(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 116896559 A
(43)申请公布日 2023.10.17

(21)申请号 202310294507.1

(22)申请日 2023.03.23

(30)优先权数据
17/709,111 2022.03.30 US

(71)申请人 辉达公司
地址 美国加利福尼亚州

(72)发明人 G·迪尔思 M·胡梅尔 姜楠
G·索尔森

(74)专利代理机构 北京市磐华律师事务所 11336
专利代理师 林琳

(51)Int.Cl.
H04L 67/10(2022.01)
H04L 45/74(2022.01)
H04L 47/72(2022.01)
H04L 41/16(2022.01)

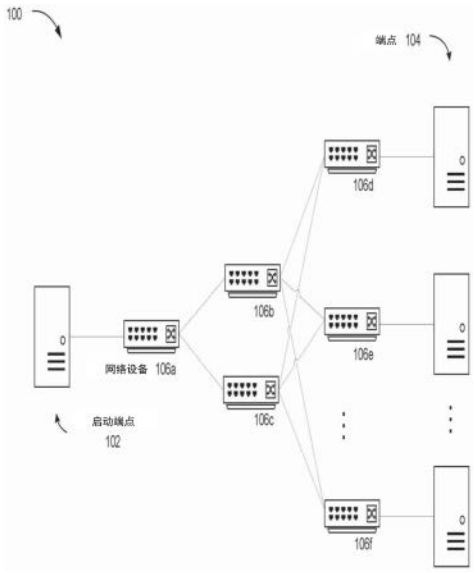
权利要求书3页 说明书65页 附图51页

(54)发明名称

由网络设备协助的多播归约

(57)摘要

本公开涉及由网络设备协助的多播归约。用于执行多播归约操作的系统和技术。在至少一个实施例中,网络设备接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据。该网络设备预留资源以处理将从端点接收的第二网络数据,并将第一网络数据发送到多个附加的网络设备。网络设备接收第二网络数据,并使用预留的资源处理第二网络数据。



CN 116896559 A

1.一种网络设备,包括:

至少一个处理器;以及

至少一个存储器,其包括指令,所述指令响应于由所述至少一个处理器执行,使所述网络设备:

接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据;

预留所述网络设备的资源,以处理将从所述多个端点接收的第二网络数据,其中要预留的资源的量至少部分地基于从所述第一网络数据的报头获得的信息来确定;

将所述第一网络数据发送至多个附加的网络设备,所述多个附加的网络设备至少部分地基于从所述报头获得的所述信息被识别;

接收所述第二网络数据;以及

使用所述预留的资源来处理所述第二网络数据。

2.根据权利要求1所述的网络设备,其中所述报头包括指示所述第一网络数据和端点设备的虚拟存储器空间之间的映射的信息。

3.根据权利要求1所述的网络设备,其中所述多个端点中的端点包括并行处理单元,并且其中所述第一网络数据的至少一部分被写入所述并行处理单元的存储器。

4.根据权利要求1所述的网络设备,其中所述第一网络数据响应于对并行处理单元的存储器的写操作而被发送至所述网络设备,其中所述第一网络数据包括由所述写操作写入所述存储器的数据。

5.根据权利要求1所述的网络设备,其中处理所述第二网络数据包括:至少部分地基于从所述报头获得的归约信息来对所述第二网络数据进行归约。

6.根据权利要求1所述的网络设备,其中所述网络设备将所述第二网络数据的归约发送至所述第一网络数据的发送者。

7.根据权利要求1所述的网络设备,所述至少一个存储器包括进一步的指令,所述指令响应于由所述至少一个处理器执行,使所述网络设备:

存储从所述第一网络数据的所述报头获得的所述信息;以及

响应于接收到所述第二网络数据来检索所述信息。

8.根据权利要求1所述的网络设备,其中所述报头包括用于所述多播操作的归约信息和路由信息。

9.根据权利要求1所述的网络设备,所述至少一个存储器包括进一步的指令,所述指令响应于由所述至少一个处理器执行,使所述网络设备:

在向所述多个附加的网络设备发送所述第一网络数据之前,更新所述报头中的归约信息和路由信息。

10.根据权利要求1所述的网络设备,其中所述多个附加的网络设备包括交换机、路由器或端点中的至少一个。

11.根据权利要求1所述的网络设备,所述至少一个存储器包括进一步的指令,所述指令响应于由所述至少一个处理器执行,使所述网络设备:

响应于确定自从发送所述第一网络数据起已经过了阈值时间量,并且所述附加的网络设备中的至少一个附加的网络设备尚未对接收到所述第一网络数据作出响应,释放所述预留的资源。

12.一种非暂时性机器可读介质,其上存储有指令,所述指令响应于由一个或更多个处理器执行,使所述一个或更多个处理器至少:

在网络设备处接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据;

预留所述网络设备的资源,以处理将从所述多个端点接收的第二网络数据,其中要预留的资源至少部分地基于从所述第一网络数据获得的信息来确定;

将所述第一网络数据发送至多个附加的网络设备,所述多个附加的网络设备至少部分地基于从所述第一网络数据获得的所述信息被识别;

接收所述第二网络数据;以及

使用所述预留的资源来处理所述第二网络数据。

13.根据权利要求12所述的非暂时性机器可读介质,其中所述第一网络数据包括一个或更多个报头,所述一个或更多个报头包括指示所述第一网络数据和端点设备的虚拟存储器空间之间的映射的信息。

14.根据权利要求12所述的非暂时性机器可读介质,其中所述多个端点中的端点包括并行处理单元,并且其中所述第一网络数据的至少一部分由所述并行处理单元用于执行所述多播操作的至少一部分。

15.根据权利要求12所述的非暂时性机器可读介质,其中所述网络设备接收响应于对端点上的并行处理单元的存储器的读或写操作中的至少一个而发送的第一网络数据。

16.根据权利要求12所述的非暂时性机器可读介质,其中处理所述第二网络数据包括:至少部分地基于从所述第一网络数据中的一个或更多个报头获得的归约信息来对所述第二网络数据进行归约。

17.根据权利要求12所述的非暂时性机器可读介质,其上存储有进一步的指令,所述指令如果由一个或更多个处理器执行,则使所述一个或更多个处理器至少:

存储从所述第一网络数据的所述报头获得的所述信息;

响应于接收到所述第二网络数据来检索所述信息;以及

使用所述信息来处理所述第二网络数据。

18.根据权利要求12所述的非暂时性机器可读介质,其中所述第一网络数据中的一个或更多个报头包括用于所述多播操作的归约信息和路由信息。

19.根据权利要求12所述的非暂时性机器可读介质,其上存储有进一步的指令,所述指令如果由一个或更多个处理器执行,则使所述一个或更多个处理器至少:

在向所述多个附加的网络设备发送所述第一网络数据之前,更新所述报头中的归约信息和路由信息。

20.根据权利要求12所述的非暂时性机器可读介质,其上存储有进一步的指令,所述指令如果由一个或更多个处理器执行,则使所述一个或更多个处理器至少:

至少部分地基于确定自从发送所述第一网络数据起已经过了阈值时间量,并且所述附加的网络设备中的至少一个附加的网络设备尚未对接收到所述第一网络数据作出响应,丢弃与所述多播操作相关联的网络数据。

21.一种方法,包括:

在网络设备处接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据;

预留所述网络设备的资源,以处理将从所述多个端点接收的第二网络数据,其中要预

留的资源至少部分地基于从所述第一网络数据获得的信息来确定；

将所述第一网络数据从所述网络设备发送至多个附加的网络设备，所述多个附加的网络设备至少部分地基于从所述第一网络数据获得的信息被识别；

在所述网络设备处，接收所述第二网络数据；以及

由所述网络设备使用所述预留的资源来处理所述第二网络数据。

22.根据权利要求21所述的方法，其中所述第一网络数据包括一个或更多个报头，所述一个或更多个报头包括指示所述第一网络数据和端点设备的虚拟存储器空间之间的映射的信息。

23.根据权利要求21所述的方法，其中所述多个端点中的端点包括并行处理单元，并且其中所述第一网络数据的至少一部分由所述并行处理单元用于执行所述多播操作的至少一部分。

24.根据权利要求21所述的方法，其中所述网络设备接收响应于对端点上的并行处理单元的存储器的读或写操作中的至少一个而发送的第一网络数据。

25.根据权利要求21所述的方法，进一步包括：

至少部分地基于从所述第一网络数据中的一个或更多个报头获得的归约信息来处理所述第二网络数据。

26.根据权利要求21所述的方法，进一步包括：

存储从所述第一网络数据的所述报头获得的所述信息；

响应于接收到所述第二网络数据来检索所述信息；以及

使用所检索的信息来处理所述第二网络数据。

27.根据权利要求21所述的方法，进一步包括：

识别拓扑中的周期；以及

至少部分地基于对所述周期的识别来预留所述资源。

28.根据权利要求21所述的方法，进一步包括：

在向所述多个附加的网络设备发送所述第一网络数据之前，更新所述报头中的归约信息和路由信息。

29.根据权利要求21所述的方法，进一步包括：

响应于确定自从发送所述第一网络数据起已经过了阈值时间量，并且所述附加的网络设备中的至少一个附加的网络设备尚未对接收到所述第一网络数据作出响应，释放所述预留的资源。

30.根据权利要求21所述的方法，进一步包括：

确定所述网络设备的资源不足以可用于处理将从所述多个端点接收的所述第二网络数据；以及

保持所述第一网络数据，直到有足够的资源可用。

由网络设备协助的多播归约

技术领域

[0001] 至少一个实施例涉及执行由网络设备协助的多播归约操作。

背景技术

[0002] 多播归约(multicast-reduction)操作可用于在若干端点(endpoint)设备之间分配计算负载,并将来自端点设备的结果组合为完整的结果。可以提高多播归约操作的效率。

附图说明

[0003] 图1示出了根据至少一个实施例的用于执行多播归约操作的系统;

[0004] 图2示出了根据至少一个实施例的用于执行多播归约操作的各个方面的网络设备;

[0005] 图3示出了根据至少一个实施例的用于执行多播归约操作的各个方面的端点和图形处理单元;

[0006] 图4示出了根据至少一个实施例的用于执行多播归约操作的各个方面的过程;

[0007] 图5示出了根据至少一个实施例的符合拓扑的多播归约数据的下行传输;

[0008] 图6示出了根据至少一个实施例的符合拓扑的多播归约数据的上行传输;

[0009] 图7示出了根据至少一个实施例的用于执行多播归约操作的各个方面的过程;

[0010] 图8示出了根据至少一个实施例的用于网络设备执行多播归约操作的各个方面的过程;

[0011] 图9示出了根据至少一个实施例的分布式系统;

[0012] 图10示出了根据至少一个实施例的示例性数据中心;

[0013] 图11示出了根据至少一个实施例的客户端-服务器网络;

[0014] 图12示出了根据至少一个实施例的计算机网络的示例;

[0015] 图13A示出了根据至少一个实施例的联网计算机系统;

[0016] 图13B示出了根据至少一个实施例的联网计算机系统;

[0017] 图13C示出了根据至少一个实施例的联网计算机系统;

[0018] 图14示出了根据至少一个实施例的系统环境的一个或更多个组件,在该系统环境中,服务可被提供为第三方网络服务;

[0019] 图15示出了根据至少一个实施例的云计算环境;

[0020] 图16示出了根据至少一个实施例的由云计算环境提供的一组功能抽象层;

[0021] 图17示出了根据至少一个实施例的在芯片级的超级计算机;

[0022] 图18示出了根据至少一个实施例的在机架模块级处的超级计算机;

[0023] 图19示出了根据至少一个实施例的在机架级处的超级计算机;

[0024] 图20示出了根据至少一个实施例的在整个系统级的超级计算机;

[0025] 图21A示出了根据至少一个实施例的推理和/或训练逻辑;

[0026] 图21B示出了根据至少一个实施例的推理和/或训练逻辑;

[0027] 图22示出了根据至少一个实施例的神经网络的训练和部署；

[0028] 图23示出了根据至少一个实施例的网络系统的架构；

[0029] 图24示出了根据至少一个实施例的网络系统的架构；

[0030] 图25示出了根据至少一个实施例的控制平面协议栈；

[0031] 图26示出了根据至少一个实施例的用户平面协议栈；

[0032] 图27示出了根据至少一个实施例的核心网的组件；

[0033] 图28示出了根据至少一个实施例的支持网络功能虚拟化(NFV)的系统的组件；

[0034] 图29示出了根据至少一个实施例的处理系统；

[0035] 图30示出了根据至少一个实施例的计算机系统；

[0036] 图31示出了根据至少一个实施例的系统；

[0037] 图32示出了根据至少一个实施例的示例性集成电路；

[0038] 图33示出了根据至少一个实施例的计算系统；

[0039] 图34示出了根据至少一个实施例的APU；

[0040] 图35示出了根据至少一个实施例的CPU；

[0041] 图36示出了根据至少一个实施例的示例性加速器集成切片；

[0042] 图37A-37B示出了根据至少一个实施例的示例性图形处理器；

[0043] 图38A示出了根据至少一个实施例的图形核心；

[0044] 图38B示出了根据至少一个实施例的GPGPU；

[0045] 图39A示出了根据至少一个实施例的并行处理器；

[0046] 图39B示出了根据至少一个实施例的处理集群；

[0047] 图39C示出了根据至少一个实施例的图形多处理器；

[0048] 图40示出了根据至少一个实施例的编程平台的软件栈；

[0049] 图41示出了根据至少一个实施例的图40的软件栈的CUDA实现；

[0050] 图42示出了根据至少一个实施例的图40的软件栈的ROCm实现；

[0051] 图43示出了根据至少一个实施例的图40的软件栈的OpenCL实现方式；

[0052] 图44示出了根据至少一个实施例的由编程平台支持的软件；以及

[0053] 图45示出了根据至少一个实施例的用于在图40-43的编程平台上执行的编译代码。

具体实施方式

[0054] 在之前和以下描述中，阐述了许多具体细节以提供对至少一个实施例的更透彻的理解。然而，对于本领域的技术人员将显而易见的是，可以在没有一个或更多个这些具体细节的情况下实践本发明构思。

[0055] 图1示出了根据至少一个实施例的用于执行多播归约操作的系统。在至少一个实施例中，多播归约操作是由分布式计算系统共同执行的计算任务。例如，分布式计算系统可以包括启动多播归约操作的端点102，该多播归约操作将由端点104共同执行，其中端点104中的每一个执行计算任务的某些部分。网络设备106a-f通过将数据分配给端点104并联合(coalescing)从端点104返回的数据来参与多播归约操作的执行。

[0056] 在至少一个实施例中，多播归约操作是分阶段执行的。在第一阶段，可被称为多播

阶段，端点102通过将数据通过网络设备106a-f的结构发送至端点104而启动多播归约操作。在第二阶段，然后端点104执行计算任务的相应部分。在第三阶段，其可称为归约阶段，来自端点104的数据通过结构流回到启动端点102。在这个阶段，网络设备可以组合或归约他们收到的数据。例如，网络设备106b可接收从网络设备106d-f中的每一个传输的数据，并将该数据组合以传输到网络设备106a。

[0057] 在至少一个实施例中，通过网络设备106a-f的结构的网络数据的流动是根据多播归约树执行的，该多播归约树在启动多播归约操作之前建立。此外，多播归约操作(有时被称为事务)在流经结构时获取资源。在至少一个实施例中，使用逐跳路由和预留方案，并包括支持将事务响应引导回代表启动事务预留的资源的设备。在至少一个实施例中，该逐跳路由支持对孤立的(orphaned)资源的自主清理和恢复，并防止在有环路的结构拓扑中出现死锁。实施例还可以包括支持最小化可能由动态分配资源引起的分段存储，支持可重复的归约行为，并最小化可能对交叉路由信道施压的独有的路由数据。在至少一个实施例中，通过在网络报头中放置逐跳路由信息来支持这些特征，以便将响应索引回到与明确预留的资源相对应的网络设备。当数据流向端点104时，该路由信息可以在每一跳的事务请求报头中被重新映射，并且当数据通过结构返回时，先前的路由信息可以在每一跳被恢复。在至少一个实施例中，预留的资源被跟踪，如果事务在系统中丢失，则超时被用来启动自动清理。

[0058] 在至少一个实施例中，端点(例如所描述的端点102-104中的任何一个)是包括一个或更多个处理器和用于存储由处理器执行的指令的存储器的计算设备。端点可以进一步包括一个或更多个网络接口，以及用于启动多播归约操作的硬件和/或软件组件。在至少一个实施例中，端点包括图形处理单元(“GPU”)或并行处理单元“PPU”，其包括支持启动或执行本文所述的多播归约操作的各个方面。请注意，尽管端点的实施例在此可被描述为包括GPU，但类似的组件(例如PPU)可在不损害相关功能的情况下被替代。

[0059] 在至少一个实施例中，端点102是启动多播归约操作的端点。如上所述，与启动多播归约操作相关联的数据源自端点102，流经网络设备的结构，并由端点104接收。在至少一个实施例中，端点104是各自接收该数据的某些部分、使用该数据来执行多播归约操作的相应部分、以及通过网络结构将数据返回到启动端点102的端点。

[0060] 在至少一个实施例中，网络结构包括多个网络设备，诸如描绘的网络设备106a-f。在至少一个实施例中，网络设备包括交换机、集线器、路由器或其他用于接收和传输数据的设备中的任何一个。这些网络设备可以包括一个或更多个网络接口、一个或更多个处理器以及用于存储由处理器执行的指令的存储器。

[0061] 在至少一个实施例中，多播归约树包括定义将被用于处理多播归约操作的网络设备和端点的拓扑的信息。拓扑可以包括指示通过网络结构将数据路由到端点104的路径，以及将数据从端点104路由到多播归约操作的启动端点102的路径的信息。

[0062] 图2示出了根据至少一个实施例的用于执行多播归约操作的各个方面的网络设备。在示例200中，网络设备206对应于图1中描绘的网络设备106a-f中的任何一个。在至少一个实施例中，网络设备206包括处理器212和用以存储指令的存储器214，当该指令由处理器212执行时，使网络设备206执行与执行多播归约操作相关联的本文所述的操作。

[0063] 网络设备206进一步包括网络接口210、220，用于从网络接收和传输数据。传入的数据可以被存储在输入缓冲区216中，而传出的数据被存储在输出缓冲区218中。在至少一

个实施例中，输入缓冲区216和/或输出缓冲区218对应于存储器214的部分，但是在其他实施例中可以包括单独的存储器。

[0064] 在至少一个实施例中，网络设备206接收与多播数据传输相关联的数据。该数据可以包括具有路由信息的一个或更多个报头，以及将由一个或更多个端点(诸如图1中描绘的端点104)使用的数据。在多播阶段，当数据从源发(originating)端点流向目的地端点时，网络设备206可以针对一组给定的输入数据，将输出数据传输到N个目的地(端点或附加的网络设备)，其中N可取决于多播归约的特定拓扑。在一些实施例中，数据被复制到网络设备206的每个输出端口，或复制到按多播归约拓扑配置的网络设备206的每个输出端口。在归约阶段，网络设备206可以从对应于先前N个目的地的N个源(source)接收返回数据。然后，该数据可以被组合，或被归约，并返回到先前的网络设备或主机。

[0065] 在至少一个实施例中，网络设备206预留资源以在多播阶段期间随着数据向外流动时执行归约。例如，在至少一个实施例中，网络设备206接收用于多播归约操作的数据，预留用于存储和处理返回数据的存储器(诸如在输入缓冲区216或存储器214中的空间)，并且将接收的数据转发到N个目的地。之后，在归约阶段，网络设备206接收从多播归约端点(诸如从图1中描绘的端点104之一)返回的数据，并使用预留的存储器资源来执行归约。然后，预留的存储器资源被释放。实施例可以以类似的方式预留、利用和释放其他资源类型。

[0066] 在至少一个实施例中，网络设备206管理传入数据和传出数据中的报头(header)信息，以促进多播归约操作的处理。在至少一个实施例中，这包括1)为传入的下行网络分组预留报头信息，2)在传出的下行网络分组中存储报头信息，3)使用传入的上行网络分组中的报头信息，以使用预留的资源执行归约，并释放预留的资源，以及4)使用预留的报头信息，向上行发送归约的网络数据。在至少一个实施例中，由于结构中的网络设备遵循上述操作，网络设备206确保多播归约操作的上行方向的网络分组将包括用于管理为归约操作预留的资源的报头信息。

[0067] 在至少一个实施例中，网络设备206在传出的下行报头中存储包括以下一项或更多项的信息：描述归约拓扑的信息、指向预留资源的指针、索引信息等等。

[0068] 在至少一个实施例中，网络设备206维护归约缓冲区。该归约缓冲区可以包括随着数据返回上行为处理多播归约数据而预留的存储器。该缓冲区可以包括用于存储在归约期间返回上行的数据的空间，并且在一些实施例中，该空间可以进一步用于组合数据。

[0069] 在至少一个实施例中，网络设备206维护包括与在归约期间返回上行的数据相关的信息的表或其他数据结构。由网络设备206存储的并且包括在返回到网络设备206的上行的数据中的报头信息可以包括索引到该表或其他数据结构的信息。

[0070] 图3示出了根据至少一个实施例的用于执行多播归约操作的各个方面的端点和图形处理单元的示例300。可以对应于图1中描绘的端点102、104中的任何一个的端点304可以包括处理器314和用以存储指令的存储器312，当该指令由处理器314执行时，使得端点执行如本文所述的与执行多播归约操作相关联的操作。

[0071] 在至少一个实施例中，端点304包括网络接口310以传输或接收网络数据。该网络数据可以与多播归约操作的启动或处理相关联。例如，在至少一个实施例中，端点304可以对应于图1中描绘的端点102，该端点通过向一个或更多个其他端点传输执行该操作的请求以及任何相关数据来启动多播归约操作。在另一个示例中，端点304可通过接收和处理与执

行该操作相关联的上行数据和下行数据来参与处理所启动的多播归约操作，类似于与图1相关地描述的端点104。

[0072] 在至少一个实施例中，端点304包括图形处理单元308。在至少一个实施例中，该图形处理单元包括主机接口320、存储器322和处理器324。主机接口包括在图形处理单元308和端点304的组件(如处理器314或存储器312)之间传输数据的电路。图形处理单元308可以包括许多处理器324，并包括被设计为优化任务的并行执行的电路，这些任务可潜在地包括但不限于图形渲染、物理学计算和机器学习。图形处理单元308可以进一步被配置为优化数据并行性、数据传输和由处理器324实现的专门指令集的优化执行。

[0073] 在至少一个实施例中，端点304包括软件(诸如设备驱动器)以与图形处理单元308接口。设备驱动器、或端点304和/或图形处理单元308的其他软件和/或电路适合于促进多播归约操作的处理。

[0074] 在至少一个实施例中，设备驱动器或其他软件和/或电路适合于促进多播归约操作的启动。例如，启动端点102中的设备驱动器可以使端点102启动多播归约操作。在至少一个实施例中，当数据被写入存储器的映射区域，例如写入端点304的存储器312的映射区域，或写入图形处理单元308的存储器322的映射区域时，多播归约操作由驱动器启动。这里，存储器的映射区域可以指与潜在的或现有的多播归约操作有关地被指定的物理或虚拟存储器区域。例如，在至少一个实施例中，虚拟存储器地址的范围可以被指定为多播归约操作的输入，因此当数据被写入该区域时，经由类似于图1中所描绘的网络设备的结构，它被驱动器自动传输到其他端点。类似地，设备驱动器或其他软件和/或电路可以被适配为促进接收端点对多播归约操作的相应部分的处理。例如，端点304可以接收与多播归约操作相关联的请求数据，并将该数据存储在图形处理单元308的存储器322的映射区域。然后，图形处理单元308可以处理与多播归约操作有关的请求数据，就像它处理任何其他操作一样。然后，该操作的结果可被写入被映射为输出的存储器322的区域，并且所产生的数据经由结构返回到源发(originating)端点。

[0075] 图4示出了根据至少一个实施例的用于执行多播归约操作的各个方面的过程。尽管示例过程400被描绘为一系列步骤或操作，但可以理解的是，过程400的实施例可以包括被改变、重新排序或省略的步骤或操作，除非明确指出或逻辑上需要，例如当一个步骤或操作的输出被用作另一个步骤或操作的输入。

[0076] 示例过程400的实施例可以由各种设备中的任何一个执行，包括但不一定限于网络设备。例如，在至少一个实施例中，示例过程400由类似于图1中描绘的网络设备106a-f或图2中描绘的网络设备206中的任何一个网络设备实现。下面的描述是指由这样的网络设备执行的步骤或操作，但在一些实施例中，也可以在其他合适的设备中实现。

[0077] 在402处，网络设备接收与多播操作相关联的网络数据。该多播操作由与多播操作相关联的多个端点共同执行，例如图1中描绘的端点104。类似于图1中描绘的网络设备，执行示例过程400的网络设备可以是网络结构(fabric)的一部分，在该结构中的设备根据拓扑接收和发送数据。在一些实施例中，该拓扑可与特定的多播操作或操作的类别相关联。因此，由网络设备接收到的网络数据可包括在多播归约操作中被一个或更多个下行端点使用的数据。在多播归约的下行阶段，数据从源发端点通过结构流向若干端点。

[0078] 在404处，网络设备预留资源以处理将从与多播操作相关联的端点接收的数据。这

样做是为了期望在多播归约操作的后续上行阶段中接收返回数据。在至少一个实施例中，在数据流向下行时预留资源有利于在数据返回上行时进行处理。

[0079] 在406处，网络设备将所接收的数据发送到多个网络设备。在至少一个实施例中，网络设备将所接收的数据的部分或全部发送到多个下行设备中的每一个，该下行设备可以包括其他网络设备或端点。例如，网络设备可以根据适用的拓扑将其接收到的网络数据转发到N个其他网络设备或端点中的每一个。在至少一个实施例中，网络设备通过使用多播网络传输来将数据的副本发送到N个目的地中的每一个来完成该步骤。

[0080] 在至少一个实施例中，该网络数据以类似的方式继续流经结构，直到被端点接收。然后，这些端点可以执行多播归约操作的它们的相应部分，并通过结构向上行发送返回数据。

[0081] 在408处，网络设备接收返回数据。在至少一个实施例中，如果网络设备将请求数据发送到下行的N个目的地，它可以期望(除非超时或其他错误)以从N个目的地中的每一个目的地接收返回数据。在至少一个实施例中，这些响应被缓存在步骤或操作404处预留的存储器资源中。

[0082] 在410处，网络设备使用预留的资源处理从端点接收的数据。例如，网络设备可以使用预留的存储空间，或其他预留的资源，以暂时存储接收到的返回数据，并将N个响应归约到单一(unitary)的结果。在该示例的另一个方面，考虑其中N＝2的情况，并且这两个响应分别对应于包括{“ABC”}和{“DEF”}的单独数据传输。在至少一个实施例中，归约这些响应然后可能得到阵列{“ABC”+“DEF”}。

[0083] 在412处，网络设备发送经处理的数据。在至少一个实施例中，根据适用的拓扑，数据被上行发送到在多播归约操作的下行阶段期间的一个或更多个设备，从该一个或更多个设备接收数据。

[0084] 图5示出了根据至少一个实施例的符合拓扑的多播归约数据的下行传输。在至少一个实施例中，被下行发送的数据包括用于命令、控制和路由中的一个或更多个的信息。

[0085] 在至少一个实施例中，拓扑是指多播归约操作中的数据将在下行方向和上行方向上行进的路线或路径。这可包括指示在多播归约操作中使用哪些网络设备和端点的信息，以及数据如何通过这些网络设备和端点进行路由。

[0086] 例如，图5的示例500中的拓扑可包括指示要使用网络设备506b、506d-e和端点502、504a-c执行的多播归约操作的信息。

[0087] 图5的示例500进一步描绘了多播归约数据可能如何流经网络设备506a-f和端点502、504的结构。在示例500中，多播归约操作由端点502启动，该端点502将多播归约数据下行发送到网络设备506b。在至少一个实施例中，这是基于与多播归约操作相关联的拓扑完成的。该关联可以基于各种因素中的任何一个，潜在地包括但不限于全局网络配置、启动端点502配置，或独立指派给每个多播归约操作的拓扑。

[0088] 然后，网络设备506b可以接收该数据并根据图4的示例过程400和适用的拓扑将数据转发给网络设备506d-e。注意，为了说明的目的，假定某些网络设备506a、c、f被排除在适用的拓扑之外。

[0089] 然后网络设备506d可根据适用的拓扑从506b接收数据，并将数据发送到端点504a。类似地，网络设备506e可根据拓扑从网络设备506b接收数据并将数据转发到端点

504b、c。这些网络设备506d-e中的每一个都根据与图4相关的示例过程400的实施例来处理所接收的数据。

[0090] 在上行返回该处理的结果之前，具有每一个在接收到与多播归约操作相关的请求数据的端点504a-c，然后可以响应于所接收到的数据而继续执行任何适用的处理。结果数据的性质将根据多播归约操作的性质而变化。作为简单的示例，端点504a-c中的每个端点可接收对数字阵列求和的请求，并返回包括这些数字之和的数据。因此，端点504a可对[1,2]求和并上行返回[3]，端点504b可对[2,2]求和并上行返回[4]，而端点504c可对[2,3]求和并上行返回[5]。

[0091] 图6示出了根据至少一个实施例的符合拓扑的多播归约数据的上行传输。继续图5的示例500，多播归约操作可以与指示多播归约数据如何在下行方向和上行方向上流经结构的拓扑相关联。在图6的示例600中，假定数据已经从源发端点602下行流向端点604a-c，并且端点604a-c已经各自处理其多播归约操作的相应部分，并且正在上行返回数据。

[0092] 在至少一个实施例中，从端点604a-c返回的数据遵循用于将数据发送到相应端点的路径的反向。例如，如果数据经由网络设备606b和606d从端点602下行发送到端点604a，则数据可以经由网络设备606d和606b从端点604a上行行进到端点602。类似地，如图6所示，来自端点604b的数据可以在包括网络设备606e和606b的路径上上行返回，同样，来自端点604c的数据也是如此。

[0093] 图7示出了根据至少一个实施例的用于执行多播归约操作的各个方面的过程。尽管示例过程700被描述为一系列的步骤或操作，但可以理解的是，过程700的实施例可以包括被改变、重新排序或省略的步骤或操作，除非明确指出或逻辑上需要，例如当一个步骤或操作的输出被用作另一个的输入时。

[0094] 示例过程700的实施例可以由各种设备中的任何一个执行，包括但不一定限于网络设备。例如，在至少一个实施例中，示例过程700由类似于图1中描绘的网络设备106a-f或图2中描绘的网络设备206的网络设备中的任何一个实现。下面的描述是指由这样的网络设备执行的步骤或操作，但在一些实施例中，也可以在其他合适的设备中实现。

[0095] 在702处，网络设备接收用于多播归约操作的网络数据。该数据从上行设备发送，作为多播归约操作的下行阶段的一部分。

[0096] 在704处，网络设备保存从传入的网络数据的报头获得的路由信息。该路由信息包括指示传入的网络数据的来源的数据，并且允许向上行移动的数据被返回到上行设备。

[0097] 在706处，网络设备预留用于归约的资源。在至少一个实施例中，预留的资源包括一个或更多个归约缓冲区，以用于随着数据从下行设备上行返回时存储数据。在至少一个实施例中，在每个端口的基础上为每个多播归约操作指派缓冲区。例如，如果网络设备包括六个端口，并且六个端口中的每一个端口被用于向下行设备多播数据，则可能预留六个归约缓冲区，为六个端口中的每一个端口预留一个归约缓冲区。

[0098] 在708处，网络设备在传出的网络数据的报头中插入路由信息。在至少一个实施例中，路由信息包括网络设备的网络地址，该网络地址随后可被下行设备用于将返回的数据上行路由到网络设备。

[0099] 在710处，网络设备向一个或更多个下行设备发送网络数据。在至少一个实施例中，这包括在708处插入的路由信息。在至少一个实施例中，至少部分地基于从传入数据的

报头获得的信息来识别数据被发送到的下行网络设备。该信息潜在地包括但不限于:识别适用的拓扑的信息,识别结构中的下一个下行跳的信息,数据应该被发送到的一个或更多个端点的地址,等等。

[0100] 在712处,网络设备接收与多播归约操作相关联返回的网络数据。在至少一个实施例中,该数据使用先前插入的路由信息从下行设备返回到网络设备。

[0101] 在714处,网络设备使用预留的资源对返回的网络数据执行归约。

[0102] 在716处,网络设备使用保存的路由信息将经归约的网络数据发送到一个或更多个上行设备。

[0103] 图8示出了根据至少一个实施例的网络设备执行多播归约操作的各个方面的过程。尽管示例过程800被描绘为一系列步骤或操作,但可以理解的是,过程800的实施例可以包括被改变、重新排序或省略的步骤或操作,除非明确指出或逻辑上需要,例如当一个步骤或操作的输出被用作另一个步骤或操作的输入。

[0104] 示例过程800的实施例可以由各种设备中的任何一个执行,包括但不一定限于网络设备。例如,在至少一个实施例中,示例过程800由类似于图1中描绘的网络设备106a-f或图2中描绘的网络设备206的网络设备中的任何一个实现。下面的描述指的是由这样的网络设备执行的步骤或操作,但是在一些实施例中,也可以在其他合适的设备中实现。

[0105] 在802处,网络设备接收将由多个端点执行的用于多播归约操作的网络数据。该网络数据可以被称为第一网络数据,以区别于被上行返回到网络设备的数据。示例过程800的网络设备可以指这些网络设备中的任何一个。在至少一个实施例中,第一网络数据包括命令、控制或路由信息中的一个或更多个。

[0106] 端点可以类似于图1中描绘的任何端点104。多播归约操作可由端点共同执行,从而每个端点使用从网络设备发送至它们的网络数据来参与执行多播归约操作。如图1所描绘的,数据从启动端点通过各种网络设备下行移动到端点。

[0107] 在至少一个实施例中,端点包括并行处理单元或其他处理器,并且由端点接收的网络数据被并行处理单元或其他处理器用于执行多播操作的一部分。每个端点执行操作的一部分,并且端点共同执行操作(不包括那些由网络设备执行的操作,如归约)。例如,在至少一个实施例中,多播归约操作可包括评估神经网络的层的输出。每个端点可使用端点中所包括的并行处理单元来执行评估的一部分。然后,评估的结果可以被上行返回,并由上行的网络设备进行归约。

[0108] 在至少一个实施例中,与多播归约操作相关联的传输由端点起动。在至少一个实施例中,这可以包括类似于图1中描绘的端点102、104中的任何端点。例如,在至少一个实施例中,端点102通过向已被指定为与多播归约操作相关联的存储器位置写入数据来启动多播归约操作。在至少一个实施例中,该存储器位置位于并行处理单元的存储器内或以其他方式可进入该存储器。当数据被写入时,启动端点可以通过发送网络数据(包括写入存储器位置的部分或全部数据)到下行的网络设备来进行响应。请注意,在某些情况下,启动端点可以将数据发送到一些介于中间(intervening)的网络设备(如网络设备106a),后者又可以将该数据的部分或全部发送到其他网络设备(如网络设备106b)。这些网络设备中的每一个都可以以类似的方式处理接收到的网络数据。在至少一个实施例中,端点(如图1中的端点104中的任何一个)可以以类似的方式上行返回多播归约数据,例如,通过将数据写入并

行处理单元的存储器位置。该方法允许端点的并行处理单元高效地执行其多播归约操作的部分，而不需要管理通过结构接收和发送数据的细节。

[0109] 在804处，网络设备预留网络设备的资源以处理将从多个端点接收的第二网络数据。这可以包括基于从端点的预期返回的数据，与多播归约操作相关联地预留资源。在至少一个实施例中，第二网络分组包括从下行设备执行的并被上行返回的多播归约操作的部分或全部结果。在每个阶段，部分结果可以被组合，直到得到完整的结果。在至少一个实施例中，关于多播归约操作的信息可以通过由网络设备接收的网络数据中的报头来传达。例如，在至少一个实施例中，网络数据中的一个或更多个报头包括路由信息。这可以包括用于根据拓扑路由数据的逐跳路由信息。在至少一个实施例中，第一网络数据中的一个或更多个报头包括与多播归约操作相关联的归约信息。这些报头还可以包括在网络数据和端点设备的虚拟存储器空间之间映射的信息。

[0110] 在至少一个实施例中，网络设备可以获得可包括在下行移动的网络数据的报头中的路由信息。然后，网络设备可以存储该路由信息。

[0111] 在至少一个实施例中，识别结构的拓扑中的周期，并且资源的分配至少部分地基于对周期的识别。例如，在至少一个实施例中，拓扑中未检测到的循环(loop)可能导致非计划中的资源预留周期，这可以通过检测到的这种循环和相应地调整资源预留来避免。

[0112] 在806处，网络设备将第一网络数据发送至多个附加的网络设备。在至少一个实施例中，在发送第一网络数据之前，网络设备将路由信息插入到传出数据的报头中。在至少一个实施例中，这个新的路由信息允许下行设备根据适用的拓扑将数据路由回上行。在上行阶段期间，网络设备检索存储的信息，并使用存储的信息将网络数据路由到下一组上行设备。

[0113] 在808处，网络设备接收正在返回上行的网络数据。为了区别于被下行发送的数据，该数据可被称为第二网络数据。该第二网络数据是在第一网络数据已经被一个或更多个端点处理并且正在返回上行之后接收的。

[0114] 在810处，网络设备处理第二网络数据。该处理可以包括归约操作，例如组合数据。在至少一个实施例中，至少部分地基于从第一网络数据中的一个或更多个报头获得的归约信息来执行归约。该归约信息可以包括用于促进归约传入数据的任何信息，潜在地包括但不限于指示归约数据应该被上行发送到的地方的信息、指示预期来自下行设备多少响应的信息以及指示应该如何组合被上行返回的数据的信息。

[0115] 在某些情况下，错误或其他条件可能中断或干扰多播归约操作的处理。这可以包括未能向上行设备返回数据的下行设备(包括网络设备和端点)。在至少一个实施例中，这种情况可以通过在经过阈值时间量之后释放预留资源来处理。在至少一个实施例中，网络设备响应于确定自从向下行设备发送网络数据起已经过了阈值时间量并确定下行网络设备中的至少一个网络设备尚未对接收向下行发送的网络数据作出响应而释放预留资源。

[0116] 在至少一个实施例中，网络设备可以响应于超时而放弃与多播归约操作相关联的网络流量。例如，网络设备可以确定自从向下行设备发送网络数据起已经过了阈值时间量，并且确定下行网络设备中的至少一个网络设备尚未响应。然后，响应于该确定，由网络设备接收的任何后续数据，以及与同一多播归约操作相关的数据，都可能被网络设备丢弃。

[0117] 服务器和数据中心

[0118] 以下附图阐述了但不限于可以用于实现至少一个实施例的基于示例性网络服务器和数据中心的系统。

[0119] 图9示出了根据至少一个实施例的分布式系统900。在至少一个实施例中，分布式系统900包括一个或更多个客户端计算设备902、904、906和908，其被配置成在一个或更多个网络910上执行和操作客户端应用，诸如网络(web)浏览器、专有客户端和/或其变体。在至少一个实施例中，服务器912可以经由网络910与远程客户端计算设备902、904、906和908通信地耦合。

[0120] 在至少一个实施例中，服务器912可适于运行一个或更多个服务或软件应用，诸如可管理跨多个数据中心的单点登录(SSO)访问的会话活动的服务和应用。在至少一个实施例中，服务器912还可以提供其他服务，或者软件应用，其可以包括非虚拟和虚拟环境。在至少一个实施例中，这些服务可作为基于web的服务或云服务或在软件即服务(SaaS)模型下被提供给客户端计算设备902、904、906和/或908的用户。在至少一个实施例中，操作客户端计算设备902、904、906和/或908的用户又可以利用一个或更多个客户端应用来与服务器912交互以利用由这些组件提供的服务。

[0121] 在至少一个实施例中，系统900的软件组件918、920和922在服务器912上实现。在至少一个实施例中，系统900的一个或更多个组件和/或由这些组件提供的服务也可由客户端计算设备902、904、906和/或908中的一个或更多个来实现。在至少一个实施例中，操作客户端计算设备的用户然后可以利用一个或更多个客户端应用来使用由这些组件提供的服务。在至少一个实施例中，这些组件可以用硬件、固件、软件或其组合来实现。应当理解，各种不同的系统配置是可能的，其可以不同于分布式系统900。因此，图9所示的实施例是用于实现实施例系统的分布式系统的一个示例，并且不旨在是限制性的。

[0122] 在至少一个实施例中，客户端计算设备902、904、906和/或908可以包括不同类型的计算系统。在至少一个实施例中，客户端计算设备可以包括便携式手持设备(例如，iPhone®、蜂窝电话、iPad®、计算平板、个人数字助理(PDA))或可穿戴设备(例如，Google Glass®头戴式显示器)，运行软件(如Microsoft Windows Mobile®)和/或各种移动操作系统(诸如iOS、Windows Phone、Android、BlackBerry 10、Palm OS和/或其变体)。在至少一个实施例中，设备可以支持不同应用，诸如不同互联网相关的应用、电子邮件、短消息服务(SMS)应用，并且可以使用各种其他通信协议。在至少一个实施例中，客户端计算设备还可以包括通用个人计算机，在通过示例，所述通用个人计算机包括运行各种版本的Microsoft Windows®、Apple Macintosh®和/或Linux操作系统的个人计算机和/或膝上型计算机。在至少一个实施例中，客户端计算设备可以是运行各种可商购的UNIX®或类似UNIX的操作系统中的任一种的工作站计算机，包括但不限于各种GNU/Linux操作系统，诸如Google Chrome OS。在至少一个实施例中，客户端计算设备还可以包括能够通过一个或更多个网络910进行通信的电子设备，诸如瘦客户端计算机、启用互联网的游戏系统(例如，具有或不具有Kinect®手势输入设备的微软Xbox游戏控制台)、和/或个人消息传递设备。尽管图9中的分布式系统900被示为具有四个客户端计算设备，但可支持任何数量的客户端计算设备。其他设备(诸如具有传感器的设备等)可与服务器912交互。

[0123] 在至少一个实施例中，分布式系统900中的网络910可以是能够使用各种可用协议

中的任何协议来支持数据通信的任何类型的网络，包括但不限于TCP/IP(传输控制协议/互联网协议)、SNA(系统网络架构)、IPX(互联网分组交换)、AppleTalk和/或其变体。在至少一个实施例中，网络910可以是局域网(LAN)，基于以太网的网络、令牌环、广域网、互联网、虚拟网络、虚拟专用网(VPN)、内联网、外联网、公共交换电话网络(PSTN)、红外网络、无线网络(例如，在电气与电子协会(IEEE)802.11协议组、Bluetooth®和/或任何其他无线协议中的任一者下运行的网络)，和/或这些和/或其他网络的任何组合。

[0124] 在至少一个实施例中，服务器912可以由一个或更多个通用计算机、专用服务器计算机(通过示例，包括PC(个人计算机)服务器、UNIX®服务器、中程服务器、大型计算机、机架式服务器等)、服务器农场、服务器集群或任何其他适当的布置和/或组合组成。在至少一个实施例中，服务器912可包括运行虚拟操作系统的一个或更多个虚拟机或涉及虚拟化的其他计算架构。在至少一个实施例中，可以虚拟化一个或更多个灵活的逻辑存储设备池，以便为服务器维护虚拟存储设备。在至少一个实施例中，虚拟网络可由服务器912使用软件定义的网络来控制。在至少一个实施例中，服务器912可适于运行一个或更多个服务或软件应用。

[0125] 在至少一个实施例中，服务器912可以运行任何操作系统，以及任何可商购的服务器操作系统。在至少一个实施例中，服务器912还可以运行各种附加服务器应用和/或中层应用中的任一种，包括HTTP(超文本传输协议)服务器、FTP(文件传输协议)服务器、CGI(公共网关接口)服务器、JAVA®服务器、数据库服务器和/或其变体。在至少一个实施例中，示例性数据库服务器包括但不限于从Oracle、Microsoft、Sybase、IBM(国际商业机器)和/或其变体可商购的那些。

[0126] 在至少一个实施例中，服务器912可包括一个或更多个应用，用于分析和合并从客户端计算设备902、904、906和908的用户接收的数据馈送和/或事件更新。在至少一个实施例中，数据馈送和/或事件更新可以包括但不限于，从一个或更多个第三方信息源和连续数据流接收的Twitter®馈送、Facebook®更新或实时更新，其可以包括与传感器数据应用、金融报价器、网络性能测量工具(例如，网络监视和业务管理应用)相关的实时事件，点击流分析工具、汽车交通监测和/或其变化。在至少一个实施例中，服务器912还可以包括用于经由客户端计算设备902、904、906和908的一个或更多个显示设备来显示数据馈送和/或实时事件的一个或更多个应用。

[0127] 在至少一个实施例中，分布式系统900还可包括一个或更多个数据库914和916。在至少一个实施例中，数据库可提供用于存储信息(诸如用户交互信息、使用模式信息、适配规则信息和其他信息)的机制。在至少一个实施例中，数据库914和916可以驻留在各种位置中。在至少一个实施例中，数据库914和916中的一个或更多个可以驻留在服务器912本地(和/或驻留在服务器912中)的非暂态存储介质上。在至少一个实施例中，数据库914和916可以远离服务器912并且经由基于网络的连接或专用连接与服务器912通信。在至少一个实施例中，数据库914和916可以驻留在存储区域网络(SAN)中。在至少一个实施例中，用于执行归属于服务器912的功能的任何必要的文件可以适当地本地存储在服务器912上和/或远程存储。在至少一个实施例中，数据库914和916可以包括关系数据库，诸如适于响应于SQL格式化的命令而存储、更新和检索数据的数据库。

[0128] 图10示出了根据至少一个实施例的示例性数据中心1000。在至少一个实施例中，

数据中心1000包括但不限于数据中心基础设施层1010、框架层1020、软件层1030和应用层1040。

[0129] 在至少一个实施例中，如图10所示，数据中心基础设施层1010可以包括资源协调器1012、分组的计算资源1014和节点计算资源(“节点C.R.”)1016(1)-1016(N)，其中“N”表示任何完整的正整数。在至少一个实施例中，节点C.R.1016(1)-1016(N)可以包括但不限于任意数量的中央处理单元(“CPU”)或其他处理器(包括加速器、现场可编程门阵列(“FPGA”)、图形处理器等)、存储器设备(例如，动态只读存储器)、存储设备(例如，固态硬盘或磁盘驱动器)、网络输入/输出(“NW I/O”)设备、网络交换机、虚拟机(“VM”)、功率模块和冷却模块等。在至少一个实施例中，节点C.R.1016(1)-1016(N)中的一个或更多个节点C.R.可以是具有一个或更多个上述计算资源的服务器。

[0130] 在至少一个实施例中，分组的计算资源1014可以包括容纳在一个或更多个机架内的节点C.R.的单独分组(未示出)，或者容纳在各个地理位置的数据中心内的许多机架(也未示出)。分组的计算资源1014内的节点C.R.的单独分组可以包括可以被配置或分配为支持一个或更多个工作负载的分组的计算、网络、存储器或存储资源。在至少一个实施例中，可以将包括CPU或处理器的几个节点C.R.分组在一个或更多个机架内，以提供计算资源来支持一个或更多个工作负载。在至少一个实施例中，一个或更多个机架还可以包括任意数量的电源模块、冷却模块和网络交换机，以任意组合。

[0131] 在至少一个实施例中，资源协调器1012可以配置或以其他方式控制一个或更多个节点C.R.1016(1)-1016(N)和/或分组的计算资源1014。在至少一个实施例中，资源协调器1012可以包括用于数据中心1000的软件设计基础结构(“SDI”)管理实体。在至少一个实施例中，资源协调器1012可以包括硬件、软件或其某种组合。

[0132] 在至少一个实施例中，如图10所示，框架层1020包括但不限于作业调度器1032、配置管理器1034、资源管理器1036和分布式文件系统1038。在至少一个实施例中，框架层1020可以包括支持软件层1030的软件1052和/或应用程序层1040的一个或更多个应用程序1042的框架。在至少一个实施例中，软件1052或应用程序1042可以分别包括基于Web的服务软件或应用程序，例如由Amazon Web Services、Google Cloud和Microsoft Azure提供的服务或应用程序。在至少一个实施例中，框架层1020可以是但不限于一种免费和开放源软件网络应用框架，例如可以利用分布式文件系统1038来进行大范围数据处理(例如“大数据”)的Apache Spark™(以下称为“Spark”)。在至少一个实施例中，作业调度器1032可以包括Spark驱动器，以促进对数据中心1000的各个层所支持的工作负载进行调度。在至少一个实施例中，配置管理器1034可以能够配置不同的层，例如软件层1030和包括Spark和用于支持大规模数据处理的分布式文件系统1038的框架层1020。在至少一个实施例中，资源管理器1036能够管理映射到或分配用于支持分布式文件系统1038和作业调度器1032的集群或分组计算资源。在至少一个实施例中，集群或分组计算资源可以包括数据中心基础设施层1010上的分组的计算资源1014。在至少一个实施例中，资源管理器1036可以与资源协调器1012协调以管理这些映射的或分配的计算资源。

[0133] 在至少一个实施例中，包括在软件层1030中的软件1052可以包括由节点C.R.1016(1)-1016(N)的至少一部分，分组计算资源1014和/或框架层1020的分布式文件系统1038使用的软件。一种或更多种类型的软件可以包括但不限于Internet网页搜索软件、电子邮件

病毒扫描软件、数据库软件和流视频内容软件。

[0134] 在至少一个实施例中，应用层1040中包括的一个或更多个应用程序1042可以包括由节点C.R.1016(1)-1016(N)的至少一部分、分组的计算资源1014和/或框架层1020的分布式文件系统1038使用的一种或更多种类型的应用程序。一种或更多种类型的应用程序可以包括但不限于CUDA应用程序、5G网络应用程序、人工智能应用程序、数据中心应用程序、和/或其变体。

[0135] 在至少一个实施例中，配置管理器1034、资源管理器1036和资源协调器712中的任何一个可以基于以任何技术上可行的方式获取的任意数量和类型的数据来实现任意数量和类型的自我修改动作。在至少一个实施例中，自我修改动作可以减轻数据中心1000的数据中心操作员做出可能不好的配置决定并且可以避免数据中心的未充分利用和/或执行差的部分。

[0136] 图11示出了根据至少一个实施例的由互连的多个网络服务器计算机1102形成的客户端-服务器网络1104。在至少一个实施例中，在系统1100中，每个网络服务器计算机1102存储其他网络服务器计算机1102和链接到广域网1104中的客户端计算机1106和网络1108可访问的数据。在至少一个实施例中，当客户端计算机1106和一个或更多个网络1108与网络1104连接和断开连接时，以及当一个或更多个干线服务器计算机1102被添加到网络1104或从网络1104移除时，客户端-服务器网络1104的配置可随时间改变。在至少一个实施例中，当客户端计算机1106和网络1108与网络服务器计算机1102连接时，客户端-服务器网络包括这样的客户端计算机1106和网络1108。在至少一个实施例中，术语计算机包括能够接受数据、将规定的过程应用于数据以及提供过程的结果的任何设备或机器。

[0137] 在至少一个实施例中，客户端-服务器网络1104存储网络服务器计算机1102、远程网络1108和客户端计算机1106可访问的信息。在至少一个实施例中，网络服务器计算机1102由大型计算机、小型计算机和/或各自具有一个或更多个处理器的微型计算机形成。在至少一个实施例中，服务器计算机1102通过有线和/或无线传输介质(诸如导线、光纤电缆)和/或微波传输介质、卫星传输介质或其他导电、光学或电磁波传输介质链接在一起。在至少一个实施例中，客户端计算机1106通过类似的有线或无线传输介质访问网络服务器计算机1102。在至少一个实施例中，客户端计算机1106可以使用调制解调器和标准电话通信网络链接到客户端-服务器网络1104中。在至少一个实施例中，替代性的运营商系统(如电缆和卫星通信系统)还可以用于链接到客户端-服务器网络1104中。在至少一个实施例中，可以使用其他私有或时间共享的运营商系统。在至少一个实施例中，网络1104是全球信息网络，诸如互联网。在至少一个实施例中，网络是使用与互联网类似的协议但具有添加的安全措施和受限的访问控制的私有内联网。在至少一个实施例中，网络1104是使用专有通信协议的私有或半私有网络。

[0138] 在至少一个实施例中，客户端计算机1106是任何终端用户计算机，并且还可以是具有一个或更多个微处理器的大型计算机、小型计算机或微型计算机。在至少一个实施例中，服务器计算机1102有时可用作访问另一服务器计算机1102的客户端计算机。在至少一个实施例中，远程网络1108可以是局域网、通过用于互联网的独立服务提供商(ISP)被添加到广域网中的网络、或通过具有固定的或随时间改变的配置的有线或无线传输介质互连的另一组计算机。在至少一个实施例中，客户端计算机1106可以独立地或通过远程网络1108

