

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04L 12/28 (2006.01)

H04L 29/08 (2006.01)



[12] 发明专利说明书

专利号 ZL 200510102887.6

[45] 授权公告日 2008 年 10 月 1 日

[11] 授权公告号 CN 100423503C

[22] 申请日 2005.9.14

[21] 申请号 200510102887.6

[30] 优先权

[32] 2004.9.14 [33] KR [31] 10-2004-0073299

[32] 2004.12.7 [33] KR [31] 10-2004-0102304

[73] 专利权人 三星电子株式会社

地址 韩国京畿道

[72] 发明人 李学求 李镛勋 金永根 金坪洙

[56] 参考文献

US6314525B1 2001.11.6

CN1292534A 2001.4.25

CN1518275A 2004.8.4

CN1490991A 2004.4.21

审查员 曹芳

[74] 专利代理机构 北京铭硕知识产权代理有限公司

代理人 韩明星 安宇宏

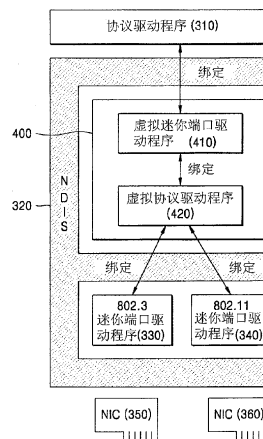
权利要求书 3 页 说明书 11 页 附图 8 页

[54] 发明名称

通信适配器切换方法和设备

[57] 摘要

提供一种在具有多个通信适配器的系统中的通信适配器切换方法，该方法包括：将多个通信适配器的地址设置为与该多个通信适配器的第一个的地址相同；将除了第一通信适配器之外的所有通信适配器的包过滤模式设置为在其中所有包被接受的模式；和在通信期间当一个通信适配器被断开时，从多个通信适配器中的该通信适配器切换到另一个。因此，由于 TCP/IP 会话以相同的 IP 地址被保持，所以从有线 LAN 到无线 LAN 的无缝切换是可行的。



1、一种在具有多个通信适配器的系统中的通信适配器切换方法，该方法包括：

将多个通信适配器的地址设置为与该多个通信适配器的第一个的相同的地址；

将除了第一通信适配器之外的所有通信适配器的包过滤模式设置为在其中所有包被接受的模式；和

在通信期间当一个通信适配器被断开时，从多个通信适配器中的该通信适配器切换到另一个。

2、一种在具有多个通信适配器的系统中的通信适配器切换方法，该方法包括：

将有线局域网通信适配器的介质访问控制地址设置为无线局域网通信适配器的介质访问控制地址相同的介质访问控制地址；

将有线局域网通信适配器的包过滤模式设置为在其中所有包被接受的不加区分模式；和

在使用所述有线局域网通信适配器进行通信期间当所述有线局域网被断开时，使用无线局域网信息从所述有线局域网通信适配器切换到所述无线局域网通信适配器。

3、如权利要求2所述的方法，还包括：

在使用所述无线局域网通信适配器进行通信期间当指示所述有线局域网被连接的事件产生时，使用有线局域网信息从所述无线局域网通信适配器切换到所述有线局域网通信适配器。

4、如权利要求2所述的方法，还包括：

将无线局域网通信适配器的包过滤模式设置为直接、多点传送、或广播模式。

5、如权利要求2所述的方法，还包括：

如果有线局域网通信适配器被断开，则将无线局域网信息映射到虚拟适配器，所述虚拟适配器用于将有线局域网通信适配器的介质访问控制地址设置为与无线局域网通信适配器的介质访问控制地址相同的介质访问控制地址，并将有线局域网通信适配器的包过滤模式设置为所有包被接受的不加区

分模式。

6、一种在具有多个通信适配器的系统中的通信适配器切换设备，该设备包括：

虚拟适配器，其将多个通信适配器的地址设置为与该多个通信适配器的第一个的地址相同，并且将除了第一通信适配器之外的所有通信适配器的包过滤模式设置为在其中所有包可被接受的模式；和

虚拟协议驱动程序，其在通信期间当一个通信适配器被断开时，从多个通信适配器中的该通信适配器切换到另一个。

7、一种在具有多个通信适配器的系统中的通信适配器切换设备，该设备包括：

虚拟适配器，其设置有线局域网通信适配器的介质访问控制地址与无线局域网通信适配器的介质访问控制地址相同，并且将有线局域网通信适配器的包过滤模式设置为在其中所有包被接受的不加区分模式；和

虚拟协议驱动程序，其在使用所述有线局域网通信适配器进行通信期间当所述有线局域网被断开时，使用无线局域网信息从所述有线局域网通信适配器切换到所述无线局域网通信适配器。

8、如权利要求7所述的设备，其中，所述虚拟协议驱动程序包括链路状态检测器，该链路状态检测器每当连接或断开事件产生时提供连接或断开状态信息。

9、如权利要求8所述的设备，其中，所述虚拟协议驱动程序还包括最佳适配器选择器，该最佳适配器选择器基于由所述链路状态检测器提供的信息确定用于通信的最佳适配器。

10、如权利要求9所述的设备，其中，从物理适配器接收的包被无条件地发送到所述虚拟适配器，并且将发送到物理适配器的包发送到被确定的最佳适配器。

11、一种在具有多个通信适配器的系统中的通信适配器切换方法，该方法包括：

将有线局域网通信适配器的介质访问控制地址设置为与无线局域网通信适配器的介质访问控制地址相同的介质访问控制地址；

将有线局域网通信适配器的包过滤模式设置为所有包被接受的不加区分模式；

检查有线局域网是否被连接;

如果该有线局域网被连接, 则产生无线局域网连接事件;

更新无线局域网连接信息; 和

用有线局域网进行通信。

12、如权利要求 11 所述的方法, 还包括:

如果有线局域网未被连接, 则将无线局域网信息映射到虚拟适配器;

用无线局域网进行通信;

