

(19) 中华人民共和国国家知识产权局



(12) 发明专利申请

(10) 申请公布号 CN 104583964 A

(43) 申请公布日 2015. 04. 29

(21) 申请号 201380045212. 5

(51) Int. Cl.

(22) 申请日 2013. 06. 27

G06F 11/08(2006. 01)

(30) 优先权数据

13/629688 2012. 09. 28 US

(85) PCT国际申请进入国家阶段日

2015. 02. 27

(86) PCT国际申请的申请数据

PCT/US2013/048086 2013. 06. 27

(87) PCT国际申请的公布数据

W02014/051797 EN 2014. 04. 03

(71) 申请人 英特尔公司

地址 美国加利福尼亚州

(72) 发明人 Z. S. 郭

(74) 专利代理机构 中国专利代理(香港)有限公司

72001

代理人 张凌苗 姜甜

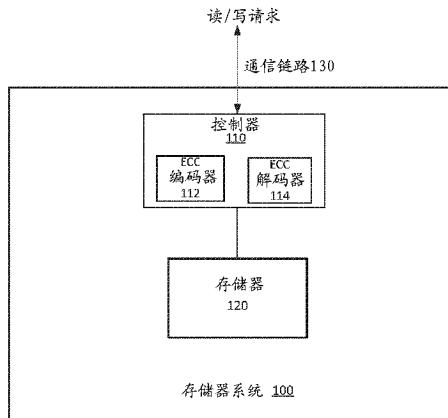
权利要求书2页 说明书14页 附图9页

(54) 发明名称

与用于编码的数据的纠错相关联的技术

(57) 摘要

公开了示例，所述示例针对于用于编码的数据的纠错相关联的技术。在某些示例中，可接收用于纠错码(ECC)编码的数据的ECC信息，其指示ECC编码的数据包括一个或多个错误。可进行关于ECC编码的数据是否包括单个错误或超过一个错误的确定。如果ECC编码的数据包括单个错误，则识别该错误的错误位置。如果ECC编码的数据包括超过一个错误，则可针对超过一个错误识别单独的错误位置。可纠正所述单个错误或超过一个错误，并且然后可将ECC编码的数据解码。描述了其他示例并要求保护。



1. 一种设备,包括:

处理器电路;

单错部件,被布置成用于由处理器电路执行以接收用于 ECC 编码的数据的纠错码(ECC)信息,其指示 ECC 编码的数据中的一个或多个错误,确定编码的数据是否包括单个错误,当 ECC 编码的数据被确定为具有单个错误时,识别 ECC 编码的数据中的错误的位置,或者生成将指示 ECC 编码的数据具有多个错误的标志;

多错部件,其被布置成用于由处理器电路执行以接收指示一个或多个错误的用于 ECC 编码的数据的 ECC 信息,并单独地识别用于每个错误的错误位置;以及

纠正器部件,其被布置成用于由处理器电路执行以接收用于单个错误的所识别位置或用于多个错误的单独识别的错误位置,并纠正 ECC 编码的数据中的一个或多个错误。

2. 权利要求 1 的设备,包括:

缓冲器部件,其被布置成用于由处理器电路执行以保持能够临时地存储 ECC 编码的数据以便由纠正器部件纠正的存储器。

3. 权利要求 1 的设备,包括 ECC 将包括 Reed-Solomon (RS) 码或二进制 Bose, Chaudhuri, Hocquenghem (BCH) 码中的一个。

4. 权利要求 3 的设备,包括多错部件,其能够实现 Berlekamp-Massey 算法(BMA)和 Chien 搜索以单独地识别错误位置。

5. 权利要求 3 的设备,包括单错部件和多错部件,被配置成基于使用 RS 码编码的 ECC 编码的数据来识别与给定识别错误位置相关联的错误值。

6. 权利要求 5 的设备,包括单错部件,其被配置成基于实现包括

$e = \frac{s_n^2}{s_{2n}}$ 和 $j = \log \frac{s_{n+1}}{s_n}$ 的算法来识别用于单个错误的错误值和给定识别错误位置,其中,e 是错误值、j 是给定识别错误位置,S 表示包括在接收到的 ECC 信息中的部分伴随式且 n 表示任何正整数。

7. 权利要求 3 的设备,包括使用二进制 BCH 码编码的 ECC 编码的数据和所述单错部件,所述单错部件被配置成基于实现包括 $j = \log s_1$ 的算法来识别 ECC 编码的数据中的错误的位置,其中,j 是给定识别错误位置且 S 表示包括在接收到的 ECC 信息中的部分伴随式。

8. 权利要求 3 的设备,包括与存储到存储器设备、存储到存储介质、无线通信或 2 维条形码读取器中的一个相关联的 ECC 编码的数据。

9. 权利要求 1 的设备,包括:

检错部件,其被布置成用于由处理器电路执行以确定 ECC 编码的数据具有一个或多个错误,并生成将指示 ECC 编码的数据中的一个或多个错误的 ECC 信息。

10. 一种方法,包括:

在处理器电路处接收用于 ECC 编码的数据的纠错码(ECC)信息,其指示 ECC 编码的数据中的一个或多个错误;

确定 ECC 编码的数据是否包括单个错误;以及

基于该确定,识别用于 ECC 编码的数据中的单个错误的错误位置,或者生成标志以指示 ECC 编码的数据具有超过一个错误。

11. 权利要求 10 的方法,包括 :

基于所识别的错误位置来纠正所述单个错误 ;以及
将 ECC 编码的数据解码。

12. 权利要求 10 的方法,包括 ECC 将包括 Reed-Solomon (RS) 码或二进制 Bose, Chaudhuri, Hocquenghem (BCH) 码中的一个。

13. 权利要求 12 的方法,包括 :

基于包括 RS 码的 ECC 来识别与所识别错误位置相关联的错误值。

14. 权利要求 12 的方法,包括 :

基于与 RS 码或二进制 BCH 码中的一个相关联的伴随式检查来确定 ECC 编码的数据具有一个或多个错误。

15. 权利要求 10 的方法,包括 :

响应于生成的标志而识别用于包括在 ECC 编码的数据中的超过一个错误的单独错误位置 ;

基于单独识别的错误位置来纠正所述超过一个错误 ;以及
将 ECC 编码的数据解码。

16. 权利要求 15 的方法,包括 ECC 将包括 Reed-Solomon (RS) 码或二进制 Bose, Chaudhuri, Hocquenghem (BCH) 码中的一个。

17. 权利要求 16 的方法,识别用于超过一个错误的单独错误位置包括实现 Berlekamp-Massey 算法(BMA) 和 Chien 搜索以单独地识别错误值和位置。

18. 权利要求 12 的方法,包括 :

基于包括 RS 码的 ECC 来识别与给定识别错误位置相关联的错误值 ;以及

基于实现包括 $e = \frac{s_n^2}{s_{2n}}$ 和 $j = \log \frac{s_{n+1}}{s_n}$ 的算法识别用于单个错误的错误值和给定识别错误位置,其中 e 是错误值、j 是给定识别错误位置,S 表示包括在接收到的 ECC 信息中的部分伴随式且 n 表示任何正整数。

19. 权利要求 16 的方法,包括 ECC 编码的数据被使用二进制 BCH 码编码,并且基于实现包括 $j = \log s_1$ 的算法来识别用于单个错误的错误位置,其中, j 是给定识别错误位置且 S 表示包括在接收到的 ECC 信息中的部分伴随式。

20. 至少一个机器可读介质,包括多个指令,其响应于在计算设备上被执行而使得计算设备执行根据权利要求 10 到 19 中的任一项所述的方法。

21. 一种包括用于执行权利要求 10 至 19 中的任一项的方法的装置的设备。

与用于编码的数据的纠错相关联的技术

背景技术

[0001] 纠错码(ECC)可用来保护数据或从与经由其发射或存储数据的介质有关的错误恢复。例如,可使用ECC对数据进行编码以可能从与有线/无线通信、到存储器设备/介质的存储或诸如2维条形码读取器之类的光学读取器相关联的错误恢复。通过从存储器设备/介质或条形码读取数据接收到或经由有线/无线通信信道接收到的ECC编码的数据可能能够识别和纠正给定数目的错误。通常,ECC编码的数据可包括具有数据和冗余或奇偶位或符号的组合的代码字。根据给定代码字的大小和期望的保护水平,代码字可在大小方面改变,并且还可在用来从可能错误中恢复的算法的复杂性方面改变。

[0002] 可将给定时间段中的错误称为误码率(BER)。已大大地增加数据传输速度的数字信号传输中的技术进步也已增加较高BER的可能性。并且,存储器存储/介质技术已导致越来越稠密的存储,其也可导致较高BER的增加的可能性。为了减少可能较高BER的影响,可用较大的代码字对数据进行编码。这些较大代码字可具有更多奇偶位。具有更多奇偶位的大的代码字可要求用增加的计算资源量实现的复杂算法。

附图说明

- [0003] 图1图示出示例存储器系统。
- [0004] 图2图示出示例存储系统。
- [0005] 图3图示出示例纠错码(ECC)系统。
- [0006] 图4图示出示例单错系统。
- [0007] 图5图示出示例单/多错系统。
- [0008] 图6图示出示例设备。
- [0009] 图7图示出示例逻辑流程。
- [0010] 图8图示出示例存储介质。
- [0011] 图9图示出示例计算平台。

具体实施方式

[0012] 如在本公开中设想的,具有更多奇偶位的大的代码字可要求用增加的计算资源量实现的复杂算法以便减小BER。在某些实施例中,存储器存储技术的用户寻求减小BER、但仍使由使用增加的计算资源量(例如,处理器时钟周期)而可能引起的延迟最小化之间的平衡。诸如与非易失性存储器(例如,相变存储器)相关联的那些之类的某些存储器存储技术可具有相对快速的数据访问时间,但是非易失性存储器的固有物理特性可导致用于BER的较高潜在性。因此,使用相对大的代码字来保护数据并抵消用于BER的较高潜在性。然而,当检测到错误时,可使用更多的计算资源来将这些大的代码字解码。更多计算资源可将访问时间增加至对于至少某些类型的非易失性存储器(例如,二级存储器(2LM)或固态驱动器(SSD))的某些期望使用而言不可接受的水平。相对于这些及其他挑战而言需要本文所述的示例。

[0013] 在某些示例中,可实现与用于编码的数据的纠错相关联的技术。这些技术可包括在处理器电路处接收用于 ECC 编码的数据的纠错码(ECC)信息,其指示 ECC 编码的数据中的一个或多个错误。然后可进行关于 ECC 编码的数据是否包括单个错误的确定。基于该确定,可识别用于 ECC 编码的数据中的单个错误的错误位置,或者可生成标志以指示 ECC 编码的数据已超过一个错误。对于包括超过一个错误的 ECC 编码的数据的示例而言,可识别单独的错误位置。然后可纠正一个或多个错误,并可将 ECC 编码的数据解码。

