(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 114026548 A
(43)申请公布日 2022.02.08

(21)申请号 202080042864.3
(22)申请日 2020.05.28
(85)PCT国际申请进入国家阶段日
2021.12.14
(86)PCT国际申请的申请数据
PCT/EP2020/064848 2020.05.28
(87)PCT国际申请的公布数据
WO2021/239228 EN 2021.12.02
(71)申请人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼
(72)发明人 本-沙哈尔·贝尔彻 亚历克斯·马戈林 谢·伯格曼 罗宁·凯悦特 丹尼·沃尔金德 利奥·赫尔莫什 塔妮娅·布罗赫曼
(74)专利代理机构 北京龙双利达知识产权代理有限公司 11329
代理人 王君 肖鹂
(51)Int.Cl.
G06F 12/1081(2016.01)

权利要求书3页 说明书7页 附图4页

(54)发明名称

直接内存访问的方法和系统

(57)摘要

描述了一种方法和装置。该方法包括接收数据包和地址数据，该数据包包括要写入计算系统的该内存的数据，地址数据包括计算系统的第一地址空间的地址集合中的地址，识别由与计算系统的内存相关联的第二地址空间中的地址子集识别的地址集合的子集，确定来自第二地址空间中的另一地址子集中的地址，将数据写入与确定地址相关联的内存的区域，以及基于确定地址更新计算系统上的地址转换表。

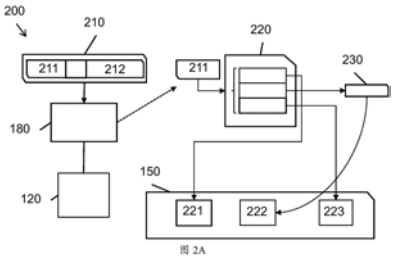

图 2A

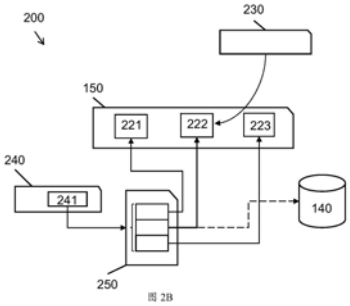

图 2B

CN 114026548 A

1.一种方法,包括:

接收数据包和地址数据,所述数据包包括要写入计算系统的内存的数据,所述地址数据包括所述计算系统的第一地址空间的地址集合中的地址;

识别由与所述计算系统的内存相关联的第二地址空间中的地址子集识别所述地址集合的子集;

确定来自所述第二地址空间中的另一地址子集的地址;

将所述数据写入与所述确定地址相关联的所述内存的区域;以及

基于所述确定地址更新所述计算系统上的地址转换表。

2.根据权利要求1所述的方法,其中,识别由所述第二地址中的地址子集识别的所述地址子集,包括:

访问所述地址转换表;以及

基于所述地址转换表识别所述子集。

3.根据权利要求1所述的方法,其中,所述第二地址空间为所述计算系统的物理内存的物理地址空间。

4.根据权利要求1所述的方法,其中,所述第一地址空间为虚拟地址空间。

5.根据权利要求1所述的方法,其中,确定来自根据所述第二地址空间中的另一地址子集的子集,包括:

访问存储的地址数据,用于所述另一子集中的一个或多个地址;以及

确定来自所述存储的地址数据的所述另一子集中的地址。

6.根据权利要求1所述的方法,其中,确定来自所述第二地址空间中的另一地址子集的地址,包括:

向所述计算系统发送请求,以识别来自所述另一子集的地址;以及

接收包括地址数据的响应,用于来自所述计算系统的另一子集中的地址。

7.权利要求1所述的方法,包括,移除来自所述第二地址空间的另一地址子集的确定地址。

8.根据权利要求1所述的方法,包括:

确定所述另一子集中所述第二地址空间的地址数;以及

当所述地址数低于阈值数时,补充所述另一子集。

9.根据权利要求8所述的方法,其中,补充所述另一子集包括:基于标准确定所述第二地址空间中的地址的所述可用性;以及

根据所述确定将所述地址包括在所述另一子集中。

10.根据权利要求9所述的方法,其中,所述标准包括所述计算系统使用所述地址的标准。

11.根据权利要求1所述的方法,其中,所述数据包的地址为直接内存访问(DMA)请求的目的地址。

12.根据权利要求1所述的方法,其中,所述数据包的地址为远程直接内存访问(RDMA)请求的目的地址。

13.根据权利要求1所述的方法,包括:

访问一个或多个另一地址转换表,每个表包括所述集合中每个地址的条目;以及

更新与所述数据包地址对应的所述一个或多个另一地址转换表的条目，以由所述确定地址识别所述数据包的地址。

14.根据权利要求1所述的方法，其中，所述地址转换表存储在所述计算系统中。

15.根据权利要求1所述的方法，其中，所述地址转化表存储在设备中，所述设备对所述计算系统执行直接内存访问或远程直接内存访问请求。

16.一种用于计算系统的装置，被布置为：

识别来自数据包和地址数据的计算系统的第一地址空间的地址集合中的地址，所述数据包包括要写入计算系统的内存的数据，所述地址数据包括所述地址；

识别用与所述计算系统的内存相关联的第二地址空间的地址子集识别的所述地址集合的子集；

确定来自所述第二地址空间中的另一地址子集的地址；

将所述数据写入与所述确定地址相关联的所述内存的区域；以及

将消息发送到所述计算系统，以基于所述确定地址更新计算系统上的地址转换表。

17.根据权利要求16所述的装置，其中，为了识别用所述第二地址中的地址子集识别的所述地址子集，所述装置被布置为：

访问所述地址转换表；以及

基于所述地址转换表识别所述子集。

18.根据权利要求16所述的装置，其中，所述第二地址空间为所述计算系统的物理内存的物理地址空间。

19.根据权利要求16所述的装置，其中，所述第一地址空间为所述计算系统的虚拟地址空间。

20.根据权利要求16所述的装置，其中，为确定来自所述第二地址空间中的另一地址子集的地址，所述装置被布置为：

将地址数据存储在所述另一子集的一个或多个地址中；以及

确定来自所述存储的地址数据的所述另一子集中的地址。

21.根据权利要求16所述的装置，其中，为确定来自所述第二地址空间中的另一地址子集的地址，所述装置被布置为：

向所述计算系统发送包括请求的查询，以识别来自所述另一子集的地址；以及

接收包括地址数据的响应，用于来自所述计算系统的另一子集的地址。

22.根据权利要求16所述的装置，其中，所述装置被布置为向所述计算系统发送包括请求的消息，以移除来自所述第二地址空间的所述另一子集的所述确定地址。

23.根据权利要求16所述的装置，其中所述装置被布置为：

确定所述另一子集中所述第二地址空间的地址数；以及

当所述地址数低于阈值数时，更新所述另一子集。

24.根据权利要求23所述的装置，其中，所述装置还被布置为存储用于所述更新的另一子集的地址数据。

25.根据权利要求16所述的装置，其中，所述数据包的地址为向所述计算系统的内存的直接内存访问(DMA)请求的目的地址。

26.根据权利要求16所述的装置，其中，所述数据包的地址为向所述计算系统的内存的

远程直接内存访问(RDMA)请求的目的地址。

27.根据权利要求16所述的装置,其中所述装置被布置为:

访问一个或多个另一地址转换表,每个表包括所述集合中每个地址的条目;以及

更新与所述数据包地址对应的所述一个或多个另一地址转换表的条目,以由所述确定地址来标识所述数据包的地址。

28.根据权利要求16所述的装置,其中,所述地址转换表存储在所述计算系统中。

29.根据权利要求16所述的装置,其中,所述地址转化表存储在设备中,所述设备对所述计算系统执行直接内存访问或远程直接内存访问请求。

直接内存访问的方法和系统

技术领域

[0001] 本公开涉及一种对计算系统的内存写入数据的系统和方法。本公开尤其涉及用于对计算系统的内存执行直接内存访问操作的系统和方法。

背景技术

[0002] 直接内存访问(Direct memory access,DMA)使计算系统中的设备或子系统可以直接从计算系统的物理内存中读取、向其写入数据。DMA可以通过诸如图形处理单元或声卡之类的设备或多核系统中的其他处理器核在计算系统中实现。DMA释放了计算系统中的计算资源。特别地,当程序在主处理器上运行时,可以同时执行DMA操作。

[0003] 远程直接内存访问(Remote direct memory access,RDMA)使计算系统可以通过网络从另一计算系统的内存读取数据或向其写入数据。RDMA可以提高网络性能,从而可以在未实现RDMA的系统上实现更高的吞吐量及更低的网络延迟。

发明内容

[0004] 本公开旨在提供一种用于计算系统的方法,诸如可以用于对计算系统的内存执行直接内存访问操作。

[0005] 前述和其它目的通过独立权利要求的特征来实现。根据从属权利要求、说明书和附图,进一步的实现方式是显而易见的。

[0006] 第一方面,提供了一种方法。所述方法包括接收数据包和地址数据,所述数据包包括要写入计算系统的所述内存的数据,所述地址数据包括所述计算系统的第一地址空间的地址集合中的地址。所述方法包括识别由与所述计算系统的内存相关联的第二地址空间中的地址子集识别所述地址集合的子集;确定来自所述第二地址空间中的另一地址子集的地址;将所述数据写入与所述确定地址相关联的所述内存的区域;以及基于所述确定地址更新所述计算系统上的地址转换表。

[0007] 根据第一方面所述的方法将数据写入计算系统的内存中,例如,当数据包的目的地址未映射并且尚未在第二地址空间中分配地址时。所述方法改善了延迟,并减少了在计算系统中执行直接内存访问操作的开销。

[0008] 根据第二方面,提供了一种用于计算系统的装置。所述装置被布置为识别来自数据包和地址数据的计算系统的第一地址空间的地址集合中的地址,所述数据包包括要写入计算系统的内存的数据,所述地址数据包括所述地址;识别用与所述计算系统的内存相关联的第二地址空间的地址子集识别的所述地址集合的子集。所述装置被布置为确定来自所述第二地址空间中的另一地址子集的地址;将所述数据写入与所述确定地址相关联的所述内存的区域;以及将消息发送到所述计算系统,以基于所述确定地址更新计算系统上的地址转换表。

[0009] 在一种实现方式中,识别用所述第二地址中的地址子集识别的所述地址子集,包括访问所述地址转换表;以及基于所述地址转换表识别所述子集。

[0010] 在另一实现方式中，所述第二地址空间为所述计算系统的物理内存的物理地址空间。

[0011] 在另一实现方式中，所述第一地址空间为虚拟地址空间。

[0012] 在另一实现方式中，确定来自所述第二地址空间中的另一地址子集的地址，包括将地址数据存储在所述另一子集的一个或多个地址中；以及确定来自所述存储的地址数据的所述另一子集中的地址。

[0013] 根据本实现方式所述的方法提供了一种方法，用于从存储地址池中选择地址，以将第一地址空间中未映射地址映射到所述地址中。

[0014] 在另一实现方式中，确定来自所述第二地址空间中的另一地址子集的地址，包括向所述计算系统发送包括请求的查询，以识别来自所述另一子集的地址；以及接收包括地址数据的响应，用于来自所述计算系统的另一子集的地址。

