(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2023 年 1 月 5 日 (05.01.2023)

WIPO | PCT

(10) 国际公布号
WO 2023/274376 A1

(51) 国际专利分类号:
G06F 9/4401 (2018.01)

(21) 国际申请号: PCT/CN2022/102981

(22) 国际申请日: 2022 年 6 月 30 日 (30.06.2022)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
202110734252.7 2021年6月30日 (30.06.2021) CN

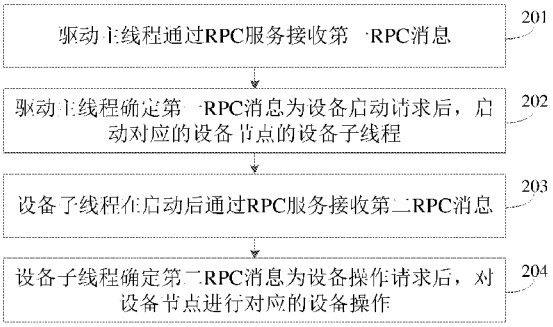
(71) 申请人: 阿里巴巴（中国）有限公司 (ALIBABA (CHINA) CO., LTD) [CN/CN]; 中国浙江省杭州市滨江区长河街道网商路 699 号 4 号楼 5 楼508室, Zhejiang 310052 (CN)。

(72) 发明人: 梁超众 (LIANG, Chaozhong); 中国浙江省杭州市余杭区文一西路969号3号楼, Zhejiang 311121 (CN)。 童武胜 (TONG, Wusheng); 中国浙江省杭州市余杭区文一西路969号3号楼, Zhejiang 311121 (CN)。 刘勇锋 (LIU, Yongfeng); 中国浙江省杭州市余杭区文一西路969号3号楼, Zhejiang 311121 (CN)。 毛熠璐 (MAO, Yilu); 中国浙江省杭州市余杭区文一西路969号3号楼, Zhejiang 311121 (CN)。

(74) 代理人: 北京三友知识产权代理有限公司 (BEIJING SANYOU INTELLECTUAL PROPERTY AGENCY LTD.); 中国北京市金融街 35 号国际企业大厦 A 座 16 层, Beijing 100033 (CN)。

(54) **Title:** DEVICE DRIVE METHOD AND APPARATUS FOR MICRO-KERNEL ARCHITECTURE, AND ELECTRONIC DEVICE AND STORAGE MEDIUM

(54) **发明名称**: 微内核架构的设备驱动方法、装置、电子设备和存储介质



201 A drive master thread receives a first RPC message by means of an RPC service
202 After determining that the first RPC message is a device start request, the drive master thread starts a device sub-thread of a corresponding device node
203 After being started, the device sub-thread receives a second RPC message by means of the RPC service
204 After determining that the second RPC message is a device operation request, the device sub-thread performs a corresponding device operation on the device node

图 2

(57) **Abstract:** The embodiments of the present disclosure relate to a device drive method and apparatus for a micro-kernel architecture, and an electronic device and a storage medium. In at least one embodiment of the present disclosure, a master-slave thread architecture mode including a drive master thread and a device sub-thread is used; when an operation system is started, only the drive master thread is started, and the device sub-thread is not started, such that system resources consumed by the device sub-thread are reduced; in addition, the drive master thread and the device sub-thread provide a device drive service by means of RPC; and specifically, when a first RPC message received by the drive master thread is a device start request, the drive master thread starts the device sub-thread of a device node, and then after determining that a received second RPC message is a device operation request, the device sub-thread performs a corresponding operation on the device node, thereby solving the problem of an application being unable to directly call a device driver in the current micro-kernel architecture.

(57) **摘要**: 本公开实施例涉及微内核架构的设备驱动方法、装置、电子设备和存储介质。本公开的至少一个实施例中，采用驱动主线程和设备子线程的主从线程架构模式，在操作系统启动时，仅启动驱动主线程，而不启动设备子线程，减少设备子线程对系统资源的消耗；并且，驱动主线程和设备子线程通过RPC方式提供设备驱动服务，具体地，当驱动主线程接收到的第一RPC消息为设备启动请求后，驱动主线程再启动设备节点的设备子线程，进而由设备子线程确定接收到的第二RPC消息为设备操作请求后，对设备节点进行对应的操作，解决目前的微内核架构中，应用无法直接调用设备驱动的问题。

WO 2023/274376 A1

[见续页]

(81) 指定国(除另有指明，要求每一种可提供的国家保护)：AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

根据细则4.17的声明：

— 关于申请人有权要求在先申请的优先权(细则4.17(iii))

本国际公布：

— 包括国际检索报告(条约第21条(3))。

微内核架构的设备驱动方法、装置、电子设备和存储介质

本申请要求2021年06月30日递交的申请号为202110734252.7、发明名称为“微内核架构的设备驱动方法、装置、电子设备和存储介质”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本公开实施例涉及计算机技术领域，具体涉及一种微内核架构的设备驱动方法、装置、电子设备和非暂态计算机可读存储介质。

背景技术

操作系统（Operation System，OS）是管理计算机硬件与软件资源的计算机程序。系统调用是操作系统的最小功能单位，系统调用将操作系统分为内核态（可以理解为内核空间）和用户态（可以理解为用户空间）。内核是操作系统的一种特殊的软件程序，直接运行在硬件上，可以理解为连接应用程序和硬件的桥梁，用于控制计算机的硬件资源，例如协调中央处理单元（Central Processing Unit， CPU）资源、分配内存资源并提供稳定的环境供应用程序运行。用户态可以理解为提供应用程序运行的空间。为了使应用程序访问到内核管理的资源，例如CPU资源、内存资源、I/O（Input/Output，输入/输出）资源，内核需提供一组通用的访问接口，这些接口即为系统调用。

目前，操作系统的内核架构有两种：宏内核架构和微内核架构。图1示出了宏内核的架构示意图和微内核架构示意图，由图1可见，在宏内核架构中组件、应用和设备驱动运行于内核态，因此，组件和应用可以直接调用设备驱动。而在微内核架构中，设备驱动运行于用户态，例如运行于进程1中，组件和应用运行于用户态的进程2中，这样，组件和应用无法直接调用设备驱动，为此，亟需提供一种微内核架构的设备驱动方案。