链接到网络1104中并且访问网络1104。

[0139] 图12示出了根据至少一个实施例的连接一个或更多个计算机器的计算机网络1208的示例1200。在至少一个实施例中，网络1208可以是任何类型的电连接的计算机组，包括例如以下网络：互联网、内联网、局域网(LAN)、广域网(WAN)或这些网络类型的互连组合。在至少一个实施例中，网络1208内的连接可以是远程调制解调器、以太网(IEEE 802.3)、令牌环(IEEE 802.5)、光纤分布式数据链路接口(FDDI)、异步传输模式(ATM)或任何其他通信协议。在至少一个实施例中，链接到网络的计算设备可以是台式机、服务器、便携式、手持式、机顶盒、个人数字助理(PDA)、终端、或任何其他期望的类型或配置。在至少一个实施例中，取决于它们的功能性，网络连接的设备可以在处理能力、内部存储器和其他性能方面广泛地变化。在至少一个实施例中，网络内的通信以及去往或来自连接到网络的计算设备的通信可以是有线或无线的。在至少一个实施例中，网络1208可以至少部分地包括世界范围的公共互联网，其通常根据客户端-服务器模型根据传输控制协议/互联网协议(TCP/IP)规范连接多个用户。在至少一个实施例中，客户端-服务器网络是用于在两个计算机之间通信的主导模型。在至少一个实施例中，客户端计算机(“客户端”)向服务器计算机(“服务器”)发出一个或更多个命令。在至少一个实施例中，服务器通过访问可用网络资源并根据客户端命令向客户端返回信息来履行客户端命令。在至少一个实施例中，客户端计算机系统和驻留在网络服务器上的网络资源被分配网络地址，用于网络的元件之间的通信期间的识别。在至少一个实施例中，从其他网络连接的系统到服务器的通信将包括作为通信的一部分的相关服务器/网络资源的网络地址，使得数据/请求的适当目的地被识别为接收者。在至少一个实施例中，当网络1208包括全球互联网时，网络地址是TCP/IP格式的IP地址，其可至少部分地将数据路由到电子邮件账户、网站或驻留在服务器上的其他互联网工具。在至少一个实施例中，驻留在网络服务器上的信息和服务可以通过域名(例如www.site.com)(其映射到网络服务器的IP地址)对客户端计算机的web浏览器可用。

[0140] 在至少一个实施例中，多个客户端1202、1204和1206经由相应的通信链路连接至网络1208。在至少一个实施例中，这些客户端中的每一个可以经由任何期望形式的通信(诸如经由拨号调制解调器连接、电缆链路、数字用户线(DSL)、无线或卫星链路、或任何其他形式的通信)来访问网络1208。在至少一个实施例中，每个客户端可以使用与网络1208兼容的任何机器(例如，个人计算机(PC)、工作站、专用终端、个人数据助理(PDA)或其他类似的设备)进行通信。在至少一个实施例中，客户端1202、1204和1206可以位于或可以不位于相同的地理区域中。

[0141] 在至少一个实施例中，多个服务器1210、1212和1214连接到网络1218以服务于与网络1218通信的客户端。在至少一个实施例中，每个服务器通常是管理网络资源并对客户端命令作出响应的强大的计算机或设备。在至少一个实施例中，服务器包括存储程序指令和数据的计算机可读数据存储介质，诸如硬盘驱动器和RAM存储器。在至少一个实施例中，服务器1210、1212、1214运行响应于客户端命令的应用程序。在至少一个实施例中，服务器1210可以运行用于响应对HTML页面的客户端请求的web服务器应用，并且还可以运行用于接收和路由电子邮件的邮件服务器应用。在至少一个实施例中，在服务器1210上还可以运行其他应用程序，诸如用于将音频/视频数据流式传输至客户端的FTP服务器或媒体服务器。在至少一个实施例中，不同的服务器可以专用于执行不同的任务。在至少一个实施例

中,服务器1210可以是为不同用户管理与网站相关的资源的专用web服务器,而服务器1212可以专用于提供电子邮件(email)管理。在至少一个实施例中,其他服务器可以专用于媒体(音频、视频等)、文件传输协议(FTP)或通常通过网络可用或提供的任何两个或更多个服务的组合。在至少一个实施例中,每个服务器可以在与其他服务器的位置相同或不同的位置中。在至少一个实施例中,可存在为用户执行镜像任务的多个服务器,从而减轻拥塞或最小化定向到和来自单个服务器的流量。在至少一个实施例中,服务器1210、1212、1214在维护和通过网络1218递送第三方内容的业务中的web托管提供者的控制下。

[0142] 在至少一个实施例中,web托管提供商向两个不同类型的客户端递送服务。在至少一个实施例中,可被称为浏览器的一种类型从服务器1210、1212、1214请求内容,诸如网页、电子邮件消息、视频剪辑等。在至少一个实施例中,第二类型(其可以被称为用户)雇佣web托管提供商来维护网络资源(诸如网站)并使其可用于浏览器。在至少一个实施例中,用户与web托管提供商签订合同,以根据用户期望利用的服务器资源的量使存储器空间、处理器容量和通信带宽可用于他们期望的网络资源。

[0143] 在至少一个实施例中,为了使web托管提供商为这两个客户端提供服务,必须适当地配置管理由服务器托管的网络资源的应用程序。在至少一个实施例中,程序配置过程涉及定义参数集,所述参数集至少部分地控制应用程序对浏览器请求的响应,并且还至少部分地定义特定用户可用的服务器资源。

[0144] 在一个实施例中,内联网服务器1216经由通信链路与网络1208通信。在至少一个实施例中,内联网服务器1216与服务器管理器1218通信。在至少一个实施例中,服务器管理器1218包括在服务器1210、1212、1214中使用的应用程序配置参数的数据库。在至少一个实施例中,用户经由内联网1216修改数据库1220,并且服务器管理器1218与服务器1210、1212、1214交互以修改应用程序参数,使得它们匹配数据库的内容。在至少一个实施例中,用户通过经由计算机1202连接到内联网1216并且输入诸如用户名和密码之类的认证信息来登录到内联网1216。

[0145] 在至少一个实施例中,当用户希望登录新服务或修改现有服务时,内联网服务器1216对用户进行认证并向用户提供允许用户访问特定应用程序的配置参数的交互式屏幕显示/控制面板。在至少一个实施例中,向用户呈现描述用户的网站或其他网络资源的配置的方面的多个可修改文本框。在至少一个实施例中,如果用户期望增加在服务器上为其网站预留的存储器空间,则向用户提供其中用户指定期望的存储器空间的字段。在至少一个实施例中,响应于接收到该信息,内联网服务器1216更新数据库1220。在至少一个实施例中,服务器管理器1218将该信息转发到适当的服务器,并且在应用程序操作期间使用新的参数。在至少一个实施例中,内联网服务器1216被配置为向用户提供对用户已与web托管服务提供商签订的托管网络资源(例如,网页、电子邮件、FTP站点、媒体站点等)的配置参数的访问。

[0146] 图13A示出了根据至少一个实施例的联网计算机系统1300A。在至少一个实施例中,联网计算机系统1300A包括多个节点或个人计算机(“PC”)1302、1318、1320。在至少一个实施例中,个人计算机或节点1302包括处理器1314、存储器1316、摄像机1304、麦克风1306、鼠标1308、扬声器1310和监视器1312。在至少一个实施例中,PC 1302、1318、1320可以各自运行例如给定公司内的内部网络的一个或更多个桌面服务器,或者可以是不限于特定环境

的通用网络的服务器。在至少一个实施例中，网络的每PC节点有一个服务器，使得网络的每个PC节点表示具有特定网络URL地址的特定网络服务器。在至少一个实施例中，每个服务器默认为该服务器的用户的默认网页，该默认网页本身可包含指向该服务器上该用户的进一步子页面、或者指向网络上的其他服务器或其他服务器上的页面的嵌入式URL。

[0147] 在至少一个实施例中，节点1302、1318、1320和网络的其他节点经由介质1322互连。在至少一个实施例中，介质1322可以是诸如综合服务数字网(“ISDN”)的通信信道。在至少一个实施例中，联网计算机系统的各个节点可以通过各种通信介质连接，包括局域网(“LAN”)、简易老式电话线(“POTS”)(有时被称为公共交换电话网络(“PSTN”))、和/或其变体。在至少一个实施例中，网络的各个节点还可以构成经由诸如互联网之类的网络互连的计算机系统用户。在至少一个实施例中，(在给定实例处从网络的特定节点运行的)网络上的每个服务器在网络内具有唯一地址或标识，该唯一地址或标识可以根据URL指定。

[0148] 在至少一个实施例中，多个多点会议单元(“MCU”)因此可以用于向会议系统的各个节点或“端点”和从会议系统的各个节点或“端点”传输数据。在至少一个实施例中，除了各种其他通信介质(诸如，通过互联网连接的节点)之外，节点和/或MCU可经由ISDN链路或通过局域网(“LAN”)互连。在至少一个实施例中，会议系统的节点通常可以直接连接到通信介质(诸如LAN)或通过MCU连接，并且会议系统可以包括其他节点或元件，诸如路由器、服务器和/或其变体。

[0149] 在至少一个实施例中，处理器1314是通用可编程处理器。在至少一个实施例中，联网计算机系统1300A的节点的处理器还可以是专用视频处理器。在至少一个实施例中，节点的不同外围设备和组件(诸如节点1302的那些)可以与其他节点的那些不同。在至少一个实施例中，节点1318和节点1320可以被配置为与节点1302相同或不同。在至少一个实施例中，除了PC系统之外，节点还可以在任何合适的计算机系统上实现。

[0150] 图13B示出了根据至少一个实施例的联网计算机系统1300B。在至少一个实施例中，系统1300B示出了网络(诸如LAN 1324)，该网络可以用于互连可以彼此通信的各种节点。在至少一个实施例中，附接到LAN 1324的是多个节点，诸如PC节点1326、1328、1330。在至少一个实施例中，节点还可经由网络服务器或其他装置连接到LAN。在至少一个实施例中，系统1300B包括其他类型的节点或元件，针对示例，其包括路由器、服务器和节点。

[0151] 图13C示出根据至少一个实施例的联网计算机系统1300C。在至少一个实施例中，系统1300C示出了具有跨主干通信网络(诸如互联网1332)的通信的WWW系统，主干通信网络可用于互连网络的各种节点。在至少一个实施例中，WWW是在互联网的顶部上操作的一组协议，并且允许图形界面系统在其上操作以便通过互联网访问信息。在至少一个实施例中，附接到WWW中的互联网1332的是多个节点，例如PC 1340、1342、1344。在至少一个实施例中，节点通过WWW HTTP服务器(诸如服务器1334、1336)与WWW的其他节点对接。在至少一个实施例中，PC 1344可以是形成网络1332的节点的PC，并且PC 1344本身运行它的服务器1336，尽管为了说明的目的在图13C中单独地示出PC 1344和服务器1336。

[0152] 在至少一个实施例中，WWW是一种分布式类型的应用程序，其特征为WWW HTTP、WWW的协议，它在互联网的传输控制协议/互联网协议(“TCP/IP”)的顶部上运行。在至少一个实施例中，WWW因此可以由在互联网上运行的一组协议(即，HTTP)作为其“主干”来表征。

[0153] 在至少一个实施例中，web浏览器是在兼容WWW类型的网络系统中在网络的节点上

运行的应用程序,其允许特定服务器或节点的用户查看这样的信息,并因此允许用户搜索使用嵌入在从理解HTTP的网络上的服务器可获得的文档或文件中的超文本链接链接在一起的图形和基于文本的文件。在至少一个实施例中,当用户使用诸如互联网之类的网络上的另一服务器来检索与第一节点相关联的第一服务器的给定网页时,所检索到的文档可具有嵌入在其中的不同超文本链接,并且在检索用户的本地创建页面的本地副本。在至少一个实施例中,当用户点击超文本链接时,与所选择的超文本链接相关的本地存储的信息通常足以允许用户的机器打开通过互联网到由超文本链接指示的服务器的连接。

[0154] 在至少一个实施例中,多于一个用户可例如通过LAN(诸如LAN 1338,诸如关于WWW HTTP服务器1334所示)耦合到每个HTTP服务器。在至少一个实施例中,系统1300C还可以包括其他类型的节点或元件。在至少一个实施例中,WWW HTTP服务器是在诸如PC的机器上运行的应用。在至少一个实施例中,每个用户可以被认为具有唯一的“服务器”,如关于PC 1344所示。在至少一个实施例中,服务器可以被认为是诸如WWW HTTP服务器1334之类的服务器,该服务器为LAN或多个节点或多个LAN提供对网络的访问。在至少一个实施例中,存在多个用户,每个用户具有台式PC或网络的节点,每个台式PC潜在地为其用户建立服务器。在至少一个实施例中,每个服务器与特定网络地址或URL相关联,当被访问时,该特定网络地址或URL为该用户提供默认网页。在至少一个实施例中,网页可以包含进一步的链接(嵌入式URL),其指向该服务器上的该用户的进一步的子页面,或者指向网络上的其他服务器或者指向网络上的其他服务器上的页面。

[0155] 云计算和服务

[0156] 以下附图阐述但不限于可以用于实现至少一个实施例的示例性的基于云的系统。

[0157] 在至少一个实施例中,云计算是一种计算风格,其中动态可扩展和通常虚拟化的资源作为服务通过互联网来提供。在至少一个实施例中,用户不需要具有支持他们的技术基础设施的知识、技术基础设施的专业知识或对技术基础设施的控制,该技术基础设施可以被称为“在云中”。在至少一个实施例中,云计算将基础设施合并为服务、平台即服务、软件即服务、以及具有依赖于互联网的常见主题以满足用户的计算需求的其他变型。在至少一个实施例中,典型的云部署(诸如在私有云(例如,企业网络)中)或公共云(例如,互联网)中的数据中心(DC)可由数千个服务器(或可替代地,VM)、数百个以太网、光纤信道或以太网光纤信道(FCoE)端口、交换和存储基础设施等组成。在至少一个实施例中,云还可以由网络服务基础设施组成,如IPsec VPN集线器、防火墙、负载平衡器、广域网(WAN)优化器等。在至少一个实施例中,远程订户可以通过经由VPN隧道(如IPsec VPN隧道)连接来安全地访问云应用和服务。

[0158] 在至少一个实施例中,云计算是一种用于使能对可配置计算资源(例如,网络、服务器、存储装置、应用程序和服务)的共享池的方便、按需的网络访问的模型,所述可配置计算资源可以用最小的管理努力或服务提供商交互来快速配置和释放。

[0159] 在至少一个实施例中,云计算的特征在于按需自助服务,其中消费者可根据需要自动地单方面供应计算能力,诸如服务器时间和网络存储,而无需与每个服务提供商的人类互动。在至少一个实施例中,云计算的特征在于广泛的网络访问,其中能力在网络上可用并且通过标准机制来访问,所述标准机制促进由异构的瘦或厚客户端平台(例如,移动电话、膝上型计算机和PDA)的使用。在至少一个实施例中,云计算的特征在于资源池,其中提

供商的计算资源被池化以使用多租户模型服务于多个消费者，其中不同的物理和虚拟资源根据消费者需求被动态地签名和重新分配。在至少一个实施例中，存在位置独立性的感觉，因为消费者通常对所提供的资源的确切位置没有控制或知识，但可能能够在较高抽象级别(例如，国家、州或数据中心)指定位置。

[0160] 在至少一个实施例中，资源的示例包括存储、处理、存储器、网络带宽和虚拟机。在至少一个实施例中，云计算的特征在于快速弹性，其中能力可被快速且弹性地供应(在一些情况下是自动地)，以快速缩小和快速释放以快速放大。在至少一个实施例中，对于消费者，可用于供应的能力通常显得不受限制并且可以在任何时间以任何数量购买。在至少一个实施例中，云计算由测量的服务来表征，其中云系统通过在适合于服务类型(例如，存储、处理、带宽和活动用户账户)的某种抽象级别处利用计量能力来自动地控制和优化资源使用。在至少一个实施例中，资源使用可以被监控、控制和报告，从而为所利用的服务的提供商和消费者两者提供透明度。

[0161] 在至少一个实施例中，云计算可与各种服务相关联。在至少一个实施例中，云软件即服务(SaaS)可以指代提供给消费者的能力是使用在云基础设施上运行的提供商的应用的服务。在至少一个实施例中，应用可通过诸如web浏览器(例如，基于web的电子邮件)的瘦客户端接口从不同客户端设备访问。在至少一个实施例中，消费者不管理或控制包括网络、服务器、操作系统、存储或甚至各个应用能力的底层云基础结构，可能的例外是有限的用户特定的应用配置设置。

[0162] 在至少一个实施例中，云平台即服务(PaaS)可以指这样的服务：其中提供给消费者的能力是将消费者创建或获取的应用程序部署到云基础设施上，这些应用程序是使用由提供商支持的编程语言和工具创建的。在至少一个实施例中，消费者不管理或控制包括网络、服务器、操作系统或存储的底层云基础结构，但是具有对所部署的应用程序以及可能的应用托管环境配置的控制。

[0163] 在至少一个实施例中，云基础设施即服务(IaaS)可以指这样的服务：其中向消费者提供的能力是提供处理、存储、网络和消费者能够部署和运行可包括操作系统和应用的任意软件的其他基本计算资源。在至少一个实施例中，消费者不管理或控制底层云基础设施，而是具有对操作系统、存储、所部署的应用程序的控制，以及对选择的联网组件(例如，主机防火墙)的可能有限的控制。

[0164] 在至少一个实施例中，可以不同方式部署云计算。在至少一个实施例中，私有云可指仅针对组织操作的云基础设施。在至少一个实施例中，私有云可由组织或第三方管理，并且可存在于场所内或场所外。在至少一个实施例中，社区云可以指由若干组织共享并且支持具有共享关注(例如，任务、安全要求、策略和合规性考虑)的特定社区的云基础设施。在至少一个实施例中，社区云可由组织或第三方管理，并且可存在于场所内或场所外。在至少一个实施例中，公共云可以指代对一般公众或大型产业组可用并且由提供云服务的组织拥有的云基础设施。在至少一个实施例中，混合云可以指云基础设施是两个或更多个云(私有、社区或公共的)的组成部分，这些云仍然是唯一的实体，但是通过实现数据和应用便携性的标准化或专有技术(例如，用于云之间的负载平衡的云突发)绑定在一起。在至少一个实施例中，云计算环境是面向服务的，其关注于无状态性、低耦合、模块性和语义互操作性。

[0165] 图14示出了根据至少一个实施例的系统环境1400的一个或更多个组件，其中服务

可被提供为第三方网络服务。在至少一个实施例中，第三方网络可被称为云、云网络、云计算网络和/或其变体。在至少一个实施例中，系统环境1400包括一个或更多个客户端计算设备1404、1406和1408，客户端计算设备1404、1406和1408可被用户用来与提供第三方网络服务(其可被称为云计算服务)的第三方网络基础设施系统1402交互。在至少一个实施例中，第三方网络基础设施系统1402可包括一个或更多个计算机和/或服务器。

[0166] 应了解，图14中所描绘的第三方网络基础设施系统1402可具有除了所描绘的那些组件之外的其他组件。进一步地，图14描绘了第三方网络基础设施系统的实施例。在至少一个实施例中，第三方网络基础设施系统1402可具有比图14中描绘的更多或更少的组件，可组合两个或更多个组件，或可具有不同的组件配置或布置。

[0167] 在至少一个实施例中，客户端计算设备1404、1406和1408可被配置成操作客户端应用，诸如web浏览器，可由客户端计算设备的用户用来与第三方网络基础设施系统1402交互以使用由第三方网络基础设施系统1402提供的服务的专有客户端应用或一些其他应用。尽管示例性系统环境1400被示为具有三个客户端计算设备，但是可以支持任何数量的客户端计算设备。在至少一个实施例中，其他设备，诸如具有传感器的设备等，可与第三方网络基础设施系统1402交互。在至少一个实施例中，一个或更多个网络1410可以促进客户端计算设备1404、1406和1408与第三方网络基础设施系统1402之间的通信和数据交换。

[0168] 在至少一个实施例中，由第三方网络基础设施系统1402提供的服务可包括按需可用于第三方网络基础设施系统的用户的服务的主机。在至少一个实施例中，还可以提供各种服务，包括但不限于在线数据存储和备份解决方案、基于Web的电子邮件服务、托管的办公套件和文档协作服务、数据库管理和处理、管理的技术支持服务、和/或其变体。在至少一个实施例中，由第三方网络基础设施系统提供的服务可以动态地扩展以满足其用户的需要。

[0169] 在至少一个实施例中，由第三方网络基础设施系统1402提供的服务的特定实例化可被称为“服务实例”。在至少一个实施例中，通常，经由通信网络(诸如互联网)从第三方网络服务提供商系统对用户可用的任何服务被称为“第三方网络服务”。在至少一个实施例中，在公共第三方网络环境中，组成第三方网络服务提供商系统的服务器和系统不同于客户自己的场所内服务器和系统。在至少一个实施例中，第三方网络服务提供商系统可以托管应用，并且用户可以经由通信网络(诸如互联网)按需订购和使用应用。

[0170] 在至少一个实施例中，计算机网络第三方网络基础设施中的服务可包括对存储、托管数据库、托管网络服务器、软件应用或由第三方网络供应商提供给用户的其他服务的受保护的计算机网络访问。在至少一个实施例中，服务可包括通过互联网对第三方网络上的远程存储装置的密码保护的访问。在至少一个实施例中，服务可以包括基于网络服务的托管关系数据库和脚本语言中间件引擎，以便由联网开发者私人使用。在至少一个实施例中，服务可包括对托管在第三方网络供应商的网站上的电子邮件软件应用的访问。

[0171] 在至少一个实施例中，第三方网络基础设施系统1402可包括以自助、基于订阅、可弹性扩展、可靠、高度可用和安全方式递送给客户的一套应用、中间件和数据库服务提供物。在至少一个实施例中，第三方网络基础设施系统1402还可以提供“大数据”相关的计算和分析服务。在至少一个实施例中，术语“大数据”通常用于指可以由分析师和研究人员存储和操纵的极大数据集，以便使大量数据可视化、检测趋势、和/或以其他方式与数据交互。

在至少一个实施例中，大数据和相关应用可以由基础设施系统在许多级别上和以不同规模托管和/或操纵。在至少一个实施例中，并行链接的数十个、数百个或数千个处理器可对此类数据起作用以便呈现该数据或模拟对数据或其所表示的内容的外力。在至少一个实施例中，这些数据集可涉及结构化数据(诸如在数据库中或以其他方式根据结构化模型组织的结构化数据)和/或非结构化数据(例如，电子邮件、图像、数据blob(二进制大对象)、网页、复杂事件处理)。在至少一个实施例中，通过利用实施例的能力来将更多(或更少)计算资源相对快速地聚焦到目标上，第三方网络基础设施系统可以更好地可用于基于来自企业、政府机构、研究组织、私人个人、想法相同的个人或组织的组、或其他实体的需求在大数据集上执行任务。

[0172] 在至少一个实施例中，第三方网络基础设施系统1402可以被适配成自动地提供、管理和跟踪顾客对由第三方网络基础设施系统1402提供的服务的订阅。在至少一个实施例中，第三方网络基础设施系统1402可以经由不同的部署模型提供第三方网络服务。在至少一个实施例中，可在公共第三方网络模型下提供服务，其中第三方网络基础设施系统1402由销售第三方网络服务的组织拥有，并且使得服务可用于一般公众或不同的行业企业。在至少一个实施例中，可在私有第三方网络模型下提供服务，在该私有第三方网络模型中，第三方网络基础设施系统1402仅针对单个组织操作，并且可为组织内的一个或更多个实体提供服务。在至少一个实施例中，第三方网络服务也可在社区第三方网络模型下提供，其中第三方网络基础设施系统1402和第三方网络基础设施系统1402提供的服务由相关社区中的若干组织共享。在至少一个实施例中，也可在混合第三方网络模型下提供第三方网络服务，该混合第三方网络模型是两个或更多个不同模型的组合。

[0173] 在至少一个实施例中，由第三方网络基础设施系统1402提供的服务可包括在软件即服务(SaaS)类别、平台即服务(PaaS)类别、基础设施即服务(IaaS)类别或包括混合服务的其他服务类别下提供的一个或更多个服务。在至少一个实施例中，客户经由订阅订单可订购由第三方网络基础设施系统1402提供的一个或更多个服务。在至少一个实施例中，第三方网络基础设施系统1402然后执行处理以在客户的订阅订单中提供服务。

[0174] 在至少一个实施例中，由第三方网络基础设施系统1402提供的服务可以包括但不限于应用服务、平台服务和基础设施服务。在至少一个实施例中，应用服务可由第三方网络基础设施系统经由SaaS平台提供。在至少一个实施例中，SaaS平台可被配置为提供属于SaaS类别的第三方网络服务。在至少一个实施例中，SaaS平台可以提供在集成开发和部署平台上构建并递送一套按需应用的能力。在至少一个实施例中，SaaS平台可以管理和控制用于提供SaaS服务的底层软件和基础设施。在至少一个实施例中，通过利用由SaaS平台提供的服务，客户可利用在第三方网络基础设施系统上执行的应用。在至少一个实施例中，客户可以获得应用服务，而不需要客户购买单独的许可证和支持。在至少一个实施例中，可提供各种不同的SaaS服务。在至少一个实施例中，示例包括但不限于为大组织的销售性能管理、企业集成和商业灵活性提供解决方案的服务。

[0175] 在至少一个实施例中，平台服务可由第三方网络基础设施系统1402经由PaaS平台提供。在至少一个实施例中，PaaS平台可被配置为提供属于PaaS类别的第三方网络服务。在至少一个实施例中，平台服务的示例可以包括但不限于使组织能够将现有应用合并在共享的公共架构上的服务，以及建立利用由平台提供的共享服务的新应用的能力。在至少一个

实施例中，PaaS平台可以管理和控制用于提供PaaS服务的底层软件和基础设施。在至少一个实施例中，客户可获取由第三方网络基础设施系统1402提供的PaaS服务，而无需客户购买单独的许可证和支持。

[0176] 在至少一个实施例中，通过利用由PaaS平台提供的服务，客户可采用由第三方网络基础设施系统支持的编程语言和工具，并且还控制所部署的服务。在至少一个实施例中，由第三方网络基础设施系统提供的平台服务可包括数据库第三方网络服务、中间件第三方网络服务和第三方网络服务。在至少一个实施例中，数据库第三方网络服务可支持共享服务部署模型，所述共享服务部署模型使组织能够汇聚数据库资源并以数据库第三方网络的形式向客户提供数据库即服务。在至少一个实施例中，在第三方网络基础设施系统中，中间件第三方网络服务可以为客户提供平台以开发和部署不同业务应用，并且第三方网络服务可以为客户提供平台以部署应用。

[0177] 在至少一个实施例中，各种不同的基础设施服务可由第三方网络基础设施系统中的IaaS平台提供。在至少一个实施例中，基础设施服务促进利用由SaaS平台和PaaS平台提供的服务的客户对底层计算资源(诸如存储、网络和其他基础计算资源)的管理和控制。

[0178] 在至少一个实施例中，第三方网络基础设施系统1402还可包括用于提供用于向第三方网络基础设施系统的客户提供各种服务的资源的基础设施资源1430。在至少一个实施例中，基础设施资源1430可包括硬件(诸如服务器、存储和联网资源)的预集成和优化的组合，用于执行由PaaS平台和SaaS平台提供的服务和其他资源。

[0179] 在至少一个实施例中，第三方网络基础设施系统1402中的资源可由多个用户共享并且按照需求动态地重新分配。在至少一个实施例中，可以向不同时区中的用户分配资源。在至少一个实施例中，第三方网络基础设施系统1402可以使得第一时区中的第一组用户能够利用第三方网络基础设施系统的资源持续指定小时数，并且随后使得能够将相同资源重新分配给位于不同时区中的另一组用户，从而使资源利用率最大化。

[0180] 在至少一个实施例中，可提供由第三方网络基础设施系统1402的不同组件或模块共享的多个内部共享服务1432，用于实现由第三方网络基础设施系统1402提供服务。在至少一个实施例中，这些内部共享服务可包括但不限于安全和身份服务、集成服务、企业库服务、企业管理器服务、病毒扫描和白名单服务、高可用性、备份和恢复服务、用于使能第三方网络支持的服务、电子邮件服务、通知服务、文件传输服务和/或其变体。

[0181] 在至少一个实施例中，第三方网络基础设施系统1402可在第三方网络基础设施系统中提供第三方网络服务(例如，SaaS、PaaS和IaaS服务)的全面管理。在至少一个实施例中，第三方网络管理功能可包括用于供应、管理和跟踪由第三方网络基础设施系统1402接收的客户的订阅的能力和/或其变体。

[0182] 在至少一个实施例中，如图14所示，第三方网络管理功能可以由一个或更多个模块提供，诸如订单管理模块1420、订单协调模块1422、订单供应模块1424、订单管理和监控模块1426和身份管理模块1428。在至少一个实施例中，这些模块可包括一个或更多个计算机和/或服务器或使用一个或更多个计算机和/或服务器来提供，所述一个或更多个计算机和/或服务器可以是通用计算机、专用服务器计算机、服务器农场、服务器集群或任何其他适当的布置和/或组合。

[0183] 在至少一个实施例中，在步骤1434，使用客户端设备(诸如客户端计算设备1404、

1406或1408)的客户可通过请求由第三方网络基础设施系统1402提供的一个或更多个服务并对由第三方网络基础设施系统1402提供的一个或更多个服务的订阅下订单来与第三方网络基础设施系统1402交互。在至少一个实施例中,客户可访问第三方网络用户界面(UI),诸如第三方网络UI 1412、第三方网络UI 1414和/或第三方网络UI 1416,并经由这些UI进行订购订单。在至少一个实施例中,由第三方网络基础设施系统1402响应于客户下订单而接收的订单信息可包括识别客户和由第三方网络基础设施系统1402提供的、客户想要订阅的一个或更多个服务的信息。

[0184] 在至少一个实施例中,在步骤1436,从客户接收的订单信息可存储在订单数据库1418中。在至少一个实施例中,如果这是新订单,则可以为订单创建新记录。在至少一个实施例中,订单数据库1418可以是由第三方网络基础设施系统1418操作的并且结合其他系统元件操作的若干数据库之一。

[0185] 在至少一个实施例中,在步骤1438,可以将订单信息转发到订单管理模块1420,该订单管理模块可以被配置成执行与订单相关的计费和记账功能,诸如验证订单,并且在验证后,预订一订单。

[0186] 在至少一个实施例中,在步骤1440,关于订单的信息可被传送到订单协调模块1422,该订单协调模块1422被配置为针对由客户下的订单协调服务和资源的供应。在至少一个实施例中,订单协调模块1422可以使用订单供应模块1424的服务进行供应。在至少一个实施例中,订单协调模块1422使得能够管理与每个订单相关联的业务过程,并且应用业务逻辑来确定订单是否应继续供应。

[0187] 在至少一个实施例中,在步骤1442,当接收到新订阅的订单时,订单协调模块1422向订单供应模块1424发送分配资源和配置满足订阅订单所需的资源的请求。在至少一个实施例中,订单供应模块1424实现针对由客户订购的服务的资源分配。在至少一个实施例中,订单供应模块1424提供由第三方网络基础设施系统1400提供的第三方网络服务与用于供应用于提供所请求的服务的资源的物理实现层之间的抽象级别。在至少一个实施例中,这使得订单协调模块1422能够与实现细节隔离,诸如服务和资源实际上是实时供应的,还是预先供应的并且仅在请求时分配/指派。

[0188] 在至少一个实施例中,在步骤1444,一旦服务和资源被供应,可以向订阅客户发送指示所请求的服务现在准备好使用的通知。在至少一个实施例中,信息(例如,链接)可以被发送到客户,其使客户能够开始使用所请求的服务。

[0189] 在至少一个实施例中,在步骤1446,客户订阅的订单可由订单管理和监控模块1426管理和跟踪。在至少一个实施例中,订单管理和监控模块1426可以被配置成收集关于订阅服务的客户使用的使用统计。在至少一个实施例中,可以针对所使用的存储量、所传输的数据量、用户数量、以及系统上电时间和系统下电时间的量和/或其变化来收集统计。

[0190] 在至少一个实施例中,第三方网络基础设施系统1400可包括身份管理模块1428,该身份管理模块1428被配置成提供身份服务,诸如第三方网络基础设施系统1400中的访问管理和授权服务。在至少一个实施例中,身份管理模块1428可控制关于希望利用由第三方网络基础设施系统1402提供的服务的客户的信息。在至少一个实施例中,这样的信息可以包括认证这样的客户的身份的信息和描述那些客户被授权相对于各种系统资源(例如,文件、目录、应用、通信端口、存储器段等)执行哪些动作的信息。在至少一个实施例中,身份管

理模块1428还可包括管理关于每个顾客的描述性信息以及关于可如何访问和修改该描述性信息和可由谁来访问和修改该描述性信息。

[0191] 图15示出了根据至少一个实施例的云计算环境1502。在至少一个实施例中，云计算环境1502包括一个或更多个计算机系统/服务器1504，诸如个人数字助理(PDA)或蜂窝电话1506A、台式计算机1506B、膝上型计算机1506C和/或汽车计算机系统1506N之类的计算设备与该一个或更多个计算机系统/服务器1504通信。在至少一个实施例中，这允许基础设施、平台和/或软件作为服务从云计算环境1502提供，以便不需要每个客户端单独地维护这样的资源。应当理解，图15中示出的计算设备1506A-N的类型旨在仅是说明性的，并且云计算环境1502可通过任何类型的网络和/或网络/可寻址连接(例如，使用web浏览器)与任何类型的计算机化设备通信。

[0192] 在至少一个实施例中，可被表示为云计算节点的计算机系统/服务器1504可与许多其他通用或专用计算系统环境或配置一起操作。在至少一个实施例中，可以适合于与计算机系统/服务器1504一起使用的计算系统、环境和/或配置包括但不限于个人计算机系统、服务器计算机系统、瘦客户机、厚客户端、手持式或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品，网络PC、小型计算机系统、大型计算机系统和包括任何上述系统或设备的分布式云计算环境，和/或其变体。

[0193] 在至少一个实施例中，计算机系统/服务器1504可以在由计算机系统执行的计算机系统可执行指令(诸如程序模块)的一般上下文中描述。在至少一个实施例中，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、逻辑、数据结构等。在至少一个实施例中，示例性计算机系统/服务器1504可以在分布式云计算环境中实践，其中任务由通过通信网络链接的远程处理设备来执行。在至少一个实施例中，在分布式云计算环境中，程序模块可位于包括存储器存储设备的本地和远程计算机系统存储介质两者中。

[0194] 图16示出了根据至少一个实施例的由云计算环境1202(图15)提供的一组功能抽象层。应提前理解，图16中所示的组件、层和功能仅旨在是说明性的，并且组件、层和功能可以变化。

[0195] 在至少一个实施例中，硬件和软件层1602包括硬件和软件组件。在至少一个实施例中，硬件组件的示例包括大型机、基于各种RISC(精简指令集计算机)架构的服务器、各种计算系统、超级计算系统、存储设备、网络、联网组件和/或其变体。在至少一个实施例中，软件组件的示例包括网络应用服务器软件、各种应用服务器软件、各种数据库软件、和/或其变体。

[0196] 在至少一个实施例中，虚拟化层1604提供抽象层，从该抽象层可以提供以下示例性虚拟实体：虚拟服务器、虚拟存储、虚拟网络(包括虚拟私有网络)、虚拟应用、虚拟客户端和/或其变体。

[0197] 在至少一个实施例中，管理层1606提供各种功能。在至少一个实施例中，资源供应提供用于在云计算环境内执行任务的计算资源和其他资源的动态获取。在至少一个实施例中，计量(metering)提供了在云计算环境内利用资源时的使用跟踪，以及针对这些资源的消耗的计费或发票。在至少一个实施例中，资源可以包括应用软件许可证。在至少一个实施例中，安全性为用户和任务提供身份验证，以及对数据和其他资源的保护。在至少一个实施

例中，用户界面为用户和系统管理员两者提供对云计算环境的访问。在至少一个实施例中，服务水平管理提供云计算资源分配和管理，使得满足所需的服务水平。在至少一个实施例中，服务水平协议(SLA)管理提供云计算资源的预布置和获取，根据SLA预期对该云计算资源的未来需求。

[0198] 在至少一个实施例中，工作负载层1608提供利用云计算环境的功能。在至少一个实施例中，可以从该层提供的工作负载和功能的示例包括：地图和导航、软件开发和管理、教育服务、数据分析和处理、交易处理和服务递送。

[0199] 超级计算

[0200] 以下附图阐述了但不限于可以用于实现至少一个实施例的示例性的基于超级计算机的系统。

[0201] 在至少一个实施例中，超级计算机可以指展现出显著并行性并且包括至少一个芯片的硬件系统，其中系统中的芯片通过网络互连并且被放置在分层组织的外壳中。在至少一个实施例中，用若干机架填充机房的大型硬件系统是超级计算机的特定示例，每个机架包含若干板/机架模块，每个板/机架模块包含全部由可扩展网络互连的若干芯片。在至少一个实施例中，这种大型硬件系统的单个机架是超级计算机的另一示例。在至少一个实施例中，展现出显著并行性并且包含若干硬件组件的单个芯片同样可以被认为是超级计算机，因为随着特征尺寸可能减小，可以结合在单个芯片中的硬件数量也可能增加。

[0202] 图17示出了根据至少一个实施例的芯片级的超级计算机。在至少一个实施例中，在FPGA或ASIC芯片内部，在被称为线程单元的有限状态机(1704)内执行主计算。在至少一个实施例中，任务和同步网络(1702)连接有限状态机并且被用于以正确的顺序分派线程和执行操作。在至少一个实施例中，使用存储器网络(1706,1710)来访问多级分区的片上高速缓存层级(1708,1712)。在至少一个实施例中，使用存储器控制器(1716)和片外存储器网络(1714)来访问片外存储器。在至少一个实施例中，当设计不适合于单个逻辑芯片时，I/O控制器(1718)用于跨芯片通信。

[0203] 图18示出了根据至少一个实施例的在机架模块级别的超级计算机。在至少一个实施例中，在机架模块内，存在连接至构成主加速器存储器的一个或更多个DRAM单元(1804)的多个FPGA或ASIC芯片(1802)。在至少一个实施例中，每个FPGA/ASIC芯片使用板上的宽总线用差分高速信令(1806)连接到其相邻的FPGA/ASIC芯片。在至少一个实施例中，每个FPGA/ASIC芯片还连接到至少一个高速串行通信电缆。

[0204] 图19示出了根据至少一个实施例的机架级的超级计算机。图20示出了根据至少一个实施例的整个系统级的超级计算机。在至少一个实施例中，参见图19和图20，在机架中的机架模块之间并且跨整个系统的机架，使用高速串行光缆或铜电缆(1902,2002)来实现可扩展的、可能不完整的超立方体网络。在至少一个实施例中，加速器的FPGA/ASIC芯片中的一个通过PCI-Express连接被连接到主机系统(2004)。在至少一个实施例中，主机系统包括应用的软件部分在其上运行的主机微处理器(2008)以及由与加速器上的存储器保持一致的一个或更多个主机存储器DRAM单元(2006)组成的存储器。在至少一个实施例中，主机系统可以是机架之一上的单独模块，或可以与超级计算机的模块之一集成。在至少一个实施例中，立方体连接的循环拓扑提供通信链路以为大型超级计算机创建超立方体网络。在至少一个实施例中，机架模块上的小组FPGA/ASIC芯片可充当单个超立方体节点，使得与单个

芯片相比，每组的外部链路的总数增加。在至少一个实施例中，组包含机架模块上的芯片A、B、C和D，该机架模块具有连接环形组织中的A、B、C和D的内部宽差分总线。在至少一个实施例中，存在将机架模块连接到外部世界的12条串行通信电缆。在至少一个实施例中，机架模块上的芯片A连接至串行通信电缆0、1、2。在至少一个实施例中，芯片B连接至电缆3、4、5。在至少一个实施例中，芯片C连接至6、7、8。在至少一个实施例中，芯片D连接至9、10、11。在至少一个实施例中，构成机架模块的整个组{A,B,C,D}可以形成超级计算机系统内的超立方体节点，其中多达212＝4096个机架模块(16384FPGA/ASIC芯片)。在至少一个实施例中，为了使芯片A在组{A,B,C,D}的链路4上向外发送消息，必须首先用板上差分宽总线连接将消息路由到芯片B。在至少一个实施例中，在链路4上到达去往芯片A的组{A,B,C,D}(即，到达B)的消息也必须首先被路由到组{A,B,C,D}内部的正确目的地芯片(A)。在至少一个实施例中，还可以实现其他大小的并行超级计算机系统。

[0205] 人工智能

[0206] 以下附图阐述了但不限于可以用于实现至少一个实施例的示例性的基于人工智能的系统。

[0207] 图21A示出了用于执行与一个或更多个实施例相关联的推理和/或训练操作的推理和/或训练逻辑2115。下面结合图21A和/或图21B提供关于推理和/或训练逻辑2115的细节。

[0208] 在至少一个实施例中，推理和/或训练逻辑2115可以包括但不限于，代码和/或数据存储2101，用于存储前向和/或输出权重和/或输入/输出数据，和/或在一个或更多个实施例的各方面中用于配置被训练和/或用于推理的神经网络的神经元或层的其他参数。在至少一个实施例中，训练逻辑2115可以包括或被耦合到代码和/或数据存储2101，用于存储图形代码或其他软件以控制定时和/或顺序，其中将加载权重和/或其他参数信息来配置逻辑，包括整数和/或浮点单元(统称为算术逻辑单元(ALU))。在至少一个实施例中，代码(诸如图代码)基于这样的代码所对应的神经网络的架构将权重或其他参数信息加载到处理器ALU中。在至少一个实施例中，代码和/或数据存储2101存储神经网络的每个层的权重参数和/或输入/输出数据，该神经网络在使用一个或更多个实施例的各方面的训练和/或推理期间在输入/输出数据和/或权重参数的前向传播期间与一个或更多个实施例结合训练或使用。在至少一个实施例中，代码和/或数据存储2101的任何部分可与其他片上或片外数据存储装置一起被包括，包括处理器的L1、L2或L3高速缓存存储器或系统存储器。

[0209] 在至少一个实施例中，代码和/或数据存储2101的任何部分可在一个或更多个处理器或其他硬件逻辑设备或电路内部或外部。在至少一个实施例中，代码和/或代码和/或数据存储2101可以是高速缓存存储器、动态随机可寻址存储器(“DRAM”)、静态随机可寻址存储器(“SRAM”)、非易失性存储器(例如，闪存)或其他存储装置。在至少一个实施例中，对代码和/或代码和/或数据存储2101是在处理器内部还是外部，例如，或包括DRAM、SRAM、闪存或一些其他存储类型的选择，可以取决于片上相对于片外的可用存储，正在执行的训练和/或推理功能的延时要求、在神经网络的推理和/或训练中使用的数据的批大小，或这些因素的一些组合。

[0210] 在至少一个实施例中，推理和/或训练逻辑2115可以包括但不限于：代码和/或数据存储2105，用于存储与在一个或更多个实施例的各方面中被训练和/或用于推理的神经

网络的神经元或层相对应的反向和/或输出权重和/或输入/输出数据。在至少一个实施例中，代码和/或数据存储2105存储神经网络的每一层的权重参数和/或输入/输出数据，该神经网络在使用一个或更多个实施例的各方面的训练和/或推理期间的输入/输出数据和/或权重参数的反后传播期间与一个或更多个实施例结合训练或使用。在至少一个实施例中，训练逻辑2115可以包括或被耦合到代码和/或数据存储2105，以存储图代码或其他软件来控制定时和/或顺序，其中将加载权重和/或其他参数信息以配置逻辑，包括整数和/或浮点单元(统称为算术逻辑单元(ALU))。

[0211] 在至少一个实施例中，代码(诸如图代码)使基于这样的代码所对应的神经网络的架构将权重或其他参数信息加载到处理器ALU中。在至少一个实施例中，代码和/或数据存储2105的任何部分可与其他片上或片外数据存储包括在一起，包括处理器的L1、L2或L3高速缓存或系统存储器。在至少一个实施例中，代码和/或数据存储2105的任何部分可以在一个或更多个处理器或其他硬件逻辑设备或电路的内部或外部。在至少一个实施例中，代码和/或数据存储2105可以是高速缓存存储器、DRAM、SRAM、非易失性存储器(例如，闪存)或其他存储装置。在至少一个实施例中，对代码和/或数据存储2105是在处理器内部还是外部，例如，或包括DRAM、SRAM、闪存或一些其他存储类型的选择，可以取决于片上相对于片外的可用存储，正在执行的训练和/或推理功能的延时要求、在神经网络的推理和/或训练中使用的数据的批大小，或这些因素的一些组合。

[0212] 在至少一个实施例中，代码和/或数据存储2101和代码和/或数据存储2105可为单独的存储结构。在至少一个实施例中，代码和/或数据存储2101和代码和/或数据存储2105可以是组合的存储结构。在至少一个实施例中，代码和/或数据存储2101和代码和/或数据存储2105可部分组合且部分分离。在至少一个实施例中，代码和/或数据存储2101和代码和/或数据存储2105的任何部分可与其他片上或片外数据存储(包括处理器的L1、L2或L3高速缓存或系统存储器)包括在一起。

[0213] 在至少一个实施例中，推理和/或训练逻辑2115可以包括但不限于，一个或更多个算术逻辑单元(“ALU”)2110，包括整数和/或浮点单元，用于至少部分地基于训练和/或推理代码(例如，图形代码)或由训练和/或推理代码(例如，图形代码)指示来执行逻辑和/或数学运算，其结果可以产生存储在激活存储2120中的激活(例如，来自神经网络内的层或神经元的输出值)，所述激活存储是存储在代码和/或数据存储2101和/或代码和/或数据存储2105中的输入/输出和/或权重参数数据的函数。在至少一个实施例中，根据响应于执行指令或其他代码、由ALU 2110执行的线性代数和/或基于矩阵的数学来生成存储在激活存储2120中的激活，其中存储在代码和/或数据存储2105和/或数据存储2101中的权重值与其他值(诸如偏置值、梯度信息、动量值或其他参数或超参数)一起被用作操作数，该其他值中的任何或全部值可被存储在代码和/或数据存储2105或代码和/或数据存储2101或芯片上或芯片外的另一存储中。

[0214] 在至少一个实施例中，一个或更多个ALU 2110被包括在一个或更多个处理器或其他硬件逻辑器件或电路内，而在另一个实施例中，一个或更多个ALU 2110可以在处理器或使用它们的其他硬件逻辑器件或电路(例如，协处理器)的外部。在至少一个实施例中，ALU 2110可以被包括在处理器的执行单元内或者以其他方式在可由处理器的执行单元可访问的ALU库内，所述处理器的执行单元在同一处理器内或分布在不同类型的不同处理器(例

如,中央处理单元、图形处理单元、固定功能单元等)之间。在至少一个实施例中,代码和/或数据存储2101、代码和/或数据存储2105、以及激活存储2120可以共享处理器或其他硬件逻辑器件或电路,而在另一个实施例中,它们可以在不同的处理器或其他硬件逻辑器件或电路中,或在相同和不同处理器或其他硬件逻辑器件或电路的某种组合中。在至少一个实施例中,激活存储2120的任何部分可以与其他片上或片外数据存储包括在一起,所述其他片上或片外数据存储包括处理器的L1、L2或L3高速缓存或系统存储器。此外,推理和/或训练代码可与处理器或其他硬件逻辑或电路可访问并使用处理器的获取、解码、调度、执行、引退(retirement)和/或其他逻辑电路来获取和/或处理的其他代码一起存储。

[0215] 在至少一个实施例中,激活存储2120可以是高速缓存存储器、DRAM、SRAM、非易失性存储器(例如,闪存)或其他存储装置。在至少一个实施例中,激活存储2120可以完全或部分地在一个或更多个处理器或其他逻辑电路之内或外部。在至少一个实施例中,对激活存储2120是在处理器内部还是外部,例如,或包括DRAM、SRAM、闪存或一些其他存储类型的选择,可以取决于片上相对于片外的可用存储,正在执行的训练和/或推理功能的延时要求、在神经网络的推理和/或训练中使用的数据的批大小,或这些因素的一些组合。

[0216] 在至少一个实施例中,图21A中所示出的推理和/或训练逻辑2115可以与专用集成电路(“ASIC”)结合使用,诸如来自谷歌的TensorFlow®处理单元、来自Graphcore™的推理处理单元(IPU)、或来自英特尔公司的Nervana®(例如,“Lake Crest”)处理器。在至少一个实施例中,图21A中所示出的推理和/或训练逻辑2115可以结合中央处理单元(“CPU”)硬件、图形处理单元(“GPU”)硬件或其他硬件(如现场可编程门阵列(“FPGA”))使用。

[0217] 图21B示出了根据至少一个实施例的推理和/或训练逻辑2115。在至少一个实施例中,推理和/或训练逻辑2115可包括但不限于其中计算资源是专用的或以其他方式结合与神经网络内的一个或更多个神经元层相对应的权重值或其他信息排他地使用的硬件逻辑。在至少一个实施例中,图21B中所示出的推理和/或训练逻辑2115可以结合专用集成电路(ASIC)(如来自谷歌的TensorFlow®处理单元、来自Graphcore™的推理处理单元(IPU)、或来自英特尔公司的Nervana®(例如,“Lake Crest”)处理器来使用。在至少一个实施例中,图21B中示出的推理和/或训练逻辑2115可结合中央处理单元(CPU)硬件、图形处理单元(GPU)硬件或其他硬件(诸如现场可编程门阵列(FPGA))使用。在至少一个实施例中,推理和/或训练逻辑2115包括但不限于代码和/或数据存储2101以及代码和/或数据存储2105,其可以用于存储代码(例如,图代码)、权重值和/或其他信息,包括偏置值、梯度信息、动量值和/或其他参数或超参数信息。在图21B中所说明的至少一个实施例中,代码和/或数据存储2101和代码和/或数据存储2105中的每一者分别与专用计算资源(例如,计算硬件2102和计算硬件2106)相关联。在至少一个实施例中,计算硬件2102和计算硬件2106中的每一个包括一个或更多个ALU,该一个或更多个ALU仅分别对存储在代码和/或数据存储2101和代码和/或数据存储2105中的信息执行数学函数(诸如线性代数函数),其结果被存储在激活存储2120中。

[0218] 在至少一个实施例中,每个代码和/或数据存储2101和2105以及相应的计算硬件2102和2106,分别对应于神经网络的不同层,使得来自代码和/或数据存储2101和计算硬件2102中的一个存储/计算对2101/2102的结果激活作为输入被提供给代码和/或数据存储

2105和计算硬件2106中的下一个存储/计算对2105/2106，以便镜像神经网络的概念组织。在至少一个实施例中，存储/计算对2101/2102和2105/2106中的每一个可对应于多于一个神经网络层。在至少一个实施例中，在存储/计算对2101/2102和2105/2106之后或与存储/计算对2101/2102和2105/2106并行的附加存储/计算对(未示出)可被包括在推理和/或训练逻辑2115中。

[0219] 图22示出了根据至少一个实施例的深度神经网络的训练和部署。在至少一个实施例中，使用训练数据集2202来训练未经训练的神经网络2206。在至少一个实施例中，训练框架2204是PyTorch框架，而在其他实施例中，训练框架2204是TensorFlow、Boost、Caffe、Microsoft Cognitive Toolkit/CNTK、MXNet、Chainer、Keras、Deeplearning4j或其他训练框架。在至少一个实施例中，训练框架2204对未经训练的神经网络2206进行训练，并使其能够使用本文中所描述的处理资源来训练以生成经训练的神经网络2208。在至少一个实施例中，权重可以随机选择或通过使用深度信念网络进行预训练来选择。在至少一个实施例中，训练可以以监督、部分监督或无监督的方式来执行。

[0220] 在至少一个实施例中，使用监督学习来训练未经训练的神经网络2206，其中训练数据集2202包括与用于输入的期望输出配对的输入，或者其中训练数据集2202包括具有已知输出的输入，并且神经网络2206的输出被手动地分级。在至少一个实施例中，以监督方式来训练未经训练的神经网络2206，并且处理来自训练数据集2202的输入，并将结果输出与预期或期望输出的集合进行比较。在至少一个实施例中，然后误差被反向传播通过未经训练的神经网络2206。在至少一个实施例中，训练框架2204调整控制未经训练的神经网络2206的权重。在至少一个实施例中，训练框架2204包括用于监视未经训练的神经网络2206朝向模型(诸如经训练的神经网络2208)收敛多好的工具，该模型适于基于输入数据(诸如新数据集2212)来生成正确答案(诸如结果2214)。在至少一个实施例中，训练框架2204重复地训练未经训练的神经网络2206，同时使用损失函数和调整算法(诸如随机梯度下降)来调整权重以精炼未经训练的神经网络2206的输出。在至少一个实施例中，训练框架2204训练未经训练的神经网络2206，直到未经训练的神经网络2206实现所期望的准确度。在至少一个实施例中，经训练的神经网络2208然后可被部署以实现任何数量的机器学习操作。

[0221] 在至少一个实施例中，使用无监督学习来训练未经训练的神经网络2206，其中未经训练的神经网络2206尝试使用未标记的数据来训练其自身。在至少一个实施例中，无监督学习训练数据集2202将包括输入数据而没有任何相关联的输出数据或“地面真值”数据。在至少一个实施例中，未经训练的神经网络2206可以学习训练数据集2202内的分组，并且可以确定各个输入如何与未经训练的数据集2202相关。在至少一个实施例中，无监督训练可被用于在经训练的神经网络2208中生成能够执行在减少新数据集2212的维度中有用的操作的自组织映射。在至少一个实施例中，无监督训练还可用于执行异常检测，其允许识别新数据集2212中偏离新数据集2212的正常模式的数据点。

[0222] 在至少一个实施例中，可以使用半监督学习，半监督学习是其中在训练数据集2202中包括标记数据和未标记数据的混合的技术。在至少一个实施例中，训练框架2204可被用于执行增量学习，诸如通过转移学习技术。在至少一个实施例中，增量学习使得经训练的神经网络2208能够适应新的数据集2212，而不会忘记在初始训练期间注入在经训练的神经网络1408内的知识。

