



(12) 发明专利申请

(10) 申请公布号 CN 103677989 A

(43) 申请公布日 2014. 03. 26

(21) 申请号 201310677376. 1

(22) 申请日 2013. 12. 13

(71) 申请人 TCL 集团股份有限公司

地址 516001 广东省惠州市鹅岭南路 6 号  
TCL 工业大厦 8 楼技术中心

(72) 发明人 刘伟

(74) 专利代理机构 深圳市君胜知识产权代理事  
务所 44268

代理人 王永文 刘文求

(51) Int. Cl.

G06F 9/48 (2006. 01)

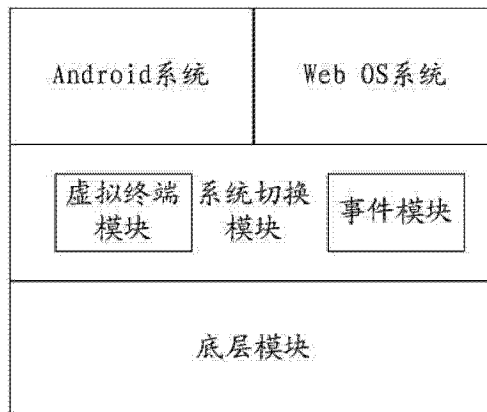
权利要求书2页 说明书5页 附图3页

(54) 发明名称

一种利用 linux 内核实现多系统的装置及方法

(57) 摘要

本发明公开一种利用 linux 内核实现多系统的装置及方法,其中,装置包括:用于不同系统共用的底层模块;设置在底层模块之上的用于在不同系统之间进行切换的系统切换模块;及设置在系统切换模块之上的不同系统;所述系统切换模块包括虚拟终端模块及事件模块;所述虚拟终端模块包括用于对不同系统进行显示的多个虚拟终端,且系统与虚拟终端一一对应;所述事件模块,用于根据用户的切换指令,将底层模块的设备文件提供给不同系统进行访问。



1. 一种利用 linux 内核实现多系统的装置,其特征在于,包括:  
用于不同系统共用的底层模块;  
设置在底层模块之上的用于在不同系统之间进行切换的系统切换模块;  
及设置在系统切换模块之上的不同系统;  
所述系统切换模块包括虚拟终端模块及事件模块;  
所述虚拟终端模块包括用于对不同系统进行显示的多个虚拟终端,且系统与虚拟终端一一对应;  
所述事件模块,用于根据用户的切换指令,将底层模块的设备文件提供给不同系统进行访问。
2. 根据权利要求1所述的利用 linux 内核实现多系统的装置,其特征在于,所述不同系统至少包括第一系统和第二系统,所述第一系统为 Android 系统。
3. 根据权利要求2所述的利用 linux 内核实现多系统的装置,其特征在于,所述事件模块包括:  
删除单元,用于当根据用户切换指令从 Android 系统切换至第二系统时,删除 linux 内核的设备文件;  
设备文件节点创建单元,用于创建新的设备文件节点,用来存储设备文件;  
读取单元,用于从新的设备文件节点读取设备文件,捕获用户的操作,并进行响应。
4. 根据权利要求2所述的利用 linux 内核实现多系统的装置,其特征在于,所述事件模块还包括:  
恢复单元,用于当根据用户切换指令从第二系统切回至 Android 系统时,恢复 linux 内核下已删除的设备文件。
5. 根据权利要求2所述的利用 linux 内核实现多系统的装置,其特征在于,所述切换指令为用户通过按下指定热键实现。
6. 根据权利要求2所述的利用 linux 内核实现多系统的装置,其特征在于,所述指定热键为组合键。
7. 一种利用 linux 内核实现多系统的方法,其特征在于,包括步骤:  
在底层模块之上设置用于在不同系统之间进行切换的系统切换模块;  
在系统切换模块之上设置不同系统的应用层;  
在系统切换模块中设置用于对不同系统进行显示的多个虚拟终端,且系统与虚拟终端一一对应;  
当接收到用户的切换指令时,将底层模块的设备文件提供给不同系统进行访问;  
并通过与系统一一对应的虚拟终端对切换后的系统进行显示。
8. 根据权利要求7所述的利用 linux 内核实现多系统的方法,其特征在于,所述将底层模块的设备文件提供给不同系统进行访问的步骤具体包括步骤:  
查询并记录当前系统使用的虚拟终端,并获取一个空闲的虚拟终端,通过所述空闲虚拟终端进行显示。
9. 根据权利要求8所述的利用 linux 内核实现多系统的方法,其特征在于,所述不同系统至少包括第一系统和第二系统,所述第一系统为 Android 系统,所述将底层模块的设备文件提供给不同系统进行访问的步骤具体包括:

当根据用户切换指令从 Android 系统切换至第二系统时,删除 linux 内核的设备文件;

创建新的设备文件节点,用来存储设备文件;

从新的设备文件节点读取设备文件,捕获用户的操作,并进行响应。

10. 根据权利要求 8 所述的利用 linux 内核实现多系统的方法,其特征在于,所述将底层模块的设备文件提供给不同系统进行访问的步骤还包括:

当根据用户切换指令从第二系统切回至 Android 系统时,恢复 linux 内核下已删除的设备文件。

## 一种利用 linux 内核实现多系统的装置及方法

### 技术领域

[0001] 本发明涉及移动终端多系统领域,尤其涉及一种利用 linux 内核实现多系统的装置及方法。

### 背景技术

[0002] 随着科技的发展和社会的进步,越来越多的智能设备走进了千家万户,智能手机、智能平板、智能电视等智能产品极大地方便了人们的工作和生活。目前,这些设备上一般只有一个操作系统,例如 Android、ios、firefox os 等。如可以在一台设备上同时安装两个或多个操作系统,那么将极大的提高用户的体验,使得用户在一台设备上就可以体验多种系统,满足了用户体验不同系统的需求。

[0003] 目前主流的 Android 设备硬件强大,CPU 基本上是双核以及四核,这也为打造多操作系统提供了坚实的硬件基础,现有技术也实现了在一台设备上安装两种操作系统的方案,但现有的双操作系统,如需要切换系统,需要重新启动系统,才能实现系统切换,切换起来不方便;并且,现有的双操作系统容易造成屏幕同时显示两个系统的问题,易导致屏幕混乱。

[0004] 因此,现有技术还有待于改进和发展。

### 发明内容

[0005] 鉴于上述现有技术的不足,本发明的目的在于提供一种利用 linux 内核实现多系统的装置及方法,旨在解决现有多系统实现技术需要重启才能切换系统导致系统切换不方便以及容易导致屏幕混乱的问题。

[0006] 本发明的技术方案如下:

一种利用 linux 内核实现多系统的装置,其中,包括:

用于不同系统共用的底层模块;

设置在底层模块之上的用于在不同系统之间进行切换的系统切换模块;

及设置在系统切换模块之上的不同系统;

所述系统切换模块包括虚拟终端模块及事件模块;

所述虚拟终端模块包括用于对不同系统进行显示的多个虚拟终端,且系统与虚拟终端一一对应;

所述事件模块,用于根据用户的切换指令,将底层模块的设备文件提供给不同系统进行访问。

[0007] 所述的利用 linux 内核实现多系统的装置,其中,所述不同系统至少包括第一系统和第二系统,所述第一系统为 Android 系统。

[0008] 所述的利用 linux 内核实现多系统的装置,其中,所述事件模块包括:

删除单元,用于当根据用户切换指令从 Android 系统切换至第二系统时,删除 linux 内核的设备文件;

设备文件节点创建单元,用于创建新的设备文件节点,用来存储设备文件;

读取单元,用于从新的设备文件节点读取设备文件,捕获用户的操作,并进行响应。

[0009] 所述的利用 linux 内核实现多系统的装置,其中,所述事件模块还包括:

