



(12) 发明专利申请

(10) 申请公布号 CN 116339759 A

(43) 申请公布日 2023. 06. 27

(21) 申请号 202310309459.9

(22) 申请日 2023.03.27

(71) 申请人 重庆长安汽车股份有限公司
地址 400023 重庆市江北区建新东路260号

(72) 发明人 韩志明

(74) 专利代理机构 北京润平知识产权代理有限公司 11283
专利代理师 赵小雨

(51) Int. Cl.
G06F 8/61 (2018.01)
G06F 11/36 (2006.01)

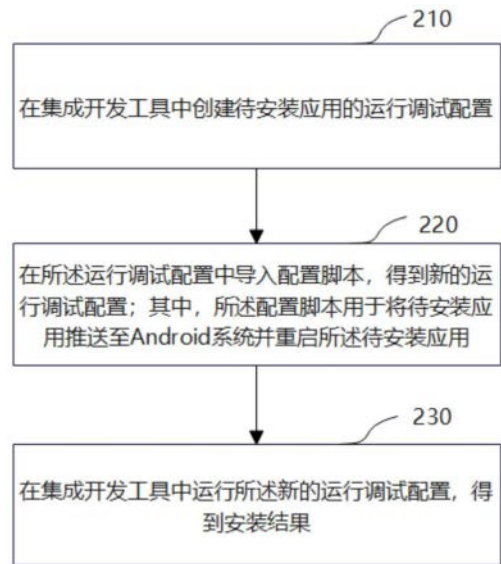
权利要求书2页 说明书8页 附图3页

(54) 发明名称

Android系统应用的安装方法及装置

(57) 摘要

本发明涉及一种Android系统应用的安装方法及装置,属于Android软件开发技术领域。Android系统应用的安装方法包括:在集成开发工具中创建待安装应用的运行调试配置;在运行调试配置中导入配置脚本,得到新的运行调试配置;其中,配置脚本用于将待安装应用推送至Android系统并重启待安装应用;在集成开发工具中运行新的运行调试配置,得到安装结果。通过以集成开发工具为基础,结合配置脚本的内容可以将系统应用直接安装到Android系统,以用于调试运行,安装过程只需要运行调试配置就可以实现安装,从而避免了在调试过程中频繁地安装打包APK并手动推送的问题,操作简单,提高了调试运行的效率。



1. 一种Android系统应用的安装方法,其特征在于,包括:
在集成开发工具中创建待安装应用的运行调试配置;
在所述运行调试配置中导入配置脚本,得到新的运行调试配置;
在集成开发工具中运行所述新的运行调试配置,得到安装结果;
其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用。
2. 根据权利要求1所述的Android系统应用的安装方法,其特征在于,所述集成开发工具为Android studio,所述在所述运行调试配置中导入配置脚本,得到新的运行调试配置,包括:
在所述运行调试配置中修改Installation Options和Launch Options;
在所述运行调试配置中的Before Launch中导入配置脚本,得到新的运行调试配置。
3. 根据权利要求1所述的Android系统应用的安装方法,其特征在于,所述配置脚本的创建过程包括:
获取APK的存放路径、名称以及推送目标位置;其中,所述APK为本地编译待安装应用后得到;根据所述APK的存放路径、名称以及推送目标位置,创建命令脚本bat文件,以得到配置脚本。
4. 根据权利要求3所述的Android系统应用的安装方法,其特征在于,所述在集成开发工具中运行所述新的运行调试配置,包括:
基于所述新的运行调试配置,对Android系统进行初始化操作;
通过所述新的运行调试配置中的脚本命令将所述APK推送至所述推送目标位置,并重启所述APK对应的应用。
5. 根据权利要求3所述的Android系统应用的安装方法,其特征在于,所述重启所述APK对应的应用,包括:
判断所述APK对应的应用的级别;
在确定所述APK对应的应用的级别为低的情况下,通过执行第一命令以强制停止所述应用,并重新启动所述APK的主Activity。
6. 根据权利要求5所述的Android系统应用的安装方法,其特征在于,还包括:
在确定所述APK对应的应用的级别为高的情况下,获取进程ID,并根据所述进程ID采用adb shell kill pid的方式重启所述应用。
7. 根据权利要求3所述的Android系统应用的安装方法,其特征在于,所述重启所述APK对应的应用,包括:
判断是否需要启动管理服务;
在确定需要启动管理服务的情况下,重启当前编译应用关联的系统服务。
8. 一种Android系统应用的安装装置,其特征在于,包括:
创建模块,用于在集成开发工具中创建待安装应用的运行调试配置;
导入模块,用于在所述运行调试配置中导入配置脚本,得到新的运行调试配置;其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用;
运行模块,用于在集成开发工具中运行所述新的运行调试配置,得到安装结果。
9. 一种处理器,其特征在于,被配置成执行权利要求1至7中任一项所述的Android系统应用的安装方法。

