



(12) 发明专利

(10) 授权公告号 CN 101790730 B

(45) 授权公告日 2013. 10. 23

(21) 申请号 200880018126. 4

(22) 申请日 2008. 05. 30

(30) 优先权数据

11/809, 613 2007. 05. 31 US

(85) PCT申请进入国家阶段日

2009. 11. 30

(86) PCT申请的申请数据

PCT/US2008/006831 2008. 05. 30

(87) PCT申请的公布数据

W02008/150435 EN 2008. 12. 11

(73) 专利权人 新思公司

地址 美国加利福尼亚州

(72) 发明人 K·S·麦克尔文

(74) 专利代理机构 北京润平知识产权代理有限公司

公司 11283

代理人 周建秋 王凤桐

(51) Int. Cl.

G06F 17/50(2006. 01)

H03K 19/173(2006. 01)

(56) 对比文件

US 6438735 B1, 2002. 08. 20,

CORSONELLO P ET AL. Design of 3:1 multiplexer standard. 《ELECTRONICS LETTERS, IEE STEVENAGE. GB》. 2000, 第 36 卷 1994-1995.

审查员 刘希

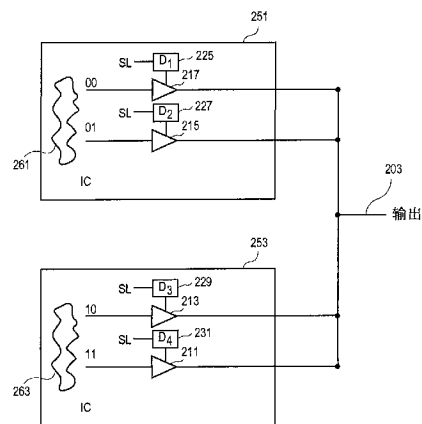
权利要求书2页 说明书7页 附图6页

(54) 发明名称

用于设计多路转换器的方法和设备

(57) 摘要

本发明公开了用于在一个或多个集成电路中设计多路转换器的方法和设备。一种示例方法包括接收第一多路转换器的表示;将所述第一多路转换器的表示转换为所述第一多路转换器的划分中枢表示;以及对所述第一多路转换器的划分中枢表示进行划分以创建多个第二多路转换器。另一种示例方法包括:将第一多路转换器的表示分解为多个第二多路转换器的表示,所述第二多路转换器被一起耦合到公共输出,且不存在任何中间多路转换器介于所述第二多路转换器与所述公共输出之间;以及将所述第二多路转换器在至少一个集成电路的部分之间进行划分。



1. 一种设计多路转换器的方法,该方法包括:
接收第一多路转换器的表示;
将所述第一多路转换器的表示分解为所述第一多路转换器的划分中枢形式,其中所述划分中枢形式包括以下中的一者:(a)多个三态驱动器;(b)有线 OR 阵列;或者(c)有线 AND 阵列;
对所述第一多路转换器的划分中枢形式进行划分以创建多个第二多路转换器。
2. 根据权利要求 1 所述的方法,其中所述多个第二多路转换器包括第一子集的多路转换器和第二子集的多路转换器,所述第一子集的多路转换器被设计为被放置在第一集成电路 IC 上,而所述第二子集的多路转换器被设计为被放置在第二集成电路 IC 上,并且其中所述第一子集包括一个或多个多路转换器,而其中所述第二子集包括一个或多个多路转换器。
3. 根据权利要求 1 所述的方法,其中所述多个第二多路转换器包括第一子集的多路转换器和第二子集的多路转换器,所述第一子集的多路转换器被设计为被放置在集成电路 IC 的第一部分上,而所述第二子集的多路转换器被设计为被放置在所述集成电路 IC 的第二部分上,并且其中所述第一子集包括一个或多个多路转换器,而其中所述第二子集包括一个或多个多路转换器。
4. 根据权利要求 1 所述的方法,其中所述第一多路转换器的表示是以寄存器传输级(RTL)网表内的 RTL 形式,并且其中转换被自动地执行。
5. 根据权利要求 1 所述的方法,其中该方法还包括:对集成电路的硬件描述语言(HDL)描述进行编译,以创建所述第一多路转换器的表示。
6. 根据权利要求 5 所述的方法,其中该方法还包括:复制所述第一多路转换器的译码逻辑。
7. 根据权利要求 1 所述的方法,该方法还包括:
扩展所述第一多路转换器的译码逻辑以将驱动所述第一多路转换器的选择线路的逻辑包括在内。
8. 根据权利要求 7 所述的方法,其中所述扩展在转换之前进行,并且其中所述扩展包括用于定义将被复制的控制逻辑的信号切断。
9. 根据权利要求 1 所述的方法,该方法还包括:
将每个划分内的划分中枢表示转换为 RTL(寄存器传输级)组件。
10. 一种设计多路转换器的机器,该机器包括:
用于接收第一多路转换器的表示的装置;
用于将所述第一多路转换器的表示分解为所述第一多路转换器的划分中枢形式的装置,其中所述划分中枢形式包括以下中的一者:(a)多个三态驱动器;(b)有线 OR 阵列;或者(c)有线 AND 阵列;
用于对所述第一多路转换器的划分中枢形式进行划分以创建多个第二多路转换器的装置。
11. 一种设计多路转换器的方法,该方法包括:
接收具有输出的第一多路转换器的表示;
将所述第一多路转换器的表示分解为多个第二多路转换器的表示,其中所述第二多路

转换器的输出被一起连接到公共输出,该公共输出与所述第一多路转换器的输出相匹配,且不存在任何中间多路转换器介于所述第二多路转换器的输出与所述公共输出之间;

将所述第二多路转换器在至少一个集成电路的部分之间进行划分。

12. 根据权利要求 11 所述的方法,其中所述多个第二多路转换器的表示是所述第一多路转换器的划分中枢形式。

13. 根据权利要求 11 所述的方法,其中所述第二多路转换器包括第一子集的多路转换器和第二子集的多路转换器,所述第一子集的多路转换器被划分为被放置在第一集成电路 IC 上,而所述第二子集的多路转换器被划分为被放置在第二集成电路 IC 上,并且其中所述第一子集包括一个或多个多路转换器,而其中所述第二子集包括一个或多个多路转换器。

14. 根据权利要求 11 所述的方法,其中所述第二多路转换器包括第一子集的多路转换器和第二子集的多路转换器,所述第一子集的多路转换器被划分为被放置在集成电路 IC 的第一部分上,而所述第二子集的多路转换器被划分为被放置在所述集成电路 IC 的第二部分上,并且其中所述第一子集包括一个或多个多路转换器,且所述第二子集包括一个或多个多路转换器。