恢复单元,用于当根据用户切换指令从第二系统切回至 Android 系统时,恢复 linux 内核下已删除的设备文件。

[0010] 所述的利用 linux 内核实现多系统的装置,其中,所述切换指令为用户通过按下指定热键实现。

[0011] 所述的利用 linux 内核实现多系统的装置,其中,所述指定热键为组合键。

[0012] 一种利用 linux 内核实现多系统的方法,其中,包括步骤:

在底层模块之上设置用于在不同系统之间进行切换的系统切换模块;

在系统切换模块之上设置不同系统的应用层;

在系统切换模块中设置用于对不同系统进行显示的多个虚拟终端,且系统与虚拟终端一一对应;

当接收到用户的切换指令时,将底层模块的设备文件提供给不同系统进行访问;

并通过与系统一一对应的虚拟终端对切换后的系统进行显示。

[0013] 所述的利用 linux 内核实现多系统的方法,其中,所述将底层模块的设备文件提供给不同系统进行访问的步骤具体包括步骤:

查询并记录当前系统使用的虚拟终端,并获取一个空闲的虚拟终端,通过所述空闲虚拟终端进行显示。

[0014] 所述的利用 linux 内核实现多系统的方法,其中,所述不同系统至少包括第一系统和第二系统,所述第一系统为 Android 系统,所述将底层模块的设备文件提供给不同系统进行访问的步骤具体包括:

当根据用户切换指令从 Android 系统切换至第二系统时,删除 linux 内核的设备文件;

创建新的设备文件节点,用来存储设备文件;

从新的设备文件节点读取设备文件,捕获用户的操作,并进行响应。

[0015] 所述的利用 linux 内核实现多系统的方法,其中,所述将底层模块的设备文件提供给不同系统进行访问的步骤还包括:

当根据用户切换指令从第二系统切回至 Android 系统时,恢复 linux 内核下已删除的设备文件。

[0016] 有益效果:本发明通过在底层模块之上设置系统切换模块,并在系统切换模块中设置虚拟终端模块及事件模块,通过该事件模块使得不同系统可以共用一内核,并且能够使当前使用的系统响应用户的操作,同时其他系统不会进行响应;通过该虚拟终端模块对不同系统进行显示,从而避免了多个系统同时在屏幕上显示导致的屏幕混乱的问题。

## 附图说明

[0017] 图 1 为本发明一种利用 linux 内核实现多系统的装置较佳实施例的结构框图。

[0018] 图 2 为本发明中事件模块的结构框图。

[0019] 图 3 为本发明一种利用 linux 内核实现多系统的方法较佳实施例的流程图。

[0020] 图 4 为本发明一种利用 linux 内核实现多系统的方法具体实施例的流程图。

### 具体实施方式

[0021] 本发明提供一种利用 linux 内核实现多系统的装置及方法,为使本发明的目的、技术方案及效果更加清楚、明确,以下对本发明进一步详细说明。应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。

[0022] 本发明提供的一种利用 linux 内核实现多系统的装置,其包括:

用于不同系统共用的底层模块;

设置在底层模块之上的用于在不同系统之间进行切换的系统切换模块;

及设置在系统切换模块之上的不同系统;

所述系统切换模块包括虚拟终端模块及事件模块;

所述虚拟终端模块包括用于对不同系统进行显示的多个虚拟终端,且系统与虚拟终端一一对应;

所述事件模块,用于根据用户的切换指令,将底层模块的设备文件提供给不同系统进行访问。

[0023] 本发明的原理是在 linux 内核的基础上建立一个新的系统(当然可以不止一个新的系统),新的系统和原有的系统共享一个 linux 内核,这就可以达到快速切换系统的目的,因为这两个系统都具有同一个核心,所以不需要重启才能切换至另一个系统,切换起来非常方便。同时,由于本发明中的事件模块可以根据用户的切换指令将底层模块的设备文件提供给不同系统的应用层访问,所以避免了两个系统同时在屏幕上显示导致屏幕混乱的问题。

