(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 116530059 A
(43)申请公布日 2023.08.01

(21)申请号 202180076516.2
(22)申请日 2021.10.12
(30)优先权数据
17/068,766 2020.10.12 US
(85)PCT国际申请进入国家阶段日
2023.05.12
(86)PCT国际申请的申请数据
PCT/US2021/054552 2021.10.12
(87)PCT国际申请的公布数据
WO2022/081559 EN 2022.04.21
(71)申请人 谷歌有限责任公司
地址 美国加利福尼亚州
(72)发明人 延斯·克里斯蒂安·波尔森
(74)专利代理机构 中原信达知识产权代理有限责任公司 11219
专利代理师 邓聪惠 周亚荣
(51)Int.Cl.
H04L 12/403(2006.01)

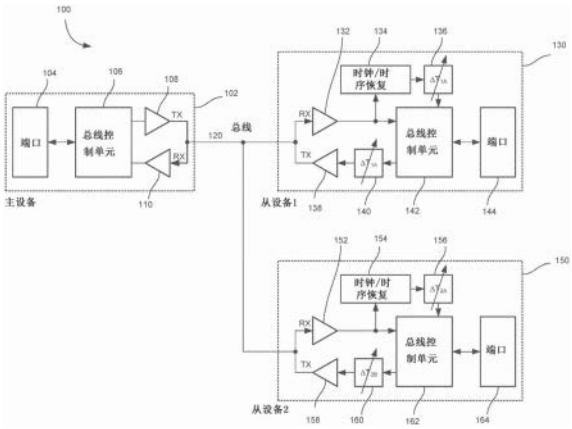
权利要求书5页 说明书19页 附图25页

(54)发明名称

动态时序校准系统和方法

(57)摘要

本文中提供了用于对连接到公共时分复用总线(120)的设备(130、150)中的内部延迟执行动态自适应和校正的系统和方法。这些方法允许设备(130、150)通过使用关于总线(120)的多个读取测量实际延迟来校正组件内和系统中的固有和未知延迟来在更高的总线频率下可靠地操作。内在噪声和抖动被用于提高测量的精度,从而实质上将这些不确定性用作自抖动以提高测量分辨率。在自适应期间,可以以多个步长调整延迟以加速自适应时间。



CN 116530059 A

1.一种方法,包括:

控制总线以促进多个设备之间的通信,其中,所述通信至少部分地基于时分复用时钟和数据协议来同步;

由第一设备将第一读取时序值写入到所述总线;

由所述第一设备读取与写入到所述总线的所述第一读取时序值相关联的第二读取时序值,其中,所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第一读取时序值和所述第二读取时序值来确定与所述第一设备相关联的读取时序,其中,与所述第一设备相关联的所述读取时序被配置为控制所述第一设备的读取延迟以校准在所述总线上从所述第一设备到所述第一设备的通信。

2.根据权利要求1所述的方法,还包括:

由第二设备将第一写入时序值写入到所述总线;

由所述第一设备读取与写入到所述总线的所述第一写入时序值相关联的第二写入时序值,其中,所述第一写入时序值和所述第二写入时序值的所述写入和所述读取发生在所述总线的相同数据行内;

由所述第一设备至少部分地基于所述第二写入时序值写入对所述第二设备的响应,其中,所述响应被配置为控制所述第二设备的写入延迟以校准在所述总线上从所述第二设备到所述第一设备的通信。

3.根据权利要求2所述的方法,还包括:

由所述第二设备将第三读取时序值写入到所述总线;

由第三设备读取与写入到所述总线的所述第三读取时序值相关联的第四读取时序值,其中所述第三读取时序值和所述第四读取时序值的所述写入和所述读取发生在所述总线的相同数据行内;

至少部分地基于所述第三读取时序值和所述第四读取时序值来确定与所述第三设备相关联的读取时序,其中,与所述第三设备相关联的所述读取时序被配置为控制所述第三设备的读取延迟以校准在所述总线上从所述第二设备到所述第三设备的通信。

4.根据权利要求2所述的方法,还包括:

由所述第二设备将第三读取时序值写入到所述总线;

由所述第一设备读取与写入到所述总线的所述第三读取时序值相关联的第四读取时序值,其中,所述第三读取时序值和所述第四读取时序值的所述写入和所述读取发生在所述总线的相同数据行内;

至少部分地基于所述第三读取时序值和所述第四读取时序值来确定与所述第一设备相关联的读取时序,其中,与所述第一设备相关联的所述读取时序被配置为控制所述第一设备的读取延迟以校准在所述总线上从所述第二设备到所述第一设备的通信。

5.根据权利要求2所述的方法,还包括:

由所述第一设备或所述第二设备将第三写入时序值写入到所述总线;

由所述第二设备读取与写入到所述总线的所述第三写入时序值相关联的第四写入时序值,其中,所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第三写入时序值和所述第四写入时序值来确定与所述第二设备相关联的写入时序,其中,与所述第二设备相关联的所述写入时序被配置为控制所述第二

设备的写入延迟以校准在所述总线上从所述第一设备到耦合到所述总线的其他设备的通信。

6.根据权利要求2所述的方法,还包括:

提供第一稳定模式以在初始校正期间使用第一延迟步长获得正确写入延迟位置的估计;

提供第二稳定模式以获得更精确的写入延迟位置以用于所述写入延迟的最终调整,其中,所述第二稳定模式采用小于所述第一延迟步长的第二延迟步长,并且比所述第一稳定模式稳定得更慢;以及

通过以下方式至少部分地基于多个读取写入时序值在所述第一稳定模式和所述第二稳定模式之间切换:

当所述第二写入时序值在预选的读取次数内没有改变时,切换到所述第一稳定模式;以及

当第二写入时序值在所述预选的读取次数内有一个或多个改变时,切换到所述第二稳定模式。

7.根据权利要求2所述的方法,还包括:

提供具有多个自适应速度的自适应模式;

确定正在读取的所述第二写入时序值是否等于先前的第二写入时序值,并且将所述自适应速度选择性地增加直到某个上限;以及

确定正在读取的所述第二写入时序值是否与所述先前的第二写入时序值不同,并且将所述自适应速度选择性地降低至某个下限。

8.根据权利要求1所述的方法,还包括:

由所述第一设备将第一写入时序值写入到所述总线;

由所述第一设备读取与写入到所述总线的所述第一写入时序值相关联的第二写入时序值,其中,所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第一写入时序值和所述第二写入时序值来确定与所述第一设备相关联的写入时序,其中,与所述第一设备相关联的所述写入时序被配置为控制所述第一设备的写入延迟以校准在所述总线上从所述第一设备到所述第一设备的通信。

9.根据权利要求1所述的方法,还包括:

调整从所述总线读取的所述第二读取时序值的平均值,以具有零和一之间的概率;以及

调整所述第一设备的所述读取延迟以避免读回相同值。

10.一种系统,包括:

第一设备,所述第一设备能够通信地耦合到总线,所述总线被配置为至少部分地基于时分复用时钟和数据协议来支持多个设备之间的同步通信,其中,所述第一设备被配置为:

将第一读取时序值写入到所述总线;

读取与所述第一读取时序值相关联的第二读取时序值,其中,所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第一读取时序值和所述第二读取时序值来确定与所述第一设备相关联的读取时序,其中,与所述第一设备相关联的所述读取时序被配置为控制所述第一

设备的读取延迟以校准在所述总线上从所述第一设备到所述第一设备的通信。

11.根据权利要求10所述的系统,还包括:

第二设备,所述第二设备能够通信地耦合到所述总线,其中,所述第二设备被配置为:

将第一写入时序值写入到所述总线;

至少部分地基于与写入到所述总线的所述第一写入时序值相关联的第二写入时序值来读取由所述第一设备生成的响应;以及

至少部分地基于由所述第一设备生成的所述响应来控制所述第二设备的写入延迟,以校准在所述总线上从所述第二设备到所述第一设备的通信。

12.根据权利要求11所述的系统,还包括能够通信地耦合到所述总线的第三设备,其中:

所述第二设备被配置为将第三读取时序值写入到所述总线;以及

所述第三设备被配置为:

读取与写入到所述总线的所述第三读取时序值相关联的第四读取时序值,其中,所述第三读取时序值和所述第四读取时序值的所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第三读取时序值和所述第四读取时序值来确定与所述第三设备相关联的读取时序,其中,与所述第三设备相关联的所述读取时序被配置为控制所述第三设备的读取延迟以校准在所述总线上从所述第二设备到所述第三设备的通信。

13.根据权利要求11所述的系统,其中:

所述第二设备被配置为将第三读取时序值写入到所述总线;以及

所述第一设备被配置为:

读取与写入到所述总线的所述第三读取时序值相关联的第四读取时序值,其中,所述第三读取时序值和所述第四读取时序值的所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第三读取时序值和所述第四读取时序值来确定与所述第一设备相关联的读取时序,其中,与所述第一设备相关联的所述读取时序被配置为控制所述第一设备的读取延迟以校准在所述总线上从所述第二设备到所述第一设备的通信。

14.根据权利要求11所述的系统,其中:

所述第二设备被配置为:

将第三写入时序值写入到所述总线;

读取与写入到所述总线的所述第三写入时序值相关联的第四写入时序值,其中,所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第三写入时序值和所述第四写入时序值来确定与所述第二设备相关联的写入时序,其中,与所述第二设备相关联的所述写入时序被配置为控制所述第二设备的写入延迟以校准在所述总线上从所述第一设备到耦合到所述总线的其他设备的通信。

15.根据权利要求10所述的系统,其中,所述第一设备被配置为:

估计所述第一设备的所述读取延迟;以及

调整所述第一设备的所述读取延迟,以引起由所述第一设备读取的所述第二读取时序

值的波动。

16.一种系统,包括:

第一设备,所述第一设备能够通信地耦合到总线,所述总线被配置为至少部分地基于时分复用时钟和数据协议来支持多个设备之间的同步通信;以及

第二设备,所述第二设备能够通信地耦合到所述总线,其中,所述第二设备被配置为:

将第一写入时序值写入到所述总线;

至少部分地基于与写入到所述总线的所述第一写入时序值相关联的第二写入时序值来读取由所述第一设备生成的响应;以及

至少部分地基于由所述第一设备生成的所述响应来控制所述第二设备的写入延迟,以校准在所述总线上从所述第二设备到所述第一设备的通信。

17.根据权利要求16所述的系统,其中,所述第一设备被配置为:

将第一读取时序值写入到所述总线;

读取与所述第一读取时序值相关联的第二读取时序值,其中,所述写入和所述读取发生在所述总线的相同数据行内;

至少部分地基于所述第一读取时序值和所述第二读取时序值来确定与所述第一设备相关联的读取时序;

将第三写入时序值写入到所述总线;

读取与所述第三读取时序值相关联的第四写入时序值,其中,所述写入和所述读取第三写入值和第四写入值发生在所述总线的相同数据行内;以及

至少部分地基于所述第一读取时序值和所述第二读取时序值来确定与所述第一设备相关联的写入时序,其中,与所述第一设备相关联的所述读取时序和所述写入时序被配置为控制所述第一设备的读取和写入延迟,以校准在所述总线上从所述第一设备到所述第一设备的通信。

18.根据权利要求16所述的系统,还包括能够通信地耦合到所述总线的第三设备,其中:

所述第二设备被配置为将第一读取时序值写入到所述总线;以及

所述第三设备被配置为:

读取与写入到所述总线的所述第一读取时序值相关联的第二读取时序值,其中,所述第一读取时序值和所述第二读取时序值的所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第一读取时序值和所述第二读取时序值来确定与所述第三设备相关联的读取时序,其中,与所述第三设备相关联的所述读取时序被配置为控制所述第三设备的读取延迟以校准在所述总线上从所述第二设备到所述第三设备的通信。

19.根据权利要求16所述的系统,其中:

所述第二设备被配置为将第一读取时序值写入到所述总线;以及

所述第一设备被配置为:

读取与写入到所述总线的所述第一读取时序值相关联的第三读取时序值,其中,所述第一读取时序值和所述第二读取时序值的所述写入和所述读取发生在所述总线的相同数据行内;以及

至少部分地基于所述第一读取时序值和所述第二读取时序值来确定与所述第一设备相关联的读取时序，其中，与所述第一设备相关联的所述读取时序被配置为控制所述第一设备的读取延迟以校准在所述总线上从所述第二设备到所述第一设备的通信。

20.根据权利要求16所述的系统，其中：

所述第二设备被配置为：

将第三写入时序值写入到所述总线；

读取与写入到所述总线的所述第三写入时序值相关联的第四写入时序值，其中，所述写入和所述读取发生在所述总线的相同数据行内；以及

至少部分地基于所述第三写入时序值和所述第四写入时序值来确定与所述第二设备相关联的写入时序，其中，与所述第二设备相关联的所述写入时序被配置为控制所述第二设备的写入延迟以校准在所述总线上从所述第一设备到耦合到所述总线的其他设备的通信。

动态时序校准系统和方法

[0001] 相关申请的交叉引用

[0002] 本申请是2020年10月12日提交的美国专利申请No.17/068,766的继续申请,其要求2019年10月11日提交的美国临时专利申请No.62/914,351的权益和优先权;并且是2019年6月27日提交的美国专利申请No.16/455,625的部分继续申请,这些申请中的每一个都通过引用整体并入本文。

[0003] 2019年6月27日提交的美国专利申请No.16/455,625要求2018年8月22日提交的美国临时专利申请No.62/721,412、2019年1月11日提交的美国临时专利申请No.62/791,607、2019年3月27日提交的美国临时专利申请No.62/824,985、以及2019年5月28日提交的美国临时专利申请No.62/853,654的权益和优先权,这些申请中的每一个都通过引用整体并入本文。

[0004] 本申请还涉及2019年9月3日提交的并且题为“Dynamic Timing Calibration Systems and Methods(动态时序校准系统和方法)”的美国临时专利申请No.62/895,449,其通过引用整体并入本文。

技术领域

[0005] 本申请总体上涉及用于利用公共线路上的数据和时钟信息的时分复用在附接到公共总线的多个设备之间的同步的系统和方法。

背景技术

[0006] 许多现代通信系统使得能够通过总线系统在两个或更多个连接的设备(例如,耳机和主机处理系统)之间传输数据。为了节省引脚数并且针对物理约束考虑,将两个或更多个信号组合在公共总线上以实现设备之间的信号的共享通常是优势。当这些信号被时分复用时,个体设备使用共同的同步源,其通常由时钟线路和标识数据的开始的帧开始标记提供。

[0007] 耦合到发送出时钟和帧开始信息的总线的设备在本文中将被称为主设备并且该信息的接收器在本文中将被称为从设备。时钟和帧开始的源可以是另一设备,和/或子电路可以将该信息提供给总线以供所有人使用。假设主设备向从设备传送信息,那么由从设备从主设备高速接收数据是很常见的,因为时钟和数据在系统中将引发相等的延迟。然而,取决于所涉及的电路中的内部传播延迟和诸如由该总线系统的总线直径和物理参数确定的外部延迟,在从从设备传播并且在主设备处接收的信号的到达时间上可能存在很大的不确定性。如果传输速度低,则这不是问题,但是对于高速链路,这种不确定性导致接收器眼图开口的尺寸减小,从而导致信号质量差或在更高传输速度下完全传输失败。

[0008] 可以通过制造非常高精度的电路来在某种程度上缓解该问题,同时有相关联的生产成本和硅面积的增加和/或个体部件的修整,这也增加了生产成本。即使努力减少由个体部件的延迟造成的这种变化性,仍然存在由于工艺、电压和温度(PVT)导致的变化。因此,虽然可以通过精确的设计和修整来提高传输速度,但任何PVT变化都将显著地限制在更高速

度下的性能。

[0009] 鉴于前述，本领域持续需要促进连接的设备之间的高速通信的低成本、低功率总线系统。

发明内容

[0010] 本公开公开了包括使得设备能够自校准的自动调整程序的总线系统和方法。由于消除了这些设备内部的精确修整或精确电路，本文中公开的方法使得能够生产低成本设备。

[0011] 在各种实施例中，一种方法包括控制总线以促进多个设备之间的通信，其中至少部分地基于时分复用时钟和数据协议来同步通信，使用第一设备将总线初始化为第一值，在总线上从第二设备接收与第一值不同的第二值，并且至少部分地基于第二值向第二设备发射对第二设备的响应，该响应被配置为控制第二设备的延迟以校准在总线上的来自第二设备的通信。

[0012] 该方法还可以包括：由第一设备估计第二设备的延迟，以及调整第二设备的延迟，以引起由第一设备读取的值的波动。在一些实施例中，该方法包括在第二值被稳定在总线上之前读回由第二设备写入的第二值。第二值可以由第二设备写入多次并且第一设备可以过滤读取以减少在总线上的当前读回值的估计的变化性。第一设备可以生成请求耦合到总线的一个或多个设备以回复来响应并且至少部分地基于回复来调整对设备的延迟的命令，其中回复包括来自每个响应设备的多个回复。该方法可以包括调整从总线读取的值的平均值以具有在零和一之间的概率并且相应地调整延迟以避免读回相同值。

[0013] 在一些实施例中，该方法还包括在总线电路中提供总线保持器，该总线保持器被配置为抑制能够引起未定义状态的泄漏电流，其中，总线保持器在总线未被驱动时维持总线的当前状态。

[0014] 该方法还可以包括提供第一稳定模式以在初始校正期间使用第一延迟步长获得正确延迟位置的估计，并且提供第二稳定模式以获得更精确的延迟位置以用于延迟的最终调整，第二稳定模式使用小于第一延迟步长的第二延迟步长并且比第一稳定模式稳定得更慢，并且基于多个读回值在第一稳定模式和第二稳定模式之间切换。在第一稳定模式和第二稳定模式之间切换可以进一步包括当值在N次读取内没有改变时切换到第一稳定模式，以及当值在N次读取内有一个或多个改变时切换到第二稳定模式。该方法可以进一步包括提供具有多个自适应速度的自适应模式，确定正在读回的值是否等于先前值，并且将自适应速度增加直到某个上限，以及确定正在读回的值是否与先前值不同，并且将自适应速度降低到某个下限。

[0015] 在各种实施例中，一种系统包括耦合到总线的第一设备，该第一设备被配置为控制总线以至少部分地基于时分复用时钟和数据协议来同步多个设备之间的通信，并且将总线初始化为第一值，以及能够通信地耦合到总线的第二设备，该第二设备被配置为在总线上发射与第一值不同的第二值。第一设备可以被配置为从总线读取第二值并且将至少部分地基于第二值向第二设备发射响应，并且该响应可以被配置为控制第二设备的延迟以校准在总线上的来自第二设备的通信。该系统可以被配置为包括对计算出的将与实际硬件值对齐的多个延迟值的限制。

[0016] 在一些实施例中，第一设备被配置为调整第二设备的延迟，使得由第一设备正在读取的值将在两个或更多个值之间波动。第一设备还可以被配置为多次读取由第二设备写入的值，并且执行对这些值的过滤；其中，过滤器被配置为减少总线上的当前读回值的估计的变化性。第一设备可以被配置为将总线设置为已知值，并且然后通过禁用输出驱动器而使总线未被驱动，其后第二设备能够改变总线的状态。

[0017] 总线可以包括电路，该电路包括总线保持器，该总线保持器被配置为消除关于能够导致未定义状态的泄漏电流的问题，其中，总线保持器在总线未被驱动时维持总线的当前状态。

[0018] 在一些实施例中，第一设备被配置为在多个连接的设备被配置为驱动总线之前将初始延迟值写入多个连接的设备中，其中，初始延迟值是系统上次运行时的先前延迟值，作为用于更新其他连接的设备的初始值。该系统可以进一步被配置为调整从总线读取的值的平均值以具有零和一之间的概率并且相应地调整延迟并且避免读回相同值。

[0019] 该系统可以被配置为包括用于在延迟的调整期间使用以获得正确延迟位置的估计的第一稳定模式和被配置为确定更精确的延迟位置以用于延迟的最终调整的第二稳定模式，与第一稳定模式相比，第二稳定模式在随后的校正期间具有更小的延迟步长。该系统还可以被配置为基于多个读回值来在第一稳定模式和第二稳定模式之间切换。

[0020] 该系统可以进一步被配置为包括具有多种自适应速度的自适应模式，其中，当读回的值与先前值相同时，自适应速度被递增直到某个上限，并且当读回的值与先前值不同时，自适应速度被递减到某个下限。

附图说明

[0021] 参考以下附图和随后的详细描述能够更好地理解本公开的各方面及其优点。应当理解，相同的附图标记用于标识一个或多个图中所图示的相同元件，其中，其中的示出是为了说明本发明的实施例，而不是为了限制本发明。附图中的组件不一定按比例绘制，而是将重点放在清楚地说明本公开的原理。