[0223] 5G网络

[0224] 以下附图阐述了但不限于可以用于实现至少一个实施例的示例性基于5G网络的系统。

[0225] 图23示出了根据至少一个实施例的网络的系统2300的架构。在至少一个实施例中，系统2300被示为包括用户设备(UE)2302和UE 2304。在至少一个实施例中，UE 2302和2304被示为智能电话(例如，可连接到一个或更多个蜂窝网络的手持触摸屏移动计算设备)，但还可包括任何移动或非移动计算设备，诸如个人数字助理(PDA)、寻呼机、膝上型计算机、台式计算机、无线手持设备或包括无线通信接口的任何计算设备。

[0226] 在至少一个实施例中，UE 2302和UE 2304中的任何一个可包括物联网(IoT)UE，该IoT UE可包括为利用短暂UE连接的低功率IoT应用设计的网络接入层。在至少一个实施例中，IoT UE可利用诸如用于经由公共陆地移动网络(PLMN)、基于邻近的服务(ProSe)或设备到设备(D2D)通信、传感器网络或IoT网络与MTC服务器或设备交换数据的技术，诸如机器对机器(M2M)或机器类型通信(MTC)。在至少一个实施例中，M2M或MTC数据交换可以是机器发起的数据交换。在至少一个实施例中，IoT网络描述互连IoT UE，该IoT UE可包括具有短寿命连接的可唯一标识的嵌入式计算设备(在互联网基础结构内)。在至少一个实施例中，IoT UE可执行后台应用(例如，保活消息、状态更新等)以促进IoT网络的连接。

[0227] 在至少一个实施例中，UE 2302和UE 2304可以被配置为与无线电接入网(RAN)2316连接(例如，通信地耦合)。在至少一个实施例中，RAN 2316例如可以是演进的通用移动电信系统(UMTS)陆地无线电接入网络(E-UTRAN)、NextGen RAN(NG RAN)或一些其他类型的RAN。在至少一个实施例中，UE 2302和UE 2304分别利用连接2312和连接2314，每个连接包括物理通信接口或层。在至少一个实施例中，连接2312和2314被示为空中接口，用于实现通信耦合，并且可以与蜂窝通信协议一致，诸如全球移动通信系统(GSM)协议，码分多址(CDMA)网络协议、即按即讲(PTT)协议、蜂窝PTT(POC)协议、通用移动电信系统(UMTS)协议、3GPP长期演进(LTE)协议、第五代(5G)协议、新无线电(NR)协议及其变型。

[0228] 在至少一个实施例中，UE 2302和2304还可经由ProSe接口2306直接交换通信数据。在至少一个实施例中，ProSe接口2306可替代地被称为边链路接口，其包括一个或更多个逻辑信道，包括但不限于物理边链路控制信道(PSCCH)、物理边链路共享信道(PSSCH)、物理边链路发现信道(PSDCH)和物理边链路广播信道(PSBCH)。

[0229] 在至少一个实施例中，UE 2304被示为配置成经由连接2308接入接入点(AP)2310。在至少一个实施例中，连接2308可以包括本地无线连接，诸如与任何IEEE 802.11协议一致的连接，其中AP 2310将包括无线保真(WiFi®)路由器。在至少一个实施例中，AP 2310被示为连接到互联网而不连接到无线系统的核心网。

[0230] 在至少一个实施例中，RAN 2316可包括启用连接2312和2314的一个或更多个接入节点。在至少一个实施例中，这些接入节点(AN)可被称为基站(BS)、NodeB、演进型NodeB(eNB)、下一代NodeB(gNB)、RAN节点等，并且可包括地面站(例如，地面接入点)或提供地理区域(例如，小区)内的覆盖的卫星站。在至少一个实施例中，RAN 2316可包括用于提供宏蜂窝小区的一个或更多个RAN节点(例如，宏RAN节点2318)和用于提供毫微微蜂窝小区或微微蜂窝小区(例如，与宏蜂窝小区相比具有较小覆盖区域、较小用户容量、或较高带宽的蜂窝小区)的一个或更多个RAN节点(例如，低功率(LP)RAN节点2320)。

[0231] 在至少一个实施例中,RAN节点2318和2320中的任一个可终止空中接口协议并且可以为UE 2302和2304的第一联系点。在至少一个实施例中,RAN节点2318和2320中的任一个可实现RAN 2316的各种逻辑功能,包括但不限于无线电网络控制器(RNC)功能,诸如无线电承载管理、上行链路和下行链路动态无线电资源管理和数据分组调度和移动性管理。

[0232] 在至少一个实施例中,UE 2302和UE 2304可被配置为使用正交频分复用(OFDM)通信信号根据各种通信技术通过多载波通信信道彼此通信或者与RAN节点2318和RAN节点2320中的任一个通信,通信技术诸如但不限于正交频分多址(OFDMA)通信技术(例如,用于下行链路通信)或单载波频分多址(SC-FDMA)通信技术(例如,用于上行链路和ProSe或边链路通信),和/或其变体。在至少一个实施例中,OFDM信号可包括多个正交子载波。

[0233] 在至少一个实施例中,下行链路资源网格可以用于从RAN节点2318和2320中的任一个到UE 2302和2304的下行链路传输,而上行链路传输可以利用类似的技术。在至少一个实施例中,网格可以是称为资源网格或时频资源网格的时频网格,其是每个时隙中下行链路中的物理资源。在至少一个实施例中,这种时频平面表示是OFDM系统的常见实践,这使得其对于无线电资源分配来说是直观的。在至少一个实施例中,资源网格的每列和每行分别对应于一个OFDM符号和一个OFDM子载波。在至少一个实施例中,时域中的资源网格的持续时间对应于无线电帧中的一个时隙。在至少一个实施例中,资源网格中的最小时间-频率单元被表示为资源元素。在至少一个实施例中,每个资源网格包括多个资源块,其描述某些物理信道到资源元素的映射。在至少一个实施例中,每个资源块包括资源元素的集合。在至少一个实施例中,在频域中,这可以表示当前可以被分配的最小数量的资源。在至少一个实施例中,存在使用这样的资源块传送的若干不同的物理下行链路信道。

[0234] 在至少一个实施例中,物理下行链路共享信道(PDSCH)可以运载用户数据和更高层信令给UE 2302和2304。在至少一个实施例中,物理下行链路控制信道(PDCCH)可运载关于与PDSCH信道相关的传输格式和资源分配的信息等。在至少一个实施例中,其还可以向UE 2302和2304通知与上行链路共享信道有关的传输格式、资源分配和HARQ(混合自动重传请求)信息。在至少一个实施例中,通常,下行链路调度(将控制和共享信道资源块分配给小区内的UE 2302)可以在RAN节点2318和2320中的任一个处基于从UE 2302和2304中的任一个反馈的信道质量信息来执行。在至少一个实施例中,下行链路资源分配信息可以在用于(例如分配给)UE 2302和2304中的每一个的PDCCH上发送。

[0235] 在至少一个实施例中,PDCCH可以使用控制信道元素(CCE)来传送控制信息。在至少一个实施例中,在被映射到资源元素之前,PDCCH复值符号可以首先被组织成四元组,然后可以使用子块交织器对其进行置换以用于速率匹配。在至少一个实施例中,可以使用这些CCE中的一个或更多个来传送每个PDCCH,其中每个CCE可以对应于被称为资源元素组(REG)的四个物理资源元素的九个集合。在至少一个实施例中,四个正交相移键控(QPSK)符号可以被映射到每个REG。在至少一个实施例中,取决于下行链路控制信息(DCI)的大小和信道条件,可以使用一个或更多个CCE来发送PDCCH。在至少一个实施例中,可以有在LTE中定义的具有不同数目的CCE的四个或更多个不同的PDCCH格式(例如,聚合等级,L=1、2、4或8)。

[0236] 在至少一个实施例中,使用PDSCH资源的增强型物理下行链路控制信道(EPDCCH)可以用于控制信息传输。在至少一个实施例中,可以使用一个或更多个增强型控制信道元

素(ECCE)来发送EPDCCH。在至少一个实施例中,每个ECCE可以对应于被称为增强型资源元素组(EREG)的四个物理资源元素的九个集合。在至少一个实施例中,ECCE在一些情况下可以具有其他数目的EREG。

[0237] 在至少一个实施例中,RAN 2316被示为经由S1接口2322通信地耦合至核心网(CN)2338。在至少一个实施例中,CN 2338可以是演进的分组核心(EPC)网络、NextGen分组核心(NPC)网络或一些其他类型的CN。在至少一个实施例中,S1接口2322被分成两部分:S1-U接口2326,其运载RAN节点2318和2320与服务网关(S-GW)2330之间的业务数据;以及S1-移动性管理实体(MME)接口2324,其为RAN节点2318和2320与MME 2328之间的信令接口。

[0238] 在至少一个实施例中,CN 2338包括MME 2328、S-GW 2330、分组数据网络(PDN)网关(P-GW)2334和归属订户服务器(HSS)2332。在至少一个实施例中,MME 2328可以在功能上类似于传统服务通用分组无线电服务(GPRS)支持节点(SGSN)的控制平面。在至少一个实施例中,MME 2328可以管理接入中的移动性方面,例如网关选择和跟踪区域列表管理。在至少一个实施例中,HSS 2332可以包括用于网络用户的数据库,该数据库包括用于支持网络实体处理通信会话的订阅相关信息。在至少一个实施例中,CN 2338可以包括一个或更多个HSS 2332,这取决于移动用户的数量、设备的容量、网络的组织等。在至少一个实施例中,HSS 2332可以提供对路由/漫游、认证、授权、命名/寻址解析、位置依赖性等的支持。

[0239] 在至少一个实施例中,S-GW 2330可以终止朝向RAN 2316的S1接口2322,并且在RAN 2316和CN 2338之间路由数据分组。在至少一个实施例中,S-GW 2330可以是用于RAN间节点切换的本地移动性锚点,并且还可以提供用于3GPP间移动性的锚点。在至少一个实施例中,其他责任可以包括合法拦截、收费和一些策略强制执行。

[0240] 在至少一个实施例中,P-GW 2334可以终止朝向PDN的SGi接口。在至少一个实施例中,P-GW 2334可以经由互联网协议(IP)接口2342在EPC网络2338和外部网络(诸如包括应用服务器2340(或者称为应用功能(AF))的网络)之间路由数据分组。在至少一个实施例中,应用服务器2340可以是采用核心网络(例如,UMTS分组服务(PS)域、LTE PS数据服务等)提供使用IP承载资源的应用的元件。在至少一个实施例中,P-GW 2334被示出为经由IP通信接口2342通信地耦合到应用服务器2340。在至少一个实施例中,应用服务器2340还可被配置为经由CN 2338支持UE 2302和2304的一个或更多个通信服务(例如,互联网协议语音(VoIP)会话、PTT会话、群组通信会话、社交网络服务等)。

[0241] 在至少一个实施例中,P-GW 2334还可以是用于策略实施和收费数据收集的节点。在至少一个实施例中,策略和计费执行功能(PCRF)2336是CN 2338的策略和计费控制元件。在至少一个实施例中,在非漫游场景中,在与UE的互联网协议连接性接入网络(IP-CAN)会话相关联的归属公共陆地移动网络(HPLMN)中可以存在单个PCRF。在至少一个实施例中,在具有本地流量突破的漫游场景中,可存在与UE的IP-CAN会话相关联的两个PCRF:HPLMN内的归属PCRF(H-PCRF)和受访公共陆地移动网络(VPLMN)内的受访PCRF(V-PCRF)。在至少一个实施例中,PCRF 2336可以经由P-GW 2334通信地耦合到应用服务器2340。在至少一个实施例中,应用服务器2340可以向PCRF 2336发信号,以指示新的服务流,并选择适当的服务质量(QoS)和计费参数。在至少一个实施例中,PCRF 2336可以将这个规则供应到具有适当的业务流模板(TFT)和标识符的QoS类(QCI)的策略和计费执行功能(PCEF)(未示出),所述PCEF开始由应用服务器2340指定的QoS和计费。

[0242] 图24示出了根据一些实施例的网络的系统2400的架构。在至少一个实施例中，系统2400被示为包括UE 2402、5G接入节点或RAN节点(被示为(R)AN节点2408)，用户平面功能(被示出为UPF 2404)，数据网络(DN 2406)，例如，其可以是运营商服务、互联网接入或第三方服务、以及5G核心网络(5GC)(示为CN 2410)。

[0243] 在至少一个实施例中，CN 2410包括认证服务器功能(AUSF 2414)；核心接入和移动性管理功能(AMF 2412)；会话管理功能(SMF 2418)；网络暴露功能(NEF 2416)；策略控制功能(PCF 2422)；网络功能(NF)储存库功能(NRF 2420)；统一数据管理(UDM 2424)；以及应用功能(AF 2426)。在至少一个实施例中，CN 2410还可包括未示出的其他元件，诸如结构化数据存储网络功能(SDSF)、非结构化数据存储网络功能(UDSF)及其变型。

[0244] 在至少一个实施例中，UPF 2404可充当RAT内和RAT间移动性的锚点、互连到DN 2406的外部PDU会话点、和支持多归属PDU会话的分支点。在至少一个实施例中，UPF 2404还可以执行分组路由和转发、分组检查、实施策略规则的用户平面部分、合法拦截分组(UP收集)；业务使用报告、为用户平面执行QoS处理(例如分组过滤、门控、UL/DL速率执行)、执行上行链路业务验证(例如，SDF到QoS流映射)、上行链路和下行链路中的传输级分组标记、以及下行链路分组缓存和下行链路数据通知触发。在至少一个实施例中，UPF 2404可包括上行链路分类器，用于支持将业务流路由到数据网络。在至少一个实施例中，DN 2406可表示各种网络运营商服务、互联网接入或第三方服务。

[0245] 在至少一个实施例中，AUSF 2414可以存储用于UE 2402的认证的数据，并且处理与认证相关的功能。在至少一个实施例中，AUSF 2414可以促进用于各种接入类型的公共认证框架。

[0246] 在至少一个实施例中，AMF 2412可以负责注册管理(例如，用于注册UE 2402等)、连接管理、可达性管理、移动性管理、和AMF相关事件的合法拦截、以及接入认证和授权。在至少一个实施例中，AMF 2412可以为SMF 2418提供SM消息的传输，并且充当用于路由SM消息的透明代理。在至少一个实施例中，AMF 2412还可以提供UE 2402与SMS功能(SMSF)(图24未示出)之间的短消息服务(SMS)消息的传输。在至少一个实施例中，AMF 2412可以充当安全锚定功能(SEA)，其可以包括与AUSF 2414和UE 2402的交互以及接收作为UE 2402认证过程的结果而建立的中间密钥。在至少一个实施例中，在使用基于USIM的认证的情况下，AMF 2412可以从AUSF 2414检索安全材料。在至少一个实施例中，AMF 2412还可以包括安全上下文管理(SCM)功能，其从SEA接收它用来导出接入网络专用密钥的密钥。此外，在至少一个实施例中，AMF 2412可以是RAN CP接口的终止点(N2参考点)、NAS(NI)信令的终止点，并且执行NAS加密和完整性保护。

[0247] 在至少一个实施例中，AMF 2412还可以支持通过N3互通功能(IWF)接口与UE 2402的NAS信令。在至少一个实施例中，N3IWF可以用于提供对不受信实体的访问。在至少一个实施例中，N3IWF可以分别是控制平面和用户平面的N2和N3接口的终止点，因此，可针对PDU会话和QoS处理来自SMF和AMF的N2信令，对IPSec和N3隧道的分组进行封装/解封装，在上行链路中标记N3用户平面分组，并且考虑到与通过N2接收的这种标记相关联的QoS要求，实施对应于N3分组标记的QoS。在至少一个实施例中，N3IWF还可以在UE 2402和AMF 2412之间中继上行链路和下行链路控制平面NAS(NI)信令，并且在UE 2402和UPF 2404之间中继上行链路和下行链路用户平面分组。在至少一个实施例中，N3IWF还提供用于与UE 2402的IPsec隧道

建立的机制。

[0248] 在至少一个实施例中,SMF 2418可负责会话管理(例如,会话建立、修改和释放,包括UPF和AN节点之间的隧道保持);UE IP地址分配和管理(包括可选的授权);UP功能的选择和控制;在UPF处配置流量转向以将流量路由到合适的目的地;朝向策略控制功能的接口终止;策略强制执行和QoS的控制部分;合法拦截(用于SM事件和到LI系统的接口);NAS消息的SM部分的终止;下行链路数据通知;AN特定SM信息的发起者,其经由AMF在N2上发送到AN;确定会话的SSC模式。在至少一个实施例中,SMF 2418可包括以下漫游功能:处理本地实施以应用QoS SLAB(VPLMN);收费数据收集和收费接口(VPLMN);合法拦截(在VPLMN中用于SM事件并且接口到LI系统);支持与外部DN交互以传输用于由外部DN进行的PDU会话授权/认证的信令。

[0249] 在至少一个实施例中,NEF 2416可以提供用于安全地暴露由3GPP网络功能为第三方提供的服务和能力、内部暴露/重新暴露、应用功能(例如,AF 2426)、边缘计算或雾计算系统等的装置。在至少一个实施例中,NEF 2416可认证、授权和/或节流AF。在至少一个实施例中,NEF 2416还可以转换与AF 2426交换的信息和与内部网络功能交换的信息。在至少一个实施例中,NEF 2416可以在AF服务标识符和内部5GC信息之间转换。在至少一个实施例中,NEF 2416还可以基于其他网络功能的暴露的能力从其他网络功能(NF)接收信息。在至少一个实施例中,该信息可作为结构化数据存储在NEF 2416处,或使用标准化接口存储在数据存储NF处。在至少一个实施例中,所存储的信息然后可由NEF 2416重新暴露给其他NF和AF,和/或用于其他目的,诸如分析。

[0250] 在至少一个实施例中,NRF 2420可以支持服务发现功能,从NF实例接收NF发现请求,以及向NF实例提供所发现的NF实例的信息。在至少一个实施例中,NRF 2420还维护可用NF实例及其支持的服务的信息。

[0251] 在至少一个实施例中,PCF 2422可向控制平面功能提供策略规则以实施它们,并且还可支持统一策略框架以管理网络行为。在至少一个实施例中,PCF 2422还可实现前端(FE),用于访问UDM 2424的UDR中与策略决策相关的订阅信息。

[0252] 在至少一个实施例中,UDM 2424可处理订阅相关信息以支持网络实体处理通信会话,并且可存储UE 2402的订阅数据。在至少一个实施例中,UDM 2424可以包括两个部分,应用FE和用户数据存储库(UDR)。在至少一个实施例中,UDM可以包括UDM FE,该UDM FE负责处理凭证、位置管理、订阅管理等。在至少一个实施例中,若干不同前端可在不同交易中服务同一用户。在至少一个实施例中,UDM-FE访问UDR中存储的子订阅信息,并执行认证凭证处理;用户标识处理;接入授权;注册/移动性管理;以及订阅管理。在至少一个实施例中,UDR可与PCF 2422交互。在至少一个实施例中,UDM 2424还可支持SMS管理,其中SMS-FE实现如前所述的类似应用逻辑。

[0253] 在至少一个实施例中,AF 2426可以提供对业务路由的应用影响、对网络能力暴露(NCE)的接入,以及与策略框架的交互以用于策略控制。在至少一个实施例中,NCE可以是允许5GC和AF 2426经由NEF 2416向彼此提供信息的机制,NEF 2416可以用于边缘计算实现。在至少一个实施例中,网络运营商和第三方服务可被托管在UE 2402的附接接入点附近,以通过减少的端到端延时和传输网络上的负载来实现高效的服务递送。在至少一个实施例中,对于边缘计算实现,5GC可选择靠近UE 2402的UPF 2404,并经由N6接口执行从UPF 2404

到DN 2406的业务引导。在至少一个实施例中，这可以基于由AF 2426提供的UE订阅数据、UE位置和信息。在至少一个实施例中，AF 2426可以影响UPF(重新)选择和业务路由。在至少一个实施例中，基于运营商部署，当AF 2426被认为是受信实体时，网络运营商可以允许AF 2426直接与相关NF交互。

[0254] 在至少一个实施例中，CN 2410可以包括SMSF，其可以负责SMS订阅检查和验证，并且中继去往/来自UE 2402的SM消息到/来自其他实体，例如SMS-GMSC/IWMSC/SMS路由器。在至少一个实施例中，SMS还可以与AMF 2412和UDM 2424交互，以用于UE 2402可用于SMS传送的通知过程(例如，设置UE不可达标志，并且当UE 2402可用于SMS时通知UDM 2424)。

[0255] 在至少一个实施例中，系统2400可以包括以下基于服务的接口:Namf:AMF展现的基于服务的接口;Nsmf:SMF展现的基于服务的接口;Nnef:NEF展现的基于服务的接口;Npcf:PCF展现的基于服务的接口;Nudm:UDM展现的基于服务的接口;Naf:AF展现的基于服务的接口;Nnrf:NRF展现的基于服务的接口;以及Nausf:AUSF展现的基于服务的接口。

[0256] 在至少一个实施例中，系统2400可以包括以下参考点:N1:UE和AMF之间的参考点;N2:(R)AN和AMF之间的参考点;N3:(R)AN和UPF之间的参考点;N4:SMF和UPF之间的参考点;以及N6:UPF和数据网络之间的参考点。在至少一个实施例中，NF中的NF服务之间可能存在更多的参考点和/或基于服务的接口，然而，为清楚起见，这些接口和参考点已经被省略。在至少一个实施例中，NS参考点可以在PCF与AF之间;N7参考点可以在PCF与SMF之间;N11参考点在AMF与SMF之间等等。在至少一个实施例中，CN 2410可以包括Nx接口，Nx接口是MME和AMF 2412之间的CN间接口，以便实现CN 2410和CN 7224之间的互通。

[0257] 在至少一个实施例中，系统2400可包括多个RAN节点(诸如(R)AN节点2408)，其中在连接到5GC 410的两个或更多个(R)AN节点2408(例如，gNB)之间，在连接到CN 2410的(R)AN节点2408(例如gNB)和eNB(例如宏RAN节点)之间，和/或在连接到CN 2410的两个eNB之间定义Xn接口。

[0258] 在至少一个实施例中，Xn接口可以包括Xn用户平面(Xn-U)接口和Xn控制平面(Xn-C)接口。在至少一个实施例中，Xn-U可以提供用户平面PDU的无保证的递送，并且支持/提供数据转发和流控制功能。在至少一个实施例中，Xn-C可提供管理和错误处理功能、管理Xn-C接口的功能;对处于连接模式(例如，CM-CONNECTED)的UE 2402的移动性支持，其包括管理针对一个或更多个(R)AN节点2408之间的连接模式的UE移动性的功能。在至少一个实施例中，移动性支持可包括从旧(源)服务(R)AN节点2408到新(目标)服务(R)AN节点2408的上下文传送;以及控制旧(源)服务(R)AN节点2408至新(目标)服务(R)AN节点2408之间的用户平面隧道。

[0259] 在至少一个实施例中，Xn-U的协议栈可以包括在互联网协议(IP)传输层上构建的传输网络层和在UDP和/或一个或更多个IP层的顶部上用于承载用户平面PDU的GTP-U层。在至少一个实施例中，Xn-C协议栈可以包括应用层信令协议(称为Xn应用协议(Xn-AP))和建立在SCTP层上的传输网络层。在至少一个实施例中，SCTP层可以在IP层的顶部上。在至少一个实施例中，SCTP层提供应用层消息的有保证的递送。在至少一个实施例中，在传输IP层中，点对点传输被用于递送信令PDU。在至少一个实施例中，Xn-U协议栈和/或Xn-C协议栈可以与本文中示出和描述的用户平面和/或控制平面协议栈相同或相似。

[0260] 图25是根据一些实施例的控制平面协议栈的图示。在至少一个实施例中，控制平

面2500被示为UE 2302(或可替代地,UE 2304)、RAN 2316和MME 2328之间的通信协议栈。

[0261] 在至少一个实施例中,PHY层2502可以通过一个或更多个空中接口发送或接收由MAC层2504使用的信息。在至少一个实施例中,PHY层2502还可以执行链路自适应或自适应调制和编码(AMC)、功率控制、小区搜索(例如,用于初始同步和切换目的)和由较高层(例如,RRC层2510)使用的其他测量。在至少一个实施例中,PHY层2502还可以进一步对传输信道执行错误检测、传输信道的前向纠错(FEC)编码/解码、物理信道的调制/解调、交织、速率匹配、映射到物理信道、以及多输入多输出(MIMO)天线处理。

[0262] 在至少一个实施例中,MAC层2504可执行逻辑信道和传输信道之间的映射,将来自一个或更多个逻辑信道的MAC服务数据单元(SDU)复用到要经由传输信道递送到PHY的传输块(TB)上,将MAC SDU从经由传输信道从PHY递送的传输块(TB)解复用到一个或更多个逻辑信道,将MAC SDU复用到TB上、调度信息报告、通过混合自动重传请求(HARD)的纠错,以及逻辑信道优先化。

[0263] 在至少一个实施例中,RLC层2506可在多种操作模式下操作,包括:透明模式(TM)、非确认模式(UM)和确认模式(AM)。在至少一个实施例中,RLC层2506可以执行上层协议数据单元(PDU)的传输、通过针对AM数据传输的自动重复请求(ARQ)的纠错、以及用于UM和AM数据传输的RLC SDU的级联、分段和重组。在至少一个实施例中,RLC层2506还可以执行用于AM数据传输的RLC数据PDU的重新分段,对用于UM和AM数据传输的RLC数据PDU进行重排序,检测用于UM和AM数据传输的重复数据,丢弃用于UM和AM数据传输的RLC SDU,检测AM数据传输的协议错误,以及执行RLC重建。

[0264] 在至少一个实施例中,PDCP层2508可以执行IP数据的报头压缩和解压缩,维持PDCP序列号(SN),在重建较低层时执行较高层PDU的序列内递送,在为映射在RLC AM上的无线电承载重建较低层时消除较低层SDU的复制,对控制平面数据进行加密和解密,对控制平面数据进行完整性保护和完整性验证,基于控制定时器的数据丢弃,以及执行安全操作(例如,加密、解密、完整性保护、完整性验证等)。

[0265] 在至少一个实施例中,RRC层2510的主要服务和功能可包括系统信息的广播(例如,包括在与非接入层(NAS)相关的主信息块(MIB)或系统信息块(SIB)中),与接入层(AS)有关的系统信息的广播、UE与E-UTRAN之间的RRC连接的寻呼、建立、维持和释放(例如,RRC连接寻呼、RRC连接建立、RRC连接修改和RRC连接释放),点对点无线承载的建立、配置、维护和释放,包括密钥管理的安全功能,无线电接入技术(RAT)间移动性,以及用于UE测量报告的测量配置。在至少一个实施例中,所述MIB和SIB可包括一个或更多个信息元素(IE),每个信息元素可包括单独的数据字段或数据结构。

[0266] 在至少一个实施例中,UE 2302和RAN 2316可以利用Uu接口(例如,LTE-Uu接口)来经由包括PHY层2502、MAC层2504、RLC层2506、PDCP层2508和RRC层2510的协议栈交换控制平面数据。

[0267] 在至少一个实施例中,非接入层(NAS)协议(NAS协议2512)形成UE 2302与MME 2328之间的控制平面的最高层。在至少一个实施例中,NAS协议2512支持UE 2302的移动性和会话管理过程以建立和维持UE 2302与P-GW 2334之间的IP连接。

[0268] 在至少一个实施例中,Si应用协议(Si-AP)层(Si-AP层2522)可以支持Si接口的功能并且包括基本过程(EP)。在至少一个实施例中,EP是RAN 2316和CN 2328之间的交互单

元。在至少一个实施例中,S1-AP层服务可以包括两个组:UE关联服务和非UE关联服务。在至少一个实施例中,这些服务执行功能,包括但不限于:E-UTRAN无线电接入承载(E-RAB)管理、UE能力指示、移动性、NAS信令传输、RAN信息管理(RIM)和配置转移。

[0269] 在至少一个实施例中,流控制传输协议(SCTP)层(可替代地称为流控制传输协议/互联网协议(SCTP/IP)层)(SCTP层2520)可以部分地基于IP层2518所支持的IP协议来确保RAN 2316与MME 2328之间的信令消息的可靠传递。在至少一个实施例中,L2层2516和L1层2514可以指由RAN节点和MME用来交换信息的通信链路(例如,有线或无线)。

[0270] 在至少一个实施例中,RAN 2316和一个或更多个MME 2328可以利用S1-MME接口来经由包括L1层2514、L2层2516、IP层2518、SCTP层2520和Si-AP层2522的协议栈交换控制平面数据。

[0271] 图26是根据至少一个实施例的用户平面协议栈的图示。在至少一个实施例中,用户平面2600被示为UE 2302、RAN 2316、S-GW 2330和P-GW 2334之间的通信协议栈。在至少一个实施例中,用户平面2600可以利用与控制平面2500相同的协议层。在至少一个实施例中,UE 2302和RAN 2316可以利用Uu接口(例如,LTE-Uu接口)来经由包括PHY层2502、MAC层2504、RLC层2506、PDCP层2508的协议栈来交换用户平面数据。

[0272] 在至少一个实施例中,用于用户平面的通用分组无线电服务(GPRS)隧道协议(GTP-U)层(GTP-U层2604)可以用于在GPRS核心网络内和在无线电接入网络和核心网络之间运载用户数据。在至少一个实施例中,所传输的用户数据例如可以是IPv4、IPv6或PPP格式中的任何格式的分组。在至少一个实施例中,UDP和IP安全(UDP/IP)层(UDP/IP层2602)可以提供数据完整性的校验和、用于在源和目的地寻址不同功能的端口号、以及对所选数据流的加密和认证。在至少一个实施例中,RAN 2316和S-GW 2330可以利用S1-U接口来经由包括L1层2514、L2层2516、UDP/IP层2602和GTP-U层2604的协议栈来交换用户平面数据。在至少一个实施例中,S-GW 2330和P-GW 2334可以利用S5/S8a接口来经由包括L1层2514、L2层2516、UDP/IP层2602和GTP-U层2604的协议栈交换用户平面数据。在至少一个实施例中,如以上关于图25所讨论的,NAS协议支持UE 2302的移动性和会话管理过程以建立和维持UE 2302与P-GW 2334之间的IP连接。

[0273] 图27示出了根据至少一个实施例的核心网络的组件2700。在至少一个实施例中,CN 2038的组件可以在一个物理节点或单独的物理节点中实现,所述单独的物理节点包括用于从机器可读介质或计算机可读介质(例如,非暂态机器可读存储介质)读取和执行指令的组件。在至少一个实施例中,网络功能虚拟化(NFV)用于经由存储在一个或更多个计算机可读存储介质(以下进一步详细描述)中的可执行指令来虚拟化任何或所有的上述网络节点功能。在至少一个实施例中,CN 2338的逻辑实例化可以被称为网络切片2702(例如,网络切片2702被示出为包括HSS 2332、MME 2328和S-GW 2330)。在至少一个实施例中,CN 2338的一部分的逻辑实例化可以被称为网络子切片2704(例如,网络子切片2704被示出为包括P-GW 2334和PCRF 2336)。

[0274] 在至少一个实施例中,NFV架构和基础设施可以用于将一个或更多个网络功能虚拟化到包括行业标准服务器硬件、存储硬件或交换机的组合的物理资源上,所述网络功能可替代地由专用硬件执行。在至少一个实施例中,NFV系统可用于执行一个或更多个EPC组件/功能的虚拟或可重新配置的实现方式。

[0275] 图28是示出了根据至少一个实施例的用于支持网络功能虚拟化(NFV)的系统2800的组件的框图。在至少一个实施例中,系统2800被示为包括虚拟化基础设施管理器(被示为VIM 2802)、网络功能虚拟化基础设施(如NFVI 2804所示)、VNF管理器(如VNFM 2806所示)、虚拟化网络功能(示出为VNF 2808)、元件管理器(示出为EM 2810)、NFV协调器(示出为NFVO 2812)、以及网络管理器(示为NM 2814)。

[0276] 在至少一个实施例中,VIM 2802管理NFVI 2804的资源。在至少一个实施例中,NFVI 2804可包括用于执行系统2800的物理或虚拟资源和应用(包括管理程序)。在至少一个实施例中,VIM 2802可以利用NFVI 2804来管理虚拟资源的生命周期(例如,与一个或更多个物理资源相关联的虚拟机(VM)的创建、维护和拆除)、跟踪VM实例、跟踪性能、VM实例和相关联的物理资源的故障和安全性、以及向其他管理系统暴露VM实例和相关联的物理资源。

[0277] 在至少一个实施例中,VNFM 2806可以管理VNF 2808。在至少一个实施例中,VNF 2808可以用于执行EPC组件/功能。在至少一个实施例中,VNFM 2806可以管理VNF 2808的生命周期并且跟踪VNF 2808的虚拟方面的性能、故障和安全性。在至少一个实施例中,EM 2810可以跟踪VNF 2808的功能方面的性能、故障和安全性。在至少一个实施例中,跟踪来自VNFM 2806和EM 2810的数据可以包括,在至少一个实施例中,由VIM 2802或NFVI 2804使用的性能测量(PM)数据。在至少一个实施例中,VNFM 2806和EM 2810两者可以放大/缩小系统2800的VNF的数量。

[0278] 在至少一个实施例中,NFVO 2812可协调、授权、释放和占用NFVI 2804的资源,以便提供所请求的服务(例如,以执行EPC功能、组件或切片)。在至少一个实施例中,NM 2814可提供负责管理网络的终端用户功能包,该网络可包括具有VNF、非虚拟化网络功能或两者的网络元件(VNF的管理可经由EM 2810发生)。

[0279] 基于计算机的系统

[0280] 以下各图提出但不限于可用于实现至少一个实施例的示例性的基于计算机的系统。

[0281] 图29示出了根据至少一个实施例的处理系统2900。在至少一个实施例中,系统2900包括一个或更多个处理器2902和一个或更多个图形处理器2908,并且可以是单处理器台式机系统、多处理器工作站系统或具有大量处理器2902或处理器核心2907的服务器系统。在至少一个实施例中,处理系统2900是结合在片上系统(SoC)集成电路内的处理平台,以用于移动、手持或嵌入式设备。

[0282] 在至少一个实施例中,处理系统2900可以包括或结合在基于服务器的游戏平台中,包括游戏和媒体控制台的游戏控制台、移动游戏控制台、手持游戏控制台或在线游戏控制台。在至少一个实施例中,处理系统2900是移动电话、智能电话、平板计算设备或移动互联网设备。在至少一个实施例中,处理系统2900还可包括与可穿戴设备耦合或集成在可穿戴设备中,例如智能手表可穿戴设备、智能眼镜设备、增强现实设备或虚拟现实设备。在至少一个实施例中,处理系统2900是电视或机顶盒设备,其具有一个或更多个处理器2902以及由一个或更多个图形处理器2908生成的图形界面。

[0283] 在至少一个实施例中,一个或更多个处理器2902每个包括一个或更多个处理器核心2907,以处理指令,该指令在被执行时执行针对系统和用户软件的操作。在至少一个实施

例中，一个或更多个处理器核心2907中的每一个被配置为处理特定指令集2909。在至少一个实施例中，指令集2909可以促进复杂指令集计算(CISC)、精简指令集计算(RISC)，或通过超长指令字(VLIW)进行计算。在至少一个实施例中，多个处理器核心2907可以各自处理不同的指令集2909，该指令集2909可以包括有助于仿真其他指令集的指令。在至少一个实施例中，处理器核心2907还可以包括其他处理设备，例如数字信号处理器(DSP)。

[0284] 在至少一个实施例中，处理器2902包括高速缓存存储器(cache)2904。在至少一个实施例中，处理器2902可以具有单个内部高速缓存或多个级别的内部高速缓存。在至少一个实施例中，高速缓存存储器在处理器2902的各个组件之间共享。在至少一个实施例中，处理器2902还使用外部高速缓存(例如，三级(L3)高速缓存或最后一级高速缓存(LLC))(未示出)，其可以使用已知的高速缓存一致性技术在处理器核心2907之间共享该逻辑。在至少一个实施例中，处理器2902中另外包括寄存器文件2906，处理器2902可以包括用于存储不同类型的数据的不同类型的寄存器(例如，整数寄存器、浮点寄存器、状态寄存器和指令指针寄存器)。在至少一个实施例中，寄存器文件2906可以包括通用寄存器或其他寄存器。

[0285] 在至少一个实施例中，一个或更多个处理器2902与一个或更多个接口总线2910耦合，以在处理器2902与系统2900中的其他组件之间传输通信信号，例如地址、数据或控制信号。在至少一个实施例中，接口总线2910在一个实施例中可以是处理器总线，例如直接媒体接口(DMI)总线的版本。在至少一个实施例中，接口总线2910不限于DMI总线，并且可以包括一个或更多个外围组件互连总线(例如，PCI，PCI Express)、存储器总线或其他类型的接口总线。在至少一个实施例中，处理器2902包括集成存储器控制器2916和平台控制器集线器2930。在至少一个实施例中，存储器控制器2916促进存储设备与处理系统2900的其他组件之间的通信，而平台控制器集线器(PCH)2930通过本地I/O总线提供到输入/输出(I/O)设备的连接。

[0286] 在至少一个实施例中，存储器设备2920可以是动态随机存取存储器(DRAM)设备、静态随机存取存储器(SRAM)设备、闪存设备、相变存储设备或具有适当的性能以用作处理器存储器。在至少一个实施例中，存储设备2920可以用作处理系统2900的系统存储器，以存储数据2922和指令2921，以在一个或更多个处理器2902执行应用或过程时使用。在至少一个实施例中，存储器控制器2916还与可选的外部图形处理器2912耦合，其可以与处理器2902中的一个或更多个图形处理器2908通信以执行图和媒体操作。在至少一个实施例中，显示设备2911可以连接至处理器2902。在至少一个实施例中，显示设备2911可以包括内部显示设备中的一个或更多个，例如在移动电子设备或便携式计算机设备或通过显示器接口(例如显示端口(DisplayPort)等)连接的外部显示设备。在至少一个实施例中，显示设备2911可以包括头戴式显示器(HMD)，诸如用于虚拟现实(VR)应用或增强现实(AR)应用中的立体显示设备。

[0287] 在至少一个实施例中，平台控制器集线器2930使外围设备能够通过高速I/O总线连接到存储设备2920和处理器2902。在至少一个实施例中，I/O外围设备包括但不限于音频控制器2946、网络控制器2934、固件接口2928、无线收发器2926、触摸传感器2925、数据存储设备2924(例如，硬盘驱动器、闪存等)。在至少一个实施例中，数据存储设备2924可以经由存储器接口(例如，SATA)或经由外围总线来连接，诸如外围组件互连总线(例如，PCI、PCIe)。在至少一个实施例中，触摸传感器2925可以包括触摸屏传感器、压力传感器或指纹

传感器。在至少一个实施例中，无线收发器2926可以是Wi-Fi收发器、蓝牙收发器或移动网络收发器，诸如3G、4G或长期演进(LTE)收发器。在至少一个实施例中，固件接口2928使能与系统固件的通信，例如，并且可以是统一的可扩展固件接口(UEFI)。在至少一个实施例中，网络控制器2934可以启用到有线网络的网络连接。在至少一个实施例中，高性能网络控制器(未示出)与接口总线2910耦合。在至少一个实施例中，音频控制器2946是多通道高清晰度音频控制器。在至少一个实施例中，处理系统2900包括可选的传统(legacy)I/O控制器2940，用于将遗留(例如，个人系统2(PS/2))设备耦合到处理系统2900。在至少一个实施例中，平台控制器集线器2930还可以连接到一个或更多个通用串行总线(USB)控制器2942，该控制器连接输入设备，诸如键盘和鼠标2943组合、相机2944或其他USB输入设备。

[0288] 在至少一个实施例中，存储器控制器2916和平台控制器集线器2930的实例可以集成到离散的外部图形处理器中，例如外部图形处理器2912。在至少一个实施例中，平台控制器集线器2930和/或存储控制器2916可以在一个或更多个处理器2902的外部。例如，在至少一个实施例中，处理系统2900可以包括外部存储控制器2916和平台控制器集线器2930，其可以配置成在与处理器2902通信的系统芯片组中的存储器控制器集线器和外围控制器集线器。

[0289] 图30示出了根据至少一个实施例的计算机系统3000。在至少一个实施例中，计算机系统3000可以是具有互连的设备和组件，SOC，或某种组合的系统。在至少一个实施例中，计算机系统3000由处理器3002形成，该处理器3002可以包括用于执行指令的执行单元。在至少一个实施例中，计算机系统3000可以包括但不限于组件，例如处理器3002，其采用包括逻辑的执行单元以执行用于过程数据的算法。在至少一个实施例中，计算机系统3000可以包括处理器，例如可从加利福尼亚圣塔克拉拉的英特尔公司(Intel Corporation of Santa Clara,California)获得的PENTIUM®处理器家族、XeonTM、Itanium®、XScaleTM和/或StrongARMTM，Intel®Core™或Intel ®Nervana™微处理器，尽管也可以使用其他系统(包括具有其他微处理器的PC、工程工作站、机顶盒等)。在至少一个实施例中，计算机系统3000可以执行可从华盛顿州雷蒙德市的微软公司(Microsoft Corporation of Redmond,Wash.)获得的WINDOWS操作系统版本，尽管其他操作系统(例如UNIX和Linux)、嵌入式软件和/或图形用户界面也可以使用。

[0290] 在至少一个实施例中，计算机系统3000可以用在其他设备中，例如手持设备和嵌入式应用。手持设备的一些示例包括蜂窝电话、互联网协议(Internet Protocol)设备、数码相机、个人数字助理(“PDA”)和手持PC。在至少一个实施例中，嵌入式应用可以包括微控制器、数字信号处理器(“DSP”)、SoC、网络计算机(“NetPC”)、机顶盒、网络集线器、广域网(“WAN”)交换机，或根据至少一个实施例可以执行一个或更多个指令的任何其他系统。

[0291] 在至少一个实施例中，计算机系统3000可包括但不限于处理器3002，该处理器3002可包括但不限于一个或更多个执行单元3008，其可以配置为执行计算统一设备架构(“CUDA”)(CUDA®由加利福尼亚州圣克拉拉的NVIDIA Corporation开发)程序。在至少一个实施例中，CUDA程序是用CUDA编程语言编写的软件应用程序的至少一部分。在至少一个实施例中，计算机系统3000是单处理器台式机或服务器系统。在至少一个实施例中，计算机系统3000可以是多处理器系统。在至少一个实施例中，处理器3002可以包括但不限于CISC微处理器、RISC微处理器、VLIW微处理器、实现指令集组合的处理器，或任何其他处理

器设备，例如，诸如数字信号处理器。在至少一个实施例中，处理器3002可以耦合到处理器总线3010，该处理器总线3010可以在处理器3002与计算机系统3000中的其他组件之间传输数据信号。

[0292] 在至少一个实施例中，处理器3002可以包括但不限于1级(“L1”)内部高速缓存存储器(“cache”)3004。在至少一个实施例中，处理器3002可以具有单个内部高速缓存或多级内部缓存。在至少一个实施例中，高速缓存存储器可以驻留在处理器3002的外部。在至少一个实施例中，处理器3002可以包括内部和外部高速缓存的组合。在至少一个实施例中，寄存器文件3006可以在各种寄存器中存储不同类型的数据，包括但不限于整数寄存器、浮点寄存器、状态寄存器和指令指针寄存器。

[0293] 在至少一个实施例中，包括但不限于执行整数和浮点运算的逻辑的执行单元3008，其也位于处理器3002中。处理器3002还可以包括微码(“ucode”)只读存储器(“ROM”)，用于存储某些宏指令的微代码。在至少一个实施例中，执行单元3008可以包括用于处理封装指令集3009的逻辑。在至少一个实施例中，通过将封装指令集3009包括在通用处理器3002的指令集中，以及要执行指令的相关电路，可以使用通用处理器3002中的封装数据来执行许多多媒体应用程序使用的操作。在至少一个实施例中，可以通过使用处理器的数据总线的全宽度来在封装的数据上执行操作来加速和更有效地执行许多多媒体应用程序，这可能不需要在处理器的数据总线上传输较小的数据单元来一次对一个数据元素执行一个或更多个操作。

[0294] 在至少一个实施例中，执行单元3008也可以用在微控制器、嵌入式处理器、图形设备、DSP和其他类型的逻辑电路中。在至少一个实施例中，计算机系统3000可以包括但不限于存储器3020。在至少一个实施例中，存储器3020可以被实现为DRAM设备、SRAM设备、闪存设备或其他存储设备。存储器3020可以存储由处理器3002可以执行的由数据信号表示的指令3019和/或数据3021。

[0295] 在至少一个实施例中，系统逻辑芯片可以耦合到处理器总线3010和存储器3020。在至少一个实施例中，系统逻辑芯片可以包括但不限于存储器控制器集线器(“MCH”)3016，并且处理器3002可以经由处理器总线3010与MCH 3016通信。在至少一个实施例中，MCH 3016可以提供到存储器3020的高带宽存储器路径3018以用于指令和数据存储以及用于图形命令、数据和纹理的存储。在至少一个实施例中，MCH 3016可以在处理器3002、存储器3020和计算机系统3000中的其他组件之间启动数据信号，并且在处理器总线3010、存储器3020和系统I/O 3022之间桥接数据信号。在至少一个实施例中，系统逻辑芯片可以提供用于耦合到图形控制器的图形端口。在至少一个实施例中，MCH 3016可以通过高带宽存储器路径3018耦合到存储器3020，并且图形/视频卡3012可以通过加速图形端口(Accelerated Graphics Port)(“AGP”)互连3014耦合到MCH 3016。

[0296] 在至少一个实施例中，计算机系统3000可以使用系统I/O 3022作为专有集线器接口总线来将MCH 3016耦合到I/O控制器集线器(“ICH”)3030。在至少一个实施例中，ICH 3030可以通过本地I/O总线提供与某些I/O设备的直接连接。在至少一个实施例中，本地I/O总线可以包括但不限于用于将外围设备连接到存储器3020、芯片组和处理器3002的高速I/O总线。示例可以包括但不限于音频控制器3029、固件集线器(“Flash BIOS”)3028、无线收发器3026、数据存储3024、包含用户输入3025的传统I/O控制器3023和键盘接口、串行扩展

端口3077(例如USB)和网络控制器3034。数据存储3024可以包括硬盘驱动器、软盘驱动器、CD-ROM设备、闪存设备或其他大容量存储设备。

[0297] 在至少一个实施例中,图30示出了包括互连的硬件设备或“芯片”的系统。在至少一个实施例中,图30可以示出示例性SoC。在至少一个实施例中,图30中示出的设备可以与专有互连、标准化互连(例如,PCIe)或其某种组合互连。在至少一个实施例中,系统3000的一个或更多个组件使用计算快速链路(CXL)互连来互连。

[0298] 图31示出了根据至少一个实施例的系统3100。在至少一个实施例中,系统3100是利用处理器3110的电子设备。在至少一个实施例中,系统3100可以是,例如但不限于,笔记本电脑、塔式服务器、机架服务器、刀片服务器、膝上型计算机、台式机、平板电脑、移动设备、电话、嵌入式计算机或任何其他合适的电子设备。

[0299] 在至少一个实施例中,系统3100可以包括但不限于通信地耦合到任何合适数量或种类的组件、外围设备、模块或设备的处理器3110。在至少一个实施例中,处理器3110使用总线或接口耦合,诸如I^2C总线、系统管理总线(“SMBus”)、低引脚数(LPC)总线、串行外围接口(“SPI”)、高清音频(“HDA”)总线、串行高级技术附件(“SATA”)总线、USB(1、2、3版)或通用异步接收器/发送器(“UART”)总线。在至少一个实施例中,图31示出了系统,该系统包括互连的硬件设备或“芯片”。在至少一个实施例中,图31可以示出示例性SoC。在至少一个实施例中,图31中所示的设备可以与专有互连线、标准化互连(例如,PCIe)或其某种组合互连。在至少一个实施例中,图31的一个或更多个组件使用计算快速链路(CXL)互连线来互连。

[0300] 在至少一个实施例中,图31可以包括显示器3124、触摸屏3125、触摸板3130、近场通信单元(“NFC”)3145、传感器集线器3140、热传感器3146、快速芯片组(“EC”)3135、可信平台模块(“TPM”)3138、BIOS/固件/闪存(“BIOS,FW Flash”)3122、DSP 3160、固态磁盘(“SSD”)或硬盘驱动器(“HDD”)3120、无线局域网单元(“WLAN”)3150、蓝牙单元3152、无线广域网单元(“WWAN”)3156、全球定位系统(GPS)3155、相机(“USB 3.0相机”)3154(例如USB 3.0相机)或以例如LPDDR3标准实现的低功耗双倍数据速率(“LPDDR”)存储器单元(“LPDDR3”)3115。这些组件可以各自以任何合适的方式实现。

[0301] 在至少一个实施例中,其他组件可以通过以上讨论的组件通信地耦合到处理器3110。在至少一个实施例中,加速度计3141、环境光传感器(“ALS”)3142、罗盘3143和陀螺仪3144可以可通信地耦合到传感器集线器3140。在至少一个实施例中,热传感器3139、风扇3137、键盘3146和触摸板3130可以通信地耦合到EC 3135。在至少一个实施例中,扬声器3163、耳机3164和麦克风(“mic”)3165可以通信地耦合到音频单元(“音频编解码器和D类放大器”)3164,其又可以通信地耦合到DSP 3160。在至少一个实施例中,音频单元3164可以包括例如但不限于音频编码器/解码器(“编解码器”)和D类放大器。在至少一个实施例中,SIM卡(“SIM”)3157可以通信地耦合到WWAN单元3156。在至少一个实施例中,组件(诸如WLAN单元3150和蓝牙单元3152以及WWAN单元3156)可以被实现为下一代形式因素(NGFF)。

[0302] 图32示出了根据至少一个实施例的示例性集成电路3200。在至少一个实施例中,示例性集成电路3200是SoC,其可使用一个或更多个IP核心制造。在至少一个实施例中,集成电路3200包括一个或更多个应用处理器3205(例如,CPU)、至少一个图形处理器3210,并且可以另外包括图像处理器3215和/或视频处理器3220,其中任意一个可能是模块化IP核心。在至少一个实施例中,集成电路3200包括外围或总线逻辑,其包括USB控制器3225、UART

控制器3230、SPI/SDIO控制器3235和I^2S/I^2C控制器3240。在至少一个实施例中，集成电路3200可以包括显示设备3245耦合到高清多媒体接口(HDMI)控制器3250和移动工业处理器接口(MIPI)显示接口3255中的一个或更多个。在至少一个实施例中，存储可以由闪存子系统3260提供，包括闪存和闪存控制器。在至少一个实施例中，可以经由存储器控制器3265提供存储器接口以用于访问SDRAM或SRAM存储器设备。在至少一个实施例中，一些集成电路还包括嵌入式安全引擎3270。

[0303] 图33示出了根据至少一个实施例的计算系统3300。在至少一个实施例中，计算系统3300包括处理子系统3301，其具有经由可以包括存储器集线器3305的互连路径通信的一个或更多个处理器3302和系统存储器3304。在至少一个实施例中，存储器集线器3305可以是芯片组组件内的单独组件，也可以集成在一个或更多个处理器3302内。在至少一个实施例中，存储器集线器3305通过通信链路3306与I/O子系统3311耦合。在至少一个实施例中，I/O子系统3311包括I/O集线器3307，其可以使计算系统3300能够接收来自一个或更多个输入设备3308的输入。在至少一个实施例中，I/O集线器3307可以使能显示控制器，其包括在一个或更多个处理器3302中，用于向一个或更多个显示设备3310A提供输出。在至少一个实施例中，与I/O集线器3307耦合的一个或更多个显示设备3310A可以包括本地、内部或嵌入式显示设备。

[0304] 在至少一个实施例中，处理子系统3301包括经由总线或其他通信链路3313耦合到存储器集线器3305的一个或更多个并行处理器3312。在至少一个实施例中，通信链路3313可以是许多基于标准的通信链路技术或协议中的一种，例如但不限于PCIe，或者可以是针对供应商的通信接口或通信结构。在至少一个实施例中，一个或更多个并行处理器3312形成计算集中的并行或向量处理系统，该系统可以包括大量的处理核心和/或处理集群，例如多集成核心(MIC)处理器。在至少一个实施例中，一个或更多个并行处理器3312形成可以将像素输出到经由I/O集线器3307耦合的一个或更多个显示设备3310A之一的图形处理子系统。在至少一个实施例中，一个或更多个并行处理器3312还可以包括显示控制器和显示接口(未示出)，以使得能够直接连接到一个或更多个显示设备3310B。

[0305] 在至少一个实施例中，系统存储单元3314可以连接到I/O集线器3307，以提供用于计算系统3300的存储机制。在至少一个实施例中，I/O交换机3316可以用于提供接口机制，以实现I/O集线器3307与其他组件之间的连接，例如可以集成到平台中的网络适配器3318和/或无线网络适配器3319，以及可以通过一个或更多个附加设备3320添加的各种其他设备。在至少一个实施例中，网络适配器3318可以是以太网适配器或另一有线网络适配器。在至少一个实施例中，无线网络适配器3319可以包括Wi-Fi、蓝牙、NFC的一个或更多个或其他包括一个或更多个无线电的网络设备。