[0024] 下面以一具体实施例来对本发明的装置进行具体说明。

[0025] 如图 1 所示,该装置包括:用于 web os 系统以及 Android 系统共用的底层模块,用于在 web os 系统以及 Android 系统之间进行系统切换的系统切换模块,设置在系统切换模块之上的不同系统。底层模块包括 linux 内核、文件系统及各种驱动,通过该系统切换模块将两个不同的系统隔离开来,在系统切换模块之上就是不同的系统:Android 系统和 web os 系统,这两个系统共享一个 linux 内核、驱动及文件系统,由于 linux 内核是一个开源的系统,所以可以通过修改其代码,来按照本发明的需求进行修改,来实现双系统切换。

[0026] 由于在一个内核上运行两个系统,会造成两个系统同时显示到屏幕导致系统显示混乱的问题,为了解决上述问题,本实施例中的虚拟终端模块设置有用用于对不同系统进行显示的多个虚拟终端(即图形虚拟终端),且系统与虚拟终端一一对应。这样将每一个虚拟终端当成一个虚拟出来的屏幕,不同的系统使用不同的虚拟终端,例如在一个虚拟终端上画图,并不会影响到另外的虚拟终端上进行画图等操作,所以两个系统分别运行在不同的虚拟终端下即可解决屏幕显示混乱的问题。

[0027] 假如 Android 系统使用的虚拟终端编号为 7,那么 web os 系统使用编号不为 7 的虚拟终端即可,例如编号为 1。当系统启动时,默认是切换到了虚拟终端 7 上进行显示,也即显示 Android 系统,当用户需要切换系统时,只需切换到虚拟终端 1,显示 web os 系统,当用户需要切换回 Android 系统时,再切换回虚拟终端 7 即可。

[0028] 虚拟终端的设备驱动文件路径是 /dev/tty,只需对这个文件进行操作,并通过调

用 `ioctl` 函数(用于向设备发出控制和配置命令的函数)即可实现切换虚拟终端的目的。由于 linux 内核最大支持 63 个虚拟终端,所以理论上在 linux 内核上可以同时安装 63 个系统。

[0029] 在本实施例中,事件模块其具有读取用户的输入操作功能,并能够将用户操作指令传递给上层,交由上层进行响应处理。事件模块的功能需要 linux 内核的 Input Driver (输入驱动)来支持。在 linux 内核中,任何设备(外设输入设备)包括遥控器、鼠标等都是被当成文件来访问的,在路径 `/dev/input/` 下以 `event` 开头的文件就是设备文件,只需对这些设备文件进行读取,并存放在一个队列中,并由上层的应用层进行读取,作出相应的处理,即可完成相应的功能,所以 Android 系统就是利用 `/dev/input/` 下的 `dev` 设备来读取用户的输入操作。本发明中,为了隔离不同系统,如图 2 所示,在事件模块 100 中设置了:

删除单元 110,用于当根据用户切换指令从 Android 系统切换至第二系统时,删除 linux 内核的设备文件;例如当用户将系统从 Android 系统切换至 web os 系统时,就删除 linux 内核的设备文件,即删除 `dev/input/` 下相关的设备文件;

设备文件节点创建单元 120,用于创建新的设备文件节点,用来存储设备文件;即在其他路径下创建新的设备文件节点,用来存储设备文件;

读取单元 130,用于从新的设备文件节点读取设备文件,捕获用户的操作,并进行响应。web os 系统可从新的设备文件节点来读取设备文件,这样就能捕获用户的输入操作,并传递给上层进行响应;

恢复单元 140,用于当根据用户切换指令从第二系统切回至 Android 系统时,恢复 linux 内核下已删除的设备文件;这是因为原有的设备文件被删除,所以 Android 系统无法获取用户输入操作,所以当用户将系统从 web os 系统切换回至 Android 系统时,就需要再恢复这些已删除的设备文件,从而达到控制两个系统分别读取用户输入操作的目的。

