



(12) 发明专利申请

(10) 申请公布号 CN 112867992 A

(43) 申请公布日 2021. 05. 28

(21) 申请号 201980069102.X

(22) 申请日 2019.10.04

(30) 优先权数据

16/153,546 2018.10.05 US

(85) PCT国际申请进入国家阶段日

2021.04.20

(86) PCT国际申请的申请数据

PCT/US2019/054784 2019.10.04

(87) PCT国际申请的公布数据

W02020/072952 EN 2020.04.09

(71) 申请人 德州仪器公司

地址 美国德克萨斯州

(72) 发明人 D·P·福莱

(74) 专利代理机构 北京律盟知识产权代理有限公司 11287

代理人 林斯凯

(51) Int.Cl.

G06F 11/08 (2006.01)

G06F 11/10 (2006.01)

H03M 13/09 (2006.01)

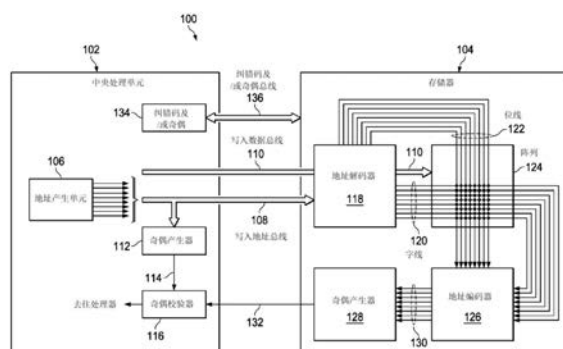
权利要求书3页 说明书11页 附图6页

(54) 发明名称

检测地址错误的系统、方法及设备

(57) 摘要

本发明公开用于主动地检测地址错误的安全机制的方法、设备、系统及制品。实例性系统包含第一奇偶产生器(112)、第二奇偶产生器(128)及奇偶校验器(116)。所述第一奇偶产生器(112)用于基于第一地址信息产生第一奇偶。所述第一地址信息对应于在存储器存储阵列(124)中存储数据的期望位置。所述第二奇偶产生器(128)用于基于第二地址信息产生第二奇偶。所述第二地址信息对应于所述存储器存储阵列(124)中存储所述数据的实际位置。所述奇偶校验器(116)用于对所述第一奇偶与所述第二奇偶进行比较以检测错误。



1. 一种系统,其包括:

第一奇偶产生器,其用于基于第一地址信息产生第一奇偶,所述第一地址信息对应于在存储器存储阵列中存储数据的期望位置;

第二奇偶产生器,其用于基于第二地址信息产生第二奇偶,所述第二地址信息对应于所述存储器存储阵列中存储数据的实际位置;及

奇偶校验器,其用于对所述第一奇偶与所述第二奇偶进行比较以检测错误。

2. 根据权利要求1所述的系统,其中所述第一奇偶包括依据所述第一地址信息中的第一组位确定的第一奇偶位,且所述第二奇偶包括依据所述第二地址信息中的第二组位确定的第二奇偶位。

3. 根据权利要求1所述的系统,其中所述错误处于地址产生单元与所述存储器存储阵列之间。

4. 根据权利要求3所述的系统,其中所述第一地址信息对应于处理器中的数据操作数,且所述第二地址信息对应于一组字线启用值及一组位线启用值。

5. 根据权利要求4所述的系统,其中所述处理器中的所述数据操作数被输入到所述地址产生单元中。

6. 根据权利要求4所述的系统,其中所述一组字线启用值及所述一组位线启用值是由地址解码器确定。

7. 根据权利要求4所述的系统,其中地址编码器对所述一组字线启用值及所述一组位线启用值进行编码。

8. 根据权利要求7所述的系统,其中所述地址编码器是通向所述第二奇偶产生器的输入。

9. 根据权利要求1所述的系统,其中所述奇偶校验器确定所述第二地址信息是否与所述第一地址信息匹配。

10. 根据权利要求1所述的系统,其中所述系统能够至少结合数据奇偶、数据与地址奇偶、数据ECC或数据与地址ECC来使用。

11. 根据权利要求1所述的系统,其中所述系统能够在应用的运行时间期间使用。

12. 根据权利要求11所述的系统,其中所述系统能够至少在所述应用的启动或关闭期间检测所述错误。

13. 一种方法,其包括:

基于第一地址信息产生第一奇偶,所述第一地址信息对应于在存储器存储阵列中存储数据的期望位置;

基于第二地址信息产生第二奇偶,所述第二地址信息对应于所述存储器存储阵列中存储数据的实际位置;

对所述第一奇偶与所述第二奇偶进行比较;及

基于对所述第一奇偶与所述第二奇偶之间的比较,检测错误。

14. 根据权利要求13所述的方法,其中所述第一奇偶包括依据所述第一地址信息中的第一组位确定的第一奇偶位,且所述第二奇偶包括依据所述第二地址信息中的第二组位确定的第二奇偶位。

15. 根据权利要求13所述的方法,其中所述错误处于地址产生单元与所述存储器存储

阵列之间。

16. 根据权利要求13所述的方法,其中所述第一地址信息对应于处理器中的数据操作数,且所述第二地址信息对应于一组字线启用值及一组位线启用值。

17. 根据权利要求16所述的方法,其中将所述一组字线启用值及所述一组位线启用值输入到地址编码器中。

18. 根据权利要求13所述的方法,其中所述方法能够至少结合数据奇偶、数据与地址奇偶、数据ECC或数据与地址ECC来使用。

19. 根据权利要求13所述的方法,其中所述方法能够在应用的运行时间期间使用。

20. 根据权利要求19所述的方法,其中所述方法能够在所述应用的启动或关闭中的至少一者期间检测所述错误。

21. 一种包括指令的有形计算机可读存储媒体,所述指令在被执行时使得处理器至少:
基于第一地址信息产生第一奇偶,所述第一地址信息对应于在存储器存储阵列中存储数据的期望位置;

基于第二地址信息产生第二奇偶,所述第二地址信息对应于所述存储器存储阵列中存储数据的实际位置;

对所述第一奇偶与所述第二奇偶进行比较;及

基于对所述第一奇偶与所述第二奇偶之间的比较,检测错误。

22. 根据权利要求21所述的有形计算机可读存储媒体,其中所述第一奇偶包括依据所述第一地址信息中的第一组位确定的第一奇偶位,且所述第二奇偶包括依据所述第二地址信息中的第二组位确定的第二奇偶位。

23. 根据权利要求21所述的有形计算机可读存储媒体,其中所述错误处于地址产生单元与所述存储器存储阵列之间。

24. 根据权利要求21所述的有形计算机可读存储媒体,其中所述第一地址信息对应于所述处理器中的数据操作数,且所述第二地址信息对应于一组字线启用值及一组位线启用值。

25. 根据权利要求24所述的有形计算机可读存储媒体,其中所述指令使得所述处理器将所述第一地址信息解码成所述一组字线启用值及所述一组位线启用值。

26. 根据权利要求24所述的有形计算机可读存储媒体,其中所述指令使得所述处理器将所述一组字线启用值及所述一组位线启用值输入到地址编码器中。

27. 根据权利要求24所述的有形计算机可读存储媒体,其中所述指令使得所述处理器至少在应用的启动或关闭期间检测所述错误。

28. 根据权利要求27所述的有形计算机可读存储媒体,其中所述指令使得所述处理器在所述应用的运行时间期间检测所述错误。

29. 根据权利要求21所述的有形计算机可读存储媒体,其中所述指令使得至少结合数据奇偶、数据与地址奇偶、数据ECC或数据与地址ECC来使用所述处理器。

30. 一种设备,其包括:

第一奇偶产生器,其用于基于第一地址信息产生第一奇偶,所述第一地址信息对应于在存储器存储阵列中存储数据的期望位置;及

奇偶校验器,其用于:

从所述第一奇偶产生器接收所述第一奇偶且接收基于第二地址信息的第二奇偶,所述第二地址信息对应于所述存储器存储阵列中存储数据的实际位置;