[0014] 图 1 图示出示例存储器系统 100。如图 1 中所示,存储器系统 100 包括控制器 110 和存储器 120。根据某些示例,控制器 110 可经由通信链路 130 来接收和 / 或履行读 / 写请求。虽然在图 1 中未示出,但在某些示例中,通信链路 130 可将控制器 110 通信耦合到与用于计算设备的操作系统相关联的元件或特征。对于这些示例而言,存储器系统 100 可以是用于计算设备的存储器设备。作为存储器设备,存储器系统 100 可充当二级存储器(2LM)系统或用于计算设备的固态驱动器(SSD)。

[0015] 在某些示例中,如图 1 中所示,控制器 110 可包括纠错码(ECC)编码器 112 和 ECC 解码器 114。ECC 编码器 112 可包括用以生成代码字以保护要写入存储器 120 的数据的逻辑和 / 或特征。如下面更详细地描述的,ECC 解码器 114 可包括将对包括在 ECC 编码的数据中的错误进行检测、定位、可能评估和纠正的逻辑和 / 或特征。根据某些示例,用来对数据进行编码的 ECC 可包括但不限于 Reed-Solomon (RS) 码或二进制 Bose, Chaudhuri, Hocquenghem (BCH) 码。

[0016] 在某些示例中,存储器 120 可包括非易失性和 / 或易失性类型的存储器。非易失性类型的存储器可包括但不限于相变存储器(PCM)、PCM 和开关(PCMS)、闪速存储器、铁电存储器、硅氧化物氮化物氧化物硅(SONOS)存储器、诸如铁电聚合物存储器之类的聚合物存储器、纳米线、铁电晶体管随机访问存储器(FeTRAM 或 FeRAM)、双向开关半导体(ovonic)存储器或电可擦可编程只读存储器(EEPROM)。易失性类型的存储器可包括但不限于动态随机访问存储器(DRAM)或静态 RAM(SRAM)。

[0017] 在某些示例中,存储器 120 还可包括存储介质类型,诸如光盘,包括但不限于紧凑式盘(CD)、数字多功能盘(DVD)、高清晰度 DVD(HD DVD)或蓝光盘。

[0018] 根据其中将存储器系统 100 配置为 2LM 系统的某些示例,存储器系统 100 可充当用于计算设备的主存储器。对于这些示例而言,存储器 120 可包括存储器的两级,包括系统盘级储存器的高速缓存子集。在此配置中,主存储器可包括被布置成包括易失性类型存储器的“近存储器”和被布置成包括易失性或非易失性类型存储器的“远存储器”。远存储器可包括比包括在近存储器中的易失性存储器更大且可能更慢的易失性或非易失性存储器。可向用于计算设备的操作系统(OS)将远存储器呈现为“主存储器”,而近存储器是用于对于 OS 而言透明的远存储器的高速缓存。2LM 系统的管理可由经由用于计算设备的控制器 110 和 / 或处理电路(例如,CPU)执行的逻辑和模块的组合完成。可经由高带宽、低延迟装置将近存储器耦合到处理电路以用于高效处理。可经由低带宽、高延迟装置将远存储器耦合到处理电路。

[0019] 图 2 图示出示例存储系统。在某些示例中,如图 2 中所示,存储系统 200 包括压缩 / 加密单元 205、ECC 编码器 112、加扰器单元 210、存储器 120、解扰器单元 205、ECC 解码器 114 和解密 / 解压缩单元 220。

[0020] 根据某些示例,如图 2 中所示,压缩 / 加密单元 205 处的数据的压缩 / 加密可导致“ u ”。ECC 编码器 112 可接收 u 并使用 ECC 代码(例如,二进制 BCH 或 RS)来生成代码字“ x ”。加扰器单元 210 可接收 x 并使得 x 被编程或写到存储器 120。可从存储器 120 读取代码字并由解扰器单元 205 解扰以导致代码字“ y ”。如图 2 中所示, $y = x + e_n$,其中,“ e_n ”表示在 x 的写然后从存储器 120 读取期间可能引入的错误,并且“ n ”表示在写和从存储器 120 读取期间引入的错误的数目。ECC 解码器 114 可接收 y 且可能纠正所识别的错误以生成“ d ”。如图 2 中所示,如果错误是可纠正的,则 $d = u$ 。解密 / 解压缩单元 220 然后可将 u 解密 / 解压缩以生成最初被压缩 / 加密单元 205 压缩 / 加密的数据。

[0021] 在某些示例中,ECC 解码器 114 可包括将接收具有 e_n 的 ECC 编码的数据 y 的逻辑和 / 或特征。ECC 解码器 114 可确定 e_n 是包括单个错误还是多个错误。对于这些示例而言,如果 e_n 包括单个错误,则 ECC 解码器 114 可包括将识别 ECC 编码的数据 y 中的错误位置和 / 或错误值的逻辑和 / 或特征或生成将指示 y 包括超过一个错误的标志。如果生成标志,则 ECC 解码器 114 可经历更加计算资源密集的 ECC 过程以单独地识别 y 中的错误位置和 / 或值。在尝试识别 y 中的多个错误位置和 / 或值之前识别用于 y 中的单个错误的位置和 / 或错误值的此过程可基本上减少用于当在 ECC 编码的数据中检测到错误时主要具有单个错误的存储器类型的纠错延迟。

[0022] 根据某些示例,当用来将 y 解码的 ECC 是 RS 码或二进制 BCH 码时识别 y 中的所述多个错误 e_n 可使用若干处理器时钟周期来识别可能错误。例如,可由包括多个计算以生成错误位置多项式的 ECC 解码器 114 的逻辑和 / 或特征来实现 Berlekamp-Massey 算法(BMA)。然后可由 ECC 解码器 114 来实现 Chien 搜索以对错误进行定位。并且,如果用来将 y 解码的 ECC 是 RS 码,则可使用附加处理器时钟周期来识别用于已定位错误的错误值。

[0023] 虽然图 2 描述了具有可能由向存储器 120 写入 / 读取数据引起的错误的 ECC 编码的数据。还可设想其他类型的 ECC 编码的数据,诸如与有线 / 无线通信相关联的 ECC 编码的数据或与用于 2 维条形码读取器的光学读取器相关联的 ECC 编码的数据。

[0024] 在某些示例中,单错识别过程可基于用来将 u 编码和将 y 解码两者的 ECC 代码的类型。例如,对于二进制 BCH 码而言,当存在一个错误时,识别 y 中的错误的部分伴随式(syndrome)可具有例如以下等式(1)所示的性质,其中, j 是错误的位置。

[0025] 示例等式(1)

$$\begin{aligned} S_1 &= \alpha^j \\ S_2 &= \alpha^{2j} \\ S_3 &= \alpha^{3j} \\ &\dots \\ S_{2t} &= \alpha^{2tj} \end{aligned}$$

根据某些示例,为了纠正单个位错误,可进行查看该部分伴随式是否形成几何级数的检查。可基于示例算法或等式(2)来确定错误位置 j 。

[0026] 示例等式(2) :

$$j = \alpha^j = \log S_i$$

在某些示例中，ECC 解码器 114 可包括将经由查找表或对数计算单元的使用来实现示例等式(2)的逻辑和 / 或特征。

[0027] 根据某些示例，在二进制 BCH 中，偶部分伴随式完全由奇部分伴随式确定，因此，可不需要检查偶部分伴随式。对于这些示例而言，示例等式(3)可检查：

示例等式(3)

$$\begin{aligned}S_3 &= (S_1)^3 \\S_5 &= (S_1)^5 \\S_7 &= (S_1)^7 \\\dots \\S_{2t-1} &= (S_1)^{2t-1}\end{aligned}$$

并且，为了检查几何级数，可由示例等式(5)示出：

示例等式(4)：

$$\alpha^{2j} = \frac{S_3}{S_1} = \frac{S_5}{S_3} = \frac{S_7}{S_5} = \dots = \frac{S_{2t-1}}{S_{2t-3}}$$

然而，使用示例等式(4)来检查几何级数可要求许多除法，其在计算资源(例如，时钟周期)方面可能是相对昂贵的。替代地，可使用示例等式(5)来检查：

示例等式(5)：

$$\alpha^{2tj} = S_1 S_{2t-1} = S_3 S_{2t-3} = S_5 S_{2t-5} = \dots = S_{2\lceil t/2 \rceil - 1} S_{2\lceil t/2 \rceil + 1}$$

示例等式(5)将显示第一 t 奇部分伴随式且第二 t 奇部分伴随式是对称的。然后可使用示例等式(6)来仅检查第一 t 奇部分伴随式形成几何级数。

[0028] 示例等式(6)：

$$\alpha^{(t+2)j} = S_1 S_{t+1} = S_3 S_{t-1} = S_5 S_{t-3} = \dots = S_{2\lceil t/2 \rceil / 2 - 1} S_{2\lceil t/2 \rceil / 2 + 1}$$

使用示例等式(6)的检查将显示第一 $\lceil t/2 \rceil$ 奇部分伴随式和第二 $\lceil t/2 \rceil$ 奇部分伴随式是对称的。然后，可使用示例等式(7)来仅检查第一 $\lceil t/2 \rceil$ 奇部分伴随式形成几何级数并继续进行分步解决(divide and conquer)直至可以显示 S_1, S_3 和 S_5 形成几何级数为止。

[0029] 示例等式(7)：

$$\begin{aligned}e^2 \alpha^{(\lceil t/2 \rceil + 1)j} &= S_1 S_{\lceil t/2 \rceil} = S_2 S_{\lceil t/2 \rceil - 1} = S_3 S_{\lceil t/2 \rceil - 2} = \dots = S_{\lceil \lceil t/2 \rceil / 2 \rceil} S_{\lceil \lceil t/2 \rceil / 2 \rceil + 1} \\\dots \\ \alpha^{6j} &= S_1 S_5 = S_3^2\end{aligned}$$

此外，为了检查形成几何级数的部分伴随式与示例等式(3)中的形式匹配，检查满足示例等式(8)：

示例等式(8)：

$$S_3 = (S_1)^3$$

根据某些示例，用于二进制 BCH 码的单错纠正可要求 $(t-1)$ 个 Galois 域恒定功率函数

和 Galois 域乘法器。解码器 114 处的逻辑和 / 或特征可包括 $(t-1)$ 个 m 位比较器和至少一个查找表或对数计算单元的使用。根据某些其他示例,解码器 114 处的逻辑和 / 或特征可包括约 t 个乘法器和 t 个比较器或者至少 $t-1$ 个乘法器和 $t-1$ 个比较器以将奇部分部分伴随式相乘以检查奇部分伴随式是否形成几何级数。

[0030] 在某些示例中,对于基于 RS 码的单错纠正而言,当存在一个错误时,识别 y 中的错误的部分伴随式可具有例如以下等式(9)所示的性质,其中, j 是错误的位置且 e 是错误值。

[0031] 示例等式(9) :

$$\begin{aligned} S_1 &= e \alpha^j \\ S_2 &= e \alpha^{2j} \\ S_3 &= e \alpha^{3j} \\ &\dots \\ S_{2t} &= e \alpha^{2tj} \end{aligned}$$