产生有线局域网的连接事件;

将有线局域网信息映射到虚拟适配器; 和

用该有线局域网进行通信。

13、如权利要求 11 所述的方法, 其中, 有线局域网是 802.3 有线局域网。

14、如权利要求 11 所述的方法, 其中, 无线局域网是 802.11 无线局域网。

15、一种选择最佳适配器的方法, 该方法包括:

将有线局域网通信适配器的介质访问控制地址设置为与无线局域网通信适配器的介质访问控制地址相同的介质访问控制地址;

将有线局域网通信适配器的包过滤模式设置为所有包被接受的不加区分模式;

查询连接状态;

检测连接状态信息;

基于所述连接状态信息选择最佳适配器; 和

映射选择的最佳适配器的信息。

16、如权利要求 15 所述的方法, 其中, 如果有线局域网和无线局域网都被连接, 则最佳适配器是有线局域网, 否则最佳适配器是被连接至其的那一个。

17、如权利要求 16 所述的方法, 其中, 有线局域网是 802.3 有线局域网。

18、如权利要求 16 所述的方法, 其中, 无线局域网是 802.11 无线局域网。

通信适配器切换方法和设备

本申请要求在韩国知识产权局于 2004 年 9 月 14 日提交的第 10-2004-0073299 号以及于 2004 年 12 月 7 日提交的第 10-2004-0102304 号韩国专利申请的利益，这些申请完整公开于此以资参考。

技术领域

本发明涉及网络领域，更具体地说，涉及一种通信适配器切换方法和设备。

背景技术

在 Windows OS 中，可安装多个用于通信的物理适配器。通过将互联网协议(IP)地址分配给每个物理适配器，通信可被执行。通过使用由网络层 IP 地址和传输层端口地址组成的对(pair)来设置会话，可尝试与另一方通信。如果对中的一个被改变，则会话被立即断开。

如果存在相同种类的两个物理适配器(例如，802.3)，则不同的 IP 地址被分配给每个物理适配器。当通信应用连接通信会话时，基于最佳路由路径选择一个 IP 地址，并且经由该 IP 地址被分配给其的物理适配器执行与另一方的通信。例如，即使存在诸如 802.3 和 802.11 的其它的方案的物理适配器，通信仍以如上所述的相同方式执行。

如果当在安装了 Windows OS 并具有不同种类的物理适配器的个人计算机(PC)中同时激活有线局域网(LAN)和无线 LAN 时用户执行通信，并且如果这些物理适配器中的两个被连接到相同的域，则通常将使用通信速度更快的有线 LAN 来执行通信。即，在这种情况下使用的 IP 地址将是分配给有线 LAN 的 IP 地址。这里，如果有线 LAN 被断开，则由于分配的 IP 地址不能再使用，所以 Windows OS 自动断开使用传输控制协议/用户数据报协议(TCP/UDP)通信的应用的会话。

图 1 显示由微软公司提供的用于提供 Windows OS 的组网功能的核心级库。这些库被称为网络驱动程序接口规范(NDIS)驱动程序。实际物理适配器

当其被连接到互连外围设备(PCI)接口时需要驱动程序,该驱动程序是由开发者提供的迷你端口驱动程序以便适合 NDIS 迷你端口规范。因此,实际物理适配器被激活以从另一方(LAN)接收包并将这些包发送到上层,并且从上层接收包并将这些包发送到 LAN。

普及的 TCP/IP 和 NETBIOS 实际上被称为协议驱动程序,并且在上层而不是在 NDIS 层运行。通常,核心级由协议驱动程序处理,并且应用属于用户级。

参照图 1,网络驱动程序系统包括 WIN32 包驱动程序 110、上级(LAN)协议驱动程序 120、NDIS 130、中间驱动程序 140、迷你端口驱动程序 150 和 160、网络接口卡(NIC) 170 和 180。

装置驱动程序将操作系统(OS)连接到输入/输出装置,分析从 OS 接收的一般请求,并将这些请求转换为特定外围控制器能够理解的命令。

NIC 170 和 180 是物理装置,其作为在网络中发送和接收数据帧的网关来运行。

NDIS 130 提供用于在至少一个 NIC 驱动程序和包括 OS 的上层中的至少一个协议驱动程序之间进行通信的接口。即,NDIS 130 提供由上级(TCP/IP)驱动程序和 NIC 驱动程序使用的功能库(称为封装器)。对于各种外部功能,NIC 驱动程序依赖于 NDIS 130,这些外部功能包括:与协议驱动程序通信、注册和截取 NIC 硬件中断、和与在下级的 NIC 170 和 180 通信。

NDIS 驱动程序包括 NIC 驱动程序、中间驱动程序 140、和上级协议驱动程序 120。

NIC 驱动程序管理 NIC。NIC 驱动程序直接被连接到在下级的硬件(NIC)并提供与上面的上级驱动程序的接口。

迷你端口驱动程序 150 和 160,即,一种 NIC 驱动程序,向 NIC 170 和 180 发送数据并从 NIC 170 和 180 接收数据,并且执行管理 NIC 170 和 180 所需要的面向硬件的操作。迷你端口驱动程序 150 和 160 不能直接调用 OS。取而代之,迷你端口驱动程序 150 和 160 可调用由 NDIS 130 导出的功能。此外,迷你端口驱动程序 150 和 160 被连接到上级驱动程序,诸如中间驱动程序 140 和传输协议驱动程序。

中间驱动程序 140 存在于现存的协议驱动程序和迷你端口驱动程序 150 和 160 之间。中间驱动程序 140 对于上级传输驱动程序而言好像迷你端口驱

动程序，对于迷你端口驱动程序 150 和 160 而言好像协议驱动程序。中间驱动程序 140 主要用于在现存的协议驱动程序和管理不被现存的协议驱动程序所识别的新介质类型的迷你端口之间执行介质转换。

