



(12)发明专利申请

(10)申请公布号 CN 105830045 A

(43)申请公布日 2016.08.03

(21)申请号 201480068720.X

(22)申请日 2014.12.17

(30)优先权数据

61/917,895 2013.12.18 US

14/572,680 2014.12.16 US

(85)PCT国际申请进入国家阶段日

2016.06.16

(86)PCT国际申请的申请数据

PCT/US2014/070935 2014.12.17

(87)PCT国际申请的公布数据

W02015/095382 EN 2015.06.25

(71)申请人 高通股份有限公司

地址 美国加利福尼亚州

(72)发明人 S·森戈库

(74)专利代理机构 上海专利商标事务所有限公司 31100

代理人 唐杰敏

(51)Int.Cl.

G06F 13/42(2006.01)

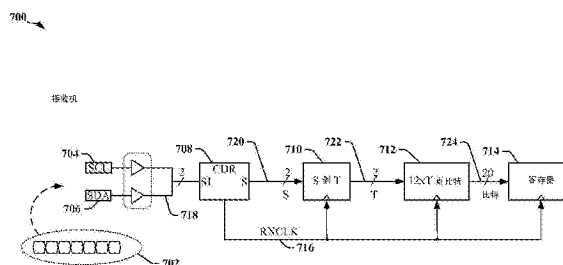
权利要求书3页 说明书16页 附图21页

(54)发明名称

仅用接收机时钟来进行的CCIE接收机逻辑寄存器写入

(57)摘要

描述了提供一种机制的方法、装置、以及计算机程序产品,该机制使得在没有自由运行的时钟的情况下数据能被写入从动设备的寄存器而同时又促成从动设备的高效休眠和苏醒机制。接收机设备可在共享总线上接收多个码元,提取嵌入在该多个码元的码元到码元转变中的时钟信号,将该多个码元转换成转变数,将该转变数转换成数据比特,以及仅使用该接收时钟信号来将这些数据比特的至少一部分存储到一个或多个寄存器中。该接收机设备可在检测到该时钟信号的第一循环之际启动递减计数器,在该递减计数器到达预定义值时触发标记,以及使用该标记来将这些数据比特的至少一部分存储到寄存器中。



1. 一种由接收机设备执行的方法,包括:
在共享总线上接收多个码元;
提取嵌入在所述多个码元的码元到码元转变中的时钟信号;
将所述多个码元转换成转变数;
将所述转变数转换成数据比特;以及
仅使用所述时钟信号来将所述数据比特的至少一部分存储到一个或多个寄存器中。
2. 如权利要求1所述的方法,其特征在于,所述码元每时钟循环发生转变,以使得没有两个顺序码元具有相同的值。
3. 如权利要求1所述的方法,其特征在于,所述接收机设备独立地进入休眠模式而不通知耦合至所述共享总线的任何其他设备。
4. 如权利要求1所述的方法,其特征在于,所述接收机设备在不使用本地自由运行的时钟的情况下接收所述数据比特并且将所述数据比特的至少一部分写入所述一个或多个寄存器。
5. 如权利要求1所述的方法,其特征在于,所述接收机设备在所述接收机处于休眠模式之时接收所述数据比特并且将所述数据比特的至少一部分写入所述一个或多个寄存器。
6. 如权利要求1所述的方法,其特征在于,所述转变数被表达为三进制数。
7. 如权利要求1所述的方法,其特征在于,所述转变数是12数位三进制数。
8. 如权利要求1所述的方法,其特征在于,所述共享总线是相机控制接口扩展(CCIe)总线。
9. 如权利要求1所述的方法,其特征在于,所述数据比特的至少一部分通过以下操作被写入所述一个或多个寄存器:
在检测到所述时钟信号的第一循环之际启动递减计数器,
在所述递减计数器到达预定义值时触发标记,以及
使用所述标记来将所述数据比特的至少一部分存储到寄存器中。
10. 如权利要求9所述的方法,其特征在于,所述预定义值在所述时钟信号的最末时钟循环到达时发生。
11. 如权利要求1所述的方法,其特征在于,
所述转变数在所述时钟信号的倒数第二个时钟循环与最末时钟循环之间被转换成所述数据比特;以及
所述数据比特的所述至少一部分在所述时钟信号的最末时钟循环处被存储到寄存器中。
12. 一种接收机设备,包括:
总线接口,所述总线接口被适配成将所述接收机设备耦合至共享总线以接收多个码元;
一个或多个寄存器;以及
接收机电路,所述接收机电路耦合到所述总线接口并且被配置成:
提取嵌入在所述多个码元的码元到码元转变中的时钟信号;
将所述多个码元转换成转变数;
将所述转变数转换成数据比特;以及

仅使用所述时钟信号来将所述数据比特的至少一部分存储到所述一个或多个寄存器中。

13. 如权利要求12所述的接收机设备,其特征在于,所述码元每时钟循环发生转变,以使得没有两个顺序码元具有相同的值。

14. 如权利要求12所述的接收机设备,其特征在于,所述接收机设备被适配成独立地进入休眠模式而不通知耦合至所述共享总线的任何其他设备。

15. 如权利要求12所述的接收机设备,其特征在于,所述接收机设备被适配成在不使用本地自由运行的时钟的情况下接收所述数据比特并且将所述数据比特的至少一部分写入所述一个或多个寄存器。

16. 如权利要求12所述的接收机设备,其特征在于,所述接收机设备被适配成在所述接收机处于休眠模式时接收所述数据比特并且将所述数据比特的至少一部分写入所述一个或多个寄存器。

17. 如权利要求12所述的接收机设备,其特征在于,所述转变数被表达为三进制数。

18. 如权利要求12所述的接收机设备,其特征在于,所述转变数是12数位三进制数。

19. 如权利要求12所述的接收机设备,其特征在于,所述共享总线是相机控制接口扩展(CCIe)总线。

20. 如权利要求12所述的接收机设备,其特征在于,进一步包括:

递减计数器,其中所述接收机设备被适配成通过以下操作来接收所述数据比特并且将所述数据比特的至少一部分写入所述一个或多个寄存器:

在检测到所述时钟信号的第一循环之际启动所述递减计数器,

在所述递减计数器到达预定义值时触发标记,

使用所述标记来将所述数据比特的至少一部分存储到寄存器中。

21. 如权利要求20所述的接收机设备,其特征在于,所述预定义值在所述时钟信号的最末时钟循环到达时发生。

22. 一种接收机设备,包括:

用于在共享总线上接收多个码元的装置;

用于提取嵌入在所述多个码元的码元到码元转变中的时钟信号的装置;

用于将所述多个码元转换成转变数的装置;

用于将所述转变数转换成数据比特的装置;以及

用于仅使用所述时钟信号来将所述数据比特的至少一部分存储到寄存器中的装置。

23. 如权利要求22所述的接收机设备,其特征在于,所述码元每时钟循环发生转变,以使得没有两个顺序码元具有相同的值。

24. 如权利要求22所述的接收机设备,其特征在于,所述接收机设备被配置成独立地进入休眠模式而不通知耦合至所述共享总线的任何其他设备。

25. 如权利要求22所述的接收机设备,其特征在于,所述用于将所述数据比特的至少一部分存储到所述寄存器中的装置被配置成:

在不使用本地自由运行的时钟的情况下将所述数据比特的至少一部分写入所述一个或多个寄存器。

26. 如权利要求22所述的接收机设备,其特征在于,所述用于将所述数据比特的至少一

部分存储到所述寄存器中的装置被配置成：

在所述接收机处于休眠模式时将所述数据比特的至少一部分写入所述一个或多个寄存器。

27. 如权利要求22所述的接收机设备，其特征在于，所述转变数是12数位三进制数。

28. 如权利要求22所述的接收机设备，其特征在于，所述用于将所述数据比特的至少一部分存储到所述寄存器中的装置被配置成：

在检测到所述时钟信号的第一循环之际启动递减计数器，

在所述递减计数器到达预定义值时触发标记，以及

使用所述标记来将所述数据比特的至少一部分存储到寄存器中，

其中所述预定义值在所述时钟信号的最末时钟循环到达时发生。

29. 如权利要求22所述的接收机设备，其特征在于，所述用于将所述数据比特的至少一部分存储到所述寄存器中的装置被配置成：

在所述时钟信号的倒数第二个时钟循环与最末时钟循环之间将所述转变数转换成所述数据比特；以及

在所述时钟信号的最末时钟循环处将所述数据比特的至少一部分存储到寄存器中。

30. 一种其上存储有指令的非瞬态机器可读存储介质，所述指令在由至少一个处理器执行时使所述至少一个处理器：

在共享总线上接收多个码元；

提取嵌入在所述多个码元的码元到码元转变中的时钟信号；

将所述多个码元转换成转变数；

将所述转变数转换成数据比特；以及

仅使用所述时钟信号来将所述数据比特的至少一部分存储到寄存器中。

仅用接收机时钟来进行的CCle接收机逻辑寄存器写入

[0001] 相关申请的交叉引用

[0002] 本申请要求于2014年12月16日在美国专利商标局提交的美国非临时专利申请No.14/572,680的优先权和权益,该非临时专利申请要求于2013年12月18日在美国专利商标局提交的美国临时申请No.61/917,895的优先权和权益,这两篇申请的全部内容通过援引纳入于此。

[0003] 领域

[0004] 本公开涉及使得能够在共享总线上进行高效操作,尤其涉及通过在经转码传输内嵌入时钟来简化共享总线上的传输和/或接收。

背景技术

[0005] I2C(也称为I²C)是被用于将低速外围设备附连至母板、嵌入式系统、蜂窝电话或其他电子设备的多主控串行单端总线。I2C总线包括具有7位寻址的时钟(SCL)和数据(SDA)线。该总线具有两个给设备的角色:主控方和从动方。主控设备是生成时钟并发起与从动设备的通信的节点。从动设备是接收时钟并在被主控设备寻址时进行响应的节点。I2C总线是多主控总线,这意味着可以有任何数目的主控设备在场。另外,主控角色和从动角色可以在消息之间(在STOP被发送之后)被改变。I2C定义了基本的消息类型,其中每种消息类型始于START(开始)并结束于停止(STOP)。

[0006] 在相机实现的此上下文中,单向传输可被用于从传感器捕捉图像并向基带处理器中的存储器传送此类图像数据,而控制数据可在该基带处理器与传感器以及其他外围设备之间被交换。在一个示例中,相机控制接口(CCI)协议可被用于基带处理器与图像传感器(和/或一个或多个从动设备)之间的此类控制数据。在一个示例中,CCI协议可在图像传感器与基带处理器之间的I2C串行总线上实现。

[0007] 已开发出对CCI的扩展,称为CCle(相机控制接口扩展),该CCle对信息进行编码以供在共享总线上传送。CCle不在共享总线上实现单独的时钟线。取而代之的是,CCle将时钟嵌入所传送的经转码信息内。然而,此类嵌入式时钟可以用于数据接收和/或同步目的,可能不足以准许将此类数据保存到寄存器中。

[0008] 另外,需要允许从动设备进入功率节省或休眠模式并且还允许主控设备将数据写入从动设备的机制。这可以通过使主控设备跟踪处于休眠模式的从动设备来达成,但是此类机制会添加不想要的开销。

[0009] 因此,需要高效地使用恢复出的嵌入在传输内的时钟以允许在没有自由运行的时钟的情况下将数据写入从动设备的寄存器中而同时又促成从动设备的高效休眠和苏醒机制的解决方案。

[0010] 概述

[0011] 在本公开的一方面,提供了一种方法、一种计算机程序产品和一种装置,该方法、计算机程序产品和装置提供了在没有自由运行的时钟的情况下使数据能够被写入从动设备的寄存器中而同时又促成从动设备的高效休眠和苏醒机制的机制。

[0012] 在某些方面,一种由接收机设备执行的方法包括:在共享总线上接收多个码元,提取嵌入在该多个码元的码元到码元转变中的时钟信号,将该多个码元转换成转变数,将该转变数转换成数据比特,以及仅使用该时钟信号来将这些数据比特的至少一部分存储到一个或多个寄存器中。

[0013] 在一个方面,这些码元每时钟循环地发生转变,以使得没有两个顺序码元具有相同的值。

[0014] 在一个方面,接收机设备独立地进入休眠模式而不通知耦合至共享总线的任何其他设备。

[0015] 在一个方面,接收机设备在不使用本地自由运行的时钟的情况下接收数据比特并且将这些数据比特的至少一部分写入一个或多个寄存器。接收机设备可在接收机处于休眠模式之时接收数据比特并且将这些数据比特的至少一部分写入一个或多个寄存器。

[0016] 在一个方面,转变数可被表达为三进制数。在一个示例中,转变数可以是12数位三进制数。

[0017] 在一个方面,共享总线是相机控制接口扩展(CCIe)总线。

[0018] 在一个方面,通过在检测到时钟信号的第一循环之际启动递减计数器、在该递减计数器达到预定义值时触发标记、以及使用该标记来将数据比特的至少一部分存储到寄存器中的方式来将这些数据比特的至少一部分写入一个或多个寄存器。该预定义值可发生在到达该时钟信号的最末时钟循环时。

[0019] 在一个方面,转变数在该时钟信号的倒数第二个时钟循环与最末时钟循环之间被转换成数据比特,并且这些数据比特的至少一部分在该时钟信号的最末时钟循环处被存储到寄存器中。

[0020] 在某些方面,一种接收机设备包括适配成将该接收机设备耦合至共享总线以接收多个码元的总线接口、一个或多个寄存器、以及耦合至该总线接口的接收机电路。接收机电路可被配置成提取嵌入在该多个码元的码元到码元转变中的时钟信号,将该多个码元转换成转变数,将该转变数转换成数据比特,以及仅使用该时钟信号来将这些数据比特的至少一部分存储到该一个或多个寄存器中。

[0021] 在某些方面,一种接收机设备包括:用于在共享总线上接收多个码元的装置,用于提取嵌入在该多个码元的码元到码元转变中的时钟信号的装置,用于将该多个码元转换成转变数的装置,用于将该转变数转换成数据比特的装置,以及用于仅使用该时钟信号来将这些数据比特的至少一部分存储到寄存器中的装置。

[0022] 在某些方面,一种机器可读存储介质具有存储于其上的指令。该存储介质可包括瞬态和/或非瞬态存储介质。这些指令可被至少一个处理器执行,并且这些指令可使该至少一个处理器:在共享总线上接收多个码元,提取嵌入在该多个码元的码元到码元转变中的时钟信号,将该多个码元转换成转变数,将该转变数转换成数据比特,以及仅使用该时钟信号来将这些数据比特的至少一部分存储到寄存器中。

[0023] 附图

[0024] 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。

[0025] 图1是解说具有基带处理器和图像传感器并且实现图像数据总线和多模控制数据

总线的设备的框图。

[0026] 图2解说了将时钟信息嵌入在数据码元内,藉此允许使用这两条I2C导线(SDA线和SCL线)来进行数据传输。

[0027] 图3是解说发射机处的产生具有嵌入式时钟信息的经转码码元的数据转码的示例的框图。

[0028] 图4解说在发射机处将比特转换成转变数以及在接收机处将转变数转换成比特的示例。

[0029] 图5解说在转变数与码元序列之间进行转换的一个示例。

[0030] 图6解说可被用于在顺序码元与转变数之间进行转换的转换函数的示例。

[0031] 图7解说配置成将从共享总线上传送的码元解码的数据写入寄存器的接收机。

[0032] 图8是对应于图7并且解说从码元解码数据以及从码元转变恢复时钟的时序图。

[0033] 图9解说影响从码元转变恢复时钟的某些状况。

[0034] 图10解说配置成仅使用从码元转变恢复的时钟并且在没有自由运行的时钟的情况下将从共享总线的传输解码的数据写入寄存器的接收机。

[0035] 图11解说可被配置成将12数位三进制数转变成比特以及仅使用从码元转变恢复的时钟并且在没有自由运行的时钟的情况下执行寄存器写操作的接收机的电路。

[0036] 图12是解说图10和11中解说的接收机和电路的操作的时序图。

[0037] 图13解说配置成通过使用从收到传输提取的时钟并且在无需从动设备苏醒的情况下写入来自该传输的数据的方式来从共享总线接收传输的CCI_e从动设备的示例。

[0038] 图14解说根据本文所公开的一个或多个方面的时钟恢复电路的示例。

[0039] 图15示出由图14的示例性时钟恢复电路生成的某些信号的定时的示例。

[0040] 图16解说将三进制数(基数为3的数)转换成二进制数的一般示例,其中 $\{T_{11}, T_{10}, \dots, T_2, T_1, T_0\}$ 中的每个转变是码元转变数。

