



(12)发明专利申请

(10)申请公布号 CN 107005250 A

(43)申请公布日 2017.08.01

(21)申请号 201580063814.2

(72)发明人 S·森戈库

(22)申请日 2015.11.24

(74)专利代理机构 上海专利商标事务所有限公司 31100

(30)优先权数据

代理人 陈小刚 陈炜

62/084,998 2014.11.26 US

62/216,692 2015.09.10 US

62/236,522 2015.10.02 US

14/949,290 2015.11.23 US

(85)PCT国际申请进入国家阶段日

2017.05.24

(86)PCT国际申请的申请数据

PCT/US2015/062511 2015.11.24

(87)PCT国际申请的公布数据

W02016/086045 EN 2016.06.02

(71)申请人 高通股份有限公司

地址 美国加利福尼亚州

(51)Int.Cl.

H03M 13/00(2006.01)

H03M 13/03(2006.01)

H04L 1/00(2006.01)

H04L 25/02(2006.01)

H04L 25/14(2006.01)

H04L 25/493(2006.01)

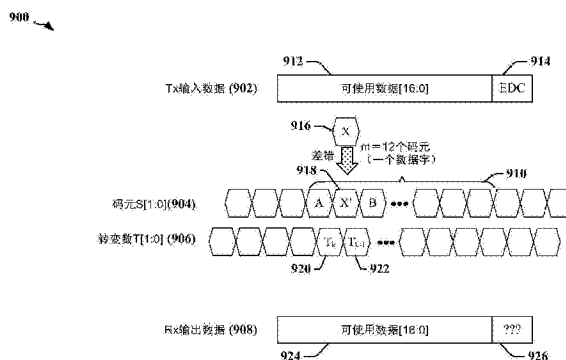
权利要求书4页 说明书20页 附图25页

(54)发明名称

多线码元转变时钟码元纠错

(57)摘要

公开了用于检测多线接口上的传输中的差错的装置、系统以及方法。一种用于纠正多线转变编码接口中的传输差错的方法可包括基于接收到的多个比特中的检错码(EDC)的值来确定码元序列中是否存在码元差错,生成该码元序列的一个或多个置换,其中每一置换包括与该码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元。在该一个或多个置换中的一个置换产生与预期EDC值相匹配的经解码EDC值时,该置换可被标识为包括经纠正码元序列。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。



1. 一种在接收机处纠正多线接口中的传输差错的方法,包括:

接收表示多个连接器的信令状态的码元序列;

解码所述码元序列,包括:

将所述码元序列转换成转变数,所述转变数的每一数位表示在所述多个连接器上传送的两个连贯码元之间的转变;以及

将所述转变数转换成接收到的多个比特;

基于所述接收到的多个比特中的检错码(EDC)的值来确定所述码元序列中是否存在码元差错;以及

在确定所述码元序列中存在码元差错时:

生成所述码元序列的一个或多个置换,每一置换包括与所述码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元;

解码所述一个或多个置换;以及

在第一置换产生与预期EDC值相匹配的经解码EDC值时将所述第一置换标识为经纠正码元序列,

其中所述预期EDC值对应于在所述多线接口上传送以使得能够在所述接收机处检测多达两个码元差错的EDC的预定义值。

2. 如权利要求1所述的方法,其特征在于,进一步包括:

使所述第一置换替代所述码元序列或使从所述第一置换取码的经纠正数据替代从所述码元序列解码的差错数据。

3. 如权利要求1所述的方法,其特征在于,进一步包括:

在确定所述码元序列中存在码元差错时,并发地生成所述码元序列的置换集合,其中所述置换集合的所有成员包括所述码元序列中的同一码元的经修改版本;以及

并发地解码所述置换集合的所有成员。

4. 如权利要求1所述的方法,其特征在于,进一步包括:

在确定所述码元序列中存在码元差错时,并发地生成和解码所述码元序列的所有可能置换,所述置换包括相对于所述码元序列中的对应码元来修改的单个码元;以及

并发地将从所述码元序列的每一可能置换解码出的EDC值与所述预期EDC值相比较。

5. 如权利要求1所述的方法,其特征在于,进一步包括:

在确定所述码元序列中存在码元差错时,并发地生成所述码元序列的置换集合,其中所述置换集合的成员包括所述码元序列中的不同码元的经修改版本;以及

并发地解码所述置换集合的所有成员。

6. 如权利要求1所述的方法,其特征在于,在所述多线接口上传送的所述EDC的预定义值具有固定长度,所述固定长度是基于为对所述多个连接器上的数据传输进行编码所定义的每码元状态总数来确定的。

7. 如权利要求6所述的方法,其特征在于,所述EDC是所述多个比特中的预定义数目的最低有效位,并且其中在所述多线接口上传送的所述EDC的预定义值是基于被用来编码所述多个比特的码元总数来确定的。

8. 如权利要求1所述的方法,其特征在于,所述码元差错使得所述EDC的经解码版本具有与所述预定义值不同的值。

9. 如权利要求1所述的方法,其特征在于,所述多个连接器包括数个(N个)单端连接器,并且能供用于编码数据传输的每码元状态总数是 $2^N - x$,其中x至少是1。

10. 如权利要求1所述的方法,其特征在于,所述多个连接器包括数个(N个)单端连接器,并且能供用于编码数据传输的每码元状态总数是 $N! - x$,其中x至少是1,并且其中所述码元序列的所述一个或多个置换包括每码元r个置换,其中r表示每码元的可能转变的数目。

11. 如权利要求1所述的方法,其特征在于,所述多个连接器包括数个(N个)单端连接器,并且能供用于编码数据传输的每码元状态的总数是 $N! - x$,其中x至少是1,并且其中所述码元序列的所述一个或多个置换包括每码元r-1个置换,其中r表示每码元的可能转变的数目。

12. 如权利要求1所述的方法,其特征在于,解码所述一个或多个置换包括:

抑制解码包括具有相同值的两个连贯码元的置换。

13. 一种设备,包括:

用于接收表示多个连接器的信令状态的码元序列的装置;

用于解码所述码元序列的装置,且其被配置成:

将所述码元序列转换成转变数,所述转变数的每一数位表示在所述多个连接器上传送的两个连贯码元之间的转变;以及

将所述转变数转换成接收到的多个比特;

用于基于所述接收到的多个比特中的检错码(EDC)的值来确定所述码元序列中是否存在码元差错的装置;以及

用于纠正码元差错的装置,其中当确定所述码元序列中存在码元差错时,所述用于纠正码元差错的装置被配置成:

生成所述码元序列的一个或多个置换,每一置换包括与所述码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元;

解码所述一个或多个置换;以及

在一置换产生与预期EDC值相匹配的经解码EDC值时将所述置换标识为经纠正码元序列,

其中所述预期EDC值对应于在所述多个连接器上传送以使得能够在所述接收机处检测多达两个码元差错的EDC的预定义值。

14. 如权利要求13所述的设备,其特征在于,所述用于纠正码元差错的装置被配置成:

在确定所述码元序列中存在码元差错时,并发地生成所述码元序列的置换集合,其中所述置换集合的所有成员包括所述码元序列中的同一码元的经修改版本;以及

并发地解码所述置换集合的所有成员。

15. 如权利要求13所述的设备,其特征在于,所述用于纠正码元差错的装置被配置成:

在确定所述码元序列中存在码元差错时,并发地生成和解码所述码元序列的所有可能置换,所述置换包括相对于所述码元序列中的对应码元来修改的单个码元;以及

并发地将从所述码元序列的每一可能置换解码出的EDC值与所述预期EDC值相比较。

16. 如权利要求13所述的设备,其特征在于,所述用于纠正码元差错的装置被配置成:

在确定所述码元序列中存在码元差错时,并发地生成所述码元序列的置换集合,其中所述置换集合的成员包括所述码元序列中的不同码元的经修改版本;以及

并发地解码所述置换集合的所有成员。

17. 一种装置, 包括:

通信收发机, 其被配置成提供表示多个连接器的信令状态的码元序列;

解码器电路, 其被配置成将表示所述码元序列中的连贯码元之间的转变的转变数转换成多个比特;

检错电路, 其被配置成基于所述多个比特中的检错码 (EDC) 的值来确定所述码元序列中是否存在码元差错; 以及

纠错电路, 其对所述码元序列中存在码元差错的确定作出响应且被配置成:

生成所述码元序列的一个或多个置换, 每一置换包括与所述码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元;

解码所述一个或多个置换; 以及

在第一置换产生与预期EDC值相匹配的经解码EDC值时将所述第一置换标识为经纠正码元序列,

其中所述预期EDC值对应于在所述多个连接器上传送以使得能够检测多达两个码元差错的EDC的预定义值, 以及

其中所述码元差错使得所述EDC的经解码版本具有与所述预定义值不同的值。

18. 如权利要求17所述的装置, 其特征在于, 所述纠错电路被配置成:

使所述第一置换替代所述码元序列或使从所述第一置换解码的经纠正数据替代从所述码元序列解码的差错数据; 以及

针对所述码元序列中的一个码元的每一可能值来生成置换。

19. 如权利要求17所述的装置, 其特征在于, 所述纠错电路被配置成:

在确定所述码元序列中存在码元差错时, 并发地生成所述码元序列的置换集合, 其中所述置换集合的所有成员包括所述码元序列中的同一码元的经修改版本; 以及

并发地解码所述置换集合的所有成员。

20. 如权利要求17所述的装置, 其特征在于, 所述纠错电路被配置成:

在确定所述码元序列中存在码元差错时, 并发地生成和解码所述码元序列的所有可能置换, 所述置换包括相对于所述码元序列中的对应码元来修改的单个码元; 以及

通过并发地将所述码元序列的每一可能置换解码出的EDC值与所述预期EDC值相比较来标识产生与所述预期EDC值相匹配的经解码EDC值的置换。

21. 如权利要求17所述的装置, 其特征在于, 所述纠错电路被配置成:

在确定所述码元序列中存在码元差错时, 并发地生成所述码元序列的置换集合, 其中所述置换集合的成员包括所述码元序列中的不同码元的经修改版本; 以及

并发地解码所述置换集合的所有成员。

22. 如权利要求17所述的装置, 其特征在于, 在所述多个连接器上传送的所述EDC的预定义值具有固定长度, 所述固定长度是基于为对所述多个连接器上的数据传输进行编码所定义的每码元状态总数来确定的。

23. 如权利要求17所述的装置, 其特征在于, 所述EDC是所述多个比特中的预定义数目的最低有效位, 并且其中在所述多个连接器上传送的所述EDC的预定义值是基于被用来编码所述多个比特的码元总数来确定的。

24. 如权利要求17所述的装置,其特征在于,所述纠错电路被配置成通过以下操作来生成所述码元序列的一个或多个置换:

针对所述码元序列中的一个码元的值的每一可能变化来生成一置换。

25. 如权利要求17所述的装置,其特征在于,所述纠错电路被配置成通过以下操作来生成所述码元序列的一个或多个置换:

生成与所述码元序列中的每一码元相对应的置换,其中在每一置换中将码元修改相同的值。

26. 如权利要求17所述的装置,其特征在于,所述多个连接器包括数个(N个)单端连接器,并且能供用于编码数据传输的每码元状态总数是 2^N-x ,其中x至少是1,或者能供用于编码数据传输的每码元状态的总数是 $N!-x$,其中x至少是1。

27. 一种其上存储有能由处理器执行的代码的处理器可读存储介质,所述代码包括用于以下操作的指令:

接收表示多个连接器的信令状态的码元序列;

解码所述码元序列,包括:

将所述码元序列转换成转变数,所述转变数的每一数位表示在所述多个连接器上传送的两个连贯码元之间的转变;以及

将所述转变数转换成接收到的多个比特;

基于所述接收到的多个比特中的检错码(EDC)的值来确定所述码元序列中是否存在码元差错;以及

在确定所述码元序列中存在码元差错时:

生成所述码元序列的一个或多个置换,每一置换包括与所述码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元;

解码所述一个或多个置换;以及

在一置换产生与预期EDC值相匹配的经解码EDC值时将所述置换标识为经纠正码元序列,

其中所述预期EDC值对应于在所述多个连接器上传送以使得能够检测多达两个码元差错的EDC的预定义值。

28. 如权利要求27所述的存储介质,其特征在于,所述代码包括用于以下操作的指令:

在确定所述码元序列中存在码元差错时,并发地生成所述码元序列的置换集合,其中所述置换集合的所有成员包括所述码元序列中的同一码元的经修改版本;以及

并发地解码所述置换集合的所有成员。

29. 如权利要求27所述的存储介质,其特征在于,所述代码包括用于以下操作的指令:

在确定所述码元序列中存在码元差错时,并发地生成和解码所述码元序列的所有可能置换,所述置换包括相对于所述码元序列中的对应码元来修改的单个码元;以及

并发地将从所述码元序列的每一可能置换解码出的EDC值与所述预期EDC值相比较。

30. 如权利要求27所述的存储介质,其特征在于,所述代码包括用于以下操作的指令:

在确定所述码元序列中存在码元差错时,并发地生成所述码元序列的置换集合,其中所述置换集合的成员包括所述码元序列中的不同码元的经修改版本;以及

并发地解码所述置换集合的所有成员。

多线码元转变时钟码元纠错

[0001] 相关申请的交叉引用

[0002] 本申请要求于2015年10月2日向美国专利商标局提交的美国临时申请No.62/236,522、于2015年9月10日向美国专利商标局提交的美国临时申请No.62/216,692、于2014年11月26日向美国专利商标局提交的美国临时申请No.62/084,998以及于2015年11月23日向美国专利商标局提交的美国非临时申请No.14/949,290的优先权和权益,它们的全部内容通过援引纳入于此。

[0003] 背景

[0004] 领域

[0005] 本公开涉及实现数据通信接口上的高效操作,且更具体地涉及促进采用码元转变时钟转码的数据通信接口上的纠错。

背景技术

[0006] 数据通信接口可以采用码元转变时钟转码来将时钟信息嵌入在编码将通过具有多条信号导线的接口传送的数据的码元序列中,从而消除对专用时钟信号线的需求。

[0007] 在多信号数据传递(多线差分信令,诸如N阶乘(N!)低电压差分信令(LVDS))的某些示例中,可执行转码(例如,一种编码类型到另一编码类型的数字-数字数据转换)以便通过在每一码元循环引起码元转变而不是在单独的数据通道(差分传输路径)中发送时钟信息的方式来嵌入码元时钟信息。通过此类转码来嵌入时钟信息还可最小化时钟与数据信号之间的偏斜并消除用锁相环(PLL)来从数据信号中恢复时钟信息的需求。在一个示例中,根据常规集成电路间(I2C)协议或相机控制接口(CCI)协议来操作的两线串行总线可被适配成提供CCI扩展(CCIe)总线,它使用码元转变时钟转码。CCIe总线支持比I2C总线或CCI总线更高的比特率。

[0008] 检错和纠错在采用转变编码的数据传递接口中可能是有问题的,因为信令状态差错与从数据传递接口解码的数据中的差错之间通常没有直接关联。数据比特与信令状态之间无关联性可以致使常规检错技术在被应用于转变编码接口时无效。

[0009] 将期望在使用码元转变时钟转码来进行通信的各设备之间的传输中提供可靠的检错和纠错。

[0010] 概述

[0011] 本公开的各方面涉及一种改进能可靠地检测影响在多线转变编码接口上传送的两个或更多个码元的差错并能纠正编码单个数据字的每一码元序列的单码元差错的设备的功能性和操作效率的方法、计算机程序、以及装置。

[0012] 根据某些方面,一种在接收机处纠正多线接口中的传输差错的方法包括:接收表示多个连接器的信令状态的码元序列;以及解码该码元序列。可通过将该码元序列转换成转变数并将转变数转换成接收到的多个比特来解码该码元序列。转变数的每一位可表示在多个连接器上传送的两个连贯码元之间的转变。该纠正传输差错的方法可包括基于接收到的多个比特中的检错码(EDC)的值来确定该码元序列中是否存在码元差错。在确定该码

元序列中存在码元差错时,可生成该码元序列的一个或多个置换,其中每一置换包括与该码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元。该一个或多个置换可被解码,并且在一个置换产生与预期EDC值相匹配的经解码EDC值时,该置换可被标识为经纠正码元序列。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。

[0013] 根据某些方面,一种设备包括:用于接收表示多个连接器的信令状态的码元序列的装置;以及用于解码该码元序列的装置。用于解码该码元序列的装置可被配置成将码元序列转换成转变数(其中转变数的每一位表示在多个连接器上传送的两个连贯码元之间的转变)以及将转变数转换成接收到的多个比特。该设备可包括:用于基于所接收到的多个比特中的EDC的值来确定该码元序列中是否存在码元差错的装置;以及用于在确定该码元序列中存在码元差错时纠正码元差错的装置。用于纠正码元差错的装置可被配置成:生成该码元序列的一个或多个置换,其中每一置换包括与该码元序列中的对应码元不同且与其它置换中的对应码元不同的一个码元;解码该一个或多个置换;以及在一置换产生与预期EDC值相匹配的经解码EDC值时将该置换标识为经纠正码元序列。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。

[0014] 根据某些方面,一种装置包括:通信收发机,其被配置成提供表示多个连接器的信令状态的码元序列;解码器电路,其被配置成将表示该码元序列中的连贯码元之间的转变的转变数转换成多个比特;检错电路,其被配置成基于接收到的多个比特中的EDC的值来确定该码元序列中是否存在码元差错;以及纠错电路。纠错电路可对该码元序列中存在码元差错的确定作出响应且可被配置成:生成该码元序列的一个或多个置换,其中每一置换包括与该码元序列中的对应码元不同且与其它置换中的对应码元不同的一个码元;解码该一个或多个置换;以及在一置换产生与预期EDC值相匹配的经解码EDC值时将该置换标识为经纠正码元序列。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。

[0015] 根据某些方面,一种处理器可读存储介质上可存储有指令。这些指令可由处理器执行。这些指令可包括用于以下操作的指令:接收表示多个连接器的信令状态的码元序列;以及解码该码元序列。在一个示例中,这些指令可包括用于以下操作的指令:将该码元序列转换成转变数,该转变数的每一位表示在多个连接器上传送的两个连贯码元之间的转变;以及将该转变数转换成接收到的多个比特。这些指令可包括用于以下操作的指令:基于所接收到的多个比特中的EDC的值来确定该码元序列中是否存在码元差错;以及在确定该码元序列中存在码元差错时,生成该码元序列的一个或多个置换,其中每一置换包括与该码元序列中的对应码元不同且与其它置换中的对应码元不同的一个码元,解码该一个或多个置换,以及在一置换产生与预期EDC值相匹配的经解码EDC值时将该置换标识为经纠正码元序列。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。