根据某些示例,为了纠正单个错误,可使用示例等式(10)来检查部分伴随式形成几何级数且然后针对 e 和 j 求解。

[0032] 示例等式(10) :

$$\alpha^j = \frac{S_2}{S_1} = \frac{S_3}{S_2} = \frac{S_4}{S_3} = \dots = \frac{S_{2t}}{S_{2t-1}}$$

然而,使用示例等式(10)来检查几何级数可要求许多除法,其在计算资源方面是相对昂贵的。替代地,可使用示例等式(11)来检查:

示例等式(11) :

$$e^2 \alpha^{(2t+1)j} = S_1 S_{2t} = S_2 S_{2t-1} = S_3 S_{2t-2} = \dots = S_t S_{t+1}$$

示例等式(11)将显示第一 t 奇部分伴随式且第二 t 部分伴随式是对称的。然后可使用示例等式(12)来仅检查第一 t 奇部分伴随式形成几何级数。

[0033] 示例等式(12) :

$$e^2 \alpha^{(t+2)j} = S_1 S_{t+1} = S_2 S_t = S_3 S_{t-1} = \dots = S_{[(t+1)/2]} S_{[(t+1)/2]+1}$$

使用示例等式(12)的检查将显示第一 $[t/2]$ 部分伴随式和第二 $[t/2]$ 部分伴随式是对称的。然后,可使用示例等式(13)来仅检查第一 $[t/2]$ 部分伴随式形成几何级数并继续进行分步解决直至可以显示 S_1 、 S_3 和 S_5 形成几何级数为止。

[0034] 示例等式(13) :

$$\begin{aligned} e^2 \alpha^{([(t+1)/2]+2)j} &= S_1 S_{[(t+1)/2]+1} = S_2 S_{[(t+1)/2]-1} = S_3 S_{[(t+1)/2]-2} = \dots \\ &= S_{[(t+1)/2]} S_{[([(t+1)/2]+1)/2]+1} \\ &\dots \\ e^2 \alpha^{4j} &= S_1 S_3 = S_2^2 \end{aligned}$$

在某些示例中,然后可使用示例等式(14)来针对 e 和 j 求解。

[0035] 示例等式(14) :

$$\begin{aligned}
 e &= \frac{e^2 \alpha^{2j}}{e \alpha^{2j}} = \frac{s_1^2}{s_2} = \frac{s_n^2}{s_{2n}} \\
 \alpha^j &= \frac{e \alpha^{3j}}{e \alpha^{2j}} = \frac{s_3}{s_2} = \frac{s_{n+1}}{s_n} \\
 j &= \log \frac{s_{n+1}}{s_n}
 \end{aligned}$$

根据某些示例,解码器 114 可包括将实现用于 RS 码的单错纠正的逻辑和 / 或特征。该逻辑和 / 或特征可包括约 $2t$ 个乘法器和 $2t$ 个比较器或者至少 $2t-1$ 个乘法器和 $2t-1$ 个比较器以将部分伴随式相乘以检查该部分伴随式是否形成几何级数。逻辑和 / 或特征还可以能够进行常数平方、Galois 域求逆,并且具有两个乘法器以计算 e 和 j 。

[0036] 图 3 图示出示例性纠错码(ECC)系统 300。在某些示例中,如图 3 中所示,ECC 系统 300 包括 ECC 编码器 112、存储器 120 和 ECC 解码器 114。并且,ECC 解码器 114 被示为包括错误检测器 / 伴随式计算器 305、错误识别单元 315、纠正器单元 325 和代码字缓冲器 335。

[0037] 在某些示例中,可由 ECC 编码器 112 使用可包括二进制 BCH 码或 RS 码的 ECC 代码来对数据进行编码(可能已加密 / 压缩)。然后可将结果得到的代码字存储到存储器 120。根据某些示例,可从存储器 120 读取所存储的代码字且其可包括可能的错误。如图 3 中所示,可将具有可能错误的代码字转发至代码字缓冲器 335。可将代码字缓冲器 335 配置成至少临时地存储代码字,而 ECC 解码器 114 的其他元件识别用于可能包括在代码字中的每个错误的位置和 / 或值。

[0038] 根据某些示例,可将错误检测器 / 伴随式计算器 305 配置成确定代码字是否包括任何错误。对于这些示例而言,如果未检测到错误,则 ECC 解码器 114 可包括指示纠正器单元 325 或代码字缓冲器 335 转发存储在代码字缓冲器 335 处的代码字的逻辑和 / 或特征。然而,如果检测到错误,则错误检测器 / 伴随式计算器 305 可计算用于 ECC 编码的数据的部分伴随式并将该计算的部分伴随式转发到错误识别单元 315。然后可由错误识别模块 315 来识别错误位置和 / 或值。如下面更多地描述的,可使用单错电路或多错电路来识别错误位置和 / 或值。

[0039] 在某些示例中,如图 3 中所示,可从错误识别电路 315 向纠正器单元 325 提供错误位置和 / 或值。对于这些示例而言,可将纠正器单元 325 配置成基于错误位置和 / 或值来纠正所识别的错误。还可将纠正器单元 325 配置成将代码字解码且然后转发数据以用于可能的解压缩 / 解密并最后递送到读请求的源。

[0040] 图 4 图示出示例单个错误系统。在某些示例中,如图 4 中所示,单错系统 400 包括错误检测器 / 伴随式计算器 305 以向单错电路 410 提供部分伴随式。对于这些示例而言,单错电路 410 可以是上文针对图 3 所述的错误识别单元 315 的一部分。

[0041] 根据某些示例,如图 4 中所示,单错电路 410 包括伴随式乘法器 412、乘积比较器 414、对数单元 415、求逆计算器 416 和伴随式乘法器 418。在某些示例中,如果用来对 ECC 编码的数据进行编码的 ECC 包括二进制 BCH 码,则可使用伴随式乘法器 412、乘积比较器 414 和对数单元 415 来确定 ECC 编码的数据是否包括单个错误并基于此确定对该单个错误进行定位。对于这些示例而言,如果所使用的 ECC 尝试防止 t 个错误,则可将伴随式乘法器 412

布置成执行 $(t-1)$ Galois 域恒定功率函数, 如上所述, 例如等式(7)。可将乘积比较器 414 布置成具有 $(t-1)$ 个 m 位比较器, 也如上所述, 例如等式(7)。并且可将对数单元 415 布置成基于从错误检测器 / 伴随式计算器 305 接收的部分伴随式和示例等式(2)的实现以针对 j 求解来确定单个错误位置。

[0042] 在某些示例中, 如果用来对 ECC 编码的数据进行编码的 ECC 包括 BCH 代码, 则可使用伴随式乘法器 412、乘积比较器 414、对数单元 415、求逆计算器 416 和伴随式乘法器 418 来确定 ECC 编码的数据是否包括单个错误, 基于此确定对该单个错误进行定位并识别用于该单个错误的值。对于这些示例而言, 如果所使用的 ECC 尝试防止 t 个错误, 则可将伴随式乘法器 412 布置成包括 t 个乘法器, 并且可将乘积比较器 414 布置成包括至少 t 个比较器, 如上所述, 例如等式(14)。并且, 可将求逆计算器 416 布置成根据示例等式(14)来执行 Galois 域求逆, 并且伴随式乘法器 418 可包括至少两个乘法器以计算用于 j 的值以确定单个错误的位置。根据某些示例, 在伴随式乘法器 412/ 乘积比较器 414 和对数单元 415/ 求逆计算器 416/ 伴随式乘法器 418 处执行的操作可并行地发生, 并且因此可用组合电路来实现。

[0043] 在某些示例中, 如果用来对 ECC 编码的数据进行编码的 ECC 包括 RS 代码, 则可使用伴随式乘法器 412、乘积比较器 414、求逆计算器 416 和伴随式乘法器 418 来确定 ECC 编码的数据是否包括单个错误, 基于此确定对该单个错误进行定位并识别用于该单个错误的值。对于这些示例而言, 如果所使用的 ECC 尝试防止 t 个错误, 则可将伴随式乘法器 412 布置成包括 $2t$ 个乘法器, 并且可将乘积比较器 414 布置成包括至少 $2t$ 个比较器, 如上所述, 例如等式(14)。并且, 可将求逆计算器 416 布置成根据示例等式(14)来执行 Galois 域求逆, 并且伴随式乘法器 418 可包括至少两个乘法器以计算用于 e 和 j 的值以确定单个错误的值和位置。根据某些示例, 在伴随式乘法器 412/ 乘积比较器 414 和对数单元 415/ 求逆计算器 416/ 伴随式乘法器 418 处执行的操作可并行地发生, 并且因此可用组合电路来实现。

[0044] 根据某些示例, 如果识别到超过一个错误, 则可将单错电路 410 配置或布置成生成多错标志。如下面更多地所述, 响应于多错标志, 可将多错电路配置成对由所使用的特定 ECC 保护的 t 个可能错误的全部数目进行定位和 / 或识别其值。

[0045] 图 5 图示出示例单 / 多错误系统 500。在某些示例中, 如图 5 中所示, 单 / 多错系统 500 包括错误检测器 / 伴随式计算器 305、错误识别单元 315、多错电路 510、复用器 (MUX) 520 和纠正器单元 335。对于这些示例而言, 多错电路 510 在图 5 中被示为包括 Berlekamp-Massey 算法 (BMA) 单元 516、Chien 搜索单元 518 和错误评估器 519。

[0046] 根据某些示例, 错误检测器 / 伴随式计算器 305 可检测到在 ECC 编码的数据中可包括一个或多个错误, 并生成指示所述一个或多个错误的部分伴随式。对于这些示例而言, 单错电路 410 可快速地确定部分伴随式是否指示单个错误并识别用于该单个错误的错误位置和 / 或值。可将 MUX 520 布置成允许来自单错电路 410 的信息默认地传递至纠正器单元 335, 例如在 MUX 520 上未声称的选择器位。然而, 如果在 ECC 编码的数据中发现多个错误, 则单错电路 410 可生成多错标志, 其可导致 MUX 520 将从单错电路 410 输入到 MUX 520 的任何信号阻塞 (blocking)。错误识别单元 315 然后可激活多错电路 510。

[0047] 在某些示例中, 可能已使用能够保护编码的数据免于 t 个错误的二进制 BCH 码来保护 ECC 编码的数据。对于这些示例而言, 可将 BMA 单元 516 和 Chien 搜索单元 518 布置

成识别包括在 ECC 编码的数据中的错误的单独位置直至至少 t 个错误为止。由于生成了多错标志, 所以可将 MUX 520 布置成允许多错电路 510 将指示所识别错误位置的信息传递 / 转发至纠正器单元 335。

[0048] 在某些示例中, 可能已使用能够保护编码的数据免于 t 个错误的 RS 码来保护 ECC 编码的数据。对于这些示例而言, 可将 BMA 单元 516、Chien 搜索单元 518 和错误评估器 519 布置成识别包括在 ECC 编码的数据中错误的单独位置和值直至至少 t 个错误为止。如上所述, 可将 MUX 520 布置成允许多错电路 510 将指示所识别错误位置和值的信息传递 / 转发至纠正器单元 335。