[0041] 图17解说用于将二进制数转换成12数位三进制数的方法的示例。

[0042] 图18解说图17的除法和取模运算的可合成实现的示例。

[0043] 图19是解说采用可根据本文所公开的某些方面来适配的处理电路的装置的示例的框图。

[0044] 图20解说在从动设备上操作的用于在共享总线上接收传输并且仅使用从该传输恢复的时钟来将此类传输内的此类数据存储到寄存器中的方法。

[0045] 图21是解说采用根据本文所公开的某些方面适配的处理电路的装置的硬件实现的示例的示意图。

[0046] 详细描述

[0047] 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。

[0048] 总览

[0049] 第一特征提供了一种用于将数据比特转换成三进制数的方式。随后,通过首先将该三进制数的最高有效数位发送给转码器来将该三进制数转换成多个码元。随后,在总线

上传送该多个码元。例如,可在相机控制接口扩展(CCIe)总线上传送该多个码元。

[0050] 第二特征提供了一种接收机设备,该接收机设备被配置成从收到传输提取嵌入式时钟并且仅使用该嵌入式时钟来将收到传输中的数据存储到寄存器中(而无需自由运行的时钟或者使用经填补的填充物传输来生成额外的时钟循环)。因此,即使在接收机设备处于休眠模式时(在没有自由运行的时钟可用时),该接收机设备也可以接收和存储信息。

[0051] 用于在CCIe总线上简化码元编码和传输的示例性方法

[0052] 图1是解说具有基带处理器104和图像传感器106并且实现图像数据总线116和多模控制数据总线108的设备102的框图。虽然图1解说了相机设备内的多模控制数据总线108,但是应当清楚,该控制数据总线108可实现在各种不同设备和/或系统中。图像数据可在图像数据总线116(例如,高速差分DPHY链路)上从图像传感器106发送给基带处理器104。在一个示例中,控制数据总线108可以是I2C总线,其包括两条导线:时钟线(SCL)和串行数据线(SDA)。时钟线SCL可以被用来同步I2C总线(控制数据总线108)上的所有数据传递。数据线SDA和时钟线SCL耦合至I2C总线(控制数据总线108)上的所有设备112、114、和118。在该示例中,可以经由控制数据总线108在基带处理器104与图像传感器106以及其他外围设备118之间交换控制数据。I2C的标准时钟(SCL)速度最高达100KHz。I2C快速模式中的标准时钟SCL速度最高达400KHz,且在I2C快速模式加(Fm+)中其最高达1MHz。I2C总线上的这些操作模式可以在用于相机应用时被称为相机控制接口(CCI)模式。

[0053] 根据一方面,改进的操作模式(即,大于1MHz)可以被实现在多模控制数据总线108上以支持相机操作。I2C总线上的这种改进的操作模式可以在被用于相机应用时称为相机控制接口扩展(CCIe)模式。在该示例中,基带处理器104包括主控设备/节点112,且图像传感器106包括从动设备/节点114,主控设备/节点112和从动设备/节点114两者都可以在控制数据总线108上根据相机控制接口扩展(CCIe)模式来操作,而不会影响耦合到控制数据总线108的其它旧式I2C设备的正确操作。根据一个方面,控制数据总线108上的这种改进模式可无需CCIe设备与旧式I2C从动设备之间有任何桥接设备的情况下来实现。

[0054] 图2解说了时钟如何可被嵌入在数据码元内以藉此允许使用这两条I2C导线(即,SDA线和SCL线)来进行数据传输。在一个示例中,通过使用转变时钟转码来嵌入时钟。转变时钟转码可涉及将原始数据转码成码元数据,以使得在连贯码元之间存在通信链路206的信令状态上的转变(即,码元值在每个码元循环处发生转变)。也就是说,要在物理链路(导线)206上传送的数据204可被转码,以使得物理链路206的信令状态在所传送码元206的每个码元循环处改变。因此,原始时钟202被嵌入在每个码元循环的码元状态改变中。

[0055] 接收机从(所传送码元206中的)每个码元处的状态转变来恢复时钟信息208并且随后反转对所传送码元206的转码以获得原始数据210。这允许I2C总线的这两条导线(图1中的控制数据总线108,SDA线和SCL线)被用于发送数据信息。另外,码元速率可得以加倍,因为不再需要在时钟信号与数据信号之间具有建立和保持时间。

[0056] 图3是解说用于在发射机处将数据比特转码成经转码码元以将时钟信号嵌入在这些经转码码元内的方法的一个示例的框图。在发射机300处,输入数据比特304被转换成多数位三进制(基数为3)数,其中每一数位可被称为“转变数”。随后,该三进制数被转换成在物理链路302的时钟线SCL 312和数据线SDA 314上传送的一组(顺序)码元。在一个示例中,原始的20比特二进制数据被输入到比特至转变数转换器块308以被转换成12数位三进制

数。该12数位三进制数的每一数位表示“转变数”。两个连贯的转变数可以具有相同的值。每个转变数在转变至码元块310处被转换成顺序码元,以使得任何两个连贯的顺序码元不具有相同值。因为在每对顺序码元中的码元之间保证码元值(以及导线312、314的信令状态)上的转变,所以该顺序码元转变可以用于嵌入时钟信号。每个顺序码元316随后在双导线物理链路302上被发送,该双导线物理链路302可包括具有SCL线312和SDA线314的I2C总线。

[0057] 图4解说在发射机处400从比特到转变数以及随后在接收机420处从转变数到比特的转换的示例。该示例解说了使用12个转变码元的双导线系统的传输。发射机400将二进制信息(比特)馈送到“比特到 $12 \times T$ ”转换器406中以生成12个码元转变数 T_{11} 到 T_0 。接收机420接收12个码元转变数(T_{11} 到 T_0),这些码元转变数被馈入到“ $12 \times T$ 到比特”转换器408以取回二进制信息(比特)。在每一个转变(T_0 – T_{11})有 r 个可能的码元转变状态时,12个转变可发送 r^{12} 个不同状态。对于双导线总线, $r=2^2-1$ 。因此,转变 $T_0 \cdots T_{11}$ 包含可具有 $(2^2-1)^{12}$ 个不同状态的数据。因此, $r=4-1=3$,并且状态数目 $= (4-1)^{12}=531441$ 。

[0058] 在该示例中,对于使用12个码元转变数的双导线系统,可以假定每一个转变的可能码元转变 r 是 $3 (=2^2-1)$ 。若群中码元的数目为12,那么可使用12数位三进制数(基数为3的数): $T_{11}, T_{10}, \cdots, T_2, T_1, T_0$,其中每个 $T_i: 0, 1, 2$ 。例如,对于 $\{T_{11}, T_{10}, \cdots, T_2, T_1, T_0\} = \{2, 1, 0, 0, 1, 1, 0, 1, 0, 1, 2, 1\}$,三进制数是:

[0059] $2100_1101_0121_3$ (三进制数)

[0060] $= 2 \times 3^{11} + 1 \times 3^{10} + 0 \times 3^9 + 0 \times 3^8$

[0061] $+ 1 \times 3^7 + 1 \times 3^6 + 0 \times 3^5 + 1 \times 3^4$

[0062] $+ 0 \times 3^3 + 1 \times 3^2 + 2 \times 3^1 + 1 \times 3^0$

[0063] $= 416356 (0x65A64)$ 。

[0064] 以此方式,12个转变数可被转换成一个数字。注意,三进制数 $2100_1101_0121_3$ 可被用作转变数(例如在图3中)以使得每一个整数可以被映射到顺序码元,反之亦然。在以逆序发送 $2100_1101_0121_3$ 时,这些转变数是以幂的降序来发送的,即, T_{11} 是要乘以 3^{11} 的数位,所以它是第11次幂,依此类推。

[0065] 图4中针对双导线系统和12个码元转变数解说的示例可被推广为 n 导线系统和 m 个码元转变数。如果每一个 T (T_0 到 T_{m-1})有 r 个可能的码元转变状态,那么 m 个转变可发送 r^m 个不同状态(即, $r=2^n-1$)。因此,转变 $T_0 \cdots T_{m-1}$ 包含可具有 $(2^n-1)^m$ 个不同状态的数据。

[0066] 图5是解说用于在三进制数(转变数)502与(顺序)码元504之间进行转换的方案的一个示例的示意图500。三进制数(基数为3的数,也被称为转变数)可具有这三个可能数位或状态0、1或2之一。虽然在两个连贯三进制数中可能出现相同值,但任何两个连贯码元不具有相同值。

[0067] 图6中解说性地阐述了转换功能。在传送侧(TX:T到S),逻辑是 $T_{tmp} = T = 0 ? 3 : T$ 和 $C_s = P_s + T_{tmp}$ 。换言之,将转变数 T 与0进行比较,并且在 $T=0$ 时, T_{tmp} (T 临时)变成等于3,否则(在 T 不等于0时) T_{tmp} 变成等于 T 。并且当前码元(C_s)变成先前码元值(P_s)加 T_{tmp} 。例如,在第一循环506中, T 为2,所以 T_{tmp} 也为2,并且在 P_s 为1的情况下,新的 C_s 现在为3。

[0068] 在第二循环508中,转变数1在此下一循环中被输入,并且转变数不为3,所以将为1的 T 值与先前码元的为3的值相加。由于该加法的结果4大于3,因此翻转数0成为当前码元。

[0069] 在第三循环510中,相同转变数1被输入。因为 T 为1,所以 T_{tmp} 也为1。转换逻辑将1与

先前码元0相加以生成当前码元1。

[0070] 在第四循环512中,转变数0被输入。在T为0时,转换逻辑使 T_{tmp} 等于3。因此将3与先前码元1相加以生成当前码元0(由于该加法的结果4大于3,因而翻转数0成为当前码元)。

[0071] 因此,即使两个连贯三进制数位502具有相同数,该转换仍保证了两个连贯码元数具有不同状态值。正因如此,码元序列504中的有保证码元转变可以用于嵌入时钟信号,藉此释放I2C总线中的时钟线SCL以用于数据传输。在接收机侧(RX:S到T),该逻辑反转: $T_{tmp} = C_s + 4 - P_s$ 并且 $T = T_{tmp} = 3 \text{?} 0 : T_{tmp}$ 。

[0072] 再次参照图3,在接收机320处,该过程被反转以将经转码码元转换回比特,并且在该过程中,从码元转变中提取时钟信号。接收机320在双导线物理链路302上接收顺序码元序列322,该双导线物理链路302可以是连接到SCL线输入324和SDA线输入326的I2C总线。所接收的顺序码元322被输入到时钟数据恢复(CDR)块328中以恢复时钟定时并采样经转码码元(S)。CDR 328可从收到码元中的码元到码元转变来恢复时钟信号336。此经恢复时钟336可以用于使得接收机组件能够操作并且提取出的比特能被写入而不需要单独的时钟。码元到转变数转换器块330随后将每个码元转换成转变数,该转变数可被表达为表示当前码元与紧前码元之间的差异的单数位三进制数。随后,转变数至比特转换器332转换12个转变数以从12数位三进制数复原20比特原始数据304'。

[0073] 本文解说的此技术可被用来提高控制总线108(图1)的链路速率,以超出I2C标准总线所提供的链路速率,并且于此被称为CCIE模式。在一个示例中,耦合至控制数据总线108的主控节点和/或从动节点可以实现在码元传输内嵌入时钟信号的发射机和/或接收机(如图2和3中所解说的),以在相同控制数据总线上达成比使用标准I2C总线所可能达成的比特率更高的比特率。注意,在其他实现中,不同数目的数据比特可被编码到三进制数(基数为3的数制)或者具有不同数值基数的数字中。

[0074] 图6是解说顺序码元与转变数之间的转换的示图。该转换将从前一顺序码元号(P_s)到当前顺序码元(C_s)的每一转变映射成转变数(T)。在发射机设备处,转变数被转换成顺序码元。由于正在使用相对转换方案,因此转变数保证了没有两个连贯的顺序码元604将会是相同的。

[0075] 在针对2导线系统的一个示例中,存在被指派给4个顺序码元S0、S1、S2和S3的4个原始码元。对于这4个顺序码元,表602解说了可如何在前一顺序码元(P_s)和基于当前转变数(T)的临时转变数 T_{tmp} 的基础上来指派当前顺序码元(C_s)。

[0076] 在该示例中,转变数 C_s 可根据下式来指派:

$$[0077] \quad C_s = P_s + T_{tmp}$$

$$[0078] \quad \text{其中 } T_{tmp} = T = 0 \text{?} 3 : T。$$

[0079] 换言之,如果T等于0,则 T_{tmp} 变成3,否则 T_{tmp} 变成等于T。并且一旦 T_{tmp} 被计算出, C_s 就被设置成 P_s 加 T_{tmp} 。此外,在接收机端,逻辑反转以恢复T,

$$[0080] \quad T_{tmp} = C_s + 4 - P_s \text{ 并且 } T = T_{tmp} = 3 \text{?} 0 : T_{tmp}。$$

[0081] 图16包括解说将三进制数(基数为3的数)转换成二进制数的一般示例的等式1600,其中 $\{T_{11}, T_{10} \dots T_2, T_1, T_0\}$ 中的每个T是码元转变数。

[0082] 图17包括解说用于将二进制数(比特)转换成12数位三进制数(基数为3的数)的示例性方法的等式1700。每一数位可如下计算:将来自更高数位计算中的余数(取模运算的结

果)除以3的该数位号次幂,丢弃小数点后面的数字。

[0083] 图18是解说图17的除法和取模运算的一个可能实现的示例的数学表示1800,其可以由任何商业合成工具来合成。

[0084] 使用嵌入式接收时钟来提取数据

[0085] 图7是解说可与图3的接收机320相当的接收机700的示例的示意性框图。接收机700可被配置成从接收自共享总线的码元序列702提取数据724,该共享总线包括可对应于图3中的线324和326的信号导线704、706。CDR电路708可提供经采样码元720和接收时钟716,该接收时钟716是从表示码元序列702的多比特信号718中提供的定时信息推导出的。接收时钟716还可被用于向从码元序列702提取数据的解码逻辑710、712提供定时信息。

[0086] 在一些实例中,在接收时钟716被用于仅使用从码元序列702中的码元之间的转变推导出的时钟循环来将提取出的数据724写入寄存器714时可能存在问题。从收到传输702内的码元到码元转变提取出的接收时钟716可能并不提供足够的时钟循环以解码数据724并且将数据724存储到寄存器714中。在最末的码元到码元转变之后可能需要额外的时钟循环以将提取出的比特写入寄存器714以供存储。在一些实例中,自由运行的时钟可被用于提供充分的时钟循环。自由运行的时钟的使用可能是不希望的,因为此类自由运行的时钟的存在可能要求主控设备需要确保从动设备在传输之前苏醒着。在某些状况(包括关于图8讨论的状况)下,提取出的时钟可能不足以从码元序列702提取数据并且将该数据写入寄存器714。

[0087] 图8是解说与提取编码在码元内的数据以及从码元转变恢复时钟相关联的定时的时序图800。跟在开始指示符(S)802之后的码元序列806通过双线总线704和706被传送。在示图800中解说了可表示码元序列702的SI信号718与码元之间的相应转变722之间的关系。从码元到码元转变722提取接收机时钟(RXCLK)716。初始时钟脉冲806对应于也可被称为“开始状况”的开始指示符(S)802。由于没有两个相同的顺序码元重复,因而可以从连贯码元 $S_{11}, S_{10}, S_9, \dots, S_0$ 718之间的转变($T_{11}, T_{10}, T_9 \dots T_0$)722提取RXCLK 716上的多个脉冲 $C_1, C_2, \dots, C_{12}$ 。该多个脉冲可被认为始于对应于经编码码元之间的第一个转变的第一脉冲(C_1)808。

[0088] 在倒数第二个时钟循环(C_{11})810之后并且在对应于经编码码元之间的转变的最末时钟循环(C_{12})812之前,最终或最末码元(S_0)804被接收到并且与其余码元 $S_{11} \dots S_1$ 组合,以使得在最末时钟循环(C_{12})812发生之后可产生20个原始数据比特816。注意,仅在接收到最末码元(例如,第12码元 S_0)之后原始比特才能被解码以获得原始数据比特816。最末时钟循环814用于将原始数据比特816或其一部分存储到寄存器714中。

[0089] 在一个示例中,所接收到的码元数目为12。这12个码元可编码20比特信息(例如,包括十六(16)个数据比特和四(4)个控制比特)。在其他示例中,不同数目的码元可被用于编码不同数目的比特。

