

(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
2019 年 3 月 21 日 (21.03.2019)



(10) 国际公布号
WO 2019/051944 A1

(51) 国际专利分类号:
G06F 9/44 (2018.01) **G06F 21/54** (2013.01)
G06F 9/48 (2006.01)

(21) 国际申请号: PCT/CN2017/108238

(22) 国际申请日: 2017 年 10 月 30 日 (30.10.2017)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201710827274.1 2017 年 9 月 14 日 (14.09.2017) CN

(71) 申请人: 平安科技 (深圳) 有限公司 (PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN];
中国广东省深圳市福田区八卦岭八卦三路平安大厦, Guangdong 518000 (CN)。

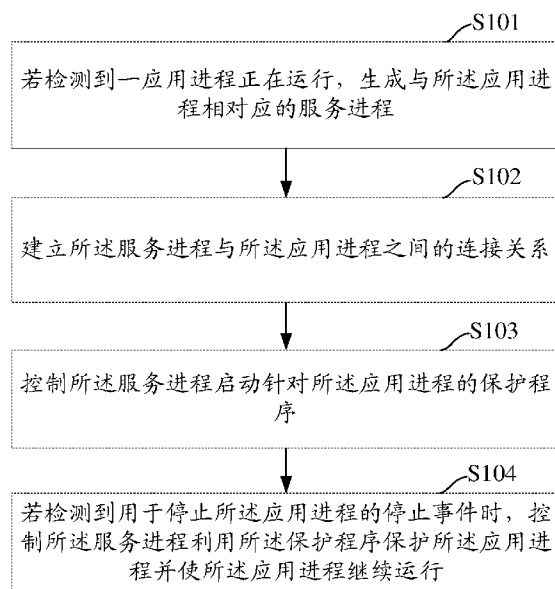
(72) 发明人: 杨文 (YANG, Wen); 中国广东省深圳市福田区八卦岭八卦三路平安大厦, Guangdong 518000 (CN)。

(74) 代理人: 深圳市精英专利事务所 (SHENZHEN TALENT PATENT SERVICE); 中国广东省深圳市福田区深南中路 6009 号绿景广场 B 栋 20 层 B, Guangdong 518000 (CN)。

(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, IR, IS, 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,

(54) **Title:** PROCESS PROTECTION METHOD, APPARATUS AND DEVICE, AND COMPUTER-READABLE STORAGE MEDIUM

(54) 发明名称: 一种进程保护方法、装置、设备以及计算机可读存储介质



S101 If it is detected that an application process is running, generate a service process corresponding to the application process
S102 Establish a connection relationship between the service process and the application process
S103 Control the service process so same starts a protection procedure for the application process
S104 When a stopping event used for stopping the application process is detected, control the service process so same uses the protection procedure to protect the application process and enable the application process to continue to run

图 1

(57) **Abstract:** Disclosed are a process protection method, apparatus and device, and a computer-readable storage medium, wherein the method comprises: if it is detected that an application process is running, generating a service process corresponding to the application process (S101); establishing a connection relationship between the service process and the application process (S102); controlling the service process so same starts a protection procedure for the application process (S103); and when a stopping event used for stopping the application process is detected, controlling the service process so same uses the protection procedure to protect the application process and enable the application process to continue to run (S104). The method can increase the efficiency of protecting an application process that a user wishes to run, improving the usage experience of a user.

SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG,
US, UZ, VC, VN, 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)。

本国际公布:

- 包括国际检索报告 (条约第21条(3))。

(57) 摘要: 一种进程保护方法、装置、设备以及计算机可读存储介质, 其中, 所述方法包括: 若检测到一应用进程正在运行, 生成与所述应用进程相对应的服务进程 (S101); 建立所述服务进程与所述应用进程之间的连接关系 (S102); 控制所述服务进程启动针对所述应用进程的保护程序 (S103); 若检测到用于停止所述应用进程的停止事件时, 控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行 (S104)。所述方法可以提高保护用户所希望运行的应用程序的效率, 改善用户的使用体验。

一种进程保护方法、装置、设备以及计算机可读存储介质

- [1] 本申请要求于2017年9月14日提交中国专利局、申请号为CN 201710827274.1、申请名称为“一种进程保护方法、装置、设备以及计算机可读存储介质”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。
- [2] 技术领域
- [3] 本申请涉及计算机技术领域，尤其涉及一种进程保护方法、装置、设备以及计算机可读存储介质。
- [4] 背景技术
- [5] 随着科技技术的飞速发展，很多智能电子产品（如智能手机、平板电脑等）可以安装各种应用软件，但是所安装的应用软件的运行需要依赖智能电子产品自身安装的系统，该系统为操作系统（如安卓操作系统、windows操作系统、ios操作系统等），由于内存资源紧张或者系统本身的运行机制，会将后台运行的应用程序的进程杀死，从而导致用户所希望运行的应用程序无法继续正常运行的问题。基于上述问题，目前所采用的处理方法比较费电，并且解决问题的效率不高、影响体验，例如，通过双进程监测的方法，或者fork子进程的方法等等。另外，在应用程序运行过程中，当主线程阻塞5s未响应或者广播接收器10s未执行完时，系统就会报错，例如出现应用程序无响应（Application Not Responding，ANR），而且此时查看调试日志并不能获得导致出现ANR问题的具体信息，导致开发人员不能准确找出导致ANR问题的原因。
- [6] 发明内容
- [7] 本申请实施例提供一种进程保护方法、装置、设备以及计算机可读存储介质，可以提高保护应用程序的效率，改善用户的使用体验。
- [8] 一方面，本申请实施例提供了一种进程保护方法，该方法包括：
- [9] 若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程；
- [10] 建立所述服务进程与所述应用进程之间的连接关系；

- [11] 控制所述服务进程启动针对所述应用进程的保护程序；
- [12] 若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [13] 另一方面，本申请实施例提供了一种进程保护装置，该装置包括：
- [14] 生成单元，用于若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程；
- [15] 建立单元，用于建立所述服务进程与所述应用进程之间的连接关系；
- [16] 第一控制单元，用于控制所述服务进程启动针对所述应用进程的保护程序；
- [17] 第二控制单元，用于若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [18] 又一方面，本申请实施例还提供了一种进程保护设备，包括：
- [19] 存储器，用于存储实现进程保护的程序；以及
- [20] 处理器，用于运行所述存储器中存储的实现进程保护的程序，以执行以下操作：
- [21] 若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程；
- [22] 建立所述服务进程与所述应用进程之间的连接关系；
- [23] 控制所述服务进程启动针对所述应用进程的保护程序；
- [24] 若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [25] 再一方面，本申请实施例还提供了一种计算机可读存储介质，所述计算机可读存储介质存储有一个或者一个以上程序，所述一个或者一个以上程序可被一个或者一个以上的处理器执行，以实现以下步骤：
- [26] 若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程；
- [27] 建立所述服务进程与所述应用进程之间的连接关系；
- [28] 控制所述服务进程启动针对所述应用进程的保护程序；
- [29] 若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [30] 本申请实施例通过若检测到一应用进程正在运行，生成与所述应用进程相对应

的服务进程；建立所述服务进程与所述应用进程之间的连接关系；控制所述服务进程启动针对所述应用进程的保护程序；若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，从而可以提高保护用户所希望运行的应用程序的效率，改善用户的使用体验。

[31] 附图说明

[32] 为了更清楚地说明本申请实施例技术方案，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，下面描述中的附图是本申请的一些实施例，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。