[0022] 图1是图示根据本公开的一个或多个实施例的连接到公共总线的主设备和两个从设备的框图，在设备内部和总线本身上具有延迟。

[0023] 图2是根据本公开的一个或多个实施例的校准后的从设备的眼图。

[0024] 图3A、图3B、图3C和图3D是图示根据本公开的一个或多个实施例的来自从设备的时序信息的时序图。

[0025] 图4图示了根据本公开的一个或多个实施例的在一行中具有十六个时隙的示例时分复用总线系统的时隙，包括用于时序信息的两个时隙。

[0026] 图5图示了根据本公开的一个或多个实施例的从设备以各种延迟(太小、校准和太大)写入到主设备。

[0027] 图6是示出根据本公开的一个或多个实施例的由于总线上的抖动和噪声而从从设备获得的个体样本的变化性的时序比较图。

[0028] 图7是图示根据本公开的一个或多个实施例的使用由主设备向总线发出的一个或多个命令的从设备延迟的示例编程的流程图。

[0029] 图8是图示根据本公开的一个或多个实施例的来自附接到具有嵌入式时钟信息的

时分复用总线的从设备的内部逻辑的框图。

[0030] 图9是图示根据本公开的一个或多个实施例的用于通过校正行或帧内的时序事件来调整一个或多个从设备的延迟的时钟恢复电路的框图。

[0031] 图10是图示根据本公开的一个或多个实施例的使用从主设备返回的延迟信息的模拟积分来调整从设备的内部延迟的电路的框图。

[0032] 图11是图示根据本公开的一个或多个实施例的被配置为使用数字控制算法来调整从设备电路的内部延迟的延迟电路和控制延迟元件的数模转换器的框图。

[0033] 图12是图示根据本公开的一个或多个实施例的被配置为使用线性搜索来控制内部主设备或从设备延迟的程序、过程和/或算法的流程图。

[0034] 图13是图示根据本公开的一个或多个实施例的被配置为使用具有两个步长的线性搜索方法来控制内部主设备或从设备延迟的程序、过程和/或算法的流程图。

[0035] 图14是图示根据本公开的一个或多个实施例的被配置为使用自适应两步线性搜索方法来控制内部主设备或从设备延迟的程序、过程和/或算法的流程图。

[0036] 图15是图示根据本公开的一个或多个实施例的被配置为使用自适应多步线性搜索方法来控制内部主设备或从设备延迟的程序、过程和/或算法的流程图。

[0037] 图16是根据本公开的一个或多个实施例的对图12的算法的修改，其包括在每个延迟调整之前的多个读取测量。

[0038] 图17是图示根据本公开的一个或多个实施例的实现调整过程的程序、过程和/或算法的流程图，其中主设备或从设备使用多个读写序列来自行调整其内部延迟。

[0039] 图18是图示根据本公开的一个或多个实施例的被配置为通过对交替事件执行两次读取来读取下降沿和上升沿两者的延迟的主设备或从设备的时序图。

[0040] 图19是图示根据本公开的一个或多个实施例的连接到公共总线的主设备和两个从设备的框图，在设备内部和总线本身上具有延迟。

[0041] 图20是图示根据本公开的一个或多个实施例的连接到公共总线的主设备和两个从设备的框图，在设备内部和总线本身上具有延迟。

[0042] 图21是图示根据本公开的一个或多个实施例的连接到公共总线的主设备和两个从设备的框图，在设备内部和总线本身上具有延迟。

[0043] 图22图示了根据本公开的一个或多个实施例的总线系统的示例校准时序，该总线系统包括连接到公共总线的主设备和两个从设备，在设备内部和总线本身上具有延迟。

[0044] 图23图示了根据本公开的一个或多个实施例的总线系统的示例校准时序，该总线系统包括连接到实现两个不同的时钟或同步速率的公共总线的主设备和两个从设备，在设备内部和总线本身上具有延迟。

[0045] 图24A是图示根据本公开的一个或多个实施例的连接到公共总线并包括写入时序调整模块的总线设备的框图，在设备内部和总线本身上具有延迟。

[0046] 图24B是图示根据本公开的一个或多个实施例的连接到公共总线并包括读取时序调整模块的总线设备的框图，在设备内部和总线本身上具有延迟。

具体实施方式

[0047] 根据本公开的各种实施例，描述了用于实现在连接到公共总线的多个设备之间的

同步的系统和方法,其中时钟和数据信息被组合。

[0048] 对于连接到总线系统的多个设备之间的同步,一个设备将发送出同步模式,该同步模式由连接到相同总线系统的其他设备接收并被用作时间参考。这些连接到总线的其他设备内部的内部延迟可能是未知的。本文中公开的实施例包括用于校准时间参考与个体设备之间的延迟的系统和方法。

[0049] 本文中描述了一种用于自校准的方法,该方法使得个体设备能够由其本身基于由这些设备写入到总线并由提供时间参考的设备复制的数据执行时间调整。在这种场境中,提供时间参考的设备将被称为主设备,而尝试校正它们自己的时序的设备将被称为从设备。在各种实施例中,执行校准的过程将包括以下内容:(i)主设备发送出时间参考(例如,同步模式),从设备能够锁定在该时间参考上;以及(ii)基于这些信息,这些从设备可以能够从主设备向从设备读取信息,因为来自主设备的同步模式和任何命令两者都将引发相同的延迟,并且因此能够可靠地被接收。

[0050] 然而,从从设备到主设备的延迟在此被测量之前是未知的,并且因此,为了促进快速总线操作并避免总线冲突,总线被配置为优化从从设备到主设备的数据传输时序。在一些实施例中,这个问题将通过使从设备将值写入到主设备来解决,并且该值将改变总线状态的当前值,并且主设备会将其已经读取的值写入到总线,以供从设备用作内部延迟的指示器。

[0051] 从设备被配置为基于已经从主设备读回的值来调整内部延迟。例如,已经读回的值接近零和一的某个概率,例如50%,从而指示该特定延迟值将导致从设备发射器在最佳时间打开,并且从该主设备可以能够在一个时间单位之后从主设备可靠地读取值,其中一个时间单位由时钟系统的时间参考定义。

[0052] 虽然本文中描述的方法将通过单端总线系统的示例示出,但是类似的方法可以被应用于差分总线,其中主设备和从设备两者都将包含差分发射器和接收器电路。

[0053] 现在将参考图1描述用于利用具有设备的嵌入式时钟延迟调整的时分复用通信执行附接到总线120的多个组件之间的总线通信的系统100的组件。系统100包括至少一个主设备102,该至少一个主设备可以包括用于外部或内部通信的端口104、总线控制单元106、连接到公共总线的发射器输出端口108和也连接到公共端口的接收器输入端口110。主设备可以包括时钟生成电路或外部时钟源(未示出),但这可以被放置在系统的其他部分中。

[0054] 总线系统可以包括一个或多个从设备,如从设备1 130。该设备将包括接收器输入端口132、时钟或时序恢复电路134、可选的输入时间调整电路136、输出发射器端口138、输出时序调整电路140、总线控制电路142和用于通信的各种端口144。除了从设备1之外,还示出了具有类似电路(例如,接收器输入端口152、时钟或时序恢复电路154、可选的输入时间调整电路156、输出发射器端口158、输出时序调整电路160、总线控制电路162和用于通信的各种端口164)的类似设备—从设备2 150,尽管这两个设备可能不相同,也不需要两个设备都附接到总线。在一些实施例中,除了从设备1和2之外,多个设备位于总线上。从设备可以使用例如DLL(延迟锁定环)或PLL(锁相环)用于与主设备同步信息同步。在一些情况下,这些子电路可以包括时间门,以移除总线上干扰PLL和DLL电路的操作的任何随机数据。在从设备已经获得对总线的锁定之后,可以首先使用该时间门(未示出)。

[0055] 从主设备到所附接的从设备的任何时钟和数据信号将引发相同的延迟。因此,从

设备可以使用时钟参考作为稳定时序来恢复任何数据。然而，主设备和从设备两者中的内部延迟都是未知的，并且存在诸如总线延迟的未知外部因素。因此，关于从从设备到主设备的方向上的数据到达将存在很大的不确定性。为了优化这个时序，将描述以低成本获得非常准确的延迟校准的程序。该程序将不需要向系统添加附加的特殊电路。通常被认为对系统操作有害的如抖动和噪声的因素将在这里用于提高系统的时序分辨率，从而使得主设备和从设备即使在信息的传输和接收中使用仅两个电压电平也能够获得准确的时序信息。与极低功率系统相关联的大抖动将导致转变点附近的接收采样值的变化性，但这种变化性将被建设性地用于改进总路径延迟的时序估计。可以调整主设备内部的数据的接收，但为了尽可能高的带宽，调整一个或多个所附接的从设备内部的实际传输事件可能更好。这样，来自从设备的任何数据都能够被紧密对齐，以在总线系统上获得尽可能高的带宽。

[0056] 所需的延迟调整的范围将取决于组件公差、总线直径和受特定技术考虑因素支配的内部延迟。作为示例，目前制造以下从设备接收器电路是可行的：与输入沿相比，输出稳定点的变化为+5纳秒(ns)(内部延迟)和+/-10ns，总变化-5ns至+15ns(静态延迟，不包括抖动)。

[0057] 主设备包括它自己的时序参考，并且在一些预期的实施方式中，该设备内的变化可以在-1到+5ns的范围内。对于0到200cm之间的总线直径(例如，相对较长的总线系统)，我们将得到0到12ns之间的延迟(假设传播速度较慢，约为16cm/ns)。这些因素的总变化导致-6到32ns之间的总时序变化。为了获得高速、低功率信令系统，期望限制时钟转变的次数，但这将增加从设备接收器电路中的抖动。

[0058] 出于成本和系统考虑，无论制造公差如何，都期望恰当地对齐从设备数据位置。为了尽可能低的功耗，接收器时钟恢复电路中的抖动将增加，因此，期望限制由于主设备和从设备数据模式的不良对齐而导致的带宽损失。对于大约0.3到1.5ns(实际值)的均方根抖动，在许多实施方式中，我们将具有+/-2.1到10.5ns的最坏情况变化(错误概率为10^{-12}的因数7)。

[0059] 对于被认为成功的校准，在一些实施例中，期望从设备数据在最佳值的1-2ns内对齐，因为这个因数是总误差预算的一小部分。该值可能非常小，以至于通过一次性工厂编程对其进行编程或控制是不现实的。因此，本文中描述的实施例动态地测量和控制设备内部的延迟以实现最佳时序控制。

[0060] 从设备延迟时序的控制能够在主设备中或在从设备中或在以另一种方式连接到总线或组件的另一设备中发生。主设备控制时序的优点是所有从设备都将不需要控制电路，因此单个电路能够控制所有设备以实现可能更简单的系统。然而，从设备也可以控制它自己的时序，前提是它知道主设备从它读取的值。

[0061] 能够解决这个问题的一种方式是通过使主设备写回它在校准阶段期间从从设备读取的值。与先前方法相比的优点是主设备只需要写回单个比特，这允许降低在校准阶段期间的带宽要求。从设备本身能够完全负责校准，并且将不需要依赖位于主设备内部的控制算法的任何特殊特性。替代地，主设备可以发信号通知从设备以调整其内部延迟(例如，不改变延迟、增加延迟或减少延迟)而不是向从设备提供绝对延迟控制值。

[0062] 图2示出了由在相对高的频率(例如，75MHz)下操作并且使用在+/-1ns内准确的延迟调整的从设备发射的数据的眼图。从图中能够看出，还为整个系统的噪声抖动和噪声留

下+/-4.5ns，并且数据仍然将是可读的。这相当于643皮秒(ps)的抖动均方根(rms)，错误概率为10^{-12}。在不进行调整并使用在-10至+30ns的范围内的需要的早前调整值的情况下，数据将跨眼图开口随机地散布。

[0063] 图3A至图3D示出了在从从设备到主设备的延迟信息的采样中涉及的事件的实施例。参看图3A，这示出了在测量系统延迟中涉及的事件。首先，主设备发送出低到高转变，以发信号通知同步事件的开始。然后主设备将使总线浮置，其后从设备能够改变总线的电平。最后，基于总线再次改变(在本实例中从高到低)之前所花费的时间指示系统延迟(即，从主设备到从设备再到主设备)。可选地并且图中未示出，主设备可以在相同帧或行中或在稍后的行或帧中发送出它已经从总线读取的值的副本，以防从设备本身要执行任何延迟调整，而没有来自主设备的任何其他交互。时序图从主设备的视角示出。总线保持器可以被包括在系统中，以在没有设备正在驱动总线时保持总线值恒定。总线保持器的输出阻抗将可能明显高于总线(传输线)的阻抗，并且明显高于个体输出驱动器的输出阻抗。总线保持器可以位于一个或多个组件中，或者对于具有小泄漏电流和高工作频率的系统可能根本不需要。

[0064] 参看图3B，在主设备已经使总线空闲之后总线改变电平的确切时间上存在一些不确定性。这种不确定性主要是由来自从设备时钟恢复电路的抖动引起的。虽然在确切的主设备时序上也将存在一点不确定性，但由于直接连接到参考时钟，所以这个时序将更加稳定。

[0065] 参看图3C，如果在从设备已经改变总线值之后主设备读取总线值明显更晚，则读取的值将始终是相同的。在图中所示的情况下，主设备将始终读取低值或零。这指示已经在从设备内部设置的延迟值太低，因为它占用的带宽可能多于正确操作实际需要的带宽。

[0066] 参看图3D，如果主设备读取总线值明显早于从设备改变总线值的时间，它将一直读取相同的值。在这种情况下，主设备将始终读取高电平或一。这指示延迟值已经被设置得太大，因为主设备无法在该时隙从从设备接收任何信息。

[0067] 图4示出了在每行中包含16个时隙的时分复用系统的实施例。每行还使用两个时隙以将同步信息从主设备传送到从设备。所有其他时隙可以被用于在主设备与从设备之间传输数据或允许改变数据传输的方向。主设备将在时隙中间和在时隙沿(即提早1/2时钟周期)从从设备读取数据以获得延迟信息。

[0068] 参看该图，最后一个时隙(15)被用于将总线预定义为低电平作为同步脉冲的开始。下一个时隙(数字0)被用于将总线设置为高电平，并且低到高转变之间的边界被用于从设备的时钟恢复电路，以获得主设备与从设备之间的正确同步。时隙1被用于设置总线空闲并允许总线方向从主设备改变为从设备写入总线。基于从设备中的内部延迟和从设备未知的外部延迟，它可能改变时隙1内部某处的输出电平。主设备能够在时隙1和时隙2之间的转变时读回总线电平。该读取将被用于延迟校正。最后，位于时隙2中间的总线值将被用作从从设备返回的值，以用于实际数据传输，因为这里总线有时间稳定下来，并且该方案允许最大程度地容忍抖动和噪声(即采样点位于“数据眼图”的中间)。时隙3至14在图中被示出为空闲，但可以涉及在设备之间传输实际数据，只要数据不会干扰延迟测量。

[0069] 图5更详细地示出了延迟测量的细节。在顶部的第一时序图示出了从设备，其中延迟校正已经被设置得太低，结果主设备将始终读回总线电平的改变，即使在时隙1和时隙2

之间的测试采样点处。在第二个时序图中，主设备将读取零和一的交替值(即有时它将看到总线电平的改变，有时它将不会)。由于抖动和噪声，正在读回的值将随机地交替。最后，在图的底部的最后一个时序图示出了从设备，其中延迟校正已经被设置得太长，以至于它对所需的改变响应得太晚。当主设备在时隙1和时隙2之间的转变时测量总线电平时，它将永远不会看到总线电平的改变。最后，将被用于从设备和主设备之间的通信的实际值将是在测试的值之后1/2时隙(即在时隙2的中间)采样的值。

[0070] 图6示出了由主设备对由从设备写入的值的测量将受噪声和时序抖动调制。然而，当多个样本被组合以进行更精确的调整时，这种变化性可以被用于获得两个以上的电平。由于从设备中需要非常低的功耗，因此时序抖动将相当大。然而，在本申请中，这将对我们有利，因为它允许我们形成多个测量值的平均值，并且它们将一起为我们提供有关转变区的信息，其中我们不仅将能够确定值是在转变之前还是之后，而且能够准确地使用平均值将延迟设置调整到最佳值，在许多实施例中导致平均返回值约为任一读取值的50％(50％低和50％高)。如果少于50％的返回总线值示出了没有改变，则意味着延迟应该更小以获得更好的信号完整性，并且如果超过50％示出了改变，则意味着延迟应该被设置为更小的值以优化总线带宽。主设备可以调整从设备本身内部的时序，或者可以简单地写回读取值并使从设备调整它们本身。如果主设备将调整它自己的接收器时序，则从设备将不需要任何调整，但是虽然这个方案更简单，但它将导致带宽的使用不是最佳的，因为从设备可能相互干扰，除非为变化分配额外的带宽。因此，建议调整发射器时序，而不是接收器时序。

[0071] 在图7中示出了根据一个或多个实施例的由主设备向总线系统发出以执行从设备延迟的读取和最佳校正的一个或多个命令的示例。向总线发出命令以如下方式进行。首先，主设备将发出同步信息达足够长的时间，以使附接到总线的所有设备都将通过它们的时钟接收器电路获得同步。现在他们能够从主设备读取命令和数据，但他们自己的发射器电路还不能针对内部和外部延迟进行恰当地校准。

[0072] 如算法700中所示的命令和程序将示出所涉及的全部步骤的示例。首先，在步骤702中，主设备将发出命令以选择对什么设备进行时间校正。在步骤704中，主设备将在该从设备中设置初始延迟(在一些实施例中可以跳过该步骤)。然后在步骤706中，主设备将总线初始化为已知值并使其未被驱动。注意，该事件可能与在每一行中发送的自然发生的同步脉冲相组合。(图4和图5中的时隙15和0)。在步骤708中，所选择的设备将基于从设备的内部时序通过改变当前总线值进行响应。在步骤710中，主设备随后将基于主设备自己的时序从总线读回当前值。在步骤712中，如果目的是使从设备本身调整内部时序，主设备可以可选地写回它已经从总线读取的值。注意，从设备能够可靠地从主设备读取数据，因为来自主设备的同步脉冲和命令被相等地延迟。在可选步骤714中，在进行任何时间校正之前对数据进行平均。在步骤716中，考虑值在该延迟位置处是否可靠，否则返回步骤706。在步骤718中，考虑延迟估计是否完成，如果是这样则改变延迟设置并以新的延迟值继续720，否则在步骤719中告诉从设备需要更多读取。如果延迟值是最佳值，则这应该被用于从设备722，并且当在步骤724中附接到总线的所有设备都已经被校准时，则算法完成。

[0073] 图8图示了用于基于DLL拓扑在从设备内部执行内部延迟校正的电路800的示例实施例。在该示例中，延迟单元被用于获得与主设备的锁定并且控制来自从设备的数据的发射。如果时钟恢复电路是基于DLL拓扑结构，将这个电路用于延迟校正可能是优势，因为这

两个电路(时钟恢复和延迟校正)可以被组合成单个电路。如图所示,电路800包括:输入接收缓冲器802,其用于接收总线数据;时间门804,一旦从设备已经被锁定到主设备,该时间门就被用于从同步脉冲中过滤掉总线流量;总线同步电路806,其确保与主设备的安全同步并控制时间门;延迟单元808(示出了八个,数量可以随应用而变化),其由总线同步电路控制并被用于将单行数据分成个体时隙;多路复用器810,其用于选择、延迟和控制输出时序;发射寄存器812,其用于包含下一个要由从设备发射的数据;以及发射输出缓冲器814,其要被连接到总线。基于测量的延迟值,主设备或从设备将控制多路复用器的设置,从而将时序调整到最佳值。该控制可以在系统启动后执行,或在持续的基础上执行,以校正环境改变(例如,可能影响总线系统中的内部和外部延迟的温度改变)。因此,通过持续监控和校正这些延迟,能够校正这些设备变化,并且系统能够以最佳方式持续工作。在所需时间调整超过由单个时隙支持的时间调整情况下,从设备可以被配置为在不同的时隙中发射脉冲以适应这些更大的延迟调整(如图所示的延迟调整支持单个时隙内的延迟调整)。

[0074] 图9示出了根据一个或多个实施例的用于基于数字校正单元在从设备内部执行内部延迟校正的示例系统900。系统900包括时钟恢复电路902,其可以包括已经被锁定到由主设备提供的同步模式的PLL、同步电路和时间门(未示出),其被提供以消除数据影响时钟恢复。该时间门可以在设备已经获得对原始同步模式的锁定以保证锁定之后首先被激活。它将包括时序计数器903和改变传输事件的时间的加法器(时间校正单元)904、以及用于检测数据的传输的开始的比较器906、用于从设备输出数据的寄存器908和实际发射输出缓冲器910。在一些情况下,实际时序事件可能被编程,而不是将延迟添加到数据的开始。这样,将不需要加法器904,而是将正确的数据的开始提供给比较器906。

[0075] 通过利用两个时钟沿进行时序调整,可以将时序精度提高几乎两倍。作为示例,通过具有250MHz的内部高速时钟,通过利用两个沿可以进行2ns的时序校正。因此,对于最佳延迟设置,误差最多为1ns(即+/-1.0ns)。在这种情况下,时序被限制为小于两倍的原因是由于实际系统中的时钟占空比不是完美的50%。

[0076] 图10示出了用于基于积分器1002、由来自积分器的输出控制的延迟单元1004和发射器输出缓冲器1006在从设备内部执行内部延迟校正的示例系统1000。这里,延迟校正通过对返回值求和并使用这些值的积分版本来控制延迟单元在持续基础上被执行。注意,由于内部泄漏和其他因素,该电路可能需要定期进行延迟测量,否则它可能会漂移出最佳位置。

[0077] 图11示出了用于基于DAC(数模转换器)1102、延迟单元1104和输出发射缓冲器1106在从设备内部执行内部延迟校正的示例电路1100。该电路将使用数字计算的最佳延迟值来控制DAC,并且来自DAC的模拟输出将控制延迟单元的延迟。这继而控制输出发射缓冲器。该电路相对于系统1000的优势在于漂移相对较小,并且内部泄漏电流将不会导致其明显移动远离最佳延迟位置。缺点是需要DAC,尽管它不需要高精度,只要它是单调的并且具有足够的步数来控制延迟单元。因此,在一些应用中,低至6比特分辨率就可以足够控制延迟。对于遇到明显温度漂移的系统,可能需要持续运行校准算法。

[0078] 在图12至图15中分别示出了用于获得主设备和从设备之间的受控延迟的四个方法。还有其他方法可以被用于控制这些设备,例如分半搜索或二分搜索。然而,这里描述的方法具有以下优势:我们能够以简单的方式确保所采取的步长足够小,使得我们不会冒从

设备将开始覆写由主设备或附接到总线的其他设备提供的任何信息的风险。此外，这里描述的算法通过选择适当的参数对影响延迟的噪声具有很强的噪声鲁棒性。

[0079] 图12示出了第一算法的流程图的示例实施例。该方法1200包括线性搜索算法。对于每个周期，主设备将设置总线，并且从设备将通过改变总线值来进行响应。基于由主设备读取的值，从设备内部的延迟将被增加或减少少量时间。换句话说，如果从设备写入总线太晚，则在执行对主设备的写入之后，从设备内部的延迟将减少少量，并且类似地，如果从设备写入太早(例如，当看到主设备的所有读取值的一半以上改变时)，延迟值将被增加。该方法将通过以小步长(例如0.05ns)改变延迟来确保延迟的抖动被很好的控制，但这也意味着，对于大的延迟改变(例如20ns)，该算法在达到理想延迟之前可能经过数百步(迭代)。因此，该方法是快速稳定时间和良好收敛之间的折衷，并且对于给定的低抖动要求，这将导致较长的稳定时间。该方法具体如下描述地工作。