[0090] 图9是解说与串行总线的两条线704和706的不同信令状态相对应的不同的恢复出的时钟状况的示图900。给出了四种不同情形912、914、916、918。在这四种情形912、914、916、918中的每一者中,在最末时钟(第12)时钟脉冲902上接收到最终码元(S_0)。这四种情形912、914、916、918覆盖最终码元的四种可能值。在第四情形918中,这两条线704和706的对应于最终码元的信令状态与这两条线704和706在终止建立状况期间的信令状态相同。相

应地,在期间传送最终码元的码元周期之后,在这两条线704和706的信令状态上没有观察到转变。第13时钟脉冲904在前三种情形912、914、916中被生成并且可被用于将收到数据写入寄存器。第13时钟脉冲在第四种情形918中不存在,并且如果接收机依赖第13时钟脉冲904的可用性,则可能妨碍接收机可靠地将收到数据写入寄存器中。

[0091] 生成有保证寄存器写入信号

[0092] 根据本文公开的某些方面并且如由图10-12解说的,字标记可被生成以使得从码元序列解码出的数据能够被可靠地写入CCI_e接收机中的寄存器。图10是解说接收机1000的示例的示意性框图,该接收机1000可使用图11的电路1100、1110和/或1130来适配以从SI信号1020解码数据。图12中提供的时序图1200解说接收机1000和电路1100、1110、1130的操作。SI信号1020可在RXCLK 1016的采样边沿发生之前由S到T解码器1010、T到比特解码器1012并且在一些实例中还有寄存器数据解码器1114的组合来处理。相应地,所寻址的寄存器1113执行与接收自串行总线的信号相关的采样,并且该采样是对经解码数据执行的。

[0093] 接收机1000可被配置成从接收自共享总线的码元序列1002提取数据,该共享总线包括可对应于图3中的线324和326的信号导线1004、1006。可使用由CDR电路1008从收到数据恢复出的接收时钟1016来将数据写入寄存器1014。不需要自由运行的时钟。接收时钟(RXCLK)1016还可被用于向从码元序列1002提取数据的解码逻辑1010、1012提供定时信息。在该电路中,码元到转变数转换器1010接收并处理SI信号1020。

[0094] 接收机1000可采用一个或多个计数器来维护转变数位置(DNCNT)1123的索引并且产生始终如一地准许将收到数据1024写入寄存器1014的字标记(包括在图9中解说的第四情形918中)。寄存器写操作可在最末恢复出的时钟RXCLK 1016循环上执行,而不需从动设备上有自由运行的时钟或者其他附加时钟,并且不需要仅为了使额外时钟循环可用的目的而插入不使用的/填充的比特。

[0095] 继续参照图10,图11中解说的某些电路1100、1110、1130可被适配或配置成将12数位三进制数转换成比特并且仅使用从对应于该三进制数的传输恢复的时钟1016来执行所提取比特的寄存器写操作。回来参照图3,20比特原始数据304被转换成三进制转变数,随后此转变数被转换(即,转码)成12个顺序码元316。在总线302上传送经转码码元316。接收方设备320(例如,从动设备)接收传输316并且执行时钟恢复和码元采样以将经转码码元转换回三进制数,该三进制数随后被提供给一个或多个电路(诸如电路332),该一个或多个电路将该三进制数转换回原始的20比特二进制数据。

[0096] 第一电路1100可被适配成从十二(12)个转变数提取二十(20)个原始比特1108。使用DNCNT 1123来选择三进制权重1102以控制复用器1104,其中DNCNT 1123表示转变数位置。这12个转变数可按基于其在转变数序列(例如,与抵达时间相关的序列)中的相应位置所确定的次序来被处理,该相应位置可由DNCNT 1123的值来指示,而该DNCNT 1123的值可由计数器、寄存器、或其他索引电路系统提供。三进制权重1102被提供作为至单输出复用器1104的输入,该单输出复用器1104被用于串行化这些三进制权重1102,以使得能够提取出20个原始比特1108。这20个原始比特1108可包括十六(16)个数据比特和四(4)个控制比特。第二复用器1106用作 $T_i \times 3^i$ 运算的乘法器并且由表示来自图10的码元到三进制块1010的2比特输出的信号1022触发和/或控制。由RXCLK 1016触发的第一触发器1125被用于随着三进制数从三进制权重1102被解码或转换成原始比特1108而对瞬时比特进行累加。注意,在

倒数第二个时钟脉冲(C11)1217之后接收的最末码元(S_0)1221(参见图12)的发生触发了第一触发器1125输出所采集瞬时比特,这些所采集瞬时比特要与来自第二复用器1106输出的最末三进制权重1103的比特相加。因此,原始比特1108(例如,图10中的数据1024)保持有效值并且在倒数第二个时钟循环(C11)1217之后、但在最末时钟循环(C12)1219之前输入了最末码元 S_0 1221之后可用。

[0097] 第二电路1110可以用于在接收到了所有码元时获得字标记1122。在检测到接收机时钟1120的开始指示符1118之际,递减计数器的值(DNCNT)1123随着RXCLK 1016上的每个脉冲从0xB hex(十六进制)减小到倒数第二个时钟(C11)1217处的0(0x0hex)并且随后在最末时钟(C12)1219处到0xF hex。在递减计数器达到0x0hex时触发字标记1122上的脉冲1215。字标记1122用作至第三电路1130的输入以使得能够将数据比特写入寄存器。注意,DNCNT 1123还用于从复用器1104选择以输入“B”(第一三进制权重1105)开始并且倒计数至输入“0”(最末三进制权重1103)的输入信号。

[0098] 第三电路1130解说配置成将经解码比特写入第二触发器或寄存器1113的电路的示例。地址解码器1112接收十七(17)个地址信息比特并且解码这些地址信息比特。类似地,数据解码器1114接收二十(20)个原始比特1108并且解码这些原始比特以在4个控制比特已被移除之后获得例如十六(16)个数据比特。当字标记1122的脉冲1215被触发并且地址被解码时,由数据解码器1114提供的经解码数据可被存储在触发器或寄存器1113中。此第三电路1130有效地使用字标记1122来触发在最末时钟循环(C12)1219上向第二触发器或寄存器1113的写入。

[0099] 在倒数第二个时钟循环(C11)1217上,DNCNT 1123已从0xB hex递减到0x0hex,并且字标记1122从逻辑低转变成逻辑高(即,脉冲1215的开始)。在最末时钟循环(C12)1219处,第二触发器或寄存器1113被启用并且存储现在携带经解码数据比特的16位总线。

[0100] 此办法准许在从动设备上没有运行的时钟的情况下将收到数据比特存储到触发器或寄存器1113中。因此,从动设备可在不通知主控设备的情况下进入休眠模式。即,不需要在从动设备进入休眠模式时通知主控设备的单独机制(例如,来自从动设备的“从动休眠请求”是不必要的)。因为嵌入式时钟允许从动设备接收所传送比特并且第三电路1130在无需从动设备苏醒的情况下生成附加时钟,所以主控设备甚至能在从动设备休眠或处于休眠模式中时(例如,在无需自由运行的时钟的情况下)将数据写入从动设备寄存器。在一些实现中,从动设备可使用被写入的寄存器数据来有条件地唤醒其部分或全部功能性。因此,主控设备不必在向从动设备发送或写入数据之前知晓从动设备是苏醒的还是正在休眠。另外,从动设备可在不通知主控设备的情况下独立地进入休眠模式。

[0101] 图12的时序图1200解说对编码在码元内的数据的接收、从码元转变进行的时钟恢复、以及用于仅使用恢复出的时钟来完成收到数据至寄存器的写操作的所生成信号的定时。跟在开始指示符(S)1206之后的码元序列1020通过双线总线1004、1006来传送。解说了码元序列1020以及码元之间的相应转变1022。从码元到码元转变1022提取接收机时钟1016。初始时钟脉冲1211对应于开始指示符或开始状况(诸如I2C规范中描述的开始状况)。由于没有两个相同的顺序码元重复,因而可以从连贯码元 $S_{11}, S_{10}, S_9, \dots, S_0$ 之间的转变($T_{11}, T_{10}, T_9 \dots T_0$)提取多个时钟脉冲C1, C2, $\dots$, C12。

[0102] 在此示例中,DNCNT 1123被用于倒计数十二(12)个循环,每个循环对应于从接收

机时钟RXCLK 1016的低到高转变至低到高转变的循环。DNCNT 1123在检测到第一时钟循环1213之后被递减,并且直至检测到最末循环1219。当DNCNT 1123到达0x0hex时,在WORD(字)信号1122上触发脉冲1215。

[0103] 在倒数第二个时钟循环(C11)1217之后并且在最末时钟循环(C12)1219之前,最终或最末码元(S_0)1221被接收到并且与其余码元 $S_{11}\cdots S_1$ 组合,以使得原始数据比特1108在最末时钟循环C12 1219发生时可用。注意,在接收到最末码元(例如,第12码元 S_0)之后这些原始比特才能被解码以获得原始数据比特1108。最末时钟循环1219随后被用于将原始数据比特1108或其一部分存储到寄存器1113中。这允许仅使用嵌入式时钟(例如,从码元到码元转变恢复出的时钟)并且在接收机(从动)设备处不使用外部或自由运行的时钟的情况下接收、解码和存储数据1222。注意,这是在无需填充或插入额外码元或比特的情况下达成的。在一个示例中,所接收到的码元数目为12。这12个码元可编码20比特信息(例如,包括十六(16)个数据比特和四(4)个控制比特)。在其他示例中,不同数目的码元可被用于编码不同数目的比特。

[0104] 图13是解说可被配置成通过使用从收到传输提取的时钟并且在无需从动设备苏醒的情况下写入来自该传输的数据的方式来从共享总线接收传输的CCIe从动设备1302的示例的框图1300。从动设备1302包括耦合至共享总线1304和1306的接收机电路1308和发射机电路1310。控制逻辑1314可以用于选择性地激活/停用接收机电路1308和/或发射机电路1310,以使得从动设备在共享总线1304和1306上进行接收或传送。从动设备1302还可包括传感器设备,该传感器设备捕捉或采集信息以供从该从动设备传送。

[0105] 接收机电路1308可包括时钟数据恢复电路1312,该时钟数据恢复电路1312可根据本文公开的某些方面从码元到码元转变来提取接收机时钟(RXCLK)。接收机电路1308还可包括第一电路1100、第二电路1110、和/或第三电路1130(图11)中的一者或多者以解码和提取在共享总线上接收到的数据以及仅使用从收到数据传输提取的时钟并且在不引入所提取时钟的延迟的情况下将此类数据存储于寄存器1318中。注意,第一电路1100、第二电路1110、和/或第三电路1130(图11)可被集成到一个电路中或者分布在不同模块或子系统之中。

[0106] 时钟发生器1320可存在于从动设备1302内,但是该时钟发生器1320仅被用于从该从动设备传送数据和/或其他从动设备操作(例如,由传感器设备进行的运动检测或温度测量)。

[0107] 图14解说了根据本文所公开的一个或多个方面的CDR电路1400的示例,并且图15示出了由CDR电路1400生成的某些信号的定时的示例。CDR电路1400可在CCIe传输方案中使用,其中时钟信息被嵌入在所传送的码元序列中。CDR电路1400可被用作CDR 328(图3)或CDR 1312(图13)。CDR电路1400包括模拟延迟元件1408a、1412和1426,这些模拟延迟元件1408a、1412和1426可被配置成使接收自CCIe双线总线324和326的码元1510、1512的建立时间最大化。CDR电路1400包括比较器1404、置位-复位锁存器1406、包括第一延迟元件1408a的单稳元件1408、第二模拟延迟元件1412、第三模拟延迟元件1426、以及电平锁存器1410。比较器1404可以将包括码元流1510和1512的输入信号(SI)1420与作为SI信号1420的电平锁存实例的信号(S)1422进行比较。该比较器输出比较信号(NE)1414。置位-复位锁存器1406从比较器1404接收NE比较信号1414并输出比较信号的经滤波版本(NEFLT)1416。第一

模拟延迟器件1408a可接收NEFLT信号1416的经滤波版本并且输出作为NEFLT信号1416的经延迟实例的信号(NEDEL信号)1428。在操作中,单稳逻辑1408接收NEFLT信号1416和经延迟的NEDEL信号1428并输出包括由NEFLT信号1416触发的脉冲1506的信号(NE1SHOT)1424。

[0108] 第二模拟延迟器件1412接收NE1SHOT信号1424并输出IRXCLK信号1418,其中IRXCLK信号1418可被用于使用第三模拟延迟元件1426来生成输出时钟信号1430。输出时钟信号1430可被用于解码S信号1422中的锁存码元。置位-复位锁存器1406可基于IRXCLK信号1418的状态来被复位。电平锁存器1410接收SI信号1420并输出电平锁存S信号1422,其中电平锁存器1410由IRXCLK信号1418启用。

[0109] 在正接收第一码元值 S_1 1510时,这导致SI信号1420开始改变其状态。由于可能因导线间偏斜、信号过冲、信号欠冲、串话等而在从先前码元 S_0 1502向第一码元 S_1 1510的信号转变处发生中间或不确定状态的可能性,因而SI信号1420的状态可不同于 S_1 信号1510的状态。NE信号1414在比较器1404检测到SI信号1420和S信号1422之间的不同值时转变为高,从而使得置位-复位锁存器1406被异步地置位。相应地,NEFLT信号1416转变为高,并且该高状态被保持直到置位-复位锁存器1406在IRXCLK 1418变为高时被复位。在对NEFLT信号1416的上升的延迟响应中,IRXCLK 1418转变为高状态,其中该延迟可部分地归因于模拟延迟元件1412。

[0110] SI信号1420上的中间状态可以被认为是无效数据,并且可以包括码元 S_0 1502的码元值达短时段,并且这些中间状态可能引起NE信号1414中的尖峰或转变1538,因为比较器1404的输出向着低状态返回达短时段。尖峰1538不影响由置位-复位锁存器1406输出的NEFLT信号1416,因为置位-复位锁存器1406在输出NEFLT信号1416之前有效地阻挡和/或滤除了NE信号1414上的尖峰1538。

[0111] 在NEFLT信号1416的上升沿之后,单稳电路1408输出NE1SHOT信号1424中的高状态。在NE1SHOT信号1424回到低状态之前,单稳电路1408将NE1SHOT信号1424保持在高状态长达延迟P时段1516。NE1SHOT信号1424上的结果所得脉冲1506在由模拟延迟S元件1412导致的延迟S时段1518之后传播到IRXCLK信号1418。IRXCLK信号1418的高状态将置位-复位锁存器1406复位,并且NEFLT信号1416转变为低。IRXCLK信号1418的高状态还启用电平锁存器1410,并且SI信号1420的值被输出作为S信号1422。

[0112] 比较器1404检测对应于 S_1 码元1510的S信号1422何时匹配于SI信号1420的码元 S_1 码元1510,并且比较器1404的输出将NE信号1414驱动为低。NE1SHOT信号1424上的脉冲1540的后沿在由模拟延迟S元件1412导致的延迟S时段1518之后传播到IRXCLK信号1418。当正在接收新码元 S_2 1512时,SI信号1420在IRXCLK信号1418的后沿之后开始其向对应于码元 S_2 1512的值的转变。

[0113] 在一个示例中,输出时钟信号RXCLK 1430被第三模拟延迟元件1426延迟达延迟R时段1520。输出时钟信号1430和S信号1422(数据)可被提供给解码电路1100、1110、和/或1130(图11)。解码电路1100、1110、和/或1130(图11)可以使用输出时钟信号1430或其衍生信号来对S信号1422上的码元进行采样。

[0114] 在所描绘的示例中,各种延迟1522a-1522d可归因于各种电路的切换时间和/或可归因于连接器的上升时间。为了提供足够建立时间以供由解码电路进行码元捕捉,码元循环周期 t_{SYM} 的定时约束可以如下定义:

[0115] $t_{dNE} + t_{dNEFLT} + t_{dIS} + \text{延迟S} + \text{延迟P} + \max(t_{HD}, t_{REC} - t_{dNE}) < t_{SYM}$

[0116] 且建立时间 t_{SU} 的定时约束可以如下:

[0117] 最大偏斜规范 $+t_{SU} < t_{dNE} + t_{dIS} + \text{延迟S}$

[0118] 其中:

[0119] t_{SYM} : 一个码元循环周期,

[0120] t_{SU} : 电平锁存器1410的SI 1420的以IRXCLK 1418的上升(前)沿为参照的建立时间,

[0121] t_{HD} : SI 1420的对于电平锁存器1410的以IRXCLK 1418的下降(后)沿为参照的保持时间,

[0122] t_{dNE} : 比较器1404的传播延迟,

[0123] t_{dRST} : 置位-复位锁存器1406从IRXCLK 1418的上升(前)沿起的复位时间。

[0124] CDR电路1400采用模拟延迟电路1408a、1412和1426以确保接收机设备(例如,从动设备1302)可在不使用自由运行的系统时钟的情况下解码CCIE编码码元并且将结果得到的比特存储到寄存器中。相应地,CCIE从动设备1302(参见图13)可被适配成在对CCIE读命令作出响应时使用传送时钟1320作为系统时钟,并且在接收数据时或者在从动设备休眠时可使用由CDR生成的时钟1430。

[0125] 图19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1902的装置的硬件实现的简化示例的概念图1900。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1902来实现。处理电路1902可包括一个或多个处理器1904,其由硬件和软件模块的某种组合来控制。处理器1904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1904可包括执行特定功能并且可由软件模块1916之一来配置、扩增或控制的专用处理器。该一个或多个处理器1904可通过在初始化期间加载的软件模块1916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1916来进一步配置。

[0126] 在所解说的示例中,处理电路1902可使用由总线1910一般化地表示的总线架构来实现。取决于处理电路1902的具体应用和整体设计约束,总线1910可包括任何数目的互连总线和桥接器。总线1910将各种电路链接在一起,包括一个或多个处理器1904、和存储1906。存储1906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1910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1908可提供总线1910与一个或多个收发机1912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1912。在一些实例中,多种联网技术可共享收发机1912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1912提供用于通过传输介质与各种其它装置通信的手段。取决于该装置的本质,也可提供用户接口1918(例如,按键板、显示器、扬声器、话筒、操纵杆),并且该用户接口1918可直接或通过总线接口1908通信地耦合至总线1910。

[0127] 处理器1904可负责管理总线1910和一般处理,包括对存储在计算机可读介质(其可包括存储1906)中的软件的执行。在这一方面,处理电路1902(包括处理器1904)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1906可被用于存储处理器1904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。

[0128] 处理电路1902中的一个或多个处理器1904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1906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1906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1906还可包括载波、传输线、和任何其它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1906可驻留在处理电路1902中、处理器1904中、在处理电路1902外部、或跨包括该处理电路1902在内的多个实体分布。计算机可读介质和/或存储1906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。

[0129] 存储1906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1916。软件模块1916中的每一者可包括在安装或加载到处理电路1902上并被一个或多个处理器1904执行时有助于运行时映像1914的指令和数据,运行时映像1904控制一个或多个处理器1904的操作。在被执行时,某些指令可使得处理电路1902执行根据本文所描述的某些方法、算法和过程的功能。

[0130] 软件模块1916中的一些可在处理电路1902初始化期间被加载,并且这些软件模块1916可配置处理电路1902以实现本文所公开的各种功能的执行。例如,一些软件模块1916可配置处理器1904的内部设备和/或逻辑电路1922,并且可管理对外部设备(诸如,收发机1912、总线接口1908、用户接口1918、定时器、数学协处理器等)的访问。软件模块1916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1902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1912的访问、用户接口1918等。

[0131] 处理电路1904的一个或多个处理器1902可以是多功能的,由此软件模块1916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1904可附加地被适配成管理响应于来自例如用户接口1918、收发机1912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1904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1904服务的任务集。在一个示例中,多任务环境可使用分时程序1920来实现,分时程序1920在不同任务之间传递对处理器1904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1904的控制权返回给分时程序1920。当任务具有对一个或多个处理器1904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1920可包括操作系统、在循环法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1904的控制权的功能、和/或通过对一个或多个处理器1904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。

[0132] 图20是在从动设备上操作的用于在共享总线上接收传输并且仅使用从该传输恢复的时钟来将此类传输内的此类数据存储到寄存器中的方法的流程图2000。例如,该方法可以由图13中的接收机设备来实现。

[0133] 在框2002,可在共享总线上接收多个码元。共享总线可以是CCIE总线。这些码元可每时钟循环发生转变,以使得没有两个顺序码元具有相同的值。

[0134] 在框2004,提取嵌入在该多个码元的码元到码元转变中的时钟信号。

[0135] 在框2006,可将该多个码元转换成转变数。该转变数可以是12数位三进制数。

[0136] 在框2008,可将该转变数转换成数据比特。该转变数可以在时钟信号的倒数第二个时钟循环与最末时钟循环之间被转换成数据比特。

[0137] 在框2010,可仅使用该时钟信号来将数据比特的至少一部分存储到一个或多个寄存器中。接收机设备可在不使用本地自由运行的时钟的情况下接收数据比特并且将这些数据比特的至少一部分写入该一个或多个寄存器。接收机设备在该接收机处于休眠模式之时接收数据比特并且将这些数据比特的至少一部分写入一个或多个寄存器。通过在检测到时钟信号的第一循环之际启动递减计数器、在该递减计数器达到预定义值时触发标记、以及使用该标记来将这些数据比特的至少一部分存储到寄存器中的方式来将这些数据比特的至少一部分写入一个或多个寄存器。该预定义值可在到达时钟信号的最末时钟循环时发生。这些数据比特的至少一部分可在时钟信号的最末时钟循环处被存储到寄存器中。

[0138] 在一个示例中,接收机设备可独立地进入休眠模式而不通知耦合至共享总线的任何其他设备。

[0139] 图21是解说采用处理电路2102的设备的硬件实现的简化示例的示图2100。该处理电路通常具有处理器2116,处理器2116可包括微处理器、微控制器、数字信号处理器、定序器和状态机中的一者或多者。处理电路2102可以用由总线2120一般化地表示的总线架构来实现。取决于处理电路2102的具体应用和整体设计约束,总线2120可包括任何数目的互连总线和桥接器。总线2120将包括一个或多个处理器和/或硬件模块(由处理器2116、模块或电路2104、2106、2108和2110、可配置成通过连接器或串行总线2114的导线来通信的线接口电路2112、与码元/三进制转换电路系统协作的一个或多个寄存器2122、以及计算机可读存储介质2118表示)的各种电路链接在一起。总线2120还可链接各种其它电路,诸如定时源、外围设备、稳压器和功率管理电路,这些电路在本领域中是众所周知的,且因此将不再进一步描述。

[0140] 处理器2116负责一般性处理,包括执行存储在计算机可读存储介质2116上的软件。该软件在由处理器2116执行时使处理电路2102执行上文针对任何特定装置描述的各种功能。计算机可读存储介质2118也可被用于存储由处理器2116在执行软件时操纵的数据,包含从通过连接器2114传送的码元解码得来的数据,连接器2114可被配置为数据通道和时钟通道。处理电路2102进一步包括模块2104、2106、2108和2110中的至少一个模块。模块2104、2106、2108和2110可以是在处理器2116中运行的软件模块、驻留/存储在计算机可读存储介质2118中的软件模块、耦合至处理器2116的一个或多个硬件模块、或其某种组合。模

块2104、2106、2108和/或2110可包括微控制器指令、状态机配置参数、或其某种组合。

[0141] 在一个配置中,用于无线通信的设备2100包括被配置成在串行总线2114上接收多个码元的模块和/或电路2104、被配置成提取嵌入在该多个码元的码元到码元转变中的时钟信号的模块和/或电路2106、被配置成将该多个码元转换成转变数的模块和/或电路2108、被配置成将转变数转换成数据比特的模块和/或电路2108、以及被配置成仅使用该时钟信号来将数据比特的至少一部分存储到寄存器中的模块和/或电路2122。

[0142] 附图中解说的组件、步骤、特征、和/或功能之中的一个或多个可以被重新编排和/或组合成单个组件、步骤、特征、或功能,或可以实施在数个组件、步骤或功能中。还可添加附加的元件、组件、步骤、和/或功能而不会脱离本文中所公开的新颖特征。附图中所图解的装置、设备和/或组件可以被配置成执行在这些附图中所描述的方法、特征、或步骤中的一个或多个。本文中描述的新颖算法还可以高效地实现在软件中和/或嵌入在硬件中。

[0143] 另外应注意,这些实施例可能是作为被描绘为流程图、流图、结构图、或框图的过程来描述的。尽管流程图可能会把诸操作描述为顺序过程,但是这些操作中有许多操作能够并行或并发地执行。另外,这些操作的次序可以被重新安排。过程在其操作完成时终止。过程可对应于方法、函数、规程、子例程、子程序等。当过程对应于函数时,它的终止对应于该函数返回调用方函数或主函数。

[0144] 此外,存储介质可以代表用于存储数据的一个或多个设备,包括只读存储器(ROM)、随机存取存储器(RAM)、磁盘存储介质、光学存储介质、闪存设备、和/或其他用于存储信息的机器可读介质。术语“机器可读介质”包括但不限于:便携或固定的存储设备、光学存储设备、无线信道以及能够存储、包含、或承载指令和/或数据的各种其它介质。

[0145] 此外,诸实施例可以由硬件、软件、固件、中间件、微代码、或其任何组合来实现。当在软件、固件、中间件、或微码中实现时,执行必要任务的程序代码或代码段可被存储在诸如存储介质之类的机器可读介质或其它存储中。处理器可以执行这些必要的任务。代码段可表示规程、函数、子程序、程序、例程、子例程、模块、软件包、类,或是指令、数据结构、或程序语句的任何组合。通过传递和/或接收信息、数据、自变量、参数、或存储器内容,一代码段可被耦合至另一代码段或硬件电路。信息、自变量、参数、数据等可以经由包括存储器共享、消息传递、令牌传递、网络传输等的任何合适的手段被传递、转发、或传输。

[0146] 结合本文中公开的示例描述的各个解说性逻辑块、模块、电路、元件和/或组件可用设计成执行本文中描述的功能的通用处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或其他可编程逻辑组件、分立的门或晶体管逻辑、分立的硬件组件、或其任何组合来实现或执行。通用处理器可以是微处理器,但在替换方案中,该处理器可以是任何常规的处理器、控制器、微控制器、或状态机。处理器还可以实现为计算组件的组合,例如DSP与微处理器的组合、数个微处理器、与DSP核心协作的一个或多个微处理器、或任何其他此类配置。

[0147] 结合本文中公开的示例描述的方法或算法可直接在硬件中、在能由处理器执行的软件模块中、或在这两者的组合中以处理单元、编程指令、或其他指示的形式实施,并且可包含在单个设备中或跨多个设备分布。软件模块可驻留在RAM存储器、闪存、ROM存储器、EPROM存储器、EEPROM存储器、寄存器、硬盘、可移动盘、CD-ROM、或本领域中所知的任何其他形式的存储介质中。存储介质可耦合至处理器以使得该处理器能从/向该存储介质读写信

