



(12) 发明专利

(10) 授权公告号 CN 101689103 B

(45) 授权公告日 2013.04.24

(21) 申请号 200880022512.0

(56) 对比文件

(22) 申请日 2008.06.07

US 6216196 B1, 2001.04.10, 说明书第3栏
45行 - 第7栏 51行.

(30) 优先权数据

11/823,239 2007.06.27 US

US 6216196 B1, 2001.04.10, 说明书第3栏
45行 - 第7栏 51行.

(85) PCT申请进入国家阶段日

2009.12.28

WO 0184301 A2, 2001.11.08, 全文.

CN 1609788 A, 2005.04.27, 全文.

(86) PCT申请的申请数据

PCT/US2008/066229 2008.06.07

审查员 钟阳雪

(87) PCT申请的公布数据

W02009/002686 EN 2008.12.31

(73) 专利权人 微软公司

地址 美国华盛顿州

(72) 发明人 S·H·米索 D·斯来茨

(74) 专利代理机构 上海专利商标事务所有限公司 31100

代理人 顾嘉运 钱静芳

(51) Int. Cl.

G06F 9/00 (2006.01)

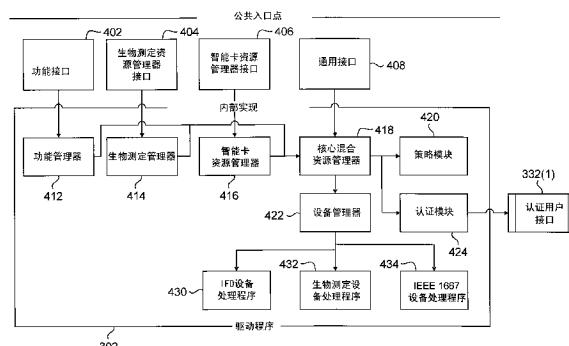
权利要求书2页 说明书7页 附图5页

(54) 发明名称

用于方便管理由计算系统使用的硬件资源的方法和系统

(57) 摘要

提供了供在计算环境中使用的混合资源管理器。该混合资源管理器用作单个资源管理器，其与操作系统协作以便管理与多功能硬件设备的各种功能相关联的各个设备驱动程序中的每一个。在一个示例中，该混合资源管理器实现用于防止在各种应用程序同时请求多个功能时发生冲突的一致的管理和策略框架。



1. 一种用于方便管理由计算系统使用的硬件资源的方法,包括:

通过多功能 API 接受来自资源消费者的访问由多功能硬件设备执行的任何单独功能的资源获取请求,其中所述多功能 API 被配置成方便 (i) 由所述多功能硬件设备执行的功能的设备枚举以及 (ii) 出于认证和验证目的的设备获取以及 (iii) 在获取所述多功能硬件设备后锁定所述多功能硬件设备;

与同所述多功能硬件设备相关联并专用于所请求的单独功能的设备驱动程序进行通信;以及

响应于与所述设备驱动程序的通信,出于允许根据所述资源获取请求来访问多功能硬件设备的目的而获取所述多功能硬件设备。

2. 如权利要求 1 所述的方法,其特征在于,还包括确定所述资源获取请求是否遵从为由所述多功能硬件设备提供的功能建立的策略。

3. 如权利要求 2 所述的方法,其特征在于,还包括当在请求由所述多功能硬件设备执行的第一功能的第一资源消费者和请求由多功能硬件设备执行的第二功能的第二资源消费者之间发生策略冲突时,拒绝获取所述多功能硬件设备。

4. 如权利要求 1 所述的方法,其特征在于,所述资源获取请求从应用程序级资源消费者接收。

5. 如权利要求 1 所述的方法,其特征在于,还包括使用多个设备处理程序中的一个来实现混合资源管理器和所述设备驱动程序之间的通信协议,其中所述设备处理程序中的每一个都专用于所述多功能硬件设备的不同的功能资源。

6. 一种用于管理由计算设备使用的硬件资源的系统,包括:

多功能接口模块,其从资源消费者接受对由多功能硬件设备执行的每一个功能的消费者功能调用,所述多功能接口模块被配置成方便 (i) 由所述多功能硬件设备执行的功能的设备枚举以及 (ii) 出于认证和验证目的的设备获取以及 (iii) 在获取所述多功能硬件设备后锁定所述多功能硬件设备;

混合资源管理器模块,其用于通过管理管理专用于由所述多功能硬件设备执行的每一个功能的不同的设备驱动程序,响应于所述消费者功能调用来分配所述多功能硬件设备的不同功能资源;以及

策略模块,其用于建立至少一个预定策略,所述混合资源管理器将实现这些策略以允许出于允许根据所述消费者功能调用来访问所述多功能硬件设备的目的而获取所述多功能硬件设备。

7. 如权利要求 6 所述的系统,其特征在于,还包括:

至少一个专用接口模块,其接受来自挑选出的资源消费者对由所述多功能硬件设备执行的一个特定功能的请求,其中所述至少一个专用接口模块允许在通过所述多功能接口模块打开的情况下是可用功能的子集的上下文;以及

至少一个专用资源管理器模块,其用于通过管理专用于所述特定功能的设备驱动程序来分配对应于所述特定功能的资源。

8. 如权利要求 7 所述的系统,其特征在于,所述专用接口模块遵从与由所述多功能硬件设备执行的特定功能相关联的 API 标准。

9. 如权利要求 6 所述的系统,其特征在于,所述策略模块被配置成建立用于实现资源

分配的预定冲突解决策略。

10. 如权利要求 9 所述的系统,其特征在于,所述预定冲突解决策略包括基于优先级的解决方案。

11. 如权利要求 6 所述的系统,其特征在于,还包括认证模块,其用于集中认证所述多功能硬件设备。

12. 如权利要求 11 所述的系统,其特征在于,还包括设备管理器模块,其用于代表所述多功能硬件设备来接受用于调用所述认证模块的内容调用。

13. 如权利要求 6 所述的系统,其特征在于,所述混合资源管理器模块驻留在操作系统内的系统层。

14. 如权利要求 11 所述的系统,其特征在于,所述认证模块被配置成独立于所分配的功能资源来呈现公共认证用户接口。

15. 如权利要求 6 所述的系统,其特征在于,所述计算设备是选自由个人计算机、游戏控制台、蜂窝电话、PDA 和机顶盒 STB 组成的组的电子设备。

16. 一种用于方便管理由计算系统使用的硬件资源的系统,包括:

用于通过多功能 API 接受来自资源消费者的访问由多功能硬件设备执行的任何单独功能的资源获取请求的装置,其中所述多功能 API 被配置成方便 (i) 由所述多功能硬件设备执行的功能的设备枚举以及 (ii) 出于认证和验证目的的设备获取以及 (iii) 在获取所述多功能硬件设备后锁定所述多功能硬件设备;

用于与同所述多功能硬件设备相关联并专用于所请求的单独功能的设备驱动程序进行通信的装置;以及

用于响应于与所述设备驱动程序的通信,出于允许根据所述资源获取请求来访问所述多功能硬件设备的目的而获取所述多功能硬件设备的装置。

17. 如权利要求 16 所述的系统,其特征在于,还包括用于确定所述资源获取请求是否遵从为由所述多功能硬件设备提供的功能建立的策略的装置。

18. 如权利要求 17 所述的系统,其特征在于,还包括用于当在请求由所述多功能硬件设备执行的第一功能的第一资源消费者和请求由多功能硬件设备执行的第二功能的第二资源消费者之间发生策略冲突时,拒绝获取所述多功能硬件设备的装置。

用于方便管理由计算系统使用的硬件资源的方法和系统

技术领域

[0001] 本发明涉及用于方便管理由计算系统使用的硬件资源的方法和系统。

背景技术

[0002] 操作系统采用资源管理方案来限制应用程序之间的干扰、实现排定资源分配的优先顺序并以其他方式控制资源分配的策略，以及一般地管理运行许多独立软件应用程序的系统的总体行为。通常，为每一类设备提供资源管理器。设置设备种类的示例包括外围设备，诸如键盘、打印机、传真机、扫描仪、CD 和 DVD 播放器、网络接口、智能卡读取器、生物测定扫描仪和存储器存储设备(例如闪存)。不同种类的设备中的每一个设备通常都具有其自己的资源管理器。因为给定种类中的特定设备一般彼此之间略有不同，所以通常为每一个特定硬件设备提供被称为设备驱动程序的附加软件组件。设备驱动程序用作资源管理器和特定硬件设备之间的转换层。

[0003] 设备驱动程序在计算机系统中通常用作用于与硬件设备通信的最底层的软件组件。通常，这些设备驱动程序以一对一关系与其控制的设备相关联。例如，串行端口驱动程序与计算机的数据 / 传真串行调制解调器相关联、综合切换数字网 (ISDN) 驱动程序与 ISDN 接口设备相关联，并且网络驱动程序提供到广域网 (WAN) 数据接口设备的接口。在这一安排中，每一个专用驱动程序都管理对硬件设备的启动、关闭、维护和运行操作。

[0004] 由于技术的进步，将多个硬件设备集成到单个多功能设备中已变为可能。例如，单个硬件设备可能能够执行数据 / 传真调制解调器、ISDN 和 WAN 接口的功能中的一个或全部。另一示例可以是具有集成的智能卡读取器和生物测定扫描仪的键盘。这些多功能设备具有胜过多个单功能设备的若干优点，包括降低的成本、较小的尺寸、主机系统上的负载降低以及更容易安装。然而，为了保持与传统应用程序兼容，这些多功能设备仍需要用于每一个功能(例如，键盘、智能卡读取器和生物测定扫描仪等)的专用驱动程序。由于每一个设备驱动程序都独立地运作，因此可能出现多个设备驱动程序向单个硬件设备发送命令的情况。因此，存在存储一个驱动程序发送的命令与另一个驱动程序发送的命令冲突，从而导致系统误动作的可能性。例如，一个驱动程序可能发送关闭硬件设备的命令，而其他独立的驱动程序正在使用该设备。另一示例是其中一个驱动程序试图运行需要独占地访问多功能设备以确保有效结果的诊断测试，而另一驱动程序试图在该诊断测试期间访问该多功能设备。多个设备驱动程序访问单个多功能硬件设备可能发生许多冲突。

[0005] 图 1 示出了计算机系统 10，其具有连接到多功能设备 8 的多个设备驱动程序，即设备驱动程序 A2、设备驱动程序 B4 以及设备驱动程序 C6。在操作中，驻留在计算机系统 10 上的应用程序可请求多功能设备 8 执行第一功能。当请求一功能时，计算机系统 10 指示对应于所请求的功能的驱动程序向多功能设备 8 发送命令以使其能够执行所请求的功能。例如，如果请求功能 A，则计算机系统 10 指示设备驱动程序 A 向多功能设备 8 发送命令以便执行功能 A。然而，如上所述，某些设备驱动程序可能向多功能设备发出冲突的命令。发自不同的设备驱动程序的冲突命令可能由具有多功能设备的系统中的众多情形引起。例如，当

设备驱动程序 A2 正在指示多功能设备 8 执行功能 A 时,计算机系统 10 可以指示设备驱动程序 B4 向多功能设备 8 发出命令以执行功能 B。由设备驱动程序 B4 发出的部分命令可能是与设备 A2 正在发出的命令冲突的设备初始化命令。在这些情况下,由于冲突,该多功能设备可能由于冲突而无法执行所请求的功能中的部分或全部。

[0006] 除了冲突之外,在多功能设备使用多个驱动程序时引发的另一问题涉及安全性。例如,确保在任何给定时间应用程序仅使用多功能设备 8 的一个特定功能并锁定所有其他功能通常是必需的。

[0007] 对该问题的一种可能的解决方案是将设备驱动程序组合到单个多功能驱动程序中,并由此使用编程方法来防止冲突命令。然而,该方法可能要求重新编写现有驱动程序的软件代码中的大部分或全部以便能够像单个驱动程序那样工作。该解决方案还降低了软件的模块性。单个组合驱动程序更大并且更复杂,并因此更难以创建、测试、维护和升级。当考虑驱动程序和多功能设备的所有不同排列和组合时,将多个驱动程序组合到单个多功能驱动程序中会增加所需驱动程序总数。因此,保持驱动程序分开是高度合乎需要的。