由于中间驱动程序 140 在驱动程序中具有中间位置，所以它与在下级的迷你端口驱动程序 150 和 160 通信，以及与上级协议驱动程序 120 通信。中间驱动程序 140 将协议进入点提供给下级以及将迷你端口进入点提供给上级。中间驱动程序 140 用于为 NDIS 130 传送用于上级协议驱动程序 120 的请求。中间驱动程序 140 对于上级协议驱动程序 120 而言好像迷你端口驱动程序。

为了向用户提供服务，上级协议驱动程序 120 提供传输驱动程序接口 (TDI) 或面向应用的接口。上级协议驱动程序 120 通过打包数据、将包分配给下级驱动程序、和调用 NDIS 130 来将包发送到下级驱动程序。此外，上级协议驱动程序 120 提供与下级的协议接口以从下级驱动程序接收包。传输协议驱动程序将接收的数据发送到适当的客户端应用。

如图 2 所示，在当前的 Windows OS 中，如果异类或同类的物理适配器之间的一条通信线路被断开，则其关联的会话被自动断开。在 TCP 的情况下，介质的信息被存储在 TCP 控制块(TCB)中，如果接口断开的信息被输入，则由于分配给该接口的 IP 地址不能再使用，所以 TCP 自动断开其关联的会话。在 Windows OS 中，即使 TCP 未断开该会话，相同的 IP 地址也不能被同时分配给另一接口。如果尝试将相同的 IP 地址分配给另一接口，那么由于 TCP/IP 协议驱动程序自身被拒绝(turn down)/找到(turn up)，所以所有其它现存的会话被断开。

由于这些问题，为了假装当从实际的协议驱动程序的角度看来时仅有一个接口存在，Windows OS 使用可以产生虚拟适配器的桥功能。桥将相同的 IP 地址提供给几个接口。然而，执行换手(handoff)花费大约 30 秒。即，即使使用桥，但如果一个物理适配器的通信被断开，则断开信号被自动输入到由桥产生的虚拟适配器，并且 TCP/IP 协议驱动程序的所有会话被自动断开。此外，在桥方案中，从一个介质切换到另一个花费太多时间。试验结果显示从一个介质切换到另一个花费 30 秒或更多，该时间对应于 TCP 超时时间(timeout time)。因此，即使断开的信息被阻塞，会话由于 TCP 超时仍被断开。

发明内容

本发明另外的方面和/或优点将会在下述的描述中被部分地阐明，并且部分地通过描述而清楚，或者通过实施本发明可被了解。

本发明提供一种通信适配器切换方法和设备，用于当在异类通信适配器之间切换时保持上层中的应用的通信会话并且提供无缝通信。

根据本发明的一方面，提供一种在具有多个通信适配器的系统中的通信适配器切换方法，该方法包括：将多个通信适配器的地址设置为与该多个通信适配器的第一个的相同的地址；将除了第一通信适配器之外的通信适配器的包过滤模式设置为在其中所有包被接受的模式；和在通信期间当一个通信适配器被断开时，从多个通信适配器中的该通信适配器切换到另一个。

根据本发明的另一方面，提供一种在具有多个通信适配器的系统中的通信适配器切换方法，该方法包括：设置有线 LAN 通信适配器的介质访问控制 (MAC) 地址与无线 LAN 通信适配器的 MAC 地址相同，并且将有线 LAN 通信适配器的包过滤模式设置为在其中所有包被接受的不加区分模式；和在使用所述有线 LAN 通信适配器进行通信期间当所述有线 LAN 被断开时，使用无线 LAN 信息从所述有线 LAN 通信适配器切换到所述无线 LAN 通信适配器。

所述通信适配器切换方法还可包括：在使用无线 LAN 通信适配器进行通信期间当通知或指示有线 LAN 被连接的事件产生时，使用有线 LAN 信息从所述无线 LAN 通信适配器切换到所述有线 LAN 通信适配器。

所述通信适配器切换方法还可包括：将无线 LAN 通信适配器的包过滤模式设置为直接、多点传送、或广播模式。

所述通信适配器切换方法还可包括：如果有线 LAN 通信适配器被断开，则将无线 LAN 信息映射到虚拟适配器。

根据本发明的另一方面，提供一种在具有多个通信适配器的系统中的通信适配器切换设备，该设备包括：虚拟适配器，其将多个通信适配器的地址设置为与该多个通信适配器的第一个的相同的地址，并且将除了第一通信适配器之外的通信适配器的包过滤模式设置为在其中所有包被接受的模式；和虚拟协议驱动程序，其在通信期间当一个通信适配器被断开时，从多个通信适配器中的该适配器切换到另一个。

根据本发明的另一方面，提供一种在具有多个通信适配器的系统中的通

信适配器切换设备, 该设备包括: 虚拟适配器, 其设置有线 LAN 通信适配器的 MAC 地址与无线 LAN 通信适配器的 MAC 地址相同, 并且将有线 LAN 通信适配器的包过滤模式设置为在其中所有包被接受的不加区分模式; 和虚拟协议驱动程序, 其在使用所述有线 LAN 通信适配器进行通信期间当所述有线 LAN 被断开时, 使用无线 LAN 信息从所述有线 LAN 通信适配器切换到所述无线 LAN 通信适配器。

所述虚拟协议驱动程序可包括: 链路状态检测器, 用于每当连接或断开事件产生时提供连接或断开状态。

所述虚拟协议驱动程序还可包括: 最佳适配器选择器, 其基于由链路状态检测器提供的信息确定用于通信的最佳适配器。

从物理适配器接收的包可被无条件地发送到所述虚拟适配器, 并且将被发送到物理适配器的包可被发送到被确定的最佳适配器。

附图说明

通过下面结合附图对实施例进行的描述, 本发明的这些和/或其他方面和优点将会变得清楚和更易于理解, 其中:

图 1 是传统网络驱动程序系统的方框图;

图 2 是传统网络驱动程序的方框图;

图 3 是根据本发明实施例的网络驱动程序系统的方框图;