[0015] 根据本实现方式所述的方法提供了一种替代方法，用于从可用地址池中确定地址以将未映射地址映射到该地址中。

[0016] 在另一实现方式中，所述方法包括移除来自所述第二地址空间的所述另一子集的确定地址。

[0017] 根据本实现方式所述的方法提供了一种方法，以移除来自所述第二地址空间的所述另一子集的所述确定地址，并且因此不再可用于将未映射地址映射到该地址中。

[0018] 在另一实现方式中，所述方法包括确定所述另一子集中所述第二地址空间的地址数；以及当所述地址数低于阈值数时，补充所述另一子集。

[0019] 根据本实现方式所述的方法补充所述地址池，以确保所述池中有足够的地址来承受包括所述第一地址空间中的未映射地址的大量请求。

[0020] 在另一实现方式中，补充所述另一子集，包括基于标准确定所述第二地址空间中的地址的所述可用性；以及根据所述确定将所述地址包括在所述另一子集中。

[0021] 在另一实现方式中，所述标准包括所述计算系统使用所述地址的标准。

[0022] 根据本实现方式所述的方法提供了一种基于内存地址的使用的标准，用于确定地址的可用性以补充地址池。所述方法的本实现方式确保所述计算系统未充分使用的地址被回收并包括在用于映射未映射地址的池中。

[0023] 在另一实现方式中，所述方法包括存储用于所述另一子集的地址数据。

[0024] 在另一实现方式中，所述数据包的地址为直接内存访问(direct memory access，DMA)请求的目的地址。

[0025] 在另一实现方式中，所述数据包的地址为直接内存访问(direct memory access，DMA)请求的目的地址。

[0026] 在另一实现方式中，所述方法包括访问至少一个其他地址转换表，每个表包括所述集合中每个地址的条目，更新与所述数据包地址对应的所述至少一个其他地址转换表的条目，以使用所述确定的地址来识别所述数据包的地址。

[0027] 在另一实现方式中，所述地址转换表存储在所述计算系统中。

[0028] 在另一实现方式中，所述地址转化表存储在设备中，所述设备对所述计算系统执行直接内存访问或远程直接内存访问请求。

[0029] 根据下文所述的实施例，本申请的以上各个方面以及其他方面是显而易见的。

附图说明

[0030] 为了更完整地理解本公开及其优点,现结合附图参考以下描述,其中:

[0031] 图1示出了根据示例的计算系统。

[0032] 图2A为根据示例的远程直接内存访问请求的示意图。

[0033] 图2B为根据示例的远程直接内存访问请求的示意图。

[0034] 图3示出了根据示例用于计算系统的方法的流程图。

具体实施方式

[0035] 以下下文将充分详尽描述示例性实施例以使本领域普通技术人员能够体现并实现本文描述的系统及过程。应理解,实施例可以以多种不同的形式体现,并且对其解释不应局限于本文所述示例。

[0036] 因此,实施例可以以各种方式进行修改,并且采取各种替代形式,其特定实施例以附图示出并且在下文中以示例形式进行详细描述。本文无意限定所公开的特定形式。相反,应包括落入所附权利要求书范围内的所有修改、等同形式和替代形式。在附图及适当详细描述中,示例性实施例的元件始终由相同的附图标记表示。

[0037] 本文中所用描述实施例的术语并不旨在限制范围。冠词由于“一(a)”,“一个(an)”和“该(the)”为单数形式,因此具有单个指称,但是在本文中使用单数形式不应排除存在多个指称。换言之,除非上下文另外明确指出,以单数形式表示的元件可以编号为一个或多个。应进一步理解为,当在本说明书中使用时,术语“包括(comprises)”、“包括”(includes)和/或“包括(comprises)”指定所述的特征、项目、步骤、操作、元件和/或组件的存在情况,但并不排除存在或添加一个或多个其他特征、项目、步骤、操作、元件、部件和/或其组合。

[0038] 除非另有定义,否则本文中使用的所有术语(包括技术术语和科学术语)都应按照本领域惯例进行解释。应进一步理解为,除非本文明确定义,否则通常使用的术语也应解释为相关领域中的惯用术语,而非理想化或过度形式化的含义。

[0039] 图1为装置100的框图。装置100包括计算系统110。装置100可以与本文所述的方法和系统一起使用。计算系统110包括中央处理单元(central processing unit,CPU)120。CPU 120包括用于读取数据或将数据写入内存的逻辑,并在计算系统110上执行进程。CPU120经由总线130连接到计算系统110的其他组件。总线130可以使计算系统110的互连组件之间进行数据传输。

[0040] 计算系统110包括存储设备140。存储设备140可以包括任何类型的非暂时性存储设备,该非暂时性存储设备用于存储数据、程序和其他信息,并通过总线130使其成为可访问的数据、程序和其它信息。存储设备140可以包括至少一种诸如固态驱动器、硬盘驱动器、磁盘驱动器、或光盘驱动器。存储设备140经由总线130连接到计算系统110的其他组件。

[0041] 计算系统110包括物理内存150。内存150可以包括任何类型的非暂时性系统内存,诸如静态随机访问内存(static random access memory,SRAM)、动态随机访问内存(dynamic random access memory,DRAM)、同步DRAM(synchronous DRAM,SDRAM)、只读内存(read-only memory,ROM),或上述的组合。内存150包括多个内存单元。每个内存单元具有地址,用于识别该内存单元在内存150中的位置。内存150的离散地址范围称为物理地址空间。

[0042] 通常情况下,仅有在启动状态下的诸如BIOS的系统软件和操作系统直接访问物理内存150。对于其他过程,计算系统110保有虚拟地址空间。虚拟地址空间类似于物理地址空间,但该地址不对应物理内存150中的位置。虚拟地址空间为进程提供了连续地址空间外观。虚拟地址到物理地址的映射存储在称为页表的数据结构中。页表的每个条目可以称为页表条目。页面、内存页面或虚拟页面为虚拟内存的固定长度连续块,由单个页表条目表示。帧为页面映射到物理内存的固定长度连续块。