发明内容

[0008] 以下详细描述呈现了供在计算环境中使用的混合资源管理器。混合资源管理器用作单个资源管理器,其与操作系统协作以管理与多功能硬件设备的各种功能相关联的各个设备驱动程序中的每一个。在一个示例中,混合资源管理器实现用于防止在各种应用程序同时请求多个功能时发生冲突的一致的管理和策略框架。

[0009] 提供本概述是为了以简化的形式介绍将在以下详细描述中进一步描述的一些概念。该概述不旨在标识所要求保护的主题的关键特征或必要特征,也不旨在用于帮助确定所要求保护的主题的范围。本发明的其它特征和优点在参考附图继续阅读以下对实施例的详细描述后将变得显而易见。

[0010] 本发明涉及一种用于方便管理由计算系统使用的硬件资源的方法,包括:通过多功能 API 接受来自资源消费者的访问由多功能硬件设备执行的任何单独功能的资源获取请求,其中所述多功能 API 被配置成方便 (i) 由所述多功能硬件设备执行的功能的设备枚举以及 (ii) 出于认证和验证目的的设备获取以及 (iii) 在获取所述多功能硬件设备后锁定所述多功能硬件设备;与同所述多功能硬件设备相关联并专用于所请求的单独功能的设备驱动程序进行通信;以及响应于与所述设备驱动程序的通信,出于允许根据所述资源获取请求来访问多功能硬件设备的目的而获取所述多功能硬件设备。

[0011] 本发明涉及一种用于管理由计算设备使用的硬件资源的系统,包括:多功能接口模块,其从资源消费者接受对由多功能硬件设备执行的每一个功能的消费者功能调用,所述多功能接口模块被配置成方便 (i) 由所述多功能硬件设备执行的功能的设备枚举以及 (ii) 出于认证和验证目的的设备获取以及 (iii) 在获取所述多功能硬件设备后锁定所述多功能硬件设备;混合资源管理器模块,其用于通过管理管理专用于由所述多功能硬件设备执行的每一个功能的不同的设备驱动程序,响应于所述消费者功能调用来分配所述多功能硬件设备的不同功能资源;以及策略模块,其用于建立至少一个预定策略,所述混合资源管理器将实现这些策略以允许出于允许根据所述消费者功能调用来访问所述多功能硬件设备的目的而获取所述多功能硬件设备。

- [0012] 本发明涉及一种用于方便管理由计算系统使用的硬件资源的系统，包括：
- [0013] 用于通过多功能 API 接受来自资源消费者的访问由多功能硬件设备执行的任何单独功能的资源获取请求的装置，其中所述多功能 API 被配置成方便 (i) 由所述多功能硬件设备执行的功能的设备枚举以及 (ii) 出于认证和验证目的的设备获取以及 (iii) 在获取所述多功能硬件设备后锁定所述多功能硬件设备；
- [0014] 用于与同所述多功能硬件设备相关联并专用于所请求的单独功能的设备驱动程序进行通信的装置；以及
- [0015] 用于响应于与所述设备驱动程序的通信，出于允许根据所述资源获取请求来访问所述多功能硬件设备的目的而获取所述多功能硬件设备的装置。

附图说明

- [0016] 图 1 示出了具有供与多功能硬件设备一起使用的多个设备驱动程序的计算机系统。
- [0017] 图 2 示出了具有计算单元以及与计算单元连接或以其他方式与其接口的多个外围组件的计算系统。
- [0018] 图 3 示出了可以由图 2 的计算系统实现以方便多功能硬件设备的资源管理的资源管理体系结构的一个示例。
- [0019] 图 4 示出了在图 3 中描述的混合资源管理器的一个具体示例。
- [0020] 图 5 是示出用于方便管理计算系统所使用的硬件资源的方法的一个示例的流程图。

具体实施方式

- [0021] 为了成功地在计算设备上运行应用程序，设备的操作系统必须向应用程序提供各种资源。在以下讨论中详细地描述了适用于实现用于管理这些资源以便于其高效分配和使用的改进的方法的计算系统。
- [0022] 此处所描述的资源管理体系结构可以在许多不同的环境和计算上下文中实现。出于讨论的目的，该体系结构在用于个人计算机的计算系统的上下文中描述，但是同等适用于其他计算环境，包括诸如游戏控制台、蜂窝电话、PDA、机顶盒 (STB) 等各种各样的电子设备。
- [0023] 如下文中详细描述的，提供混合资源管理器，而非提供用于支持多功能硬件设备的单个多功能驱动程序。混合资源管理器可以与多功能硬件设备的各个设备驱动程序一起使用，但提供可以由访问硬件设备的所有应用程序使用的一致的管理和策略框架。
- [0024] 图 2 示出了具有计算单元 22 以及与计算单元 22 连接或以其他方式与之接口的多个外围组件。计算单元 22 具有一个或多个处理器 24(1)、24(2)、…、24(P)、易失性存储器 26 (例如 RAM) 和非易失性存储器 28 (例如 ROM、闪存、硬盘、CD ROM 等)。
- [0025] 操作系统 30 被存储在非易失性存储器 28 中。操作系统 30 是多任务操作系统，其在其被加载到易失性存储器 26 中并在一个或多个处理器 24 上执行时，支持多个应用程序 32(1)、32(2)、…、32(A) 同时执行。一个优选操作系统是微软公司销售的 Windows™ 品牌操作系统。然而，应该注意，可以使用其他操作系统。

[0026] 应用程序 32(1)、32(2)、…、32(A) 是需要访问硬件资源的、可以在系统上加载并执行的不同资源消费者的示例。资源消费者可以包括诸如文字处理应用程序、电子表格应用程序、数据库应用程序、日程安排应用程序、财务应用程序、教育应用程序、娱乐应用程序等应用程序。