图 4 是图 3 中显示的中间驱动程序的方框图;

图 5A 和图 5B 是示出根据本发明实施例的虚拟驱动程序加载过程的流程图;

图 6 是示出根据本发明实施例的连接操作的流程图; 和

图 7 是示出根据本发明实施例的断开操作的流程图。

具体实施方式

现在对本发明实施例进行详细的描述, 其示例表示在附图中, 其中, 相同的标号始终表示相同部件。下面通过参照附图对实施例进行描述以解释本发明。

在下文中, 将参照附图更充分地描述本发明, 附图中显示了本发明的实施例。

图 3 是根据本发明实施例的网络驱动程序系统的方框图。

根据本实施例，实际迷你端口驱动程序仅与包括在互联网驱动程序中的虚拟协议驱动程序绑定。虚拟迷你端口驱动程序被产生，虚拟协议驱动程序与该虚拟迷你端口驱动程序绑定。然后诸如 TCP/IP 的实际协议驱动程序仅与该虚拟迷你端口驱动程序绑定。因此，实际协议驱动程序仅用来自虚拟驱动程序的信息得到连接状态的信息，并通过虚拟驱动程序发送和接收所有包。

参照图 3，网络驱动程序系统包括根据本实施例的中间驱动程序 400、NIC 350 和 360、802.3 迷你端口驱动程序 330、802.11 迷你端口驱动程序 340、协议驱动程序 310、和 NDIS 320。

中间驱动程序 400 包括在下级的虚拟协议驱动程序 420 和在上级的虚拟迷你端口驱动程序 410。即，在处于中间驱动程序 400 的下级的虚拟协议驱动程序 420 与迷你端口驱动程序 330 和 340 之间的相互关系中，中间驱动程序 400 具有协议驱动程序的功能，并且在处于中间驱动程序 400 的上级的虚拟迷你端口驱动程序 410 与协议驱动程序 310 之间的相互关系中，中间驱动程序 400 具有迷你端口驱动程序的功能。虚拟迷你端口驱动程序 410 好像是用于协议驱动程序 310 的适配器。产生虚拟适配器以防止 TCP/IP 协议驱动程序的会话由于实际物理适配器的断开信息而被断开。即，从实际物理适配器到协议驱动程序 310 的断开信息的传输可通过在中间驱动程序 400 中进行过滤而被防止。因此，TCP 会话被连续地维持。

此外，如果存在两个实际物理适配器，那么由于两个 IP 地址必须被分别分配给该两个适配器，所以通信的连续性不能被保证。因此，中间驱动程序 400 被用作链路到该两个物理适配器的虚拟适配器。因此，由于位于协议驱动程序 310 上的应用层(未显示)与一个虚拟适配器通信，所以该应用层不关心位于虚拟驱动程序下的物理适配器的类型和状态。

因此，在中间驱动程序 400 中，可改变包路由路径。即，在中间驱动程序 400 中，基于情况选择通信介质，并且经由物理适配器发送和接收包。因此，有线和无线通信几乎可被实时切换，而不影响协议驱动程序 310。

图 4 是图 3 所示的中间驱动程序 400 的方框图。

参照图 4，中间驱动程序 400 包括虚拟迷你端口驱动程序 410 和虚拟协议驱动程序 420。

虚拟迷你端口驱动程序 410 包括虚拟适配器控制器 411。虚拟适配器控

制器 411 产生虚拟适配器并控制该虚拟适配器以便协议驱动程序 310 与该虚拟适配器绑定并使用该虚拟适配器。

此外，当虚拟协议驱动程序 420 被绑定到物理适配器时，每个物理适配器的属性被设置。这里，虚拟适配器控制器 411 将相同的地址设置到每个绑定的物理适配器。当执行设置时，在有线 LAN 的情况下，包过滤值被设置为不加区分模式，在该模式中所有包被接受，并且在无线 LAN 中，包过滤值被设置为直接/多点传送/广播模式，如通常所使用的情況。虚拟适配器的 MAC 地址被设置为无线 LAN 的 MAC 地址。即，在有线 LAN 的情况下，不加区分模式被设置并且有线 LAN 的 MAC 地址看起来没有被设置，并且在无线 LAN 的情况下，由于使用了一般模式，所以虚拟适配器的 MAC 地址依赖于无线 LAN 的 MAC 地址。因此，在这两种情况下，在协议驱动程序 310 这方面，仅仅无线 LAN 的 MAC 地址呈现出被设置。

通过使用相同的 MAC 地址可防止为了更新集线器地址解析协议(ARP)表而重新传输 ARP 包。此外，当使用动态主机配置协议(DHCP)来执行动态地址分配时，通过作为 DHCP 使用的一个选项，MAC 地址被用来对 PC 区分。在这种情况下，因为 DHCP 服务器由于连接的 PC 使用相同的 MAC 地址而将它们识别为相同的 PC，所以虚拟适配器可毫无问题地使用在 DHCP 环境中。此外，当根据使用 MAC 地址的自动配置而分配互联网协议版本 6 (IPv6)地址时，在下一代互联网 IPv6 中，由于 MAC 地址都是相同的，所以可以毫无问题地执行地址分配。

当虚拟适配器控制器 411 从上层中的协议驱动程序请求信息时，虚拟适配器控制器 411 将由最佳适配器选择器 421 选择的最佳适配器信息报告给上层中的协议驱动程序。即，虚拟适配器控制器 411 选择下层链路状态信息并将该选择的信息报告给上层中的协议驱动程序。此外，当虚拟适配器控制器 411 将包发送到物理适配器时，虚拟适配器控制器 411 使用由最佳适配器选择器 421 选择的最佳适配器的绑定句柄来发送这些包。