[0080] 该算法开始于步骤1202，这是调整程序的入口点。然后主设备将在步骤1204处将总线设置为已知值，并且在步骤1206中从设备将改变总线上的值。在主设备已经从总线读取当前值之后，它可以可选地写回它在步骤1209中从总线读取的值，并且从设备可以使用该值进行自我调整。如果主设备正在控制从设备延迟，则可能不需要该步骤。注意，主设备基于它自己的时序在步骤1204和步骤1208中操作，并且类似地，从设备将基于从设备内部时序来执行步骤1206。通常，这些时序参考将不同，并且因此可能需要时序上的校正。基于从总线返回的值(在步骤1208中读取)，在步骤1210中做出判定。如果该值已经改变，我们知道—从统计学上讲，因为返回值受抖动和噪声严重影响—延迟太大。因此，如果是这种情况，当前延迟将稍微向上调整，例如+0.05ns。否则，即没有检测到总线值的改变，延迟太大(再次在统计学意义上)，并且我们将在相反的方向上进行小的校正，例如-0.05ns。这些值只是示例，并且可以使用许多其他值。如果我们选择更大的值，寻找最佳延迟的搜索时间将更快，但由于随机变化导致的最终输出抖动将更大，所以这是一种折衷。在步骤1216中，我们将确保输出延迟在系统限制内，即我们不会将值增加到超出我们能够校正的最大值，也不会减少到超出电路能够处理的最低值。然后在步骤1218中，积分延迟值从内部寄存器映射到实际延迟设置(例如，实际延迟设置的数量可能小于用于计算该值的精度)，并且在步骤1220中，实际从设备时序被校正。最后，在步骤1220中确定算法是否完成。如果算法未完成，则其从步骤1204继续。可以使用算法能够进行的已知最大迭代次数或通过使用该次数的返回的零和一的平均值来确定算法是否完成。一旦在多次迭代(例如100次)上的平均值达到0.50，我们就达到了最佳延迟时序校正。为了允许一定的容差，一旦平均值达到某个范围(例如0.47-0.53)内的值，算法就可以退出。在另一实施例中，主设备可以在步骤1209中调整从设备延迟(例如，不改变延迟、增加延迟或减少延迟)。

[0081] 图13示出了第二算法的流程图1300。该方法是一种线性搜索算法，其已经通过包括两种自适应速度得到改进。最初，该方法将进行较大的延迟调整(例如1-2ns)，并且在已经找到第一转变之后(通过由主设备从从设备测量与主设备从从设备采样的先前值不同的值)，它将改变为较慢的跟踪模式，其中时间步长约为0.05ns，以确保抖动被控制。这将相当大地改进简单方法一的稳定时间，但仅单次噪声测量就将使方法改变为慢速模式并且然后自适应速度将会更慢。与方法一相比，该方法的抖动稍高。该方法具体如下描述地工作。

[0082] 算法开始于步骤1302，并且在步骤1304中将标志(第一转变)设置为零。该标志被

用于查找在从设备响应之后返回的读取值与先前读取的总线值不同的点。在步骤1306中，主设备将总线设置为已知值，然后释放总线，而总线持有者可以维持该值。在步骤1308中，从设备将基于其内部时序来改变总线，并且在步骤1310中，主设备将基于主设备内部时序来读取总线。在步骤1312中，主设备可以可选地写回它已经从总线读取的值，使得从设备能够自校准它自己的时序。如果主设备直接设置从设备的时序值，则不需要该可选步骤；如果从设备将本身控制时序，则可以包括该可选步骤。在另一实施例中，主设备还可以通过发出对当前延迟值的改变(例如，不改变、增加延迟、减少延迟)在步骤1312中控制从设备时序。

[0083] 在步骤1314中确定由主设备读回的总线值是否与由主设备读回的先前值相同(注意，在运行该算法的第一序列中，这两个值应假定为相同，因为没有信息可用并且标志“第一转变”将保持为零)。如果两个值相同，标志“第一转变”将保持为零，但如果两个值不同，现在将在步骤1316中将标志设置为一或真。该标志被用于最初执行快速搜索，并且一旦已经发现转变，就将执行更慢且更准确的搜索。在步骤1318中，基于“第一转变”标志的值做出判定。如果没有值的改变，我们将继续快速搜索1322，但是一旦已经发现单个转变，我们将继续进行如步骤1320中定义的更慢且更准确的延迟校正。在快进模式下的延迟值的值可以是例如0.75ns，但将取决于应用，并且慢速且更精确的步进的值可以是例如0.125ns，这取决于应用。这些值能够容易地用二进制算术表示。在步骤1324至1336中执行的过程可以与在方法一中描述的步骤1210至1222相同，并且将不再重复。该算法的优点是搜索时间更快，同时维持与算法一相同的抖动性能。

[0084] 第三算法在图14中示出并且是第二算法的演变，其中算法能够基于先前的N个值一直在慢速和快速模式之间改变。如果所有N个先前的读取值都相同(即我们在相同方向进行适应)，则将采取两个时间步长中的较大者(ΔT_{LARGE})。如果读取的N个最后值之间存在任何差异，则将采取较小的时间步长(ΔT_{SMALL})。与方法二相比，该方法对噪声和抖动的敏感度要低得多，并且与方法二相比，以略微更大的抖动实现了更快的稳定时间。该方法具体如下描述地工作。

[0085] 算法1400在步骤1402中开始，并且然后主设备在步骤1404中将总线设置为已知值并释放总线。当没有设备正在使用中等输出阻抗(例如，在1至2kΩ的范围内)驱动总线时，总线保持器可以存在并且保持总线状态。从设备在步骤1406中改变总线值并且主设备将在步骤1408中基于它自己的时序来读回总线值。在步骤1410中，主设备可以可选地将读取的值写回到总线，以供从设备用于自我调整。如果主设备正在控制从设备的延迟，则不需要该步骤，并且主设备可以通过发出确切的延迟值或通过控制当前值(例如，不改变、增加延迟、减少延迟)来控制从设备的延迟。在步骤1412中，基于由主设备读取的过去N个样本做出判定。如果最后N个样本中没有改变，我们将采取过程1414，其中使用了较大的自适应值。快速自适应值(过程1414)的值可以是0.75ns，而慢速自适应值(过程1416)的值可以是0.125ns。在各种实施例中，N的值可以是8以确保即使在有明显噪声的情况下也不存在高自适应值和低自适应值之间的振荡。步骤1418-1432与步骤1210-1224相同，因此将不再重复描述。该算法即使在存在明显噪声的情况下也能很好地工作，虽然可能有几个步骤进行得太慢，但通常它将使用仅两个步长值来尽可能快地适应。

[0086] 第四算法在图15中示出并且与图14中所示的第三算法相比是进一步的演变。这里，每当主设备在总线上采样相同值两次时，我们将增加时间延迟改变，否则延迟将被减

少。在某个点处将达到平衡，取决于更新系数给出两个可能的返回值(低或高，零和一)的接近50％的平均读取值。换句话说，对于值的改变，下一延迟调整的长度将是如下：$\Delta T=\alpha_1 x \Delta T$，其中$\alpha_1$将通常在1.05到1.2的范围内。选择容易在二进制算术中没有舍入误差的表示的数字可能是优势，例如1+1/8＝1.125，其能够使用加法而不是乘法被实现为系数。

[0087] 如果没有值的改变，下一延迟调整的长度将是如下：$\Delta T=\alpha_2 x \Delta T$，其中$\alpha_2$将通常在0.8到0.95的范围内。选择容易在二进制算术中没有舍入误差的表示的数字可能是优势，例如1-1/8＝0.875，其能够使用加法而不是乘法被实现为系数。由于自适应步长，该方法工作得最快，并且具有与方法一几乎相同的抖动性能(稳定时间的略微增加可能进一步降低抖动)。为了保证算法的稳定性，延迟改变的最小值和最大值存在界限，因此延迟中的最小和最大步长被实现。该方法具体如下描述地工作。

[0088] 在步骤1502中，算法1500开始。在步骤1504中，主设备将总线设置为已知值，并且在步骤1506中，从设备基于它自己的时序来改变总线值。在步骤1508中，主设备基于主设备的时序来读取总线上的值。在可选的步骤1510中，主设备写回它已经读取的值。仅当从设备要自行调整其内部时序时才需要该步骤。如果主设备正在控制从设备的延迟，则不需要该步骤。在这种情况下，主设备将通过写入绝对延迟值或通过发出对当前延迟值的改变(例如，不改变、增加延迟、减少延迟)来控制从设备的时序。在步骤1512中，做出由主设备从总线读取的前两个值是否相同的判定。在该算法的第一次迭代中，将在步骤1514中使用默认的初始延迟值，例如0.75ns。如果两个返回值相同，则这指示我们处于延迟调整的范围内，其中几乎没有噪声。这意味着我们应该尝试更快地移动。这是通过将延迟调整值乘以大于一的因数(例如1.125)来完成的。然而，如果这两个值不同，则使用较小的步长(步骤1516)，因为这指示我们处于转变的中间的有噪声区域中，其中我们通常将在零和一之间波动。延迟调整校正因数现在将小于一(例如0.875)。从这一点开始，由于算法精度有限以及避免覆写来自其他设备的数据，我们现在将在步骤1518中将时间调整限制在某个范围(例如1/128至1.0ns)内。步骤1520-1534可以与步骤1210-1224相同，因此将不再重复描述。

[0089] 最后，返回值的平均值可以被用作算法何时已经完成的指示符，例如通过形成加权平均值，例如平均值＝0.99*平均值+0.01x(最后一个读取值)。在一些应用中，可以基于有关噪声和抖动水平的已知信息以及已知的稳定特性来代替地使用固定和有限迭代次数，例如，可以认为该算法在40次迭代之后完成。

[0090] 这些方法中的每一个都能够通过形成在总线上采样的两个或更多个值的平均值来补充，这可能是特别有效的，如果主设备正在控制从设备，因为在这种情况下，在每次写入操作之前执行多个读取操作可能是有益的，因为写入操作将比读取单个比特花费更长的时间。

[0091] 延迟的初始猜测被用于实际实施方式。由于存在整个系统将必须收敛在其内的已知范围，例如从-10到+30ns延迟，选择等于该范围的中点的初始值可能是合理的。然而，为了避免覆写关于其他设备输出的问题，这些问题包括覆写主设备输出，略微增加开始点可能是有利的，例如在上面的示例中使用延迟＝15ns的初始值。这可以意味着与最大正延迟设置相比，收敛到最大负延迟设置需要略微更长的时间(即收敛到-10ns将花费比+30ns更长的时间)。在最保守的设计方法中，将选择最大延迟作为初始值，以最大程度地防止覆写来自其他设备的早前值。

[0092] 在图16中示出了在执行延迟调整之前将算法一扩展为使用多个读取样本的示例。除了在执行调整之前对多个样本进行平均之外，算法1600可以类似于图12的第一算法。例如，步骤1606、1608、1610、1612和1614分别对应于步骤1204、1206、1208、1209和1210。一些改变包括将值初始化为零(步骤1604)，并且在步骤1616和1618中基于过去从总线读取的值形成总和，并且基于在步骤1622中做出的判定重复从总线读取K次并形成步骤1624中的最终延迟校正。步骤1620、1626、1628和1630可以分别对应于图12的步骤1216、1218、1220和1222。

[0093] 第二、第三和第四算法可以类似地被扩展到在执行延迟调整之前使用多个样本。然而，当控制算法位于从设备内部时，动态地改变每个读和写操作之间的延迟值可能更好，因为从设备本身能够基于从总线读取单个比特来改变设备内部的多个比特。因此，更新操作将需要更少带宽，并且因此可能更高效。在另一实施例中，代替更新从设备中的绝对延迟值，主设备将发出对当前值的改变以减少调整所需的带宽。

[0094] 除了这些方法之外，例如，使用分半搜索和二分搜索的方法可以被使用。然而，由于调整延迟的问题如此负面以至于从设备输出数据开始与主设备值和相关联的未知值发生冲突，建议这些方法要在对要包括的最大步长有某种限制的情况下使用，以避免这种情形。

[0095] 通过使用计算机模拟，我们能够更精确地比较四个算法。仿真基于算法的以下特定设置：一般抖动水平为1.2ns(均方根)，高斯噪声。对于实际电路，这通常在从设备内部接收和控制之后的0.3-1.5ns范围内。

[0096] 第一算法(图12)：步长被设置为0.05ns，并且延迟调整精度被设置为0.1ns。

[0097] 第二算法(图13)：步长被设置为0.125ns和0.75ns。延迟调整精度被设置为0.1ns。初始步长值是较大的步长(0.75ns)。

[0098] 第三算法(图14)：步长被设置为0.125ns和0.75ns。延迟调整精度被设置为0.1ns。如果从主设备读回的电平在N＝8个连续测量中未改变，则步长被设置为0.75ns。如果在已经完成8次相同值测量之前有任何改变，则该值将被改变回较低值(0.05ns)。初始步长值是较大的步长。

[0099] 第四算法(图15)：延迟大小被设置为在1/128ns和1ns之间。延迟调整精度被设置为0.1ns或2.0ns，参见表1。初始延迟步长为0.75ns。如果先前的延迟值与主设备的当前读取值相同，则延迟步长改变将改变1.125倍。如果在由主设备读取的当前值与先前值之间存在差异，则延迟步长乘以0.875。请注意，1.125和0.875的乘法能够使用二进制算术被实现为(1+1/8)和(1-1/8)的简单加法。实际延迟是基于当前读取值更新的，如果该值较低，则最后一个延迟值被递增延迟步长值。

[0100] 如果当前读取值较高，则实际延迟被递减当前延迟步长值。

[0101]

表 1

参数	单位	线性搜索	两步初始值	两步自适应	多步自适应
延迟抖动*（ns）/0.1dt	ns	0.91	1.35	1.56	0.94
延迟抖动*（ns）/2.0dt	ns	2.21	3.46	3.53	2.03
锁定（T=-10ns）	迭代	490.3	37.6	38.9	25.7
Std dev 锁定	迭代	3.5	6.2	3.2	1.0
锁定（T=10ns）	迭代	87.5	7.7	11.9	5.0
Std dev 锁定	迭代	4.4	3.9	3.2	0.0
锁定（T=30ns）	迭代	291.7	115.9	27.3	17.3
Std dev 锁定	迭代	3.9	1.9	3.2	0.8
平均值	一的数量	0.48	0.48	0.49	0.45

[0102] 在该实施例中，错误概率是10-12并且锁定搜索时间和基于锁定时间的变化是基于12次模拟。0.1dt的延迟抖动意味着延迟调整的输出分辨率为100ps，并且2.0dt意味着延迟调整的输出分辨率为2.0ns。

[0103] 图17示出了根据一个或多个实施例的用于调整从设备的延迟的示例算法或总线命令。算法1700开始于步骤1702，其中主设备选择应该响应的设备。在步骤1704中，主设备可以在从设备中设置初始延迟值。请注意，该步骤是可选的。从设备可能已经适应了良好的延迟值，并且算法可能仅针对较小的调整而运行。此外，从设备可以具有将不需要任何调整的初始内部值。同样，在这种情况下，主设备不需要在从设备内部设置初始值。

[0104] 在步骤1706中，主设备将基于主设备的内部时序将总线设置为已知值。通过在从设备改变总线上的值之前改变主设备在步骤1706中向总线发送出的已知值(例如，在低或高之间交替)，调整后的延迟可以是基于来自主设备和从设备两者的下降和上升数据沿两者，从而形成更准确的平均值。

[0105] 在步骤1708中，从设备将基于它自己的内部时序通过将不同的值写回到总线来进行响应。在步骤1710中，主设备将基于它自己的时序读回总线上的值。通常，该读取事件将在来自从设备的数据应该已经完全稳定之前的1/2时隙发生。在步骤1712中，主设备将读取值写回到总线。这可能发生在与读取事件相同的行中(例如，在具有如图4至图5所示的时序的时隙3至14中)或在后续行中。替代地，主设备将控制绝对延迟或发出对从设备中的当前延迟的改变。

[0106] 在步骤1714中，从设备将基于由主设备写回的值来调整它自己的内部延迟。延迟的适应和调整可以如四个算法(分别如图12至图15所示)中所示的执行。

[0107] 在步骤1716中，做出调整程序是否完成的确定。这可以基于有限且已知的迭代次数(例如，每个延迟调整命令可以使用32次延迟调整)或基于延迟的收敛(基于接收到的逻辑零和一的近似相等的次数，读回值达到约0.50的平均值，或者位置没有改变达特定时间)。如果步骤的数量不是有限的，则主设备或从设备将需要提供关于告知算法何时完成的

反馈，否则从步骤1706继续调整。最后，在步骤1718中做出是否已经恰当地调整了所有设备的确定。如果不是这种情况，我们将继续步骤1702。算法何时完成的确定能够基于：当前步长、所使用的迭代次数或者平均读取值何时在给定区间(例如0.4至0.6)内以及已经达到一定的迭代次数或者如果调整后的延迟值对于给定的迭代次数没有改变。

[0108] 在其他实施例中，主设备将在步骤1712中通过发出对当前延迟值的改变(例如，不改变、增加或减少)来调整从设备的内部时序。

[0109] 在一些实施例中，两个从设备可以以与主设备和从设备之间类似的方式调整这两个设备之间的延迟。在这种情况下，通过反射读取值或调整其他从设备中的实际延迟，将读取当前总线值的改变并且调整发射从设备的时序的将是接收从设备。

[0110] 在一些实施例中，从设备可以在没有来自任何其他设备的任何交互的情况下调整它自己的时序，在这种情况下，时序可以与接收到的同步模式紧密对齐，但将不包括对总线直径和系统中的其他延迟的调整。

[0111] 图18示出了根据一个或多个实施例的主设备读取两个从设备值的示例。这里，从设备将以下降和上升转变两者进行响应。如果延迟的测量是基于从设备内部的未知或未校准的时间步长，这可以被用于通过寻找两个设置之间的数字或延迟数来找出给定时间单位需要多少时间步长。通过将该值除以二，我们能够找到等于半个时钟周期的延迟单元数。此外，这种最终方法可以允许主设备获得上升沿和下降沿延迟两者的测量值。

[0112] 如本文中所描述的，当从设备改变总线值并且主设备读取并且然后反射该值使得从设备能够调整其总线写入时序以获得最佳接收可靠性时，系统时序拓扑实质上是表2中的情况1并且如由图19中的时间线1902所标识。图19图示了根据本公开的一个或多个实施例的连接到公共总线的主设备和两个从设备，在设备内部和总线本身上具有延迟，并且类似于图1的系统100。这是标准从设备到主设备校准的情况。

[0113] 如果另一从设备从设备2反射了它已经从由从设备1对总线状态的改变中读取的值，那么该从设备反射的值能够被用于优化从从设备1到从设备2的写入时序，其采用本文中描述的各种时序调整过程。这种系统时序拓扑实质上是表2中的情况2，并且如由图20中的时序线2002所标识。

[0114] 如果从设备使其写入时间适应于它自己的输出，则从设备校正从其内部时序到外部观察到的输出的固有延迟，并且因此其写入与对应的主设备同步沿对齐。这是因为其内部读取逻辑将观察到主设备同步信号和自生成的写入值或信号两者的相同延迟。因此，在这种情况下，从设备的时序将非常准确地与接收到的主设备时序对齐，而不需要昂贵的调谐/编程。这种系统时序拓扑实质上是表2中的情况3，并且如由图21中的时序线2102所标识。

[0115] 在情况3中，从设备1写入总线，同时从总线读回。该过程被用于调整从设备1的写入时序，使得其输出数据与由主设备提供的同步模式对齐。该过程可以被用于校正总线设备之间的任何内部时序差异，并且确保输出与由主设备提供的时序沿(如由内部时序划分提供的真实或虚拟)完美对齐。如果多个设备靠近从设备1定位，则可以同时写入所有这种设备，而无需对设备进行任何校正，因为沿将与主设备时序对齐，并且其他设备将接收具有相同时间参考的数据。表2中的情况4与情况3类似，但侧重于主设备。如果主设备内部的接收时序具有固定时序，则能够通过执行类似于情况2的程序来校正发射和接收时序的差异

以获得完美的时序，例如用于自测试和冲突测试。在各种实施例中，情况4可以在总线的启动时执行，甚至在从设备附接到总线之前，以便节省时间/减少完成完整系统时序校准所需的时间。主设备可以使用初始0-1同步脉冲来调整内部采样点以用于从总线的读取。通过调整写入时序并且保持读取点固定，可以将外部0-1事件与主设备的读取点对齐。这样，可以消除微小的时序差异，并且因此更高的总线操作速度是可实现的。例如，来自主设备的同步和数据两者都可以是时移的。

[0116] 图24A是根据本公开的一个或多个实施例的图示连接到总线120并包括写入时序调整模块或逻辑2406A的总线设备(例如，主设备102)的框图，其具有与发射器/输出端口108相关联的固有写入延迟 ΔT_1 和与总线设备内部的接收器/输入端口110相关联的固有读取延迟 ΔT_2。在图24A中，主设备102可以被配置为使用写入时序调整模块2406A来调整主设备102的写入延迟以补偿固有读取延迟 ΔT_2 并且相对于总线通信时序恰当地对齐主设备写入事件2406B和主设备读取事件2406C，如关于情况4所描述的。在各种实施例中，写入时序调整模块2406A、主设备写入事件2406B和/或主设备读取事件2406C中的一个或多个可以由如图1所示的总线控制单元106实现和/或在总线控制单元106内实现。类似的时序调整模块可以在本文中描述的任何从设备内实现。

[0117] 各种写入时序调整情况总结于下表1中。

[0118]

表 2

情况	用于写入的设备	用于反射的设备	过程	结果	使用情况
1	从设备 1	主设备	调整从设备 1 写入时序	当写入到主设备时从设备 1 时序的完美对齐	优化从设备与主设备的通信
2	从设备 1	从设备 2	调整从设备 1 写入时序	当写入到主设备时从设备 1 时序的完美对齐	优化从设备与从设备的通信
3	从设备 1	从设备 1（内部）	调整从设备 1 写入时序	从设备 1 将具有与主设备同步沿精确对齐的输出	优化从设备与多个从设备的通信（如果未调整读取时序，则紧密接近）
4	主设备	主设备（内部）	调整主设备写入时序	校正主设备中的接收延迟	冲突测试，主设备自测试，多设备写入

