



(12)发明专利

(10)授权公告号 CN 105009535 B

(45)授权公告日 2018.05.18

(21)申请号 201480012199.8

(22)申请日 2014.03.07

(65)同一申请的已公布的文献号
申请公布号 CN 105009535 A

(43)申请公布日 2015.10.28

(30)优先权数据

61/774,408 2013.03.07 US

61/774,247 2013.03.07 US

61/778,768 2013.03.13 US

14/199,898 2014.03.06 US

(85)PCT国际申请进入国家阶段日
2015.09.02

(86)PCT国际申请的申请数据
PCT/US2014/021979 2014.03.07

(87)PCT国际申请的公布数据
W02014/138644 EN 2014.09.12

(73)专利权人 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 G·A·威利 C·李
J·张

(74)专利代理机构 上海专利商标事务所有限公
司 31100

代理人 李小芳

(51)Int.Cl.

H04L 25/49(2006.01)

H04L 25/493(2006.01)

H04L 25/02(2006.01)

(56)对比文件

US 2008212709 A1, 2008.09.04,

US 2008159432 A1, 2008.07.03,

US 4839907 A, 1989.06.13,

审查员 何思佳

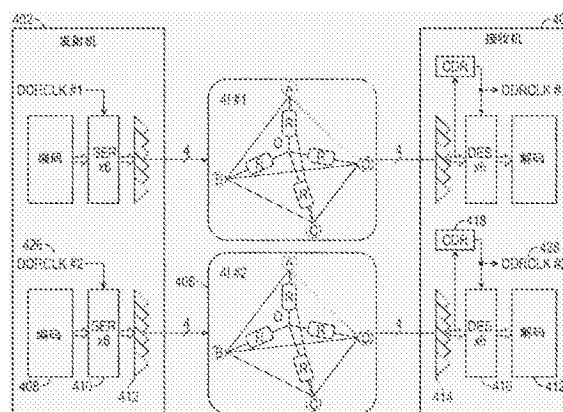
权利要求书3页 说明书12页 附图30页

(54)发明名称

用于将时钟信息嵌入在信号状态转变中的
多导线信令的转码方法

(57)摘要

提供了一种用于执行多导线信令编码的方法,其中时钟信号被编码在码元转变内。数据比特序列被转换成多个(m个)转变数量。每一转变数量被转换成来自顺序码元编号集合的顺序码元编号。顺序码元编号被转换成可以在多个差分驱动器上传送的原始码元。原始码元散布在多条(n条)导线上传送,其中时钟信号被有效地嵌入在原始码元的传输中,因为从转变数量至顺序码元编号的转换保证了没有两个连贯原始码元是相同的。原始码元被保证具有跨多条(n条)导线中的所有导线对的非零差分电压。



1. 一种用于执行多导线信令编码的方法,包括:
将数据比特序列转换成 m 个的多个转变数量;
使用适配成产生不相同连贯顺序编号的第一预定义映射将每一转变数量转换成来自顺序编号集合的顺序编号;
使用从所述不相同连贯顺序编号产生不相同连贯原始码元的第二预定义映射将所述顺序编号中的每一者转换成原始码元;以及
经由多个差分驱动器且散布在 n 条的多条导线上传送每个原始码元,以使得能从原始码元转变中提取时钟信号。
2. 如权利要求1所述的方法,其特征在于,将每一转变数量转换成来自顺序编号集合的顺序编号包括:
将多个转变数量转换成顺序编号。
3. 如权利要求1所述的方法,其特征在于,将所述顺序编号转换成原始码元包括:
将所述顺序编号转换成多个原始码元。
4. 如权利要求1所述的方法,其特征在于,所述 n 条的多条导线大于或等于3。
5. 如权利要求1所述的方法,其特征在于,所述 n 条的多条导线大于或等于4。
6. 如权利要求1所述的方法,其特征在于,所述原始码元被保证具有跨所述 n 条的多条导线中的所有导线对的非零差分电压。
7. 如权利要求1所述的方法,其特征在于,对于跨所述 n 条导线中的各导线对的 n 阶乘差分信号, r^n 个可能的不同状态由所述转变数量来表示,其中 r 是 $n!-1$ 。
8. 如权利要求1所述的方法,其特征在于,所述顺序编号是从基于自紧挨着的前一顺序码元编号的转变的转变数量来选择的。
9. 如权利要求1所述的方法,其特征在于,进一步包括:
使用所述时钟信号来同步原始码元传输。
10. 一种用于执行多导线信令编码的编码电路,包括:
比特至转变数量转换器,用于将数据比特序列转换成 m 个的多个转变数量;
转变数量至顺序编号转换器,用于使用适配成产生不相同连贯顺序编号的第一预定义映射将每一转变数量转换成来自顺序编号集合的顺序编号;
顺序编号至原始码元转换器,用于基于从所述不相同连贯顺序编号产生不相同连贯原始码元的第二预定义映射将所述顺序编号中的每一者转换成原始码元;以及
多个差分驱动器,用于散布在 n 条的多条导线上传送每个原始码元,以使得能从原始码元转变中提取时钟信号。
11. 如权利要求10所述的编码电路,其特征在于,将每一转变数量转换成来自顺序编号集合的顺序编号包括将多个转变数量转换成顺序编号。
12. 如权利要求10所述的编码电路,其特征在于,将所述顺序编号转换成原始码元包括将所述顺序编号转换成多个原始码元。
13. 如权利要求10所述的编码电路,其特征在于,所述 n 条的多条导线大于或等于3。
14. 如权利要求10所述的编码电路,其特征在于,所述 n 条的多条导线大于或等于4。
15. 如权利要求10所述的编码电路,其特征在于,所述原始码元被保证具有跨所述 n 条导线中的所有导线对的非零差分电压。

16. 如权利要求10所述的编码电路,其特征在于,对于跨所述 n 条导线中的各导线对的 n 阶乘差分信号, r^n 个可能的不同状态由所述转变数量来表示,其中 r 是 $n!-1$ 。

17. 如权利要求10所述的编码电路,其特征在于,所述多个差分驱动器等于 nC_2 ,其中 $nC_2 = n(n-1)/2$ 。

18. 如权利要求10所述的编码电路,其特征在于,所述多个差分驱动器等于 n 。

19. 如权利要求10所述的编码电路,其特征在于,所述顺序编号是从基于自紧挨着的前一顺序码元编号的转变的所述转变数量来选择的。

20. 一种编码电路,包括:

用于将数据比特序列转换成 m 个的多个转变数量的装置;

用于使用适配成产生不相同连贯顺序编号的第一预定义映射将每一转变数量转换成来自顺序编号集合的顺序编号的装置;

用于基于从所述不相同连贯顺序编号产生不相同连贯原始码元的第二预定义映射将所述顺序编号中的每一者转换成原始码元的装置;以及

用于经由多个差分驱动器且散布在 n 条的多条导线上传送每个原始码元的装置,以使得能从原始码元转变中提取时钟信号。

21. 一种用于执行多导线信令解码的方法,包括:

经由多个差分接收机接收散布在 n 条的多条导线上的原始码元;

使用第一预定义映射将所述原始码元转换成来自顺序编号集合的顺序编号;

使用第二预定义映射将每一顺序编号转换成转变数量;

将多个转变数量转换成数据比特序列;以及

从接收到的原始码元转变中提取时钟信号。

22. 如权利要求21所述的方法,其特征在于,将每一顺序编号转换成转变数量包括:

将顺序编号转换成多个转变数量。

23. 如权利要求21所述的方法,其特征在于,将所述原始码元转换成来自顺序编号集合的顺序编号包括:

将多个原始码元转换成顺序编号。

24. 如权利要求21所述的方法,其特征在于,所述 n 条的多条导线大于或等于3。

25. 如权利要求21所述的方法,其特征在于,所述 n 条的多条导线大于或等于4。

26. 如权利要求21所述的方法,其特征在于,所述原始码元被保证具有跨所述 n 条导线中的所有导线对的非零差分电压。

27. 如权利要求21所述的方法,其特征在于,对于跨 n 个驱动器的 n 阶乘差分信令, r^n 个可能的不同状态由所述转变数量来表示,其中 r 是 $n!-1$ 。

28. 如权利要求21所述的方法,其特征在于,所述转变数量是从所述顺序编号并基于紧挨着的前一顺序码元编号来选择的。

29. 如权利要求21所述的方法,其特征在于,进一步包括:

使用所述时钟信号来同步原始码元接收。

30. 一种用于执行多导线信令解码的解码电路,包括

多个差分接收机,用于接收散布在 n 条的多条导线上的原始码元;

原始码元至顺序编号转换器,用于使用第一预定义映射将所述原始码元转换成来自顺

序编号集合的顺序编号；

顺序编号至转变数量转换器，用于使用第二预定义映射将每一顺序编号转换成转变数量；

转变数量至比特转换器，用于将多个转变数量转换成数据比特序列；以及
时钟数据恢复电路，用于从接收到的原始码元转变中提取时钟信号。

31. 如权利要求30所述的解码电路，其特征在于，将每一顺序编号转换成转变数量包括将顺序编号转换成多个转变数量。

32. 如权利要求30所述的解码电路，其特征在于，将所述原始码元转换成来自顺序编号集合的顺序编号包括将多个原始码元转换成顺序编号。

33. 如权利要求30所述的解码电路，其特征在于，所述n条的多条导线大于或等于3。

34. 如权利要求30所述的解码电路，其特征在于，所述n条的多条导线大于或等于4。

35. 如权利要求30所述的解码电路，其特征在于，所述原始码元被保证具有跨所述n条的多条导线中的所有导线对的非零差分电压。

36. 如权利要求30所述的解码电路，其特征在于，对于跨n个驱动器的n阶乘差分信令， r^m 个可能的不同状态由所述转变数量来表示，其中r是 $n!-1$ 。

37. 如权利要求30所述的解码电路，其特征在于，所述多个差分驱动器等于 nC_2 ，其中 $nC_2 = n(n-1)/2$ 。

38. 如权利要求30所述的解码电路，其特征在于，所述多个差分驱动器等于n。

39. 如权利要求30所述的解码电路，其特征在于，所述转变数量是从所述顺序编号并基于紧挨着的前一顺序码元编号来选择的。

40. 如权利要求30所述的解码电路，其特征在于，原始码元接收是使用所述时钟信号来同步的。

41. 一种用于执行多导线信令解码的解码电路，包括：

用于经由多个差分接收机接收散布在n条的多条导线上的原始码元的装置；

用于使用第一预定义映射将所述原始码元转换成来自顺序码元编号集合的顺序码元编号的装置；

用于使用第二预定义映射将每一顺序码元编号转换成转变数量的装置；以及

用于将多个转变数量转换成数据比特序列的装置；以及

用于从接收到的原始码元转变中提取时钟信号的装置。

用于将时钟信息嵌入在信号状态转变中的多导线信令的转码方法

[0001] 根据35U.S.C. §119的优先权要求

[0002] 本专利申请要求以下优先权：

[0003] 2013年3月7日提交的题为“Transcoding Method For Multi-Wire Signaling That Embeds Clock Information In Transition Of Signal State (用于将时钟信息嵌入在信号状态转变中的多导线信令的转码方法)”的美国临时申请No.61/774,408,

[0004] 2013年3月13日提交的题为“Transcoding Method For Multi-Wire Signaling That Embeds Clock Information In Transition Of Signal State (用于将时钟信息嵌入在信号状态转变中的多导线信令的转码方法)”的美国临时申请No.61/778,768,以及

[0005] 2013年3月7日提交的题为“Circuit To Recover A Clock Signal From Multiple Wire Data Signals That Changes State Every State Cycle And Is Immune To Data Inter-Lane Skew As Well As Data State Transition Glitches (用于从多导线数据信号中恢复在每一状态循环改变状态并且对数据通道间偏斜以及数据状态转变毛刺免疫的时钟信号的电路)”的美国临时申请No.61/774,247,所有这些申请被转让给本申请受让人并且由此通过援引明确纳入于此。

技术领域

[0006] 本公开涉及在多信号数据传输的循环内传送和/或编码时钟信号。

背景技术

[0007] 在多信号数据传输(例如,多导线差分信令,诸如3相或N阶乘低电压差分信令(LVDS))中,可进行转码(例如,一种编码到另一种编码的数字到数字数据转换)以通过在每一码元循环导致码元转变来嵌入码元时钟信息,而不是在单独的数据通道(传输路径)中发送时钟信息。通过此种转码来嵌入时钟信息是使时钟与数据信号之间的偏斜最小化并消除用于从数据信号中恢复时钟信息的锁相环(PLL)的必要性的有效方式。

[0008] 一些转码方案(例如,3导线差分信令和/或3相信令)已经实现了嵌入式时钟信息,但缺乏可伸缩性。

[0009] 因此,需要支持在多导线信令(诸如N阶乘差分信令)中使用的任何数量的导线以及将从任何二进制数转码得到的群中的任何数量的码元的高效转码方法。

发明内容

[0010] 提供了一种用于执行多导线信令编码的方法。数据比特序列被转换成多个(m个)转变数量。每一转变数量被转换成来自顺序编号集合的顺序编号。顺序编号被转换成原始码元。原始码元经由多个差分驱动器且散布在多条(n条)导线上传送,其中时钟信号被有效地嵌入在原始码元的传输中,因为从转变数量至顺序编号的转换保证没有两个连贯原始码元是相同的。原始码元传输可使用时钟信号来同步。将每一转变数量转换成来自顺序编号

集合的顺序编号可包括将多个转变数量转换成顺序编号。将顺序编号转换成原始码元可包括将顺序编号转换成多个原始码元。在一个示例中,多条(n 条)导线可以大于或等于3或4。原始码元被保证具有跨多条(n 条)导线中的所有导线对的非零差分电压。对于跨 n 条导线中的各导线对的 n 阶乘差分信号, r^m 个可能的不同状态由转变数量来表示,其中 r 是 $n!-1$ 且 m 是该多个转变数量。顺序编号可以从基于自紧挨着的前一顺序码元编号的转变的转变数量来选择。

[0011] 提供了一种用于执行多导线信令编码的编码电路。比特至转变数量转换器可将数据比特序列转换成多个(m 个)转变数量。转变数量至顺序编号转换器可将每一转变数量转换成来自顺序编号集合的顺序编号。顺序编号至原始码元转换器可将顺序编号转换成原始码元。多个差分驱动器可以散布在多条(n 条)导线上传送原始码元,其中时钟信号被有效地嵌入在原始码元的传输中,因为从转变数量至顺序编号的转换保证没有两个连贯原始码元是相同的。

[0012] 对于跨 n 条导线中的各导线对的 n 阶乘差分信号, r^m 个可能的不同状态由转变数量来表示,其中 r 是 $n!-1$ 且 m 是该多个转变数量。

[0013] 在一个示例中,该多个差分驱动器等于 nC_2 ,其中 $nC_2 = n(n-1)/2$ 。在另一示例中,该多个差分驱动器等于 n 。

[0014] 顺序编号可以从基于自紧挨着的前一顺序码元编号的转变的转变数量来选择。

[0015] 还提供了一种用于执行多导线信令解码的方法。经由多个差分接收机接收散布在多条(n 条)导线上的原始码元。原始码元可被转换成来自顺序编号集合的顺序编号。每一顺序编号可被转换成转变数量。多个转变数量可被转换成数据比特序列。可以从原始码元接收中提取时钟信号。原始码元接收可使用该时钟信号来同步。

[0016] 将每一顺序编号转换成转变数量可包括将顺序编号转换成多个转变数量。将原始码元转换成来自顺序编号集合的顺序编号可包括将多个原始码元转换成顺序编号。在各种示例中,多条(n 条)导线大于或等于3或4。原始码元被保证具有跨这 n 条导线中的所有导线对的非零差分电压。对于跨 n 个驱动器的 n 阶乘差分信令, r^m 个可能的不同状态由转变数量来表示,其中 r 是 $n!-1$ 。转变数量可以从顺序编号并基于紧挨着的前一顺序码元编号来选择。

[0017] 还提供了一种用于执行多导线信令解码的解码电路。多个差分接收机可以接收散布在多条(n 条)导线上的原始码元。原始码元至顺序编号转换器可将原始码元转换成来自顺序编号集合的顺序编号。顺序编号至转变数量转换器可将每一顺序编号转换成转变数量。转变数量至比特转换器可将多个转变数量转换成数据比特序列。时钟数据恢复电路可以从原始码元接收中提取时钟信号。原始码元接收可使用该时钟信号来同步。