10. 一种机器可读存储介质,该机器可读存储介质上存储有指令,其特征在于,该指令在被处理器执行时使得所述处理器被配置成执行权利要求1至7中任一项所述的Android系统应用的安装方法。

Android系统应用的安装方法及装置

技术领域

[0001] 本发明涉及Android软件开发技术领域,具体地涉及一种Android系统应用的安装方法、一种Android系统应用的安装装置、一种机器可读存储介质及一种处理器。

背景技术

[0002] 针对以Android系统为基础的应用开发,目前通用的开发方式有两种,一种是将应用代码放入系统源码中一起参与编译,最终通过编译完成的系统镜像烧录到真机达到安装的目的。这种方式依赖源码环境,且编译较慢,修改代码费时费力。为了提高编译效率,另一种方式是将代码独立作为Android Studio工程,通过gradle脚本编译成APK,然后将APK应用推送到系统对应目录,最后重启系统达到替换APK应用的目的。这种方式在进行调试时,需要频繁推送APK到Android系统存储目录并重启Android系统,效率不高且繁琐。

[0003] 因此,现有技术在进行Android系统应用调试运行时,采用上述第二种方式存在需要频繁地安装打包APK并手动推送,导致效率不高且操作繁琐的问题。

发明内容

[0004] 本发明实施例的目的是提供一种Android系统应用的安装方法、一种Android系统应用的安装装置、一种机器可读存储介质及一种处理器,该Android系统应用的安装方法可以使不同位置区域的人员对车机系统进行不同的控制,以满足不同乘坐者的个性需求。

[0005] 为了实现上述目的,本申请第一方面提供一种Android系统应用的安装方法,包括:

[0006] 在集成开发工具中创建待安装应用的运行调试配置;

[0007] 在所述运行调试配置中导入配置脚本,得到新的运行调试配置;

[0008] 在集成开发工具中运行所述新的运行调试配置,得到安装结果;

[0009] 其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用。

[0010] 在本申请实施例中,所述集成开发工具为Android studio,所述在所述运行调试配置中导入配置脚本,得到新的运行调试配置,包括:

[0011] 在所述运行调试配置中修改Installation Options和Launch Options;

[0012] 在所述运行调试配置中的Before Launch中导入配置脚本,得到新的运行调试配置。

[0013] 在本申请实施例中,所述配置脚本的创建过程包括:

[0014] 获取APK的存放路径、名称以及推送目标位置;其中,所述APK为本地编译待安装应用后得到;

[0015] 根据所述APK的存放路径、名称以及推送目标位置,创建命令脚本bat文件,以得到配置脚本。

[0016] 在本申请实施例中,所述在集成开发工具中运行所述新的运行调试配置,包括:

- [0017] 基于所述新的运行调试配置,对Android系统进行初始化操作;
- [0018] 通过所述新的运行调试配置中的脚本命令将所述APK推送至所述推送目标位置,并重启所述APK对应的应用。
- [0019] 在本申请实施例中,所述重启所述APK对应的应用,包括:
- [0020] 判断所述APK对应的应用的级别;
- [0021] 在确定所述APK对应的应用的级别为低的情况下,通过执行第一命令以强制停止所述应用,并重新启动所述APK的主Activity。
- [0022] 在本申请实施例中,还包括:
- [0023] 在确定所述APK对应的应用的级别为高的情况下,获取进程ID,并根据所述进程ID采用adb shell kill pid的方式重启所述应用。
- [0024] 在本申请实施例中,所述重启所述APK对应的应用,包括:
- [0025] 判断是否需要启动管理服务;
- [0026] 在确定需要启动管理服务的情况下,重启当前编译应用关联的系统服务。
- [0027] 本申请第二方面提供一种Android系统应用的安装装置,包括:
- [0028] 创建模块,用于在集成开发工具中创建待安装应用的运行调试配置;
- [0029] 导入模块,用于在所述运行调试配置中导入配置脚本,得到新的运行调试配置;其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用;
- [0030] 运行模块,用于在集成开发工具中运行所述新的运行调试配置,得到安装结果。
- [0031] 本申请第三方面提供一种处理器,被配置成执行上述的Android系统应用的安装方法。
- [0032] 本申请第四方面提供一种机器可读存储介质,该机器可读存储介质上存储有指令,该指令在被处理器执行时使得所述处理器被配置成执行上述的Android系统应用的安装方法。
- [0033] 本发明的有益效果:
- [0034] 通过在集成开发工具中创建待安装应用的运行调试配置,然后在所述运行调试配置中导入配置脚本,得到新的运行调试配置;其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用;最后在集成开发工具中运行所述新的运行调试配置,得到安装结果。通过以集成开发工具为基础,结合配置脚本的内容可以将系统应用直接安装到Android系统,以用于调试运行,安装过程只需要运行调试配置就可以实现安装,从而避免了在调试过程中频繁地安装打包APK并手动推送的问题,操作简单,提高了调试运行的效率。同时,在安装过程中,采用一键编译代码并安装系统应用APK就可以及时生效,无需重启Android系统,大大提高了开发效率,减少了开发过程出错的概率。
- [0035] 本发明实施例的其它特征和优点将在随后的具体实施方式部分予以详细说明。

附图说明

[0036] 附图是用来提供对本发明实施例的进一步理解,并且构成说明书的一部分,与下面的具体实施方式一起用于解释本发明实施例,但并不构成对本发明实施例的限制。在附图中:

[0037] 图1示意性示出了根据本申请实施例的一种Android系统应用的安装方法的流程

示意图；

[0038] 图2示意性示出了根据本申请实施例的多维度编译流程图；

[0039] 图3示意性示出了根据本申请实施例的一种Android系统应用的安装装置的结构框图；

[0040] 图4示意性示出了根据本申请实施例的计算机设备的内部结构图。

[0041] 其中,410-创建模块;420-导入模块;430-运行模块;A01-处理器;A02-网络接口;A03-内存储器;A04-显示屏;A05-输入装置;A06-非易失性存储介质;B01-操作系统;B02-计算机程序。

具体实施方式

[0042] 以下将参照附图和优选实施例来说明本发明的实施方式,本领域技术人员可由本说明书中所揭露的内容轻易地了解本发明的其他优点与功效。本发明还可以通过另外不同的具体实施方式加以实施或应用,本说明书中的各项细节也可以基于不同观点与应用,在没有背离本发明的精神下进行各种修饰或改变。应当理解,优选实施例仅为了说明本发明,而不是为了限制本发明的保护范围。

[0043] 以下结合附图对本发明实施例的具体实施方式进行详细说明。应当理解的是,此处所描述的具体实施方式仅用于说明和解释本发明实施例,并不用于限制本发明实施例。

[0044] 请参看图1-图2,图1示意性示出了根据本申请实施例的一种Android系统应用的安装方法的流程示意图;图2示意性示出了根据本申请实施例的多维度编译流程图。

[0045] 本实施例提供了一种Android系统应用的安装方法,包括以下步骤:

[0046] 步骤210:在集成开发工具中创建待安装应用的运行调试配置;

[0047] 在本实施例中,集成开发工具可以根据实际情况选用,比如Android studio。在集成开发工具为Android studio的情况下,可以在Android studio运行配置中新增run/debug configurations,以创建待安装应用的运行调试配置。

[0048] 步骤220:在所述运行调试配置中导入配置脚本,得到新的运行调试配置;其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用。

[0049] 在本实施例中,所述配置脚本可以是预先根据待安装应用的相关信息编写好的。其主要向Android系统的设备推送app到相应目录并重启应用,比如将应用推送到车上的车载娱乐Android系统。

[0050] 相应地,在一些实施例中,所述集成开发工具为Android studio,所述在所述运行调试配置中导入配置脚本,得到新的运行调试配置,包括以下步骤:

[0051] 首先,在所述运行调试配置中修改Installation Options和Launch Options;

[0052] 然后,在所述运行调试配置中的Before Launch中导入配置脚本,得到新的运行调试配置。

[0053] 在本实施例中,运行调试配置为run/debug configurations,可以在run/debug configurations配置项中修改Installation Options和Launch Options,然后在Before Launch配置中增加配置脚本,从而得到新的run/debug configurations。

