



(12) 发明专利

(10) 授权公告号 CN 110225048 B

(45) 授权公告日 2022.09.23

(21) 申请号 201910532613.2

CN 105868111 A, 2016.08.17

(22) 申请日 2019.06.19

CN 107273748 A, 2017.10.20

(65) 同一申请的已公布的文献号

CN 108132873 A, 2018.06.08

申请公布号 CN 110225048 A

CN 101770004 A, 2010.07.07

(43) 申请公布日 2019.09.10

CN 102156661 A, 2011.08.17

(73) 专利权人 腾讯科技(成都)有限公司

CN 103607587 A, 2014.02.26

地址 610041 四川省成都市高新区拓东新街81号天府软件园二期C区4号楼

US 9471466 B1, 2016.10.18
高书亮等. 基于多线程和消息的局域网通信程序实现.《计算机工程与设计》.2008, (第02期),

(72) 发明人 黄造军 余强 沈姝莺 唐志旭

周广川. 多线程应用程序调试技术.《现代计算机(专业版)》.2011, (第03期),

(74) 专利代理机构 北京三高永信知识产权代理

有限责任公司 11138

专利代理师 张所明

李国徽等. 事件循环机制在嵌入式多线程应用级调试器设计中的应用.《中国科技论文》.2013, (第01期),

(51) Int. Cl.

H04L 67/06 (2022.01)

H04L 67/146 (2022.01)

H04L 9/40 (2022.01)

Michael N. Gagnon. Software Protection through Anti-Debugging.《IEEE Security & Privacy》.2007,

(56) 对比文件

CN 106980574 A, 2017.07.25

审查员 邓成

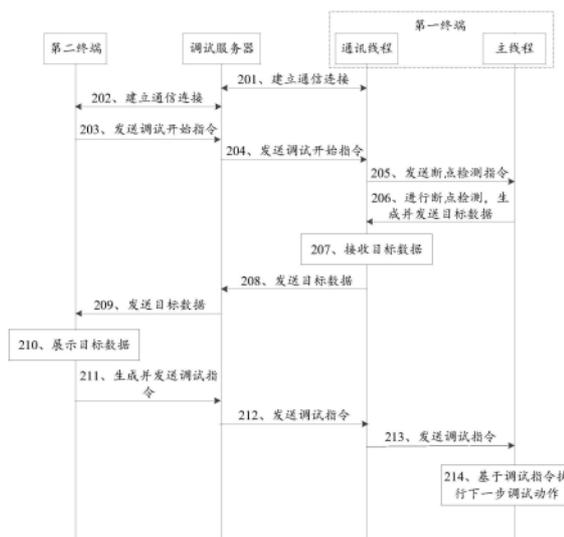
权利要求书2页 说明书15页 附图7页

(54) 发明名称

数据传输方法、装置、第一终端及存储介质

(57) 摘要

本发明公开了一种数据传输方法、装置、第一终端及存储介质,属于网络技术领域。在本发明中,第一终端通过通讯线程,实现主线程与第二终端之间的数据传输,由于在通过通讯线程向第二终端传输数据时,没有基于套接字的执行逻辑,因此第一终端和第二终端可以不处于同一局域网内,使得第二终端可以在任何地点实现对第一终端的调试,大大提升了调试过程的便捷性,提升了第一终端上应用程序的调试效率。



1. 一种数据传输方法,其特征在于,由第一终端执行,所述第一终端上运行有通讯线程,所述通讯线程用于实现调试服务器与所述第一终端之间的数据传输,所述方法包括:

当接收到对第一终端上任一应用程序的调试开始指令时,通过所述通讯线程向所述应用程序的主线程发送断点检测指令,所述断点检测指令用于指示所述主线程进行断点检测,所述断点是指在逐行执行脚本的过程中由错误导致运行中断的代码行数;

通过所述通讯线程接收所述主线程响应于所述断点检测指令所生成的目标数据,所述目标数据基于所述主线程在检测到断点后的执行数据生成,所述执行数据是指所述主线程在执行所述断点前在寄存器的函数调用栈中的函数或变量的执行数据;

通过所述通讯线程,基于钩子函数向所述调试服务器发送所述目标数据,由所述调试服务器向第二终端发送所述目标数据,所述第二终端与所述第一终端不处于同一局域网内,所述钩子函数用于在检测到命中回调事件时回调通知所述调试服务器;

当接收到所述第二终端返回的调试指令时,通过所述通讯线程向所述主线程发送所述调试指令,所述调试指令与所述目标数据对应。

2. 根据权利要求1所述的方法,其特征在于,所述通过所述通讯线程,基于钩子函数向所述调试服务器发送所述目标数据包括:

每间隔目标时长,通过所述通讯线程,基于所述钩子函数向所述调试服务器发送所述目标数据。

3. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

当接收到热更新文件时,通过所述通讯线程向所述主线程发送所述热更新文件;

当所述主线程接收到所述热更新文件时,基于所述热更新文件对所述应用程序进行热更新。

4. 根据权利要求3所述的方法,其特征在于,所述基于所述热更新文件对所述应用程序进行热更新包括:

将所述热更新文件复制至所述应用程序的运行目录;

当检测到所述应用程序处于启动状态时,从所述运行目录中加载所述热更新文件。

5. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

当接收到日志拉取指令时,通过所述通讯线程向所述主线程发送所述日志拉取指令,所述日志拉取指令用于拉取所述应用程序的日志文件;

通过所述通讯线程接收所述主线程响应于所述日志拉取指令所返回的日志文件;

通过所述通讯线程向所述第二终端发送所述日志文件。

6. 一种数据传输装置,其特征在于,所述装置上运行有通讯线程,所述通讯线程用于实现调试服务器与所述装置之间的数据传输,所述装置包括:

发送模块,用于当接收到对第一终端上任一应用程序的调试开始指令时,通过所述通讯线程向所述应用程序的主线程发送断点检测指令,所述断点检测指令用于指示所述主线程进行断点检测,所述断点是指在逐行执行脚本的过程中由错误导致运行中断的代码行数;

接收模块,用于通过所述通讯线程接收所述主线程响应于所述断点检测指令所生成的目标数据,所述目标数据基于所述主线程在检测到断点后的执行数据生成,所述执行数据是指所述主线程在执行所述断点前在寄存器的函数调用栈中的函数或变量的执行数据;

所述发送模块还包括发送单元,所述发送单元用于通过所述通讯线程,基于钩子函数向所述调试服务器发送所述目标数据,由所述调试服务器向第二终端发送所述目标数据,所述第二终端与所述第一终端不处于同一局域网内,所述钩子函数用于在检测到命中回调事件时回调通知所述调试服务器;

所述发送模块,还用于当接收到所述第二终端返回的调试指令时,通过所述通讯线程向所述主线程发送所述调试指令,所述调试指令与所述目标数据对应。

7. 根据权利要求6所述的装置,其特征在于,所述发送单元用于:

每间隔目标时长,通过所述通讯线程,基于所述钩子函数向所述调试服务器发送所述目标数据。

8. 根据权利要求6所述的装置,其特征在于,所述发送模块还用于:

当接收到热更新文件时,通过所述通讯线程向所述主线程发送所述热更新文件;

所述装置还包括:

热更新模块,用于当所述主线程接收到所述热更新文件时,基于所述热更新文件对所述应用程序进行热更新。

9. 根据权利要求8所述的装置,其特征在于,所述热更新模块用于:

将所述热更新文件复制至应用程序的运行目录;

当检测到所述应用程序处于启动状态时,从所述运行目录中加载所述热更新文件。

10. 根据权利要求6所述的装置,其特征在于,所述发送模块还用于:

当接收到日志拉取指令时,通过所述通讯线程向所述主线程发送所述日志拉取指令,所述日志拉取指令用于拉取所述应用程序的日志文件;

所述接收模块还用于:

通过所述通讯线程接收所述主线程响应于所述日志拉取指令所返回的日志文件;

所述发送模块还用于:

通过所述通讯线程向所述第二终端发送所述日志文件。

11. 一种第一终端,其特征在于,所述第一终端包括一个或多个处理器和一个或多个存储器,所述一个或多个存储器中存储有至少一条指令,所述至少一条指令由所述一个或多个处理器加载并执行以实现如权利要求1至权利要求5任一项所述的数据传输方法。

12. 一种存储介质,其特征在于,所述存储介质中存储有至少一条指令,所述至少一条指令由处理器加载并执行以实现如权利要求1至权利要求5任一项所述的数据传输方法。

数据传输方法、装置、第一终端及存储介质

技术领域

[0001] 本发明涉及网络技术领域,特别涉及一种数据传输方法、装置、第一终端及存储介质。

背景技术

[0002] 随着网络技术的发展,在终端上可以执行嵌入有lua脚本的应用程序,由于兼容性问题在执行过程中可能会产生错误(bug),为了修正应用程序中的错误,需要由技术人员对应用程序进行调试(debug)。

[0003] 目前,在上述调试过程中,终端上的本地套接字(localsocket)与调试设备上的lua套接字(luasocket)互相输入对方的互联网协议地址(internet protocol address,IP地址),从而建立起调试设备与终端之间的通信连接,当终端检测到报错事件时,终端上的本地套接字回调通知到调试设备上的lua套接字,并且本地套接字向lua套接字发送报错事件的相关信息,技术人员在调试设备上可以根据该报错事件的相关信息,定位出引发错误的代码位置。

[0004] 在上述过程中,如果调试设备与终端处于同一局域网(也即是内网),本地套接字与lua套接字互相输入对方的内网IP地址即可建立连接,而在更普遍的广域网场景下,调试设备无法获取到终端的内网IP地址,也就无法与外网的终端建立通信连接,导致调试设备无法对外网的终端进行调试,因此,上述调试方法仅能够支持局域网内调试,导致调试过程存在诸多不便,降低了调试过程的便捷性,影响了调试效率。

发明内容

[0005] 本发明实施例提供了一种数据传输方法、装置、第一终端及存储介质,能够解决调试过程便捷性低、调试效率低的问题。该技术方案如下:

[0006] 一方面,提供了一种数据传输方法,该方法包括:

[0007] 当接收到对第一终端上任一应用程序的调试开始指令时,通过所述第一终端的通讯线程向所述应用程序的主线程发送断点检测指令,所述通讯线程用于实现不同局域网内终端之间的通信;

[0008] 通过所述通讯线程接收所述主线程响应于所述断点检测指令所生成的目标数据;