息。在替换方案中,存储介质可以被整合到处理器。

[0148] 本领域技术人员将可进一步领会,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。

[0149] 本文所述的本发明的各种特征可实现于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。

100

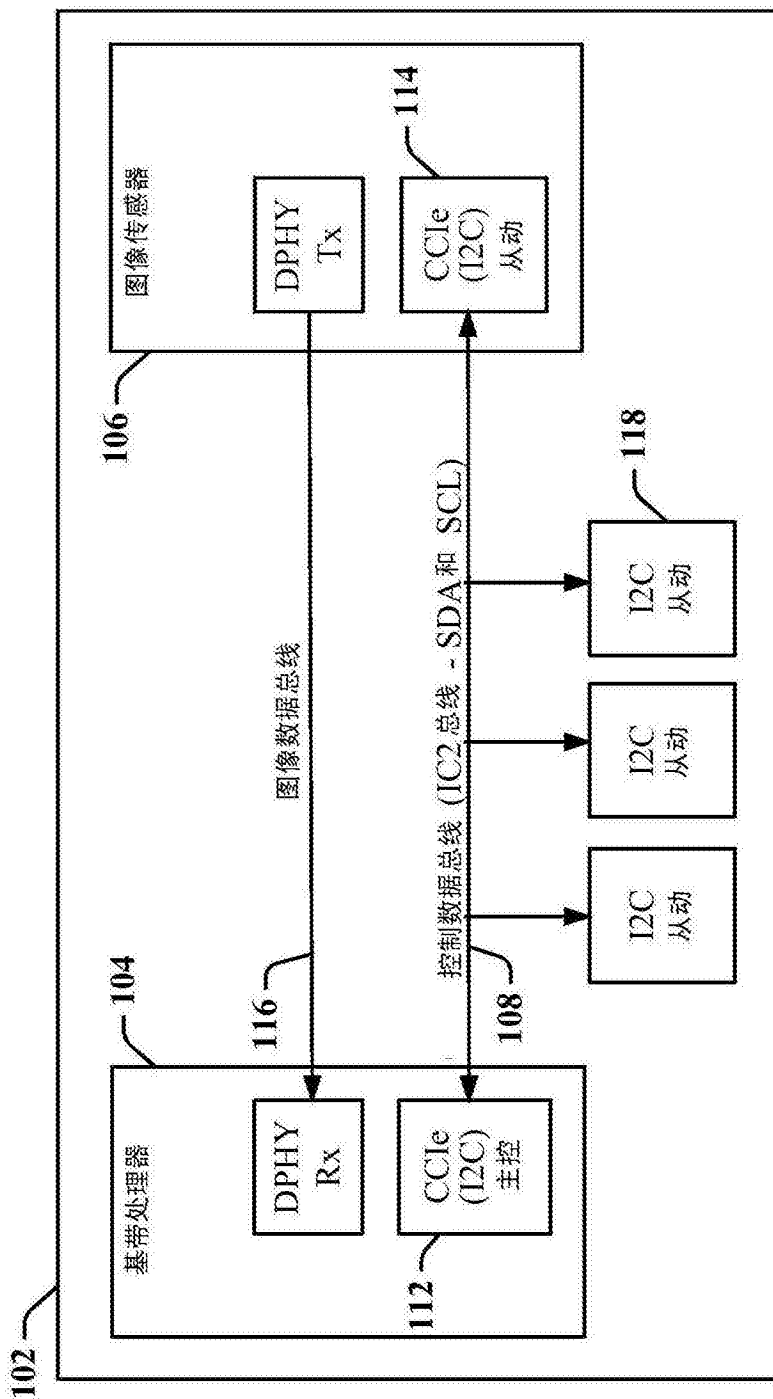


图1

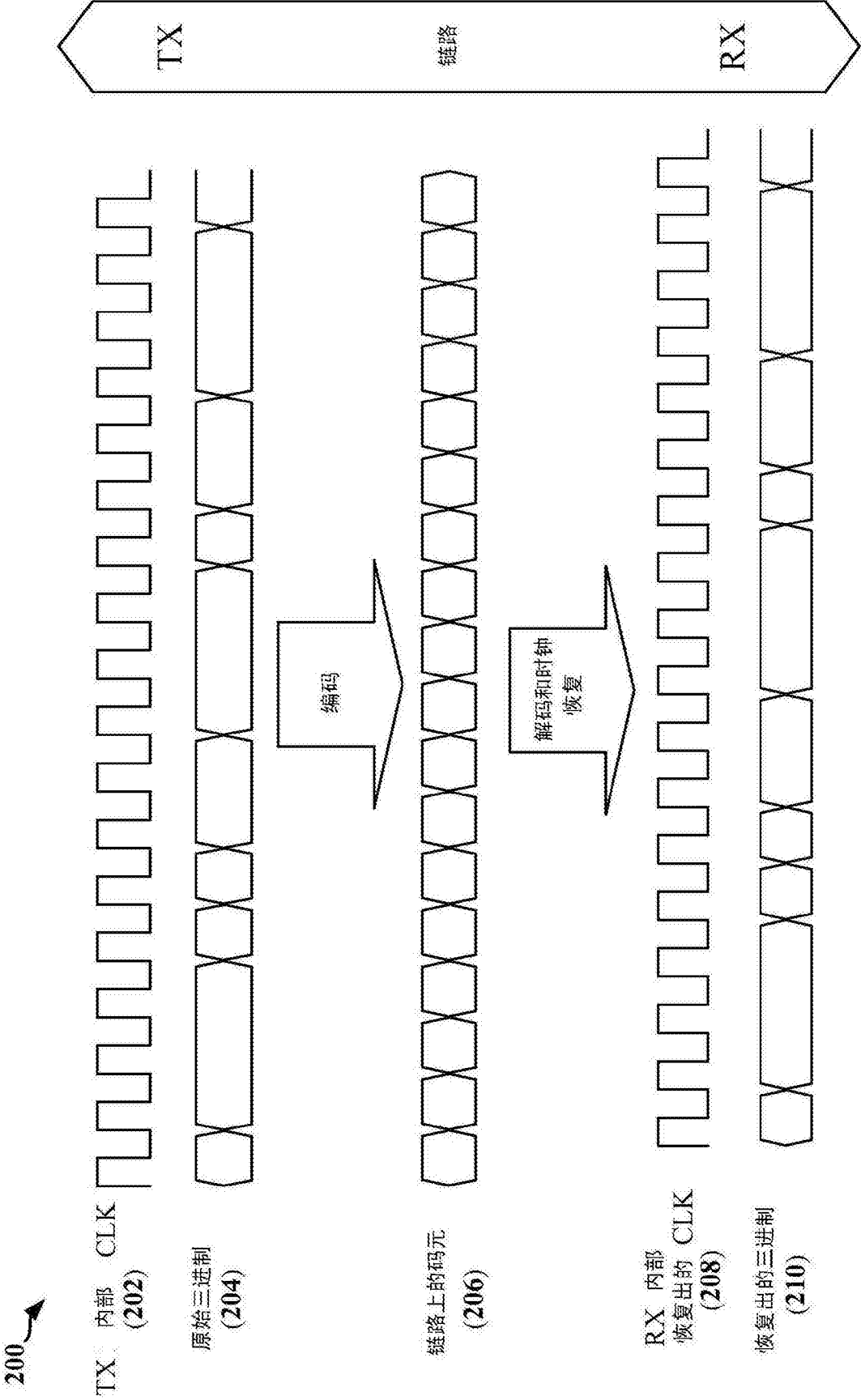


图2

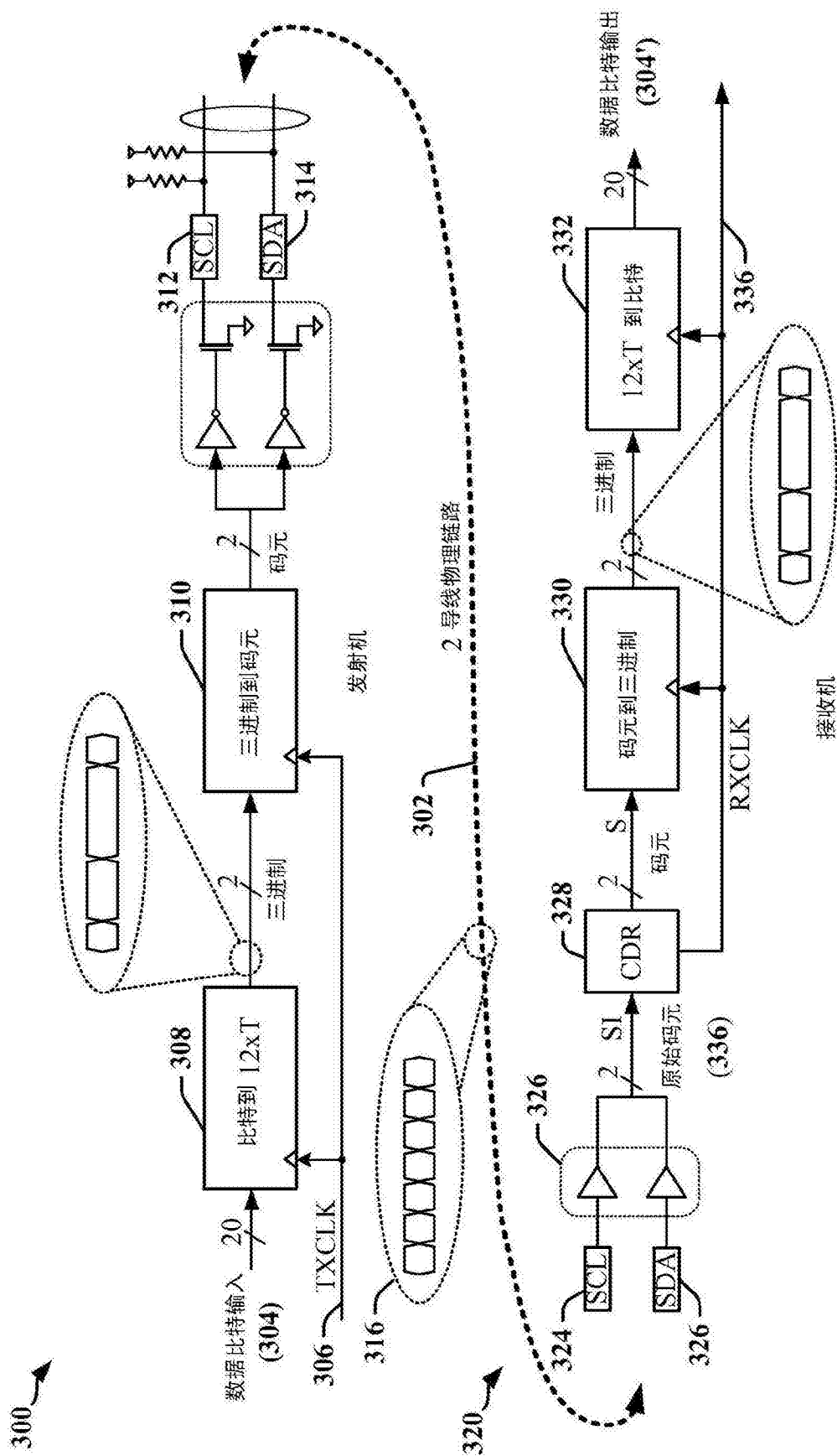


图3