[0016] 附图

[0017] 在结合附图理解下面阐述的详细描述时,各种特征、本质、和优点会变得明显,在附图中,相像的附图标记贯穿始终作相应标识。

[0018] 图1描绘了在各IC设备之间采用数据链路的装置,该数据链路根据多个可用标准

之一来选择性地操作。

[0019] 图2解说了在IC设备之间采用数据链路的装置的系统架构。

[0020] 图3解说了在两个设备之间提供的N!接口的示例。

[0021] 图4解说了可根据本文公开的某些方面来适配的发射机和接收机。

[0022] 图5解说了可被用来控制转变数与顺序码元之间的转换的编码方案。

[0023] 图6解说了转变编码接口的一个示例中的码元与转变数之间的关系。

[0024] 图7解说了在3!接口中码元边界处可能的转变数到码元编码。

[0025] 图8解说了3!接口中的转变数与码元之间的数学关系。

[0026] 图9解说了其中通过多线通信接口传送的码元序列受单码元差错的影响的示例。

[0027] 图10是解说表征通过多线通信接口传送的码元序列中的单码元差错的数学关系的示意图。

[0028] 图11通过表格示出 r^n 的值,其中n在范围0-15中并且在 $r=3$ 和 $r=5$ 时。

[0029] 图12通过表格示出与码元序列中的单码元差错相对应的差错系数。

[0030] 图13解说了差错系数中的最长非零LSB部分。

[0031] 图14解说了其中单码元差错导致单个转变数中的差错的情况。

[0032] 图15解说了影响通过多线通信接口传送的码元序列中的两个码元的信令差错的第一示例。

[0033] 图16解说了影响通过多线通信接口传送的两个连贯码元的信令差错的第二示例。

[0034] 图17解说了根据本文公开的某些方面的在EDC中提供的用于检测两个码元差错的比特数。

[0035] 图18解说了根据本文公开的某些方面的可被用来纠正单码元差错的电路的操作。

[0036] 图19解说了根据本文公开的某些方面的可被用来纠正码元差错的解码器的第一配置。

[0037] 图20解说了根据本文公开的某些方面的可被用来纠正码元差错的解码器的第二配置。

[0038] 图21解说了根据本文公开的某些方面的可被使用来降低解码开销的用于纠正码元差错的解码器的某些方面。

[0039] 图22是解说采用可根据本文所公开的某些方面来适配的处理系统的装置的示例的框图。

[0040] 图23是根据本文所公开的某些方面的用于纠正码元差错的方法的流程图。

[0041] 图24是解说根据本文公开的某些方面的用于解码各码元的方法的示例的流程图。

[0042] 图25是解说采用根据本文所公开的某些方面的用于在提供码元检错的接口中使用的装置的硬件实现的第二示例的示意图。

[0043] 详细描述

[0044] 在以下描述中,给出了具体细节以提供对诸实施例的透彻理解。然而,本领域普通技术人员将理解,没有这些具体细节也可实践这些实施例。例如,电路可能用框图示出以免使这些实施例混淆在不必要的细节中。在其他实例中,公知的电路、结构、和技术可能不被详细示出以免使这些实施例不明朗。

[0045] 概览

[0046] 某些数据传递接口采用转变编码(包括3相和N!多线LVDS接口)和多线单端接口(包括CCIE接口)。转变编码将时钟信息嵌入通过该接口传送的信令状态中。在某些实例中,数据被转码成转变数,其中每一转变数选择要在当前码元之后传送的下一码元。每一码元可以表示该接口的信令状态。例如,转变数可以表示用来在可在接口上传送的有序码元集合中的各码元之间进行选择的偏移。通过确保连贯码元彼此不同,该接口的信令状态的变化在每一码元边界处发生,从而提供被用来生成接收机处的接收时钟的信息。

[0047] 信令状态中将所传送的码元 S_1 改变成接收到的码元 Se_1 的差错可以使接收机产生与紧接在前的码元 S_2 与已改变码元 Se_1 之间的转变相关联的不正确转变数 T_1+e_1 。 T_1 表示 S_2 与正确地传送的码元 S_1 之间的差异,而 e_1 是信令差错所引入的偏移的值。第二不正确转变数 T_0+e_0 与已改变码元 Se_1 相关联,其中 T_0 表示正确地传送的码元 S_1 与下一码元 S_0 之间的差异,而 e_0 表示由信令差错引入的偏移的值。 e_1 和 e_0 的值没有直接对应于信令状态中的差错,并且数据比特差错与信令状态差错之间的无关联性可致使常规检错技术在被应用于转变编码接口时无效。

[0048] 根据本文公开的某些方面,通过提供检错常数(EDC)来在转变编码接口中启用可靠检错,EDC可以检测被用来传送数据字的码元序列中的两个码元差错。EDC可包括具有已知、固定值的预定义数目的比特。在一个示例中,EDC的值可具有零值,并且可被提供作为追加到要在多线接口上传送的每一字的预定义或预选择数目的最低有效位(LSB)。

[0049] 根据某些方面,每传送数据字的单码元差错可被纠正。在一个示例中,追加到数据字的EDC可以能够检测两个码元差错,并且这一能力可被用来实现有效地用于纠正单码元差错的纠错方案。在操作中,接收机可以监视经解码字中的EDC值并在例如EDC具有非预期值时确定至少一个码元差错已发生。在检测到差错时可尝试纠错。

[0050] 纠错方案可通过测试与不匹配发射机所插入的预定义值的EDC相关联的对应于收到数据字的收到码元序列中的每一码元的值的不同变化来操作。例如,码元序列的诸置换可以使用一个码元的替换值来解码,同时其他码元未修改。在经修改码元使得追加到所得的经解码数据字的EDC与在无错传输之后EDC的预期预定义值相匹配时,纠错可完成。因为EDC检测两个码元差错,所以纠错逻辑对正确地接收到的码元作出的任何修改都会将第二码元差错引入码元序列,这可通过比较EDC值来检测。

[0051] 采用转变编码的设备的示例

[0052] 图1描绘了采用IC设备之间的通信链路的装置100的一个示例。在一个示例中,装置100可包括无线通信设备,该无线通信设备通过RF收发机与无线电接入网(RAN)、核心接入网、因特网和/或另一网络通信。装置100可包括可操作地耦合至处理电路102的通信收发机106。处理电路102可包括一个或多个IC设备,诸如专用IC(ASIC) 108。ASIC 108可包括一个或多个处理设备、逻辑电路等等。处理电路102可包括和/或耦合至处理器可读存储(诸如存储器设备112),该处理器可读存储可维护可由处理电路102执行的指令和数据。处理电路102可由操作系统以及应用编程接口(API) 110层中的一者或多者来控制,该API 110层支持并允许执行驻留在存储介质(诸如无线设备的存储器设备112)中的软件模块。存储器设备112可包括只读存储器(ROM)或随机存取存储器(RAM)、电可擦除可编程ROM(EEPROM)、闪存卡、或可以在处理系统和计算平台中使用的任何存储器设备。处理电路102可包括或访问本地数据库114,该本地数据库114可维护用于配置和操作该装置100的操作参数和其他信息。

本地数据库114可使用数据库模块、闪存、磁介质、EEPROM、光学介质、磁带、软盘或硬盘等中的一者或多者来实现。处理电路也可以可操作地耦合至外部设备,诸如天线122、显示器124、操作者控件(诸如按钮128和按键板126以及其他组件)。

[0053] 图2是解说连接到通信链路220的装置200的某些方面的示意框图,其中该装置200可被实施在以下一者或多者中:无线移动设备、移动电话、移动计算系统、无线电话、笔记本计算机、平板计算设备、媒体播放器、游戏设备等。装置200可包括通过通信链路220交换数据和控制信息的多个IC设备202和230。通信链路220可被用于连接彼此紧邻地定位或者物理上位于装置200的不同部分中的IC设备202和230。在一个示例中,通信链路220可被设在搭载IC设备202和230的芯片载体、基板或电路板上。在另一示例中,第一IC设备202可位于折叠式电话的按键板区段中,而第二IC设备230可位于该折叠式电话的显示器区段中。在另一示例中,通信链路220的一部分可包括电缆或光学连接。

[0054] 通信链路220可包括多个信道222、224和226。一个或多个信道226可以是双向的,并且可以工作在半双工和/或全双工模式下。一个或多个信道222和224可以是单向的。通信链路220可以是非对称的,由此在一个方向上提供较高带宽。在本文描述的一个示例中,第一通信信道222可提供或被称为前向链路,而第二通信信道224可提供或被称为反向链路。第一IC设备202可以被指定为主机系统或发射机,而第二IC设备230可以被指定为客户端系统或接收机,即便IC设备202和230两者都被配置成在通信信道222上传送和接收。在一个示例中,前向链路可以在将数据从第一IC设备202传达给第二IC设备230时以较高数据率操作,而反向链路可以在将数据从第二IC设备230传达给第一IC设备202时以较低数据率操作。

[0055] IC设备202和230可各自具有处理器206、236和/或处理和/或计算电路或其他这样的设备或电路。在一个示例中,第一IC设备202可执行装置200的核心功能,包括通过无线收发机204和天线214来维持无线通信,而第二IC设备230可支持管理或操作显示器控制器232的用户接口。第一IC设备202或第二IC设备230可使用相机控制器234来控制相机或视频输入设备的操作。IC设备202和230中的一者或多者所支持的其他特征可包括键盘、语音识别组件、以及其他输入或输出设备。显示器控制器232可包括支持显示器(诸如液晶显示器(LCD)面板、触摸屏显示器、指示器等)的电路和软件驱动器。存储介质208和238可包括瞬态和/或非瞬态存储设备,其被适配成维持由相应处理器206和236、和/或IC设备202和230的其他组件所使用的指令和数据。每个处理器206、236与其相应的存储介质208和238以及其他模块和电路之间的通信可分别由一条或多条总线212和242来促成。

[0056] 反向链路(在此是第二通信信道224)可以按与前向链路(在此是第一通信信道222)相同的方式操作,并且第一通信信道222和第二通信信道224可以能够以相当的速度或以不同的速度进行传送,其中速度可被表达为数据传输速率和/或时钟速率。取决于应用,前向和反向数据率可以基本上相同或相差几个数量级。在一些应用中,单个双向链路(在此是第三通信信道226)可支持第一IC设备202与第二IC设备230之间的通信。当例如前向和反向链路共享相同的物理连接并以半双工方式工作时,第一通信信道222和/或第二通信信道224可以可配置以按双向模式工作。在一个示例中,通信链路220可被操作以根据行业或其他标准在第一IC设备202与第二IC设备230之间传达控制、命令以及其他信息。

[0057] 在一个示例中,前向和反向链路可被配置或适配成支持宽视频图形阵列(WVGA)、

每秒80帧的LCD驱动器IC而不需要帧缓冲器,以810Mbps递送像素数据以供显示器刷新。在另一示例中,前向和反向链路可被配置或适配成与动态随机存取存储器(DRAM)(诸如双倍数据率同步动态随机存取存储器(SDRAM)实现通信。编码设备210和/或230可以在每一时钟转变编码多个比特,且多组导线可被用来传送和接收来自SDRAM的数据、控制信号、地址信号等。

[0058] 前向和反向信道可以遵循或兼容因应用而异的行业标准。在一个示例中,MIPI标准定义包括应用处理器的IC设备202与控制/或支持移动设备中的相机或显示器的IC设备230之间的物理层接口。MIPI标准包括管控遵循移动设备的MIPI规范的产品可操作特性的规范。MIPI标准可定义采用互补金属氧化物半导体(CMOS)并行总线的接口。

[0059] 在一个示例中,图2的通信链路220可被实现为包括多条信号线(记为N条导线)的有线总线。这N条导线可被配置成携带编码在码元中的数据,其中每一码元定义N条导线的信令状态,并且其中时钟信息被嵌入在多条导线上传送的码元序列中。

[0060] 图3是解说在两个设备之间提供的N线接口300的一个示例的示图。在发射机302处,转码器306可被用于使用N阶乘($N!$)编码来将数据比特304和时钟信息编码在要在一组N条导线314上传送的码元中。时钟信号推导自传送时钟312并且可以通过以下方式被编码到在这N条导线314上在 $N C_2$ 个差分信号中传送的码元序列中:确保在连贯码元之间在这 $N C_2$ 个信号中的至少一个信号上发生信令状态转变。当 $N!$ 编码被用来驱动这N条导线314时,码元的每一比特作为差分信号由一组差分线路驱动器310中的一者来传送,其中该组差分线路驱动器310中的诸差分驱动器被耦合到这N条导线中的不同导线对。导线对的可用组合的数目($N C_2$ 个)决定了能够在这N条导线314上传送的信号的数目。可以基于可用于每个码元传输区间的可用信令状态的数目来计算能够被编码在码元中的数据比特304的数目。

[0061] 端接阻抗(通常为电阻性的)将这N条导线314中的每一条耦合到端接网络316中的共用中心点318。将会领会,这N条导线314的信令状态反映了终接网络316中的电流的组合,该电流组合归因于耦合到每条导线的差分线路驱动器310。将会进一步领会,中心点318是零点,藉此终接网络316中的电流在该中心点处彼此抵消。

[0062] 因为该链路中的这 $N C_2$ 个信号中的至少一个信号在连贯码元之间转变,所以 $N!$ 编码方案不需要使用单独的时钟信道和/或非归零解码。有效地,转码器306通过产生其中每个码元都不同于其紧接的前趋码元的码元序列来确保在这N条导线314上传送的每对码元之间发生转变。在图3中所描绘的示例中,提供了 $N=4$ 条导线,并且这4条导线能够携带 $4 C_2=6$ 个差分信号。转码器306可采用映射方案来生成原始码元以供在这N条导线314上进行传输。转码器306可将数据比特304映射到转变数集合。这些转变数随后可以被用于基于在前码元的值来选择供传输的原始码元,以使得所选择的原始码元与在前原始码元不同。在一个示例中,转变数可以被用于参照连贯原始码元中的第一原始码元来查找对应于这些连贯原始码元中的第二原始码元的数据值。例如,在接收机320处,转码器328可以采用映射来确定在查询表中表征一对连贯原始码元之间的差别的转变数。转码器306、328在每一对连贯原始码元包括两个不同码元的基础上操作。

[0063] 发射机302处的转码器306可以在每次码元转变处可用的 $N!-1$ 个码元之间进行选择。在一个示例中, $4!$ 系统在每个码元转变处为要被传送的下一码元提供 $4!-1=23$ 个信令状态。比特率可以被计算为每传送时钟循环 $\log_2$ (可用状态)。在使用双倍数据率(DDR)时钟

(藉此码元转变在传送时钟312的上升沿和下降沿两者处发生)的系统中,每传送时钟循环传送两个码元。对于 $N=4$,传送时钟循环中的总可用状态是 $(N!-1)^2=(23)^2=529$ 个,并且每码元可传送的数据比特304的数目可被计算为 $\log_2(529)=9.047$ 比特。

[0064] 接收机320使用一组线接收机322来接收该码元序列,其中该组线接收机322中的每个接收机确定 N 条导线314中的一对导线上的信令状态的差异。相应地,使用 $N C_2$ 个接收机,其中 N 表示导线的数目。 $N C_2$ 个接收机产生相应数目的原始码元作为输出。在所描绘的 $N=4$ 导线示例中,在这4条导线314上接收到的信号由6个接收机($4 C_2=6$)处理以产生被提供到相应的CDR 324和解串器326的状态转变信号。CDR 324可产生可由解串器326使用的接收时钟信号334。接收时钟信号334可以是能够由外部电路系统用来接收由转码器328提供的数据的DDR时钟信号。转码器328通过将每个下一码元与紧挨在其前的码元作比较来解码来自解串器326的收到码元块。转码器328产生与提供给发射机302的数据比特304相对应的输出数据330。

[0065] 转变编码示例

[0066] 图4是解说根据本文所公开的某些方面来配置的发射机400和接收机420的框图。发射机400和接收机420可被适配成与各种编码技术(包括 $N!$ 和CCIE接口中使用的转变编码)联用。发射机400包括被配置成将数据402转换成转变数414的第一转换器404。转变数414可用于基于当前码元的值来选择供传输的下一码元,其中该下一码元不同于当前码元。第二转换器(诸如编码器406)接收转变数并产生供使用适当配置的线驱动器408来在接口上传送的码元序列。由于没有连贯码元对包括两个等同码元,因此在每一次码元转变时在接口的信号线418中的至少一者中发生信令状态的转变。在接收机420处,一组线接收机426向CDR 428提供原始码元(SI) 436,该CDR 428提取接收时钟438并将捕捉到的码元(S) 434提供给将捕捉到的码元434转换成转变数432的电路。转变数可由电路422解码以提供输出数据430。

[0067] 在 $3!$ 系统的示例中,发射机400可被配置或适配成将数据402转码成由3个比特来表示的五进制(基数为5)转变数414。在CCIE系统的示例中,发射机400可被配置或适配成将数据402转码成由2个比特来表示的三进制(基数为3)转变数414。转变数414可被编码在码元序列416中以在信号线418上传送。提供给发射机400的数据402可以是一个或多个字,每一字具有20比特。第一转换器404(它可以是转码器)接收数据402并为每一数据元素产生转变数序列414。转变数序列414可包括足够数目的三进制数以编码预定义数目的数据比特、检错、以及其他信息。编码器406产生通过线驱动器408传送的码元序列416。在一个示例中,线驱动器408可包括开漏输出晶体管。在另一示例中,线驱动器408可包括推挽式驱动器。通过确保没有连贯码元对包括两个相同码元,由编码器生成的输出码元序列416在码元序列416中的每一对连贯码元之间具有信号线418中的至少一者的状态转变。至少一条信号线中的状态转变可用性准许接收机420从码元序列416中提取接收时钟438。

[0068] 图5是解说编码方案500的简单示例的示图。在这一示例中,编码方案可由配置成产生码元序列416以供在两线CCIE接口上传输的编码器406使用。编码方案500还由转码器424用来从接收自在接口的信号线418上传送的信号的码元中提取数据。在所解说的编码方案500中,使用两根信号线418准许定义4个基本码元S: {0,1,2,3}。码元序列416、434中的任何两个连贯码元具有不同状态,并且码元序列0,0、1,1、2,2和3,3是连贯码元的无效组合。

相应地,在每个码元边界处仅3个有效码元转变可用,其中码元边界由传送时钟确定并且表示第一码元(Ps)终止且第二码元(Cs)开始的点。第一码元可被称为在前或先前码元522,且第二码元可被称为当前码元524。

[0069] 根据本文所公开的某些方面,针对每个先前码元522,这三个可用转变被指派转变数(T) 526。T 526的值可以由三进制数表示。在一个示例中,转变数526的值通过指派用于编码方案的码元排序圆502来确定。码元排序圆502为4个可能码元分配码元排序圆502上的位置504a-504d以及位置504a-504d之间的旋转方向506。在所描绘的示例中,旋转方向506为顺时针。转变数526可以表示有效的当前码元524与前一紧邻码元522之间的间隔。间隔可被定义为从前一码元522到达当前码元524所需要的在码元排序圆502上沿旋转方向506的步数。步数可被表达为一位基数为3的数。将领会,码元之间的三步差异可被表示为 $0_{\text{基数}=3}$ 。图5中的表520概述了采用这种办法的编码方案。

[0070] 在发射机400处,在给定先前码元522和用作转变数526的输入三进制数的知识的情况下,表520可被用于查找要被传送的当前码元524。在接收机420处,表520可被用作查找表以确定表示先前码元522与当前码元524之间的转变的转变数526。转变数526可作为三进制数来被输出。

[0071] 使用将时钟信息嵌入到码元序列中的转码器可将发射机400接收到以供传输的数据402与在信号线418上传送的码元序列416解除关联。因此,在不考虑至少一个先前传送的码元的情况下,所接收到的原始码元436无法被直接解码以获得提供给发射机400的数据402。该解除关联可使得常规纠错技术无效。例如,常规系统可将纠错码(ECC)追加到数据402,其中该ECC可以是数据402的预定义块大小或者分组长度计算出的循环冗余码(CRC)。该ECC可被用于标识和/或纠正在常规接口中的传输期间发生的差错,其中这些差错可包括一个或多个比特差错。

[0072] 在使用转变编码的接口中,码元差错以比特差错突发来显现在接收机处。即,单个码元传输差错可引起多个比特差错。在这些情形中,CRC通常超过汉明距离且不是用于检错的切实解决方案。

[0073] 图6是解说码元602与转变数604(其在本文中也可被称为“转变码元”)之间的关系与时序图600。在这一示例中,每一数据字被编码到在多线接口上传送的m个码元中。在m个码元中传送的字可以使用以下公式来解码:

$$[0074] \quad \sum_{k=0}^{m-1} T_k r^k$$

[0075] 其中 T_k 是第k迭代处的转变数,且r是码元之间的每一转变处的可用码元的数目。例如,在其中自转变被禁止(以确保接收时钟能被可靠地生成)的3!接口中,6个定义状态中的r=5个状态在每一码元转变处可用。在各种示例中,3!接口可以将数据编码在m=4个码元或m=7个码元的序列中。在4!接口中,24个定义状态中的r=23个状态在每一码元转变处可用,并且4!接口可以将数据编码在m=2个码元的序列中。在CCIE接口中,4个定义状态中的r=3个状态在每一码元处可用,并且数据字可被编码在m=12个码元的序列中。对于3线单端接口,m=12和r=7的值可被使用。对于4线单端接口,m=10和r=15的值可被使用。