[0009] 通过所述通讯线程向第二终端发送所述目标数据,所述第二终端与所述第一终端不处于同一局域网内;

[0010] 当接收到所述第二终端返回的调试指令时,通过所述通讯线程向所述主线程发送所述调试指令,所述调试指令与所述目标数据对应。

[0011] 一方面,提供了一种数据传输装置,该装置包括:

[0012] 发送模块,用于当接收到对第一终端上任一应用程序的调试开始指令时,通过所述第一终端的通讯线程向所述应用程序的主线程发送断点检测指令,所述通讯线程用于实现不同局域网内终端之间的通信;

- [0013] 接收模块,用于通过所述通讯线程接收所述主线程响应于所述断点检测指令所生成的目标数据;
- [0014] 所述发送模块,还用于通过所述通讯线程向第二终端发送所述目标数据,所述第二终端与所述第一终端不处于同一局域网内;
- [0015] 所述发送模块,还用于当接收到所述第二终端返回的调试指令时,通过所述通讯线程向所述主线程发送所述调试指令,所述调试指令与所述目标数据对应。
- [0016] 在一种可能实施方式中,所述发送模块包括:
- [0017] 发送单元,用于通过所述通讯线程向调试服务器发送所述目标数据,由所述调试服务器向所述第二终端发送所述目标数据。
- [0018] 在一种可能实施方式中,所述发送单元用于:
- [0019] 每间隔目标时长,通过所述通讯线程向所述调试服务器发送所述目标数据。
- [0020] 在一种可能实施方式中,所述发送单元用于:
- [0021] 通过所述通讯线程,基于钩子函数向所述调试服务器发送所述目标数据。
- [0022] 在一种可能实施方式中,所述装置还包括:
- [0023] 所述发送模块,还用于当接收到热更新文件时,通过所述通讯线程向所述主线程发送所述热更新文件;
- [0024] 热更新模块,用于当所述主线程接收到所述热更新文件时,基于所述热更新文件对所述应用程序进行热更新。
- [0025] 在一种可能实施方式中,所述热更新模块用于:
- [0026] 将所述热更新文件复制至应用程序的运行目录;
- [0027] 当检测到所述应用程序处于启动状态时,从所述运行目录中加载所述热更新文件。
- [0028] 在一种可能实施方式中,所述装置还包括:
- [0029] 当接收到日志拉取指令时,通过所述通讯线程向所述主线程发送所述日志拉取指令,所述日志拉取指令用于拉取所述应用程序的日志文件;
- [0030] 通过所述通信线程接收所述主线程响应于所述日志拉取指令所返回的日志文件;
- [0031] 通过所述通信线程向所述第二终端发送所述日志文件。
- [0032] 一方面,提供了一种第一终端,该第一终端包括一个或多个处理器和一个或多个存储器,该一个或多个存储器中存储有至少一条指令,该至少一条指令由该一个或多个处理器加载并执行以实现如上述任一种可能实现方式的数据传输方法所执行的操作。
- [0033] 一方面,提供了一种存储介质,该存储介质中存储有至少一条指令,该至少一条指令由处理器加载并执行以实现如上述任一种可能实现方式的数据传输方法所执行的操作。
- [0034] 本发明实施例提供的技术方案带来的有益效果至少包括:
- [0035] 在本发明中,第一终端能够通过通讯线程,实现主线程与第二终端之间的数据传输,由于在通过通讯线程向第二终端传输数据时,没有基于套接字的执行逻辑,因此第一终端和第二终端可以不处于同一局域网内,使得第二终端可以在任何地点实现对第一终端的调试,大大提升了调试过程的便捷性,提升了第一终端上应用程序的调试效率。

附图说明

[0036] 为了更清楚地说明本发明实施例中的技术方案,下面将对实施例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。

[0037] 图1是本发明实施例提供的一种数据传输方法的实施环境示意图;

[0038] 图2是本发明实施例提供的一种数据传输方法的交互流程图;

[0039] 图3是本发明实施例提供的一种建立通信连接的原理性示意图;

[0040] 图4是本发明实施例提供的一种登录页面的示意图;

[0041] 图5是本发明实施例提供的一种调试页面的示意图;

[0042] 图6是本发明实施例提供的一种数据传输方法的原理性示意图;

[0043] 图7是本发明实施例提供的一种数据传输方法的交互流程图;

[0044] 图8是本发明实施例提供的一种调试页面的示意图;

[0045] 图9是本发明实施例提供的一种数据传输方法的交互流程图;

[0046] 图10是本发明实施例提供的一种调试页面的菜单栏的示意图;

[0047] 图11是本发明实施例提供的一种数据传输装置的结构示意图;

[0048] 图12示出了本发明一个示例性实施例提供的第一终端1200的结构框图。

具体实施方式

[0049] 为使本发明的目的、技术方案和优点更加清楚,下面将结合附图对本发明实施方式作进一步地详细描述。

[0050] 图1是本发明实施例提供的一种数据传输方法的实施环境示意图。参见图1,在该实施环境中,可以包括第一终端101,调试服务器102和第二终端103,下面进行详述:

[0051] 其中,该第一终端101可以为任一能够进行数据传输的电子设备,在该第一终端101上可以运行有主线程和通讯线程,该主线程可以用于执行任一脚本,例如该脚本可以是lua脚本,该通讯线程可以用于实现调试服务器与第一终端101之间的数据传输,使得当主线程运行报错时,主线程能够通过通讯线程实现与调试服务器之间的数据传输。

[0052] 其中,该调试服务器102可以为任一能够进行数据传输的计算机设备,调试服务器可以分别与第一终端101以及第二终端103建立通信连接,使得调试服务器102能够向第一终端101以及第二终端103传输数据。可选地,该调试服务器102可以是单机设备,也可以是集群设备,本发明实施例不对调试服务器102的设备类型进行具体限定。

[0053] 其中,该第二终端103可以为任一能够进行数据传输的电子设备,第二终端103与调试服务器102之间可以基于B/S(browser/server,浏览器/服务器)模式交互,也可以是基于C/S(client/server,客户端/服务器)模式交互,本发明实施例不对第二终端103与调试服务器102之间的交互模式进行具体限定,具体交互过程将在下述实施例中进行详述。

[0054] 示意性地,在一些数据传输场景中,第一终端101和第二终端103分别与调试服务器102签订通信协议,例如,该通信协议可以是TCP(transmission control protocol,传输控制协议)、UDP(user datagram protocol,用户数据报协议)、DHCP(dynamic host configuration protocol,动态主机设置协议)等,在签订通信协议之后,第一终端101和第

二终端103能够分别与调试服务器102进行基于通信协议的数据传输。

[0055] 需要说明的是,实际上对于同一个终端的实体而言,可能既是第二终端也是第一终端。另外,在上述过程中,第一终端101的数量可以是一个或多个,第二终端103的数量可以是一个或多个,调试服务器102也可以为单机设备或集群设备。

[0056] 图2是本发明实施例提供的一种数据传输方法的交互流程图。参见图2,该实施例包括:

[0057] 201、第一终端与调试服务器建立通信连接。

[0058] 其中,第一终端可以是任一能够进行数据传输的电子设备,在该第一终端上可以安装有待调试的应用程序(application,APP),在应用程序在运行时可以包括主线程,例如该应用程序可以是游戏应用程序、即时通讯应用程序、直播应用程序等。

[0059] 可选地,当该应用程序为lua脚本时,在应用程序的运行过程中,通常除了主线程之外还可以包括一个或多个子线程,该一个或多个子线程用于为应用程序提供多线程并行处理服务,而在主线程(或任一子线程)中还可以调用一个或多个协程(coroutine),该一个或多个协程用于为主线程(或任一子线程)提供辅助功能。

[0060] 当然,该应用程序也可以为其他脚本,例如该脚本类型可以为JSON(JavaScript object notation,java脚本对象简谱)、Python等,本发明实施例不对应用程序采用的脚本类型进行具体限定。

[0061] 可选地,在调试应用程序时,第一终端除了运行主线程以及一个或多个协程之外,还需要运行通讯线程,该通讯线程用于实现第一终端与调试服务器之间的通信。

[0062] 其中,该调试服务器可以是任一能够提供数据传输服务的计算机设备,该调试服务器可以作为第一终端与第二终端之间的数据中转点,也即是,在本发明实施例中第一终端和第二终端之间不直连,而是均通过访问调试服务器来获取传输的数据。

[0063] 在一些实施例中,第一终端在创建某一应用程序的主线程之后,主线程可以调用通讯线程,通过通讯线程与调试服务器之间建立通信连接,当检测到报错事件时,主线程向通讯线程发送报错事件的相关信息,通讯线程接收到该报错事件的相关信息之后,向调试服务器转发该报错事件的相关信息,进而实现第一终端与调试服务器之间的数据传输。可选地,在通讯线程与调试服务器之间的通信连接可以是无线连接、蓝牙连接等。

[0064] 在一些实施例中,第一终端在于调试服务器建立通信连接之后,调试服务器还可以在第二终端上注册钩子函数,从而当第二终端命中回调事件时,能够基于钩子函数回调通知到调试服务器。

[0065] 在上述情况中,例如,在第一终端启动某一应用程序的主线程时,需要对主线程进行初始化,此时遍历主线程的脚本上下文(例如lua脚本中的luastate)中所有的可收集对象(collectable objects),从而能够获取到所有可收集对象中可运行的所有协程,并初始化hook函数(一种钩子函数),当主线程新建任一个协程的脚本上下文时,协程的脚本上下文需要继承上述hook函数,从而实现事件回调。

[0066] 由于在相关技术中,第二终端与第二终端在基于套接字的执行逻辑进行数据传输时,第二终端仅能够对第一终端开启调试模式之后创建的协程进行调试,导致第二终端无法对开启调试模式之前已创建的协程进行调试,导致第二终端对调试时协程支持性差,而在本发明实施例中,由于每当主线程新建任一个协程的脚本上下文时,都从主线程的脚本

上下文中继承hook函数,因此本发明既能够支持对开启调试模式之后创建的协程进行调试,也能够支持对开启调试模式之前已创建的协程进行调试,因此本发明对协程的支持性高。

[0067] 202、第二终端与调试服务器建立通信连接。

[0068] 其中,该第二终端可以是任一能够进行数据传输的电子设备。可选地,该第二终端上可以安装有IDE(integrated development environment,集成开发环境)工具,便于用户基于IDE工具进行调试。