[0306] 在至少一个实施例中，计算系统3300可以包括未明确示出的其他组件，包括USB或其他端口连接、光存储驱动器、视频捕获设备和/或其变体，也可以连接到I/O集线器3307。在至少一个实施例中，对图33中的各个组件进行互连的通信路径可以使用任何合适的协议来实现，诸如基于PCI(外围组件互连)的协议(例如，PCIe)，或其他总线或点对点通信接口和/或协议(例如，NVLink高速互连或互连协议)。

[0307] 在至少一个实施例中，一个或更多个并行处理器3312包括针对图形和视频处理而优化的电路(例如，包括视频输出电路)，并构成图形处理单元(GPU)。在至少一个实施例中，

一个或更多个并行处理器3312包括针对通用处理而优化的电路。在至少一个实施例中，计算系统3300的组件可以与单个集成电路上的一个或更多个其他系统元件集成。在至少一个实施例中，一个或更多个并行处理器3312、存储器集线器3305、处理器3302和I/O集线器3307可以被集成到片上系统(SoC)集成电路中。在至少一个实施例中，计算系统3300的组件可以被集成到单个封装中以形成系统级封装(SIP)配置。在至少一个实施例中，计算系统3300的组件的至少一部分可以被集成到多芯片模块(MCM)中，该多芯片模块可以与其他多芯片模块互连到模块化计算系统中。在至少一个实施例中，从计算系统3300中省略了I/O子系统3311和显示设备3310B。

[0308] 处理系统

[0309] 以下各图阐述了但不限于可用于实现至少一个实施例的示例性处理系统。

[0310] 图34示出了根据至少一个实施例的加速处理单元(“APU”)3400。在至少一个实施例中，APU 3400由加利福尼亚州圣克拉拉市的AMD公司开发。在至少一个实施例中，APU 3400可以被配置为执行应用程序，诸如CUDA程序。在至少一个实施例中，APU 3400包括但不限于核心复合体3410、图形复合体3440、结构3460、I/O接口3470、存储器控制器3480、显示控制器3492和多媒体引擎3494。在至少一个实施例中，APU 3400可以包括但不限于任意数量的核心复合体3410、任意数量的图形复合体3440、任意数量的显示控制器3492和任意数量的多媒体引擎3494的任何组合。为了说明的目的，在本文中用附图标记表示相似对象的多个实例，其中附图标记标识该对象，并且括号中的数字标识所需要的实例。

[0311] 在至少一个实施例中，核心复合体3410是CPU，图形复合体3440是GPU，并且APU 3400是将不限于3410和3440集成到单个芯片上的处理单元。在至少一个实施例中，一些任务可以被分配给核心复合体3410，而其他任务可以被分配给图形复合体3440。在至少一个实施例中，核心复合体3410被配置为执行与APU 3400相关联的主控制软件，例如操作系统。在至少一个实施例中，核心复合体3410是APU 3400的主处理器，其控制和协调其他处理器的操作。在至少一个实施例中，核心复合体3410发出控制图形复合体3440的操作的命令。在至少一个实施例中，核心复合体3410可以被配置为执行从CUDA源代码派生的主机可执行代码，并且图形复合体3440可以被配置为执行从CUDA源代码派生的设备可执行代码。

[0312] 在至少一个实施例中，核心复合体3410包括但不限于核心3420(1)-3420(4)和L3高速缓存3430。在至少一个实施例中，核心复合体3410可以包括但不限于任意数量的核心3420以及任意数量和类型的高速缓存的任何组合。在至少一个实施例中，核心3420被配置为执行特定指令集架构(“ISA”)的指令。在至少一个实施例中，每个核心3420是CPU核心。

[0313] 在至少一个实施例中，每个核心3420包括但不限于获取/解码单元3422，整数执行引擎3424，浮点执行引擎3426和L2高速缓存3428。在至少一个实施例中，获取/解码单元3422获取指令，对这些指令进行解码，生成微操作，并将单独的微指令分派给整数执行引擎3424和浮点执行引擎3426。在至少一个实施例中，获取/解码单元3422可以同时分派一个微指令到整数执行引擎3424和另一微指令到浮点执行引擎3426。在至少一个实施例中，整数执行引擎3424执行不限于整数和存储器操作。在至少一个实施例中，浮点引擎3426执行不限于浮点和向量运算。在至少一个实施例中，获取-解码单元3422将微指令分派给单个执行引擎，该执行引擎代替整数执行引擎3424和浮点执行引擎3426两者。

[0314] 在至少一个实施例中，每个核心3420(i)可以访问包括在核心3420(i)中的L2高速

缓存3428(i),其中i是表示核心3420的特定实例的整数。在至少一个实施例中,包括在核心复合体3410(j)中的每个核心3420经由包括在核心复合体3410(j)中的L3高速缓存3430(j)连接到包括在核心复合体3410(j)中的其他核心3420,其中j是表示核心复合体3410的特定实例的整数。在至少一个实施例中,包括在核心复合体3410(j)中的核心3420可以访问包括在核心复合体3410(j)中的所有L3高速缓存3430(j),其中j是表示核心复合体3410的特定实例的整数。在至少一个实施例中,L3高速缓存3430可以包括但不限于任意数量的切片(slice)。

[0315] 在至少一个实施例中,图形复合体3440可以被配置为以高度并行的方式执行计算操作。在至少一个实施例中,图形复合体3440被配置为执行图形管线操作,诸如绘制命令、像素操作、几何计算以及与将图像渲染至显示器相关联的其他操作。在至少一个实施例中,图形复合体3440被配置为执行与图形无关的操作。在至少一个实施例中,图形复合体3440被配置为执行与图形有关的操作和与图形无关的操作。

[0316] 在至少一个实施例中,图形复合体3440包括但不限于任意数量的计算单元3450和L2高速缓存3442。在至少一个实施例中,计算单元3450共享L2高速缓存3442。在至少一个实施例中,L2高速缓存3442被分区。在至少一个实施例中,图形复合体3440包括但不限于任意数量的计算单元3450以及任意数量(包括零)和类型的高速缓存。在至少一个实施例中,图形复合体3440包括但不限于任意数量的专用图形硬件。

[0317] 在至少一个实施例中,每个计算单元3450包括但不限于任意数量的SIMD单元3452和共享存储器3454。在至少一个实施例中,每个SIMD单元3452实现SIMD架构并且被配置为并行执行操作。在至少一个实施例中,每个计算单元3450可以执行任意数量的线程块,但是每个线程块在单个计算单元3450上执行。在至少一个实施例中,线程块包括但不限于任意数量的执行线程。在至少一个实施例中,工作组是线程块。在至少一个实施例中,每个SIMD单元3452执行不同的线程束(warp)。在至少一个实施例中,线程束是一组线程(例如16个线程),其中线程束中的每个线程属于单个线程块,并且被配置为基于单个指令集来处理不同的数据集。在至少一个实施例中,可以使用预测(predication)来禁用线程束中的一个或更多个线程。在至少一个实施例中,通道是线程。在至少一个实施例中,工作项是线程。在至少一个实施例中,波前是线程束。在至少一个实施例中,线程块中的不同波前可一起同步并经由共享存储器3454进行通信。

[0318] 在至少一个实施例中,结构3460是系统互连,其促进跨核心复合体3410、图形复合体3440、I/O接口3470、存储器控制器3480、显示控制器3492和多媒体引擎3494的数据和控制传输。在至少一个实施例中,除了结构3460之外或代替结构3460,APU 3400还可以包括但不限于任意数量和类型的系统互连,该结构3460促进跨可以在APU 3400内部或外部的任意数量和类型的直接或间接链接的组件的数据和控制传输。在至少一个实施例中,I/O接口3470表示任意数量和类型的I/O接口(例如,PCI,PCI-Extended(“PCI-X”),PCIe,千兆以太网(“GBE”),USB等)。在至少一个实施例中,各种类型的外围设备耦合到I/O接口3470。在至少一个实施例中,耦合到I/O接口3470的外围设备可以包括但不限于键盘,鼠标,打印机,扫描仪,操纵杆或其他类型的游戏控制器、媒体记录设备、外部存储设备、网络接口卡等。

[0319] 在至少一个实施例中,显示控制器AMD92在一个或更多个显示设备(例如液晶显示器(LCD)设备)上显示图像。在至少一个实施例中,多媒体引擎240包括但不限于任意数量和

类型的与多媒体相关的电路，例如视频解码器、视频编码器、图像信号处理器等。在至少一个实施例中，存储器控制器3480促进APU 3400与统一系统存储器3490之间的数据传输。在至少一个实施例中，核心复合体3410和图形复合体3440共享统一系统存储器3490。

[0320] 在至少一个实施例中，APU 3400实现种存储器子系统，其包括但不限于任意数量和类型的存储器控制器3480和可以专用于一个组件或在多个组件之间共享的存储器设备(例如，共享存储器3454)。组件。在至少一个实施例中，APU 3400实现高速缓存子系统，其包括但不限于一个或更多个高速缓存存储器(例如，L2高速缓存2728，L3高速缓存3430和L2高速缓存3442)，每个高速缓存存储器可以是组件私有的或在任意数量的组件(例如，核心3420，核心复合体3410，SIMD单元3452，计算单元3450和图形复合体3440)之间共享。

[0321] 图35示出了根据至少一个实施例的CPU 3500。在至少一个实施例中，CPU 3500由加利福尼亚州圣克拉拉市的AMD公司开发。在至少一个实施例中，CPU 3500可以被配置为执行应用程序。在至少一个实施例中，CPU 3500被配置为执行主控制软件，例如操作系统。在至少一个实施例中，CPU 3500发出控制外部GPU(未示出)的操作的命令。在至少一个实施例中，CPU 3500可以被配置为执行从CUDA源代码派生的主机可执行代码，并且外部GPU可以被配置为执行从这种CUDA源代码派生的设备可执行代码。在至少一个实施例中，CPU 3500包括但不限于任意数量的核心复合体3510，结构3560，I/O接口3570和存储器控制器3580。

[0322] 在至少一个实施例中，核心复合体3510包括但不限于核心3520(1)-3520(4)和L3高速缓存3530。在至少一个实施例中，核心复合体3510可以包括但不限于任意数量的核心3520以及任意数量和类型的高速缓存的任何组合。在至少一个实施例中，核心3520被配置为执行特定ISA的指令。在至少一个实施例中，每个核心3520是CPU核心。

[0323] 在至少一个实施例中，每个核心3520包括但不限于获取/解码单元3522，整数执行引擎3524，浮点执行引擎3526和L2高速缓存3528。在至少一个实施例中，获取/解码单元3522获取指令，对这些指令进行解码，生成微操作，并将单独的微指令分派给整数执行引擎3524和浮点执行引擎3526。在至少一个实施例中，获取/解码单元3522可以同时分派一个微指令至整数执行引擎3524和另一微指令至浮点执行引擎3526。在至少一个实施例中，整数执行引擎3524执行不限于整数和存储器操作。在至少一个实施例中，浮点引擎3526执行不限于浮点和向量运算。在至少一个实施例中，获取-解码单元3522将微指令分派给单个执行引擎，该引擎代替整数执行引擎3524和浮点执行引擎3526两者。

[0324] 在至少一个实施例中，每个核心3520(i)可以访问包括在核心3520(i)中的L2高速缓存3528(i)，其中i是表示核心3520的特定实例的整数。在至少一个实施例中，包括在核心复合体3510(j)中的每个核心3520经由包括在核心复合体3510(j)中的L3高速缓存3530(j)连接到核心复合体3510(j)中的其他核心3520，其中j是表示核心复合体3510的特定实例的整数。在至少一个实施例中，包括在核心复合体3510(j)中的核心3520可以访问包括在核心复合体3510(j)中的所有L3高速缓存3530(j)，其中j是表示核心复合体3510的特定实例的整数。在至少一个实施例中，L3高速缓存3530可以包括但不限于任意数量的切片。

[0325] 在至少一个实施例中，结构3560是系统互连，其促进跨核心复合体3510(1)-3510(N)(其中N是大于零的整数)、I/O接口3570和存储器控制器3580的数据和控制传输。在至少一个实施例中，除了结构3560之外或代替结构3560，CPU 3500还可以包括但不限于任意数量和类型的系统互连，该结构3560促进跨可以在CPU 3500内部或外部的任意数量和类型的

直接或间接链接的组件的数据和控制传输。在至少一个实施例中，I/O接口3570表示任意数量和类型的I/O接口(例如PCI，PCI-X，PCIe，GBE，USB等)。在至少一个实施例中，各种类型的外围设备耦合到I/O接口3570。在至少一个实施例中，耦合到I/O接口3570的外围设备可以包括但不限于显示器，键盘，鼠标，打印机，扫描仪，操纵杆或其他类型的游戏控制器、媒体记录设备、外部存储设备、网络接口卡等。

[0326] 在至少一个实施例中，存储器控制器3580促进CPU 3500与系统存储器3590之间的数据传输。在至少一个实施例中，核心复合体3510和图形复合体3540共享系统存储器3590。在至少一个实施例中，CPU 3500实现存储器子系统，其包括但不限于任意数量和类型的存储器控制器3580和可以专用于一个组件或在多个组件之间共享的存储器设备。在至少一个实施例中，CPU 3500实现了高速缓存子系统，其包括但不限于一个或更多个高速缓存存储器(例如，L2高速缓存3528和L3高速缓存3530)，每个高速缓存存储器可以是组件私有的或在任意数量的组件(例如，核心3520和核心复合体3510)之间共享。

[0327] 图36示出了根据至少一个实施例的示例性加速器集成切片3690。如本文所使用的，“切片”包括加速器集成电路的处理资源的指定部分。在至少一个实施例中，加速器集成电路代表多个图形加速模块种的多个图形处理引擎提供高速缓存管理、存储器访问、环境管理和中断管理服务。图形处理引擎可以各自包括单独的GPU。可选地，图形处理引擎可包括GPU内的不同类型的图形处理引擎，例如图形执行单元、媒体处理引擎(例如，视频编码器/解码器)、采样器和blit引擎。在至少一个实施例中，图形加速模块可以是具有多个图形处理引擎的GPU。在至少一个实施例中，图形处理引擎可以是集成在通用封装、线卡或芯片上的各个GPU。

[0328] 系统存储器3614内的应用程序有效地址空间3682存储进程元素3683。在一个实施例中，响应于来自处理器3607上执行的应用程序3680的GPU调用3681而存储进程元素3683。进程元素3683包含对应应用程序3680的处理状态。包含在进程元素3683中的工作描述符(WD)3684可以是应用程序请求的单个作业或可能包含指向作业队列的指针。在至少一个实施例中，WD 3684是指向应用程序有效地址空间3682中的作业请求队列的指针。

[0329] 图形加速模块3646和/或各个图形处理引擎可以由系统中的全部或部分进程共享。在至少一个实施例中，可以包括用于建立处理状态并将WD 3684发送到图形加速模块3646以在虚拟化环境中开始作业的基础设施。

[0330] 在至少一个实施例中，专用进程编程模型是针对实现的。在该模型中，单个进程拥有图形加速模块3646或个体图形处理引擎。由于图形加速模块3646由单个进程拥有，因此管理程序为拥有的分区初始化加速器集成电路，并且当分配图形加速模块3646时操作系统对加速器集成电路进行初始化以用于拥有的分区。

[0331] 在操作中，加速器集成切片3690中的WD获取单元3691获取下一个WD 3684，其中包括要由图形加速模块3646的一个或更多个图形处理引擎完成的工作的指示。来自WD 3684的数据可以存储在寄存器3645被存储器管理单元(MMU)3639、中断管理电路3647和/或环境管理电路3648使用，如图所示。例如，MMU 3339的实施例包括用于访问OS虚拟地址空间3685内的段/页表3686的段/页面漫游电路。中断管理电路3647可以处理从图形加速模块3646接收到的中断事件(INT)3692。当执行图操作时，由图形处理引擎产生的有效地址3693由MMU 3639转换为实际地址。

[0332] 在一个实施例中，为每个图形处理引擎和/或图形加速模块3646复制相同的寄存器组3645，并且可以由系统管理程序或操作系统来初始化。这些复制的寄存器中的每一个都可以包含在加速器集成切片3690中。表1中显示了可由管理程序初始化的示例性寄存器。

[0333] 表1-管理程序初始化的寄存器

[0334]

1	切片控制寄存器
2	实地址(RA)计划的处理区域指针
3	授权掩码覆盖寄存器
4	中断向量表输入偏移
5	中断向量表入口限制
6	状态寄存器
7	逻辑分区ID
8	实地址(RA)管理程序加速器利用率记录指针
9	存储描述寄存器

[0335] 表2中示出了可以由操作系统初始化的示例性寄存器。

[0336] 表2-操作系统初始化寄存器

[0337]

1	进程和线程识别
2	有效地址(EA)环境保存/还原指针
3	虚拟地址(VA)加速器利用率记录指针
4	虚拟地址(VA)存储分段表指针
5	权威面具
6	工作描述符

[0338] 在一个实施例中，每个WD 3684特定于特定的图形加速模块3646和/或特定图形处理引擎。它包含图形处理引擎进行工作或工作所需的所有信息，或者它可以是指向存储器位置的指针，其中应用程序建立了要完成的工作的命令队列。

[0339] 图37A-37B示出了根据本文至少一个实施例的示例性图形处理器。在至少一个实施例中，任何示例性图形处理器可以使用一个或更多个IP核心来制造。除了图示之外，在至少一个实施例中可以包括其他逻辑和电路，包括附加的图形处理器/核心、外围接口控制器或通用处理器核心。在至少一个实施例中，示例性图形处理器用于SoC内。

[0340] 图37A示出了根据至少一个实施例的SoC集成电路的示例性图形处理器3710，其可以使用一个或更多个IP核心来制造。图37B示出了根据至少一个实施例的SoC集成电路的的附加示例性图形处理器3740，其可以使用一个或更多个IP核心来制造。在至少一个实施例中，图37A的图形处理器3710是低功耗图形处理器核心。在至少一个实施例中，图37B的图形处理器3740是更高性能的图形处理器核心。在至少一个实施例中，每个图形处理器3710、3740可以是图5的图形处理器510的变体。

[0341] 在至少一个实施例中，图形处理器3710包括顶点处理器3705和一个或更多个片段处理器3715A-3715N(例如3715A、3715B、3715C、3715D至3715N-1和3715N)。在至少一个实施例中，图形处理器3710可以经由单独的逻辑来执行不同的着色器程序，使得顶点处理器3705被优化以执行针对顶点着色器程序的操作，而一个或更多个片段处理器3715A-3715N执行片段(例如，像素)着色操作用于片段或像素或着色器程序。在至少一个实施例中，顶点

处理器3705执行3D图形管线的顶点处理阶段并生成图元和顶点数据。在至少一个实施例中，片段处理器3715A-3715N使用由顶点处理器3705生成的图元和顶点数据来生成在显示设备上显示的帧缓冲区。在至少一个实施例中，片段处理器3715A-3715N被优化以执行如在OpenGL API中所提供的片段着色器程序，其可以用于执行与在Direct 3D API中所提供的像素着色器程序类似的操作。

[0342] 在至少一个实施例中，图形处理器3710附加地包括一个或更多个MMU 3720A-3720B、高速缓存3725A-3725B和电路互连3730A-3730B。在至少一个实施例中，一个或更多个MMU 3720A-3720B提供用于图形处理器3710的虚拟到物理地址的映射，包括用于顶点处理器3705和/或片段处理器3715A-3715N，其可以引用存储在存储器中的顶点或图像/纹理数据，除了存储在一个或更多个高速缓存3725A-3725B中的顶点或图像/纹理数据之外。在至少一个实施例中，一个或更多个MMU 3720A-3720B可以与系统内的其他MMU同步，包括与图32的一个或更多个应用处理器3205、图像处理器3215和/或视频处理器3220相关联的一个或更多个MMU，使得每个处理器3205-3220可以参与共享或统一的虚拟存储器系统。在至少一个实施例中，一个或更多个电路互连3730A-3730B使图形处理器3710能够经由SoC的内部总线或经由直接连接与SoC内的其他IP核心相连接。

[0343] 在至少一个实施例中，图形处理器3740包括图37A的图形处理器3710的一个或更多个MMU 3720A-3720B、高速缓存3725A-3725B和电路互连3730A-3730B。在至少一个实施例中，图形处理器3740包括一个或更多个着色器核心3755A-3755N(例如，3755A、3755B、3755C、3755D、3755E、3755F、至3755N-1和3755N)，其提供了统一的着色器核心架构，其中单个核心或类型或核心可以执行所有类型的可编程着色器代码，包括用于实现顶点着色器、片段着色器和/或计算着色器的着色器程序代码。在至少一个实施例中，多个着色器核心可以变化。在至少一个实施例中，图形处理器3740包括核心间任务管理器3745，其充当线程分派器以将执行线程分派给一个或更多个着色器核心3755A-3755N和分块单元3758，以加速基于图块渲染的分块操作，其中在图像空间中细分了场景的渲染操作，例如，以利用场景内的局部空间一致性或优化内部缓存的使用。

[0344] 图38A示出了根据至少一个实施例的图形核心3800。在至少一个实施例中，图形核心3800可以包括在图32的图形处理器3210内。在至少一个实施例中，图形核心3800可以是图37B中统一的着色器核心3755A-3755N。在至少一个实施例中，图形核心3800包括共享指令高速缓存3802、纹理单元3818和高速缓存/共享存储器3820，它们是图形核心3800内的执行资源所共有的。在至少一个实施例中，图形核心3800可以包括多个切片(slice)3801A-3801N或每个核心的分区，图形处理器可以包括图形核心3800的多个实例。切片3801A-3801N可以包括支持逻辑，该支持逻辑包括本地指令高速缓存3804A-3804N、线程调度器3806A-3806N、线程分派器3808A-3808N和一组寄存器3810A-3810N。在至少一个实施例中，切片3801A-3801N可以包括一组附加功能单元(AFU)3812A-3812N、浮点单元(FPU)3814A-3814N、整数算术逻辑单元(ALU)3816A-3816N、地址计算单元(ACU)3813A-3813N、双精度浮点单元(DPFPU)3815A-3815N和矩阵处理单元(MPU)3817A-3817N。

[0345] 在一个实施例中，FPU 3814A-3814N可以执行单精度(32位)和半精度(16位)浮点运算，而DPFPU 3815A-3815N可以执行双精度(64位)浮点运算点操作。在至少一个实施例中，ALU 3816A-3816N可以以8位、16位和32位精度执行可变精度整数运算，并且可以被配置

用于混合精度运算。在至少一个实施例中,MPU 3817A-3817N还可被配置用于混合精度矩阵运算,包括半精度浮点运算和8位整数运算。在至少一个实施例中,MPU 3817A-3817N可以执行各种矩阵操作以加速CUDA程序,包括使得能够支持加速的通用矩阵到矩阵乘法(GEMM)。在至少一个实施例中,AFU 3812A-3812N可以执行浮点数或整数单元不支持的附加逻辑运算,包括三角运算(例如,Sine、Cosine等)。

[0346] 图38B示出了在至少一个实施例中的通用图形处理单元(GPGPU)3830。在至少一个实施例中,GPGPU 3830是高度并行的并且适合于部署在多芯片模块上。在至少一个实施例中,GPGPU 3830可以被配置为使得高度并行的计算操作能够由GPU阵列来执行。在至少一个实施例中,GPGPU 3830可以直接链路到GPGPU 3830的其他实例,以创建多GPU集群以提高用于CUDA程序的执行时间。在至少一个实施例中,GPGPU 3830包括主机接口3832以实现与主机处理器的连接。在至少一个实施例中,主机接口3832是PCIe接口。在至少一个实施例中,主机接口3832可以是厂商专用的通信接口或通信结构。在至少一个实施例中,GPGPU 3830从主机处理器接收命令,并使用全局调度器3834将与那些命令相关联的执行线程分派给一组计算集群3836A-3836H。在至少一个实施例中,计算集群3836A-3836H共享高速缓存存储器3838。在至少一个实施例中,高速缓存存储器3838可以用作计算集群3836A-3836H内的高速缓存存储器的高级高速缓存。

[0347] 在至少一个实施例中,GPGPU 3830包括经由一组存储器控制器3842A-3842B与计算集群3836A-3836H耦合的存储器3844A-3844B。在至少一个实施例中,存储器3844A-3844B可以包括各种类型的存储器设备,包括动态随机存取存储器(DRAM)或图形随机存取存储器,例如同步图形随机存取存储器(SGRAM),包括图形双倍数据速率(GDDR)存储器。

[0348] 在至少一个实施例中,计算集群3836A-3836H各自包括一组图形核心,诸如图38A的图形核心3800,其可以包括多种类型的整数和浮点逻辑单元,可以以各种精度执行计算操作,包括适合与CUDA程序相关的计算。在至少一个实施例中,每个计算集群3836A-3836H中的浮点单元的至少一个子集可以配置为执行16位或32位浮点运算,而不同的浮点单元的子集可以配置为执行64位浮点运算。

[0349] 在至少一个实施例中,GPGPU 3830的多个实例可以被配置为操作为计算集群。在至少一个实施例中,计算集群3836A-3836H可以实现用于同步和数据交换的任何技术上可行的通信技术。在至少一个实施例中,GPGPU 3830的多个实例通过主机接口3832进行通信。在至少一个实施例中,GPGPU 3830包括I/O集线器3839,其将GPGPU 3830与GPU链路3840耦合,使得能够直接连接至GPGPU 3830的其他的实例。在至少一个实施例中,GPU链路3840耦合到专用GPU到GPU桥接器,其使得能够在GPGPU 3830的多个实例之间进行通信和同步。在至少一个实施例中,GPU链路3840与高速互连耦合,以向其他GPGPU或并行处理器发送和接收数据。在至少一个实施例中,GPGPU 3830的多个实例位于单独的数据处理系统中,并经由可经由主机接口3832访问的网络设备进行通信。在至少一个实施例中,GPU链路3840可被配置为能够连接到主机处理器,附加或替代主机接口3832。在至少一个实施例中,GPGPU 3830可以配置为执行CUDA程序。

[0350] 图39A示出了根据至少一个实施例的并行处理器3900。在至少一个实施例中,并行处理器3900的各种组件可以使用一个或更多个集成电路设备来实现,例如可编程处理器、专用集成电路(ASIC)或FPGA。

[0351] 在至少一个实施例中，并行处理器3900包括并行处理单元3902。在至少一个实施例中，并行处理单元3902包括I/O单元3904，其使得能够与其他设备进行通信，包括并行处理单元3902的其他实例。在至少一个实施例中，I/O单元3904可以直接连接到其他设备。在至少一个实施例中，I/O单元3904通过使用集线器或交换机接口(例如，存储器集线器1405)与其他设备连接。在至少一个实施例中，存储器集线器1405与I/O单元3904之间的连接形成通信链路。在至少一个实施例中，I/O单元3904与主机接口3906和存储器交叉开关3916连接，其中主机接口3906接收用于执行处理操作的命令，而存储器交叉开关3916接收用于执行存储器操作的命令。

[0352] 在至少一个实施例中，当主机接口3906经由I/O单元3904接收命令缓冲区时，主机接口3906可以引导工作操作以执行那些命令到前端3908。在至少一个实施例中，前端3908与调度器3910耦合，调度器3910配置成将命令或其他工作项分配给处理阵列3912。在至少一个实施例中，调度器3910确保在将任务分配给处理阵列3912中的处理阵列3912之前，处理阵列3912被正确地配置并且处于有效状态。在至少一个实施例中，调度器3910通过在微控制器上执行的固件逻辑来实现。在至少一个实施例中，微控制器实现的调度器3910可配置成以粗粒度和细粒度执行复杂的调度和工作分配操作，从而实现对在处理阵列3912上执行的线程的快速抢占和环境切换。在至少一个实施例中，主机软件可以证明用于通过多个图形处理门铃之一在处理阵列3912上进行调度的工作负载。在至少一个实施例中，工作负载然后可以由包括调度器3910的微控制器内的调度器3910逻辑在处理阵列3912上自动分配。

[0353] 在至少一个实施例中，处理阵列3912可以包括多达“N”个处理集群(例如，集群3914A、集群3914B到集群3914N)。在至少一个实施例中，处理阵列3912的每个集群3914A-3914N可以执行大量并发线程。在至少一个实施例中，调度器3910可以使用各种调度和/或工作分配算法将工作分配给处理阵列3912的集群3914A-3914N，其可以根据每种程序或计算类型产生的工作负载而变化。在至少一个实施例中，调度可以由调度器3910动态地处理，或者可以在配置为由处理阵列3912执行的程序逻辑的编译期间部分地由编译器逻辑来辅助。在至少一个实施例中，可将处理阵列3912的不同的集群3914A-3914N分配用于处理不同类型的程序或用于执行不同类型的计算。

[0354] 在至少一个实施例中，处理阵列3912可以配置成执行各种类型的并行处理操作。在至少一个实施例中，处理阵列3912配置成执行通用并行计算操作。在至少一个实施例中，处理阵列3912可以包括执行处理任务的逻辑，该处理任务包括对视频和/或音频数据的过滤，执行建模操作，包括物理操作以及执行数据转换。

[0355] 在至少一个实施例中，处理阵列3912配置成执行并行图形处理操作。在至少一个实施例中，处理阵列3912可以包括附加逻辑以支持这种图形处理操作的执行，包括但不限于执行纹理操作的纹理采样逻辑，以及镶嵌逻辑和其他顶点处理逻辑。在至少一个实施例中，处理阵列3912可以配置成执行与图形处理有关的着色器程序，例如但不限于顶点着色器、曲面细分着色器、几何着色器和像素着色器。在至少一个实施例中，并行处理单元3902可以经由I/O单元3904从系统存储器传送数据以进行处理。在至少一个实施例中，在处理期间，可以在处理期间将传送的数据存储到片上存储器(例如，并行处理器存储器3922)，然后将其写回到系统存储器。

[0356] 在至少一个实施例中，当并行处理单元3902用于执行图处理时，调度器3910可以配置成将处理工作负载划分为近似相等大小的任务，以更好地将图形处理操作分配给处理阵列3912的多个集群3914A-3914N。在至少一个实施例中，处理阵列3912的部分可以配置成执行不同类型的处理。例如，在至少一个实施例中，第一部分可以配置成执行顶点着色和拓扑生成，第二部分可以配置成执行镶嵌和几何着色，并且第三部分可以配置成执行像素着色或其他屏幕空间操作，以生成用于显示的渲染图像。在至少一个实施例中，可以将由集群3914A-3914N中的一个或更多个产生的中间数据存储在缓冲区中，以允许在集群3914A-3914N之间传输中间数据以进行进一步处理。

[0357] 在至少一个实施例中，处理阵列3912可以经由调度器3910接收要执行的处理任务，该调度器3910从前端3908接收定义处理任务的命令。在至少一个实施例中，处理任务可以包括要被处理的数据的索引，例如可以包括表面(补丁)数据、原始数据、顶点数据和/或像素数据，以及状态参数和定义如何处理数据的命令(例如，要执行什么程序)。在至少一个实施例中，调度器3910可以配置成获取与任务相对应的索引，或者可以从前端3908接收索引。在至少一个实施例中，前端3908可以配置成确保在启动由传入命令缓冲区(例如，批缓冲区(batch-buffer)、推送缓冲区等)指定的工作负载之前，处理阵列3912配置成有效状态。

[0358] 在至少一个实施例中，并行处理单元3902的一个或更多个实例中的每一个可以与并行处理器存储器3922耦合。在至少一个实施例中，可以经由存储器交叉开关3916访问并行处理器存储器3922，所述存储器交叉开关3916可以接收来自处理阵列3912以及I/O单元3904的存储器请求。在至少一个实施例中，存储器交叉开关3916可以经由存储器接口3918访问并行处理器存储器3922。在至少一个实施例中，存储器接口3918可以包括多个分区单元(例如，分区单元3920A、分区单元3920B到分区单元3920N)，其可各自耦合至并行处理器存储器3922的一部分(例如，存储器单元)。在至少一个实施例中，多个分区单元3920A-3920N为配置为等于存储器单元的数量，使得第一分区单元3920A具有对应的第一存储器单元3924A，第二分区单元3920B具有对应的存储器单元3924B，第N分区单元3920N具有对应的第N存储器单元3924N。在至少一个实施例中，分区单元3920A-3920N的数量可以不等于存储器设备的数量。

[0359] 在至少一个实施例中，存储器单元3924A-3924N可以包括各种类型的存储器设备，包括动态随机存取存储器(DRAM)或图形随机存取存储器，例如同步图形随机存取存储器(SGRAM)，包括图形双倍数据速率(GDDR)存储器。在至少一个实施例中，存储器单元3924A-3924N还可包括3D堆叠存储器，包括但不限于高带宽存储器(HBM)。在至少一个实施例中，可以跨存储器单元3924A-3924N来存储诸如帧缓冲区或纹理映射的渲染目标，从而允许分区单元3920A-3920N并行地写入每个渲染目标的部分，以有效地使用并行处理器存储器3922的可用带宽。在至少一个实施例中，可以排除并行处理器存储器3922的本地实例，以有利于利用系统存储器与本地高速缓存存储器结合的统一存储器设计。

[0360] 在至少一个实施例中，处理阵列3912的集群3914A-3914N中的任何一个都可以处理将被写入并行处理器存储器3922内的任何存储器单元3924A-3924N中的数据。在至少一个实施例中，存储器交叉开关3916可以配置为将每个集群3914A-3914N的输出传输到任何分区单元3920A-3920N或另一个集群3914A-3914N，集群3914A-3914N可以对输出执行其他

处理操作。在至少一个实施例中，每个集群3914A-3914N可以通过存储器交叉开关3916与存储器接口3918通信，以从各种外部存储设备读取或写入各种外部存储设备。在至少一个实施例中，存储器交叉开关3916具有到存储器接口3918的连接以与I/O单元3904通信，以及到并行处理器存储器3922的本地实例的连接，从而使不同处理集群3914A-3914N内的处理单元与系统存储器或不是并行处理单元3902本地的其他存储器进行通信。在至少一个实施例中，存储器交叉开关3916可以使用虚拟通道来分离集群3914A-3914N和分区单元3920A-3920N之间的业务流。

[0361] 在至少一个实施例中，可以在单个插入卡上提供并行处理单元3902的多个实例，或者可以将多个插入卡互连。在至少一个实施例中，并行处理单元3902的不同实例可以配置成相互操作，即使不同实例具有不同数量的处理核心，不同数量的本地并行处理器存储器和/或其他配置差异。例如，在至少一个实施例中，并行处理单元3902的一些实例可以包括相对于其他实例而言更高精度的浮点单元。在至少一个实施例中，结合并行处理单元3902或并行处理器3900的一个或更多个实例的系统可以以各种配置和形式因素来实现，包括但不限于台式机、膝上型计算机或手持式个人计算机、服务器、工作站、游戏机和/或嵌入式系统。

[0362] 图39B示出了根据至少一个实施例的处理集群3994。在至少一个实施例中，处理集群3994被包括在并行处理单元内。在至少一个实施例中，处理集群3994是图39A的处理集群3914A-3914N之一的实例。在至少一个实施例中，处理集群3994可以配置成并行执行许多线程，其中术语“线程”是指在特定的一组输入数据上执行的特定程序的实例。在至少一个实施例中，单指令多数据(SIMD)指令发布技术用于支持大量线程的并行执行而无需提供多个独立的指令单元。在至少一个实施例中，使用单指令多线程(SIMT)技术来支持并行执行大量一般同步的线程，这使用了公共指令单元，该公共指令单元配置成向每个处理集群3994内的一组处理引擎发出指令。

[0363] 在至少一个实施例中，可以通过将处理任务分配给SIMT并行处理器的管线管理器3932来控制处理集群3994的操作。在至少一个实施例中，管线管理器3932从图39A的调度器3910接收指令，通过图形多处理器3934和/或纹理单元3936管理这些指令的执行。在至少一个实施例中，图形多处理器3934是SIMT并行处理器的示例性实例。然而，在至少一个实施例中，处理集群3994内可以包括不同架构的各种类型的SIMT并行处理器。在至少一个实施例中，在处理集群3994内可以包括图形多处理器3934的一个或更多个实例。在至少一个实施例中，图形多处理器3934可以处理数据，并且数据交叉开关3940可以用于将处理后的数据分发到多个可能的目的(包括其他着色器单元)地之一。在至少一个实施例中，管线管理器3932可以通过指定要经由数据交叉开关3940分配的处理后的数据的目的地来促进处理后的数据的分配。

[0364] 在至少一个实施例中，处理集群3994内的每个图形多处理器3934可以包括相同的一组功能执行逻辑(例如，算术逻辑单元、加载存储单元(LSU)等)。在至少一个实施例中，可以以管线方式配置功能执行逻辑，其中可以在先前的指令完成之前发出新的指令。在至少一个实施例中，功能执行逻辑支持多种运算，包括整数和浮点算术、比较操作、布尔运算、移位和各种代数函数的计算。在至少一个实施例中，可以利用相同的功能单元硬件来执行不同的操作，并且可以存在功能单元的任何组合。

[0365] 在至少一个实施例中，传送到处理集群3994的指令构成线程。在至少一个实施例中，跨一组并行处理引擎执行的一组线程是线程组。在至少一个实施例中，线程组在不同的输入数据上执行程序。在至少一个实施例中，线程组内的每个线程可被分配给图形多处理器3934内的不同处理引擎。在至少一个实施例中，线程组可包括比图形多处理器3934内的多个处理引擎更少的线程。在至少一个实施例中，当线程组包括的线程数少于处理引擎的数量时，一个或更多个处理引擎在正在处理该线程组的循环期间可能是空闲的。在至少一个实施例中，线程组还可以包括比图形多处理器3934内的多个处理引擎更多的线程。在至少一个实施例中，当线程组包括比图形多处理器3934内的处理引擎的数量更多的线程时，可以在连续的时钟周期内执行处理。在至少一个实施例中，可以在图形多处理器3934上同时执行多个线程组。

[0366] 在至少一个实施例中，图形多处理器3934包括内部高速缓存存储器，以执行加载和存储操作。在至少一个实施例中，图形多处理器3934可以放弃内部高速缓存并使用处理集群3994内的高速缓存存储器(例如，L1高速缓存3948)。在至少一个实施例中，每个图形多处理器3934还可以访问分区单元(例如，图39A的分区单元3920A-3920N)内的L2高速缓存，这些分区单元在所有处理集群3994之间共享并且可以用于在线程之间传输数据。在至少一个实施例中，图形多处理器3934还可以访问片外全局存储器，其可以包括本地并行处理器存储器和/或系统存储器中的一个或更多个。在至少一个实施例中，并行处理单元3902外部的任何存储器都可以用作全局存储器。在至少一个实施例中，处理集群3994包括图形多处理器3934的多个实例，它们可以共享可以存储在L1高速缓存3948中的公共指令和数据。

[0367] 在至少一个实施例中，每个处理集群3994可以包括配置成将虚拟地址映射为物理地址的MMU 3945。在至少一个实施例中，MMU 3945的一个或更多个实例可以驻留在图39A的存储器接口3918内。在至少一个实施例中，MMU 3945包括一组页表条目(PTE)，其用于将虚拟地址映射到图块(谈论有关图块的更多信息)的物理地址以及可选地映射到高速缓存行索引。在至少一个实施例中，MMU 3945可以包括地址转换后备缓冲区(TLB)或可以驻留在图形多处理器3934或L1高速缓存3948或处理集群3994内的高速缓存。在至少一个实施例中，处理物理地址以分配表面数据访问局部性，以便在分区单元之间进行有效的请求交织。在至少一个实施例中，高速缓存行索引可以用于确定对高速缓存线的请求是命中还是未命中。

[0368] 在至少一个实施例中，可以配置处理集群3994，使得每个图形多处理器3934耦合到纹理单元3936，以执行纹理映射操作，例如，可以涉及确定纹理样本位置、读取纹理数据以及过滤纹理数据。在至少一个实施例中，根据需要从内部纹理L1高速缓存(未示出)或从图形多处理器3934内的L1高速缓存中读取纹理数据，并从L2高速缓存、本地并行处理器存储器或系统存储器中获取纹理数据。在至少一个实施例中，每个图形多处理器3934将处理后的任务输出到数据交叉开关3940，以将处理后的任务提供给另一处理集群3994以进行进一步处理或将处理后的任务存储在L2高速缓存、本地并行处理器存储器、或经由存储器交叉开关3916的系统存储器中。在至少一个实施例中，光栅前操作单元(preROP)3942配置成从图形多处理器3934接收数据，将数据引导至ROP单元，该ROP单元可以与本文所述的分区单元(例如，图39A的分区单元3920A-3920N)一起定位。在至少一个实施例中，PreROP 3942单元可以执行用于颜色混合的优化、组织像素颜色数据以及执行地址转换。

[0369] 图39C示出了根据至少一个实施例的图形多处理器3996。在至少一个实施例中，图形多处理器3996是图39B的图形多处理器3934。在至少一个实施例中，图形多处理器3996与处理集群3994的管线管理器3932耦合。在至少一个实施例中，图形多处理器3996具有执行管线，该执行管线包括但不限于指令高速缓存3952、指令单元3954、地址映射单元3956、寄存器文件3958、一个或更多个GPGPU核心3962和一个或更多个LSU 3966。GPGPU核心3962和LSU 3966与高速缓存存储器3972和共享存储器3970通过存储器和高速缓存互连3968耦合。

[0370] 在至少一个实施例中，指令高速缓存3952从管线管理器3932接收要执行的指令流。在至少一个实施例中，将指令高速缓存在指令高速缓存3952中并将其分派以供指令单元3954执行。在一个实施例中，指令单元3954可以分派指令作为线程组(例如，线程束)，将线程组的每个线程分配给GPGPU核心3962内的不同执行单元。在至少一个实施例中，指令可以通过在统一地址空间内指定地址来访问任何本地、共享或全局地址空间。在至少一个实施例中，地址映射单元3956可以用于将统一地址空间中的地址转换成可以由LSU 3966访问的不同的存储器地址。

[0371] 在至少一个实施例中，寄存器文件3958为图形多处理器3996的功能单元提供了一组寄存器。在至少一个实施例中，寄存器文件3958为连接到图形多处理器3996的功能单元(例如，GPGPU核心3962、LSU 3966)的数据路径的操作数提供了临时存储。在至少一个实施例中，在每个功能单元之间划分寄存器文件3958，使得为每个功能单元分配寄存器文件3958的专用部分。在至少一个实施例中，寄存器文件3958在图形多处理器3996正在执行的不同线程组之间划分。

[0372] 在至少一个实施例中，GPGPU核心3962可以各自包括用于执行图多处理器3996的指令的FPU和/或ALU。GPGPU核心3962在架构上可以相似或架构可能有所不同。在至少一个实施例中，GPGPU核心3962的第一部分包括单精度FPU和整数ALU，而GPGPU核心的第二部分包括双精度FPU。在至少一个实施例中，FPU可以实现用于浮点算法的IEEE 754-3908标准或启用可变精度浮点算法。在至少一个实施例中，图形多处理器3996可以另外包括一个或更多个固定功能或特殊功能单元，以执行特定功能，诸如复制矩形或像素混合操作。在至少一个实施例中，GPGPU核心3962中的一个或更多个也可以包括固定或特殊功能逻辑。

[0373] 在至少一个实施例中，GPGPU核心3962包括能够对多组数据执行单个指令的SIMD逻辑。在至少一个实施例中，GPGPU核心3962可以物理地执行SIMD4、SIMD8和SIMD16指令，并且在逻辑上执行SIMD1、SIMD2和SIMD32指令。在至少一个实施例中，用于GPGPU核心的SIMD指令可以在编译时由着色器编译器生成，或者在执行针对单程序多数据(SPMD)或SIMT架构编写和编译的程序时自动生成。在至少一个实施例中，可以通过单个SIMD指令来执行为SIMT执行模型配置的程序的多个线程。例如，在至少一个实施例中，可以通过单个SIMD8逻辑单元并行执行执行相同或相似操作的八个SIMT线程。

[0374] 在至少一个实施例中，存储器和高速缓存互连3968是将图形多处理器3996的每个功能单元连接到寄存器文件3958和共享存储器3970的互连网络。在至少一个实施例中，存储器和高速缓存互连3968是交叉开关互连，其允许LSU 3966在共享存储器3970和寄存器文件3958之间实现加载和存储操作。在至少一个实施例中，寄存器文件3958可以以与GPGPU核心3962相同的频率操作，从而在GPGPU核心3962和寄存器文件3958之间进行数据传输的延迟非常低。在至少一个实施例中，共享存储器3970可以用于启用在图形多处理器3996内的

功能单元上执行的线程之间的通信。在至少一个实施例中，例如，高速缓存存储器3972可以用作数据高速缓存，以高速缓存在功能单元和纹理单元3936之间通信的纹理数据。在至少一个实施例中，共享存储器3970也可以用作程序管理的高速缓存。在至少一个实施例中，除了存储在高速缓存存储器3972中的自动高速缓存的数据之外，在GPGPU核心3962上执行的线程还可以以编程方式将数据存储在共享存储器中。

[0375] 在至少一个实施例中，如本文所述的并行处理器或GPGPU通信地耦合到主机/处理器核心，以加速图形操作、机器学习操作、图案分析操作以及各种通用GPU(GPGPU)功能。在至少一个实施例中，GPU可以通过总线或其他互连(例如，诸如PCIe或NVLink的高速互连)通信地耦合到主机处理器/核心。在至少一个实施例中，GPU可以与核心集成在相同的封装或芯片上，并通过内部处理器总线/互连(即，封装或芯片的内部)通信地耦合到核心。在至少一个实施例中，不管GPU连接的方式如何，处理器核心可以以WD包含的命令/指令序列的形式向GPU分配工作。在至少一个实施例中，GPU然后使用专用电路/逻辑来有效地处理这些命令/指令。

[0376] 通用计算

[0377] 以下各图阐述但不限于在通用计算中用来实现至少一个实施例的示例性软件配置。

[0378] 图40示出了根据至少一个实施例的编程平台的软件栈。在至少一个实施例中，编程平台是用于利用计算系统上的硬件来加速计算任务的平台。在至少一个实施例中，软件开发人员可以通过库、编译器指令和/或对编程语言的扩展来访问编程平台。在至少一个实施例中，编程平台可以是但不限于CUDA，Radeon开放计算平台(“ROCm”)，OpenCL(由Khronos group开发的OpenCL™)，SYCL或Intel One API。

[0379] 在至少一个实施例中，编程平台的软件栈4000为应用程序4001提供执行环境。在至少一个实施例中，应用程序4001可以包括能够在软件栈4000上启动的任何计算机软件。在至少一个实施例中，应用程序4001可以包括但不限于人工智能(“AI”)/机器学习(“ML”)应用程序，高性能计算(“HPC”)应用程序，虚拟桌面基础架构(“VDI”)或数据中心工作负载。

[0380] 在至少一个实施例中，应用程序4001和软件栈4000在硬件4007上运行。在至少一个实施例中，硬件4007可以包括一个或更多个GPU，CPU，FPGA，AI引擎和/或支持编程平台的其他类型的计算设备。在至少一个实施例中，例如采用CUDA，软件栈4000可以是厂商专用的，并且仅与来自特定厂商的设备兼容。在至少一个实施例中，例如在采用OpenCL中，软件栈4000可以与来自不同供应商的设备一起使用。在至少一个实施例中，硬件4007包括连接到一个或更多个设备的主机，该设备可经由应用程序编程接口(API)调用被访问以执行计算任务。在至少一个实施例中，与硬件4007内的主机相比，其可以包括但不限于CPU(但还可以包括计算设备)及其存储器，硬件4007内的设备可以包括但不限于GPU，FPGA，AI引擎或其他计算设备(但还可以包括CPU)及其存储器。

[0381] 在至少一个实施例中，编程平台的软件栈4000包括但不限于多个库4003，运行时(runtime)4005和设备内核驱动器4006。在至少一个实施例中，库4003中的每个库可以包括可以由计算机程序使用并在软件开发期间利用的数据和编程代码。在至少一个实施例中，库4003可以包括但不限于预写的代码和子例程，类，值，类型规范，配置数据，文档，帮助数据和/或消息模板。在至少一个实施例中，库4003包括被优化用于在一种或更多种类型的设

备上执行的函数。在至少一个实施例中，库4003可以包括但不限于用于在设备上执行数学、深度学习和/或其他类型的运算的函数。在至少一个实施例中，库3803与对应的API 3802相关联，API 3802可包括一个或更多个API，其暴露在库3803中实现的函数。

[0382] 在至少一个实施例中，将应用程序4001编写为源代码，该源代码被编译成可执行代码，如下面结合图45更详细讨论的。在至少一个实施例中，应用程序4001的可执行代码可以至少部分地在由软件栈4000提供的执行环境上运行。在至少一个实施例中，在应用程序4001的执行期间，可以得到需要在设备(与主机相比)上运行的代码。在这种情况下，在至少一个实施例中，可以调用运行时4005以在设备上加载和启动必需的代码。在至少一个实施例中，运行时4005可以包括能够支持应用程序4001的执行的任何技术上可行的运行时系统。

[0383] 在至少一个实施例中，运行时4005被实现为与对应的API(其被示为API 4004)相关联的一个或更多个运行时库。在至少一个实施例中，一个或更多个这样的运行时库可以包括但不限于用于存储器管理，执行控制，设备管理，错误处理和/或同步等等的函数。在至少一个实施例中，存储器管理函数可以包括但不限于用于分配、解除分配和复制设备存储器以及在主机存储器和设备存储器之间传输数据的函数。在至少一个实施例中，执行控制函数可以包括但不限于在设备上启动函数(当函数是可从主机调用的全局函数时，有时称为“内核”)的函数，和用于在运行时库为要在设备上执行的给定函数维护的缓冲区中设置属性值的函数。

[0384] 在至少一个实施例中，可以任何技术上可行的方式来实现运行时库和相应的API 4004。在至少一个实施例中，一个(或任意数量的)API可以公开用于设备的细粒度控制的低级函数集，而另一(或任意数量的)API可以公开这样的较高级的函数集。在至少一个实施例中，可以在低级API之上构建高级运行时API。在至少一个实施例中，一个或更多个运行时API可以是在与语言无关的运行时API之上分层的特定于语言的API。

[0385] 在至少一个实施例中，设备内核驱动器4006被配置为促进与底层设备的通信。在至少一个实施例中，设备内核驱动器4006可以提供诸如API 4004之类的API和/或其他软件所依赖的低级函数。在至少一个实施例中，设备内核驱动器4006可以被配置为在运行时将中间表示(“IR”)代码编译成二进制代码。在至少一个实施例中，对于CUDA，设备内核驱动器4006可以在运行时将非硬件专用的并行线程执行(“PTX”)IR代码编译为用于特定目标设备的二进制代码(高速缓存已编译的二进制代码)，其有时也称为“最终”代码。在至少一个实施例中，这样做可以允许最终代码在目标设备上运行，而当源代码最初被编译为PTX代码时，该目标设备可能不存在。备选地，在至少一个实施例中，设备源代码可以离线地编译成二进制代码，而不需要设备内核驱动器4006在运行时编译IR代码。

[0386] 图41示出了根据至少一个实施例的图40的软件栈4000的CUDA实现。在至少一个实施例中，可在其上启动应用程序4101的CUDA软件栈4100包括CUDA库4103，CUDA运行时4105，CUDA驱动器4107和设备内核驱动器4108。在至少一个实施例中，CUDA软件栈4100在硬件4109上执行，该硬件4109可以包括支持CUDA的GPU，其由加利福尼亚州圣克拉拉市的NVIDIA公司开发。

[0387] 在至少一个实施例中，应用程序4101、CUDA运行时4105和设备内核驱动器4108可以分别执行与应用程序4001、运行时4005和设备内核驱动器4006类似的功能，以上结合图

40对其进行了描述。在至少一个实施例中,CUDA驱动器4107包括实现CUDA驱动器API 4106的库(libcuda.so)。在至少一个实施例中,类似于由CUDA运行时库(cudart)实现的CUDA运行时API 4104,CUDA驱动器API 4106可以公开但不限于用于存储器管理、执行控制、设备管理、错误处理、同步和/或图形互操作性等的函数。在至少一个实施例中,CUDA驱动器API 4106与CUDA运行时API 4104的不同之处在于,CUDA运行时API 4104通过提供隐式初始化、上下文(类似于进程)管理和模块(类似于动态加载的库)管理来简化设备代码管理。与高级CUDA运行时API 4104相反,在至少一个实施例中,CUDA驱动器API 4106是提供对设备的更细粒度控制的低级API,特别是关于上下文和模块加载。在至少一个实施例中,CUDA驱动器API 4106可以公开没有由CUDA运行时API 4104公开的用于上下文管理的函数。在至少一个实施例中,CUDA驱动器API 4106也与语言无关,并且除了支持CUDA运行时API 4104之外,还支持例如OpenCL。此外,在至少一个实施例中,包括CUDA运行时4105在内的开发库可被视为与驱动器组件分离,包括用户模式的CUDA驱动器4107和内核模式的设备驱动器4108(有时也称为“显示”驱动器)。

[0388] 在至少一个实施例中,CUDA库4103可以包括但不限于数学库,深度学习库,并行算法库和/或信号/图像/视频处理库,并行计算应用程序(例如应用程序4101)可以利用这些库。在至少一个实施例中,CUDA库4103可包括数学库,例如cuBLAS库,其是用于执行线性代数运算的基本线性代数子程序(“BLAS”)的实现;用于计算快速傅立叶变换(“FFT”)的cuFFT库,以及用于生成随机数的cuRAND库等。在至少一个实施例中,CUDA库4103可以包括深度学习库,诸如用于深度神经网络的基元的cuDNN库和用于高性能深度学习推理的TensorRT平台等等。

