(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101848203 B
(45)授权公告日 2011.08.31

(21)申请号 201010113628.4

(22)申请日 2005.06.23

(30)优先权数据
10/890,710 2004.07.14 US

(62)分案原申请数据
200580023049.8 2005.06.23

(73)专利权人 国际商业机器公司
地址 美国纽约

(72)发明人 道格拉斯·M.·弗里姆斯
埃尔伯特·C·胡 罗纳德·姆拉兹
埃里奇·M·纳胡姆
普拉沙恩特·普拉德汉
萨姆比特·撒胡 约翰·M·翠斯

(74)专利代理机构 中国国际贸易促进委员会专利商标事务所 11038
代理人 袁玥

(51)Int.Cl.
H04L 29/06(2006.01)

(56)对比文件
WO 2004/021150 A2,2004.03.11, 全文.
US 2004/0042483 A1,2004.03.04, 全文.
CN 1497448 A,2004.05.19, 全文.
WO 2004/051489 A2,2004.06.17, 全文.
US 6,735,620 B1,2004.05.11, 全文.
Piyush Shivam et al.EMP:Zero-copy OS-bypass NIC-driven Gigabit Ethernet Message Passing.《ACM 2001》.2011,1-8.

审查员 加玉

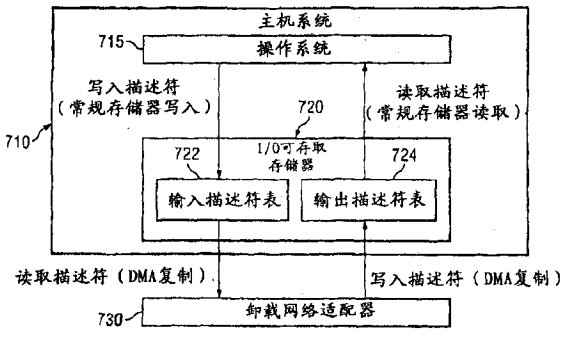
权利要求书 3 页 说明书 24 页 附图 11 页

(54)发明名称

用于从网络适配器向主机系统传送数据的方法和系统

(57)摘要

在用于从主机处理器中卸载协议处理的网络适配器方面提供了多个改进。具体来讲,提供了一种用于在利用卸载网络适配器(730)的系统中改进连接建立的机制。所述连接建立机制提供了将连接建立(1030)和连接状态信息的维护卸载给所述卸载网络适配器(730)的能力。作为这种卸载连接建立(1030)和状态信息维护的结果,在主机系统(710)和卸载网络适配器(730)之间所需的通信次数可以得以减少。此外,把这些功能卸载给所述卸载网络适配器(730),允许向主机系统(710)成批地通知已建立的连接和状态信息,而不是像已知的计算系统中出现的那样逐个地进行通知。



CN 101848203 B

1. 一种在数据处理系统中用于把数据从网络适配器传送至主机系统的方法，包括如下步骤：

在网络适配器中接收用于传送到主机系统的数据；

确定是否为与所述数据相关联的连接分配了连接专用应用缓冲器；

如果还没有为与所述数据相关联的连接分配连接专用应用缓冲器，那么确定是否等待为所述连接分配专用应用缓冲器；

如果确定不等待分配专用应用缓冲器，那么从缓冲器池中选择用于接收所述数据的非连接专用应用缓冲器；并且

把所述数据传送至所选的非连接专用应用缓冲器。

2. 如权利要求 1 所述的方法，还包括如下步骤：

如果确定要等待分配专用应用缓冲器，那么等待为与所述数据相关联的连接分配连接专用应用缓冲器。

3. 如权利要求 2 所述的方法，其中等待连接专用应用缓冲器的步骤包括如下步骤：

确定是否满足或者超出最大等待时间；并且

如果满足或者超出了最大时间，那么中断对连接专用应用缓冲器的等待。

4. 如权利要求 3 所述的方法，其中所述最大等待时间是根据用于提供连接专用应用缓冲器的主机系统的历史数据来确定的。

5. 如权利要求 1 所述的方法，其中确定是否等待分配连接专用应用缓冲器的步骤包括：确定是否已经在网络适配器中设置了等待参数。

6. 如权利要求 1 所述的方法，其中确定是否等待连接专用应用缓冲器的步骤包括：在网络适配器中，根据用于提供连接专用应用缓冲器的主机系统的历史数据，来确定是否等待连接专用应用缓冲器。

7. 如权利要求 1 所述的方法，其中从缓冲器池中选择用于接收数据的非连接专用应用缓冲器的步骤包括如下步骤：

使用缓冲器池中的每个非连接专用应用缓冲器的特性信息来确定与所述数据最佳匹配的非连接专用应用缓冲器。

8. 如权利要求 7 所述的方法，其中所述特性信息包括缓冲器大小、缓冲器速度以及缓冲器在主机处理器体系结构中的位置中的至少一个。

9. 如权利要求 1 所述的方法，还包括如下步骤：

确定是否应该把数据注入到主机系统的 L3 高速缓存中；并且

如果确定应该把数据注入到主机系统的 L3 高速缓存中，那么使用高速缓存注入机制来把数据直接传送至 L3 高速缓存。

10. 如权利要求 1 所述的方法，还包括如下步骤：

检查由网络适配器为所述数据生成的描述符；

确定所述描述符是否标识连续的物理地址空间；并且

如果所述描述符标识连续的物理地址空间，那么在网络适配器和主机系统之间的单次批事务处理中把所述描述符提供至主机系统。

11. 如权利要求 1 所述的方法，还包括如下步骤：

查看数据的首部以便确定数据的类型；并且

根据数据的类型来标识用于路由数据的连接。

12. 如权利要求 11 所述的方法，还包括如下步骤：

根据对首部进行的查看来确定所述首部的大小；并且

根据首部的大小来生成和存储一个偏移，用于跳过首部读取所述数据的有效负载。

13. 一种用于从网络适配器向主机系统传送数据的系统，包括：

用于在网络适配器中接收用于传送到主机系统的数据的装置；

用于确定是否为与所述数据相关联的连接分配连接专用应用缓冲器的装置；

用于如果还没有为与所述数据相关联的连接分配连接专用应用缓冲器，那么确定是否等待为所述连接分配专用应用缓冲器的装置；

用于如果确定不等待分配专用应用缓冲器，那么从缓冲器池中选择用于接收所述数据的非连接专用应用缓冲器的装置；以及

用于把所述数据传送至所选的非连接专用应用缓冲器的装置。

14. 如权利要求 13 所述的系统，还包括：

用于如果确定等待分配专用应用缓冲器，那么等待为与所述数据相关联的连接分配连接专用应用缓冲器的装置。

15. 如权利要求 14 所述的系统，其中用于等待连接专用应用缓冲器的装置包括：

用于确定是否满足或者超出最大等待时间的装置；以及

用于如果满足或者超出了最大时间，那么中断对连接专用应用缓冲器的等待的装置。

16. 如权利要求 15 所述的系统，其中所述最大等待时间是根据用于提供连接专用应用缓冲器的主机系统的历史数据来确定的。

17. 如权利要求 13 所述的系统，其中用于确定是否等待分配连接专用应用缓冲器的装置包括：用于确定是否已经在网络适配器中设置了等待参数的装置。

18. 如权利要求 13 所述的系统，其中用于确定是否等待连接专用应用缓冲器的装置包括：用于在网络适配器中根据用于提供连接专用应用缓冲器的主机系统的历史数据来确定是否等待连接专用应用缓冲器的装置。

19. 如权利要求 13 所述的系统，其中用于从缓冲器池中选择用于接收数据的非连接专用应用缓冲器的装置包括：

用于使用缓冲器池中的每个非连接专用应用缓冲器的特性信息来确定与所述数据最佳匹配的非连接专用应用缓冲器的装置。

20. 如权利要求 19 所述的系统，其中所述特性信息包括缓冲器大小、缓冲器速度以及缓冲器在主机处理器体系结构中的位置中的至少一个。

21. 如权利要求 13 所述的系统，还包括：

用于确定是否应该把所述数据注入到主机系统的 L3 高速缓存中的装置；以及

用于如果确定应该把所述数据注入到主机系统的 L3 高速缓存中，那么使用高速缓存注入机制来把所述数据直接传送至 L3 高速缓存的装置。

22. 如权利要求 13 所述的系统，还包括：

用于检查由网络适配器为所述数据生成的描述符的装置；

用于确定所述描述符是否标识连续的物理地址空间的装置；以及

用于如果所述描述符标识连续的物理地址空间，那么在网络适配器和主机系统之间的

单次批事务处理中把所述描述符提供至主机系统的装置。

23. 如权利要求 13 所述的系统，还包括：

用于查看数据的首部以便确定数据的类型的装置；以及

用于根据数据的类型来标识用于路由数据的连接的装置。

24. 如权利要求 23 所述的系统，还包括：

用于根据对首部进行的查看来确定所述首部的大小的装置；以及

用于根据首部的大小来生成和存储一个用于跳过首部读取所述数据的有效负载的偏移的装置。

用于从网络适配器向主机系统传送数据的方法和系统

[0001] 本申请是申请号为 200580023049.8、申请日为 2005 年 6 月 23 日、名称为“在网络协议处理的卸载中支持连接建立的设备和方法”的发明专利申请的分案申请。

[0002] 技术领域

[0003] 本发明总体上致力于一种改进的数据处理系统。更具体地说，本发明致力于一种用于在卸载 (offload) 网络适配器中支持连接建立操作的方法和设备。

[0004] 背景技术

[0005] 在已知的系统中，仅就数据传送而言，操作系统通过向常规的网络接口提供两个缓冲器队列来与网络接口进行通信。缓冲器的第一队列由用于指向主机存储器中可读取的数据分组的描述符组成，所述数据分组被读取用于进行传输。缓冲器的第二队列包括用于指向装满了主机存储器中的未处理的数据分组的缓冲器的描述符，其中所述数据分组已经被接收用于进行处理。所述网络接口提供了一个存储器映射的输入 / 输出 (I/O) 接口，所述输入 / 输出接口用于向网络接口通知所述队列在物理存储器中处于什么位置，并且所述网络接口还提供了一个用于某些控制信息、诸如当数据分组到达时生成什么样的中断的接口。

[0006] 常规的网络接口的网络协议处理全部在主机内执行，并且只有数据分组被提供至网络适配器以便进行传输。然而，网络链接速度的增长已经快于微处理器性能的发展。因此，主机处理器承担了大量的 TCP/IP 协议处理、重组无序数据分组、资源密集的存储器拷贝和中断。在某些高速网络中，与处理正在运行的应用相比，主机处理器必须进行更多的处理来解决网络通信量的问题。由此，数据分组在主机中以低于网络速度的速率被处理。

[0007] 为了解决此问题，最近的重点已经放在了将 TCP/IP 协议的处理从主机处理器中卸载 (offload) 到网络适配器上的硬件中。这种网络适配器 (有时也称为智能网络适配器或者 TCP/IP 卸载引擎 (TOE)) 可以利用网络处理器和固件、专用 ASIC 或者两者的组合来实现。这些网络适配器不仅卸载主机处理器的处理以便提高应用性能，而且能够与新型的网络和装置进行通信，所述新型的网络和装置诸如是 iSCSI 存储区域网络 (SAN) 和高性能网络附属存储 (NAS) 应用。

[0008] 虽然这些网络适配器卸载了数据分组的 TCP/IP 协议处理，但是经由网络进行通信所需要的大部分处理仍保留在主机系统内。例如，主机系统仍负责建立连接、为每个已建立的连接维护状态信息、处理存储器管理等。由此，因为必须在主机系统中执行这些操作，并且还因为要在主机系统中执行这些操作而在主机系统和网络适配器间所需要的通信量，主机系统仍旧承受处理器负荷。由此，具有一种用于改进网络适配器的操作以便使主机系统上的处理负荷最小化并且使更多的处理在网络适配器中执行的设备和方法，将会是十分有益的。

发明内容

[0009] 本发明提供了用于从主机处理器卸载协议处理的网络适配器方面的多个改进，此后将该网络适配器称为卸载网络适配器。具体来讲，本发明提供了利用卸载网络适配器来

处理系统内的存储器管理和优化的机制。另外，本发明提供了利用卸载网络适配器来改进系统中的连接建立的机制。此外，本发明提供了利用卸载网络适配器来处理系统中的数据分组的接收的改进机制。

[0010] 本发明的一个方面在于能够把连接建立和连接状态信息的维护卸载给所述卸载网络适配器。作为这种卸载连接建立和状态信息维护的结果，主机系统和卸载网络适配器之间所需的通信次数得以减少。 另外，把这些功能卸载给所述卸载网络适配器，允许向主机系统成批地通知已建立的连接和状态信息，而不是像已知计算系统中出现的那样逐个地通知。

[0011] 除连接建立以外，本发明还对使用卸载网络适配器的数据处理系统中的存储器管理加以改进。根据本发明的存储器管理不仅允许数据的缓冲后发送和接收，而且允许数据的零拷贝 (zero-copy) 发送和接收。另外，本发明允许根据任意数目的属性来对可以在指定的连接当中共享的 DMA 缓冲器进行分组。本发明还允许部分发送和接收缓冲操作，延迟 DMA 请求，以便使 DMA 请求可以被成批地传递至主机系统，并且本发明还提供了一种用于加快向主机系统传送数据的机制。

[0012] 除连接建立和存储器管理以外，本发明还对使用卸载网络适配器的数据处理系统中的已接收数据的处理加以改进。本发明的卸载网络适配器可以包括用于允许卸载网络适配器以不同的方式延迟向主机系统通知数据接收情况的逻辑。延迟向主机系统通知数据分组接收情况的优点在于 :可能在单个通知中合并多个紧接在例如第一个数据分组后到达的数据分组。考虑到具有连续数据分组到达的流，可以为通知延迟设置一个值，并且可以每一通信套接字地为主机系统配置该值。

[0013] 本发明的这些以及其它特征和优点将在随后对优选实施例的详细说明中进行描述，并且对本领域普通技术人员而言，根据这些描述，本发明将变得更加显而易见。

附图说明

[0014] 在所附权利要求书中阐明了被认为是本发明特性的新颖性特征。然而，当结合附图阅读并参照以下对图示实施例的详细说明时，将会更好地理解本发明本身及其优选的使用方式、进一步的目的和优点，其中 :

[0015] 图 1 是可以实现本发明各方面的分布式数据处理系统的示例性 简图 ;

[0016] 图 2 是可以实现本发明各方面的服务器计算装置的示例性简图 ;

[0017] 图 3 是可以实现本发明各方面的客户端计算装置的示例性简图 ;

[0018] 图 4 是根据本发明一个示例性实施例的网络适配器的示例性简图 ;

[0019] 图 5 是举例说明利用常规网络接口卡的系统中的 TCP/IP 处理的简图 ;

[0020] 图 6 是举例说明利用 TCP/IP 卸载引擎或者卸载网络适配器的系统中的 TCP/IP 处理的简图 ;

[0021] 图 7 是针对本发明的卸载网络适配器编程接口来举例说明本发明一个示例性实施例的各方面的示例性简图 ;

[0022] 图 8 是针对使用卸载网络适配器和卸载网络适配器编程接口建立连接来举例说明本发明一个示例性实施例的各方面的示例性简图 ;

[0023] 图 9 是概述当使用卸载网络适配器建立连接时本发明的主机系统的示例性操作

的流程图；

[0024] 图 10 是概述当按照本发明的一个示例性实施例建立连接时卸载网络适配器的示例性操作的流程图；

[0025] 图 11 是举例说明依照本发明的使用数据的缓冲后发送和接收的存储器管理机制的示例性简图；

[0026] 图 12 是根据本发明一个示例性实施例举例说明零拷贝操作的示例性简图；

[0027] 图 13 是根据本发明一个示例性实施例来举例说明共享缓冲器设置的示例性简图；

[0028] 图 14 举例说明了按照本发明一个示例性实施例的部分接收 / 发送缓冲操作的方式；

[0029] 图 15 举例说明了按照本发明一个示例性实施例的示例性 DMA 传送顺序决策处理过程；

[0030] 图 16 是概述当按照本发明一个示例性实施例的各方面、使用主机系统和卸载网络适配器发送数据时的示例性操作的流程图；

[0031] 图 17 是概述当按照本发明一个示例性实施例的各方面在主机系统和卸载网络适配器之间执行数据的零拷贝传送时的示例性操作的流程图；并且

[0032] 图 18 是概述按照本发明一个示例性实施例的各方面用于确定要向其发送数据的应用缓冲器的示例性操作的流程图。

具体实施方式

[0033] 本发明致力于一种用于改进卸载网络适配器的操作的设备和方法，所述卸载网络适配器是一种用于执行网络协议处理的某些或者全部处理并由此从主机中卸载处理的网络适配器。由于本发明涉及卸载网络适配器，所以本发明尤其适合于供具有一个或多个网络的分布式数据处理系统使用。图 1-3 是作为可以实现本发明各方面的这种分布式数据处理环境的示例而提供的。应该理解的是，图 1-3 只是示例性的，而且在不脱离本发明的精神和范围的情况下，可以对这些示例性环境做出许多修改。

[0034] 现在参考附图，图 1 描述了可以实现本发明的数据处理系统的网络的图示。网络数据处理系统 100 是一种可以实现本发明的计算机网络。网络数据处理系统 100 包含网络 102，其是用于在网络数据处理系统 100 内连接在一起的计算机与各种装置之间提供通信链路的媒介。网络 102 可以包括诸如有线、无线通信链路或者光纤电缆之类的连接。

[0035] 在所述的例子中，服务器 104 与网络 102 以及存储单元 106 相连。另外，客户端 108、110 和 112 与网络 102 连接。这些客户端 108、110 和 112 例如可以是个人计算机或者网络计算机。在所述的例子中，服务器 104 向客户端 108-112 提供诸如引导文件、操作系统映像和应用程序之类的数据。客户端 108、110 和 112 是服务器 104 的客户端。网络数据处理系统 100 可以包括另外的、未示出的服务器、客户端以及其它装置。在所述的例子中，网络数据处理系统 100 是因特网，网络 102 表示世界范围的、使用传输控制协议 / 网际协议 (TCP/IP) 的 协议组来彼此通信的网关和网络的集合。处于因特网心脏的是位于主节点或者主机计算机之间的高速数据通信线路中枢，其包含数以千计的用于路由数据和消息的商业、政府、教育以及其它计算机系统。当然，网络数据处理系统 100 还可以被实现为多个不