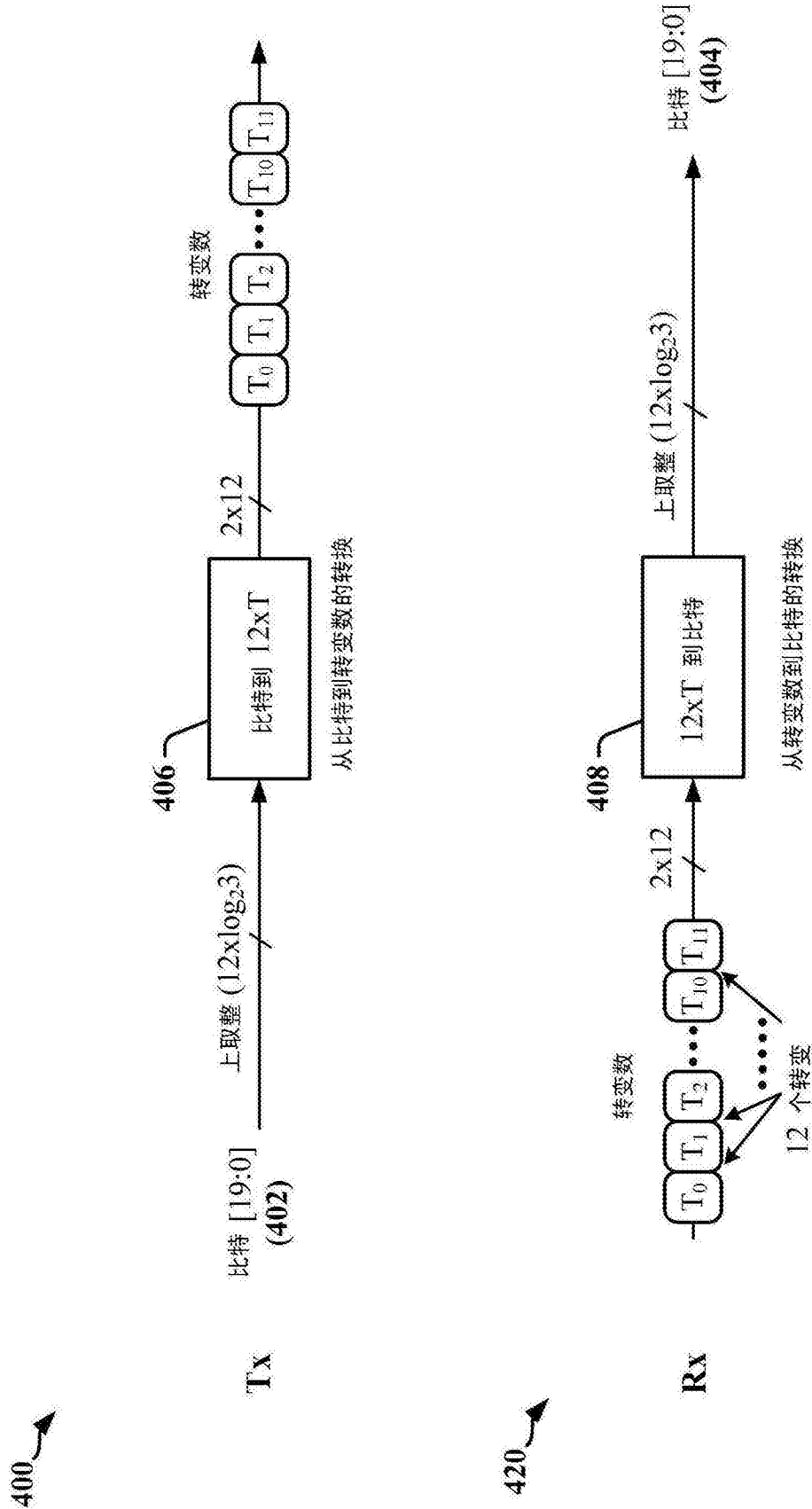


图4

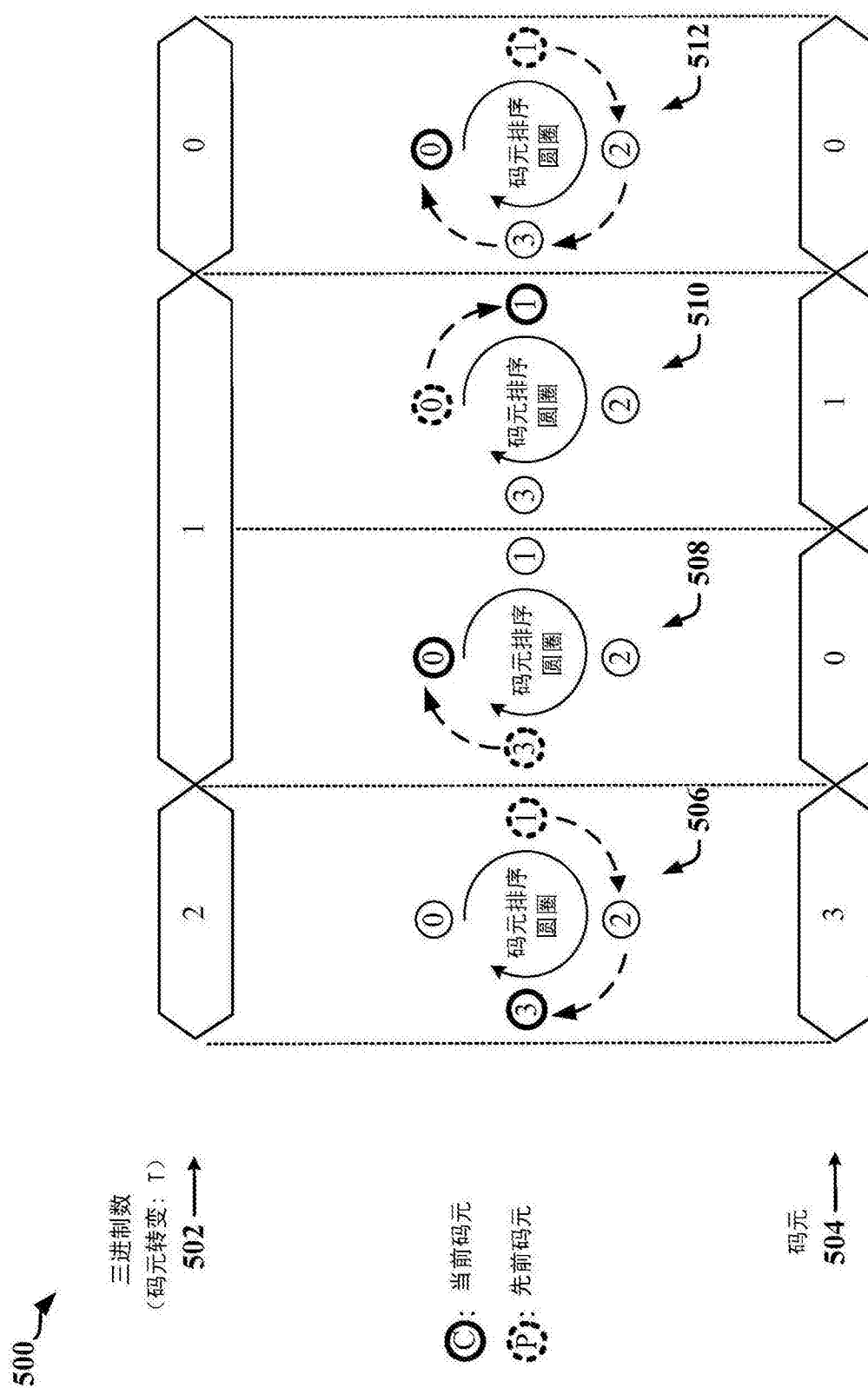


图5

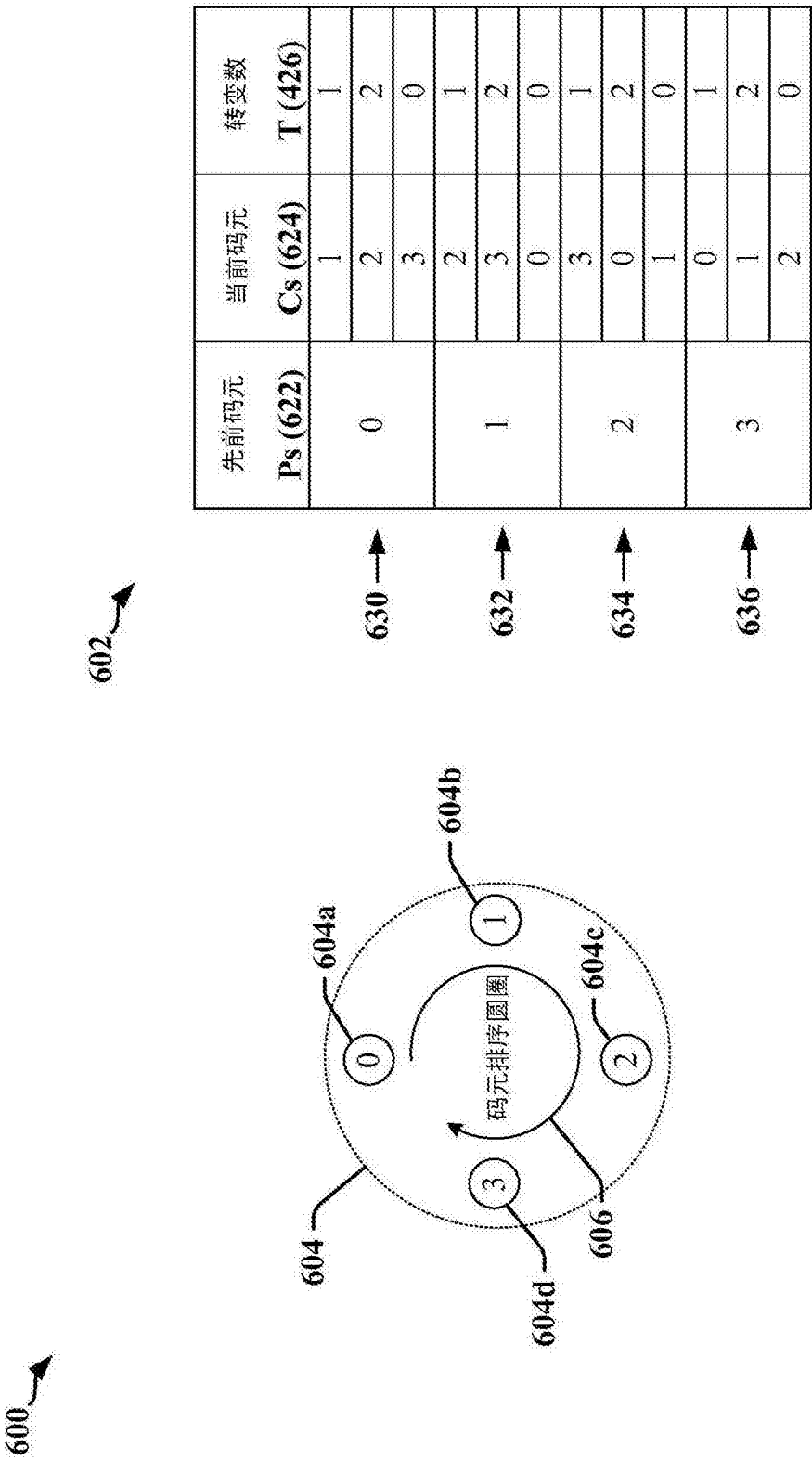


图6

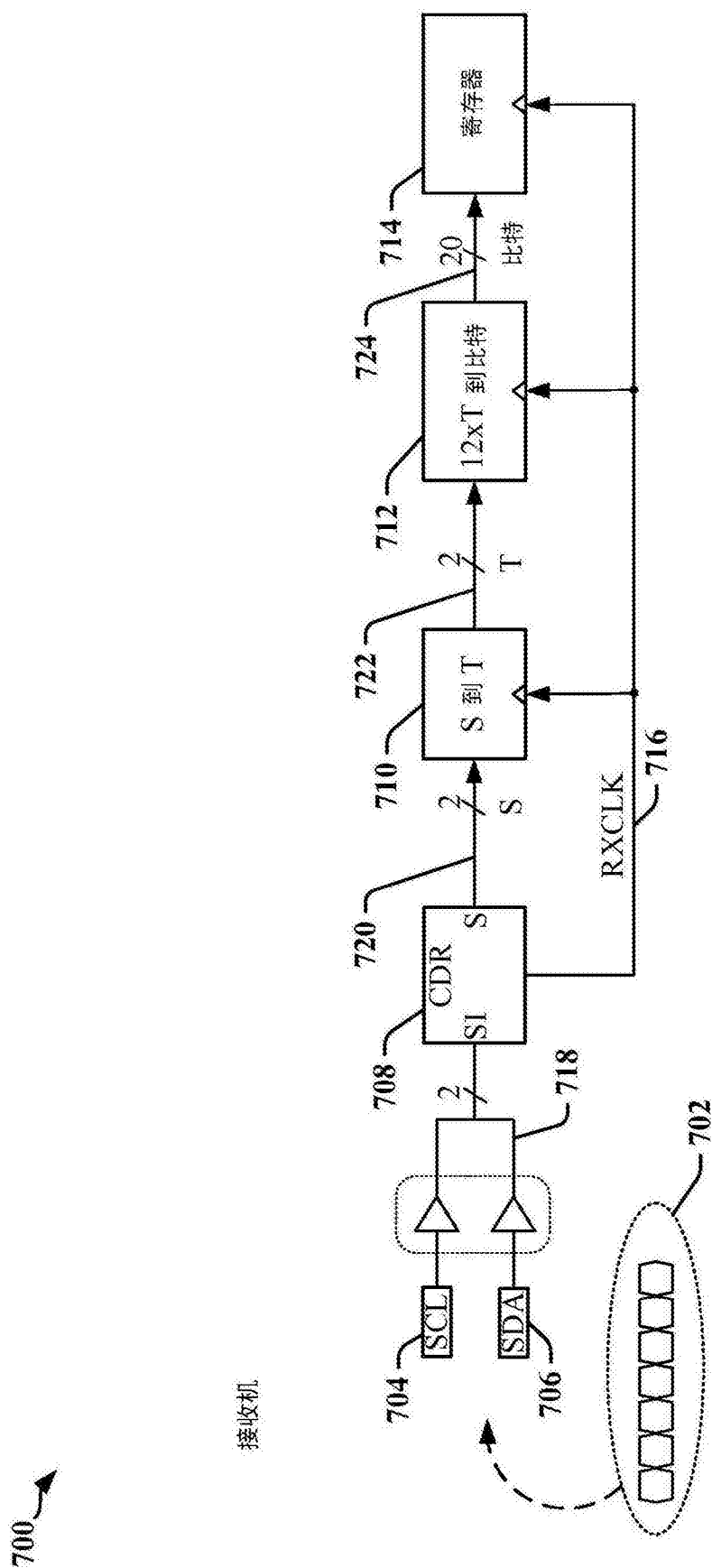


图7

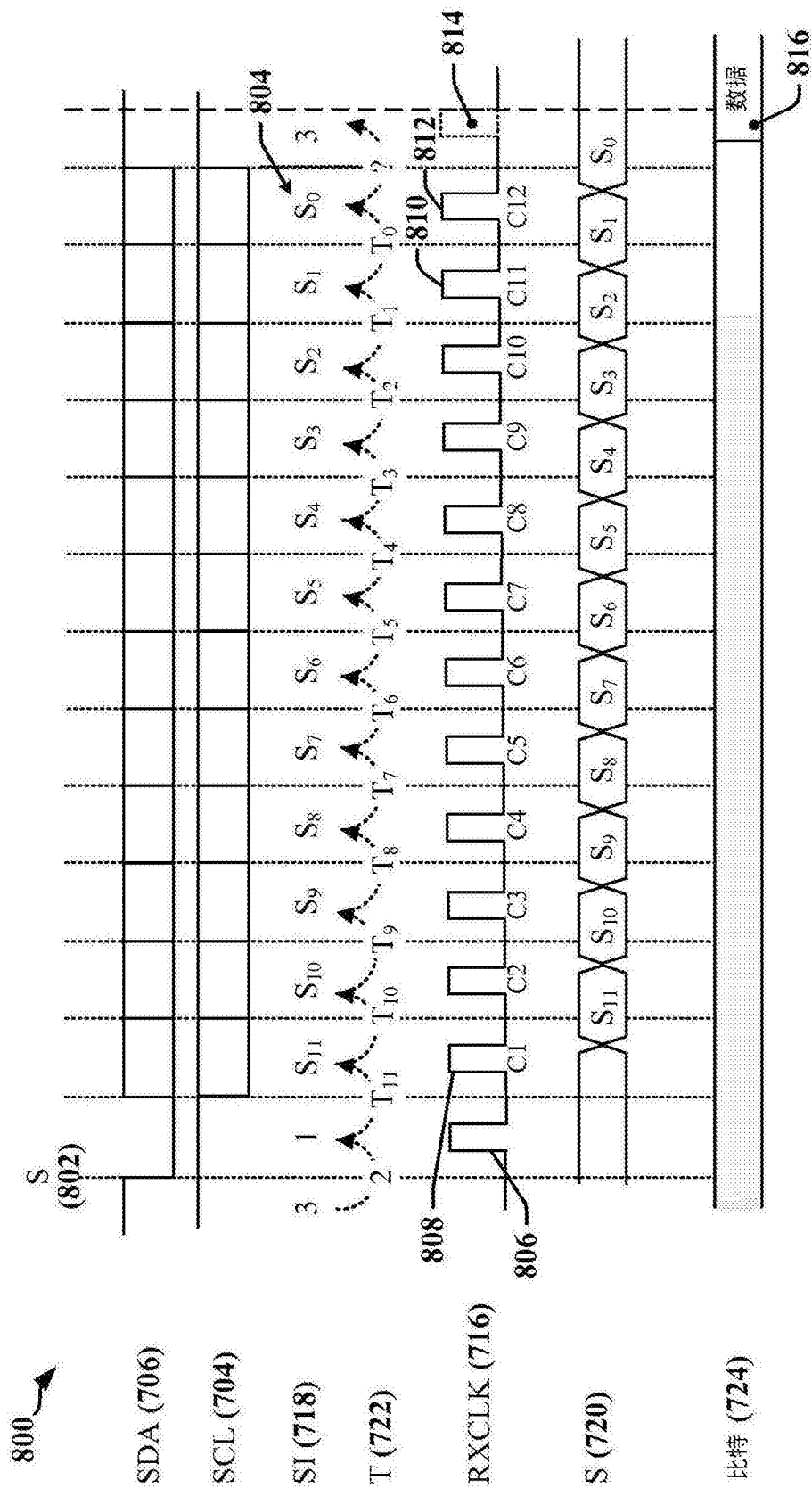


图8

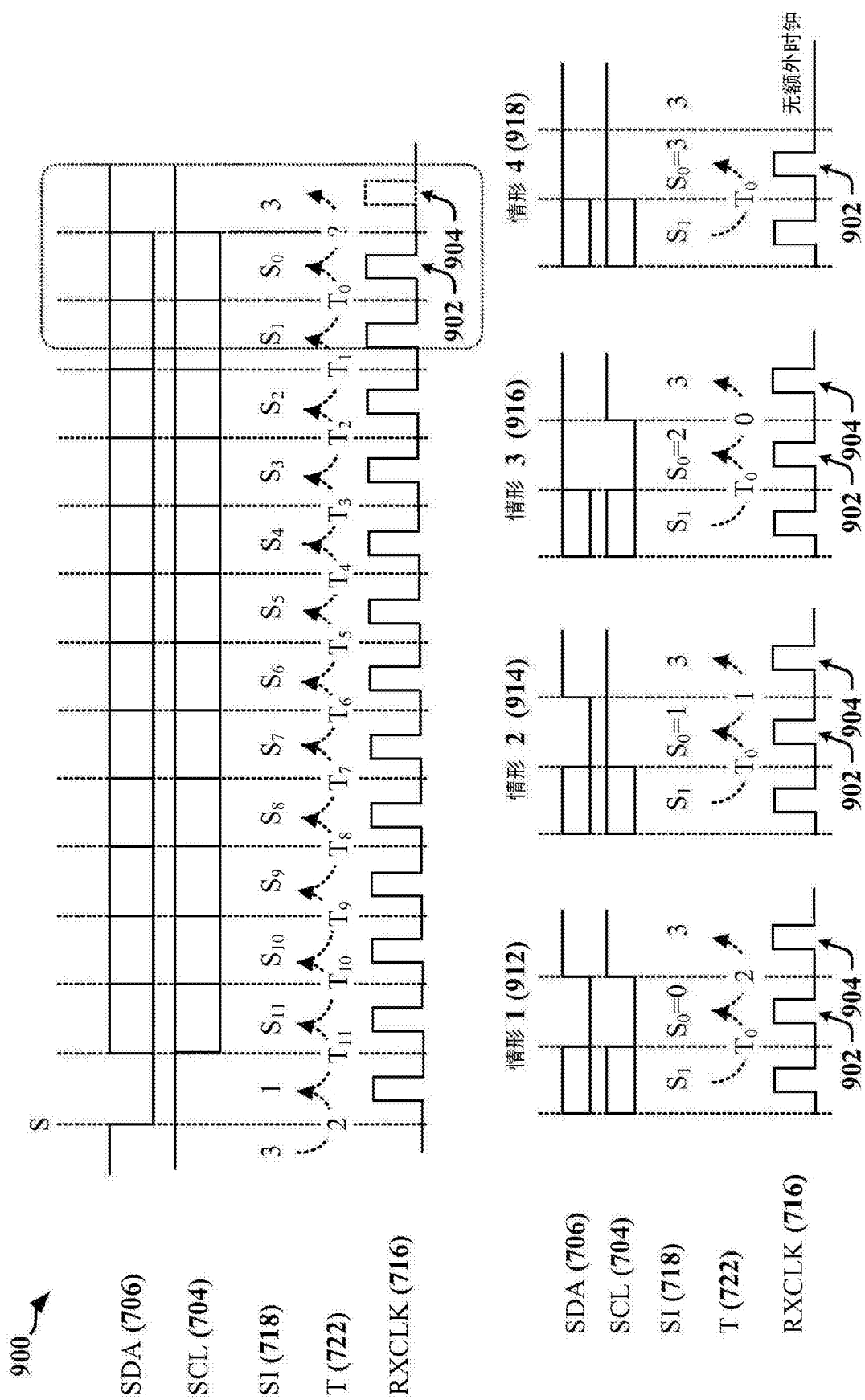


图9

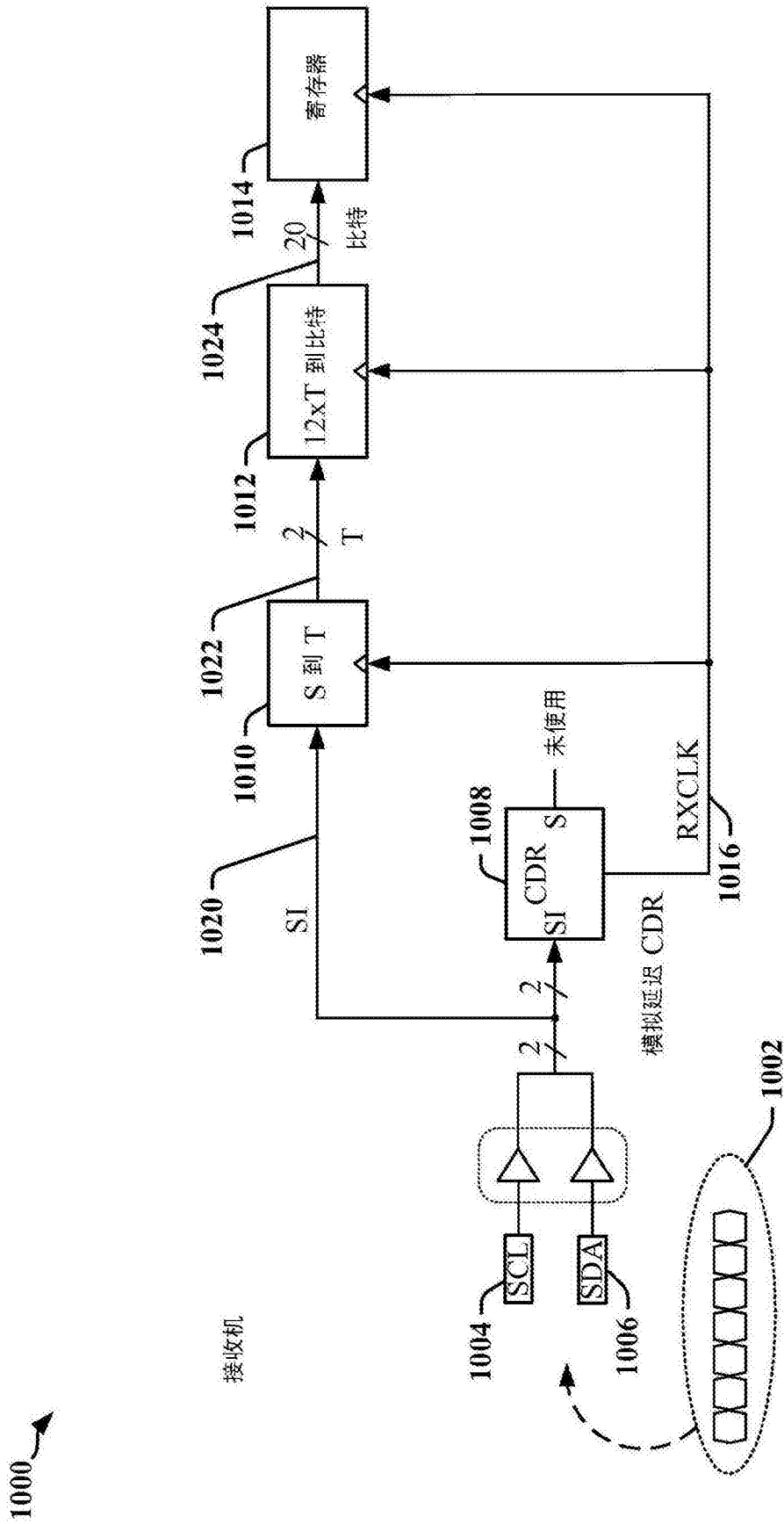
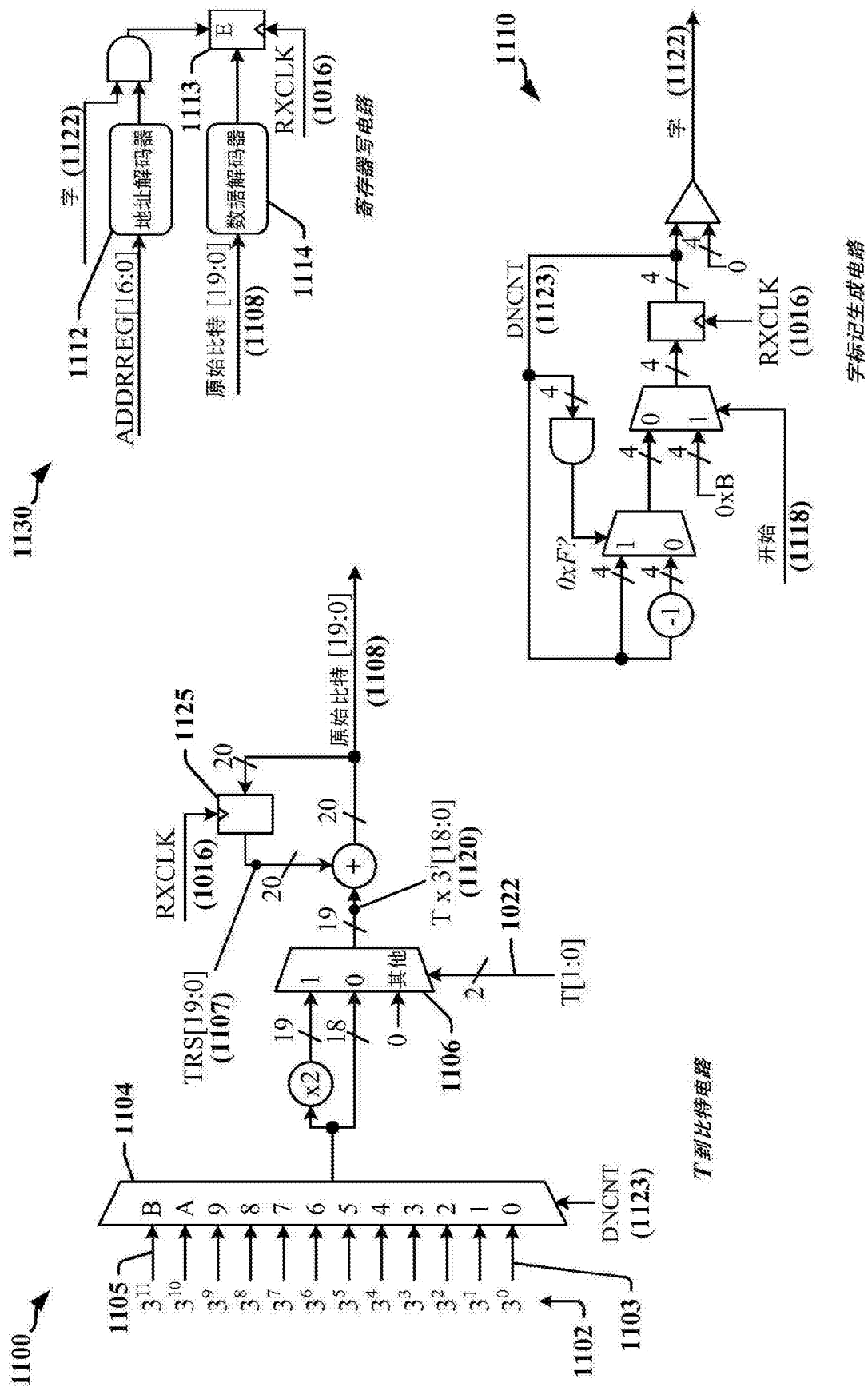