对所述第一奇偶与所述第二奇偶进行比较;及

基于对所述第一奇偶与所述第二奇偶之间的比较,检测错误。

31. 根据权利要求30所述的设备,其中所述第一地址信息是基于处理器中的数据操作数,且所述第二地址信息是基于的一组字线启用值及一组位线启用值。

32. 根据权利要求30所述的设备,其中所述第一奇偶包括依据所述第一地址信息中的第一组位确定的第一奇偶位,且所述第二奇偶包括依据所述第二地址信息中的第二组位确定的第二奇偶位。

33. 根据权利要求30所述的设备,其中所述错误处于所述第一地址信息的产生与所述存储器存储阵列之间。

检测地址错误的系统、方法及设备

技术领域

[0001] 本公开大体来说涉及处理系统错误,且更确切来说涉及检测地址错误系统、方法及设备。

背景技术

[0002] 微控制器用于进行可被写入到一组经译码指令中的自动化过程,所述微控制器可执行所述一组经译码指令以实施所期望过程。举例来说,微控制器可用于车辆引擎控制系统、生物医学装置、远程控制器、电器、电动工具等中。

[0003] 微控制器可含有至少一个中央处理单元(CPU)以及存储器及可编程输入/输出装置。所述存储器可以是随机存取存储器(RAM)、铁电随机存取存储器(FRAM)、非异或(NOR)快闪存储器、一次性可编程只读存储器(OTP ROM)等。存储器可位于CPU外部。另一选择为,存储器及CPU可以是同一装置上原有的。

[0004] 通过控制数据在处理器与存储器之间的传送来调节各种过程的能力使微控制器称为自动化系统不可或缺的一部分。

发明内容

[0005] 本文中所公开的某些实例能增强保护以免可在最初产生地址信息与将所述地址信息解码成特定的字线启用值及位线启用值期间出现错误。实例性系统包含第一奇偶产生器、第二奇偶产生器及奇偶校验器。所述第一奇偶产生器用于基于第一地址信息产生第一奇偶。所述第一地址信息对应于在存储器存储阵列中存储数据的期望位置。所述第二奇偶产生器用于基于第二地址信息产生第二奇偶。所述第二地址信息对应于存储器存储阵列中存储数据的实际位置。奇偶校验器用于对所述第一奇偶与第二奇偶进行比较以检测错误。

附图说明

[0006] 图1是图解说明实例性错误检测系统的框图。

[0007] 图2是图解说明实例性CPU奇偶产生器的框图。

[0008] 图3是图解说明实例性存储器奇偶产生器的框图。

[0009] 图4是图解说明实例性奇偶校验器的框图。

[0010] 图5是表示可经执行以实施图1的错误检测系统的机器可读指令的流程图。

[0011] 图6是表示可经执行以实施图2的CPU奇偶产生器的机器可读指令的流程图。

[0012] 图7是表示可经执行以实施图3的存储器奇偶产生器的机器可读指令的流程图。

[0013] 图8是表示可经执行以实施图4的奇偶校验器的机器可读指令的流程图。

[0014] 图9是经构建以执行图5、6、7及8的指令的实例性处理平台的框图。

[0015] 所述图未按比例。一般来说,在图式及随附书面说明通篇,将使用相同的参考编号来指代相同或相似的零件。尽管图展示具有清晰的线及边界的层及区,但这些线及/或边界中的一些或所有可能是理想化的。事实上,边界及/或线可能是难以看到的、混杂的及/或不

规则的。

具体实施方式

[0016] 微控制器在过程自动化中扮演着重要的角色。所述过程可能是冗余的或对应用的成功至关重要；无论如何，微控制器的适当功能性均是最重要的。微控制器中出现的典型错误可能是由存储器存储阵列或支持所述存储器存储阵列的外围装置中的电干扰或磁干扰引起。这些干扰可导致存储器中的一或多个位或逻辑改变状态，因此导致出现错误。这些错误的典型解决方案是纠错码 (ECC) 及奇偶逻辑及/或软件。这些解决方案在数据被写入到正确地址的情况下才有效且通常伴随着在写入之后读回所写入的数据。然而，即使CPU所呈现的地址信息包含到ECC及/或奇偶逻辑产生算法中，但如果地址路径中存在永久性错误，那么无法保证检测方案将在没有明显延迟的情况下能够感测到错误并警告处理器采用错误纠正程序。举例来说，ECC及奇偶逻辑及/或软件可以是数据奇偶、数据与地址奇偶、数据ECC或数据与地址ECC。

[0017] 检测永久性地址错误的传统方法涉及在启动期间对存储器进行测试以确保地址位置是正确的。然而，此方法仅能够检测自最后的启动程序开始出现的错误。此外，启动程序可能不经常执行，且因此永久性错误可在长的时间段内不被检测到。

[0018] 举例来说，CPU可旨在将一些数据写入到第一地址，但错误存在于地址路径或逻辑中。此错误将导致数据被写入到第二地址。如果错误是暂时性错误，那么上文所提及的传统先写入再读取方法将读取存储在第二地址处的数据且辨识第二地址处所存储的数据并非在写入过程传输的相同的数据。然而，如果错误是在CPU与ECC及/或奇偶逻辑及/或软件之间出现的永久性错误，那么传统方法将感知到其正在读取存储在第二地址处的数据，但其正在读取存储在第二地址处的数据。在永久性错误出现在CPU与ECC及奇偶逻辑及/或软件之间的情形中，第二地址处的数据将被旨在写入第一地址的数据盖写。CPU将无法确定在第二地址处的数据已被盖写且此可导致应用的不安全状况。

[0019] 永久性错误的另一情形是错误出现在ECC及/或奇偶逻辑及/或软件与存储器之间。在此情形中，传统方法将感知到旨在写入第一地址的数据被存储在第二地址处，但所述数据被存储在了第二地址处。当CPU读取第二地址时，CPU将注意到存储在第二地址处的数据不是所述地址的正确数据。虽然传统方法最终将捕获到此类型的永久性错误，但检测将延迟且使应用面临不安全状况。

[0020] 为解决出现在CPU与存储有数据的存储器单元之间的永久性错误，可在处理器中以及在存储器中包含奇偶产生器以从在处理器中所使用的地址信息以及存储器中存储数据的地址信息产生至少两个奇偶。然后，处理器可使用奇偶校验器来对两个或多于两个奇偶值进行比较并确定在地址信息产生与存储器中存储数据的地址信息之间是否已出现永久性错误。针对在处理器中产生的地址信息产生第一奇偶并对所述奇偶与从用于将数据存储在存储器中的地址信息产生的第二奇偶进行比较的所提出方法可与ECC及奇偶逻辑及/或软件的传统方法结合使用。

[0021] 图1是实例性错误检测系统100的框图，可根据本公开的教示实施实例性错误检测系统100以对实例性存储器存储阵列124实行错误检测。错误检测系统100可包含实例性处理器102及实例性存储器104。在所图解说明的实例中，处理器102是实例性中央处理单元

(CPU)且存储器104例如是RAM。另一选择为,存储器104可以是任何其他类型的易失性或非易失性存储器或逻辑寄存器。

[0022] 在图1所图解说明的实例中,CPU 102可包含实例性地址产生单元106。

[0023] 地址产生单元106耦合到实例性写入地址总线108。写入地址总线108传递识别期望位置的信息(被称为地址信息)以将数据写入到存储器104。通过实例性写入数据总线110将数据写入到存储器104。写入地址总线108也耦合到实例性CPU奇偶产生器112。CPU奇偶产生器112通过CPU奇偶产生器输出114耦合到实例性奇偶校验器116。然后,奇偶校验器116耦合到在奇偶校验器116外部的CPU 102的一部分。CPU 102还包含ECC及/或奇偶逻辑及/或软件134,ECC及/或奇偶逻辑及/或软件134通过实例性ECC与奇偶总线136耦合到存储器104。上文所列示的不同组件之间的耦合可根据应用而是有线的或无线的。

[0024] 在图1所图解说明的实例中,CPU 102包含地址产生单元106。在一些实例中,地址产生单元106用于对CPU中的地址数据操作数实行必要计算并产生地址。地址产生单元106包含四组寄存器三元组以及全加法器、偏移加法器及模数加法器。寄存器三元组每一包含地址寄存器、偏移寄存器及修饰符寄存器。加法器与寄存器结合地发挥功能以对存储在寄存器中的信息实行运算,从而产生地址信息。地址产生单元106可以各种数据类型来表示信息,包含散列值、二进制值、十六进制值等。