[0069] 图3是本发明实施例提供的一种建立通信连接的原理性示意图,参见图3,在一些实施例中,第二终端可以基于B/S模式与调试服务器建立通信连接,用户在第二终端的浏览器客户端中输入调试页面的URL,由于尚未通过验证,因此在浏览器客户端中展示的是登录页面,用户在该登录页面中输入账号和密码之后,第二终端生成携带该账号和密码的用户标识信息,将该用户标识信息发送至第一网关,其中,该第一网关为调试服务器的前置网关,由第一网关对该用户标识信息进行鉴权验证,当验证通过时,第一网关向第二终端返回客户端标识信息,在上述过程中相当于建立了第二终端与第一网关之间的通信连接。

[0070] 图4是本发明实施例提供的一种登录页面的示意图,如图4所示,用户除了在登录页面中提供账号密码登录的功能选项之外,还可以提供快速登录的功能选项,从而由浏览器客户端获取已登录的应用客户端所授权的账号信息,根据该账号信息生成用户标识信息,执行与上述过程中类似的与第一网关建立通信连接的步骤,这里不做赘述。

[0071] 当第二终端接收到第一网关返回的客户端标识信息之后,第二终端可以根据用户标识信息、客户端标识信息以及待调试的应用程序标识信息,生成访问请求,第二终端将该访问请求发送至第二网关,第二网关对该访问请求进行解析,对用户标识信息、客户端标识信息以及应用程序标识信息进行鉴权验证,当验证通过时,第二网关可以根据该访问请求生成验证令牌(token),在该验证令牌中可以携带时间戳(timestamp)、序列号、用户标识信息、客户端标识信息、应用程序标识信息、用户姓名或者数字签名中的至少一项,第二网关向调试网页的后台服务器发送该验证令牌,上述后台服务器对该验证令牌验证通过时,根据该验证令牌生成验证票据(ticket),该验证票据中也可以携带与验证令牌中相同的信息,该验证票据用于表示后台服务器和第二网关均对该第二终端验证通过,后台服务器向调试服务器发送该验证票据,当调试服务器接收到该验证票据时,调试服务器才通过第一网关向第二终端发送该应用程序的调试网页。

[0072] 在调试服务器向第二终端发送该调试网页之后,第二终端从登录页面跳转至该调试页面,在该调试页面中可以包括多个第一终端的调试选项,当第二终端检测到用户对任一第一终端的调试选项的触控操作时,第二终端从调试服务器处获取该第一终端的目标数据,在调试页面中显示该第一终端的目标数据,便于用户基于该目标数据进行调试。

[0073] 图5是本发明实施例提供的一种调试页面的示意图,第二终端可以显示如图5所示的调试页面,用户可以在该调试页面上部的输入栏中输入第一终端的ID(identification,标识),以进行第二终端的选择,当然,用户也可以点击输入栏右侧的下拉“箭头”后,显示多个第一终端的调试选项,当用户点击任一第一终端的调试选项之后,在调试页面的中部展示目标数据。

[0074] 在一些实施例中,第二终端还可以基于C/S模式与调试服务器建立通信连接,也即

是用户在第二终端的调试客户端上输入账号和密码之后,由调试客户端直接向调试服务器发送访问请求,调试服务器执行与上述B/S模式中类似的鉴权验证的步骤,当验证通过时第二终端与调试服务器建立通信连接,从而无需经过后台服务器的中转,简化了建立通信连接的流程,这里不做赘述。

[0075] 需要说明的是,本发明实施例仅以单个第二终端的调试过程为例进行说明,但在一些实施例中,该第二终端的数量还可以是多个,从而能够支持在下述过程中的多人同时协作调试,能够与项目组进行调试的场景更好地适配,提升了调试过程的协作效率,由于每个第二终端所执行的操作与本发明实施例中单个第二终端类似,这里不做赘述。

[0076] 203、第二终端向调试服务器发送对任一应用程序的调试开始指令。

[0077] 其中,该应用程序可以是第一终端上任一待调试的应用客户端,例如,该应用程序可以是游戏应用程序、即时通讯应用程序、直播应用程序等。

[0078] 其中,该调试开始指令用于表示对该应用程序开始进行调试,该调试开始指令中可以携带终端标识信息以及应用程序标识信息,例如,该调试开始指令可以是一个starthook函数。

[0079] 204、当调试服务器接收到该调试开始指令时,向第一终端的通讯线程发送该调试开始指令。

[0080] 在一些实施例中,调试服务器上可以安装有第一插件或第二插件,该第一插件用于管理各个第二终端发送的数据,而第二插件则用于管理各个第一终端发送的数据,第一插件和第二插件之间基于RPC(remote procedure call,远程过程调用)机制进行数据传输。需要说明的是,当调试服务器为包括多个节点设备的分布式集群时,该第一插件和第二插件通常可以在不同的节点设备上运行。

[0081] 图6是本发明实施例提供的一种数据传输方法的原理性示意图,参见图6,在上述过程中,调试服务器通过节点设备1的第一插件接收该调试开始指令,通过该第一插件对该调试开始指令进行解析,得到终端标识信息,从而由第一插件基于RPC机制向节点设备2的第二插件发送该终端标识信息,由第二插件将该调试开始指令转发至该终端标识信息所对应的第一终端的通讯线程。

[0082] 可选地,由于在实时的动态数据传输过程中,由于第一终端会源源不断的产生目标数据,而第二终端会源源不断的产生热更新文件(在下面的实施例中会进行详述),通常在调试服务器上还可以包括安装有存储插件的节点设备3,节点设备1上的第一插件可以通过RPC机制路由至该节点设备3的存储插件,节点设备2上的第二插件可以通过RPC机制路由至该节点设备3的存储插件,从而当节点设备1的存储空间小于第一目标阈值时,节点设备1的第一插件向节点设备3的存储插件发送热更新文件,当节点设备2的存储空间小于第二目标阈值时,节点设备2的第二插件向节点设备3的存储插件发送目标数据,从而由节点设备3按照应用程序标识信息来对各个热更新文件或目标数据进行持久化的存储,增强数据传输过程中的安全性。其中,该第一目标阈值或第二目标阈值为任一大于或等于1的数值。

[0083] 205、当接收到对第一终端上任一应用程序的调试开始指令时,第一终端通过通讯线程向该应用程序的主线程发送断点检测指令,该通讯线程用于实现不同局域网内终端之间的通信。

[0084] 其中,该断点检测指令用于指示应用程序的主线程进行断点检测,该断点是指在

一行一行地执行脚本的过程中由错误导致执行中断的代码行数。

[0085] 其中,该应用程序可以包括主线程和一个或多个子线程,在主线程或任一子线程中还可以调用一个或多个协程,当然,该应用程序中也可以不包括子线程,而是仅通过一个主线程来执行。

[0086] 可选地,该通讯线程可以是应用程序中的一个子线程,从而该通讯线程仅用于实现单个应用程序的主线程在跨局域网情况下的通信,也即是不同的应用程序可以具有不同的通讯线程。

[0087] 当然,该通讯线程还可以第一终端上的另一个独立的主线程,该通讯线程不随着应用程序的变化而发生改变,从而在第一终端上的所有应用程序均可以通过同一个通讯终端实现跨局域网情况下的通信。

[0088] 206、当主线程接收到该断点检测指令,主线程进行断点检测,根据检测到的断点生成目标数据,向通讯线程发送该目标数据。

[0089] 在上述过程中,主线程在接收到该断点检测指令后,切换至调试模式,在该调试模式下进行断点检测,该调试模式可以用于指示主线程一行一行的执行脚本,而当脚本的某一行中包含错误时,能够中断脚本的执行,从而主线程将执行中断的行数作为断点。

[0090] 当检测到断点后,可以从寄存器中获取主线程在执行断点前函数调用栈(call stack)中函数或变量的执行数据,基于该执行数据生成目标数据,例如,该执行数据中可以包括函数的调用关系、入栈出栈信息、函数返回值、变量类型等。

[0091] 207、第一终端通过通讯线程接收该主线程响应于该断点检测指令所生成的目标数据。

[0092] 在上述过程中,通讯线程可以接收主线程发送的任一数据报文,对该数据报文的目標字段进行检测,当该目标字段中携带目标数据标识时,将该数据报文确定为目标数据。例如,该目标字段可以是数据报文的包头的包头字段。

[0093] 208、每间隔目标时长,第一终端通过通讯线程,基于钩子函数向调试服务器发送该目标数据。

[0094] 其中,目标时长为任一大于或等于0的数值。

[0095] 在上述过程中,通讯线程可以每间隔目标时长,对钩子函数的回调事件进行检测,当命中回调事件时,基于该钩子函数向调试服务器发送该目标数据,例如,该回调事件可以是目标数据的缓存区不为空。

[0096] 在一些实施例中,通讯线程还可以在接收到目标数据之后,立即执行向调试服务器发送该目标数据的步骤,从而简化了通讯线程的处理逻辑。

[0097] 209、当调试服务器接收到该目标数据时,向第二终端发送该目标数据。

[0098] 在上述过程中,调试服务器可以基于上述步骤204中涉及的第一终端插件接收该目标数据,第一终端插件基于RPC机制将目标数据发送至第二终端插件,由第二终端插件向第二终端发送该目标数据。

[0099] 在上述步骤208-209中,该通讯线程向第二终端发送该目标数据,该第二终端与该第一终端不处于同一局域网内,由于第一终端通过调试服务器实现了与不同局域网内的第二终端之间的通信,能够使得调试操作不局限于局域网场景,拓展了调试过程的可移植性。

[0100] 在一些应用场景中,如果该应用程序为游戏应用程序,在更加普遍的场景下,玩家

对应的第一终端与技术人员对应的第二终端通常不是在同一个局域网内的,使用本发明实施例提供的数据传输方法,能够消除调试过程中,第一终端与第二终端需要位于同一局域网的网络限制,只需要第一终端和第二终端能够与调试服务器接入即可进行调试,使得调试过程更加方便快捷。