[0027] 操作系统 30 具有管理计算系统 20 的资源以便分配给应用程序 32(1)-32(A) 的资源管理系统 40。资源管理系统 40 可以与操作系统 30 分开实现,但是在该示例中被示为集成在操作系统中。以下参考图 3 更详细地描述了资源管理系统 40。

[0028] 操作系统 30 还具有用于计算系统 20 中的各种相关联的外围组件的多个软件驱动程序 42(1)、…、42(D)。一个或多个 COM (通信)端口 44 也被示为是操作系统 30 的一部分。代表性外围组件的集合在计算单元 22 的周围示出。

[0029] USB 总线 54 连接到计算单元 22 以便与许多不同种类的兼容 USB 的外围组件接口。这些组件的示例包括调制解调器 56、扬声器 58、键盘 60 和其他 USB 设备 62。计算单元 22 也耦合到网络 66 以便与其他计算机接口。网络 66 表示许多不同类型的网络,包括 LAN、WAN、因特网、内联网和无线网络。

[0030] 1394 串行总线 68 连接到计算单元 22 以便与许多不同种类的兼容 1394 的外围组件接口。这些组件的示例包括存储器驱动器 70 (例如盘驱动器、磁带驱动器、CD ROM 驱动器等)、调制解调器 72、扬声器 74、CPU (中央处理单元)76 和其他 1394 设备 78。应该注意,虽然在该示例性系统中示出了 USB 和 1394 总线,但可另外地或另选地使用其他总线体系结构,诸如 SCSI、ISA (工业标准体系结构) 和 PCI (外围部件互连) 总线。

[0031] 计算系统 20 具有显示器 80,其可以是电视机或计算机监视器。显示器经由一个或多个显示接口 82(1)、82(2)、…、82(C) 与计算单元 22 接口,这些显示接口表示视频端口、覆盖图和视频存储器。耦合到计算单元 22 的其他示例性外围设备包括 DVD 播放器 84、生物测定扫描仪 86 和智能卡读取器 88。

[0032] 图 2 所示的组件的集合示出了由资源管理系统 40 管理的示例性资源类型。其中有各种硬件设备资源,诸如键盘、调制解调器、USB 驱动器、生物测定扫描仪、智能卡读取器、1394 设备、显示接口、记录器、存储器驱动器等等。许多其他组件可被添加到该系统,并且可以移除所示组件中的一个或多个。如上所述,将这些设备中两个或更多设备的功能组合到单个多功能硬件设备中正变得越来越常见。

[0033] 图 3 示出了可以由图 2 的计算系统 20 实现以方便多功能硬件设备的资源管理的资源管理体系结构 300 的一个示例。体系结构 300 用软件来实现,并且在该示例中包括用户级组件以及内核级组件。体系结构 300 具有混合资源管理器 302 和设备驱动程序 304(1)、304(2)、304(3)、…、304(P),这些设备驱动程序支持一个或多个资源消费者。资源消费者的示例包括用户级资源消费者,诸如应用程序 332(1)、332(2)、332(3)、…、332(N)。如图所示,可以提供的一个应用程序级资源消费者是认证用户接口 332(1),其用于提供应用程序级认证和安全性以确保例如特洛伊木马或其他恶意软件不会欺骗或以其他方式获取对诸如口令、公钥等用户凭证的访问权。虽然未示出,但是也可以提供内核级资源消费者。

[0034] 每一个设备驱动程序 304(1)-304(4) 都与一硬件设备相关联并且跟踪该硬件设备的可用性。即,如上所述,设备驱动程序可以是与硬件设备接口的任何软件组件或模块。设备驱动程序 304(1) 可以是例如允许多功能设备 340 将串行数据传递给调制解调器的串

行驱动程序。设备驱动程序 304(2) 可以是例如允许多功能设备 340 发出和接收呼叫的 ISDN 接口驱动程序, 而设备驱动程序 304(3) 可以是例如允许多功能设备 340 对 WAN 网络接口设备收发数据的 WAN 驱动程序。多功能设备的另一示例可以是集成具有闪存的智能卡读取器和生物测定读取器的键盘。在这种情况下, 设备驱动程序 304(1) 可以是键盘, 设备驱动程序 304(2) 可以是智能卡驱动程序, 设备驱动程序 304(3) 可以是生物测定扫描仪驱动程序, 而设备驱动程序 304(4) 可以是闪存驱动程序。又一多功能设备可以是组合 WAN 数据接口、X.25 数据接口和数据加密处理器的虚拟专用网络(VPN)卡。这些设备驱动程序可能非常复杂, 并且如上所述, 一般最好被实现为单独的驱动程序。当然, 上述多功能设备和设备驱动程序仅是示例性的并且此处所描述的原理可以应用于任何多功能硬件设备以及用于与这些设备接口的驱动程序。本领域的技术人员可以理解, 随着技术的进步, 更多硬件功能将变得可用, 并且将众多硬件功能组合到单个设备中将变为可能, 并且此处所描述的原理可同等地应用于这些以后开发的多功能设备和设备驱动程序。虽然设备驱动程序被示为在内核层, 但是一个或多个设备驱动程序也可以在用户层实现。

[0035] 每一个设备驱动程序 304 都向混合资源管理器 302 注册自己, 并且提供由混合资源管理器 302 使用的一组回调以获取信息。例如, 一个回调用于执行资源计算, 而另一个用于将成功的保留通知给提供者。混合资源管理器 302 允许对混合硬件设备使用相同的上下文并且获取由特定设备资源管理器使用以使得硬件设备能够执行该功能对应的设备种类称号(handle)。

[0036] 混合资源管理器 302 展示用于与体系结构中的其他模块交互的公共接口 320。公共接口 320 包括定义的 API (应用程序编程接口), 其具有由设备驱动程序 304 使用的一组提供者 API 调用和用于从应用程序 332(1)–332(N) 或其他资源消费者接收对资源的请求的一组消费者 API 调用。API 是到程序的接口, 其定义程序将为执行所需任务而接受的输入类别以及程序将在执行所需任务后返回的输出类别。当应用程序 332 想要执行一任务时, 它使用公共接口 320 来在混合资源管理器 302 处创建活动, 并且构建描述执行该活动所需的各资源集的一个或多个配置。活动是与正在系统中执行的任务相关联的数据结构。混合资源管理器 302 确定哪些活动能够从有限的资源池中得到完全满足, 并允许具有可以被满足的活动的应用程序访问所请求的资源。