[0076] 图7是解说用于3!接口的转变数到码元编码的示图700。在这一示例中,存在6个可

能码元, S: {0, 1, 2, 3, 4, 5}, 其围绕码元排序圆702布置。通过确保同一码元不出现在任何两个连贯码元区间中, 时钟信息被嵌入在码元序列中。在这一示例中, $r=5$, 并且对于每一类型的转变704、706、708、710、712, 转变数(T)可被指派不同值。转变数的值可以相对于码元排序圆702上的当前码元的位置来指示码元排序圆702上的下一码元的位置。转变数可以取在范围1-5中的值。因为当前码元不能与前一码元相同, 所以当前码元与下一码元之间的步数不能是零。

[0077] 转变数可根据以下公式来被指派:

[0078] $T = Ps+1 \leq Cs ? Cs - (Ps+1) : Cs - (Ps+1) + 6$

[0079] 相反, 当前顺序码元编号(Cs)可根据以下规则来指派:

[0080] $Cs = Ps+1+T < 6 ? Ps+1+T : Ps+1+T-6$

[0081] 其中Cs是当前码元, 且Ps是前一收到码元。

[0082] 图8是解说码元转变时钟转码的一般化示例800的示图。在这一示例800中, 接口对于在多线通信接口上传送的每一码元提供六个可能的信令状态, 其中通过确保每一对连贯地传送的码元包括两个不同码元, 时钟信息被嵌入在连贯码元之间的每一转变处。相应地, 在码元之间的每一转变处有5个状态可用。通过将数据字的各比特转换成转变数来编码该数据字, 这基于当前所传送的码元来选择要传送的下一码元。在示例800中, 三个顺序码元812、814、816在多线通信接口上传送, 其中每一码元812、814、816定义多线通信接口的六个信令状态之一。数据和时钟信息被编码在码元812、814、816的连贯对之间的转变中。转变可被表示为转变数808、810的各个数位。转变数的每一数位标识码元序列中的一对连贯码元之间的转变, 并且在这一上下文中, 各数位也可被称为转变数。如本文所述, 对于m个码元的序列, 数据被编码为:

[0083] 数据 =
$$\sum_{k=0}^{m-1} T_k r^k$$

[0084] 其中k具有在0和m-1之间的值。第一转变数(T_k)808对应于第一码元812(A)与第二码元814(X)之间的转变, 并且第二转变数(T_{k-1})810对应于第二码元814(X)与第三码元816(B)之间的转变。在此, 第一码元812可以编码数据字的最高有效位。

[0085] 在一个示例中, 多比特数据字可被转换成m个转变数的序列。每一转变数可以使用三进制数、四进制数、五进制数、六进制数或使用可表示r个转变的某一其他数字系统来表达。即, 数字系统可以是提供能跨范围0到r-1的数字的r基系统。每一转变数可以基于正传送的当前码元来选择供传输的下一码元。从与当前码元不同的诸码元中选择下一码元以确保发生信令状态转变, 以便将时钟信息嵌入码元序列802中。即, 连贯码元对中的两个不同码元的传输造成多线接口的至少一条导线的信令状态的变化, 并且接收机可以基于在连贯码元之间的信令状态中检测到的变化来生成接收时钟。

[0086] 在示例800中, 码元排序圆806解说了一种选择下一码元的方法。在此, 转变数可被表达为五进制数(基数为5), 具有可能值{0, 1, 2, 3, 4}。对于六个可能码元804a-804f中的每一者, 六个信令状态之一在多线通信接口上传送。六个码元804a-804f被布置在绕码元排序圆806的不同位置中。给定在码元排序圆806上的当前码元位置, 可通过选择在码元排序圆806上位于T个顺时针步数处的码元作为下一码元来对转变数T进行编码。在一个示例中, 在

当前码元是码元0 804a时,转变数值 $T=1$ 选择码元1 804b作为下一码元,转变数值 $T=2$ 选择码元2 804c作为下一码元,转变数值 $T=3$ 选择码元3 804c作为下一码元,且转变数值 $T=4$ 选择码元4 804d作为下一码元。转变数值 $T=0$ 可造成翻转,因为该转变数选择距当前码元(码元0 804a)为5个顺时针步数(或1个逆时针步数)的码元,从而选择码元5 804f作为下一码元。

[0087] 在所传送的码元序列802的示例中,码元序列802中的第一码元812可对应于码元1 804b。输入数据可被处理以产生具有值 $T_k=2$ 的第一转变数808、以及具有值 $T_{k-1}=1$ 的第二转变数810。第二码元814可基于 T_k 的值被确定为码元3 804d,且第三码元816可基于 T_{k-1} 的值被确定为码元4 804e。

[0088] 在接收机处,码元排序圆806可被用来确定连贯码元812、814和/或816之间的每一转变的转变数。在一个示例中,接收机基于连贯码元812、814和/或816之间发生的信令状态变化来提取接收时钟。接收机随后可从多线接口捕捉码元812、814、816并确定表示每一对连贯码元812、814和/或816之间的转变的转变数。在一个示例中,可通过计算该对连贯码元812、814之间在码元排序圆806上的步数来确定转变数。

[0089] 转变编码接口中的检错

[0090] 根据本文公开的某些方面,可使用添加到要在转变编码接口上传送的数据的EDC来在转变编码接口中实现可靠检错。EDC可包括具有预定义位数的预定义数目的比特,其中EDC具有已知、固定值。在一个示例中,EDC在被传送时具有零值。在一些情形中,EDC被提供作为要在该接口上传送的每一字的最低有效位(LSB)。EDC字的形式和结构可被选择成使得影响字的单个信令状态差错造成在接收机处解码的EDC具有不同于该固定值的值(例如,非零值)。

[0091] 图9解说了影响转变编码接口的单个差错的效应的示例900。在该示例中,数据字912被提供以供在该接口上传输。EDC 914被追加到数据字912以产生输入到编码器的传输字902。传输字902在码元序列910中传送,其中码元序列910包括12个码元。码元序列910在被配置用于CCIE操作的两线接口上传送并且在接收机处以码元流904被接收。在传输中,发生信令差错,使得初始传送的码元916被修改并被接收为错误码元918。与接收到的码元流904相对应的转变数流906包括包含差错偏移的转变数920、922。第一转变数920表示前一码元与错误码元918之间的差异,且第二转变数922表示错误码元918与在受影响码元之后传送的下一码元之间的差异。

[0092] EDC 914的大小、位置以及结构可被选择成使得单码元差错的发生在接收机处产生与所传送的EDC 914不同的EDC 926。在一个示例中,EDC 914包括多个比特且可被设置成零值。在CCIE接口的示例中,EDC 914可具有三个比特。

[0093] 图10是解说其中在多线通信接口上传送的码元序列1002受单码元差错1018影响而导致在接收到的码元序列1004中捕捉到错误码元1014的示例的示图。所传送的码元序列1002包括第一码元1008(A码元)、第二码元(X码元)1010以及第三码元1012(B码元)。在接收到的码元序列1004中,第一码元1008和第三码元1012被正确地接收,而第二码元1014被码元差错1018修改(位移e)且被接收为错误码元(X'码元1014)。

[0094] 单码元差错1018的发生导致两个转变数差错。第一不正确转变数1020表示正确接收到的第一码元1018与X'码元1014之间的转变。第二不正确转变数1022表示X'码元1014与

正确接收到的第三码元1012之间的转变。第一不正确转变数1020可被表达为 $T_k + e_k$ ，其中 T_k 是与第一码元1008和X码元1010之间的转变相对应的第一正确转变数1016，而 e_k 是在第一不正确转变数1020中造成的相对于第一正确转变数1016的差错值。第二不正确转变数1022可被表达为 $T_{k-1} + e_{k-1}$ ，其中 T_{k-1} 是与X码元1010和第三码元1012之间的转变相对应的第二正确转变数1024，而 e_{k-1} 是在第二不正确转变数1022中造成的相对于第一正确转变数1024的差错值。

[0095] 单码元差错1018的效应在解码转变圆1006中解说。初始从多线接口接收与码元1相对应的第一码元1008。下一码元由于差错而被不正确地捕捉为X'码元1014。X'码元1014可对应于码元0。随后从多线接口接收与码元4相对应的第三码元1012。在这一示例中，最高有效码元被首先传送，并且：

[0096] $e = 3, T_k = 2, T_{k-1} = 1$

[0097] $T_k + e = 2 + 3 = 5 = 0_{\text{基数5}}$ ，以及 $e_k = -2$

[0098] $T_{k-1} - e = 1 - 3 = -2 = 4_{\text{基数5}}$ ，以及 $e_{k-1} = -3$

[0099] 每一数据字可由转变数序列 $\{T_0, T_1, \dots, T_{m-1}\}$ 来表示。

[0100] 位移差错 e 表示所传送的X码元1010与接收到的X'码元1014之间的差异，它可对应于解码转变圆1006中的步数。由于用来表达转变数的数字系统中的翻转， e 的值在取值上不一定等于 e_k 。例如，具有值3的转变数可表示解码转变圆1006上的第一正确转变数1016由位移差错 e 造成的在所传送的X码元1014与接收到的X'码元1014之间的差异，而 e_k 的值具有值2。

[0101] 对于两个连贯码元转变：

[0102] 比特 $= T_k r^k + T_{k-1} r^{k-1}$

[0103] 影响两个连贯码元的单个差错的结果可被表达为：

[0104] 比特' $= (T_k + e_k) r^k + (T_{k-1} - e_{k-1}) r^{k-1}$

[0105] $= (T_k r^k + T_{k-1} r^{k-1}) + (e_k r - e_{k-1}) r^{k-1}$

[0106] 其中：

[0107] $(e_k r - e_{k-1}) r^{k-1}$ 可被称为差错效应，

[0108] $(e_k r - e_{k-1})$ 可被称为差错系数，以及

[0109] r^{k-1} 可被称为基幂。

[0110] 根据某些方面，转变编码接口可被配置成使得 r 是奇数。在 r 是奇数时，得出 r^{k-1} 也是奇数（LSB非零）。相应地， $(e_k r - e_{k-1})$ 的值确定EDC所需的LSB的数目。图11提供了在 $r=3$ 和5时 r^n 的列表（其中 n 在范围0-15中）。第一表1100可与CCIe接口相关，其中在每一码元区间处有 $r=3$ 个转变可用。在每一实例中，基幂的LSB 1104被设置成‘1’。第二示例1102可以与3线3!接口相关，其中在每一码元区间处有 $r=5$ 个转变可用（6个可能码元）。在每一实例中，基幂的LSB 1206被设置成‘1’。

[0111] 图12是通过表格示出差错系数并且解说在码元差错不涉及连贯码元区间中的码元重复（这可造成时钟缺失）的差错系数的表1200。 $|e_k|$ 总是小于 r 。即：

[0112] $1 \leq |e_k| \leq r-1$ ，

[0113] $1 \leq |e_{k-1}| \leq r-1$ 。

[0114] 因为 $|e_k|$ 的最小值是1，所以 $|e_k r|$ 的最小值是 r 。 $|e_{k-1}|$ 的最大值是 $r-1$ 。在存在单码

元差错时,差错系数($e_k r - e_{k-1}$)永不为零。

[0115] 图13解说了差错系数中的最长非零LSB部分的计算和表格化的示例1300。在此,在 $|e_k|$ 和 $|e_{k-1}|$ 两者是 $2(2^n)$ 的最长幂且 $e_k = e_{k-1}$ 时, $(e_k r - e_{k-1})$ 的2个LSB的幂是最长的。差错系数的2个LSB的最长幂确定“检错常数LSB”的大小。

[0116] 图14解说了其中单码元差错导致单个转变数1408、1426中的差错的情况的两个示例1400、1420。在第一示例1400中,信令差错影响在前码元序列中最后传送的码元1402。该信令差错导致接收机检测到经修改码元1404作为在前码元序列中最后接收的码元。该差错可在表示在前码元序列中最后传送的码元1402与当前码元序列的第一码元之间的差异的转变数1406中引入偏移。在第一示例1400中,该差错的效应可被表达为: $e_{m-1} r^{m-1}$,其中差错系数是 e_{m-1} 且基幂是 r^{m-1} 。

[0117] 在第二示例1420中,信令差错影响当前码元序列中最后传送的码元1422。该信令差错导致接收机检测到经修改码元1424作为当前码元序列中最后接收的码元。该差错可在表示当前码元序列中最后传送的码元1422与下一码元序列的第一码元之间的差异的转变数1426中引入偏移。在第一示例1400中,该差错的效应可被表达为 e_0 。

[0118] 表1列出可检测使用转变编码的多线接口中的单码元差错的EDC中的LSB的数目。

[0119]

r	EDC长度(比特)	示例
3	3	2线单端(例如,CCIE)
5	5	3线多级差分(3!)
7	6	3线单端
9	7	
11	5	
13	6	
15	8	4线单端
17	9	
19	8	
21	7	
23	6	4线多级差分(4!)

[0120] 表1

[0121] 图14中解说的情形不影响EDC中准许检测单码元差错所需的LSB的最大数目。

[0122] 检测每字多个码元差错

[0123] 图15是解说影响对单个数据字进行编码的码元序列1502中的两个码元1504、1506的信令差错的第一示例的时序图1500。图15涉及其中信令差错影响两个非连贯码元的示例。码元1504、1506中的差错导致相应的各对转变差错1508、1510。这些转变差错导致错误的转变数1512、1514、1516、1518。可归因于第一受影响码元1504的差错效应可被表述为 $(e_k r - e_{k-1}) r^{k-1}$,而可归因于第二受影响码元1506的差错效应可被表述为 $(e_j r - e_{j-1}) r^{j-1}$ 假定如果差错的总效应 $(e_k r - e_{k-1}) r^{k-1} + (e_j r - e_{j-1}) r^{j-1}$ 总是修改具有预定长度和值的EDC,则多个码元差错可被检测。

[0124] 图16是解说影响对单个字进行编码的码元序列1602中的两个连贯码元1604、1606

的信令差错的第二示例的时序图1600。连贯码元1604、1606中的差错导致转变差错1608,转变差错1608导致生成三个错误的转变数1610、1612、1614。可归因于受影响码元1504、1506的差错效应可被表述为 $(e_k r^2 + e_{k-1} r + e_{k-2}) r^{k-2}$ 。在根据本文公开的某些方面适配的接收机中,与非连贯码元1504、1506中的差错相比,可以用更短EDC来检测可归因于影响连贯码元1604、1606的差错的差错效应。

[0125] 图17是解说用于 r (每码元边界可用转变) 和 m (用来对数据元素进行编码的码元数目) 的各种值的EDC比特数的表1700。用于检测两个码元差错的EDC大小随 m 的值而变化。表1700的第一行(带阴影)对应于用来检测单码元差错的EDC。

[0126] 根据本文公开的某些方面,在足够长度的EDC与数据字一起传送时,接收机可被配置成检测表示数据字的码元序列中的两个码元差错。EDC的长度可基于用来对数据字进行编码的码元数目以及一对连贯地传送的码元之间的边界处可用的转变的数目来确定。

[0127] 由时钟缺失或额外时钟造成的码元滑动差错可能不被检错常数检测到。然而,这些类型的差错中的大多数可由较高协议层、在下一字处、和/或使用接收机设备处的状态机被检测到。

[0128] 每字单码元差错的纠正

[0129] 图18是解说根据本文公开的可被用来纠正每字单码元差错的某些电路的操作的示图1800。在码元序列1802中传送的EDC允许检测码元序列1802中的两个码元差错时,纠错电路可以纠正对字进行编码的码元序列1802中的单码元差错。纠错电路可被配置成当已在码元序列1802中检测到差错之后生成并解码从通信链路接收到的码元序列1802中的码元不同置换。每一置换可包括对码元序列1802中的一个码元的修改。当在传输期间引入单码元差错时,改变码元序列1802中正确地接收到的码元使得解码器解码出指示码元序列1802中存在一个或多个差错的EDC。将码元序列1802中具有差错的码元改变成另一不正确码元会得到指示单个差错仍然保留在码元序列1802中的EDC。当码元置换中的经修改码元逆转在传输期间引入的差错时,解码器解码出与预期的、初始传送的EDC值相匹配的EDC,并且该码元置换和/或从该码元置换解码出的数据可由接收机用来代替接收到的码元序列1802和/或从中解码出的数据。

[0130] 在图18的上下文中,发射机可将组合数据字和EDC编码在 m 个码元中。在图9解说的示例中,传输字902可包括数据字912和EDC 914。如果在解码了码元序列1802中的 m 个码元之后,经解码EDC具有非预期值,则接收机可以确定至少一个码元已被传输差错影响。在接收到的EDC不匹配用于通信链路上的传输且被发射机追加到数据字912的EDC 914的预定义值时,接收到的EDC可具有非预期值。在一个示例中,发射机可以追加数个零值比特作为传输字902的最低有效位,并且经解码的非零EDC可指示在码元序列1802的传输中已发生至少一个码元差错。

[0131] 在EDC可被用来检测一个码元序列1802中的两个码元差错的发生时,纠错电路可以纠正单码元差错。在一个示例中,作为足够数目的最低有效位追加到数据字的EDC可以准许检测两个码元差错以及由纠错电路纠正单码元差错。

[0132] 在从码元序列1802解码出非预期EDC值时,纠错电路可被启用。纠错电路可以通过解码通过修改码元序列1802中的单个码元所获得的每一码元置换来纠正单码元差错。产生具有预期EDC值的经解码EDC的码元置换可被认为已经纠正了传输差错。即,该码元置换包

括已经按逆转由传输差错引起的先前修改的方式来被修改的码元。

[0133] 在图18的示例1800中解说的示例中,码元序列1802中的每一码元可以取 $r+1$ 个可能信令状态之一,其中 r 表示每码元的可能转变的数目。即,在嵌入时钟信息的编码方案中,每一对连贯地传送的码元包括两个不同码元以确保每一码元边界处发生转变,并且 $r+1$ 个所定义的码元之一在每一码元边界处不可用。在第一示例中,两线接口可被操作,使得 $r+1=4$ 个信令状态可用。在第二示例中,三线接口可被操作,使得 $r+1=6$ 个信令状态可用。传输差错可以产生由接收机指派与本来要被指派给初始传送的码元的值 $V_{\text{正确}}$ 不同的值 $V_{\text{错误}}$ 的码元 S_i 。解码器在解码包括受差错影响的码元(即, $S_i = V_{\text{错误}}$)的码元序列1802之后可以产生不具有预期值的收到EDC。预期EDC值1818可具有预定义EDC值。

[0134] 根据某些方面, S_i 的值可被迭代地改变并在码元序列1802的置换中提供,它随后被解码以确定该码元置换是否产生与预期EDC值1818相匹配的EDC。当在码元置换中用初始传送的码元值($S_i = V_{\text{正确}}$)替换了受差错影响的码元时,经解码EDC具有预期EDC值1818,并且该码元置换可替代接收到的码元序列1802,和/或从该码元置换解码出的数据字可替代从接收到的码元序列1802解码出的数据字。

[0135] 在一些实现中,码元序列1802中所有码元1812的值可被提供给包括 r (即, $(r+1)-1$)个码元-到-数据解码器的解码器集合1808。该解码器集合1808中的每一解码器接收码元序列1802中的每一码元的值,除了 S_i 的值。代替 S_i 的值,解码器集合1808中的每一解码器接收 S_i 的经修改版本,并且相应地,解码器集合1808中的每一解码器解码从码元序列1802导出的码元置换。加法器集合1806可被配置成将 S_i 的不同版本提供给解码器集合1808。加法器集合1806可包括模 (r) 加法器,其每一者都提供不同码元值以在码元置换中替代 S_i 。

[0136] 在图18中描绘的示例中,涉及对单个码元的改变的所有置换可被并行地处理和解码。可使用多比特比较器1810来比较从多个码元置换解码出的EDC值1816,其中每个比较器可以在预期EDC值1818与从对应码元置换解码出的EDC值1816之间存在匹配的情况下产生逻辑1。如果多比特比较器1810之一指示匹配,则影响码元序列1802的传输差错可被确定为已被纠正。

[0137] 在一些实现中,可为码元序列1802中的每一码元提供纠错块,并且可针对每一码元并发地生成和解码诸置换集合,其中每一置换集合包括码元序列1802中单个码元的修改。每一纠错块可包括加法器集合1806、解码器集合1808、以及多比特比较器1810。在这些实现中,在纠错块提供来自该纠错块中的比较器1810之一的匹配信号时,纠错完成,并且该码元置换和/或对应的数据字可被提供作为纠错电路的输出。