[0119] 替代地，或者除了写入时序调整之外，可以执行读取时序调整。对于读取时序情况，接收事件在从设备或主设备内部发生的确切时间将通过校正对应的接收时序来校正。这些情况将采用类似于本文中描述的各种过程的过程，但重要的是要确保时序的改变以正确的方式发生。

[0120] 例如，可以修改写入时序调整算法，使得如果最后一个值示出了改变(即恰当读取)，则读取时间将被减少(例如，以达到存在零和一的大约50％几率的点)，并且如果没有总线值的改变，读取时间将被增加。换句话说，在通常写入位置将被增加的情况下，如果改变一直被读取，那么如果改为读取位置被调整，则读取延迟将被减少(即相反的调整)。

[0121] 在表3的情况5中，主设备的读取时间基于由从设备1提供的值来调整。因为确切的读取事件的内部延迟是在主设备内部调整的，所以没有必要将读取值反射回到总线，因为主设备已经具有读取延迟调整所需的信息。在这方面，与写入时序调整相比，该读取时序调

整将只需要一半的带宽。因此，虽然早前的描述使用两行用于每个调整步骤(从设备写入总线，主设备在下一行写回所写入的值)，但可以在单个行中执行该读取调整。

[0122] 在一些实施例中，表3的情况5和情况6可以被组合成单个读取调整，其中所有设备同时调整它们的读取延迟时序。这是可能的，因为从从设备1读取的附接的设备能够基于它们自己先前的读取值来调整它们的读取时序。因此，多个设备能够同时调整它们的时序，而不增加完成程序的时间。唯一的实际限制是总线上最慢的设备将确定总的调整时间。对于情况6，从设备从设备2的读取时序被调整，而总线被从设备1改变。因为从设备2从总线读取值并使用该值来调整其读取时序(即尝试实现大约50％的零和一)，所以从设备2没有必要写回从总线读取的值。图22图示了读取时序校准和写入时序校准之间的差异，其中在顶部行中，每个设备都能够在相同数据行内写入和读取以校准读取时序，并且其中在底部行中，每个设备都需要反射以便校准写入时序。

[0123] 对于情况7，主设备将调整它自己的接收时序，使得其与它自己的发射时序完美对齐。这可以被用于校正主设备内部的延迟以及使主设备被用作组写入的一部分，其中该组的所有成员都已经将其内部时序调整为来自主设备的输出。这样，主设备将不需要时钟恢复电路来获得完美的接收时序。在各种实施例中，情况7可以在总线的启动时执行，甚至在从设备附接到总线之前，以便节省/减少完成完整系统时序校准所需的时间。主设备可以使用初始0-1同步脉冲来调整内部采样点以用于从总线读取。通过调整读取时序并且保持写入时间固定，可以将外部0-1事件与主设备的读取点对齐。这样，可以消除微小的时序差异，并且因此更高的总线操作速度是可实现的。

[0124] 图24B是图示连接到总线120并且包括读取时序调整模块或逻辑2406D的总线设备(例如，主设备102)的框图，其具有与总线设备内部的发射器/输出端口108相关联的固有写入延迟ΔT_1和与接收器/输入端口110相关联的固有读取延迟ΔT_2，类似于图24A中关于情况4描述的固有延迟。在图24B中，主设备102可以被配置为使用读取时序调整模块2406A来调整主设备102的读取延迟以补偿固有写入延迟ΔT_2并且相对于总线通信时序恰当地对齐主设备写入事件2406B和主设备读取事件2406C，如关于情况7所描述的。在各种实施例中，读取时序调整模块2406D、主设备写入事件2406B和/或主设备读取事件2406C中的一个或多个可以由如图1所示的总线控制单元106实现和/或在总线控制单元106内实现。类似的时序调整模块可以在本文中描述的任何从设备内实现。

[0125] 对于情况8，从设备调整它自己的接收时序以与主设备同步脉冲对齐。这对于系统时序通常是不必要的或不相关的，因为同步和数据已经在该方向上对齐(同步和数据被相等地延迟)。

[0126]

表 3

情况	用于写入的设备	用于调整的设备	过程	结果
5	从设备 1	主设备，内部调整到读取值	调整主设备读取时序	当主设备从从设备 1 读取时的完美对齐
6	从设备 1	从设备 2，内部调整到读取值	调整从设备 2 读取时序	当从设备 2 从从设备 1 读取时的完美对齐
7	主设备	主设备，内部调整到读取值	调整主设备读取时序	当主设备读取它自己的输出时的完美对齐
8	主设备	从设备，内部调整到读取值	调整从设备读取时序	不相关，我们已经具有准确的时间参考（零偏移）

[0127] 一般系统时序调整可以如下进行。对于单个从设备1写入到多个设备(2-N)的一般情况，首先，优化从设备1写入时序以与主设备时序对齐(情况3)以节省带宽，并且其次，优化设备2-N的读取时序(情况5-6)。对于简单配置，情况1-2将覆盖所有场景并提供完美对齐。在线缆延迟较长的情况下，可能有必要写入到稍后的时隙以避免传出和返回信号之间的冲突。作为替代，当同时与多个设备通信时可以使用更长的时隙。在反射值是内部值的情况下，不需要在命令序列中插入额外的行。这将使自适应过程更快。

[0128] 在附加的实施例中，一般系统时序调整可以包括根据各种不同的操作顺序来执行情况1、6和7。例如，在一个实施例中，系统100可以执行情况7，其中主设备将调整它自己的接收或读取时序以便与它自己的发射时序完美对齐。在一些实施例中，这可以与系统100执行情况1基本上同时或在系统100执行情况1之前执行，其中从设备改变总线值并且主设备读取并且然后反射该值，使得从设备能够调整它的总线写入或发射时序，如本文中所描述的。在各种实施例中，系统100然后可以执行情况6和/或情况8以调整所有其他从设备读取时序。

[0129] 在一些进一步的实施例中，系统100可以执行情况2和/或3的实施例以调整所有从设备写入和读取时序(例如，发射和接收时序)而根本不需要任何主设备读取/写入时序调整，或以最小化主设备读取/写入时序调整(例如，以最小化当新的从设备在主设备和多个从设备已经稳定在相应的读取/写入时序上之后试图参与总线通信时系统100的动态重新校准所需的时间)。

[0130] 更一般地，情况1至8中的任何一个或多个可以由系统100的实施例以任何顺序或组合来执行以提供动态写入和读取校准以用于由系统100的总线120支持的和/或通过系统100的总线120支持的总线连接设备之间的总线通信，如本文中所描述的。在各种实施例中，这样的时序调整可以包括适当时隙的标识和选择(例如，如图4所示)以促进通过总线120通信耦合的每个设备(例如，主设备或从设备)的可实现范围内的时序校准。此外，这种时序调整可以使用本文中所标识的过程和/或算法中的任一者来执行，其可以被修改以调整读取时序和/或写入时序，如本文中所描述的。

[0131] 在一些实施例中，可以期望总线的时钟速率是可调节的。只要用于发射和接收的电路的带宽以及由此内部延迟保持不变，就不需要新的时序校准。一般原理是在校准期间

在一个时钟沿上进行调整，并且在下一时钟沿上读取以供正常操作，以确保完美的接收眼图模式。如果在同步/时钟速率的改变期间内部延迟略微改变，则校准准确度将被降低，并且，在一些情况下，如果内部延迟的改变太大，则可能需要重新校准。否则，可以使用相同的校准值。例如，图23图示了根据本公开的一个或多个实施例的总线系统的示例校准时序，该总线系统包括连接到实现两个不同的时钟或同步速率的公共总线的主设备和两个从设备，在设备内部和总线本身上具有延迟。

[0132] 可以使总线以原生脉冲密度调制(PDM)模式启动。这将确保与PDM的向后兼容性。在一些实施例中，可以包括差分电压检测算法，该算法被配置为检测差分数据并且从单端标准PDM模式(除了时钟频率的改变之外没有允许的直接编程)改变为差分模式。类似地，可以使接口以差分模式启动，并且当检测到单端标准PDM信号时，接口变为单端模式。当信号被保持在某个电平达某个时间量或通过时钟或数据线的占空比调制时，这种改变也可能随着使用在这两种操作模式之间改变的某个时间常数而发生。

[0133] 在特定实施例中，系统可以包括第三单端模式，其包括编程，诸如可用于受控PDM(PDM+)系统。三种模式(向后兼容PDM、可编程PDM和差分模式)之间的改变可以通过时钟和数据线的编程发生。在各种实施例中，差分操作模式可以被用于低EMI，并且向后兼容的PDM模式可以被用于确保存在用于通用系统和/或总线连接设备的备份模式。

[0134] 在一些实施例中，系统100可以被配置为使用特殊的同步字来执行时序校准。当使用这种特殊的同步字时，需要特别小心，以便发射的命令不干扰选取的(多个)同步字。因此，在校准命令的执行期间，以下响应是可用的：(1)测试值—未知值且不可控；(2)反射值—未知值但可控(反射读取的内容)；(3)可选的可控的已知值—该值可以被插入到读取序列中，以避免生成特殊的同步值。

[0135] 通常，应当注意确保已知值(2)和可控值(3)在一起使用时将不会生成同步字。因为值(3)是受控的，所以只要同步字的长度至少长于两个比特，就可以始终保证不生成同步字。对于由设备从本身的特殊读取或读取延迟调整，通常将不会发送出值(2)，其应当在控制值(3)时被考虑。

[0136] 在适用的情况下，可以使用硬件、软件或硬件和软件的组合来实现由本公开提供的各种实施例。此外，在适用的情况下，在不脱离本公开的范围的情况下，可以将本文中阐述的各种硬件组件和/或逻辑组件组合成包括软件、硬件和/或两者的复合组件。在适用的情况下，在不脱离本公开的范围的情况下，本文中阐述的各种硬件组件和/或逻辑组件可以被分离成包括软件、硬件或两者的子组件。另外，在适用的情况下，预见到软件组件可以被实现为硬件组件，反之亦然。

[0137] 前述公开内容不旨在将本公开内容限制为所公开的精确形式或特定使用领域。因此，预见到鉴于本公开可以进行对本公开的各种替代实施例和/或修改，无论是在本文中明确描述还是暗示。已经因此描述了本公开的实施例，本领域普通技术人员将认识到，可以在不脱离本公开的范围的情况下，在形式和细节上做出改变。因此，本公开仅受权利要求限制。

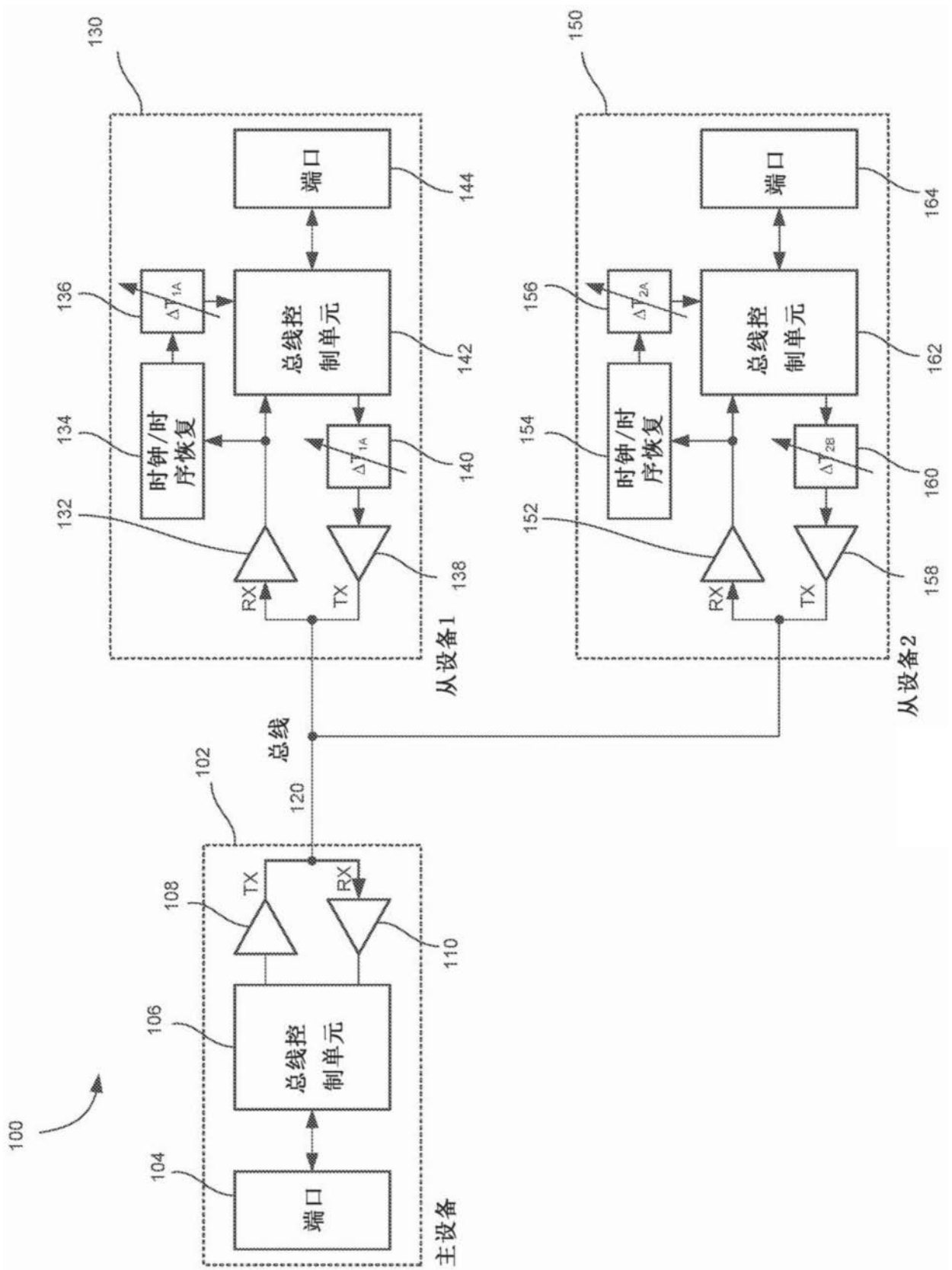

100
主设备
102
104
端口
106
总线控制单元
108
TX
110
RX
120
总线
从设备1
130
132
RX
134
时钟/时序恢复
136
ΔT1A
138
TX
140
ΔT1A
142
总线控制单元
144
端口
从设备2
150
152
RX
154
时钟/时序恢复
156
ΔT2A
158
TX
160
ΔT2B
162
总线控制单元
164
端口