[0018] 在一个示例中,该多个差分驱动器等于 nC_2 ,其中 $nC_2 = n(n-1)/2$ 。在另一示例中,该多个差分驱动器等于 n 。转变数量可以从顺序编号并基于紧挨着的前一顺序码元编号来选择。

附图说明

[0019] 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。

- [0020] 图1解说了在发射机设备和接收机设备之间使用的四阶乘调制数据编码/解码方案。
- [0021] 图2解说了在发射机设备和接收机设备之间使用的另一四阶乘调制数据编码/解码方案。
- [0022] 图3解说了在发射机设备和接收机设备之间使用的另一四阶乘调制数据编码/解码方案。
- [0023] 图4解说了经由多导线传输路径在发射机设备和接收机设备之间使用的另一四阶乘调制数据编码/解码方案。
- [0024] 图5是其中时钟被集成到码元转变中的发射机(编码器)设备的框图。
- [0025] 图6是具有集成时钟恢复的接收机(解码器)设备的框图。
- [0026] 图7解说了发射机处从比特至码元转变数量以及随后在接收机处从码元转变数量至比特的转换。
- [0027] 图8解说了顺序码元编号与码元转变数量之间的转换。
- [0028] 图9是解说原始码元(对于 $n=3$ 导线系统)与顺序码元编号之间的转换的表。
- [0029] 图10解说示出输入驱动器和输出接收机以及各节点之间的电流的3导线端接网络。
- [0030] 图11(包括图11A和11B)解说示出输入驱动器和输出接收机以及各节点之间的电流的4导线端接网络。
- [0031] 图12解说示出发射机驱动器、接收机驱动器和4导线端接网络的发射机至接收机转码方案。
- [0032] 图13(包括图13A和13B)是对于 $n=4$ 示出原始码元与具有零差分电压的那些码元的各种组合的表。
- [0033] 图14解说具有已经从图13A和13B获得的非零差分电压原始码元(Sraw)的表。
- [0034] 图15是解说对于 n 导线系统的比特至码元转变数量的转换的框图。
- [0035] 图16是解说对于3导线系统(对于 $n=3$ 且每一群有7个码元)的比特至码元转变数量的转换的示例的示例性编码器的框图。
- [0036] 图17是解说对于3导线系统(对于 $n=3$ 且每一群有7个码元)的码元转变数量至比特的转换的示例的示例性解码器的框图。
- [0037] 图18是解说用于一般化的 n 阶乘(或 $n!$)转码方案的示例性发射机和接收机设备的框图。
- [0038] 图19是解说当 $w=8$ 时对于 $4!m=2$ 的总线化转码(仅RX侧)的示例的框图。
- [0039] 图20解说对于3导线系统以每群的各种码元的利用率表。
- [0040] 图21解说对于4导线系统以每群的各种码元的利用率表。
- [0041] 图22解说对于5导线系统以每群的各种码元的利用率表。
- [0042] 图23解说对于6导线系统以每群的各种码元的利用率表。
- [0043] 图24解说对于7导线系统以每群的各种码元的利用率表。
- [0044] 图25解说用于执行多导线信令编码的方法,其中时钟信号被编码在码元转变内。
- [0045] 图26解说用于执行多导线信令解码的方法,其中从码元转变中提取时钟信号。
- [0046] 图27解说了具有基于状态转变(原始码元转变)的集成时钟的3导线差分信令方

案。

[0047] 图28解说了N阶乘差分信令的表2802。

具体实施方式

[0048] 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。

[0049] 总览

[0050] 用于在多条导线上使用差分信令来传递转码系统的各种转码方法。

[0051] 第一方面提供了多导线信令编码,其中时钟信号被编码在码元转变内。数据比特序列被转换成多个(m个)转变数量。每一转变数量被转换成来自顺序码元编号集合的顺序码元编号。顺序码元编号被转换成原始码元。原始码元经由多个差分驱动器且散布在多条(n条)导线上传送,其中时钟信号被有效地嵌入在原始码元的传输中,因为从转变数量至顺序码元编号的转换保证没有两个连贯原始码元是相同的。

[0052] 第二方面提供了多导线信令解码,其中从码元转变中提取时钟信号。经由多个差分接收机接收散布在多条(n条)导线上的原始码元。原始码元被转换成来自顺序码元编号集合的顺序码元编号。每一顺序码元编号被转换成转变数量。多个转变数量被转换成数据比特序列。从原始码元转变中提取时钟信号。

[0053] 具有专用时钟信道和/或锁相环硬件的示例性转码系统

[0054] 图1解说了在发射机设备102和接收机设备104之间使用的四阶乘调制数据编码/解码方案。该阶乘数据编码/解码方案可包括编码器110、发射机电路112、物理导线107(例如,四个导体、线路或电路径)、四面体端接网络106、以及接收机电路114和解码器116。发射机电路112可包括使数据流串行化的串行化器(SER)118以及多个驱动器120。接收机电路114可包括多个接收机122以及使数据流反串行化的反串行化器(DES)124。应当清楚的是,四面体端接网络106实际上存在于发射机102和接收机104中的每一者的接口处,并且多条导线(即,该示例中的导线A、B、C和D)耦合在发射机102和接收机104的端接网络之间。在该示例中,六个驱动器120可用于生成在四个导体上传送的四个差分信号。然而,可以在其他实现中使用其他数目的驱动器和差分信号。

[0055] 单独通道可用作时钟信道108。该办法的缺点是它导致数据偏斜并需要两条额外线路来实现。

[0056] 图2解说了在发射机设备202和接收机设备204之间使用的另一四阶乘调制数据编码/解码方案206。该阶乘该数据编码/解码方案可包括编码器210、发射机电路212、物理导线(例如,四个导体或线路)、四面体端接网络206、以及接收机电路214和解码器216。在该示例中,编码器/解码器系统中的一个驱动器-接收机对208a/208b现在被用来传送/接收时钟信号。然而,并非使用六个驱动器来进行数据传输,而是只使用五个驱动器220来进行数据传输,同时将一个驱动器208a用于时钟信号。作为将驱动器-接收机对208a和208b专用于时钟信号的结果,(用于数据传输的)每码元传送的状态数量减半。例如,在4阶乘信令的情况下,在将所有比特都用于数据的情况下,每码元的二十四个(24)状态通常可以被传送。在一

个驱动器-接收机对被用于时钟信号的情况下,每码元只有十二个(12)状态被传送。

[0057] 图3解说了在发射机设备302和接收机设备304之间使用的另一四阶乘调制数据编码/解码方案306。在该示例中,串行化器318可由时钟信号326来同步。锁相环308可在接收机304处被用于提取时钟信号328。例如,可实现不归零解码以从数据信号中提取时钟信号。但该办法需要锁相环308,该锁相环消耗功率并占据半导体设备上的空间。

[0058] 具有基于码元转变的集成时钟的示例性多导线系统。

[0059] 图4解说了经由多导线传输路径在发射机设备402和接收机设备404之间使用的另一四阶乘调制数据编码/解码方案。在发射机设备402处,编码器408、串行化器(SER)模块410、多个差分驱动器412和端接网络406可被适配成实现四阶乘调制数据编码方案。在接收机404处,端接网络406、接收机或阱414、反串行化器(DES)模块410以及解码器412可被适配成实现四阶乘调制数据解码方案。应当清楚的是,端接网络406实际上存在于发射机设备402和接收机设备404中的每一者的接口处,并且多条导线(即,该示例中的导线A、B、C和D)耦合在发射机设备402和接收机设备404的端接网络之间。

[0060] 与图1和2、或图3中所解说的办法形成对比,图4中的办法不使用单独的时钟信道、单独的发射机/接收机,也不使用需要PLL来提取时钟信息的不归零解码。取而代之,时钟信号426可被嵌入在携带数据的码元转变内。时钟数据恢复电路418解码这些码元转变以提取时钟428。

[0061] 在该办法中,多条导线(例如,导体、线路等)差分地携带经编码信号,这些经编码信号携带多个码元。如果 n 条导线(例如,导体或线路)被用来传送差分信号,则可使用这 n 条导线来表示最多 $n!-1$ 个码元。构想了术语“导线”和/或“多条导线”被宽泛地解释并包括可用于在两个点之间携带信号的所有形式的物理导体、路径、线路等。

[0062] 在该转码系统内,时钟信号是从码元转变中提取的。在该转码方案中构想了三个转换:(a)原始码元与顺序码元编号之间的转换、(b)顺序码元编号与码元转变数量之间的转换、以及(c)码元转变数量与比特之间的转换。该转码方案被设计成使得其保证连贯原始码元被保证是不同的(即,在每一循环转变)。因此,原始码元转变可被接收机404用来生成、构造或提取时钟信号。

[0063] 图5是其中时钟被集成到码元转变中的发射机(编码器)设备402的框图。发射机设备402可包括比特至转变码元转换器502、转变码元至顺序码元转换器504、顺序码元至原始码元转换器506、以及在多条(n 条)导线上传送原始码元的多个驱动器508。

[0064] 图6是具有集成时钟恢复的接收机(解码器)设备404的框图。接收机设备404可包括在多条(n 条)导线上接收原始码元的多个接收机608、原始码元至顺序码元转换器606、顺序码元至转变码元转换器604、以及码元转变至比特转换器602。

[0065] 比特与码元转变数量之间的示例性转换

[0066] 图7解说了发射机702处从比特到码元转变数量以及随后在接收机704处从码元转变数量到比特的转换。发射机702将二进制信息(即,比特)馈送到“比特至 $m \times T$ ”转换器706以生成 m 个码元转变数量 T_0 到 T_{m-1} 。接收机704接收 m 个码元转变数量 T_0 到 T_{m-1} ,这些码元转变数量被馈送到“ $m \times T$ 至比特”转换器708以检索二进制信息(即,比特)。如果每一个 T (T_0 到 T_{m-1})存在 r 个可能码元转变状态,则 m 个转变能发送 r^m 个不同状态。对于 $N!$ 差分信令, $r = n! - 1$ 。因此,转变 $T_0 \cdots T_{m-1}$ 包含可具有 $(n! - 1)^m$ 个不同状态的数据。

[0067] 在一个示例中,可以假设每一个T的可能码元转变 r 为10。还假设一群中的码元数量 m 为3,以使得码元转变数量为 T_2 、 T_1 、 T_0 ,其中 $T_i:0,1,2,\dots,9$ 。由此,每一个T可具有10个不同状态。由此,对于 T_2 、 T_1 、 T_0 ,码元转变数量可以是例如3位数,诸如 $T_2=3$ 、 $T_1=9$ 、 $T_0=1$ (或者十进制数391)。以此方式,比特序列可被转换成多个码元转变数量 T ,以及反之。

[0068] 在另一个示例中, $3!(n=3)$,可以假设每一个T的可能码元转变 r 为 $5(=3!-1)$ 。如果一群中的码元数量 m 为4,则四位五进制数(基数为5的数字): T_3 、 T_2 、 T_1 、 T_0 ,其中每一个 $T_i:0,1,2,3,4$ 。例如,对于 $T_3=1$ 、 $T_2=2$ 、 $T_1=0$ 、 $T_0=3$,五进制数是 $1203_5=1\times 5^3+2\times 5^2+0\times 5^1+3\times 5^0=178$ 。以此方式,四个转变数量可被转换成一个编号。注意,转变数量 1203_5 可用作转变数量,例如在图8中,以使得每一个整数可以映射到顺序码元,反之亦然。

[0069] 图15是解说对于 n 导线系统的比特至码元转变数量的转换的框图。从比特到码元转变数量 T 的这一转换可由可以是发射机1502的一部分的编码器1504内的比特至码元转变数量转换器1506来执行。发射机1502和编码器1504可以与图5中所解说的相类似地操作。

[0070] 图16是解说对于3导线系统(对于 $n=3$ 且每一群有7个码元)的比特至码元转变数量的转换的示例的示例性编码器的框图。发射机设备1602可包括实现比特至码元转变数量转换器1606、码元转变数量至顺序码元编号转换器1608和顺序码元编号至原始码元转换器1610的编码器1604以及在多条(3条)导线上传送原始码元的三个驱动器1612。发射机1602和编码器1604可以与图5中所解说的相类似地操作。

[0071] 图17是解说对于3导线系统(对于 $n=3$ 且每一群有7个码元)的码元转变数量至比特的转换的示例的示例性解码器的框图。接收机设备1702可包括在多条(n 条)导线上接收原始码元的多个接收机1703、原始码元至顺序码元编号转换器1706、顺序码元编号至码元转变数量转换器1708、以及码元转变数量至原始比特转换器1710。接收机1702和解码器1704可以与图6中所解说的相类似地操作。

[0072] 顺序码元与码元转变数量之间的示例性转换

[0073] 图8解说了顺序码元与码元转变数量之间的转换。该转换将从前一顺序码元编号(P_s)到当前顺序码元编号(C_s)的每一转变映射到转变数量(T)。在发射机设备处,码元转变数量被转换成顺序码元编号。由于使用相对转换方案,因此转变数量保证了没有两个连贯的顺序码元编号804将会是相同的。

[0074] 在针对3导线系统的一个示例中,存在被指派给六个顺序码元编号 S_0 、 S_1 、 S_2 、 S_3 、 S_4 和 S_5 的六个原始码元。对于这六个顺序码元编号804,表802解说了如何可基于前一顺序码元编号(P_s)和当前顺序码元编号(C_s)来指派转变数量(T)。

[0075] 在该示例中,转变数量 T 可根据以下规则来指派:

[0076] $T = P_s + 1 \leq C_s$

[0077] $?C_s - (P_s + 1)$

[0078] $:C_s - (P_s + 1) + 6$

[0079] 相反,当前顺序码元编号(C_s)可根据以下规则来指派:

[0080] $C_s = P_s + 1 + T < 6$

[0081] $?P_s + 1 + T$

[0082] $:P_s + 1 + T - 6$

[0083] 相似的办法可用于 n 导线系统,这通过对于任何 n (例如, $n=4,5,6,\dots$)构造保证原

始码元将改变的顺序码元编号至转变数量的映射表来实现。

[0084] 例如,在n导线系统的情况下,转变数量T可根据以下规则来指派:

[0085] $T = Ps + 1 \leq Cs$

[0086] $?Cs - (Ps + 1)$

[0087] $:Cs - (Ps + 1) + n!$

[0088] 相反,在n导线系统的情况下,当前顺序码元编号(Cs)可根据以下规则来指派:

[0089] $Cs = Ps + 1 + T < n!$

[0090] $?Ps + 1 + T$

[0091] $:Ps + 1 + T - n!$

[0092] 顺序码元编号与原始码元之间的示例性转换

[0093] 图9是解说原始值(对于n=3导线系统)与顺序码元编号之间的转换的表。在该示例中,对于n=3,原始数据值X、Y和Z的组合可以被映射到顺序码元编号0-5的集合中的一个顺序码元编号。原始值X、Y和Z的组合(例如,{Z,Y,Z})表示原始码元。助记符-x、+x、+y、-y、+z和-z被用来表示3相系统(其中n=3)的六个原始码元中的每一个。因为原始码元被保证在每一循环改变(即,没有两个连贯原始码元是相同的),所以从一个原始码元到下一原始码元的转变可被用于提取时钟信号。

[0094] 具有嵌入式定时信息的示例性3导线转码系统

[0095] 图27解说了发射机设备2700与接收机设备2701之间的3导线差分信令方案,其中时钟(定时信息)基于由导体、导线或线路A、B和C之间的差分信号所定义的状态而被嵌入。

[0096] 发射机设备2700和接收机设备2701可以在多线路总线2708上通信。在该示例中,三条线路A、B和C被用于总线2708。接收机设备2701可包括将该接收机设备2701耦合到总线2708的三端口接收机2710。

[0097] 在一个示例中,如图4-9中解说的差分信号编码/解码可用于将信号从发射机设备2700传送到接收机设备2701。作为接收机端接网络的一部分,多个接收机2712中的每一个可被配置成采用三条线路A、B和C中的两条并提供不同信号。例如,第一线路A和第二线路B可用于提供第一差分信号RX_AB 2714,第二线路B和第三线路C可用于提供第二差分信号RX_BC 2716,而第一线路A和第三线路C可用于提供第三差分信号RX_CA 2718RX。这些差分信号2714、2716和2718可用作至解码器(DEC)电路2720的输入。解码器电路2720解码这三个差分信号RX_AB 2714、RX_BC 2716和RX_CA 2718,并输出六个状态XM、YM、ZM、ZP、YP和XP(在图9中也被表示为-x、-y、-z、+z、+y和+x)。