15. 根据权利要求 11 所述的方法,其中所述第一多路转换器的表示是以寄存器传输级(RTL)网表内的 RTL 形式,并且其中所述分解被自动地执行。

16. 根据权利要求 11 所述的方法,其中所述多个第二多路转换器的表示是以下述中的一者的形式:(a) 多个三态驱动器;(b) 有线 OR 阵列;或者(c) 有线 AND 阵列。

17. 根据权利要求 11 所述的方法,其中该方法还包括:对集成电路的硬件描述语言(HDL)描述进行编译,以创建所述第一多路转换器的表示。

18. 根据权利要求 17 所述的方法,其中该方法还包括:复制所述第一多路转换器的译码逻辑。

19. 一种设计多路转换器的机器,该机器包括:

用于接收具有输出的第一多路转换器的表示的装置;

用于将所述第一多路转换器的表示分解为多个第二多路转换器的表示的装置,其中所述第二多路转换器的输出被一起连接到公共输出,该公共输出与所述第一多路转换器的输出相匹配,且不存在任何中间多路转换器介于所述第二多路转换器的输出与所述公共输出之间;

用于将所述第二多路转换器在至少一个集成电路的部分之间进行划分的装置。

20. 一种设计多路转换器的方法,该方法包括:

接收能够进行划分中枢分解的第一逻辑组件的表示;

将所述第一逻辑组件的表示分解为所述第一逻辑组件的划分中枢形式,其中所述划分中枢形式包括以下中的一者:(a) 多个三态驱动器;(b) 有线 OR 阵列;或者(c) 有线 AND 阵列;

对所述第一逻辑组件的所述划分中枢形式进行划分以创建多个逻辑组件,所述多个逻辑组件一起提供所述第一逻辑组件的输出结果。

21. 根据权利要求 20 所述的方法,其中所述第一逻辑组件是 N- 输入加法器。

22. 根据权利要求 21 所述的方法,其中转换将所述 N- 输入加法器表示为具有一组导线的总线。

用于设计多路转换器的方法和设备

技术领域

[0001] 本发明主要涉及集成电路设计领域,并且更特别地,涉及通过机器实施的自动过程进行集成电路设计,所述自动过程例如为从描述集成电路(例如使用电路的硬件描述语言进行描述)开始的综合过程。

背景技术

[0002] 对于 VLSI(超大规模集成电路)工艺级别的电路设计,设计者通常采用计算机辅助设计技术。已经开发了诸如硬件描述语言(HDL)的标准语言来辅助进行复杂电路(例如复杂的数字或模拟电路)的设计和模拟。若干种硬件设计语言,例如 VHDL 和 Verilog,已经演进为工业标准。VHDL 和 Verilog 是通用硬件描述语言,其允许使用抽象数据类型来在门级、寄存器传输级(RTL)或者行为级对硬件模型进行定义。随着器件工艺持续发展,已经开发了各种产品设计工具来使 HDL 适用于更新的器件和设计风格。

[0003] 当使用 HDL 编译器设计电路时,设计者通常用 HDL 源代码来描述电路元件,并且然后编译源代码以生成合成 RTL 网表(netlist)。RTL 网表与电路元件的示意性表示(representation)相对应。包含合成电路元件的电路通常被优化以改进时序关系并去除不必要的或冗余的元件。这种优化通常包括替换电路中的不同门类型、或组合和去除电路中的门。可以在以下过程之前执行所述优化:将诸如 RTL 网表的 RTL 描述映射到集成电路的特定预定体系结构,例如现场可编程门阵列(FPGA)的目标体系结构。本领域所公知的是,不同厂商的 FPGA 通常具有不同的体系结构,不同之处在于用于实施某个功能的特定集成电路内的电路在不同厂商之间可能并不相同。具有独特体系结构的 FPGA 厂商的示例包括 Xilinx 和 Altera。在美国专利 6,438,735、6,449,762 以及 6,973,632 中描述了用于进行集成电路的计算机辅助设计的不同方法和系统,在此合并这些申请作为参考。

[0004] 通常,期望在多于一个集成电路中实施设计,例如在多于一个 FPGA 集成电路中实施设计。这在设计比较复杂且庞大以至于不适于单个 FPGA 上的可用资源时通常是有必要的。这对于 ASIC(专用集成电路)的设计者来说也是有用的。越来越多的 ASIC 设计者通常通过在若干个 FPGA 中实施设计来制作其设计的样品并测试其设计。ASIC 设计的复杂度可以高到需要多个 FPGA 来实施该设计所需要的所有功能。这意味着设计者必须努力将其设计划分(partition)到多个 FPGA 中,并且现有技术中存在若干种工具和方法来实现这一过程。在美国专利 6,438,735 中描述了通过自动过程来进行划分的技术的一个示例,该专利还描述了用于将设计划分到多个 FPGA 上的其它现有技术。在 RT 组件级执行自动划分的软件工具的一个示例是来自加利福尼亚州森尼维耳市 Synplicity 股份有限公司的产品 Certify(验证)。在工艺映射网表级执行自动划分的工具的一个示例是来自 Auspy 发展股份有限公司的 Auspy 划分系统 II。

[0005] 即使使用现有的用于划分的技术,也难以对能够具有极大宽度并且可能具有许多个输入(其中一个输入被选择以驱动多路转换器(multiplexer)的输出)的多路转换器(MUX)进行划分。多路转换器是公知的电路元件,而图 1A 中示出了多路转换器的一个示例。

图 1A 示出了电路 10, 该电路 10 包括数据驱动器 12 和 14, 该驱动器 12 和 14 将时钟控制的数据驱动到多路转换器 20 的输入 16 和 18 中。可以理解的是, 可存在两个或多个到多路转换器 20 的输入。例如, 可以有 20 个输入, 每个输入由数据驱动器驱动到多路转换器 20 中。多路转换器 20 在其输出 22 处产生输出值, 并且该输出值由选择线路 24 上出现的数据来决定, 该数据使得多路转换器选择所述输入中的一个输入以连接到所述多路转换器的输出 22。

[0006] 对较大的多路转换器进行划分通常需要创建至少三个较小的多路转换器。这些较小的多路转换器中的两个多路转换器用于各从两组不同的输入中选择一个输入 (以产生两个选定输入), 并且然后由第三多路转换器在这两个选定输入之间进行选择。图 1B 和 1C 示出了这种将较大的多路转换器分为三个较小的多路转换器的示例。