[0049] 在某些示例中, 可使用其他算法方案来识别 ECC 编码的数据中的错误位置和 / 或值。这些其他算法可包括但不限于 Peterson-Goresntein-Zierler 算法或欧几里德算法。

[0050] 图 6 图示出示例设备 600。虽然图 6 中所示的设备 600 在某个拓扑中具有有限数目的元件, 但可认识到的是设备 600 可针对给定实施根据期望在替代拓扑中包括更多或更少的元件。

[0051] 设备 600 可包括计算机实现设备, 其可包括上文针对 ECC 解码器 114 所述的逻辑和 / 或特征中的至少某些, 如上文针对图 1—5 所述。可将计算机实现设备 600 布置成执行一个或多个软件部件 622-a。值得注意的是如本文所使用的“a”和“b”和“c”和类似标志符(designator)意图是表示任何正整数的变量。因此, 例如, 如果实施设定用于 $a = 5$ 的值, 则一整套软件部件 622-a 可包括模块 622-1、622-2、622-3、622-4 或 622-5。在此背景下实施例不受限制。

[0052] 根据某些示例, 设备 600 可能能够与用于存储器系统的控制器或 ECC 解码器一起定位, 例如作为诸如存储器系统 100 之类的存储器系统的一部分。对于这些示例而言, 可将设备 600 包括在处理器、处理器电路、微控制器电路或专用集成电路(ASIC)中或由它们实现。在其他示例中, 可将设备 600 实现为固件(BIOS)的一部分或实现为中间件应用。在此背景下示例不受限制。

[0053] 在某些示例中, 如果在处理器中实现, 则一般地可将处理器布置成执行一个或多个软件部件 622-a。处理器可以是各种市售处理器中的任何一个, 在没有限制的情况下包括 AMD® Athlon®、Duron® 和 Opteron® 处理器; ARM® 应用、嵌入式和安全处理器; IBM® 和 Motorola® DragonBall® 和 PowerPC® 处理器; IBM 和 Sony® Cell 处理器; Intel® Celeron®、Core (2) Duo®、Core i3、Core i5、Core i7、Pentium® 和 XScale® 处理器; 以及类似处理器。还可采用多核处理器及其他多处理器架构来实现设备 600。

[0054] 根据某些示例, 设备 600 可包括检错部件 622-1。检错部件 622-1 可被布置成用于由处理器电路 620 执行以确定经由代码字 610 接收到的 ECC 编码的数据可具有一个或多个错误。对于这些示例而言, 检错部件 622-1 可计算部分伴随式, 其可导致指示 ECC 编码的数据中的错误的非零值。这些部分伴随式可至少临时地被检错部件 622-2 保持(例如, 存储在诸如寄存器之类的数据结构中)。在某些示例中, 部分伴随式可基于用于纠错的代码, 诸如二进制 BCH 码或 RS 码。

[0055] 在某些示例中, 设备 600 还可包括单错部件 622-1。单错部件 622-2 可被布置成用于由处理器电路 620 执行以接收用于 ECC 编码的数据的 ECC 信息, 其指示 ECC 编码的数据中的一个或多个错误(例如, 包括在部分伴随式中)。还可将单错部件 622-2 布置成确定

ECC 编码的数据是否包括单个错误,如果 ECC 编码的数据被确定为具有单个错误,则识别该单个错误的位置和 / 或值。如果 ECC 信息指示多个错误,则单错部件 622-2 可被布置成生成将指示多个错误的标志。

[0056] 根据某些示例,单错部件 622-2 可至少临时地保持伴随式乘法 / 比较 626-b 和错误评估 626-c(例如,在诸如寄存器之类的数据结构中)。对于这些示例而言,如果用来对 ECC 编码的数据进行编码的 ECC 是二进制 BCH 码,则乘法 / 比较 626-b 可包括用于该单个错误的错误位置信息。如果用来对 ECC 编码的数据进行编码的 ECC 是 RS 码,则错误评估 626-c 可包括用于该单个错误的错误位置信息和错误值两者。

[0057] 在某些示例中,设备 600 还可包括多错部件 622-3。可将多错部件 622-3 布置成用于由处理器电路 620 执行以接收用于 ECC 编码的数据的 ECC 信息,指示多个错误。可将多错部件 622-3 布置成单独地识别用于每个错误的错误位置和 / 或值直至至少 t 个为止。

[0058] 根据某些示例,多错电路 622-3 可至少临时地保持 BMA 结果 626-d、Chien 搜索结果 626-e 和错误评估 626-f,例如在诸如寄存器之类的数据结构中。对于这些示例而言,如果用来对 ECC 编码的数据进行编码的 ECC 是 RS 码或二进制 BCH 码,则 BMA 结果 626-d 和 Chien 搜索结果 626-e 可被多错部件 622-3 用来识别单独的错误位置。如果所使用的 ECC 是 RS 码,则错误评估 626-f 可包括与单独识别的错误位置相关联的错误值。

[0059] 设备 600 还可包括缓冲器部件 622-4。可将缓冲器部件 622-4 布置成用于由处理器电路 620 执行以保持能够临时地存储在代码字 610 中接收到的 ECC 编码的数据的存储器。

[0060] 根据某些示例,设备 600 还可包括纠正器部件 622-5。可将纠正器部件 622-5 布置成用于由处理器电路 620 执行以接收用于来自单错部件 622-2 的单个错误的所识别位置或者用于由多错部件 622-3 识别的多个错误的单独识别错误位置。还可将纠正器部件 622-5 布置成从由缓冲器部件 622-4 保持的存储器获得 ECC 编码的数据且然后纠正包括在代码字 610 中的 ECC 编码的数据。然后可将已纠正的 ECC 编码的数据解码。

[0061] 在这里包括一组逻辑流程,其表示用于执行公开架构的新颖方面的示例方法。虽然为了说明的简单起见将这里所示的一个或多个方法示出并描述为一系列动作,但本领域的技术人员将理解并认识到该方法不受动作顺序的限制。据此,某些动作可按照与本文所示和所述的不同的顺序和 / 或与其他动作同时地发生。例如,本领域的技术人员将理解并认识到的是可以替代地将该方法表示为一系列的互相关状态或事件,诸如用状态图。此外,对于新颖实施而言并非方法中所示的所有动作都可能是要求的。

[0062] 可用软件、固件和 / 或硬件来实现逻辑流程。在软件和固件实施例中,可用存储在至少一个非临时计算机可读介质或机器可读介质(诸如光学、磁或半导体储存器)上的计算机可执行指令来实现逻辑流程。在此背景下实施例不受限制。

[0063] 图 7 图示出逻辑流程 700。逻辑流程 700 可表示由本文所述的一个或多个逻辑、特征或器件,诸如设备 600,执行的某些或所有操作。更特别地,可用检错部件 622-1、单错部件 622-2、多错部件 622-3、缓冲器部件 622-4 或纠正器部件 622-5 来实现逻辑流程 700。

[0064] 根据某些示例,在方框 702 处,逻辑流程 700 可接收用于 ECC 编码的数据的 ECC 信息,指示 ECC 编码的数据中的一个或多个错误。对于这些示例而言,单错部件 622-2 可接收 ECC 信息作为具有至少某些非零值的部分伴随式,其指示 ECC 编码的数据中的一个或多个

错误。

[0065] 在某些示例中，在方框 704 处，逻辑流程 700 可确定编码的数据是否包括单个错误。

[0066] 根据某些示例，基于 ECC 编码的数据是否包括单个错误的确定，在方框 706 处，逻辑流程 700 可识别用于 ECC 编码的数据中的单个错误的错误位置，或者生成将指示 ECC 编码的数据具有超过一个错误的标志。如果确定单个错误，则在方框 708 处，逻辑流程可基于所识别的错误位置来纠正单个错误并将 ECC 编码的数据解码。

[0067] 在某些示例中，在方框 710 处，逻辑流程 700 可响应于生成的标志而识别用于 ECC 编码的数据中的超过一个错误的单独错误位置。对于这些示例而言，单错部件 622-2 可能已生成将指示超过一个错误的多错标志。并且，单独错误位置的识别可包括使用保持在多错部件 622-3 处的 BMA 信息 626-d 和 Chien 搜索信息 626-e。

[0068] 根据某些示例，在方框 712 处，逻辑流程 700 可基于单独识别的错误位置来纠正超过一个错误并将 ECC 编码的数据解码。对于这些示例而言，纠正器部件 622-5 可接收单独识别的错误位置并纠正可能已被临时地存储在由缓冲器部件 622-4 保持的存储器处的 ECC 编码的数据。然后可由纠正器部件 622-5 将已纠正的 ECC 编码的数据解码。

[0069] 图 8 图示出存储介质 800 的实施例。存储介质 800 可包括制品。在某些示例中，存储介质 800 可包括任何非临时计算机可读介质或机器可读介质，诸如光学、磁或半导体储存器。存储介质 800 可存储各种类型的计算机可执行指令，诸如用以实现逻辑流程 700 的指令。计算机可读或机器可读存储介质的示例可包括能够存储电子数据的任何有形介质，包括易失性存储器或非易失性存储器、可移动或不可移动存储器、可擦或不可擦存储器、可写或可重写存储器等。计算机可执行指令的示例可包括任何适当类型的代码，诸如源代码、编译代码、解释代码、可执行代码、静态代码、动态代码、面向对象代码、可视代码等。在此背景下示例不受限制。

[0070] 图 9 图示出示例计算平台 900。在某些示例中，如图 9 中所示，计算平台 900 可包括存储器系统 930、处理部件 940、其他平台部件 950 或通信接口 960。根据某些示例，可在计算设备中实现计算平台 900。

[0071] 根据某些示例，存储器系统 930 可类似于存储器系统 100。对于这些示例而言，驻留于存储器系统 930 处或与之一起定位的逻辑和 / 或特征(例如，包括在 ECC 解码器 / 控制器中)可执行用于设备 600 的至少某些处理操作或逻辑。并且，存储器系统 930 可包括易失性或非易失性类型的存储器(未示出)，其可存储以与如上所述类似的方式被写入包括在存储器系统 100 中的存储器 120 或从其读取的 ECC 编码的数据。

[0072] 根据某些示例，处理部件 940 还可执行用于设备 600 和 / 或存储介质 800 的至少某些处理操作。处理部件 910 可包括各种硬件元件、软件元件或两者的组合。硬件元件的示例可包括器件、逻辑器件、部件、处理器、微处理器、电路、处理器电路、电路元件(例如，晶体管、电阻器、电容器、电感器等)、集成电路、专用集成电路(ASIC)、可编程逻辑器件(PLD)、数字信号处理器(DSP)、现场可编程门阵列(FPGA)、存储器单元、逻辑门、寄存器、半导体器件、芯片、微芯片、芯片组等。软件元件的示例可包括软件部件、程序、应用、计算机程序、应用程序、系统程序、软件开发程序 / 机器程序、操作系统软件、中间件、固件、软件模块、例程、子例程、函数、方法、程序、软件接口、应用程序接口(API)、指令集、计算代码、计算机代