[0054] 需要说明的是,可以预先创建待安装应用,并通过集成开发工具编译通过,确保代码正常编译没有问题。比如:创建对应的系统应用Android studio项目并编译通过。

[0055] 其中,所述配置脚本可以是用于向Android系统推送APK的命令脚本bat文件,该文件的主要内容包括:对Android系统进行root,remount等操作;定义本地编译生成APK路径和APK名称,定义将要推送的Android系统目标路径与目标位置;通过脚本命令将编译好的APK推送到Android系统目标位置;通过命令重启编译的系统APK应用。

[0056] 其中,所述配置脚本的创建过程包括以下步骤:

[0057] 首先,获取APK的存放路径、名称以及推送目标位置;其中,所述APK为本地编译待安装应用后得到;

[0058] 然后,根据所述APK的存放路径、名称以及推送目标位置,创建命令脚本bat文件,以得到配置脚本。

[0059] 需要说明的是,本发明针对的是系统级应用,一般位于Android系统目录/system/目录下,这种应用无法通过普通安装命令直接进行安装,且大部分需要安装后重启才能生效。

[0060] 在本实施例中,所述推送目标位置目标路径与目标位置。上述创建命令脚本bat文件包括:创建对Android系统进行root,remount等操作的脚本;创建通过脚本命令将编译好的APK推送到目标位置的脚本;创建通过命令重启APK对应的应用的脚本。

[0061] 步骤230:在集成开发工具中运行所述新的运行调试配置,得到安装结果。

[0062] 在本实施例中,新的运行调试配置中设置有配置脚本,在运行时就可以执行配置脚本。运行时执行了以下步骤:

[0063] 首先,基于所述新的运行调试配置,对Android系统进行初始化操作;在本实施例中,初始化操作包括root,remount等操作,以获取应用权限等属性。

[0064] 然后,通过所述新的运行调试配置中的脚本命令将所述APK推送至所述推送目标位置,并重启所述APK对应的应用。

[0065] 在一些实施例中,由于不同系统应用权限及重要程度不同,重启策略也会有所区别,所述重启所述APK对应的应用包括以下步骤:

[0066] 首先,判断所述APK对应的应用的级别;在本实施例中,由于Android系统会分配给应用相应的属性信息,根据这些属性信息可以判断级别。比如:可以根据目标路径来确定出应用的级别。

[0067] 然后,在确定所述APK对应的应用的级别为低的情况下,通过执行第一命令以强制停止所述应用,并重新启动所述APK的主Activity。

[0068] 在本实施例中,所述第一命令是adb am force-stop packageName (packageName为对应程序包名,即APK的名称)。比如:相册,日历等级别较低的应用可以直接通过命令adb am force-stop packageName (packageName为对应程序包名)来强制停止,然后重新启动对应apk的主Activity。

[0069] 然后,在确定所述APK对应的应用的级别为高的情况下,获取进程ID,并根据所述进程ID采用adb shell kill pid的方式重启所述应用。

[0070] 在本实施例中,一些级别较高的应用如systemui launcher等无法用强制停止的命令,这时需要获取当前应用的进程ID强制kill,由于Android系统机制的影响,高级别系统应用在被杀掉后会自动重启,所以可以直接通过获取到进程ID,然后使用adb shell kill pid的方式达到重启的目的。

[0071] 在一些实施例中,由于Android版本及设备的差异,重启应用的方式可能并不通用。所述重启所述APK对应的应用包括以下步骤:

[0072] 首先,判断是否需要启动管理服务;在本实施例中,对于应用关联有其他子应用的情况下,则需要启动管理服务,管理服务用于管理应用。

[0073] 然后,在确定需要启动管理服务的条件下,重启systemui关联的系统服务。

[0074] 在本实施例中,以Android 9上的systemui为例,它的启动是由系统指定的服务来启动并加载的,首先任何通过adb shell kill pid的方式重启的应用均需另起sh脚本用以获取应用进程id。其次根据进程id强制关闭进程后,应用重启可能会不生效,不同的应用重启策略有所不同,systemui重启后相关的服务没有重启机制,因此需要手动重启systemui关联的系统服务:adb shell am startservice -n com.android.systemui/com.android.systemui.SystemUIService。