虚拟协议驱动程序 420 包括最佳适配器选择器 421，链路状态检测器 422、和适配器绑定单元 423。

最佳适配器选择器 421 根据从链路状态检测器 422 输入的有线或无线 LAN 的连接信息来确定用于通信的物理适配器。

链路状态检测器 422 检测有线或无线 LAN 的链路状态信息并将检测到的

链路状态信息输出到最佳适配器选择器 421。

适配器绑定单元 423 绑定所有激活的适配器并产生绑定列表 424。适配器绑定单元 423 包括用于控制绑定的物理适配器的绑定句柄。在本实施例中，适配器绑定单元 423 包括用于有线 LAN 的 802.3 绑定句柄 425 和用于无线 LAN 的 802.11 绑定句柄 426。每个绑定句柄类似于用于虚拟适配器与下级中的迷你端口驱动程序进行通信的密钥。此外，当适配器绑定单元 423 接收包时，它无条件地将这些包发送到虚拟适配器。

图 5A 和图 5B 是示出根据本发明实施例的使用 DriverEntry 功能初始化中间驱动程序 400 的过程的流程图。DriverEntry 功能为 ARP 分配包池和缓冲池，注册用于迷你端口驱动程序 330 和 340 的驱动程序句柄以及用于协议驱动程序 310 的协议句柄，并且向 NDIS320 通知所述驱动程序句柄与所述协议句柄相关联。图 5A 示出初始化中间驱动程序 400 的过程。

参照图 5A，在操作 510 中，中间驱动程序 400 的初始化从调用 DriverEntry 功能开始，该功能通过 OS 作为中间驱动程序 400 的进入点来操作。

在操作 520 中，虚拟适配器控制器 411 绑定 802.3 适配器。每当虚拟适配器控制器 411 绑定物理适配器时，虚拟适配器数据结构被产生以存储物理适配器的信息。因此，物理适配器的数量是多少，就有相同数量的数据结构被产生。该虚拟适配器数据结构与上层中的实际协议驱动程序绑定。通过将数据结构与物理适配器绑定来获得物理适配器的绑定句柄。

虚拟适配器数据结构中的绑定句柄数据结构存储下级中的多个数据结构的列表。

在 802.3 适配器绑定过程中，802.3 有线 LAN 适配器的包过滤属性在操作 521 中被设置为不加区分模式。不加区分模式下的适配器接受它接收到的所有包。

在操作 522 中，有线 LAN 适配器的地址被设置为 802.11 层 2 无线 LAN 适配器的 MAC 地址。

因此，802.3 有线 LAN 适配器具有 802.11 无线 LAN 适配器的 MAC 地址，并可接收通过有线 LAN 输入的所有包。

在操作 530 中，虚拟适配器控制器 411 绑定 802.11 适配器。

在操作 531 中，802.11 无线 LAN 适配器的包过滤属性被设置为直接、多点传送、或广播模式，如通常所使用的情況。

在直接模式下，被发送到特定机器的帧具有分配为目的地址的目的机器的物理地址(以太网地址)。具有该物理地址的机器接受该帧，其它机器不接受该帧。

在多点传送模式下，被发送到一批特定机器的帧(多点传送帧)具有分配为目的地址的多点传送地址。这批特定机器形成多点传送组。因此，包括在该多点传送组中的每一台机器可接受该帧。即使 NIC 不属于该多点传送组，它仍可被编程从而进入这样的多点传送模式，在该模式下，所有多点传送帧被接受。

在广播模式下，被发送到网络中的所有机器的帧被广播到具有目的地址“0xffffffff”的网络。该目的地址“0xffffffff”被视为帧广播地址。在广播模式下的 NIC 接受具有目的地址“0xffffffff”的所有帧。通常，所有 NIC 被配置以便广播帧能被接受。

在为所有物理适配器执行了绑定之后，虚拟适配器控制器 411 在操作 540 中产生虚拟适配器，并在操作 550 中初始化该虚拟适配器。虚拟适配器的初始化包括：通过分配绑定句柄数据结构来存储绑定句柄信息；产生绑定句柄列表；和将绑定的适配器的绑定句柄信息和点信息存储在该绑定句柄列表中。

参照图 5B，在操作 560 中，使用虚拟协议驱动程序 420 来查询连接状态。

包括在已经接收所述查询的虚拟协议驱动程序 420 中的链路状态检测器 422 在操作 570 中检测 802.3 连接状态信息和 802.11 连接状态信息，并将该检测到的连接状态信息发送到最佳适配器选择器 421。

在操作 580 中，最佳适配器选择器 421 选择最佳适配器。当有线 LAN 和无线 LAN 两者都被连接时，由于有线 LAN 更快，所以有线 LAN 被选择作为最佳适配器，当没有有线 LAN 被连接时，无线 LAN 被选择作为最佳适配器。最佳适配器选择器 421 将该选择的最佳适配器的信息发送到虚拟适配器控制器 411。

在操作 590 中，虚拟适配器控制器 411 映射接收的被选择的最佳适配器的信息。然后，中间驱动程序 400 准备好进行通信。

图 6 是示出根据本发明实施例的连接操作的流程图。

参照图 6，当在操作 601 中 802.3 有线 LAN 和 802.11 无线 LAN 都被断开时，如果在操作 602 中产生连接事件，则在操作 603 中虚拟迷你端口驱动程序 410 将连接状态报告给上层中的协议驱动程序 310。上层协议驱动程序

310 在操作 604 中确定连接的适配器是否是 802.3 适配器，并且如果该连接的适配器是 802.3 适配器，则在操作 607 中上层协议驱动程序 310 用 802.3 有线 LAN 进行通信。当用 802.3 有线 LAN 进行通信时，如果在操作 605 中产生了 802.11 连接事件，则在操作 606 中更新 802.11 连接状态信息。

如果在操作 604 中连接的适配器是 802.11 适配器而不是 802.3 适配器，则在操作 608 中 802.11 适配器信息被映射到虚拟适配器。在操作 609 中，上层协议驱动程序 310 用 802.11 无线 LAN 进行通信。