码、代码段、计算机代码段、字、值、符号或其任何组合。确定是否使用硬件元件和 / 或软件元件来实现示例可根据许多因素而改变,按照给定示例所期望的,诸如期望计算速率、功率水平、耐热性、处理周期预算、输入数据速率、输出数据速率、存储器资源、数据总线速度以及其他设计或性能约束。

[0073] 在某些示例中,其他平台部件 950 可包括公共计算元件,诸如一个或多个处理器、多核处理器、协处理器、存储器单元、芯片组、控制器、外围设备、接口、振荡器、定时设备、视频卡、声卡、多媒体输入 / 输出(I/O)部件(例如,数字显示器)、电源等。与其他平台部件 950 或存储器系统 930 相关联的存储器单元的示例在没有限制的情况下可包括采取一个或多个高速存储器单元形式的各种类型的计算机可读和机器可读存储介质,诸如只读存储器(ROM)、RAM、DRAM、双倍数据速率 DRAM(DDR AM)、同步 DRAM(SDRAM)、SRAM、可编程 ROM(PROM)、可擦可编程 ROM(EPROM)、电可擦可编程 ROM(EEPROM)、闪速存储器、纳米线、铁电晶体管随机访问存储器(FeTRAM 或 FeRAM)、聚合物存储器,诸如铁电聚合物存储器、双向开关半导体存储器、相变或铁电存储器、硅氧化物氮化物氧化物硅(SONOS)存储器、磁或光学卡、器件阵列,诸如独立磁盘冗余阵列(RAID)驱动器、固态存储器设备(例如,USB 存储器)、固态驱动器(SSD)和适合于存储信息的任何其他类型的存储介质。

[0074] 在某些示例中,通信接口 960 可包括将支持通信接口的逻辑和 / 或特征。对于这些示例而言,通信接口 960 可包括一个或多个通信接口,其根据各种通信协议或标准进行操作以在直接或网络通信链路上进行通信。直接通信可经由在一个或多个行业标准(包括后代和变体)中所述的通信协议或标准的使用发生,诸如与系统管理总线(SMBus)规范、PCI 快速规范、串行高级技术附着(SAT A)规范、串行附着 SCSI(SAS)或通用串行总线(USB)规范相关联的那些。网络通信可经由通信协议或标准的使用发生,诸如在以太网标准中所述的那些。

[0075] 计算平台 900 可以是计算设备的一部分,其可以是例如用户设备、计算机、个人计算机(PC)、台式计算机、膝上型计算机、笔记本计算机、上网本计算机、平板电脑、智能电话、嵌入式电子装置、游戏控制台、服务器、服务器阵列或服务器场、web 服务器、网络服务器、因特网服务器、工作站、微型计算机、主机计算机、超级计算机、网络器械、web 设备、分布式计算系统、多处理器系统、基于处理器的系统或其组合。因此,按照适当的期望,可在计算平台 900 的各种实施例中包括或省略本文所述的计算平台 900 的功能和 / 或特定配置。

[0076] 可使用分立电路、专用集成电路(ASIC)、逻辑门和 / 或单片架构的任何组合来实现计算平台 900 的部件和特征。此外,可使用微控制器、可编程逻辑阵列和 / 或微处理器或者在合适地适当的情况下前述各项的任何组合来实现计算平台 900 的特征。应注意的是在本文中可将硬件、固件和 / 或软件元件共同地或单独地称为“逻辑”或“电路”。

[0077] 应认识到的是图 9 的框图中所示的示例性计算平台 900 可表示许多潜在实施的一个功能描述示例。因此,附图中所描述的方框功能的划分、省略或包括并不推断用于实现这些功能的硬件部件、电路、软件和 / 或元件将必须被划分、省略或包括在实施例中。

[0078] 可用存储在至少一个机器可读介质上的典型指令来实现至少一个示例的一个或多个方面,该典型指令表示处理器内的各种逻辑,其在被机器、计算设备或系统读取时使得机器、计算设备或系统制造将执行本文所述技术的逻辑。可将称为“IP 核”的此类表示存储在有形、机器可读介质上并供应给各种客户或制造机构以加载到实际上实现逻辑或处理器

的制造机器。

[0079] 可使用硬件元件、软件元件或两者的组合来实现各种示例。在某些示例中，硬件元件可包括设备、处理器、微处理器、电路、电路元件（例如，晶体管、电阻器、电容器、电感器等）、集成电路、专用集成电路（ASIC）、可编程逻辑器件（PLD）、数字信号处理器（DSP）、现场可编程门阵列（FPGA）、存储器单元、逻辑门、寄存器、半导体器件、芯片、微芯片、芯片组等。在某些示例中，软件元件的示例可包括软件部件、程序、应用、计算机程序、应用程序、系统程序、机器程序、操作系统软件、中间件、固件、软件模块、例程、子例程、函数、方法、程序、软件接口、应用程序接口（API）、指令集、计算代码、计算机代码、代码段、计算机代码段、字、值、符号或其任何组合。确定是否使用硬件元件和 / 或软件元件来实现示例可根据许多因素而改变，按照给定实施方式的期望，诸如期望计算速率、功率水平、耐热性、处理周期预算、输入数据速率、输出数据速率、存储器资源、数据总线速度以及其他设计或性能约束。

[0080] 某些示例可包括制品或至少一个计算机可读介质。计算机可读介质可包括将存储逻辑的非临时存储介质。在某些示例中，非临时存储介质可包括能够存储电子数据的一个或多个类型的计算机可读存储介质，包括易失性存储器或非易失性存储器、可移动或不可移动存储器、可擦或不可擦存储器、可写或可重写存储器等。在某些示例中，所述逻辑可包括各种软件元件，诸如软件部件、程序、应用、计算机程序、应用程序、系统程序、机器程序、操作系统软件、中间件、固件、软件模块、例程、子例程、函数、方法、程序、软件接口、API、指令集、计算代码、计算机代码、代码段、计算机代码段、字、值、符号或其任何组合。

[0081] 根据某些示例，计算机可读介质可包括将存储或保持指令的非临时存储介质，该指令在被机器、计算设备或系统执行时使得机器、计算设备或系统执行根据所述示例的方法和 / 或操作。所述指令可包括任何适当类型的代码，诸如源代码、编译代码、解释代码、可执行代码、静态代码、动态代码等。可根据预定义计算机语言、方式或语法来实现该指令以便命令机器、计算设备或系统执行某个功能。可使用任何适当的高级、低级、面向对象、可视、编译和 / 或解释编程语言来实现指令。

[0082] 可使用表述“在一个示例中”或“示例”连同它们的派生词来描述某些示例。这些术语意味着结合示例所述的特定特征、结构或特性被包括在至少一个示例中。短语“在一个示例中”在本说明书中的不同地方的出现不一定全部指相同示例。

[0083] 可使用表述“耦合”和“连接”连同它们的派生词来描述某些示例。这些术语并不一定意图作为用于彼此的同义词。例如，使用术语“连接”和 / 或“耦合”的描述可指示两个或更多元件和相互进行直接的物理或电接触。然而，术语“耦合”还可意指两个或更多元件并未相互进行直接接触，但是仍相互合作或彼此交互。

[0084] 在某些示例中，示例方法可包括在处理器电路处接收用于 ECC 编码的数据的 ECC 信息，其指示 ECC 编码的数据中的一个或多个错误。可进行关于 ECC 编码的数据是否包括单个错误的确定。基于该确定，可识别用于 ECC 编码的数据中的单个错误的错误位置，或者可生成标志以指示 ECC 编码的数据已超过一个错误。

[0085] 根据用于示例方法的某些示例，可基于所识别的错误位置来纠正单个错误，并且然后可将 ECC 编码的数据解码。

[0086] 在用于示例方法的某些示例中，ECC 可包括 RS 码或二进制 BCH 码中的一个。

[0087] 根据用于示例方法的某些示例，识别与所识别错误位置相关联的错误值可基于包

括 RS 码的 ECC。

[0088] 在用于示例方法的某些示例中,确定 ECC 编码的数据具有一个或多个错误可基于与 RS 码或二进制 BCH 码中的一个相关联的伴随式检查。

[0089] 根据用于示例方法的某些示例,响应于生成的标志而识别用于包括在 ECC 编码的数据中的超过一个错误的单独错误位置可包括基于单独识别的错误位置来纠正超过一个错误且然后将 ECC 编码的数据解码。对于这些示例而言,ECC 可包括 RS 码或二进制 BCH 码中的一个。并且,如果是 RS 码,则识别与所识别错误位置相关联的错误值。并且,对于这些示例而言,确定 ECC 编码的数据具有一个或多个错误可基于与 RS 码或二进制 BCH 码中的一个相关联的伴随式检查。

[0090] 在用于示例方法的某些示例中,响应于生成的标志而识别用于包括在 ECC 编码的数据中的超过一个错误的单独错误位置可包括基于单独识别的错误位置来纠正超过一个错误且将 ECC 编码的数据解码。对于这些示例而言,ECC 可包括一个或 RS 码或二进制 BCH 码。并且,用于超过一个错误的单独错误位置可包括实现 BMA 和 Chien 搜索以单独地识别错误值和位置。并且,对于这些示例而言,识别与给定识别错误位置相关联的错误值可基于包括 RS 码的 ECC。

[0091] 根据用于示例方法的某些示例,识别用于单个错误的错误值和给定识别错误位置可基于实现包括 $e = \frac{s_n^2}{s_{zn}}$ 和 $j = \log \frac{s_{n+1}}{s_n}$ 的算法,其中 e 是错误值、j 是给定的识别错误位置,S 表示包括在接收到的 ECC 信息中的部分伴随式且 n 表示任何正整数。

[0092] 在用于示例方法的某些示例中,可使用二进制 BCH 码对已编码的 ECC 编码的数据进行编码。识别用于单个错误的错误位置可基于实现包括 $j = \log S_i$ 的算法,其中, j 是给定的识别错误位置且 S 表示包括在接收到的 ECC 信息中的部分伴随式。

[0093] 在用于示例方法的某些示例中,可使 ECC 编码的数据与存储到存储器设备、存储到存储介质、无线通信或 2 维条形码读取器中的一个相关联。

[0094] 根据用于示例方法的某些示例,存储器设备可包括非易失性存储器,诸如 PCM、PCMS、闪速存储器、铁电存储器、SONOS 存储器、聚合物存储器、纳米线、FeTRAM、FeRAM 或 EEPROM。