[0098] 状态图2703解说了可由三个导体A、B和C 2708携带的差分信号2714、2716和2718定义的六个(6)状态XM、YM、ZM、ZP、YP和XP。如可以观察到的,跨这三个差分信号2714、2716和2718的电压电平可以被映射到一(1)和零(0)的不同组合。例如,关于状态XM的差分信号电压电平可以与“011”相关联,状态YM可以与“101”相关联,状态ZP可以与“001”相关联,状态ZM可以与“110”相关联,状态YP可以与“010”相关联,而状态XP可以与“100”相关联。

[0099] 除了在状态中编码的信息(例如,每一状态有3比特)之外,还可基于这些状态之间的转变来编码信息。注意,任何两个状态(XM、YM、ZM、ZP、YP和XP)之间的转变发生在单个步骤中,而不穿越中间状态。由此,基于状态图2703的差分数据传输方案将没有状态转变解码问题。由于该编码,没有两个连贯原始码元是相同的。原始码元的保证转变(即,每一循环中

状态XM、YM、ZM、ZP、YP和XP的改变)准许使用此类转变来在接收机设备2701处生成或提取时钟。

[0100] 总线2708的导体、导线或线路中的每一者都可被驱动为高、驱动为低或不被驱动,且在任何单个循环中只有一个导体未被驱动。在一个实施例中,三个差分信号RX_AB 2714、RX_BC 2716和RX_CA 2718(例如,由接收机设备2701内的解码器2720接收)分别在导体A相对于导体B、导体B相对于导体C以及导体C相对于导体A之间被定义为正差分电压是逻辑1而负差分电压是逻辑0。这三个差分信号2714、2716和2718的示例波形在示图2704中解说。

[0101] 六个可能状态(不包括导致导体A相对于导体B、导体B相对于导体C、以及导体C相对于导体A之间的零差分电压的状态)由信号RX_AB 2714、RX_BC 2716和RX_CA 2718的状态根据状态图2703来定义为XM、YM、ZP、ZM、YP、XP和XM。

[0102] 对应于这六个可能状态XM、YM、ZP、ZM、YP、XP和XM的状态信号由接收机设备2701中的解码器块2720(DEC)从差分信号RX_AB 2714、RX_BC 2716和RX_CA 2718生成,并且这些状态信号的示例性波形在示图2705中示出。

[0103] 在一个实施例中,在每一个循环总是发生从状态XM、YM、ZP、ZM、YP、XP、或XM到不同状态的状态转变。因此,状态转变可被用于表示从发射机设备2700传送到接收机设备2701的时钟信号。接收机设备2701然后可以从保证状态转变(即,连贯原始码元之间的保证转变)中提取时钟信号。

[0104] 示例性端接网络

[0105] 图10解说示出至驱动器的输入和来自接收机的输出以及各节点之间的电流的3导线端接网络1002。对于 $n=3$ 的该转码方案具有两种模式 $(ZYX) = (000)$ 和 (111) 导致零差分电压并由此是非法状态。零差分电压是非法的,因为它们无法彼此区分开,由此妨碍准确解码。非法状态必须不被用于编码。其余六个(合法)状态是 $(ZYX) = (001)$ 、 (010) 、 (100) 、 (110) 、 (101) 和 (011) 。

[0106] 图11(包括图11A和11B)解说示出至驱动器的输入和来自接收机的输出以及各节点之间的电流1104的4导线端接网络1102。该4导线端接网络1102包括四个端接电阻器、六个差分驱动器以及六个差分接收机(驱动器/阱),并使用三个差分电压电平。表1106解说了跨每一端接电阻R的可能的单位电流。如从端接网络1102可观察到的,每一端接电阻R都耦合到三个(3)驱动器/阱1107。假定来自每一驱动器/阱的单位电流为+1或-1,则表1106解说了对于每一端接电阻R的可能净电流:+3、+1、-1、-3。

[0107] 图12解说示出发射机驱动器、接收机驱动器和4导线端接网络的发射机至接收机转码方案。

[0108] 图11和12的4导线端接网络1102具有64个可能状态(即,六个驱动器和 2^6 个状态)。然而,如以下参照图13进一步描述的,在这64个可能状态中,40个是非法状态,因为它们导致零差分电压(如图13所解说的)。对于每一端接电阻R,对于一个驱动器存在四个可能电流(幅度和方向):-3、-1、+1和+3。差分电压XY在两个电阻器(X0和Y0)具有相同大小和相同方向的电流时变为零(0)。由此,仅仅剩余的24个“非零”状态可被用作原始码元。

[0109] 图13(包括图13A和13B)是对于 $n=4$ 示出原始码元与具有零差分电压的那些码元的各种组合的表。该表可关于图11和12的4导线端接网络来解释。Sraw是指在图12中的CD、DB、AD、AC、CB和AB处的差分信号形成的原始码元。每一Rterm处的电压是指跨图11中的端接

网络1102中的中心节点0与每一节点A之间的每一电阻器R的电压。在该示例中,假定单位电阻R,跨每一电阻Rterm的电压被给出为+3、+1、-1、或-3,如图11中的表1106中的可能电流中那样。作为这些跨每一Rterm的电压的结果,可以在图11中的端接网络1102中查明跨每一驱动器/阱的差分电压(DB、CD、AD、AC、CD和AB)。在某些情况下,对于给定差分发射机/接收机的跨两个电阻器对Rterm的电压相互抵消,从而导致“0”差分电压。结果,此类状况是无法区分的,发生此类状况的原始码元Sraw是无法解码/无法区分的并被认为是“非法”的且不被使用。

[0110] 图14解说具有已从图13A和13B获得的非零差分电压原始码元(Sraw)的表。因为只有可能组合的子集可以用作有效原始码元,所以可使用映射来保证非法状态从不被用作原始码元。

[0111] 对于任何给定数目n条导线(例如,导体、线路等),生成类似的表并用于定义原始码元(来自有效“非零”状态)。原始码元到顺序码元的映射可以如图9中所解说的那样进行。注意,该映射保证只传送有效码元(“非零差分电压状态”)。由于在传送原始码元之前在发射机处发生的码元转变数量至顺序码元转换,没有两个连贯原始码元将会是相同的,它们始终变化。即使恰好相同的比特群一连串地(例如,全1或全0比特)被传送两次或更多次,原始比特也将会是不同的。原始码元的这种改变保证在每一次原始码元被传送时的“转变”。因此,能够从原始码元的此类保证转变中提取时钟。

[0112] 示例性N!转码系统

[0113] 图18是解说用于一般化的n阶乘(n!)转码方案的示例性发射机和接收机设备的框图,其中n是所使用的导线(例如,导体、线路)的数目,m是码元/群,且w是反串行化器(DES)1806和串行化器(SER)1826块的位宽。在该示例中,接收机设备1802可包括包含DES块1806的解码器1804。解码器1804可以与图6中所解说和描述的相类似地操作。另外,发射机设备1822可包括包含SER块1826的编码器1824。编码器1824可以与图5中所解说和描述的相类似地操作。

[0114] 在一个示例中,如果 $n=4$ 、 $m=2$ 且 $w=8$ 以使得2码元能够传送9比特,则8个码元通过TX和RX侧的转码被同时处理。

[0115] 图19是解说对于 $n=4$ 、 $m=2$ 且 $w=8$ 的总线化转码(仅RX侧)的示例的框图。该示例解说了图18的接收机1802和解码器1804的详细视图。如图所示,六个接收机1912中的每一个可使用六个反串行化器1914,且使用单独的解码器电路来解码比特序列中的不同比特。

[0116] 具有集成时钟的示例性数据编码设备和方法

[0117] 提供了一种用于执行多导线信令编码的编码电路。此类编码电路和其中执行的功能的各种示例在图4(发射机402)、图5、图7、图8、图9、图13、图14、图15(编码器1504)、图16(编码器1604)和图18(编码器1824)中找到。比特至转变数量转换器用于将数据比特序列转换成多个(m个)转变数量。转变数量至顺序编号转换器用于将每一转变数量转换成来自顺序编号集合的顺序编号。将每一转变数量转换成来自顺序编号集合的顺序编号可包括将多个转变数量转换成顺序编号。顺序编号至原始码元转换器用于将顺序编号转换成原始码元。

[0118] 将顺序编号转换成原始码元可包括将顺序编号转换成多个原始码元。原始码元可被保证具有跨n条导线中的所有导线对的非零差分电压。例如,如图27所示,跨导线AB、BC和

CA的差分电压2703是非零的。

[0119] 多个差分驱动器可用于散布在多条(n条)导线上传送原始码元,其中时钟信号被有效地嵌入在原始码元的传输中,因为从转变数量至顺序编号的转换保证没有两个连贯原始码元是相同的。在各种示例中,这多条(n条)导线大于或等于3或4。

[0120] 对于跨这n条导线中的各导线对的n阶乘差分信号, r^m 个可能的不同状态由转变数量来表示,其中r是 $n!-1$ 。顺序编号可以从基于自紧挨着的前一顺序码元编号的转变的转变数量来选择。例如,对顺序码元编号的此类选择在图8中解说。

[0121] 在一个示例中,对于n阶乘差分信号,多个差分驱动器等于 nC_2 ,其中 $nC_2 = n! / (2! (n-2)!) = n(n-1)/2$ 。图28解说了N阶乘差分信令的表2802。例如,如果n=6条导线,则可使用每循环的720个状态、5个差分电压电平和15个驱动器/接收机来传送每循环的9.49个比特。因此,相对于传统差分信令办法,可以每循环在相同数量的导线上传送显著更多的信息。

[0122] 在另一示例中,这多个差分驱动器等于n。例如,所使用的端接网络可被设计成使得对于n=3、4、5、6等的所有值都使用与导线相同数量的差分驱动器。

[0123] 图25解说用于执行多导线信令编码的方法,其中时钟信号被编码在码元转变内。数据比特序列被转换成多个(m个)转变数量2502。每一转变数量被转换成来自顺序码元编号集合的顺序码元编号(2504)。顺序码元编号被转换成原始码元(2506)。原始码元散布在多个差分驱动器上且散布在多条(n条)导线(例如,导体或线路)上传送,其中时钟信号被有效地嵌入在原始码元的传输中,因为从转变数量至顺序码元的转换保证没有两个连贯原始码元是相同的(2508)。

[0124] 具有时钟恢复的示例性解码设备和方法

[0125] 提供了一种用于执行多导线信令解码的解码电路。

[0126] 此类解码电路和其中执行的功能的各种示例在图4(接收机402)、图6、图7、图8、图9、图13、图14、图17(解码器1704)和图18(解码器1804)中找到。多个差分接收机可用于接收散布在多条(n条)导线上的原始码元。在各种示例中,这多条(n条)导线大于或等于3或4。

[0127] 原始码元至顺序编号转换器可用于将原始码元转换成来自顺序编号集合的顺序编号。将原始码元转换成来自顺序编号集合的顺序编号可包括将多个原始码元转换成顺序编号。原始码元被保证具有跨这多条(n条)导线中的所有导线对的非零差分电压。

[0128] 顺序编号至转变数量转换器可用于将每一顺序编号转换成转变数量。将每一顺序编号转换成转变数量可包括将顺序编号转换成多个转变数量。转变数量可以从该顺序编号并基于紧挨着的前一顺序码元编号来选择。

[0129] 转变数量至比特转换器可用于将多个转变数量转换成数据比特序列。时钟数据恢复电路可以从原始码元接收中提取时钟信号。原始码元接收使用时钟信息被同步。

[0130] 对于跨n个驱动器的n阶乘差分信令, r^m 个可能的不同状态可由转变数量来表示,其中r是 $n!-1$ 。

[0131] 图26解说用于执行多导线信令解码的方法,其中从码元转变中提取时钟信号。经由多个差分接收机接收散布在多条(n条)导线上的原始码元(2602)。原始码元被转换成来自顺序编号集合的顺序编号(2604)。每一顺序码元编号被转换成转变数量(2606)。多个转变数量被转换成数据比特序列(2608)。从原始码元接收中提取时钟信号(2610)。

[0132] 示例性利用率

[0133] 取决于所使用的导线数量以及所选择的码元/群,可达成不同的利用率百分比。“利用率”可以指可发送的每群的比特数量的效率。在这些示例中,“利用率”可被表示为对于给定数目的导体和每群的码元,所传送的每群的整数比特数目与可以传送的每群的理论比特数目的百分比。

[0134] 图20解说对于3导线系统以每群的各种码元的利用率表。在该示例中,通过28码元/群可发送2.321比特/循环,而1码元/群只能发送2比特/循环。

[0135] 图21解说对于4导线系统以每群的各种码元的利用率表。在该示例中,通过2码元/群可发送4.5比特/循环,而1码元/群只能发送4比特/循环。

[0136] 图22解说对于5导线系统以每群的各种码元的利用率表。在该示例中,通过19码元/群可发送6.8947比特/循环,而1码元/群只能发送6比特/循环。值得注意的是,14码元/群能发送96比特(32×3)。

[0137] 图23解说对于6导线系统以每群的各种码元的利用率表。在该示例中,通过31码元/群可发送9.483比特/循环,而1码元/群只能发送9比特/循环。

[0138] 图24解说对于7导线系统以每群的各种码元的利用率表。在该示例中,通过27码元/群可发送12.296比特/循环,而1码元/群只能发送12比特/循环。

[0139] 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。

[0140] 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。

[0141] 还应注意,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。

[0142] 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,但不被限定于,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。

[0143] 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，一代码段可被耦合到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。

[0144] 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。

[0145] 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。存储介质可耦合到处理器以使得该处理器能从/向该存储介质读写信息。替换地，存储介质可以被整合到处理器。

[0146] 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、和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。

