



(12) 发明专利

(10) 授权公告号 CN 1805329 B

(45) 授权公告日 2011.09.21

(21) 申请号 200510121564.1

审查员 庞艳

(22) 申请日 2005.12.15

(30) 优先权数据

60/637192 2004.12.16 US

11/069194 2005.02.28 US

(73) 专利权人 塔米拉斯珀私人有限责任公司

地址 美国特拉华州

(72) 发明人 A·J·埃格利特

(74) 专利代理机构 中科专利商标代理有限责任
公司 11021

代理人 王波波

(51) Int. Cl.

H04L 1/20 (2006.01)

(56) 对比文件

US 20030215038

A1, 2003.11.20, [0026]-[0055] 段、图 1, 3, 6.

US 20040066864

A1, 2004.04.08, [0010]-[0012] 段, 图 1, 2.

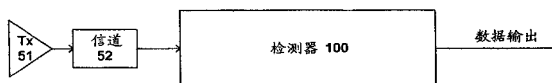
权利要求书 1 页 说明书 12 页 附图 9 页

(54) 发明名称

通过数字传输链路接收数据的方法

(57) 摘要

通过数字传输链路接收数据的方法和装置。一种过采样序列检测器对采样数据进行操作并跟踪采样数据的检测可靠性。检测器对于不同的采样相位独立地分析采样序列,然后拾取允许最可靠检测的采样序列。对于不同的采样相位,检测器检查一些回顾和预见的信息量以便改善简单的逐个符号的检测。过采样信息被用于更进一步改善检测性能。



1. 一种用于在接收机单元接收在数字传输链路上传输的数据的方法,包括:

接收产生用于表示符号序列的采样序列,该采样序列包括对应于该采样序列的第一和第二相位的第一和第二采样流;

从第一和第二采样流中检测第一和第二符号;

通过弹性缓存器对第一和第二符号进行延时;

估计第一和第二符号的第一和第二检测可靠性;

选择第一和第二符号中具有较高检测可靠性的那个符号;以及

基于所选符号和先前所选符号之间的相位差,确定所选符号是否与来自符号序列的相关符号的符号边界交叉,如果是,则调整弹性缓存器的延时长度,以控制边界交叉,如果不是,则不调整弹性缓存器。

2. 根据权利要求1所述的方法,包括:

通过多个采样流表示编码符号流,其中该多个采样流的每一个是通过根据对该多个采样流的每一个都不同的相位对符号流进行L倍的过采样而产生的,这样每个采样流包括L个采样,以表示从编码符号流中采样的每个相关符号;

通过将长度为W的观测窗中来自每个相关的采样流的采样与检测模式表中的模式进行比较,分别检测来自每个采样流的符号,其中W大于L,并且检测模式表中的模式至少与观测窗等长,这样检测来自每一个采样流的符号以表示来自编码符号流的相关符号;

在每个符号的基础上根据估计的每个检测的符号的可靠性给从该多个采样流的每一个中检测的每个符号分配检测量度,其中根据检测模式表中的模式进行给每个检测的符号的检测量度的分配;

用弹性缓存器对来自多个采样流的每个检测的符号进行延时;

至少部分地基于分配给从采样流中检测的符号的相关检测量度,确定从该多个采样流的每一个中检测的检测符号中的哪一个是最可靠的符号;

选择最可靠的符号来表示来自符号流的相关符号;

基于所选符号和先前所选符号之间的相位差,确定所选符号是否与来自符号流的相关符号的符号边界交叉,如果是,则调整弹性缓存器的延时长度,以控制边界交叉,如果不是,则不调整弹性缓存器;并且

在接收机单元接收所选符号。

通过数字传输链路接收数据的方法

技术领域

[0001] 本发明一般涉及通过模拟或者数字传输链路进行的数据接收,特别涉及在存在确定性和非确定性的抖动时提供鲁棒操作的接收机。

背景技术

[0002] 高速数字通信代表一种日益重要的技术。越来越多数量的模拟通信链路被提供可靠性、链路质量(例如,低误码率(BER))、对终端用户来说使用简单和低成本的高速串行链路所取代。

[0003] 例如,现代计算机显示器和高清晰度电视(HDTV)显示器常常采用根据诸如数字可视界面(DVI)和高清晰度多媒体界面(HDMI)这样的工业标准的高速串行链路。这种链路总的数字速率通常在吉比特/秒的范围。例如,DVI链路波特率接近5吉波特,有很多厂商能提供超出这个限制很大余量的产品。

[0004] 另一个实例是用于连接计算机系统存储设备的串行高级技术附件(ATA)接口。这个应用也要求大约2吉比特/秒和更高的数据速率。

[0005] 其它的实例是诸如吉比特以太网(例如1000Base-T)以及新兴的10-吉比特的以太网(10GBASE-T)标准的高速局域网(LAN)应用。

[0006] 支承这些和相似实例的链路有许多共同的要求。它们要求高速运行,有容许大的确定性和/或非确定性的数据抖动的能力,容许大的频率偏移和/或频率调制(FM)的能力,较有效的BER,可在小的水平眼图张开度下运行(例如,0.4单位间隔(UI)的眼图宽度(eye width)规格是很难得的)的能力,高跟踪带宽、快速相位捕获和良好的跟踪范围,这仅仅是举出的几个例子。

[0007] 关于这种高速串行链路接收机的一个普遍的实施是基于结合所谓的“相位拾取”的数据过采样(over-sampling)来选择“好的”采样流的接收机。常见的DVI链路实施方式使用一个过采样因子 $L=3$,数据采集常在没有任何相位跟踪的情况下进行。一旦过采样数据流是可用的,相位拾取器通过使用一些类型的相位跟踪方法(例如传输相位平均)保留 L 个采样中的一个采样,以便选择最佳的采样。

[0008] 尽管这种技术实施起来相对简单和廉价,但也经受着各种问题。例如,今天所使用的一些线路码有非常宽的频谱(即,差的游程长度 (d,k) -约束constraint)。这个问题对可以具有从1到14个符号的游程的DVI/HDMI链路尤为尖锐。因为稳定性和鲁棒性,相位拾取器必须在相对窄的跟踪带宽下运行。这种方法的不利方面是不能跟踪在过采样数据中出现的确定性抖动。窄的跟踪带宽还限制稳定速度(即,相位捕获时间),引起差的瞬变性能,有时使捕捉范围变窄。

[0009] 相位拾取方法中的一个基本问题是它不能在检测处理中使用全部可用的过采样数据。常见的相位拾取器实施方式简单地丢弃每个符号的 $L-1$ 个采样,仅保留 L 个接收采样中的一个采样。因此,过采样数据仅被用于相位跟踪,不用于数据检测。因此,相位拾取器丢弃许多携带关于正在被解码的符号的有用信息的数据采样。在常见的DVI/HDMI实施

方式中,大约 2/3 的对接收机可用的信息被简单地丢弃。

[0010] 如果数据速率适度并且眼图张开度足够,例如当链路采用短的高质量屏蔽双绞线 (STP) 电缆时,可以容许由这种相位拾取方式引起的损失,并有助于廉价的接收机实施。然而,一旦数据速率增加和 / 或使用更高或者更低质量的电缆,例如非屏蔽双绞线 (UTP) 电缆,那么可用的眼图宽度变得更小,接收机性能开始迅速下降。

[0011] 例如,可见这种传统的取相位 - 平均的相位拾取器要求眼图张开度大于 2 个采样间隔,以便可靠地检测数据符号。在使用 $L = 3$ 个采样的常见实施方式的情况中,这意味着如果可避免由相位拾取引起的数据误差,那么眼图宽度应当至少是 $0.7UI$ 。在低速和 / 或短传输距离的情况中完全可以接受这种限制,但是在更高数据速率的情况下这种限制变得花费昂贵,并且常常需要相对昂贵的良好均衡和预加重技术。

[0012] 此外,因为高速和 / 或高过采样因子通常不提供平行处理采样的简单方式,所以在上述情况中许多传统的相位拾取方法都难于实施。从接收机实施方式的立场看,常希望接收机的复杂性线性地依赖过采样因子,由此规定更窄的眼图规范 (eye specification)。许多现有的相位拾取方法都不显示这种复杂度的线性增加,而是代替随过采样因子的平方或者甚至更多地增长。

[0013] 所谓的“多数表决”过采样接收机代表本领域另一类公知的接收机。在过去,这种技术被广泛地用于低速链路,如果良好的眼图张开度可用,例如 $0.9UI$ 或者更高的眼图张开度,便可以提供一些线性的和 / 或非线性的数据过滤,并常常改善链路的鲁棒性和 BER。这种接收机实施起来还很便宜,这一点也是这种接收机在过去能够流行的重要原因。然而,一旦眼图张开度开始收缩,如在今天的高速串行链路中常见的那样,这种“多数表决”接收机的性能就会迅速变得不令人满意。此外,通常这种接收机在处理高频率偏移和频率调制的数据流以及有高度非确定性抖动的数据流时非常困难。