[33] 图1是本申请实施例提供的一种进程保护方法的示意图；

[34] 图2是本申请实施例提供的一种进程保护方法的另一示意图；

[35] 图3是本申请实施例提供的一种进程保护方法的另一示意图；

[36] 图4是本申请实施例提供的一种进程保护方法的另一示意图；

[37] 图5是本申请实施例提供的一种进程保护方法的另一示意图；

[38] 图6是本申请实施例提供的一种进程保护装置的示意性框图；

[39] 图7是本申请实施例提供的一种进程保护装置的另一示意性框图；

[40] 图8是本申请实施例提供的一种进程保护装置的另一示意性框图；

[41] 图9是本申请实施例提供的一种进程保护装置的另一示意性框图；

[42] 图10是本申请实施例提供的一种进程保护装置的另一示意性框图；

[43] 图11是本申请实施例提供的一种进程保护设备结构组成示意图。

[44] 具体实施方式

[45] 下面将结合本申请实施例中的附图，对本申请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申请一部分实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有做出创造性劳

[46] 动前提下所获得的所有其他实施例，都属于本申请保护的范围。

[47] 应当理解，当在本说明书和所附权利要求书中使用时，术语“包括”和“包含”指示所描述特征、整体、步骤、操作、元素和/或组件的存在，但并不排除一个或

多个其它特征、整体、步骤、操作、元素、组件和/或其集合的存在或添加。

[48] 还应当理解，在此本申请说明书中所使用的术语仅仅是出于描述特定实施例的目的而并不意在限制本申请。如在本申请说明书和所附权利要求书中所使

[49] 用的那样，除非上下文清楚地指明其它情况，否则单数形式的“一”、“一个”及“该”意在包括复数形式。

[50] 请参阅图1，图1是本申请实施例提供的一种进程保护方法的示意图。该方法可以运行在智能手机（如Android手机、IOS手机等）、平板电脑、笔记本电脑以及智能设备等终端中。本申请实施例所述的进程保护方法可以提高保护用户所希望运行的应用程序的效率，改善用户的使用体验。图1是本申请实施例提供的一种进程保护方法的示意图。该方法包括步骤S101~S104。

[51] S101，若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。

[52] 在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

[53] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming

Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函

数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

[54] S102，建立所述服务进程与所述应用进程之间的连接关系。

[55] 在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。

[56] S103，控制所述服务进程启动针对所述应用进程的保护程序。

[57] 在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[58] S104，若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。

[59] 在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序

的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[60] 由以上可见，通过若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程；建立所述服务进程与所述应用进程之间的连接关系；控制所述服务进程启动针对所述应用进程的保护程序；若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，从而可以提高保护用户所希望运行的应用程序的效率，改善用户的使用体验。

[61] 请参阅图2，图2是本申请实施例提供的一种进程保护方法的示意图。该方法可以运行在智能手机（如Android手机、IOS手机等）、平板电脑、笔记本电脑以及智能设备等终端中。本申请实施例所述的进程保护方法可以提高保护用户所希望运行的应用程序的效率，改善用户的使用体验。图2是本申请实施例提供的一种进程保护方法的示意图。该方法包括步骤S201~S208。

[62] S201，若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。

[63] 在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

[64] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的

服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application

Programming

Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

[65] S202，建立所述服务进程与所述应用进程之间的连接关系。

[66] 在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。

[67] S203，获取与所述服务进程相对应的分叉子进程。

[68] 在本申请实施例中，将所述服务进程fork出来的子进程作为所述分叉子进程，将分叉子进程的上一级进程称为父进程，在分叉子进程与父进程之间具体某种特性，即当父进程被杀掉后，与其相关的分叉子进程仍然可以存活，并且不受父进程的影响。

[69] S204，在所述分叉子进程中创建定时器。

[70] 在本申请实施例中，在获得所述分叉子进程后，针对所获得的分叉子进程添加一个定时器，所述定时器可以是反复性定时器，也可以是一次性定时器，具体的定时器类型可以根据用户的需要自行选择，在此不作限定。