图1

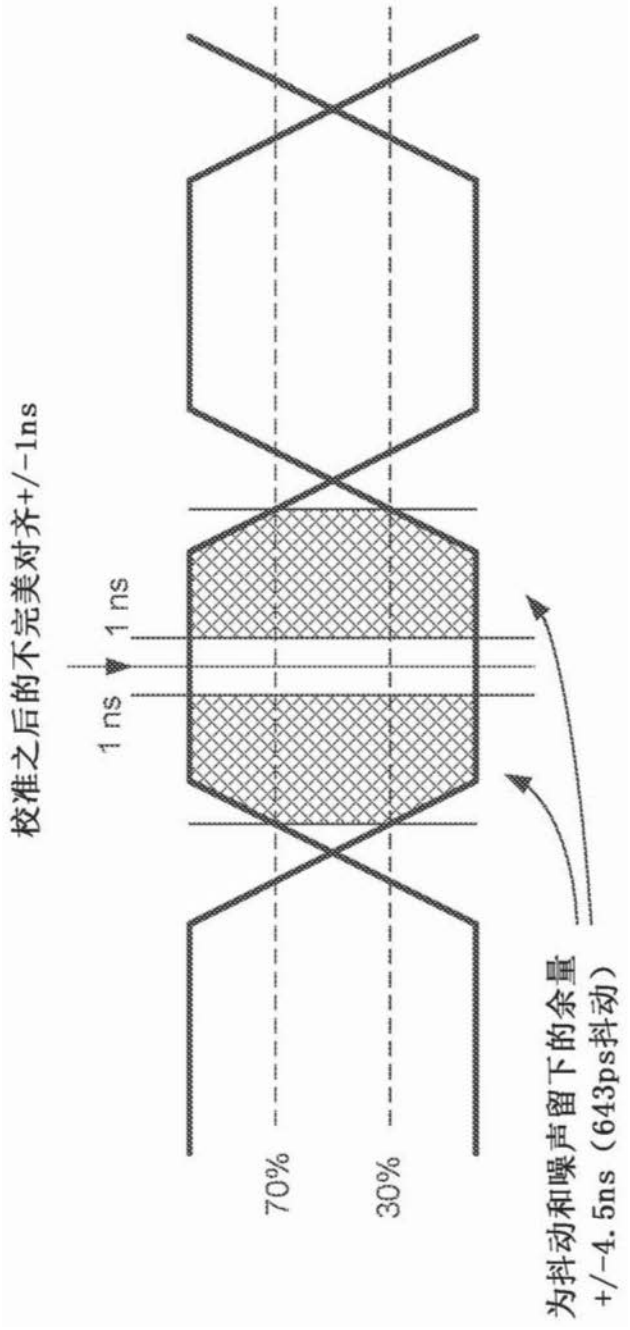


图2

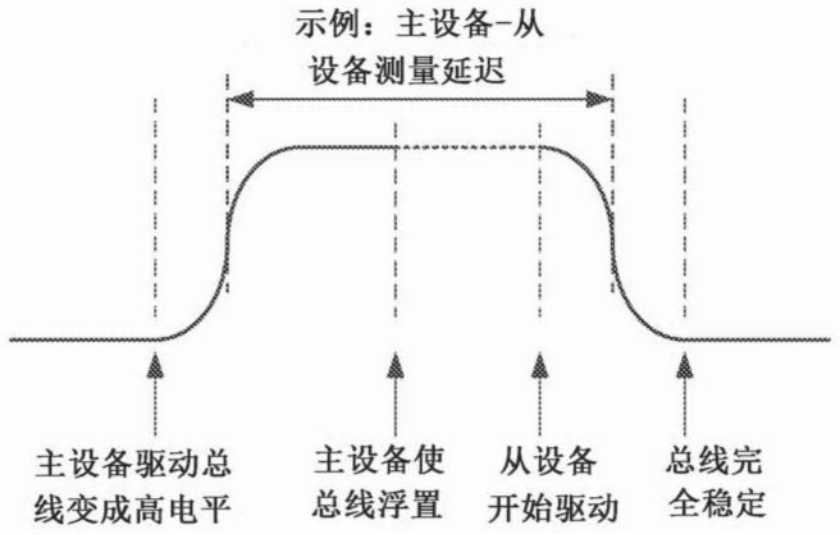

示例：主设备-从
设备测量延迟
主设备驱动总
线变成高电平
主设备使
总线浮置
从设备
开始驱动
总线完
全稳定

图3A

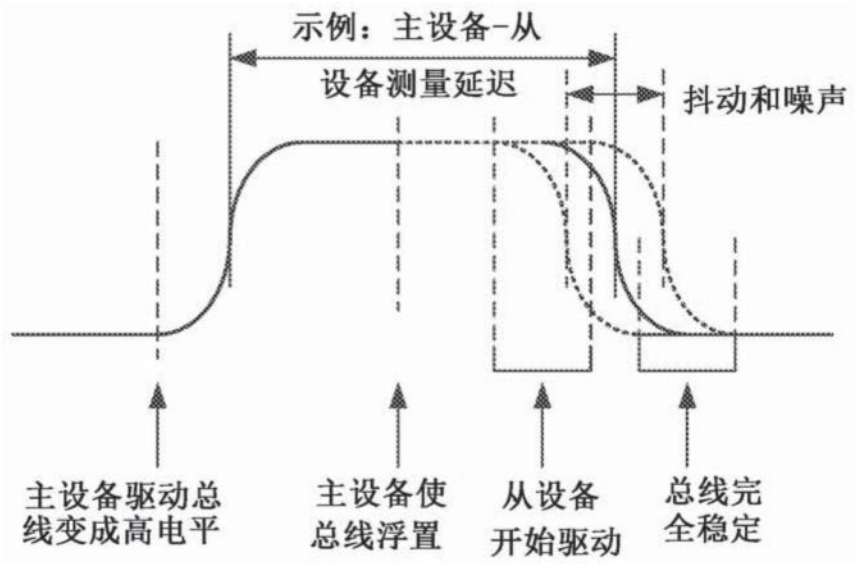

示例：主设备-从
设备测量延迟
抖动和噪声
主设备驱动总
线变成高电平
主设备使
总线浮置
从设备
开始驱动
总线完
全稳定

图3B

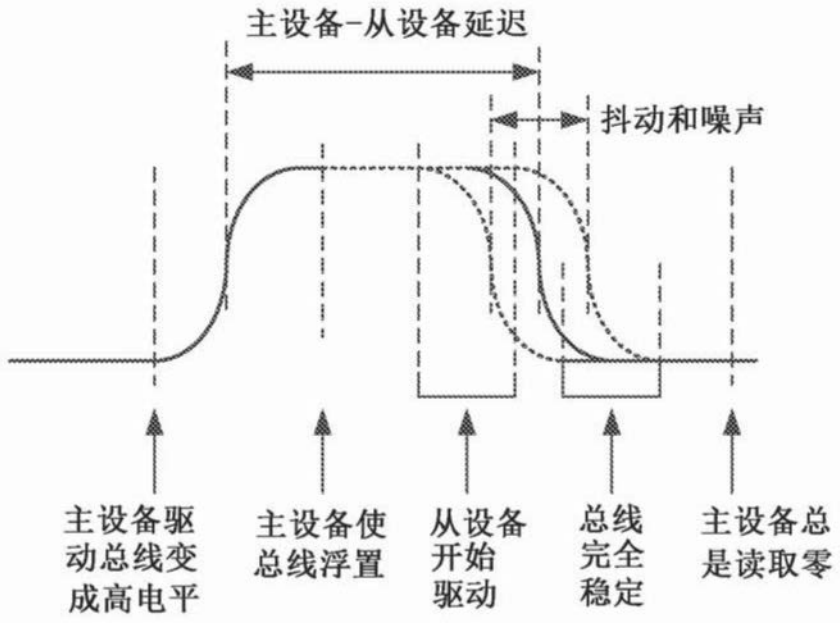

主设备-从设备延迟
抖动和噪声
主设备驱动总线变成高电平
主设备使总线浮置
从设备开始驱动
总线完全稳定
主设备总是读取零

图3C

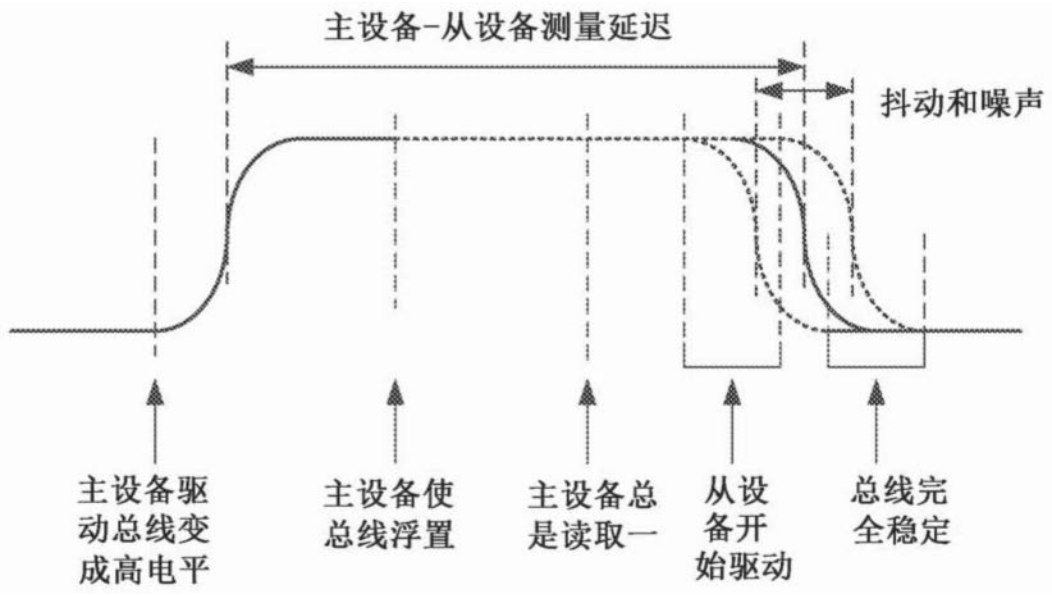

主设备-从设备测量延迟
抖动和噪声
主设备驱动总线变成高电平
主设备使总线浮置
主设备总是读取一
从设备开始驱动
总线完全稳定

图3D

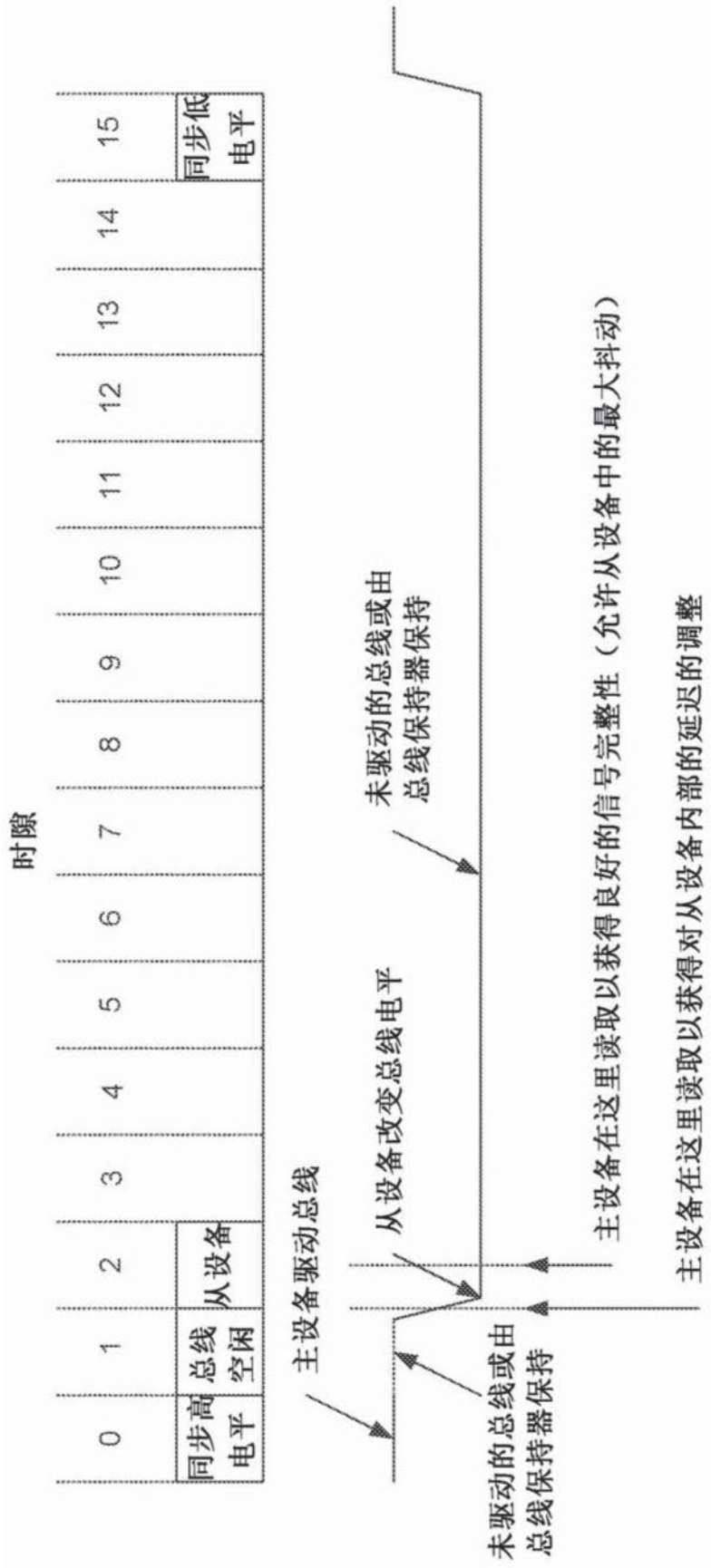

时隙
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
同步高电平
总线空闲
从设备
同步低电平
主设备驱动总线
从设备改变总线电平
未驱动的总线或由总线保持器保持
未驱动的总线或由总线保持器保持
主设备在这里读取以获得良好的信号完整性（允许从设备中的最大抖动）
主设备在这里读取以获得对从设备内部的延迟的调整

图4

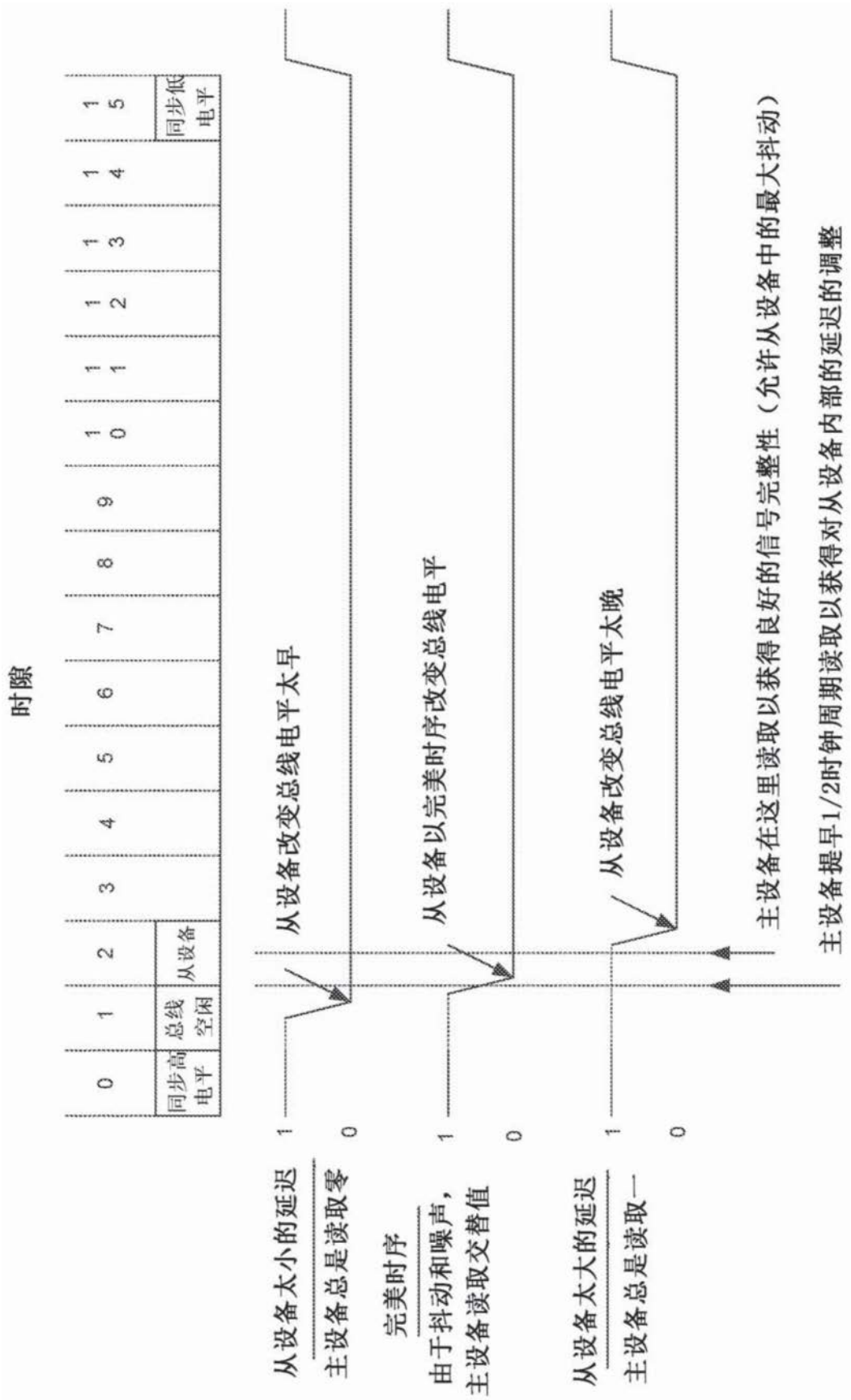

时隙
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
同步高电平
总线空闲
从设备
同步低电平
从设备改变总线电平太早
从设备以完美时序改变总线电平
从设备改变总线电平太晚
从设备太小的延迟
主设备总是读取零
完美时序
由于抖动和噪声，主设备读取交替值
从设备太大的延迟
主设备总是读取一
主设备在这里读取以获得良好的信号完整性（允许从设备中的最大抖动）
主设备提早1/2时钟周期读取以获得对从设备内部的延迟的调整

图5

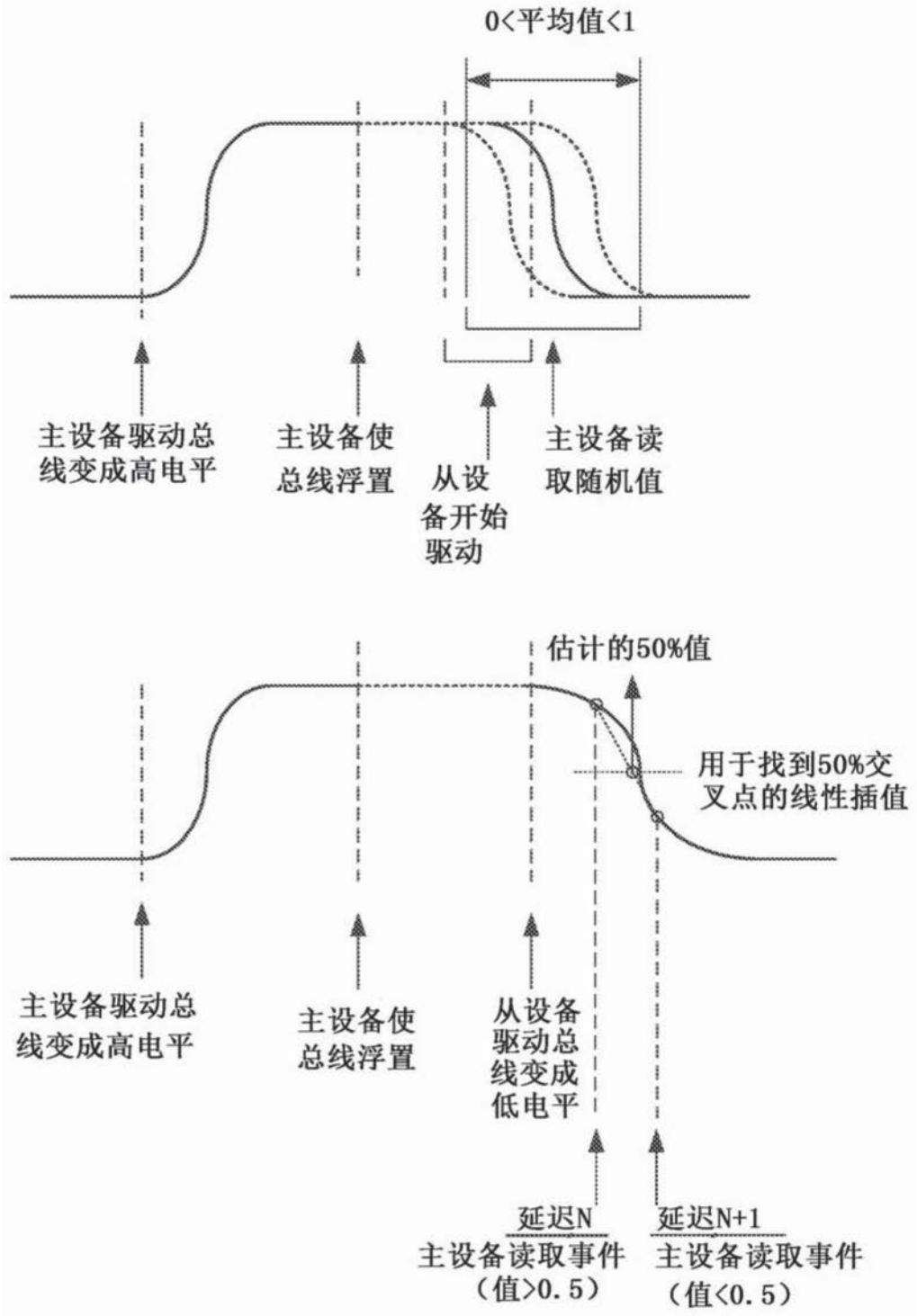

0<平均值<1
主设备驱动总线变成高电平
主设备使总线浮置
从设备开始驱动
主设备读取随机值
估计的50%值
用于找到50%交叉点的线性插值
主设备驱动总线变成高电平
主设备使总线浮置
从设备驱动总线变成低电平
延迟N
主设备读取事件
（值>0.5）
延迟N+1
主设备读取事件
（值<0.5）

图6

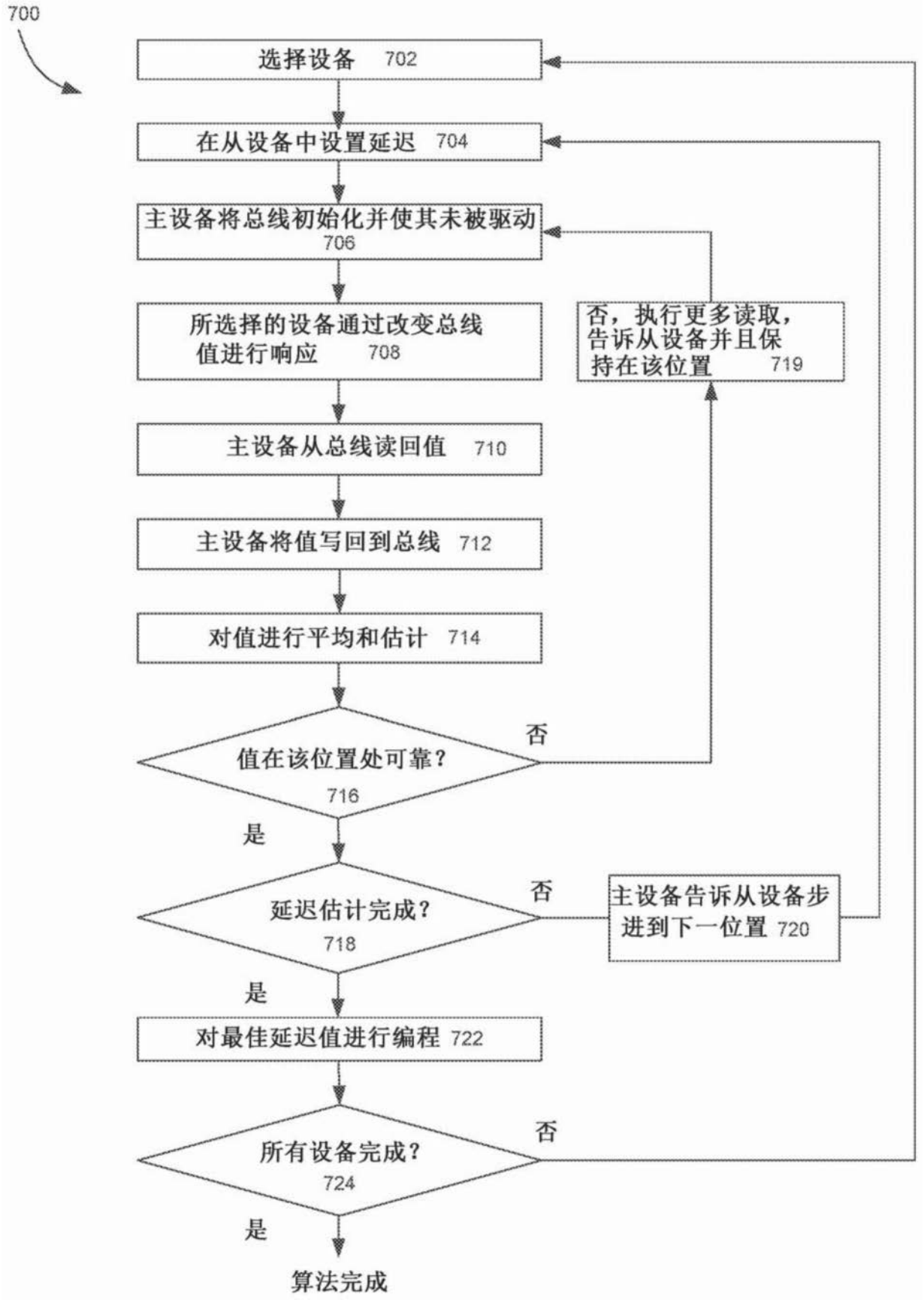

700
选择设备 702
在从设备中设置延迟 704
主设备将总线初始化并使其未被驱动 706
所选择的设备通过改变总线值进行响应 708
否，执行更多读取，告诉从设备并且保持在该位置 719
主设备从总线读回值 710
主设备将值写回到总线 712
对值进行平均和估计 714
值在该位置处可靠？ 716
否
是
延迟估计完成？ 718
否
主设备告诉从设备步进到下一位置 720
是
对最佳延迟值进行编程 722
所有设备完成？ 724
否
是
算法完成

图7

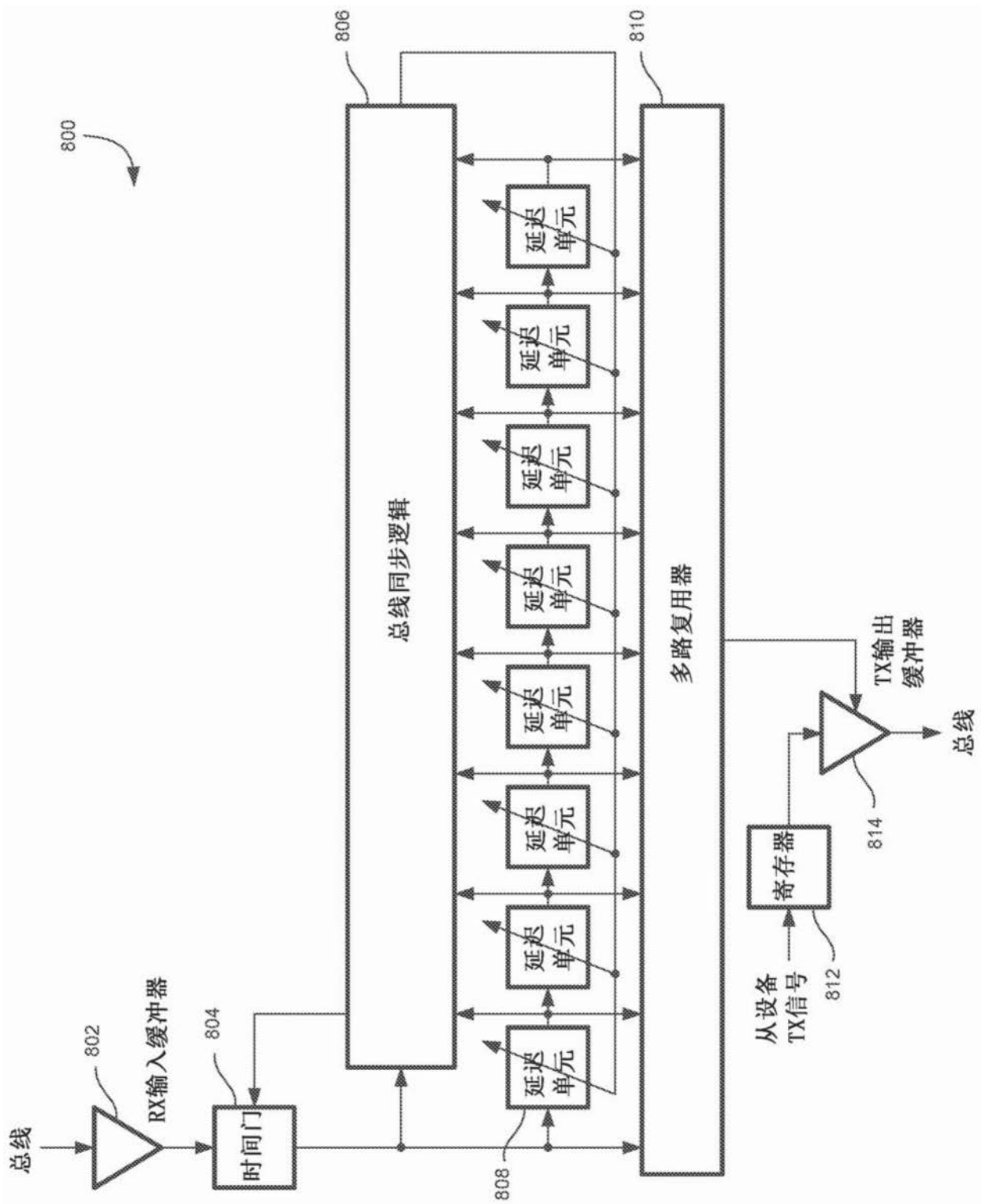

800
806
810
总线同步逻辑
多路复用器
总线
802
RX输入缓冲器
804
时间门
808
延迟单元
从设备TX信号
812
寄存器
814
TX输出缓冲器
总线