同类型的网络，诸如例如企业内部网、局域网 (LAN) 或者广域网 (WAN) 等。图 1 意在作为一个例子而不是作为对本发明的体系结构的限制。

[0036] 参考图 2，按照本发明的优选实施例描述了可以被实现为服务器、诸如图 1 中的服务器 104 的数据处理系统的框图。数据处理系统 200 可以是包括连接至系统总线 206 的多个处理器 202 和 204 的对称多处理器 (SMP) 系统。作为选择，也可以采用单处理器系统。此外，存储器控制器 / 高速缓存 208 也连接至系统总线 206，其用于提供与本地存储器 209 的接口。I/O 总线桥 210 与系统总线 206 相连，并且提供与 I/O 总线 212 的接口。存储器控制器 / 高速缓存 208 和 I/O 总线桥 210 可以如所描述的那样被集成。

[0037] 连接至 I/O 总线 212 的外围部件互联 (PCI) 总线桥 214 提供了与 PCI 本地总线 216 的接口。多个调制解调器可以连接至 PCI 本地总线 216。典型的 PCI 总线实现方式将会支持四个 PCI 扩展槽或者内插式 (add-in) 连接器。与图 1 中的客户端 108-112 的通信链路可以通过经由内插式连接器连接至 PCI 本地总线 216 的调制解调器 218 和网络适配器 220 来提供。

[0038] 附加的 PCI 总线桥 222 和 224 为附加的 PCI 本地总线 226 和 228 提供了接口，借此，使附加的调制解调器或者网络适配器得以支持。依照此方式，数据处理系统 200 允许与多个网络计算机进行连接。存储器映射的图形适配器 230 和硬盘 232 可以如所描述的那样直接或者间接地与 I/O 总线 212 相连。

[0039] 本领域普通技术人员将会理解的是，图 2 中描述的硬件可以改变。例如，除了所描述的硬件之外，或者作为对它们的替代，可以使用诸如光盘驱动器等的其它外围设备。所描述的示例不意味着隐含对本发明的体系结构的限制。

[0040] 图 2 中描述的数据处理系统例如可以是运行高级交互执行体 (AIX) 操作系统或者 LINUX 操作系统的 IBM eServer pSeries 系统，它是位于纽约的 Armonk 的国际商业机器公司 (IBM) 的产品。

[0041] 现在参考图 3，描述了用于图示说明可以实现本发明的数据处理系统的框图。数据处理系统 300 是客户端计算机的示例。数据处理系统 300 采用外围部件互联 (PCI) 本地总线体系结构。虽然所述的例子采用了 PCI 总线，但是也可以使用诸如加速图形端口 (AGP) 和工业标准体系结构 (ISA) 之类的其它总线体系结构。处理器 302 和主存储器 304 经由 PCI 桥 308 与 PCI 本地总线 306 相连。PCI 桥 308 还可以包括用于处理器 302 的集成的存储器控制器和高速缓冲存储器。与 PCI 本地总线 306 的附加连接可以通过直接部件互连或者通过内插板来进行。在所述的例子中，局域网 (LAN) 适配器 310、SCSI 主机总线适配器 312 和扩展总线接口 314 通过直接部件连接来与 PCI 本地总线 306 相连。与之不同的是，音频适配器 316、图形适配器 318 和音频 / 视频适配器 319 通过插入到扩展槽中的内插板来与 PCI 本地总线 306 相连。扩展总线接口 314 为键盘和鼠标适配器 320、调制解调器 322 和附加存储器 324 提供连接。小型计算机系统接口 (SCSI) 主机总线适配器 312 为硬盘驱动器 326、磁带驱动器 328 和 CD-ROM 驱动器 330 提供连接。典型的 PCI 本地总线实现方式将会支持三个或者四个 PCI 扩展槽或者内插式连接器。

[0042] 操作系统在处理器 302 上运行，并且用来协调和提供对图 3 中的数据处理系统 300 内的各种部件的控制。所述操作系统可以是从微软公司购买到的操作系统，诸如 Windows XP。诸如 Java 之类的面向对象的编程系统可以结合操作系统来运行，并且提供从数据处理

系统300上执行的Java程序或者应用程序对操作系统的调用。“Java”是Sun Microsystems公司的注册商标。操作系统、面向对象的编程系统和应用程序或者程序的指令位于诸如硬盘驱动器326之类的存储设备上，并且可以被载入到主存储器304中以便由处理器302执行。

[0043] 本领域普通技术人员将会理解的是，图3中的硬件可以取决于实 现方式而有所改变。除了图3中所描述的硬件之外，或者作为对它们的替代，可以使用诸如快闪只读存储器(ROM)、等效的非易失性存储器或者光盘驱动器等之类的其它内部硬件或者外围设备。同时，本发明的处理过程可以应用于多处理器数据处理系统。

[0044] 作为另一个例子，数据处理系统300可以是一个独立的系统，其被配置为能够不依赖于某些类型的网络通信接口而进行引导。作为一个进一步的示例，数据处理系统300可以是个人数字助理(PDA)装置，其具备ROM和/或快闪ROM，以便提供非易失性存储器来存储操作系统文件和/或用户生成的数据。

[0045] 图3中所述的例子和上述例子不意味着隐含对体系结构的限制。例如，除了采取PDA的形式以外，数据处理系统300还可以是笔记本计算机或者手持式计算机。数据处理系统300还可以是一个信息站(kiosk)或者Web设备。

[0046] 现在转向图4，按照本发明的优选实施例描述了网络适配器的简图。网络适配器400可以被实现为图2中的网络适配器220、图3中的LAN适配器310等。如图所示，网络适配器400包括以太网接口402、数据缓冲器404和PCI总线接口406。这三个部件提供了网络和数据处理系统的总线之间的通路。以太网接口402提供了与连接至数据处理系统的网络的接口。PCI总线接口406提供了与总线的接口，所述总线诸如是PCI总线216或者306。数据缓冲器404用来存储经由网络适配器400传输和接收的数据。此数据缓冲器还包括与SRAM接口的连接以便提供附加的存储。

[0047] 网络适配器400还包括电可擦可编程只读存储器(EEPROM)接口408、寄存器/配置/状态/控制单元410、振荡器412和控制单元414。EEPROM接口408提供与EEPROM芯片的接口，该芯片可以包含用于网络适配器400的指令以及其它配置信息。不同的参数和设置可以经由EEPROM接口408存储在EEPROM芯片上。寄存器/配置/状态/控制单元410提供了用于存储用来在网络适配器400上配置和运行处理的信息的地方。例如，用于定时器的定时器值可以被存储 在这些寄存器内。另外，不同处理的状态信息也可以被存储在此单元中。振荡器412提供了用于在网络适配器400上执行处理的时钟信号。

[0048] 控制单元414控制由网络适配器400执行的不同处理与功能。控制单元414可以采取各种形式。例如，控制单元414可以是处理器或者专用集成芯片(ASIC)。在这些例子中，用于管理数据流控制的本发明的处理由控制单元414执行。如果是作为处理器来实现的，那么用于这些处理的指令可以被存储在可经由EEPROM接口408存取的芯片中。

[0049] 经由以太网接口402在接收操作中接收数据。把此数据存储在数据缓冲器404中，以便跨越PCI总线接口406传送至数据处理系统上。反之，从主机系统接收数据，以便经由PCI总线接口406来传输，并且将其存储在数据缓冲器404中。

[0050] 在常规的数据处理系统中，对经由网络适配器向/从主机系统传输的数据的处理是在主机系统内执行的。图5举例说明了执行TCP/IP协议栈中数据分组的常规处理的方式。如图5所示，应用软件510经由操作系统520和网络适配器530来发送和接收数据。经

由 TCP/IP 协议栈处理数据，是利用操作系统 520 执行 TCP/IP 协议处理来执行的，以便生成格式化的数据分组来进行传输，或者提取数据分组中的数据并将其路由至适当的应用 510。这些操作是在主机系统上的软件中执行的。

[0051] 已格式化的数据分组是经由网络适配器 530 通过硬件发送 / 接收的。所述网络适配器 530 对来自媒体存取控制和物理层的数据分组进行操作。所述媒体存取控制层是用于控制对网络上的物理传输媒介进行存取的服务。MAC 层的功能被嵌入在网络适配器中，并且包括用于标识每一个网络适配器的唯一序号。所述物理层是用于提供服务以便在网络媒介上进行比特传输的层。

[0052] 如图 5 所示，在常规的网络接口中，当将要从主机系统经由网络发送数据时，首先，把所述数据从用户空间中的应用缓冲器 540 复制到锁定的 (pinned) 内核缓冲器 550，并且在网络适配器队列 560 中 生成一个条目，以便对到达网络适配器 530 的数据进行排队用以进行传输。当从网络接收到用于主机系统上的应用 510 的数据时，使用直接存储器存取 (DMA) 操作把所述数据分组写入主机内核缓冲器 540 中。然后，在以后当该应用调用 receive() 时，由主机把所述数据复制到用户空间中的应用缓冲器 540 中。

[0053] 图 6 举例说明了卸载网络适配器处理 TCP/IP 协议栈中的数据分组的方式。如图 6 所示，通常在主机系统的操作系统 620 中执行的 TCP 和 IP 处理被移动，以便使其得以在卸载网络适配器 630 内执行。因此，减少了由主机系统执行的处理，从而使得应用 610 可以更加有效地被执行。

[0054] 采用已知的卸载网络适配器，以上就图 5 所述的缓冲后发送和接收仍是必需的，即便 TCP/IP 栈的处理已经被转入到网络适配器 630 中也一样。也就是说，如图 6 所示，为了从主机系统发送数据分组，首先把数据从用户空间中的应用缓冲器 640 复制到内核缓冲器 650，其中所述数据在网络适配器队列 660 中被排队以便由网络适配器进行处理。同样地，随着数据分组的接收，所述数据被直接存储器存取 (DMA) 至内核缓冲器 650，并且在以后的时间被复制到用户空间中的应用缓冲器 640 中。

[0055] 由此，如同上述的常规情况一样，在已知的卸载网络适配器中，仍需要在用户空间的应用缓冲器 640 和内核空间的内核缓冲器 650 之间复制数据。这种复制操作必须在主机系统中对正发送或接收的每一数据分组执行。与这种复制操作相关联的系统开销降低了主机处理器运行应用程序的有效性。

[0056] 另外，虽然数据分组的 TCP/IP 协议处理可以被卸载到卸载网络适配器 630，但是实际的连接建立和对每个已建立的连接的状态信息维护仍然由主机系统、例如操作系统 620 负责。也就是说，主机仍必须执行必要的操作来建立出站 (outbound) 和进站 (inbound) 连接。另外，当每一连接的状态改变时，主机必须与网络适配器交换消息，以便使主机系统中存储的每个连接的状态信息得以维护。

[0057] 因此，虽然把 TCP/IP 协议处理从主机系统卸载至网络适配器已经改进了计算系统的吞吐量，但是通过改进在这种卸载网络适配器系统中管理存储器的方式，并且改进建立连接的方式，以便卸载连接建立并且最小化主机和网络适配器之间的消息传送，可以获得额外的改进。另外，通过改进在卸载网络适配器中接收数据的方式以便使网络适配器和主机系统之间的交互最小化，可以获得网络适配器操作方面的改进。

[0058] 本发明提供了用于改进卸载网络适配器的操作以便使主机系统和网络适配器之

间的交互最小化的机制。本发明提供了一种位于主机系统的操作系统和卸载网络适配器之间的改进的接口。此接口包括控制部分和数据部分。所述接口利用了与显式数据结构一起使用的缓冲器队列，所述显式数据结构用于表明接口的控制部分和数据部分。接口的控制部分允许主机系统向卸载网络适配器发布命令，并且允许卸载网络适配器向主机系统发布命令。例如，就将要监听哪些端口号，所述主机系统可以向网络接口发布命令，并且就新连接的建立、数据的接收等，卸载网络适配器可以向主机系统发布命令。所述接口的数据部分提供了一种用于在已建立的连接上传送数据的机制，以便既用于发送又用于接收。所述接口的控制部分可以通过使用用于控制连接的常规套接字应用编程接口 (API)、例如 socket()、bind()、listen()、connect()、accept()、setsockopt() 等来启用。所述接口的数据部分可以通过用于发送或者接收数据的套接字 API、例如 send()、sendto()、write()、writev()、read()、readv() 等来启用。

[0059] 图 7 是举例说明使用本发明的卸载网络适配器编程接口在主机系统和卸载网络适配器之间通信的示例性简图。所述卸载网络适配器编程接口在主机系统和主要基于直接存储器存取 (DMA) 操作或 DMA 的卸载网络适配器之间提供了一个通信接口，以便在主机系统上的 I/O 可存取的存储器的保留部分中写入并且从中读取请求和响应描述符。

[0060] 如图 7 所示，所述主机系统 710 提交用于向卸载网络适配器 730 传送数据或从中传送数据的请求，并且卸载网络适配器 730 用请求成功或者失败的通知来做出响应。把请求和响应打包为被称为请求描述符和响应描述符的数据结构。把所述描述符写入到主机系统 710 上的 I/O 可存取的存储器 720 中的两个物理区域中并且从中进行读取。这些区域被称为输入描述符表 722 和输出描述符表 724，并且按照生产方 - 消费方的方式来使用。

[0061] 所述输入描述符表 722 由卸载网络适配器 730 读取，并且由主机系统 710 写入以便提交控制和数据接口请求。所述输出描述符表 724 由主机系统 710 读取并且由卸载网络适配器 730 写入，其中卸载网络适配器 730 使用输出描述符表 724 来表明先前请求的结果并且向主机系统 710 通知数据到达。

[0062] 虽然主机系统 710 和卸载网络适配器 730 都可以从这些描述符表 722 和 724 中进行读取并且向其中进行写入，但是它们不以相同的方式来存取描述符。所述主机系统 710 使用常规的存储器读写来存取描述符表 722 和 724。然而，所述卸载网络适配器使用 DMA 操作来向 / 从描述符表 722 和 724 复制任意的描述符集合。

[0063] 如同常规的网络适配器一样，所述主机系统 710 例如可以通过轮询或者接收中断来从卸载网络适配器 730 获悉输出描述符表 724 中的新的响应描述符。也就是说，当在卸载网络适配器中接收到数据分组，并且对于向主机系统 710 进行的数据分组到达通知而言、某些标准得以满足时，如此后更加详细描述的那样，可以由卸载网络适配器 730 生成响应描述符，并且其被写入到输出描述符表 724 中。然后，可以由操作系统 715 接收到中断，以表明输出描述符表 724 中的新描述符。作为选择，所述主机系统 710 可以周期性地轮询输出描述符表 724 以确定是否有新的描述符。如果输出描述符表 724 有溢出的危险，那么卸载网络适配器 730 可以对主机系统 710 发起中断，以便通知它这种情况。

[0064] 在本发明的一个示例性实施例中，被写入到描述符表 722 和 724 中的描述符是 256 比特 /32 字节，并且其被构造如下 ：描述符所有者 (1 比特)、描述符类型 (5 比特)、描述符内容 (250 比特)。所有者比特用于描述符表 722 和 724 中的描述符的生产方 / 消费方关

系。换言之，由于存在两个进行通信的部件，例如主机操作系统和卸载网络适配器，所以存在生产方 / 消费方关系。可以使用单个比特来表示描述符的所有权。例如，“1”可以表示主机生成的描述符，而“0”可以表示卸载网络适配器生成的描述符，反之亦然。

[0065] 所述描述符类型用来标识与描述符相关联的操作和 / 或请求。例如，请求描述符可以包含如下类型之一 :缓冲发送 (buffer send)、缓冲器可用 (buffer available)、连接请求、终止请求、监听请求、取消请求、连接属性控制和网络适配器属性控制。

[0066] 缓冲发送描述符类型与请求分配用于存储待发送数据的缓冲器的请求相关联，并且标识所述缓冲器、待使用的连接标识符和 ASAP 比特值，将在下文中描述这部分内容。缓冲器可用描述符类型与请求分配用于存储已接收数据的缓冲器的请求相关联，并且标识用于存储已接收数据的缓冲器和经由其接收数据的连接标识符。连接请求描述符类型与请求启动在指定的本地端口和协议上的连接的请求相关联。终止请求描述符类型与请求卸下指定连接的请求相关联。监听请求描述符类型与表明自愿接收端口和协议上的连接的请求相关联。取消请求描述符类型与请求取消先前提交的发送、连接或者监听请求的请求相关联。连接属性控制描述符类型与请求获得或者设置连接属性的请求相关联。网络适配器属性控制描述符类型与请求获得或者设置网络适配器范围的属性的请求相关联。

[0067] 响应描述符也可以具有各种类型。例如，响应描述符可以是如下类型之一 :缓冲接收 (buffer receive)、缓冲器可用、连接到达、连接完成、监听响应、终止响应、取消响应、连接属性和网络适配器属性。缓冲接收描述符类型标识具有可用数据的缓冲器，并且标识数据用于哪个连接。缓冲器可用描述符类型用于标识 DMA 已完成并且发送用缓冲器可用。连接到达描述符类型向主机通知新的连接已经到达，并且包括连接标识符。连接完成描述符类型向主机通知连接请求 已经成功或者失败。监听响应描述符类型表明已提交的监听请求的成功 / 失败。终止响应描述符类型表明已提交的关闭请求的成功 / 失败。取消响应描述符类型表明已提交的取消请求的成功 / 失败。连接属性描述符类型表明旧的连接属性值或者新的值成功 / 失败。网络适配器属性描述符类型表明旧的网络适配器属性值或者新的网络适配器属性值成功 / 失败。