发明内容

为了解决现有技术存在的至少一个问题，本公开的至少一个实施例提供了一种微内核架构的设备驱动方法、装置、电子设备和非暂态计算机可读存储介质。

第一方面，本公开实施例提出一种微内核架构的设备驱动方法，其中，操作系统响应启动指令而启动驱动主线程，且不启动设备子线程，该设备驱动方法包括：

驱动主线程通过RPC服务接收第一RPC消息；

驱动主线程确定第一RPC消息为设备启动请求后，启动对应的设备节点的设备子线程；

设备子线程在启动后通过RPC服务接收第二RPC消息；

设备子线程确定第二RPC消息为设备操作请求后，对设备节点进行对应的设备操作。

在一些实施例中，驱动主线程通过 RPC 服务接收第一 RPC 消息之前，该设备驱动方法还包括：

驱动主线程注册并启动主线程 RPC 服务；

驱动主线程在启动主线程 RPC 服务后，注册一个或多个设备节点的 RPC 服务并将注册每个设备节点。

在一些实施例中，驱动主线程确定第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程，包括：

驱动主线程确定第一 RPC 消息为设备启动请求后，打开设备启动请求对应的设备节点；

驱动主线程在打开设备节点后，启动设备节点的设备子线程。

在一些实施例中，驱动主线程确定第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程，包括：

驱动主线程确定第一 RPC 消息为设备启动请求后，调用操作系统提供的创建线程 API，以创建设备启动请求对应的设备节点的设备子线程。

在一些实施例中，该设备驱动方法还包括：

设备子线程在启动后打开对应的设备节点；

设备子线程在打开设备节点后，且确定第二 RPC 消息为设备操作请求后，对设备节点进行对应的设备操作。

在一些实施例中，注册一个或多个设备节点的 RPC 服务并将注册每个设备节点，包括：

驱动主线程扫描驱动列表，得到一个或多个设备类型以及每个设备类型对应的设备节点；

驱动主线程注册每个设备节点的 RPC 服务；

驱动主线程在注册每个设备节点的 RPC 服务后注册每个设备节点。

在一些实施例中，注册每个设备节点包括：

向第一 VFS 库注册每个设备节点，其中，第一 VFS 库为驱动主线程所在进程的 VFS 库；或，向 PM 进程注册每个设备节点。

在一些实施例中，设备启动请求由应用进程通过如下方式发送：

应用进程在第二 VFS 库中查找待启动的设备节点是否已注册，其中，第二 VFS 库为应用进程的 VFS 库；若未注册，则应用进程查找主线程 RPC 服务；

应用进程基于主线程 RPC 服务向驱动主线程发送设备启动请求。

在一些实施例中，应用进程查找主线程 RPC 服务的同时还查找待启动的设备节点的 RPC 服务；

应用进程接收到驱动主线程反馈的设备启动结果后，若设备启动结果为成功，则应

用进程基于待启动的设备节点的RPC服务向设备子线程发送设备操作请求。

在一些实施例中，驱动主线程在PM进程中注册并启动主线程RPC服务；应用进程在PM进程中查找主线程RPC服务。

在一些实施例中，该设备驱动方法还包括：

驱动主线程确定第一RPC消息为设备关闭请求后，结束对应的设备节点的设备子线程；

驱动主线程在结束对应的设备节点的设备子线程后，关闭设备节点。

第二方面，本公开实施例还提出一种微内核架构的设备驱动装置，其中，该设备驱动装置包括：驱动主线程和设备子线程；操作系统响应启动指令而启动驱动主线程，且不启动设备子线程；

驱动主线程，用于通过RPC服务接收第一RPC消息，并确定第一RPC消息为设备启动请求后，启动对应的设备节点的设备子线程；

设备子线程，用于在启动后通过RPC服务接收第二RPC消息，并确定第二RPC消息为设备操作请求后，对设备节点进行对应的设备操作。

第三方面，本公开实施例还提出一种电子设备，包括：处理器和存储器；所述处理器通过调用所述存储器存储的程序或指令，用于执行如第一方面所述微内核架构的设备驱动方法的步骤。

第四方面，本公开实施例还提出一种非暂态计算机可读存储介质，用于存储程序或指令，所述程序或指令使计算机执行如第一方面所述微内核架构的设备驱动方法的步骤。

可见，本公开的至少一个实施例中，采用驱动主线程和设备子线程的主从线程架构模式，在操作系统启动时，仅启动驱动主线程，而不启动设备子线程，减少设备子线程对系统资源的消耗；并且，驱动主线程和设备子线程通过RPC方式提供设备驱动服务，具体地，当驱动主线程接收到的第一RPC消息为设备启动请求后，驱动主线程再启动设备节点的设备子线程，进而由设备子线程确定接收到的第二RPC消息为设备操作请求后，对设备节点进行对应的操作，解决目前的微内核架构中，应用无法直接调用设备驱动的问题。

附图说明

为了更清楚地说明本公开实施例的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本公开的一些实施例，对于本领域普通技术人员来讲，还可以根据这些附图获得其他的附图。

图1是宏内核的架构示意图和微内核架构示意图；

图2是本公开实施例提供的一种微内核架构的设备驱动方法的流程图；

图3是本公开实施例提供的一种启动设备子线程的流程图；

图 4 是本公开实施例提供的另一种启动设备子线程的流程图；

图 5 是本公开实施例提供的一种微内核架构的设备驱动方法的交互图；

图 6 是本公开实施例提供的一种微内核架构的设备驱动装置的示例性框图；

图 7 是本公开实施例提供的一种电子设备的示例性框图。

具体实施方式

为了能够更清楚地理解本公开的上述目的、特征和优点，下面结合附图和实施例对本公开作进一步的详细说明。可以理解的是，所描述的实施例是本公开的一部分实施例，而不是全部的实施例。此处所描述的具体实施例仅仅用于解释本公开，而非对本公开的限定。基于所描述的本公开的实施例，本领域普通技术人员所获得的所有其他实施例，都属于本公开保护的范围。

需要说明的是，在本文中，诸如“第一”和“第二”等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来，而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。

微内核是一种能够提供必要服务的操作系统内核，其中，必要服务包括但不限于任务、线程、交互进程通信（IPC，Inter-Process Communication）以及内存管理等。所有服务（包括设备驱动）运行于用户态，而处理这些服务同处理其他的任何一个程序一样，因为每个服务只是在自己的地址空间运行。

为了解决现有技术存在的至少一个问题，本公开的至少一个实施例提供了一种微内核架构的设备驱动方法、装置、电子设备和非暂态计算机可读存储介质，采用主从线程的架构模式对设备驱动加载方式及设备驱动对外提供服务的应用程序编程接口（Application Programming Interface，API）进行了详细的设计。其中，设备驱动加载包括：设备节点的 RPC 服务注册和设备节点的注册。设备驱动对外提供的服务包括：设备启动请求（open）、设备关闭请求（close）和设备操作请求，其中，设备操作请求包括但不限于：写请求（write）、读请求（read）、输入/输出请求（ioctl）等。

本公开的至少一个实施例中，采用驱动主线程和设备子线程的主从线程架构模式，在操作系统启动时，仅启动驱动主线程，而不启动设备子线程，减少设备子线程对系统资源的消耗；并且，驱动主线程和设备子线程通过 RPC 方式提供设备驱动服务，具体地，当驱动主线程接收到的第一 RPC 消息为设备启动请求后，驱动主线程再启动设备节点的设备子线程，进而由设备子线程确定接收到的第二 RPC 消息为设备操作请求后，对设备节点进行对应的操作，解决目前的微内核架构中，应用无法直接调用设备驱动的问题。