图8

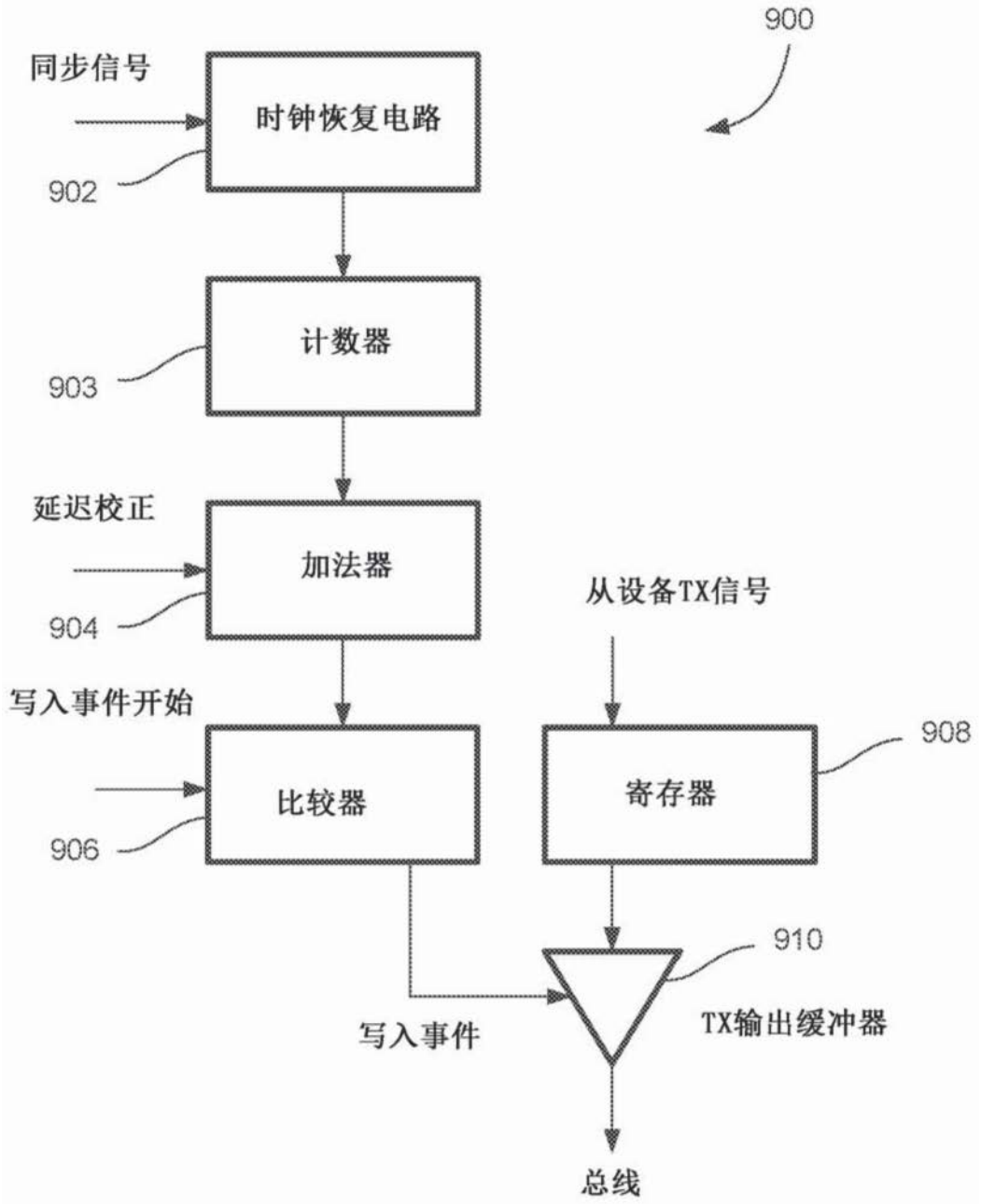

900
同步信号
时钟恢复电路
902
计数器
903
延迟校正
加法器
904
从设备TX信号
写入事件开始
比较器
906
寄存器
908
910
写入事件
TX输出缓冲器
总线

图9

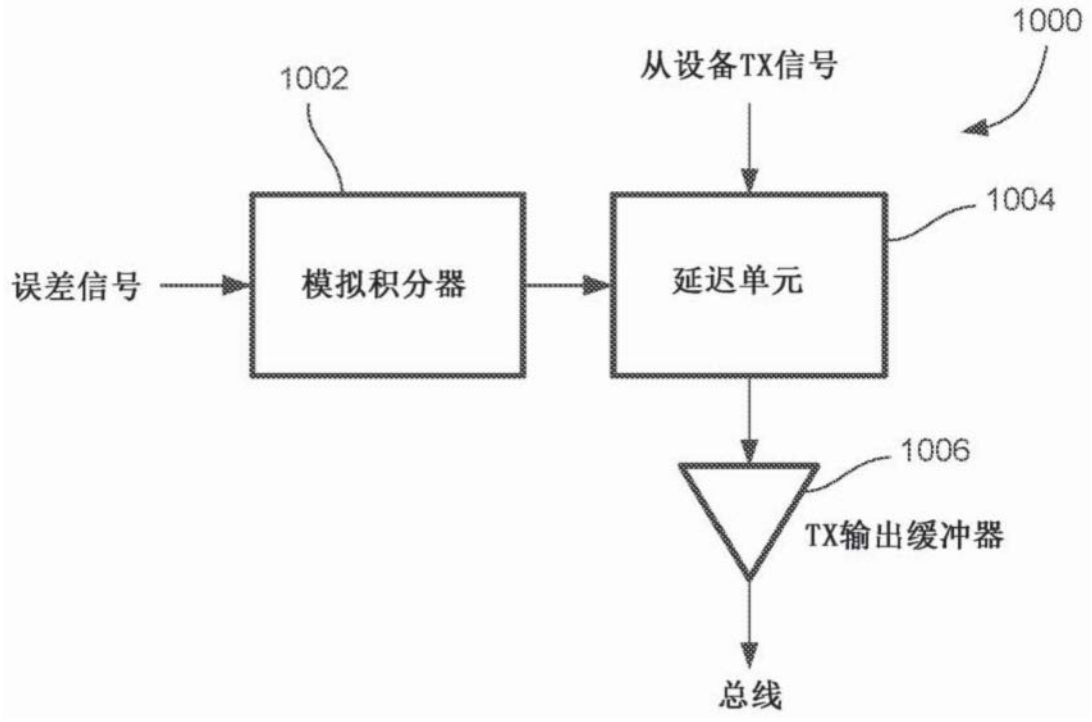

1000
从设备TX信号
1002
1004
误差信号
模拟积分器
延迟单元
1006
TX输出缓冲器
总线

图10

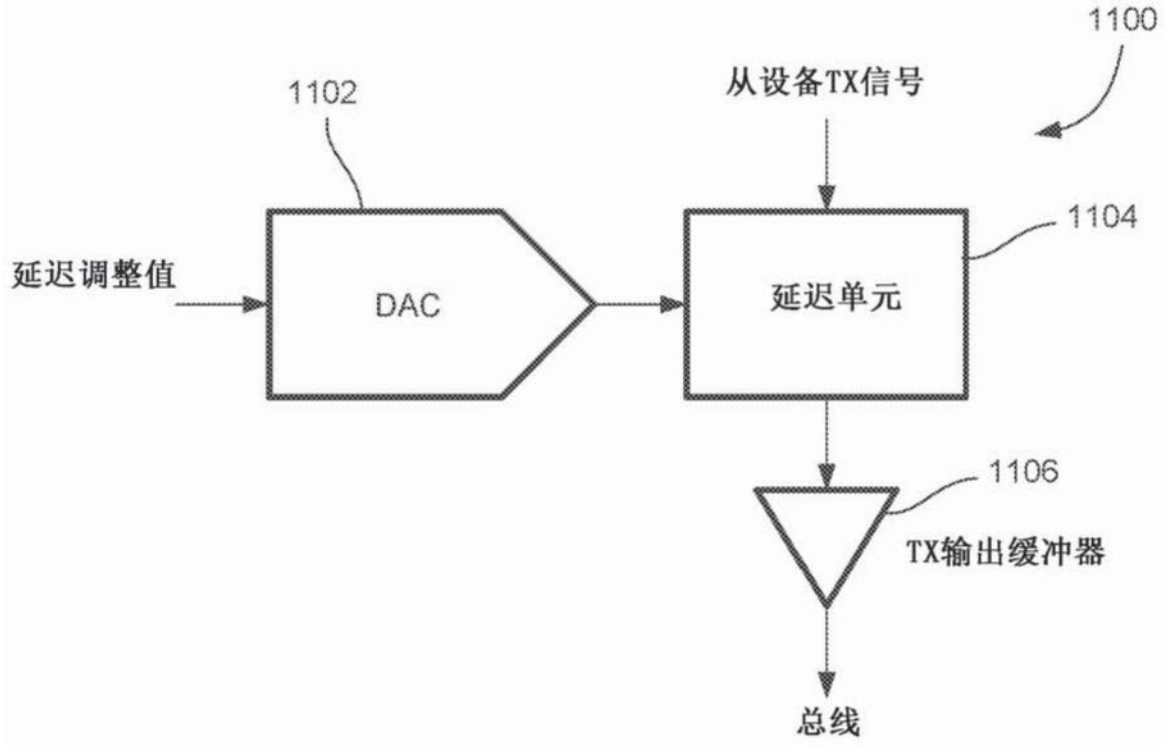

1100
从设备TX信号
1102
1104
延迟调整值
DAC
延迟单元
1106
TX输出缓冲器
总线

图11

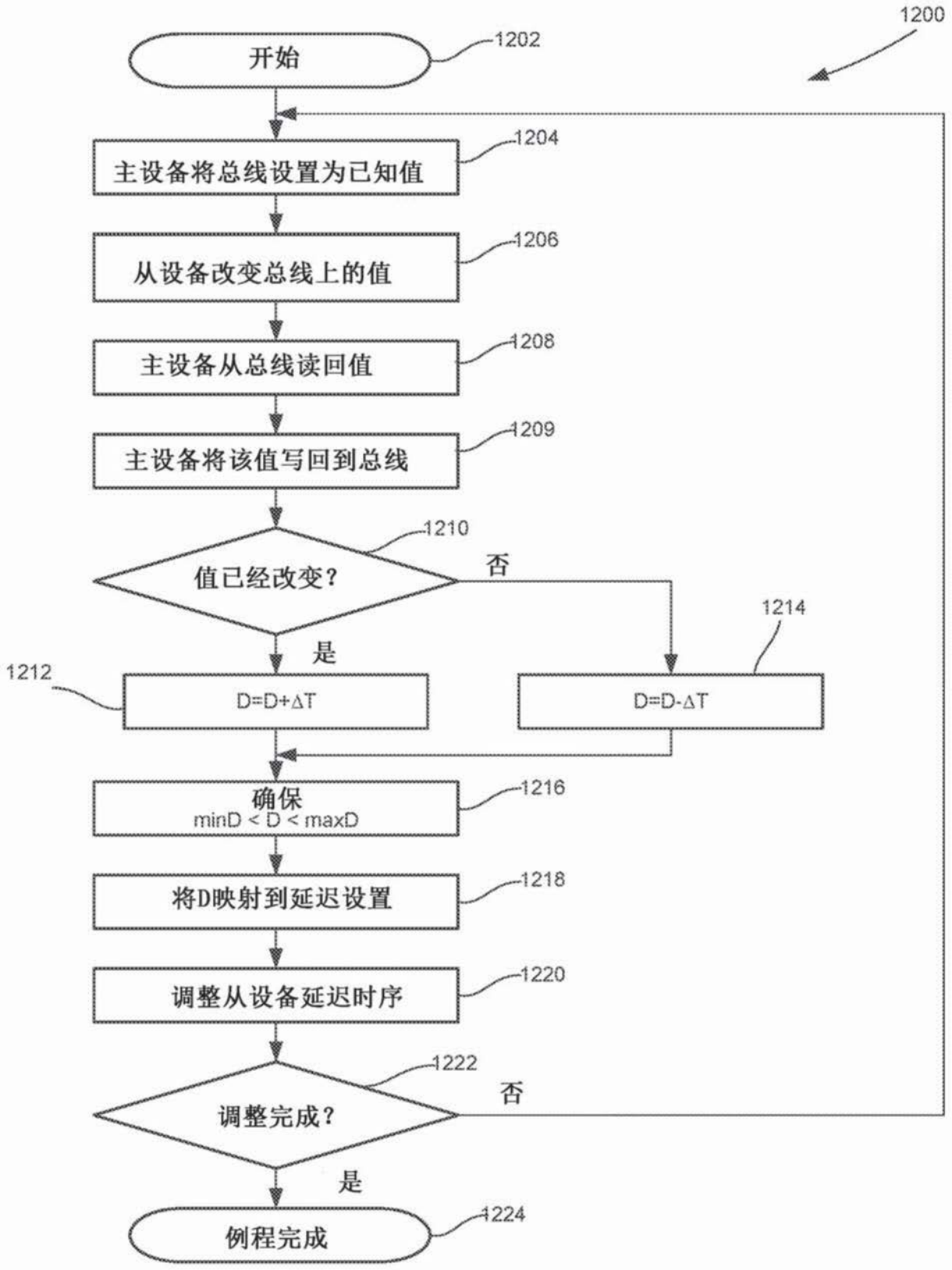

1200
开始
1202
主设备将总线设置为已知值
1204
从设备改变总线上的值
1206
主设备从总线读回值
1208
主设备将该值写回到总线
1209
值已经改变?
1210
否
是
1212
D=D+ΔT
1214
D=D-ΔT
确保
minD < D < maxD
1216
将D映射到延迟设置
1218
调整从设备延迟时序
1220
调整完成?
1222
否
是
例程完成
1224

图12

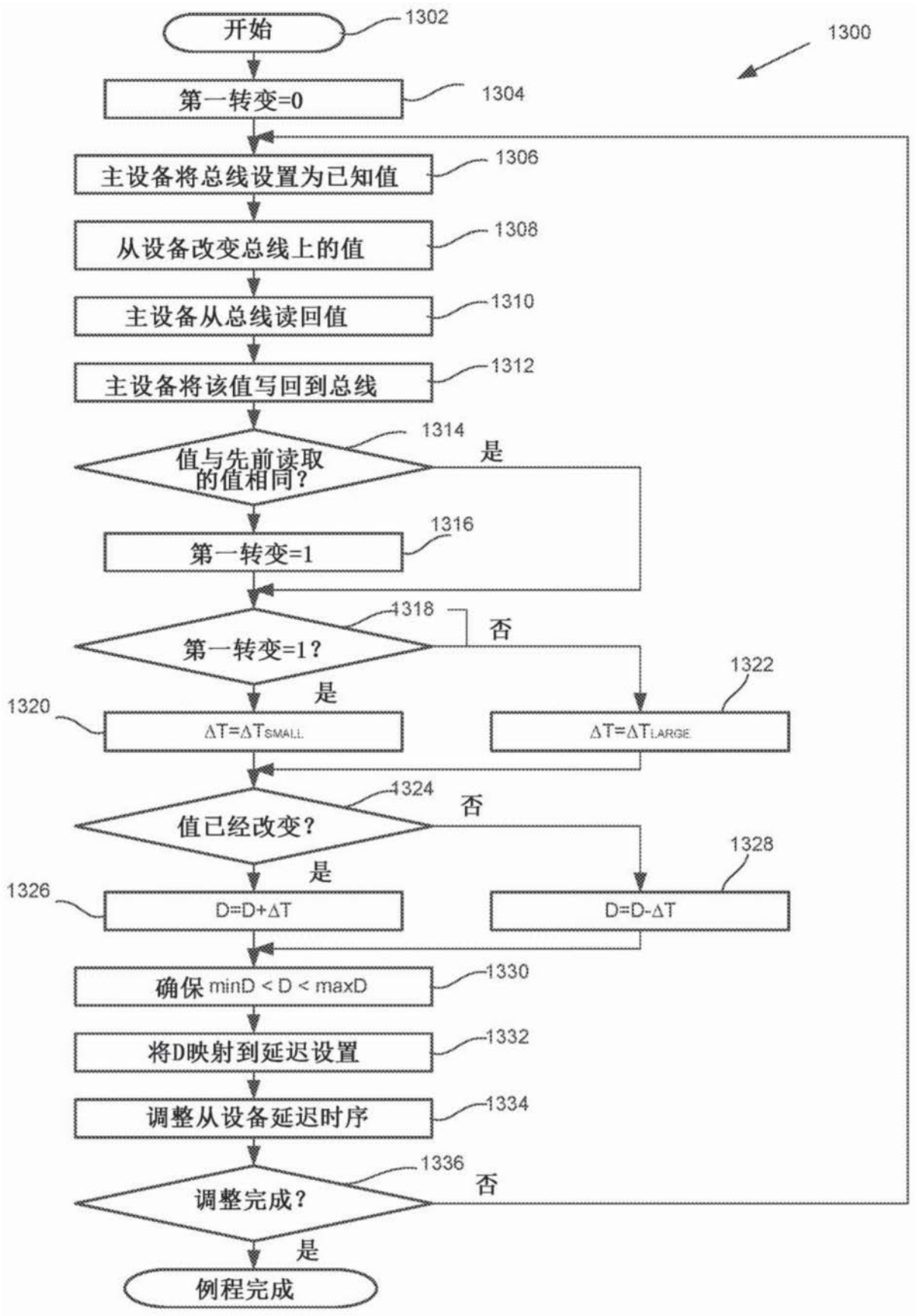

开始
1302
1300
第一转变=0
1304
主设备将总线设置为已知值
1306
从设备改变总线上的值
1308
主设备从总线读回值
1310
主设备将该值写回到总线
1312
值与先前读取的值相同?
1314
是
第一转变=1
1316
第一转变=1?
1318
否
是
ΔT=ΔTSMALL
1320
ΔT=ΔTLARGE
1322
值已经改变?
1324
否
是
D=D+ΔT
1326
D=D-ΔT
1328
确保 minD < D < maxD
1330
将D映射到延迟设置
1332
调整从设备延迟时序
1334
调整完成?
1336
否
是
例程完成

图13

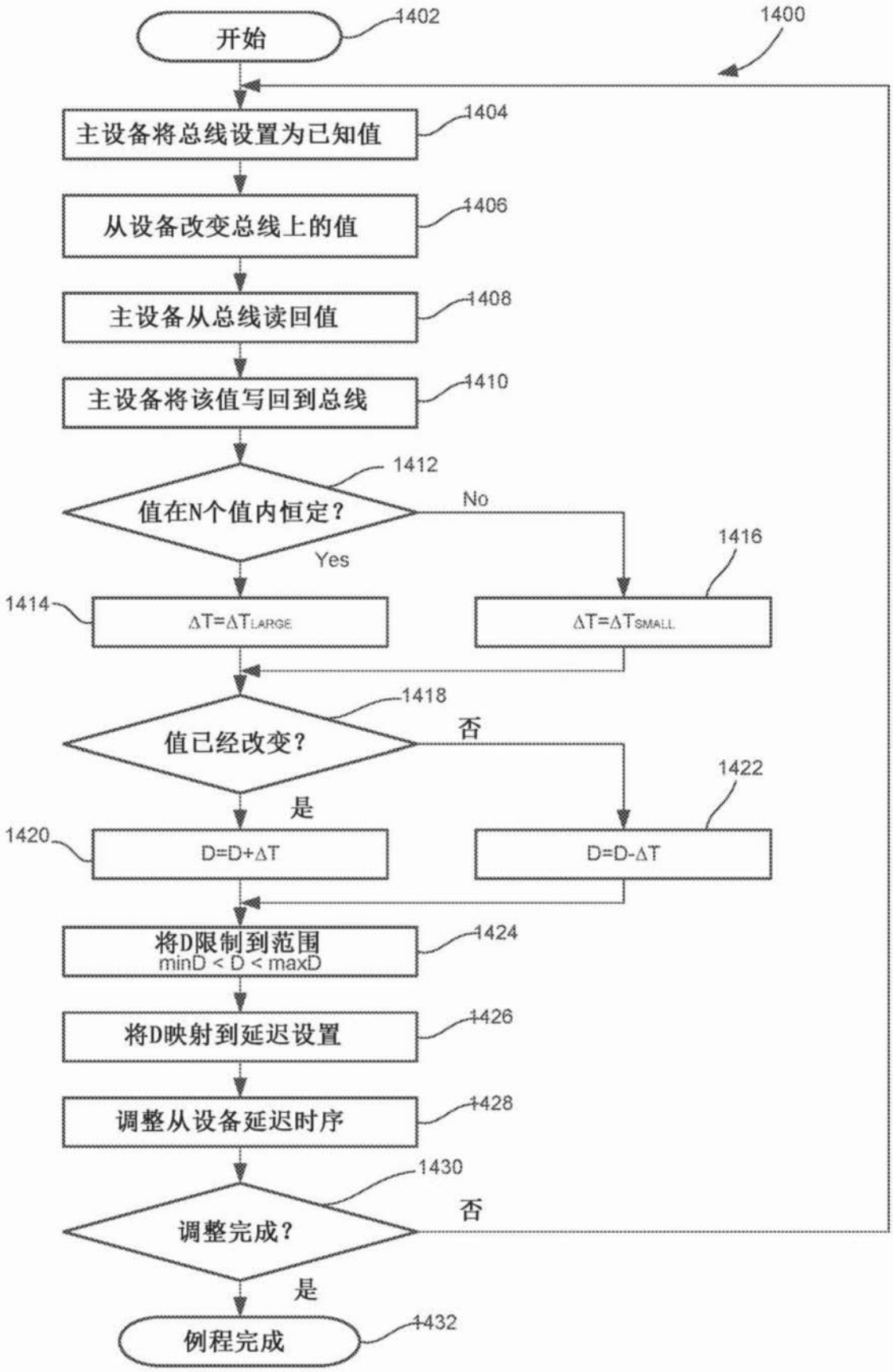

开始
1402
1400
主设备将总线设置为已知值
1404
从设备改变总线上的值
1406
主设备从总线读回值
1408
主设备将该值写回到总线
1410
值在N个值内恒定?
1412
No
Yes
1416
1414
ΔT=ΔTLARGE
ΔT=ΔTSMALL
值已经改变?
1418
否
是
1422
1420
D=D+ΔT
D=D-ΔT
将D限制到范围
minD < D < maxD
1424
将D映射到延迟设置
1426
调整从设备延迟时序
1428
调整完成?
1430
否
是
例程完成
1432