[0068] 在本发明的一个示例性实施例中，用于缓冲发送请求、缓冲器可用请求、缓冲接收响应和缓冲器可用响应描述符的描述符内容字段全部被格式化为具有如下字段 :

[0069]	Base	64 比特	缓冲器的基本物理地址
[0070]	Len	32 比特	以字节为单位的缓冲器长度
[0071]	Conn ID	64 比特	由网络适配器给出的唯一的连接标识符
[0072]	ASAP	1 比特	尽可能快地请求 DMA(此后论述)
[0073]	Modify	1 比特	表明此缓冲器是否已经被修改 (此后论述)

[0074] 连接 ID(Conn ID) 是用于唯一地标识连接的值，并且由卸载网络适配器响应于连接请求而且作为连接到达的响应而提供。连接 ID 0(零) 被保留以便意味着“不连接”。使用它来表明例如缓冲器可用于任何连接 (例如，用于仍没有 ID 的被动接受的连接上的数据)。不与任何特定连接相关联的缓冲器被称为“批缓冲器 (bulk buffer)”。

[0075] ASAP 和 modify 字段只用于缓冲发送请求描述符。ASAP 比特表明希望让此缓冲器尽快地 DMA。modify 比特用于向卸载网络适配器通知自从它上一次被提供给卸载网络适配器以来此特定的缓冲器是否已经改变。这样做允许所述卸载网络适配器确定在本地存储器

中它是否早已具有此缓冲器的拷贝，由此能够避免 DMA 传送。

[0076] 控制描述符描述了控制缓冲器，该控制缓冲器包含可变数目的任意长度的属性元组。用于控制描述符、连接请求、终止请求、监听请 求、取消请求和它们的各自响应的描述符内容字段全部被格式化为具有如下字段：

[0077] Number 8 比特 控制缓冲器中的属性元组数目

[0078] Base 64 比特 控制缓冲器的基本物理地址

[0079] Len 32 比特 以字节为单位的控制缓冲器长度

[0080] Conn ID 64 比特 唯一的连接标识符

[0081] 用于连接属性请求、卸载网络适配器属性请求和它们的各自响应的控制缓冲器和描述符内容字段全部被格式化为具有如下字段：

[0082] Get/Set 1 比特 表明属性是要被检取还是要被更新

[0083] Attribute 15 比特 标识用于读 / 写的属性

[0084] Length 32 比特 属性数据的长度

[0085] Value N/A 实际属性值，由 prev.field 指定的长度

[0086] 上述控制描述符具有尽可能通用的含义。因为可以由控制描述符指定大量属性，所以无法在此对其进行全部举例说明。网络接口控制属性的示例包括 IP 地址、域名和路由信息。每一连接的控制属性的示例包括接收窗口大小、Nagle 算法设置和 SACK 支持。

[0087] 采用本发明，所述卸载网络适配器 730 具有卸载网络适配器 730 的诸如以固件、ASIC 等形式的逻辑，其用于利用本发明的卸载网络适配器编程接口。也就是说，所述卸载网络适配器 730 具有用于识别请求描述符、处理请求描述符和相应数据的逻辑，以及用于生成将被写入到输出描述符表 724 中的响应描述符的逻辑。同样地，主机系统的操作系统 715、由操作系统 715 加载的设备驱动程序等具有用于生成将被写入到输入描述符表 722 中的请求描述符、识别从输出描述符表 724 中读取的响应描述符的逻辑，以及用于处理响应描述符以及相 应数据的逻辑。

[0088] 已经给出了使用本发明的卸载网络适配器编程接口的描述符在主机系统和网络适配器之间进行交互的总体概述，随后的描述将举例说明这个接口如何使用卸载网络适配器来有助于改进的连接建立、存储器管理以及数据接收。

[0089] 连接建立

[0090] 本发明的一个方面在于能够把连接建立和连接状态信息的维护卸载给卸载网络适配器。作为这种卸载连接建立和状态信息维护的结果，主机系统和卸载网络适配器之间所需的通信次数可以得以减少。另外，如此后将论述的那样，把这些功能卸载给所述卸载网络适配器，允许向主机系统成批通知已建立的连接和状态信息，而不是像已知计算系统中出现的那样逐个地通知。

[0091] 图 8 是按照本发明一个示例性实施例当建立通信连接时、在主机系统和卸载网络适配器之间进行的通信的示例性简图。如图 8 所示，出站连接的建立是通过操作系统 815 接收到来自应用 805 的用于请求建立连接的请求而启动的。因此，操作系统 815 生成连接请求描述符，并且将其写入到输入描述符表 822。所述连接请求描述符和相关联的控制缓冲器包括建立所请求的连接所需要的全部信息。例如，所述控制缓冲器和连接请求描述符可以包含 AF_INET、SOCK_STREAM、IP VERSION 信息和连接标识符，以便参考远程和本地连接。

[0092] 所述卸载网络适配器 830 从输入描述符表 822 中读取连接请求描述符，然后卸载网络适配器 830 内的连接建立逻辑 832 根据连接请求描述符中接收到的信息来试图建立连接。根据连接请求描述符建立连接的处理，包括建立用于连接的套接字描述符，即，用于描述主机系统和远程计算设备的套接字、把连接标识符与所述连接相关联并且分配卸载网络适配器 830 中的缓冲器用于该连接的数据结构。也就是说，所述卸载网络适配器可以执行与常规的系统调用 connect()、setsockopt()、bind()、accept() 等相关联的操作。只有当建立了连接或 者满足错误条件（所述错误条件诸如是持续时间超时条件）时，才向主机系统 810 通知连接建立操作所产生的状态。

[0093] 这种响应可以是把一个或多个响应描述符写入到输出描述符表 824。例如，连接完成描述符可以由卸载网络适配器 830 生成，并且被写入到输出描述符表 824，以便由此向主机系统 810 通知连接已经被建立。

[0094] 进站连接的建立是依照稍微不同的方式来执行的。如果应用程序要求“监听”特定端口上的连接的能力，那么操作系统 815 可以把监听请求描述符写入至输入描述符表 822。所述监听请求描述符标识在其上进行监听的端口以及用于要监听的连接的协议。卸载网络适配器 820 的连接建立逻辑 832 然后从输入描述符表 822 中读取监听请求描述符，并且执行必要的操作以便在适当的输入套接字连接上建立连接。此操作例如可以包括执行类似于常规的 accept() 和 bind() 系统调用的操作，但是，是在卸载网络适配器 830 内执行它们。只有当建立了连接或者满足错误条件（所述错误条件诸如是持续时间超时条件）时，才向主机系统 810 通知连接的所产生的状态。在已知的“卸载”实现方式中，所述主机系统在连接建立的每一阶段上进行交互。本发明发布高级命令来进行连接或监听连接，并且只有当建立了连接或者满足了超时或错误条件时才做出应答。

[0095] 当建立连接时，在卸载网络适配器的存储器 834 中依照连接状态数据结构来保存与连接有关的信息。此状态信息用来经由已建立的连接发送并且接收数据。此状态信息也可以用来更新由主机系统 810 维护的连接状态信息，如此后论述的那样。

[0096] 正如可以从上述描述中看到的那样，在卸载网络适配器内执行连接建立操作并且使用本发明的卸载网络适配器编程接口的关键结果之一在于：主机系统和网络适配器之间的通信在连接建立期间被最小化了。因此，主机系统处理的信息较少。当主机系统是利用其建立和卸下大量连接的服务器计算系统时，这是尤其重要的。

[0097] 如上所述，在本发明的一个实施例中，可以向主机系统通知建立 连接或者遇到错误条件之后的连接状态。由此，作为结果，每当建立了连接或者试图建立连接失败时，把连接完成响应描述符写入到输出描述符表 824。通过把每一连接完成响应描述符写入到输出描述符表 824，可以生成一个中断，并且将其发送至操作系统 815，以便通知主机系统 810 在输出描述符表 824 中存在新的响应描述符要处理。

[0098] 为了最小化把连接完成响应描述符写入到输出描述符表 824 的次数，并且由此最小化所生成的而且被发送至主机系统 810 的中断数目，本发明可以依照多种不同的方式延迟把连接完成响应描述符写入到输出描述符表 824。延迟向主机通知连接建立状态的优点在于：可能在单个通知中合并多个连接。以这种方式，用于相同或者不同连接的多个完成响应描述符可以被“分批组合”在一起，并且在卸载网络适配器和主机系统之间的一个事务中被提供至主机系统。

[0099] 例如，可以根据建立套接字连接的速率，接收连接请求的速率等，来设置可配置的延迟值。这个延迟值可以标识在生成连接完成响应描述符之前在卸载网络适配器 830 中可以累积的连接建立信息的合并量，该连接完成响应描述符用于指明合并内的每一连接的状态。此值可以被存储在卸载网络适配器 830 上的存储器中。

[0100] 所述延迟值可以被静态地或者动态地确定，并且可以采取如下的形式，即：建立连接与使用连接完成响应描述符向主机系统进行通知之间的预定时间量，所接收的多个连接建立状态更新，即，连接建立操作的成功 / 失败等。如果所述延迟值被动态地确定，那么它例如可以根据在一定时段内接收的连接的速率和数量、套接字连接时序的历史观察数据等来加以确定。例如，如果一个特定的套接字接收连接具有在 10 毫秒上有 10 个连接请求然后平静达 10 秒的脉冲串，那么可能谨慎的是，延迟向主机系统的所有通知，直到 10 个连接完成，以便减少向主机系统的总体通知。可将 1 秒的超时特征用来等待附加的套接字连接。

[0101] 用于确定何时把连接完成响应描述符写入到输出描述符表 824 的另一选择是使卸载网络适配器 830 等待已建立的连接的单元数据到 达。以这种方式，所述卸载网络适配器 830 在存储器中保存与已建立连接有关的信息，直到数据被接收以便由主机系统 810 处理为止。在这时，连接完成响应描述符可以被写入到输出描述符表 824 中，以向主机系统 810 通知连接的建立，然后缓冲接收响应描述符可以被写入到输出描述符表 824 中，以表明经由已建立的连接接收到数据。

[0102] 在本发明的又一个实施例中，经由输出描述符表 824 向主机系统进行的通知可以被延迟，直到经由连接接收到特定的数据模式为止。这些特定的数据模式例如可以是特定的 HTTP GET 请求、特定的元标签等，所述特定的元标签被预先确定以便表明可以作为单个单元处理的数据序列的结束。

[0103] 一旦经由已建立的连接接收了此数据模式，卸载网络适配器 830 就可以把连接完成响应描述符写入到输出描述符表 824，以标识在接收到该数据模式之前的时段期间已成功建立或者失败了的所有连接。以这种方式，直到主机系统 810 具有特定的数据要处理时，才会向主机系统 810 通知新的连接的建立。换言之，所述主机系统不会因要处理的描述符而增加负担，除非存在专门要主机系统去执行的某些事情。所述“某些事情”是由正在被搜索的数据模式来定义的。

[0104] 由此，本发明允许合并在建立连接时已建立的连接或连接失败的通知，以便使发送至主机系统的通知数目最小化。这样做减轻了必须由主机系统执行的处理量，并且允许主机系统使用其资源来处理在主机系统上运行的应用程序。

[0105] 采用本发明，由于连接建立由所述卸载网络适配器 830 来执行，所以已建立的连接的状态被保存在卸载网络适配器 830 的存储器中。然而，如果出现故障转移、网络错误条件，那么主机系统 810 需要具有这个状态信息，或者需要做出路由判断。因此，本发明提供了一种用于把保存在卸载网络适配器 830 中的已建立连接的状态信息转移至主机系统 810 的机制。

[0106] 在本发明的一个示例性实施例中，连接属性响应描述符可以被周期性地生成并且被写入到输出描述符表 824。此连接属性响应描述符 标识每一个连接的当前状态。通过向操作系统 815 发送中断，来向主机系统 810 通知把连接属性响应描述符添加至输出描述符表 824。然后，所述主机系统 810 读取连接属性响应描述符，并且对它进行处理，以便使主机

系统的连接状态信息被更新。由此，主机系统 810 具备已更新的信息，借此，如果发生网络错误或者故障转移，则主机系统 810 可以做出路由判断，并且执行适当的操作。

[0107] 由此，本发明提供了用于把连接建立卸载给卸载网络适配器以便在建立连接期间使主机系统和卸载网络适配器之间的通信最小化的机制。这样做可以允许主机系统在单个连接请求描述符中向卸载网络适配器发送成批的连接建立请求，然后卸载网络适配器不必再与主机系统进行进一步的通信，直到满足了某些标准为止，其中所述标准例如是建立了预定数目的连接、在连接上到达了预定量的数据、过去了预定量的时间、接收到了预定的数据模式等。同样地，主机系统可以命令卸载网络适配器监听特定端口上的连接，然后接受并且绑定这些连接。因此，就正在监听的端口上的连接建立而言，所述主机系统可以发送一个监听请求描述符，并且不会再次与其通信，直到预定的标准得以满足为止。另外，本发明提供了一种用于在卸载网络适配器中存储连接状态信息，然后把此状态信息转移至主机，以便用于路由判断以及用于发生网络错误或者故障转移的情况的机制。

[0108] 图 9 和 10 是概述按照本发明一个示例性实施例的本发明的元件操作的流程图。应当理解的是，这些流程图中的每个块、此后描述的其它流程图以及流程图中的块的组合可以通过计算机程序指令来实现。这些计算机程序指令可以被提供给处理器或其它可编程数据处理设备，以便产生一种机制，使得在该处理器或者其它可编程数据处理设备上执行的指令创建用于实现流程图的一个或多个块中所指定的功能的装置。这些计算机程序指令还可以被存储在计算机可读存储器或者存储介质中，其可以引导处理器或其它可编程数据处理设备依照特殊的方式来起作用，从而使得存储在计算机可读存储器或存储介质中的指令产生一种制造物品，所述制造物品包括用于实现流程图的一 个或多个块中所指定的功能的指令装置。

[0109] 因此，流程图中的各块支持用于执行特定功能的装置的组合、用于执行特定功能的步骤的组合、以及用于执行特定功能的程序指令装置。还将会理解的是，流程图中的每个块和流程图中各块的组合可以通过基于专用硬件的用于执行特定功能或者步骤的计算机系统来实现，或者通过专用硬件和计算机指令的组合来实现。

[0110] 图 9 是概述当使用卸载网络适配器建立连接时本发明的主机系统的示例性操作的流程图。如图 9 所示，所述操作通过从一个应用接收连接建立请求而开始（步骤 910）。此连接建立请求例如可以是用于建立特定连接的请求或者用于在特定端口监听连接的请求。把连接建立请求描述符写入到输入描述符表（步骤 920）。此连接建立请求描述符例如可以是连接请求描述符或者监听请求描述符。

[0111] 然后，所述操作等待来自卸载网络适配器的关于连接建立操作完成的响应（步骤 930）。“等待”意味着 :就此连接而言，主机系统不再执行进一步的操作，直到接收到响应为止。显然，当出现这种“等待”时，主机系统正在执行其它操作。

[0112] 就是否已经接收到响应做出确定（步骤 940）。如果没有，那么就连接建立请求是否已经超时做出确定（步骤 950）。如果没有，则操作返回到步骤 930 并且继续等待。如果连接建立请求已经超时，那么把取消请求描述符写入到输入描述符表（步骤 960），并且所述操作终止。

[0113] 如果接收到响应，那么从输出描述符表中读取连接完成响应描述符（步骤 970）。然后，由主机系统处理所述连接完成响应描述符（步骤 980），并且所述操作终止。

[0114] 应该注意的是，在步骤 920 中被写入到输入描述符表的原始连接建立请求描述符可以指明要建立多个连接，即，一个成批连接建立请求。由此，采用本发明，所述主机只需利用输入描述符表进行一个事务处理，以便执行这种成批连接建立，其中建立这些连接所需的全部处理都被卸载给所述卸载网络适配器。同样地，如果原始连接建立请 求描述符是“监听”请求描述符，那么可以在卸载网络适配器监听端口时，建立许多连接，然而主机系统只执行一个事务来启动这些连接的建立。

[0115] 图 10 是概述当按照本发明的一个示例性实施例建立连接时卸载网络适配器的示例性操作的流程图。如图 10 所示，所述操作通过从输入描述符表中读取连接建立请求而开始（步骤 1010）。执行连接建立操作，以生成套接字描述符、连接标识符等，以便建立在连接建立请求描述符中标识出的一个或多个连接（步骤 1020）。把涉及每个已建立连接的状态信息连同用于标识自从前一次向主机系统进行通知以来、已经建立了哪些连接和哪些连接已经失败的信息存储在存储器中（步骤 1030）。

[0116] 对于写入连接完成响应描述符而言，就是否已经满足延迟标准做出确定（步骤 1040）。如上所述，所述延迟标准可以采取多种不同的形式。例如，所述延迟标准可以是自从上一次向主机系统发送通知以来建立了多个连接，经由多个连接之一到达了预定量数据，正在接收特定数据模式，自从上一次向主机系统进行通知以来过去了预定量时间等。

[0117] 如果延迟标准尚未得以满足，那么操作返回到步骤 1020，并且继续建立连接，并且在存储器中保存连接建立信息和状态信息。如果延迟标准已经得以满足，那么生成连接完成响应描述符并且将其写入到输出描述符表，用以标识自从上一次向主机系统发送通知以来所建立的连接以及未能建立的连接（步骤 1050）。然后，所述操作终止。

[0118] 由此，本发明提供了一种使用卸载网络适配器来建立连接的改进机制。本发明的此方面尤其适合于成批连接建立，这是因为，主机系统和卸载网络适配器之间的通信被最小化，从而使得只利用主机系统和卸载网络适配器之间的最小量的交互即可建立许多连接。这样做使主机系统解放出来从而可以将其资源集中于运行应用程序和执行其它有用工作上。