[0025] 在所图解说明的实例中,CPU 102包含CPU奇偶产生器112,所述CPU奇偶产生器112依据某些输入数据类型确定至少一个奇偶位。如下文所阐释,奇偶位用于校验产生所述奇偶位所依据的数据中的一位差错。CPU 102还包含对来自两个不同的奇偶产生器的至少两个奇偶进行比较的奇偶校验器116。在对奇偶位进行比较之后,奇偶校验器116与CPU 102的不同部分进行通信且传达奇偶校验的结果。

[0026] 在图1所图解说明的实例中,可以各种方式实施写入地址总线108及写入数据总线110。写入地址总线108及写入数据总线110将地址信息及数据传输到存储器存储阵列124中的期望位置。地址总线及数据总线的实例性实施方案可包含针对总线上的每一数据或地址信息值使用一条导线、使用多路复用器作为减小导线配置的复杂性的方法等。

[0027] 在所图解说明的实例中,存储器104包含地址解码器118、存储器存储阵列124、地址编码器126及存储器奇偶产生器128。地址解码器118通过字线120及位线122耦合到写入地址总线108且耦合到存储器存储阵列124。存储器存储阵列124通过字线120及位线122耦合到地址编码器126。地址编码器126还耦合到存储器奇偶产生器128。存储器奇偶产生器128耦合到CPU 102中的奇偶校验器116。

[0028] 地址解码器118可包含硬件或软件元件以将地址信息的数据类型转换成存储器存储阵列124所需的输入数据类型。此转换可通过使用物理逻辑门或通过使用呈软件形式的布尔代数来进行。作为硬件实施方案,地址解码器118可包含一定数目个解多路复用器,所述一定数目个解多路复用器将写入地址总线108上的信号转换成更大数目个信号,然后使用所述更大数目个信号为字线120及位线122供能。

[0029] 被供能的120及位线122形成一组字线启用值及一组位线启用值,所述一组字线启用值及一组位线启用值用于规定存储器存储阵列124中用于存储写入数据总线110上的信息的实际位置。

[0030] 在图1所图解说明的实例中,存储器存储阵列124可以是存储器104中所使用的可

编程逻辑阵列、现场可编程栅极阵列等。存储器104可被实施为动态随机存取存储器(DRAM)、静态随机存取存储器(SRAM)、硬盘驱动器等。在所图解说明的实例中,字线120及位线122用作行-列对的指定器以在存储器存储阵列124中选择用于存储数据的特定地址信息。存储器存储阵列124可用于是DRAM的实例性存储器104中。如果实例性存储器存储阵列124是DRAM,那么其可包含数个开关及电容器。所述开关可以是晶体管,例如金属氧化物半导体场效晶体管(MOSFET)、双极结晶体管(BJT)、结栅极场效晶体管(JFET)、异质结双极晶体管(HBT)等。晶体管及电容器一起用于形成对应于单个数据位的存储器单元。字线120及位线122用于规定存储有数据的个别存储器单元的位置。实例性存储器存储阵列124中的电容器可随时间推移而经历泄漏,此将损坏其中存储的数据且在读取事件之后可能需要再充电来维持所存储的数据。此通过为字线120及位线122再次供能以将存储器存储阵列124中的电容器再充电来进行。此重复的再次供能可导致字线120及位线122发生应变且最终导致字线120及/或位线122上出现错误。

[0031] 在图1所图解说明的实例中,存储器104包含地址编码器126,地址编码器126将对字线120及位线122上的信息分别进行编码。被启用的特定一组字线120及位线122成为字线启用值及位线启用值,然后地址编码器126将所述字线启用值及位线启用值编码到地址信息中。举例来说,关于字线120及位线122的字线启用值及位线启用值输入(例如值输入)到多路复用器中以减小经由地址校验总线130发送到存储器奇偶产生器128的地址信息的大小。在将重新编码的地址信息发送到存储器奇偶产生器128之后,将存储器奇偶产生器128的输出发送奇偶校验器116。存储器奇偶产生器128耦合到奇偶校验器116,以使得可对来自存储器奇偶产生器128的输出与从CPU 102中的CPU奇偶产生器112产生的奇偶进行比较。换句话说,如果没有错误,那么来自地址编码器126的输出将与来自地址产生单元106的输出相匹配。在CPU 102中通过CPU奇偶产生器112产生奇偶、在存储器104中通过存储器奇偶产生器128产生奇偶及通过奇偶校验器116对两个奇偶进行比较将允许对在CPU 102与存储有数据的位置之间出现的错误进行适当错误检测。

[0032] 在图1所图解说明的实例中,错误检测系统100主动地检测地址线上的错误。在存储器104中存储数据的所期望地址信息首先是通过地址产生单元106计算。此地址信息对应于存储写入数据总线110上的数据的期望位置。然后,将地址信息发送到写入地址总线108且发送到存储器104以及CPU奇偶产生器112。CPU奇偶产生器112从写入地址总线108上的已编码地址信息产生一或多个奇偶位。将所述一或多个奇偶位发送到奇偶校验器116。

[0033] 在存储器104中,地址解码器118从写入地址总线108接收地址信息。地址解码器118使用硬件或软件对地址信息进行解码。所解码的地址信息产生字线120及位线122的字线启用值及位线启用值,以确定要存储写入数据总线110上的数据的实际地址信息。然后,地址编码器126使用硬件或软件对字线启用值及位线启用值进行编码。将此地址信息经由地址校验总线130发送到存储器奇偶产生器128。存储器奇偶产生器128从其所接收到的地址信息产生一或多个奇偶位。

[0034] 将所述一或多个奇偶位经由存储器奇偶产生器输出132发送到CPU 102中的奇偶校验器116。对由存储器奇偶产生器128产生的所述一或多个奇偶位与由CPU奇偶产生器112产生的所述一或多个奇偶位进行比较。然后,将结果发送到CPU 102中在奇偶校验器116外部的一部分。奇偶校验器116中的比较结果可能是数字高值或数字低值。举例来说,如果结

果是数字高值且CPU 102被设计成使数字高值对应于布尔真值,那么将通知CPU102已出现错误。另一选择为,如果CPU 102被设计成使数字高值对应于布尔假值,那么将通知CPU 102未出现错误。如果未出现错误,那么CPU 102将继续进行校验直到发现错误。然而,如果已出现错误,那么CPU 102将制定软件减轻差错对应用的影响。

[0035] 图2是图1的实例性CPU奇偶产生器112的框图。CPU奇偶产生器112包含实例性接口202及实例性逻辑确定器204。在实例中,接口202耦合到写入地址总线108及逻辑确定器204。逻辑确定器204可通过CPU奇偶产生器输出114耦合到奇偶校验器116。根据应用,上文所列示的不同的组件之间的耦合可以是有线的或无线的。

[0036] 在图2所图说明的实例中,接口202可以硬件形式实施为写入地址总线108与CPU奇偶产生器112之间的连接。接口202可将写入地址总线108上的地址信息分离成8个信息位的片段。另一选择为,接口202可被实施为实现相同功能的软件。接口202可连接到各种数据类型,包含二进制、十六进制、八进制等。接口202可将数据类型中的一者转换成逻辑确定器204可作为输入接受的任何数据类型。

[0037] 在所图说明的实例中,逻辑确定器204可被实施为将输入值减小成至少一个奇偶位的一组个别异或(XOR)门或软件布尔方程,或者被实施为将把输入数据减小到一组奇偶位的另一类型的软件。当被实施为XOR门时,逻辑确定器204可将8个位的群组散列在一起以为每一群组形成单个奇偶位。

[0038] 图3是图1的实例性存储器奇偶产生器128的框图。存储器奇偶产生器128可包含实例性接口302及实例性逻辑确定器304。接口302可耦合到地址校验总线130。逻辑确定器304通过存储器奇偶产生器输出132进一步耦合到奇偶校验器116。根据应用,上文所列示的不同组件之间的耦合可以是有线的或无线的。

[0039] 在图3所图说明的实例中,接口302可以硬件形式实施为写入地址总线130与存储器奇偶产生器128之间的连接。接口302可将写入地址总线130上的地址信息分离成8个信息位的片段。另一选择为,接口302可被实施为实现相同功能的软件。接口302可连接到各种数据类型,包含二进制、十六进制、八进制等。接口302可将数据类型中的一者转换成逻辑确定器304可作为输入接受的任何数据类型。