[0147] 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。

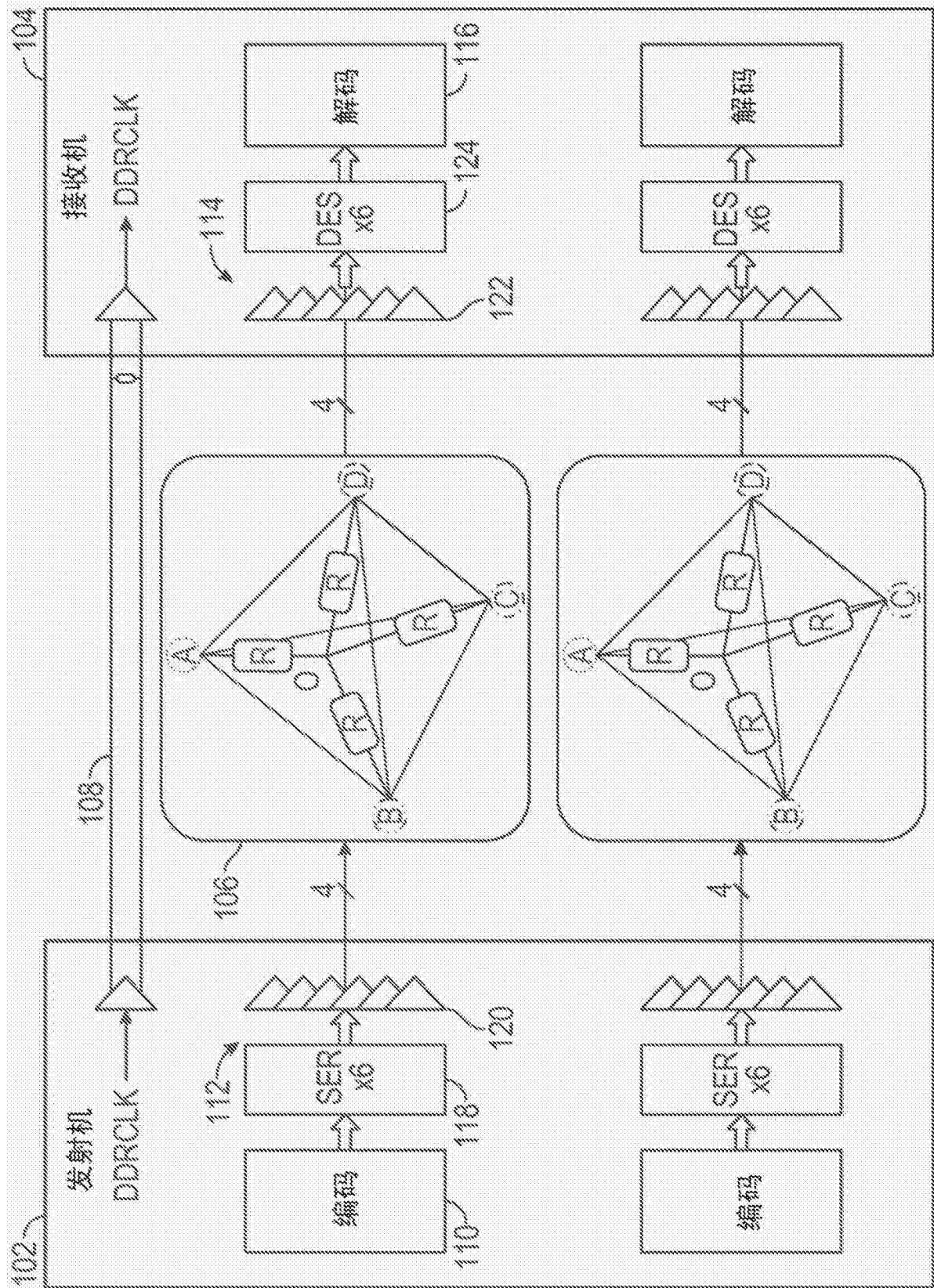


图1

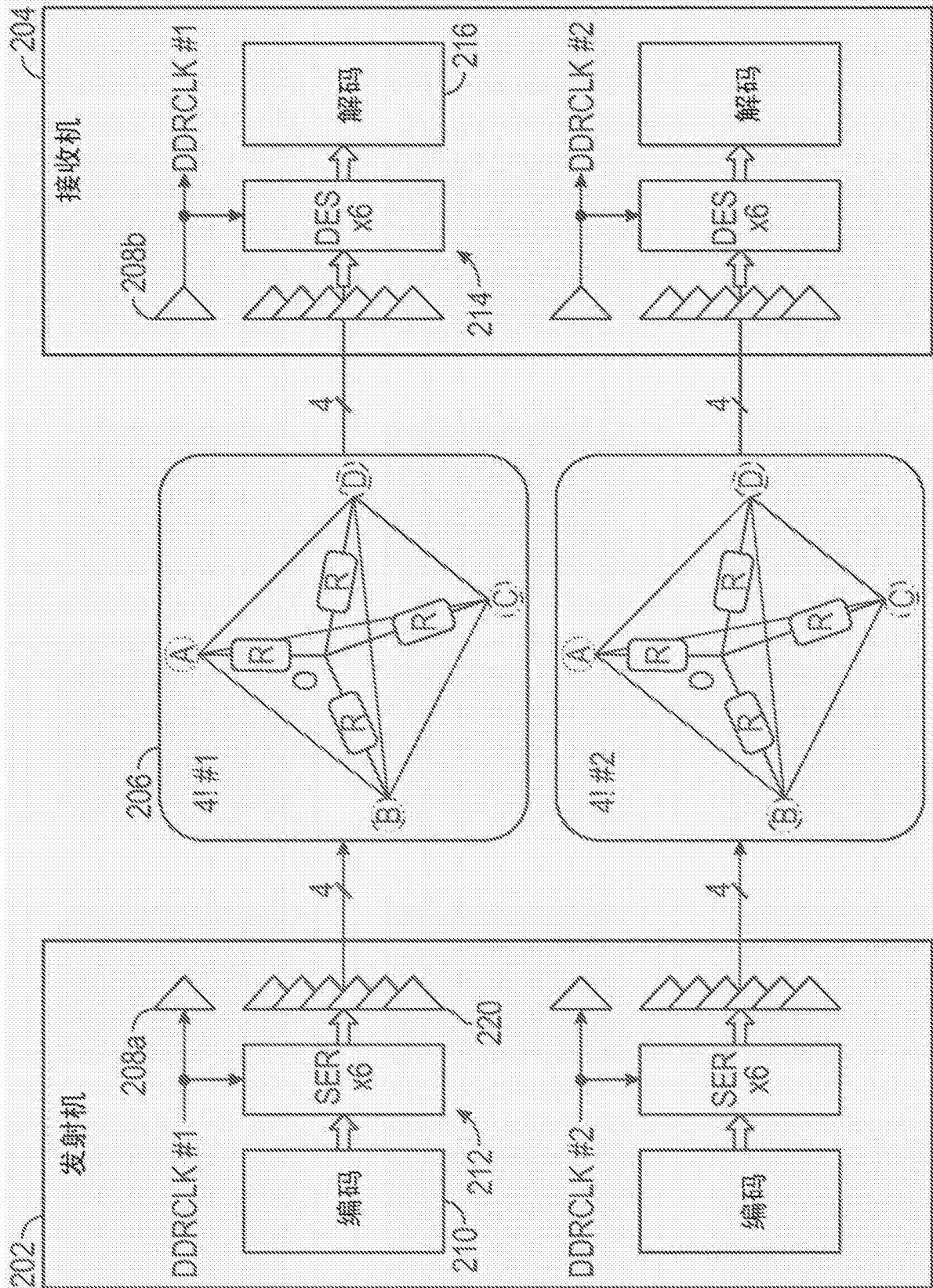


图2

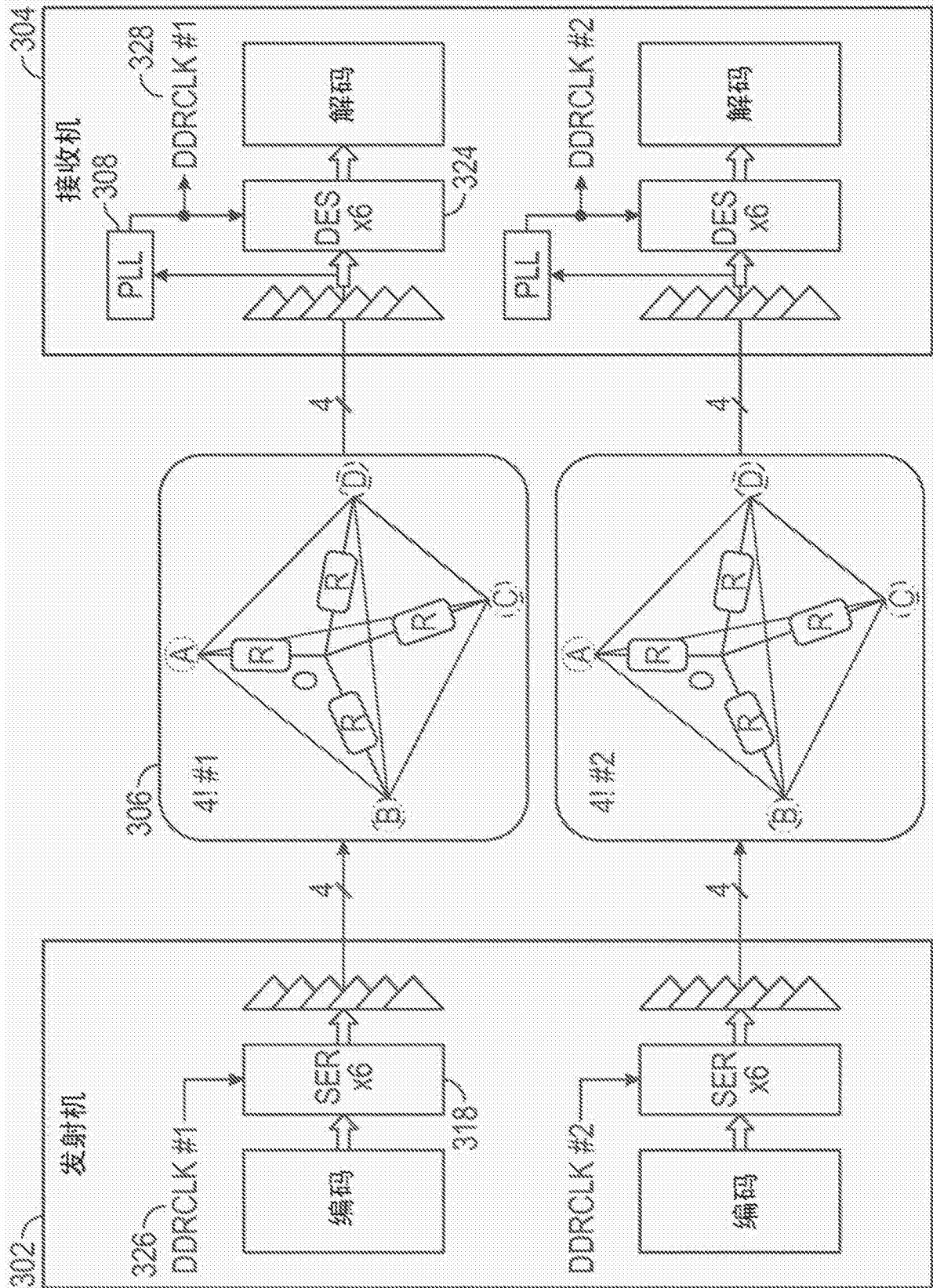


图3

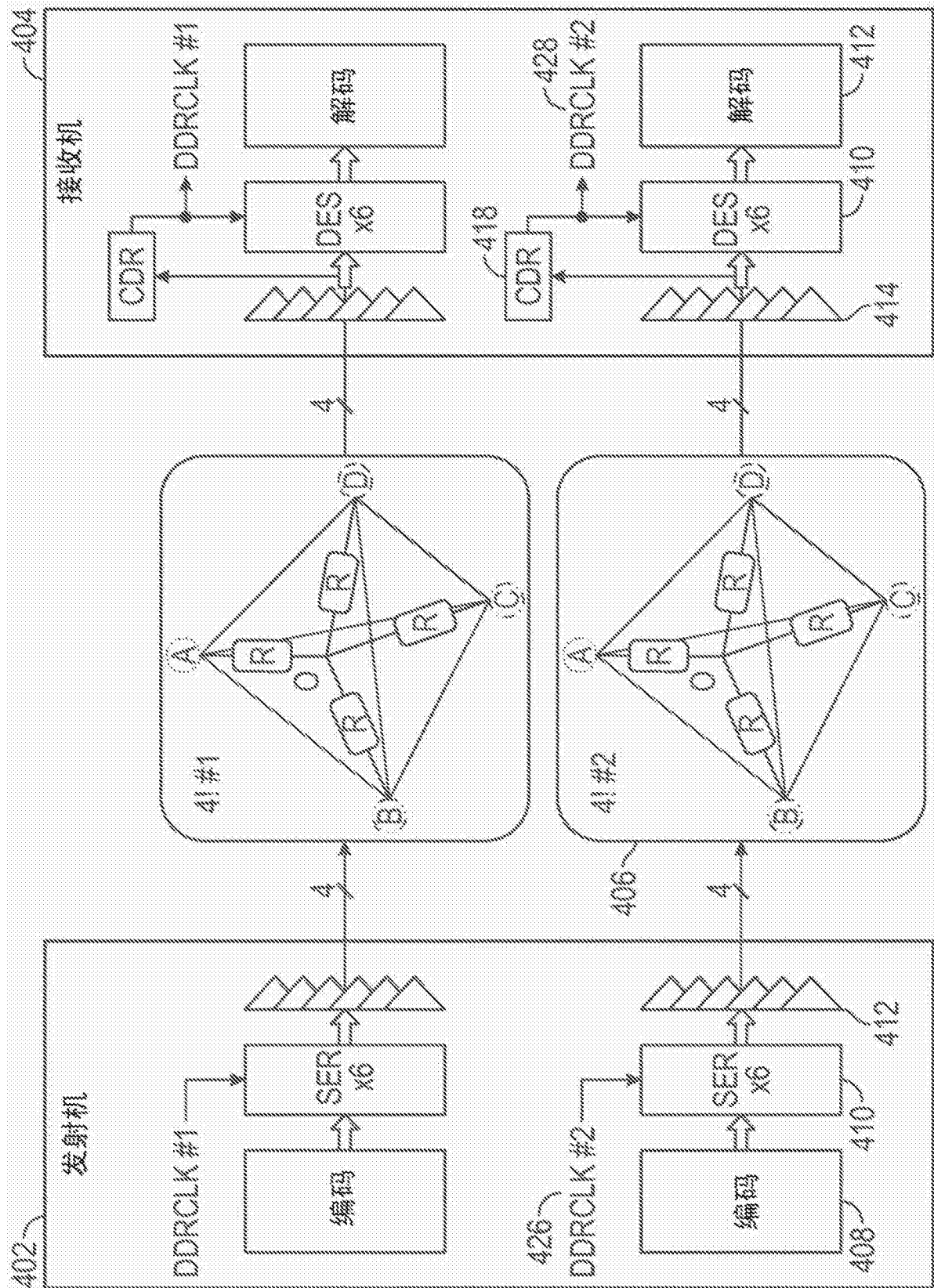


图4

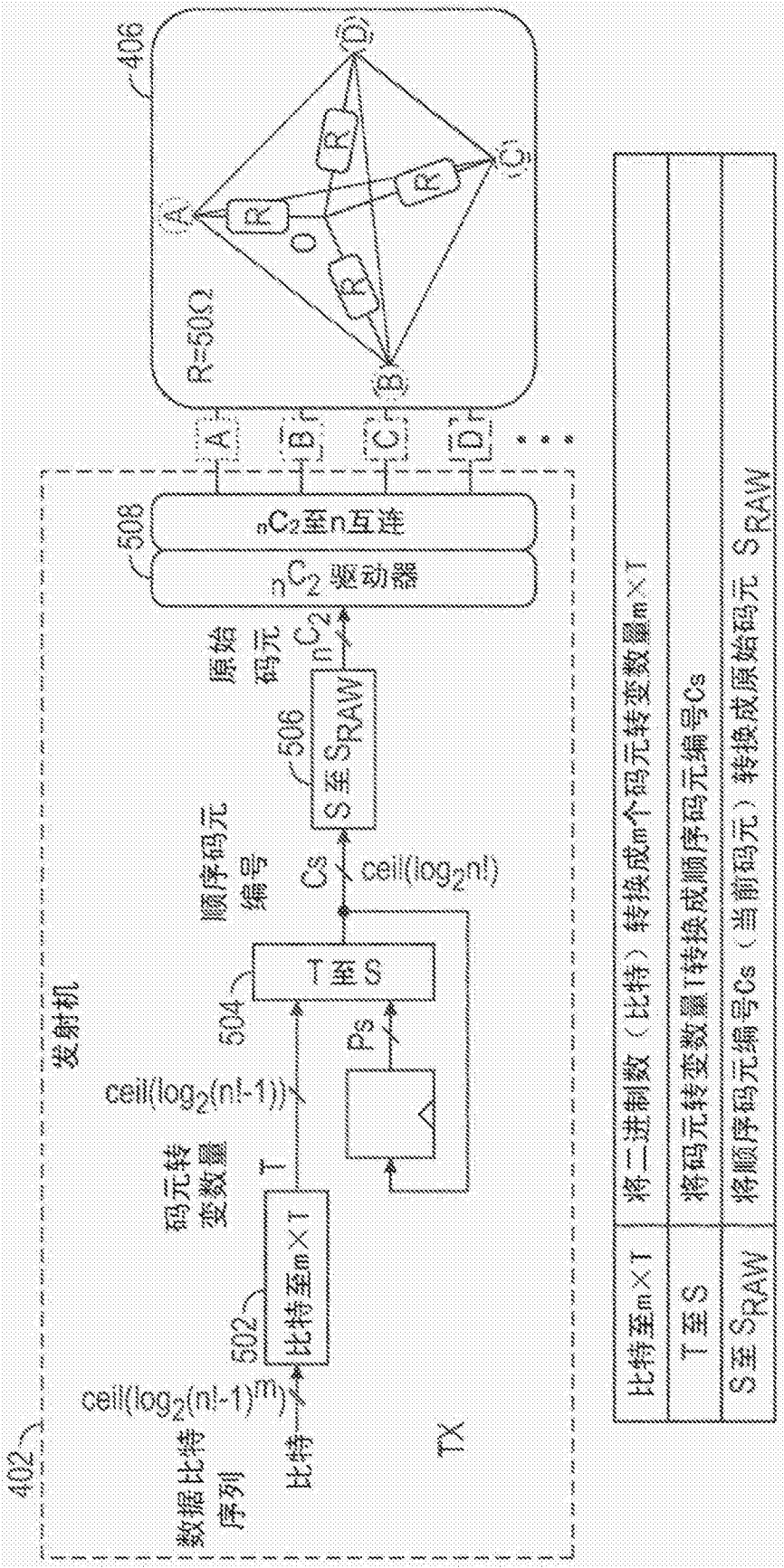


图5

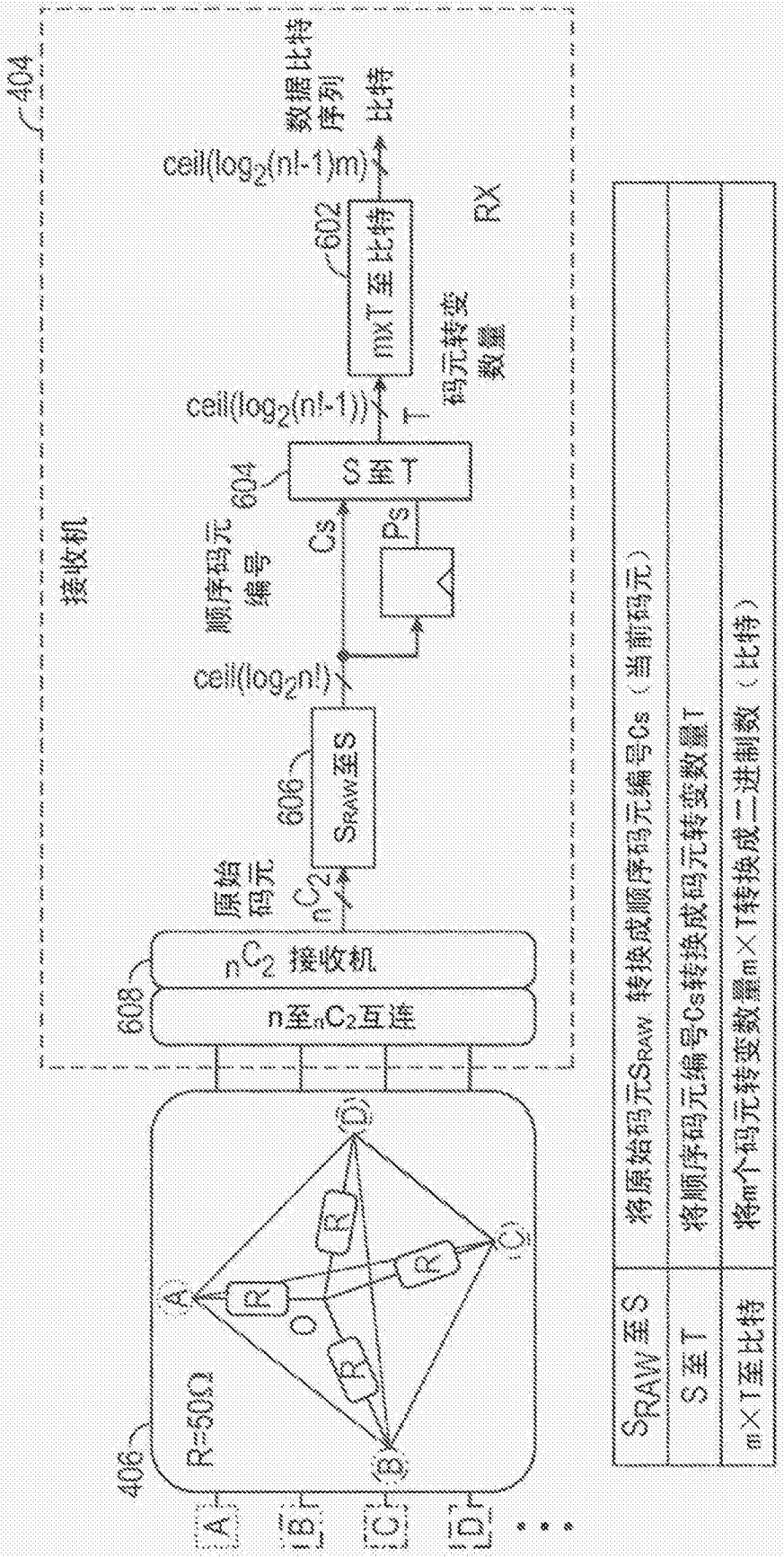


图6