[0014] 与传统的“模拟”时钟和数据恢复 (也称为 CDR) 技术相比,在数字域的数据检测和 / 或相位跟踪常在不同制造处理之间的可移植性方面和简单设计,低廉成本方面有优势。此外,许多在模拟实施方式中不可用的非线性检测和 / 或过滤方法在数字域中实施起来常常是廉价的。此外,即使传统的模拟 CDR 经常提供良好的相位跟踪性能,但这很少能改善数据检测本身,大多数模拟 CDR 有用于数据检测的简单采样器,仅对每个符号进行一次采样。因此,如果出现由于高的确定性抖动引起的实质的眼图闭合,模拟 CDR 的性能则变得不令人满意。

[0015] 通常,如果在眼图张开度的最大高度这一点检测 (采样) 每个符号,则可以实现最好的接收机操作性能。然而,找到这种采样点非常困难,也常常不可能找到。为了确定一个“最佳”的采样点,接收机通常考虑用以指导的数据转换。然而,这种数据转换通常不能恰好在符号的边界发生。它们的位置受到确定性抖动 (由先前符号历史确定的符号间干扰 (ISI) 引起的),由传输系统中各个部件的噪声引入的非确定性抖动 (例如时钟发生器相位噪声,导致振幅和时间偏移的采样器的不精确的信号限幅,加性和倍增的信道噪声等),串音诱发的抖动和频率偏移,以及在这种链路中经常出现的频率调制的影响。此外,链路还常常承受进一步使接收变复杂的非线性失真。

[0016] 因为转换的位置可以不提供关于当前符号最佳采样点的可靠信息,所以接收机尝试过滤从转换的观测中得到的采样相位信息,以便获得一些可接受的采样相位。例如,许多

现有的实施方式应用有低通转移特性的线性过滤器或者一些有噪声抑制特性的非线性过滤器（例如，中值滤波器）。然而，在逐个符号的意义上，使用这种方法获得的采样相位位置常常不是最佳的。这是因为难于从纯非确定性相位噪声和频率偏移中区分确定性分量（例如 ISI 和串音）。尽管非确定性的相位噪声应该被滤除（即，抑制），但是应当跟踪确定性部分以及频率偏移以便提供良好的接收机性能。过滤方法常不能区分这种区别，因此总体检测性能受到损害。

[0017] 还有其它已知的尝试改善这种性能的方法，例如基于维特比算法（VA）及其改进的序列检测方法，以及反馈检测方案和判决反馈均衡（DFE）。这些方法起到很好的作用，也允许计算 ISI 和其它确定性分量，因此能改善整体性能并简化采样相位跟踪任务（因为目前仅需要处理非确定性部分以及频率偏移）。然而，基于序列检测的方法在实施起来常常很复杂并且昂贵，通常需要多比特采样（即，模拟 - 数字转换（ADC））和逐个符号进行的密集计算。一旦数据速率开始接近吉比特 / 秒的范围，序列检测器迅速变得不实用。即使在低速时，基于 VD（维特比解码）和 DFE 的接收机实施起来也是昂贵得几乎不允许实施。

[0018] 因此，需要一种简单且有效的方式来构建有过采样数据接收的数据接收机，它不会遭受困扰传统相位拾取器、“多数表决”和模拟 CDR 接收机的问题。

发明内容

[0019] 公开一种对采样数据进行操作的过采样序列检测器。该检测器基于新颖的跟踪采样数据的检测可靠性的方法。检测器分别分析关于不同采样相位的采样序列，然后拾取允许最可靠检测的采样序列。对于不同的采样相位，检测器插入一些回顾和预见的信息，以用于改善简单的逐个符号的检测。此外，过采样的信息被用于进一步改善检测性能。

附图说明

[0020] 在附图的各个图形中借助实例但不是限制地举例说明本发明，其中相同的参考数字涉及相似的元件，其中：

[0021] 图 1 是举例说明依照本发明实施例的在数字传输链路上接收数据的方法和系统的高级框图。

[0022] 图 2 是举例说明依照本发明实施例的在数字传输链路上接收数据的方法的流程图。

[0023] 图 3 是举例说明依照本发明实施例的数据检测器的高级框图。

[0024] 图 4 是举例说明依照本发明实施例的模式（pattern）析器的框图。

[0025] 图 5 是举例说明依照本发明一个实施例的模式（pattern）分析器元件的框图。

[0026] 图 6 是举例说明依照本发明一个实施例的弹性缓存器的框图。

[0027] 图 7 是举例说明依照本发明一个实施例的流切换器的框图。

[0028] 图 8 是举例说明依照本发明一个实施例的弹性缓存器控制器的框图。

[0029] 图 9 是举例说明用于实施本发明实施例的示例性计算机系统的框图。

具体实施方式

[0030] 现在详细地参考本发明的特定实施例，在附图中举例说明其实例。尽管将结合特

定的实施例描述本发明,但是应该理解并不意图将本发明限制为所描述的实施例。相反地,是意图覆盖可能包括如附加权利要求所定义的、在本发明精神和范围内的替换、修改和等效方案。

[0031] 当通过数字传输链路发送过采样数据时,所希望的是在存在确定性和非确定性的抖动时提供鲁棒操作的数字接收机。由于 ISI,即使过采样数据重复超过每一比特一个采样(例如 3 倍,其示例性的过采样因子 $L = 3$),比特值 b 不能作为极佳的重复采样序列 bbb 到达接收机,但是可能代替地在采样序列中包括不等于 b 的数值。例如,比特值 0 可以明确地得到 000,或者根据抖动得到 001、100、101 或者任何其它组合。因此,难题是不能根据多个不能明确指示符号的采样推断符号。

[0032] 如上所述,一种显而易见的方法是使用采样的多数表决。不幸地,这种方法在很多情况中不起作用,因为假定知道在已发送的采样流中的符号边界,但是实质上确定和跟踪本身就是难题。另一种方法通过检查转换,以及在有些情况下通过执行跟踪回路来跟踪信号相位。然而,相位跟踪不能区分由影响理想采样位置(采样时钟相对于符号边界的位置)的非确定性抖动引起的相移(因此,应当跟踪这种相移)以及由信道存储器引起的不影响理想采样位置的相移(因此不必跟踪)。关于相位跟踪的变形包括根据由 ISI 引起的边界移动是高速过程并且以高于跟踪回路带宽的速率发生的观测结果,通过修改跟踪回路的带宽尝试使跟踪回路对信道变化的灵敏度降低。尽管该方法在处理低频的不稳定性,例如这种涉及发生器频率或者无线电链路多普勒效应的问题,由电缆串音引起的宽噪声谱和起因于限幅器的高频抖动时正常工作,但是锁相环路(PLL)限制相位跟踪器的性能。因此,所希望的是改善接收机的性能。

[0033] 图 1 是举例说明依照本发明实施例的用于通过数字传输链路接收数据的方法和系统的高级框图。发送器 51 通过信道 52 发送采样数据。过采样序列检测器 100(以下也称为“接收机”)接收并解码传输信号。检测器 100 是根据跟踪采样数据的检测可靠性,而不是跟踪数据的最佳采样相位。仅为了举例的简单而不是作为限制,公开的一个或多个示例性的实施例具有使用二进制字母表(alphabet)的链路,其中每个符号取两个值的其中一个,如 +1 和 -1,或者 0 和 1。因此,公开的一个或多个实施例以每个采样一个比特的速率运行,过采样因子保持的低,在很多情况下可以在 2 和 4 个采样之间。然而,这些并不是所公开实施例运行的必要条件,本发明同样可适用于有更复杂(非二进制的)字母表的链路,并且可以类似地处理包含超过一比特信息的采样以及更高的过采样因子。

[0034] 举例来说,但不作为限制,在此显示的检测器 100 的操作中使用过采样因子 $L = 3$,并且使用单比特采样。不同于传统的序列检测器,检测器 100 分别分析不同采样相位(有 $1/L$ 的标称相位偏移)的采样序列,然后拾取允许最可靠检测的采样序列。对于不同的采样相位,检测器 100 检查一些预见和回顾信息(即预兆和后兆的数据)以便改善简单的逐符号检测。另外,过采样信息还被用于进一步改善检测性能。这一点与一旦已经决定采样相位就常常丢弃过采样数据的传统“相位拾取器”相反。

[0035] 本发明在检测性能方面是计算有效的,因此可以成功地应用在正好超过目前对传统序列检测器而言是可能的那些的数据速率和/或成本约束。

[0036] 可以通过将转换-驱动的相位跟踪(模拟或者数字)与本发明的一个或多个实施例进行组合来更进一步地改善检测性能。例如,可以通过跟踪环路减少相位噪声和频率偏