[0119] 存储器管理

[0120] 除了连接建立以外，本发明还对使用卸载网络适配器的数据处理系统中的存储器管理加以改进。根据本发明的存储器管理允许数据的缓冲后发送和接收，而且允许数据的零拷贝发送和接收。另外，本发明允许根据任意数目的属性来对可以在指定的连接当中共享的 DMA 缓冲器进行分组。本发明还允许部分发送和接收缓冲操作，延迟 DMA 请求，以便使其可以被成批地传递至主机系统，并且本发明还提供了一种用于加快向主机系统传送数据的机制。

[0121] 所述卸载网络适配器编程接口支持常规的用户级应用编程接口 (API)，诸如套接字接口，以及允许更加直接地对用户存储器进行存取的更新的 API。本发明的卸载体系结构允许数据的缓冲后发送和接收，而且还允许数据的零拷贝发送和接收。从卸载网络适配器的角度看，缓冲后传输和零拷贝传输几乎被同等地处理。区别这两种数据传送的方式取决于主机系统如何使用卸载网络适配器。

[0122] 图 11 是举例说明依照本发明的、其中使用数据的缓冲后发送和接收的存储器管理机制的示例性简图。为了便于描述，假定主机系统 1110 和其它计算设备（未示出）之间

的连接已经通过上述机制建立了。当参考此连接进行 read() 调用时，可以为此连接建立应用缓冲器 1130。所述操作系统 1150 还可以包括锁定的内核缓冲器 1140，其可以被称为批缓冲器，用于接收各种连接的数据，在把数据发送至网络适配器或者特定的连接缓冲器、例如应用缓冲器 1130 之前，将数据写入其中。所述内核缓冲器 1140 在连接发布时被创建，并且当在一个连接上发送数据之前没有发布 (post) 用于该连接的应用缓冲器 1130 时，使用所述内核缓冲器 1140。如果应用缓冲器 1130 在发送数据以前被发布了，那么应用缓冲器可用来接收所述数据。作为选择，如此后将论述的那样，应用缓冲器 1130 和内核缓冲器 1140 都可用于某些缓冲后传输的实施例中。

[0123] 如图 11 所示，当主机系统 1110 希望经由卸载网络适配器 1120 向其它计算设备发送数据时，主机系统 1110 把来自用户空间的应用 缓冲器 1130 的数据复制到操作系统内核空间中的操作系统 1150 的锁定的内核缓冲器 1140。这个锁定的内核缓冲器 1140 是用于从卸载网络适配器 1120 和一个或多个已建立连接的应用缓冲器 1130 接收数据的批缓冲器。由此，如果目前开启了多个连接，并且这些连接的数据可以经由锁定的内核缓冲器 1140 进行发送 / 接收，那么所述主机系统 1110 可以具有多个应用缓冲器 1130。

[0124] 以这种方式，所述数据被排队以便由卸载网络适配器 1120 传输。当具有要发送的数据时，所述主机系统 1110 然后可以在输入描述符表上发布缓冲发送描述符，以标识锁定的内核缓冲器 1140。响应于从输入描述符表中读取缓冲发送请求描述符，所述卸载网络适配器 1120 然后可以从锁定的内核缓冲器 1140 中读取数据，并且可以经由网络（未示出）把数据传输至目的地计算设备。此后，卸载网络适配器 1120 可以在输出描述符表上发布缓冲器可用响应描述符，以表明数据传输已经完成。由此，通过使用缓冲后传输机制进行数据发送，本发明把来自应用缓冲器 1130 的数据复制到锁定的内核缓冲器 1140 以便进行传输。

[0125] 缓冲后接收按类似方式进行。采用缓冲后接收操作，卸载网络适配器 1120 执行直接存储器存取 (DMA) 操作，以便把数据从卸载网络适配器 1120 传输到锁定的内核缓冲器 1140 中。响应于主机系统 1110 在输入描述符表上发布缓冲器可用请求描述符，所述卸载网络适配器 1120 可以在输出描述符表上发布缓冲接收响应描述符。然后，主机系统 1110 可以从输出描述符表中读取缓冲接收响应描述符，并且可以调用 read() 套接字调用，以便把数据从锁定的内核缓冲器 1140 复制到用户空间中的应用缓冲器 1130。

[0126] 由于把数据从应用缓冲器 1130 传送至锁定的内核缓冲器 1140 或者反过来把数据从锁定的内核缓冲器 1140 传送至应用缓冲器 1130 所必须执行的数据复制操作的数目，所以缓冲后传送势必比最优情况要慢。然而，缓冲后传送提供了两个优点。因为数据被保留在主机内核存储器中，即，被保留在锁定的内核缓冲器 1140 中，所以因直到 将要发送它们时卸载网络适配器 1120 才会对缓冲器进行 DMA，故而降低了卸载网络适配器 1120 上的存储器压力。另外，因为如果卸载网络适配器 1120 失败，那么在主机系统的锁定的内核缓冲器中数据仍然可用于经由其它网络适配器进行发送，所以更加易于实现故障转移。

[0127] 本发明的体系结构还提供了一种在卸载网络适配器和主机系统之间进行数据的零拷贝传输的机制。术语“零拷贝”指的是消除由主机系统执行的存储器至存储器的复制。图 12 是根据本发明一个示例性实施例举例说明零拷贝操作的示例性简图。为了向 / 从主机系统 1210 传输数据，主机系统 1210 可以阻断用户应用，并且锁定其应用缓冲器 1230。所述主机系统 1210 然后可以启动卸载网络适配器 1220，以便把数据直接 DMA 至应用缓冲器

1230 或者把数据从应用缓冲器 1230 直接 DMA 至卸载网络适配器 1220。

[0128] 在当前的系统中，为了从已建立的连接中进行读取，应用程序调用具有三个自变量的 read() 套接字调用。第一个自变量指定要使用的套接字描述符，第二个自变量指定应用缓冲器 1230 的地址，而第三个自变量指定缓冲器的长度。读取过程提取已经到达该套接字的数据字节，并且把它们复制到用户的缓冲区，例如应用缓冲器 1230。如果已经到达的数据比适合用户缓冲区的数据要少，那么 read() 提取所有数据并且返回它所找到的字节数目。

[0129] 通过在根据本发明的系统中采用零拷贝，应用缓冲器 1230、即 DMA 缓冲器的创建使得生成一个描述符通信分组并且将其从主机系统 1210 发送至卸载网络适配器 1220，例如，可以生成一个缓冲器可用请求描述符通信分组并且将其发布给输入描述符表。所述描述符描述了应用缓冲器 1230 及其属性，并且把应用缓冲器 1230 与已建立连接的连接信息相关联。当所述应用缓冲器可以用于卸载网络适配器 1220 时，并且当执行 read() 套接字调用时，执行 DMA 操作以便把数据从卸载网络适配器 1220 传送至应用缓冲器 1230。然后，创建来自卸载网络适配器 1220 的响应描述符，用于描述 read() 调用完成通知所 需的 DMA 数据属性，例如，可以生成缓冲器可用响应描述符并且将其发布给主机系统的输入描述符表。

[0130] 应该注意的是，卸载网络适配器 1220 在存储器中保存每个开启的连接的信息，以便用于执行其功能。此信息可以包括与开启的连接相关联的应用缓冲器的标识以及其它连接专用信息。然后，当卸载网络适配器 1220 需要在其自身和主机系统 1210 上的应用之间进行数据通信时，使用此信息。

[0131] 由此，采用本发明，所述卸载网络适配器可以使用直接存储器存取操作把数据直接发送至用户空间中的应用缓冲器。这样做时，可避免把数据从锁定的内核缓冲器复制到应用缓冲器。当然，本发明可以依照任一种模式、即缓冲后发送 / 接收或者零拷贝发送 / 接收来进行操作，或者可互换地或者大致同时地使用这两种模式。也就是说，可以使用缓冲后发送 / 接收在主机系统和卸载网络适配器之间传送某些数据，并且可以使用零拷贝发送 / 接收来传送其它数据。例如，每当应用程序的 read() 调用在接收到套接字上的各个数据之前进行时，可以使用所述零拷贝发送 / 接收。以这种方式，应用缓冲器将被预先发布，以便在已建立的连接上接收数据。如果 read() 调用不是在接收到套接字上的数据之前进行的，那么可以使用缓冲后发送 / 接收。

[0132] 在优选的实施例中，零拷贝发送 / 接收是用于向 / 从主机系统发送 / 接收数据的优选方式。然而，可能出现其中无法进行零拷贝发送 / 接收的情况。例如，如果应用缓冲器的可用存储器将要被超出，或者如果应用缓冲器不可用，那么卸载网络适配器可能不能使用直接存储器存取操作来把数据直接发送至应用缓冲器。因此，可以要求把数据缓冲后发送至共享缓冲器。

[0133] 本发明的卸载网络适配器具有根据任意数目的属性来对可以在指定的连接当中共享的应用缓冲器进行分组的能力。在优选的实施例中，应用缓冲器的分组是基于连接端口号进行的。也就是说，全部使用同一端口号的应用缓冲器可以共享应用缓冲器。例如，在 web 服务方案中，每一端口可能有多个连接。一个示例是 web 服务器的 TCP/IP 端口 80。在端口 80 上可以有数以千计的客户端 HTTP 连接请求信息。分配给端口 80 的缓冲器可以被分组在一起，即，可以建立已分配缓冲器池，以便处理在端口 80 上进入的这些信息请求。

[0134] 在发送操作上共享应用缓冲器，允许为基于主机系统的广播或者多播类型的连接重新使用数据。也就是说，只需要把数据写入共享应用缓冲器一次，但是可以在共享这些应用缓冲器的多个连接之上传输数据。对于已接收的数据共享应用缓冲器，允许为具有低带宽要求或者瞬态的业务脉冲串的有效连接更加有效地利用存储器。也就是说，与必须具有其自身专用的单独应用缓冲器（其中缓冲器的大部分存储空间对于低带宽或者瞬态脉冲串连接而言可能不使用）相比，多个连接可以共享一个更小的共享应用缓冲器。另外，共享应用缓冲器允许单独的应用程序和进程共享已接收的数据。

[0135] 图 13 是根据本发明一个示例性实施例举例说明共享的缓冲器设置的示例性简图。在所述的例子中，在主机系统 1310 上目前运行有三个进程 X、Y 和 Z。已经建立了五个连接 A、B、C、D 和 E，并且在主机系统 1310 中已经为这些连接建立了相对应的应用缓冲器 1350-1370。应用缓冲器 1350 和 1360 是可以使用 DMA 操作把数据直接发送至其中的单独的应用缓冲器。作为选择，如上所述，作为缓冲后发送 / 接收操作的一部分，可以使用锁定的内核缓冲器 1330 来把数据复制到这些应用缓冲器 1350-1360 中。

[0136] 应用缓冲器 1370 是在连接 C、D 和 E 之间共享的共享应用缓冲器。例如，连接 C、D 和 E 可以为其套接字连接全部使用相同的端口号，可以是低带宽连接，并且由此可以共享缓冲器空间。作为选择，连接 C、D 和 E 可以是共享缓冲器 1370 以便多播或广播数据的多播或者广播组的一部分。

[0137] 如图 13 所示，当使用数据的缓冲后发送 / 接收传送时，首先使用 DMA 操作把数据从卸载网络适配器 1320 发送至主机系统 1310 的操作系统 1340 中的锁定的内核缓冲器 1330。响应于主机系统 1310 在输出缓冲器表中发布缓冲器可用请求描述符，所述卸载网络适配器 1320 在输入描述符表中发布缓冲接收响应描述符。然后，所述主机系统 1310 可以调用 read() 来把数据从锁定的内核缓冲器 1330 复制到用于连接 C、D 和 E 的共享应用缓冲器 1370。数据可以由共享该共享应用缓冲器 1370 的一个或多个进程从这些共享应用缓冲器 1370 中读取。例如，进程 Z 可以从共享的缓冲器 1370 中读取数据。监听连接 C、D 或者 E 上的数据的任何进程可以执行这些操作以便从锁定的内核缓冲器 1330 中将其连接上的数据读入到共享的缓冲器 1370。

[0138] 作为选择，如同单独的应用缓冲器 1350 和 1360 那样，连接 C、D 和 E 的数据可以从卸载网络适配器 1320 被直接 DMA 到共享的缓冲器 1370 中。以这种方式，本发明的零拷贝实现方式可以利用共享的缓冲器 1370 来保存数据以便从多个连接进行发送 / 接收。

[0139] 共享的缓冲器 1370 尤其有用的一种情况是：当卸载网络适配器 1320 需要在应用程序已经建立了用于在其中接收数据的应用缓冲器之前向主机系统 1310DMA 数据时。例如，这种情况可能会出现在当在卸载网络适配器 1320 上持续接收的数据超出预定阈值并且卸载网络适配器可能存在存储器用完了的危险时。考虑到可能存在这种情况，把数据复制到主机存储器中的共享系统缓冲器 1370 中的中间复制操作往往有助于减轻这种情况。也就是说，对于所有开启的连接而言，可以把数据复制到共享的缓冲器 1370 中，而不是复制到专用的连接应用缓冲器、诸如缓冲器 1350 中。

[0140] 由此，除了与主机系统和卸载网络适配器之间的数据的零拷贝传送相关联的优点以外，本发明还提供了这样一种机制，通过该机制，连接可以共享缓冲器以便使连接缓冲器所使用的主机系统存储器的量最小化，本发明还提供了这样一种机制，用于当卸载网络适

配器的存储器溢出时处理数据，并且避免分配给专用连接缓冲器的未使用的主机系统存储器。

[0141] 除了上述的存储器管理机制以外，本发明还提供了用于已建立连接的部分接收和发送缓冲器。本发明中的所述“部分接收和发送缓冲器”功能指的是，本发明中的把接收到的数据添加至具有早已接收 / 发 送的应用数据的缓冲器中的能力。对于应用数据传送，重新使用缓冲器，而不是分配两个单独的缓冲器。

[0142] 图 14 举例说明了按照本发明一个示例性实施例的部分接收 / 发送缓冲器进行操作的方式。采用部分接收 / 发送缓冲器，主机系统 1410 向卸载网络适配器 1420 通知为特定的连接分配了应用缓冲器 1430。例如，缓冲器可用请求描述符可以被发布至输入描述符表。以这种方式，主机系统 1410 把应用缓冲器 1430 的所有权移交至卸载网络适配器 1420。

[0143] 然后，所述卸载网络适配器 1420 在所述连接上接收数据，并且把数据 DMA 至主机系统 1410 上的应用缓冲器 1430。所述卸载网络适配器 1420 然后可以在输出描述符表中发布缓冲接收响应描述符。在所述的例子中，被 DMA 至应用缓冲器 1430 的数据只够部分地填充应用缓冲器 1430。

[0144] 当向主机系统 1410 通知数据到达应用缓冲器 1430 中时，网络接口把对这个“部分”应用缓冲器 1430 的控制移交给主机系统 1410。原始缓冲器的任何其余部分仍在卸载网络适配器 1420 的控制之下。Read() 调用的语义要求在响应中添加“Byte Offset(字节偏移)”值。当所返回的数据的 Offset(偏移)+Length(长度) 等于原始应用缓冲器 1430 的总长度时，主机系统 1410 中的应用将会知道对应用缓冲器 1430 的全部控制都被返回给主机系统 1410。如果数据的 Offset+Length 不等于原始应用缓冲器 1430 的总长度，那么卸载网络适配器 1420 仍保留对缓冲器的部分控制。作为选择，还可以提供一个额外的字段，用于表明应用缓冲器 1430 的最终数据传送。如果这是应用缓冲器 1430 的最终数据传送，那么控制已经被返回给主机系统 1410，并且卸载网络适配器 1430 不保留对应用缓冲器 1430 的部分控制。

[0145] 此后，如果在所述连接上接收到附加数据，那么卸载网络适配器 1420 可以把这个附加数据 DMA 到主机系统 1410 上的同一应用缓冲器 1430 中，从而使得数据被添加到应用缓冲器 1430 中。然后，诸如通过在输出描述符表中发布另一缓冲接收响应描述符，来由卸载网络 适配器 1420 向主机系统 1410 通知对于该连接而言附加数据已经达到。

[0146] 采用如上所述的这种机制，如果网络分组大小不等于主机存储器的缓冲器大小，那么分段可能是一个难题。然而，在提供了较大的连续的虚拟缓冲器以供应用程序使用的情况下，可以使用缓冲器分段，以便保持虚拟连续空间优先选择。这样做使得应用免遭虚拟存储器上并置的缓冲器的增加了的杂务的影响。

[0147] 例如，考虑对于待传送的数据提供 4 兆字节的应用缓冲器的应用程序 Read() 调用。这可能是期待接收例如大的数据文件或者多媒体流用于进行显示。当从网络接收到数据时，卸载网络适配器可以直接把此数据的 1500 字节的部分返回给应用缓冲器。这种设置允许在连续的虚拟 (应用程序) 空间中接收此数据，由此避免应用程序侧上的数据重组的额外复杂性。

[0148] 另一方面，当应用缓冲器不是较大的连续虚拟缓冲器的一部分时，所述卸载网络适配器 1420 可以推荐允许分段，以便优化所接收的数据的位置。允许分段可以有助于减少

从卸载网络适配器 1430 移交到主机系统 1410 的缓冲器数目，并且反之亦然。由此，除了允许数据的零拷贝传送、数据的缓冲后传送和共享缓冲器以外，本发明还提供了这样一种机制，用于重新使用部分填充的缓冲器从而最小化分配给连接使用的缓冲器数目。

[0149] 如上所述，卸载网络适配器在其自身和主机系统之间进行通信并且传送数据的方式是通过 DMA 操作实现的。如同连接的建立一样，当向 / 从卸载网络适配器和主机系统传送数据时，卸载网络适配器可以延迟这些 DMA 操作，以便可以实现数据的成批传送。也就是说，一旦主机系统请求数据传送，所述卸载网络适配器未必启动 DMA 请求。当所述卸载网络适配器认为适合时，所述卸载网络适配器可以判定何时将对传输的数据启动 DMA 操作。