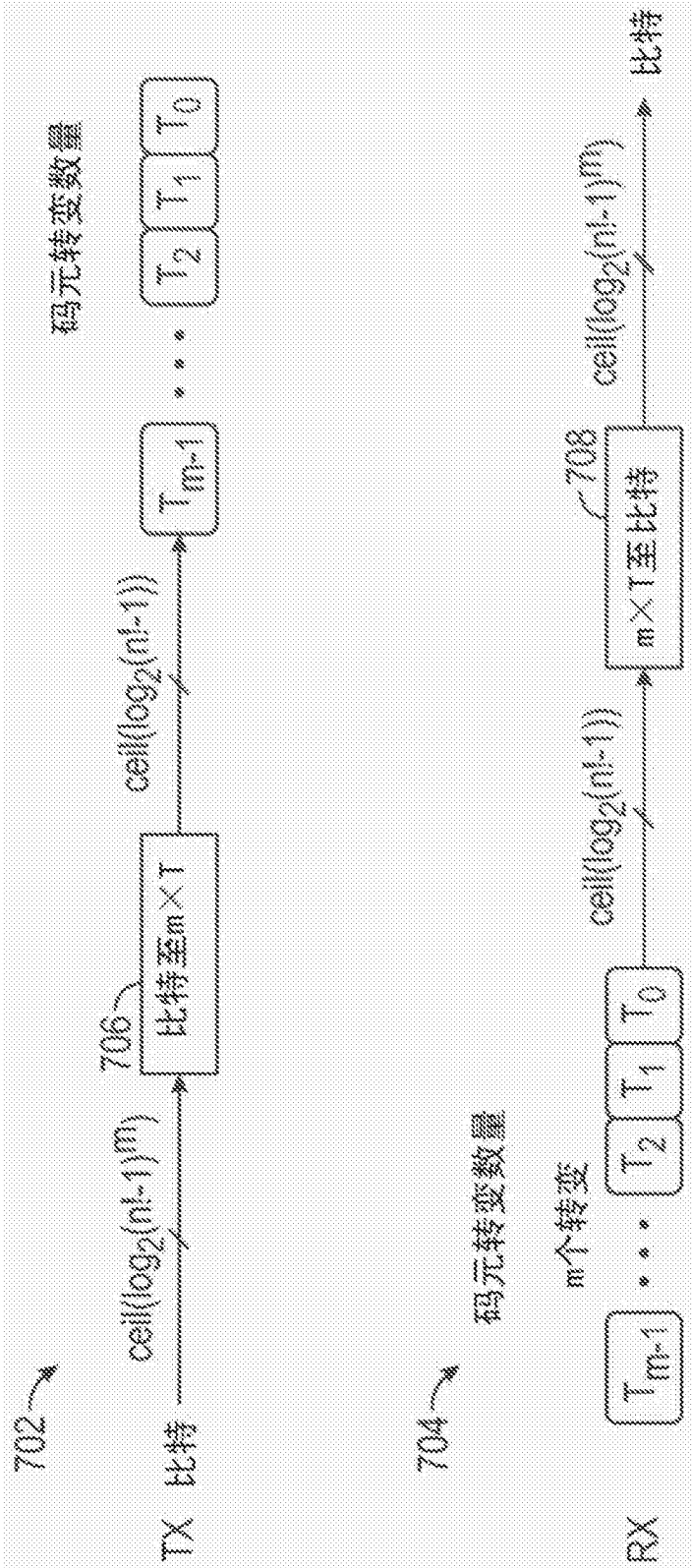


图7

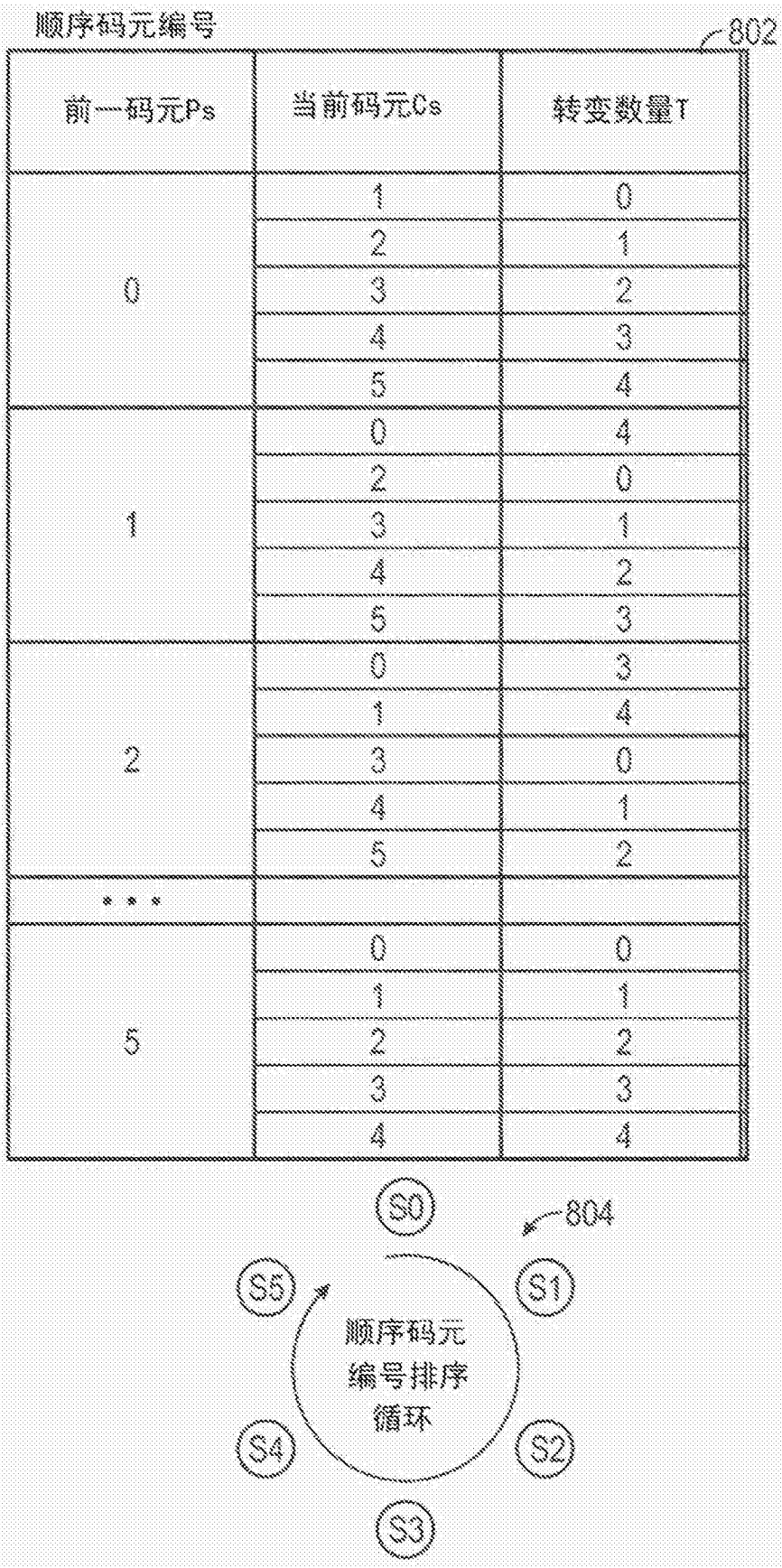


图8

对于n=3的码元编号转换如以下那样简单:

$$V[2:0] = \{Z, Y, X\} == 6 ? 0 : \{Z, Y, X\};$$

$$\{Z, Y, X\} = V[2:0] == 0 ? 6 : V[2:0];$$

原始码元			顺序码元 编号	“3相”码元	
Z	Y	X	V[2:0]		
1	1	0	0	-x	仅码元 $\{Z, Y, X\} := V[2:0]$
0	0	1	1	+x	
0	1	0	2	+y	
0	1	1	3	-z	
1	0	0	4	+z	
1	0	1	5	-y	