[0040] 在所图说明的实例中,逻辑确定器304可被实施为将输入值减小成至少一个奇偶位的一组个别异或(XOR)门或软件布尔方程,或被实施为将把输入数据减小到一组奇偶位的另一类型的软件。当被实施为XOR门时,逻辑确定器304可将8个位的群组散列在一起以为每一群组形成单个奇偶位。

[0041] 图4是图1的实例性奇偶校验器116的框图。奇偶校验器116包含实例性接口402、实例性比较器404及实例性错误产生器406。接口402耦合到CPU 102中在奇偶校验器116外部的一部分、CPU奇偶产生器输出114、存储器奇偶产生器输出132等。根据应用,上文所列示的不同组件之间的耦合可以是有线的或无线的。

[0042] 在图4所图说明的实例中,接口402耦合到CPU奇偶产生器输出114、存储器奇偶产生器输出132、CPU 102中在奇偶校验器116外部的一部分及比较器404。接口402将CPU奇偶产生器输出114及存储器奇偶产生器输出132视作输入且将与两个奇偶产生器对应的奇偶值发送到比较器404。在其它实施方案中,接口402可从CPU奇偶产生器112及存储器奇偶产生器128接收地址信息以及奇偶位。在先前提及的实施方案中,接口402可针对从CPU奇偶

产生器112接收的输入及从存储器奇偶产生器128接收的输入而从地址信息分离出奇偶位。根据应用,上文所列示的不同组件之间的耦合可以是有线的或无线的。

[0043] 在图4所图解说明的实例中,比较器404耦合到接口402及错误产生器406。比较器404可被实施为硬件及/或软件。在硬件实施方案中,XOR逻辑门可用于对两个或多于两个奇偶位进行比较且产生结果并将所述信息发送到错误产生器406。在比较器404的软件实施方案中,可使用一组布尔代数方程来对至少两个奇偶位一起进行比较且然后将所述结果发送到错误产生器406。根据应用,上文所列示的不同组件之间的耦合可以是有线的或无线的。

[0044] 在所图解说明的实例中,错误产生器406耦合到比较器404及接口402。错误产生器406可被实施为硬件或软件。作为硬件实施方案,错误产生器406可以是用于产生高电平或低电平信号以供发送到接口402的一组逻辑门。当被实施为软件时,错误产生器406可以是经受比较器404的输出的else/if语句。将来自错误产生器406的输出发送到接口402以使得接口402可传达到CPU 102中在奇偶校验器116外部的一部分。根据应用,上文所列示的不同组件之间的耦合可以是有线的或无线的。

[0045] CPU奇偶产生器112、存储器奇偶产生器128及奇偶校验器116允许在应用的运行时间(例如,在运行时间期间使用)期间在允许进行错误检测的同时最大程度地涵盖字线错误及位线错误,而由于先前解决方案的破坏性,先前解决方案仅可在启动或关闭应用期间实施。由于必须从功能使用中停用存储器以使先前解决方案不中断地实施全面测试,因此先前解决方案是破坏性的。由于正在接受测试的存储器在很长的时间量内不可用,因此此举明显地影响系统性能。反之,当使用CPU奇偶产生器112、存储器奇偶产生器128及奇偶校验器116时,在监测错误时存储器104完全能发挥功能且不会像先前解决方案一样使系统性能受到削弱。CPU奇偶产生器112、存储器奇偶产生器128及奇偶校验器116将地址信息的自适应实时反馈提供给CPU 102。CPU奇偶产生器112、存储器奇偶产生器128及奇偶校验器116通过将CPU 102中的地址信息产生与处理系统的存储器104中的地址信息使用之间的环路闭合来提高使用计算装置的效率。此允许增强保护以免可能在CPU 102中最初产生地址信息与在存储器104中将所述地址信息解码成特定字线启用值及位线启用值之间出现错误。

[0046] 虽然图2、3及4中图解说明实施图1的错误检测系统100的实例性方式,但可组合、划分、重新布置、省略、去除及/或以任何其它方式实施图2、3及4中所图解说明的元件、过程及/或装置中的一或多个。此外,可通过硬件、软件、固件及/或硬件、软件及/或固件的任何组合来实施图1的实例性CPU奇偶产生器112、实例性存储器奇偶产生器128、实例性奇偶校验器116及/或更笼统来说实例性错误检测系统100。因此,举例来说,实例性CPU奇偶产生器112、实例性存储器奇偶产生器128、实例性奇偶校验器116及/或更笼统来说实例性错误检测系统100中的任一者可通过一或多个模拟或数字电路、逻辑电路、可编程处理器、可编程控制器、图形处理单元(GPU)、数字信号处理器(DSP)、特殊应用集成电路(ASIC)、可编程逻辑装置(PLD)及/或现场可编程逻辑装置(FPLD)来实施。当阅读本专利的设备或系统技术方案中的任一者以涵盖纯软件及/或固件实施方案时,实例性CPU奇偶产生器112、实例性存储器奇偶产生器128、实例性奇偶校验器116及/或实例性错误检测系统100中的至少一者特此明确地定义为包含非暂时性计算机可读存储装置或存储盘(例如存储器、数字多功能盘(DVD)、光盘(CD)、蓝光盘等),其包含软件及/或固件。此外,除图2、3及4中所图解说明的元件、过程及/或装置之外或者替代图2、3及4中所图解说明的元件、过程及/或装置,图1中的

实例性错误检测系统100可还包含一或多个元件、过程及/或装置,及/或可包含所图解说明的元件、过程及装置中的任一者或全部中的一者以上。如本文中所使用,短语“在通信中”(包含其变化形式在内)囊括直接通信及/或通过一或多个中间组件的间接通信,且不要求直接物理(例如,有线)通信及/或连续通信,而是另外包含以周期性间隔、排定间隔、周期性间隔及/或一次性事件进行的选择性通信。

[0047] 图5中展示表示用于实施图1的错误检测系统的实例性硬件逻辑、机器可读指令、硬件实施状态机及/或其任何组合的流程图。机器可读指令可以是由计算机处理器(例如,下文结合图9所论述的实例性处理器平台900中所展示的处理器912)执行的可执行程序或可执行程序的一部分。程序可体现为存储在非暂时性计算机可读存储媒体(例如,与处理器912相关联的CD-ROM、软盘、硬盘驱动器、DVD、蓝光盘或存储器)上的软件,但另一选择为,整个程序及/或其部分可由除处理器912之外的装置执行及/或体现为固件或专用硬件。此外,尽管参考图5中所图解说明的流程图描述实例性程序,但可替代地使用实施实例性错误检测系统100的许多其它方法。举例来说,可改变方框的执行次序及/或可改变、去除或组合所描述的方框中的一些方框。另外或另一选择为,方框中的任一者或全部可由经构建以实行对应操作而不执行软件或固件的一或多个硬件电路(例如,离散及/或集成模拟及/或数字电路系统、FPGA、ASIC、比较器、运算放大器(op-amp)、逻辑电路等)实施。

[0048] 如上文所提及,图1、2、3及4的实例性过程可使用可执行指令(例如,计算机及/或机器可读指令)来实施,所述可执行指令存储在非暂时性计算机及/或机器可读媒体上,例如硬盘驱动器、快闪存储器、只读存储器、光盘、数字多功能盘、高速缓冲存储器、随机存取存储器及/或任何其它存储装置或存储盘,信息在所述非暂时性计算机及/或机器可读媒体中存储达任何持续时间,例如延长的时间段、永久性、短暂时刻、暂时缓冲及/或用于高速缓存信息。如本文中所使用,术语非暂时性计算机可读媒体明确地定义为包含任何类型的计算机可读存储装置及/或存储盘,并且不包括传播信号且不包括传输媒体。