[0037] 图 4 示出了图 3 所描绘的混合资源管理器 302 的一个具体示例, 其中出于支持与现有应用程序的向后兼容的目的而提供了多个公共接口。对混合资源管理器 302 的访问权是通过公共接口(例如图 3 中的公共接口 320)来获取, 该公共接口包括遵从适当的 API 标准的一系列接口。例如, 在该示例中, 示出了分别用于生物测定读取器和智能卡读取器的标准 API 404 和 406。还示出了功能接口 402, 其表示可从可信的第三方获得的任何其他硬件设备功能(例如闪存)。这些接口不直接访问硬件设备。由于这些接口 402、404 和 406 与当前用于获取对资源管理器的访问权的那些接口相同, 因此维持了向后兼容性。通过公共接口 320 打开的上下文可能由于可由策略模块建立的策略(在下文中更详细地讨论)而无法完全访问所有底层系统硬件。在某些情况下, 混合资源管理器 302 具有维护硬件设备支持的所有设备种类 ID 的列表和对应的上下文的上下文结构。

[0038] 该公共接口还包括通用混合资源接口 408, 其用于方便实现混合资源管理器的增强的功能。即, 也可以被实现为 API 的通用混合资源管理器接口 408 由利用混合资源管理

器 302 的特征的应用程序使用。通用混合资源管理器接口 408 提供诸如如下功能：对于混合设备内的设备的混合设备枚举，其标识由硬件设备提供的不同功能的数量和类型；设备获取，其由计算系统和硬件设备用于验证和认证目的；以及设备锁定，其确保在获取多功能硬件设备之后，给予应用程序（例如应用程序 332）对该多功能硬件设备的适当且可能是独占的访问权以便执行特定操作。

[0039] 通用混合资源管理器接口 408 帮助使得能够在不同的设备资源管理器并因此即设备（例如经由设备驱动程序，例如设备驱动程序 304）之间进行通信。通用混合资源管理器接口 408 还允许在受限视图中打开上下文，这意味着上下文在通过实际公共 API 打开的情况下可以是可用功能的子集。

[0040] 混合资源管理器 302 还包括通过接口访问的专用资源管理器。在图 3 所示的示例中，功能管理器 412、生物测定管理器 414 和智能卡资源管理器 416 分别通过功能接口 402、生物测定接口 404 和智能卡接口 406 来访问。这些接口中的每一个都是可扩展的但完整性是根据安全模块所建立的安全过程而得到验证的，如下所述。专用资源管理器 412、414 和 416 允许向后兼容性并且基本上提供与当前资源管理器相同的功能，但是在诸如上下文分配、外部资源锁定和硬件管理等某些关键方面进行革新。

[0041] 核心混合资源管理器 418 负责管理不同的资源管理器并因此即由能够使用混合资源管理体系结构的多功能硬件设备和应用程序使用的设备之间的访问。专用资源管理器 412、414 和 416 中的每一个都负责其各自到核心混合资源管理器 418 的连接。核心混合资源管理器 418 成为可扩展的以方便产生附加功能以及处理可开发的新型混合硬件设备。核心混合资源管理器 418 仲裁由设备驱动程序提供的对资源（本地或远程）的访问。诸如图 3 中的应用程序 332 等资源消费者请求由驱动程序提供的一个或多个资源的集合，而核心混合资源管理器 418 确定哪些应用程序要使用设备驱动程序的哪些资源。核心混合资源管理器 418 可基于预定的冲突解决机制来作出资源分配判定。

[0042] 核心混合资源管理器 418 强制实施由策略模块 420 建立的安全策略和其他策略。策略模块 420 可以接收将从设备或系统层建立的策略（例如，组或本地机器策略）。可以实现的策略的一个示例如下。例如，在包括智能卡读取器和生物测定扫描仪的多功能设备的情况下，可以建立要求对集成的生物测定扫描仪独占地使用智能卡读取器的策略。如果用户试图使用专用智能卡读取器以及物理上单独的生物测定扫描仪，则策略模块 420 将阻止获取这两个设备。当然，策略模块 420 可以建立确保混合硬件设备适当地运作而没有冲突等的许多其他策略。例如，就冲突解决而言，如果混合资源管理器 302 采用基于优先级的解决方案，则策略模块 420 根据用户或系统所归结的任务的相对重要性来先验地对任务进行排序，从而使得核心资源管理器 302 可以在存在冲突以使得无法为所有任务分配其资源时确定哪一个任务应该获取对资源的访问权。其他可行策略包括首先预订获胜、最近预订获胜、“公平”资源共享、用户挑选什么胜过什么等等。许多不同的策略是可能的。系统或用户设置策略，并且策略管理器模块将它们转换成绝对优先级。

[0043] 混合资源管理器 302 可以包括各种其他模块，诸如在图 3 中示出的那些附加模块。例如，可以出于集中认证硬件设备的目的而提供认证模块 424，以使得无需在每一次不同的应用程序试图访问混合硬件设备时执行单独的认证过程。该认证模块可以代表应用程序（作为上下文调用的结果）调用或由设备管理器模块 422 代表混合硬件设备调用。此外，设

备管理器模块 422 监视和管理混合硬件设备。设备管理器模块 422 还可在混合硬件设备和功能层锁定资源。

[0044] 诸如处理程序 430、432 和 434 等设备功能处理程序使用输入 / 输出控制代码来管理混合资源管理器 302 和混合资源设备之间的详细通信协议。也可以提供认证用户接口或用户体验模块以便集成可能需要的各种用户级认证并为用户提供一致的接口。例如可以在用户、系统和 / 或硬件设备层提供这种认证。