[0101] 210、当第二终端接收到该目标数据时,在该应用程序的调试页面中展示该目标数据。

[0102] 当第二终端接收该目标数据之后,第二终端可以对该目标数据进行渲染,从而实现了在调试页面中绘制出该目标数据,便于用户查看和编辑该目标数据。

[0103] 211、第二终端基于该调试页面生成调试指令,向调试服务器发送该调试指令,该调试指令与该目标数据对应。

[0104] 其中,该调试指令用于指示主线程的下一步调试动作的执行逻辑,例如,该调试指令可以用于指示主线程不执行某一个函数,或者该调试指令还可以用于指示主线程检测下一个断点,又或者该调试指令还可以用于指示主线程退出调试模式等,本发明实施例不对该调试指令的内容进行具体限定。上述步骤211与上述步骤203类似,这里不做赘述。

[0105] 212、当调试服务器接收到该调试指令时,向第一终端的通讯线程发送该调试指令。

[0106] 上述步骤212与上述步骤204类似,这里不做赘述。

[0107] 213、当接收到第二终端返回的调试指令时,每间隔目标时长,第一终端通过该通讯线程基于钩子函数向该主线程发送该第二终端返回的调试指令。

[0108] 上述步骤213中,当接收到第二终端返回的调试指令时,通讯线程向主线程发送调试指令,该调试指令与该目标数据对应,由于上述步骤213与上述步骤208类似,这里不做赘述。

[0109] 214、当该主线程接收到该调试指令时,基于该调试指令执行下一步调试动作。

[0110] 可选地,当该调试指令指示主线程不执行某一个函数时,主线程可以重新执行检测断点过程中执行的lua代码,从而试运行出是否是这个函数出现了错误,有利于帮助精准定位到产生的问题。

[0111] 可选地,当该调试指令指示主线程检测下一个断点时,主线程返回到步骤206,从上述断点的下一行开始执行lua代码,直到检测到下一个断点,执行与上述步骤207-214类似的操作,这里不做赘述。

[0112] 可选地,当该调试指令指示主线程退出调试模式时,主线程响应于该调试指令,退出调试模式。

[0113] 上述所有可选技术方案,可以采用任意结合形成本公开的可选实施例,在此不再一一赘述。

[0114] 本发明实施例提供的方法,第一终端能够通过通讯线程,实现主线程与第二终端之间的数据传输,由于在通过通讯线程向第二终端传输数据时,没有基于套接字的执行逻辑,因此第一终端和第二终端可以不处于同一局域网内,使得第二终端可以在任何地点实现对第一终端的调试,大大提升了调试过程的便捷性,提升了第一终端上应用程序的调试效率。

[0115] 进一步地,由于在相关技术中基于套接字的执行逻辑进行数据传输,导致数据传

输过程严重依赖于套接字所在的第三方库,而在本发明实施例中基于通讯线程进行数据传输,能够大大降低对第三方库的外部依赖。

[0116] 进一步地,由于在相关技术中在创建套接字之后,需要第一终端和第二终端互相知道对方的内网IP地址才可以建立通信连接,而在本发明实施例中,双方分别与调试服务器建立通信连接,第一终端无需获取第二终端的内网IP地址,第二终端也无需获取第一终端的内网IP地址,从而提升了调试操作的可拓展性。

[0117] 进一步地,由于在相关技术中不通过通讯线程进行数据传输,所以数据解析是由主线程完成的,导致第二终端对主线程的侵入性强,降低了主线程的执行效率,而在本发明实施例中在第一终端上开设了专用的通讯线程,将数据解析、压缩、打包等工作转移至通讯线程上执行,大大降低了对主线程的侵入性,提升了主线程的执行效率。

[0118] 在上述过程中,提供了第一终端向第二终端发送目标数据,第二终端向第一终端返回调试指令的数据传输的过程,实现了不同局域网内终端在文件级别的通信,基于上述实施例,在执行上述步骤205之后,为了提升调试效率,用户还可以通过执行本发明实施例中的方法,在第二终端上查看第一终端的日志文件,下面进行详述。

[0119] 图7是本发明实施例提供的一种数据传输方法的交互流程图,参见图7,该实施例包括下述步骤:

[0120] 701、第二终端向调试服务器发送对任一应用程序的日志拉取指令。

[0121] 其中,该日志拉取指令用于拉取该应用程序的日志文件。

[0122] 图8是本发明实施例提供的一种调试页面的示意图,参见图8,第二终端可以在调试过程中的任一时刻点击菜单栏中的“日志拉取”功能选项,从而当第二终端检测到用户对“日志拉取”功能选项的触控操作时,生成日志拉取指令,可选地,该日志拉取指令中可以携带待拉取日志的起始时间戳和终止时间戳,从而能够对指定时间段内的日志文件进行拉取,提高了日志文件拉取过程的便捷性。

[0123] 上述步骤701与上述步骤203类似,这里不做赘述。

[0124] 702、当调试服务器接收到该日志拉取指令时,向第一终端的通讯线程发送该日志拉取指令。

[0125] 上述步骤702与上述步骤204类似,这里不做赘述。

[0126] 703、当接收到日志拉取指令时,第一终端通过该通讯线程向该主线程发送该日志拉取指令。

[0127] 上述步骤703与上述步骤205类似,这里不做赘述。

[0128] 704、当主线程接收到该日志拉取指令时,主线程获取日志文件,向通讯线程发送该日志文件。

[0129] 上述步骤704与上述步骤206类似,这里不做赘述。

[0130] 705、第一终端通过该通信线程接收该主线程响应于该日志拉取指令所返回的日志文件。

[0131] 上述步骤705与上述步骤207类似,这里不做赘述。

[0132] 706、每间隔目标时长,第一终端通过该通信线程,基于钩子函数向调试服务器发送该日志文件。

[0133] 上述步骤706与上述步骤208类似,这里不做赘述。

- [0134] 707、当接收到该日志文件时,调试服务器向第二终端发送该日志文件。
- [0135] 上述步骤707与上述步骤209类似,这里不做赘述。
- [0136] 在上述步骤706-707中,通讯线程向第二终端发送日志文件,能够便于用户查看日志文件,并基于日志文件进行调试。
- [0137] 708、第二终端接收该日志文件。
- [0138] 上述步骤708与上述步骤210中的接收过程类似,这里不做赘述。
- [0139] 可选地,第二终端接收该日志文件之后,可以在调试页面中展示该日志文件。
- [0140] 在一些应用场景中,以应用程序为游戏应用程序为例进行说明,由于第一终端通常是玩家所对应的设备,第二终端通常是技术人员所对应的设备,而由于不同的第一终端的型号、操作系统、处理能力的不同,使得第一终端的调试环境趋于复杂多样,在相关技术中,若想查看某一游戏应用程序的日志文件,通常需要第二终端通过USB(universal serial bus,通用串行总线)连接至第一终端,借助第二终端上外部的第三方软件来进行日志文件的查看,导致查看日志文件的操作繁琐,并且必须要第一终端与第二终端进行USB线的连接。
- [0141] 在本发明实施例中,第一终端与第二终端无需“碰面”,两者均通过调试服务器进行日志文件的传输,通过调试服务器来保证数据传输过程的安全性,而由于应用程序的日志文件能够反映出应用程序的主线程运行时的工作记录,因此用户能够结合日志文件和目标数据进行调试,为调试过程提供更多的参考信息,便于用户快速定位到问题,大大降低了调试成本,提升了调试效率。
- [0142] 在上述过程中,提供了第一终端与第二终端之间传输日志文件的过程,实现了不同局域网内终端的通信,基于上述实施例,在执行上述步骤210之后,为了提升调试效率,用户还可以通过执行本发明实施例中的方法,实现第一终端与第二终端之间的动态链接库的传输,下面进行详述。
- [0143] 图9是本发明实施例提供的一种数据传输方法的交互流程图,参见图9,该实施例包括下述步骤:
- [0144] 901、第二终端向调试服务器发送对任一应用程序的热更新文件。
- [0145] 在通过多次重复执行上述步骤206-214之后,用户能够定位出应用程序的主线程中所存在至少一个问题,从而用户可以针对上述至少一个问题,在调试页中编辑用于修正问题的目标代码,当用户编辑完成后,可以点击调试页面中的“热更新(hotfix)”功能选项,触发第二终端将上述目标代码封装为热更新文件,向调试服务器发送该热更新文件。
- [0146] 在一些实施例中,第二终端可以将该目标代码封装为热更新文件,从而当第一终端的主线程接收到该热更新文件时,能够基于该热更新文件进行热更新,热更新的过程将在下述步骤904中进行详述。
- [0147] 图10是本发明实施例提供的一种调试页面的菜单栏的示意图,参见图10,例如,用户在第二终端完成调试之后,可以点击该调试页面的菜单栏中的“热更新(hotfix)”功能选项,当第二终端检测到对“热更新(hotfix)”功能选项的触控操作时,将调试指令封装为热更新文件,从而将该热更新文件发送至调试服务器,执行下述步骤902。
- [0148] 902、当调试服务器接收到该热更新文件时,向第一终端的通讯线程发送该热更新文件。

[0149] 上述步骤902与上述步骤204类似,这里不做赘述。

[0150] 903、当接收到热更新文件时,第一终端通过该通讯线程向应用程序的主线程发送该热更新文件。

[0151] 上述步骤903与上述步骤205类似,这里不做赘述。

[0152] 904、当该主线程接收到该热更新文件时,第一终端基于该热更新文件对该应用程序进行热更新。

[0153] 在上述进行热更新的过程中,第一终端可以执行下述步骤:将该热更新文件复制至应用程序的运行目录;当检测到该应用程序处于启动状态时,从该运行目录中加载该热更新文件。

[0154] 需要说明的是,在Windows系统下,热更新文件通常为DLL(dynamic link library,动态链接库)文件,DLL文件是一种PE格式(windows preinstallation environment,Windows预安装环境)的文件,因此在DLL文件头中会写入该文件所引用的DLL文件所在的位置符号,PE文件加载器从DLL文件头中查找引用的位置符号表,从而根据该位置符号表加载该DLL文件引用的其他DLL文件,因此在Windows系统下提供DLL文件时通常还需要提供一个链接库(lib),并且DLL文件通常对其他的DLL文件具有运行依赖。

[0155] 而在Unix系统下,热更新文件通常表现为SO文件,SO文件是一种elf格式(executable and linkable format,可执行与可链接格式)的文件,SO文件无需在文件头中声明所引用的SO文件所在的位置符号,而是在执行SO文件时,由调用dlopen函数(一个用于打开热更新文件的函数)的进程提供该SO文件所引用的其他SO文件。