[0043] 除了提供连续地址空间外观,虚拟寻址还可以在两个不相连的区域中创建内存150的虚拟分区。称为内核空间的第一区域被保留以用于受保护的进程,诸如BIOS和操作系统。称为用户空间的第二个区域分配给其他进程。计算系统110通过防止在用户空间中执行的进程在内核空间寻址,来保持内核空间和用户空间的分离。

[0044] 图1所示的CPU 120包括内存管理单元(memory management unit,MMU)160。MMU 160执行地址转换以将通过处理寻址的页面虚拟地址映射到内存150中的对应帧的物理地址。MMU 160还为在计算系统110上运行的进程执行虚拟内存管理。当页面在不同物理内存位置之间移动时,MMU 160管理相应的页面表条目,从而根据需要更新页表。

[0045] 使用称为分页的虚拟内存管理进程,将存储在虚拟地址的数据在物理内存150和其他内存(诸如存储设备140)之间移动。当进程请求虚拟地址空间中的页面时,MMU 160通过执行地址转换来确定所请求的页面在内存150中是否可用。当页面可用时,返回物理地址,并且在CPU 120上执行计算。当页面在内存150中不可用时,MMU 160返回页面错误。在操作系统上运行的软件(称为页面管理程序)访问存储设备140,还原与导致页面错误的页面虚拟地址对应的帧,使用虚拟地址和物理地址之间的新映射关系更新MMU 160中的页表,其中,该页面已恢复到内存150中。

[0046] 分页通过将虚拟地址空间扩展到诸如存储设备140的二级存储设备中,可以使计算系统为超出物理内存150可用空间的进程分配连续虚拟地址范围。然而,当所有帧都在物理内存150中使用时,操作系统必须选择一个帧以重新用作该进程所需的页面。分页管理程序可以使用诸如最近最少使用(Least Recently Used,LRU)或先进先出(First In First Out,FIFO)之类的页面替换算法来确定内存150中的哪个内存位置要释放所请求的页面。分页管理程序可以根据页面替换算法从内存150向存储设备140页面调出或“交换”页面。分页管理程序更新页表,使得该进程所请求的页面指向内存中已释放的位置。为此目的而保留的存储设备140的区域称为交换空间。在某些情况下,可以将页面锁定或“固定”在内存150中,以防止该页面被交换出至存储设备140。

[0047] 计算系统110还包括直接内存访问(DMA)设备170。DMA设备170可以为磁盘驱动器、图形卡、声卡或其他硬件设备。在其他示例中,DMA设备170可以为类似于CPU 120的其他处理核心。DMA设备170连接到总线130,并且可以通过总线130与计算系统110的其他组件进行交互。DMA设备170可以对内存150执行DMA请求。从设备170到内存150的DMA请求为诸如数据写操作的操作,其独立于CPU 120且直接对内存150中的位置执行。若没有DMA,则当CPU 120使用编程的输入/输出时,CPU 120将在读或写操作的整个过程中完全被占用,无法执行其他工作。DMA允许CPU 120在处理源自设备170的DMA请求的同时执行其他操作。DMA对于诸如在设备170与内存150之间执行大型数据传输很有益。一旦完成对内存150的DMA操作,设备170将中断请求发送回CPU 120,从而使CPU 120可以处理来自设备170的数据,该数据在DMA

操作之后被写入内存150。

[0048] 与在计算系统110上运行的进程相似，来自DMA设备170的DMA请求可以在虚拟地址空间指定地址。在一些示例中，DMA设备170被布置为执行在DMA请求中指定地址的地址转换。例如，在一些情况下，DMA设备170和/或计算系统110包括输入输出内存管理单元(input-output memory management unit，IOMMU)(图1中未示出)，该输入输出内存管理单元对计算系统110中的I/O设备执行地址转换，且地址转换方式类似于MMU 160为CPU 120执行地址转换的方式。在其他示例中，DMA设备170使用对操作系统的远程进程调用来跟踪或查询页表条目。

[0049] 计算系统110进一步包括将计算系统110连接到网络190的网络接口控制器(NIC)180。NIC 180可以包括到网络190的有线或无线链路，诸如以太网(Ethernet)或无线发射器和接收器。在一些示例中，网络190可以为局域网(LAN)。在其他示例中，网络190为广域网。网络190可以使计算系统110与远程设备之间进行通信，诸如其他计算系统、网络服务器以及远程存储和数据处理设备。

[0050] 在图1中，计算系统110跨网络190与远程计算设备195通信。根据示例，NIC 180支持从远程设备195到内存150的远程直接内存访问(RDMA)请求。与来自DMA设备170的DMA请求类似，来自远程设备195的RDMA请求为执行直接对内存150中位置的操作请求，该请求绕过了计算系统110的操作系统。RDMA通过使用到内存150的数据零拷贝而无需CPU 120将数据拷贝到其他内存位置，来实现计算系统110与远程设备195之间的高吞吐量、低延迟联网。

[0051] 与在计算系统110上运行的进程以及来自DMA设备170的DMA请求类似，在NIC 180处接收到的RDMA请求可以包括虚拟地址空间的地址。在一些示例中，NIC 180被布置为执行在RDMA请求中指定地址的地址转换。在其他示例中，NIC180使用诸如板载IOMMU来跟踪或查询页表条目。一旦物理地址确定，NIC 180就可以将RDMA请求中的数据直接写入内存150。

[0052] 本文所述的方法和系统可以用于执行DMA(或RDMA)操作，因为发送至未映射虚拟地址的DMA请求在内存150中不具有对应物理地址。

[0053] 解决对未映射虚拟地址的(R)DMA请求的问题的一种方法是将可由DMA设备170或NIC 180使用的子集虚拟地址固定到内存150中的物理地址。这样可以确保(R)DMA操作永远不会遇到未映射的内存。然而，由于固定内存不可供计算系统120中的其他进程或设备使用，因此需要付出相当的代价。此外，由于必须更频繁地将其他内存从内存150换到存储设备140来容纳所需的空间，因此，在此情况下的内存消耗会有损性能。