[0007] 在图 1B 中示出了对于设计 30 的情况, 多路转换器已经被分为位于两个不同的集成电路 32 和 34 上的三个多路转换器 40、42 以及 44。两个不同的集成电路上的多路转换器被多线总线 64 和导线 66 耦合到一起。实际上, 多路转换器 40 和 44 在到原始多路转换器的两组不同的输入之间进行选择, 由此提供两个输出, 由多路转换器 42 在这两个输出之间进行选择。所述多路转换器中的每个多路转换器受到选择逻辑或选择线路的控制, 该选择逻辑或选择线路使得选择的输入以本领域已知的方式连接到多路转换器的相应输出。选择线路通常由本领域已知的译码逻辑驱动。如图 1B 所示, 逻辑 50 和 56 提供到多路转换器 40 的输入, 而逻辑 54 通过总线 62 (其包括多条线路) 提供到多路转换器 44 的输入。多路转换器 44 还接收来自总线 61 (其包括多条线路) 的另一输入。逻辑 50 通过总线 60 被耦合到多路转换器 40, 而逻辑 56 通过总线 64 被耦合到多路转换器 40。多路转换器 40 的输出被耦合到多路转换器 42 的多个输入中的一个输入, 而多路转换器 44 的输出被耦合作为多路转换器 42 的另一输入, 该多路转换器 42 在这两个输入之间选择以通过线路 22 驱动该多路转换器 42 的输出, 以驱动逻辑 52。对于多路转换器 40、42 以及 44 中的每个多路转换器, 每个数据输入和每个数据输出均是总线 (其包括多条线路), 并且由多路转换器进行的切换过程在输入总线之间进行切换。可以理解的是, 图 1B 中示出的示例是多种逻辑元件和多路转换器可以在两个或多个集成电路之间被布置或划分的多种不同方式中的一种方式。但是, 在每种情况中, 驱动多路转换器 42 的输入必须相对于多路转换器 42 而被分组到一起, 以保持由划分后的多路转换器实施的逻辑。换句话说, 驱动多路转换器 42 的输入不能被用于驱动多路转换器 40 或 44, 而是必须被用于驱动树形 (tree) 驱动器中的最后一个多路转换器, 以维持划分后的多路转换器的正确的逻辑功能, 从而提供与原始多路转换器相同的逻辑复用。图 1C 示出了这种关系, 其中省去了多个集成电路的复杂关系。特别地, 输入 70 和 71 必须被划分为用于多路转换器 44。因此, 例如用于所述输入 70 和 71 的驱动器必须被划分为用于多路转换器 44。这使得划分操作变得复杂。

[0008] 如果在划分为芯片之前, 在包含较大多路转换器的电路上执行逻辑合成, 则所得到的电路将根据由逻辑合成系统选择的多路转换器分解而对多路转换器的驱动器进行分组。许多不同分解都是可行的: 较小的树形多路转换器、与 - 或分解以及使用 FPGA 专用组件进行的专用分解。如果分解后的多路转换器在芯片之间被划分, 则所选择的分解可能会引起在划分之间使用过多的交叉连接。如果另一方面, 在多路转换器被分解之前进行了划分, 则连接到多路转换器的所有信号必须在单个芯片中获得, 这也会引起划分之间过多的

交叉连接。一个偏好的解决方案是将多路转换器组件分片为单个比特宽度的多路转换器。这一偏好的解决方案在 Certify 中被实施。这并不有助于对具有较大数量输入的多路转换器进行划分,并且对于将在片段之间被共享的多路转换器来说,还将产生对多路转换器的译码逻辑难以划分的问题。需要一种同时用于多路转换器分解和划分的解决方案。

发明内容

[0009] 本发明公开了用于设计集成电路的方法和设备。根据本发明的一个方面,用于设计集成电路的一种示例方法包括接收第一多路转换器的表示;将所述第一多路转换器的表示转换为所述第一多路转换器的划分中枢(neutral)表示;以及对所述第一多路转换器的划分中枢表示进行划分以创建多个第二多路转换器。在一个实施方式的一种实施中,所述多个第二多路转换器包括第一子集的多路转换器和第二子集的多路转换器,其中所述第一子集的多路转换器被设计为被放置在第一集成电路(IC)上,而所述第二子集的多路转换器被设计为被放置在第二 IC 上,并且其中所述第一子集包括一个或多个多路转换器,而其中所述第二子集包括一个或多个多路转换器。在其它实施中,所述多个第二多路转换器包括第一子集的多路转换器和第二子集的多路转换器,所述第一子集的多路转换器被设计为被放置在集成电路(IC)的第一部分上,而所述第二子集的多路转换器被设计为被放置在所述 IC 的第二部分上。所述第一多路转换器的表示可以是以寄存器传输级(RTL)网表内的 RTL 形式,并且其中所述转换由机器自动地执行而不需要人工干预。在至少一些实施方式中,所述划分中枢表示可以包括下述中的一者或多者:(a) 多个三态驱动器;(b) 有线 OR(或)阵列;或者(c) 有线 AND(与)阵列。在至少一些实施方式中,所述方法还可以包括:对集成电路的 HDL 描述进行编译以创建所述第一多路转换器的表示,并且所述方法还可以包括复制所述第一多路转换器的译码逻辑以提供用于所述第二多路转换器的选择线路。

[0010] 根据本发明的另一方面,一种示例方法包括接收具有输出的第一多路转换器的表示;将所述第一多路转换器的表示分解为多个第二多路转换器的表示,其中所述第二多路转换器被一起耦合到公共输出,该公共输出与所述第一多路转换器的输出相匹配,且不存在任何中间多路转换器介于所述第二多路转换器与所述公共输出之间;以及将所述第二多路转换器在至少一个集成电路的部分之间进行划分。在至少一些实施方式中,所述多个第二多路转换器的表示是所述第一多路转换器的划分中枢形式。所述划分可以在多个集成电路之间或者在同一集成电路的不同部分(例如,平面布置部分)之间进行。在至少一些实施方式中,所述第一多路转换器的表示可以是以 RTL 网表内的寄存器传输级形式,并且其中所述分解可以被自动地执行而不需要人工干预。在至少一些实施方式中,所述多个第二多路转换器的表示可以是以下述中的一者或多者的形式:(a) 多个三态驱动器;(b) 有线 OR 阵列;或者(c) 有线 AND 阵列。虽然本发明的一些实施方式是相对于多路转换器进行描述的,但是也可以将其应用于允许划分中枢分解的任何组件。例如,N-输入加法器可以被表示为由导线组成的抽象总线,该总线将其驱动器的值相加。在进行划分之前,N-输入加法器将被分解为这种总线,其中 N 大于 2。在进行划分之后,如果所述加法器的若干个输入在同一划分中,则这些输入被变换到 M-输入 RTL 加法器,其中 M 是该划分中的原始加法器输入的数量。可以通过将所估计出的加法器的区域分配给其驱动器来获得对用于划分的区域的较好逼近。划分软件将需要较小改变,以将目的划分处的输入数考虑在内,因为目的划分将具

