



(12) 发明专利申请

(10) 申请公布号 CN 112083953 A

(43) 申请公布日 2020.12.15

(21) 申请号 202010873725.7

(22) 申请日 2020.08.26

(71) 申请人 武汉普利商用机器有限公司
地址 430000 湖北省武汉市东湖开发区光谷大道70号

申请人 精伦电子股份有限公司

(72) 发明人 吉纲 彭骏 李会丽 谭尧 龚琦
郑卓鑫 韩达

(74) 专利代理机构 北京路浩知识产权代理有限公司 11002

代理人 郭亮

(51) Int. Cl.

G06F 8/72 (2018.01)

G06F 8/71 (2018.01)

G06F 8/41 (2018.01)

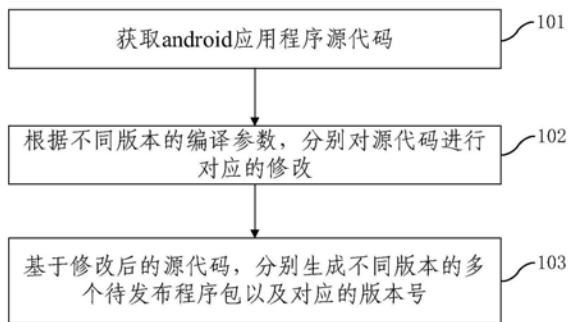
权利要求书2页 说明书6页 附图2页

(54) 发明名称

Android应用程序构建方法及装置

(57) 摘要

本发明实施例提供一种android应用程序构建方法及装置,该方法包括:获取android应用程序源代码;根据不同版本的编译参数,分别对源代码进行对应的修改;基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。通过设置自定义编译参数,在编译前、编译中都可以灵活地修改源代码、可选地执行一些自定义工作,一次性生成多个不同的程序发布版本,无需在每次有不同需求时重复修改程序版本,并且可将步骤繁琐的应用程序构建工作变成自动化操作。



1. 一种android应用程序构建方法,其特征在于,包括:
获取android应用程序源代码;
根据不同版本的编译参数,分别对源代码进行对应的修改;
基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。
2. 根据权利要求1所述的android应用程序构建方法,其特征在于,所述编译参数包括:
Meta-data属性值、jni目标框架、批处理脚本、平台签名文件标识、版本名称和版本号。
3. 根据权利要求1所述的android应用程序构建方法,其特征在于,分别对源代码进行对应的修改,包括:
将源代码中版本号修改为编译参数中的版本号;
将源代码中版本名称修改为编译参数中的版本名称;
将源代码中属性字段修改为编译参数中的属性字段。
4. 根据权利要求1所述的android应用程序构建方法,其特征在于,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号,包括:
使用ant编译工具,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。
5. 根据权利要求1所述的android应用程序构建方法,其特征在于,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:
对待发布程序包进行系统签名,得到最终待发布程序包。
6. 根据权利要求1所述的android应用程序构建方法,其特征在于,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:
将编译参数生成的对应文件与待发布程序包进行绑定;
若检测到编译参数的动态修改,则用apktool工具将所述待发布程序包进行解包,得到应用清单文件;
根据动态修改的编译参数,修改应用清单文件的属性值;
用apktool工具对修改后的应用清单文件,重新打包成程序文件。
7. 根据权利要求1所述的android应用程序构建方法,其特征在于,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:
提取版本信息和日期,将待发布程序包、编译参数配置文件、脚本文件复制到创建的程序发布文件夹中;
根据不同程序版本,生成多个不同的程序发布文件夹。
8. 一种android应用程序构建装置,其特征在于,包括:
源代码提取模块,用于获取android应用程序源代码;
编译预处理模块,用于根据不同版本的编译参数,分别对源代码进行对应的修改;
应用包构建模块,用于基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。
9. 一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述处理器执行所述程序时实现如权利要求1至7任一项所述android应用程序构建方法的步骤。
10. 一种非暂态计算机可读存储介质,其上存储有计算机程序,其特征在于,该计算机