[0156] 有鉴于上述Windows系统和Unix系统下处理热更新文件的区别,由于Unix系统下的lua解释器能够通过上述dlopen函数,实现某一SO文件对任一lua API(application programming interface,应用程序编程接口)的连接,因此,在本发明中第二终端可以向配置有Unix系统的第一终端提供动态链接库SO文件,使得主线程可以方便地加载该SO文件。

[0157] 可选地,对于配置有Windows系统的第一终端而言,第二终端可以自定义使用到的lua函数指针,通过在Windows系统下遍历所有的组件(module),可以通过lua函数指针,重定向到lua脚本的位置符号,从而实现了动态查找lua符号的策略,使得第二终端提供的热更新文件对lua内核(luacore)没有外部依赖,避免了在运行时出现multiple lua vms detected的异常。

[0158] 另外,第二终端不仅能够向第一终端提供热更新文件,还可以向第一终端提供静态链接库文件(通常为.c文件),对于静态链接库文件而言,第一终端直接将静态链接库文件加入编译器即可实现加载。

[0159] 由于套接字的传输逻辑下不支持文件级别的数据传输,因此在相关技术中主线程无法进行热更新,而在本发明实施例提供的方法中,第二终端和第一终端之间能够通过调试服务器直接进行文件级别的数据传输,当调试指令为热更新文件时,主线程能够通过该热更新文件实现热更新。

[0160] 进一步地,在相关技术中,lua套接字必须运行在lua虚拟机上,因此传输的调试指令必须是lua代码编写的数据包,由于本发明实施例中的热更新文件对lua内核没有外部依赖,因此该热更新文件可以由非lua代码编写,例如该热更新文件可以由C/C++代码编写,提升了用户编辑调试指令的效率。

[0161] 图11是本发明实施例提供的一种数据传输装置的结构示意图,参见图11,该装置包括:

[0162] 发送模块1101,用于当接收到对第一终端上任一应用程序的调试开始指令时,通过该第一终端的通讯线程向该应用程序的主线程发送断点检测指令,该通讯线程用于实现不同局域网内终端之间的通信;

[0163] 接收模块1102,用于通过该通讯线程接收该主线程响应于该断点检测指令所生成的目标数据;

[0164] 该发送模块1101,还用于通过该通讯线程向第二终端发送该目标数据,该第二终端与该第一终端不处于同一局域网内;

[0165] 该发送模块1101,还用于当接收到该第二终端返回的调试指令时,通过该通讯线程向该主线程发送该调试指令,该调试指令与该目标数据对应。

[0166] 本发明实施例提供的装置中,第一终端能够通过通讯线程,实现主线程与第二终端之间的数据传输,由于在通过通讯线程向第二终端传输数据时,没有基于套接字的执行逻辑,因此第一终端和第二终端可以不处于同一局域网内,使得第二终端可以在任何地点实现对第一终端的调试,大大提升了调试过程的便捷性,提升了第一终端上应用程序的调试效率。

[0167] 在一种可能实施方式中,基于图11的装置组成,该发送模块1101包括:

[0168] 发送单元,用于通过该通讯线程向调试服务器发送该目标数据,由该调试服务器向该第二终端发送该目标数据。

[0169] 在一种可能实施方式中,该发送单元用于:

[0170] 每间隔目标时长,通过该通讯线程向该调试服务器发送该目标数据。

[0171] 在一种可能实施方式中,该发送单元用于:

[0172] 通过该通讯线程,基于钩子函数向该调试服务器发送该目标数据。

[0173] 在一种可能实施方式中,基于图11的装置组成,该装置还包括:

[0174] 该发送模块1101,还用于当接收到热更新文件时,通过该通讯线程向该主线程发送该热更新文件;

[0175] 热更新模块,用于当该主线程接收到该热更新文件时,基于该热更新文件对该应用程序进行热更新。

[0176] 在一种可能实施方式中,该热更新模块用于:

[0177] 将该热更新文件复制至应用程序的运行目录;

[0178] 当检测到该应用程序处于启动状态时,从该运行目录中加载该热更新文件。

[0179] 在一种可能实施方式中,基于图11的装置组成,该装置还包括:

[0180] 当接收到日志拉取指令时,通过该通讯线程向该主线程发送该日志拉取指令,该日志拉取指令用于拉取该应用程序的日志文件;

[0181] 通过该通信线程接收该主线程响应于该日志拉取指令所返回的日志文件;

[0182] 通过该通信线程向该第二终端发送该日志文件。

[0183] 需要说明的是:上述实施例提供的数据传输装置在传输数据时,仅以上述各功能模块的划分进行举例说明,实际应用中,可以根据需要而将上述功能分配由不同的功能模块完成,即将第一终端的内部结构划分成不同的功能模块,以完成以上描述的全部或者部

分功能。另外,上述实施例提供的数据传输装置与数据传输方法实施例属于同一构思,其具体实现过程详见数据传输方法实施例,这里不再赘述。

[0184] 图12示出了本发明一个示例性实施例提供的第一终端1200的结构框图。该第一终端1200可以是:智能手机、平板电脑、MP3播放器(Moving Picture Experts Group Audio Layer III,动态影像专家压缩标准音频层面3)、MP4(Moving Picture Experts Group Audio Layer IV,动态影像专家压缩标准音频层面4)播放器、笔记本电脑或台式电脑。第一终端1200还可能被称为用户设备、便携式终端、膝上型终端、台式终端等其他名称。

[0185] 通常,第一终端1200包括有:处理器1201和存储器1202。

[0186] 处理器1201可以包括一个或多个处理核心,比如4核心处理器、8核心处理器等。处理器1201可以采用DSP(Digital Signal Processing,数字信号处理)、FPGA(Field-Programmable Gate Array,现场可编程门阵列)、PLA(Programmable Logic Array,可编程逻辑阵列)中的至少一种硬件形式来实现。处理器1201也可以包括主处理器和协处理器,主处理器是用于对在唤醒状态下的数据进行处理的处理器,也称CPU(Central Processing Unit,中央处理器);协处理器是用于对在待机状态下的数据进行处理的低功耗处理器。在一些实施例中,处理器1201可以在集成有GPU(Graphics Processing Unit,图像处理器),GPU用于负责显示屏所需要显示的内容的渲染和绘制。一些实施例中,处理器1201还可以包括AI(Artificial Intelligence,人工智能)处理器,该AI处理器用于处理有关机器学习的计算操作。

[0187] 存储器1202可以包括一个或多个计算机可读存储介质,该计算机可读存储介质可以是非暂态的。存储器1202还可包括高速随机存取存储器,以及非易失性存储器,比如一个或多个磁盘存储设备、闪存存储设备。在一些实施例中,存储器1202中的非暂态的计算机可读存储介质用于存储至少一个指令,该至少一个指令用于被处理器1201所执行以实现本申请中数据传输方法实施例提供的数据传输方法。

[0188] 在一些实施例中,第一终端1200还可选包括有:外围设备接口1203和至少一个外围设备。处理器1201、存储器1202和外围设备接口1203之间可以通过总线或信号线相连。各个外围设备可以通过总线、信号线或电路板与外围设备接口1203相连。具体地,外围设备包括:射频电路1204、触摸显示屏1205、摄像头1206、音频电路1207和电源1209中的至少一种。

[0189] 外围设备接口1203可被用于将I/O(Input/Output,输入/输出)相关的至少一个外围设备连接到处理器1201和存储器1202。在一些实施例中,处理器1201、存储器1202和外围设备接口1203被集成在同一芯片或电路板上;在一些其他实施例中,处理器1201、存储器1202和外围设备接口1203中的任意一个或两个可以在单独的芯片或电路板上实现,本实施例对此不加以限定。

[0190] 射频电路1204用于接收和发射RF(Radio Frequency,射频)信号,也称电磁信号。射频电路1204通过电磁信号与通信网络以及其他通信设备进行通信。射频电路1204将电信号转换为电磁信号进行发送,或者,将接收到的电磁信号转换为电信号。可选地,射频电路1204包括:天线系统、RF收发器、一个或多个放大器、调谐器、振荡器、数字信号处理器、编解码芯片组、用户身份模块卡等等。射频电路1204可以通过至少一种无线通信协议来与其它终端进行通信。该无线通信协议包括但不限于:城域网、各代移动通信网络(2G、3G、4G及5G)、无线局域网和/或WiFi(Wireless Fidelity,无线保真)网络。在一些实施例中,射频电

路1204还可以包括NFC (Near Field Communication, 近距离无线通信) 有关的电路, 本申请对此不加以限定。

[0191] 显示屏1205用于显示UI (User Interface, 用户界面)。该UI可以包括图形、文本、图标、视频及其它们的任意组合。当显示屏1205是触摸显示屏时, 显示屏1205还具有采集在显示屏1205的表面或表面上方的触摸信号的能力。该触摸信号可以作为控制信号输入至处理器1201进行处理。此时, 显示屏1205还可以用于提供虚拟按钮和/或虚拟键盘, 也称软按钮和/或软键盘。在一些实施例中, 显示屏1205可以为一个, 设置第一终端1200的前面板; 在另一些实施例中, 显示屏1205可以为至少两个, 分别设置在第一终端1200的不同表面或呈折叠设计; 在再一些实施例中, 显示屏1205可以是柔性显示屏, 设置在第一终端1200的弯曲表面上或折叠面上。甚至, 显示屏1205还可以设置成非矩形的不规则图形, 也即异形屏。显示屏1205可以采用LCD (Liquid Crystal Display, 液晶显示屏)、OLED (Organic Light-Emitting Diode, 有机发光二极管) 等材质制备。

[0192] 摄像头组件1206用于采集图像或视频。可选地, 摄像头组件1206包括前置摄像头和后置摄像头。通常, 前置摄像头设置在终端的前面板, 后置摄像头设置在终端的背面。在一些实施例中, 后置摄像头为至少两个, 分别为主摄像头、景深摄像头、广角摄像头、长焦摄像头中的任意一种, 以实现主摄像头和景深摄像头融合实现背景虚化功能、主摄像头和广角摄像头融合实现全景拍摄以及VR (Virtual Reality, 虚拟现实) 拍摄功能或者其它融合拍摄功能。在一些实施例中, 摄像头组件1206还可以包括闪光灯。闪光灯可以是单色温闪光灯, 也可以是双色温闪光灯。双色温闪光灯是指暖光闪光灯和冷光闪光灯的组合, 可以用于不同色温下的光线补偿。