移,同时可以通过本发明的一个或多个实施例处理 ISI 和大的“相位跳跃”。实际上,可以通过窄带反馈环路跟踪缓慢的相位变化以及非确定性的相位噪声,同时本发明的过采样检测器 100 处理剩余的相位偏移(例如 ISI)。

[0037] 图 2 是举例说明依照本发明实施例的用于通过数字传输链路接收数据的方法流程图。从 12 开始解码第一个符号,按照下面的步骤描述该方法:

[0038] 对于每个采样相位(在总共 L 个相位之中,在 14 从第一个相位开始),在 16 检查观测窗 W 上的采样。这个窗口典型地包括对应于当前符号的 L 个采样以及来自先前的和下一个符号的一些采样。例如,对于过采样因子 $L = 3$ 的系统来说,典型的观测窗 W 可以包含 5 或者 6 个采样。在 $W = 5$ 的示例情况中,除来自当前符号的 $L = 3$ 个的采样以外可以使用一个预见采样和一个回顾采样。在 $W = 6$ 的示例情况中,除当前符号的 $L = 3$ 个采样以外可以使用两个回顾采样和一个预见采样,对于其它 L 和 W 的值是类似的。

[0039] 接下来在 18 使用上面选择的采样执行对每个可能采样相位的当前符号值的检测。如下所述,预见的和回顾的采样,以及关于采样流过去的判决和 / 或检测器 100 的输出流都可用来改善检测性能。

[0040] 在 20 估计每个采样相位(即估计判决不确定性)的检测可靠性。

[0041] 在 22 选择有最高检测可靠性(即具有最低的判决不确定性)的采样流。如下所述,一个或多个线性和 / 或非线性的过滤器可以控制选择处理以避免不必要的流转换。替换地,如果特定的实施例中需要,则可以使用没有过滤的逐个符号的转换。

[0042] 如果在 32 确定转换处理与符号边界交叉,则在 36 调整弹性缓存器。通常,如果在旧的和新的相位之间的差异的绝对值大于 $1/2 UI$ 则与符号边界交叉。例如,对于过采样因子 $L = 3$,当发生 $0 \rightarrow 2$ 或者 $2 \rightarrow 0$ 的转换时发生交叉。对于更大的过采样因子例如 $L = 5$,任何的下列转换将引起符号丢弃或者复制: $0 \rightarrow 4$ 、 $1 \rightarrow 4$ 、 $0 \rightarrow 3$ 、 $4 \rightarrow 0$ 、 $4 \rightarrow 1$ 和 $3 \rightarrow 0$,因为在所有这些情况中,在旧的和新的相位之间的差异是 3 或者 4,即大于 $1/2 UI$ (在这个情况中是 2.5)。下面描述弹性缓存器。可替换地,可以针对每个采样流调整当前符号指针以便解决边界交叉。

[0043] 在 38 对该流中的剩余符号重复执行上述步骤。

[0044] 下面更详细地描述以上的步骤。

[0045] 本发明的方法是假定采样相位正确,设法猜测当前的符号值,以及估计这种猜测的可靠性。经常地,每个可能的采样相位都会产生相同的符号值。例如,如果符号流具有相同值的长的符号游程(例如值为 1),检测器 100 处于这种游程的中间,那么,因为不管相位是多少检测到的值都是 1,所以采样相位没有意义。此外,这种猜测非常可靠,即与这种猜测相关的不确定性是零。尽管这是一个小例子,但是它举例说明了检测器 100 并不总是需要知道采样相位以便正确地检测符号值。

[0046] 作为另一个有相位 - 不变的检测结果的示例性序列,考虑以下的实例(这个实例中 $L = 3$ 和 $W = 5$):

[0047] 相位 0 :000. 111. 101. 111. 000,被解码作为 01010(见下文)。

[0048] 解码的数据 :--0-----0-----1----0---

[0049] 相位 1 :001. 111. 011. 110. 0,也被解码为 01010。

[0050] 相位 2 :011. 110. 111. 100. 0,被解码为 1010x(即早 1 个符号)。

[0051] 在上述实例中,有可能不考虑已选的采样相位进行正确的检测判决,因为在四个所示符号上全部三个被检测数据流都相同。

[0052] 然而,在上述实例中,对于不同的相位,检测可靠性是不同的。例如,因为所有的符号都有一种采样模式,它可以仅表示一个可能的符号值,所以相位 0 数据可以被可靠地解码。除了相位 0 的一个符号之外的所有符号都有清晰定义的符号边界。因为两个相邻的符号 1 可以被单个 0 采样分离,所以也可以确定地解码采样模式 1. 101. 1。这对应于有相对较高 ISI 的,引起过浅的零峰值的情况,因此产生窄的符号。因此,对应于相位 0 的序列被可靠地解码,并且具有指示低等级的不确定性的累积“检测量度”0。该想法用于向符号序列分配判决量度(以下也简单地称作量度),以便指示它们的不确定等级,并在解码输入的采样序列中的符号时使用这些判决量度。

[0053] 相反,猜测相位 1 和相位 2 流的数据的可靠性低于相位 0 流的。一些采样模式显得混淆和模糊,并且即使可以不模糊地检测到序列,在存在结合数据的相位抖动和 / 或频率调制的严重的 ISI 时检测也可能出错。因此,设置相位 1 和相位 2 流的量度高于相位 0 流,从而指示在对相位 1 和相位 2 流的检测处理中增加的不确定性(或者减少的可靠性)。例如,同相位 0 流的量度 0 相比,给相位 1 和相位 2 流的量度都是 2。

[0054] 作为举例说明而不是限制,下面的部分举例说明一些在接收机的示例性实现中建议的示例性检测模式。

[0055] 对于 $L = 3$ 和 $W = 5$ 的示例数据检测模式

[0056] 与其它的实施例一起,在这里举例说明两个检测器 100 的实施例:一个使用先前检测值(即对先前的符号判决的值)的实施例和一个不使用这种先前检测值的实施例。在此将这种使用先前值减少当前符号的检测的不确定性的方案称作判决反馈检测或者 DF 方案。这种方案有时以附加的硬件为代价改善检测的可靠性(即减少 BBR)。

[0057] 下面的列表给出了建议用于 DF 检测器的检测模式。请注意,为了节省空间(在这里的和在真实的硬件中),仅列表的第一半是明确提供的。通过反转该模式比特以及被检测的值产生第二半。可能使用其它压缩方案以便更简洁地表示列表数据。

模式	先前值	量度	相位	数值	模式数目
00000	0	0	0	0	0/63
00000	1	0	0	0	1/62
00001	0	0	0	0	2/61
00001	1	0	0	0	3/60
00010	0	0	-1	1	4/59
00010	1	0	-1	1	5/58
00011	0	1	1	0	6/57
00011	1	1	1	0	7/56
00100	0	0	0	1	8/55
00100	1	0	0	1	9/54
00101	0	1	0	0	10/53
00101	1	1	0	0	11/52
00110	0	1	-1	1	12/51
00110	1	0	1	0	13/50
00111	0	1	-1	1	14/49
00111	1	1	1	0	15/48
01001	0	0	1	1	16/47
01001	1	0	1	1	17/46
01001	0	1	0	0	18/45
01001	1	1	0	0	19/44
01010	0	1	0	1	20/43
01010	1	1	0	1	21/42
01011	0	1	0	1	22/41
01011	1	1	0	0	23/40
01100	0	0	1	1	24/39
01100	1	0	1	1	25/38
01101	0	1	1	1	26/37
01101	1	1	-1	0	27/36
01110	0	0	0	1	28/35
01110	1	0	0	1	29/34
01111	0	0	0	1	30/33
01111	1	0	0	1	31/32

[0058]

表 1

[0059] 在表 1 中,注意到仅仅 4 个模式 0 011 0,0 011 1,0 101 1 和 0 110 1(十六进记数法 06、07、0B 和 0D) 是取决于 DF 值(在名为“先前值”的列中显示)。因此可以得到结论,在这些模式的检测质量合理降低的情况下可以实现简化的非 DF 检测器。

[0060] 流选择

[0061] 一旦计算每个流中的当前符号的量度便进行流选择。典型地,可以使用一些过滤器类型(例如有限脉冲响应(FIR)、无限脉冲响应(IIR)或非线性的),以便保持一些期望的相位相干量,以免在检测流之间的迅速转换,因为这种转换可以降低检测可靠性。

[0062] 如果在得到转换到另一个流的判决之前考虑规定长度的过去的检测历史,则可使检测器 100 更精确。这是因为如先前所述,几个独立过程,例如 ISI、相位抖动、频率调制和/或时钟频率偏移量,或者其它的处理会影响符号边界。一些处理,例如抖动、和在某程度上的 ISI 有短的相位相干,即逐个符号地高速变化。在每个流中的模式解码器处理这种短期的移位。

[0063] 其它的处理,例如频率偏移或者频率调制通常有长得多的相位相干周期,因此不适合由模式检测器单独处理。在这种情况下执行流转换,因此完成(间接地)采样相位跟踪。