图 2 是本公开实施例提供的一种微内核架构的设备驱动方法的流程图。采用微内核架构的操作系统在启动时，例如操作系统接收到启动指令，启动指令例如用户按压开机按键而产生的指令，操作系统响应启动指令仅启动驱动主线程，而不启动设备子线程，

减少设备子线程对操作系统软硬件资源的消耗。在一些实施例中，设备子线程可以有多个，每个设备子线程与一个设备节点对应，不论设备子线程有多少个，操作系统在启动时，均不启动这些设备子线程。

在操作系统启动驱动主线程后，在步骤 201 中，驱动主线程通过 RPC 服务接收第一 RPC 消息。

其中，RPC（Remote Procedure Call，远程过程调用）服务是微服务架构中不同进程之间进行交互的一种方式，虽然微服务架构中应用无法直接调用设备驱动，但是应用进程可以通过 RPC 方式向设备驱动所处的进程发送第一 RPC 消息，这样，驱动主线程可以通过 RPC 方式提供设备驱动服务。

在步骤 202 中，驱动主线程确定第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程。

驱动主线程通过 RPC 方式提供的设备驱动服务包括设备启动请求（open），这样，应用进程可以通过 RPC 方式向驱动主线程发送设备启动请求，在该设备启动请求中携带待启动的设备节点的信息。

驱动主线程确定接收到的第一 RPC 消息为设备启动请求后，可以基于该设备启动请求中携带的设备节点的信息，确定该设备启动请求对应的设备节点，进而启动对应的设备节点的设备子线程。

可见，设备子线程的启动是由驱动主线程来控制，当有应用进程需要使用某个设备节点时，驱动主线程才会启动该设备节点的设备子线程，这样，可以减少设备节点未被使用时，设备子线程对操作系统软硬件资源的消耗。

在步骤 203 中，设备子线程在启动后通过 RPC 服务接收第二 RPC 消息。

设备子线程在启动后，可以通过 RPC 方式提供设备驱动服务，设备驱动服务例如为设备操作请求，设备操作请求包括但不限于：写请求（write）、读请求（read）、输入/输出请求（ioctl）等。这样，应用进程可以通过 RPC 方式向设备子线程发送设备操作请求。

需要说明的是，设备子线程在启动后所接收的第二 RPC 消息均是针对该设备子线程所对应的设备节点。

在步骤 204 中，设备子线程确定第二 RPC 消息为设备操作请求后，对设备节点进行对应的设备操作。

若设备操作请求为写请求，则设备子线程对设备节点进行写操作；若设备操作请求为读请求，则设备子线程对设备节点进行写操作；若设备操作请求为输入/输出请求，则设备子线程对设备节点进行输入/输出操作。

可见，实施例中采用驱动主线程和设备子线程的主从线程架构模式，在操作系统启动时，仅启动驱动主线程，而不启动设备子线程，减少设备子线程对系统资源的消耗；

并且，驱动主线程和设备子线程通过 RPC 方式提供设备驱动服务，具体地，当驱动主线程接收到的第一RPC 消息为设备启动请求后，驱动主线程再启动设备节点的设备子线程，进而由设备子线程确定接收到的第二 RPC 消息为设备操作请求后，对设备节点进行对应的操作，解决目前的微内核架构中，应用无法直接调用设备驱动的问题。

在一些实施例中，驱动主线程通过 RPC 服务接收第一 RPC 消息之前，驱动主线程需要先注册并启动主线程 RPC 服务，进而才能通过主线程 RPC 服务接收第一 RPC 消息。

驱动主线程在注册并启动主线程 RPC 服务后，可以注册一个或多个设备节点的 RPC 服务并注册每个设备节点。在一些实施例中，驱动主线程可以扫描驱动列表，得到一个或多个设备类型以及每个设备类型对应的设备节点，进而注册每个设备节点的 RPC 服务并注册每个设备节点。这样，任一设备子线程在启动后可以通过该设备子线程对应的设备节点的 RPC 服务接收第二 RPC 消息。

在一些实施例中，驱动主线程确定第一 RPC 消息为设备启动请求后，有多种方式启动对应的设备节点的设备子线程。

例如，图 3 为本公开实施例提供的一种启动设备子线程的流程图。在步骤 301 中，驱动主线程接收设备启动请求。该步骤中，驱动主线程等待接收第一 RPC 消息，当驱动主线程确定接收的第一 RPC 消息为设备启动请求后，执行步骤 302。

在步骤 302 中，驱动主线程打开设备启动请求对应的设备节点。该步骤中，驱动主线程可以基于设备启动请求中携带的设备节点的信息（例如设备节点的名称），确定对应的设备节点，进而打开该设备节点。在一些实施例中，驱动主线程通过将 ops（操作方法）指针指向 ops 结构体的成员：open，也即 ops-> open，以打开该设备节点。

在步骤 303 中，驱动主线程在打开设备节点后，启动设备节点的设备子线程。由于设备子线程是在设备节点打开后才启动，因此，设备子线程在启动后可以直接通过 RPC 服务接收第二 RPC 消息。

又例如，图 4 为本公开实施例提供的另一种启动设备子线程的流程图。步骤 401 与图 3 中的步骤 301 相同，不再赘述。

在步骤 402 中，驱动主线程调用操作系统提供的创建线程 API（），以创建设备启动请求对应的设备节点的设备子线程。本实施例中，与图 3 实施例不同的是，驱动主线程在接收到设备启动请求后，不是先打开设备节点再启动设备节点的设备子线程，而是直接调用创建线程 API 来创建设备节点的设备子线程。

在步骤 403 中，设备子线程在启动后打开对应的设备节点。与图 3 实施例不同的是，设备节点并非由驱动主线程打开，而是由设备子线程打开。具体地，设备子线通过将 ops（操作方法）指针指向 ops 结构体的成员：open，也即 ops-> open，以打开该设备节点。

设备子线程在打开设备节点后，且确定第二 RPC 消息为设备操作请求后，对设备节点进行对应的设备操作。

在一些实施例中，驱动主线程注册每个设备节点时，可以向第一 VFS（Virtual File System，虚拟文件系统）库注册每个设备节点，其中，第一 VFS 库为驱动主线程所在进程的 VFS 库；或者，驱动主线程可以向 PM（Pass Manager，路径管理）进程注册每个设备节点。

在一些实施例中，应用进程在第二 VFS 库中查找待启动的设备节点是否已注册，其中，第二 VFS 库为应用进程的 VFS 库；若未注册，则应用进程查找主线程 RPC 服务。在一些实施例中，主线程 RPC 服务由驱动主线程在 PM 进程中注册并启动，因此，应用进程在 PM 进程中查找主线程 RPC 服务，PM 进程将查询结果反馈给应用进程。

应用进程查找到主线程 RPC 服务后，可以基于主线程 RPC 服务向驱动主线程发送设备启动请求。

在一些实施例中，应用进程查找主线程 RPC 服务的同时还可以查找待启动的设备节点的 RPC 服务。在一些实施例中，主线程 RPC 服务由驱动主线程在 PM 进程中注册并启动，设备节点的 RPC 服务也由驱动主线程在 PM 进程中注册，因此，应用进程在 PM 进程中查找主线程 RPC 服务以及待启动的设备节点的 RPC 服务，PM 进程将查询结果反馈给应用进程。