[0193] 音频电路1207可以包括麦克风和扬声器。麦克风用于采集用户及环境的声波, 并将声波转换为电信号输入至处理器1201进行处理, 或者输入至射频电路1204以实现语音通信。出于立体声采集或降噪的目的, 麦克风可以为多个, 分别设置在第一终端1200的不同部位。麦克风还可以是阵列麦克风或全向采集型麦克风。扬声器则用于将来自处理器1201或射频电路1204的电信号转换为声波。扬声器可以是传统的薄膜扬声器, 也可以是压电陶瓷扬声器。当扬声器是压电陶瓷扬声器时, 不仅可以将电信号转换为人类可听见的声波, 也可以将电信号转换为人类听不见的声波以进行测距等用途。在一些实施例中, 音频电路1207还可以包括耳机插孔。

[0194] 电源1209用于为第一终端1200中的各个组件进行供电。电源1209可以是交流电、直流电、一次性电池或可充电电池。当电源1209包括可充电电池时, 该可充电电池可以支持有线充电或无线充电。该可充电电池还可以用于支持快充技术。

[0195] 在一些实施例中, 第一终端1200还包括有一个或多个传感器1210。该一个或多个传感器1210包括但不限于: 加速度传感器1211、陀螺仪传感器1212、压力传感器1213、光学传感器1215以及接近传感器1216。

[0196] 加速度传感器1211可以检测以第一终端1200建立的坐标系的三个坐标轴上的加速度大小。比如, 加速度传感器1211可以用于检测重力加速度在三个坐标轴上的分量。处理器1201可以根据加速度传感器1211采集的重力加速度信号, 控制触摸显示屏1205以横向视图或纵向视图进行用户界面的显示。加速度传感器1211还可以用于游戏或者用户的运动数据的采集。

[0197] 陀螺仪传感器1212可以检测第一终端1200的机体方向及转动角度,陀螺仪传感器1212可以与加速度传感器1211协同采集用户对第一终端1200的3D动作。处理器1201根据陀螺仪传感器1212采集的数据,可以实现如下功能:动作感应(比如根据用户的倾斜操作来改变UI)、拍摄时的图像稳定、游戏控制以及惯性导航。

[0198] 压力传感器1213可以设置在第一终端1200的侧边框和/或触摸显示屏1205的下层。当压力传感器1213设置在第一终端1200的侧边框时,可以检测用户对第一终端1200的握持信号,由处理器1201根据压力传感器1213采集的握持信号进行左右手识别或快捷操作。当压力传感器1213设置在触摸显示屏1205的下层时,由处理器1201根据用户对触摸显示屏1205的压力操作,实现对UI界面上的可操作性控件进行控制。可操作性控件包括按钮控件、滚动条控件、图标控件、菜单控件中的至少一种。

[0199] 光学传感器1215用于采集环境光强度。在一个实施例中,处理器1201可以根据光学传感器1215采集的环境光强度,控制触摸显示屏1205的显示亮度。具体地,当环境光强度较高时,调高触摸显示屏1205的显示亮度;当环境光强度较低时,调低触摸显示屏1205的显示亮度。在另一个实施例中,处理器1201还可以根据光学传感器1215采集的环境光强度,动态调整摄像头组件1206的拍摄参数。

[0200] 接近传感器1216,也称距离传感器,通常设置在第一终端1200的前面板。接近传感器1216用于采集用户与第一终端1200的正面之间的距离。在一个实施例中,当接近传感器1216检测到用户与第一终端1200的正面之间的距离逐渐变小时,由处理器1201控制触摸显示屏1205从亮屏状态切换为息屏状态;当接近传感器1216检测到用户与第一终端1200的正面之间的距离逐渐变大时,由处理器1201控制触摸显示屏1205从息屏状态切换为亮屏状态。

[0201] 本领域技术人员可以理解,图12中示出的结构并不构成对第一终端1200的限定,可以包括比图示更多或更少的组件,或者组合某些组件,或者采用不同的组件布置。

[0202] 在示例性实施例中,还提供了一种计算机可读存储介质,例如包括至少一条指令的存储器,上述至少一条指令可由终端中的处理器执行以完成上述实施例中数据传输方法。例如,该计算机可读存储介质可以是ROM、随机存取存储器(RAM)、CD-ROM、磁带、软盘和光数据存储设备等。

[0203] 本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成,也可以通过程序来指令相关的硬件完成,该程序可以存储于一种计算机可读存储介质中,上述提到的存储介质可以是只读存储器,磁盘或光盘等。

[0204] 以上所述仅为本发明的较佳实施例,并不用以限制本发明,凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