[0075] 通过对应用的级别进行判断,不同的级别采用不同的重启策略以及不同的Android版采用不同的重启策略,保证了应用的成功重启,进而保证了Android系统应用安装的成功率。

[0076] 上述实现过程中,通过在集成开发工具中创建待安装应用的运行调试配置,然后在所述运行调试配置中导入配置脚本,得到新的运行调试配置;其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用;最后在集成开发工具中运行所述新的运行调试配置,得到安装结果。通过以集成开发工具为基础,结合配置脚本的内容可以将系统应用直接安装到Android系统,以用于调试运行,安装过程只需要运行调试配置就可以实现安装,从而避免了在调试过程中频繁地安装打包APK并手动推送的问题,操作简单,提高了调试运行的效率。同时,在安装过程中,采用一键编译代码并安装系统应用APK就可以及时生效,无需重启Android系统,大大提高了开发效率,减少了开发过程出错的概率。

[0077] 图1为一个实施例中Android系统应用的安装方法的流程示意图。应该理解的是,虽然图1的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,这些步骤可以以其它的顺序执行。而且,图1中的至少一部分步骤可以包括多个子步骤或者多个阶段,这些子步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,这些子步骤或者阶段的执行顺序也不必然是依次进行。

[0078] 在一个实施例中,如图3所示,图3示意性示出了根据本申请实施例的一种Android系统应用的安装装置的结构框图。提供了一种Android系统应用的安装装置,包括创建模块410、导入模块420和运行模块430,其中:

[0079] 创建模块410,用于在集成开发工具中创建待安装应用的运行调试配置;

[0080] 导入模块420,用于在所述运行调试配置中导入配置脚本,得到新的运行调试配置;其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用;

[0081] 运行模块430,用于在集成开发工具中运行所述新的运行调试配置,得到安装结果。

[0082] 所述Android系统应用的安装装置包括处理器和存储器,上述创建模块410、导入模块420和运行模块430等均作为程序单元存储在存储器中,由处理器执行存储在存储器中的上述程序模块中实现相应的功能。

[0083] 处理器中包含内核,由内核去存储器中调取相应的程序单元。内核可以设置一个或以上,通过调整内核参数来避免在调试过程中频繁地安装打包APK并手动推送的问题,简化操作,提高调试运行的效率。

[0084] 存储器可能包括计算机可读介质中的非永久性存储器,随机存取存储器(RAM)和/或非易失性内存等形式,如只读存储器(ROM)或闪存(flash RAM),存储器包括至少一个存储芯片。

[0085] 本发明实施例提供了一种机器可读存储介质,其上存储有程序,该程序被处理器执行时实现所述Android系统应用的安装方法。

[0086] 本发明实施例提供了一种处理器,所述处理器用于运行程序,其中,所述程序运行时执行所述Android系统应用的安装方法。

[0087] 在一个实施例中,提供了一种计算机设备,该计算机设备可以是终端,其内部结构图可以如图4所示。该计算机设备包括通过系统总线连接的处理器A01、网络接口A02、显示屏A04、输入装置A05和存储器(图中未示出)。其中,该计算机设备的处理器A01用于提供计算和控制能力。该计算机设备的存储器包括内存储器A03和非易失性存储介质A06。该非易失性存储介质A06存储有操作系统B01和计算机程序B02。该内存储器A03为非易失性存储介质A06中的操作系统B01和计算机程序B02的运行提供环境。该计算机设备的网络接口A02用于与外部的终端通过网络连接通信。该计算机程序被处理器A01执行时以实现一种Android系统应用的安装方法。该计算机设备的显示屏A04可以是液晶显示屏或者电子墨水显示屏,该计算机设备的输入装置A05可以是显示屏上覆盖的触摸层,也可以是计算机设备外壳上设置的按键、轨迹球或触控板,还可以是外接的键盘、触控板或鼠标等。

[0088] 本领域技术人员可以理解,图4中示出的结构,仅仅是与本申请方案相关的部分结构的框图,并不构成对本申请方案所应用于其上的计算机设备的限定,具体的计算机设备可以包括比图中所示更多或更少的部件,或者组合某些部件,或者具有不同的部件布置。