[0095] 根据某些示例,至少一个机器可读介质包括多个指令,其响应于在系统上被执行而使得系统执行如上所述的示例方法。

[0096] 根据某些示例,示例设备可包括处理器电路单错部件,其被布置成用于由处理器电路执行以接收用于 ECC 编码的数据的 ECC 信息,其指示 ECC 编码的数据中的一个或多个错误。还可将单错部件布置成确定编码的数据是否包括单个错误,当 ECC 编码的数据被确定为具有单个错误时,识别 ECC 编码的数据中的错误的位置,或者生成将指示 ECC 编码的数据具有多个错误的标志。示例设备还可包括多错部件,其被布置成用于由处理器电路执行以接收指示一个或多个错误的用于 ECC 编码的数据的 ECC 信息,并单独地识别用于每个错误的错误位置。示例设备还可包括纠正器部件,其被布置成用于由处理器电路执行以接收用于单个错误的所识别位置或用于多个错误的单独识别错误位置,并纠正 ECC 编码的数据中的一个或多个错误。

[0097] 在某些示例中,示例设备还可包括缓冲器部件,其被布置成用于由处理器电路执

行以保持能够临时地存储 ECC 编码的数据以便由纠正器部件纠正的存储器。

[0098] 在某些示例中,示例设备还可包括检错部件,其被布置成用于由处理器电路执行以确定 ECC 编码的数据具有一个或多个错误,并生成将指示 ECC 编码的数据中的一个或多个错误的 ECC 信息。

[0099] 根据用于示例设备的某些示例,ECC 可包括 RS 码或二进制 BCH 码的一个。

[0100] 在用于示例设备的某些示例中,多错部件能够实现 BMA 和 Chien 搜索以单独地识别错误位置。

[0101] 根据用于示例设备的某些示例,单错部件和多错部件被配置成基于使用 RS 码编码的 ECC 编码的数据来识别与给定的识别错误位置相关联的错误值。

[0102] 在用于示例设备的某些示例中,可将单错部件配置成基于实现包括 $e = \frac{s_n^2}{s_{2n}}$ 和

$j = \log \frac{s_{n+1}}{s_n}$ 的算法来识别用于单个错误的错误值和给定识别错误位置,其中,e 是错误值、

j 是给定识别错误位置,S 表示包括在接收到的 ECC 信息中的部分伴随式且 n 表示任何正整数。

[0103] 根据用于示例设备的某些示例,可使用二进制 BCH 码对 ECC 编码的数据进行编码。对于这些示例而言,可将单错部件配置成基于实现包括 $j = \log s_1$ 的算法来识别 ECC 编码的数据中的错误的位置,其中,j 是给定识别错误位置且 S 表示包括在接收到的 ECC 信息中的部分伴随式。

[0104] 在用于示例设备的某些示例中,可使 ECC 编码的数据与存储到存储器设备、存储到存储介质、无线通信或 2 维条形码读取器中的一个相关联。

[0105] 根据用于示例设备的某些示例,存储器设备将包括非易失性存储器,包括 PCM、PCMS、闪速存储器、铁电存储器、SONOS 存储器、聚合物存储器、纳米线、FeTRAM、FeRAM 或 EEPROM。

[0106] 在用于示例设备的某些示例中,存储器设备可以是用于计算设备的二级存储器(2LM)系统。对于这些示例而言,2LM 系统还可包括易失性存储器。

[0107] 应强调的是提供本公开的摘要是为了服从 37 C. F. R. 第 1. 72 节(b),要求将允许读者快速地确定本技术公开的性质的摘要。其是在这样的条件下提交的,即其将不用来解释或限制权利要求的范围或意义。另外,在前述详细描述中,可以看到出于使本公开流畅的目的而在单个示例中将各种特征分组在一起。不应将本公开方法解释为反映要求保护的示例要求比在每个权利要求中明确地叙述的更多的特征的意图。相反地,如以下权利要求所反映的,发明主题在于少于单个公开示例的所有特征中。因此,以下权利要求因此被结合到详细描述中,每个权利要求独立地作为单独示例存在。在所附权利要求中,术语“包括”和“其中”分别地被用作各术语“包含”和“在其中”的口语英语等价物。此外,术语“第一”、“第二”、“第三”等仅仅被用作标签,并且并不意图对其对象施加数值要求。

[0108] 虽然已用结构特征和 / 或方法动作特定的语言描述了主题,但应理解的是在所附权利要求中定义的主题不一定局限于上述特定特征或动作。相反地,上文描述的特定特征和动作是作为实现权利要求的示例性形式而公开的。