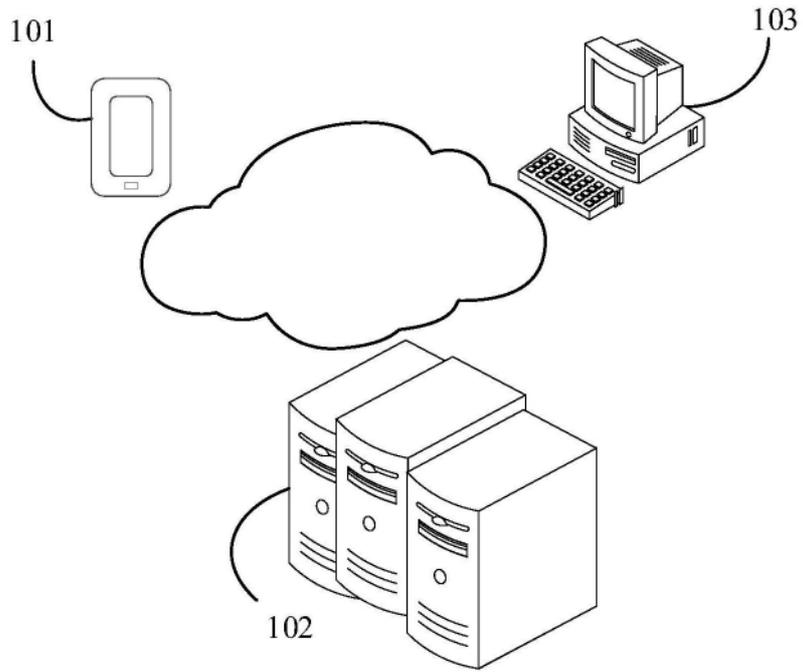


图1

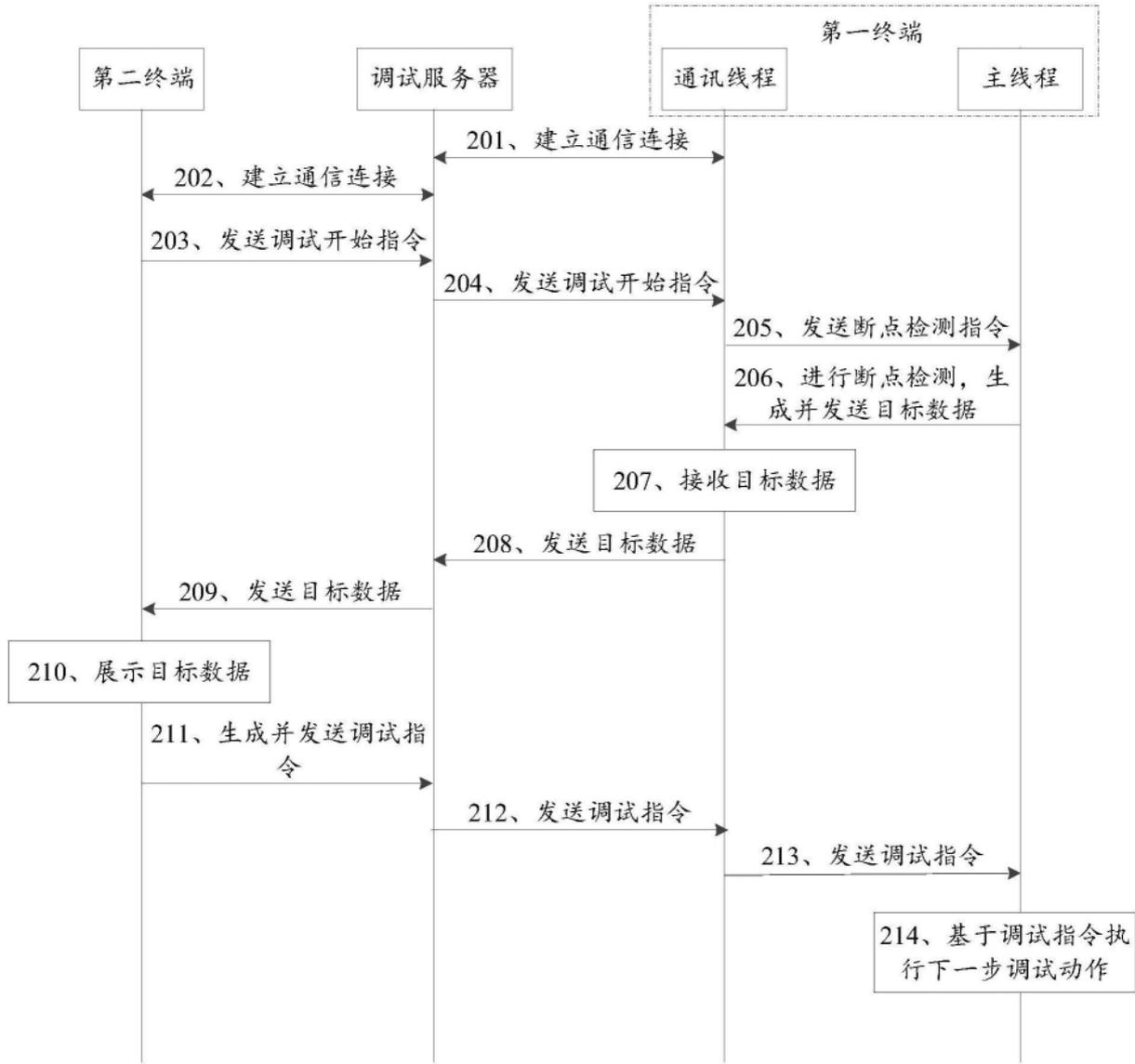


图2

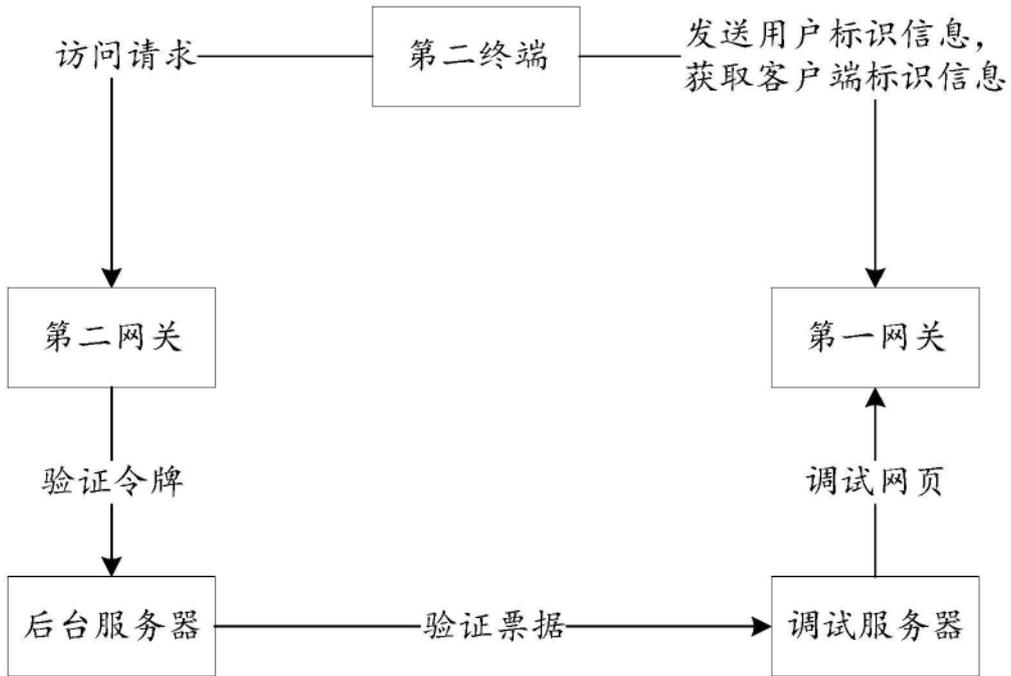


图3

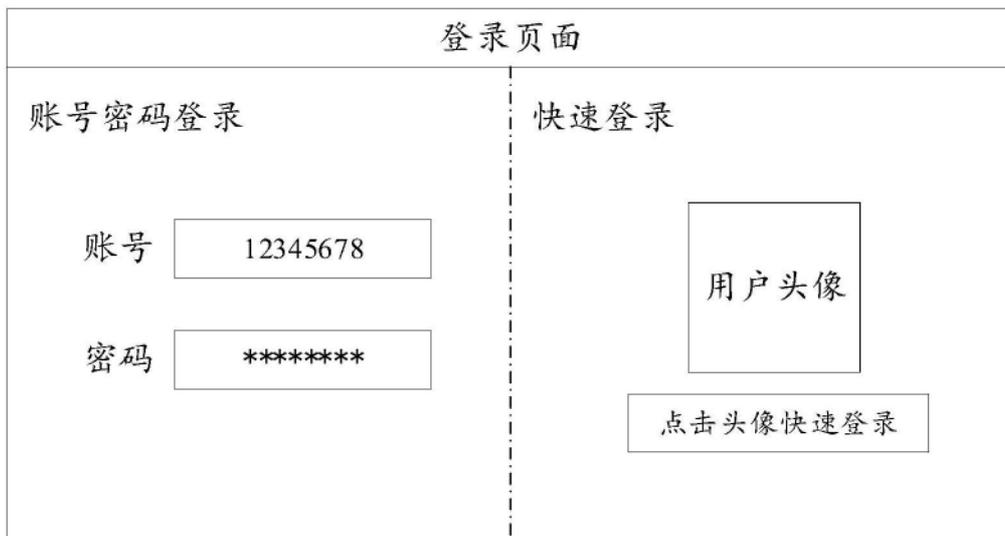


图4

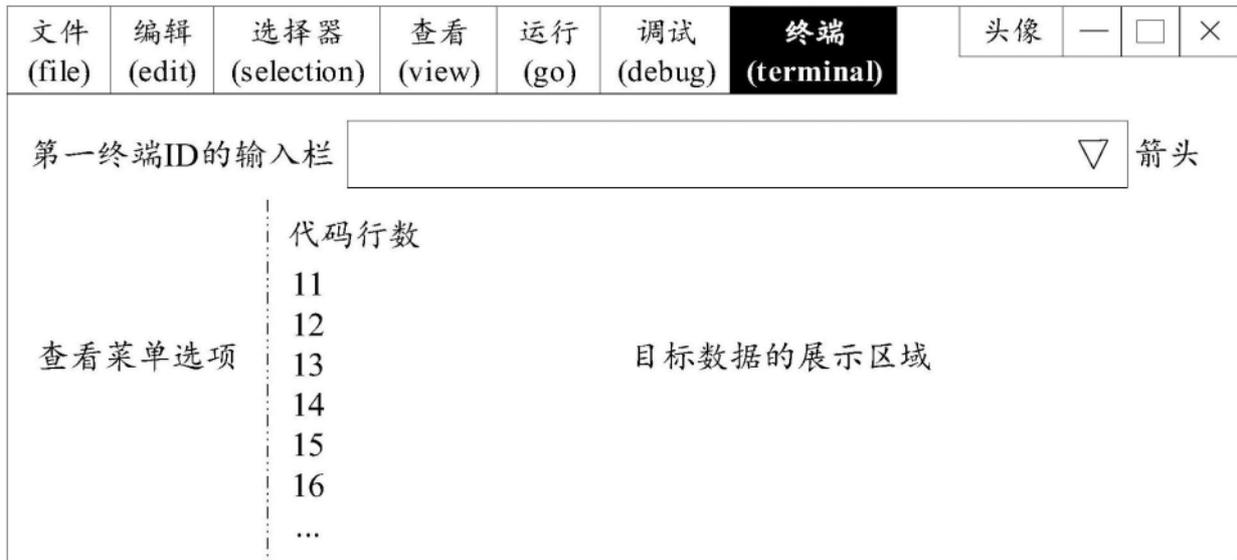


图5

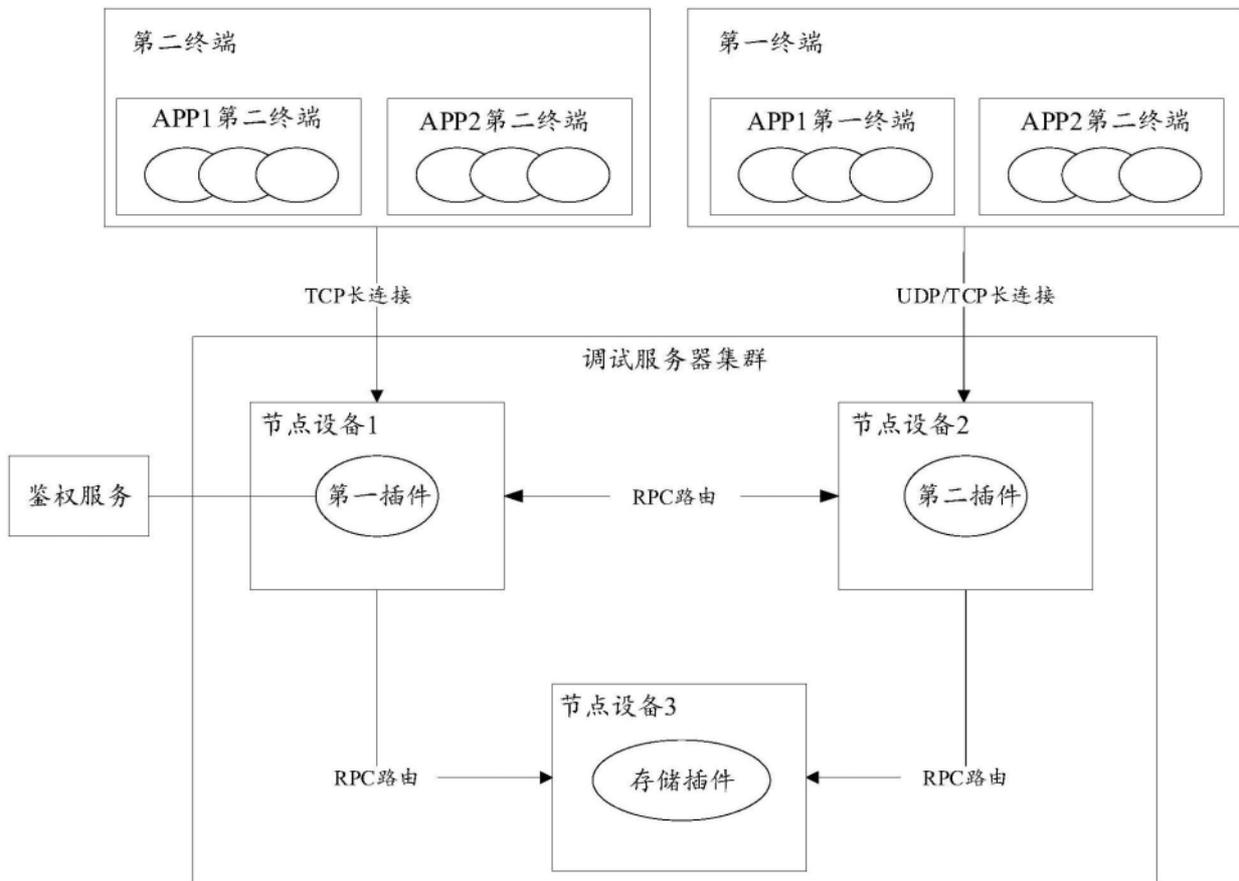