图14

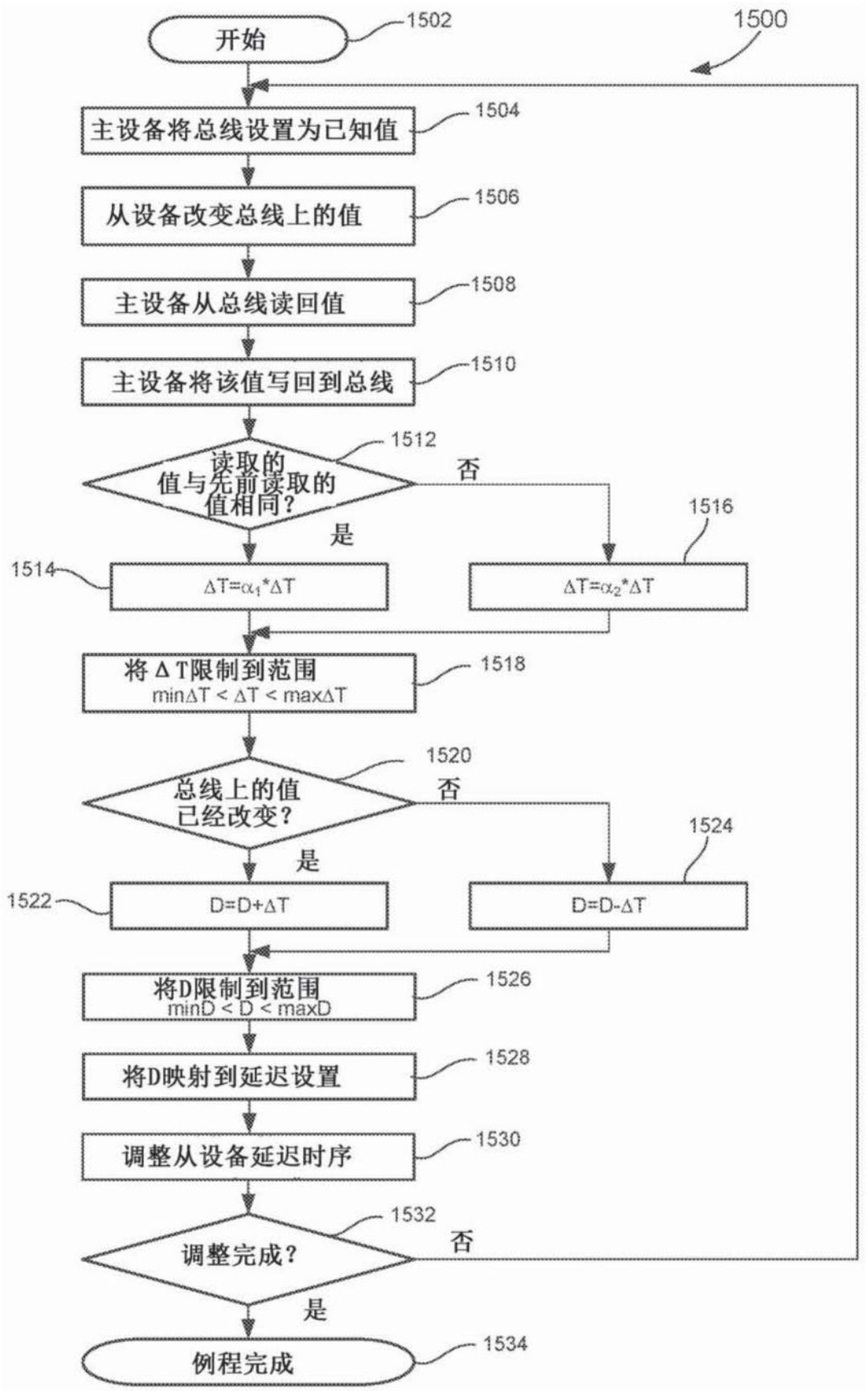

开始
1502
1500
主设备将总线设置为已知值
1504
从设备改变总线上的值
1506
主设备从总线读回值
1508
主设备将该值写回到总线
1510
读取的值与先前读取的值相同?
1512
否
是
1516
1514
ΔT=α1*ΔT
ΔT=α2*ΔT
将ΔT限制到范围
minΔT < ΔT < maxΔT
1518
总线上的值已经改变?
1520
否
是
1524
1522
D=D+ΔT
D=D-ΔT
将D限制到范围
minD < D < maxD
1526
将D映射到延迟设置
1528
调整从设备延迟时序
1530
调整完成?
1532
否
是
例程完成
1534

图15

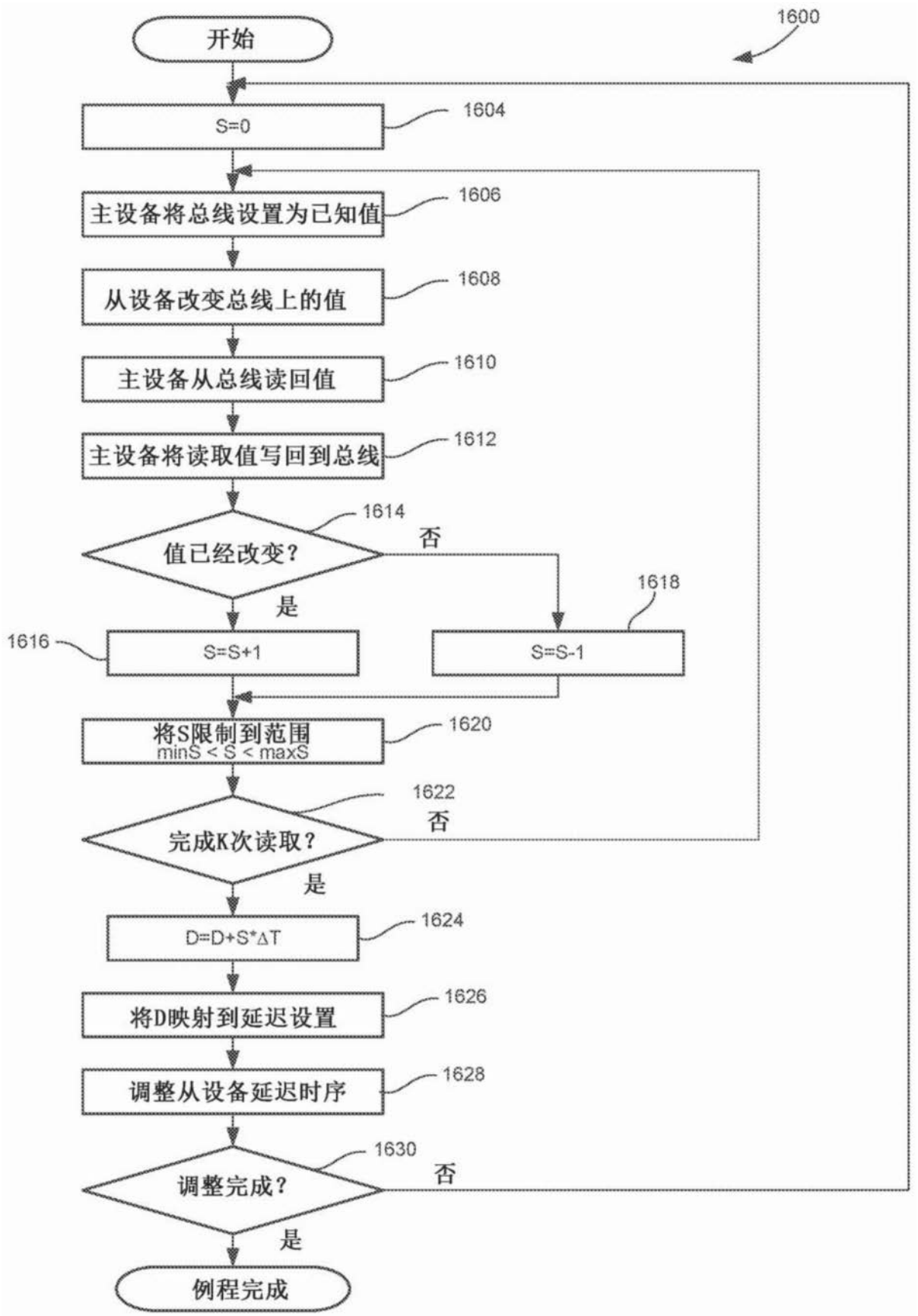

1600
开始
S=0
1604
主设备将总线设置为已知值
1606
从设备改变总线上的值
1608
主设备从总线读回值
1610
主设备将读取值写回到总线
1612
值已经改变?
1614
否
是
1618
1616
S=S+1
S=S-1
将S限制到范围
minS < S < maxS
1620
完成K次读取?
1622
否
是
D=D+S*ΔT
1624
将D映射到延迟设置
1626
调整从设备延迟时序
1628
调整完成?
1630
否
是
例程完成

图16

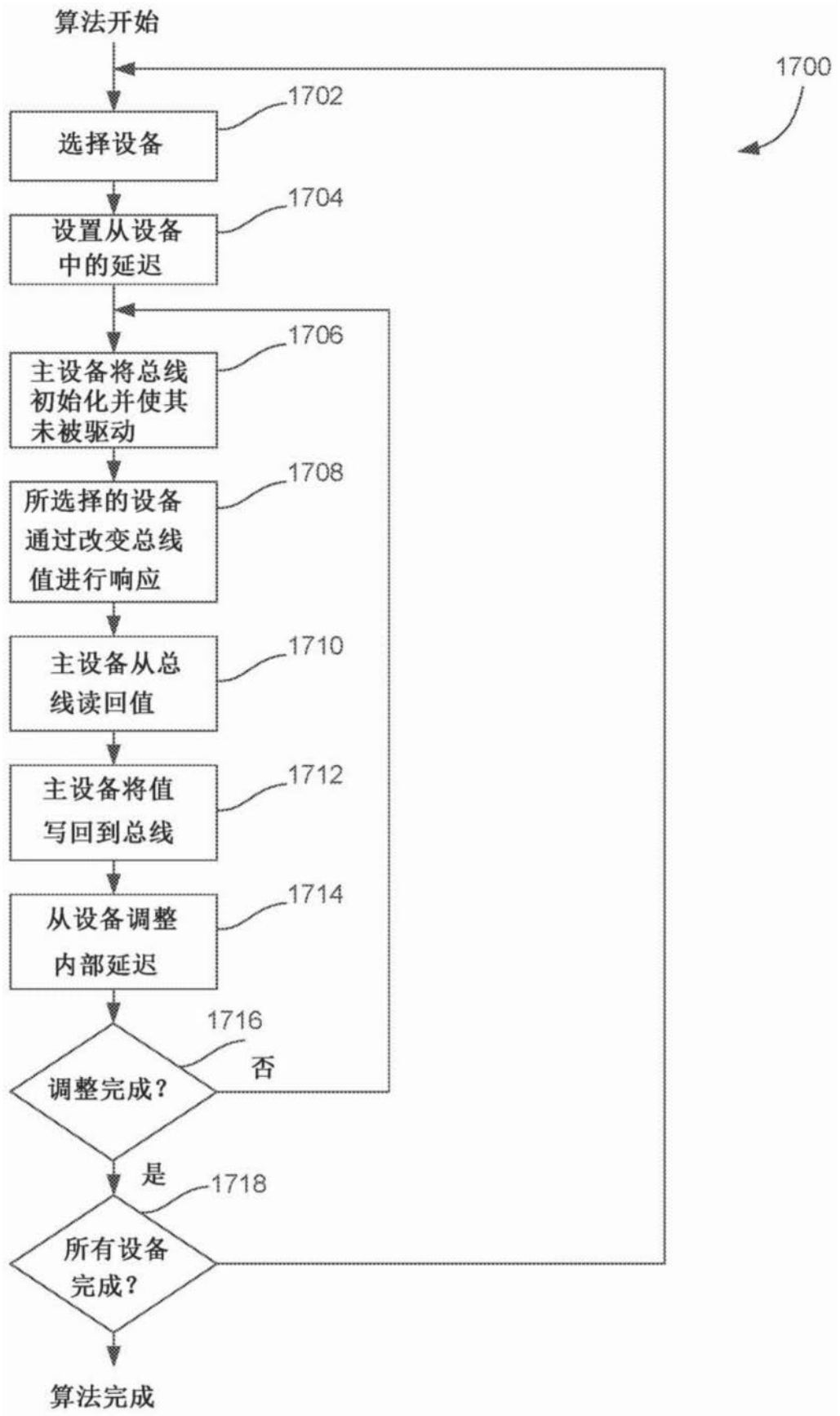

算法开始
1700
1702
选择设备
1704
设置从设备中的延迟
1706
主设备将总线初始化并使其未被驱动
1708
所选择的设备通过改变总线值进行响应
1710
主设备从总线读回值
1712
主设备将值写回到总线
1714
从设备调整内部延迟
1716
调整完成?
否
是
1718
所有设备完成?
算法完成

图17

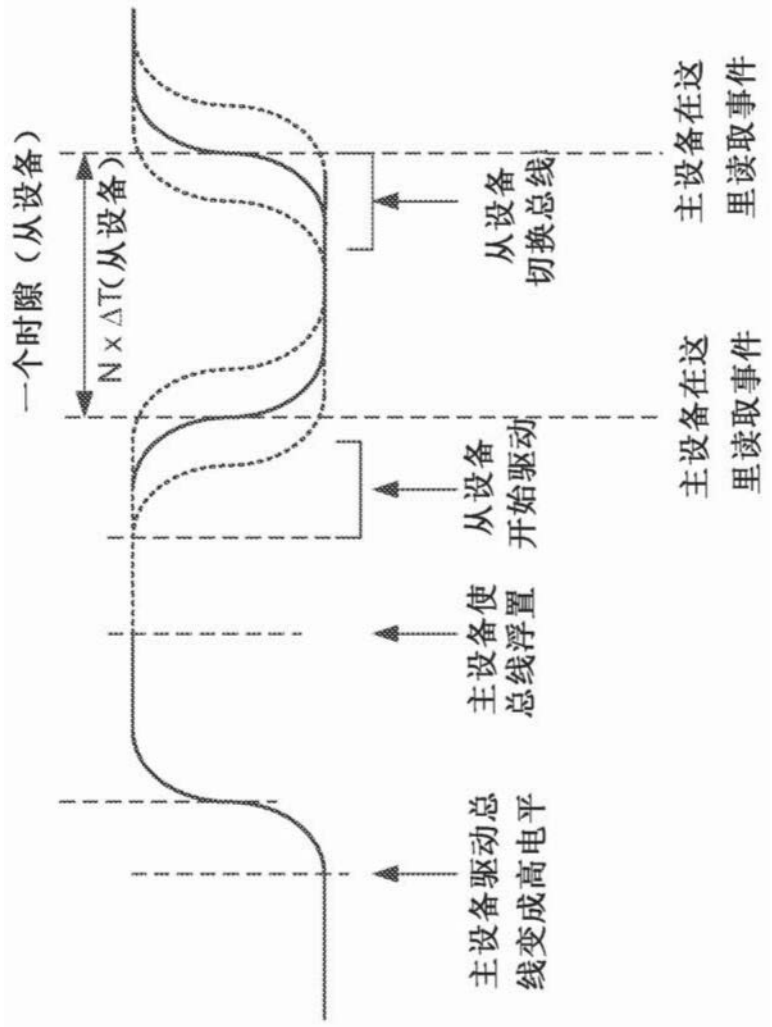

一个时隙（从设备）
N x ΔT(从设备)
主设备驱动总线变成高电平
主设备使总线浮置
从设备开始驱动
从设备切换总线
主设备在这里读取事件
主设备在这里读取事件

图18

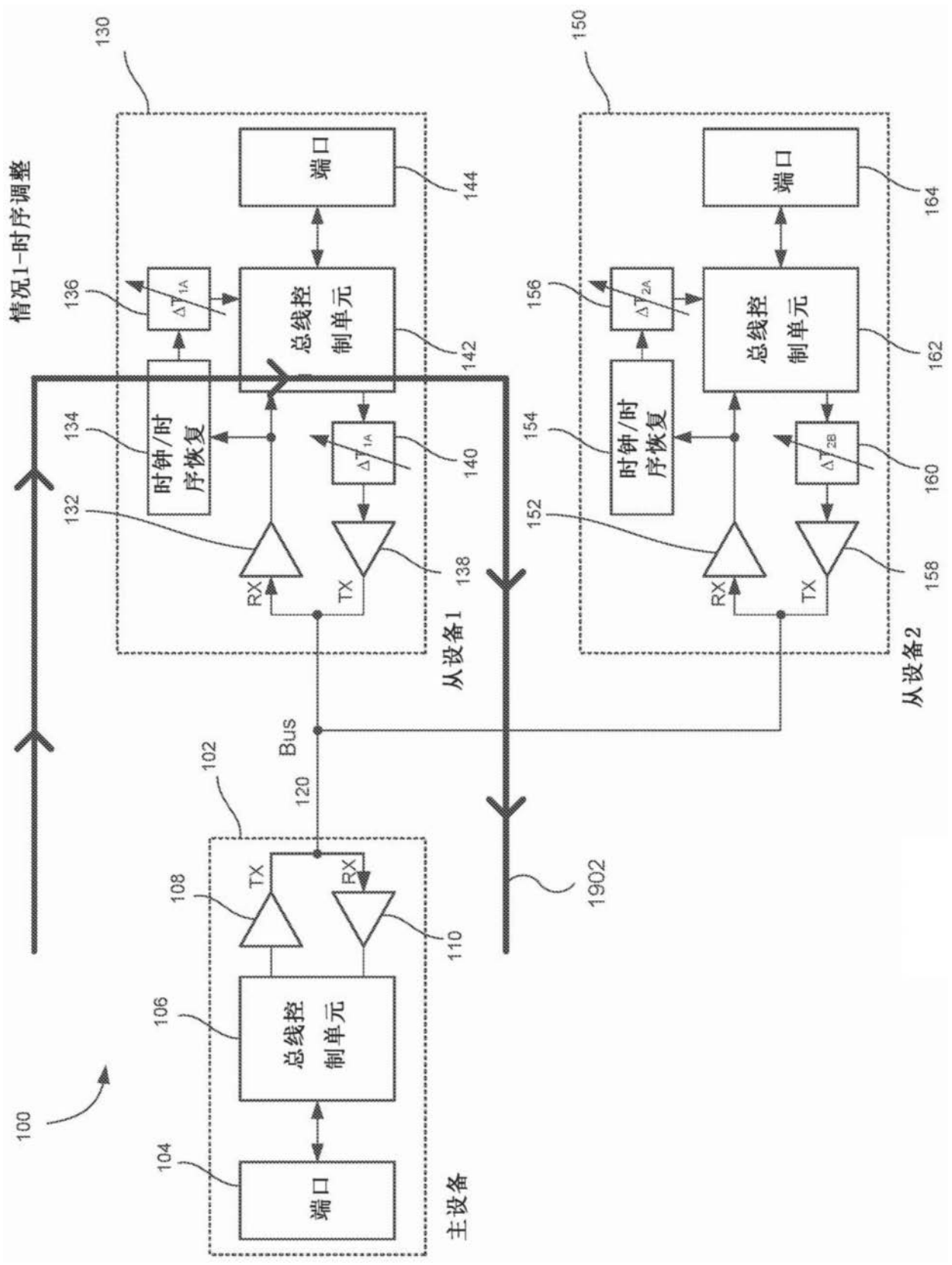

情况1-时序调整
100
主设备
从设备1
从设备2
总线控制单元
端口
时钟/时序恢复
RX
TX
Bus
102
104
106
108
110
120
130
132
134
136
138
140
142
144
150
152
154
156
158
160
162
164
1902