应用进程接收到驱动主线程反馈的设备启动结果后，若设备启动结果为成功，则应用进程基于待启动的设备节点的 RPC 服务向设备子线程发送设备操作请求。

在一些实施例中，驱动主线程通过 RPC 方式提供的设备驱动服务包括设备关闭请求（open），这样，应用进程可以通过 RPC 方式向驱动主线程发送设备关闭请求，在该设备启动请求中携带待关闭的设备节点的信息或者设备启动结果，以便驱动主线程可以确定设备关闭请求对应的设备节点。

驱动主线程确定第一 RPC 消息为设备关闭请求后，结束对应的设备节点的设备子线程。驱动主线程在结束对应的设备节点的设备子线程后，关闭设备节点。在一些实施例中，驱动主线程通过将 ops（操作方法）指针指向 ops 结构体的成员：close，也即 ops-> close，以关闭该设备节点。

结合以上各方法实施例的描述，图 5 是本公开实施例提供的一种微内核架构的设备驱动方法的交互图，需要说明的是，图 5 仅为本公开的一些实施例，而非全部实施例。

在图 5 中，为了便于描述，仅示出一个设备子线程，本领域技术人员可以理解，设备子线程可以有多个，且交互流程类似。

在图 5 中，采用微内核架构的操作系统在启动时，例如操作系统接收到启动指令，启动指令例如用户按压开机按键而产生的指令，操作系统响应启动指令仅启动驱动主线程，而不启动设备子线程，减少设备子线程对操作系统软硬件资源的消耗。

在图 5 中，驱动主线程启动后，在 PM 进程中注册并启动主线程 RPC 服务，并且在 PM 进程中注册设备节点的 RPC 服务以及注册设备节点。在完成以上注册工作后，驱动

主线程可以等待 RPC 消息。

在图 5 中，应用进程在第二 VFS 库中查找待启动的设备节点是否已注册，其中，第二 VFS 库为应用进程的 VFS 库；若未注册，则应用进程在 PM 进程中进行 RPC 服务查询，RPC 服务查询包括查找主线程 RPC 服务和待启动的设备节点的 RPC 服务。

PM 进程将查询结果反馈给应用进程，其中，查询结果中包括：主线程 RPC 服务信息和待启动的设备节点的服务信息。应用进程在获取到查询结果后，基于主线程 RPC 服务向驱动主线程发送设备启动请求，设备启动请求中携带设备节点的信息（例如设备节点的名称）。

在图 5 中，驱动主线程接收到设备启动请求后，基于设备启动请求中携带的设备节点的信息，确定对应的设备节点，进而通过将 ops（操作方法）指针指向 ops 结构体的成员：open，也即 ops-> open，以打开该设备节点。

驱动主线程在打开设备节点后，启动设备节点的设备子线程，并将设备启动结果反馈给应用进程。由于设备子线程是在设备节点打开后才启动，因此，设备子线程在启动后可以直接等待 RPC 消息。

在图 5 中，应用进程接收到驱动主线程反馈的设备启动结果后，若设备启动结果为成功，则应用进程基于待启动的设备节点的 RPC 服务向设备子线程发送设备操作请求。

设备子线程接收到设备操作请求后，对设备节点进行对应的设备操作。若设备操作请求为写请求，则设备子线程对设备节点进行写操作；若设备操作请求为读请求，则设备子线程对设备节点进行写操作；若设备操作请求为输入/输出请求，则设备子线程对设备节点进行输入/输出操作。

在图 5 中，应用进程可以向驱动主线程发送设备关闭请求，在该设备启动请求中携带待关闭的设备节点的信息或者设备启动结果，以便驱动主线程可以确定设备关闭请求对应的设备节点。

驱动主线程接收到设备关闭请求后，结束对应的设备节点的设备子线程。驱动主线程在结束对应的设备节点的设备子线程后，通过将 ops（操作方法）指针指向 ops 结构体的成员：close，也即 ops-> close，以关闭该设备节点。

可见，图 5 所示的微内核架构的设备驱动方法的交互流程，采用驱动主线程和设备子线程的主从线程架构模式，在操作系统启动时，仅启动驱动主线程，而不启动设备子线程，减少设备子线程对系统资源的消耗；并且，驱动主线程和设备子线程通过 RPC 方式提供设备驱动服务，具体地，当驱动主线程接收到设备启动请求后，驱动主线程再启动设备节点的设备子线程，进而由设备子线程基于接收到的设备操作请求，对设备节点进行对应的操作，解决目前的微内核架构中，应用无法直接调用设备驱动的问题。

图 6 是本公开实施例提供的一种微内核架构的设备驱动装置，该设备驱动装置包括：驱动主线程 61 和设备子线程 62。采用微内核架构的操作系统响应启动指令而启动驱动

主线程 61，且不启动设备子线程 62。该设备驱动装置可以包括多个设备子线程 62，每个设备子线程 62 与一个设备节点对应。

驱动主线程 61，用于通过 RPC 服务接收第一 RPC 消息，并确定第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程 62。

设备子线程 62，用于在启动后通过 RPC 服务接收第二 RPC 消息，并确定第二 RPC 消息为设备操作请求后，对设备节点进行对应的设备操作。

在一些实施例中，驱动主线程 61，还用于通过 RPC 服务接收第一 RPC 消息之前，注册并启动主线程 RPC 服务，并在启动主线程 RPC 服务后，注册一个或多个设备节点的 RPC 服务并将注册每个设备节点。

在一些实施例中，驱动主线程 61 确定第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程 62，具体包括：确定第一 RPC 消息为设备启动请求后，打开设备启动请求对应的设备节点；在打开设备节点后，启动设备节点的设备子线程 62。

在一些实施例中，驱动主线程 61 确定第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程 62，具体包括：确定第一 RPC 消息为设备启动请求后，调用操作系统提供的创建线程 API，以创建设备启动请求对应的设备节点的设备子线程 62。

在一些实施例中，设备子线程 62 在启动后打开对应的设备节点，在打开设备节点后，且确定第二 RPC 消息为设备操作请求后，对设备节点进行对应的设备操作。

在一些实施例中，驱动主线程 61 注册一个或多个设备节点的 RPC 服务并将注册每个设备节点，具体包括：扫描驱动列表，得到一个或多个设备类型以及每个设备类型对应的设备节点；注册每个设备节点的 RPC 服务；并在注册每个设备节点的 RPC 服务后注册每个设备节点。

在一些实施例中，驱动主线程 61 注册每个设备节点，具体包括：向第一 VFS 库注册每个设备节点，其中，第一 VFS 库为驱动主线程所在进程的 VFS 库；或，向 PM 进程注册每个设备节点。