[0054] 使用固定为DMA操作永久分配内存的另一种方法是在内存150中提供临时固定的缓冲区，用作DMA操作的目的。一旦完成最后一个DMA请求，就可以重新使用每个缓冲区。例如，来自设备195的RDMA请求的输入数据可以首先被放置在内存150中的固定缓冲区中，接着被复制到由进程寻址的虚拟地址空间子空间中的另一缓冲区中。然后，原始固定缓冲区可以在进一步的DMA请求中重复使用。

[0055] 不幸的是，此方法同样具有许多缺点。首先，由于固定缓冲区的额外复制操作，使得延迟显著减少。然而该操作是必需的，否则缓冲区将无法用于其他DMA请求。此外，需要分配专用的固定缓冲池，因此导致计算系统110中产生管理开销，类似于先前描述的固定方法。由于内存150区域中有过多固定区，其他类似方法也有延迟损失或引起大的内存占用。

[0056] 本文所述的方法和系统将来自(R)DMA请求的数据写入缓冲区，然后重新映射虚拟

地址以指向缓冲区地址。本方法不需要停止或将数据进一步复制到缓冲区。

[0057] 图2A为根据本文所述方法的RDMA请求的示例200的简化图。图2A所示的示例200为由图1所示的NIC 180处理的RDMA请求。图1所示的DMA设备170的DMA请求以类似的方式处理，并且图2A所示的示例并不旨在将本文所述的其他方法和示例限制为RDMA请求。

[0058] 在2A所示的示例200中，图1所示的NIC 180接收RDMA请求210。RDMA请求210可由诸如图1所示的远程设备195接收。RDMA请求210包括目的(虚拟)地址211，该目的(虚拟)地址211为诸如在计算系统110上运行的目标进程的虚拟地址空间地址，以及将要写入计算系统110的内存150的数据212。在一些情况下，数据212可以被分成一个或多个数据包，其中各个数据包的虚拟地址由第一数据包虚拟地址的偏移来确定，该偏移由RDMA请求210的虚拟地址211表示。当NIC 180接收到RDMA请求210时，NIC 180执行地址转换(由其本身，或如前所述使用诸如板载IOMMU)，以识别内存150中用于目的虚拟地址211的地址。

[0059] 框220中示出了转换目的地址211的示例。本文示出了三个示例。当RDMA请求的虚拟地址211已经映射到物理内存150中时，NIC 180只需执行地址转换并将数据212写入内存150中的相应位置。例如，在图2中，若虚拟地址211映射至物理地址221或223，则NIC 180将数据212分别写入内存150中的位置221或223。如果NIC 180确定目的地址211的页面未被映射，则NIC 180确定地址222以从对应于内存150中的物理内存位置的可用地址池写入数据212，并将数据写入位于内存150中的物理地址222的缓冲器230中。

[0060] 图2B示出了与图2A所示相同的示例200。在图2B中，示出了虚拟地址空间240和在虚拟地址空间240中寻址地址范围的进程241。一旦NIC 180确定了写入图2A中的RDMA请求210的数据212的地址222，NIC 180通知CPU 120更新进程241的地址转换表。在一些示例中，CPU 120(重新)映射进程241的地址转换表250中的虚拟地址以指向缓冲区230的物理地址222。作为说明性示例，图2B中示出了先前指向存储设备140的交换空间中的位置的虚拟地址到地址222的重新映射。在一些情况下，NIC 180被配置为更新地址转换表250。

[0061] 图3为根据示例示出的方法300的框图。图3所示的方法300可以与本文所述的其他方法和系统结合使用。特别地，方法300可以在图1所示的计算系统110上实现，以处理对内存150的DMA和RDMA请求。

[0062] 在框310处，数据包包括要写入计算系统的内存的数据，地址数据用于接收来自计算系统的第一地址空间的地址集合中的地址。根据示例，数据包可以为DMA请求或RDMA请求的数据包。计算系统和内存可为图1所示的计算系统110和内存150。根据示例，第一地址空间为计算系统的虚拟地址空间。

[0063] 在框320处，识别由与计算系统的内存相关联的第二地址空间中的地址子集识别地址集合的子集。第二地址空间可以为与计算系统的内存相关联的物理地址空间。根据示例，可以从地址转换表识别子集，该地址转换表由第二地址空间中的地址子集识别来自第一地址空间的地址集合的子集。地址转换表包括集合中每个地址的条目。在其他示例中，可以通过访问将第一地址空间中的地址映射到第二地址空间中的地址的函数的输出来确定子集的识别。

[0064] 在框330处，确定来自第二地址空间中的另一地址子集的地址。根据示例，确定第二地址空间另一子集中的地址，包括访问存储的地址数据以获取另一子集中的至少一个地址，从存储的地址数据中确定另一子集中的地址。例如，当方法300在图1中所示的计算系统

110上实现时,NIC 180可以存储地址池,该地址池被用作在请求包含未映射的虚拟地址的情况下写入RDMA请求的数据。

[0065] 在框340,将数据写入与第二地址空间中的确定地址相关联的内存区域。在框350,基于确定地址更新地址转换表的条目。例如,当方法300在图1所示的装置上实现时,NIC 180通知CPU 120更新地址转换表。在一些示例中,CPU 120(重新)映射地址转换表中的虚拟地址,以指向第二(物理)地址空间中的缓冲区地址。

[0066] 根据示例,确定第二地址空间中来自另一子集中的地址,可以包括向计算系统发送请求以识别来自另一子集的地址,从计算系统接收响应,该响应包括另一地址子集的数据的地址数据。例如,NIC 180可以确定IOMMU的缓冲区地址,而不在本地存储地址。