[0389] 图42示出了根据至少一个实施例的图40的软件栈4000的ROCm实现。在至少一个实施例中,可在其上启动应用程序4201的ROCm软件栈4200包括语言运行时4203,系统运行时4205,thunk 4207,ROCm内核驱动器4208和设备内核驱动器4209。在至少一个实施例中,ROCm软件栈4200在硬件4209上执行,硬件4209可以包括支持ROCm的GPU,其由加利福尼亚州圣克拉拉市的AMD公司开发。

[0390] 在至少一个实施例中,应用程序4201可以执行与以上结合图40讨论的应用程序4001类似的功能。另外,在至少一个实施例中,语言运行时4203和系统运行时4205可以执行与以上结合图40讨论的运行时4005类似的功能。在至少一个实施例中,语言运行时4203和系统运行时4205的不同之处在于,系统运行时4205是实现ROCr系统运行时API 4204并利用异构系统架构(“HAS”)运行时API的语言无关运行时。在至少一个实施例中,HAS运行时API是一种瘦用户模式API,它公开接口以供访问和与AMD GPU交互,包括用于存储器管理、通过架构分派内核的执行控制、错误处理、系统和代理信息以及运行时初始化和关闭等的函数。在至少一个实施例中,与系统运行时4205相比,语言运行时4203是ROCr系统运行时API 4204之上分层的特定于语言的运行时API 4202的实现。在至少一个实施例中,语言运行时API可以包括但不限于可移植异构计算接口(“HIP”)语言运行时API,异构计算编译器(“HCC”)语言运行时API或OpenCL API等等。特别是,HIP语言是C++编程语言的扩展,具有CUDA机制的功能相似版本,并且在至少一个实施例中,HIP语言运行时API包括与以上结合图41讨论的CUDA运行时API 4104相似的函数,例如用于存储器管理、执行控制、设备管理、错误处理和同步等的函数。

[0391] 在至少一个实施例中，thunk(ROCt)4207是可用于与底层ROCm驱动器4208交互的接口。在至少一个实施例中，ROCm驱动器4208是ROCk驱动器，其是AMDGPU驱动器和HAS内核驱动器(amdkfd)的组合。在至少一个实施例中，AMDGPU驱动器是由AMD开发的用于GPU的设备内核驱动器，其执行与以上结合图41讨论的设备内核驱动器4106类似的功能。在至少一个实施例中，HAS内核驱动器是允许不同类型的处理器经由硬件特征更有效地共享系统资源的驱动器。

[0392] 在至少一个实施例中，各种库(未示出)可以被包括在语言运行时4203上方的ROCm软件栈4200中，并且提供与以上结合图41讨论的CUDA库4103相似的功能。在至少一个实施例中，各种库可以包括但不限于数学、深度学习和/或其他库，例如实现与CUDA cuBLAS类似的函数的hipBLAS库，类似于CUDA cuFFT用于计算FFT的rocFFT库等。

[0393] 图43示出了根据至少一个实施例的图40的软件栈4000的OpenCL实现。在至少一个实施例中，可以在其上启动应用程序4301的OpenCL软件栈4300包括OpenCL框架4305，OpenCL运行时4306和驱动器4307。在至少一个实施例中，OpenCL软件栈4300在不是特定于供应商的硬件4109上执行。在至少一个实施例中，由于由不同厂商开发的设备支持OpenCL，因此可能需要特定的OpenCL驱动器才能与来自此类厂商的硬件进行互操作。

[0394] 在至少一个实施例中，应用程序4301，OpenCL运行时4306，设备内核驱动器4307和硬件4308可以分别执行与上面结合图40讨论的应用程序4001、运行时4005、设备内核驱动器4006和硬件4007类似的功能。在至少一个实施例中，应用程序4301还包括具有将在设备上执行的代码的OpenCL内核4302。

[0395] 在至少一个实施例中，OpenCL定义了一种“平台”，其允许主机控制连接到该主机的设备。在至少一个实施例中，OpenCL框架提供平台层API和运行时API，示出为平台API 4303和运行时API 4305。在至少一个实施例中，运行时API 4305使用上下文来管理设备上内核的执行。在至少一个实施例中，每个标识的设备可以与各自的上下文相关联，运行时API 4305可以使用该上下文来管理该设备的命令队列、程序对象和内核对象、共享存储器对象等。在至少一个实施例中，平台API 4303公开了允许设备上下文用于选择和初始化设备，经由命令队列将工作提交给设备，以及使得能够进行来自和去往设备的数据传输等的函数。另外，在至少一个实施例中，OpenCL框架提供各种内置函数(未示出)，包括数学函数、关系函数和图像处理函数等。

[0396] 在至少一个实施例中，编译器4304也被包括在OpenCL框架4305中。在至少一个实施例中，源代码可以在执行应用程序之前被离线编译或者在执行应用程序期间被在线编译。与CUDA和ROCm相反，至少一个实施例中的OpenCL应用程序可以由编译器4304在线编译，编译器4304被包括以代表可以用于将源代码和/或IR代码(例如标准可移植中间表示(“SPIR-V”)代码)编译为二进制代码的任意数量的编译器。可替代地，在至少一个实施例中，可以在执行这样的应用程序之前离线编译OpenCL应用程序。

[0397] 图44示出了根据至少一个实施例的由编程平台支持的软件。在至少一个实施例中，编程平台4404被配置为支持应用程序4400可以依赖的各种编程模型4403，中间件和/或库4402以及框架4401。在至少一个实施例中，应用程序4400可以是使用例如深度学习框架(例如，MXNet，PyTorch或TensorFlow)实现的AI/ML应用，其可以依赖于诸如cuDNN，NVIDIA Collective Communications Library(“NCCL”)”和/或NVIDIA开发人员数据加载库

("DALI")CUDA库之类的库,以在底层硬件上提供加速的计算。

[0398] 在至少一个实施例中,编程平台4404可以是以上分别结合图41、图42和图43描述的CUDA、ROCm或OpenCL平台之一。在至少一个实施例中,编程平台4404支持多个编程模型4403,其是底层计算系统的抽象,其允许算法和数据结构的表达。在至少一个实施例中,编程模型4403可以暴露底层硬件的特征以便改善性能。在至少一个实施例中,编程模型4403可以包括但不限于CUDA,HIP,OpenCL,C++加速大规模并行性("C++AMP"),开放多处理("OpenMP"),开放加速器("OpenACC")和/或Vulcan计算(Vulcan Compute)。

[0399] 在至少一个实施例中,库和/或中间件4402提供编程模型4404的抽象的实现。在至少一个实施例中,这样的库包括可由计算机程序使用并在软件开发期间利用的数据和编程代码。在至少一个实施例中,除了可以从编程平台4404获得的那些之外,这样的中间件还包括向应用程序提供服务的软件。在至少一个实施例中,库和/或中间件4402可以包括但不限于cuBLAS、cuFFT、cuRAND和其他CUDA库,或rocBLAS、rocFFT、rocRAND和其他ROCm库。另外,在至少一个实施例中,库和/或中间件4402可以包括NCCL和ROCm通信集合库("RCCL")库,其提供用于GPU的通信例程,用于深度学习加速的MIOpen库和/或用于线性代数、矩阵和向量运算、几何变换、数值求解器以及相关算法的本征库。

[0400] 在至少一个实施例中,应用程序框架4401依赖于库和/或中间件4402。在至少一个实施例中,每个应用程序框架4401是用于实现应用软件的标准结构的软件框架。在至少一个实施例中,可以使用框架(诸如Caffe,Caffe2,TensorFlow,Keras,PyTorch或MxNet深度学习框架)来实现AI/ML应用。

[0401] 图45示出了根据至少一个实施例的编译代码以在图40-43的编程平台之一上执行。在至少一个实施例中,编译器4501接收源代码4500,其包括主机代码以及设备代码两者。在至少一个实施例中,编译器4501被配置为将源代码4500转换为用于在主机上执行的主机可执行代码4502以及用于在设备上执行的设备可执行代码4503。在至少一个实施例中,源代码4500可以在执行应用程序之前离线编译,或者在执行应用程序期间在线编译。

[0402] 在至少一个实施例中,源代码4500可以包括编译器4501支持的任何编程语言的代码,例如C++、C、Fortran等。在至少一个实施例中,源代码4500可以包括在单源(single-source)文件中,其具有主机代码和设备代码的混合,并在其中指示了设备代码的位置。在至少一个实施例中,单源文件可以是包括CUDA代码的.cu文件或包括HIP代码的.hip.cpp文件。备选地,在至少一个实施例中,源代码4500可以包括多个源代码文件,而不是单源文件,在该单源文件中主机代码和设备代码是分开的。

[0403] 在至少一个实施例中,编译器4501被配置为将源代码4500编译成用于在主机上执行的主机可执行代码4502和用于在设备上执行的设备可执行代码4503。在至少一个实施例中,编译器4501执行操作,包括将源代码4500解析为抽象系统树(AST),执行优化以及生成可执行代码。在源代码4500包括单源文件的至少一个实施例中,编译器4501可以将设备代码与主机代码在这种单源文件中分开,将设备代码和主机代码分别编译成设备可执行代码4503和主机可执行代码4502,以及将设备可执行代码4503和主机可执行代码4502在单个文件中链接到一起,如下面关于图34更详细讨论的。

[0404] 在至少一个实施例中,主机可执行代码4502和设备可执行代码4503可以是任何合适的格式,例如二进制代码和/或IR代码。在CUDA的情况下,在至少一个实施例中,主机可执

行代码4502可以包括本地对象代码，而设备可执行代码4503可以包括PTX中间表示的代码。在至少一个实施例中，在ROCm的情况下，主机可执行代码4502和设备可执行代码4503都可以包括目标二进制代码。

[0405] 本公开的至少一个实施例可以考虑以下条款来描述：

[0406] 1.一种网络设备，包括：

[0407] 至少一个处理器；以及

[0408] 至少一个存储器，其包括指令，所述指令响应于由所述至少一个处理器执行，使所述网络设备：

[0409] 接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据；

[0410] 预留所述网络设备的资源，以处理将从所述多个端点接收的第二网络数据，其中要预留的资源的量至少部分地基于从所述第一网络数据的报头获得的信息来确定；

[0411] 将所述第一网络数据发送至多个附加的网络设备，所述多个附加的网络设备至少部分基于从所述报头获得的所述信息被识别；

[0412] 接收所述第二网络数据；以及

[0413] 使用所述预留的资源来处理所述第二网络数据。

[0414] 2.根据条款1所述的网络设备，其中所述报头包括指示所述第一网络数据和端点设备的虚拟存储器空间之间的映射的信息。

[0415] 3.根据条款1或2所述的网络设备，其中所述多个端点中的端点包括并行处理单元，并且其中所述第一网络数据的至少一部分被写入所述并行处理单元的存储器中。

[0416] 4.根据条款1至3中的任一项所述的网络设备，其中所述第一网络数据响应于对并行处理单元的存储器的写操作而被发送至所述网络设备，其中所述第一网络数据包括由所述写操作写入所述存储器的数据。

[0417] 5.根据条款1至4中的任一项所述的网络设备，其中处理所述第二网络数据包括：至少部分地基于从所述报头获得的归约信息来对所述第二网络数据进行归约。

[0418] 6.根据条款1至5中的任一项所述的网络设备，其中所述网络设备将所述第二网络数据的归约发送至所述第一网络数据的发送者。

[0419] 7.根据条款1至6中的任一项所述的网络设备，所述至少一个存储器包括进一步的指令，所述指令响应于由所述至少一个处理器执行，使所述网络设备：

[0420] 存储从所述第一网络数据的所述报头获得的所述信息；以及

[0421] 响应于接收到所述第二网络数据来检索所述信息。

[0422] 8.根据条款1至7中的任一项所述的网络设备，其中所述报头包括用于所述多播操作的归约信息和路由信息。

[0423] 9.根据条款1至8中的任一项所述的网络设备，所述至少一个存储器包括进一步的指令，所述指令响应于由所述至少一个处理器执行，使所述网络设备：

[0424] 在向所述多个附加的网络设备发送所述第一网络数据之前，更新所述报头中的归约信息和路由信息。

[0425] 10.根据条款1至9中的任一项所述的网络设备，其中所述多个附加的网络设备包括交换机、路由器或端点中的至少一个。

[0426] 11.根据条款1至10中的任一项所述的网络设备，所述至少一个存储器包括进一步

的指令，所述指令响应于由所述至少一个处理器执行，使所述网络设备：

[0427] 响应于确定自从发送所述第一网络数据起已经过了阈值时间量，并且所述附加的网络设备中的至少一个附加的网络设备尚未对接收到所述第一网络数据作出响应，释放所述预留的资源。

[0428] 12.一种非暂时性机器可读介质，其上存储有指令，所述指令响应于由一个或更多个处理器执行时，使所述一个或更多个处理器至少：

[0429] 在网络设备处接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据；

[0430] 预留所述网络设备的资源，以处理将从所述多个端点接收的第二网络数据，其中要预留的资源至少部分地基于从所述第一网络数据获得的信息来确定；

[0431] 将所述第一网络数据发送至多个附加的网络设备，所述多个附加的网络设备至少部分地基于从所述第一网络数据获得的所述信息被识别；

[0432] 接收所述第二网络数据；以及

[0433] 使用所述预留的资源来处理所述第二网络数据。

[0434] 13.根据条款12所述的非暂时性机器可读介质，其中所述第一网络数据包括一个或更多个报头，所述一个或更多个报头包括指示所述第一网络数据和端点设备的虚拟存储器空间之间的映射的信息。

[0435] 14.根据条款12或13所述的非暂时性机器可读介质，其中所述多个端点中的端点包括并行处理单元，并且其中所述第一网络数据的至少一部分由所述并行处理单元用于执行所述多播操作的至少一部分。

[0436] 15.根据条款12至14中的任一项所述的非暂时性机器可读介质，其中所述网络设备接收响应于对端点上的并行处理单元的存储器的读或写操作中的至少一个而发送的第一网络数据。

[0437] 16.根据条款12至15中的任一项所述的非暂时性机器可读介质，其中处理所述第二网络数据包括：至少部分地基于从所述第一网络数据中的一个或更多个报头获得的归约信息来对所述第二网络数据进行归约。

[0438] 17.根据条款12至16中的任一项所述的非暂时性机器可读介质，其上存储有进一步的指令，所述指令如果由一个或更多个处理器执行，则使所述一个或更多个处理器至少：

[0439] 存储从所述第一网络数据的所述报头获得的所述信息；

[0440] 响应于接收到所述第二网络数据来检索所述信息；以及

[0441] 使用所述信息来处理所述第二网络数据。

[0442] 18.根据条款12至17中的任一项所述的非暂时性机器可读介质，其中所述第一网络数据中的一个或更多个报头包括用于所述多播操作的归约信息和路由信息。

[0443] 19.根据条款12至18中的任一项所述的非暂时性机器可读介质，其上存储有进一步的指令，所述指令如果由一个或更多个处理器执行，则使所述一个或更多个处理器至少：

[0444] 在向所述多个附加的网络设备发送所述第一网络数据之前，更新所述报头中的归约信息和路由信息。

[0445] 20.根据条款12至19中的任一项所述的非暂时性机器可读介质，其上存储有进一步的指令，所述指令如果由一个或更多个处理器执行，则使所述一个或更多个处理器至少：

[0446] 响应于确定自从发送所述第一网络数据起已经过了阈值时间量，并且所述附加的网络设备中的至少一个附加的网络设备尚未对接收到所述第一网络数据作出响应，释放预留的资源。

[0447] 21.一种方法，包括：

[0448] 在网络设备处接收与将由至少多个端点共同执行的多播操作相关联的第一网络数据；

[0449] 预留所述网络设备的资源，以处理将从所述多个端点接收的第二网络数据，其中要预留的资源至少部分地基于从所述第一网络数据获得的信息来确定；

[0450] 将所述第一网络数据从所述网络设备发送至多个附加的网络设备，所述多个附加的网络设备至少部分地基于从所述第一网络数据获得的信息被识别；

[0451] 在所述网络设备处，接收所述第二网络数据；以及

[0452] 由所述网络设备使用所述预留的资源来处理所述第二网络数据。

[0453] 22.根据条款21所述的方法，其中所述第一网络数据包括一个或更多个报头，所述一个或更多个报头包括指示所述第一网络数据和端点设备的虚拟存储器空间之间的映射的信息。

[0454] 23.根据条款21或22所述的方法，其中所述多个端点中的端点包括并行处理单元，并且其中所述第一网络数据的至少一部分由所述并行处理单元用于执行所述多播操作的至少一部分。

[0455] 24.根据条款2至23中的任一项所述的方法，其中所述网络设备接收响应于对端点上的并行处理单元的存储器的读或写操作中的至少一个而发送的第一网络数据。

[0456] 25.根据条款21至24中的任一项所述的方法，进一步包括：

[0457] 至少部分地基于从所述第一网络数据中的一个或更多个报头获得的归约信息来处理所述第二网络数据。

[0458] 26.根据条款21至25中的任一项所述的方法，进一步包括：

[0459] 存储从所述第一网络数据的所述报头获得的所述信息；

[0460] 响应于接收到所述第二网络数据来检索所述信息；以及

[0461] 使用所检索的信息来处理所述第二网络数据。

[0462] 27.根据条款21至26中的任一项所述的方法，其中所述第一网络数据中的一个或更多个报头包括用于所述多播操作的归约信息和路由信息。

[0463] 28.根据条款21至27中的任一项所述的方法，进一步包括：

[0464] 在向所述多个附加的网络设备发送所述第一网络数据之前，更新所述报头中的归约信息和路由信息。

[0465] 29.根据条款21至28中的任一项所述的方法，进一步包括：

[0466] 响应于确定自从发送所述第一网络数据起已经过了阈值时间量，并且所述附加的网络设备中的至少一个附加的网络设备尚未对接收到所述第一网络数据作出响应，释放所述预留的资源。

[0467] 30.根据条款21至29中的任一项所述的方法，进一步包括：

[0468] 确定所述网络设备的资源不足以可用于处理将从所述多个端点接收的所述第二网络数据；以及

[0469] 保持所述第一网络数据，直到有足够的资源可用。

[0470] 其他变型在本公开的精神内。因此，尽管公开的技术易于进行各种修改和替代配置，但是某些示出的其实施例在附图中示出并且已经在上面进行了详细描述。然而，应理解，无意将公开内容限制为所公开的一种或更多种特定形式，而是相反，其意图是涵盖落入如所附权利要求书所定义的本公开内容的精神和范围内的所有修改、替代配置和等同物。

[0471] 除非另有说明或显然与上下文矛盾，否则在描述所公开的实施例的上下文中(特别是在所附权利要求的上下文中)，术语“一”和“一个”和“该”以及类似指代的使用应被解释为涵盖单数和复数，而不是作为术语的定义。除非另有说明，否则术语“包括”、“具有”、“包含”和“含有”应被解释为开放式术语(意味着“包括但不限于”)。术语“连接”(在未经修改时指的是物理连接)应解释为部分或全部包含在内、附接到或连接在一起，即使有某些介入。除非本文另外指出，否则本文中对数值范围的引用仅旨在用作分别指代落入该范围内的每个单独值的简写方法，并且每个单独值都被并入说明书中，就如同其在本文中被单独叙述一样。在至少一个实施例中，除非另外指出或与上下文矛盾，否则术语“集”(例如“项目集”)或“子集”的使用应解释为包括一个或更多个成员的非空集合。此外，除非另外指出或与上下文矛盾，否则术语相应集的“子集”不一定表示对应集的适当子集，而是子集和对应集可以相等。

[0472] 除非以其他方式明确指出或与上下文明显矛盾，否则诸如“A,B和C中的至少一个”或“A,B与C中的至少一个”形式的短语之类的连接语在上下文中理解为通常用来表示项目、条款等，其可以是A或B或C，也可以是A和B和C集的任何非空子集。例如，在具有三个成员的集的说明性示例中，连接短语“A,B和C中的至少一个”和“A,B与C中的至少一个”是指以下任意集：{A}，{B}，{C}，{A,B}，{A,C}，{B,C}，{A,B,C}。因此，这种连接语言通常不旨在暗示某些实施例要求存在A中的至少一个，B中的至少一个和C中的至少一个。另外，除非另有说明或与上下文矛盾，否则术语“多个”表示复数的状态(例如，“多个项目”表示多个项目)。在至少一个实施例中，多个项目中项目的数量至少为两个，但如果明确指示或通过上下文指示，则可以更多。此外，除非另有说明或从上下文中可以清楚得知，否则短语“基于”是指“至少部分基于”而不是“仅基于”。

[0473] 除非本文另外指出或与上下文明显矛盾，否则本文描述的过程的操作可以任何合适的顺序执行。在至少一个实施例中，诸如本文所述的那些过程(或其变形和/或其组合)之类的过程在配置有可执行指令的一个或更多个计算机系统的控制下执行，并且被实现为代码(例如，可执行指令，一个或更多个计算机程序或一个或更多个应用程序)，该代码通过硬件或其组合在一个或更多个处理器上共同执行。在至少一个实施例中，代码以计算机程序的形式存储在计算机可读存储介质上，在至少一个实施例中，该计算机程序包括可由一个或更多个处理器执行的多个指令。在至少一个实施例中，计算机可读存储介质是非暂时性计算机可读存储介质，其排除了暂时性信号(例如，传播的瞬态电或电磁传输)，但包括非暂时性数据存储电路(例如，缓冲区、高速缓存和队列)。在至少一个实施例中，代码(例如，可执行代码或源代码)被存储在其上存储有可执行指令的一组一个或更多个非暂时性计算机可读存储介质(或用于存储可执行指令的其他存储器)上，该可执行指令在由计算机系统的一个或更多个处理器执行时(即，作为被执行的结果)，使得计算机系统执行本文所述的操作。在至少一个实施例中，一组非暂时性计算机可读存储介质包括多个非暂时性计算机可

读存储介质，并且多个非暂时性计算机可读存储介质中的个体非暂时性存储介质中的一个或更多个缺少全部代码，而是多个非暂时性计算机可读存储介质共同存储全部代码。在至少一个实施例中，可执行指令被执行，以使得不同的指令由不同的处理器执行，在至少一个实施例中，非暂时性计算机可读存储介质存储指令，并且主中央处理单元(“CPU”)执行一些指令，而图形处理单元(“GPU”)执行其他指令。在至少一个实施例中，计算机系统的不同组件具有单独的处理器，并且不同的处理器执行指令的不同子集。

[0474] 因此，在至少一个实施例中，计算机系统被配置为实现单独地或共同地执行本文所述的过程的操作的一个或更多个服务，并且这样的计算机系统被配置有使能实施操作的适用的硬件和/或软件。此外，实现本公开的至少一个实施例的计算机系统是单个设备，并且在另一实施例中是分布式计算机系统，其包括以不同方式操作的多个设备，使得分布式计算机系统执行本文所述的操作，并且使得单个设备不执行所有操作。

[0475] 本文提供的任何和所有示例或示例性语言(例如，“诸如”)的使用仅旨在更好地阐明本公开的实施例，并且不对公开的范围构成限制，除非另有要求。说明书中的任何语言都不应被解释为表示任何未要求保护的要素对于实践公开内容是必不可少的。

[0476] 本文引用的所有参考文献，包括出版物、专利申请和专利，均通过引用并入本文，其程度就如同每个参考文献被单独且具体地指示为以引用的方式并入本文并且其全部内容在本文中阐述一样。

[0477] 在说明书和权利要求中，可以使用术语“耦合”和“连接”以及它们的派生词。应当理解，这些术语可能不旨在作为彼此的同义词。相反，在特定的示例中，“连接”或“耦合”可用于指示两个或更多个元件彼此直接或间接物理或电接触。“耦合”也可能意味着两个或更多个元素彼此不直接接触，但仍彼此协作或交互。

[0478] 除非另有明确说明，否则可以理解，在整个说明书中，诸如“处理”、“计算”、“计算”、“确定”等之类的术语，是指计算机或计算系统或类似的电子计算设备的动作和/或过程，其将计算系统的寄存器和/或存储器中表示为物理量(例如电子)的数据处理和/或转换为类似表示为计算系统的存储器、寄存器或其他此类信息存储、传输或显示设备中的物理量的其他数据。

[0479] 以类似的方式，术语“处理器”可以指处理来自寄存器和/或存储器的电子数据并将该电子数据转换成可以存储在寄存器和/或存储器中的其他电子数据的任何设备或存储器的一部分。作为非限制性示例，“处理器”可以是CPU或GPU。“计算平台”可以包括一个或更多个处理器。如本文所使用的，在至少一个实施例中，“软件”进程可以包括随时间执行工作的软件和/或硬件实体，诸如任务、线程和智能代理。同样，每个过程可以指代多个过程，以连续地或间歇地顺序地或并行地执行指令。术语“系统”和“方法”在本文中可以互换使用，只要系统可以体现一种或更多种方法，并且方法可以被认为是系统。

[0480] 在至少一个实施例中，算术逻辑单元是一组组合逻辑电路，其采用一个或更多个输入来产生结果。在至少一个实施例中，处理器使用算术逻辑单元来实现数学运算，例如加法、减法或乘法。在至少一个实施例中，算术逻辑单元用于实现逻辑运算，例如逻辑与/或(AND/OR)或者异或(XOR)。在至少一个实施例中，算术逻辑单元是无状态的，并且由诸如半导体晶体管之类的物理开关组件制成，所述半导体晶体管布置成形成逻辑门。在至少一个实施例中，算术逻辑单元可以在内部作为具有关联时钟的有状态逻辑电路来操作。在至少

一个实施例中，算术逻辑单元可以被构造为异步逻辑电路，其内部状态未保持在相关联的寄存器组中。在至少一个实施例中，处理器使用算术逻辑单元来组合存储在处理器的一个或更多个寄存器中的操作数并产生可由处理器存储在另一寄存器或存储器位置中的输出。

[0481] 在至少一个实施例中，作为处理由处理器检索的指令的结果，处理器将一个或更多个输入或操作数呈现给算术逻辑单元，使得算术逻辑单元产生至少部分基于提供给算术逻辑单元的输入的指令代码的结果。在至少一个实施例中，由处理器提供给ALU的指令代码至少部分地基于由处理器执行的指令。在至少一个实施例中，ALU中的组合逻辑处理输入并产生输出，该输出放置在处理器内的总线上。在至少一个实施例中，处理器在输出总线上选择目的地寄存器、存储器位置、输出设备或输出存储位置，从而对处理器进行计时使得由ALU产生的结果被发送到期望的位置。

[0482] 在本文件中，可以参考获得、获取、接收或将模拟或数字数据输入子系统、计算机系统或计算机实现的机器中。在至少一个实施例中，可以通过多种方式来完成获得、获取、接收或输入模拟和数字数据的过程，例如通过接收作为函数调用或对应用程序编程接口的调用的参数的数据。在一些实现方式中，可以通过经由串行或并行接口传输数据来完成获得、获取、接收或输入模拟或数字数据的过程。在另一实现方式中，可以通过经由计算机网络将数据从提供实体传输到获取实体来完成获得、获取、接收或输入模拟或数字数据的过程。也可以参考提供、输出、传送、发送或呈现模拟或数字数据。在各种示例中，提供、输出、传送、发送或呈现模拟或数字数据的过程可以通过将数据作为函数调用的输入或输出参数、应用程序编程接口或进程间通信机制的参数进行传输来实现。

[0483] 尽管上面的讨论阐述了所描述的技术的示例实现方式，但是其他架构可以用于实现所描述的功能，并且旨在落入本公开的范围内。此外，尽管出于讨论的目的在上面定义了具体的职责分配，但是根据情况，可以以不同的方式分配和划分各种功能和职责。

[0484] 此外，尽管已经用特定于结构特征和/或方法动作的语言描述了主题，但是应当理解，所附权利要求书所要求保护的主题不必限于所描述的特定特征或动作。而是，公开了特定的特征和动作作为实现权利要求的示例性形式。

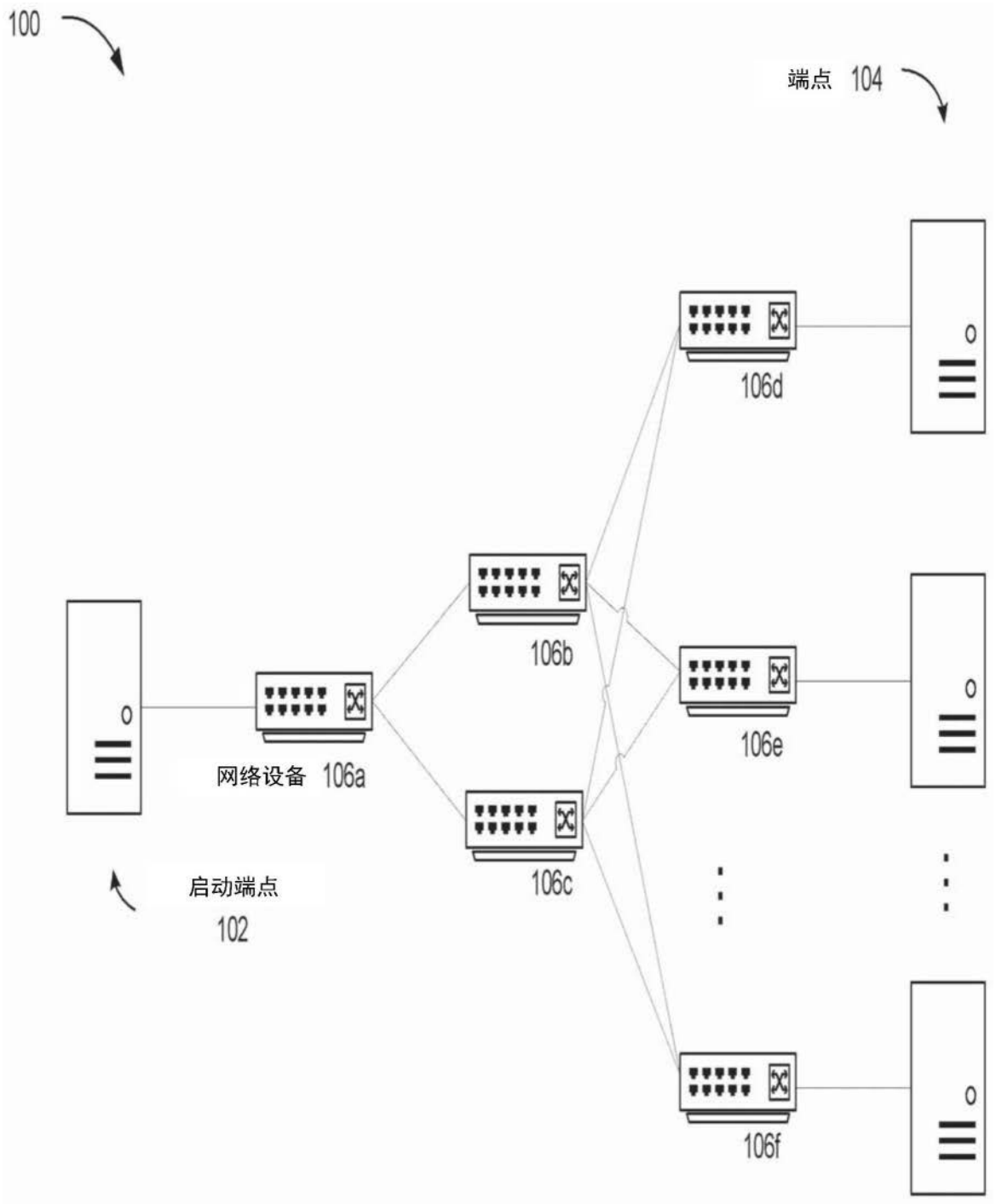


图1


200
网络设备 206
一个或更多个网络接口
210
处理器
212
输入缓冲区
216
存储器
214
输出缓冲区
218
一个或更多个网络接口
220

图2

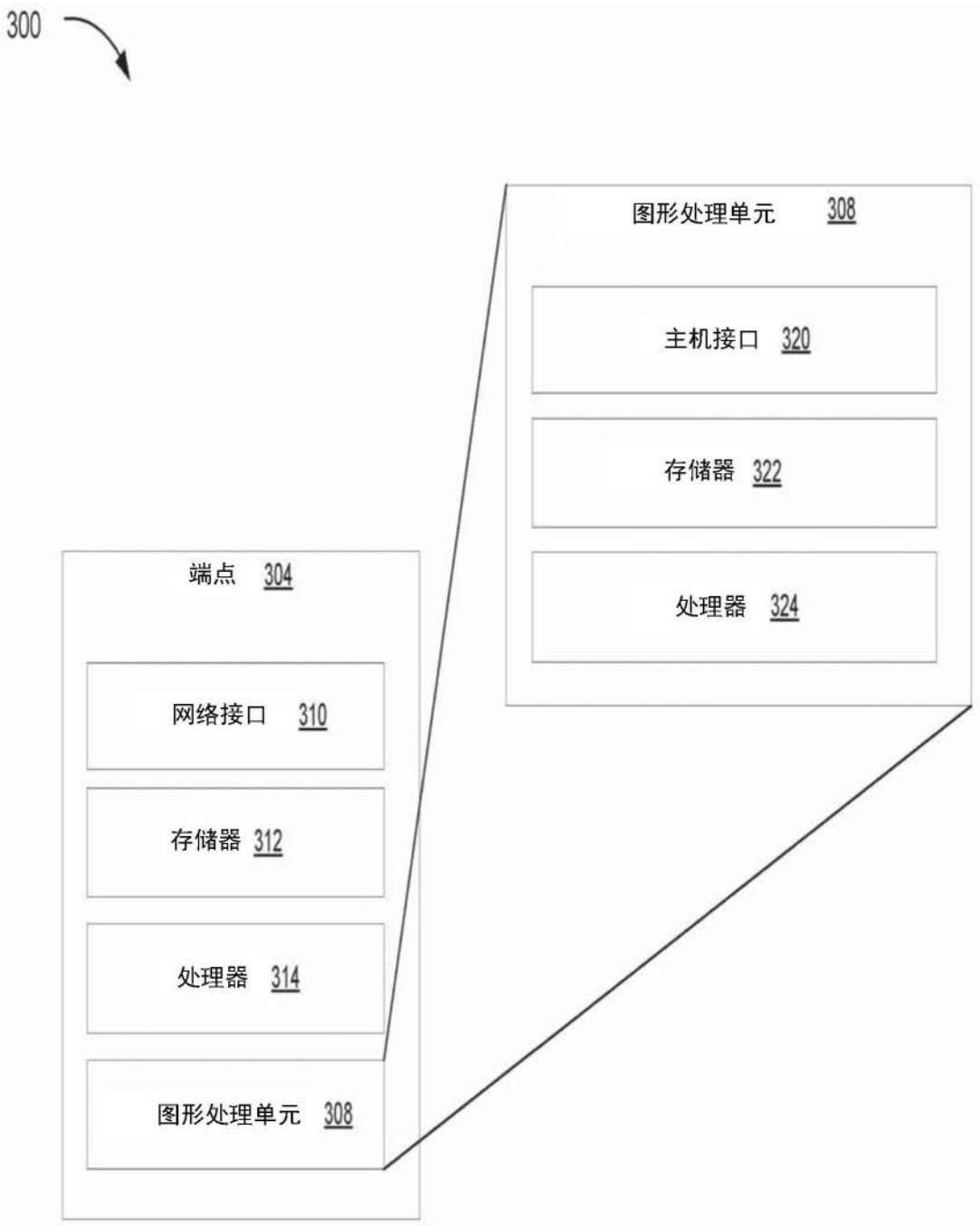

300
图形处理单元 308
主机接口 320
存储器 322
处理器 324
端点 304
网络接口 310
存储器 312
处理器 314
图形处理单元 308

图3

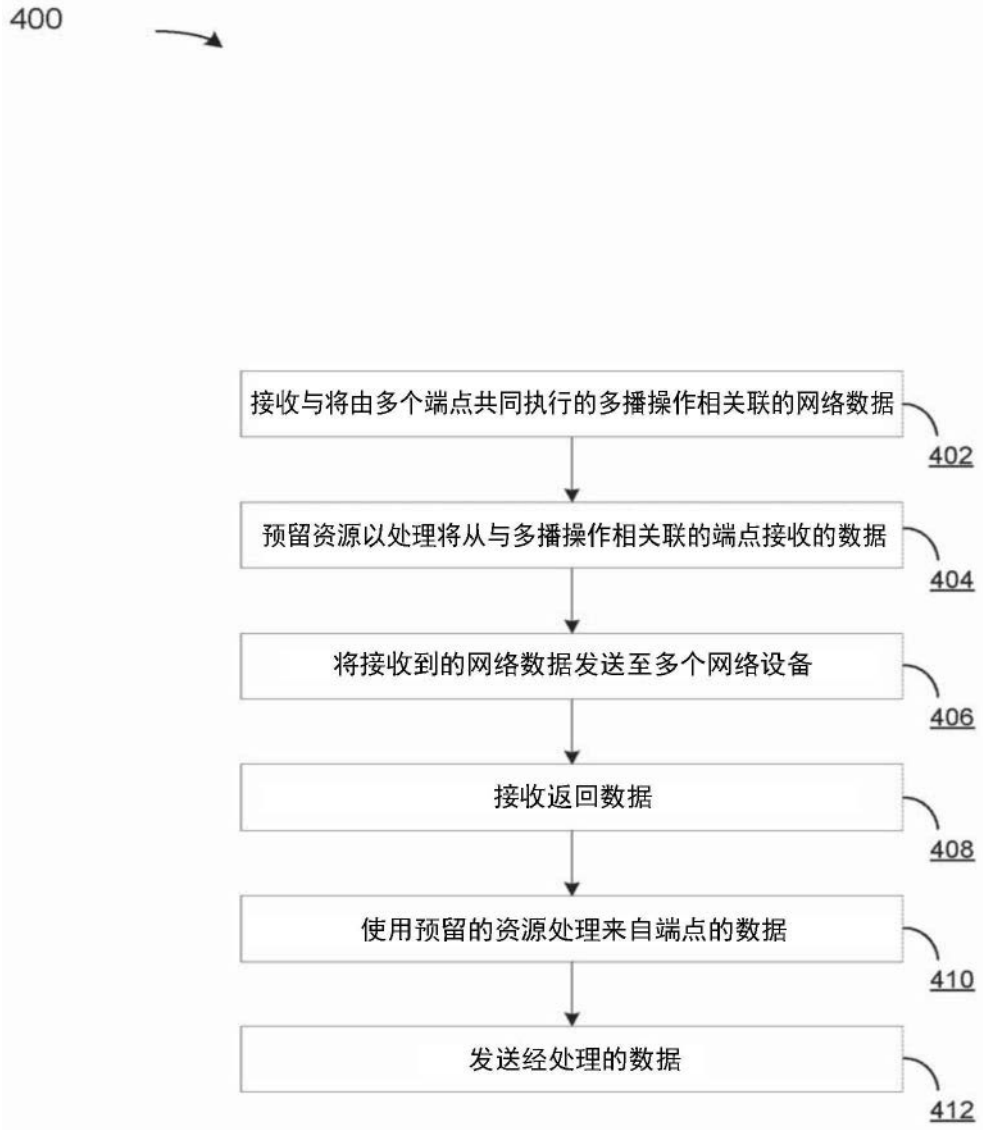

400
接收与将由多个端点共同执行的多播操作相关联的网络数据
402
预留资源以处理将从与多播操作相关联的端点接收的数据
404
将接收到的网络数据发送至多个网络设备
406
接收返回数据
408
使用预留的资源处理来自端点的数据
410
发送经处理的数据
412

图4

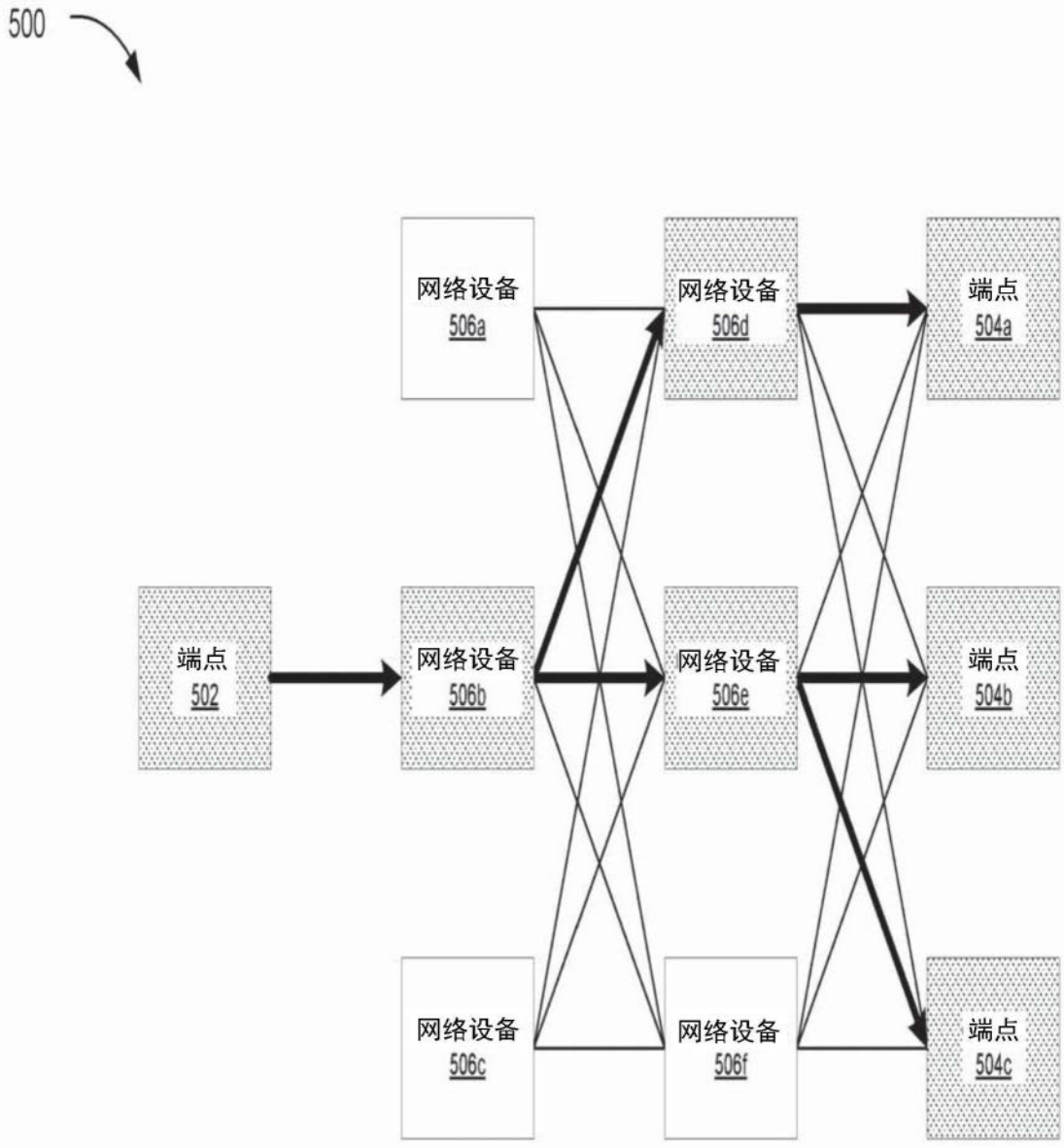

500
网络设备
506a
网络设备
506d
端点
504a
端点
502
网络设备
506b
网络设备
506e
端点
504b
网络设备
506c
网络设备
506f
端点
504c

图5

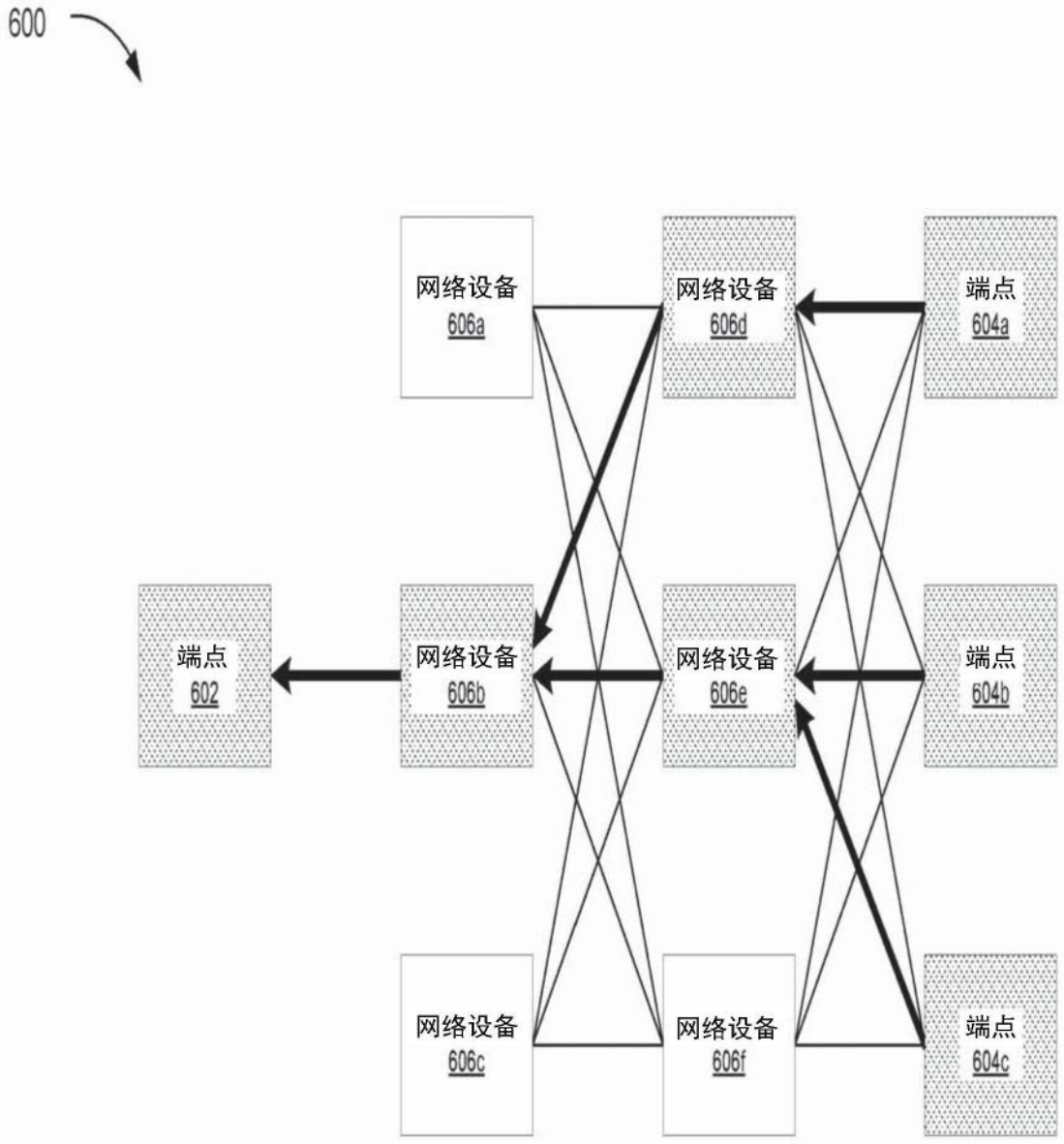

600
网络设备
606a
网络设备
606d
端点
604a
端点
602
网络设备
606b
网络设备
606e
端点
604b
网络设备
606c
网络设备
606f
端点
604c

图6

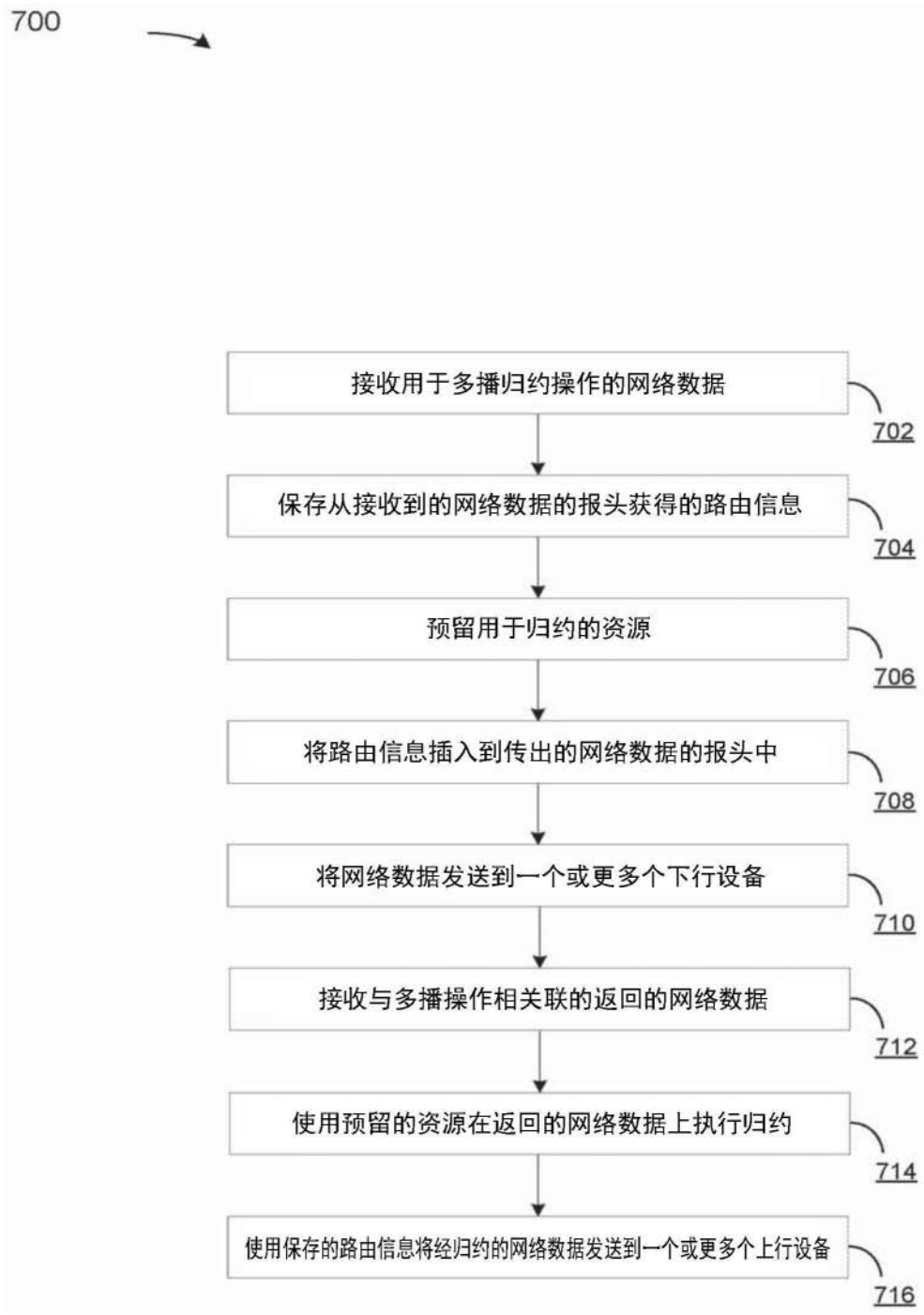

700
接收用于多播归约操作的网络数据
702
保存从接收到的网络数据的报头获得的路由信息
704
预留用于归约的资源
706
将路由信息插入到传出的网络数据的报头中
708
将网络数据发送到一个或更多个下行设备
710
接收与多播操作相关联的返回的网络数据
712
使用预留的资源在返回的网络数据上执行归约
714
使用保存的路由信息将经归约的网络数据发送到一个或更多个上行设备
716

图7

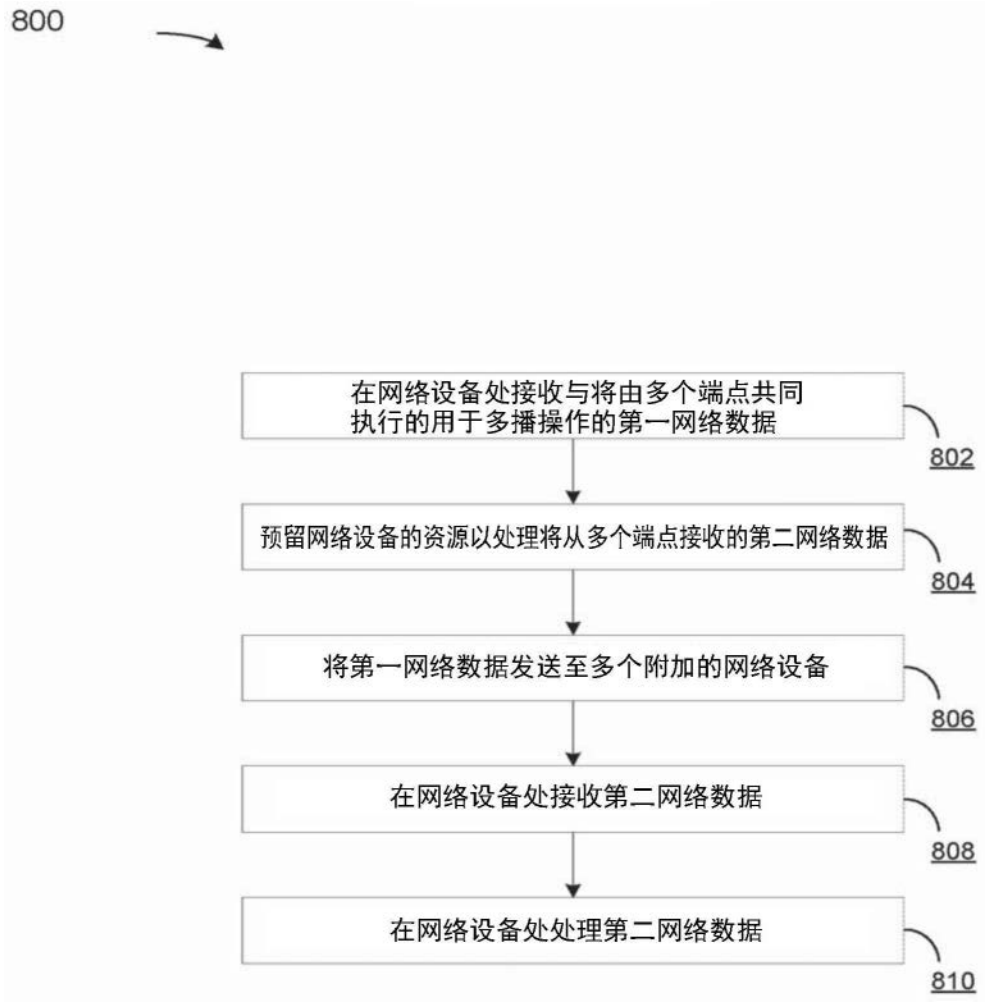

800
在网络设备处接收与将由多个端点共同执行的用于多播操作的第一网络数据
802
预留网络设备的资源以处理将从多个端点接收的第二网络数据
804
将第一网络数据发送至多个附加的网络设备
806
在网络设备处接收第二网络数据
808
在网络设备处处理第二网络数据
810