图10



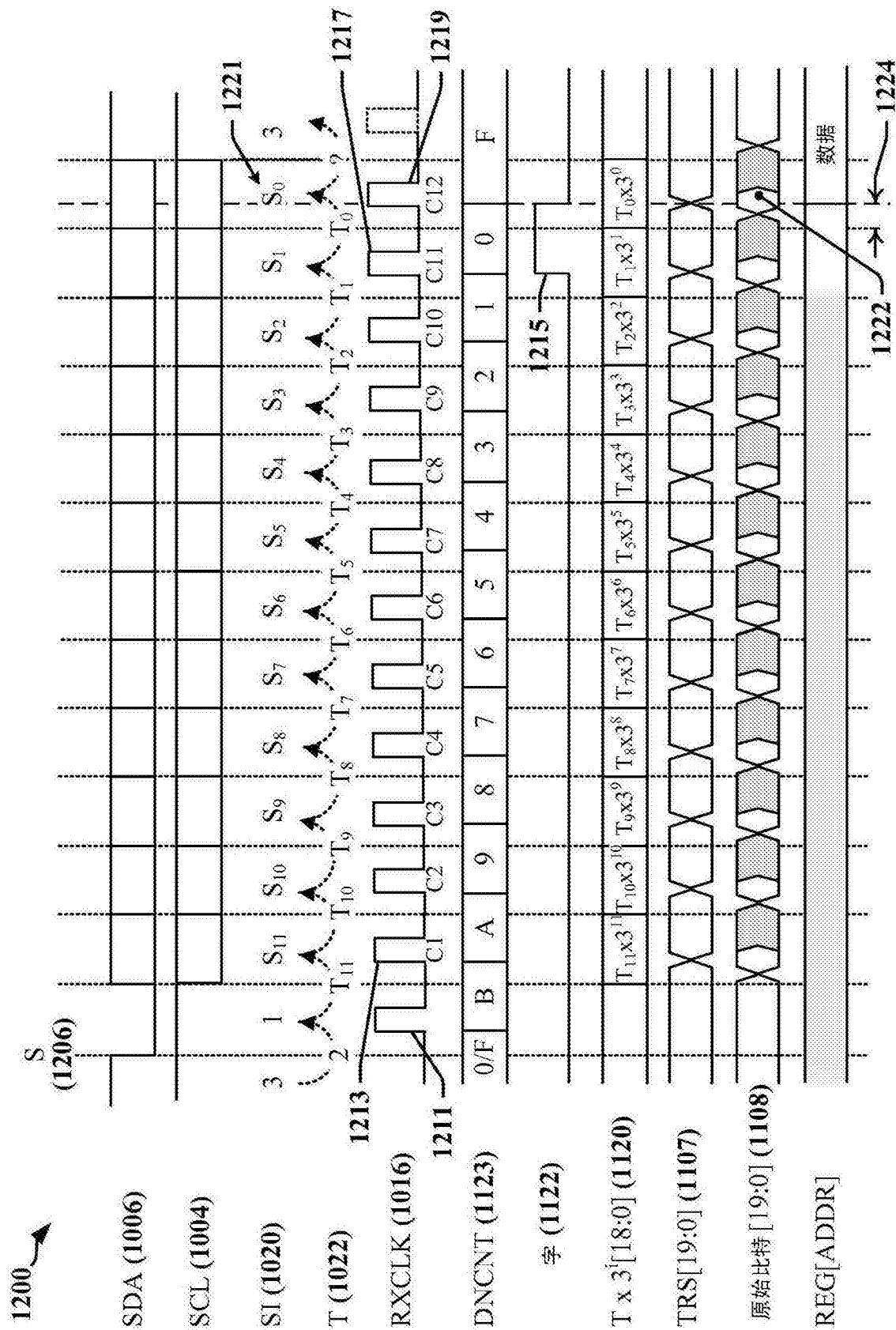


图12

1300 ↗

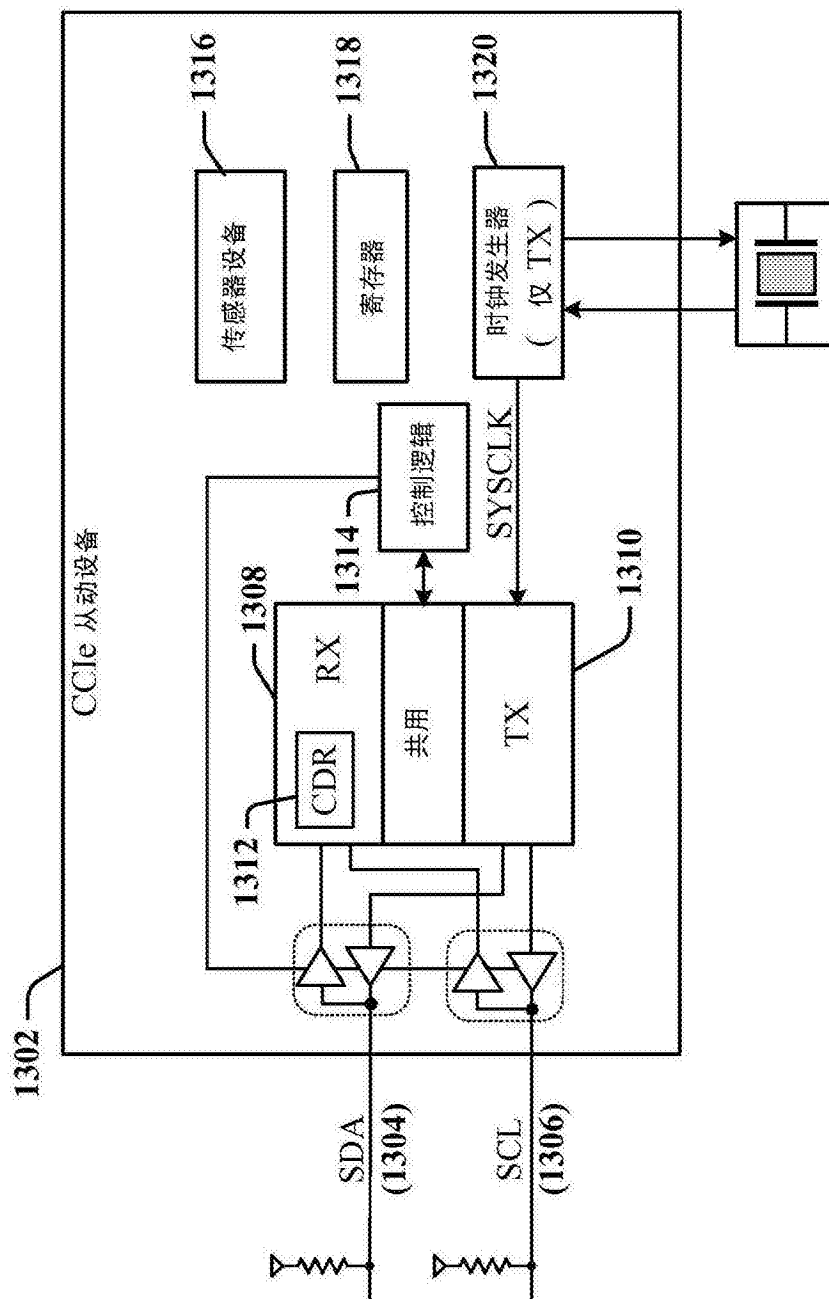


图13

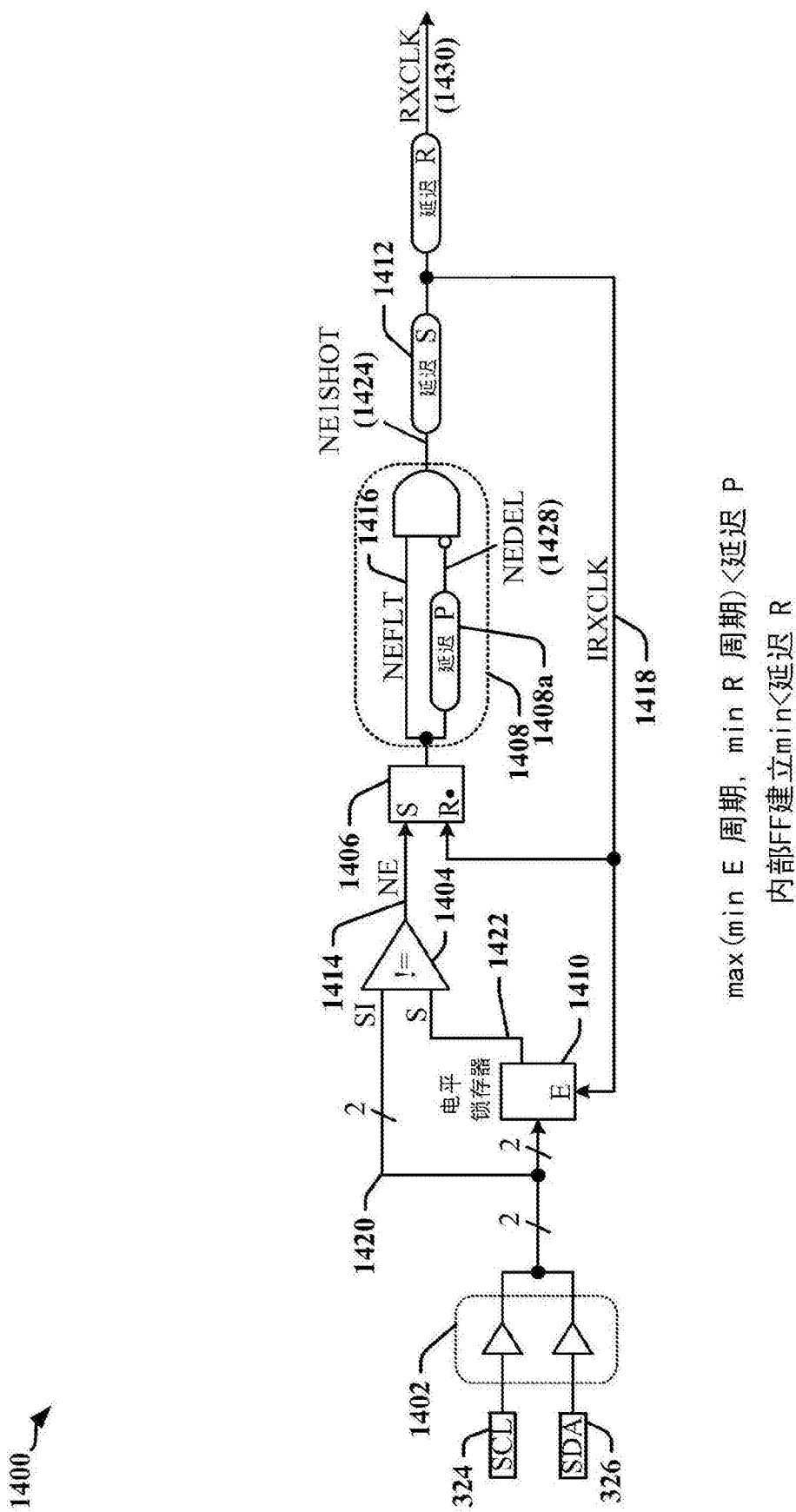


图14

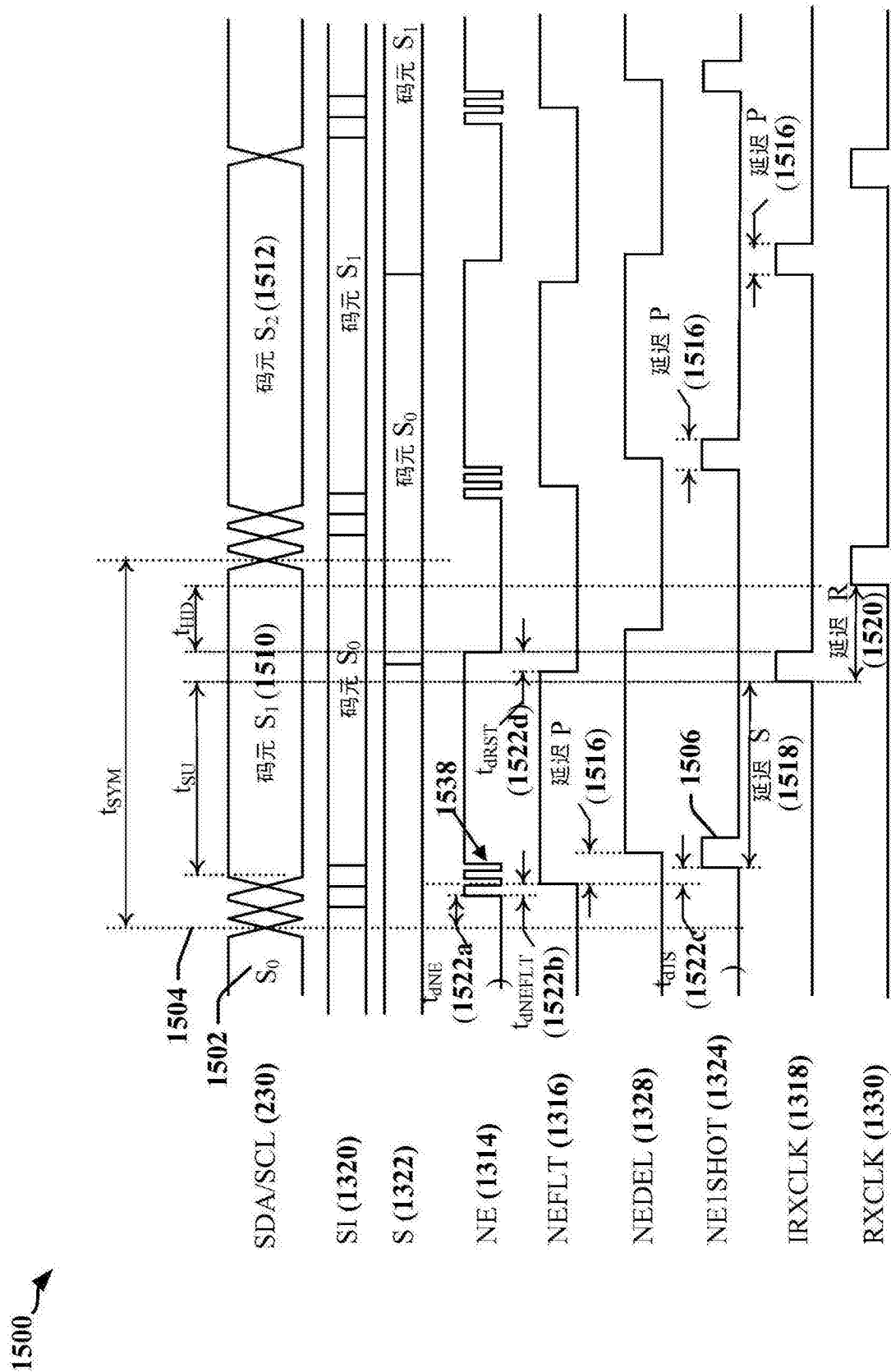


图15

1600 ↗

比特

3^{11}

$\times$

T_{11}

$=$

3^{10}

$\times$

T_{10}

$+$

3^9

$\times$

T_9

$+$

3^8

$\times$

T_8

$+$

3^7

$\times$

T_7

$+$

3^6

$\times$

T_6

$+$

3^5

$\times$

T_5

$+$

3^4

$\times$

T_4

$+$

3^3

$\times$

T_3

$+$

3^2

$\times$

T_2

$+$

3

$\times$

T_1

$+$

T_0

$+$

图16

1700 ↗

T_{11}	$=$	$\frac{\text{比特}}{3^{11}}$	M_{11}	$=$	$\frac{\text{比特}}{3^{11}}$
T_{10}	$=$	$\frac{M_{11}}{3^{10}}$	M_{10}	$=$	$\frac{M_{11}}{3^{10}}$