[0064] 在一个关于 DVI/HDMI 应用的实施例中,过滤器可以简单地实现为每个流上采用(即 $1-D$ 个过滤器的 L) 的移动平均,接下来是非线性的阈值处理。在当前流和候选流的量度之间的差异没有超过指定的应用相关的阈值,这样做防止了流的转换。

[0065] 在一个实施例中,选择有最小量度的候选流。然而,也可以使用其它的策略。

[0066] 如下所述,在最后选择符号值之前,可以在弹性缓存器的帮助下对候选流进行一些调整。

[0067] 流转换以及弹性缓存器的作用

[0068] 如果当前以及建议流之间的相位差异在选择新的候选流的过程中大于 $L/2$ (即大于 $1/2$ UI 相位跳跃),则检测处理与符号边界交叉。除非检测器 100 采用附加的步骤,要不然因为一个符号的复制或者丢弃将导致错误的检测(由相位跳跃方向确定)。

[0069] 例如,当使用接近于 UI 开始的流(例如具有过采样因子 $L = 3$ 的系统中的相位 0 流)以及选择在 UI 末端的流(例如在这种情况下中的相位 2 流)时,将会与符号边界交叉,除非采取必要的步骤,要不然将丢失被检测流中的一个符号(即,丢弃)。

[0070] 相反地,如果相位在反向移动 $1/2$ 的 UI 或更多(例如在上述实例中从相位 2 流到相位 0 流),符号可能被采样两次,因此引起在检测流中的符号复制。

[0071] 一个可能解决上述问题的解决方案包括增加弹性缓存器(EB),或者更普遍地在执行流选择之前增加可变符号延迟。在前向相位移动的情况下(例如在上述实例中从相位 0 流到相位 2 流),弹性缓存器的长度(即延迟的量)减少一个符号。相反地,在相位延迟的情况下(例如,在上述实例中从相位 2 流到相位 0 流),弹性缓存器的长度被增加一个符号。

[0072] 注意到在上述讨论中使用的抽象弹性缓存器(以及可变延迟)仅是为了举例说明这种概念,并不打算作为限制。在实际的实现中,可以使用各种方法。例如,可以使用基于多路复用器和保留数据的几个符号值的公共共享寄存器的平行实施方式。然而,下列描述沿用弹性缓存器的概念,可以理解的是实际的接收机可以使用任何足以充分地缓存和/或延迟数据的实施例。

[0073] 定时是这样的,使得在完成当前符号检测之后,解码器计算量度并做出转换判决。在这点上,也可以选择性地调整弹性缓存器的长度。然而,在下一个符号间隔期间选择新的

候选流。换句话说,可以使在符号检测中间发生的相位转换过程可视化,即在已经检测前一个符号之后以及处理下一个符号之前。其它的实现也是可能的。例如,可以没有任何延迟地对当前符号执行迅速的转换。然而,这种方法增加硬件复杂性,并且在许多实际的情况中没有显著地改善检测质量。

[0074] 在接收机运行的不稳定周期期间,例如启动或者建立新的链路连接时,弹性缓存器可以达到调整极限。在这种情况下,缓存器(即延迟的量)必须被强制地回到中心位置(re-center),导致可能的数据丢失。这种回到中心位置防止由弹性缓存器相对于调整极限弹回所引起的误差传播现象,引起由于在与符号边界交叉的情况中不能补偿复制或者丢弃的符号引起的多重误差。

[0075] 回到中心位置的处理通过检测弹性缓存器长度的减少低于零点(即要求负延迟)或者弹性缓存器长度的增加超过最大长度限制来工作。在两种情况中,缓存器长度(即延迟)被设置为预定义的值。这个值可能是固定的或者可被选择性地适当调整。

[0076] 在具备长度表示为 EB_{length} 的弹性缓存器的简单实现中,可以使用 $EB_{length}/2$ 的固定中心值。例如,对于7个符号的 EB_{length} ,中心值可以被设置为3个符号。

[0077] 可替换地,更详细的实现可以跟踪弹性缓存器的溢出趋势,使其回到中心位置,以便在最可能边界交叉的方向提供更多的缓冲空间。例如,如果有瞬时的正频率偏移,接收机采样器滞后在输入数据相位之后,采样相位可能提前并且可能碰到零延迟限制。在这种情况下,缓存器可以被集中到最大延迟的 $2/3$ 以便减少再次到达极限和引起数据误差的概率。

[0078] 总体结构

[0079] 现在描述当前数据检测方法和系统的示范性实现的结构。图3是举例说明依照本发明实施例的数据检测器100的高级框图。数据检测器100包括模式分析器110、量度过滤器130、流转换器140、弹性缓存器120和候选流选择器150。

[0080] 输入的 L 倍过采样数据101被馈送到模式分析器110。在这个块中,按照每个相位来分析输入数据模式,与检测量度一起产生 L 个候选值流。此外,这个块可以产生附加的辅助信息流152,例如采样相位估计、均衡测量的质量,等等。当前讨论限于值(V)的流111和量度(M)的流112

[0081] 量度过滤器130更进一步地处理总共 L 个每相位量度流,以下也叫作“原始量度”,以便抑制度量值的短期不稳定性和利用量度的内在一致性。检测器100通常尝试按照候选流质量变化的稳定趋势而不是按照每个符号来选择流。这是因为对流中每个符号的量度估计可能不可靠,尝试按照每个符号(即,以符号速率)跟踪它们可能引起错误的和/或不必要的流转换。量度过滤器130块完成这些目的并且基本上确定整个检测器100的跟踪动态。

[0082] 一旦已经过滤度量流(以下也称为“窜改的(cooked)”量度), L 个窜改的量度131被发给决定选择哪个流的流转换器块140。流转换器块140还控制弹性缓存器120的长度以便在符号边界交叉的情况下调整缓冲延迟。通过“相位-选择”信号142发送候选流选择数据,通过“符号-选择”信号141发送弹性缓存器120延迟值。

[0083] 弹性缓存器块120按照流转换器140指定的量延迟候选流。 L 个候选流被延迟相同的符号间隔数。在实际的硬件实现中,弹性缓存器块120能够作为在候选流之间共享的

多抽头移位寄存器和一系列选择适当的抽头的多路复用器来实现。其它的实现也是可能的,例如每个流一个可变长度的移位寄存器。

[0084] 延迟的候选流被施加到候选流选择器 (CSS) 150, 该候选流选择器保留 L 个流中的一个流, 丢弃剩余的数据。保留的流表示检测器 100 的输出数据流 151。本发明重要和有利的方面是在整个检测器 100 拓扑中缺少反馈环路。这一点非常便于平行检测器 100 实施方式, 并使得该技术适合非常高速的操作, 例如吉比特 / 秒数据速率以及更高。

[0085] 模式分析器

[0086] 图 4 是举例说明依照本发明实施例的模式分析器 110 的框图。输入数据采样 201 被发送到具有指定延迟的移位寄存器 210。移位寄存器 210 足够大以便保留 L_{SR} 个采样, 其中:

$$[0087] \quad L_{SR} = 2L + N_{LB} + N_{LA} - 1$$

[0088] 其中 L 是过采样因子, N_{LA} 是模式分析所需的预见 (LA) 采样数, N_{LB} 是回顾 (LB) 采样数, L_{SR} 是移位寄存器 210 的最小长度 (单位为采样)。

[0089] 例如, 在 3x 过采样以及 5 个采样模式分析 (即一个 LA 和一个 LB 采样, 不包括判决反馈 (DF) 采样) 的示例情况中, 移位寄存器 210 并行地产生 7 个采样。然而, 尽管使用移位寄存器 210 的特定示范性实施例使该操作更容易理解, 但是其它的实现也是可能的。例如, 有多路复用器的并行寄存器可用于方便数据的并行处理 (在高数据速率时可能是必须的), 并减少复杂性和 / 或功耗。

[0090] 采样 211 从移位寄存器 210 发送给一系列模式分析器元件 220、230...240。为了清楚起见, 图 4 中仅显示开始两个和最后一个模式分析器元件。每个模式分析器元件 220...240 观察按照每个符号呈现给它的适当的采样集合, 估计最可能的检测值以及估计量度和其它的辅助信息 (按照需要)。因此, 在线路 202 和 203 分别发送 L 对候选值和它们的量度。

[0091] 模式分析器元件

[0092] 图 5 是举例说明依照本发明一个实施例的模式分析器元件的框图。在线路 301 上从模式分析器 110 接收输入的总共 $L_{PA} = N_{LB} + L + N_{LA}$ 个采样的集合, 并应用其作为查阅表 (LUT) 310 的输入。此外, 从寄存器 320 接收先前决断的值 321。寄存器 320 按照每个符号被更新, 即每个检测器循环一次。

[0093] 查阅表 310 产生值“V”312、量度“M”331, 以及关于相位、均衡等等 的附加的辅助信息 313。可以使用只读存储器 (ROM) 或者使用随机逻辑实现查阅表 310。候选值“V”312 被保存在寄存器 320 中, 在处理下一个符号期间使用。