图8

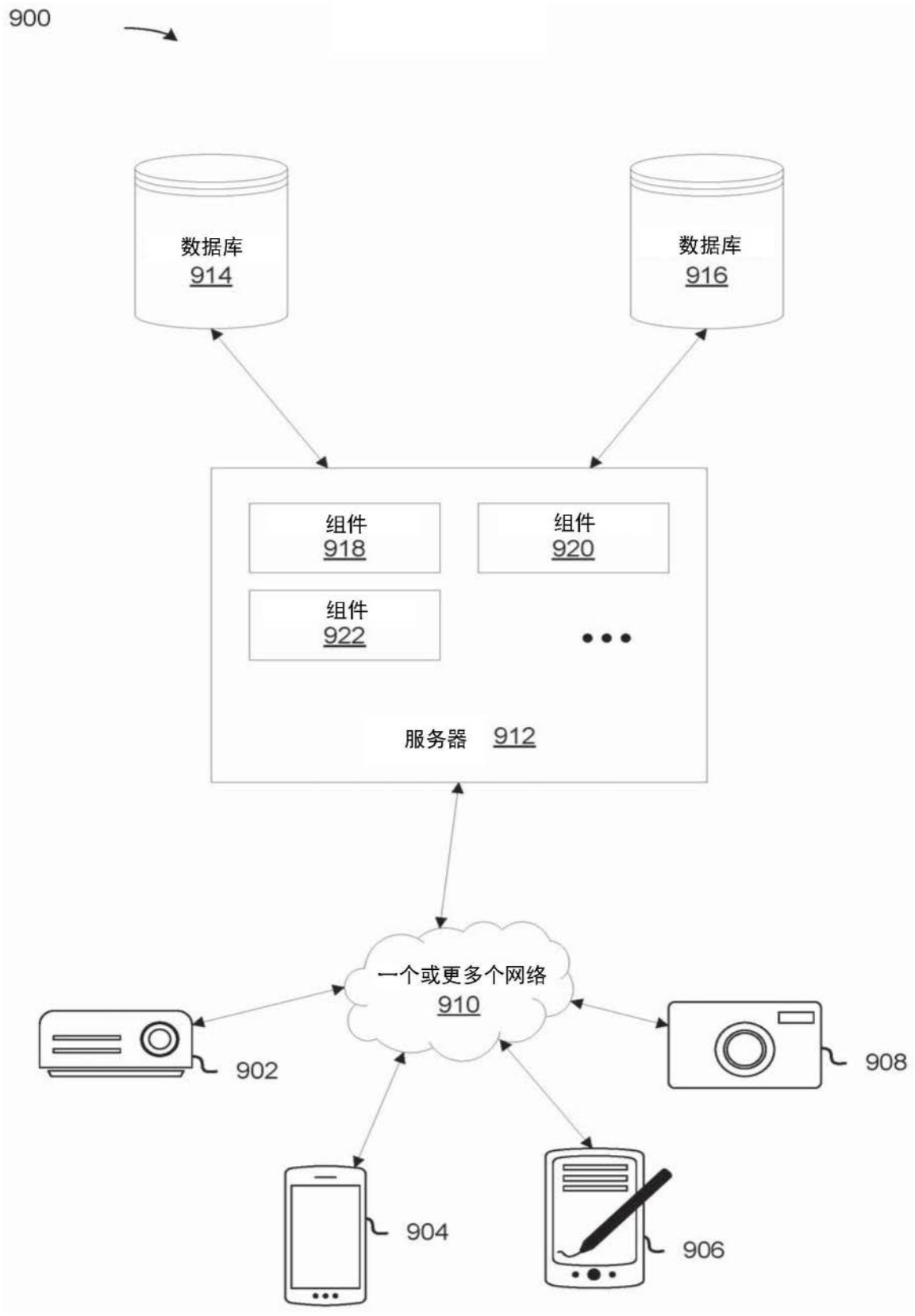

900
数据库
914
数据库
916
组件
918
组件
920
组件
922
服务器 912
一个或更多个网络
910
902
908
904
906

图9


数据中心
1000
应用层 1040
一个或更多个应用程序 1042
软件层 1030
软件 1052
框架层 1020
作业调度器 1032
配置管理器 1034
分布式文件系统 1038
资源管理器 1036
数据中心基础设施层 1010
资源协调器 1012
分组的计算资源 1014
节点C. R. 1016(1)
节点C. R. 1016(2)
节点C. R. 1016(N)

图10

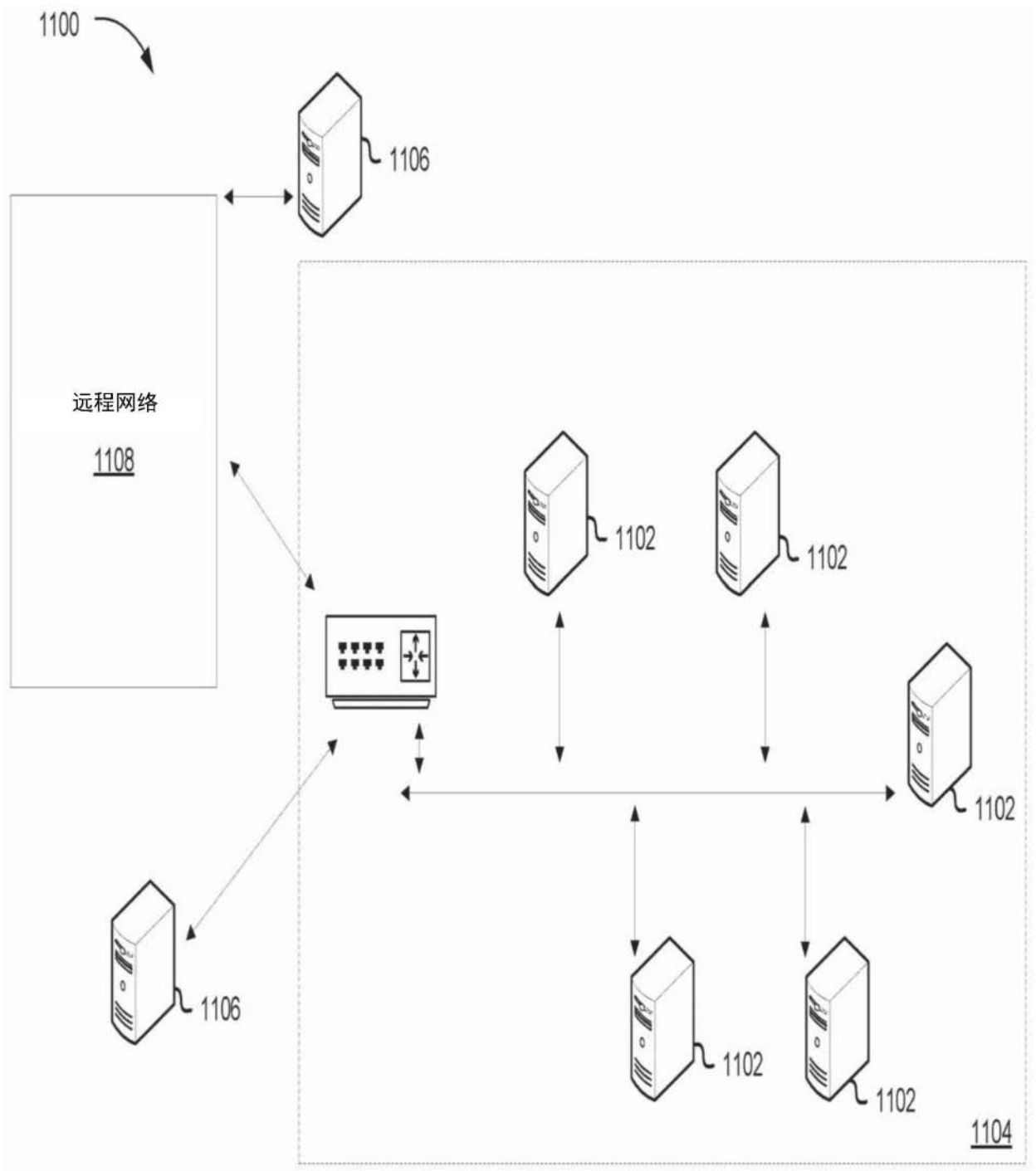


图11

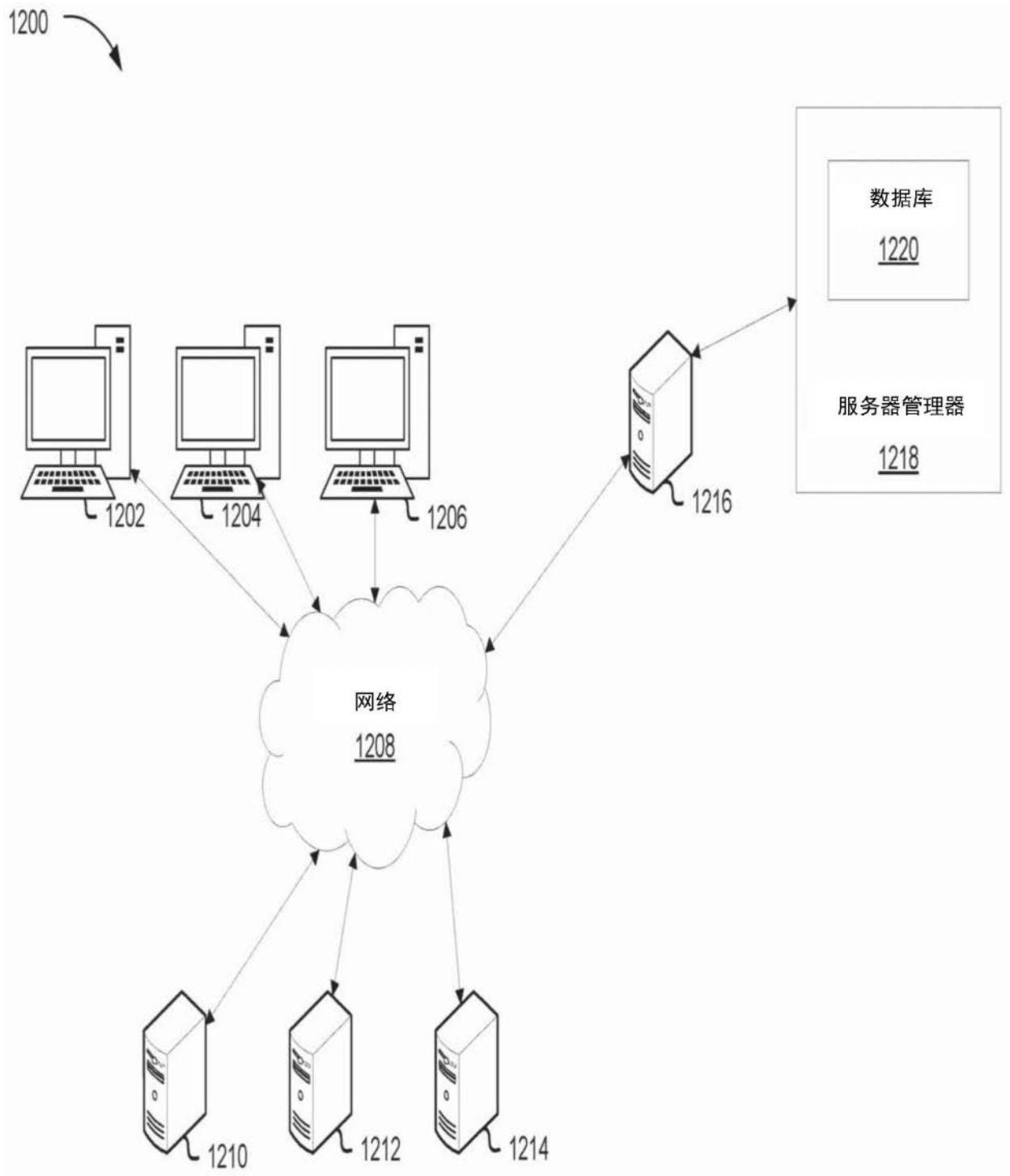

1200
数据库
1220
服务器管理器
1218
1202
1204
1206
1216
网络
1208
1210
1212
1214

图12

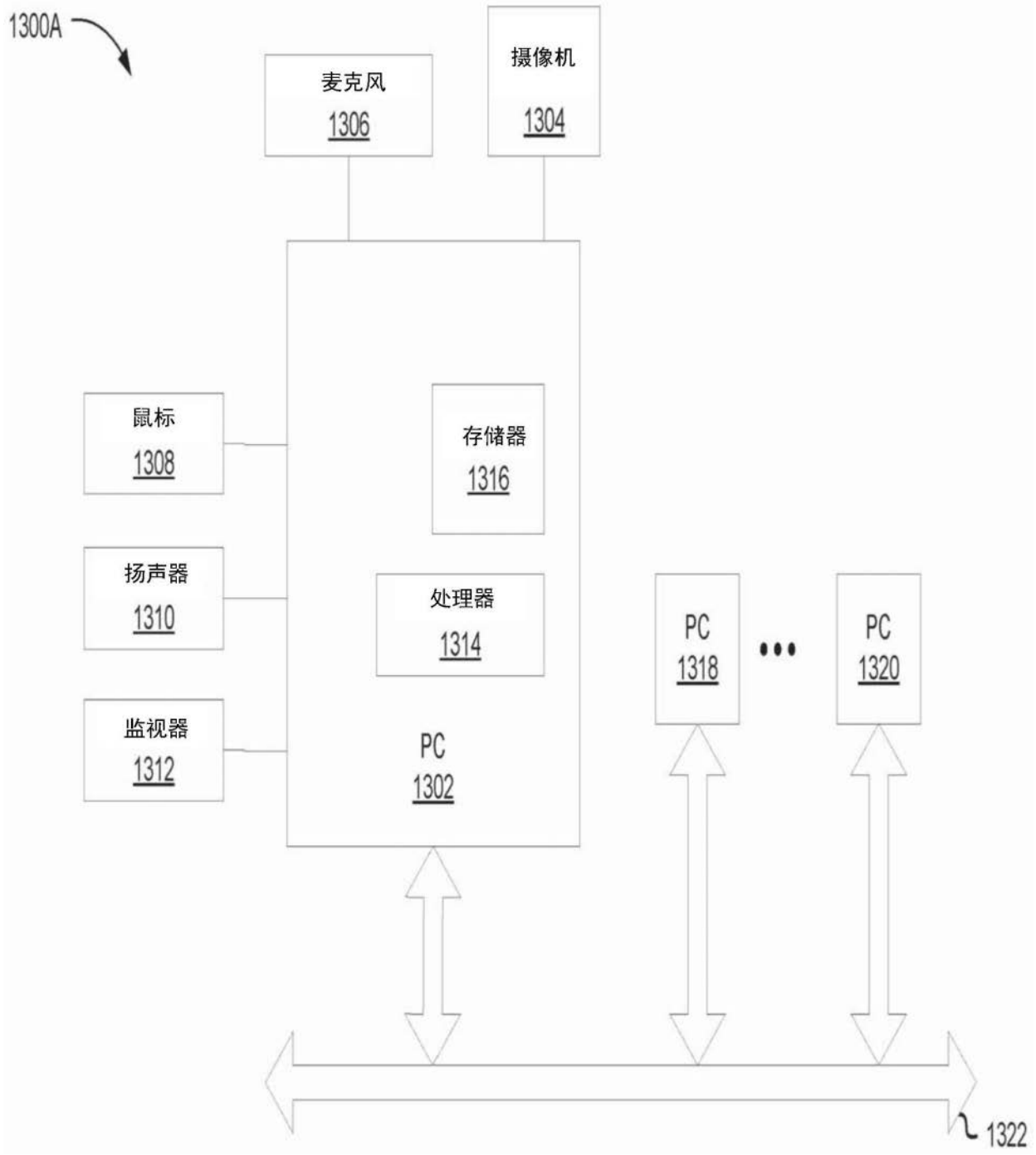

1300A
麦克风
1306
摄像机
1304
鼠标
1308
存储器
1316
扬声器
1310
处理器
1314
监视器
1312
PC
1302
PC
1318
PC
1320
1322

图13A

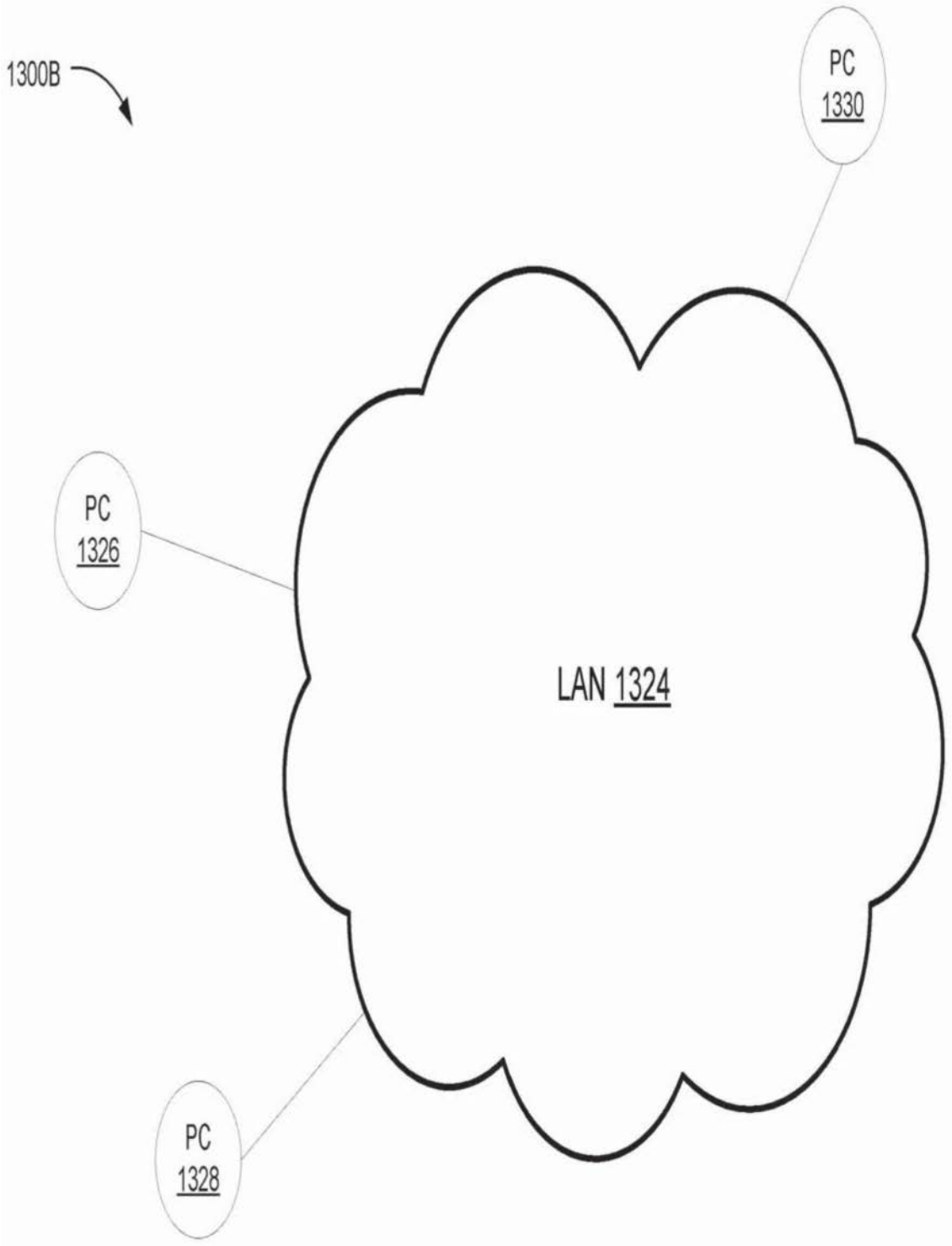


图13B

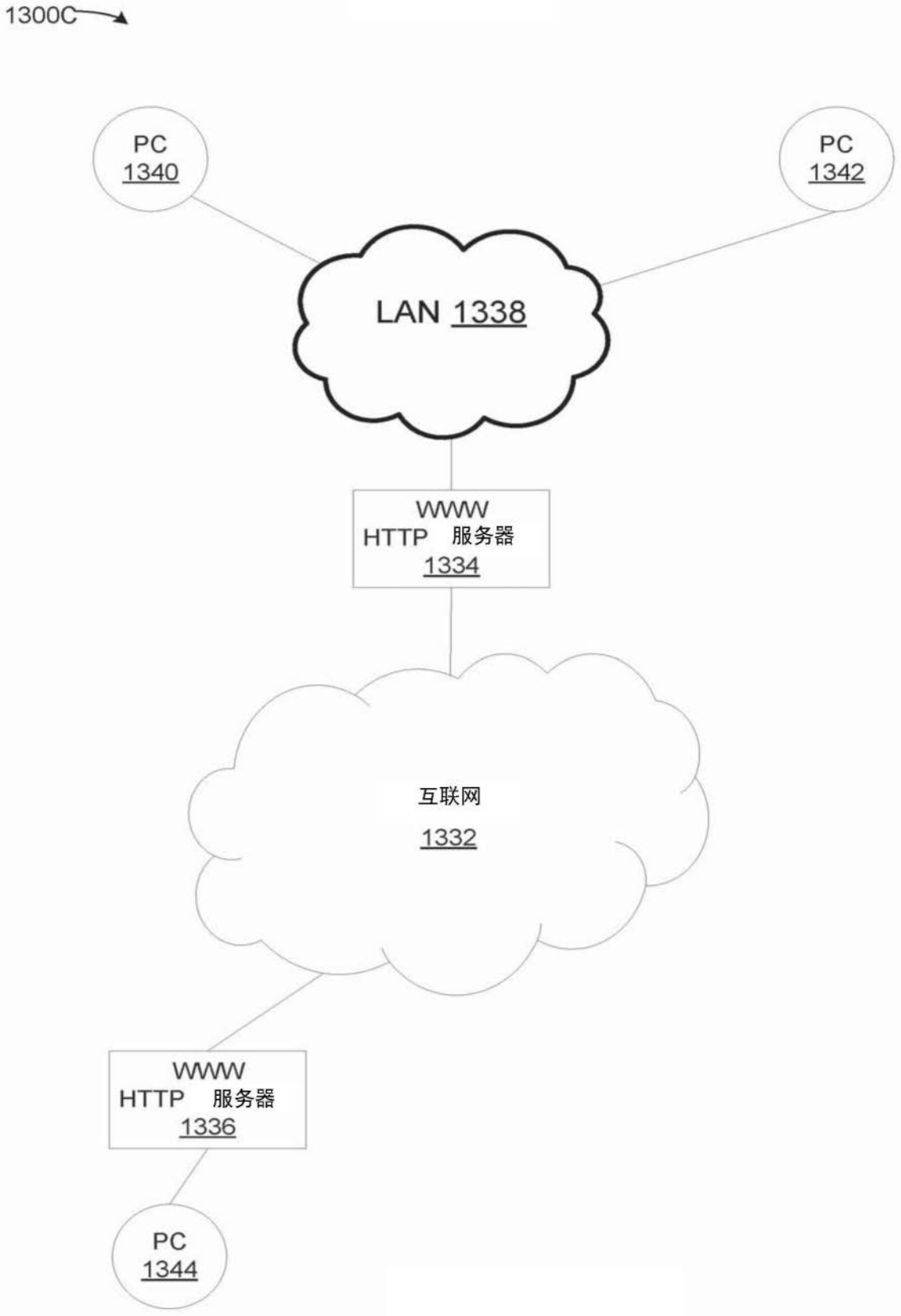

1300C
PC
1340
PC
1342
LAN 1338
WWW
HTTP 服务器
1334
互联网
1332
WWW
HTTP 服务器
1336
PC
1344

图13C

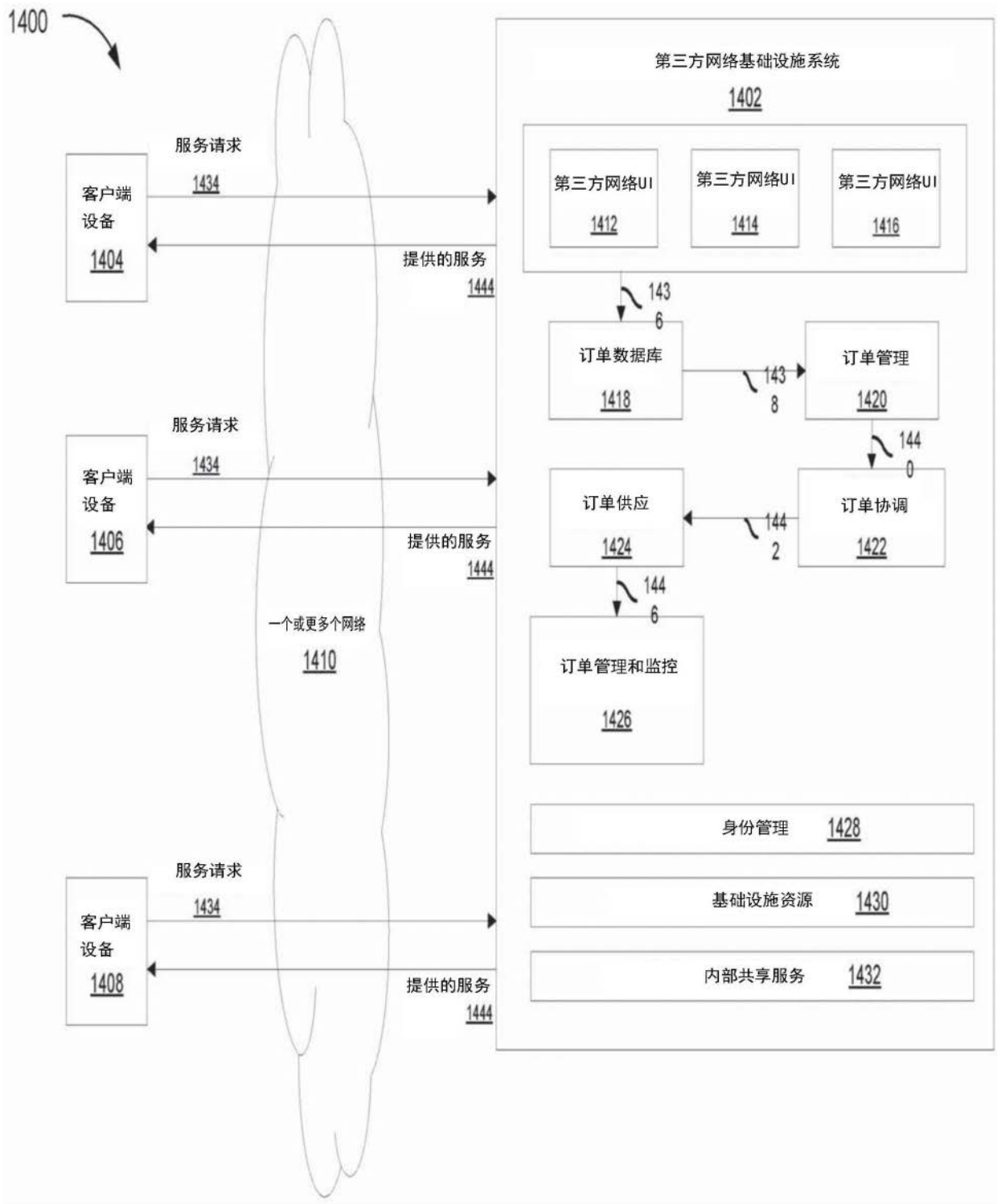

1400
第三方网络基础设施系统
1402
第三方网络UI
1412
第三方网络UI
1414
第三方网络UI
1416
服务请求
1434
客户端设备
1404
提供的服务
1444
1436
订单数据库
1418
1438
订单管理
1420
1440
订单协调
1422
1442
订单供应
1424
1446
订单管理和监控
1426
服务请求
1434
客户端设备
1406
提供的服务
1444
一个或更多个网络
1410
身份管理 1428
基础设施资源 1430
内部共享服务 1432
服务请求
1434
客户端设备
1408
提供的服务
1444

图14

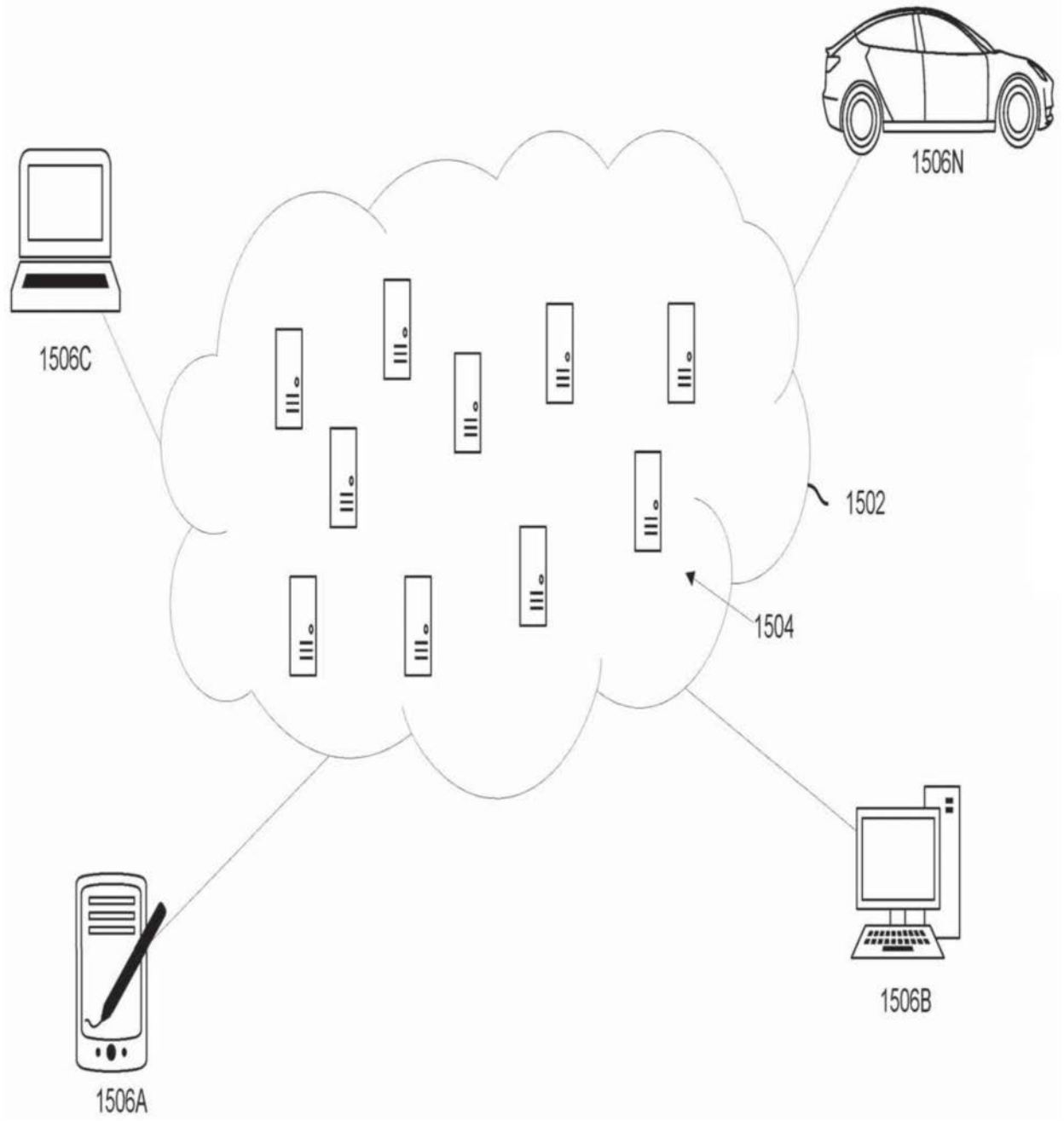


图15

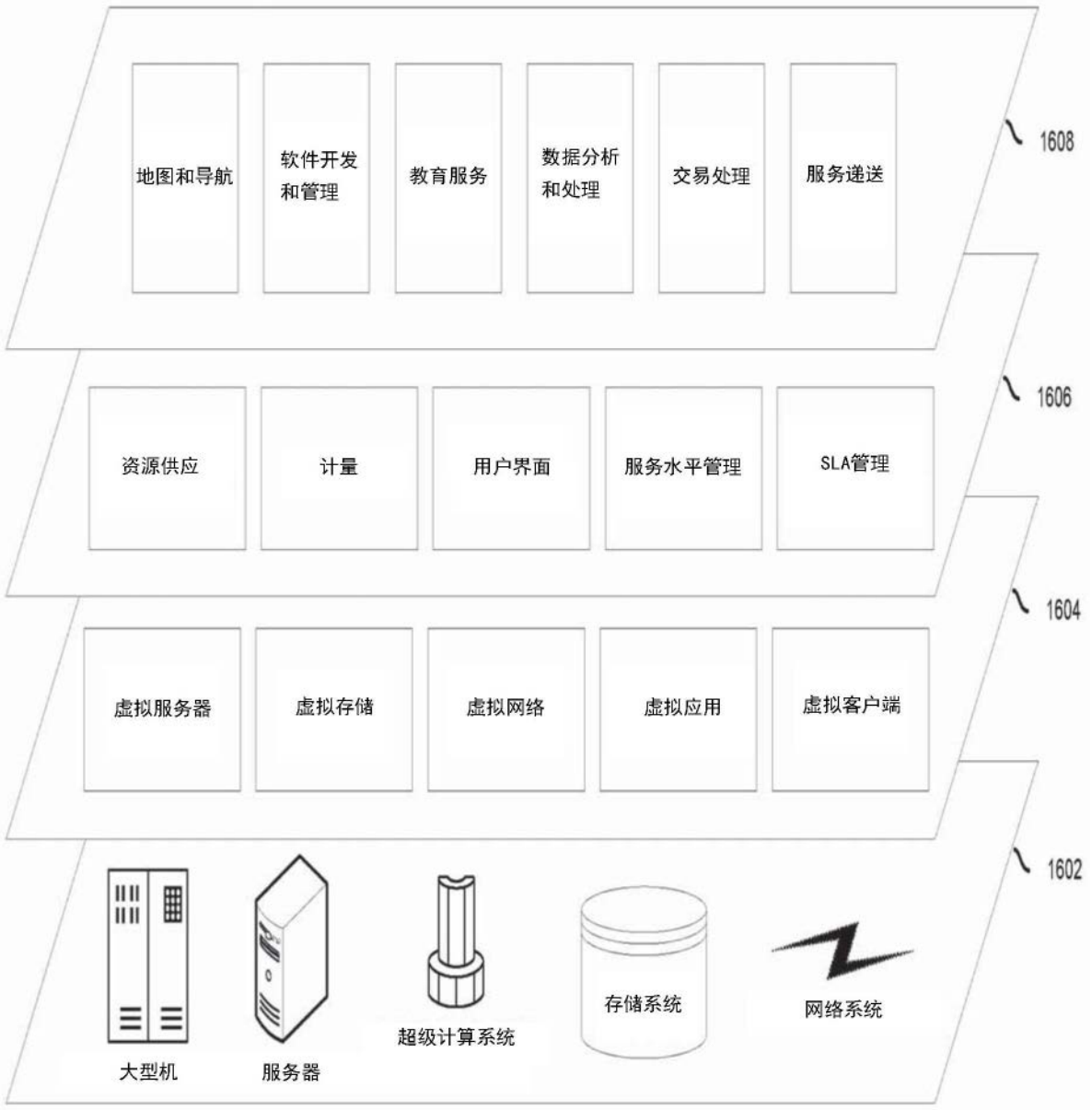

地图和导航
软件开发和管理
教育服务
数据分析和处理
交易处理
服务递送
1608
资源供应
计量
用户界面
服务水平管理
SLA管理
1606
虚拟服务器
虚拟存储
虚拟网络
虚拟应用
虚拟客户端
1604
大型机
服务器
超级计算系统
存储系统
网络系统
1602

图16

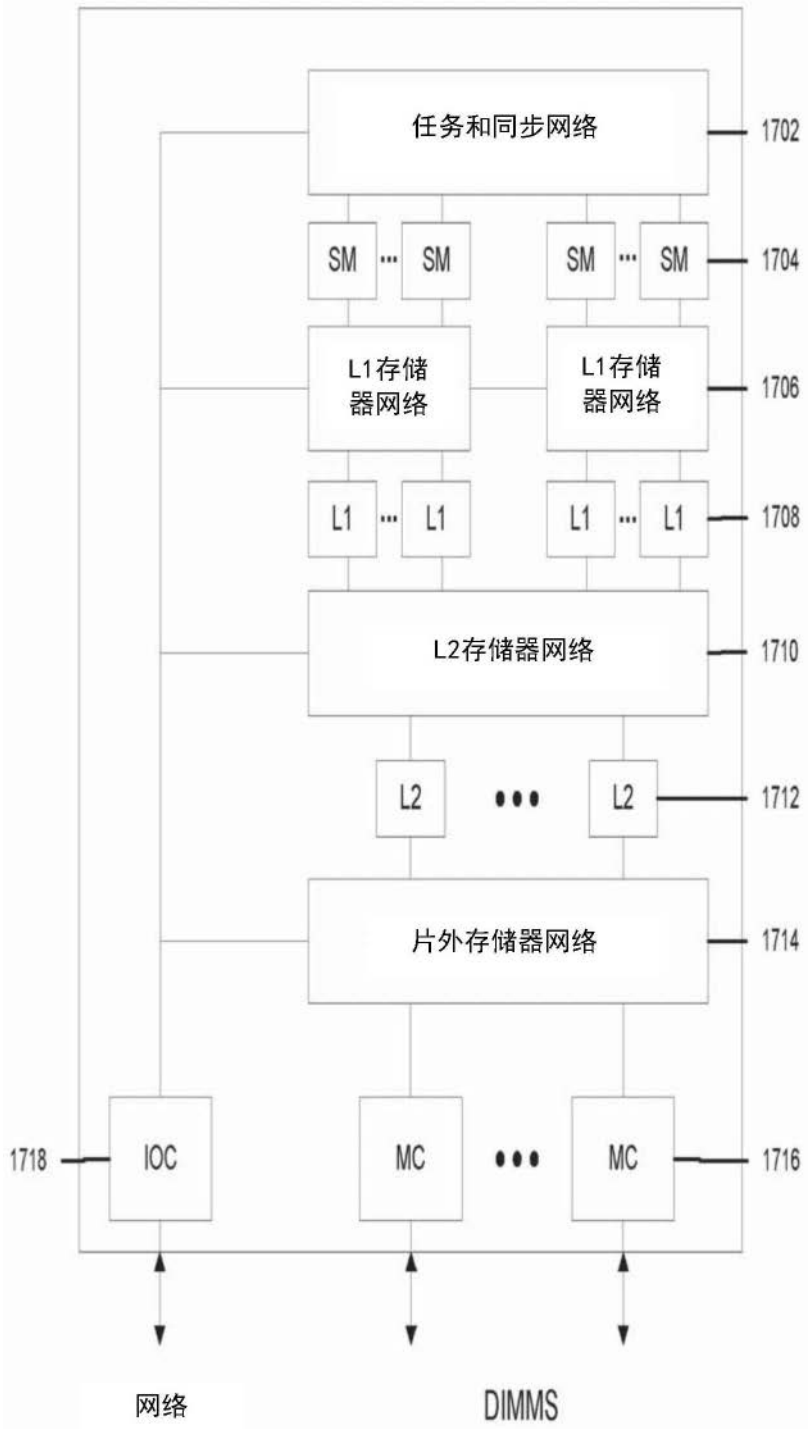

任务和同步网络
1702
SM
SM
SM
SM
1704
L1存储器网络
L1存储器网络
1706
L1
L1
L1
L1
1708
L2存储器网络
1710
L2
L2
1712
片外存储器网络
1714
1718
IOC
MC
MC
1716
网络
DIMMS