在一些实施例中，设备启动请求由应用进程通过如下方式发送：

应用进程在第二 VFS 库中查找待启动的设备节点是否已注册，其中，第二 VFS 库为应用进程的 VFS 库；若未注册，则应用进程查找主线程 RPC 服务；应用进程基于主线程 RPC 服务向驱动主线程发送设备启动请求。

在一些实施例中，应用进程查找主线程 RPC 服务的同时还查找待启动的设备节点的 RPC 服务。应用进程接收到驱动主线程反馈的设备启动结果后，若设备启动结果为成功，则应用进程基于待启动的设备节点的 RPC 服务向设备子线程发送设备操作请求。

在一些实施例中，驱动主线程 61 在 PM 进程中注册并启动主线程 RPC 服务。应用进程在 PM 进程中查找主线程 RPC 服务。

在一些实施例中，驱动主线程 61 还用于确定第一 RPC 消息为设备关闭请求后，结

束对应的设备节点的设备子线程 62；驱动主线程 61 在结束对应的设备节点的设备子线程 62 后，关闭设备节点。

以上各装置实施例的细节可参考各方法实例的细节，为避免重复，不再赘述。

图 7 是本公开实施例提供的一种电子设备的结构示意图。如图 7 所示，电子设备包括：至少一个处理器 71、至少一个存储器 72 和至少一个通信接口 73。电子设备中的各个组件通过总线系统 74 耦合在一起。通信接口 73，用于与外部设备之间的信息传输。可理解地，总线系统 74 用于实现这些组件之间的连接通信。总线系统 74 除包括数据总线之外，还包括电源总线、控制总线和状态信号总线。但为了清楚说明起见，在图 7 中将各种总线都标为总线系统 74。

可以理解，本实施例中的存储器 72 可以是易失性存储器或非易失性存储器，或可包括易失性和非易失性存储器两者。

在一些实施方式中，存储器 72 存储了如下的元素，可执行单元或者数据结构，或者他们的子集，或者他们的扩展集：操作系统和应用程序。

其中，操作系统，包含各种系统程序，例如框架层、核心库层、驱动层等，用于实现各种基础任务以及处理基于硬件的任务。应用程序，包含各种应用程序，例如媒体播放器（Media Player）、浏览器（Browser）等，用于实现各种应用任务。实现本公开实施例提供的微内核架构的设备驱动方法的程序可以包含在应用程序或操作系统中。

在本公开实施例中，处理器 71 通过调用存储器 72 存储的程序或指令，具体的，可以是应用程序中存储的程序或指令，处理器 71 用于执行本公开实施例提供的微内核架构的设备驱动方法各实施例的步骤。

本公开实施例提供的微内核架构的设备驱动方法可以应用于处理器 71 中，或者由处理器 71 实现。处理器 71 可以是一种集成电路芯片，具有信号的处理能力。在实现过程中，上述方法的各步骤可以通过处理器 71 中的硬件的集成逻辑电路或者软件形式的指令完成。上述的处理器 71 可以是通用处理器、数字信号处理器（Digital Signal Processor，DSP）、专用集成电路（Application Specific Integrated Circuit，ASIC）、现成可编程门阵列（Field Programmable Gate Array，FPGA）或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。

本公开实施例提供的微内核架构的设备驱动方法的步骤可以直接体现为硬件译码处理器执行完成，或者用译码处理器中的硬件及软件单元组合执行完成。软件单元可以位于随机存储器，闪存、只读存储器，可编程只读存储器或者电可擦写可编程存储器、寄存器等本领域成熟的存储介质中。该存储介质位于存储器 72，处理器 71 读取存储器 72 中的信息，结合其硬件完成方法的步骤。

需要说明的是，对于前述的各方法实施例，为了简单描述，故将其都表述为一系列

的动作组合，但是本领域技术人员能够理解，本公开实施例并不受所描述的动作顺序的限制，因为依据本公开实施例，某些步骤可以采用其他顺序或者同时进行。另外，本领域技术人员能够理解，说明书中所描述的实施例均属于可选实施例。

本公开实施例还提出一种非暂态计算机可读存储介质，所述非暂态计算机可读存储介质存储程序或指令，所述程序或指令使计算机执行如微内核架构的设备驱动方法各实施例的步骤，为避免重复描述，在此不再赘述。

需要说明的是，在本文中，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、物品或者装置不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、物品或者装置所固有的要素。在没有更多限制的情况下，由语句“包括……”限定的要素，并不排除在包括该要素的过程、方法、物品或者装置中还存在另外的相同要素。

本领域的技术人员能够理解，尽管在此所述的一些实施例包括其它实施例中所包括的某些特征而不是其它特征，但是不同实施例的特征的组合意味着处于本公开的范围之内并且形成不同的实施例。

本领域的技术人员能够理解，对各个实施例的描述都各有侧重，某个实施例中没有详述的部分，可以参见其他实施例的相关描述。

虽然结合附图描述了本公开的实施方式，但是本领域技术人员可以在不脱离本公开的精神和范围的情况下做出各种修改和变型，这样的修改和变型均落入由所附权利要求所限定的范围之内。

权 利 要 求 书

1. 一种微内核架构的设备驱动方法，其中，操作系统响应启动指令而启动驱动主线程，且不启动设备子线程，所述方法包括：

所述驱动主线程通过 RPC 服务接收第一 RPC 消息；

所述驱动主线程确定所述第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程；

所述设备子线程在启动后通过 RPC 服务接收第二 RPC 消息；

所述设备子线程确定所述第二 RPC 消息为设备操作请求后，对所述设备节点进行对应的设备操作。

2. 根据权利要求 1 所述的方法，其中，所述驱动主线程通过 RPC 服务接收第一 RPC 消息之前，所述方法还包括：

所述驱动主线程注册并启动主线程 RPC 服务；

所述驱动主线程在启动所述主线程 RPC 服务后，注册一个或多个设备节点的 RPC 服务并将注册每个所述设备节点。

3. 根据权利要求 1 所述的方法，其中，所述驱动主线程确定所述第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程，包括：

所述驱动主线程确定所述第一 RPC 消息为设备启动请求后，打开所述设备启动请求对应的设备节点；

所述驱动主线程在打开所述设备节点后，启动所述设备节点的设备子线程。

4. 根据权利要求 1 所述的方法，其中，所述驱动主线程确定所述第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程，包括：

所述驱动主线程确定所述第一 RPC 消息为设备启动请求后，调用所述操作系统提供的创建线程 API，以创建所述设备启动请求对应的设备节点的设备子线程。

5. 根据权利要求 4 所述的方法，其中，所述方法还包括：

所述设备子线程在启动后打开对应的设备节点；

所述设备子线程在打开所述设备节点后，且确定所述第二 RPC 消息为设备操作请求后，对所述设备节点进行对应的设备操作。

6. 根据权利要求 2 所述的方法，其中，所述注册一个或多个设备节点的 RPC 服务并将注册每个所述设备节点，包括：

所述驱动主线程扫描驱动列表，得到一个或多个设备类型以及每个所述设备类型对应的设备节点；

所述驱动主线程注册每个所述设备节点的 RPC 服务；

所述驱动主线程在注册每个所述设备节点的 RPC 服务后注册每个所述设备节点。

7. 根据权利要求 2 所述的方法，其中，所述设备启动请求由应用进程通过如下方式发送：

所述应用进程在第二 VFS 库中查找待启动的设备节点是否已注册，其中，所述第二 VFS 库为所述应用进程的 VFS 库；若未注册，则所述应用进程查找主线程 RPC 服务；