图9

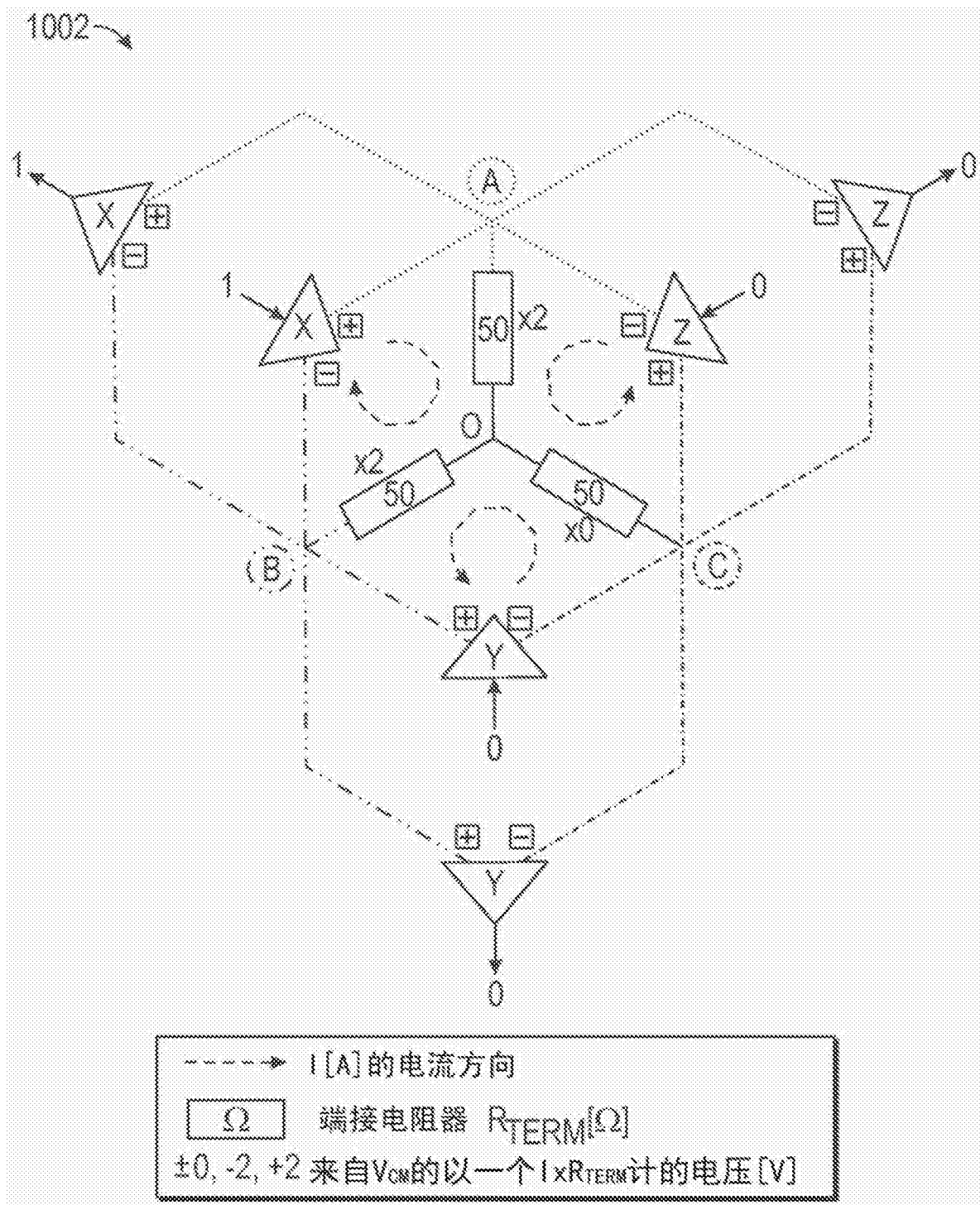
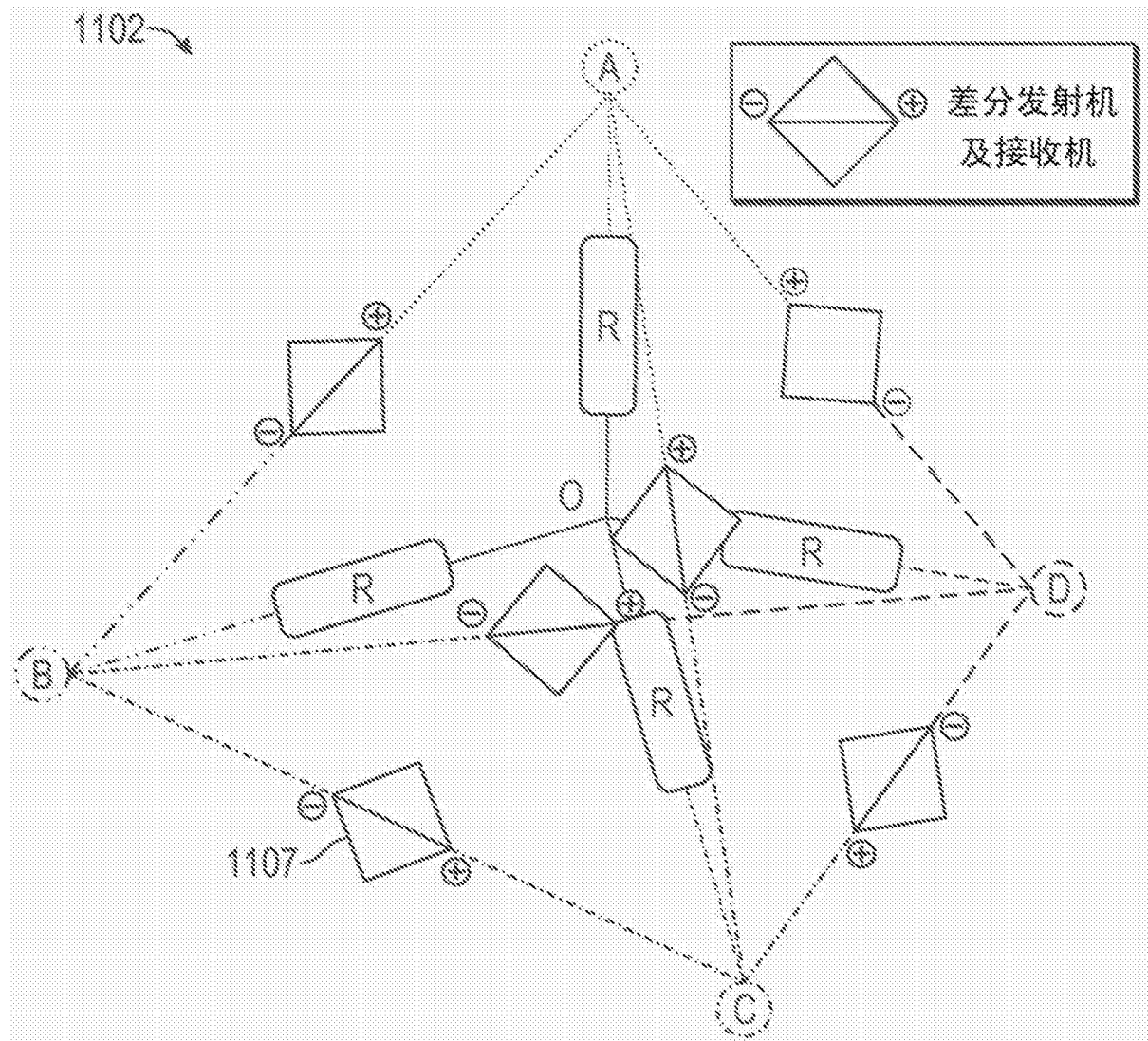
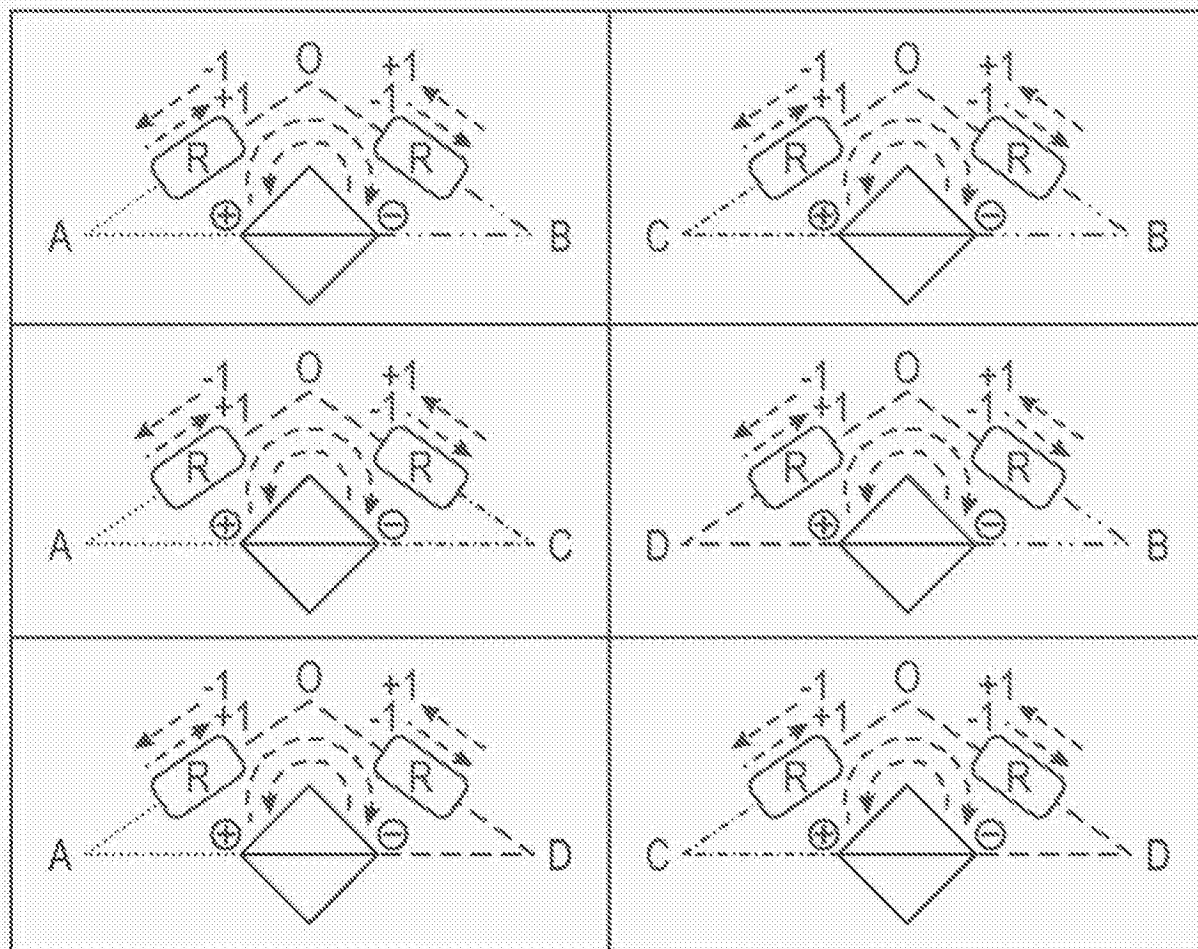


图10



1104



1106

来自连接的 驱动器的 电流计数			R处的 可能总电流	
+1	+1	+1	+3	全部驱动+1
+1	+1	-1	+1	一个驱动-1
+1	-1	-1	-1	一个驱动+1
-1	-1	-1	-3	全部驱动-1

图11B

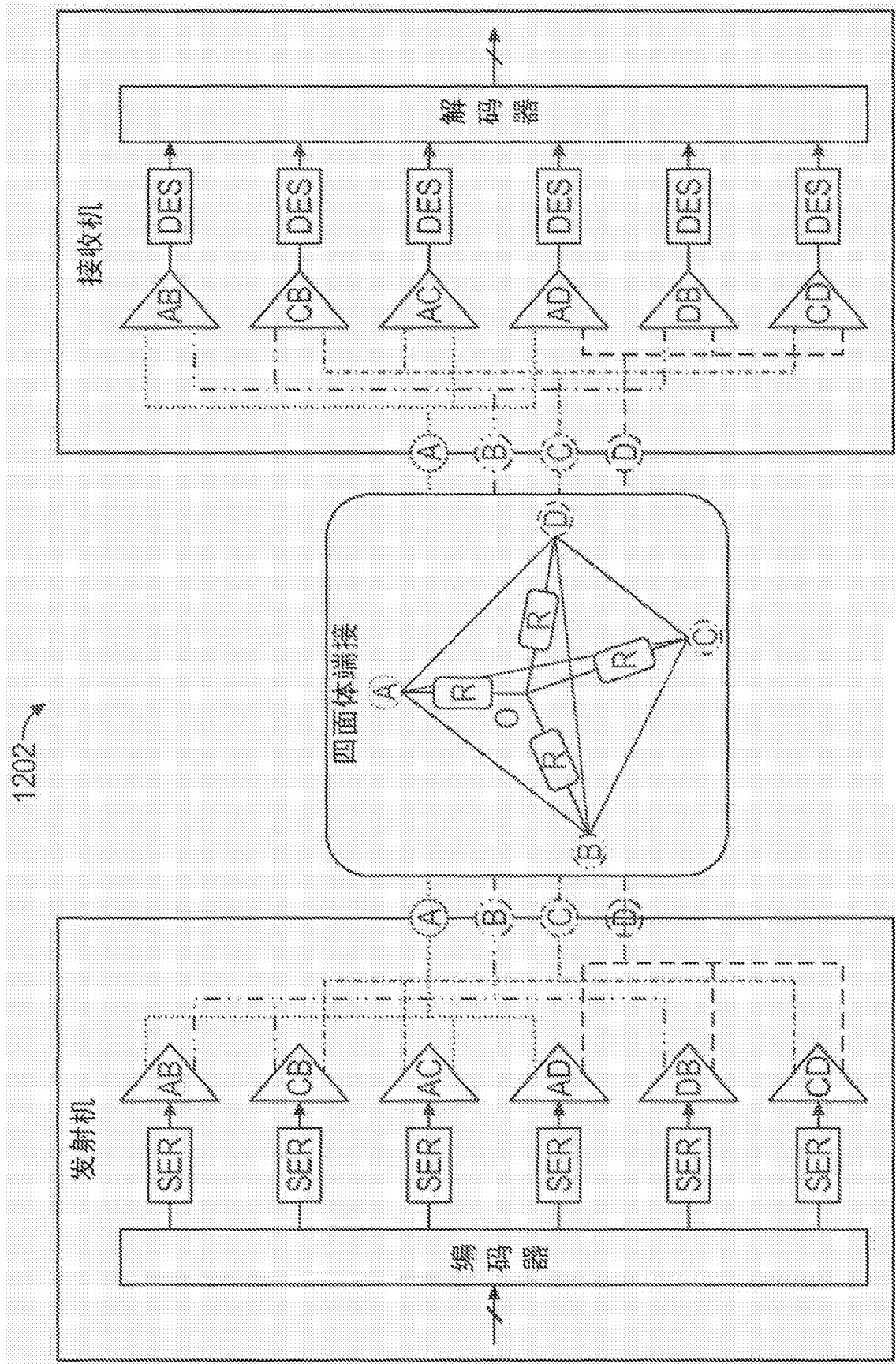


图12

四面体：每6比特输入的差分电压（1）

S _{RAW} (CD, DB, AD, AC, CB, AB)	每个R _{TERM} 处的电压				差分电压						0差分 电平?
	DO	CO	BO	AO	DB	CB	AD	AC	CD	AB	
000000	1	-1	3	-3	-2	-4	-4	-2	-2	-6	
000001	1	-1	1	-1	0	-2	-2	0	-2	-2	零
000010	-1	1	3	-3	-4	-2	-2	-4	2	-6	
000011	-1	1	1	-1	-2	0	0	-2	2	-2	零
000100	1	-3	3	-1	-2	-6	-2	2	-4	-4	
000101	1	-3	1	1	0	-4	0	4	-4	0	零
000110	-1	-1	3	-1	-4	-4	0	0	0	-4	零
000111	-1	-1	1	1	-2	-2	2	2	0	0	零
001000	-1	-1	3	-1	-4	-4	0	0	0	-4	零
001001	-1	-1	1	1	-2	-2	2	2	0	0	零
001010	-3	1	3	-1	-6	-2	2	-2	4	-4	
001011	-3	1	1	1	-4	0	4	0	4	0	零
001100	-1	-3	3	1	-4	-6	2	4	-2	-2	
001101	-1	-3	1	3	-2	-4	4	6	-2	2	
001110	-3	-1	3	1	-6	-4	4	2	2	-2	
001111	-3	-1	1	3	-4	-2	6	4	2	2	
010000	1	1	1	-3	0	0	-4	-4	0	-4	零
010001	1	1	-1	-1	2	2	-2	-2	0	0	零
010010	-1	3	1	-3	-2	2	-2	-6	4	-4	
010011	-1	3	-1	-1	0	4	0	-4	4	0	零
010100	1	-1	1	-1	0	-2	-2	0	-2	-2	零
010101	1	-1	-1	1	2	0	0	2	-2	2	零
010110	-1	1	1	-1	-2	0	0	-2	2	-2	零
010111	-1	1	-1	1	0	2	2	0	2	2	零
011000	-1	1	1	-1	-2	0	0	-2	2	-2	零
011001	-1	1	-1	1	0	2	2	0	2	2	零
011010	-3	3	1	-1	-4	2	2	-4	6	-2	
011011	-3	3	-1	1	-2	4	4	-2	6	2	
011100	-1	-1	1	1	-2	-2	2	2	0	0	零
011101	-1	-1	-1	3	0	0	4	4	0	4	零
011110	-3	1	1	1	-4	0	4	0	4	0	零
011111	-3	1	-1	3	-2	2	6	2	4	4	

图13A

四面体：每6比特输入的差分电压（2）

S _{RAW} (CD, DB, AD, AC, CB, AB)	每个R _{TERM} 处的电压				差分电压						0差分 电平?
100000	3	-1	1	-3	2	-2	-6	-2	-4	-4	
100001	3	-1	-1	-1	4	0	-4	0	-4	0	零
100010	1	1	1	-3	0	0	-4	-4	0	-4	零
100011	1	1	-1	-1	2	2	-2	-2	0	0	零
100100	3	-3	1	-1	2	-4	-4	2	-6	-2	
100101	3	-3	-1	1	4	-2	-2	4	-6	2	
100110	1	-1	1	-1	0	-2	-2	0	-2	-2	零
100111	1	-1	-1	1	2	0	0	2	-2	2	零
101000	1	-1	1	-1	0	-2	-2	0	-2	-2	零
101001	1	-1	-1	1	2	0	0	2	-2	2	零
101010	-1	1	1	-1	-2	0	0	-2	2	-2	零
101011	-1	1	-1	1	0	2	2	0	2	2	零
101100	1	-3	1	1	0	-4	0	4	-4	0	零
101101	1	-3	-1	3	2	-2	-2	6	-4	4	
101110	-1	-1	1	1	-2	-2	2	2	0	0	零
101111	-1	-1	-1	3	0	0	4	4	0	4	零
110000	3	1	-1	-3	4	2	-6	-4	-2	-2	
110001	3	1	-3	-1	6	4	-4	-2	-2	2	
110010	1	3	-1	-3	2	4	-4	-6	2	-2	
110011	1	3	-3	-1	4	6	-2	-4	2	2	
110100	3	-1	-1	-1	4	0	-4	0	-4	0	零
110101	3	-1	-3	1	6	2	-2	2	-4	4	
110110	1	1	-1	-1	2	2	-2	-2	0	0	零
110111	1	1	-3	1	4	4	0	0	0	4	零
111000	1	1	-1	-1	2	2	-2	-2	0	0	零
111001	1	1	-3	1	4	4	0	0	0	4	零
111010	-1	3	-1	-1	0	4	0	-4	4	0	零
111011	-1	3	-3	1	2	6	2	-2	4	4	
111100	1	-1	-1	1	2	0	0	2	-2	2	零
111101	1	-1	-3	3	4	2	2	4	-2	6	
111110	-1	1	-1	1	0	2	2	0	2	2	零
111111	-1	1	-3	3	2	4	4	2	2	6	

图13B

手动指派的原始码元 $\longleftrightarrow$ 编号转换表

S _{RAW} (CD, DB, AD, AC, CB, AB)	每个R _{TERM} 处的电压				差分电压						S(4:0)
	DO	CO	BO	AO	DB	CB	AD	AC	CD	AB	
010010	-1	3	1	-3	-2	2	-2	-6	4	-4	0x0
001101	-1	-3	1	3	-2	-4	4	6	-2	2	0x1
110010	1	3	-1	-3	2	4	-4	-6	2	-2	0x2
101101	1	-3	-1	3	2	-2	2	6	-4	4	0x3
100000	3	-1	1	-3	2	-2	-6	-2	-4	-4	0x4
001111	-3	-1	1	3	-4	-2	6	4	2	2	0x5
110000	3	1	-1	-3	4	2	-6	-4	-2	-2	0x6
011111	-3	1	-1	3	-2	2	6	2	4	4	0x7
000100	1	-3	3	-1	-2	-6	-2	2	-4	-4	0x8
110011	1	3	-3	-1	4	6	-2	-4	2	2	0x9
001100	-1	-3	3	1	-4	-6	2	4	-2	-2	0xA
111011	-1	3	-3	1	2	6	2	-2	4	4	0xB
001010	-3	1	3	-1	-6	-2	2	-2	4	-4	0xC
110001	3	1	-3	-1	6	4	-4	-2	-2	2	0xD
001110	-3	-1	3	1	-6	-4	4	2	2	-2	0xE
110101	3	-1	-3	1	6	2	-2	2	-4	4	0xF
011010	-3	3	1	-1	-4	2	2	-4	6	-2	0x10
011011	-3	3	-1	1	-2	4	4	-2	6	2	0x11
100100	3	-3	1	-1	2	-4	-4	2	-6	-2	0x12
100101	3	-3	-1	1	4	-2	-2	4	-6	2	0x13
000000	1	-1	3	-3	-2	-4	-4	-2	-2	-6	0x14
111101	1	-1	-3	3	4	2	2	4	-2	6	0x15
000010	-1	1	3	-3	-4	-2	-2	-4	2	-6	0x16
111111	-1	1	-3	3	2	4	4	2	2	6	0x17