程序被处理器执行时实现如权利要求1至7任一项所述android应用程序构建方法的步骤。

Android应用程序构建方法及装置

技术领域

[0001] 本发明涉及计算机软件技术领域,尤其涉及一种android应用程序构建方法及装置。

背景技术

[0002] 在软件项目的开发过程中,开发人员需要在本地开发并编译打包,然后远程登入服务器,并且将编译打包后的软件包上传至服务器,同时根据不同的现场环境规则,在服务器上修改与现场环境相对应的配置文件,接着根据目的地址而执行分发操作,以完成发布任务。

[0003] 使用Apache的java生成工具ant工具可以帮助开发人员构建通用的android应用程序。但是,程序包里还需要动态添加配置文件/脚本等资源文件;编译过程中需要根据预设参数动态修改java代码或者xml文件。面对这些问题,目前的方法均需要一一对应手动修改后重新打包,导致开发效率大大降低。

发明内容

[0004] 本发明实施例提供一种android应用程序构建方法及装置,用以解决现有技术中的缺陷。

[0005] 本发明实施例提供一种android应用程序构建方法,包括:获取android应用程序源代码;根据不同版本的编译参数,分别对源代码进行对应的修改;基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0006] 根据本发明一个实施例的android应用程序构建方法,编译参数包括:Meta-data属性值、jni目标框架、批处理脚本、平台签名文件标识、版本名称和版本号。

[0007] 根据本发明一个实施例的android应用程序构建方法,分别对源代码进行对应的修改,包括:将源代码中版本号修改为编译参数中的版本号;将源代码中版本名称修改为编译参数中的版本名称;将源代码中属性字段修改为编译参数中的属性字段。

[0008] 根据本发明一个实施例的android应用程序构建方法,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号,包括:使用ant编译工具,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0009] 根据本发明一个实施例的android应用程序构建方法,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:对待发布程序包进行系统签名,得到最终待发布程序包。

[0010] 根据本发明一个实施例的android应用程序构建方法,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:将编译参数生成的对应文件与待发布程序包进行绑定;若检测到编译参数的动态修改,则用apktool工具将所述待发布程序包进行解包,得到应用清单文件;根据动态修改的编译参数,修改应用清单文件的属性值;用apktool工具对修改后的应用清单文件,重新打包成程序文件。

[0011] 根据本发明一个实施例的android应用程序构建方法,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:提取版本信息和日期,将待发布程序包、编译参数配置文件、脚本文件复制到创建的程序发布文件夹中;根据不同程序版本,生成多个不同的程序发布文件。

[0012] 本发明实施例还提供一种android应用程序构建装置,包括:源代码提取模块,用于获取android应用程序源代码;编译预处理模块,用于根据不同版本的编译参数,分别对源代码进行对应的修改;应用包构建模块,用于基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0013] 本发明实施例还提供一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,所述处理器执行所述程序时实现如上述任一种所述android应用程序构建方法的步骤。

[0014] 本发明实施例还提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现如上述任一种所述android应用程序构建方法的步骤。

[0015] 本发明实施例提供的android应用程序构建方法及装置,通过设置自定义编译参数,在编译前、编译中都可以灵活地修改源代码、可选地执行一些自定义工作,一次性生成多个不同的程序发布版本,无需在每次有不同需求时重复修改程序版本,并且可将步骤繁琐的应用程序构建工作变成自动化操作。

附图说明

[0016] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0017] 图1是本发明实施例提供的android应用程序构建方法的流程示意图;

[0018] 图2是本发明实施例提供的android应用程序构建装置的结构示意图;

[0019] 图3是本发明实施例提供的一种电子设备的结构示意图。

具体实施方式

[0020] 为使本发明实施例的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0021] 下面结合图1-图3描述本发明实施例的android应用程序构建方法及装置。图1是本发明实施例提供的android应用程序构建方法的流程示意图,如图1所示,本发明实施例提供一种android应用程序构建方法,包括:

[0022] 101、取android应用程序源代码。

[0023] 可选的,同时还包括获取源代码的修订号。使用svn工具从代码仓库提取最新版的android应用程序源码,同时使用svn命令提取代码的修订号,生成版本信息文本文件,初始只有修订号字段,例:app_version=7153。

[0024] 102、根据不同版本的编译参数,分别对源代码进行对应的修改。

[0025] 这些参数包括:是否需要动态修改源代码,以及对应属性值的修改,如meta-data属性值的修改。不同版本,可以是一个集合的形式体现,比如[人脸门禁考勤app轻量级版、人脸门禁考勤app云平台版、人证比对标准版app]。每个版本对于不同的修改方式和编译参数。