[0049] “包含”及“包括”(及其所有形式及时态)在本文中用作开放性术语。因此,每当技术方案采用任何形式的包含或包括(例如,comprises、includes、comprising、including、having等)作为前文或者在任何类型的技术方案引述内使用任何形式的包含或包括时,应理解可存在额外元件、物项等,且此仍处于对应技术方案或引述的范围内。如本文中所使用,当例如在技术方案的前文中使用短语“至少”作为过渡术语时,其是开放式的,与开放式术语“包括”及“包含”相同。如本文中所使用,当例如在技术方案的前文中使用短语“至少”作为过渡术语时,其是开放式的,与开放式术语“包括”及“包含”相同。当使用例如以例如A、B及/或C形式使用术语“及/或”时,术语“及/或”指代A、B、C的任何组合或子组,例如(1)只有A、(2)只有B、(3)只有C、(4)A与B、(5)A与C、(6)B与C及(7)A与B及C。如本文中对描述结构、组件、物项、物体及/或事物的上下文中所使用,短语“A及B中的至少一者”意在指代包含(1)至少一个A、(2)至少一个B以及(3)A中的至少一个及B中的至少一者中的任一者的实施方案。类似地,如本文中对描述结构、组件、物项、物体及/或事物的上下文中所使用,短语“A或B中的至少一者”意在指代包含(1)至少一个A、(2)至少一个B及(3)至少一个A及至少一个B中的任一者的实施方案。如本文中对描述过程、指令、动作、活动及/或步骤的实行或执行的上下文中所使用,短语“A及B中的至少一者”意在指代包含(1)至少A、(2)至少B及(3)至少A及至少B中的任一者的实施方案。类似地,如本文中对描述过程、指令、动作、活动及/或步骤的实

行或执行的上下文中所使用,短语“A或B中的至少一者”意在指代包含(1)至少A、(2)至少B以及(3)至少A及至少B中的任一者的实施方案。

[0050] 图5的程序展示实施图1的错误检测系统100的过程500。过程500包含作为过程500的开始的方框502。方框502是可指示处理器产生第一奇偶的方框。过程500可对应于图2的实例性CPU奇偶产生器112。在方框502处,CPU奇偶产生器112用于产生与第一地址信息相关联的第一奇偶位。第一地址信息与图1的CPU 102中的数据操作数(例如,基于数据操作数)相关联。通过写入地址总线108将第一地址信息发送到存储器104。第一地址信息可以是存储器104中存储写入数据总线110上的数据的期望位置。

[0051] 过程500包含方框504,其是指示处理器将第一地址信息解码成字线启用值及位线启用值的方框。这些字线启用值及位线启用值对应于图1的存储器104的字线120及位线122。方框504可对应于图1的实例性硬件实施方案,其中方框504可以如地址解码器118等硬件形式来施行。

[0052] 过程500包含方框506。方框506是指示处理器将数据存储存储在存储器存储阵列中的方框。存储器存储阵列可对应于图1的存储器存储阵列124且数据可对应于图1的写入数据总线110上的数据。

[0053] 过程500进一步包含方框508。方框508是指示处理器将字线启用值及位线启用值重新编码成第二地址信息的方框。此第二地址信息对应于字线120及位线122上的字线启用值及位线启用值。此第二地址信息是存储器存储阵列中存储数据的实际位置。方框508可对应于图1的实例性硬件实施方案,其中方框508是以实例性地址编码器126来实施。存储数据的存储器存储阵列可对应于图1的存储器存储阵列124。

[0054] 过程500包含方框510,其是指示处理器基于第二地址信息产生第二奇偶位的方框。此奇偶位对应于字线120及位线122上的字线启用值及位线启用值。方框510可对应于图1的实例性存储器104的实例性存储器奇偶产生器128。

[0055] 过程500还包含方框512,其是指示处理器对对应于第一地址信息的第一奇偶位与对应于第二地址信息的第二奇偶位进行比较的方框。方框512可对应于图1的实例性奇偶校验器116。

[0056] 图6的过程展示子过程600,子过程600可用于实施图5中所展示的过程500的方框502。在图1中的实例性硬件实施方案中通过图2中详细地展示的CPU奇偶产生器112展示子过程600。子过程600指示处理器产生基于CPU 102中的第一地址信息的奇偶位。第一地址信息对应于CPU 102中由图1中所展示的地址产生单元106使用的数据操作数。

[0057] 子过程600开始于方框602处,方框602规定在将数据存储存储在存储器存储阵列之前从写入地址总线接收第一地址信息。存储器存储阵列可对应于图1的实例性错误检测系统100中的存储器104的存储器存储阵列124。第一地址信息可位于写入地址总线108上。方框602可通过图2中所展示的实例性CPU奇偶产生器112的接口202施行。

[0058] 子过程600中的下一方框是方框604,方框604是指示实例性CPU奇偶产生器112的接口202将地址信息发送到图2的逻辑确定器204的方框。方框604可表示图2的接口202与逻辑确定器204之间的耦合。

[0059] 子过程600中接下来是方框606,方框606是确定发送到方框606的地址信息的一或多个奇偶位的方框。方框606可以如图2的CPU奇偶产生器112中的逻辑确定204等硬件施行。

所确定的奇偶位的数目基于地址信息的大小而定。通常针对每一组8个信息位确定奇偶位。举例来说,接口202将输入到CPU奇偶产生器112中的数据分离成8位区段以供逻辑确定器204使用。针对32位地址信息,逻辑确定器204将确定4个奇偶位。

[0060] 子过程600中接下来是方框608,方框608是将所确定奇偶位发送到奇偶校验器的方框。奇偶校验器可对应于图1的CPU 102的奇偶校验器116。方框608可以如CPU奇偶产生器输出114等硬件表示。子过程600中接下来是方框610,方框610是指示子过程600返回到图5的过程500的方框。

[0061] 图7的过程展示子过程700,子过程700可用于实施图5中所展示的过程500的方框510。在图1实例性硬件实施方案中通过图3中详细地展示的存储器奇偶产生器128展示子过程700。子过程700可指示处理器基于存储器104中的第二地址信息产生奇偶位。第二地址信息对应于与图1中所展示的存储器104中的字线120及位线122对应的字线启用值及位线启用值。

[0062] 子过程700开始于方框702处,方框702规定在将数据存储在存储器存储阵列中之后在地址校验总线上接收第二地址信息。地址校验总线可对应于图1的地址校验总线130。存储器存储阵列可对应于图1的存储器104的存储器存储阵列124。方框702可通过图3中所展示的实例性存储器奇偶产生器128的接口302来施行。

[0063] 子过程700中下一方框是方框704,方框704是指示实例性存储器奇偶产生器128的接口302将地址信息发送到图3的逻辑确定器304的方框。方框704可表示图3的接口302与逻辑确定器304之间的耦合。

[0064] 子过程700中接下来是方框706,方框706是方框针对发送到方框706的地址信息确定一或多个奇偶位。方框706可以图3的存储器奇偶产生器128中的逻辑确定器304等硬件来施行。所确定的奇偶位的数目是基于地址信息的大小。通常针对每一组个8信息位确定奇偶位。举例来说,接口302将输入到存储器奇偶产生器128中的数据分离成8位区段以供逻辑确定器304使用。针对32位地址信息,逻辑确定器304将确定个4奇偶位。

[0065] 子过程700中接下来是方框708,方框708是将所确定奇偶位发送到奇偶校验器的方框。奇偶校验器可对应于图1的CPU 102的奇偶校验器116。方框708可以如存储器奇偶产生器输出132等硬件来表示。子过程700中接下来是方框710,方框710是指示子过程700返回到图5的过程500的方框。

[0066] 图8的过程展示可用于实施图5中所展示的过程500的方框512的子过程800。在图1中的实例性硬件实施方案中通过图4中详细展示奇偶校验器116展示子过程800。子过程800可指示处理器对两个或多于两个奇偶位进行比较。

[0067] 子过程800开始于方框802处,其方框802规定处理器从第一奇偶产生器接收第一奇偶位。第一奇偶产生器可对应于图2中所展示的CPU奇偶产生器112。子过程800中下一方框是方框804,方框804规定处理器从第二奇偶产生器接收第二奇偶位。第二奇偶产生器可对应于图3中所展示的存储器奇偶产生器128。子过程800的实例性实施方案在图4中展示,其中子过程800的方框802及804可通过接口402来施行。