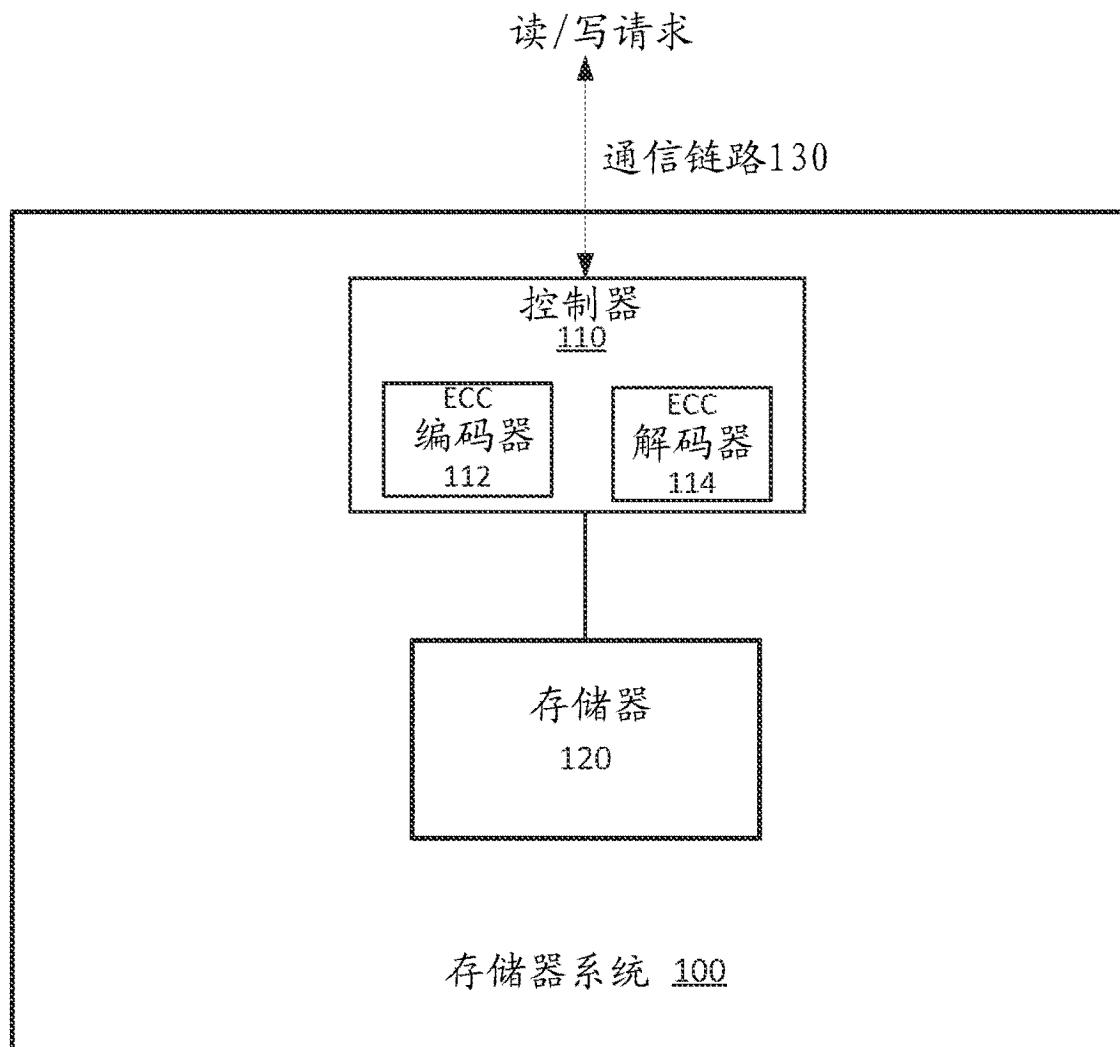


图 1

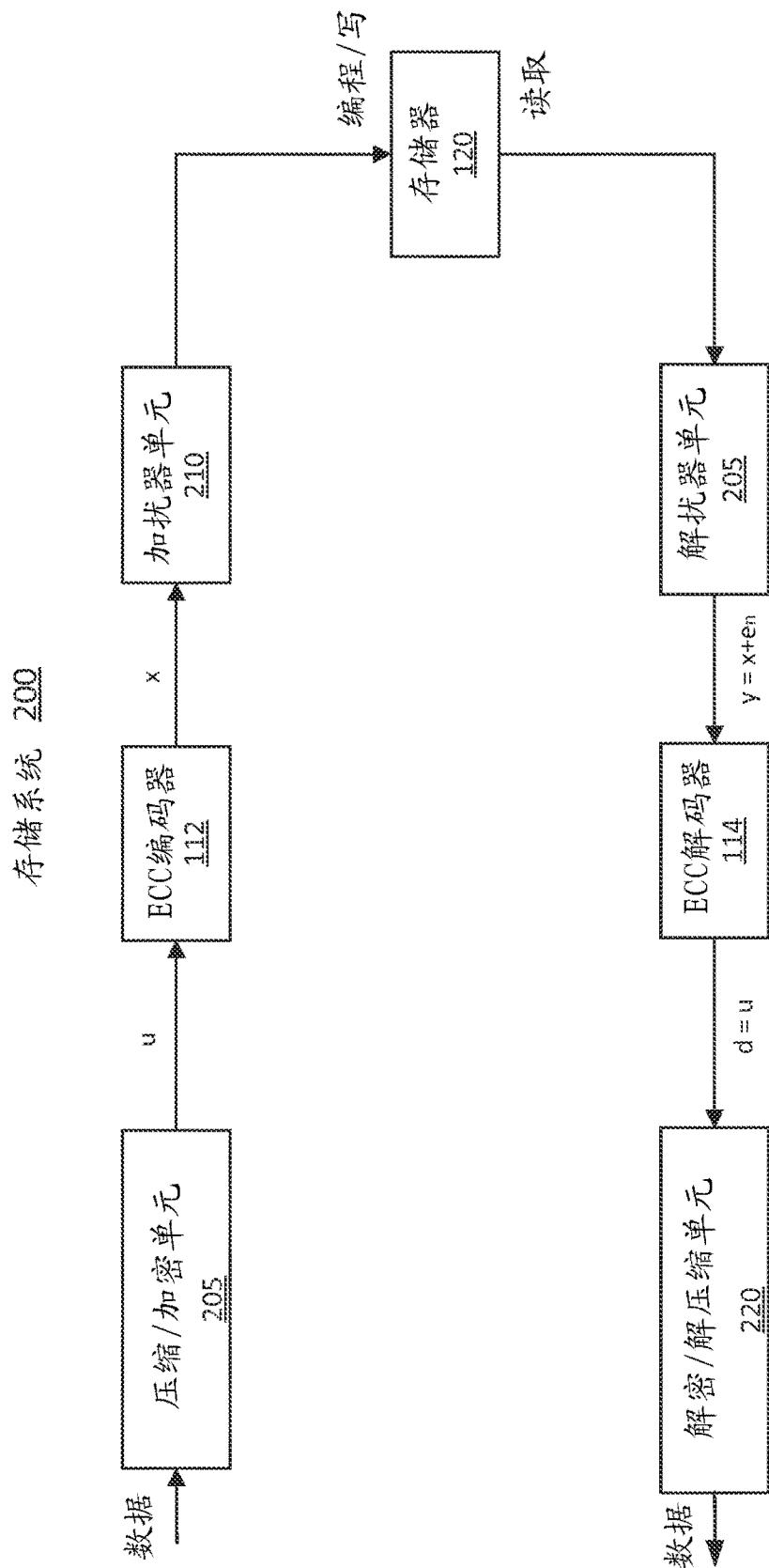


图 2

ECC 系统 300

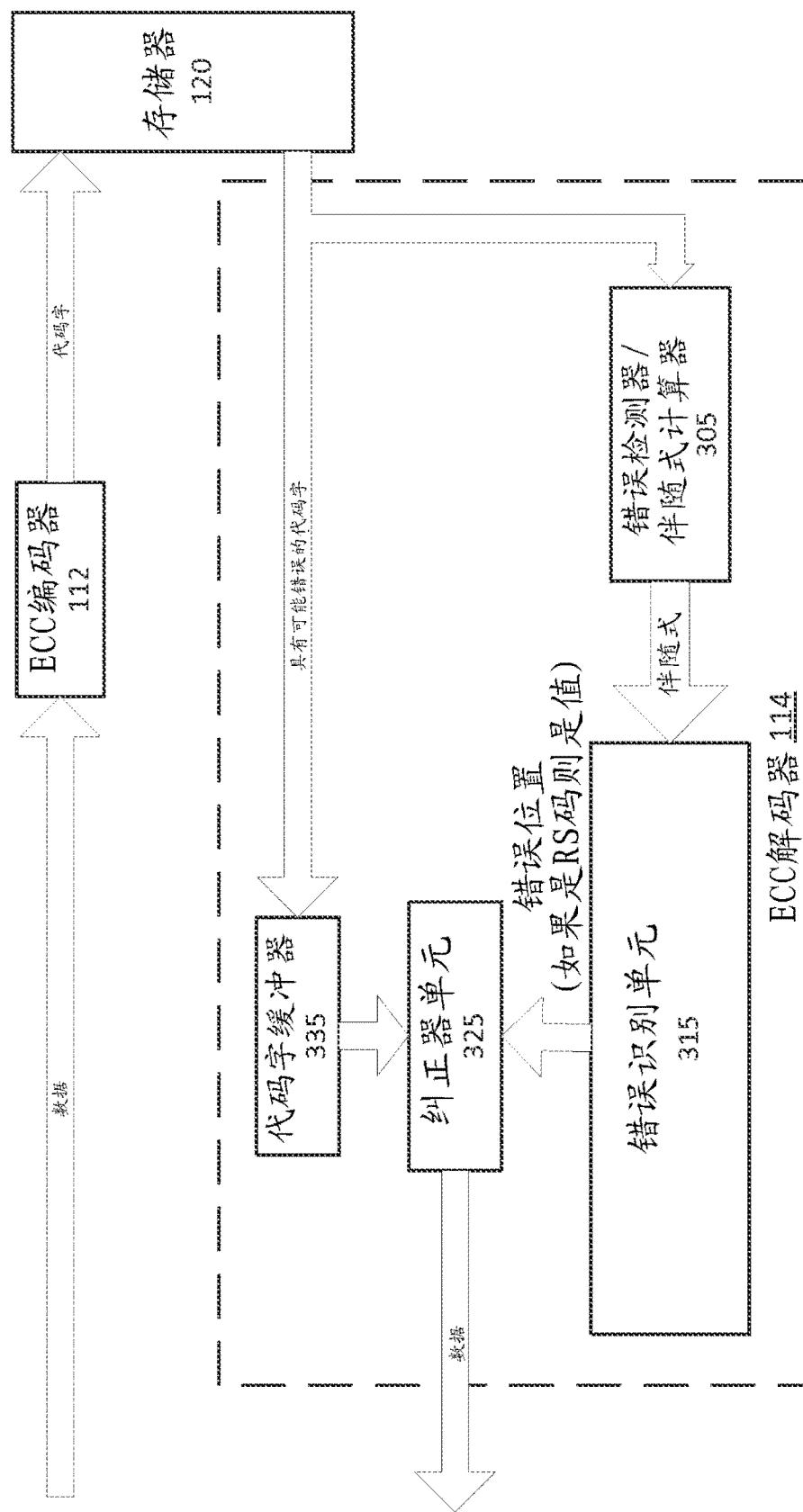


图 3

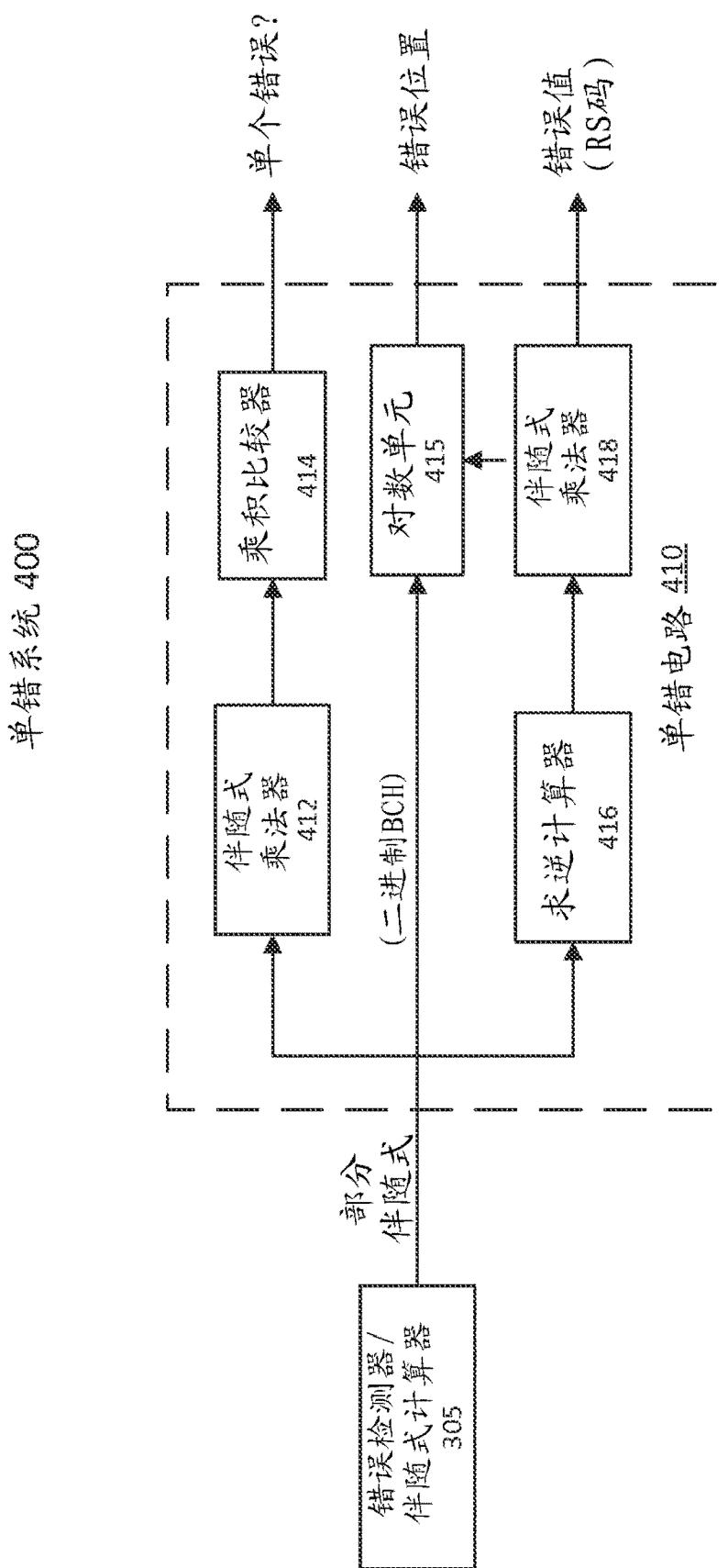
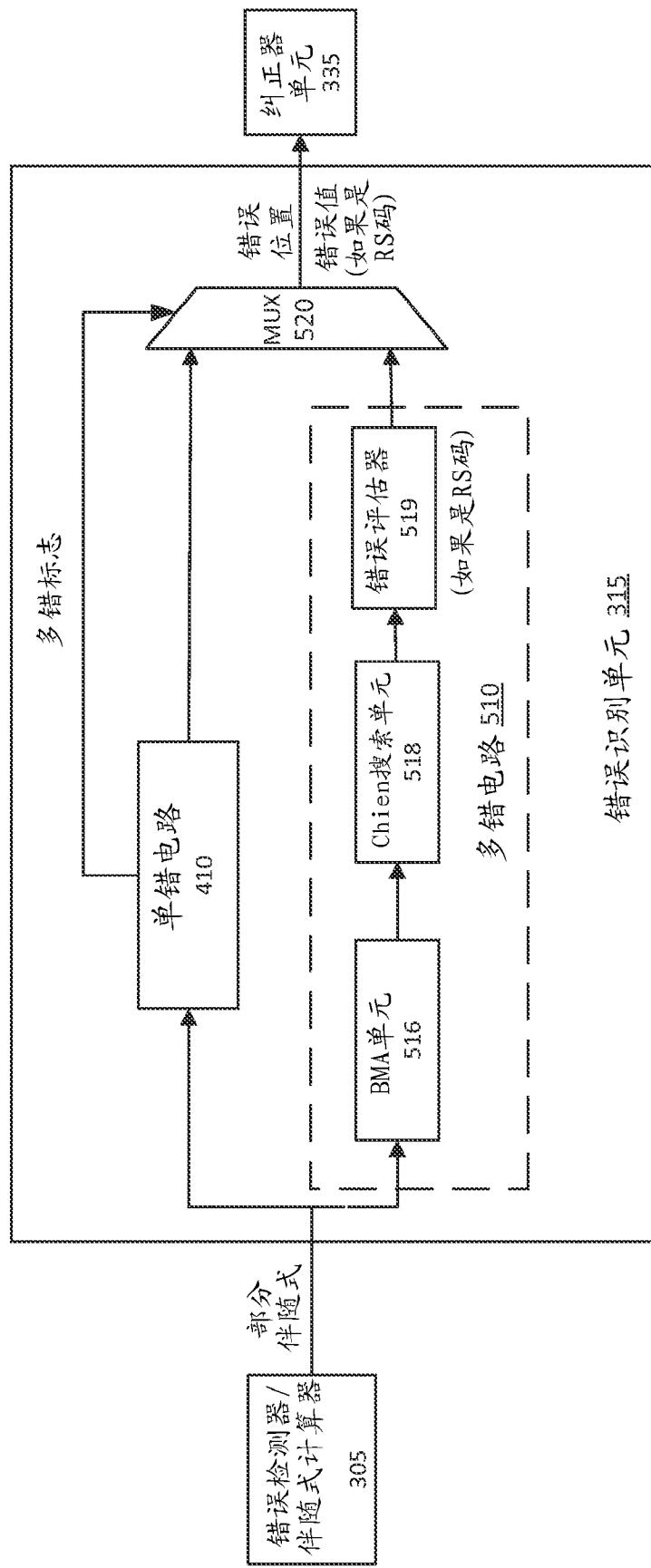