[0094] 弹性缓存器

[0095] 图 6 是举例说明依照本发明一个实施例的弹性缓存器 120 的框图。在线路 401 上接收总共 L 个候选流, 并按照每个符号将其移动到一系列 L 个移位寄存器 410、420...430。在延迟信号 402 的控制下, 通过相应的总共 L 个多路复用器 440、450...460 选择每个移位寄存器 410、420...430 中适当的抽头。多路复用器 440、450...460 的输出 441、451...461 包括 L 个候选流的集合, 其被适当地延迟并经由信号 405 发送。弹性缓存器 120 的其它实现也是可能的, 例如在使用适当的多路复用器的 L 个流之间共享的平行的装入寄存器, 或者其它的这种实施例。

[0096] 流转换器

[0097] 图 7 是举例说明依照本发明一个实施例的流转换器 140 的框图。在线路 501 上接收来自量度过滤器 130 的、被适当过滤（窜改的）的量度 M_{in} 。窜改的量度被发送到 L-路（L-way）比较器 510，该比较器按照每个符号来比较输入量度并确定当前 L 个候选流中的哪个候选流有最高的正确似然率。在该实施例中，量度越低，关于候选值的判决是正确的概率越高。因此，在这种情况下，比较器 510 可确定具备最低窜改量度的流。在线路 511 上将这种流的数目发送到转换过滤器 520。

[0098] 转换过滤器 (SF) 520 执行线性的和 / 或非线性的过滤以避免由于在量度方面有小的短期变化而转换候选流。这通常改善了系统操作。过滤算法取决于操作环境。例如固定的或者自适应阈值处理这样的简单非线性方案在很多情况下都工作得不错（例如 DVI/HDMI 实现）。例如，仅在当前使用的流和建议的流之间的量度差异大于指定的与实现相关的值时执行流转换。

[0099] 然后在线路 521 上将当前确定的候选流选择器“相位 - 选择”发送到候选流选择器块 150。此外，还将相位 - 选择信号发送给弹性缓存器控制器 (EBC) 530。弹性缓存器控制器 530 观察已选候选流的变化，如果该选择在任何方向与符号边界交叉便调整弹性缓存器延迟。在线路 531 将当前所需的延迟量发送到弹性缓存器 120。

[0100] 弹性缓存器控制器

[0101] 图 8 是举例说明依照本发明一个实施例的弹性缓存器控制器 530 的框图。从转换过滤器 520 接收建议的候选流数“相位”601，并在寄存器 610 将其延迟一个符号，通过减法器电路 620 和绝对值电路 630 确定在当前和以前的流数之间的差异。

[0102] 在线路 631 将选择器差异的绝对值发送到双路 (two-way) 比较器 640，该双向比较器 640 确定是否该差异大于 $L/2$ ，即大于符号间隔的一半。如果是的话，在线路 641 标记边界交叉条件，允许借助于与门 660 和 670 在延迟累积器 680 中调整该值。

[0103] 通过双路比较器 650 确定延迟调整的方向，确定采样是超前还是滞后输入数据。在边界交叉的情况下，延迟累积器 680 的值依照比较器 650 所作的比较结果增加或者减少。回到中心位置逻辑 690 检查出界延迟值并在此情况下复位延迟累积器 680 的值。参见上面关于回到中心位置处理的讨论。

[0104] 图 9 举例说明用于实现本发明实施例的示范性计算机系统 300。计算机系统 300 包括一个或多个中央处理单元 (CPU) 301、随机存取存储器 (RAM) 302、只读存储器 (ROM) 303、一个或多个外设 305 以及主存储设备 306 和 307。如在本领域众所周知的，ROM 用来将数据和指令单向传递到 CPU 301，而 RAM 典型地被用于以双向方式传递数据和指令。CPU 301 通常可以包括任何数目处理器。两个主存储设备 306 和 307 都可以包括任何适合的计算机可读介质。辅助存储介质 308 典型地是大容量存储器设备，其双向耦合到 CPU 301 并提供附加的数据存储容量。大容量存储器设备 308 是可用于存储包括计算机代码、数据等程序的计算机 - 可读介质。典型地，大容量存储器设备 308 是例如一般比主存储设备 306 和 307 慢的硬盘或者磁带之类的存储介质。大容量存储器存储设备 308 可以采取磁带或者纸带读出器或者其它的已知设备的形式。应当理解在适当的情况下，在大容量存储器设备 308 内保存的信息可以标准方式作为虚拟存储器被合并为 RAM 302 的一部分。

[0105] CPU 301 还耦合到一个或多个输入 - 输出设备 309，包括但不限于诸如视频监控

器、跟踪球、鼠标、键盘、扩音器、触敏显示器、转换器卡阅读器、磁带或者纸带读出器、写字板、指示笔、语音或者手写识别器的设备,或者其它已知的输入设备,当然例如包括其它的计算机。最后,使用通常在 304 所示的网络连接将 CPU 301 耦合到通信链路或者计算机或电信网 304,例如数字传输链路、互联网或内联网。通过具备这种通信链路,预计 CPU 301 可以在执行上述过采样数据接收和 / 或发送步骤的过程中在该链路上从网络接收信息,或者在该链路上向网络输出信息。这种常被表示为要由 CPU 301 执行的指令序列的信息,例如以嵌入到载波的计算机数据信号的形式从网络接收或者输出到网络。在计算机硬件和软件领域的技术人员熟知上述的设备和材料。计算机系统 300 接收一个或多个过采样数据流并处理它们,以便解码传送的符号以及执行如上所述的接收、检测和 / 或其它的处理步骤。用于执行这种接收、检测和 / 或其它的处理步骤的计算机指令可以保存在 RAM 302、ROM 303、主存储设备 306 和 307 和 / 或任何其它的计算机 - 可读介质中。

[0106] 所提供的之前描述的本发明的实施例作为举例说明和描述。不意图将本发明限制为已描述的精确形式。按照上述教导,其它的变化和实施例也是可能的,因此本发明的范围不是由这些详细说明来限制,而是由后面的权利要求来限制。