有与源划分的输出计数的总数相等的输入数,因此将在该目的划分处进行最终的汇总。由于使用多路转换器的划分,同时进行的分解和划分解方案可以大大减少导线数。

[0011] 本发明还公开了诸如能够根据本发明的一个或多个实施方式来设计集成电路的数字处理系统的设备。本发明还公开了诸如计算机可读媒介或机器可读媒介的设备,所述计算机可读媒介或机器可读媒介可以用于根据这里描述的方法通过使得诸如数据处理系统的机器执行这里描述的一种或多种方法来设计集成电路。

[0012] 通过附图和以下详细描述,将会清楚本发明的其它特征。

附图说明

[0013] 本发明通过示例来说明,并且不限于附图中的图示,其中相同的参考标记指示类似的元件。

[0014] 图 1A 示出了包括多路转换器的电路的一个示例;

[0015] 图 1B 示出了通过将单个多路转换器划分为若干个多路转换器而得到的一组多路转换器的布置的一个示例;

[0016] 图 1C 示出了通过将一个多路转换器划分为若干个多路转换器而得到的一组多路转换器的另一个示例;

[0017] 图 2A 示出了根据本发明的一个实施方式在被划分之前的多路转换器的一个示例;

[0018] 图 2B 示出了根据本发明的一个实施方式的多路转换器的划分中枢形式的一个示例;

[0019] 图 2C 示出了对图 2B 中示出的划分中枢多路转换器进行划分的结果的一个示例,其中图 2C 所示的划分是在两个集成电路上进行的,在可替换实施方式中,所述划分可以在同一集成电路的部分(例如同一个集成电路的平面布置部分)上进行;

[0020] 图 3 是示出了根据本发明的一个实施方式的一种方法的一个示例的流程图;

[0021] 图 4 示出了根据本发明的一个实施方式的机器可读介质的一个示例;以及

[0022] 图 5 是可以用于实施本发明的一个或多个实施方式的数据处理系统的框图示例。

具体实施方式

[0023] 这里描述用于设计一个集成电路或多个集成电路的方法和设备。在以下描述中,出于解释说明的目的,提出了许多具体细节以提供本发明的全面理解。但是,对于本领域技术人员来说,显然本发明可以在缺少这些特定细节的情况下实现。在其它示例中,公知的结构、过程以及装置以框图的形式示出,并且以概括的方式提及,从而在缺少过多细节的情况下提供解释。

[0024] 现在参考图 2A、图 2B 以及图 2C,提供本发明的一些实施方式的主要示例。虽然本发明的实施方式要用于 HDL 设计合成软件中,但是本发明不必被限制为这些用途。虽然使用其它语言和计算机程序是可行的,但是本发明的实施方式将以使用 HDL 合成系统(并且更特别地,那些设计用于具有厂商特定技术/体系结构(例如 FPGA 的厂商的特定体系结构)的集成电路的系统)为背景而进行描述。但是,可以理解的是,本发明的一些实施方式还可以采用其它类型的集成电路,例如 ASIC。

[0025] 图 2A 示出了具有四个输入 202、并提供一个输出 203 的多路转换器 201。在工作时,多路转换器 201 选择所述四个输入中的一个输入作为用于驱动输出 203 的输入。特定输入的选择受到选择线路 204 的控制,该选择线路 204 通常由本领域已知的译码逻辑驱动。根据本发明的至少一些实施方式,多路转换器 201 可以从其作为单个多路转换器的初始表示被转换或分解为所述多路转换器的划分中枢表示,从而提供比多路转换器的原始表示更容易被划分的表示。

[0026] 图 2B 示出了原始多路转换器 201 的划分中枢表示的一个示例。该特定划分中枢表示通过将三态驱动器的输出一起耦合到公共输出处而被形成,以提供原始多路转换器的输出 203。三态驱动器中的每个驱动器接收原始的四个输入 202 中的一个输入,并且所述三态驱动器中的每个驱动器包括被耦合到相应译码逻辑的启动 (enable) 输入。相应的译码逻辑被轮流耦合到原始选择线路 204 中的所有选择线路或其子集。如图 2B 所示,三态驱动器 217 接收 00 输入,并且通过三态驱动器 217 的启动输入被耦合到译码逻辑 225。类似地,三态驱动器 215 接收输入 01,并且通过三态驱动器 215 的启动输入被耦合到译码逻辑 227。三态驱动器 213 被耦合以接收 10 输入,并且通过三态驱动器 213 的启动输入被耦合到译码逻辑 229,而三态驱动器 211 被耦合到输入 11,并且还通过三态驱动器 211 的启动输入被耦合到译码逻辑 231。这些三态驱动器中的全部四个驱动器的输出被耦合到一起以提供输出 203。

[0027] 每个三态驱动器的译码逻辑被实施为使得所述三态驱动器中的仅一个三态驱动器将驱动所述输出,而其它三态驱动器将被他们的耦合到相应译码逻辑的启动输入设定为高阻抗状态。例如,如果选择线路指定 00 输入将被选择以通过多路转换器输出,则译码逻辑 225 将使三态驱动器 217 被设定为使得三态驱动器 217 的输入驱动其输出,并且由此驱动输出 203,这是因为所有其它三态驱动器 215、213 以及 211 将其启动输入设定为使得其输出处于高阻抗状态。例如,译码逻辑 227 使三态驱动器 215 的启动输入将三态驱动器 215 的输出设定为其高阻抗状态。类似地,译码逻辑 229 和译码逻辑 231 分别使三态驱动器 213 和 211 处于类似的状态。这使得来自三态驱动器 217 的输出能够驱动输出 203。

[0028] 使用这种划分中枢设计,自动设计系统不需要担心输入与多路转换器的特定部分之间的匹配。如现有技术那样,译码逻辑必须继续将其正在译码的输入考虑在内,而这与现有技术并无不同。这种划分中枢表示,例如如图 2B 所示的形式,可以被分解为分散在不同集成电路之间、或者甚至如图 2C 所示的相同集成电路的不同部分的若干个多路转换器。可以看出,这些分散的多路转换器的输出被一起耦合到公共输出,该公共输出与原始多路转换器的输出相匹配,而且不存在任何中间多路转换器介于在被划分的多路转换器与所述公共输出之间。换句话说,不存在在第一级多路转换器的输出与树形多路转换器的最后一个输出之间介入多路转换器的树形多路转换器。这也可以通过将类似于树的图 1C 所示的多路转换器的集合与现在将描述的图 2C 所示的多路转换器的集合进行比较而得出。