[0068] 子过程800中接下来是方框806,方框806是对两个或多于两个奇偶位进行比较且规定两个或多于两个奇偶位是否不同的方框。子过程800的方框806可被实施为图4的实例性奇偶校验器116中的比较器404。如果方框806对两个或多于两个奇偶位进行比较且规定

两个或多于两个奇偶位不同,然后子过程800移动到方框808上,如果否,子过程800移动到方框810上。

[0069] 在子过程800中,方框808是产生规定所述两个或多于两个奇偶位不同的错误的方框。错误标示第一地址信息不同于第二地址信息。第一地址信息对应于CPU 102中由地址产生单元106使用的数据操作数。第二地址信息对应于所述一组字线启用值及所述一组位线启用值以及存储数据的实际位置。方框808可被实施为图4的错误产生器406。

[0070] 然后,子过程800从方框808进行到方框810。方框810是将两个或多于两个奇偶位的比较结果发送到处理器中在奇偶校验器外部的一部分的方框。为实现此目的,可首先将结果发送到接口402。然后,接口402可将比较结果发送到CPU 102中在奇偶校验器116外部的一部分。依据此结果的值,CPU 102可确定是否制定安全协议来寻找错误。针对错误的安全协议是减轻错误的影响且防止应用中出现不安全状况。子过程800中下一方框是方框812,方框812是指示子过程800返回到图5的过程500的方框。

[0071] 图9是经构建以执行图5、6、7及8的指令以实施图1的系统的实例性处理器平台900的框图。处理器平台900例如可以是服务器、个人计算机、工作站、自学习机器(例如,神经网络)、移动装置(例如,手机、智能电话、平板计算机,例如iPad™)、个人数字助理(PDA)、因特网电器、DVD播放器、CD播放器、数字视频记录器、蓝光播放器、游戏机、个人视频记录器、机顶盒、耳机或其它可穿戴装置或任何其他类型的计算装置。

[0072] 所图解说明实例的处理器平台900包含处理器912。所图解说明实例的处理器912是硬件。举例来说,处理器912可通过来自任何所期望族群或制造商的一或多个集成电路、逻辑电路、微处理器、CPU、DSP或控制器来实施。硬件处理器可以是基于半导体(例如,基于硅)的装置。在此实例中,处理器实施CPU奇偶产生器112及奇偶校验器116。

[0073] 所图解说明的实例处理器912包含本地存储器913(例如,高速缓冲存储器)。所图解说明实例的处理器912经由总线918与包含易失性存储器914及非易失性存储器916的主存储器进行通信。易失性存储器914可由同步动态随机存取存储器(SDRAM)、动态随机存取存储器(DRAM)、RAMBUS®动态随机存取存储器(RDRAM®)及/或任何其它类型的随机存取存储器装置实施。非易失性存储器916可由快闪存储器及/或任何其它所期望类型的存储器装置实施。对主存储器914、916的存取由存储器控制器加以控制。在此实例中,易失性存储器914包含存储器奇偶产生器128。

[0074] 所图解说明实例的处理器平台900还包含接口电路920。接口电路920可由任何类型的接口标准(例如以太网接口、通用串行总线(USB)、Bluetooth®接口、近场通信(NFC)接口及/或PCI高速接口)实施。

[0075] 在所图解说明的实例中,一或多个输入装置922连接到接口电路920。输入装置922准许用户将数据及/或命令键入到处理器912中。举例来说,输入装置可由音频传感器、麦克风、摄像机(静物或视频)、键盘、按钮、鼠标、触摸屏、跟踪垫、轨迹球、等位点装置(isopoint)及/或声音辨识系统实施。

[0076] 一或多个输出装置924也连接到所图解说明的实例的接口电路920。举例来说,输出装置924可由显示装置(例如,发光二极管(LED)、有机发光二极管(OLED)、液晶显示器(LCD)、阴极射线管显示器(CRT)、共面切换(IPS)显示器、触摸屏等)、触觉输出装置、打印机及/或扬声器实施。因此,所图解说明实例的接口电路920通常包含图形驱动器卡、图形驱动

器芯片及/或图形驱动器处理器。

[0077] 所图解说明实例的接口电路920还包含例如发射器、接收器、收发器、调制解调器、常驻网关、无线接入点及/或网络接口的通信装置以便于经由网络926与外部机器(例如,任何类型的计算装置)交换数据。通信可经由例如以太网连接、数字订户线(DSL)连接、电话线连接、同轴缆线系统、卫星系统、直线对传式无线系统、蜂窝电话系统等。

[0078] 所图解说明实例的处理器平台900还包含用于存储软件及/或数据的一或多个大容量存储装置928。此类大容量存储装置928的实例包含软盘驱动器、硬驱动盘、光盘驱动器、蓝光盘驱动器、独立盘冗余阵列(RAID)系统及数字多功能盘(DVD)驱动器。

[0079] 图5、6及8的机器可执行指令932可存储在大容量存储装置928中、存储在易失性存储器914中、存储在非易失性存储器916中及/或存储在可拆卸式非暂时性计算机可读存储媒体(例如CD或DVD)上。

[0080] 从上文看,将了解,已公开在允许在应用的运行时间(例如,在运行时间期间使用)期间进行错误检测的同时最大程度地涵盖字线及位线错误的实例性方法、设备及制品,而由于先前解决方案的破坏性,先前解决方案仅可在启动或关闭应用期间实施。由于必须从功能使用中停用存储器以使先前解决方案不中断地实施全面测试,因此先前解决方案是破坏性的。由于正在接受测试的存储器在很长的时间量内不可用,因此此举明显地影响系统性能。反之,当使用本文中所公开的方法、设备及制品时,在监测错误时存储器完全能发挥功能且不会像先前解决方案一样使系统性能受到削弱。还将了解,已公开将地址信息的自适应实时反馈提供给CPU的实例性方法、设备及制品。所公开的方法、设备及制品通过闭合处理器中的地址信息产生与处理系统的存储器中的地址信息使用之间的环路提高使用计算装置的效率。此允许增强保护以免最初产生地址信息与将所述地址信息解码成特定的字线启用值及位线启用值期间可能出现的错误。因此,所公开的方法、设备及制品旨在计算机的功能的一或多项提高。