[0138] 在其他实现中,纠错电路可以使用状态机、定序逻辑、和/或一个或多个处理电路来在多次迭代中执行纠错。在这些实现中的一些中,多达 m 个纠错循环可被执行以轮流测试对每一码元($S_0, S_1, \dots, S_i, \dots, S_m$)的置换的效应。在第一纠错循环中,与码元序列1802中的第一码元(S_0)相关的 r 个置换被解码以确定是否可解码出预期EDC值。在第二纠错循环中,与码元序列1802中的第一码元(S_1)相关的 r 个置换被解码以确定是否可解码出预期EDC值。这一过程可以继续,直至发现在解码时与预期EDC值1818相匹配的置换。在获得了EDC匹配时,纠错过程可以终止。

[0139] 根据某些方面的纠错技术的示例

[0140] 可通过共享用于生成和解码各码元置换的组件来减少用于实现纠错电路的硬件。

例如,可以用预加载有接收到的码元序列1802中的 S_i 的值的模(r)计数器来替换加法器集合1806。对于 r 次迭代中的每一迭代,计数器可被递增且其输出可被提供给解码器作为接收到的码元序列1802中的 S_i 的值的替代。解码器所产生的EDC可以与预期EDC值1818相比较。在一些示例中,可为码元序列1802中的每一码元提供计数器和解码器,使得在检错过程的每一循环中生成并解码 m 个置换,这可延及 r 个循环。在其他示例中,可使用单个计数器并且最大数目的检错循环可被计算为 $m(r-1)$ 个循环。

[0141] 各种硬件组合可被采用以根据本文公开的某些方面来纠错。在一个硬件密集示例中,计数器、加法器以及解码器可被提供以使得每一可能的码元置换可被并发地测试。在后一示例中,纠错电路能以低等待时间来操作。在另一示例中,可使用单个解码器来顺序地测试每一置换,使得纠错可能需要大量时间来完成纠错过程。根据某些方面,可基于应用要求作出硬件与纠错过程时间之间的折衷。在一些实例中,置换生成、解码和/或纠错决策可被实现在由软件指令控制的定序器和/或处理设备中。

[0142] 图19是解说根据本文公开的可被用来纠正码元差错的解码器的配置的第一示例的示图1900。在这一示例中,可使用定序逻辑、状态机或控制器来在多次迭代中执行纠错,其中在每一迭代中,通过修改单个码元所创建的诸置换可被测试。在该示例中,数据字可被编码在包括10个码元 $\{S_0, S_1, \dots, S_9\}$ 的码元序列中,并且每一码元可具有4个值之一。在已确定接收到的码元序列1902中存在码元差错时,可针对多次测试迭代中的每一迭代来并发地生成码元序列1902的三个置换1904、1906、1908。在测试迭代中生成的每一置换1904、1906、1908中,只改变码元序列1902中的一个码元位置(目标码元1910)。在该示例,第一迭代正在进行中,并且在三个置换1904、1906、1908中只改变了码元序列1902中的 S_0 。在一个示例中, S_0 的值可通过简单的模3加法来修改。随后可使用包括3个解码器和比较器的解码器电路和/或逻辑1912来测试这三个置换1904、1906、1908,使得针对所有三个置换1904、1906、1908生成的EDC可与预期EDC值1914相比较。如果纠正未完成,则针对下一码元(在此是 S_1)执行另一迭代,直至所有10个码元已被测试。这一办法使用最多10次测试迭代和3条解码器路径。

[0143] 图20是解说根据本文公开的可被用来纠正码元差错的解码器的配置的第二示例的示图2000。在这一示例中,可使用定序逻辑、状态机或其他控制器来在多次迭代中执行纠错,其中在每一迭代中,在每迭代生成的置换集合2004之一中修改码元序列2002中的每一码元。在该示例中,数据字可被编码在包括10个码元 $\{S_0, S_1, \dots, S_9\}$ 的码元序列中,并且每一码元可具有4个值之一。在已确定接收到的码元序列2002中存在码元差错时,可针对多次测试迭代中的每一迭代来并发地生成码元序列2002的10个置换2004。在每测试迭代生成的10个置换2004中的每一者中,码元序列2002中的诸码元位置 $\{S_0, S_1, \dots, S_9\}$ 中不同的一个码元位置被改变。在该示例中,第一迭代正在进行中且10个置换2004中的每一者中的单个码元与码元序列2002中的对应码元相差值1。通过使用模3加法来改变码元以执行 S_{j+1} ,其中 S_j 表示码元序列2002中对应码元的值。随后可使用包括10个解码器和比较器的解码器电路和/或逻辑2006来测试10个置换2004,使得为所有10个置换2004生成的EDC可与预期EDC值2008相比较。如果纠正未完成,则使用下一码元值增量(S_{j+2} ,随后 S_{j+3})来执行另一迭代,直至码元序列2002中的所有码元的所有3个可能值已被测试。这一办法使用最多3次测试迭代和10条解码器路径。

[0144] 图21是解说根据本文公开的某些方面的可被用来降低解码开销的用于纠正码元差错的解码器的某些方面的示图2100。所解说的示例基于图19中解说的解码器的配置,其中数据字可被编码在包括10个码元 $\{S_0, S_1, \dots, S_9\}$ 的码元序列中,并且每一码元可具有4个值之一,并且其中解码器电路和/或逻辑1912包括3个解码器和比较器,使得针对所有三个置换1904、1906、1908生成的EDC可以与预期EDC值1914相比较。可通过考虑改变接收到的码元序列2102中的码元的效应来降低处理开销以及解码器数目。在这一示例中,纠错是在第五迭代中,且在每一置换中修改了 S_4 。码元的4个可能状态可被表示为 $\{A, B, C, D\}$,其中 $B=A+1$ 、 $C=B+1$ 、 $D=C+1$ 以及 $A=D+1$ 。在接收到的码元序列2102中, $S_3=A$ 且 $S_4=B$ 。可以看到,第三置换2108改变了 S_4 的值,使得 $S_4=S_3=A$ 。在转变编码方案中,连贯地传送的各对码元不能具有相同值。在一些实施例中,纠错电路和模块可被适配成识别引入错误状况(在一对连贯码元2112相同时)的置换2108,并且这样的置换2108不必被测试。在一些实例中,置换2104、2106、2108可被动态地指派到解码器电路和/或逻辑1912中提供减少数目的解码器和比较器从而反映由于创建无效连贯码元对2112而消除至少一个置换2108的能力的解码路径。

[0145] 图18-21中解说的示例可被伸缩。例如,在图19中,解码器电路和/或逻辑1912可被适配成包括更大数目的解码器和比较器,使得与码元序列1902中的两个或更多个码元相对应的诸置换集合可被并发地测试(即,每一迭代可测试6个、9个、12个等置换)。在另一示例中,图20的解码器电路和/或逻辑2006可被适配成包括更少解码器和比较器,使得使用增加数目的迭代来测试码元序列1902的所有置换。

[0146] 处理电路的示例

[0147] 图22是解说采用可被配置成执行本文所公开的一个或多个功能的处理电路2202的装置的硬件实现的简化示例的概念图2200。根据本公开的各种方面,本文所公开的元素、或元素的任何部分、或者元素的任何组合可使用处理电路2202来实现。处理电路2202可包括一个或多个处理器2204,其由硬件和软件模块的某种组合来控制。处理器2204的示例包括:微处理器、微控制器、数字信号处理器(DSP)、现场可编程门阵列(FPGA)、可编程逻辑设备(PLD)、状态机、定序器、门控逻辑、分立的硬件电路、以及其他配置成执行本公开中通篇描述的各种功能性的合适硬件。该一个或多个处理器2204可包括执行特定功能并且可由软件模块2216之一来配置、增强或控制的专用处理器。该一个或多个处理器2204可通过在初始化期间加载的软件模块2216的组合来配置,并且通过在操作期间加载或卸载一个或多个软件模块2216来进一步配置。

[0148] 在所解说的示例中,处理电路2202可使用由总线2210一般化地表示的总线架构来实现。取决于处理电路2202的具体应用和整体设计约束,总线2210可包括任何数目的互连总线和桥接器。总线2210将各种电路链接在一起,包括一个或多个处理器2204、和存储2206。存储2206可包括存储器设备和大容量存储设备,并且在本文可被称为计算机可读介质和/或处理器可读介质。总线2210还可链接各种其他电路,诸如定时源、定时器、外围设备、稳压器、和功率管理电路。总线接口2208可提供总线2210与一个或多个收发机2212之间的接口。可针对处理电路所支持的每种联网技术来提供收发机2212。在一些实例中,多种联网技术可共享收发机2212中找到的电路系统或处理模块中的一些或全部。每个收发机2212提供用于通过传输介质与各种其他装置通信的手段。取决于该装置的本质,用户接口2218(例如,按键板、显示器、触摸接口、扬声器、话筒、操纵杆)也可被提供,并且可直接或通过总

线接口2208通信地耦合至总线2210。

[0149] 处理器2204可负责管理总线2210和一般处理,包括对存储在计算机可读介质(其可包括存储2206)中的软件的执行。在这一方面,处理电路2202(包括处理器2204)可被用来实现本文所公开的方法、功能和技术中的任一种。存储2206可被用于存储处理器2204在执行软件时操纵的数据,并且该软件可被配置成实现本文所公开的方法中的任一种。

[0150] 处理电路2202中的一个或多个处理器2204可执行软件。软件应当被宽泛地解释成意为指令、指令集、代码、代码段、程序代码、程序、子程序、软件模块、应用、软件应用、软件包、例程、子例程、对象、可执行件、执行的线程、规程、函数、算法等,无论其是用软件、固件、中间件、微代码、硬件描述语言、还是其他术语来述及皆是如此。软件可按计算机可读形式驻留在存储2206中或驻留在外部计算机可读介质中。外部计算机可读介质和/或存储2206可包括非瞬态计算机可读介质。作为示例,非瞬态计算机可读介质包括:磁存储设备(例如,硬盘、软盘、磁条)、光盘(例如,压缩碟(CD)或数字多功能碟(DVD))、智能卡、闪存设备(例如,“闪存驱动器”、卡、棒、或钥匙驱动器)、随机存取存储器(RAM)、只读存储器(ROM)、可编程ROM(PROM)、可擦式PROM(EPROM)、电可擦式PROM(EEPROM)、寄存器、可移动盘、以及任何其他用于存储可由计算机访问和读取的软件和/或指令的合适介质。作为示例,计算机可读介质和/或存储2206还可包括载波、传输线、和任何其他用于传送可由计算机访问和读取的软件和/或指令的合适介质。计算机可读介质和/或存储2206可驻留在处理电路2202中、处理器2204中、在处理电路2202外部、或跨包括该处理电路2202在内的多个实体分布。计算机可读介质和/或存储2206可实施在计算机程序产品中。作为示例,计算机程序产品可包括封装材料中的计算机可读介质。本领域技术人员将认识到如何取决于具体应用和加诸于整体系统上的总体设计约束来最佳地实现本公开中通篇给出的所描述的功能性。

[0151] 存储2206可维持以可加载代码段、模块、应用、程序等来维持和/或组织的软件,其在本文中可被称为软件模块2216。软件模块2216中的每一者可包括在安装或加载到处理电路2202上并由一个或多个处理器2204执行时有助于运行时映像2214的指令和数据,运行时映像2214控制一个或多个处理器2204的操作。在被执行时,某些指令可使得处理电路2202执行根据本文所描述的某些方法、算法和过程的功能。

[0152] 软件模块2216中的一些可在处理电路2202初始化期间被加载,并且这些软件模块2216可配置处理电路2202以实现本文所公开的各种功能的执行。例如,一些软件模块2216可配置处理器2204的内部设备和/或逻辑电路2222,并且可管理对外部设备(诸如,收发机2212、总线接口2208、用户接口2218、定时器、数学协处理器等)的访问。软件模块2216可包括控制程序和/或操作系统,其与中断处理程序和设备驱动器交互并且控制对由处理电路2202提供的各种资源的访问。这些资源可包括存储器、处理时间、对收发机2212的访问、用户接口2218等。

[0153] 处理电路2202的一个或多个处理器2204可以是多功能的,由此软件模块2216中的一些被加载和配置成执行不同功能或相同功能的不同实例。这一个或多个处理器2204可附加地被适配成管理响应于来自例如用户接口2218、收发机2212和设备驱动器的输入而发起的后台任务。为了支持多个功能的执行,这一个或多个处理器2204可被配置成提供多任务环境,由此多个功能中的每个功能按需或按期望实现为由一个或多个处理器2204服务的任务集。在一个示例中,多任务环境可使用分时程序2220来实现,分时程序2220在不同任务之

间传递对处理器2204的控制权,由此每个任务在完成任何未决操作之际和/或响应于输入(诸如中断)而将对一个或多个处理器2204的控制权返回给分时程序2220。当任务具有对一个或多个处理器2204的控制权时,处理电路有效地专用于由与控制方任务相关联的功能所针对的目的。分时程序2220可包括操作系统、在循环基础上转移控制权的主循环、根据各功能的优先级化来分配对一个或多个处理器2204的控制权的功能、和/或通过对一个或多个处理器2204的控制权提供给处置功能来对外部事件作出响应的中断驱动式主循环。

[0154] 图23和24是解说可被合作地使用以在接收机处接收并纠正多线接口中的传输差错的方法的流程图2300和2400。

[0155] 在第一流程图2300(参见图23)中,在框2302,接收机可以接收表示多个连接器的信令状态的码元序列。

[0156] 在框2304,接收机可以解码该码元序列。码元解码的示例由第二流程图2400(参见图24)来解说。

[0157] 在框2304,接收机可以基于接收到的多个比特中的EDC的值来确定码元序列中是否存在码元差错。

[0158] 如果没有检测到差错,则在框2308,接收机可前进至框2310并跳过检错。如果检测到差错,则在框2308,接收机可通过在框2312继续来发起检错。

[0159] 在框2312,接收机可生成该码元序列的一个或多个置换。每一置换可包括与该码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元。接收机可针对该码元序列中的一个码元的每一可能值来生成置换。在一个示例中,接收机可针对该码元序列中的一个码元的值的每一可能变化来生成置换。在另一示例中,接收机可以生成与该码元序列中的每一码元相对应的置换,其中在每一置换中将码元修改相同的值。

[0160] 在框2314,接收机可解码该一个或多个置换。码元解码的示例由第二流程图2400(参见图24)来解说。接收机可抑制解码包括具有相同值的两个连贯码元的置换。

[0161] 在框2316,接收机可标识产生与预期EDC值相匹配的经解码EDC值的置换。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。信令差错可以使得EDC的经解码版本具有与预定义值不同的值。

[0162] 在第一示例中,在确定码元序列中存在码元差错之后,接收机可并发地生成该码元序列的置换集合,其中该置换集合的所有成员包括该码元序列中的一个码元的经修改版本。即,可使用置换集合来测试该码元序列中的一个目标码元,其中逐个置换只修改了该目标码元。接收机可以并发地解码该置换集合的所有成员。如果没有解码出正确的EDC,则选择下一目标码元并且生成置换集合以包括该下一码元的不同值。这一过程被重复,直至该码元序列中的所有码元已被测试。

[0163] 在第二示例中,在确定码元序列中存在码元差错之后,接收机可并发地生成和解码该码元序列的所有可能置换,其中每一置换只包括相对于初始码元序列被修改的一个码元并且每一置换不同于所有其他置换。接收机可并发地将从该码元序列的每一可能置换解码出的EDC值与预期EDC值相比较。在这一示例中,可需要最大数目的解码器。在这一示例中,纠错可以在最小化延迟的情况下执行。

[0164] 在第三示例中,在确定码元序列中存在码元差错之后,接收机可生成该码元序列的置换集合,其中该置换集合的成员包括该码元序列中的不同码元的经修改版本。即,第一

置换集合中的每一置换例如包括被修改了固定值 x (使用模 (r) 加法) 的一个码元, 第二置换集合中的每一置换包括被修改了固定值 $x+1$ 的一个码元, 以此类推。每一置换集合可在生成下一置换集合之前被测试。接收机可以并发地解码该置换集合的所有成员。

[0165] 在一些实例中, 在多线接口上传送的EDC的预定义值具有基于为对多个连接器上的数据传输进行编码所定义的每码元状态总数来确定的固定长度。EDC可被提供作为多个比特中的预定义数目的最低有效位。在多线接口上传送的EDC的预定义值可基于用来编码该多个比特的码元总数来确定。

[0166] 时钟可被嵌入在码元序列中的诸码元之间的转变中。

[0167] 多个连接器可具有数个(N 个) 单端连接器。在一个示例中, 可供用于对数据传输进行编码的每码元状态总数是 $2^N - x$, 其中 x 是大于零的整数。在另一示例中, 可供用于对数据传输进行编码的每码元状态总数是 $N! - x$, 其中 x 是大于零的整数。

[0168] 第二流程图2400 (参见图24) 解说了用于解码诸码元的方法的示例。在框2404, 接收机可将码元序列转换成转变数。转变数的每一数位可表示在多个连接器上传送的两个连贯码元之间的转变。

[0169] 在框2406, 接收机可将转变数转换成接收到的多个比特。

[0170] 图25是解说采用处理电路2502的装置2500的硬件实现的示例的概念图。在该示例中, 处理电路2502可被实现成具有由总线2516一般化地表示的总线架构。取决于处理电路2502的具体应用和整体设计约束, 总线2516可包括任何数目的互连总线和桥接器。总线2516将包括一个或多个处理器(由处理器2512一般化地表示) 和计算机可读介质(由处理器可读存储介质2514一般化地表示) 的各种电路链接在一起。总线2516还可链接各种其他电路, 诸如定时源、定时器、外围设备、稳压器、和功率管理电路。收发机或通信接口2518提供用于通过多线接口2520与各种其他装置通信的手段。取决于该装置的本质, 也可提供用户接口(例如, 按键板、显示器、扬声器、话筒、操纵杆)。一个或多个时钟生成电路可以设在处理电路2502内或者由处理电路2502和/或一个或多个处理器2512控制。在一个示例中, 时钟生成电路可包括一个或多个晶体振荡器、一个或多个锁相环设备和/或一个或多个可配置的时钟树。

[0171] 处理器2512负责管理总线2516和一般处理, 包括对存储在处理器可读存储介质2514上的软件的执行。该软件可包括代码和/或指令, 该代码和/或指令在由处理器2512执行时使处理电路2502执行上文针对任何特定装置描述的各种功能。处理器可读存储介质2514可被用于存储由处理器2512在执行软件时操纵的数据。

[0172] 在一种配置中, 处理电路可包括被配置成用于接收表示多线接口2520的多个连接器的信令状态的码元序列的一个或多个模块和/或电路2504以及被配置成用于解码该码元序列的一个或多个解码模块和/或电路2506。例如, 解码模块和/或电路2506可被配置成: 将该码元序列转换成转变数, 其中该转变数的每一数位表示在该多个连接器上传送的两个连贯码元之间的转变; 以及将转变数转换成接收到的多个比特。处理电路可包括被配置用于的一个或多个模块和/或电路2504。处理电路可包括被配置成用于基于接收到的多个比特中的EDC的值来确定该码元序列中是否存在码元差错的一个或多个检错模块和/或电路2508。处理电路可包括被配置用于纠正码元差错的一个或多个纠错模块和/或电路2510。在一个示例中, 在确定该码元序列中存在码元差错时, 纠错模块和/或电路2510可被启用。纠

错模块和/或电路2510可被配置成：生成该码元序列的一个或多个置换，其中每一置换包括与该码元序列中的对应码元不同且与其他置换中的对应码元不同的一个码元；解码该一个或多个置换；以及标识产生与预期EDC值相匹配的经解码EDC值的置换。预期EDC值可对应于在多线接口上传送以使得能够在接收机处检测多达两个码元差错的EDC的预定义值。

[0173] 本领域技术人员将领会，结合本文中公开的实施例描述的各种解说性逻辑块、模块、电路、和算法步骤可被实现为电子硬件、计算机软件或两者的组合。为清楚地解说硬件与软件的这一可互换性，各种解说性组件、块、模块、电路、以及步骤在上面是以其功能性的形式作一般化描述的。此类功能性是被实现为硬件还是软件取决于具体应用和施加于整体系统的设计约束。