[0030] 较优选的,所述切换指令为用户通过按下指定热键实现,即用户通过按下某个指定热键即可实现系统切换,对于不同的设备其指定热键可能不同,所以需对不同的设备进行不同的定义,例如对于手机和带有遥控器的电视其定义是不同的。更优选的,所述指定热键为组合键,这样能够避免用户错按或误按影响用户使用的问题。指定热键的定义,可以在 linux 内核中修改事件驱动,来接受相应的指定按键切换到不同的系统中。

[0031] 修改的代码文件路径是 `/kernel /drivers/input/input.c`,在该文件中,增加对指定热键的接收以及处理函数。当接受到切换指定热键后,调用系统切换模块的接口去进行系统切换。

[0032] 上述实施例是以设备同时安装 Android 系统和 web os 系统为例进行说明,显然,本发明还可安装目前流行的其他系统,例如 firefox web os 等系统,或者基于现有的硬件基础,同时安装三个或以上的系统,同样能实现三系统切换操作的目的。另外,本发明中,假如设备上的某个系统出现故障,还可切换到另外一个系统上去进行修复,这也方便了用户解决故障。

[0033] 基于上述装置,本发明还提供一种利用 linux 内核实现多系统的方法,如图 3 所示,其包括步骤:

S101、在底层模块之上设置用于在不同系统之间进行切换的系统切换模块;

S102、在系统切换模块之上设置不同系统的应用层;

S103、在系统切换模块中设置用于对不同系统进行显示的多个虚拟终端，且系统与虚拟终端一一对应；

S104、当接收到用户的切换指令时，将底层模块的设备文件提供给不同系统进行访问；

S105、并通过与系统一一对应的虚拟终端对切换后的系统进行显示。

[0034] 进一步，所述步骤 S104 具体包括：

查询并记录当前系统使用的虚拟终端，并获取一个空闲的虚拟终端，通过所述空闲虚拟终端进行显示。

[0035] 进一步，所述不同系统至少包括第一系统和第二系统，所述第一系统为 Android 系统，所述步骤 S104 具体包括：

当根据用户切换指令从 Android 系统切换至第二系统时，删除 linux 内核的设备文件；

创建新的设备文件节点，用来存储设备文件；

从新的设备文件节点读取设备文件，捕获用户的操作，并进行响应。

[0036] 进一步，所述步骤 S104 还包括：

当根据用户切换指令从第二系统切回至 Android 系统时，恢复 linux 内核下已删除的设备文件。

[0037] 下面提供一利用 linux 内核实现多系统的方法具体实施例，如图 4 所示，其包括：

开始；

S201、用户开机；

S202、默认进入到 Android 系统；

S203、接收到用户的切换指令时，准备切换至第二系统；

S204、内核接收热键，启动切换进程；

S205、查询当前 Android 系统使用的虚拟终端，并记录其编号，同时查询一个空闲的虚拟终端，并切换至该空闲虚拟终端；

S206、删除 linux 内核的设备文件，同时创建新的设备文件节点，用来存储设备文件；

S207、调用第二系统的初始化进程，并将控制权交给第二系统；

S208、当用户按下切换回 Android 系统的热键时，使用之前记录的 Android 系统的虚拟终端；

S209、恢复原有的 linux 内核的设备文件，同时删除当前第二系统使用的设备文件；

结束。

[0038] 关于上述方法的具体技术细节在前面的方法中已有详述。

[0039] 综上所述，本发明通过在底层模块之上设置系统切换模块，并在系统切换模块中设置虚拟终端模块及事件模块，通过该事件模块使得不同系统可以共用一内核，并且能够使当前使用的系统响应用户的操作，同时其他系统不会进行响应；通过该虚拟终端模块对不同系统进行显示，从而避免了多个系统同时在屏幕上显示导致的屏幕混乱的问题。

[0040] 应当理解的是，本发明的应用不限于上述的举例，对本领域普通技术人员来说，可以根据上述说明加以改进或变换，所有这些改进和变换都应属于本发明所附权利要求的保护范围。