当用 802.11 适配器进行通信时，如果在操作 610 中产生了 802.3 连接事件，则在操作 611 中 802.3 适配器信息被映射到虚拟适配器。在操作 612 中，上层协议驱动程序 310 用 802.3 有线 LAN 进行通信。

图 7 是示出根据本发明实施例的断开操作的流程图。

参照图 7，当在操作 701 中上层协议驱动程序 310 用 802.3 有线 LAN 进行通信时，如果在操作 702 中产生断开事件，则在操作 703 中虚拟适配器控制器 411 确定断开的适配器是否是 802.3 适配器，并且如果该断开的适配器是 802.3 适配器，则在操作 704 中 802.11 适配器信息被重新映射到虚拟适配器。然后，在操作 705 中，上层协议驱动程序 310 用 802.11 无线 LAN 进行通信。

如果断开的适配器不是 802.3 适配器，则在操作 706 中检查 802.3 有线 LAN 的连接状态。

如果发现 802.3 有线 LAN 是连接的，则在操作 707 中上层协议驱动程序 310 用 802.3 有线 LAN 进行通信。

如果发现 802.3 有线 LAN 是断开的，则在操作 708 中虚拟迷你端口驱动程序 410 将断开状态报告给上层协议驱动程序 310，并且在操作 709 中 802.3 适配器信息被重新映射到虚拟适配器。然后，802.3 有线 LAN 和 802.11 无线 LAN 都处于断开状态。

如上所述，根据本发明的实施例，产生用于异类通信介质的唯一的虚拟适配器，并且一个 IP 地址被分配给该产生的虚拟适配器。当有线 LAN 和无线 LAN 都被连接时，高速的有线 LAN 被使用，并且如果该有线 LAN 被断开，则断开被自动检测并且包被重新路由到无线 LAN。因此，诸如 TCP/IP 的协议驱动程序仅仅识别虚拟适配器。因此，由于即使端口地址改变 IP 地址也不改变，所以可保持连续的会话。

即，由于 TCP/IP 会话以相同的 IP 地址被保持，所以从有线 LAN 到无线 LAN 的无缝换手是可行的。此外，由于使用相同的 MAC 地址，所以用由 DHCP 分配的 IP 地址，通信连续性被保证。此外，在下一代互联网 IPv6 中，在无状态自动配置环境下可进行无缝通信，在所述环境中，根据 MAC 地址产生 IPv6 地址。

通过从计算机可读介质运行程序，本发明可被实现在通用计算机中，所述计算机可读介质包括但不限于以下存储介质，诸如磁存储介质(ROM、RAM、软盘、磁带等)、光可读介质(CD-ROM、DVD 等)、和载波(通过互联网的传输)。本发明可被实现为具有在其中实现了计算机可读程序代码单元的计算机可读介质，以使许多经由网络而连接的计算机系统实现分布式处理。用于实现本发明的功能性程序。代码和代码段可由本发明所属的技术领域中的程序员容易地推导出。

尽管已参照其优选实施例显示和描述了本发明，但本领域的技术人员应该理解，在不脱离由所附权利要求限定的本发明的精神和范围的情况下，可对本发明进行形式和细节上的各种改变。这些优选实施例应该仅仅被认为是描述意义上的，而不是为了限制的目的。因此，本发明的范围不是由本发明的详细描述所限定，而是由所附权利要求限定，该范围内的所有差别将被解释为被包括在本发明中。

尽管已显示和描述了本发明的一些实施例，但本领域的技术人员应该理解，在不脱离由所附权利要求及其等同物限定其范围的本发明的原理和精神的情况下，可对这些实施例进行改变。