[0089] 在一个实施例中,本申请提供的Android系统应用的安装装置可以实现为一种计算机程序的形式,计算机程序可在如图4所示的计算机设备上运行。计算机设备的存储器中可存储组成该Android系统应用的安装装置的各个程序模块。各个程序模块构成的计算机程序使得处理器执行本说明书中描述的本申请各个实施例的Android系统应用的安装方法中的步骤。

[0090] 图4所示的计算机设备可以通过如图3所示的Android系统应用的安装装置中的创建模块410执行步骤210。计算机设备可通过导入模块420执行步骤220、通过运行模块430执行步骤230。

[0091] 本申请实施例提供了一种设备,设备包括处理器、存储器及存储在存储器上并可在处理器上运行的程序,处理器执行程序时实现以下步骤:

[0092] 在集成开发工具中创建待安装应用的运行调试配置;

[0093] 在所述运行调试配置中导入配置脚本,得到新的运行调试配置;

[0094] 在集成开发工具中运行所述新的运行调试配置,得到安装结果;

[0095] 其中,所述配置脚本用于将待安装应用推送至Android系统并重启所述待安装应用。

[0096] 在一个实施例中,所述集成开发工具为Android studio,所述在所述运行调试配

置中导入配置脚本,得到新的运行调试配置,包括:

[0097] 在所述运行调试配置中修改Installation Options和Launch Options;

[0098] 在所述运行调试配置中的Before Launch中导入配置脚本,得到新的运行调试配置。

[0099] 在一个实施例中,所述配置脚本的创建过程包括:

[0100] 获取APK的存放路径、名称以及推送目标位置;其中,所述APK为本地编译待安装应用后得到;

[0101] 根据所述APK的存放路径、名称以及推送目标位置,创建命令脚本bat文件,以得到配置脚本。

[0102] 在一个实施例中,所述在集成开发工具中运行所述新的运行调试配置,包括:

[0103] 基于所述新的运行调试配置,对Android系统进行初始化操作;

[0104] 通过所述新的运行调试配置中的脚本命令将所述APK推送至所述推送目标位置,并重启所述APK对应的应用。

[0105] 在一个实施例中,所述重启所述APK对应的应用,包括:

[0106] 判断所述APK对应的应用的级别;

[0107] 在确定所述APK对应的应用的级别为低的情况下,通过执行第一命令以强制停止所述应用,并重新启动所述APK的主Activity。

[0108] 在一个实施例中,还包括:

[0109] 在确定所述APK对应的应用的级别为高的情况下,获取进程ID,并根据所述进程ID采用adb shell kill pid的方式重启所述应用。

[0110] 在一个实施例中,所述重启所述APK对应的应用,包括:

[0111] 判断是否需要启动管理服务;

[0112] 在确定需要启动管理服务的情况下,重启当前编译应用关联的系统服务。

[0113] 本领域内的技术人员应明白,本申请的实施例可提供为方法、系统、或计算机程序产品。因此,本申请可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0114] 本申请是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0115] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0116] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计

计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0117] 在一个典型的配置中,计算设备包括一个或多个处理器(CPU)、输入/输出接口、网络接口和内存。

[0118] 存储器可能包括计算机可读介质中的非永久性存储器,随机存取存储器(RAM)和/或非易失性内存等形式,如只读存储器(ROM)或闪存(flash RAM)。存储器是计算机可读介质的示例。

[0119] 计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。

[0120] 还需要说明的是,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、商品或者设备不仅包括那些要素,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、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括要素的过程、方法、商品或者设备中还存在另外的相同要素。

[0121] 以上仅为本申请的实施例而已,并不用于限制本申请。对于本领域技术人员来说,本申请可以有各种更改和变化。凡在本申请的精神和原理之内所作的任何修改、等同替换、改进等,均应包含在本申请的权利要求范围之内。