[0174] 本文所述的本发明的各种特征可实现在不同系统中而不脱离本发明。应注意，以上实施例仅是示例，且不应被解释成限定本发明。这些实施例的描述旨在是说明性的，而并非旨在限定权利要求的范围。由此，本发明的教导可以现成地应用于其他类型的装置，并且许多替换、修改和变形对于本领域技术人员将是显而易见的。

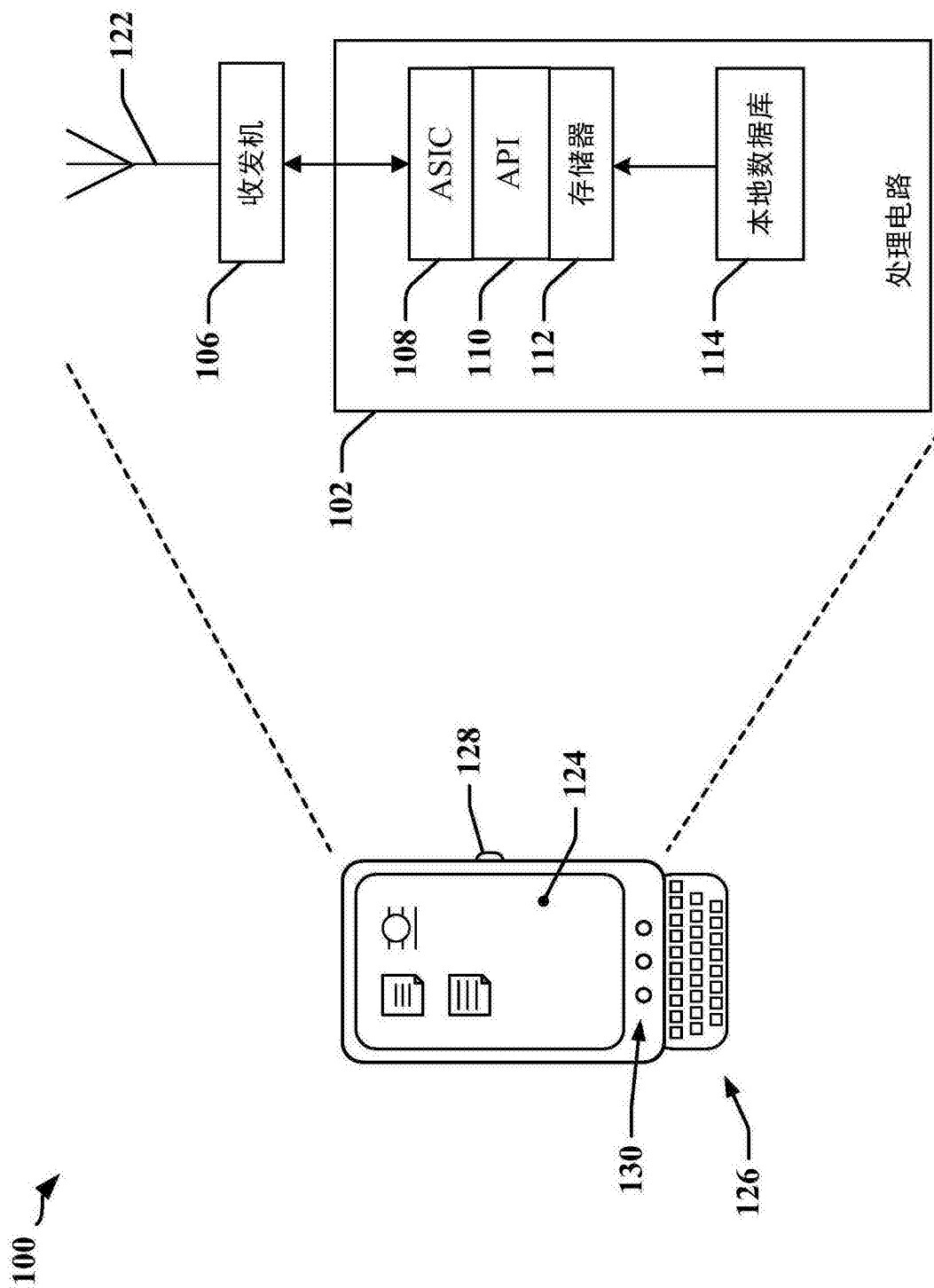


图1

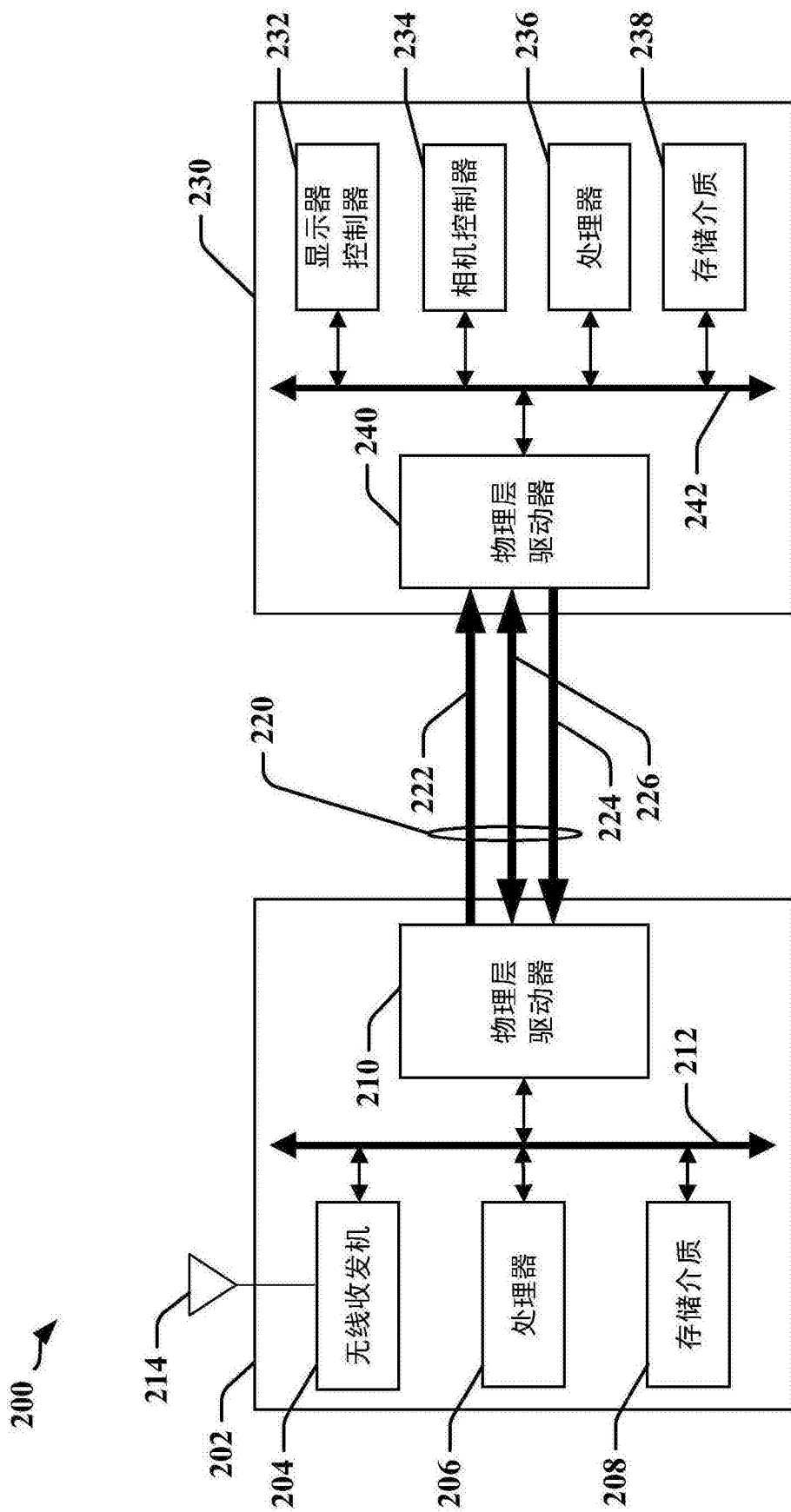


图2

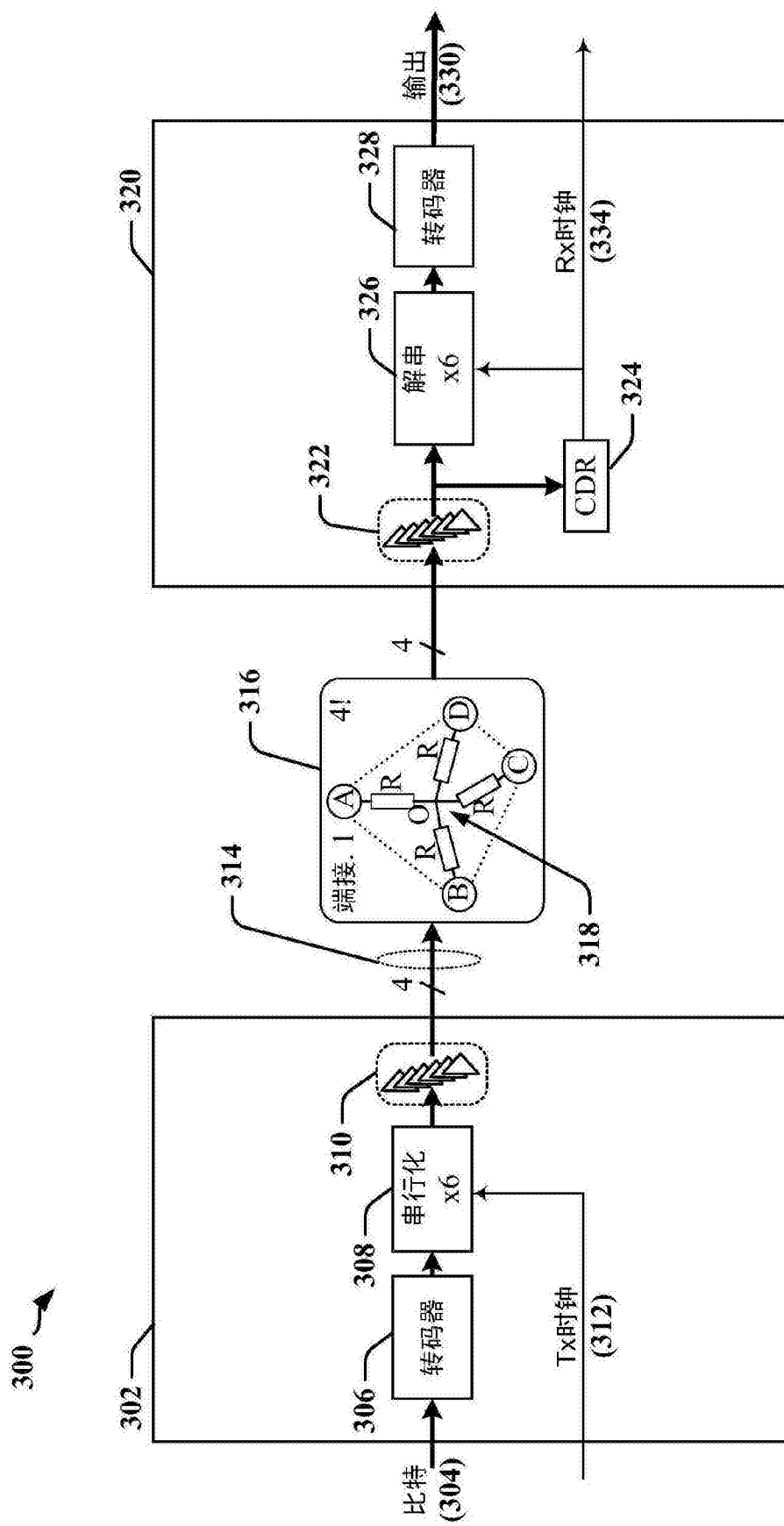


图3

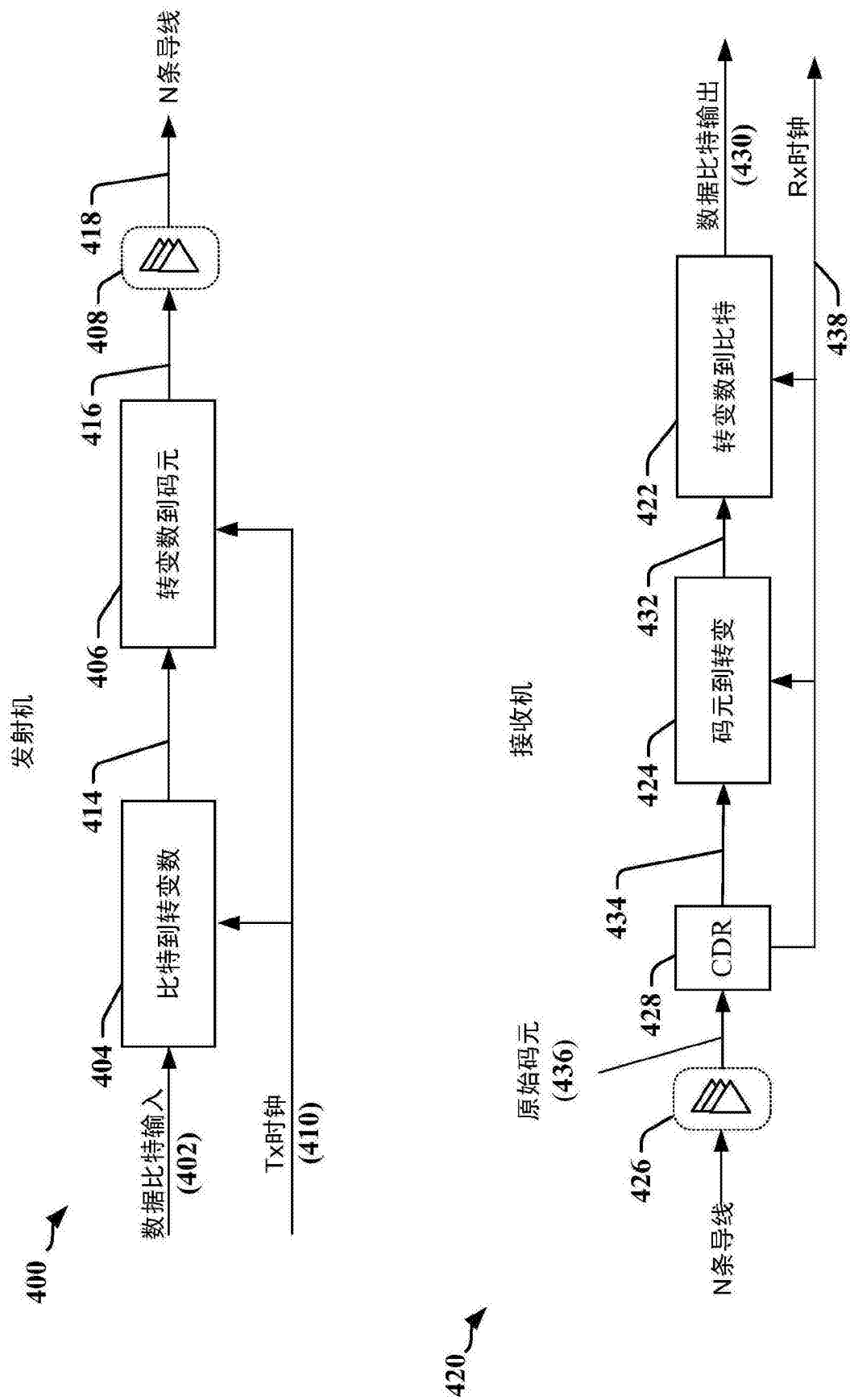


图4

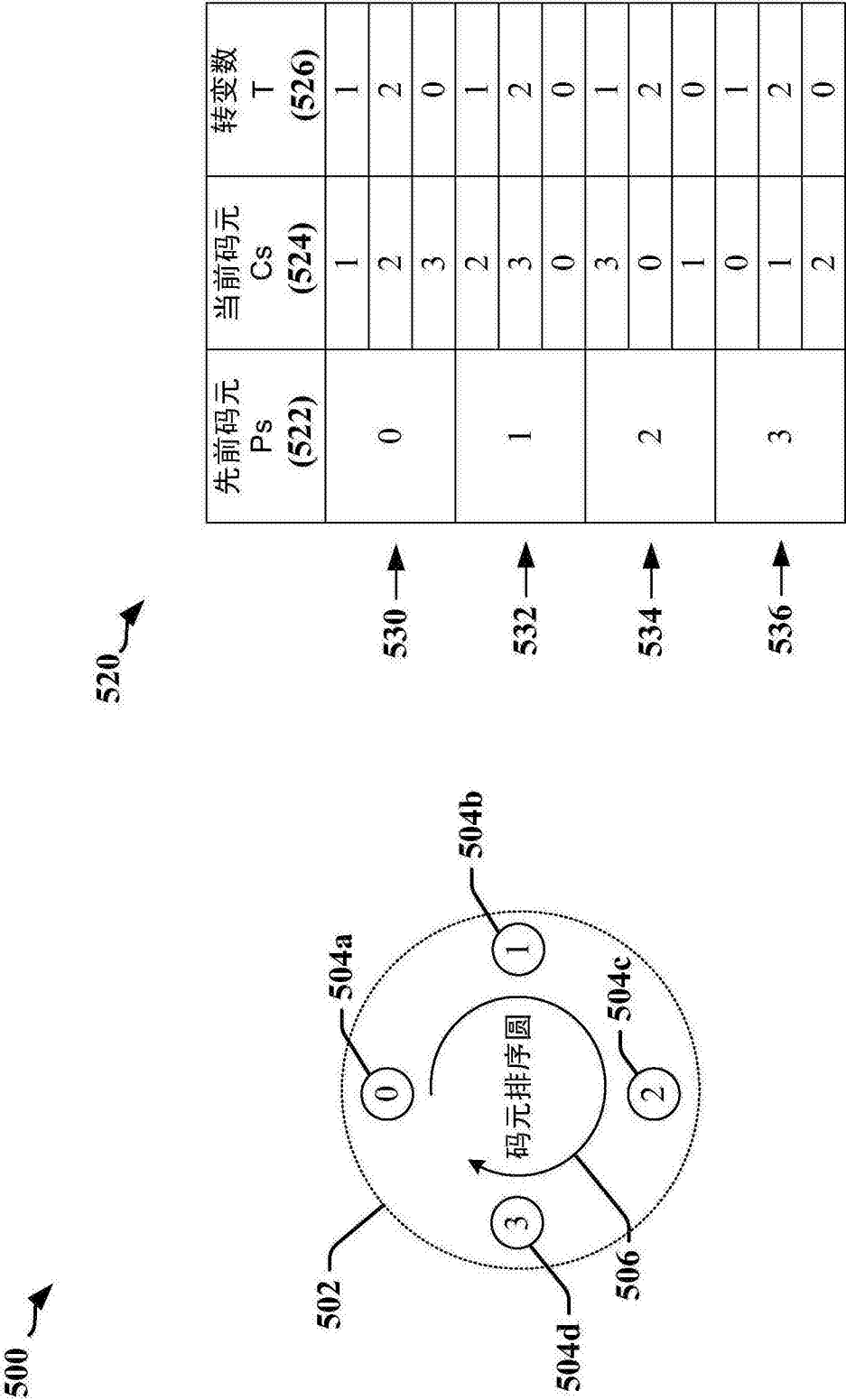


图5

600 ↗

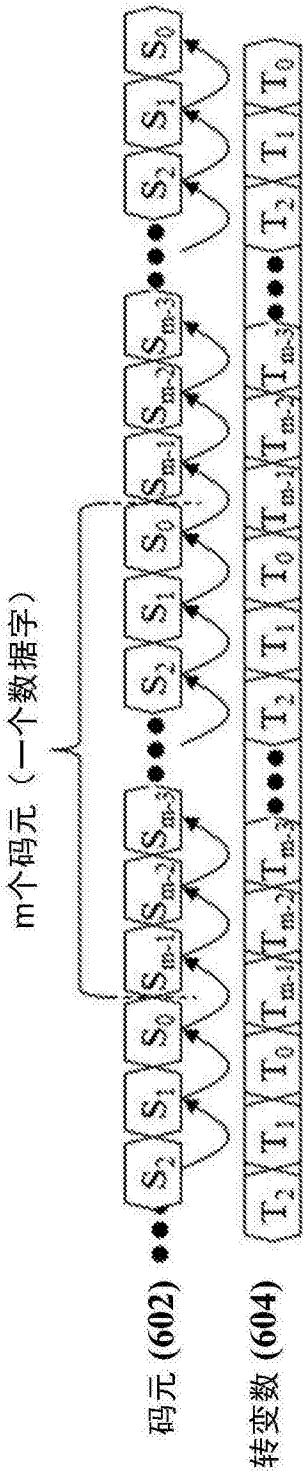


图6

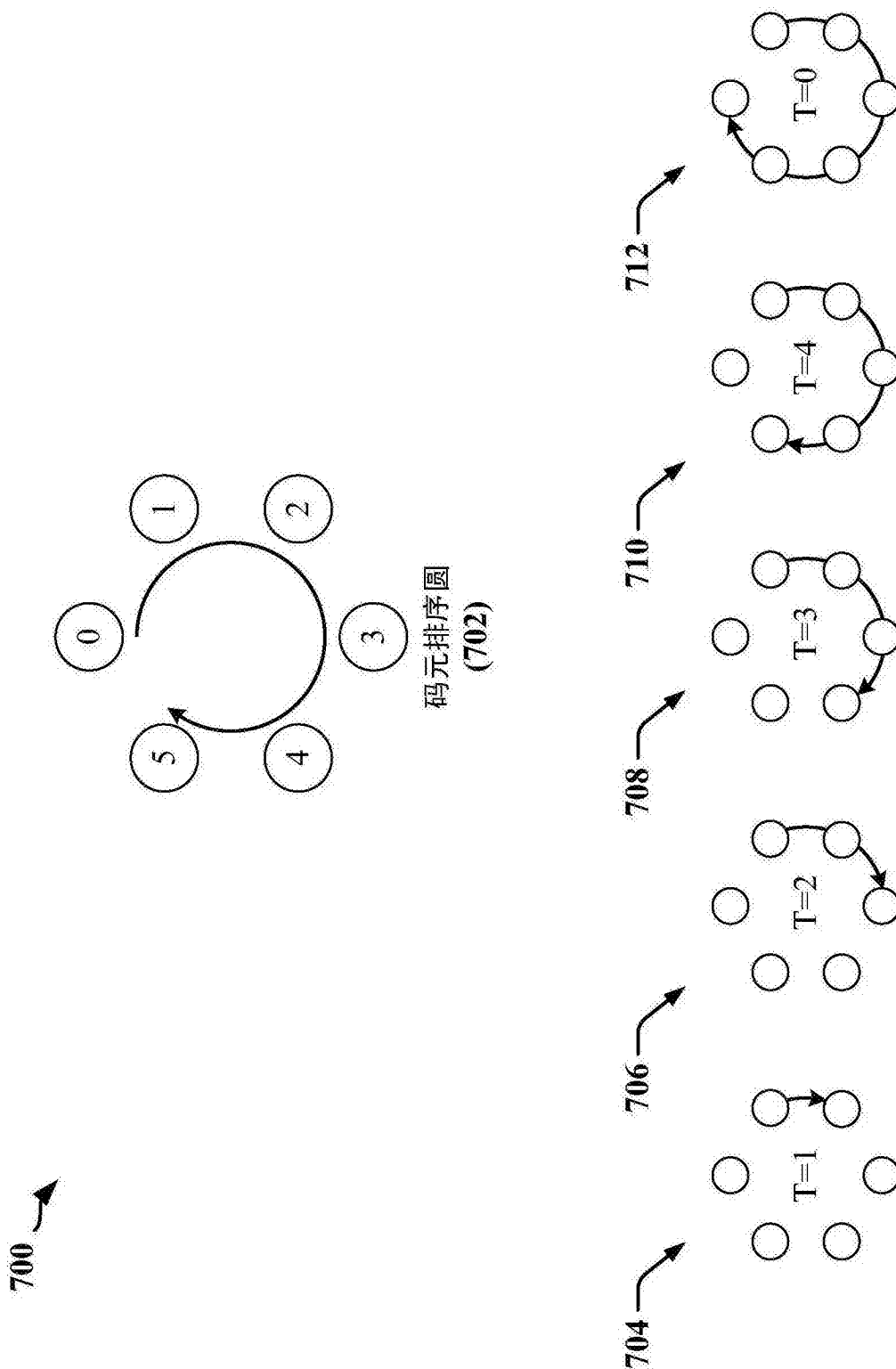


图7

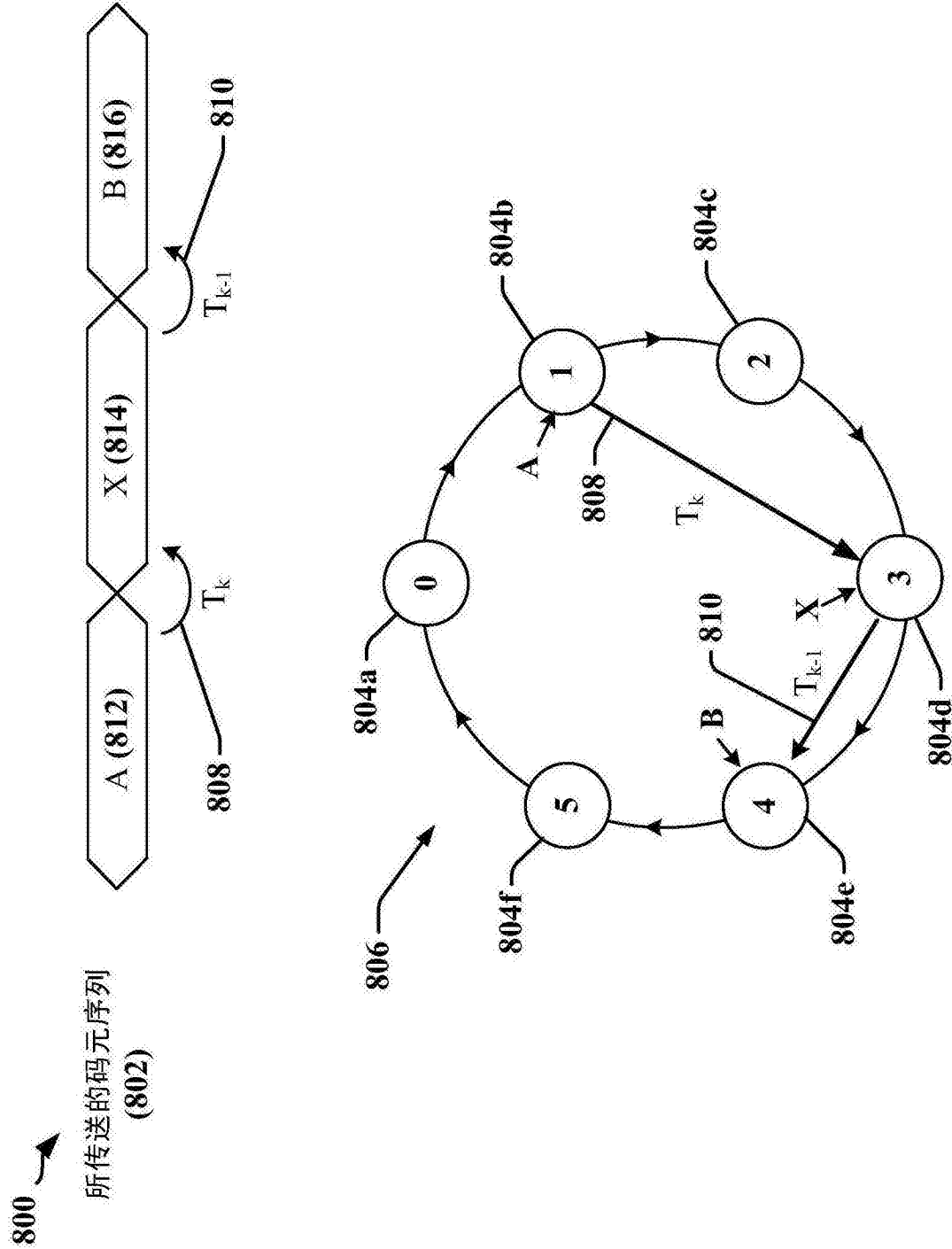


图8

900 ↗

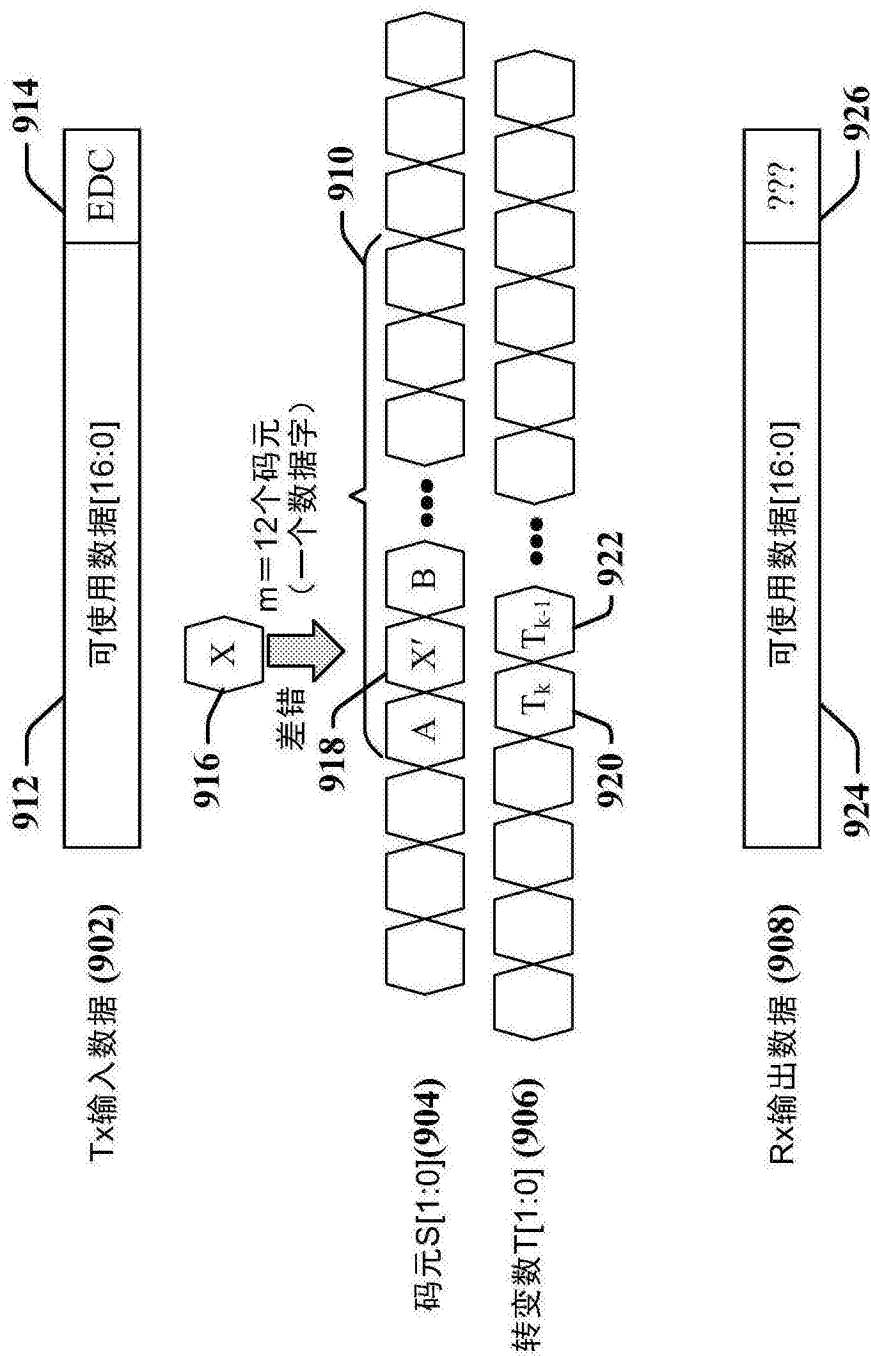


图9

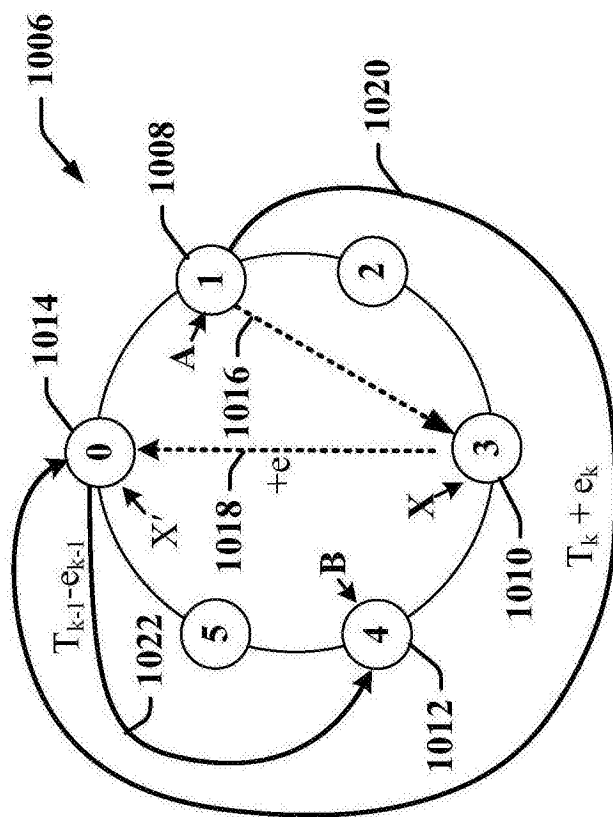
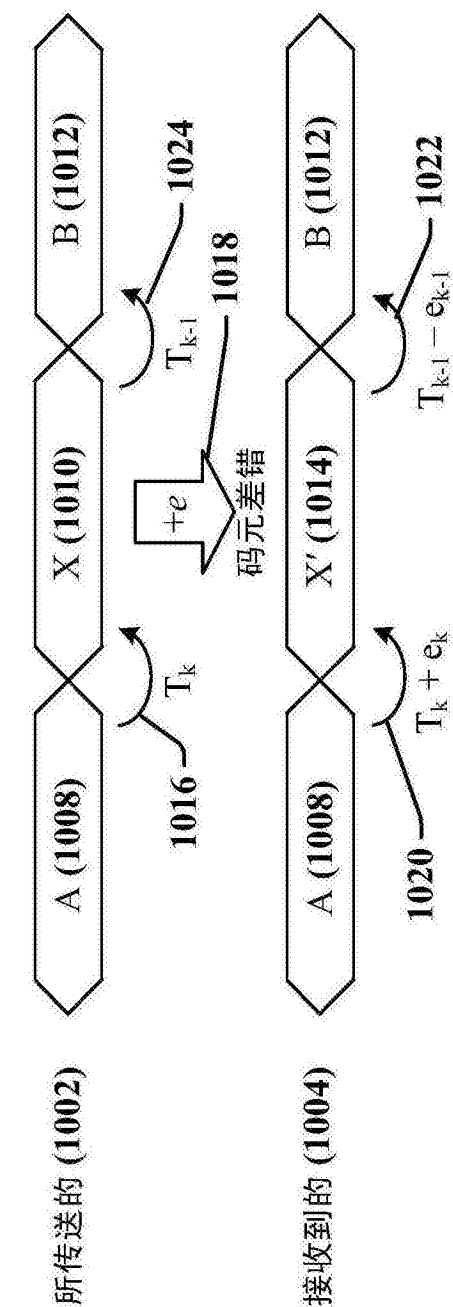


图10

$r = 3$		
n	r^n 十六进制	r^n 二进制
0	000001	00000000000000000000000000000001
1	000003	00000000000000000000000000000011
2	000009	000000000000000000000000000001001
3	00001B	0000000000000000000000000000011011
4	000051	000000000000000000000000000001010001
5	0000F3	0000000000000000000000000000011110011
6	0002D9	000000000000000000000000000001011011001
7	00088B	000000000000000000000000000001000100001011
8	0019A1	000000000000000000000000000001001101000001
9	004CE3	0000000000000000000000000000010011100011
10	00E6A9	0000000000000000000000000000010011010101001
11	02B3FB	00000001010110011111111011
12	081BF1	0000100000001101111110001
13	1853D3	0001100000101000111010011
14	48FB79	010010001111101101111001
15	DAF26B	1101101011110010001101011

1100 ↗

$r = 5$		
n	r^n 十六进制	r^n 二进制
0	000000001	0000000000000000000000000000000000000001
1	000000005	000000000000000000000000000000000000000101
2	000000019	00000000000000000000000000000000000000011001
3	00000007D	000000000000000000000000000000000000000111101
4	000000271	0000000000000000000000000000000000000001001110001