[0067] 在一些示例中,方法300还包括移除来自第二地址空间的地址的另一子集的确定地址。方法300还可以包括,确定另一子集中第二地址空间的地址数,以及当地址数低于阈值数时,补充另一子集。补充另一子集可以包括基于标准确定第二地址空间中的地址的可用性,以及根据确定将地址包括在另一子集中。根据示例,所述标准可以包括计算系统使用地址的标准。这些示例使得计算系统能够保持缓冲区的供应,以无停顿的方式将DMA请求中的数据按需写入系统,而无需将数据复制到其他缓冲区中。

[0068] 本文描述的方法和示例提供资源友好、高效的(R)DMA操作。特别地,所述方法提供了用于处理(R)DMA操作的控制流,目的地为未映射的内存位置。此外,本文描述的方法和系统利用充当(R)DMA操作的明确目的地的可移动缓冲器。已经映射到内存的(R)DMA操作不受影响。

[0069] 本文描述的方法可以在任何具有(R)DMA能力的设备上实现,并且不限于某些设备类型。该方法和系统还可以与虚拟机以及常规进程页面一起使用。实现本文描述的方法的系统的性能接近将整个地址空间固定到内存中的系统的性能,同时还允许内存超额预订和分页。

[0070] 应当理解,本文提供的实施例方法的一个或多个步骤可以由相应的单元或模块执行。各个单元或模块可以是硬件、软件、或其组合。例如,这些单元或模块中的一个或多个单元或模块可以是集成电路,例如,现场可编程门阵列(field programmable gate array,FPGA)或专用集成电路(application-specific integrated circuit,ASIC)。

[0071] 尽管对本公开及其优势进行了详细的描述,但是应当理解的是,在不脱离由所附权利要求限定的本公开的精神和范围的情况下,可以在此进行各种改变、替换和变更。

[0072] 本申请可以在其他特定设备和/或方法中体现。应认为,所描述的实施例在所有方面都是说明性的而非限制性的。特别地,本申请的范围由所附权利要求说明指定而非本文说明书及附图指定。凡在权利要求的含义及等效范围内的变化,都应包含在其范围内。