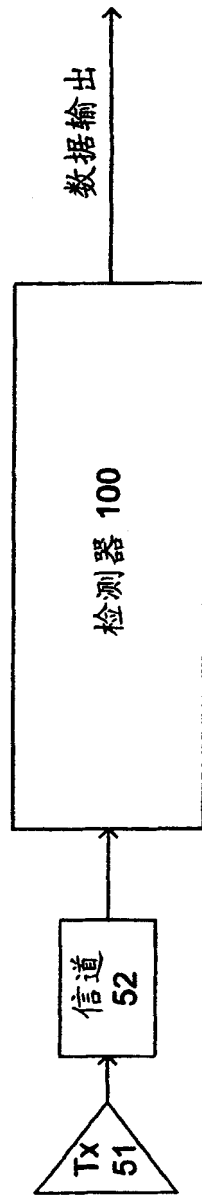


图 1

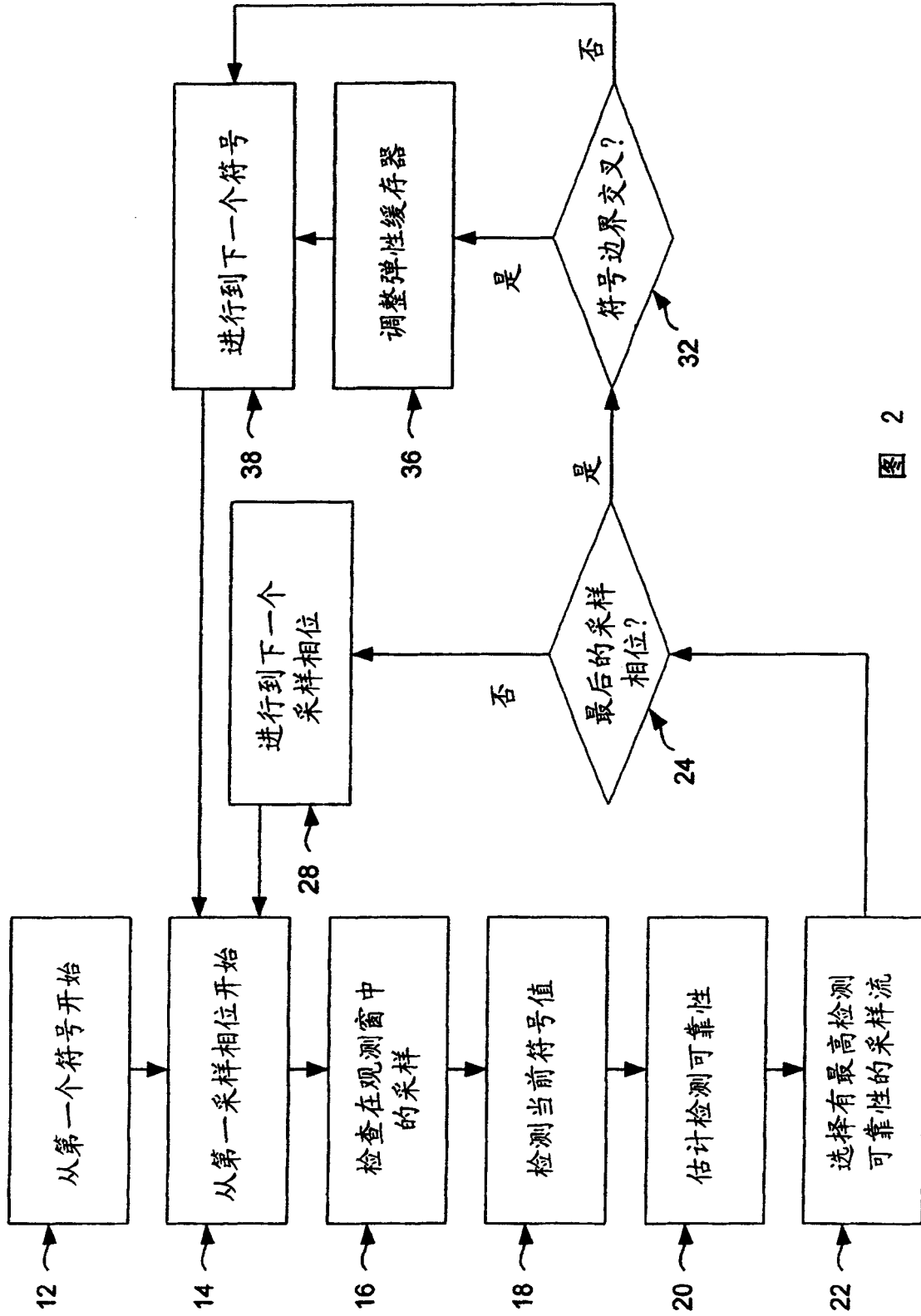


图 2

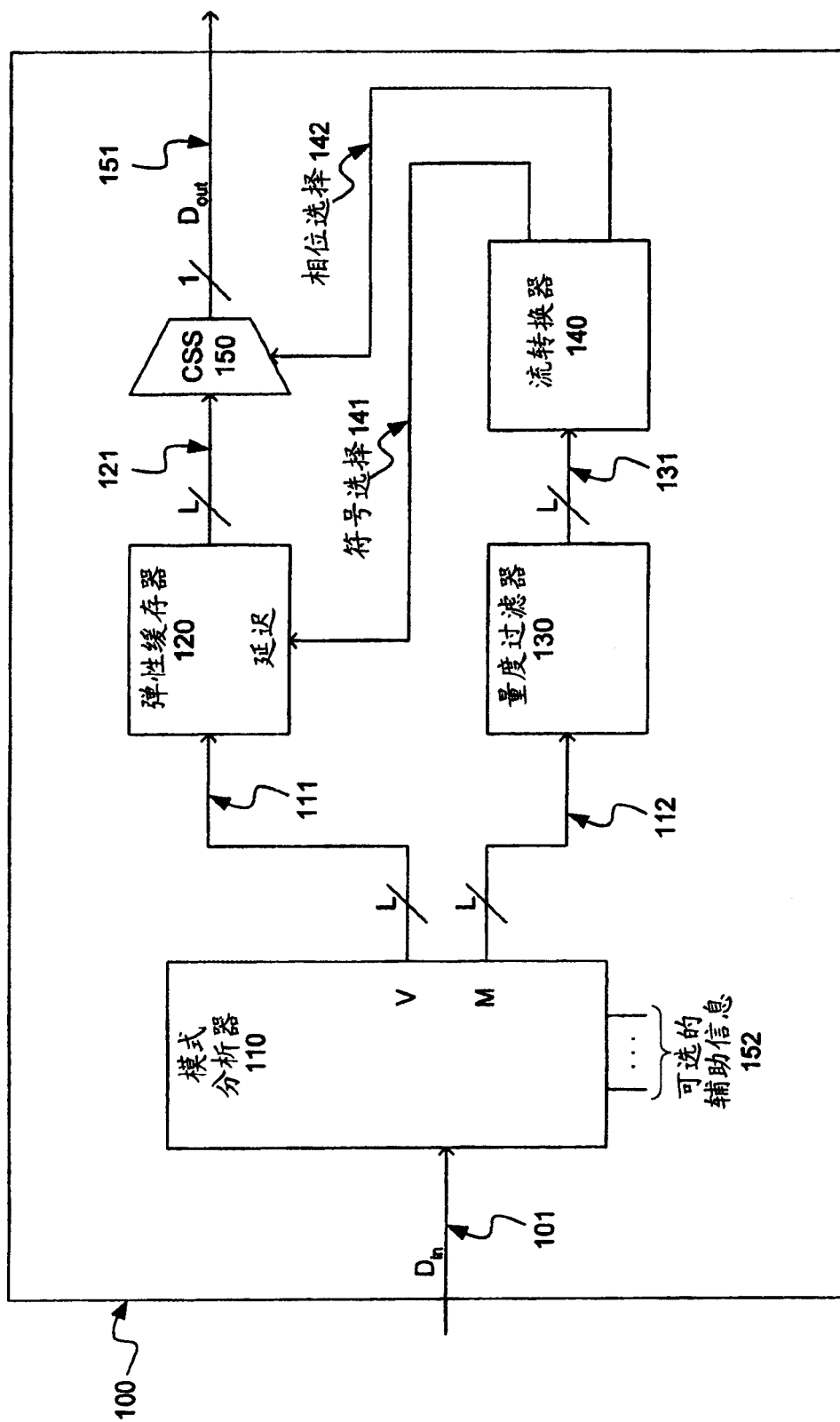


图 3