[71] S205，设置所述定时器的时间间隔。

[72] 在本申请实施例中，通过设置定时器时间间隔，决定分叉子进程所运行的两个间隔时间，如设置10秒，则10秒内分叉子进程只可运行一次。

[73] S206, 根据所述时间间隔控制所述应用进程显示运行信息的操作。

[74] 在本申请实施例中, 分叉子进程的运行用于显示与其对应的应用进程的运行信息, 根据所述时间间隔控制所述应用进程显示运行信息的操作, 例如, 可以将定时器的设置改为手动设置, 分叉子进程运行一次之后必须响应某些操作才可再次运行。通过设置定时器以控制应用进程显示运行信息的操作, 可以达到动态显示当前应用进程信息的效果。

[75] S207, 控制所述服务进程启动针对所述应用进程的保护程序。

[76] 在本申请实施例中, 启动针对所述应用进程的保护程序, 可选的, 若当前应用程序 (Application Programming Interface, API) 编程接口数量小于18, 在前台Service中直接传入Notification()函数, 设置定时唤醒应用程序的时间间隔, 并通过发送唤醒广播来促使已挂掉的应用进程重新启动, 其中, 该时间间隔可以是5分钟、10分钟, 或者其他时间, 具体的时间间隔在此不作限定; 若API编程接口数量大于或者等于18, 同时启动两个id相同的前台Service, 然后再将后启动的Service做stop处理, 并通过使用Hack手段对正在被杀掉的应用进程作保活, 即通过提高被保护的应用进程的优先级, 在应用进程将要面临被杀掉时, 利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[77] S208, 若检测到用于停止所述应用进程的停止事件时, 控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。

[78] 在本申请实施例中, 控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行, 可选的, 若应用进程中断时, 应用进程和服务进程之间的连接便会中断, 服务进程会接收到应用进程中断通知消息, 服务进程一旦接收到中断通知消息, 即确定应用程序可能已被中断, 将重新启动应用程序的应用进程中的部分或者全部功能; 可选的, 若应用进程将被中断时, 通过使用Hack手段对正在被杀掉的应用进程作保活, 通过提高被保护的应用进程的优先级, 在应用进程将要面临被杀掉时, 利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[79] 由以上可见, 本申请实施例通过创建定时器, 设置所述定时器的时间间隔, 根

据所述时间间隔控制所述应用进程显示运行信息的操作，可以达到动态显示当前应用程序信息的效果，改善用户的使用体验。

[80] 请参阅图3，图3是本申请实施例提供的一种进程保护方法的示意图。该方法可以运行在智能手机（如Android手机、IOS手机等）、平板电脑、笔记本电脑以及智能设备等终端中。本申请实施例所述的进程保护方法可以提高保护用户所希望运行的应用程序的效率，改善用户的使用体验。图3是本申请实施例提供的一种进程保护方法的示意图。该方法包括步骤S301~S308。

[81] S301，若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。

[82] 在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

[83] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

[84] S302，建立所述服务进程与所述应用进程之间的连接关系。

- [85] 在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。
- [86] S303，获取与所述服务进程相对应的分叉子进程。
- [87] 在本申请实施例中，将所述服务进程fork出来的子进程作为所述分叉子进程，将分叉子进程的上一级进程称为父进程，在分叉子进程与父进程之间具体某种特性，即当父进程被杀掉后，与其相关的分叉子进程仍然可以存活，并且不受父进程的影响。
- [88] S304，在所述分叉子进程中设置无障碍服务类。
- [89] 在本申请实施例中，通过在分叉子进程设置无障碍服务类，并回调无障碍服务类中的方法，可以达到动态显示当前运行的应用进程信息的效果。可选的，所设置的无障碍类可以是AccessibilityService类。
- [90] S305，调用所述无障碍服务类中预设的方法项。
- [91] 在本申请实施例中，在的无障碍服务类中的预设的方法项可以是onAccessibilityEvent(AccessibilityEvent event)方法。
- [92] 进一步地，如图4所示，步骤S305包括步骤S401~S402。
- [93] S401，若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项。
- [94] 在本申请实施例中，当系统检测到一个符合无障碍服务设定的无障碍事件过滤参数的事件时，预设的方法项被系统回调。
- [95] S402，传递关联的无障碍事件并向用户反馈。
- [96] 在本申请实施例中，例如，当用户在应用程序中点击一个按钮或聚焦一个用户界面控件时，无障碍服务会为此应用程序提供反馈，此时，系统调用所预设的