图 4

单 / 多错系统 500



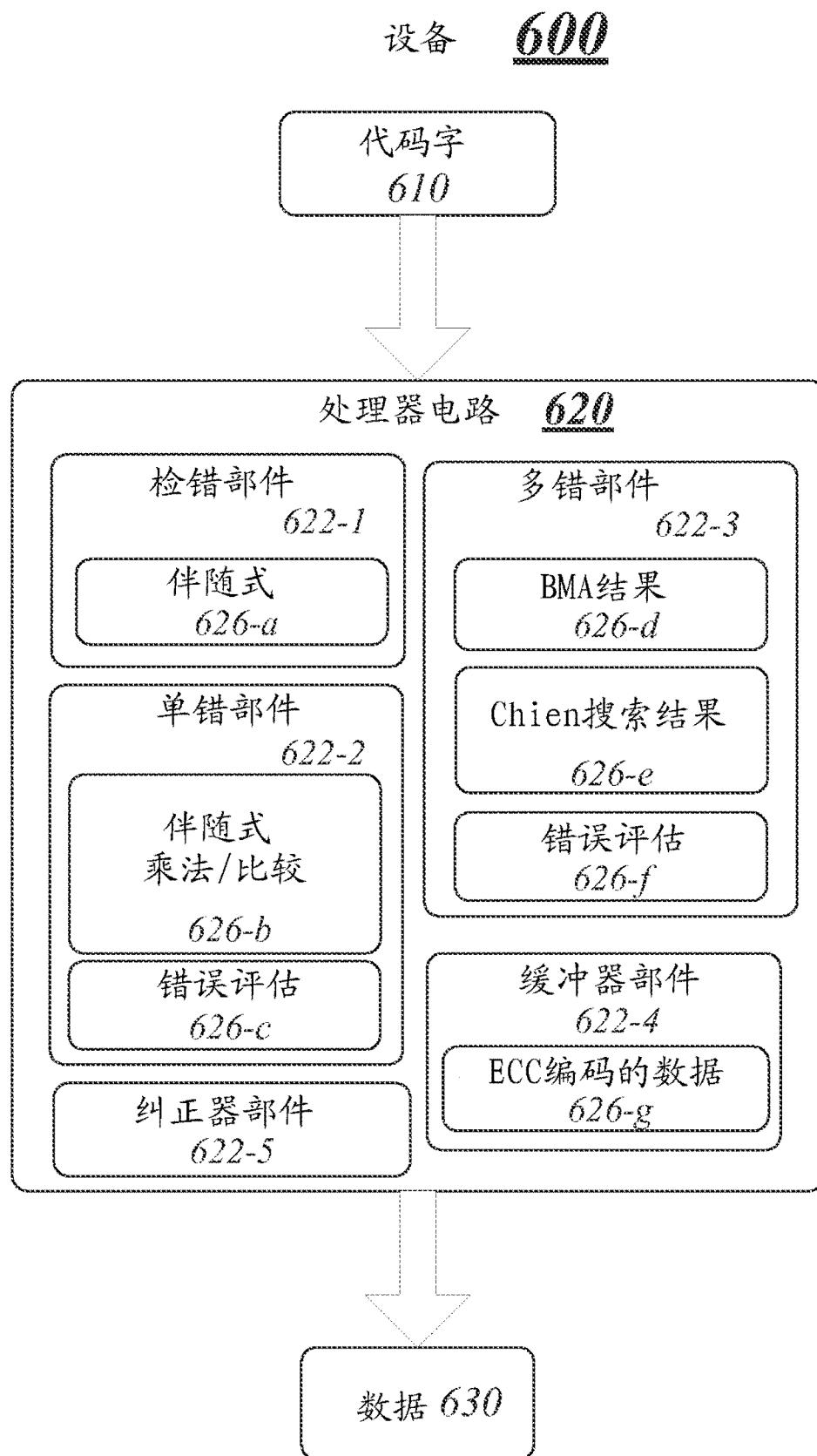


图 6

700

接收用于ECC编码的数据的ECC信息，
指示ECC编码的数据中的一个或多个错误

702

确定编码的数据是否包括单个错误

704

基于该确定，识别用于ECC编码的数据中的
单个错误的错误位置，或者生成标志以指示
ECC编码的数据具有超过一个错误

706

基于所识别的错误位置来纠正单个
错误并将ECC编码的数据解码

708

响应于生成的标志而识别用于ECC
编码的数据中的超过一个错误的单独错误位置

710

基于单独识别错误位置来纠正超过
一个错误并将ECC编码的数据解码

712

图 7

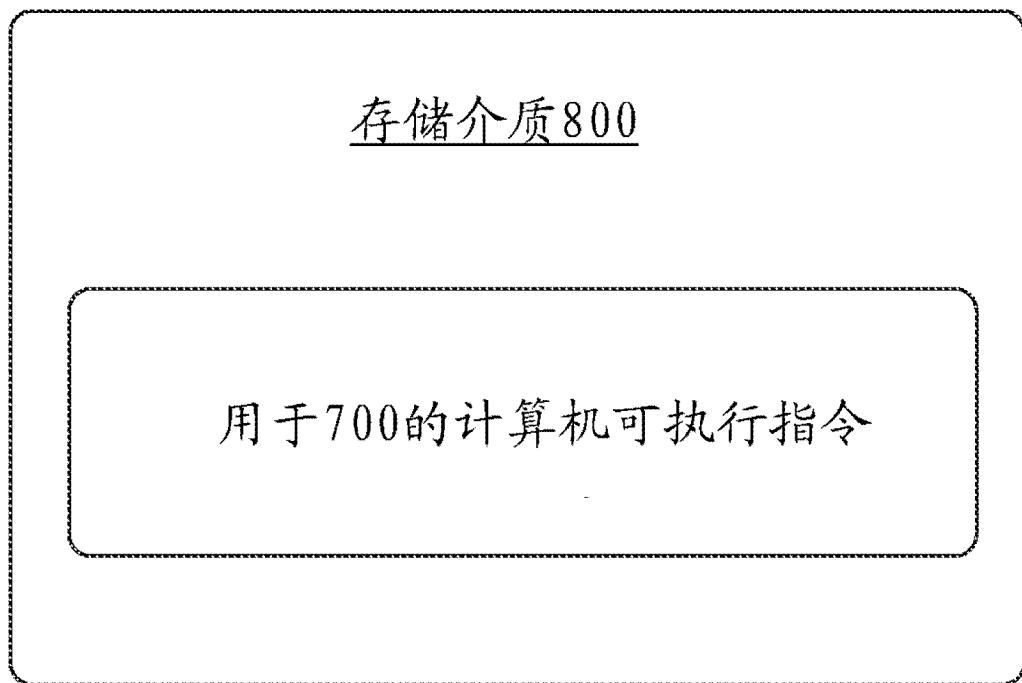


图 8

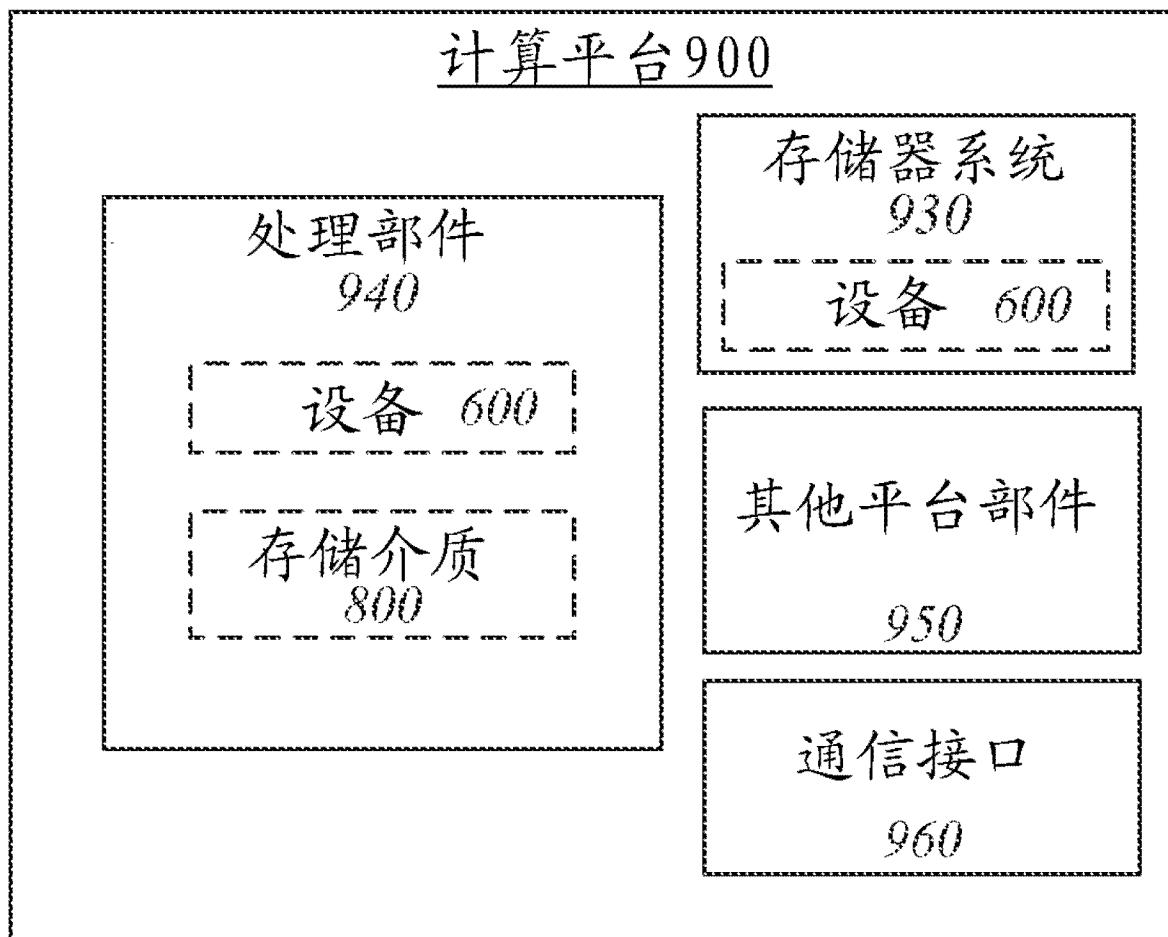


图 9