[0150] 例如，如果所述卸载网络适配器的存储器中早已具有足够的数据在一个连接上进行发送，那么卸载网络适配器可以延迟用于在所述连 接上传送数据的 DMA 操作。所述卸载网络适配器可以根据各种标准来确定由什么构成了“足够”量的数据，所述标准例如是带宽和延迟的乘积的当前估计值、拥挤窗口、以及卸载网络适配器上的可用存储器等。所述卸载网络适配器也可以根据其它可能的标准来做出判定，所述标准诸如是公平的排队、与和所述连接相关联的应用相关联的服务质量、服务的差别等。

[0151] 例如，考虑这样一种情况，其中应用程序 Read() 调用为待传送的数据提供 4 兆字节的缓冲器。当从网络中接收到数据时，卸载网络适配器可以把此数据的 1500 字节的部分直接返回到缓冲器。所述卸载网络适配器可以意识到的是 :应用程序提供了非常大的缓冲器以待进行成批数据传送，然后可以分批处理从网络接收的多个 1500 字节的分组以待接收附加分组。成批传送中的 1500 字节的分组的数目往往取决于主机系统和卸载网络适配器之间的连接的特性而定。作为一个示例，与早先的 PCI 2.1 总线互连相比，诸如 PCI-Express 的更新的技术可以更加有效地移动更大的数据块，例如 64K。

[0152] 正如前面提到的那样，当把数据置于应用缓冲器中以便发送时，可以把缓冲发送请求描述符发布给输入描述符表。此缓冲发送请求描述符可以包括一个 ASAP（尽可能快的）比特，用于表明是否要加快数据的发送。在确定是否应该延迟 DMA 操作以及延迟多长时间的过程中，ASAP 比特的设置还可以是由卸载网络适配器使用的标准。当然，只要有可能，通过这个 ASAP 比特的设置，所述卸载网络适配器应该试图承兑主机系统对加快数据传输的请求。

[0153] 就处理器周期、所要求的存储器资源等而言，DMA 操作势必具有固定的设置成本以及每字节的传送成本。为了更好地使用 I/O 总线并且相对于高字节成本来降低设置成本，所述卸载网络适配器可以通过识别对 DMA 传送的两个请求用于相邻的物理存储器区域来合并 DMA 传送。例如，通过为每一连接分配大的应用缓冲器，逐渐地填写应用缓冲器的子集，并且由此生成对存储器相邻子集的请求，主机系统可以试图促进该处理。所述卸载网络适配器可以将这些子集识别 为相邻的，并且合并 DMA 传送。

[0154] 作为一个例子，描述符队列包含 DMA 传送的地址和长度的详细信息。在执行 DMA 操作之前对相邻描述符的检查可以表明 :接下来的 DMA 请求仅仅是当前请求的继续，即涉及存储器的相邻部分。在此情况下，两次 DMA 传送可以满足于涉及需要执行的两个 DMA 操作的单个组合的请求。通过提供对这些 DMA 传送的成批通知，降低了处理主机系统和卸载网络适配器之间的 DMA 传送请求的系统开销。

[0155] 本发明可以“积存 (store up)”DMA 数据传送，直到存在足够数目的 DMA 数据传送

为止。用于确定“足够”的标准可以如上所述那样发生改变。一旦有足够数目的 DMA 数据传送准备好执行，本发明就使用一种用于确定要进行这些 DMA 数据传送的顺序的优先级机制。由此，在本发明的一个示例性实施例中，由卸载网络适配器根据优先级机制来对 DMA 操作重新排序，以便可以向饥饿连接 (starvedconnection) 和高优先级连接给予优先选择。

[0156] 图 15 举例说明了按照本发明一个示例性实施例的示例性 DMA 传送顺序判定处理过程。如图 15 所示，已经建立了三个连接，即连接 A、B 和 C。已经向这些连接给予了语义优先级顺序 A、B 和 C，其中 A 是最高或优选的连接。此优先级顺序例如可以根据由用户或者主机系统分配给应用程序或者应用程序连接的优先级来确定。正如早先提及的那样，所述卸载网络适配器可以存储有关已建立连接的信息。此优先级信息可以作为连接信息的一部分存储在卸载网络适配器中，并且可以连同连接信息的其余部分一起在主机系统上进行复制。依照此方式，使所述优先级信息对于卸载网络适配器和主机系统而言都是可用的，以便用于确定 DMA 操作的顺序。

[0157] 在所述时间上，所有连接在卸载网络适配器 1520 上具有足够的数据用于在连接 A、B 和 C 上进行发送。需要确定应该把数据从应用缓冲器 1530、1540 和 1550DMA 至卸载网络适配器的缓冲器 1560、1570 和 1580 以便进行传输的顺序。

[0158] 采用本发明，通过在输入描述符表 1590 中存储多组描述符，数据的成批传送变得更为方便，其中所述描述符用于描述发送操作和可用来发送数据的应用缓冲器 1530-1550 的地址。所述卸载网络适配器根据所指定的连接优先级来对输入描述符表 1590 中的描述符列表重新排序。

[0159] 在一个示例性的实施例中，描述符列表的重新排序最初根据目前缺少数据的 (data starved) 连接来执行。也就是说，如果连接是缺少数据的，即，在预定时段内尚未在连接上传输数据，那么首先在描述符列表中对与用于在这种连接上传输的数据相关联的描述符进行排序。此后，根据与所述连接相关联的优先级来对描述符重新排序。

[0160] 由此，按照所述示例，输入描述符表条目 1590，即，用于连接 A、B 和 C 的缓冲发送请求描述符将由卸载网络适配器 1520 读取并重新排序，以便使重新排序的描述符列表具有如下顺序 :A1、A2、A3、B1、B2、B3、C1、C2、C3。然后，依照此顺序从应用缓冲器 1530-1550 中读取数据，并且将其存储在卸载网络适配器的缓冲器 1560-1580 中，从而把优先级给予连接 A。

[0161] 由此，本发明还提供了这样一种机制，其使用应用缓冲器、缓冲发送请求描述符、输入描述符表和 DMA 操作在主机系统和卸载网络适配器之间进行数据的成批传送。以这种方式，DMA 操作可以被延迟，以便使它们可以被成批地执行，而不会逐个地中断在主机系统上运行的应用程序。

[0162] 图 16 是概述当按照本发明一个示例性实施例的各方面使用主机系统和卸载网络适配器发送数据时的示例性操作的流程图。如图 16 所示，所述操作从请求传输由应用程序发送至操作系统的数据开始 (步骤 1610)。然后把所述数据从应用缓冲器复制到锁定的内核缓冲器 (步骤 1620)。然后，把缓冲发送描述符发布给输入描述符表 (步骤 1630)。

[0163] 然后，所述卸载网络适配器通过 DMA 操作读取输入描述符表中的下一条目 (步骤 1640)。为了便于描述，假定下一条目是缓冲发送 描述符。所述输入描述符表被存储在成批传送列表中 (步骤 1650)，并且就是否已经满足了延迟标准做出确定 (步骤 1660)。如果

没有，那么操作返回到步骤1640，以便读取输入描述符表中的下一条目。然而，如果延迟标准已经得以满足，那么根据就是否有任何连接是饥饿的而进行的确定以及连接优先级，来重组成批传送列表（步骤1670）。

[0164] 如上所述，作为此确定过程的一部分，可以确定缓冲发送描述符是否表明已经设置了ASAP比特。倘若如此，那么可以确定出所述延迟标准已经得以满足，并且如果有可能，就立即执行数据的传输。

[0165] 此后，经由DMA操作从锁定的内核缓冲器中读取数据，并且依照根据成批传送列表的重组而确定的顺序来由卸载网络适配器传输（步骤1680）。然后，缓冲器可用响应描述符可以被发布给输出描述符表，该输出描述符表然后由主机系统读取以便确认由卸载网络适配器发送了所述数据（步骤1690）。然后，所述操作终止。

[0166] 图17是概述当按照本发明一个示例性实施例的各方面在主机系统和卸载网络适配器之间执行数据的零拷贝传送时的示例性操作的流程图。如图17所示，所述操作通过在卸载网络适配器中经由已建立的连接接收数据而开始（步骤1710）。然后，所述卸载网络适配器1420把缓冲接收响应描述符发布给输出描述符表（步骤1720）。所述主机系统读取输出描述符表中的下一条目（步骤1730）。为了便于描述，假定所述输出描述符表中的下一条目是缓冲接收响应描述符。然后，可以把所述输出描述符表的条目存储在成批传送列表中（步骤1740）。

[0167] 就延迟标准是否得到满足做出确定（步骤1750）。如果没有，则操作返回到步骤1730。如果延迟标准已经得以满足，那么根据连接是否饥饿以及连接优先级来重新排序成批传送列表（步骤1760）。然后，使用DMA操作，依照根据成批传送列表的重新排序而确定的顺序，把数据直接传送至与存在数据的每一连接相关联的应用缓冲器（步骤1770）。然后，对于已完成的每一个DMA操作，所述主机系统可以把缓冲器可用响应描述符发布给输入描述符表（步骤1780）。 然后，所述操作终止。

[0168] 应该理解的是，使用DMA操作向其发送数据的应用缓冲器可以包括一个或多个共享应用缓冲器。由此，对于共享该一个或多个共享应用缓冲器的各种连接所接收的数据可以被DMA至共享应用缓冲器，并且应用程序可以从共享应用缓冲器中检取所述数据。对于图16中描述的数据发送操作也同样如此，即，从中发送数据的应用缓冲器可以是共享应用缓冲器。

[0169] 由此，本发明提供了这样一种机制，其用于共享应用缓冲器，延迟主机系统和卸载网络适配器之间的通信，从而使得数据的成批传送得以实现，并且主机系统和卸载网络适配器之间的数据的零拷贝传送也可以得以实现。另外，本发明提供了一种用于部分缓冲器数据传送的机制，从而使得数据可以被传送至早已具有传输到其上的数据的同一应用缓冲器。

[0170] 处理接收到的数据

[0171] 除了连接建立和存储器管理以外，本发明还对使用卸载网络适配器的数据处理系统中的已接收数据的处理加以改进。如上所述，本发明的卸载网络适配器可以包括用于允许卸载网络适配器以不同的方式延迟向主机系统通知数据接收情况的逻辑。延迟向主机系统通知数据分组接收情况的优点在于：可能在单个通知中合并在例如第一个数据分组后立即到达的多个数据分组。考虑到具有连续数据分组到达的流，可以为通知延迟设置一个值，

并且可以每一通信套接字地为主机系统配置该值。

[0172] 所述延迟值可以被静态地或者动态地设置。例如，所述延迟值可以通过对套接字连接中所接收的数据的历史观察、根据在某一时段内所接收的数据的速率或数量来设置。一个示例可以是：如果一个特定的接收连接依照在 10 毫秒上有 10 个数据分组然后平静达 10 秒的脉冲串进行操作，那么可能谨慎的是，延迟 10 毫秒内的全部分组到达通知，以便减少向主机系统进行的总体通知。

[0173] 作为选择，主机系统把应用缓冲器发布给连接所依据的速率可以被监控，并且被用作动态设置该延迟值的基础。如果所述主机以特定速率发布应用缓冲器，例如，每 10 毫秒一次，那么把数据到达通知延迟 10 毫秒，以便确保缓冲器可用于从卸载网络适配器到主机系统的数据的零拷贝传送，将会是有意义的。

[0174] 作为进一步的选择，在已经把数据到达通知发送至主机系统之后主机系统为连接发布新的缓冲器所依照的速率可以被监控，并且被用作设置延迟值的基础。这表明主机系统消耗来自特定连接的数据的速率。例如，主机系统可能花费 10 毫秒来消耗缓冲器内的数据，并且把缓冲器发布给卸载网络适配器以供使用。由此，把通知延迟 10 毫秒可能是谨慎的，以便确保为卸载网络适配器和主机系统之间的数据的零拷贝传送代替数据缓冲器。

[0175] 在又一个可替换的实施例中，可以使用数据量而不是时间度量来用于缓冲接收发布延迟。在此情况下，所述延迟值被设置为在向主机系统通知接收到数据分组之前等待接收一定量的数据。该数据量可以作为连接设置中的选项来由主机系统静态地设置，或者根据历史观察数据由卸载网络适配器动态地设置。在不脱离本发明的精神和范围的情况下，还可以使用用于确定延迟值设置的其它方法和机制。

[0176] 无论选择哪种替换实施例用于确定延迟量，在卸载网络适配器中都可以保存最大延迟值，以便标识第一个数据到达和最后向主机系统通知该数据到达之间的最大延迟。这样做确保在数据到达和向主机系统通知该数据到达之间不会存在过量延迟。延迟值、最大延迟值以及用于确定延迟值所需的其它信息可以被存储在卸载网络适配器上的存储器中，以便用于设置延迟值，并且用于确定从卸载网络适配器向主机系统进行通知要延迟多久。

[0177] 在对本发明的操作的先前描述中，依照一个或多个上述替代方式所确定的延迟值以及最大延迟值可用于确定延迟标准是否得以满足。例如，当确定延迟标准是否被满足时，可以将从接收到第一个数据分组开始的时间延迟与所述延迟值进行比较。一旦时间延迟满足或者超 出了延迟值，那么可以从卸载网络适配器向主机系统进行数据分组的成批传送，反之亦然。同样，如果就数据量而言存在延迟值，那么可以把从已经接收到第一个数据分组开始在连接上所接收到的数据量与所述延迟值进行比较，以便确定数据量是否满足或超出了在延迟值中设置的数据量。倘若如此，那么可以通过把成批数据接收通知发送给主机系统或者卸载网络适配器，例如，把缓冲接收响应描述符发布给输入 / 输出描述符表，来启动从卸载网络适配器向主机系统进行的数据的成批传送，或者启动从主机系统向卸载网络适配器进行的数据的成批传送。

[0178] 在当前的非智能主机网络适配器系统中，所有数据都经过主机的操作系统层中的非连接专用应用缓冲器池。考虑到能够使用本发明的机制来向连接专用应用缓冲器进行数据的零拷贝传送，本发明提供了一种判定处理过程，用于在应用程序当前没有发布连接专用应用缓冲器或者共享应用缓冲器来接收数据时的情况。作为默认，如果连接专用应用缓

冲器或者共享应用缓冲器还未分配给连接，那么本发明的判定处理过程使用来自非连接专用应用缓冲器池的缓冲器来把数据从卸载网络适配器传送至应用程序。

[0179] 然而，采用本发明，可以提供一个由主机系统提供的配置参数，以便如果不存在连接专用缓冲器，那么卸载网络适配器可以等待，直到分配了连接专用应用缓冲器为止，而不使用非连接专用应用缓冲器。这个参数可以被存储在卸载网络适配器的存储器中，并且可用来代替系统的默认行为，以便在把数据 DMA 至主机系统之前，使卸载网络适配器等待，直到为连接分配了连接专用应用缓冲器为止。可以执行这种等待，直到分配了连接专用应用缓冲器或者满足或超出了最大等待时间为止。如果满足或者超出了最大等待时间，那么为所述连接而存储在卸载网络适配器中的数据可以被 DMA 至非连接专用应用缓冲器。

[0180] 所述卸载网络适配器自身可以具有用于允许它根据提供连接专用应用缓冲器的主机系统的历史数据来确定是否等待连接专用应用 缓冲器、等待连接专用缓冲器达多长时间、或者不等待连接专用应用缓冲器的逻辑，而不是设置预定的主机提供的配置参数来代替使用非连接专用应用缓冲器的默认行为。

[0181] 例如，主机系统可能已经为零拷贝操作提供连接专用应用缓冲器长达历史数据中观察到的时间帧中的 100％的时间。也就是说，在先前的 x 次数据传送中，使用连接专用应用缓冲器达 100％的时间以便利于这些数据传送。因此，可以执行用于等待连接专用应用缓冲器的上述操作。

[0182] 然而，如果所述历史数据表明没有使用连接专用应用缓冲器执行数据传送达 100％的时间，那么就使用连接专用应用缓冲器的时间的百分比是否少于预定阈值量做出确定。倘若如此，那么卸载网络适配器不必等待分配连接专用应用缓冲器，并且可以利用非连接专用应用缓冲器。作为选择，卸载网络适配器等待连接专用应用缓冲器的时间量可以根据百分比值是否低于预定阈值来减少。当数据传送继续进行时，卸载网络适配器内保留的历史数据可以是随着每一次数据传送移动的时窗。由此，当使用连接专用应用缓冲器来执行更多的数据传送时，百分比值可能增加到超过预定阈值，并且系统可以返回到等待分配连接专用应用缓冲器，或者可以返回到连接专用应用缓冲器的原始等待时间。

[0183] 依照本发明的示例性实施例的另一方面，如果必须从所述池中选择非连接专用应用缓冲器以便用于把数据从卸载网络适配器 DMA 至主机系统，那么本发明在卸载网络适配器内提供了用于选择将向其发送数据的非连接专用应用缓冲器的逻辑。此逻辑检查缓冲器池中各个非连接专用应用缓冲器的每一个特性，并且选择用于提供将从卸载网络适配器传送至主机系统的数据的最佳匹配的一个缓冲器。可以从保存在主机系统和 / 或卸载网络适配器中的连接信息中获得与缓冲器有关的信息。

[0184] 例如，当所述卸载网络适配器确定它必须使用来自缓冲器池的非连接专用应用缓冲器时，所述卸载网络适配器从主机系统中读取所述 池中的缓冲器的特性信息。此特性信息例如可以是缓冲器的大小、缓冲器的速度、缓冲器在主机处理器体系结构中的位置等。根据这些特性，所述卸载网络适配器从所述池中选择作为用于在把数据从卸载网络适配器传送至主机系统的过程中使用的最佳候选的缓冲器。

[0185] 作为一个例子，把缓冲器大小作为选择处理过程所关注的特性时，在缓冲器池中存在可用的具有不同大小的多个非连接专用应用缓冲器。假定一定量的数据要被传送至主机系统，则所述卸载网络适配器将从缓冲器池中选择具有足够大小以便完全包含数据的非