[0026] 103、基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0027] 可选的,生成不同版本的多个待发布程序包以及对应的版本号,同时还将版本号中加入上述修订号。

[0028] 每生成一个待发布程序包文件,就提取该程序的版本信息,同时将源代码的修订号加入进行标识。例如,把属性为"software_version"的值修改为真实的软件版本+svn修订号,示例:V1.2.0(6000)。

[0029] 本发明实施例的android应用程序构建方法,通过设置自定义编译参数,在编译前、编译中都可以灵活地修改源代码、可选地执行一些自定义工作,一次性生成多个不同的程序发布版本,无需在每次有不同需求时重复修改程序版本,并且可将步骤繁琐的应用程序构建工作变成自动化操作。

[0030] 基于上述实施例的内容,作为一种可选实施例,所述编译参数包括:修改后的Meta-data属性值、jni目标框架、批处理脚本、平台签名文件标识、版本名称和版本号。

[0031] 编译参数可以包括:是否需要动态修改源代码;是否需要修改meta-data属性值;是否需要编译jni代码;设置jni的目标架构;是否需要执行批处理脚本;是否需要平台签名;使用哪个平台签名文件。版本名称比如,人脸门禁考勤app轻量级版、人脸门禁考勤app云平台版和人证比对标准版app。

[0032] 基于上述实施例的内容,作为一种可选实施例,分别对源代码进行对应的修改,包括:将源代码中版本号修改为编译参数中的版本号;将源代码中版本名称修改为编译参数中的版本名称;将源代码中属性字段修改为编译参数中的属性字段。

[0033] 针对人脸算法SDK,修改源代码,把标识SDK版本的变量SDK_VERSION的值修改为参数指定的版本号,如V4.0;针对人脸服务app,修改源代码,修改应用名称为参数指定的业务名称,如人脸考勤门禁、人证比对等;针对读卡SDK,修改源代码,把标识授权方式的变量authWay属性字段,修改为参数指定的授权方式,如按型号授权、按单片机版本授权、按安全模块号授权等。如果需要修改程序的应用清单文件AndroidManifest.xml中的meta-data属性值,则进行修改,比如把属性为software_version的值修改为真实的软件版本+svn修订号,示例:V1.2.0(6000)。

[0034] 基于上述实施例的内容,作为一种可选实施例,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号,包括:使用ant编译工具,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。Ant是跨平台的构建工具,ant是使用java实现的,所以它跨平台使用简单,与编译工具make比起来语法清晰且功能强大。

[0035] 基于上述实施例的内容,作为一种可选实施例,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:对待发布程序包进行系统签名,得到最终待发布程

序包。

[0036] 在编译之前,如果需要编译jni代码,先编译jni代码,生成动态库。编译生成普通apk之后,配置为需要平台签名的apk,还需要执行一次系统签名,才能得到最终的程序文件。生成的版本信息文件中添加一行对应的版本信息,例如:iDRFaceService.apk=V1.2.0(7153)。

[0037] 基于上述实施例的内容,作为一种可选实施例,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:将编译参数生成的对应文件与待发布程序包进行绑定;若检测到编译参数的动态修改,则用apktool工具将所述待发布程序包进行解包,得到应用清单文件;根据动态修改的编译参数,修改应用清单文件的属性值;用apktool工具对修改后的应用清单文件,重新打包成程序文件。

[0038] 考虑到编译后仍存在修改需求,每个版本的待发布程序包有绑定的编译参数生成的对应文件。当有修改需求时,直接修改该文件中的编译参数。

[0039] 检测到修改后,用apktool工具把生成的程序文件解包,得到应用清单文件AndroidManifest.xml;再修改应用清单文件的meta-data属性值,把软件名称software_name修改成指定的名称;把软件erp编码software_erp修改成指定的erp编码;用apktool工具重新打包程序文件;给新程序文件签名;对齐新程序文件。