图17

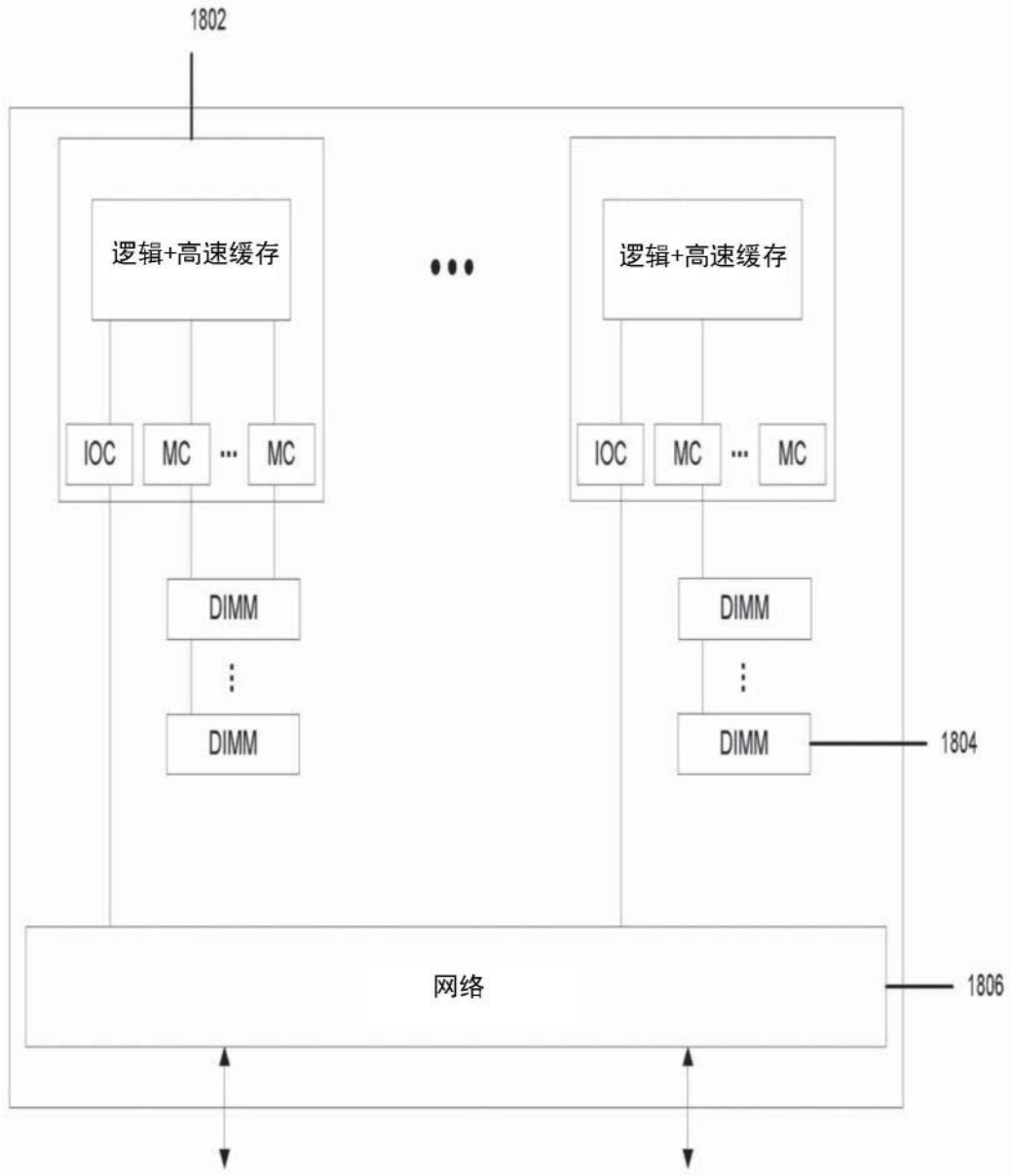

1802
逻辑+高速缓存
逻辑+高速缓存
IOC
MC
MC
IOC
MC
MC
DIMM
DIMM
DIMM
DIMM
1804
网络
1806

图18

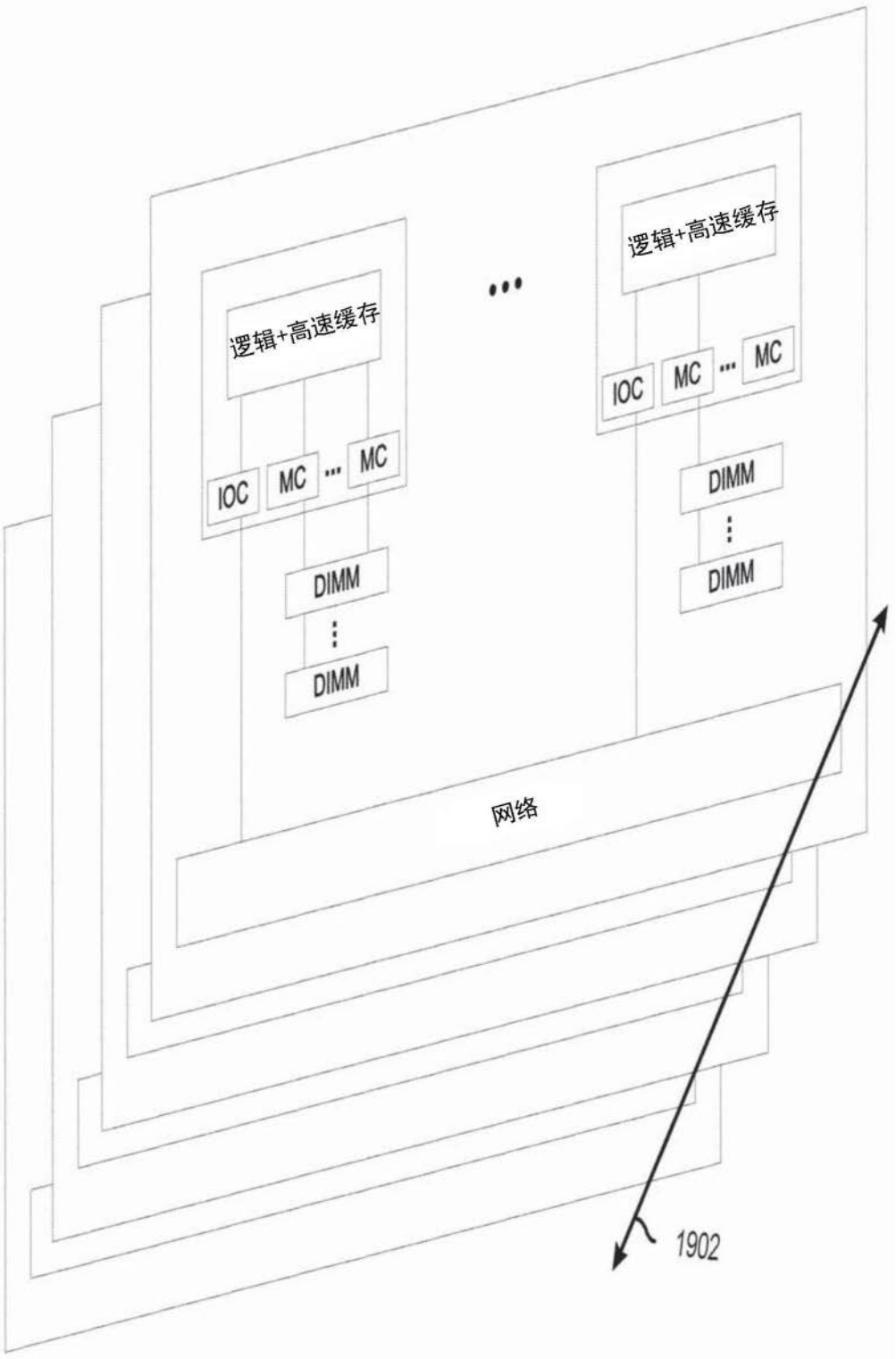


图19

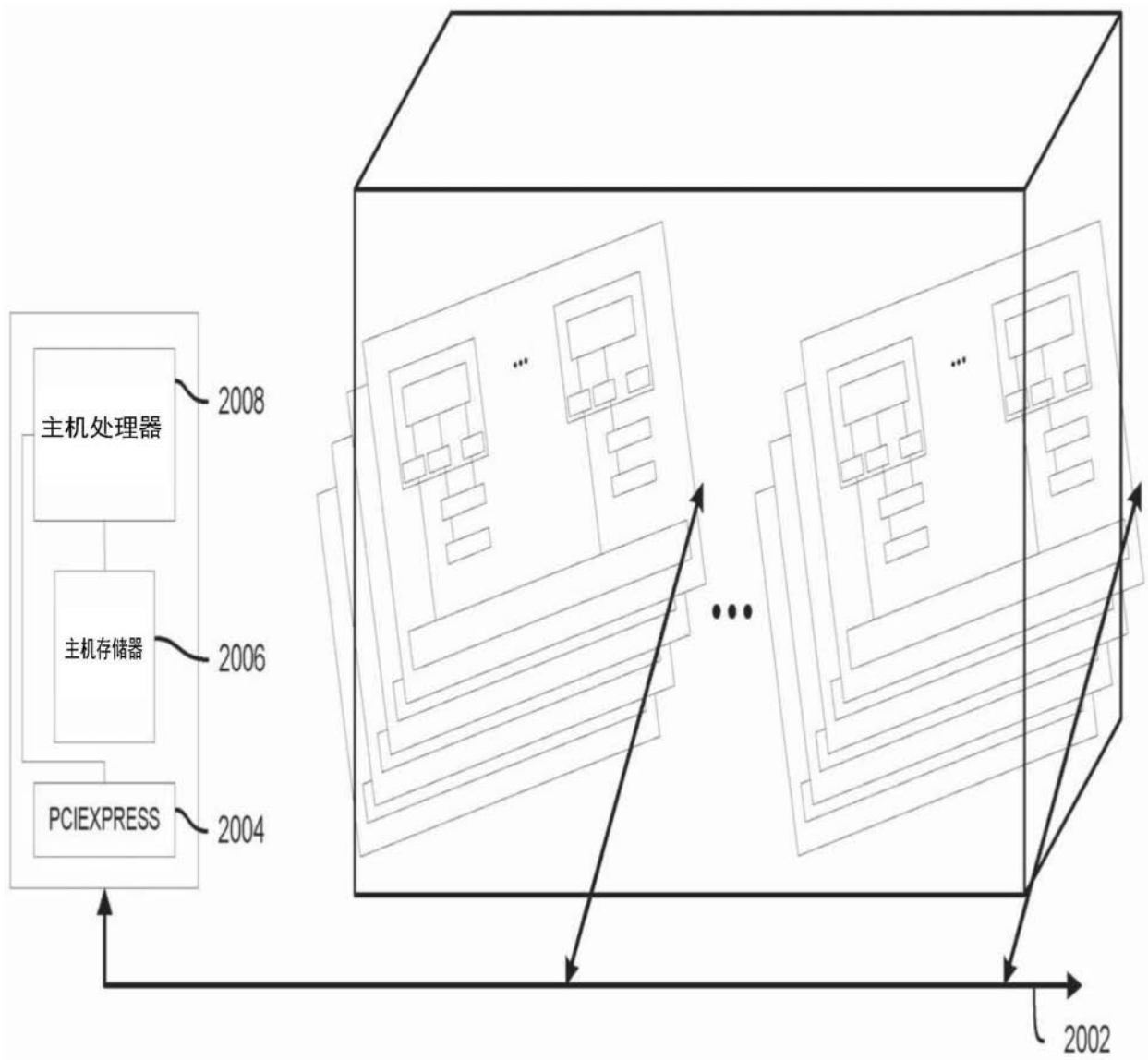


图20

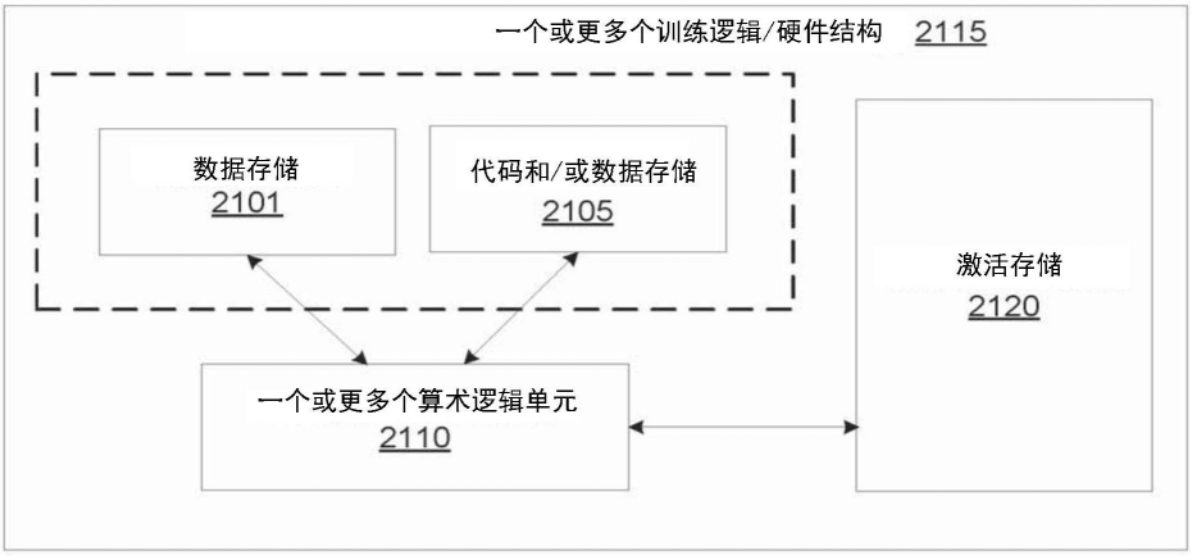

一个或更多个训练逻辑/硬件结构 2115
数据存储
2101
代码和/或数据存储
2105
激活存储
2120
一个或更多个算术逻辑单元
2110

图21A

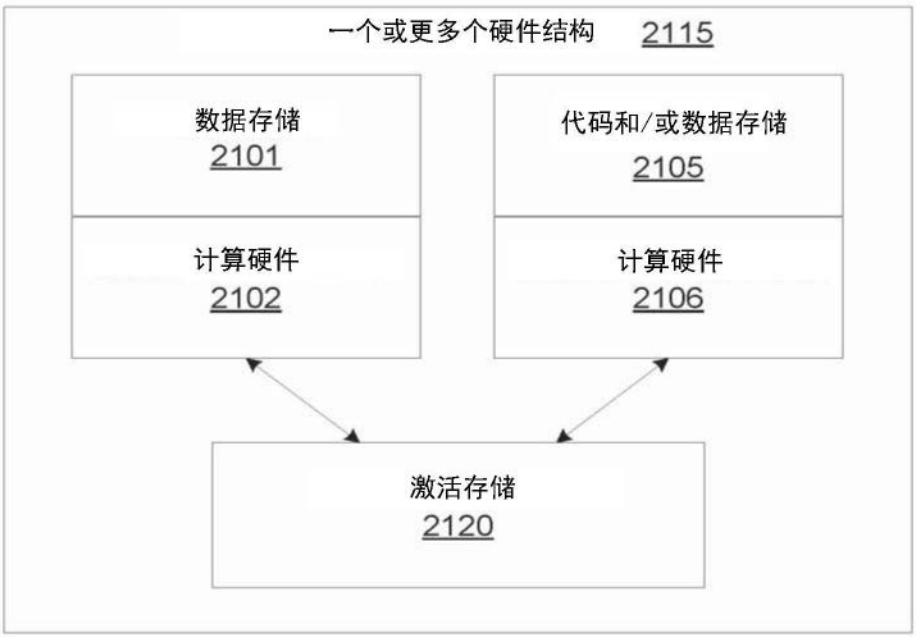

一个或更多个硬件结构 2115
数据存储
2101
代码和/或数据存储
2105
计算硬件
2102
计算硬件
2106
激活存储
2120

图21B

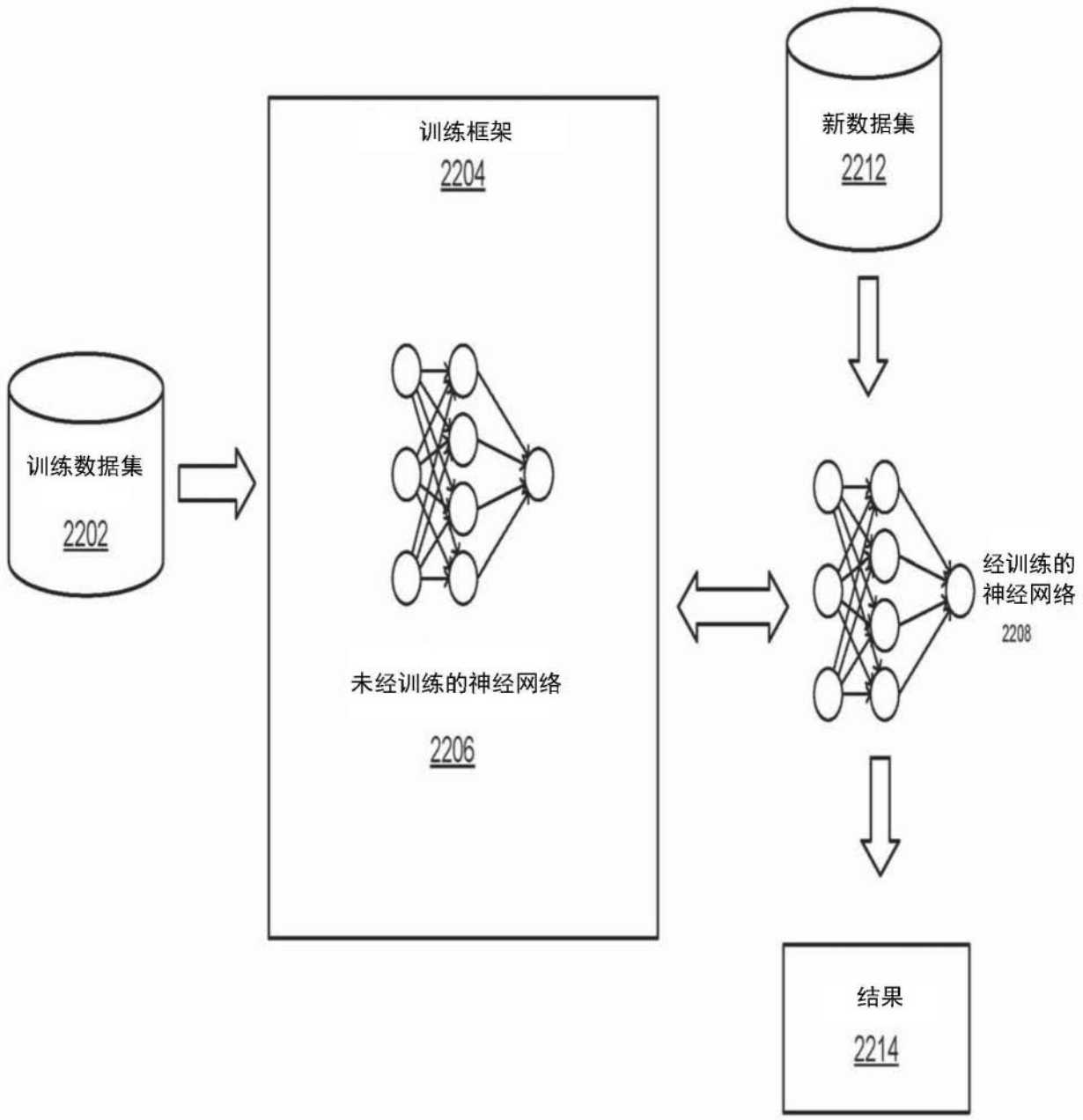

训练框架
2204
新数据集
2212
训练数据集
2202
未经训练的神经网络
2206
经训练的
神经网络
2208
结果
2214

图22

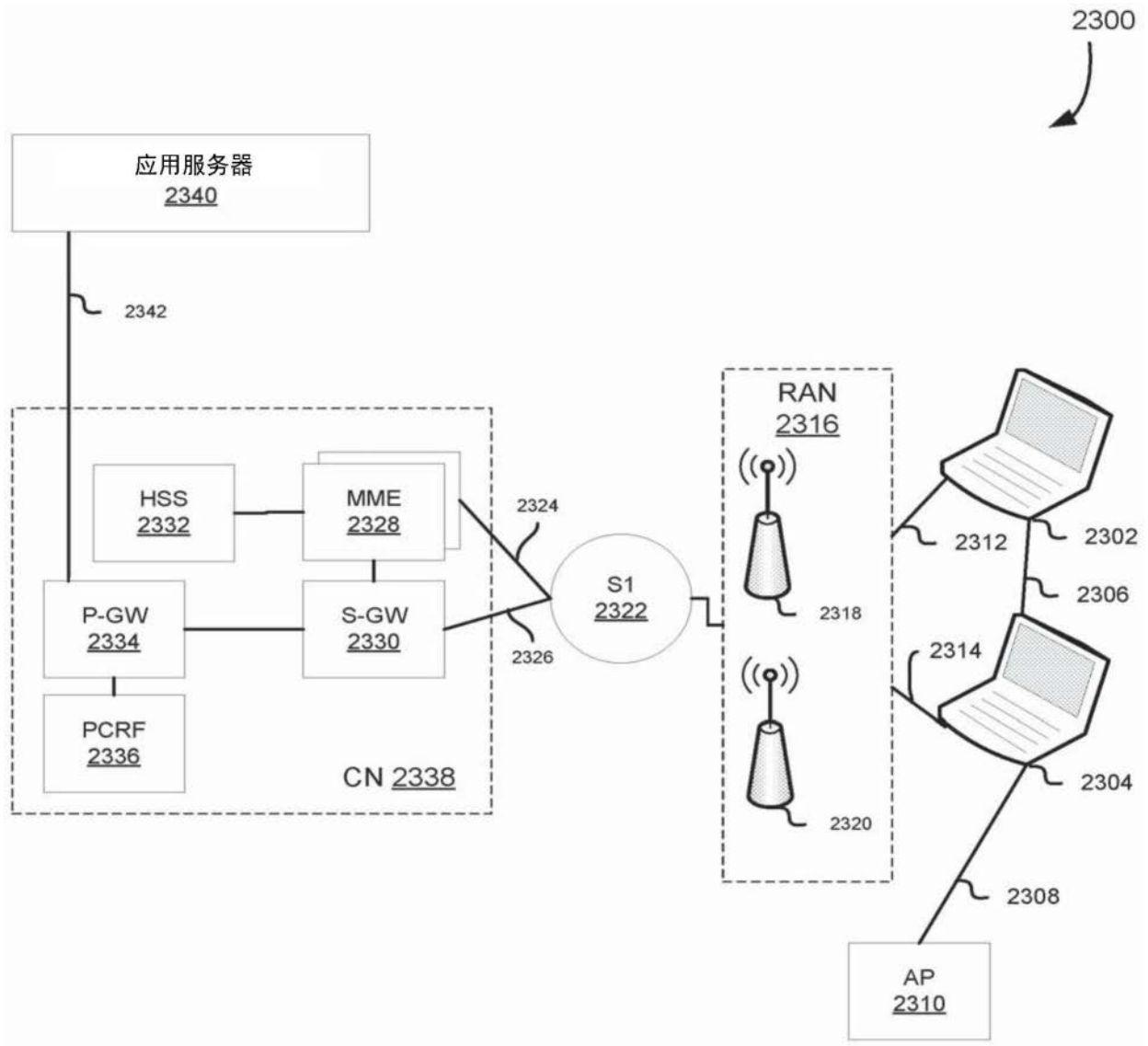

2300
应用服务器
2340
2342
HSS
2332
MME
2328
2324
P-GW
2334
S-GW
2330
2326
PCRF
2336
CN 2338
S1
2322
RAN
2316
2318
2320
2312
2302
2306
2314
2304
2308
AP
2310

图23

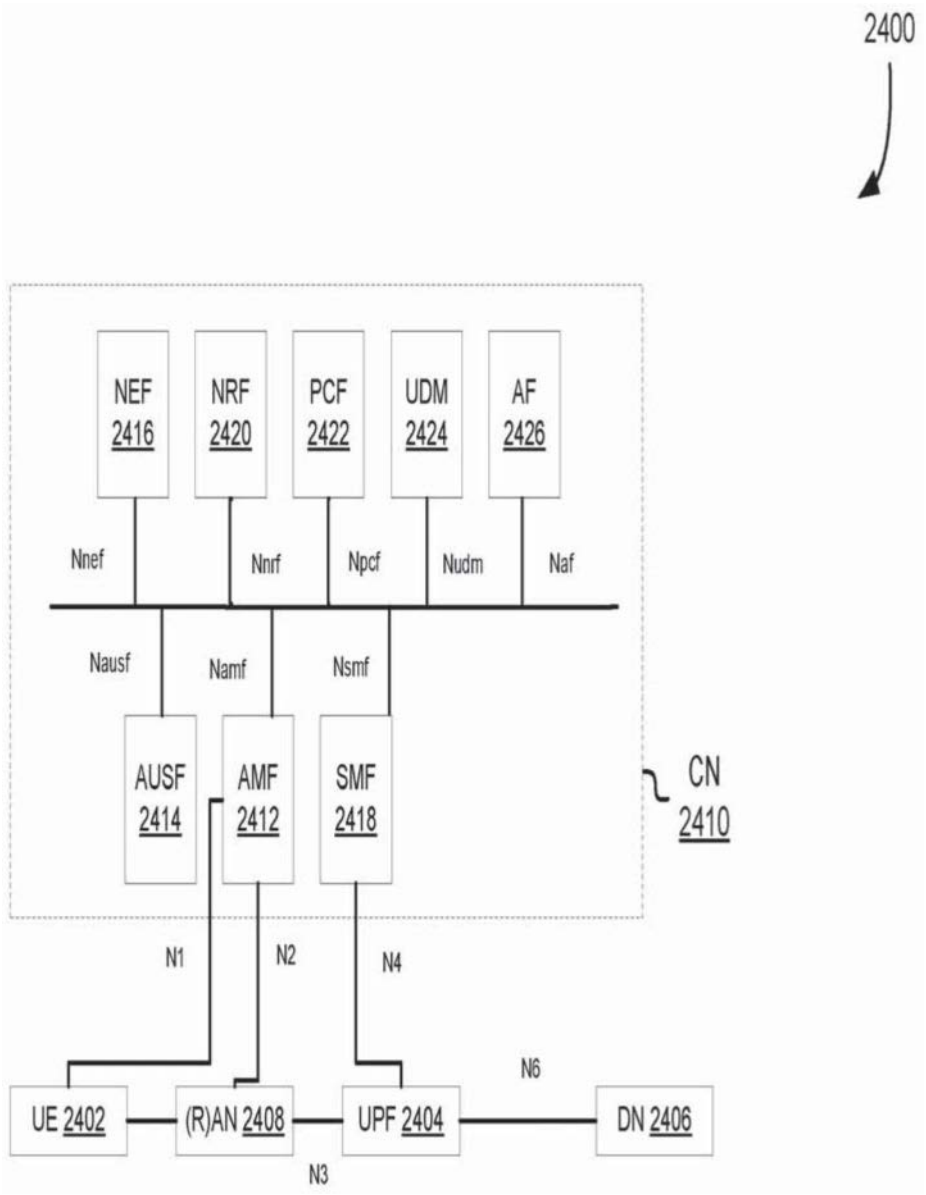

2400
NEF 2416
NRF 2420
PCF 2422
UDM 2424
AF 2426
Nnef
Nnrf
Npcf
Nudm
Naf
Nausf
Namf
Nsmf
AUSF 2414
AMF 2412
SMF 2418
CN 2410
N1
N2
N4
N6
UE 2402
(R)AN 2408
UPF 2404
DN 2406
N3

图24

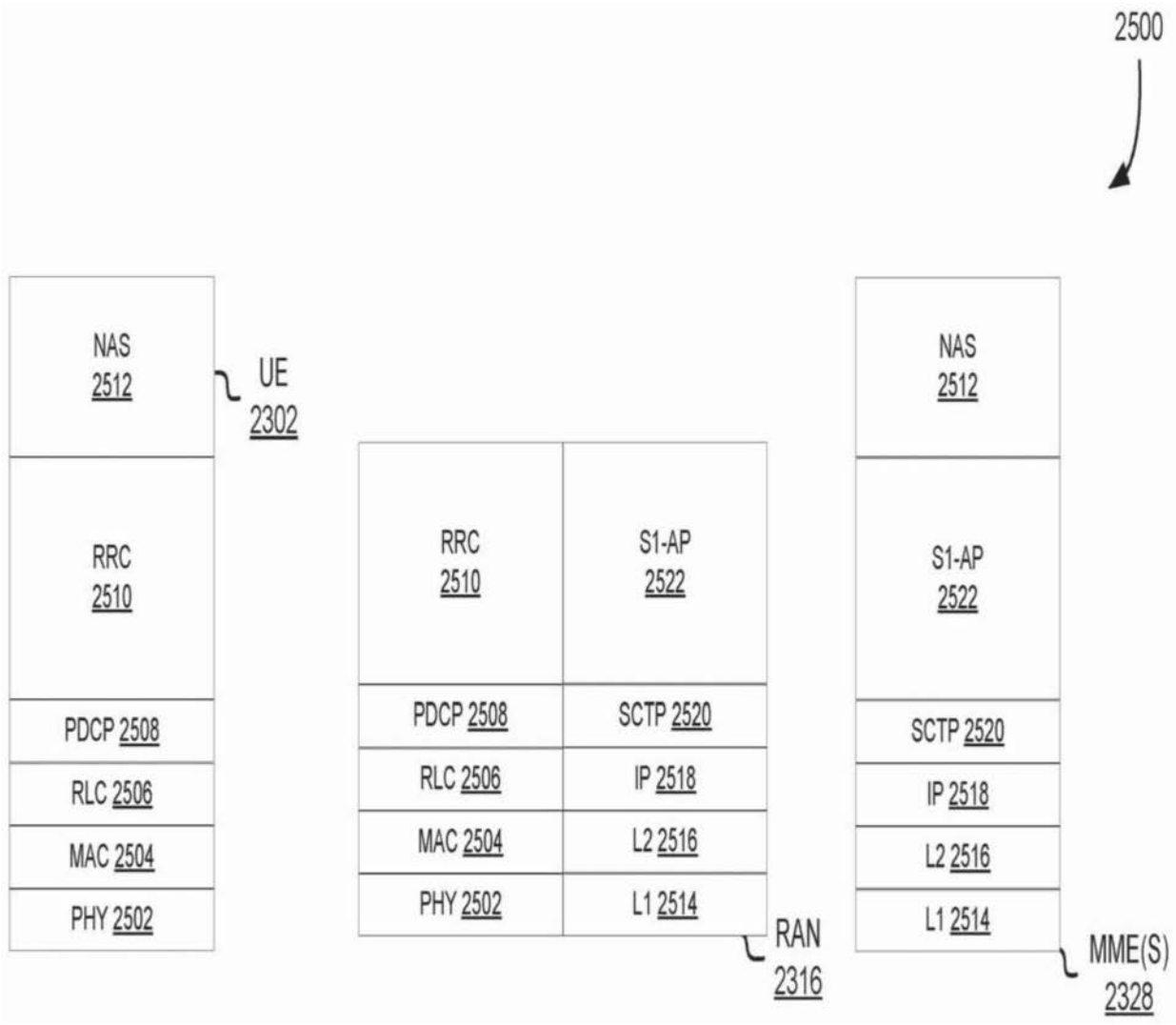

2500
NAS 2512
UE 2302
RRC 2510
PDCP 2508
RLC 2506
MAC 2504
PHY 2502
RRC 2510
S1-AP 2522
PDCP 2508
SCTP 2520
RLC 2506
IP 2518
MAC 2504
L2 2516
PHY 2502
L1 2514
RAN 2316
NAS 2512
S1-AP 2522
SCTP 2520
IP 2518
L2 2516
L1 2514
MME(S) 2328

图25

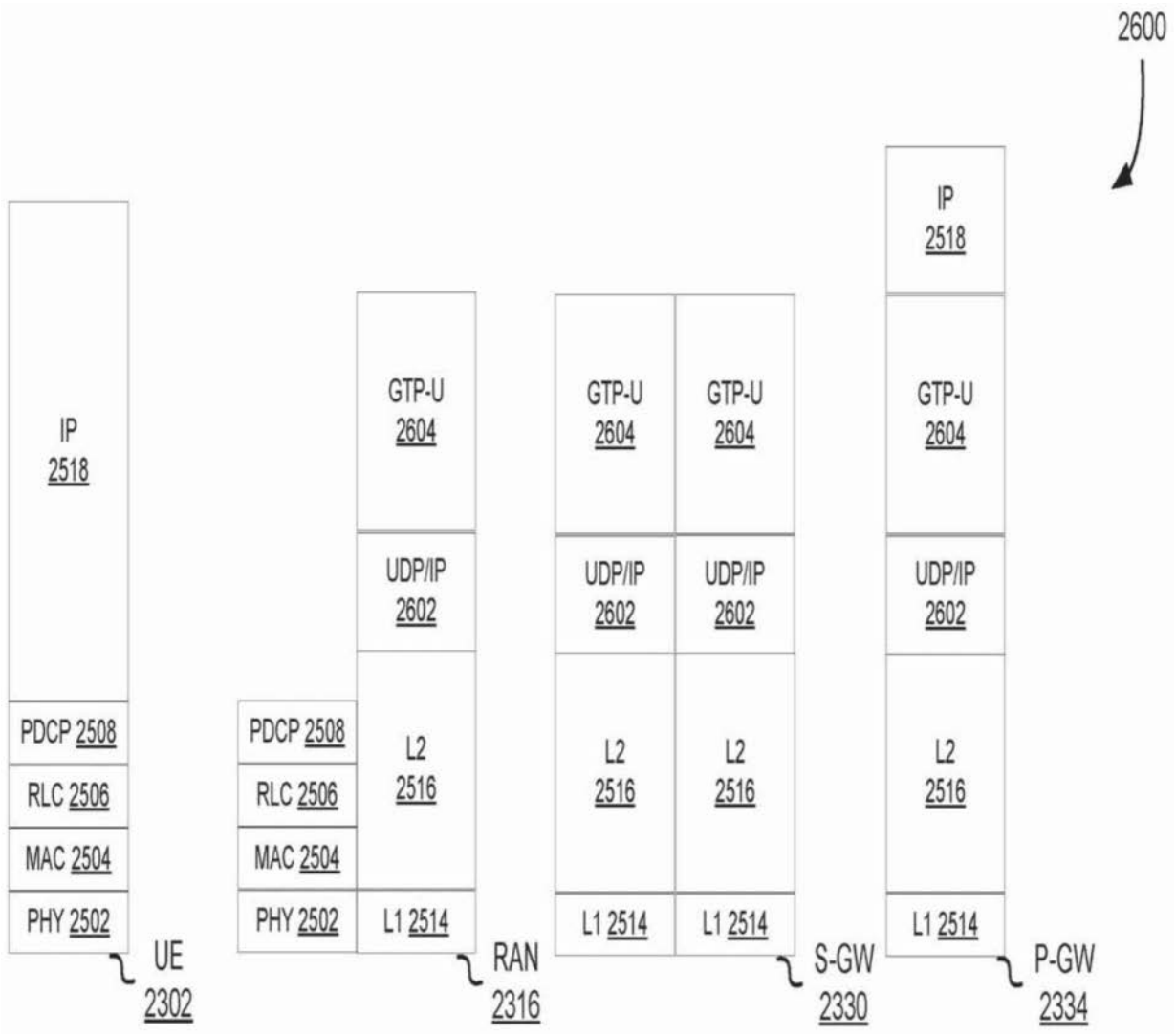

2600
IP 2518
PDCP 2508
RLC 2506
MAC 2504
PHY 2502
UE 2302
GTP-U 2604
UDP/IP 2602
L2 2516
L1 2514
RAN 2316
S-GW 2330
P-GW 2334

图26

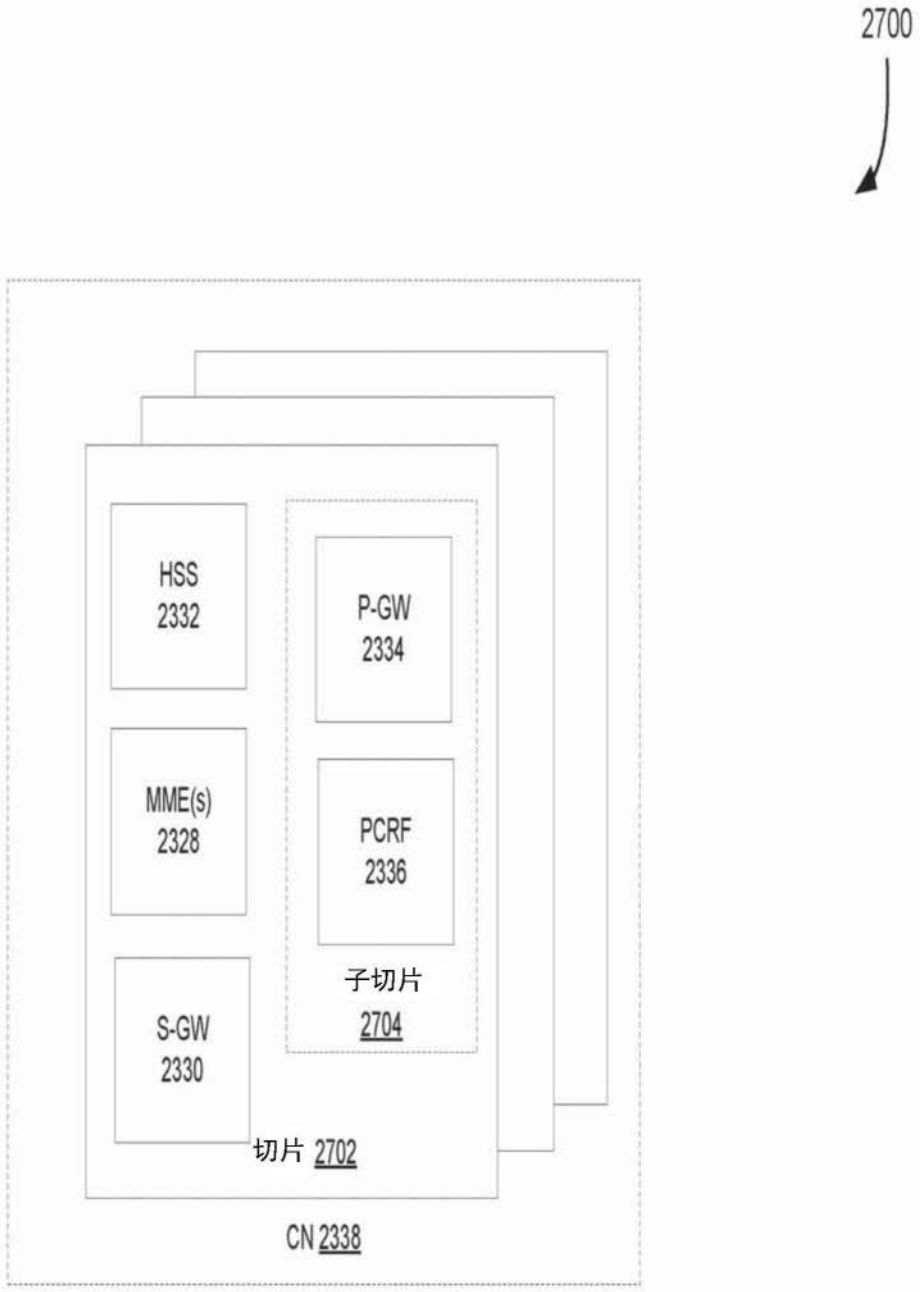

2700
HSS
2332
P-GW
2334
MME(s)
2328
PCRF
2336
子切片
2704
S-GW
2330
切片 2702
CN 2338

图27

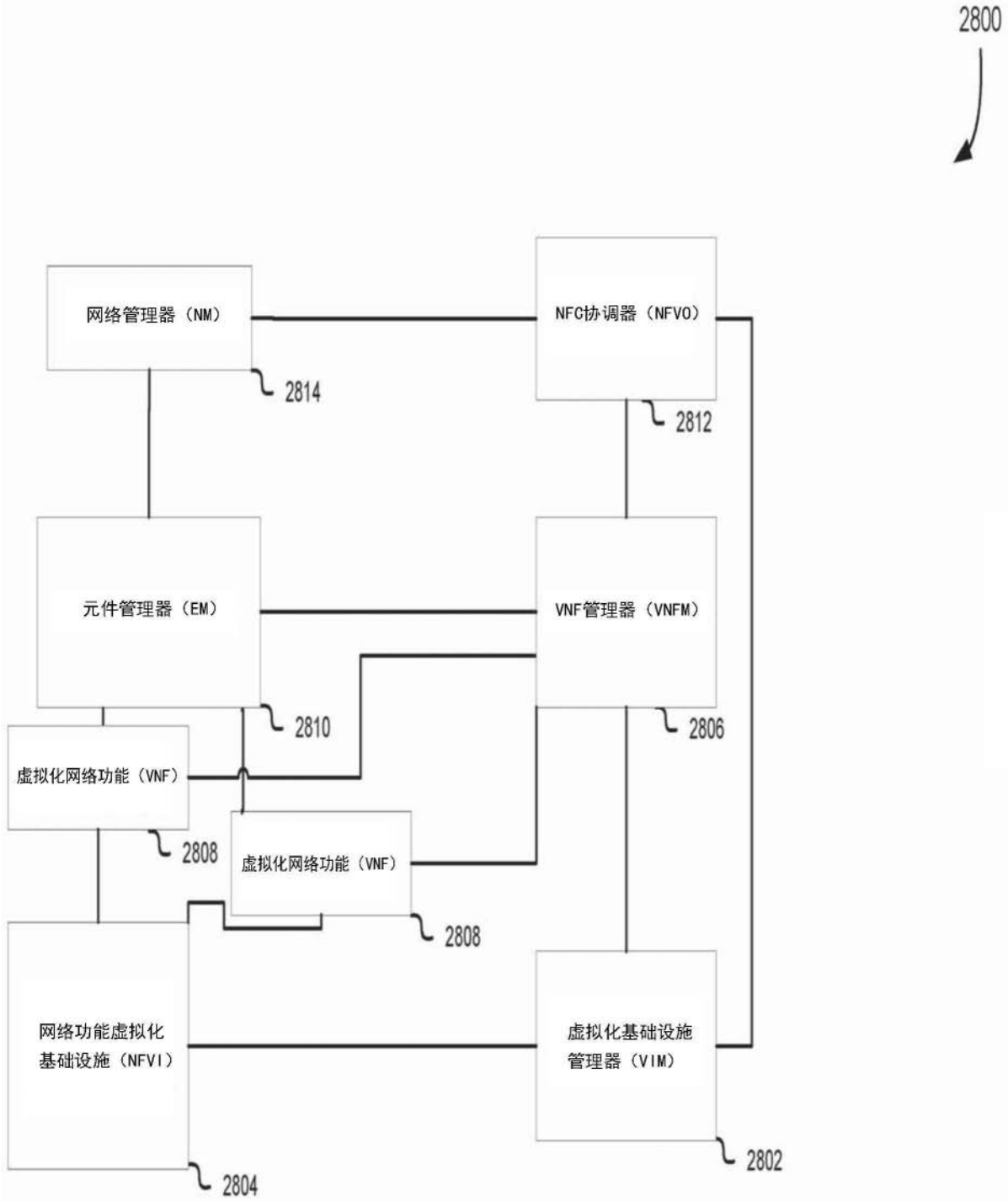

2800
网络管理器（NM）
2814
NFC协调器（NFVO）
2812
元件管理器（EM）
2810
VNF管理器（VNFM）
2806
虚拟化网络功能（VNF）
2808
虚拟化网络功能（VNF）
2808
网络功能虚拟化
基础设施（NFVI）
2804
虚拟化基础设施
管理器（VIM）
2802

图28

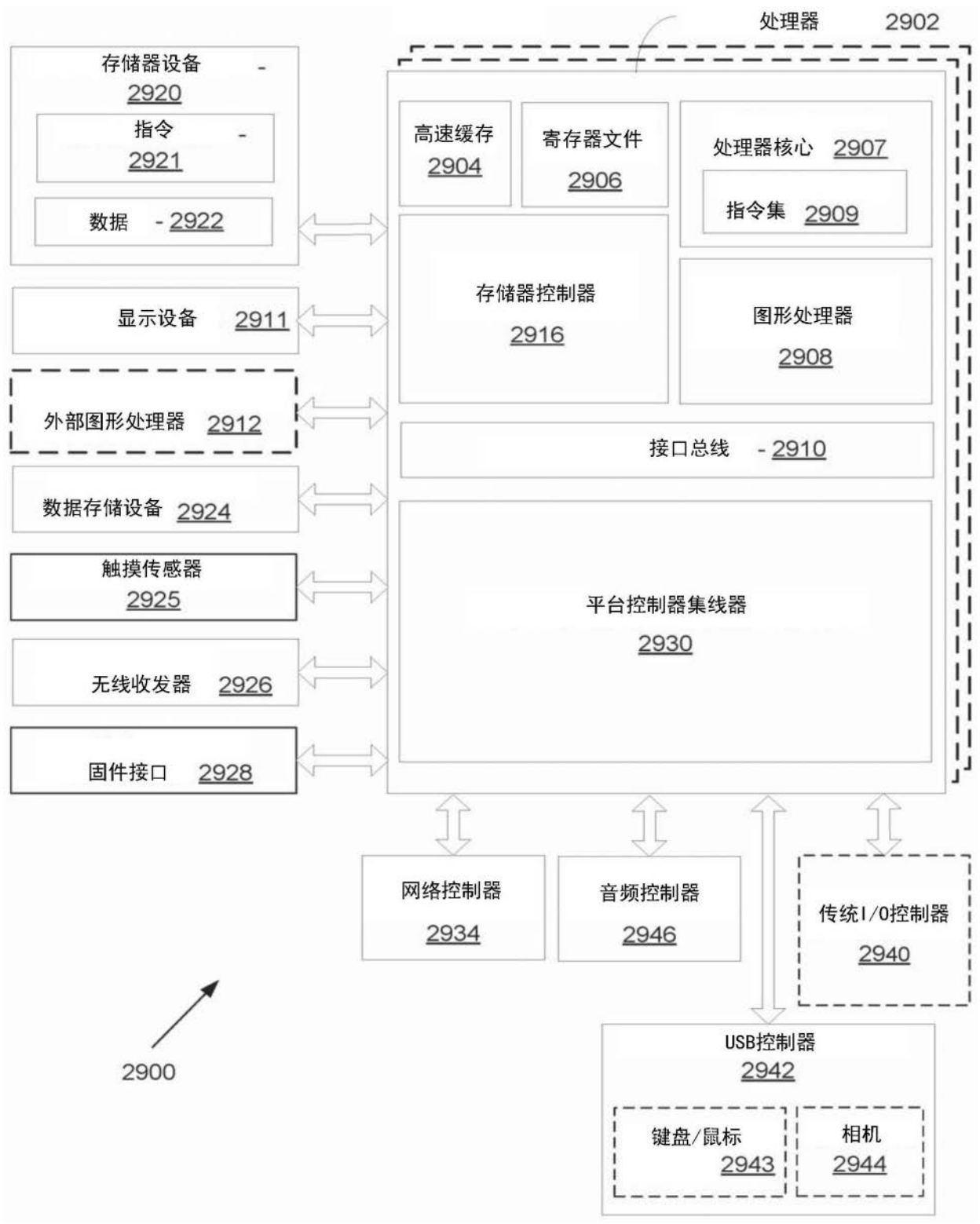

处理器 2902
存储器设备 2920
指令 2921
数据 2922
显示设备 2911
外部图形处理器 2912
数据存储设备 2924
触摸传感器 2925
无线收发器 2926
固件接口 2928
高速缓存 2904
寄存器文件 2906
处理器核心 2907
指令集 2909
存储器控制器 2916
图形处理器 2908
接口总线 2910
平台控制器集线器 2930
网络控制器 2934
音频控制器 2946
传统I/O控制器 2940
USB控制器 2942
键盘/鼠标 2943
相机 2944
2900

图29


处理器 3002
高速缓存 3004
寄存器文件 3006
执行单元 3008
封装指令集 3009
处理器总线 3010
图形/视频卡 3012
3014
存储器控制器集线器 3016
3018
存储器 3020
指令 3019
数据 3021
3022
数据存储器 3024
传统I/O控制器 3023
用户输入接口 3025
无线收发器 3026
I/O控制器集线器 3030
串行扩展端口 3027
固件集线器 3028
音频控制器 3029
网络控制器 3034
3000

图30

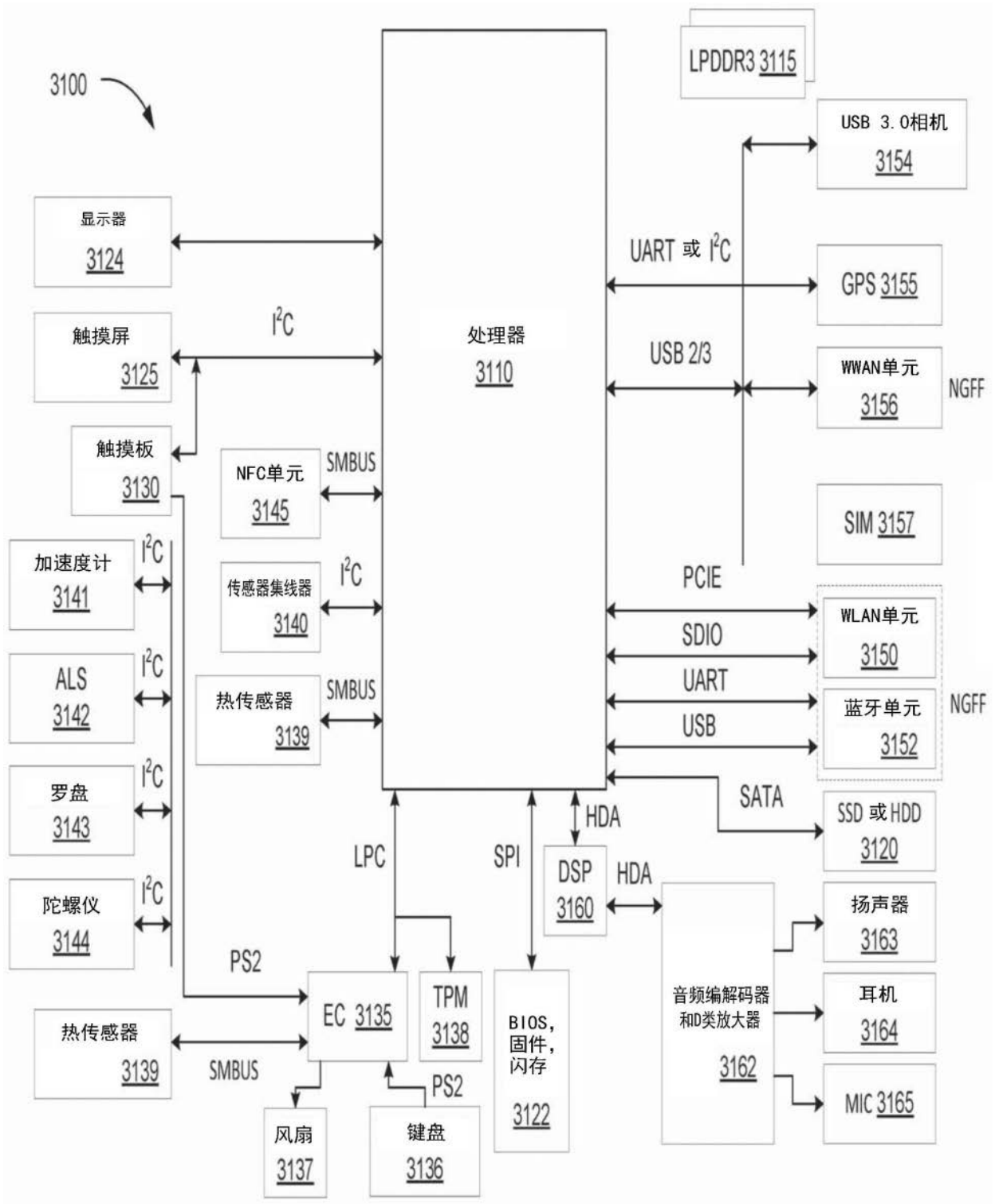

3100
处理器
3110
LPDDR3 3115
USB 3.0相机
3154
显示器
3124
UART 或 I²C
GPS 3155
触摸屏
3125
I²C
USB 2/3
WWAN单元
3156
NGFF
触摸板
3130
NFC单元
3145
SMBUS
SIM 3157
加速度计
3141
I²C
传感器集线器
3140
I²C
PCIE
WLAN单元
3150
SDIO
ALS
3142
I²C
热传感器
3139
SMBUS
UART
蓝牙单元
3152
NGFF
USB
罗盘
3143
I²C
SATA
SSD 或HDD
3120
HDA
LPC
SPI
DSP
3160
HDA
陀螺仪
3144
I²C
扬声器
3163
PS2
EC 3135
TPM
3138
BIOS,
固件,
闪存
3122
音频编解码器
和D类放大器
3162
耳机
3164
热传感器
3139
SMBUS
PS2
MIC 3165
风扇
3137
键盘
3136

图31

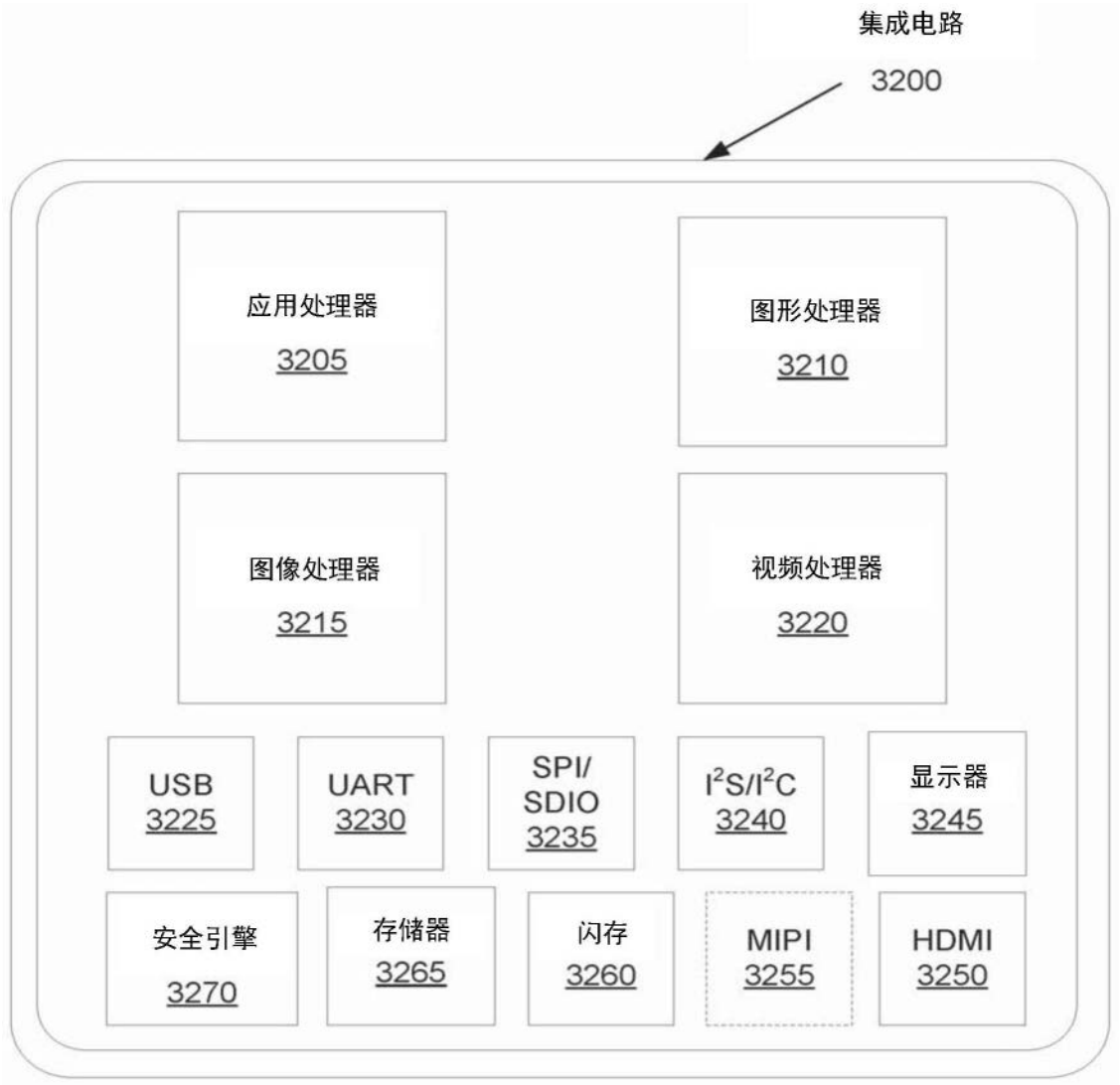

集成电路
3200
应用处理器
3205
图形处理器
3210
图像处理器
3215
视频处理器
3220
USB
3225
UART
3230
SPI/
SDIO
3235
I²S/I²C
3240
显示器
3245
安全引擎
3270
存储器
3265
闪存
3260
MIPI
3255
HDMI
3250

图32

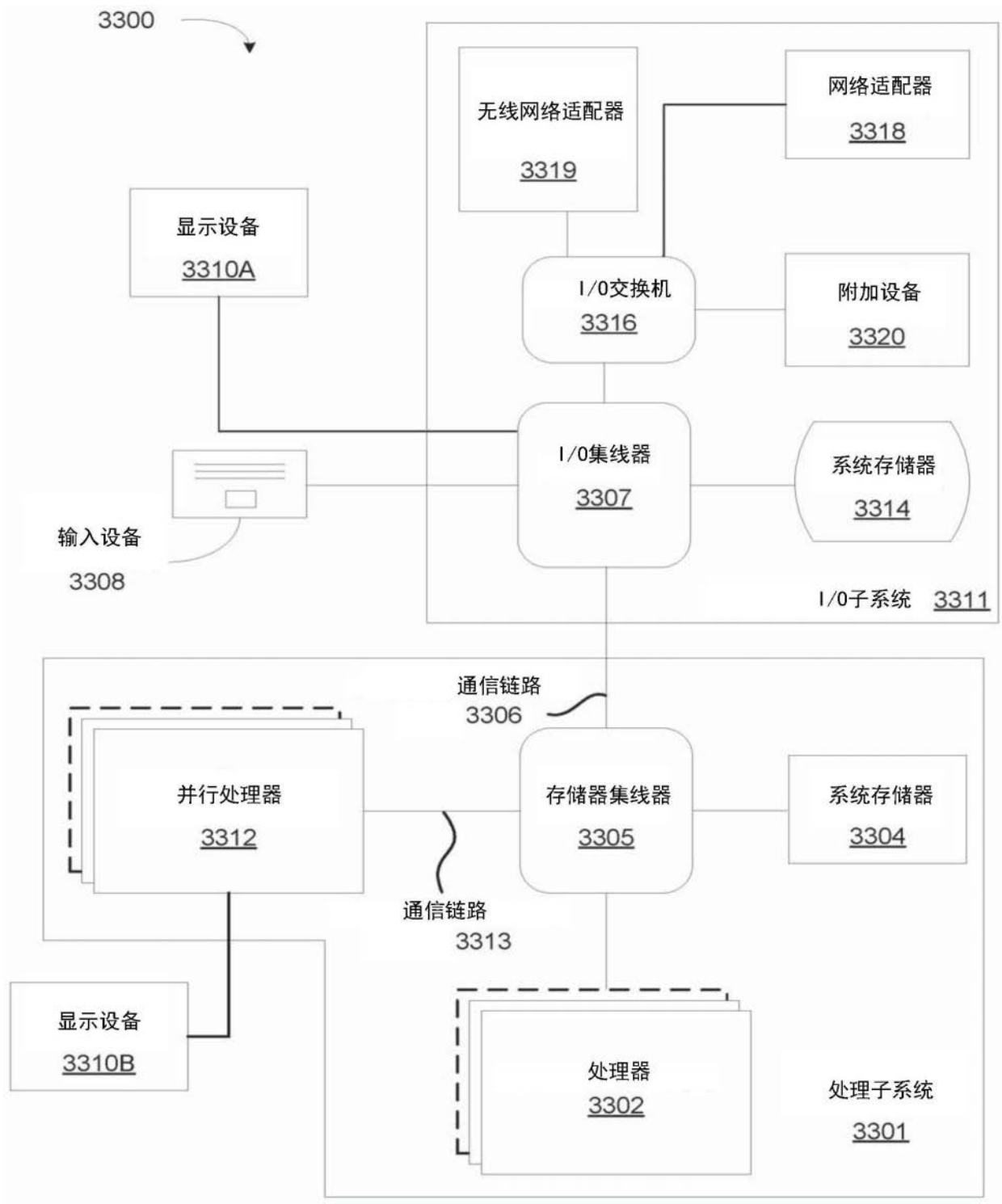

3300
无线网络适配器
3319
网络适配器
3318
显示设备
3310A
I/0交换机
3316
附加设备
3320
I/0集线器
3307
系统存储器
3314
输入设备
3308
I/0子系统 3311
通信链路
3306
并行处理器
3312
存储器集线器
3305
系统存储器
3304
通信链路
3313
显示设备
3310B
处理器
3302
处理子系统
3301

图33

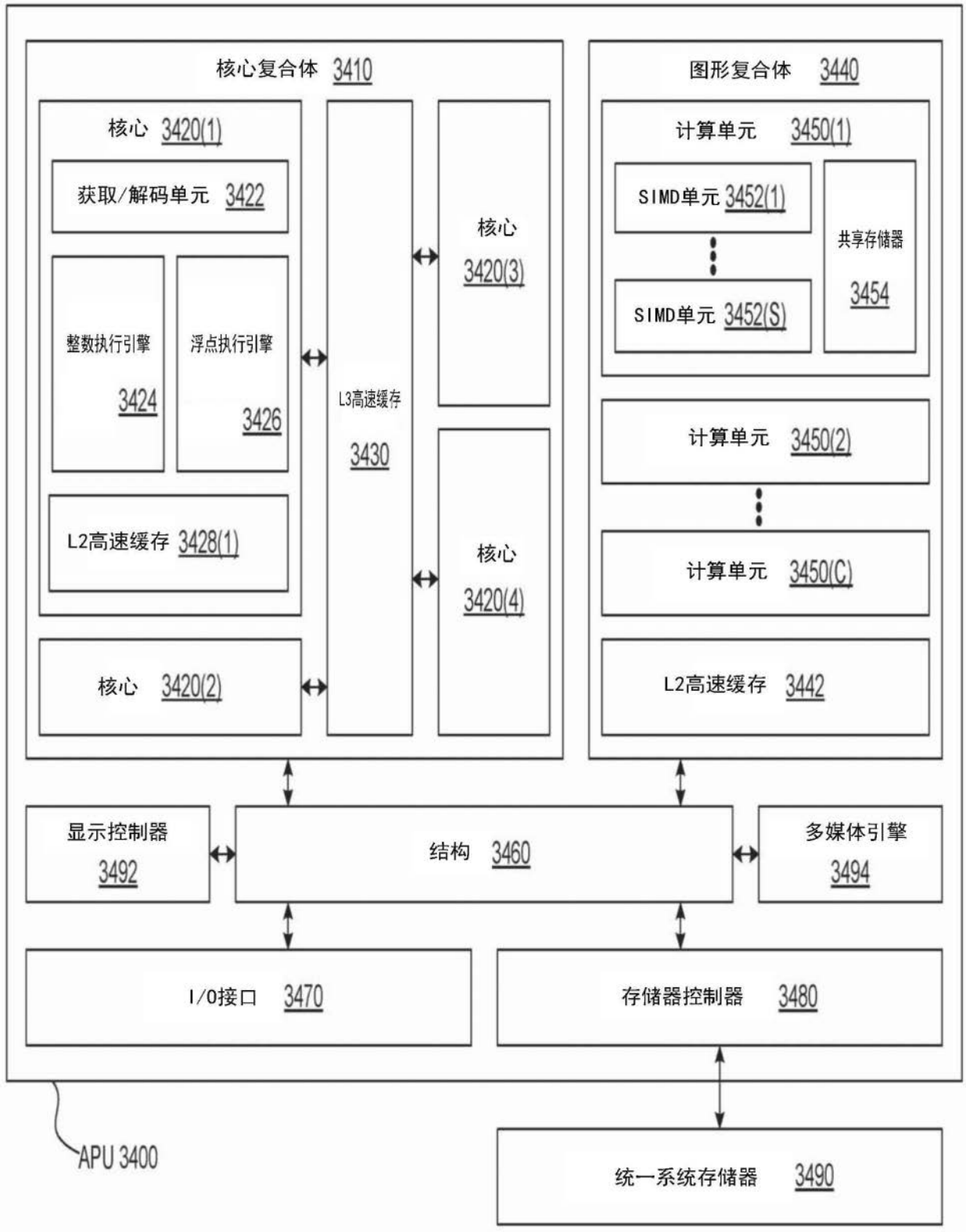

核心复合体 3410
核心 3420(1)
获取/解码单元 3422
整数执行引擎 3424
浮点执行引擎 3426
L2高速缓存 3428(1)
核心 3420(2)
L3高速缓存 3430
核心 3420(3)
核心 3420(4)
图形复合体 3440
计算单元 3450(1)
SIMD单元 3452(1)
SIMD单元 3452(S)
共享存储器 3454
计算单元 3450(2)
计算单元 3450(C)
L2高速缓存 3442
显示控制器 3492
结构 3460
多媒体引擎 3494
I/O接口 3470
存储器控制器 3480
APU 3400
统一系统存储器 3490