[0045] 图 5 是示出方便管理由计算系统使用的硬件资源的方法的一个示例的流程图。方法从步骤 510 开始,在那里通用混合资源管理器接口接受访问由多功能硬件资源执行的任何单独功能的资源获取请求。在判定步骤 520 中,确定资源获取请求是否遵从为多功能硬件资源所提供的功能而建立的策略。如果否,则在步骤 530 中拒绝访问。如果该请求确实遵从策略,则在步骤 540,混合资源管理器与同多功能硬件资源相关联并专用于所请求的单独功能的设备驱动程序进行通信。响应于与设备驱动程序的通信,在步骤 550,混合资源管理器使得能够出于允许根据资源获取请求来访问多功能硬件设备的目的而获取多功能硬件设备。

[0046] 尽管用对结构特征和 / 或方法动作专用的语言描述了本主题,但可以理解,所附权利要求书中定义的主题不必限于上述具体特征或动作。相反,上述具体特征和动作是作为实现权利要求的示例形式公开的。

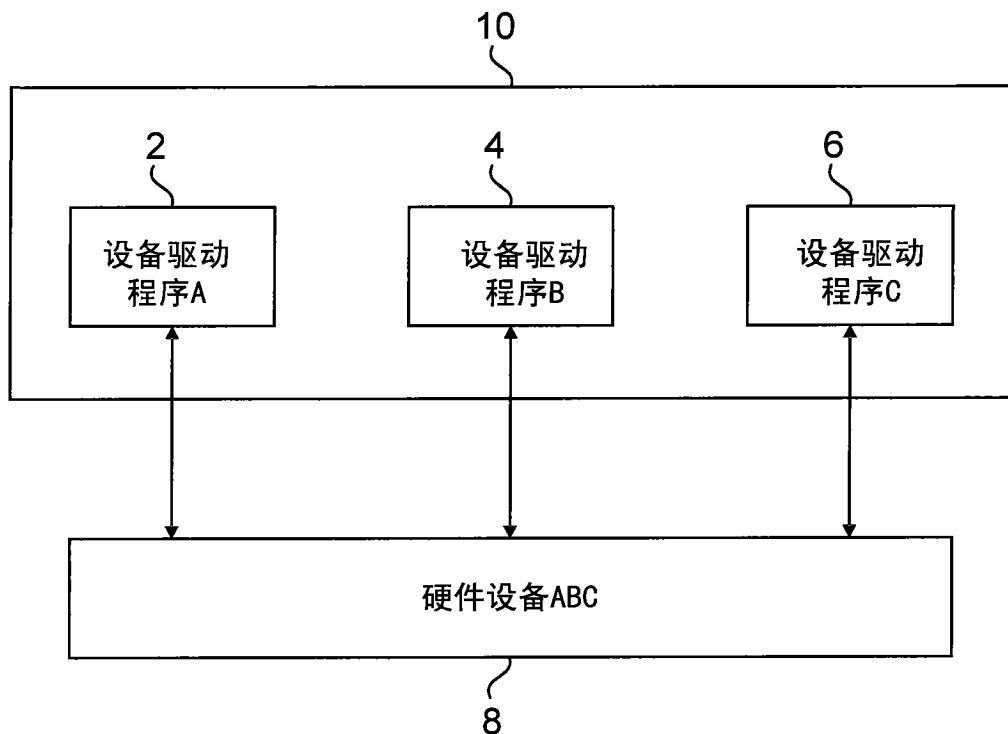