基数为16的码元“主要”

替代码元“替代”

图14

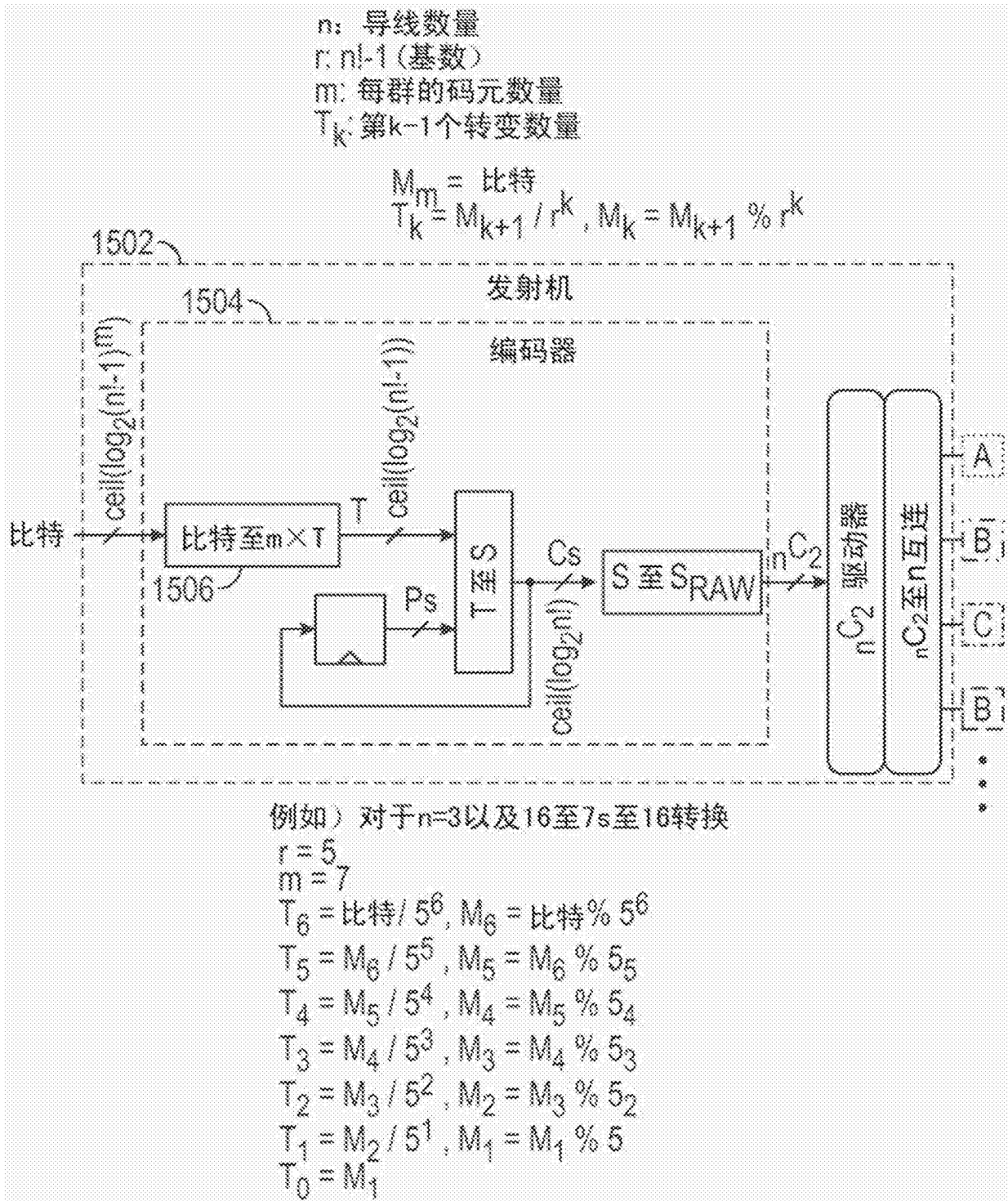
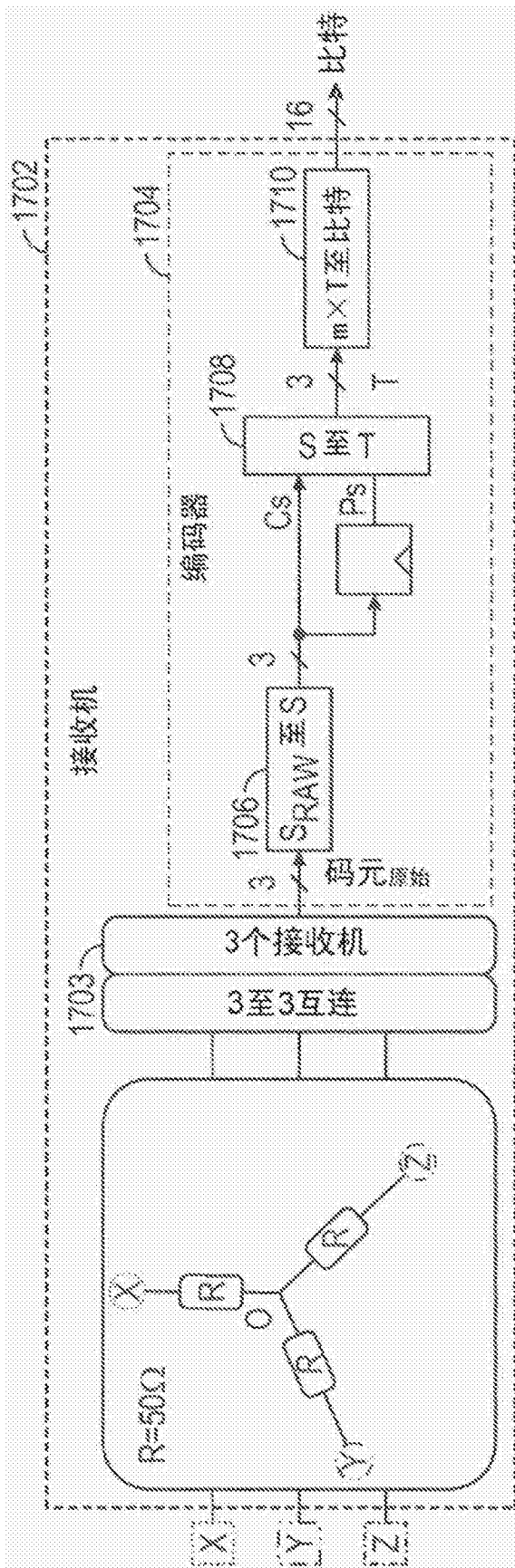


图15



CS = 码元_{开始} == 6? 0: 码元_{开始}

$$1 = p_{s+1}^2 \leq p_s^2 \leq (p_s+1) \leq p_{s+1} + 6$$

$$\begin{aligned} \text{比特} &= T_6x^6 + T_5x^5 + T_4x^4 + T_3x^3 + T_2x^2 + T_1x + T_0 \\ &= T_6x^6 + T_5x^5 + T_4x^4 + T_3x^3 + T_2x^2 + T_1x + T_0 \end{aligned}$$

图17

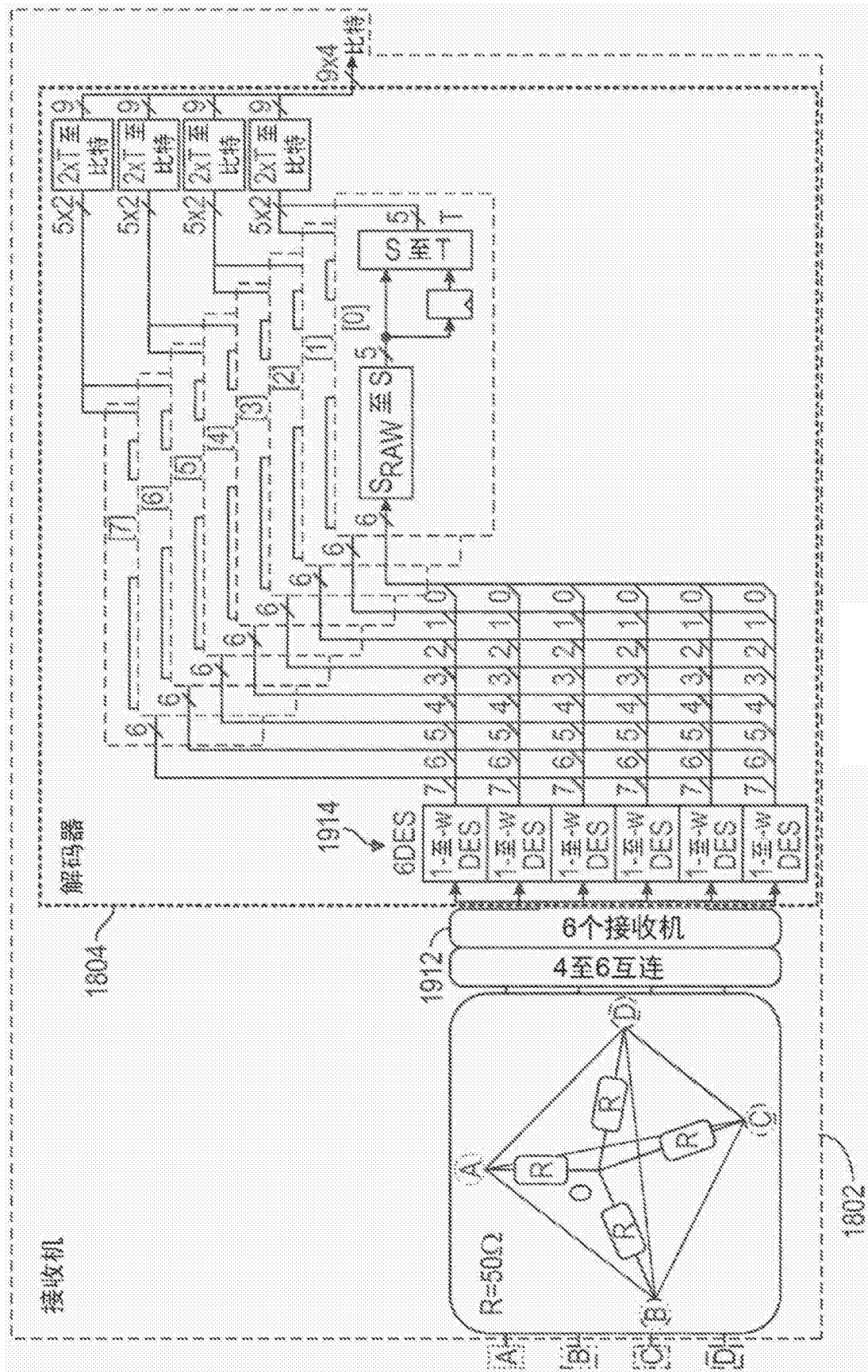


图19

n	利用率 %	码元/群	比特/群	Int(bpg)	比特/ 循环	比特/ 导线	比特/导线 ×2
3	99.98	28	65.0140	65	2.3214	0.7738	1.5476
3	99.92	25	58.0482	58	2.3200	0.7733	1.5467
3	99.84	22	51.0824	51	2.3182	0.7727	1.5455
3	99.74	19	44.1166	44	2.3158	0.7719	1.5439
3	99.59	32	74.3017	74	2.3125	0.7708	1.5417
3	99.59	16	37.1508	37	2.3125	0.7708	1.5417
3	99.50	29	67.3359	67	2.3103	0.7701	1.5402
3	99.39	26	60.3701	60	2.3077	0.7692	1.5385
3	99.39	13	30.1851	30	2.3077	0.7692	1.5385
3	99.24	23	53.4043	53	2.3043	0.7681	1.5362
3	99.06	30	69.6578	69	2.3000	0.7667	1.5333
3	99.06	20	46.4386	46	2.3000	0.7667	1.5333
3	99.06	10	23.2193	23	2.3000	0.7667	1.5333
3	98.90	27	62.6921	62	2.2963	0.7654	1.5309
3	98.80	17	39.4728	39	2.2941	0.7647	1.5294
3	98.70	24	55.7263	55	2.2917	0.7639	1.5278
3	98.64	31	71.9798	71	2.2903	0.7634	1.5269
3	98.44	21	48.7605	48	2.2857	0.7619	1.5238
3	98.44	14	32.5070	32	2.2857	0.7619	1.5238
3	98.44	7	16.2535	16	2.2857	0.7619	1.5238
3	98.10	18	41.7947	41	2.2778	0.7593	1.5185
3	97.88	11	25.5412	25	2.2727	0.7576	1.5152
3	97.62	15	34.8289	34	2.2667	0.7556	1.5111
3	96.90	12	27.8631	27	2.2500	0.7500	1.5000
3	96.90	8	18.5754	18	2.2500	0.7500	1.5000
3	96.90	4	9.2877	9	2.2500	0.7500	1.5000
3	95.71	9	20.8974	20	2.2222	0.7407	1.4815
3	94.75	5	11.6096	11	2.2000	0.7333	1.4667
3	93.31	6	13.9316	13	2.1667	0.7222	1.4444
3	86.14	3	6.9658	6	2.0000	0.6667	1.3333
3	86.14	2	4.6439	4	2.0000	0.6667	1.3333
3	86.14	1	2.3219	2	2.0000	0.6667	1.3333

图20

n	利用率 %	码元/群	比特/群	Int(bpg)	比特/ 循环	比特/ 导线	比特/导线 ×2
4	99.96	23	104.0419	104	4.5217	1.1304	2.2609
4	99.92	25	113.0890	113	4.5200	1.1300	2.2600
4	99.89	27	122.1362	122	4.5185	1.1296	2.2593
4	99.86	29	131.1833	131	4.5172	1.1293	2.2586
4	99.84	31	140.2304	140	4.5161	1.1290	2.2581
4	99.48	32	144.7540	144	4.5000	1.1250	2.2500
4	99.48	30	135.7069	135	4.5000	1.1250	2.2500
4	99.48	28	126.6597	126	4.5000	1.1250	2.2500
4	99.48	26	117.6126	117	4.5000	1.1250	2.2500
4	99.48	24	108.5655	108	4.5000	1.1250	2.2500
4	99.48	22	99.5184	99	4.5000	1.1250	2.2500
4	99.48	20	90.4712	90	4.5000	1.1250	2.2500
4	99.48	18	81.4241	81	4.5000	1.1250	2.2500
4	99.48	16	72.3770	72	4.5000	1.1250	2.2500
4	99.48	14	63.3299	63	4.5000	1.1250	2.2500
4	99.48	12	54.2827	54	4.5000	1.1250	2.2500
4	99.48	10	45.2356	45	4.5000	1.1250	2.2500
4	99.48	8	36.1885	36	4.5000	1.1250	2.2500
4	99.48	6	27.1414	27	4.5000	1.1250	2.2500
4	99.48	4	18.0942	18	4.5000	1.1250	2.2500
4	99.48	2	9.0471	9	4.5000	1.1250	2.2500
4	98.95	21	94.9948	94	4.4762	1.1190	2.2381
4	98.90	19	85.9477	85	4.4737	1.1184	2.2368
4	98.83	17	76.9006	76	4.4706	1.1176	2.2353
4	98.74	15	67.8534	67	4.4667	1.1167	2.2333
4	98.63	13	58.8063	58	4.5615	1.1154	2.2308
4	98.47	11	49.7592	49	4.4545	1.1136	2.2273
4	98.25	9	40.7121	40	4.4444	1.1111	2.2222
4	97.90	7	31.6649	31	4.4286	1.1071	2.2143
4	97.27	5	22.6178	22	4.4000	1.1000	2.2000
4	95.79	3	13.5707	13	4.3333	1.0833	2.1667
4	88.43	1	4.5236	4	4.0000	1.0000	2.0000