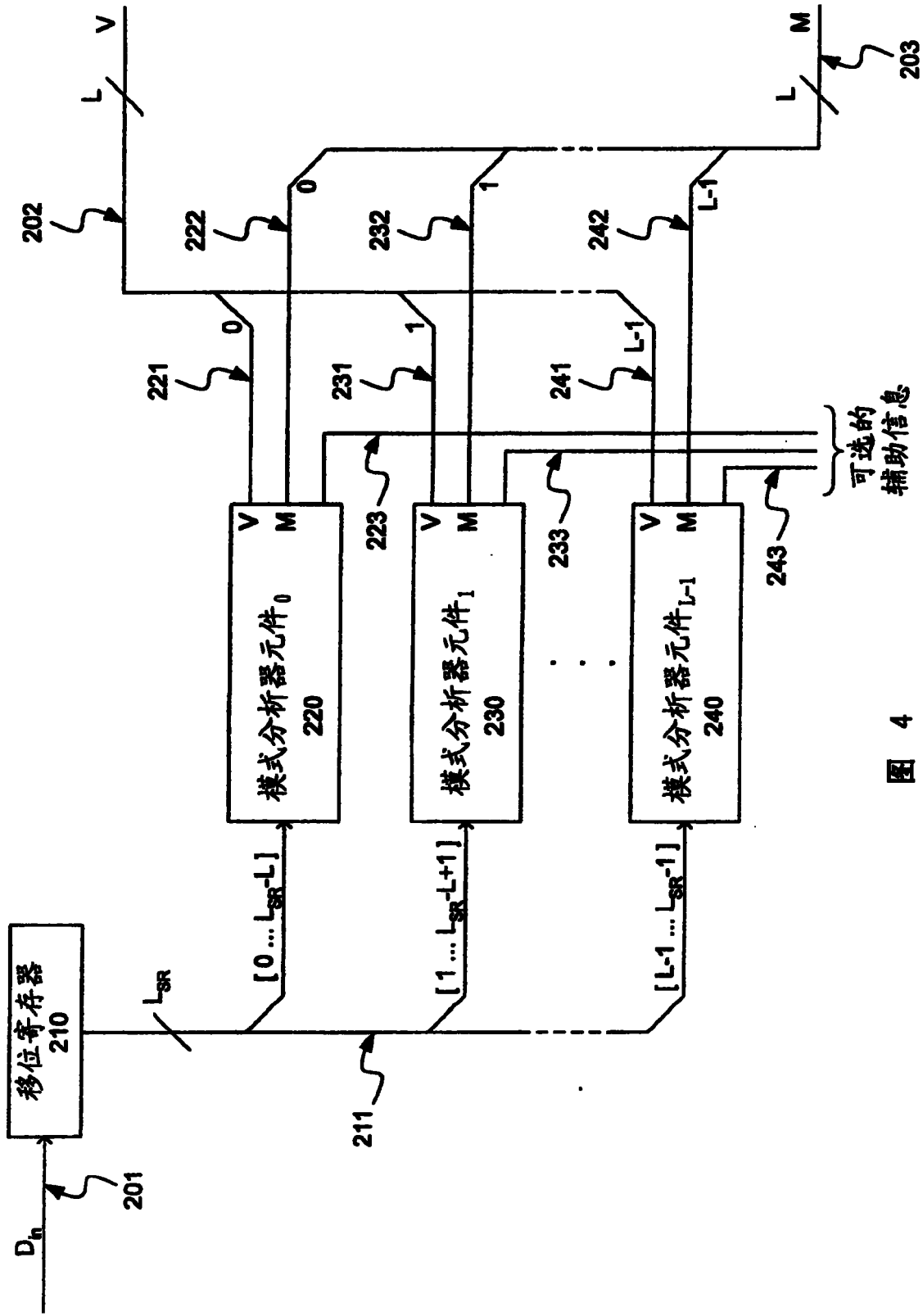


图 4

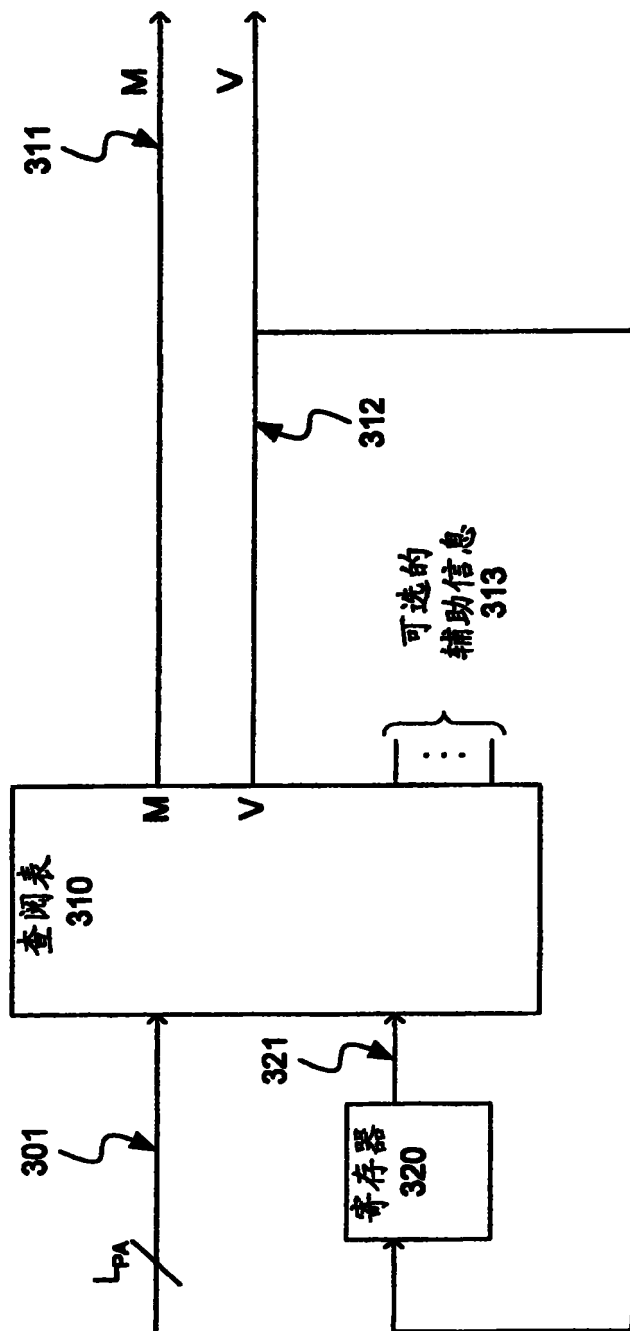


图 5

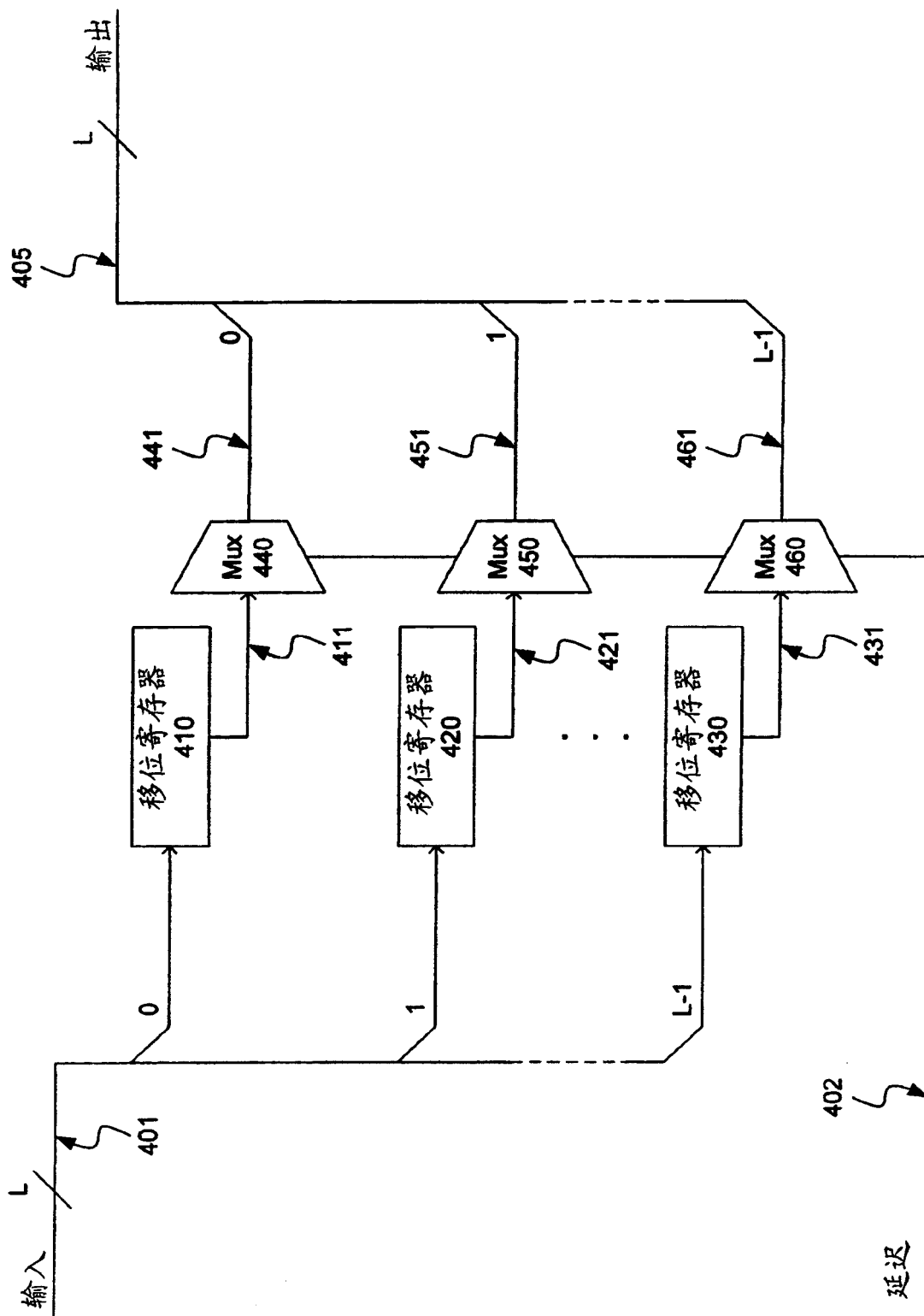


图 6