T_9	$=$	$\frac{M_{10}}{3^9}$	M_9	$=$	$\frac{M_{10}}{3^9}$
T_8	$=$	$\frac{M_9}{3^8}$	M_8	$=$	$\frac{M_9}{3^8}$
T_7	$=$	$\frac{M_8}{3^7}$	M_7	$=$	$\frac{M_8}{3^7}$
T_6	$=$	$\frac{M_7}{3^6}$	M_6	$=$	$\frac{M_7}{3^6}$
T_5	$=$	$\frac{M_6}{3^5}$	M_5	$=$	$\frac{M_6}{3^5}$
T_4	$=$	$\frac{M_5}{3^4}$	M_4	$=$	$\frac{M_5}{3^4}$
T_3	$=$	$\frac{M_4}{3^3}$	M_3	$=$	$\frac{M_4}{3^3}$
T_2	$=$	$\frac{M_3}{3^2}$	M_2	$=$	$\frac{M_3}{3^2}$
T_1	$=$	$\frac{M_2}{3}$	M_1	$=$	$\frac{M_2}{3}$
T_0	$=$	M_1			

图17

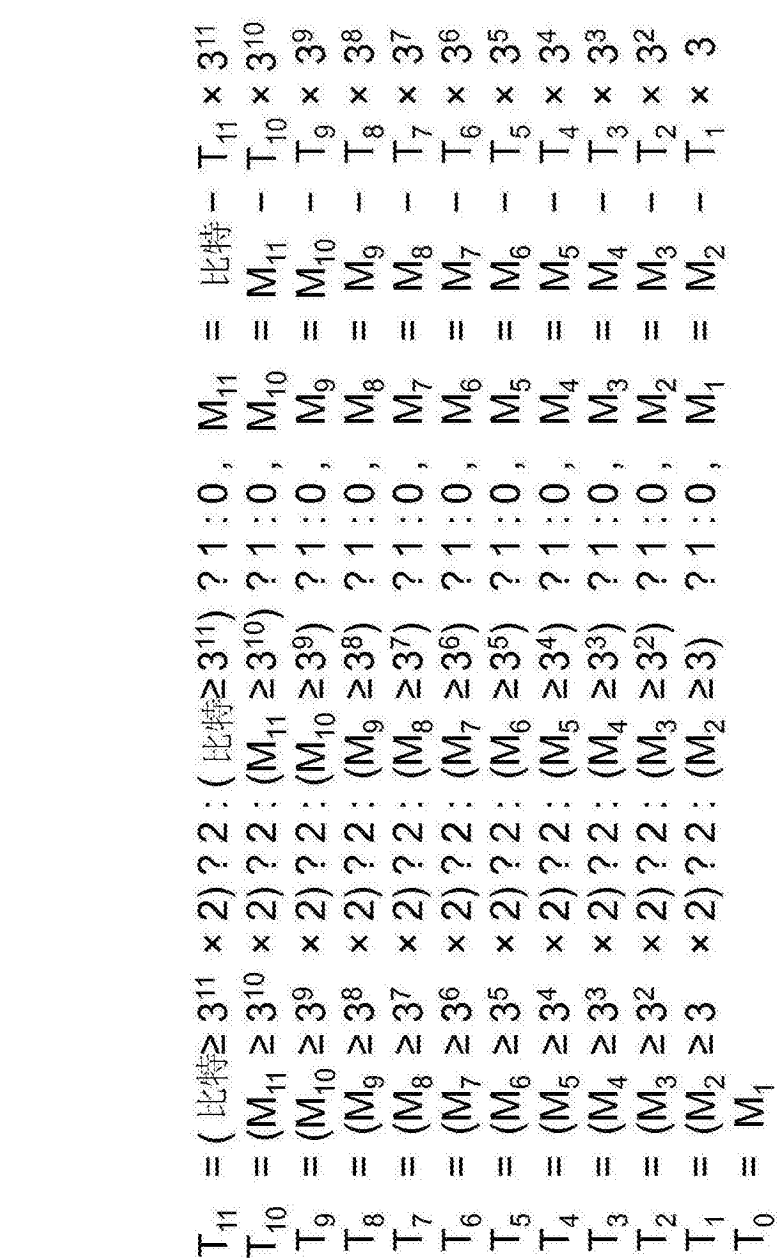


图18

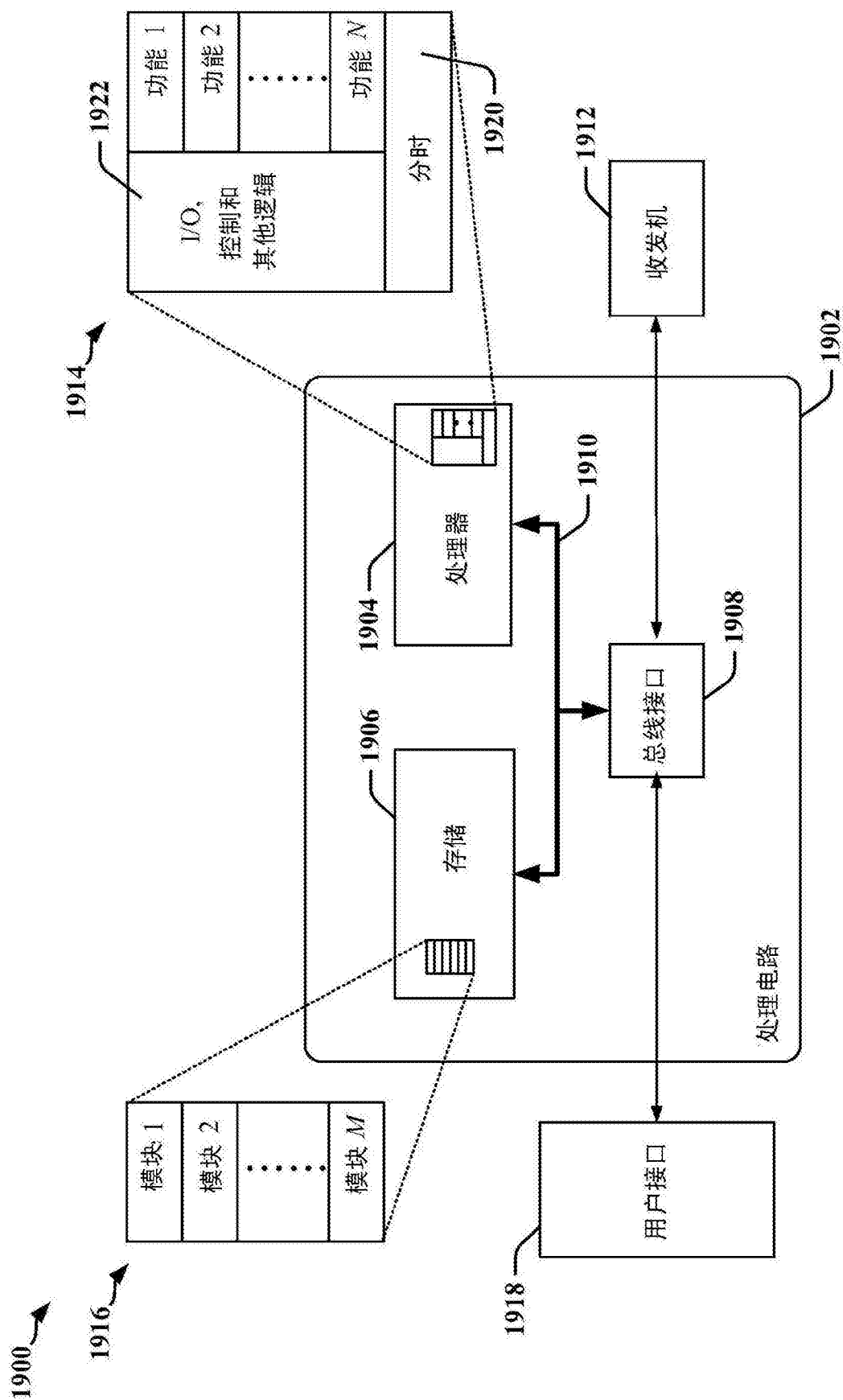


图19

2000

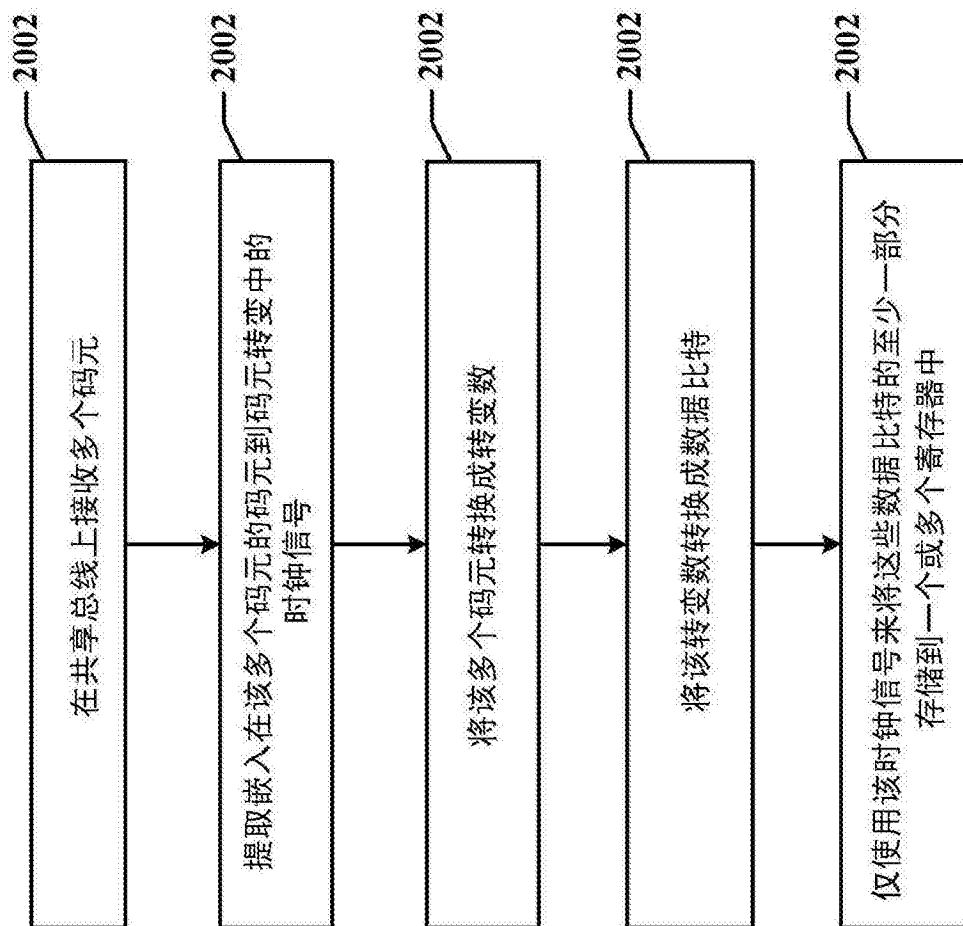


图20

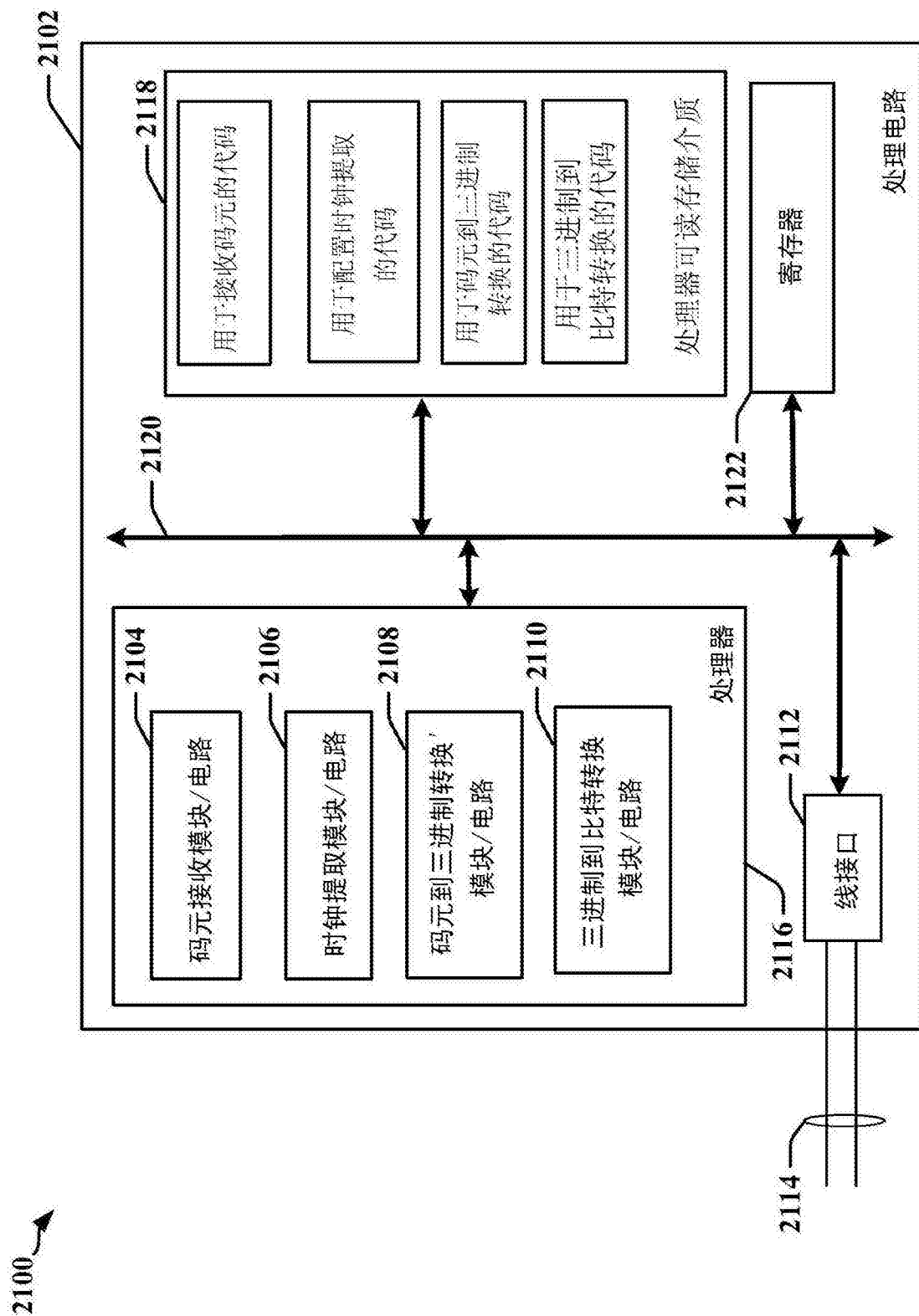


图21