图 1

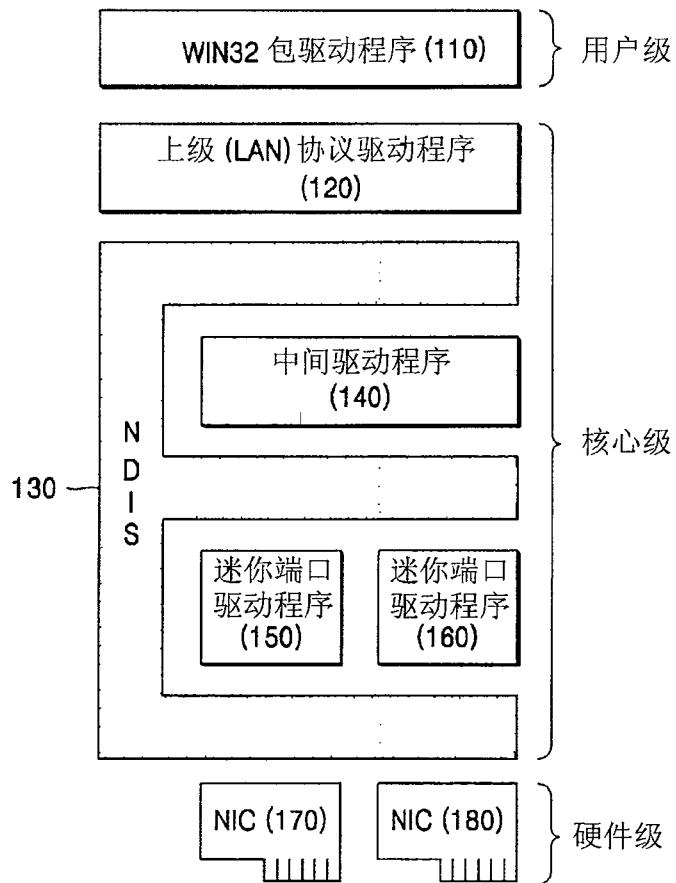


图 2

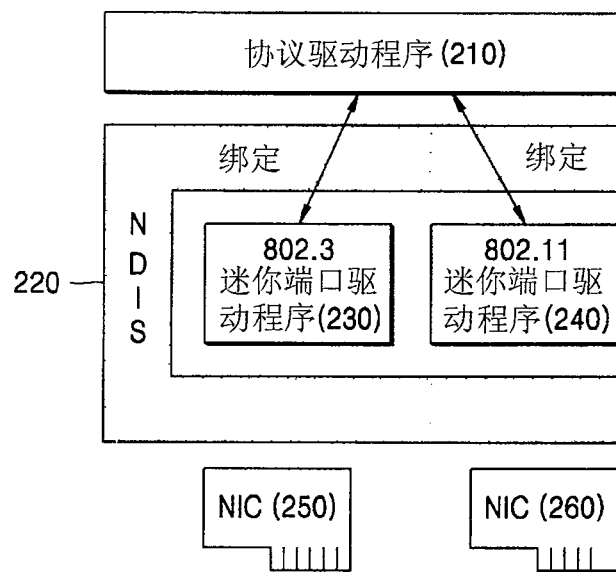


图 3

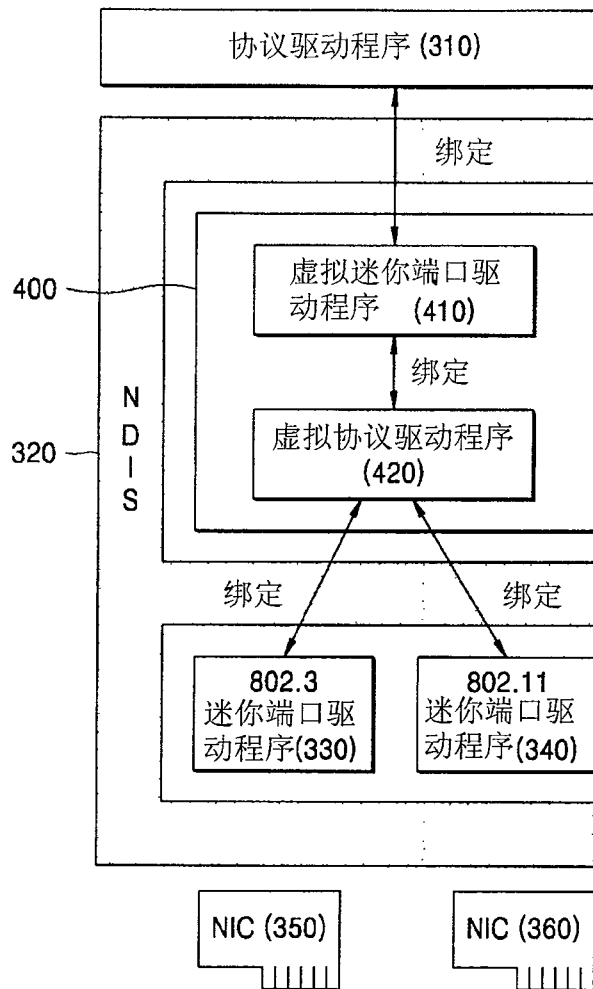


图 4

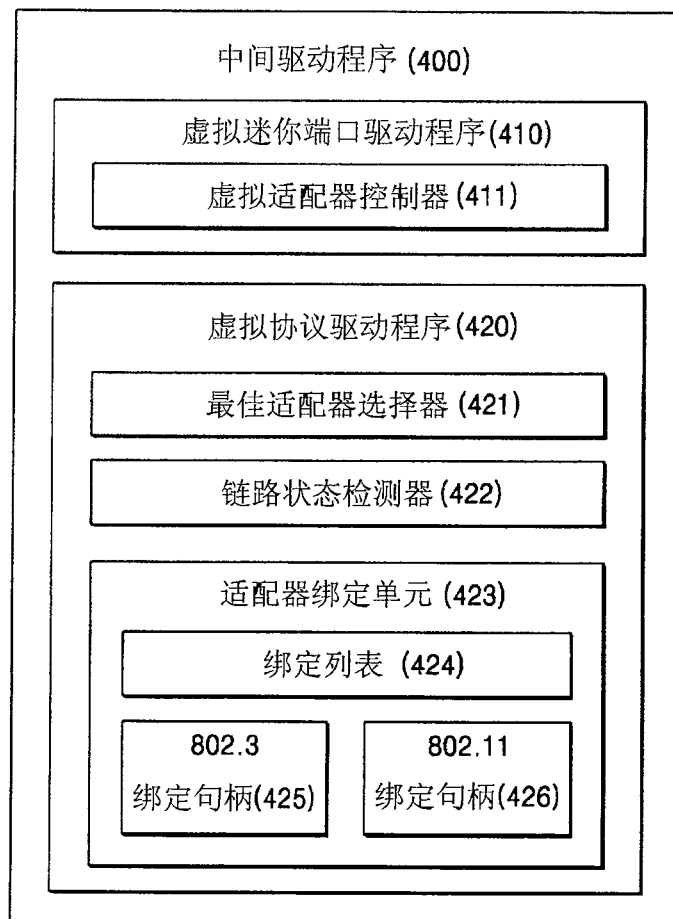