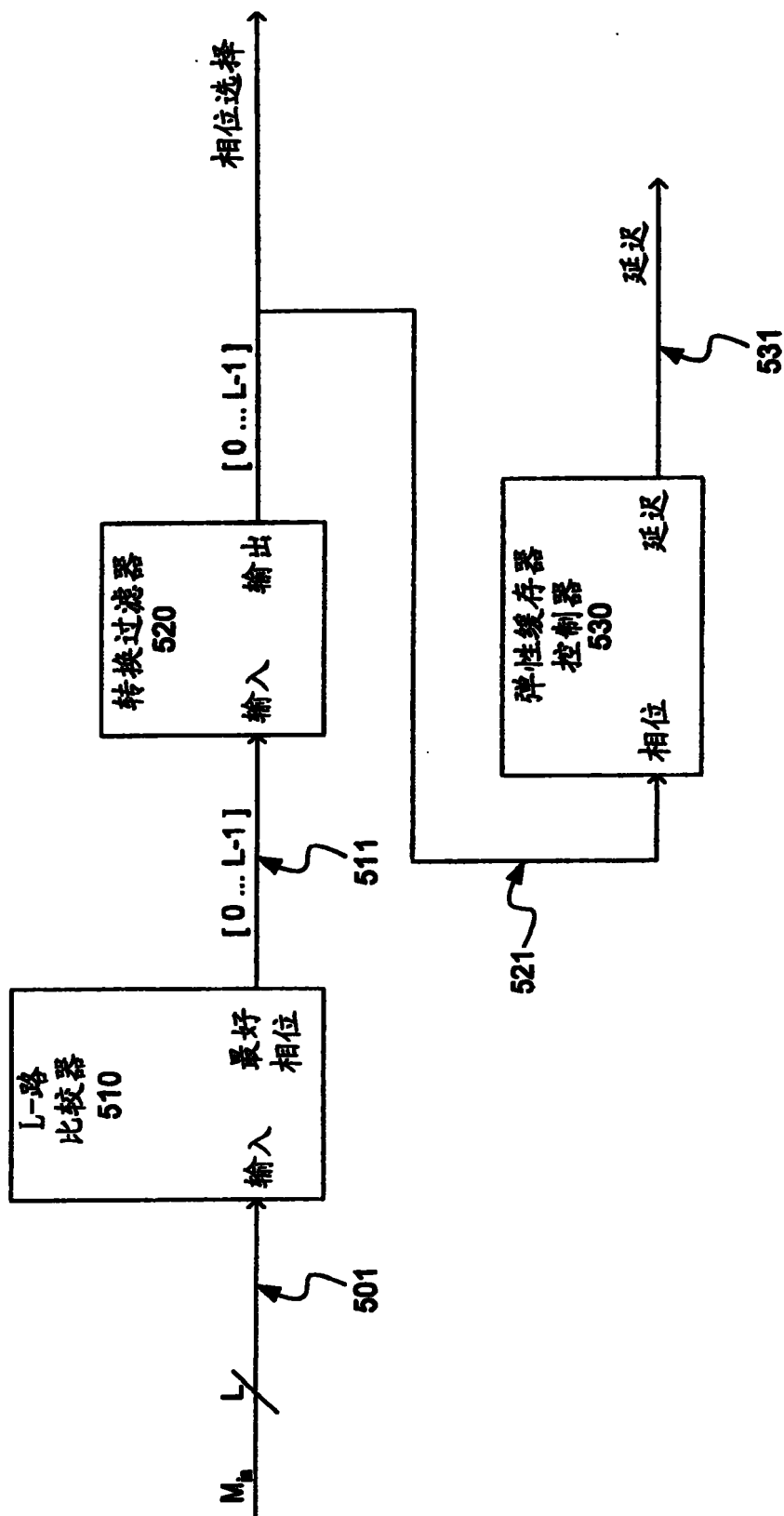


图 7

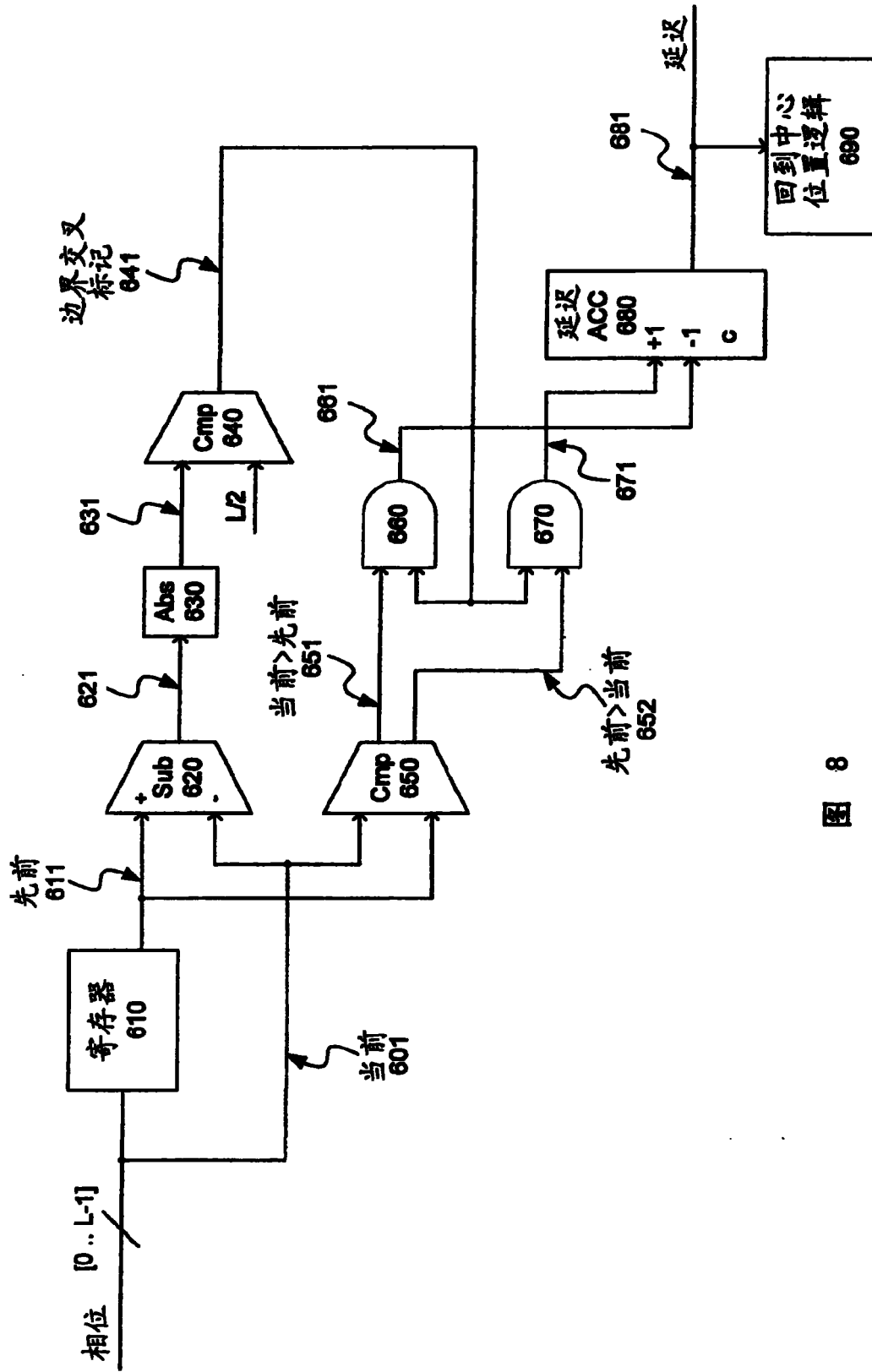


图 8

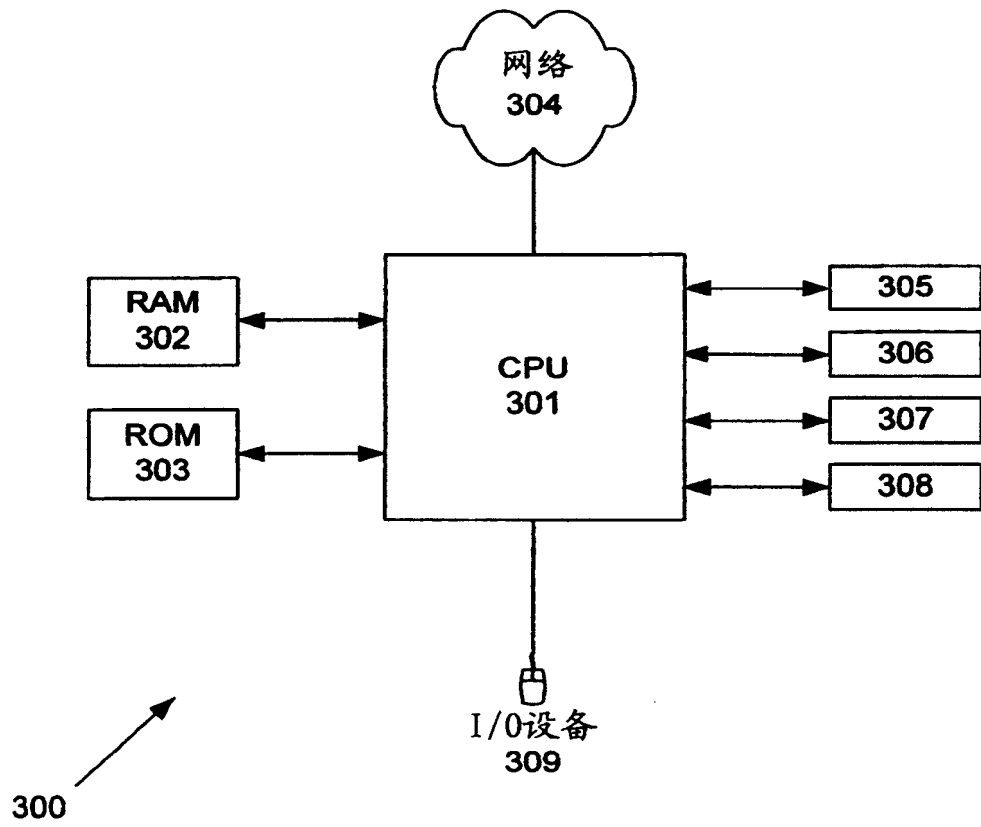


图 9