图19

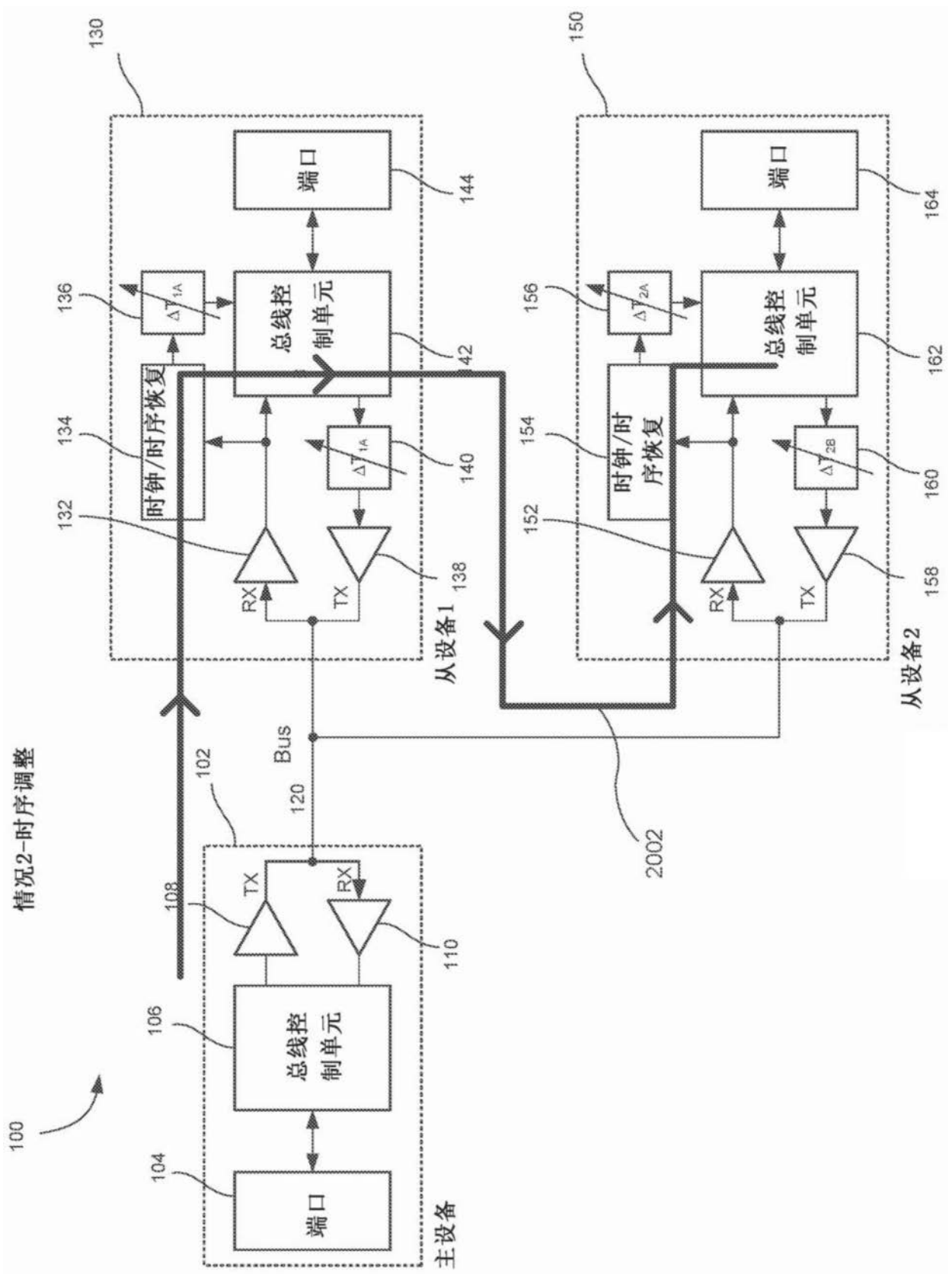

情况2-时序调整
100
主设备
从设备1
从设备2
端口
总线控制单元
时钟/时序恢复
TX
RX
Bus
102
104
106
108
110
120
130
132
134
136
138
140
144
150
152
154
156
158
160
162
164
42
2002

图20

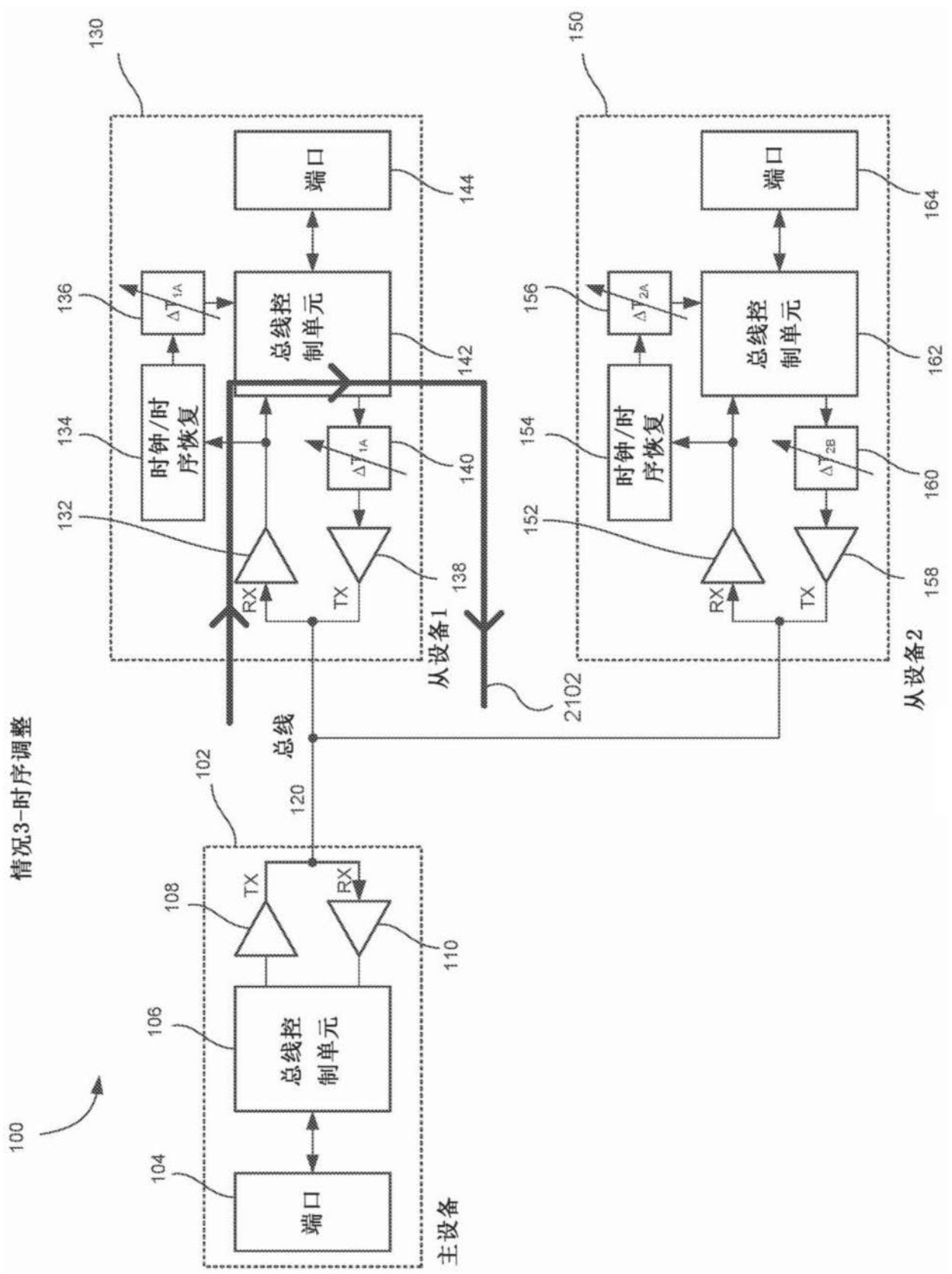

情况3-时序调整
100
主设备
端口
104
总线控制单元
106
TX
108
RX
110
102
总线
120
从设备1
从设备2
2102
130
150
端口
144
164
总线控制单元
142
162
时钟/时序恢复
134
154
ΔT_{1A}
136
ΔT_{2A}
156
ΔT_{1A}
140
ΔT_{2B}
160
RX
132
152
TX
138
158

图21

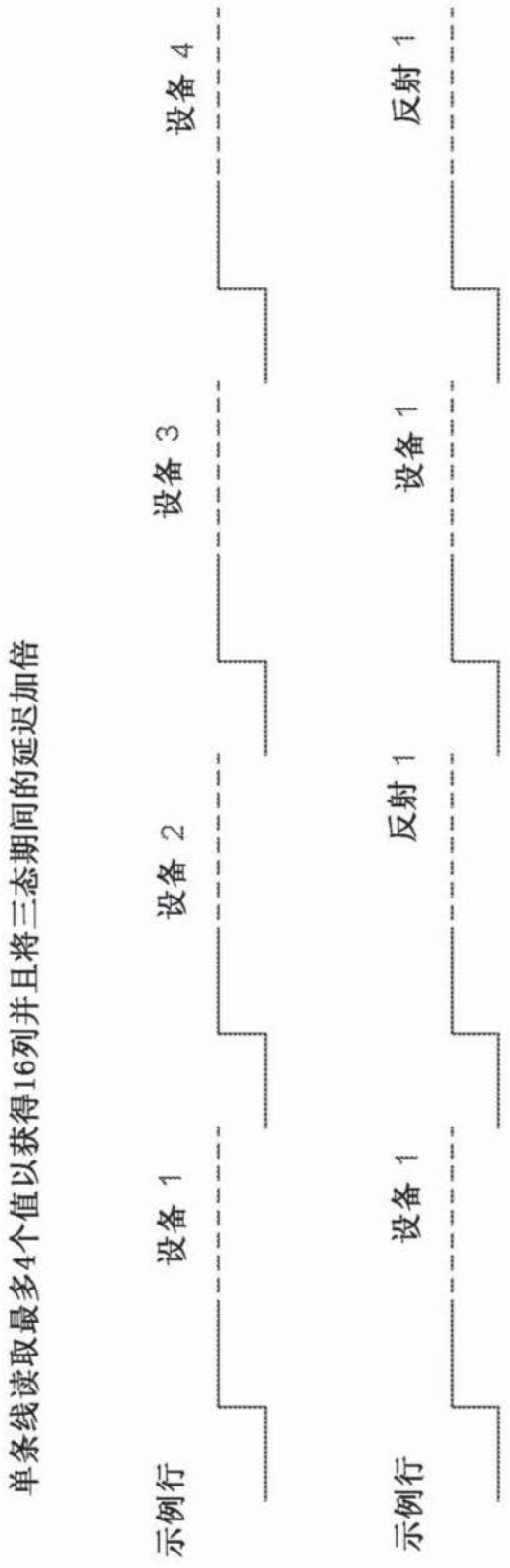

单条线读取最多4个值以获得16列并且将三态期间的延迟加倍
示例行
设备 1
设备 2
设备 3
设备 4
示例行
设备 1
反射 1
设备 1
反射 1

图22

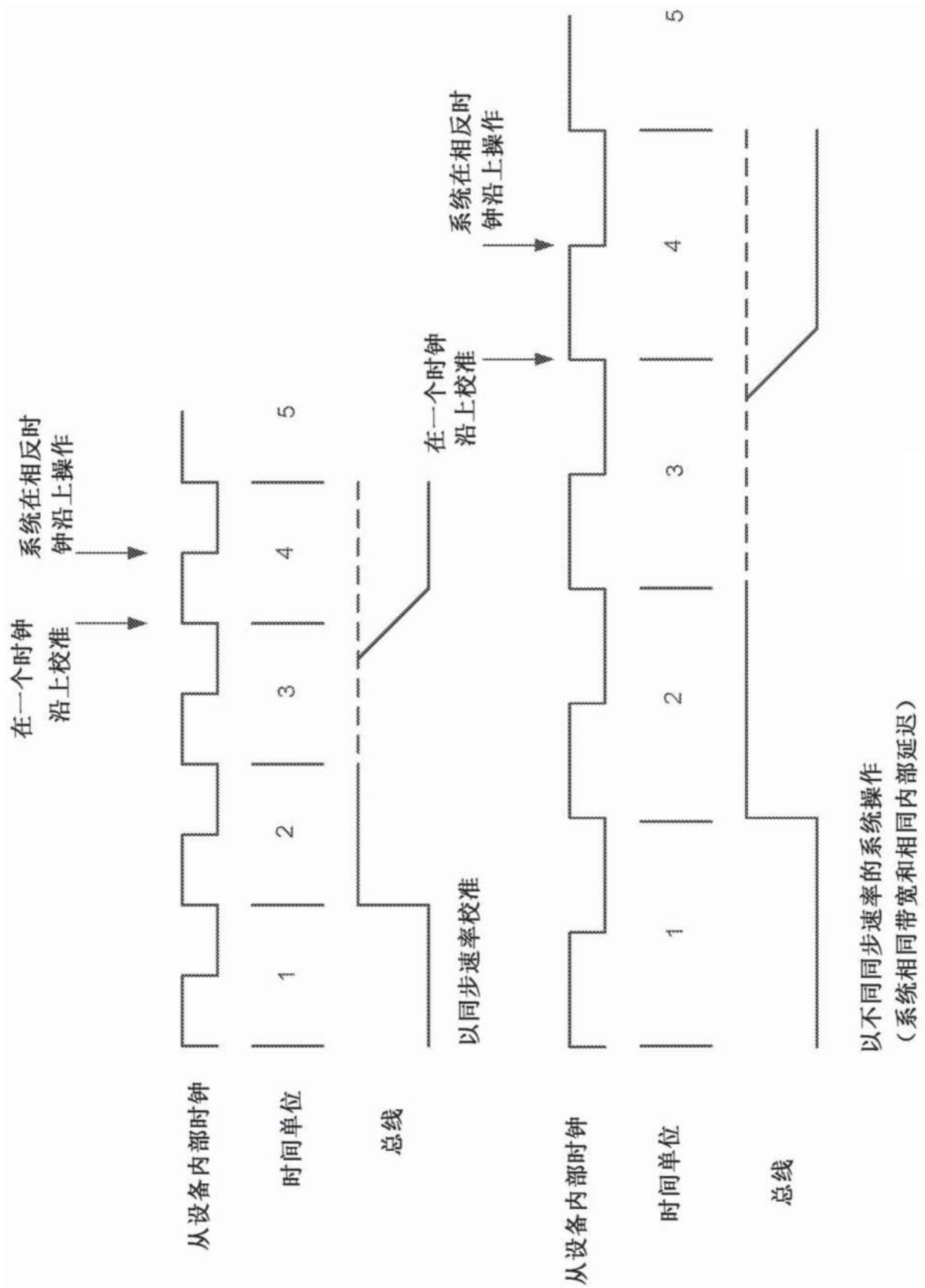

在一个时钟沿上校准
系统在相反时钟沿上操作
从设备内部时钟
时间单位
1
2
3
4
5
总线
以同步速率校准
在一个时钟沿上校准
系统在相反时钟沿上操作
从设备内部时钟
时间单位
1
2
3
4
5
总线
以不同同步速率的系统操作
（系统相同带宽和相同内部延迟）

图23

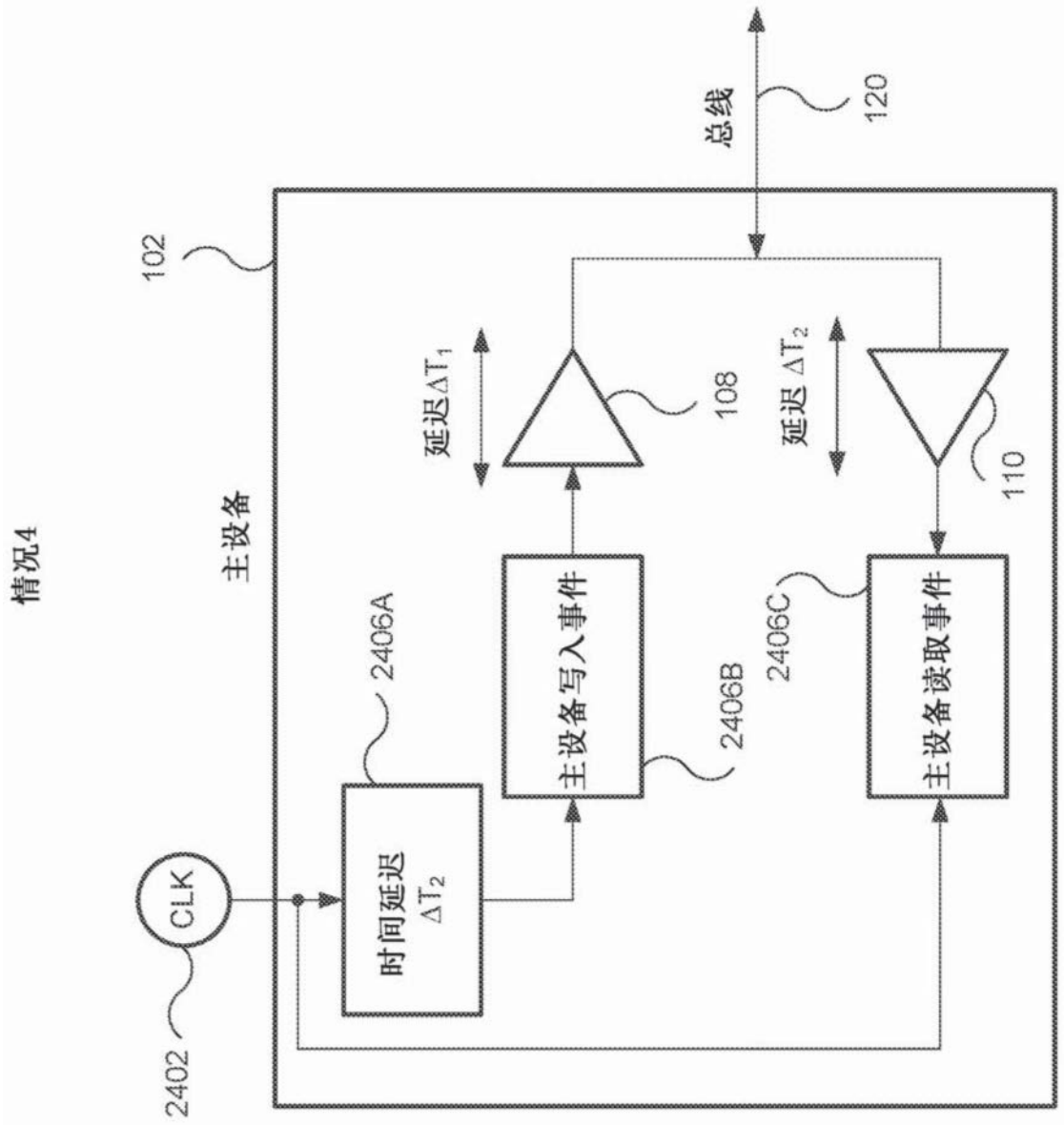

情况4
主设备
102
总线
120
延迟ΔT_1
108
延迟ΔT_2
110
2406A
时间延迟
ΔT_2
主设备写入事件
2406B
2406C
主设备读取事件
CLK
2402

图24A

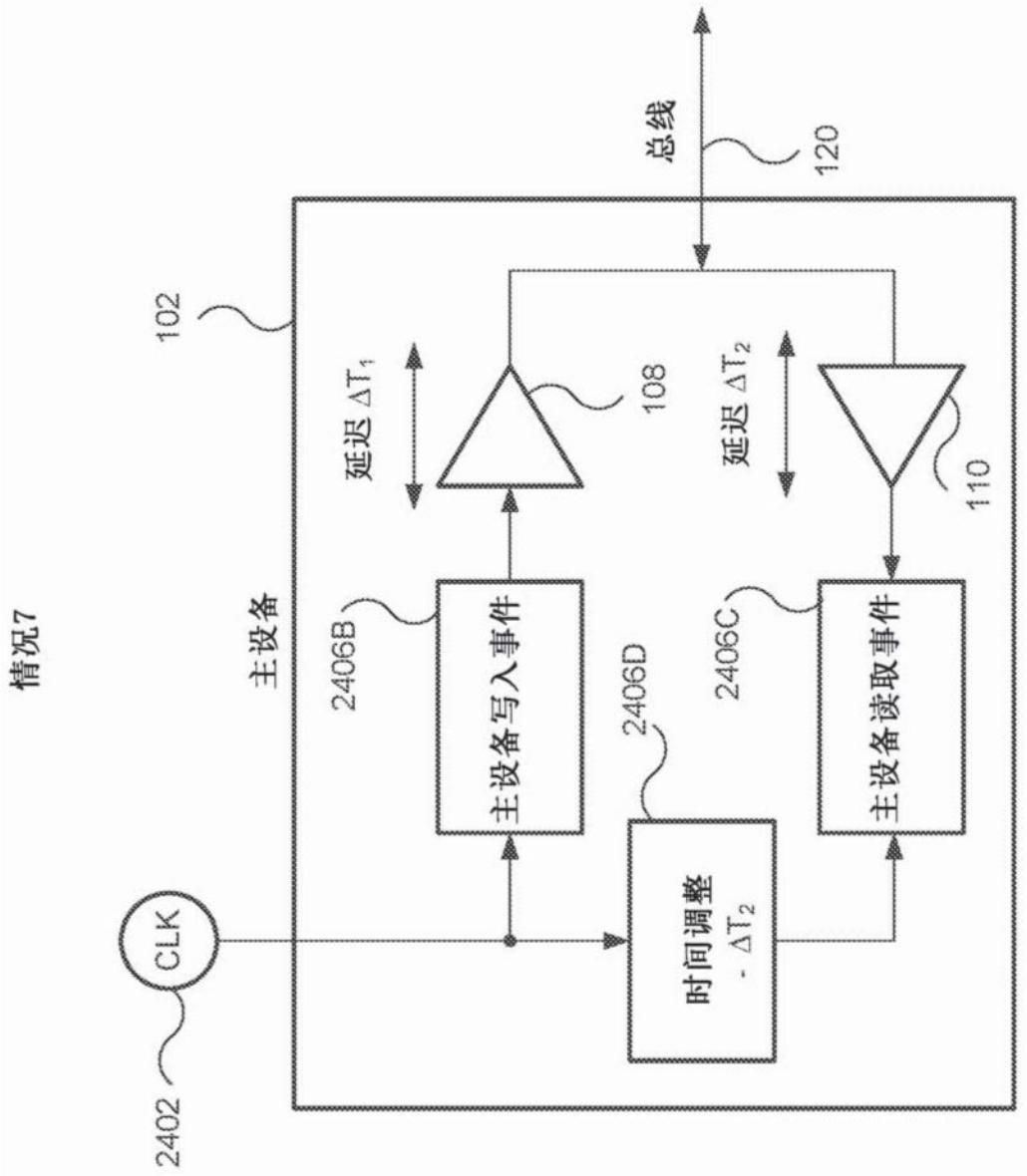

情况7
主设备
102
总线
120
延迟 ΔT_1
108
延迟 ΔT_2
110
2406B
主设备写入事件
2406C
主设备读取事件
2406D
时间调整
$-\Delta T_2$
CLK
2402

图24B