5	0000000C35	0000000000000000000000000000000000000001101000010101
6	0000003D09	0000000000000000000000000000000000000001110100001001
7	00001312D	00000000000000000000000000000000000000010011000100101101
8	00005F5E1	000000000000000000000000000000000000000101111010111000001
9	0001DCD65	00000000000000000000000000000000000000011011100110101100101
10	0009502F9	00000000000000000000000000000000000000010010101000000101111001
11	002E90EDD	00000000000000000000000000000000000000011010010000111011011101
12	00E8D4A51	0000000000000000000000000000000000000001101000110101001010001010001
13	048C27395	00000010010000110000000000000000000000011000111001010101
14	16BCC41E9	000101101011110011000100000000000000000111010001111010001
15	71AFD498D	01110000110101111101010010001000110001101

1102 ↗

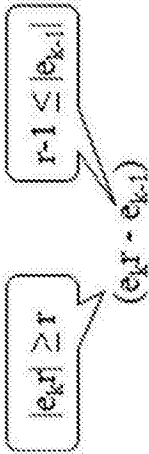
1104 ↗

1106 ↗

图11

十进制	十六进制	二进制	$\{e^k, e^{k+1}\}$ 列表
-21	EB	11101011	-4, 1
-18	EE	11101110	-3, 3
-17	EF	11101111	-3, 2
-16	F0	11110000	-4, -4
-13	F3	11110011	-2, 3
-12	F4	11110100	-3, -3
-9	F7	11110111	-1, 4
-8	F8	11111000	-2, -2
-7	F9	11111001	-2, -3
-4	FC	11111100	-1, -1
-3	FD	11111101	-1, -2
3	03	00000011	1, 2
4	04	00000100	1, 1
7	07	00000111	2, 3
8	08	00001000	2, 2
9	09	00001001	1, -4
13	0D	00001101	2, 3
16	10	00010000	4, 4
17	11	00010001	3, -2
18	12	00010010	3, -3
21	15	00010101	4, -1

1200



差错系数($e_k r - e_{k-1}$)永不变成0

图12

1300 ↗

$(2^n r - 2^n) = 2^n (r-1)$

将r-1左移n-1位

用0替换r的LSB，例如
r=3=00011₂ → r-1=00010₂
r=5=00101₂ → r-1=00100₂
r=7=00111₂ → r-1=00110₂
r=15=01111₂ → r-1=01110₂
r=23=10111₂ → r-1=10110₂

r	r-1	对于e _k 或e _{k-1} 的最长2 ⁿ	(e _k r-e _{k-1}) 的最长非零 LSB
3	00011 ₂	2	00010 ₂
5	00101 ₂	4	00100 ₂
7	00111 ₂	4	001100 ₂
9	01001 ₂	8	0100000 ₂
11	01011 ₂	8	01010000 ₂
15	01111 ₂	8	01110000 ₂
23	10111 ₂	16	101100000 ₂