图21

n	利用率 %	码元/群	比特/群	Int(bpg)	比特/ 循环	比特/ 导线	比特/导线 ×2
5	100.00	19	131.0015	131	6.8947	1.3789	2.7579
5	99.97	28	193.0549	193	6.8929	1.3786	2.7571
5	99.91	27	186.1601	186	6.8889	1.3778	2.7556
5	99.91	18	124.1067	124	6.8889	1.3778	2.7556
5	99.91	9	62.0534	62	6.8889	1.3778	2.7556
5	99.85	26	179.2653	179	6.8846	1.3769	2.7538
5	99.82	17	117.2119	117	6.8824	1.3765	2.7529
5	99.79	25	172.3704	172	6.8800	1.3760	2.7520
5	99.71	32	220.6342	220	6.8750	1.3750	2.7500
5	99.71	24	165.4756	165	6.8750	1.3750	2.7500
5	99.71	16	110.3171	110	6.8750	1.3750	2.7500
5	99.71	8	55.1585	55	6.8750	1.3750	2.7500
5	99.65	31	213.7394	213	6.8710	1.3742	2.7484
5	99.63	23	158.5808	158	6.8696	1.3739	2.7478
5	99.59	30	206.8445	206	6.8667	1.3733	2.7467
5	99.59	15	103.4223	103	6.8667	1.3733	2.7467
5	99.55	22	151.6860	151	6.8636	1.3727	2.7455
5	99.53	29	199.9497	199	6.8621	1.3724	2.7448
5	99.45	21	144.7912	144	6.8571	1.3714	2.7429
5	99.45	14	96.5274	96	6.8571	1.3714	2.7429
5	99.45	7	48.2637	48	6.8571	1.3714	2.7429
5	99.35	20	137.8964	137	6.8500	1.3700	2.7400
5	99.29	13	89.6326	89	6.8462	1.3692	2.7385
5	99.11	12	82.7378	82	6.8333	1.3667	2.7333
5	99.11	6	41.3689	41	6.8333	1.3667	2.7333
5	98.89	11	75.8430	75	6.8182	1.3636	2.7273
5	98.62	10	68.9482	68	6.8000	1.3600	2.7200
5	98.62	5	34.4741	34	6.8000	1.3600	2.7200
5	97.90	4	27.5793	27	6.7500	1.3500	2.7000
5	96.69	3	20.6845	20	6.6667	1.3333	2.6667
5	94.27	2	13.7896	13	6.5000	1.3000	2.6000
5	87.02	1	6.8948	6	6.0000	1.2000	2.4000

图22

n	利用率 %	码元/群	比特/群	Int(bpg)	比特/ 循环	比特/ 导线	比特/导线 ×2
6	99.94	31	294.1853	294	9.4839	1.5806	3.1613
6	99.93	29	275.2056	275	9.4828	1.5805	3.1609
6	99.91	27	256.2259	256	9.4815	1.5802	3.1605
6	99.90	25	237.2462	237	9.4800	1.5800	3.1600
6	99.88	23	218.2665	218	9.4783	1.5797	3.1594
6	99.86	21	199.2868	199	9.4762	1.5794	3.1587
6	99.83	19	180.3071	180	9.4737	1.5789	3.1579
6	99.80	17	161.3274	161	9.4706	1.5784	3.1569
6	99.78	32	303.6751	303	9.4688	1.5781	3.1556
6	99.76	30	284.6954	284	9.4667	1.5778	3.1556
6	99.76	15	142.3477	142	9.4667	1.5778	3.1556
6	99.73	28	265.7157	265	9.4643	1.5774	3.1548
6	99.70	26	246.7360	246	9.4615	1.5769	3.1538
6	99.70	13	123.3680	123	9.4615	1.5769	3.1538
6	99.67	24	227.7564	227	9.4583	1.5764	3.1528
6	99.63	22	208.7767	208	9.4545	1.5758	3.1515
6	99.63	11	104.3883	104	9.4545	1.5758	3.1515
6	99.58	20	189.7970	189	9.4500	1.5750	3.1500
6	99.52	18	170.8173	170	9.4444	1.5741	3.1481
6	99.52	9	85.4086	85	9.4444	1.5741	3.1481
6	99.45	16	151.8376	151	9.4375	1.5729	3.1458
6	99.35	14	132.8579	132	9.4286	1.5714	3.1429
6	99.35	7	66.4289	66	9.4286	1.5714	3.1429
6	99.23	12	113.8782	113	9.4167	1.5694	3.1389
6	99.05	10	94.8985	94	9.4000	1.5667	3.1333
6	99.05	5	47.4492	47	9.4000	1.5667	3.1333
6	98.79	8	75.9188	75	9.3750	1.5625	3.1250
6	98.35	6	56.9391	56	9.3333	1.5556	3.1111
6	98.35	3	28.4695	28	9.3333	1.5556	3.1111
6	97.47	4	37.9594	37	9.2500	1.5417	3.0833
6	94.84	2	18.9797	18	9.0000	1.5000	3.0000
6	94.84	1	9.4898	9	9.0000	1.5000	3.0000

图23

n	利用率 %	码元/群	比特/群	Int(bpg)	比特/ 循环	比特/ 导线	比特/导线 ×2
7	99.98	27	332.0709	332	12.2963	1.7566	3.5132
7	99.96	17	209.0817	209	12.2941	1.7563	3.5126
7	99.94	24	295.1741	295	12.2917	1.7560	3.5119
7	99.93	31	381.2666	381	12.2903	1.7558	3.5115
7	99.89	28	344.3698	344	12.2857	1.7551	3.5102
7	99.89	21	258.2774	258	12.2857	1.7551	3.5102
7	99.89	14	172.1849	172	12.2857	1.7551	3.5102
7	99.89	7	86.0925	86	12.2857	1.7551	3.5102
7	99.86	32	393.5655	393	12.2812	1.7545	3.5089
7	99.85	25	307.4730	307	12.2800	1.7543	3.5086
7	99.83	18	221.3806	221	12.2778	1.7540	3.5079
7	99.81	29	356.6687	356	12.2759	1.7537	3.5074
7	99.79	22	270.5763	270	12.2727	1.7532	3.5065
7	99.79	11	135.2881	135	12.2727	1.7532	3.5065
7	99.76	26	319.7720	319	12.2692	1.7527	3.5055
7	99.74	30	368.9677	368	12.2667	1.7524	3.5048
7	99.74	15	184.4838	184	12.2667	1.7524	3.5048
7	99.71	19	233.6795	233	12.2632	1.7519	3.5038
7	99.69	23	282.8752	282	12.2609	1.7516	3.5031
7	99.60	20	245.9784	245	12.2500	1.7500	3.5000
7	99.60	16	196.7827	196	12.2500	1.7500	3.5000
7	99.60	12	147.5871	147	12.2500	1.7500	3.5000
7	99.60	8	98.3914	98	12.2500	1.7500	3.5000
7	99.60	4	49.1957	49	12.2500	1.7500	3.5000
7	99.45	13	159.8860	159	12.2308	1.7473	3.4945
7	99.38	9	110.6903	110	12.2222	1.7460	3.4921
7	99.20	10	122.9892	122	12.2000	1.7429	3.4857
7	99.20	5	61.4946	61	12.2000	1.7429	3.4857
7	98.92	6	73.7935	73	12.1667	1.7381	3.4762
7	97.57	3	36.8968	36	12.0000	1.7143	3.4286
7	97.57	2	24.5978	24	12.0000	1.7143	3.4286
7	97.57	1	12.2989	12	12.0000	1.7143	3.4286

图24

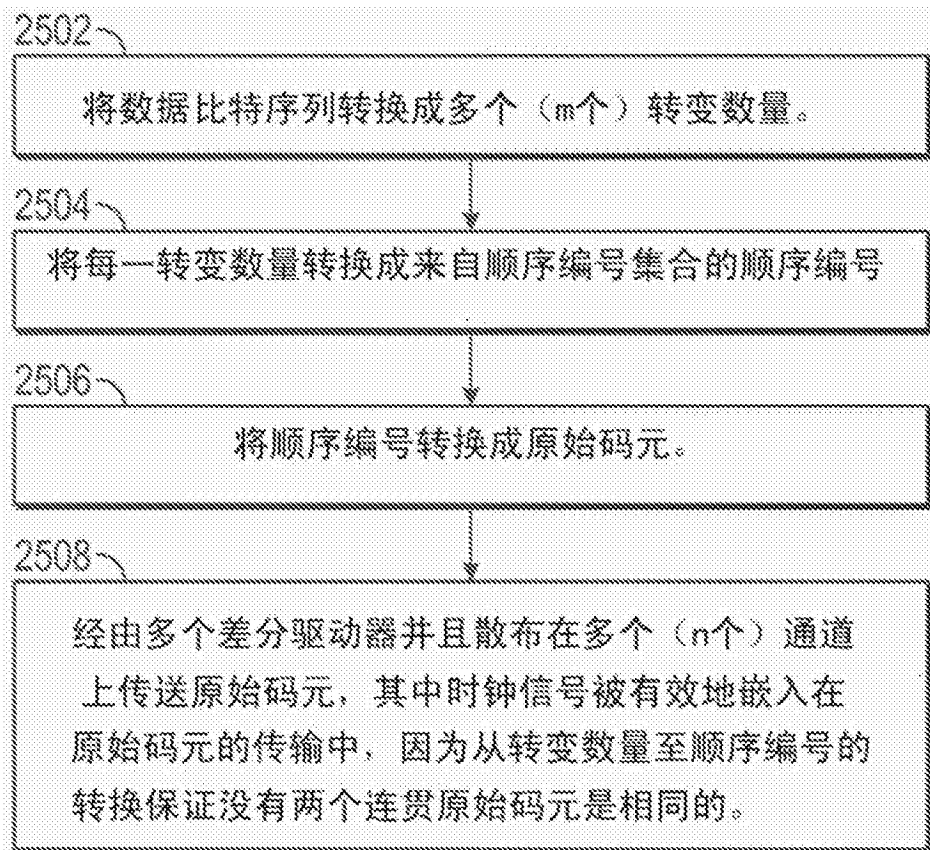


图25

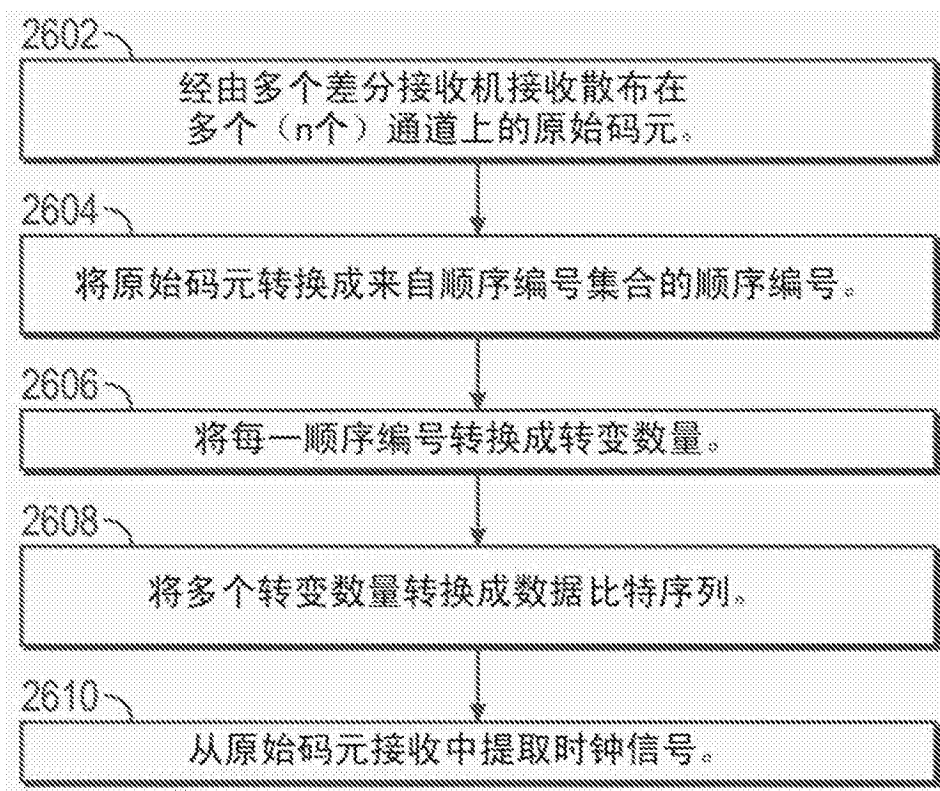


图26

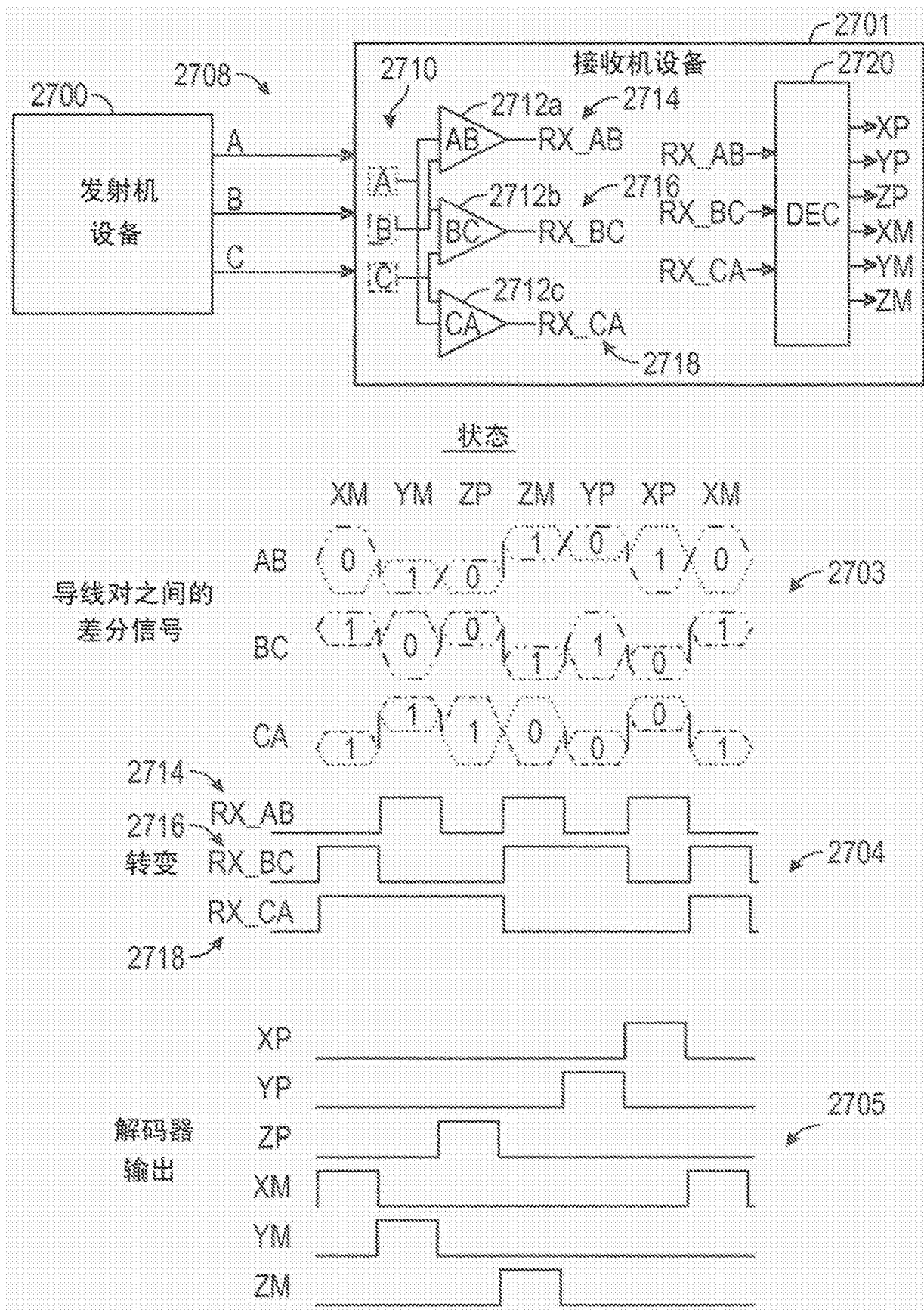


图27



图28