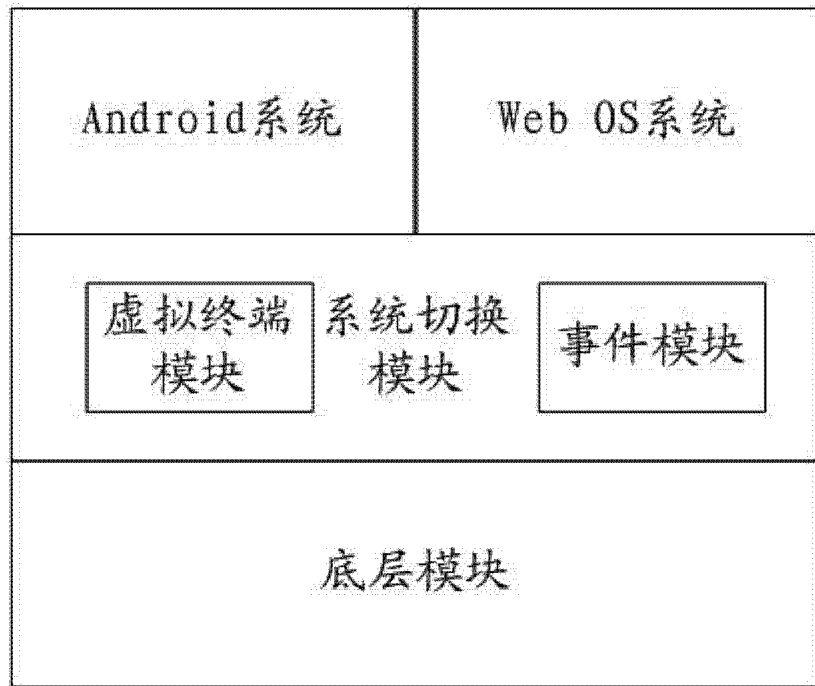


图 1

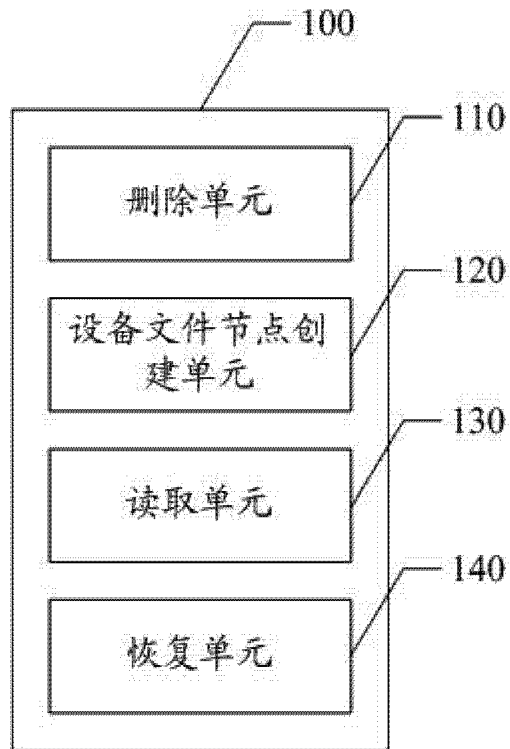


图 2