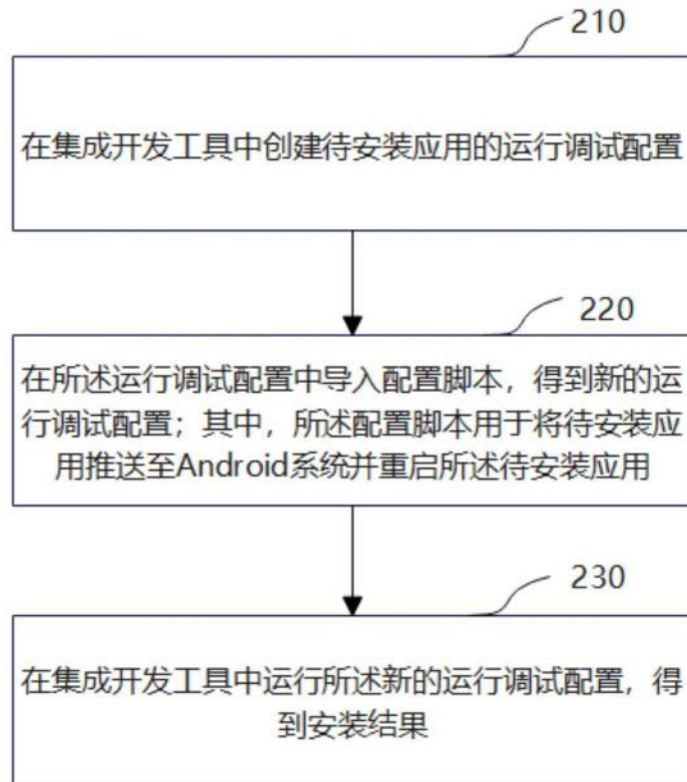


图1



图2

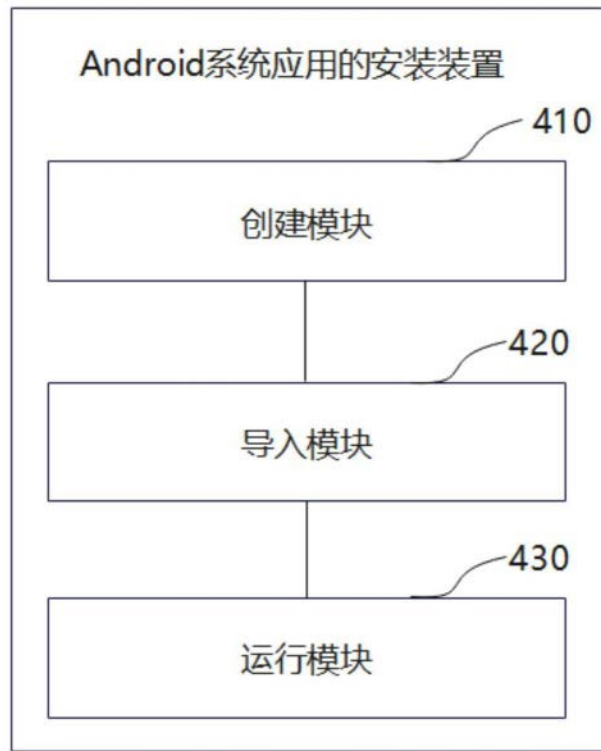


图3

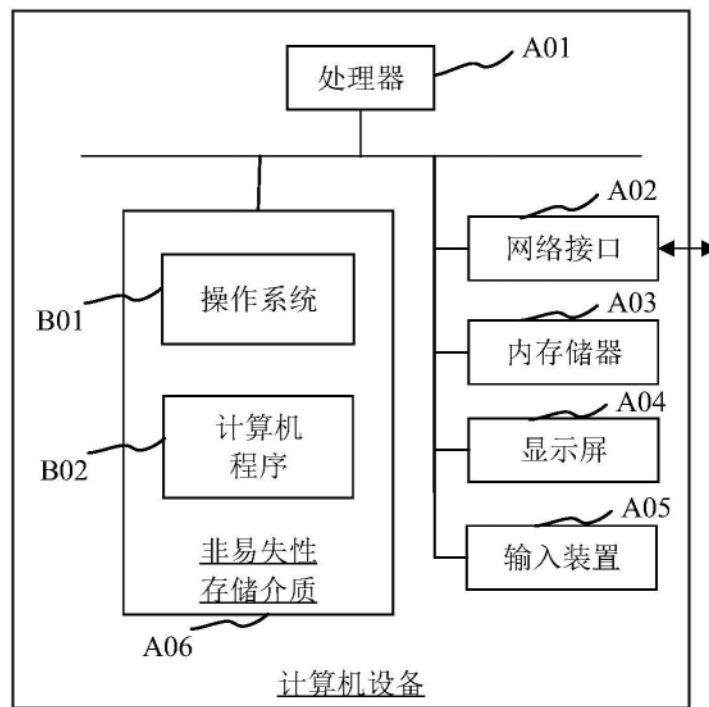


图4