图6

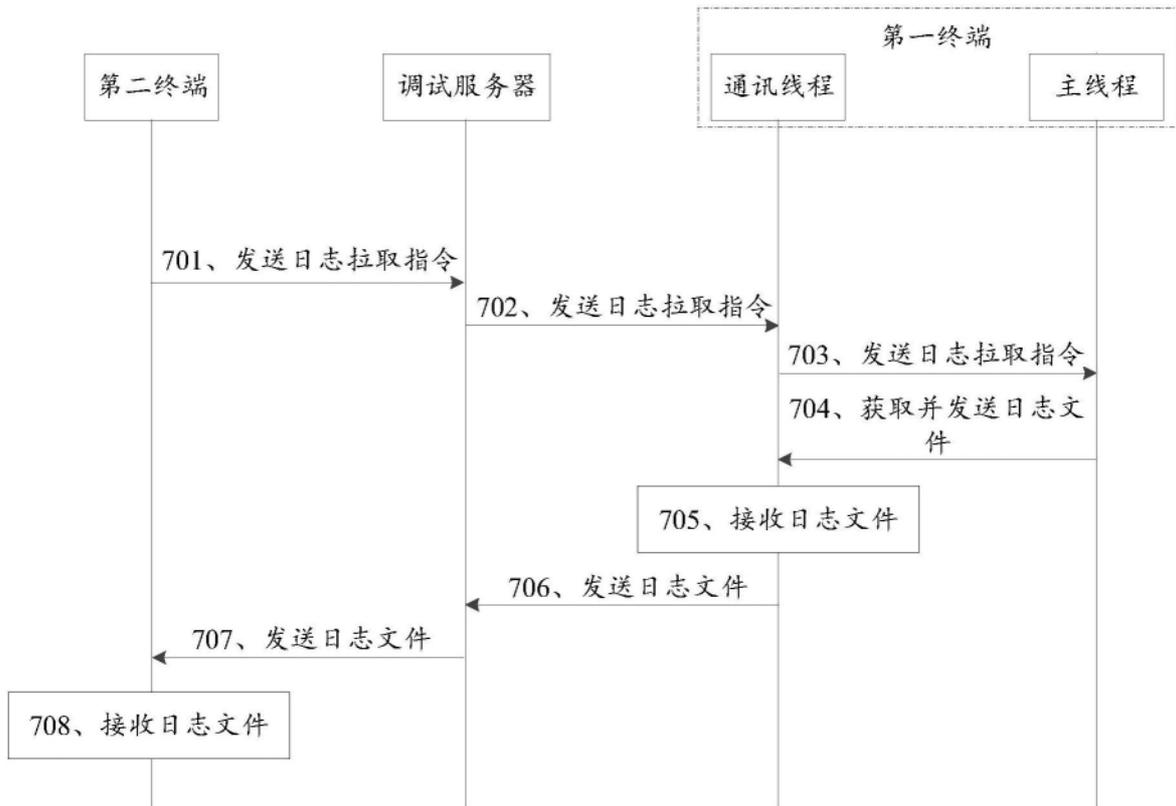


图7

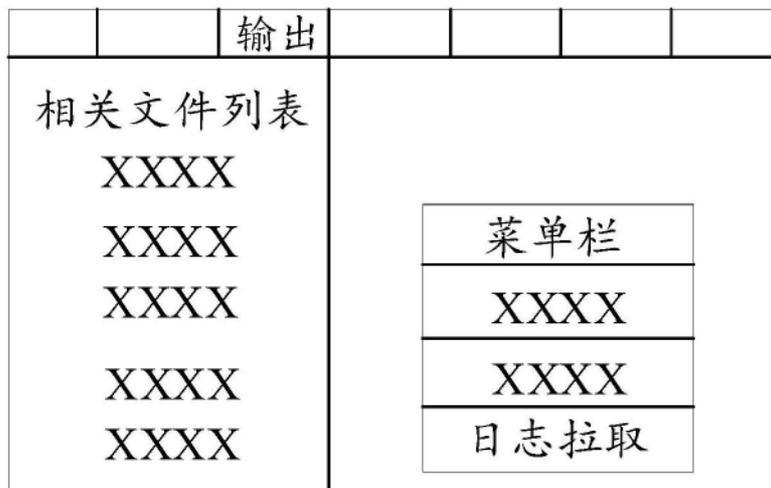


图8

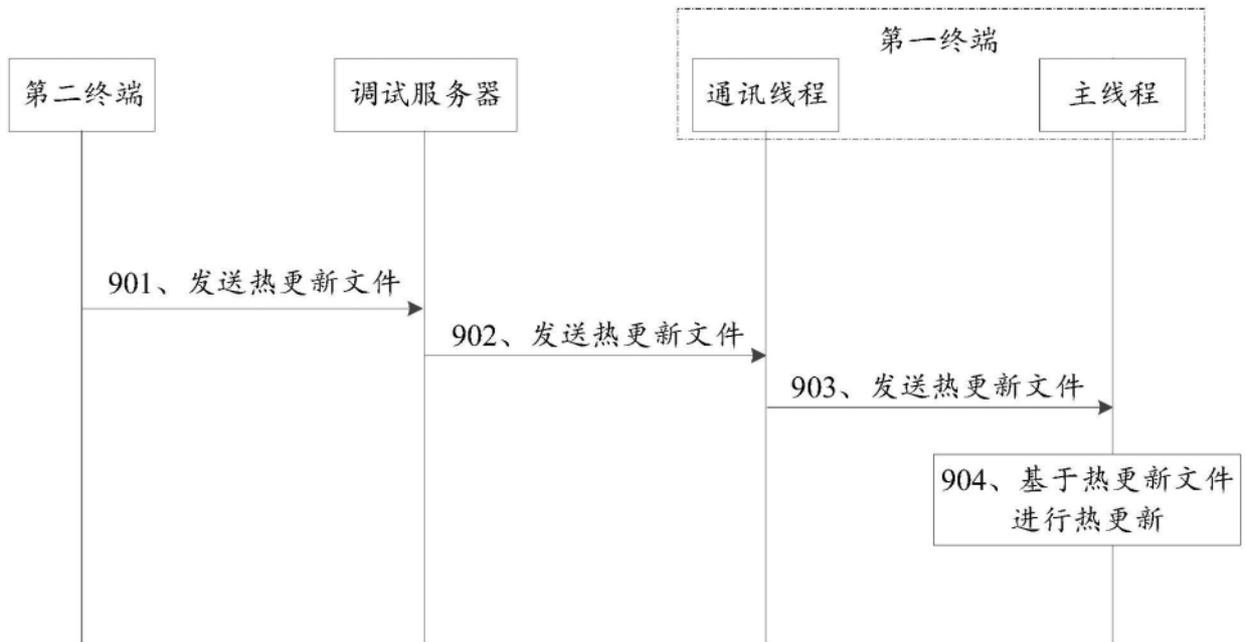


图9



图10

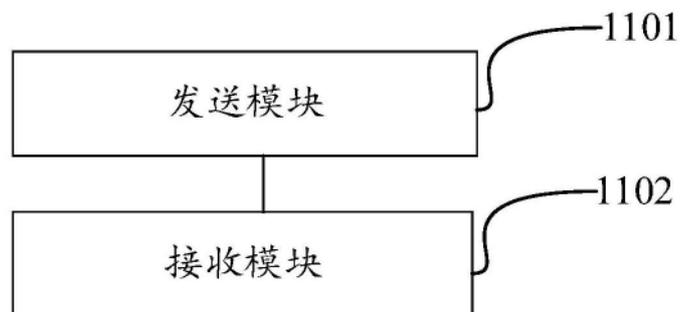


图11

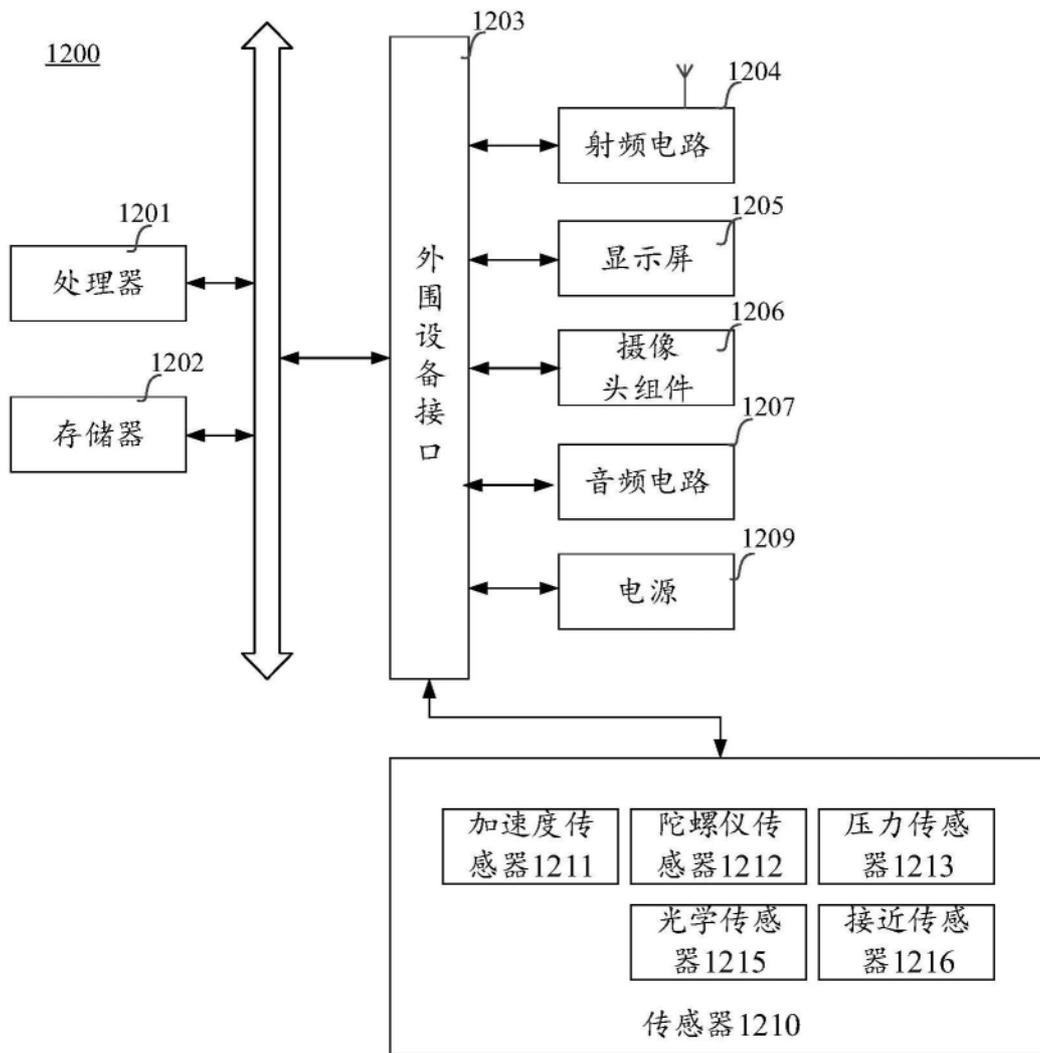


图12