[0029] 图 2C 示出了将图 2B 所示的多路转换器的划分中枢表示在两个集成电路 251 和 253 之间进行划分的示例。三态驱动器 217 和 215 已经与其相应的译码逻辑一起被放置在集成电路 251 上,并且被集成电路 251 上的逻辑 261 驱动。同时,三态驱动器 213 和 211 已经与其相应的译码逻辑一起被放置在集成电路 253 上,该集成电路 253 具有逻辑 263,该逻辑 263 提供用于三态驱动器 213 和 211 的输入。这些三态驱动器中的所有四个三态驱动器

的输出被一起耦合到所述两个集成电路外侧的印刷电路板（或其它基底）上的导线上，该导线将所述两个集成电路耦合以提供输出 203。取决于设计，该输出可以依次返回到一个或多个集成电路，以驱动其他逻辑。可以理解的是，在至少一些实施方式中，通常处于集成电路的边界处的三态驱动器，例如驱动集成电路上的焊接点的三态驱动器，可以用于实施多路转换器的划分中枢表示。特别地，如果各种集成电路上存在足够的输入 / 输出管脚可用于这种用途，则集成电路边界处的三态驱动器可以用于形成划分中枢表示。

[0030] 可以理解的是，存在可以使用有线 OR 阵列或有线 AND 阵列的可替换划分中枢表示来实施多路转换器的划分中枢表示。

[0031] 多路转换器的译码逻辑可以被扩展以包括为多路转换器的控制线路提供信号的逻辑。这尤其有利于在比直接连接到多路转换器的控制线路的多个信号更窄的信号切断 (cut) 通过先前的逻辑时、或者在新的切断可以被其它多路转换器共享时的情况。然后每个输入均被这一新的、可能更窄的或者共享程度更高的切断的布尔函数启动。当创建了多路转换器的划分中枢分解形式时，所述分解的每个片段都将连接到所需控制信号，并且承载布尔函数。在大多数情况下，将控制或译码逻辑复制到多个划分中的区域开销将远不如通过减少必须在划分之间传送的控制信号的数量而实现划分之间的交叉连接的节省重要。

[0032] 图 3 示出了根据本发明的一个实施方式的方法的一个示例。在操作 301 中，创建电路设计的表示。例如，可以通过将电路设计的 HDL 描述编译为包括至少一个多路转换器的 RTL 网表来创建所述表示。在操作 303 中，可以将多路转换器转换为多路转换器的划分中枢形式。例如，可以将所述多路转换器从其 RTL 形式转换为使用三态驱动器来创建具有划分中枢形式的多路转换器的形式。可选地，可以使用集成电路边界处的三态驱动器来实施这一形式。在其它实施方式中，可以使用有线 OR 阵列或有线 AND 阵列来实施多路转换器的划分中枢形式。在操作 305 中，将多路转换器的划分中枢形式划分为若干个多路转换器，以使得该若干个多路转换器可以被放置在不同集成电路上或者同一集成电路上不同部分中，例如同一个集成电路上不同平面布置区域。然后在操作 307 中，可以将译码逻辑复制以用于划分后的逻辑。在操作 307 之后，可以执行电路的放置和布线 (route)，以及其它操作，例如优化操作以及包括将 RTL 映射到工艺相关 (dependent) 的体系结构的操作，所述体系结构例如为集成电路（例如现场可编程门阵列集成电路）的目标体系结构。

[0033] 图 4 示出了可用于实施本发明的一种或多种方法的机器可读介质或计算机可读介质的一个示例。例如，机器可读介质可以包括可执行计算机程序指令，当由数据处理系统执行该计算机程序指令时，其使得数据处理系统执行这里描述的一种或多种方法。机器或计算机可读介质可以被布置在 CD-ROM 或 DVD-ROM 或磁性硬盘或其它盘或者存储装置上。所述介质可以包括多个组件以及图 4 的介质 400 中未示出的其它组件。在图 4 所示的示例中，用于设计的 HDL 资源描述如标记 401 所示。HDL 编译器 403 包括可执行程序指令，该可执行程序指令将 HDL 资源编译为例如工艺无关的 RTL 网表 405。划分程序 407 可以用于划分所述工艺无关的 RTL 网表，从而所述设计的不同组件被划分在多个集成电路上。所述划分程序可以包括或调用划分中枢转换程序 409，其可以实施例如图 3 所示的方法以创建多路转换器的划分中枢形式，并且然后将所述形式划分为多个集成电路或者同一集成电路的多个部分（例如同一个集成电路的多个平面布置部分）上的若干个多路转换器。介质 400 还可以包括工艺映射程序 413，其将来自己已经被划分的工艺无关的 RTL 网表的划分后的设计

映射到工艺相关的网表,然后该工艺相关的网表还可以通过常规放置或布线工具被处理以创建最终的网表、或者用于特定集成电路厂商的目标体系结构的设计,所述目标体系结构例如为 Xilinx 的 FPGA 的目标体系结构。介质 400 还可以包括优化程序 411,可以在所述网表已经被划分之后或者在该网表被划分之前利用所述最优程序 411 来优化所述设计。

[0034] 图 5 示出了可以使用介质 400 来执行这里描述的一种或多种方法的数据处理系统的一个示例。数据处理系统 500 包括一个或多个处理器 501,所述一个或多个处理器 501 通过总线 507 被耦合到存储器 503。处理器 501 和存储器 503 还被耦合到海量存储装置 505,该海量存储装置 505 可以代表硬盘或其它非易失性或易失性存储装置。在一个实施方式中,所述海量存储装置可以是包括图 4 所示的介质 400 的磁性硬盘。所述数据处理系统还包括显示装置 509、一个或多个输入装置 511(包括例如键盘、光标控制装置)以及诸如以太网接口或无线接口的网络接口。可以理解的是,图 5 所示的数据处理装置的体系结构仅被提供用作说明的目的,并且结合本发明使用的计算机系统或其它数据处理系统并不限于这种特定体系结构。

[0035] 在前述说明书中,已经参考本发明的特定示例性实施方式对本发明进行了描述。显然可以对本发明进行不偏离所附权利要求书所提出的本发明的较宽的实质和范围的各种修改。因此,说明书和附图被认为是说明性的理解而不是限制性的理解。