图 1

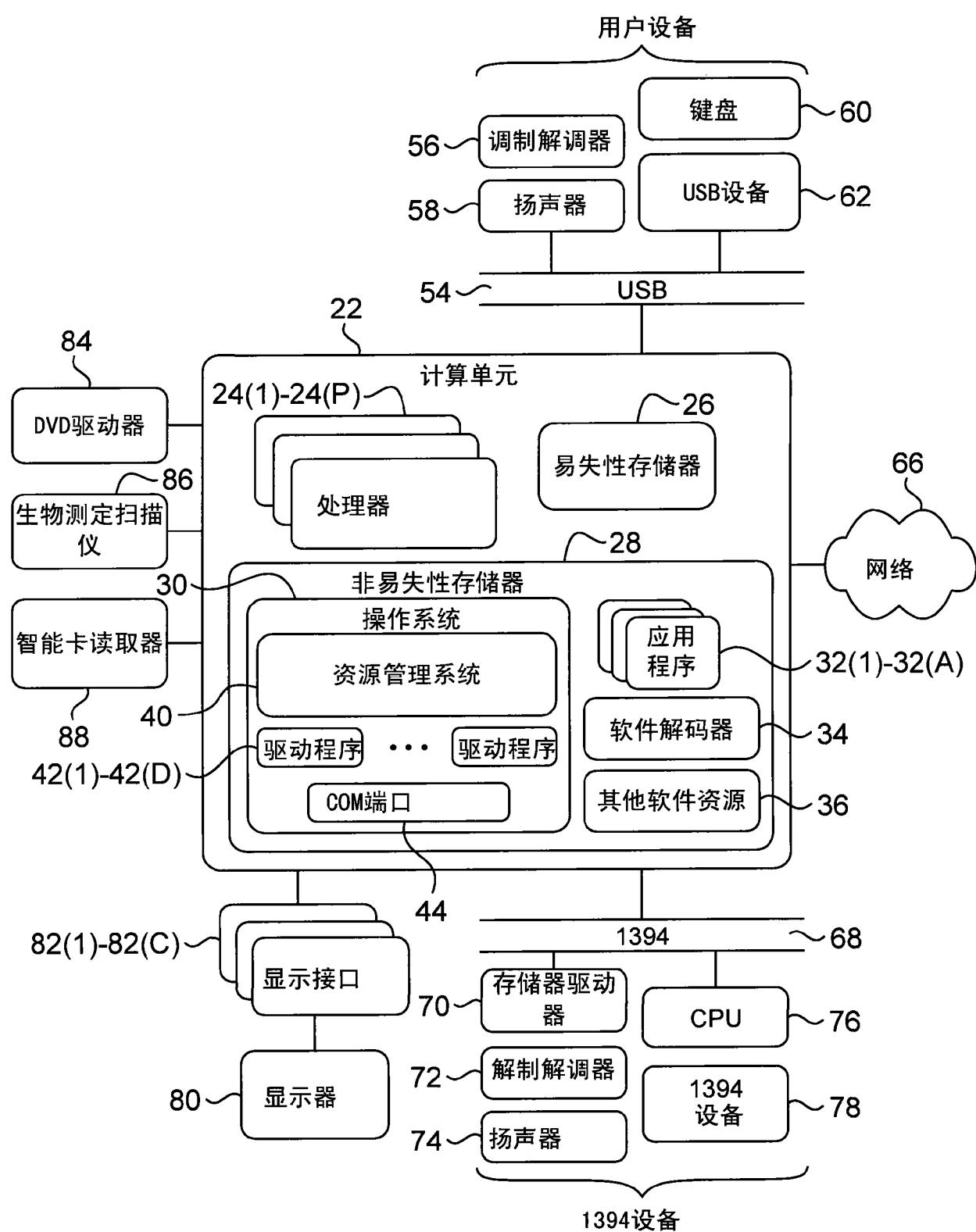


图 2

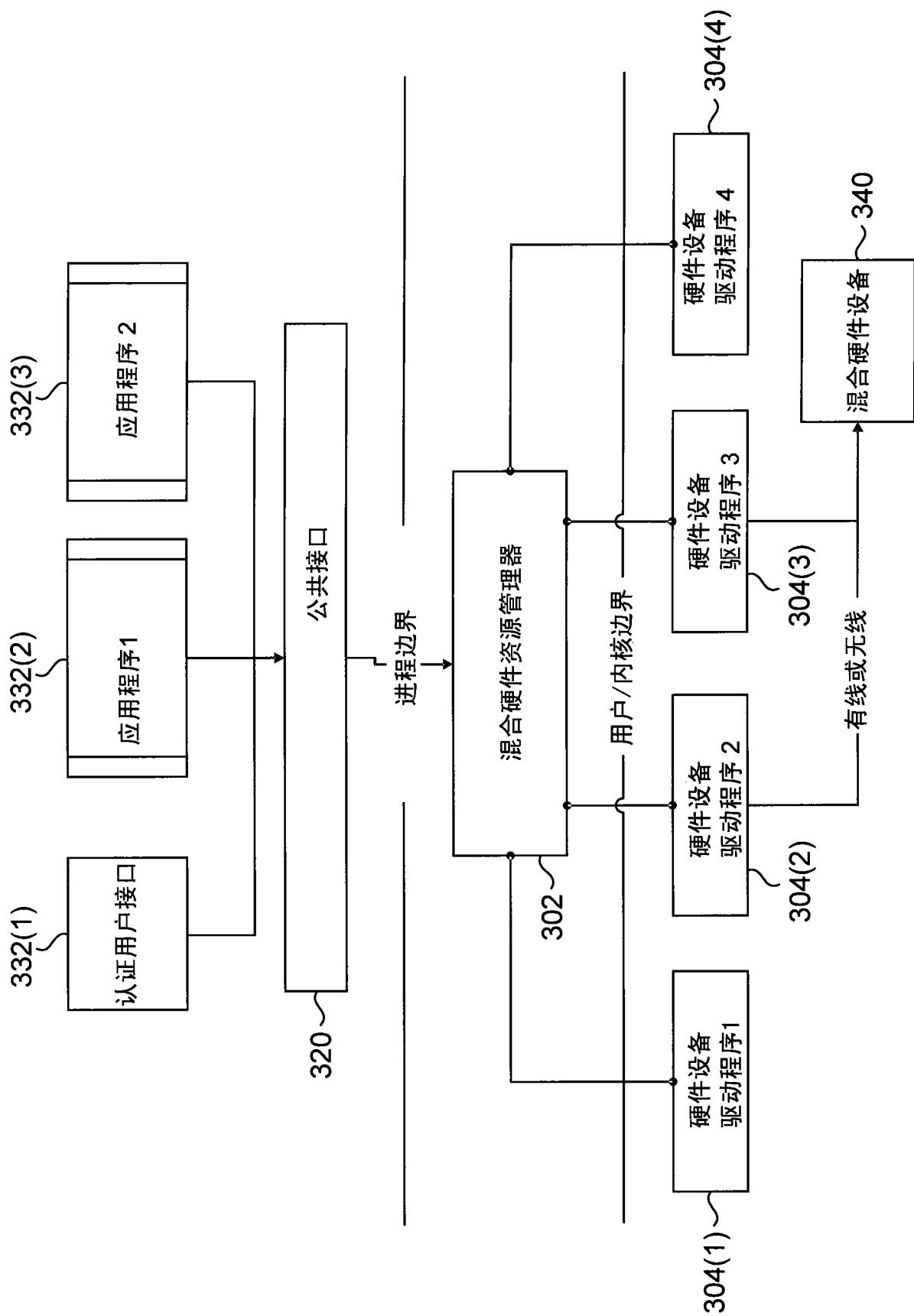


图 3

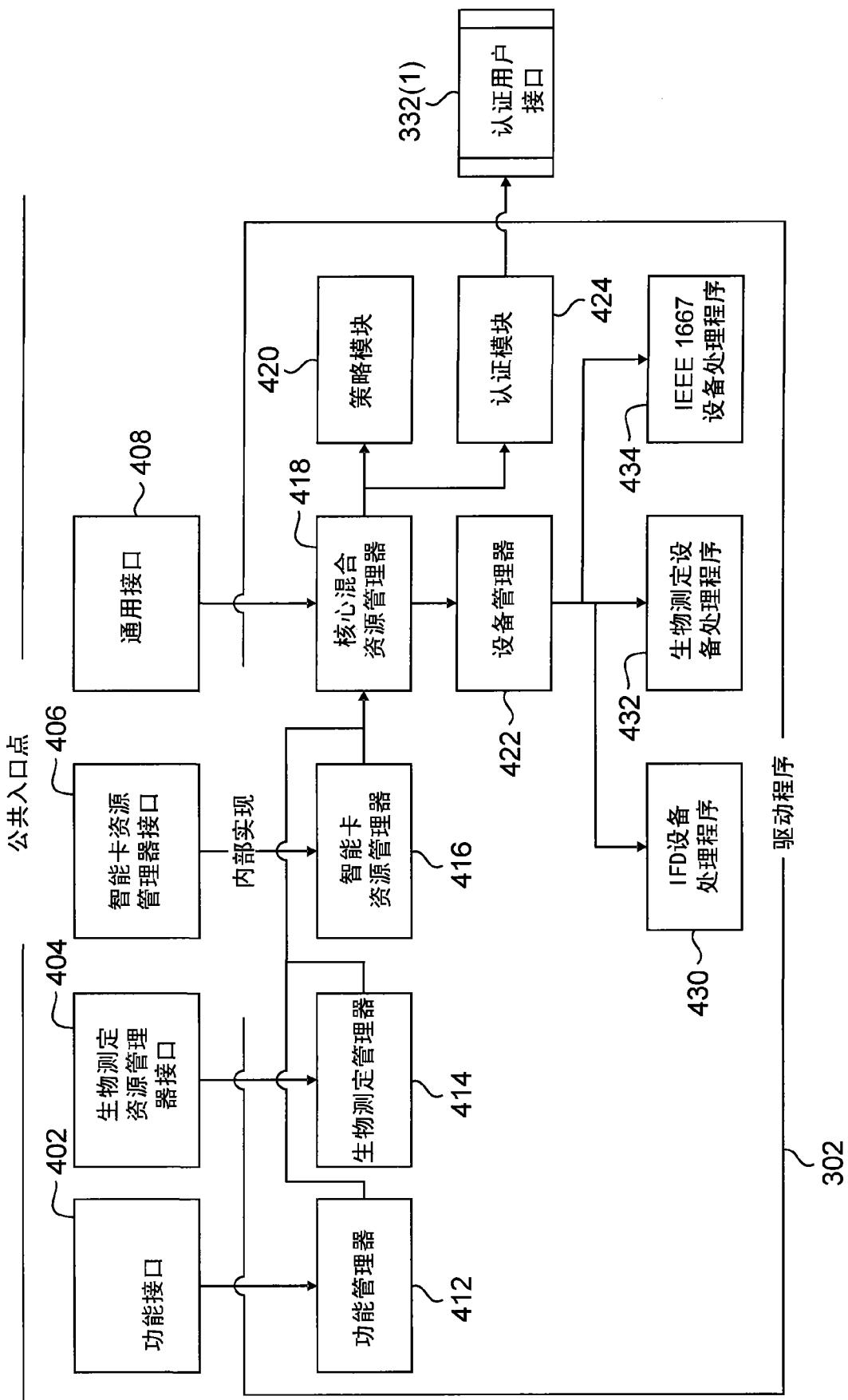


图 4

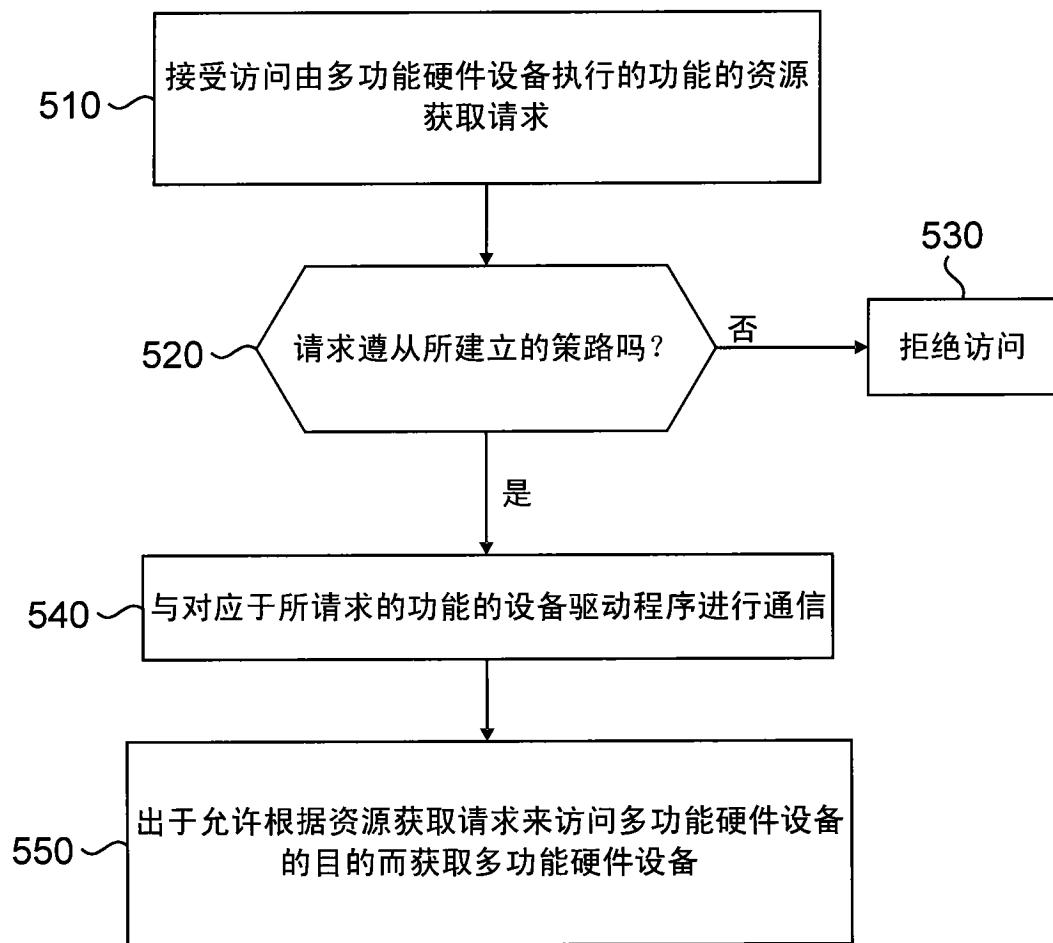


图 5