连接专用应用缓冲器，而不是在多个缓冲器上散布数据。也可依照类似方式使用上述其它特性，以便确定用于特定数据传送的最佳缓冲器。

[0186] 图 18 是概述按照本发明一个示例性实施例的各方面用于确定要向其发送数据的应用缓冲器的示例性操作的流程图。如图 18 所示，所述操作通过在卸载网络适配器中接收用于传送至主机系统的数据而开始（步骤 1810）。然后，就是否为与所接收的数据被引导到至此的一个或多个连接分配了连接专用应用缓冲器做出确定（步骤 1820）。倘若如此，那么使用 DMA 操作把数据传输到所分配的一个或多个连接专用应用缓冲器（步骤 1830），并且所述操作终止。

[0187] 如果没有为数据被引导至此的连接分配连接专用应用缓冲器（步骤 1820），那么就是否已经设置了等待参数做出确定（步骤 1840）。倘若如此，那么就是否已经超出等待阈值做出确定（步骤 1850）。如果没有，则所述操作循环返回至步骤 1820，并且继续循环，直到超出了等待阈值或者直到分配了连接专用应用缓冲器为止。

[0188] 如果已经超出了所述等待阈值（步骤 1850），或者如果等待参数尚未被设置（步骤 1840），那么可以从主机系统中检取缓冲器池中的非连接专用应用缓冲器的特性信息（步骤 1860）。然后，根据所检取的特性信息来从该池中选择非连接专用应用缓冲器（步骤 1870）。然后使用 DMA 操作把所述数据直接传送至所选的非连接专用应用缓冲器中（步骤 1880），并且所述操作终止。

[0189] 附加的设计可以允许把数据直接放置至 L3 高速缓存体系结构 中，作为对 DMA 放置的选项。也就是说，可以使用高速缓存注入 (injection) 机制和主机系统所提供的虚拟地址，来把数据推入 L3 高速缓存中。作为对把数据 DMA 放置在应用缓冲器中的替代，或者除此之外，可以把需要迅速处理的数据提供给 L3 高速缓存以便即时处理。

[0190] 存在许多方式来判定是否应该把特定的数据注入到 L3 高速缓存中。例如，有关应该把哪些数据注入到 L3 高速缓存中的确定，可以取决于由主机系统为每一连接建立的显式配置信息。作为选择，此确定可以取决于最近监控到有多少数据已经被注入到 L3 高速缓存中，以便确定是否存在高速缓存溢出情况的可能。还可以使用用于确定向 L3 高速缓存注入数据是否将获得任何优点或者导致高速缓存溢出的其它机制。

[0191] 如上所述，这类存储器管理机制最好是用于要求即时 CPU 关注的某些业务，诸如 web 请求 / 响应业务。诸如 ISCSI 数据（为文件系统所预取的）之类的其它类型数据可能情况像 DMA 那样好，这是由于不必经过若干时间。此参数可以根据网络读取或者配置参数的请求的起源来标识。

[0192] 应该理解的是，虽然如上所述的替代实施例提到把数据注入到 L3 高速缓存中，但是此实施例不局限于和 L3 高速缓存一起使用。在示例性的实施例中，由于 L3 具有依照许多已知体系结构映射的物理地址，所以它是优选的。这降低了从输入 / 输出设备中直接移动数据的设计的复杂性。然而，在新出现的网络适配器、例如诸如 InfiniBand 之类的系统区域网的 RDMA 网络适配器中，可以提供用户地址，以便允许把数据注入到虚拟的可寻址的 L3 高速缓存中以及存储器分层结构中的任何其它高速缓存中。另外，可以从实际地址至虚拟地址来执行地址转换，由此为任何类型的高速缓存提供必要的地址。由此，根据系统的特殊体系结构，示例性的可替代实施例中的机制可以应用于任何级别的高速缓存。

[0193] 在本发明的其它方面中，所述卸载网络适配器可以包含用于重新 组合单独的但

是有序的数据缓冲器分段的逻辑。在由卸载网络适配器生成描述符的过程中，可以在将描述符发布给输出描述符表之前检查所述描述符，以便查看待移动的数据是否被移动到连续的物理地址空间。如果生成用于标识存储器中的连续物理地址的多个描述符，那么待传送的数据可以在卸载网络适配器中进行组合，并且可以使用单个组合的描述符来标识每一个数据传送，而不是把多个描述符发布给输出描述符表。例如，TCP/IP 分段可以被重新组合为适当大小的缓冲器（例如，4K 页对准的数据），并且向主机系统成批地进行通信。这样做提供了对主机系统的更加容易的数据缓冲器管理，并且提供了更高的效率。这样做可以潜在地减少服务于这多个连接所需要的缓冲器数量。

[0194] 在本发明示例性实施例的其它方面中，所述卸载网络适配器具备用于检查所接收分组内的数据但不消耗所述数据的逻辑。接收调用可以指定“查看 (peek)”选项，其可以向主机应用程序提供所接收的数据分组的一部分（例如，首部）的拷贝。这样做可以允许主机应用程序检查首部数据，并且对如何消耗有效负载做出判定。作为一个例子，可能会期待应用程序接收由首部标识符标记的不同类型的数据。如果所述首部和有效负载数据是可变长度的，那么这是特别有用的。所述程序可以仅仅对任何首部的最大长度进行“查看”以便检查首部信息。对首部进行查看，可以允许程序根据预想的程序流来确定要把数据分组的有效负载发送到哪个应用缓冲器。

[0195] 由此，当在卸载网络适配器中为连接设置了“查看”选项时，然后当确定正接收何种类型的数据并且要在哪个套接字（即，连接）上传输数据分组有效负载时，将所接收的数据分组的首部的拷贝提供给主机应用程序。例如，应用程序可以具有用于视频数据和音频数据的独立的连接。所述应用程序能够根据首部确定数据分组的有效负载中的数据类型。如果所述数据是视频数据，那么查看操作允许主机应用程序指明所述数据分组有效负载应被 DMA 至与第一连接相关联的应用缓冲器。如果所述数据是音频数据，那么查看操作允许主机应用程序 指明所述数据分组有效负载应被 DMA 至与第二连接相关联的应用缓冲器。

[0196] 为了示意这种查看操作，提供了一个选项以便利用偏移读取数据。以这种方式，数据分组的有效负载可以容易地与正被查看的首部相分离。也就是说，由于主机应用程序知道首部的实际大小，所以可以生成偏移，并且将其存储用于当处理数据分组时跳过首部。当所述首部小于查看操作中所指定的字节数时，这是最有用的。

[0197] 重要的是应该注意到，虽然已经在完全发挥功能的数据处理系统的环境下描述了本发明，但是本领域普通技术人员将会理解的是，本发明的处理过程能够以指令的计算机可读介质的形式以及各种其它形式来分布，并且不管实际上用于执行所述分布的信号承载介质的具体类型如何，本发明都同样适用。计算机可读介质的例子包括可记录类型的介质，诸如软盘、硬盘驱动器、RAM、CD-ROM、DVD-ROM，以及传输类型的介质，诸如使用例如射频和光波传输的传输形式的数字和模拟通信链路、有线或无线通信链路。计算机可读介质可以采取编码格式的形式，该编码格式被解码用于在特定数据处理系统中实际利用。

[0198] 已经出于举例说明和描述的目的给出了对本发明的描述，但这不是穷举的，也不意味着把本发明限制为所公开的形式。许多修改和变化对于本领域普通技术人员都将会是显而易见的。选择并且描述了该实施例，是为了更好地解释本发明的原理和实际应用，并且是为了使本领域普通技术人员能够针对具有适用于特定预期用途的各种修改的各种实施例而理解本发明。

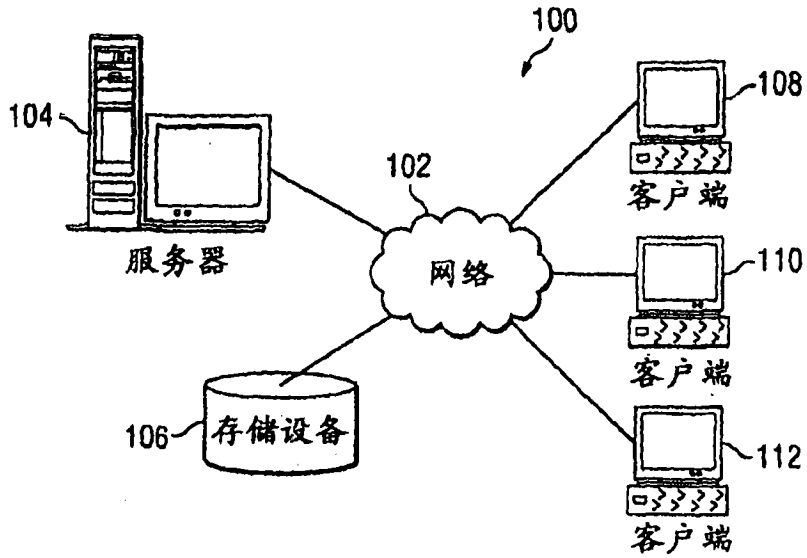

100
104
服务器
102
网络
108
客户端
110
客户端
112
客户端
106
存储设备

图 1

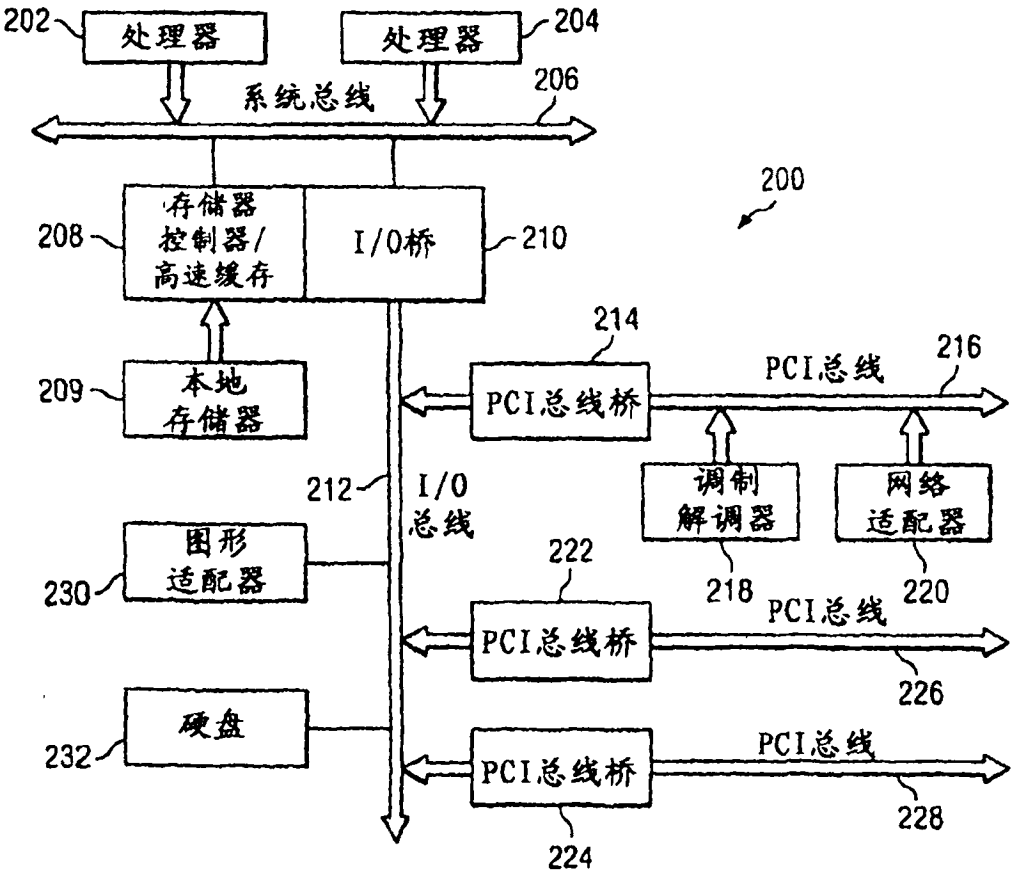

202
处理器
处理器
204
系统总线
206
200
208
存储器
控制器/
高速缓存
I/O桥
210
209
本地
存储器
214
PCI总线桥
PCI总线
216
212
I/O
总线
调制
解调器
网络
适配器
218
220
222
图形
适配器
230
PCI总线桥
PCI总线
226
硬盘
232
PCI总线桥
PCI总线
228
224

图 2

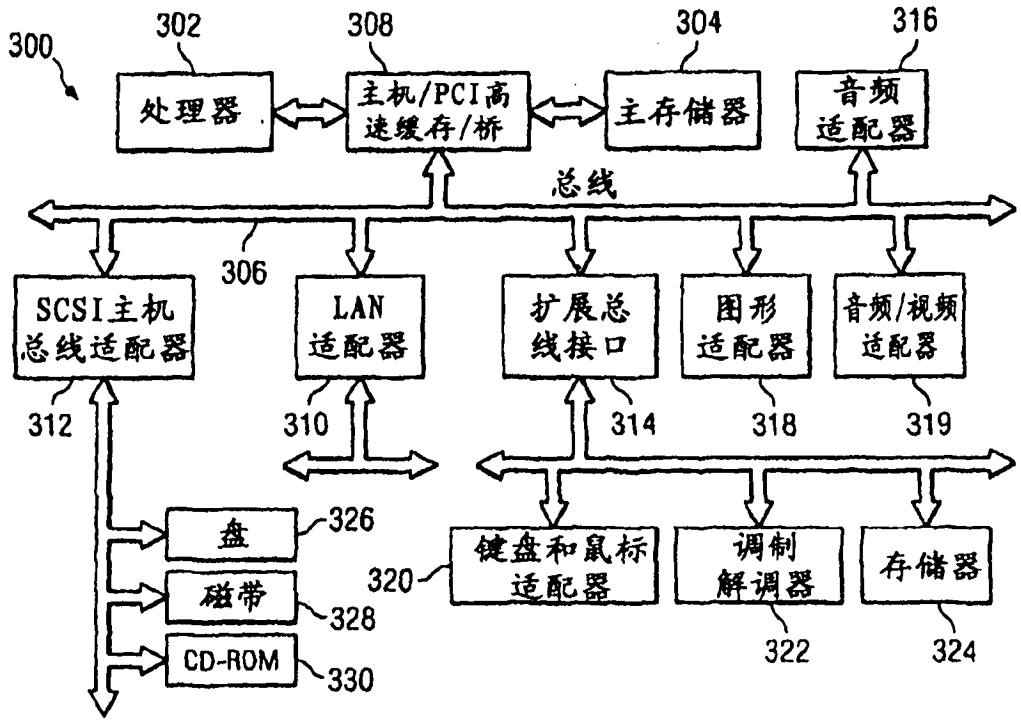

300
302
处理器
308
主机/PCI高速缓存/桥
304
主存储器
316
音频适配器
总线
306
SCSI主机总线适配器
LAN适配器
扩展总线接口
图形适配器
音频/视频适配器
312
310
314
318
319
盘
326
磁带
328
CD-ROM
330
320
键盘和鼠标适配器
调制解调器
存储器
322
324

图 3

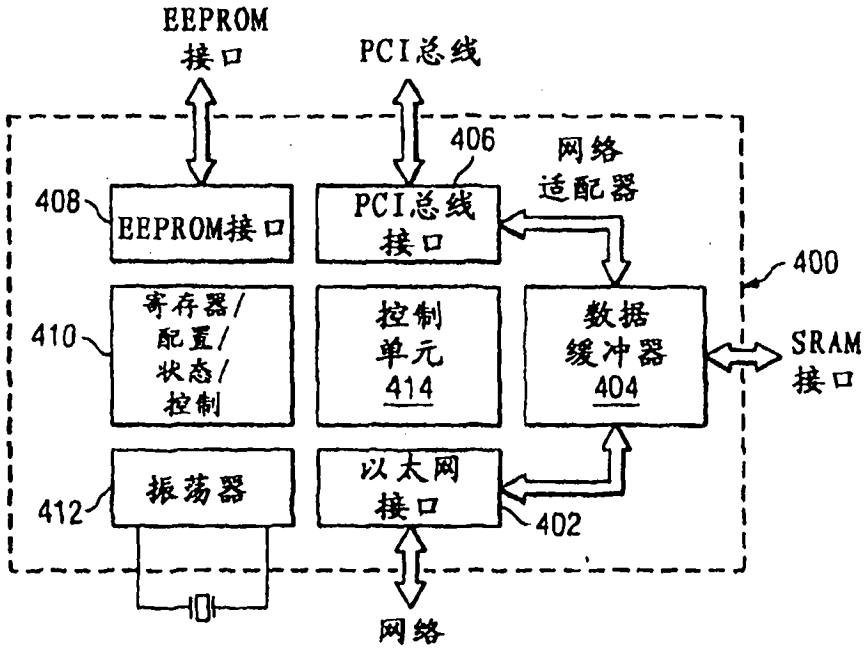

EEPROM接口
PCI总线
406
网络适配器
408
EEPROM接口
PCI总线接口
400
寄存器/配置/状态/控制
410
控制单元
414
数据缓冲器
404
SRAM接口
振荡器
412
以太网接口
402
网络