[0081] 本专利涵盖处于本专利的权利要求书的范围内的所有方法、设备及制品。

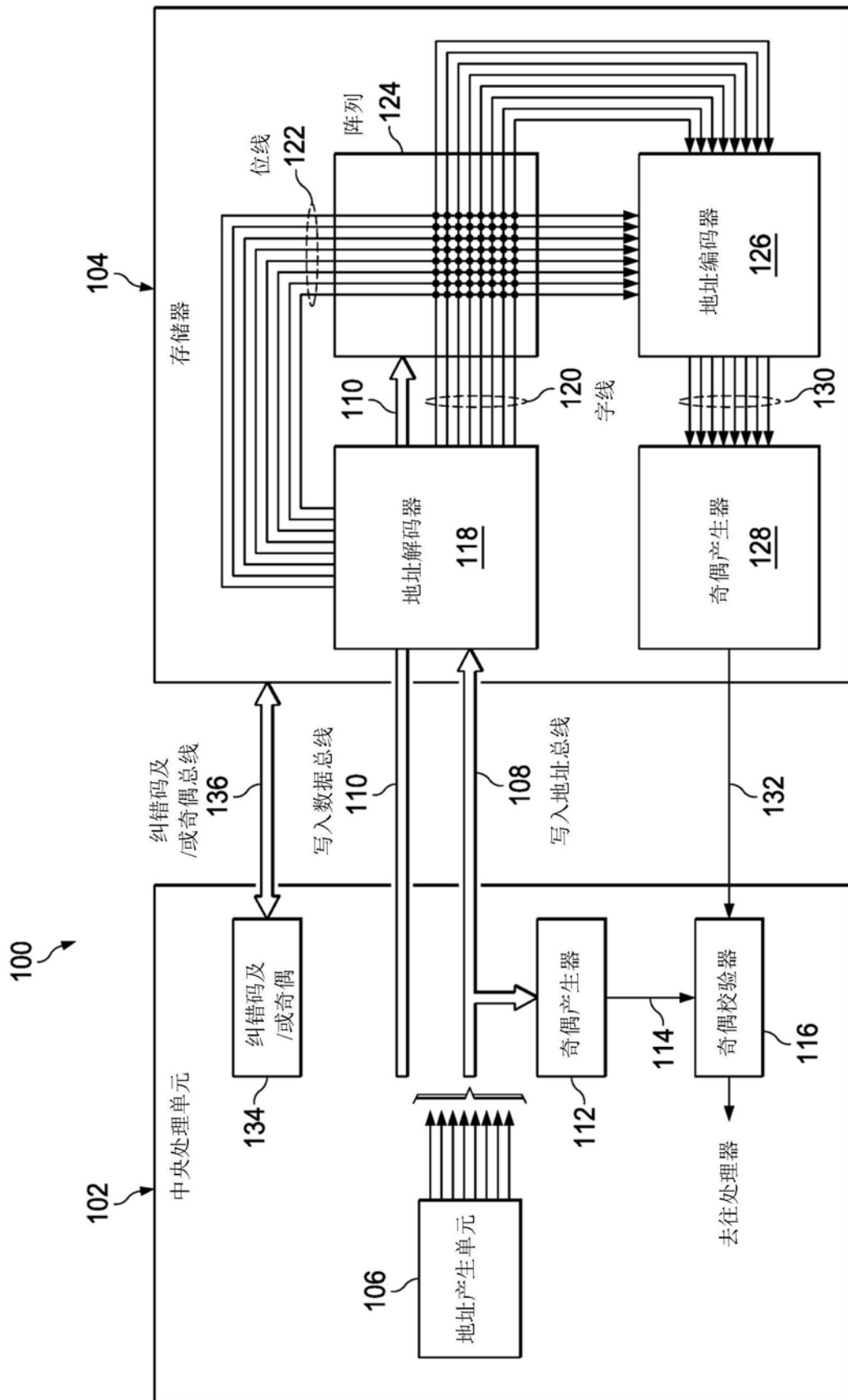


图1

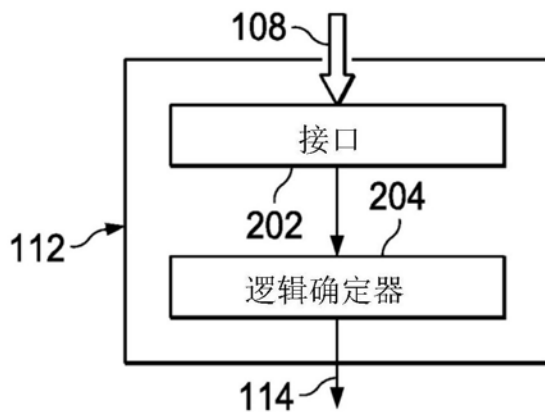


图2

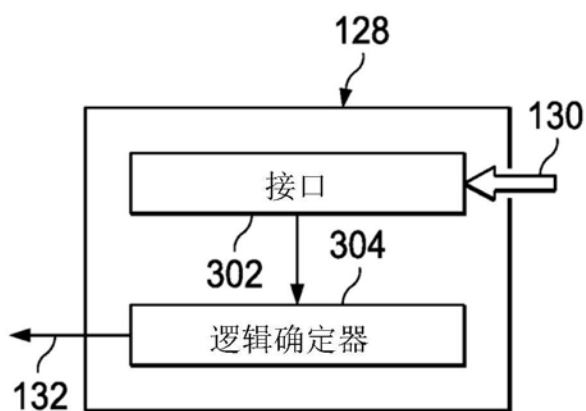


图3

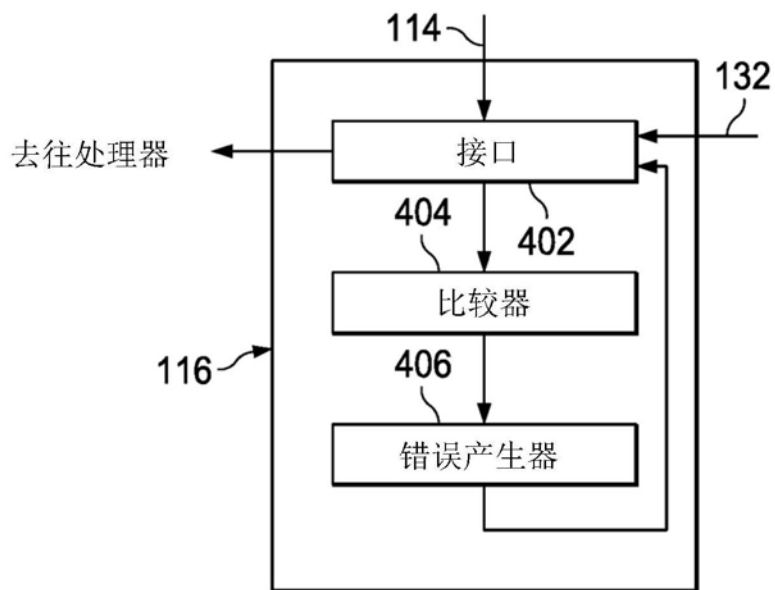


图4