[0040] 本发明实施例的方法,在编译后仍可以灵活地修改每个版本的源代码。

[0041] 基于上述实施例的内容,作为一种可选实施例,基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号之后,还包括:提取版本信息和日期,将待发布程序包、编译参数配置文件、脚本文件复制到创建的程序发布文件夹中;根据不同程序版本,生成多个不同的程序发布文件夹。

[0042] 创建带版本信息和日期的程序发布文件夹。提取版本信息和日期,把待发布的程序文件、业务配置文件、脚本文件按一定规则复制到所创建文件夹。根据要生成的版本列表,生成多个不同的程序发布文件夹。

[0043] 自动发布到程序发布服务器:使用一组辅助工具命令spawn/expect/ssh/scp,实现自动登录到程序发布服务器,并自动把多个不同的程序发布文件夹复制到程序发布服务器,达到自动发布的目的。

[0044] 下面对本发明实施例提供的android应用程序构建装置进行描述,下文描述的android应用程序构建装置与上文描述的android应用程序构建方法可相互对应参照。

[0045] 图2是本发明实施例提供的android应用程序构建装置的结构示意图,如图2所示,该android应用程序构建装置包括:源代码提取模块201、编译预处理模块202和应用包构建模块203。其中,源代码提取模块201取android应用程序源代码;编译预处理模块202据不同版本的编译参数,分别对源代码进行对应的修改;应用包构建模块203于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0046] 本发明实施例提供的装置实施例是为了实现上述各方法实施例的,具体流程和详细内容请参照上述方法实施例,此处不再赘述。

[0047] 本发明实施例提供的android应用程序构建装置,通过设置自定义编译参数,在编译前、编译中都可以灵活地修改源代码、可选地执行一些自定义工作,一次性生成多个不同的程序发布版本,无需在每次有不同需求时重复修改程序版本,并且可将步骤繁琐的应用

程序构建工作变成自动化操作。

[0048] 图3是本发明实施例提供的一种电子设备的结构示意图,如图3所示,该电子设备可以包括:处理器(processor)301、通信接口(Communications Interface)302、存储器(memory)303和通信总线304,其中,处理器301,通信接口302,存储器303通过通信总线304完成相互间的通信。处理器301可以调用存储器303中的逻辑指令,以执行android应用程序构建方法,该方法包括:获取android应用程序源代码;根据不同版本的编译参数,分别对源代码进行对应的修改;基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0049] 此外,上述的存储器303中的逻辑指令可以通过软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读取存储介质中。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM, Read-Only Memory)、随机存取存储器(RAM, Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0050] 另一方面,本发明实施例还提供一种计算机程序产品,所述计算机程序产品包括存储在非暂态计算机可读存储介质上的计算机程序,所述计算机程序包括程序指令,当所述程序指令被计算机执行时,计算机能够执行上述各方法实施例所提供的android应用程序构建方法,该方法包括:获取android应用程序源代码;根据不同版本的编译参数,分别对源代码进行对应的修改;基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0051] 又一方面,本发明实施例还提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现以执行上述各实施例提供的android应用程序构建方法,该方法包括:获取android应用程序源代码;根据不同版本的编译参数,分别对源代码进行对应的修改;基于修改后的源代码,分别生成不同版本的多个待发布程序包以及对应的版本号。

[0052] 以上所描述的装置实施例仅仅是示意性的,其中所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性的劳动的情况下,即可以理解并实施。

[0053] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到各实施方式可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件。基于这样的理解,上述技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品可以存储在计算机可读存储介质中,如ROM/RAM、磁碟、光盘等,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行各个实施例或者实施例的某些部分所述的方法。