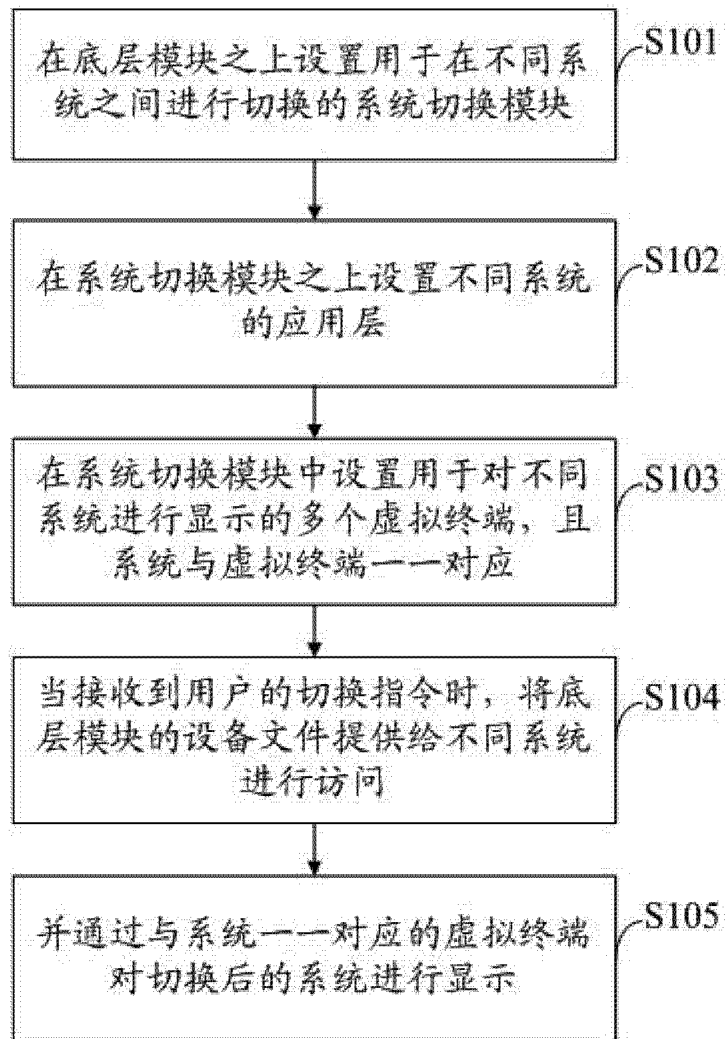


图 3

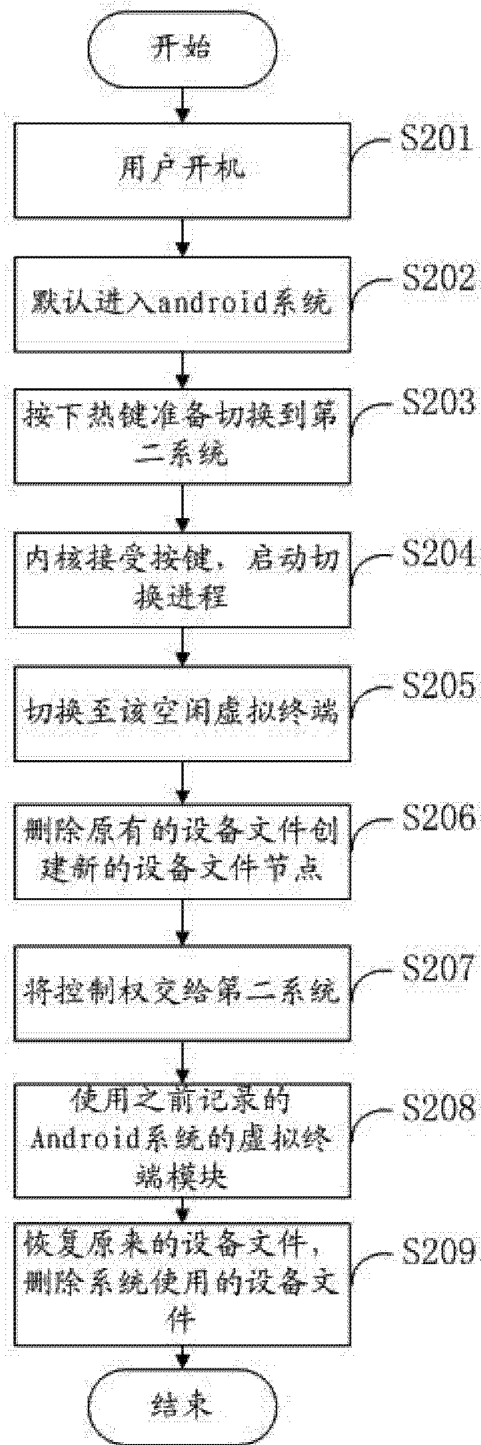


图 4