图34

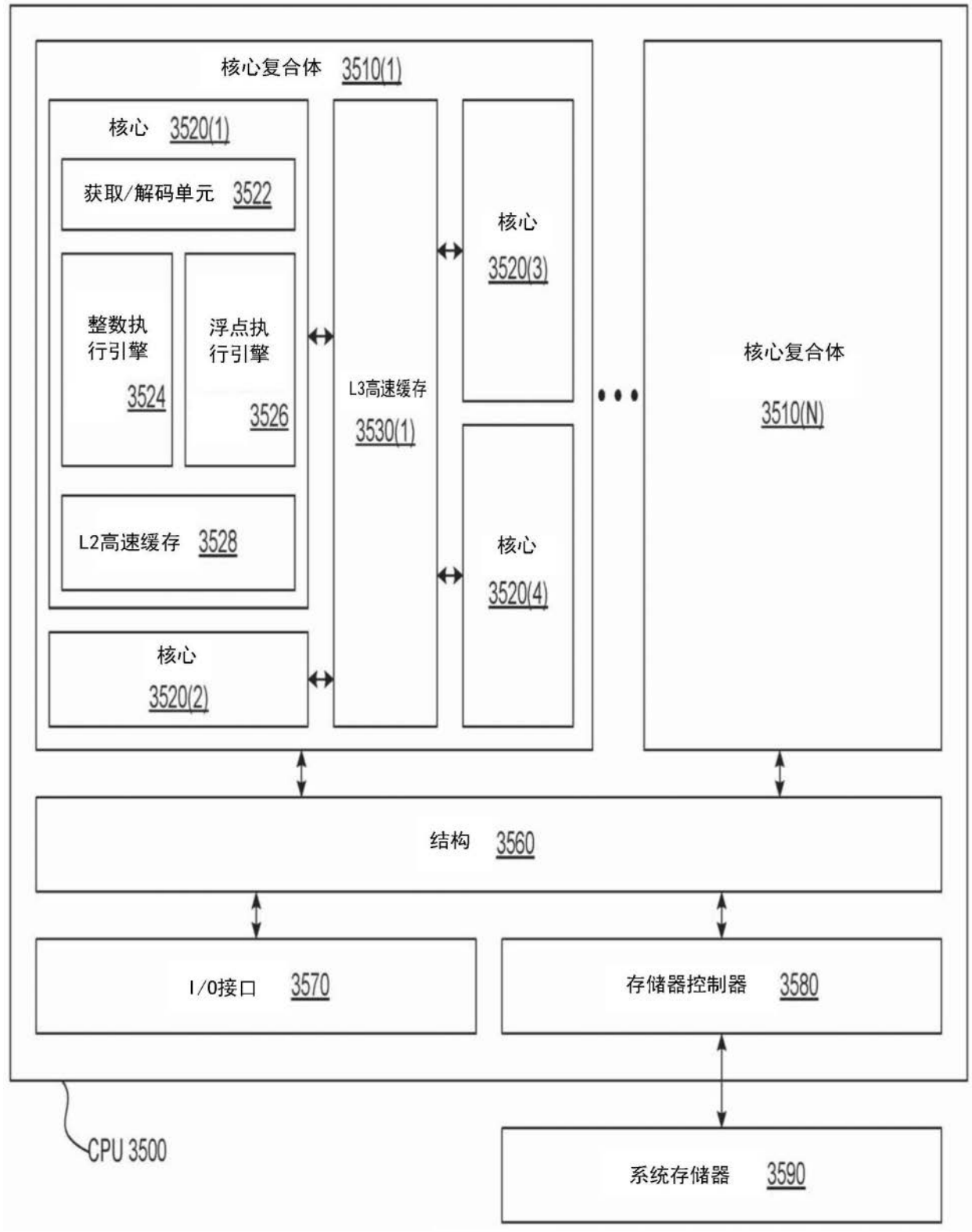

核心复合体 3510(1)
核心 3520(1)
获取/解码单元 3522
整数执行引擎 3524
浮点执行引擎 3526
L2高速缓存 3528
核心 3520(2)
L3高速缓存 3530(1)
核心 3520(3)
核心 3520(4)
核心复合体 3510(N)
结构 3560
I/O接口 3570
存储器控制器 3580
CPU 3500
系统存储器 3590

图35

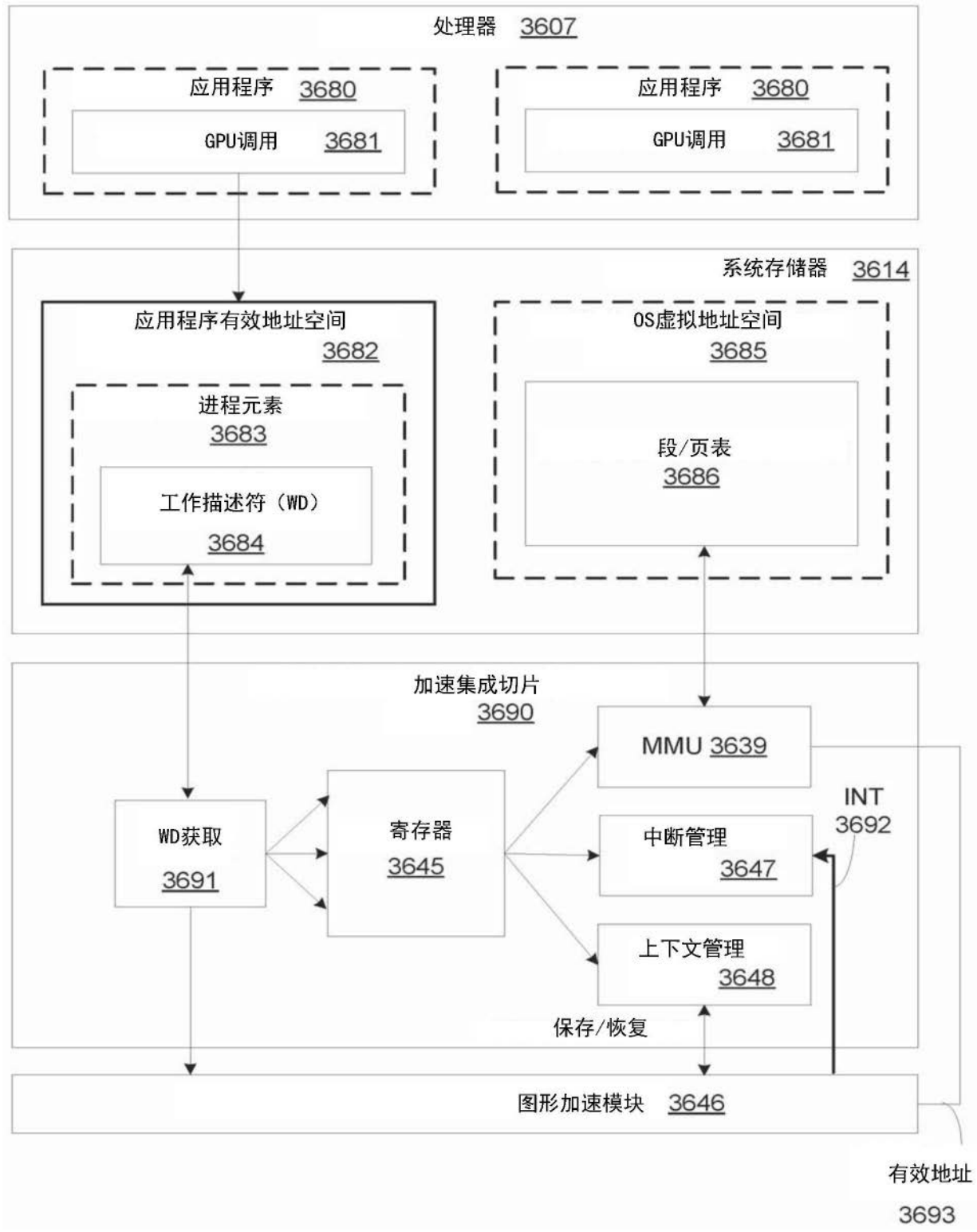

处理器 3607
应用程序 3680
GPU调用 3681
应用程序 3680
GPU调用 3681
系统存储器 3614
应用程序有效地址空间 3682
进程元素 3683
工作描述符（WD） 3684
OS虚拟地址空间 3685
段/页表 3686
加速集成切片 3690
MMU 3639
INT 3692
WD获取 3691
寄存器 3645
中断管理 3647
上下文管理 3648
保存/恢复
图形加速模块 3646
有效地址 3693

图36


图形处理器
3710
顶点处理器
3705
片段处理器
3715A
片段处理器
3715C
片段处理器
3715N-1
片段处理器
3715B
片段处理器
3715D
片段处理器
3715N
MMU
3720A
MMU
3720B
高速缓存
3725A
高速缓存
3725B
互连
3730A
互连
3730B

图37A

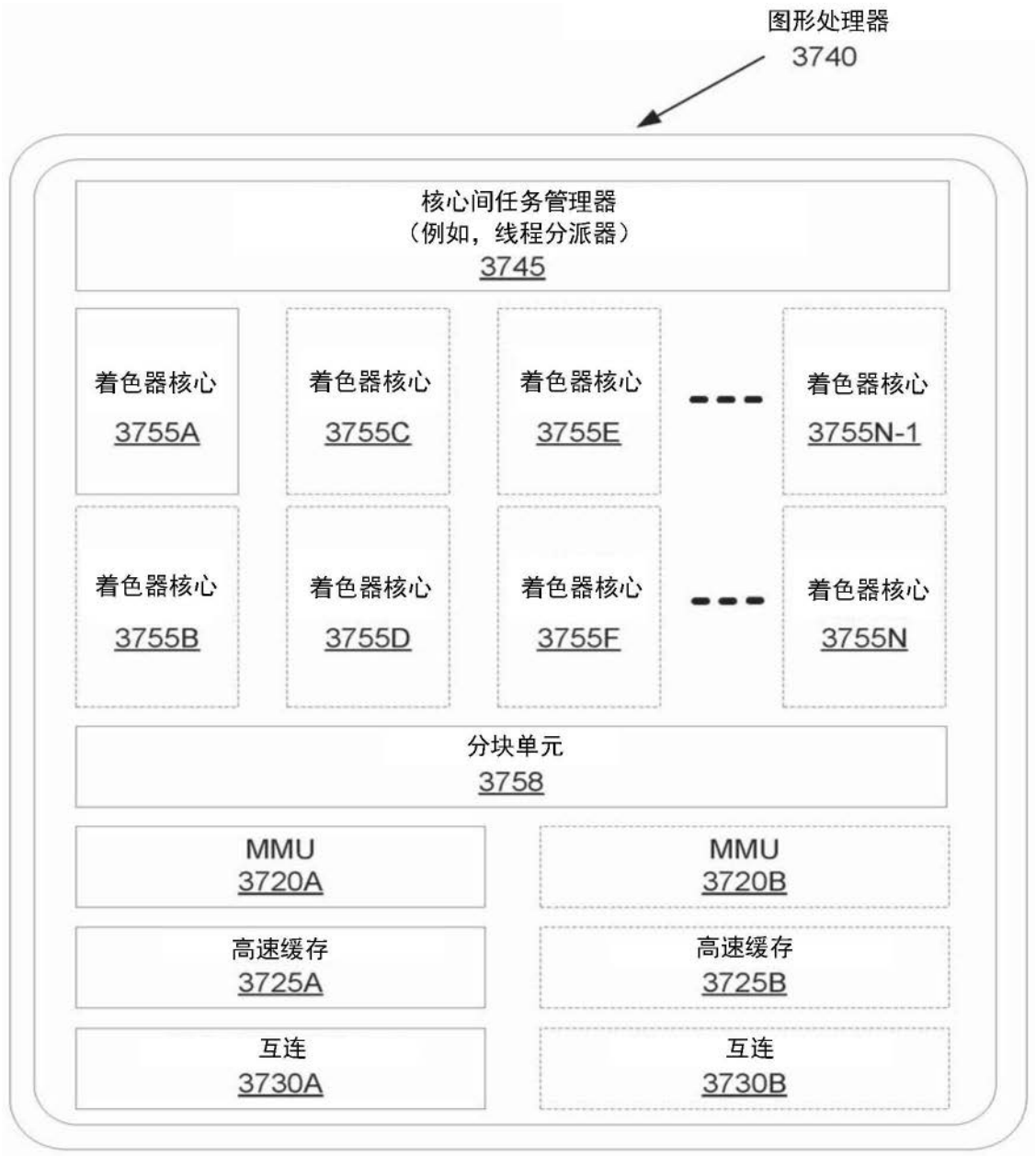

图形处理器
3740
核心间任务管理器
（例如，线程分派器）
3745
着色器核心
3755A
着色器核心
3755C
着色器核心
3755E
着色器核心
3755N-1
着色器核心
3755B
着色器核心
3755D
着色器核心
3755F
着色器核心
3755N
分块单元
3758
MMU
3720A
MMU
3720B
高速缓存
3725A
高速缓存
3725B
互连
3730A
互连
3730B

图37B

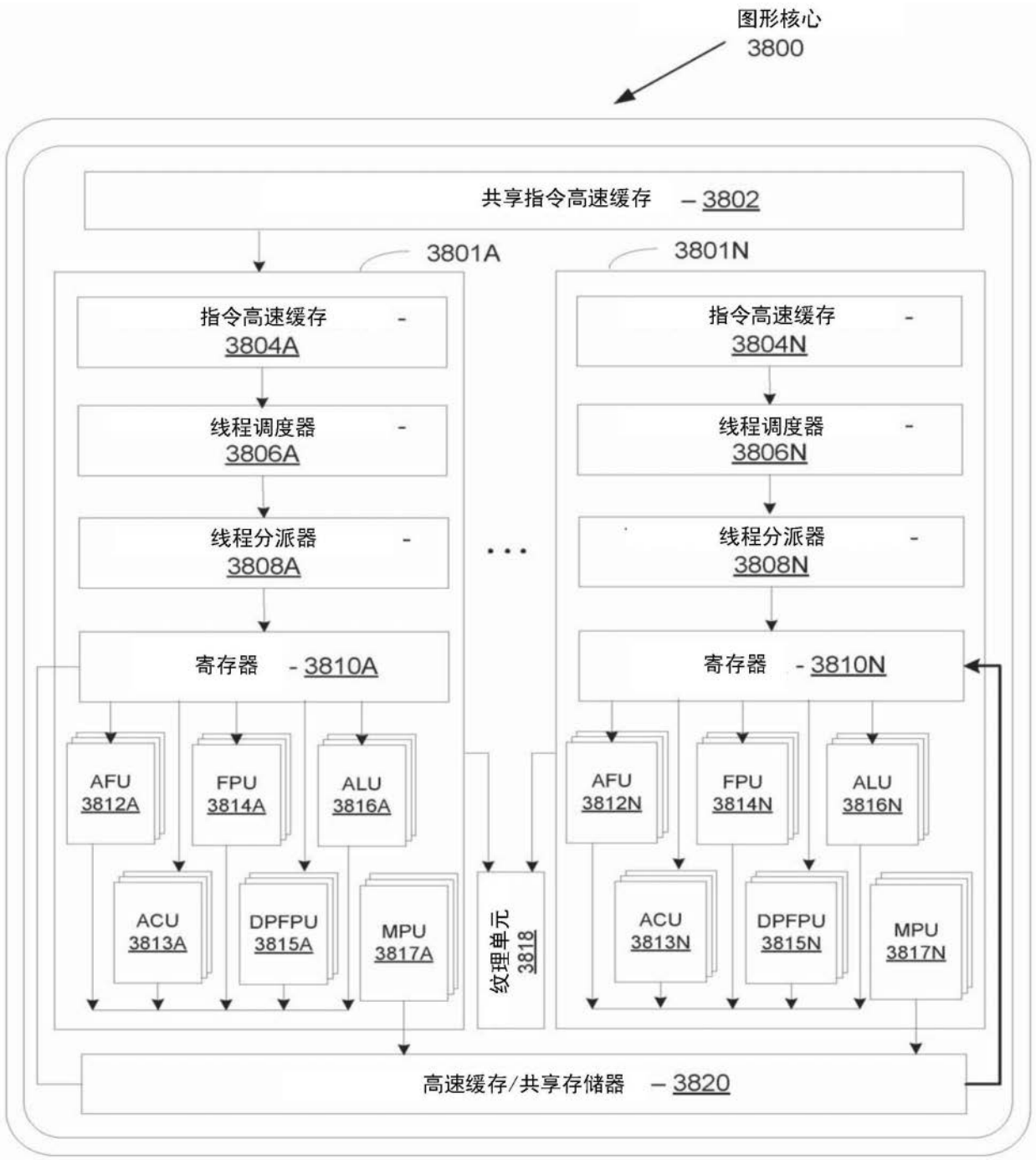

图形核心
3800
共享指令高速缓存 – 3802
3801A
3801N
指令高速缓存 - 3804A
线程调度器 - 3806A
线程分派器 - 3808A
寄存器 - 3810A
AFU 3812A
FPU 3814A
ALU 3816A
ACU 3813A
DPFPU 3815A
MPU 3817A
指令高速缓存 - 3804N
线程调度器 - 3806N
线程分派器 - 3808N
寄存器 - 3810N
AFU 3812N
FPU 3814N
ALU 3816N
ACU 3813N
DPFPU 3815N
MPU 3817N
纹理单元 3818
高速缓存/共享存储器 – 3820

图38A

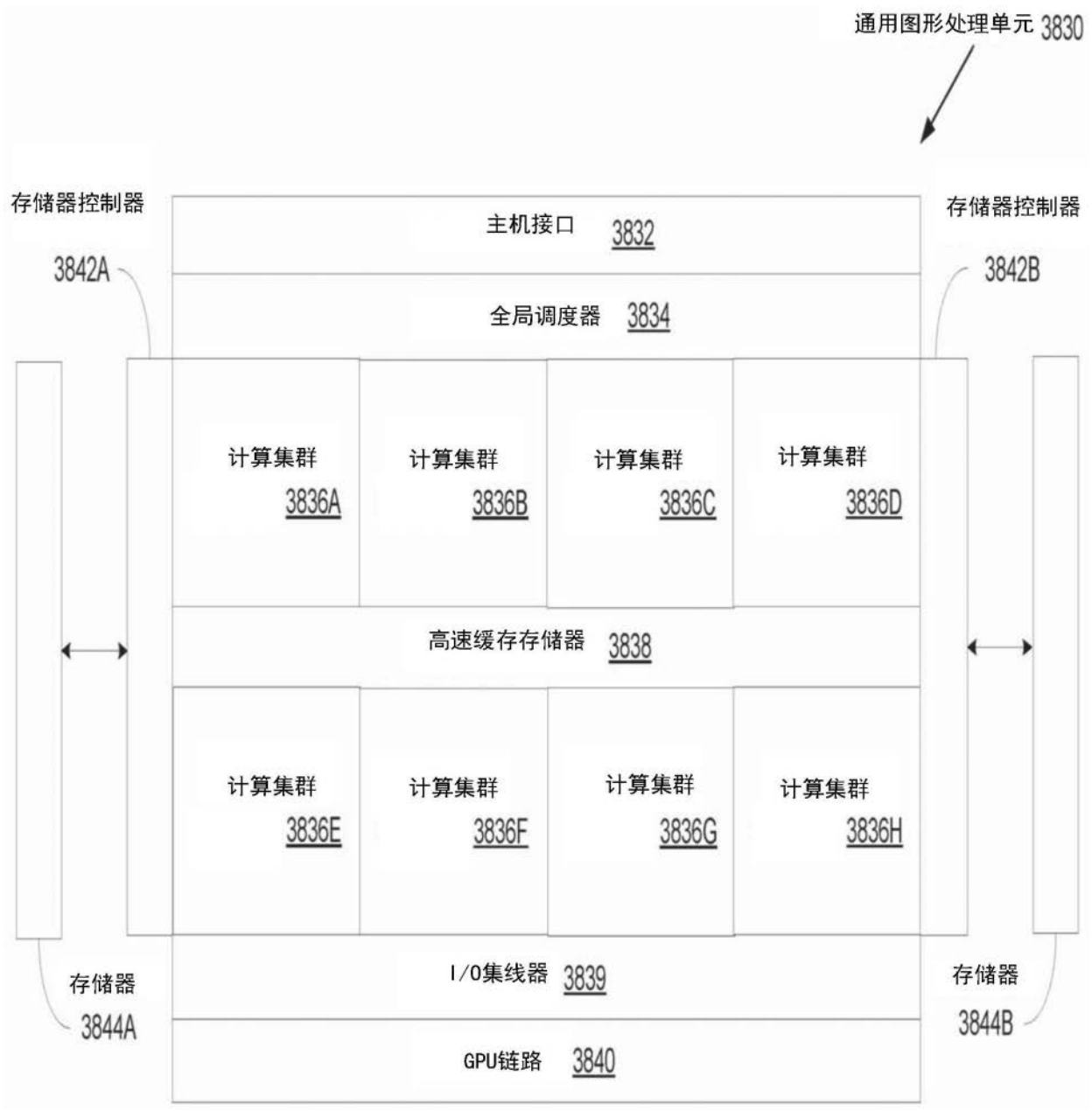

通用图形处理单元 3830
存储器控制器
3842A
主机接口 3832
全局调度器 3834
存储器控制器
3842B
计算集群 3836A
计算集群 3836B
计算集群 3836C
计算集群 3836D
高速缓存存储器 3838
计算集群 3836E
计算集群 3836F
计算集群 3836G
计算集群 3836H
存储器
3844A
I/O集线器 3839
GPU链路 3840
存储器
3844B

图38B

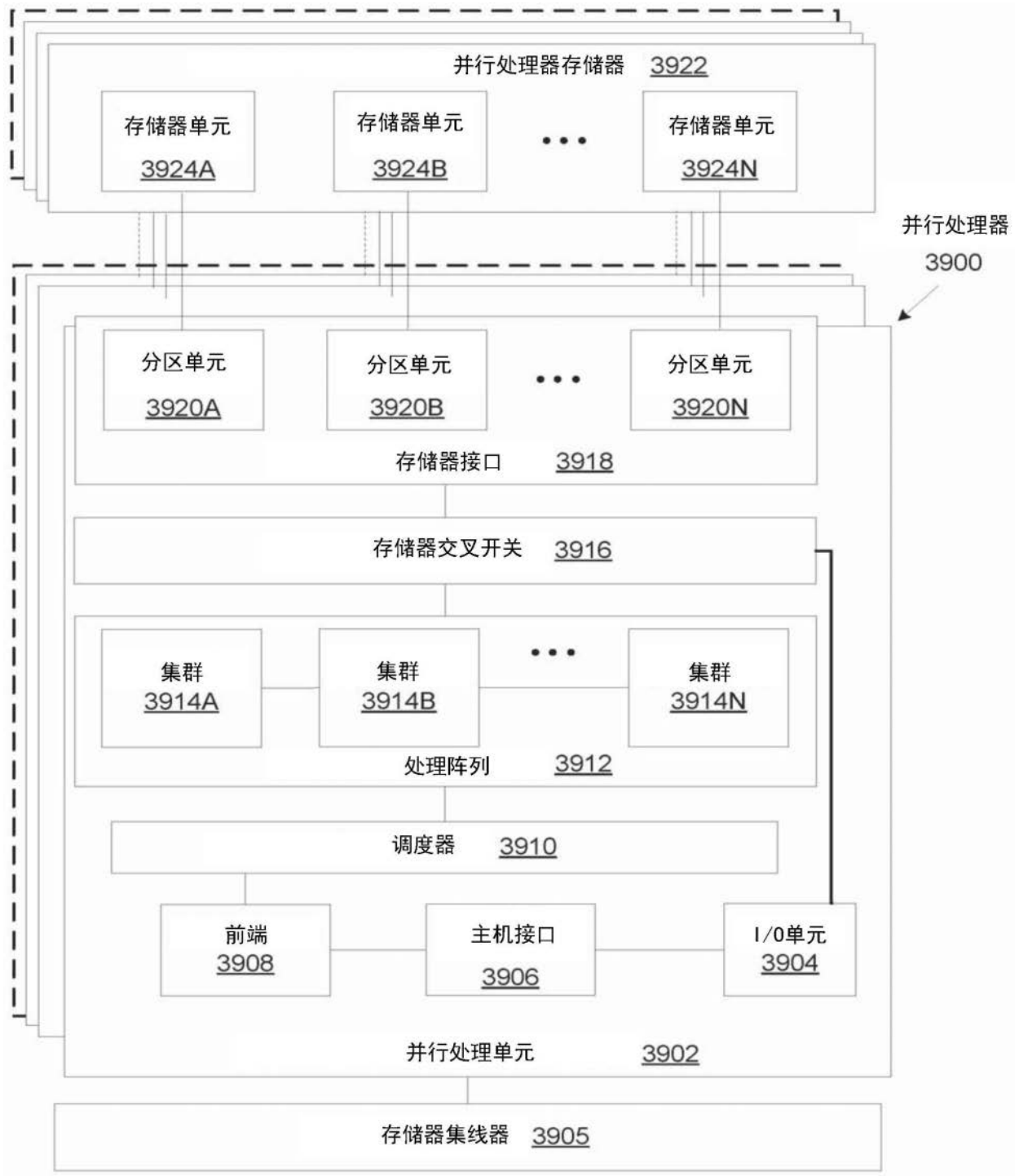

并行处理器存储器 3922
存储器单元 3924A
存储器单元 3924B
存储器单元 3924N
并行处理器 3900
分区单元 3920A
分区单元 3920B
分区单元 3920N
存储器接口 3918
存储器交叉开关 3916
集群 3914A
集群 3914B
集群 3914N
处理阵列 3912
调度器 3910
前端 3908
主机接口 3906
I/O单元 3904
并行处理单元 3902
存储器集线器 3905

图39A

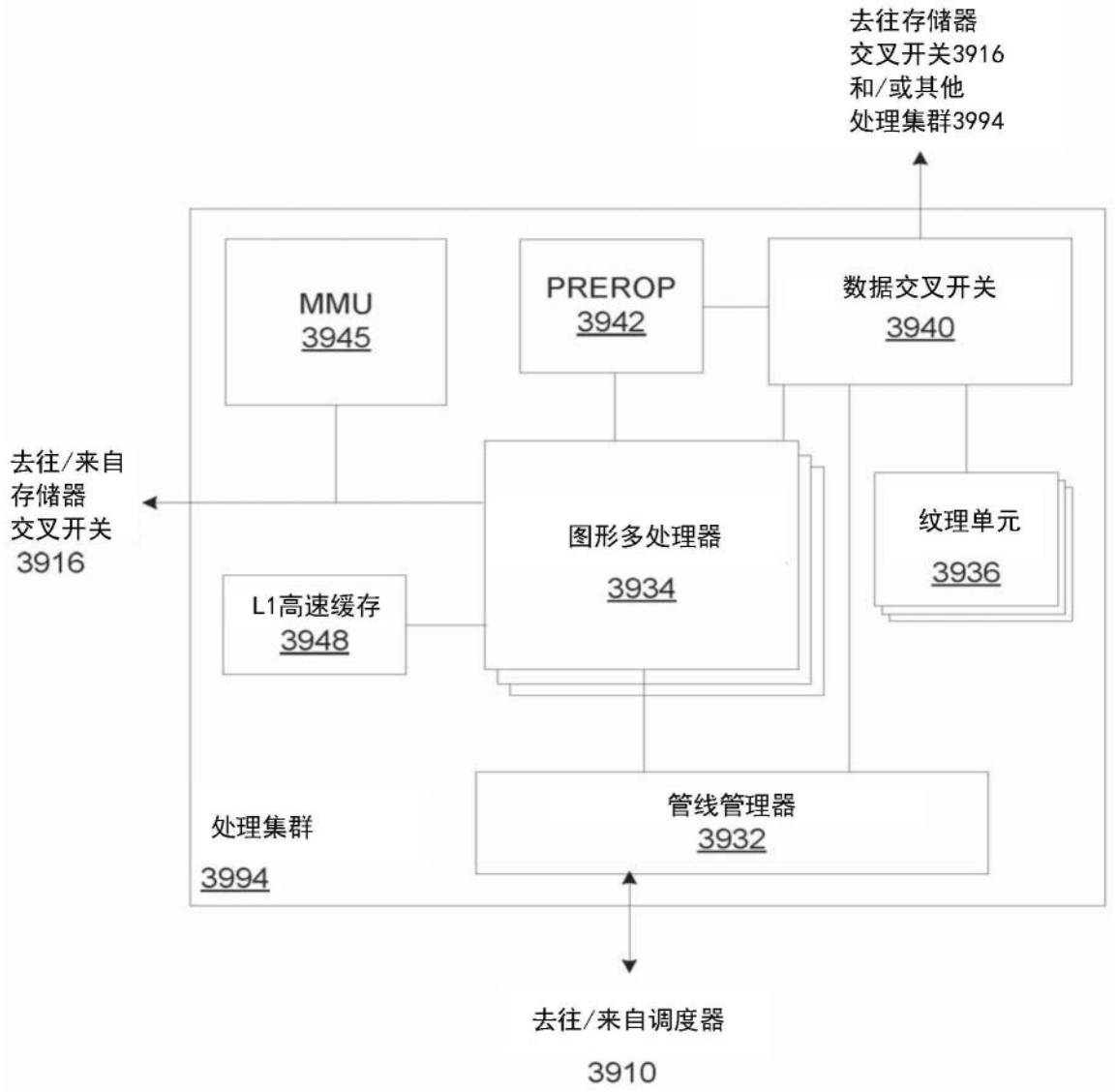

去往存储器
交叉开关3916
和/或其他
处理集群3994
MMU
3945
PREROP
3942
数据交叉开关
3940
去往/来自
存储器
交叉开关
3916
图形多处理器
3934
纹理单元
3936
L1高速缓存
3948
管线管理器
3932
处理集群
3994
去往/来自调度器
3910

图39B

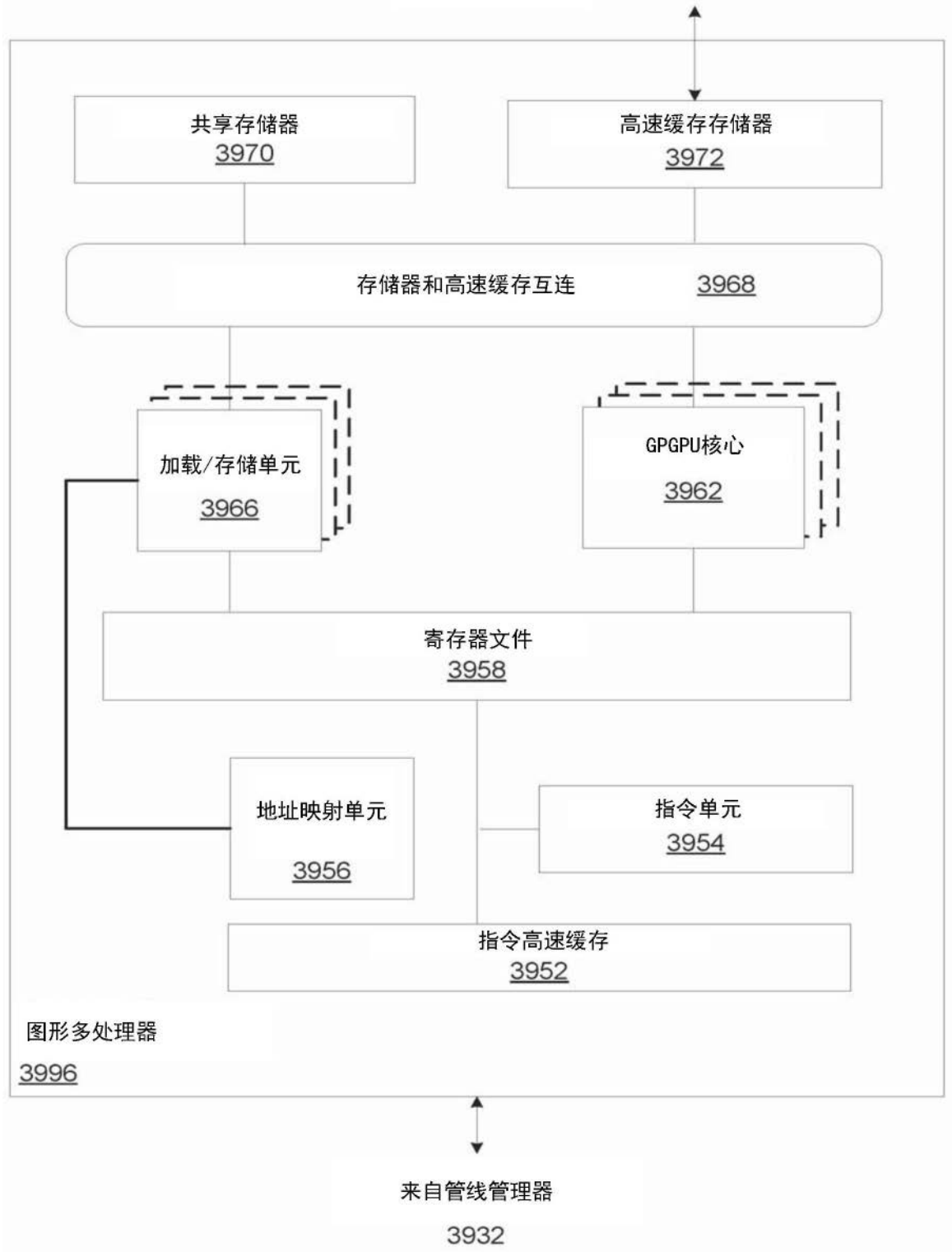

共享存储器
3970
高速缓存存储器
3972
存储器和高速缓存互连
3968
加载/存储单元
3966
GPGPU核心
3962
寄存器文件
3958
地址映射单元
3956
指令单元
3954
指令高速缓存
3952
图形多处理器
3996
来自管线管理器
3932

图39C

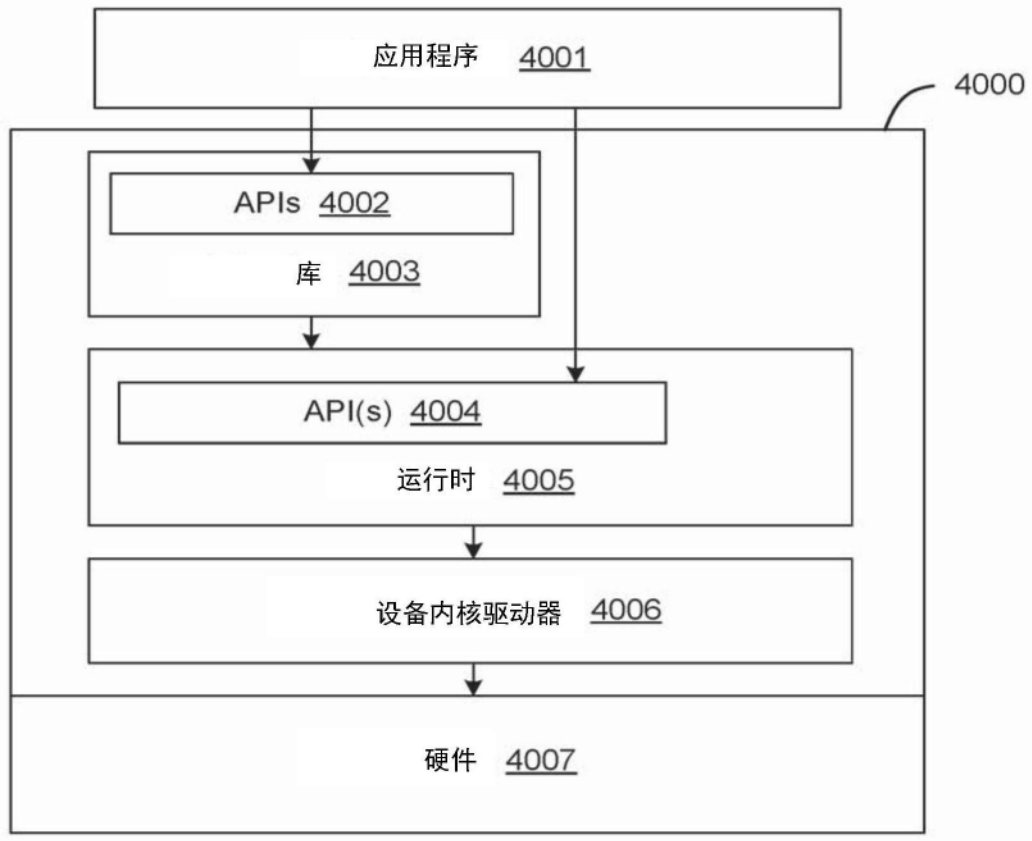

应用程序 4001
4000
APIs 4002
库 4003
API(s) 4004
运行时 4005
设备内核驱动器 4006
硬件 4007

图40

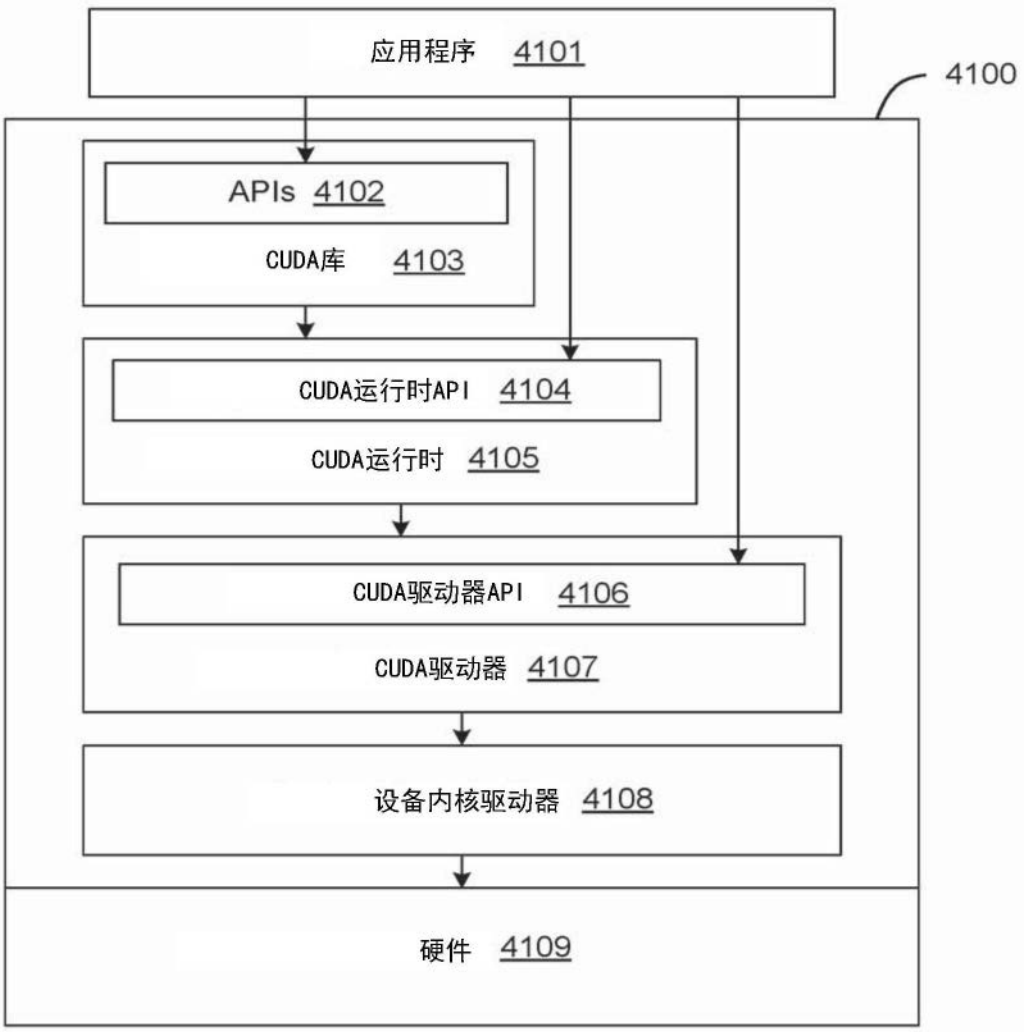

应用程序 4101
4100
APIs 4102
CUDA库 4103
CUDA运行时API 4104
CUDA运行时 4105
CUDA驱动器API 4106
CUDA驱动器 4107
设备内核驱动器 4108
硬件 4109

图41

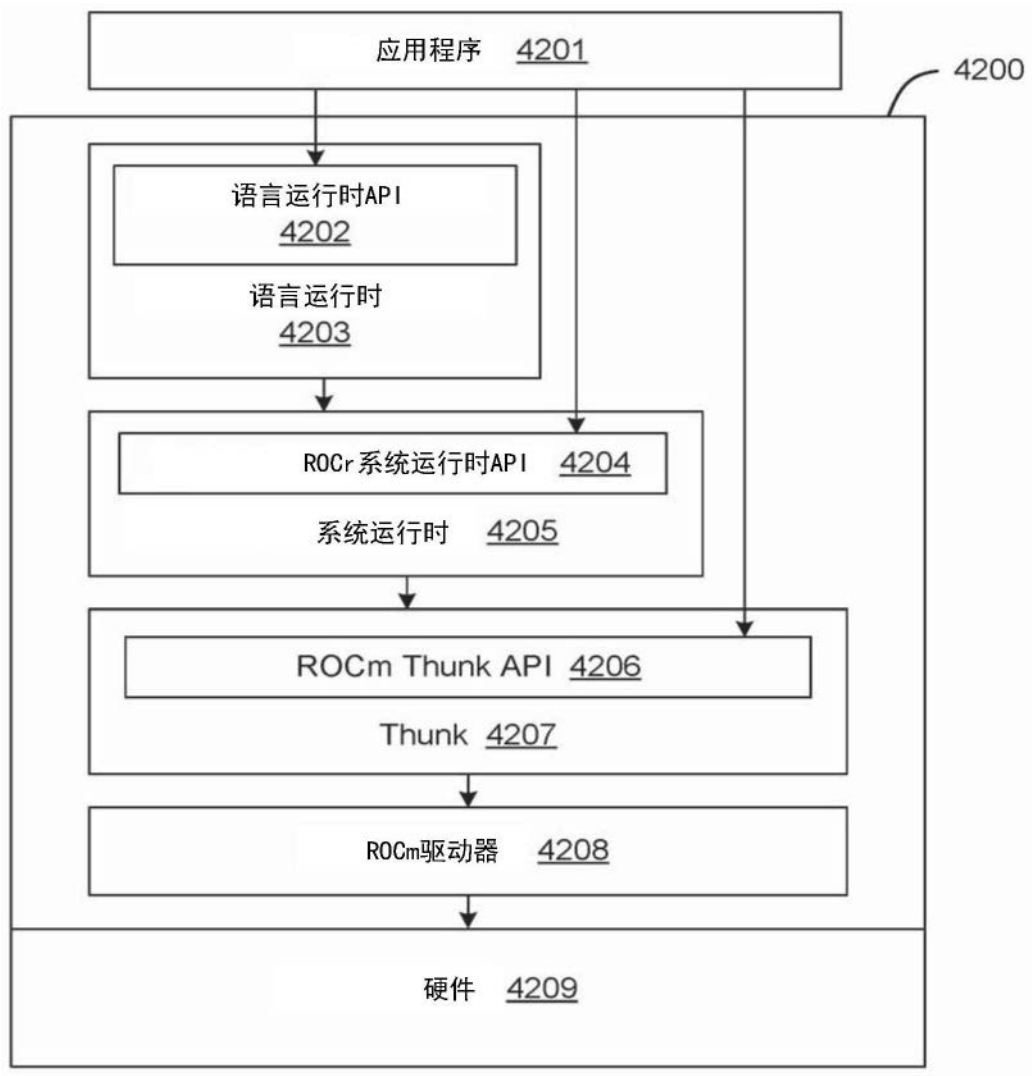

应用程序 4201
4200
语言运行时API
4202
语言运行时
4203
ROCr系统运行时API 4204
系统运行时 4205
ROCm Thunk API 4206
Thunk 4207
ROCm驱动器 4208
硬件 4209

图42

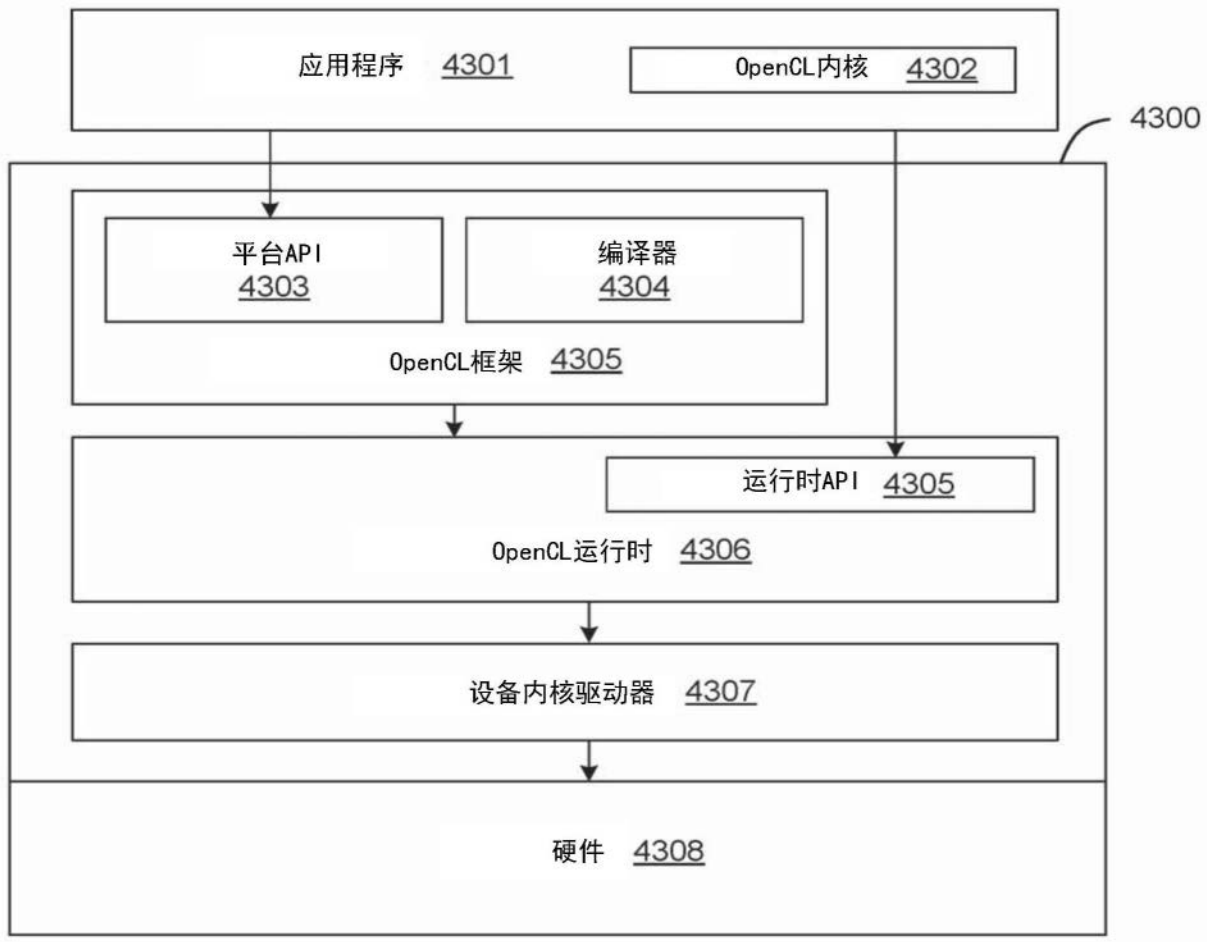

应用程序 4301
OpenCL内核 4302
4300
平台API 4303
编译器 4304
OpenCL框架 4305
运行时API 4305
OpenCL运行时 4306
设备内核驱动器 4307
硬件 4308

图43

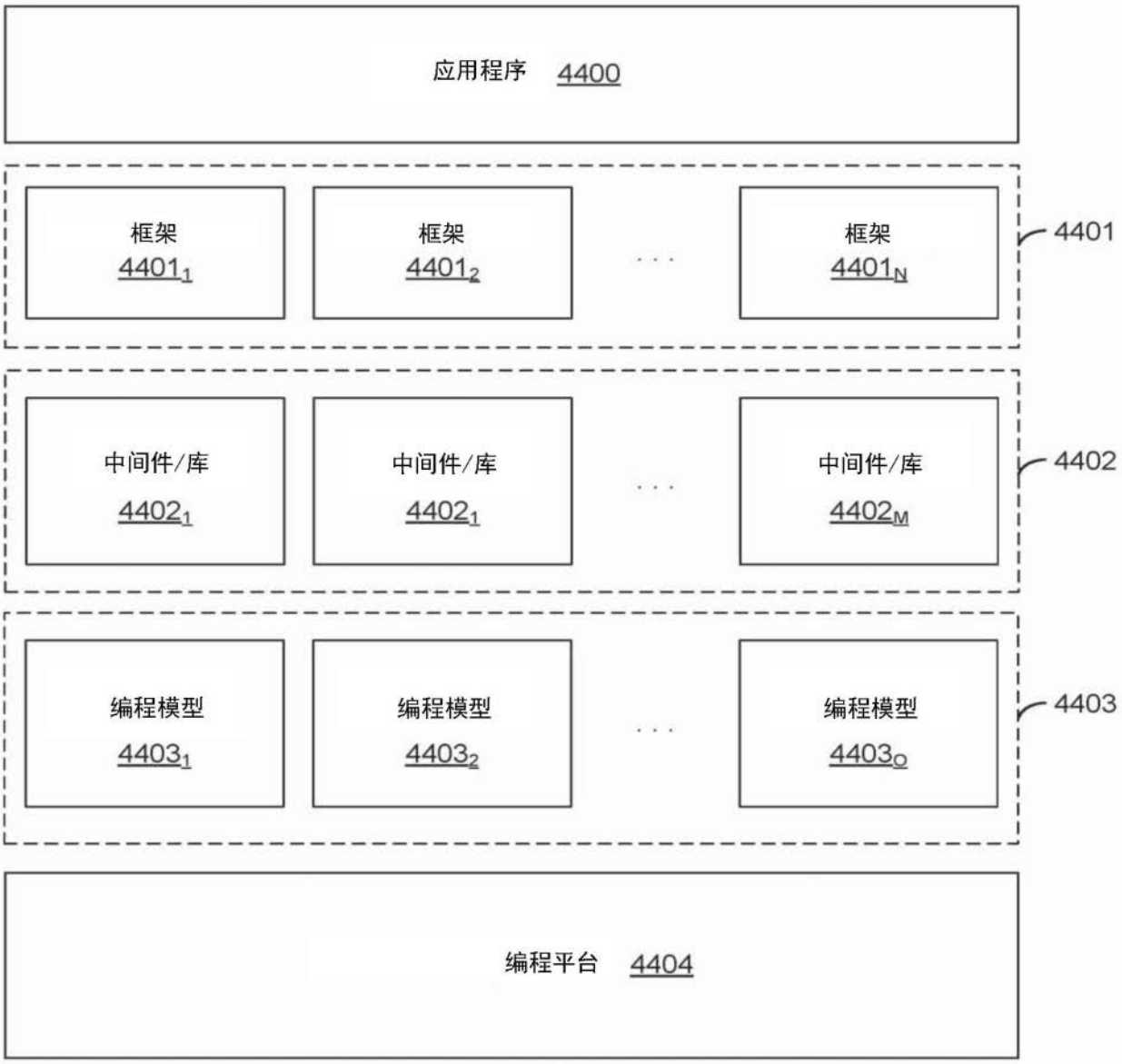

应用程序 4400
框架 4401₁
框架 4401₂
框架 4401N
4401
中间件/库 4402₁
中间件/库 4402₁
中间件/库 4402M
4402
编程模型 4403₁
编程模型 4403₂
编程模型 4403O
4403
编程平台 4404

图44

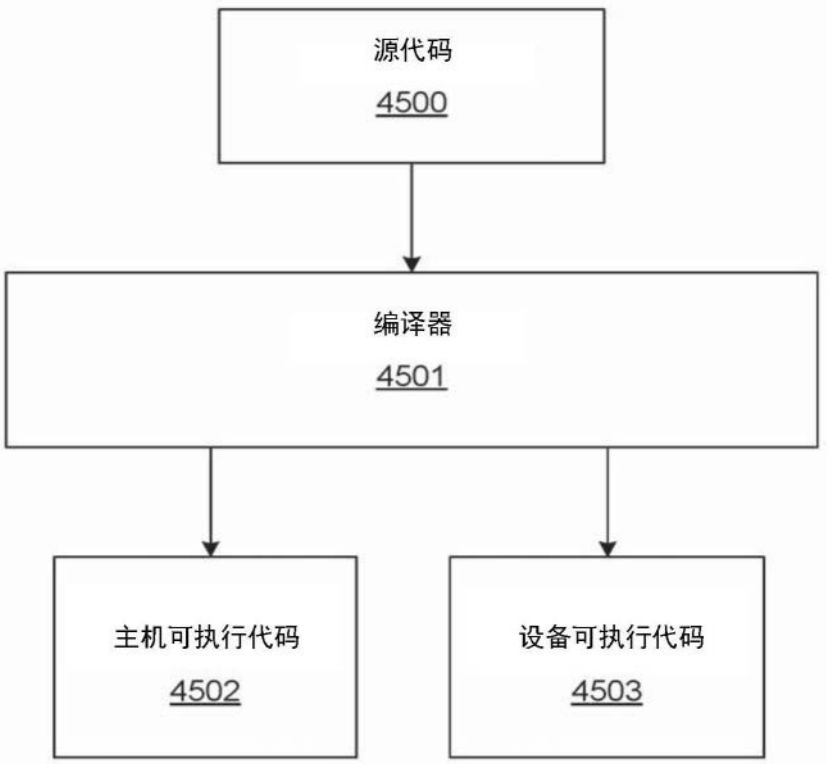


图45