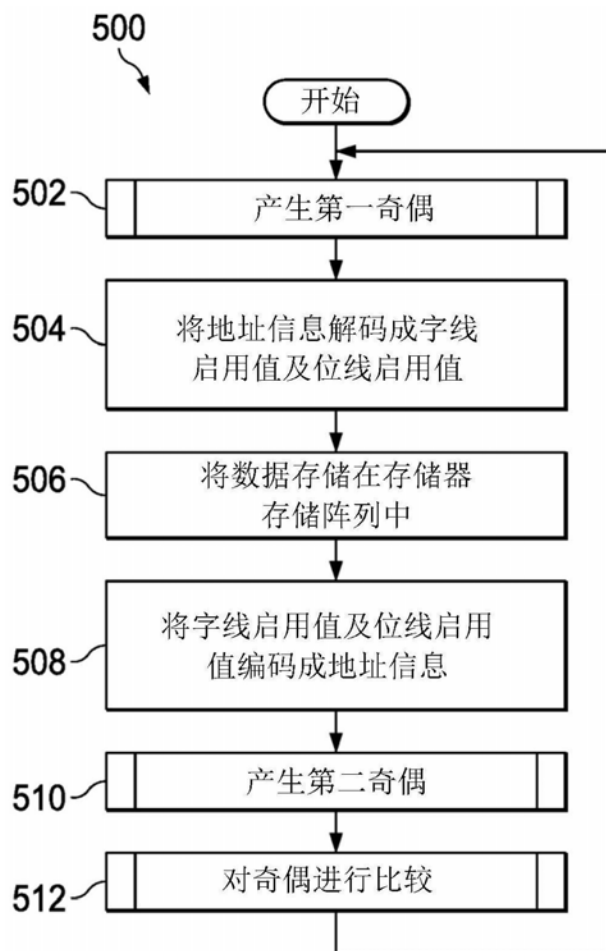


图 5

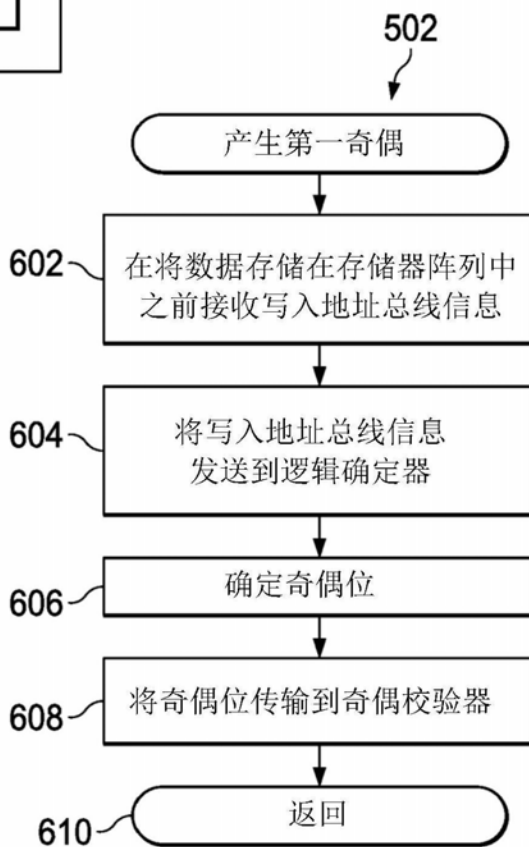


图 6

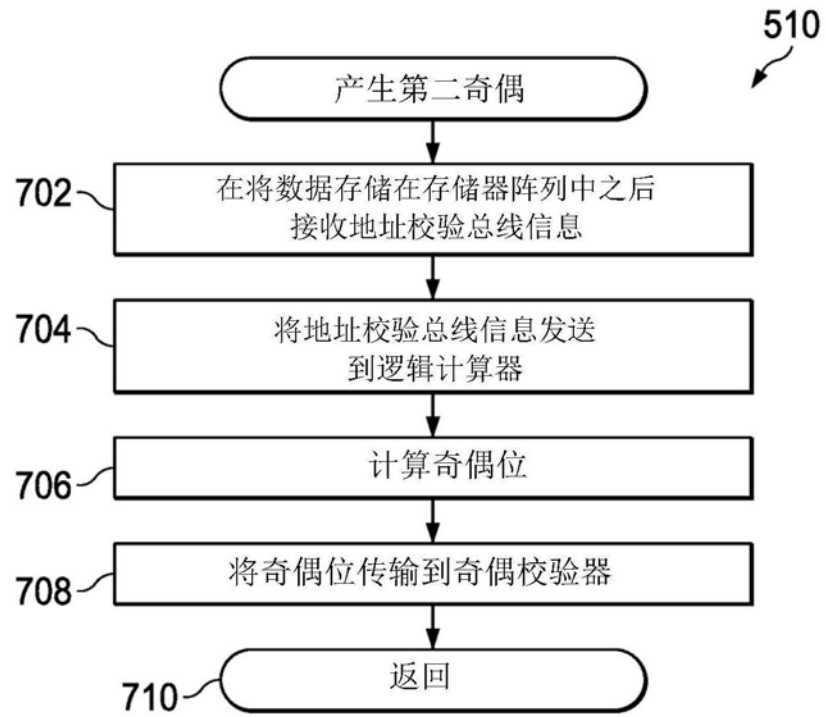


图7

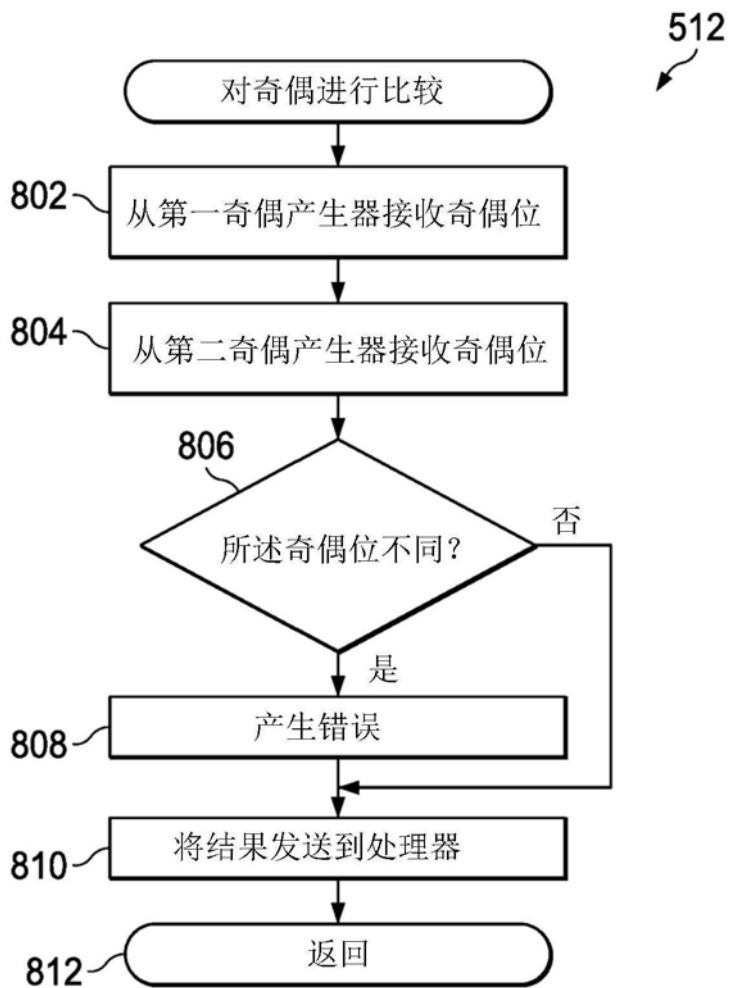


图8

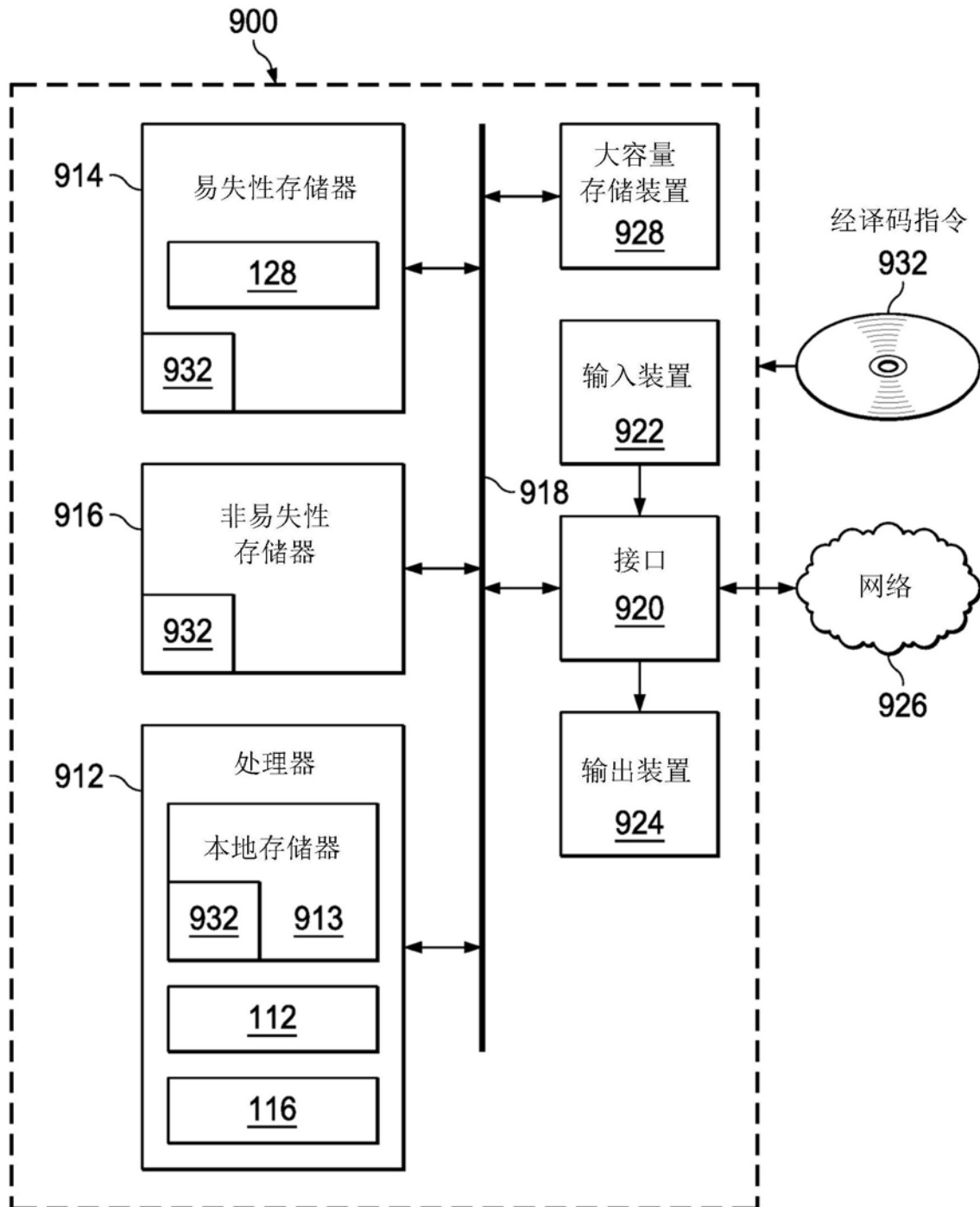


图9