图 5A

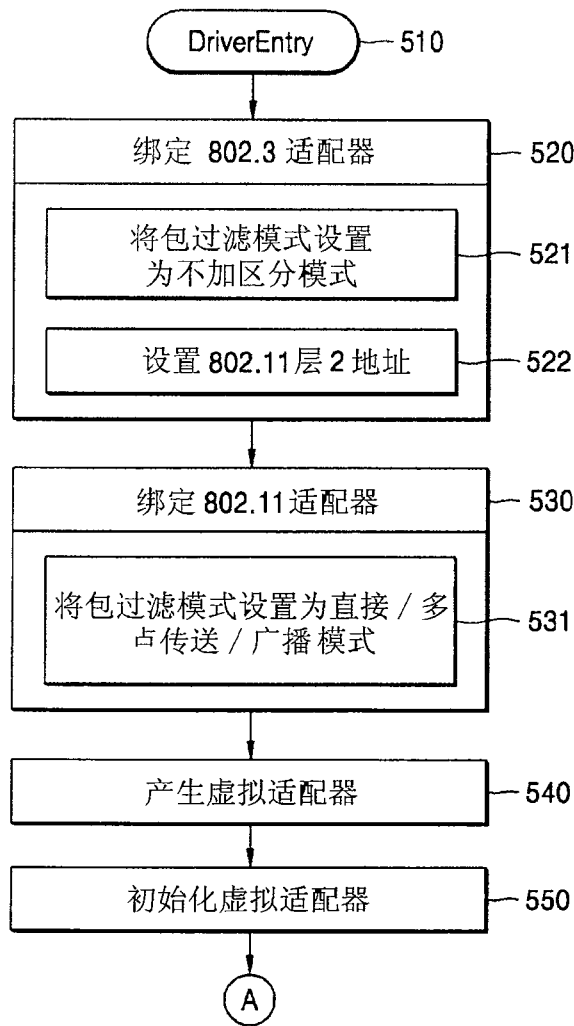


图 5B

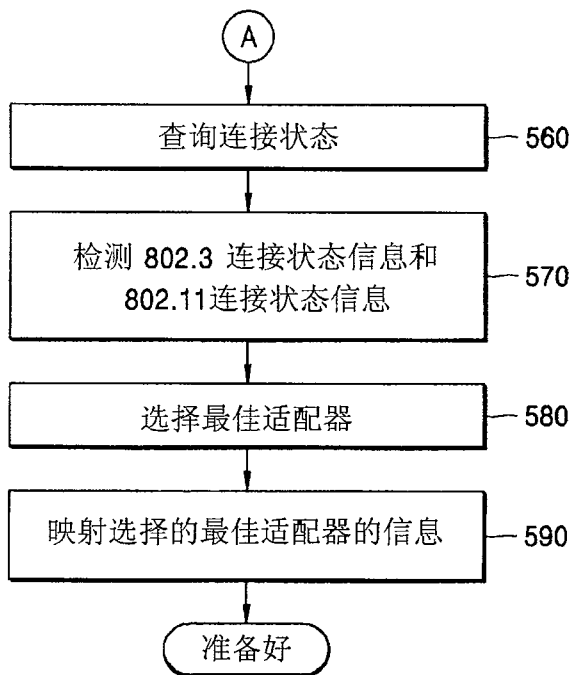


图 6

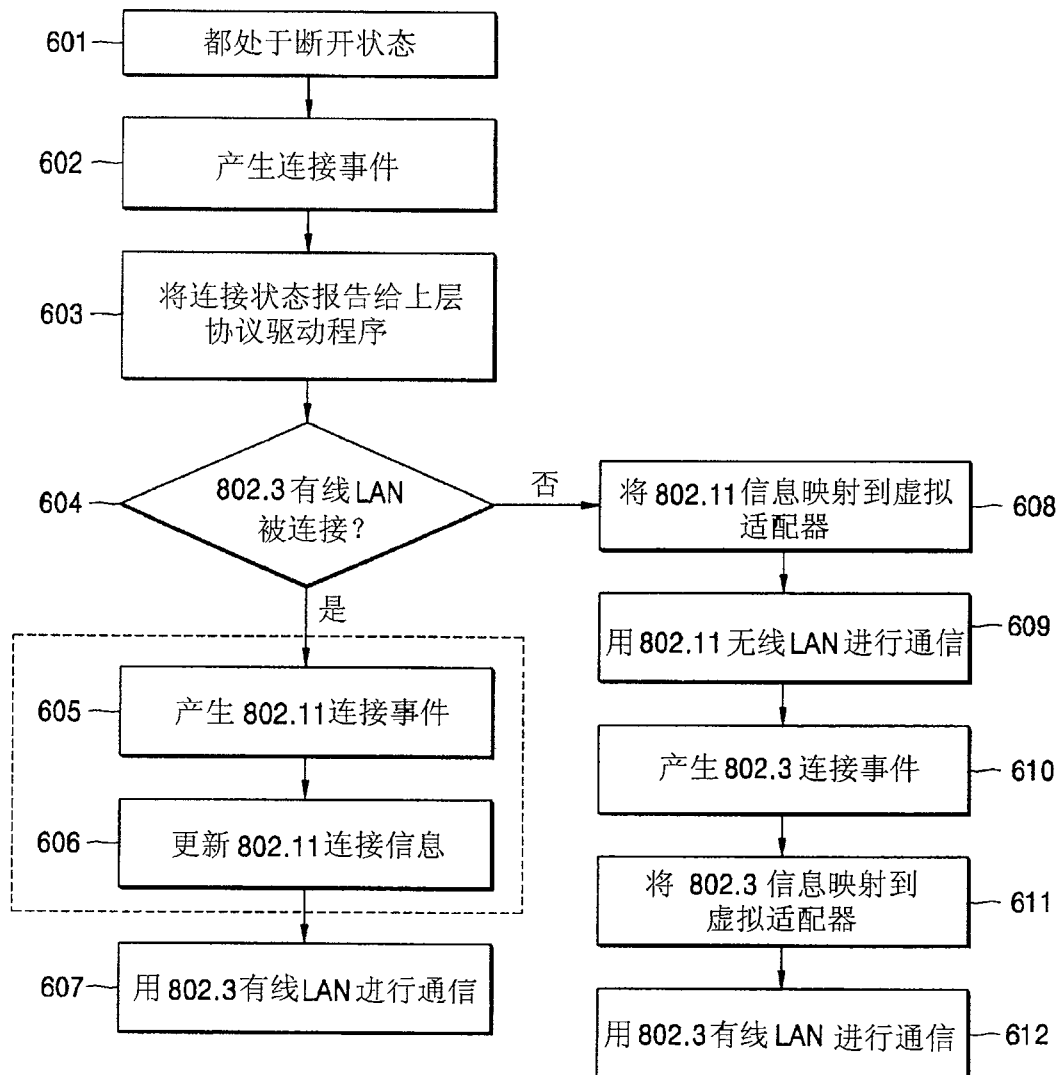


图 7