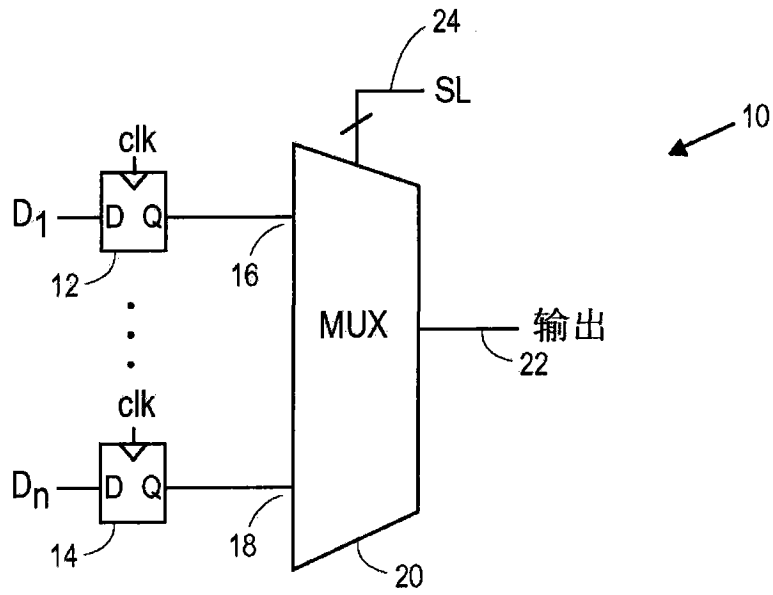


图 1A

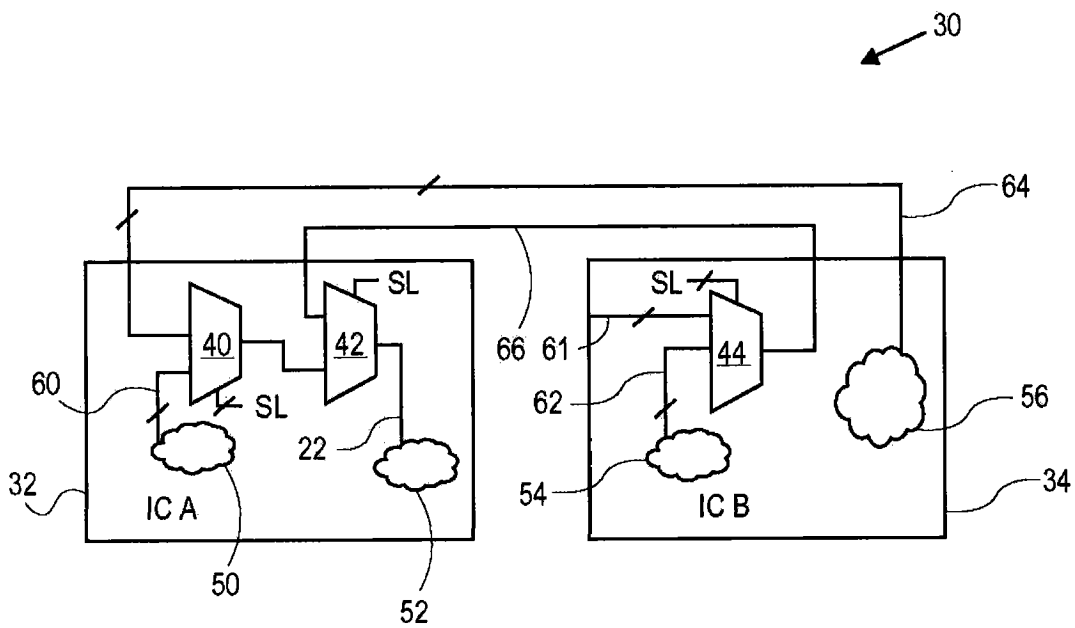


图 1B

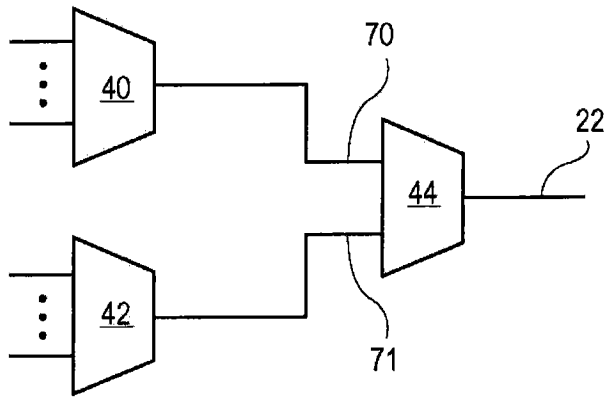


图 1C