所述应用进程基于所述主线程 RPC 服务向所述驱动主线程发送所述设备启动请求。

8. 根据权利要求 7 所述的方法，其中，所述应用进程查找主线程 RPC 服务的同时还查找所述待启动的设备节点的 RPC 服务；

所述应用进程接收到所述驱动主线程反馈的设备启动结果后，若所述设备启动结果为成功，则所述应用进程基于所述待启动的设备节点的 RPC 服务向所述设备子线程发送所述设备操作请求。

9. 根据权利要求 1 所述的方法，其中，所述方法还包括：

所述驱动主线程确定所述第一 RPC 消息为设备关闭请求后，结束对应的设备节点的设备子线程；

所述驱动主线程在结束对应的设备节点的设备子线程后，关闭所述设备节点。

10. 一种微内核架构的设备驱动装置，其中，所述装置包括：驱动主线程和设备子线程；操作系统响应启动指令而启动所述驱动主线程，且不启动所述设备子线程；

所述驱动主线程，用于通过 RPC 服务接收第一 RPC 消息，并确定所述第一 RPC 消息为设备启动请求后，启动对应的设备节点的设备子线程；

所述设备子线程，用于在启动后通过 RPC 服务接收第二 RPC 消息，并确定所述第二 RPC 消息为设备操作请求后，对所述设备节点进行对应的设备操作。

11. 一种电子设备，包括：处理器和存储器；

所述处理器通过调用所述存储器存储的程序或指令，用于执行如权利要求 1 至 9 任一项所述方法的步骤。

12. 一种非暂态计算机可读存储介质，所述非暂态计算机可读存储介质存储程序或指令，所述程序或指令使计算机执行如权利要求 1 至 9 任一项所述方法的步骤。

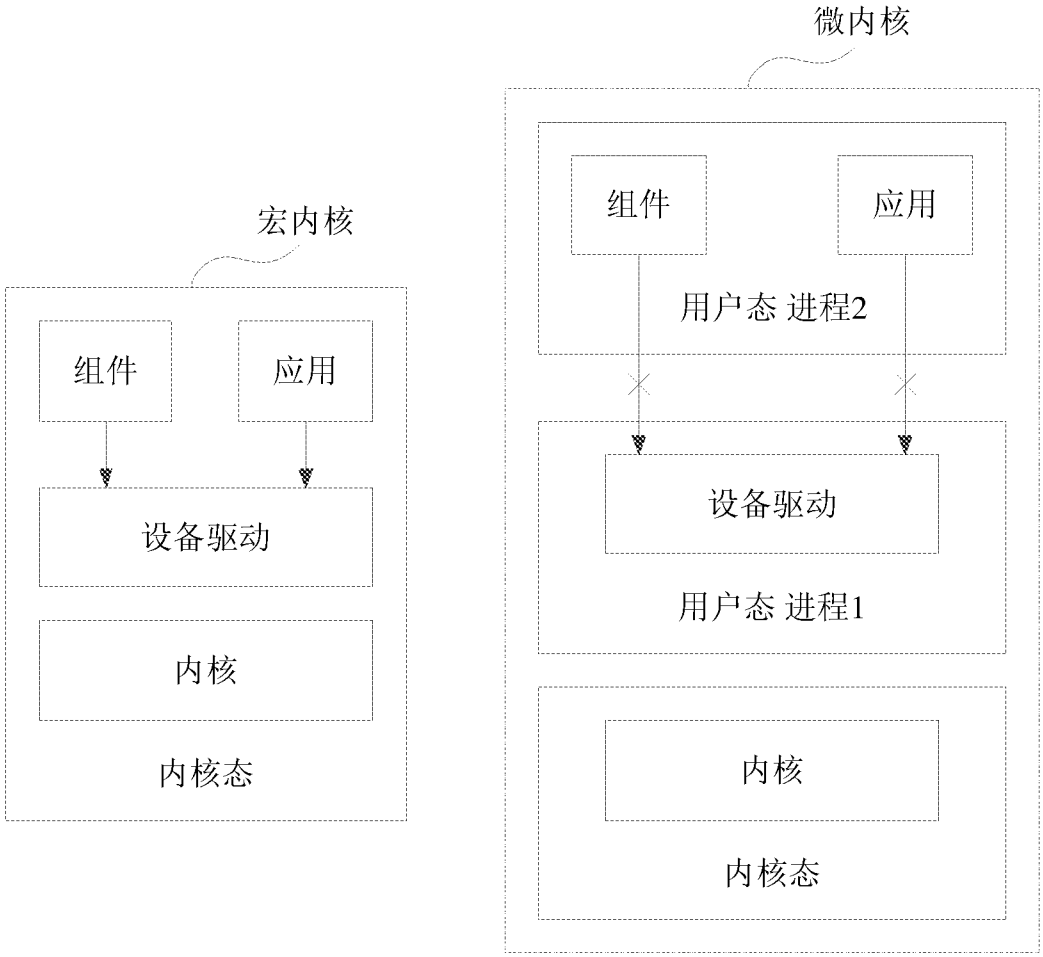

微内核
组件
应用
用户态 进程2
设备驱动
用户态 进程1
内核
内核态
宏内核
组件
应用
设备驱动
内核
内核态

图 1

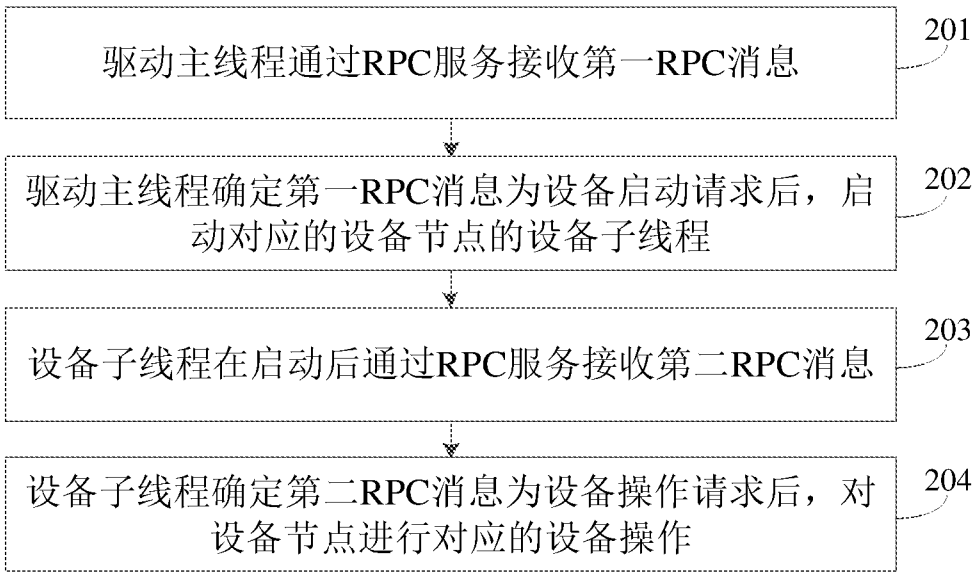

驱动主线程通过RPC服务接收第一RPC消息
201
驱动主线程确定第一RPC消息为设备启动请求后，启动对应的设备节点的设备子线程
202
设备子线程在启动后通过RPC服务接收第二RPC消息
203
设备子线程确定第二RPC消息为设备操作请求后，对设备节点进行对应的设备操作
204