十进制	十六进制	二进制	{e ^k , e ^{k-1} }列表
-21	EB	111101011	-4, 1
-18	EE	111101110	-3, 3
-17	EF	111101111	-3, 2
-16	F0	111100000	-4, -4
-13	F3	111100111	-2, 3
-12	F4	111101000	-3, -3
-9	F7	111101111	-1, 4
-8	F8	111110000	-2, -2
-7	F9	111110001	-2, -3
-4	FC	111111100	-1, -1
-3	FD	111111101	-1, -2
3	03	00000011	1, 2
4	04	00000100	1, 1
7	07	00000111	2, 3
8	08	00001000	2, 2
9	09	00001001	1, -4
13	0D	00001101	2, -3
16	10	00010000	4, 4
17	11	00010001	3, -2
18	12	00010010	3, 3
21	15	00010101	4, -1

图13

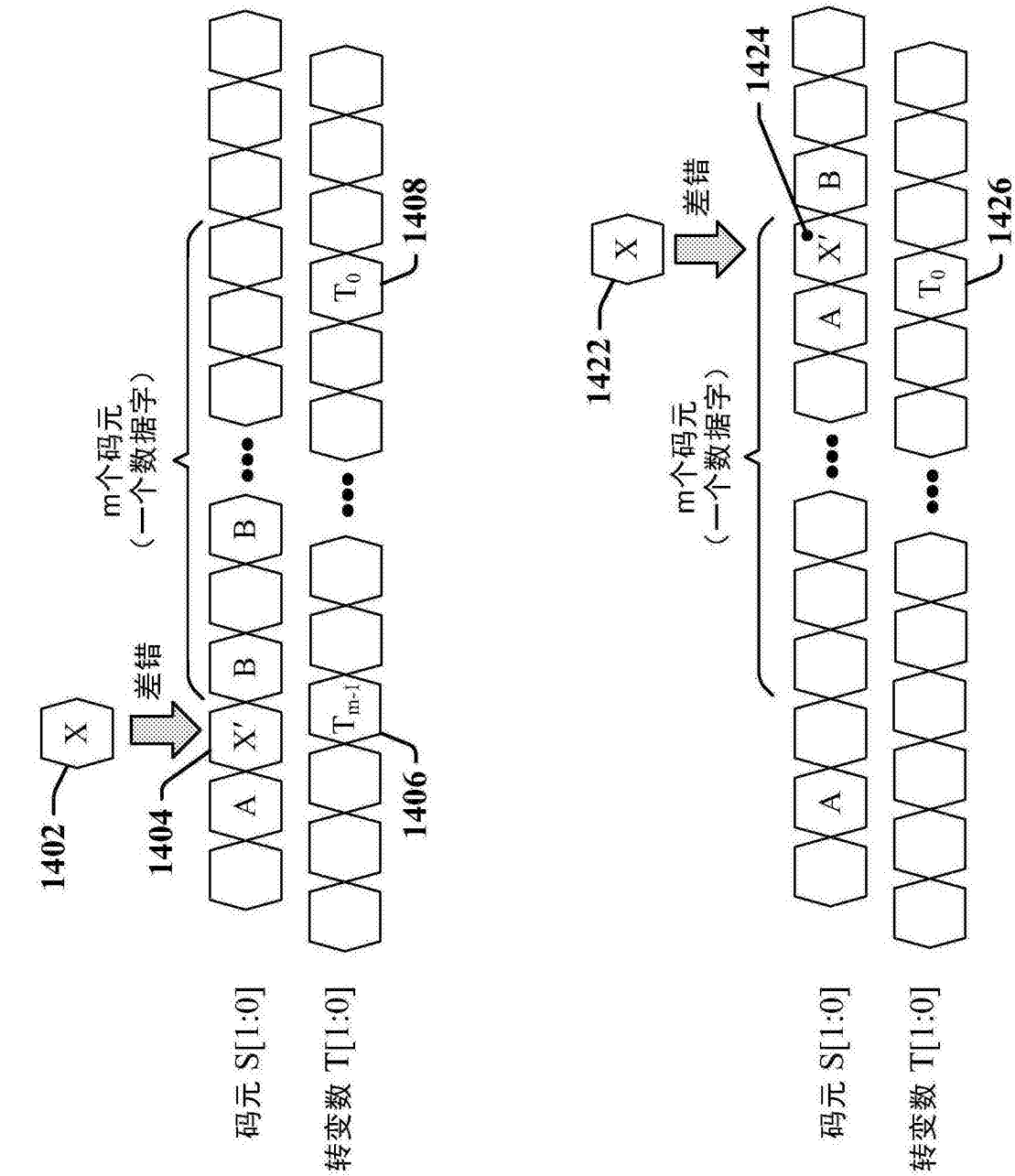


图14

1500 ↗

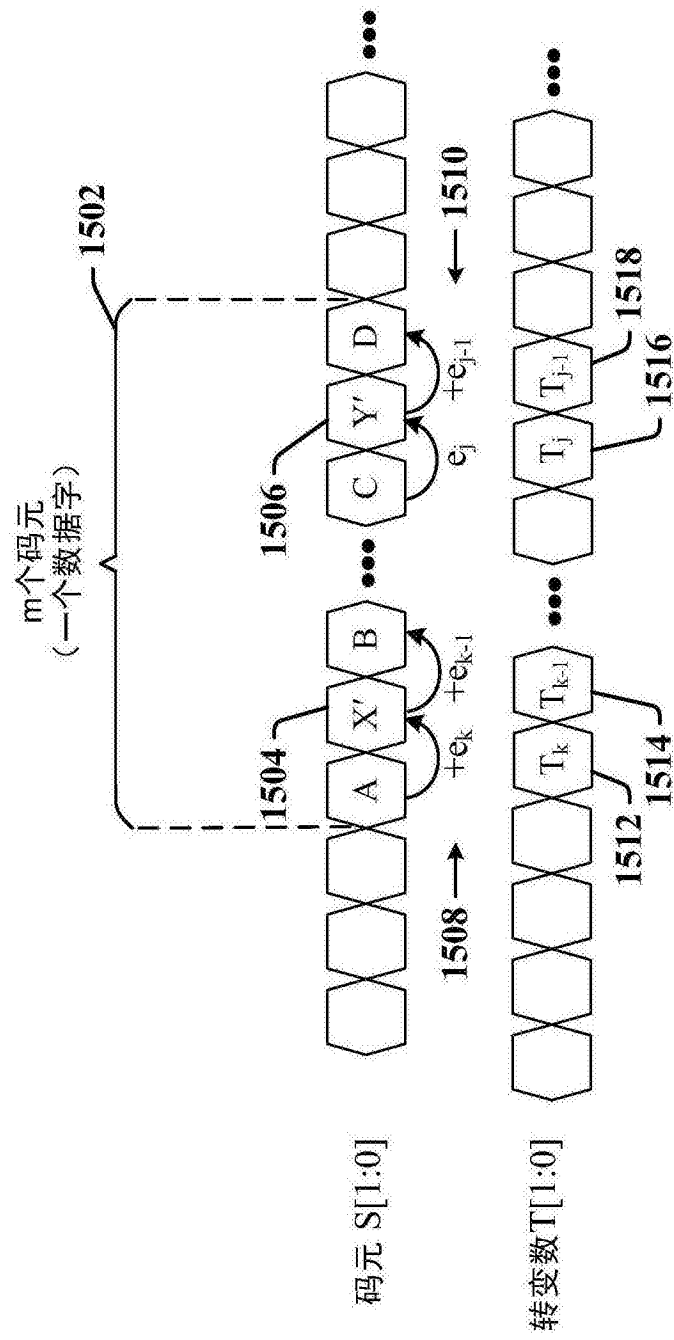


图15

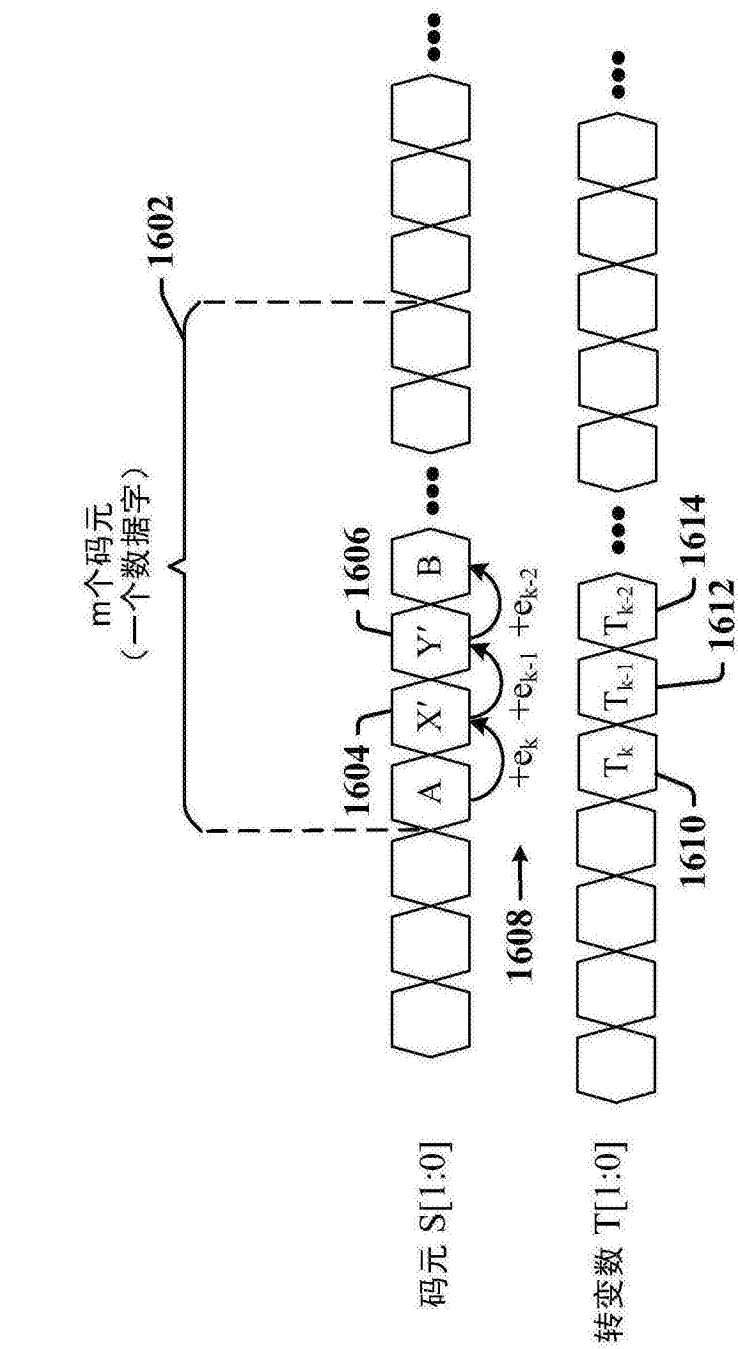


图16

1700

	r											
m	3	5	7	9	11	13	15	17	19	21	23	
	3	5	6	7	5	6	8	9	8	7	6	单码元差错
4	8	10	12	13	12	14	15	18	16	16	14	两码元差错检测
5	8	10	12	13	12	14	15	18	16	16	15	
6	8	10	12	14	13	14	15	18	16	16	15	
7	8	10	12	14	13	14	15	20	16	16	15	
8	8	11	12	14	14	14	18	20	16	16	15	
9	8	11	12	14	14	14	18	20	16	16	16	
10	8	11	12	14	15	14	18	20	16	16	18	
11	8	11	12	17	15	14	18	20	16	16	18	
12	8	11	12	17	15	14	18	20	16	19	18	
13	8	11	12	17	15	15	18	20	16	19	18	
14	8	11	12	17	15	15	18	20	16	19	18	
15	8	11	14	17	15	15	18	20	16	19	18	
16	8	11	14	17	15	15	18	20	16	19	18	
17	9	11	14	17	15	15	18	20	16	19	18	
18	9	11	14	17	15	15	18	20	16	19	18	
19	9	11	14	17	15	15	18	20	16	19	18	
20	9	11	14	17	15	15	18	20	16	19	18	
21	9	11	14	17	15	16	18	20	16	19	18	
22	9	11	14	17	15	16	18	20	16	19	18	
23	9	11	14	17	15	16	18	20	16	19	18	
24	9	11	14	17	15	16	18	20	16	19	18	
25	9	11	14	17	15	16	18	20	16	19	18	
26	9	11	14	17	15	16	18	20	16	19	18	
27	9	11	14	17	15	16	18	20	17	19	18	
28	9	11	14	17	15	16	18	20	17	19	18	
29	9	11	14	17	15	16	18	20	17	19	18	
30	9	11	14	17	15	16	18	20	17	19	18	
31	9	11	14	17	15	16	18	20	17	19	18	
32	9	11	14	17	15	16	18	20	17	19	18	