[0054] 最后应说明的是:以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管

参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

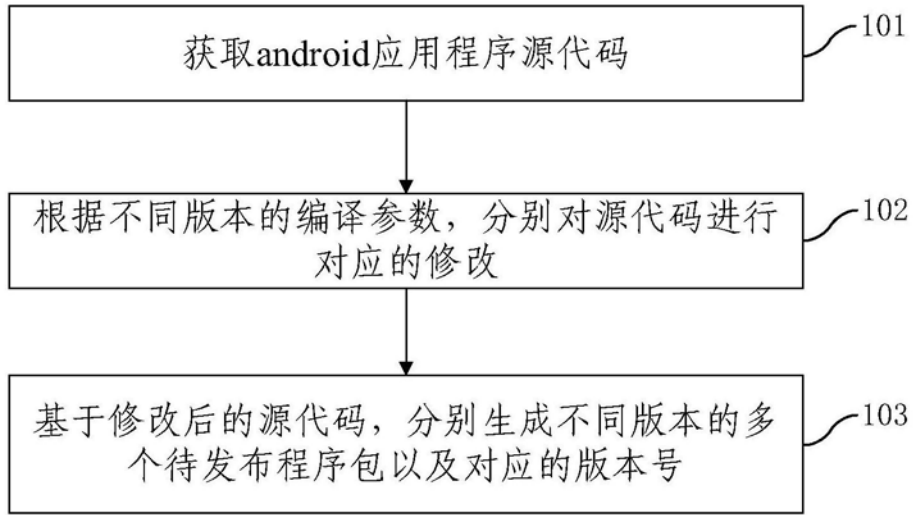


图1

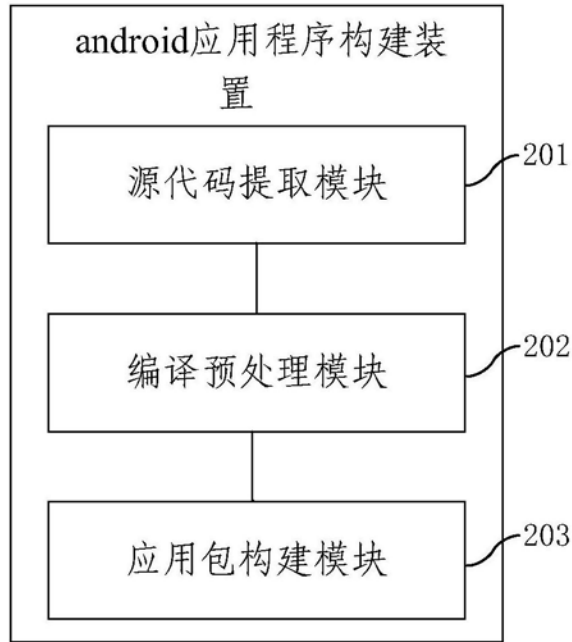


图2

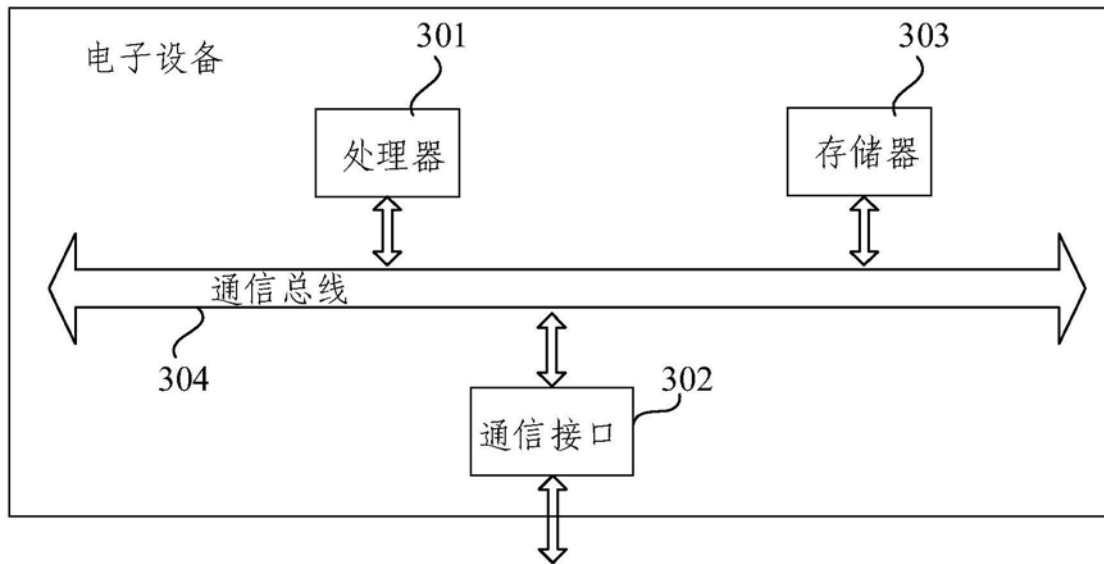


图3