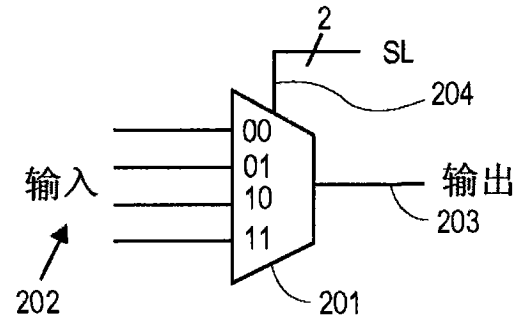


图 2A

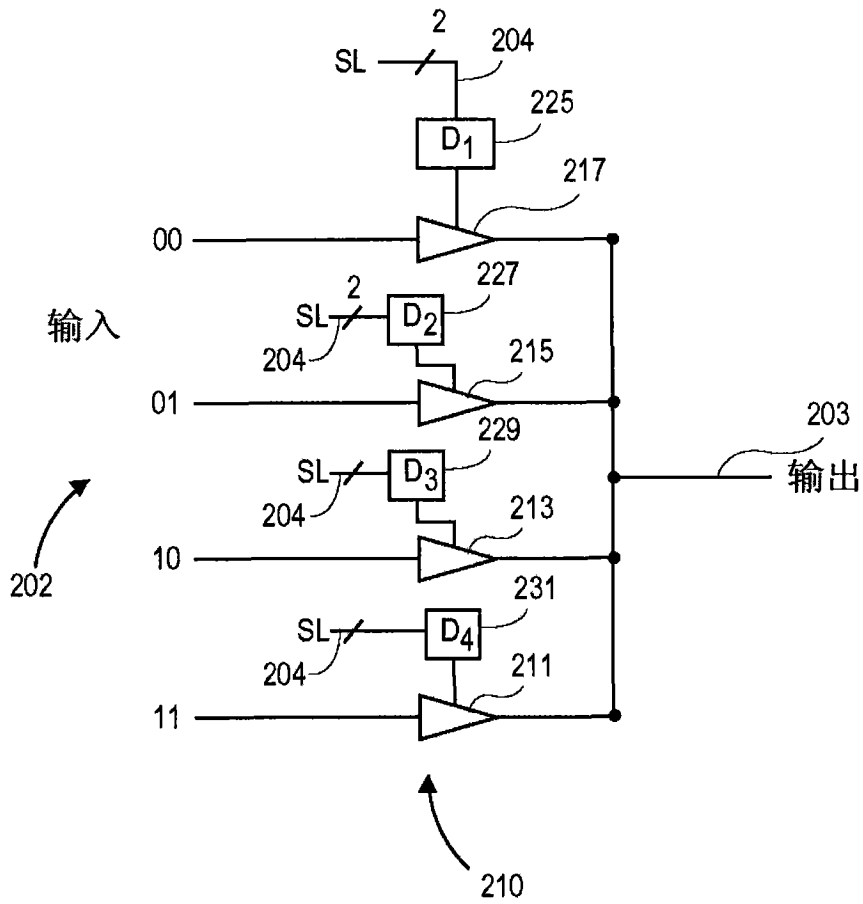


图 2B

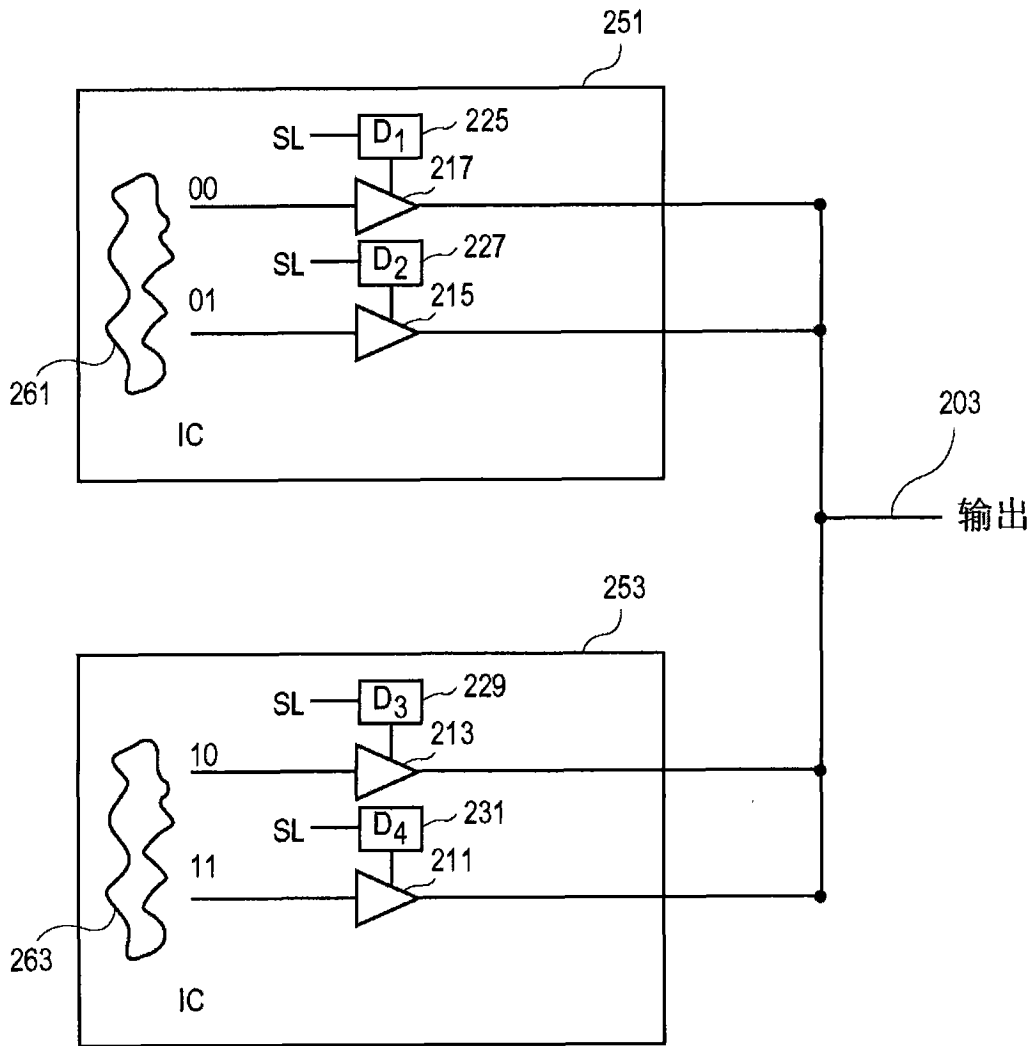