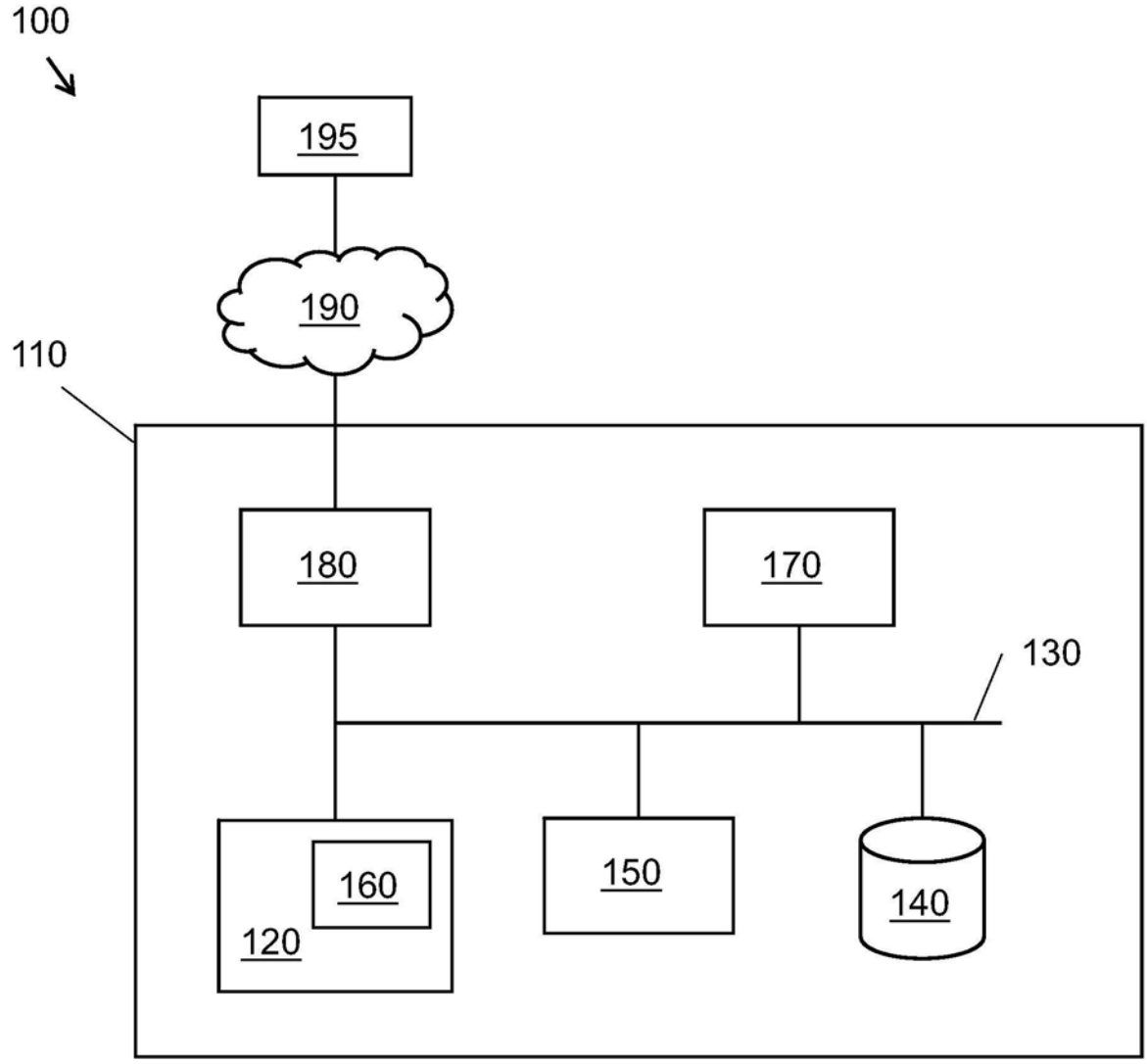

图1

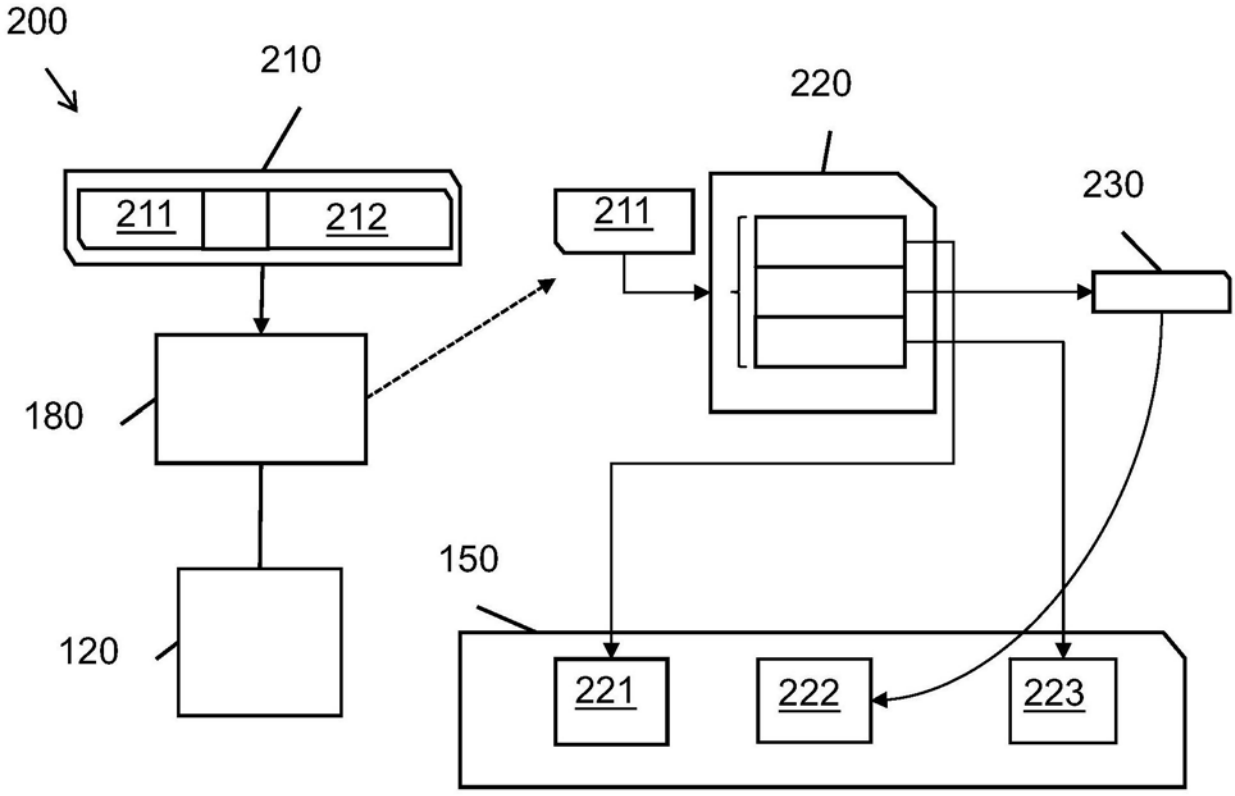

图2A

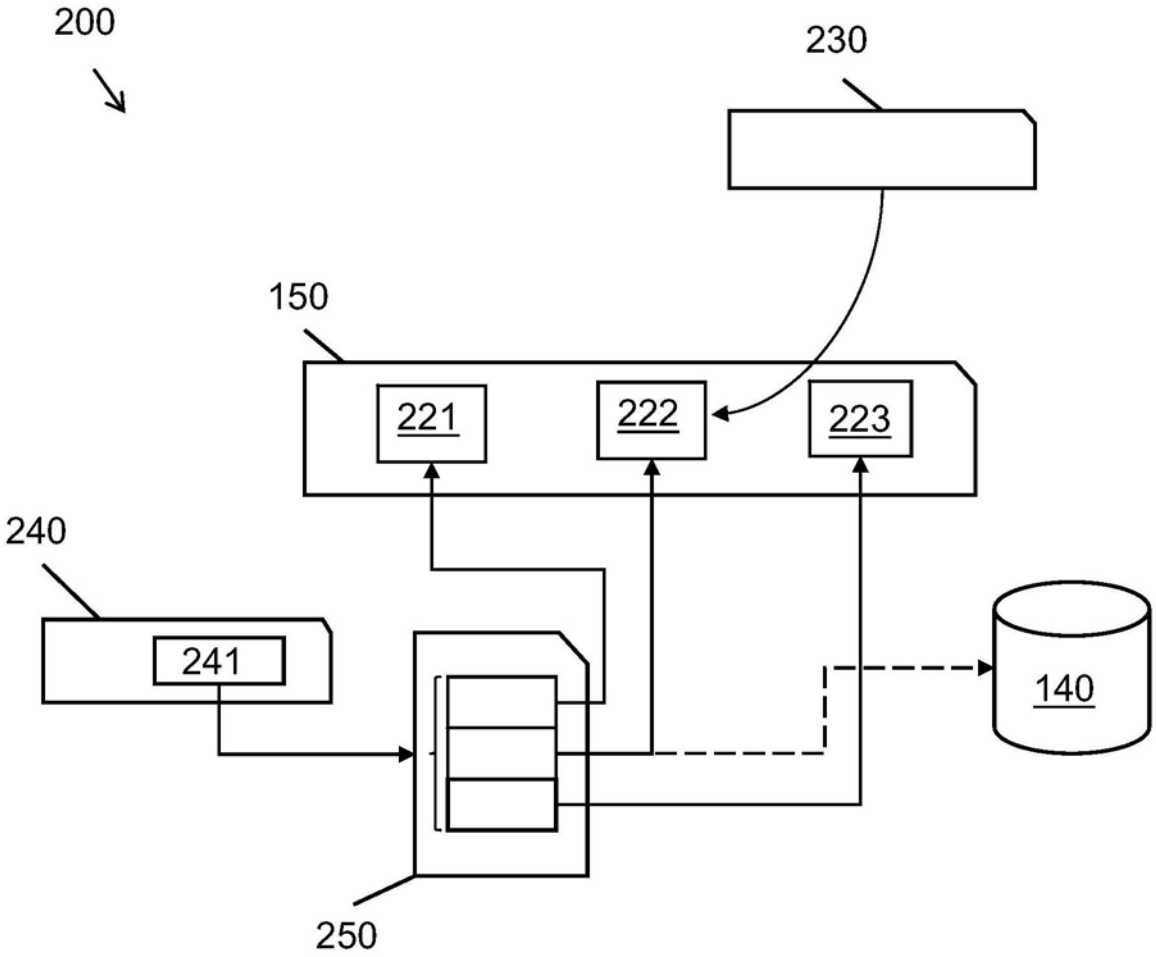

图2B

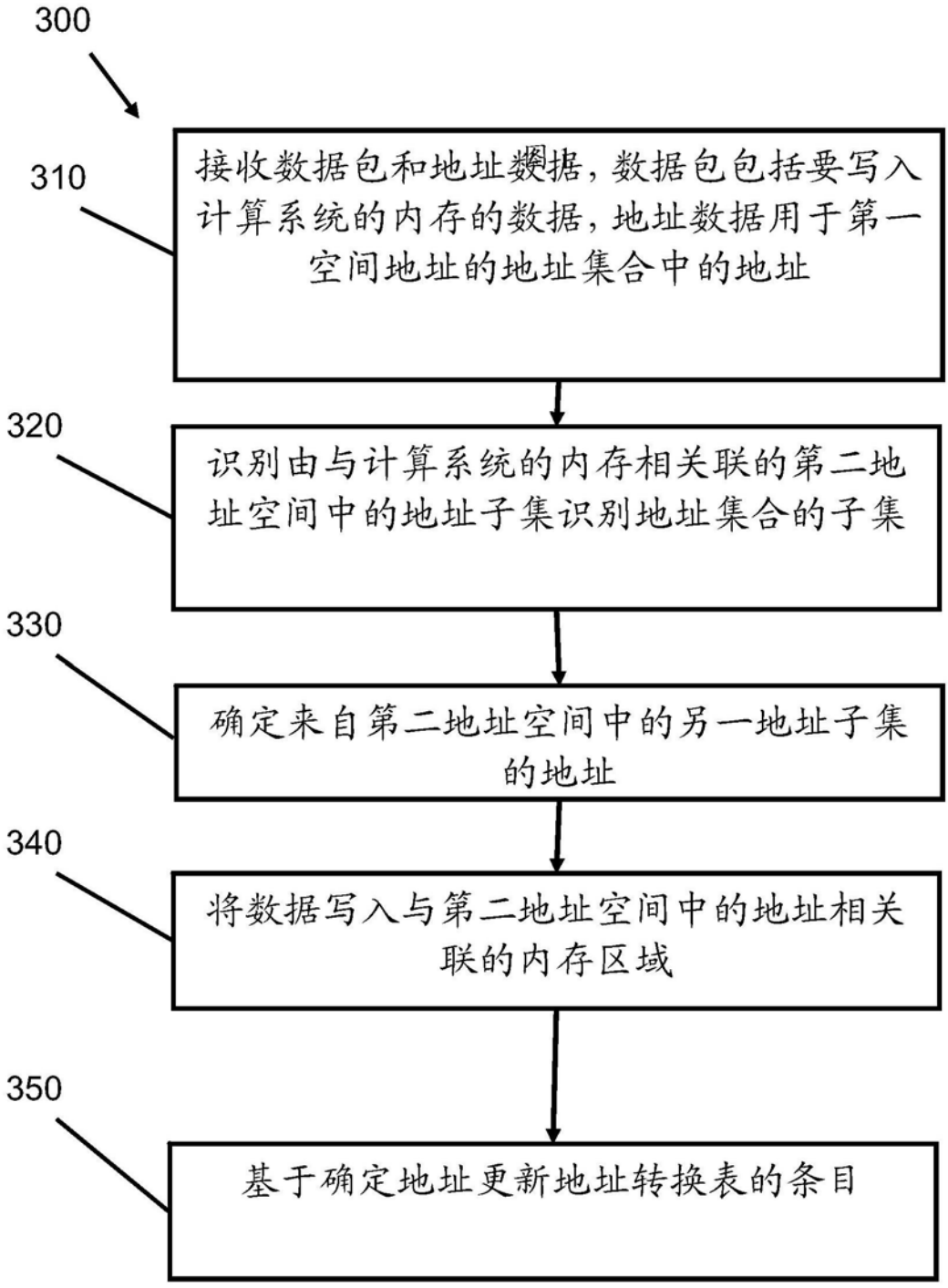
300
310
接收数据包和地址数据，数据包包括要写入计算系统的内存的数据，地址数据用于第一空间地址的地址集合中的地址
320
识别由与计算系统的内存相关联的第二地址空间中的地址子集识别地址集合的子集
330
确定来自第二地址空间中的另一地址子集的地址
340
将数据写入与第二地址空间中的地址相关联的内存区域
350
基于确定地址更新地址转换表的条目

图3