图 4

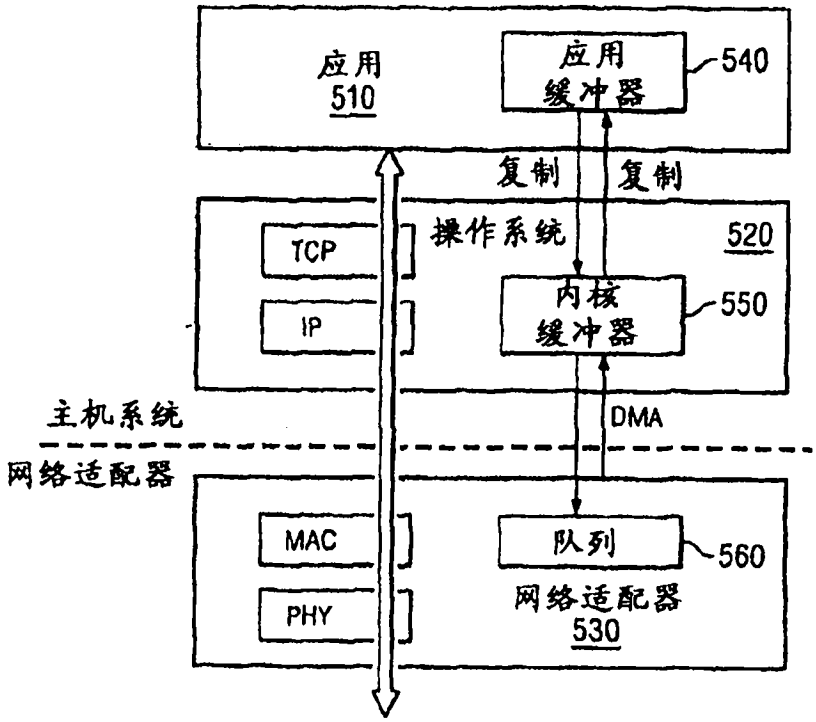

应用
510
应用
缓冲器
540
复制
复制
操作系统
520
TCP
IP
内核
缓冲器
550
主机系统
DMA
网络适配器
MAC
PHY
队列
560
网络适配器
530

图 5

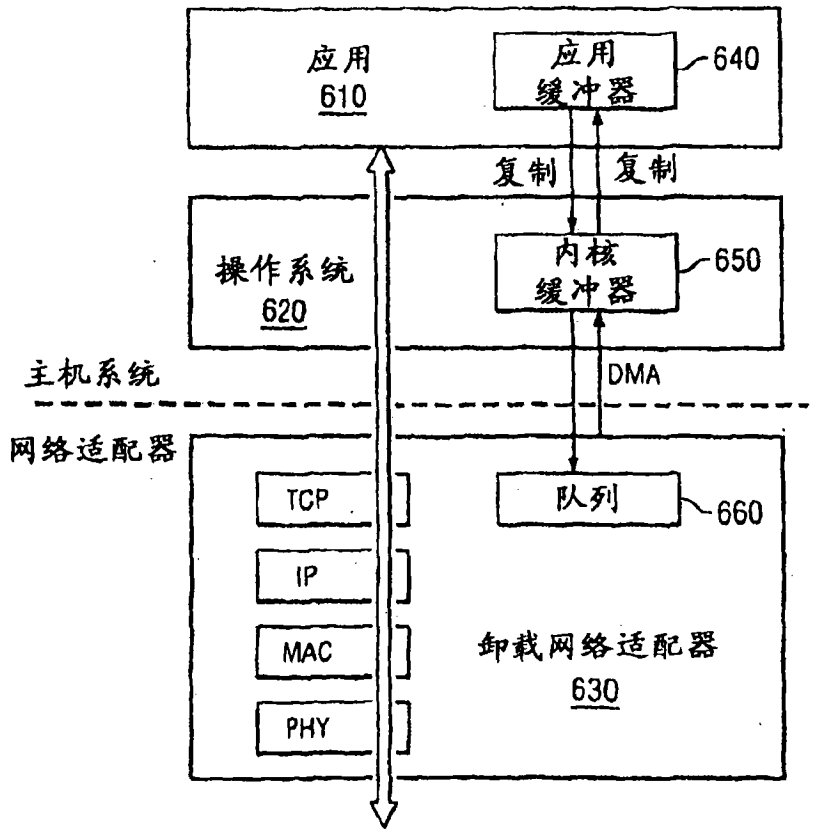

应用
610
应用
缓冲器
640
复制
复制
操作系统
620
内核
缓冲器
650
主机系统
DMA
网络适配器
TCP
IP
MAC
PHY
队列
660
卸载网络适配器
630

图 6

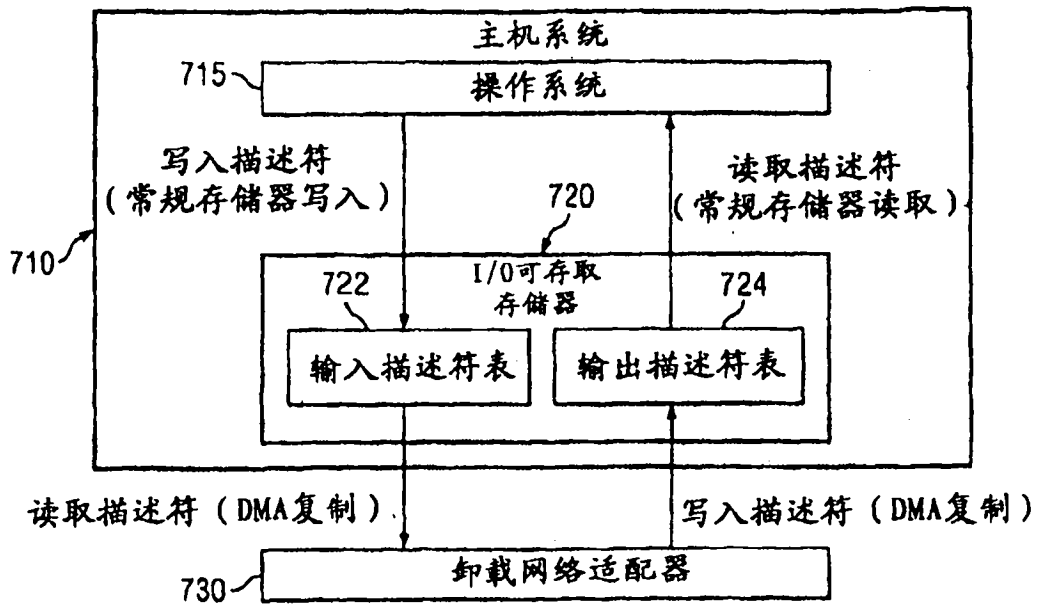

主机系统
715
操作系统
写入描述符
（常规存储器写入）
读取描述符
（常规存储器读取）
720
710
722
I/O可存取
存储器
724
输入描述符表
输出描述符表
读取描述符（DMA复制）
写入描述符（DMA复制）
730
卸载网络适配器

图 7

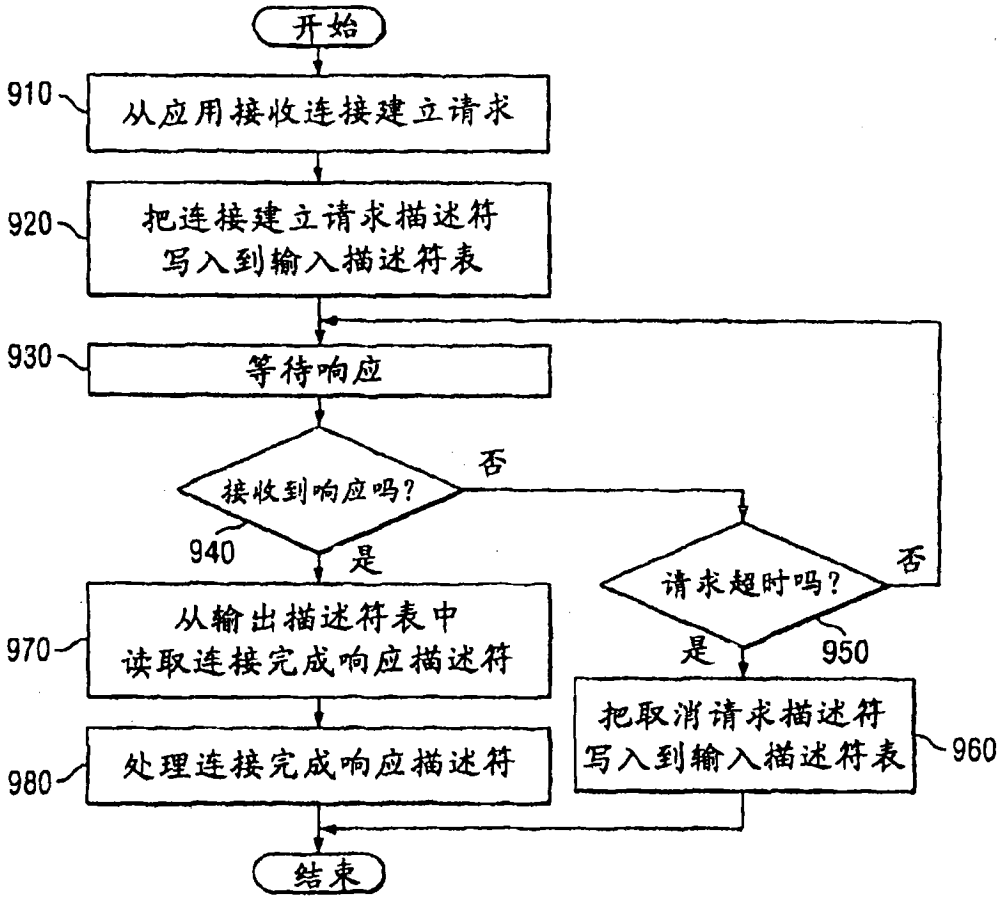

开始
910
从应用接收连接建立请求
920
把连接建立请求描述符
写入到输入描述符表
930
等待响应
接收到响应吗?
否
940
是
请求超时吗?
否
是
950
970
从输出描述符表中
读取连接完成响应描述符
把取消请求描述符
写入到输入描述符表
960
980
处理连接完成响应描述符
结束

图 9

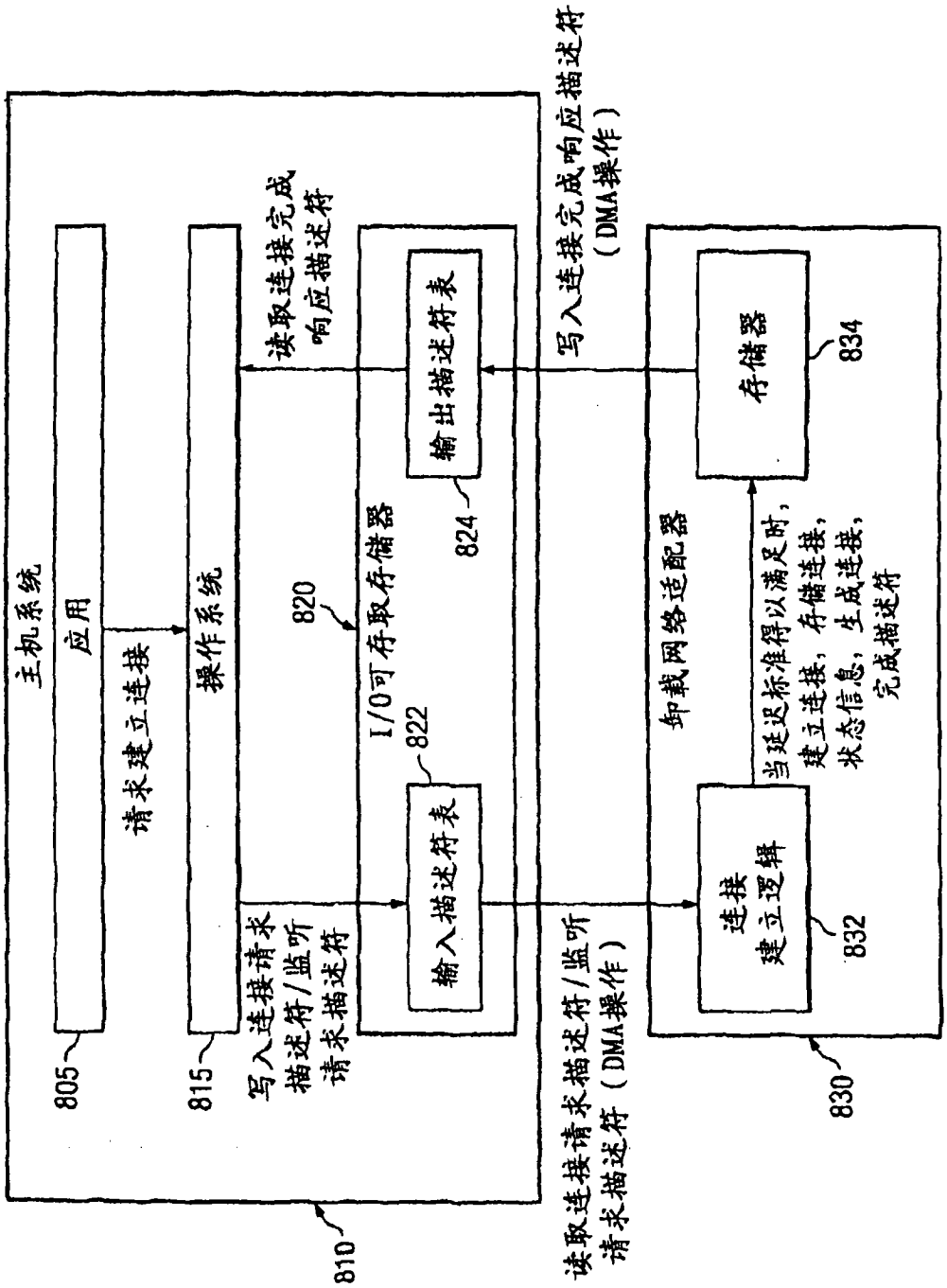

主机系统
应用
805
请求建立连接
操作系统
815
写入连接请求描述符/监听请求描述符
读取连接完成响应描述符
I/O可存取存储器
820
输入描述符表
822
输出描述符表
824
写入连接完成响应描述符（DMA操作）
卸载网络适配器
连接建立逻辑
832
当延迟标准得以满足时，建立连接，存储连接状态信息，生成连接完成描述符
存储器
834
830
读取连接请求描述符/监听请求描述符（DMA操作）
810

图 8

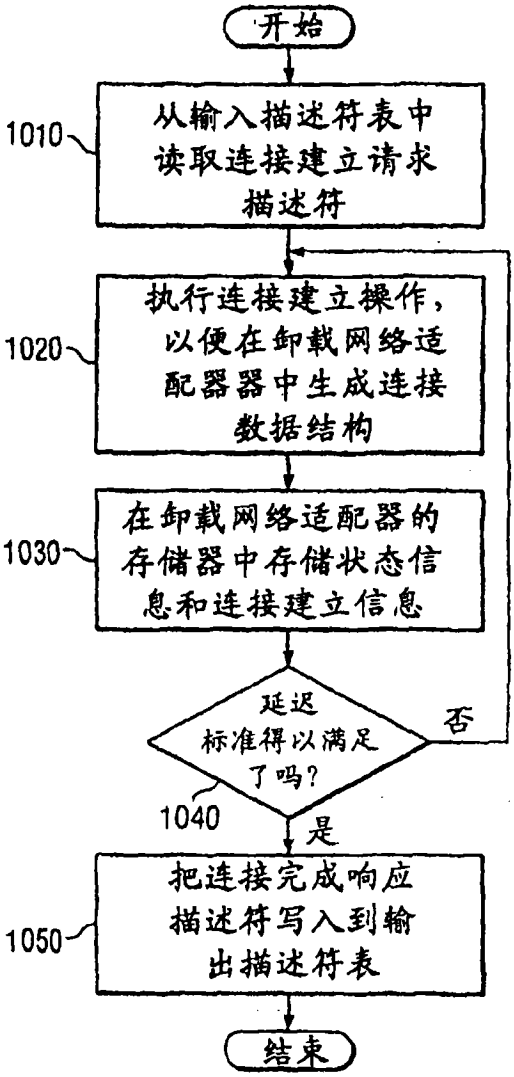

开始
从输入描述符表中读取连接建立请求描述符
1010
执行连接建立操作，以便在卸载网络适配器器中生成连接数据结构
1020
在卸载网络适配器的存储器中存储状态信息和连接建立信息
1030
延迟标准得以满足了吗?
否
1040
是
把连接完成响应描述符写入到输出描述符表
1050
结束

图 10

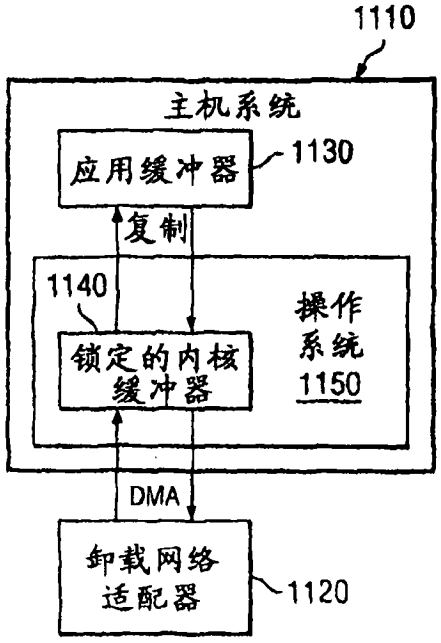

1110
主机系统
应用缓冲器
1130
复制
1140
锁定的内核缓冲器
操作系统
1150
DMA
卸载网络适配器
1120

图 11

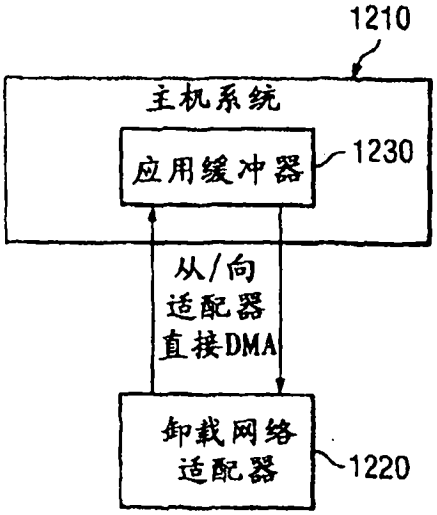

1210
主机系统
应用缓冲器
1230
从/向
适配器
直接DMA
卸载网络
适配器
1220

图 12

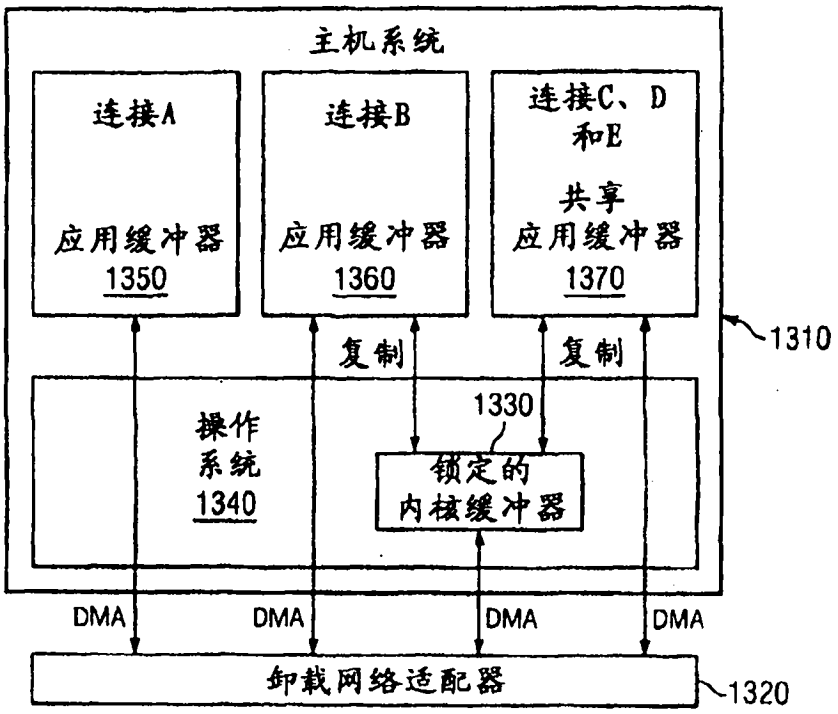

主机系统
连接A
应用缓冲器
1350
连接B
应用缓冲器
1360
连接C、D
和E
共享
应用缓冲器
1370
1310
复制
复制
1330
操作
系统
1340
锁定的
内核缓冲器
DMA
DMA
DMA
DMA
卸载网络适配器
1320