图 2C

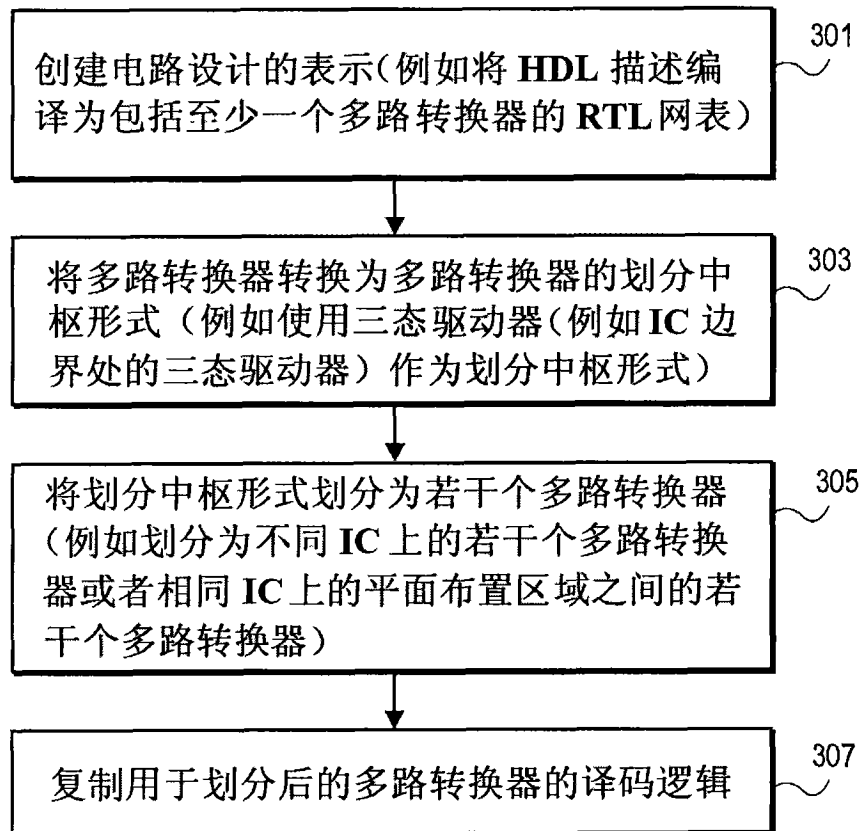


图 3

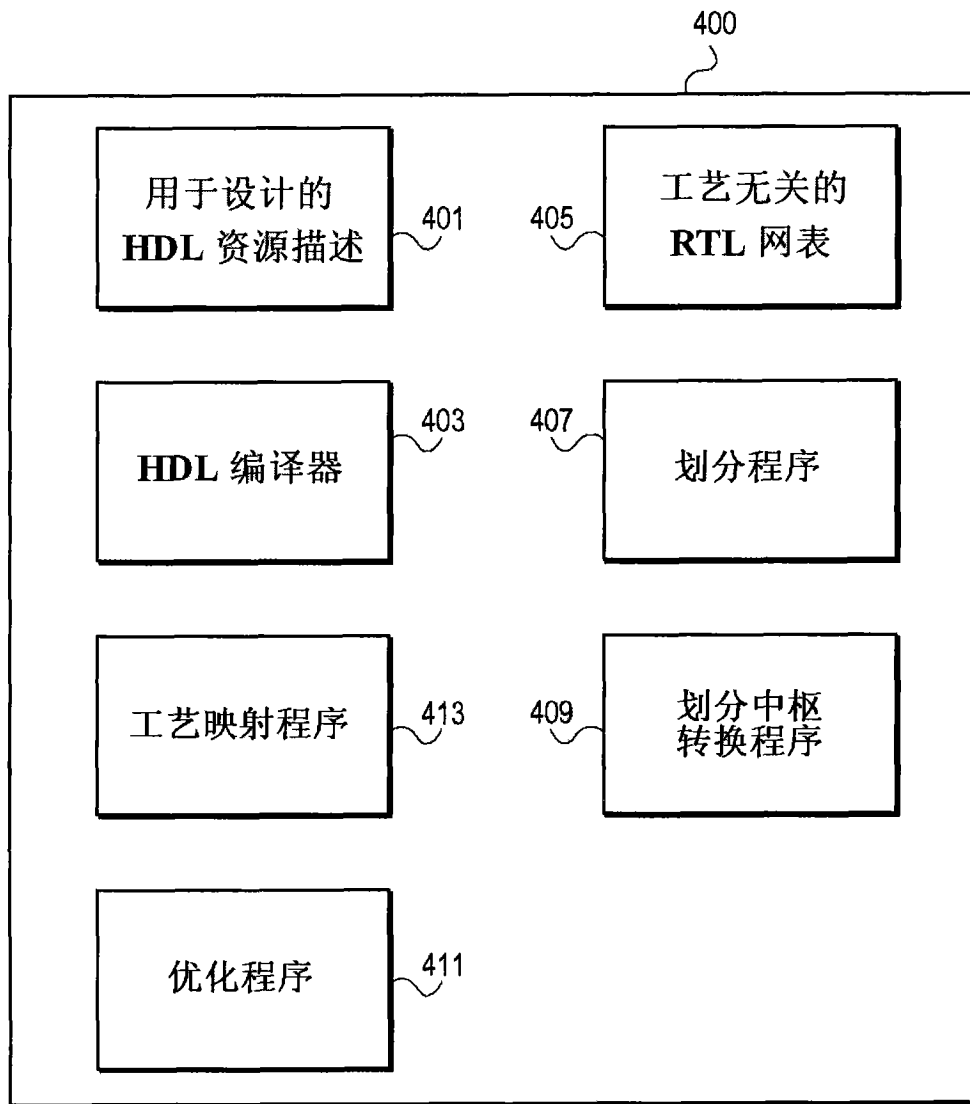


图 4

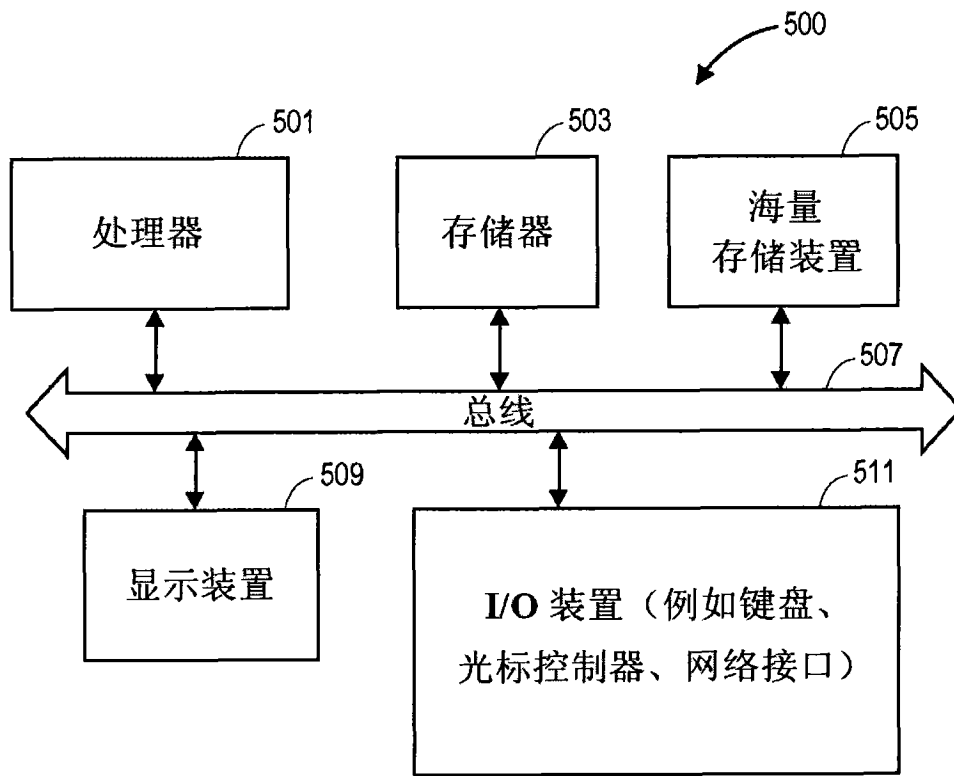


图 5