图17

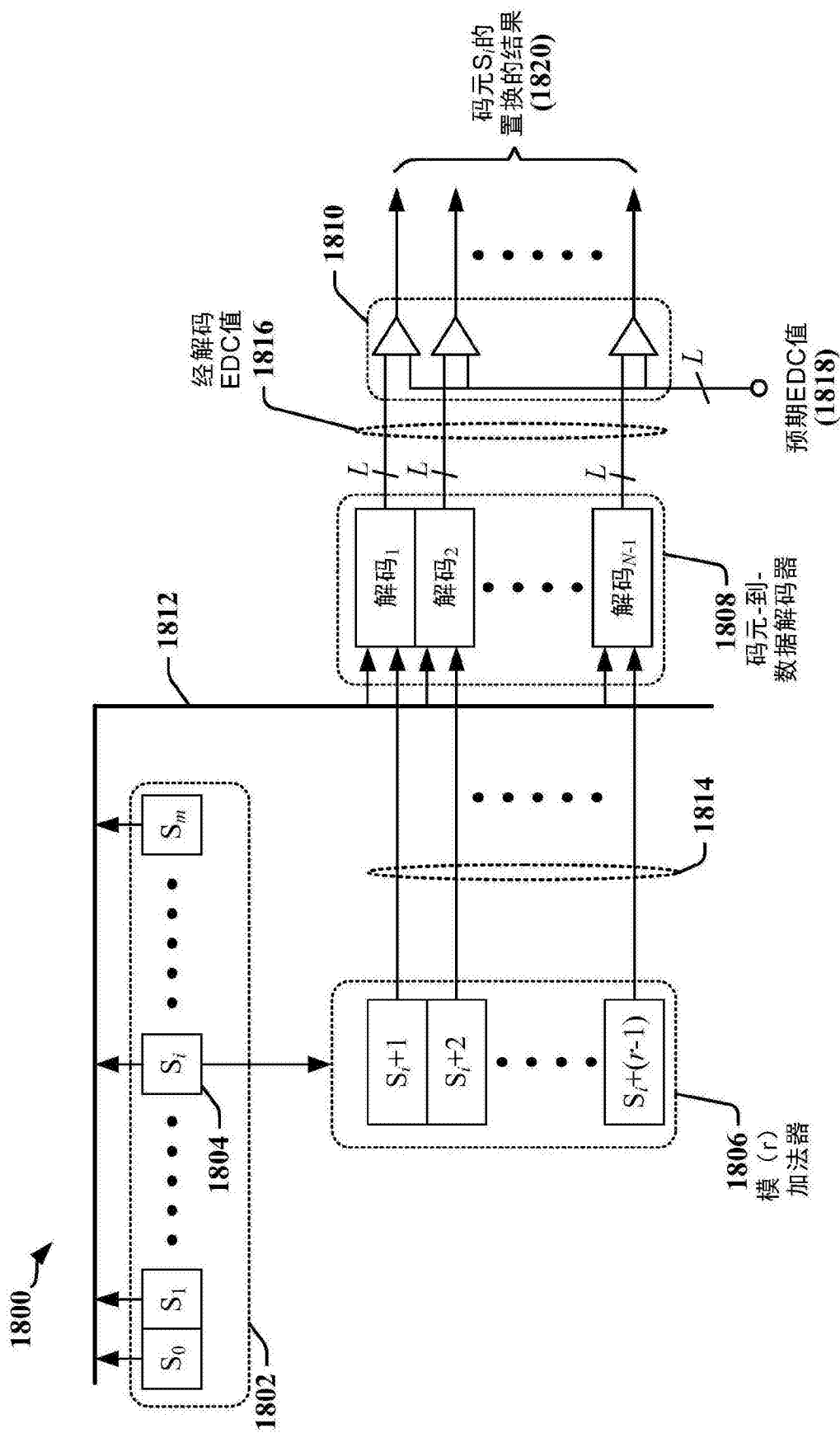


图18

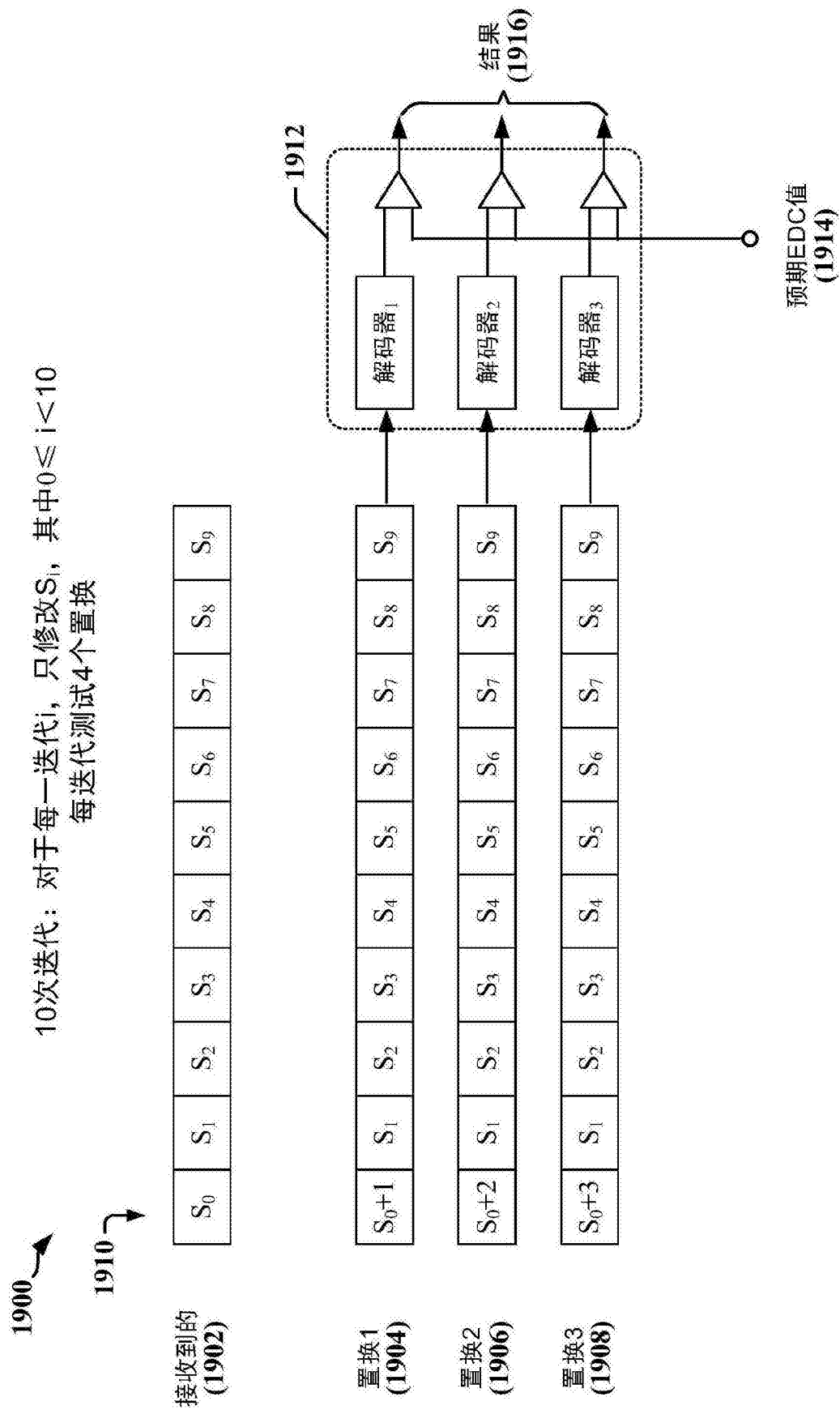


图19

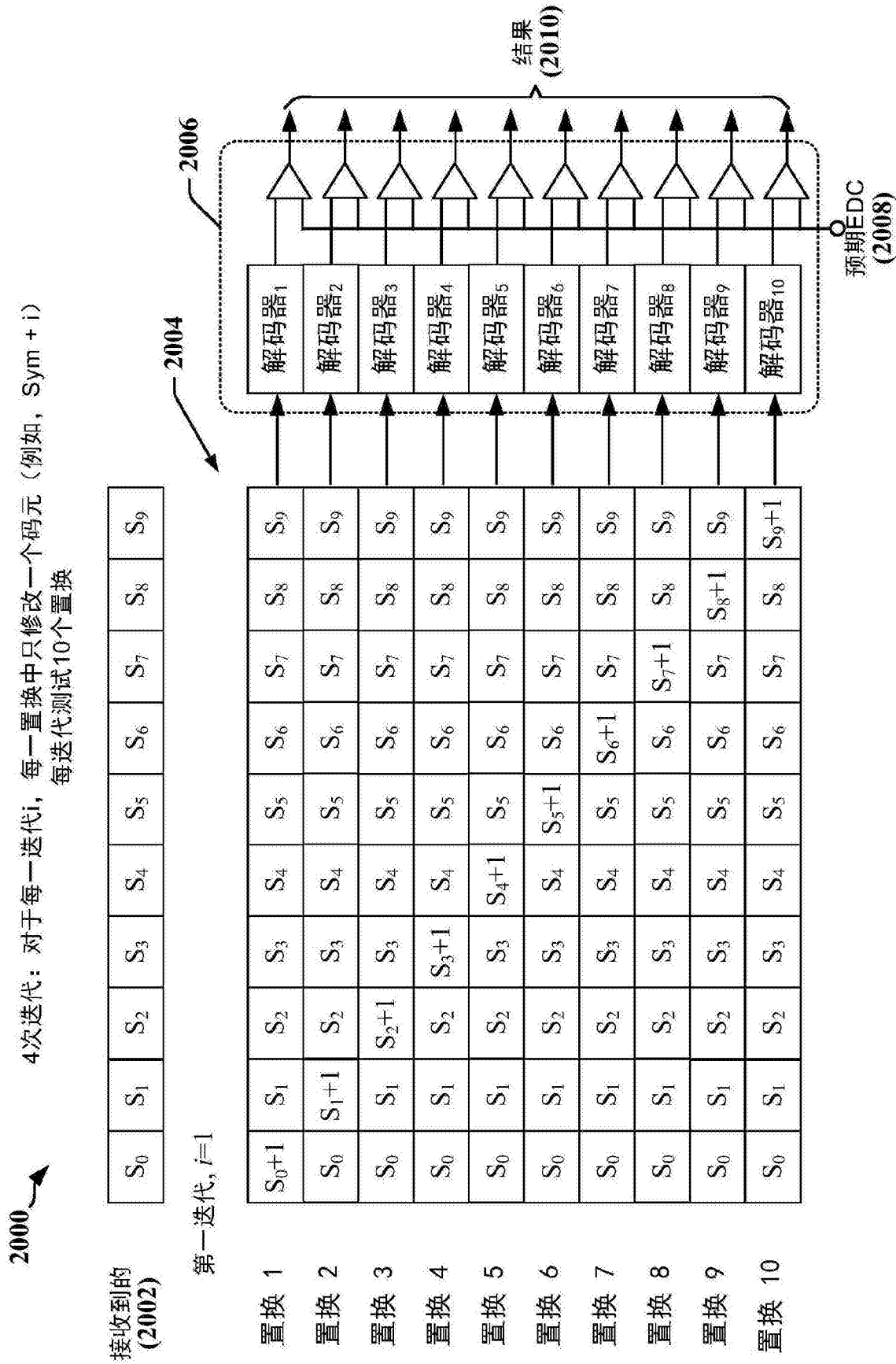


图20

2100 ↗

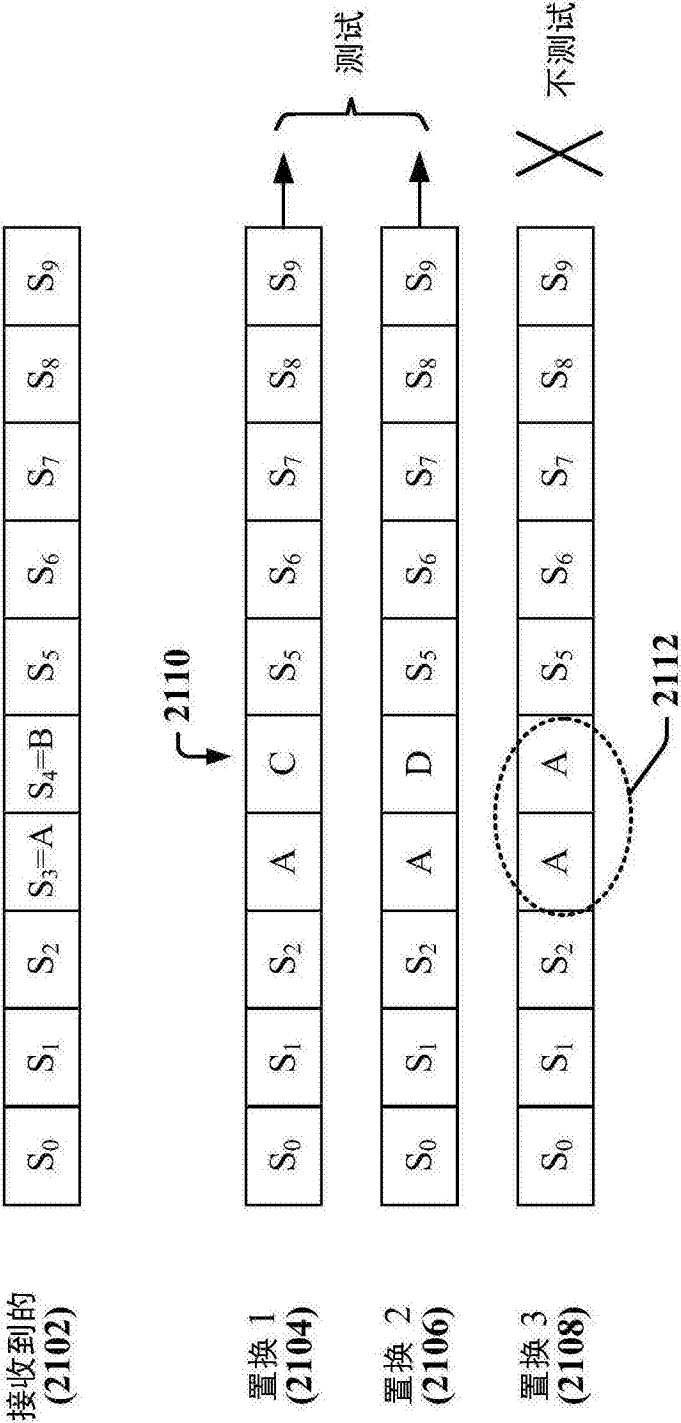


图21

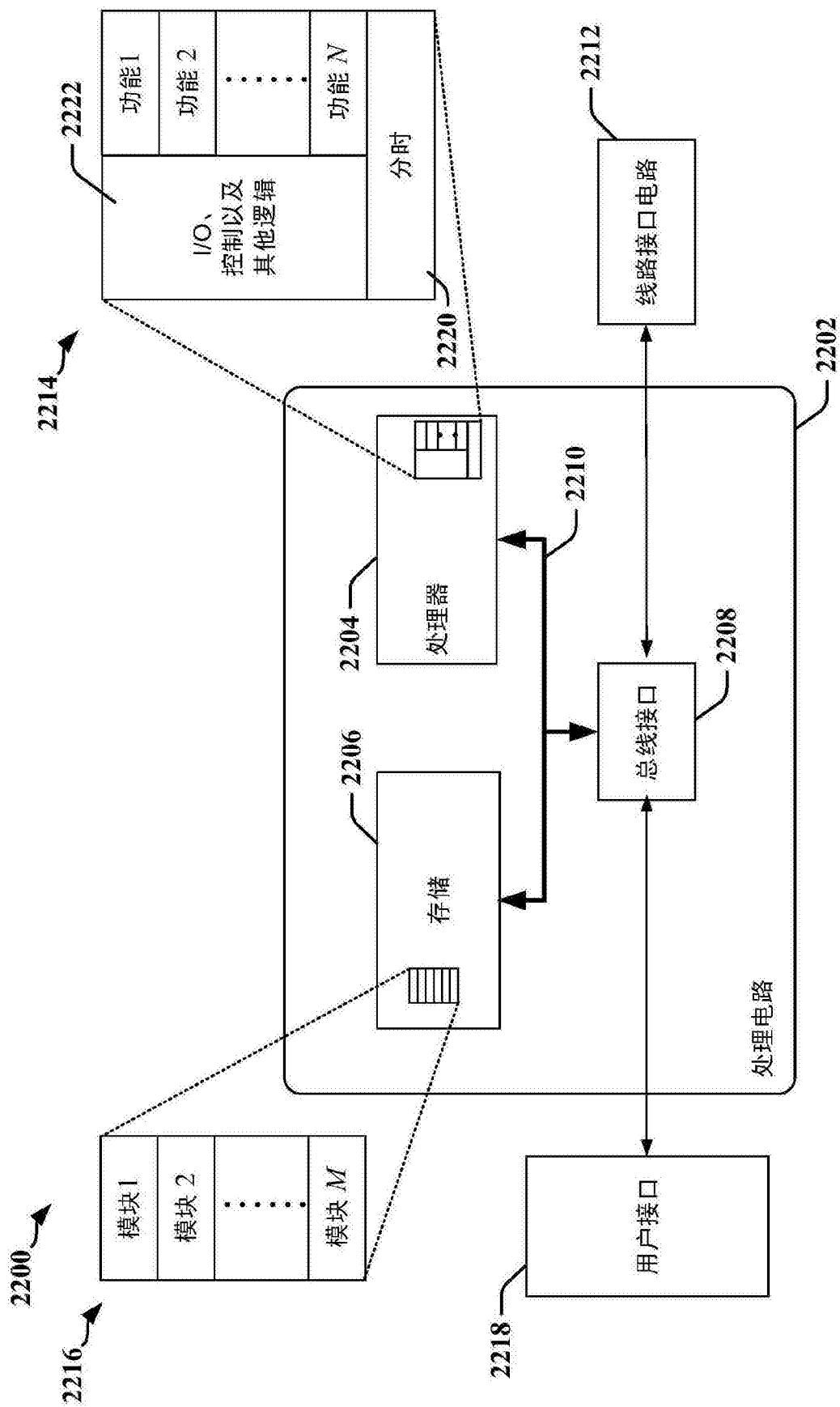


图22

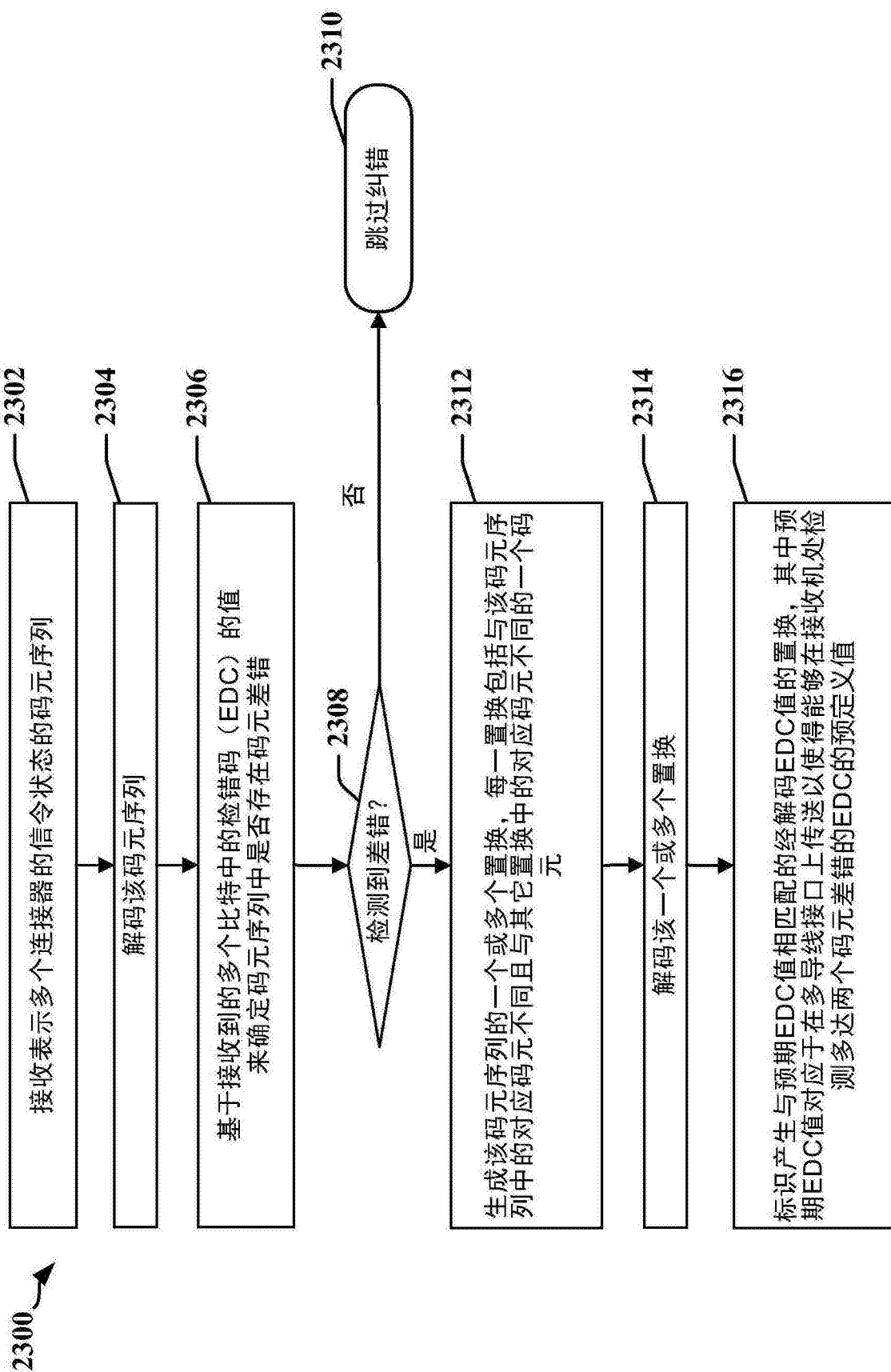


图23

2400 ↗

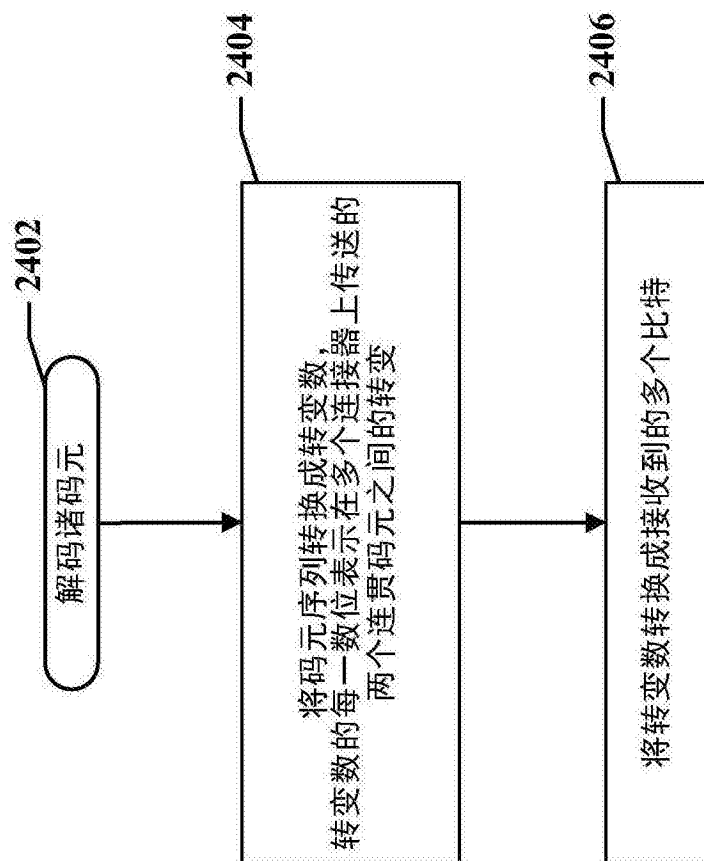


图24

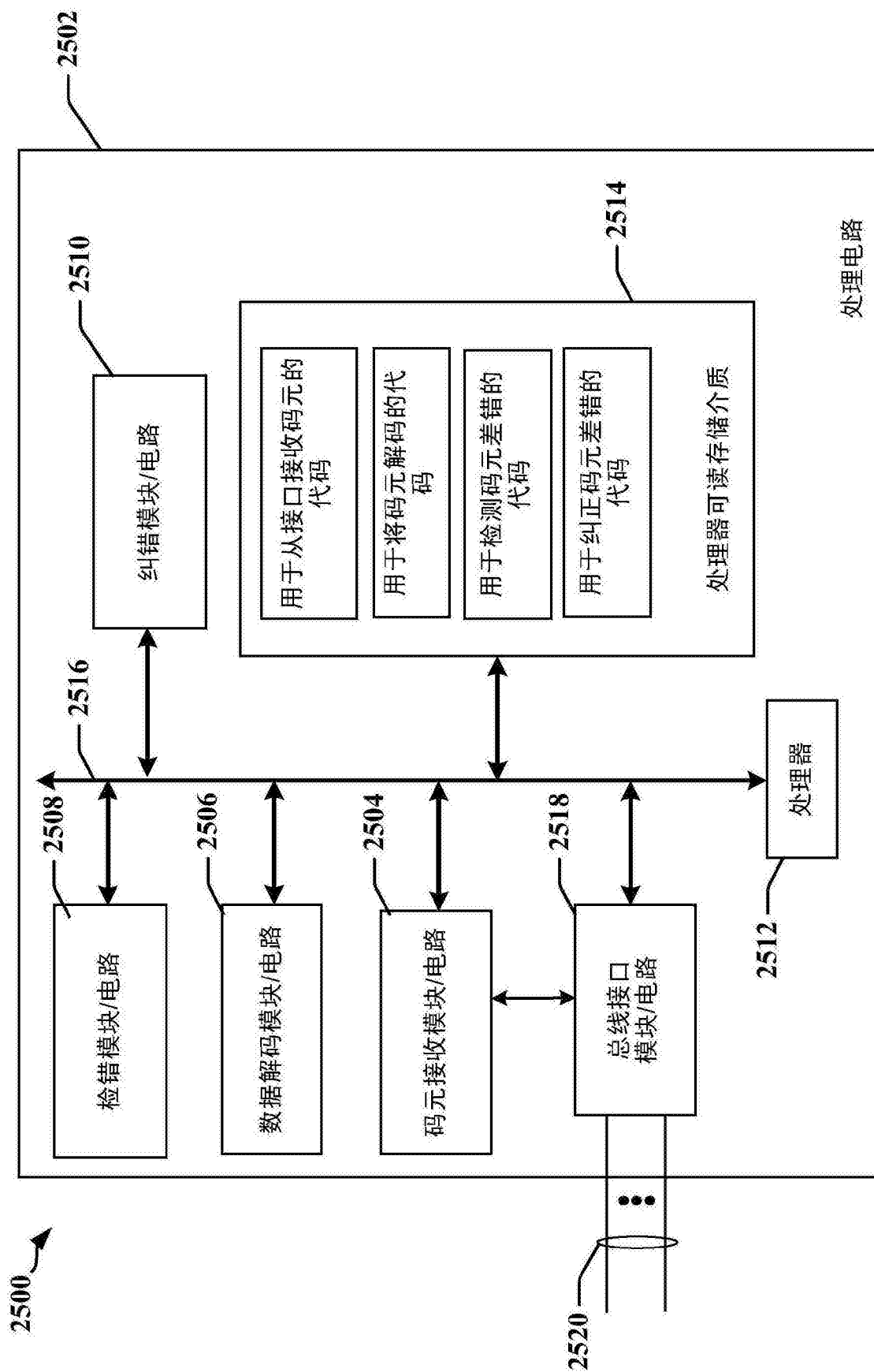


图25