图 2

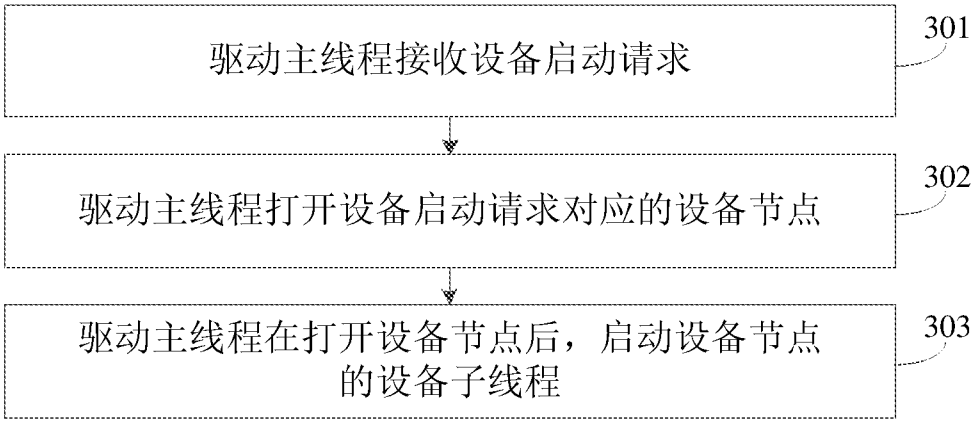

驱动主线程接收设备启动请求
301
驱动主线程打开设备启动请求对应的设备节点
302
驱动主线程在打开设备节点后，启动设备节点的设备子线程
303

图 3

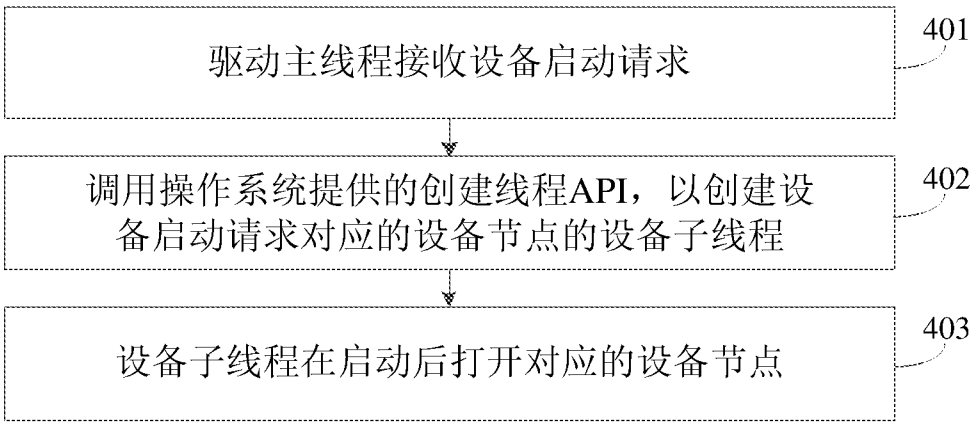

驱动主线程接收设备启动请求
401
调用操作系统提供的创建线程API，以创建设备启动请求对应的设备节点的设备子线程
402
设备子线程在启动后打开对应的设备节点
403