图 13

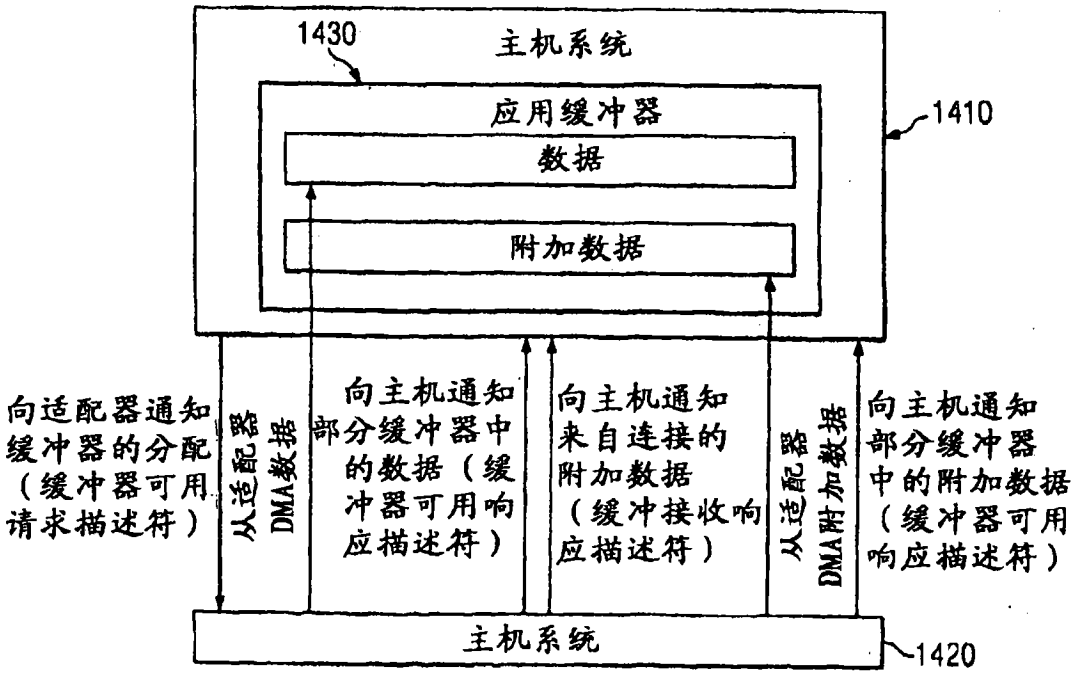

1430
主机系统
应用缓冲器
数据
附加数据
1410
向适配器通知缓冲器的分配（缓冲器可用请求描述符）
从适配器DMA数据
向主机通知部分缓冲器中的数据（缓冲器可用响应描述符）
向主机通知来自连接的附加数据（缓冲接收响应描述符）
从适配器DMA附加数据
向主机通知部分缓冲器中的附加数据（缓冲器可用响应描述符）
主机系统
1420

图 14

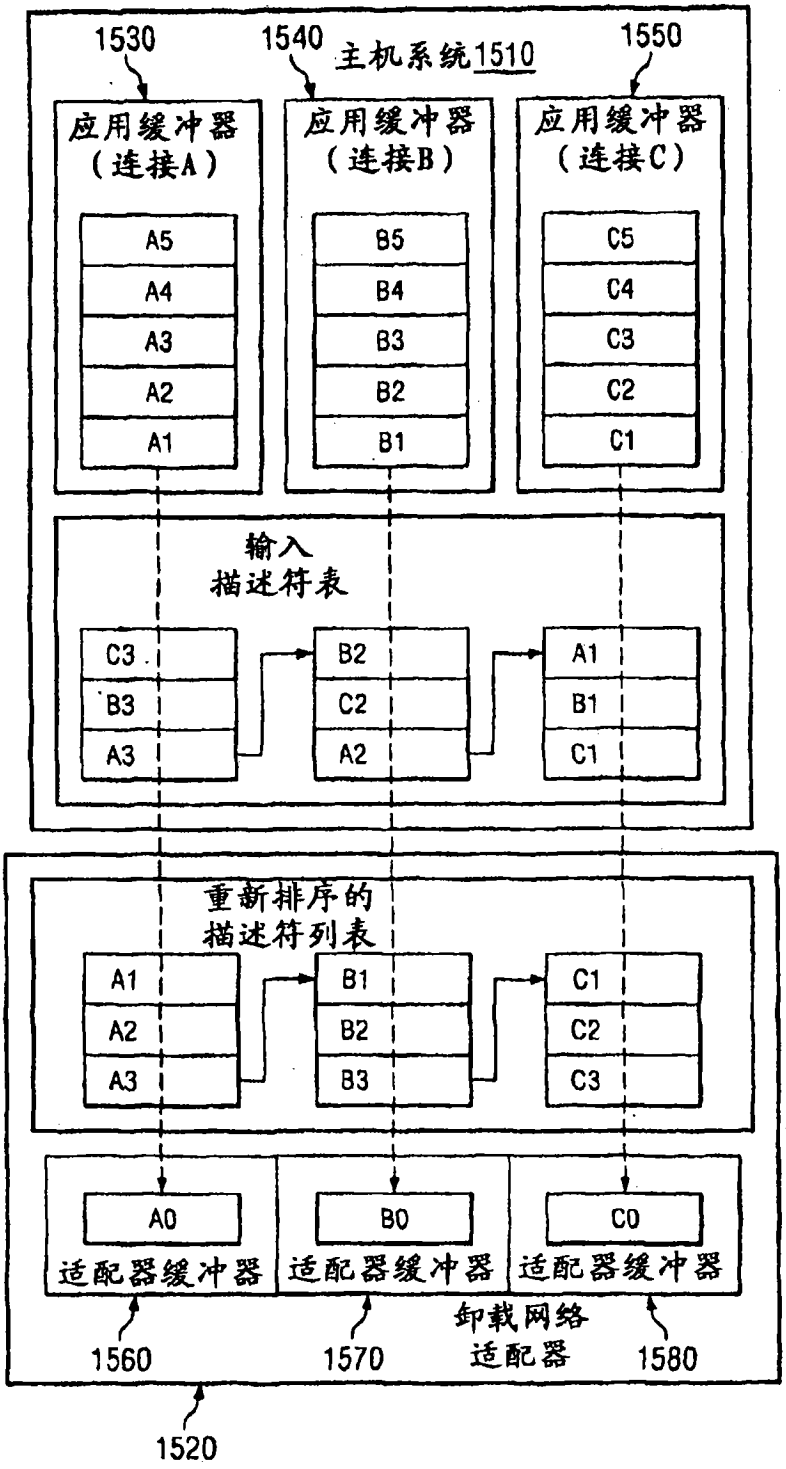

1530
1540
主机系统1510
1550
应用缓冲器（连接A）
应用缓冲器（连接B）
应用缓冲器（连接C）
A5
A4
A3
A2
A1
B5
B4
B3
B2
B1
C5
C4
C3
C2
C1
输入描述符表
C3
B3
A3
B2
C2
A2
A1
B1
C1
重新排序的描述符列表
A1
A2
A3
B1
B2
B3
C1
C2
C3
A0
B0
C0
适配器缓冲器
适配器缓冲器
适配器缓冲器
卸载网络适配器
1560
1570
1580
1520

图 15

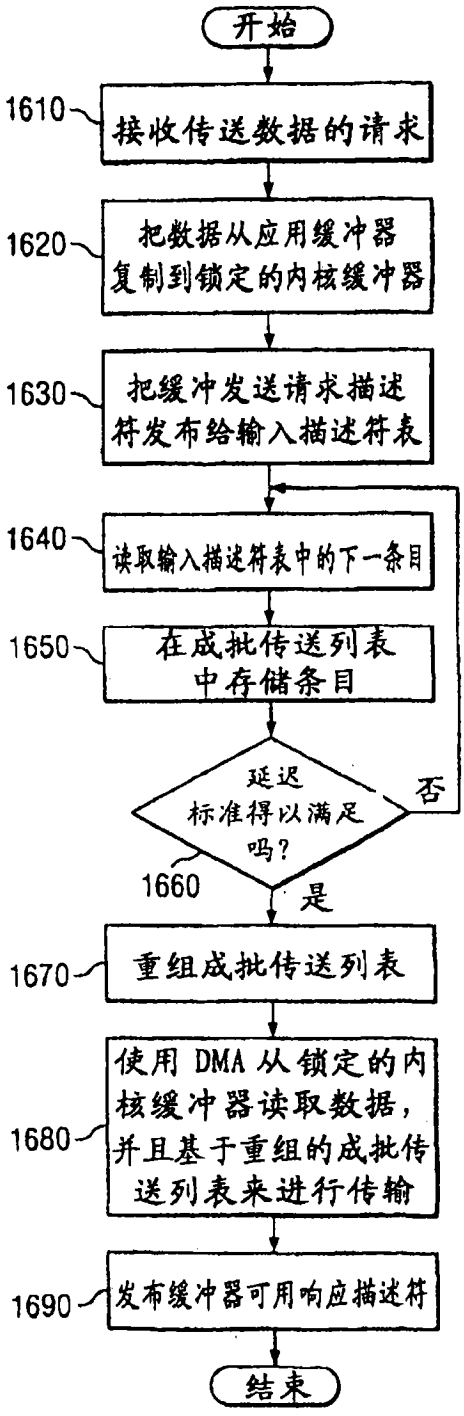

开始
1610
接收传送数据的请求
1620
把数据从应用缓冲器复制到锁定的内核缓冲器
1630
把缓冲发送请求描述符发布给输入描述符表
1640
读取输入描述符表中的下一条目
1650
在成批传送列表中存储条目
延迟标准得以满足吗?
否
1660
是
1670
重组成批传送列表
1680
使用 DMA 从锁定的内核缓冲器读取数据，并且基于重组的成批传送列表来进行传输
1690
发布缓冲器可用响应描述符
结束

图 16

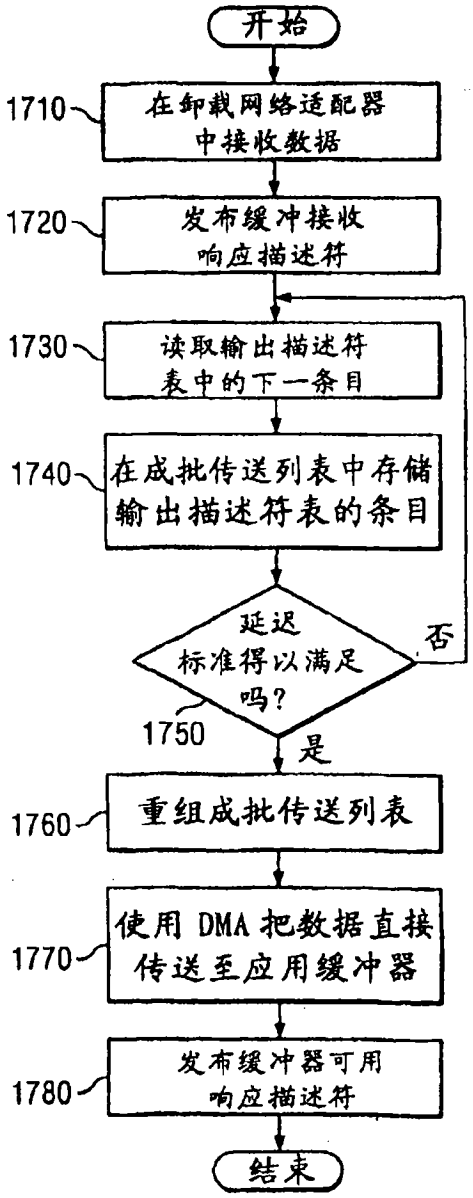

开始
1710
在卸载网络适配器中接收数据
1720
发布缓冲接收响应描述符
1730
读取输出描述符表中的下一条目
1740
在成批传送列表中存储输出描述符表的条目
延迟标准得以满足吗?
否
1750
是
1760
重组成批传送列表
1770
使用 DMA 把数据直接传送至应用缓冲器
1780
发布缓冲器可用响应描述符
结束

图 17

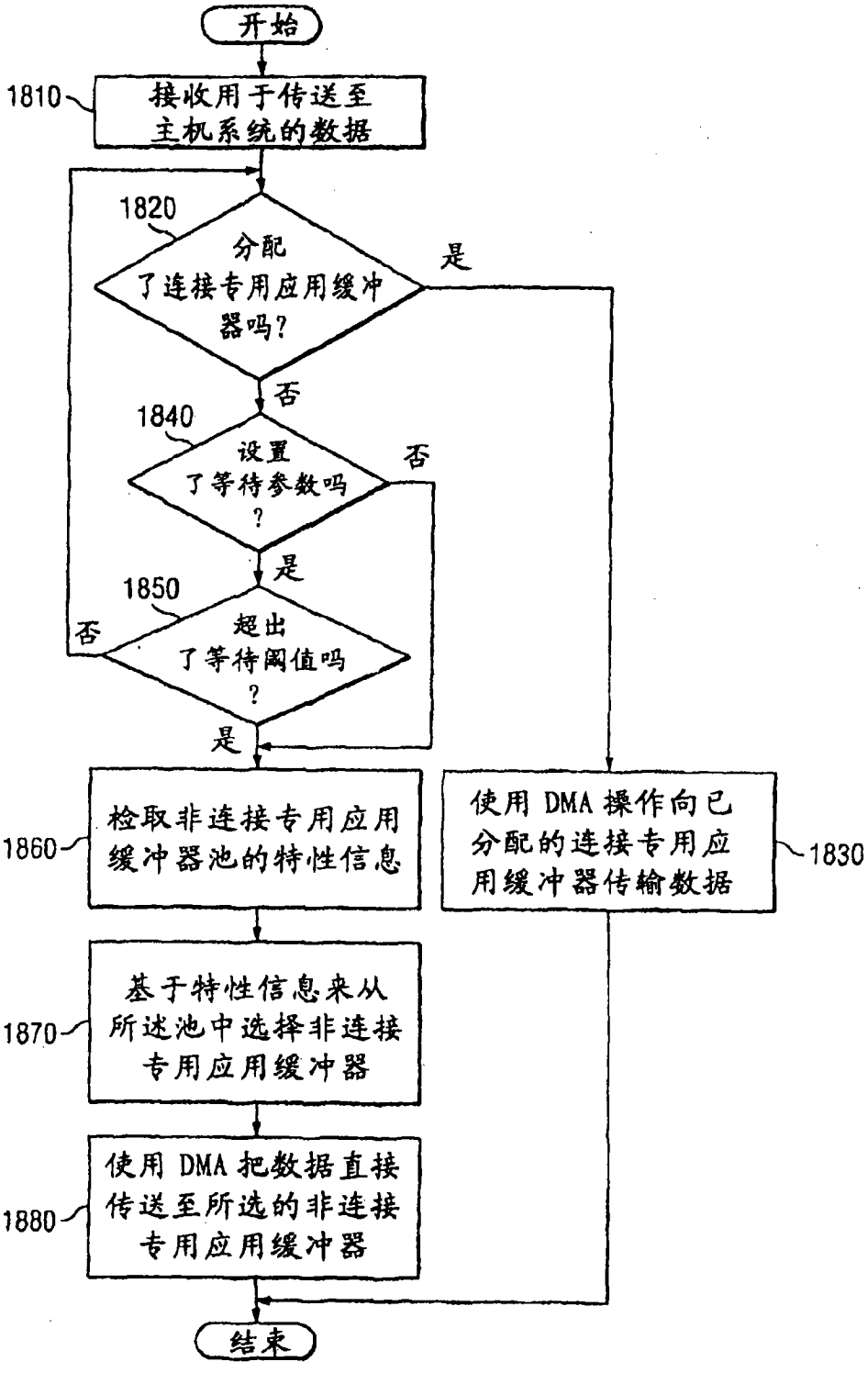

开始
1810
接收用于传送至主机系统的数据
1820
分配了连接专用应用缓冲器吗?
是
否
1840
设置了等待参数吗?
否
是
1850
超出了等待阈值吗?
否
是
1860
检取非连接专用应用缓冲器池的特性信息
使用 DMA 操作向已分配的连接专用应用缓冲器传输数据
1830
1870
基于特性信息来从所述池中选择非连接专用应用缓冲器
1880
使用 DMA 把数据直接传送至所选的非连接专用应用缓冲器
结束

图 18