图 4

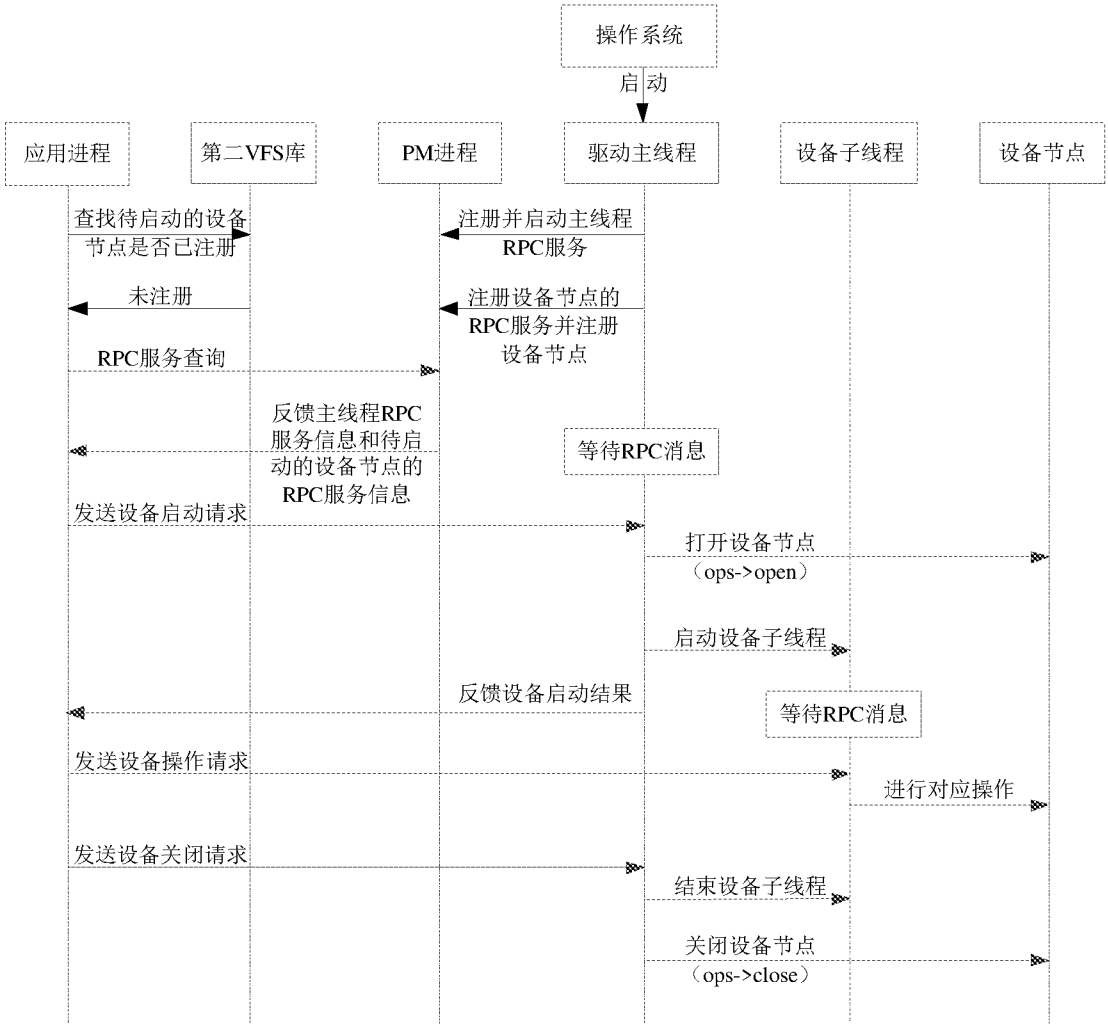

操作系统
启动
应用进程
第二VFS库
PM进程
驱动主线程
设备子线程
设备节点
查找待启动的设备节点是否已注册
注册并启动主线程
RPC服务
未注册
注册设备节点的RPC服务并注册设备节点
RPC服务查询
反馈主线程RPC服务信息和待启动的设备节点的RPC服务信息
等待RPC消息
发送设备启动请求
打开设备节点
（ops->open）
启动设备子线程
反馈设备启动结果
等待RPC消息
发送设备操作请求
进行对应操作
发送设备关闭请求
结束设备子线程
关闭设备节点
（ops->close）

图 5

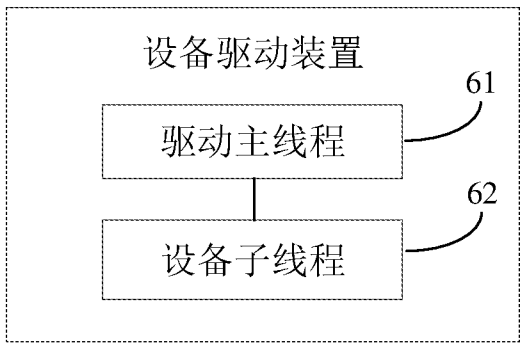

设备驱动装置
61
驱动主线程
62
设备子线程

图 6

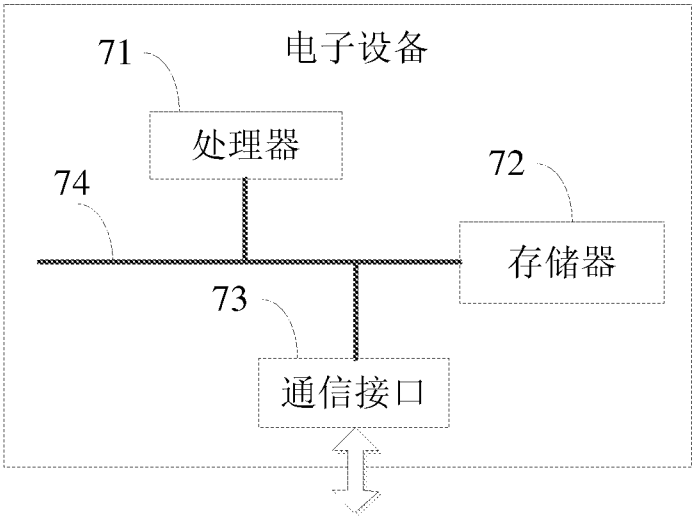


图 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2022/102981

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/4401(2018.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS; CNTXT; DWPI; VEN; WOTXT; EPTXT; USTXT; CNKI: 微内核, 设备, 驱动, 进程, 线程, 启动, 远程过程调用, 消息, microkernel, device, driver, process, thread, start, rpc, Remote Procedure Call, message

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 107368342 A (BEIJING YUANXIN SCIENCE & TECHNOLOGY CO., LTD.) 21 November 2017 (2017-11-21) description, paragraphs 57-104	1-12
A	CN 101980170 A (BEIJING RUIJIE NETWORKS CO., LTD.) 23 February 2011 (2011-02-23) description, paragraphs 30-106	1-12
A	CN 110764825 A (HUAWEI TECHNOLOGIES CO., LTD.) 07 February 2020 (2020-02-07) entire document	1-12
A	CN 109117253 A (ALIBABA GROUP HOLDING LTD.) 01 January 2019 (2019-01-01) entire document	1-12
A	US 5675795 A (IBM) 07 October 1997 (1997-10-07) entire document	1-12

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
05 September 2022	**16 September 2022**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao, Haidian District, Beijing 100088, China** Facsimile No. **(86-10)62019451**	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2022/102981

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	107368342	A	21 November 2017	CN	107368342	B	14 August 2020
CN	101980170	A	23 February 2011	CN	101980170	B	05 September 2012
CN	110764825	A	07 February 2020	CN	110764825	B	09 February 2021
CN	109117253	A	01 January 2019	WO	2019001276	A1	03 January 2019
				US	2020133723	A1	30 April 2020
				CN	109117253	B	24 May 2022
US	5675795	A	07 October 1997	CA	2122162	A1	27 October 1994
				EP	0622731	A2	02 November 1994
				JP	H06324849	A	25 November 1994
				JP	3212007	B2	25 September 2001

国际检索报告

国际申请号
PCT/CN2022/102981

A. 主题的分类

G06F 9/4401(2018.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

CNABS;CNTXT;DWPI;VEN;WOTXT;EPTXT;USTXT;CNKI:微内核，设备，驱动，进程，线程，启动，远程过程调用，消息，microkernel, device, driver, process, thread, start, rpc, Remote Procedure Call, message

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
A	CN 107368342 A (北京元心科技有限公司) 2017年11月21日 (2017 - 11 - 21) 说明书第57-104段	1-12
A	CN 101980170 A (北京星网锐捷网络技术有限公司) 2011年2月23日 (2011 - 02 - 23) 说明书第30-106段	1-12
A	CN 110764825 A (华为技术有限公司) 2020年2月7日 (2020 - 02 - 07) 全文	1-12
A	CN 109117253 A (阿里巴巴集团控股有限公司) 2019年1月1日 (2019 - 01 - 01) 全文	1-12
A	US 5675795 A (IBM) 1997年10月7日 (1997 - 10 - 07) 全文	1-12

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2022年9月5日	2022年9月16日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 张楚湖 电话号码 86-(20)-28958956

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2022/102981

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	107368342	A	2017年11月21日	CN	107368342	B	2020年8月14日
CN	101980170	A	2011年2月23日	CN	101980170	B	2012年9月5日
CN	110764825	A	2020年2月7日	CN	110764825	B	2021年2月9日
CN	109117253	A	2019年1月1日	WO	2019001276	A1	2019年1月3日
				US	2020133723	A1	2020年4月30日
				CN	109117253	B	2022年5月24日
US	5675795	A	1997年10月7日	CA	2122162	A1	1994年10月27日
				EP	0622731	A2	1994年11月2日
				JP	H06324849	A	1994年11月25日
				JP	3212007	B2	2001年9月25日