方法项，传递相关联的无障碍事件，无障碍服务就可以翻译和使用该事件为用户提供反馈。需要说明的是，在无障碍服务的生命周期中，所预设的方法项会被调用多次。

[97] S306，根据所述预设的方法项获取所述应用进程的运行信息。

[98] 在本申请实施例中，通过回调无障碍服务类中的方法，可以动态显示当前运行的应用进程信息。

[99] S307，控制所述服务进程启动针对所述应用进程的保护程序。

[100] 在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[101] S308，若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。

[102] 在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[103] 由以上可见，本申请实施例通过设置无障碍服务类，并调用所述无障碍服务类

中预设的方法项，可以达到动态显示当前应用程序信息的效果，改善用户的使用体验。

[104] 请参阅图5，图5是本申请实施例提供的一种进程保护方法的示意图。该方法可以运行在智能手机（如Android手机、IOS手机等）、平板电脑、笔记本电脑以及智能设备等终端中。本申请实施例所述的进程保护方法可以提高保护用户所希望运行的应用程序的效率，改善用户的使用体验，并且可以帮助开发人员准确查找、定位出现ANR问题的原因。图5是本申请实施例提供的一种进程保护方法的示意图。该方法包括步骤S501~S509。

[105] S501，若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。

[106] 在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

[107] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

[108] S502, 创建监测线程。

[109] 在本申请实施例中, 一般实时监控功能放在一个单独开辟的线程比较合适, 因为这样可以大大减轻主线程的负担。创建监测线程, 可选的, 通过在终端的平台上新建一个对话框工程, 如Test, 并在 `BOOL C TestDlg::OnInitDialog()` 函数里创建所述监测线程。

[110] S503, 根据所述监测线程向所述应用进程指示或者派发任务。

[111] 在本申请实施例中, 在所述监测线程内的Looper (有消息循环的线程) 使用Handler (异步消息的处理) 不断发消息往应用进程中的UI线程, 并派发一个任务由系统去处理。

[112] S504, 记录所述任务指示或者派发的时间戳。

[113] 在本申请实施例中, 用户可以记录下所派发任务的时间戳, 而这个任务会修改派发任务时记录下的时间戳。派发完任务后, 线程睡眠固定时间 (一般为5s), 并等待线程重新开启。

[114] S505, 监测所述时间戳是否发生变化。

[115] 在本申请实施例中, 待线程重新开启后, 通过判断记录的时间是否有改变来检测之前派发的任务是否执行了, 如果任务未被执行, 证明线程阻塞了, 成功监测到应用程序无响应 (Application Not Responding, ANR)。

[116] S506, 若检测到所述时间戳未发生变化, 判定所述应用进程处于ANR状态。

[117] S507, 建立所述服务进程与所述应用进程之间的连接关系。

[118] 在本申请实施例中, 当为应用程序创建服务进程之后, 终端需要建立服务进程和应用进程之间的连接, 通过监控该连接的状态, 确定是否需要与服务进程相对应的应用程序进行保护。其中, 终端为服务进程创建了套接口, 监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系, 终端可以建立应用程序的服务进程和应用进程之间的连接, 并且通过该连接来实现数据传输, 并通过监听该连接是否中断来判断应用程序的状态, 对应用程序的状态确定过程方便快捷, 系统时延小, 系统资源消耗小。

[119] S508, 控制所述服务进程启动针对所述应用进程的保护程序。

- [120] 在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。
- [121] S509，若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [122] 在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。
- [123] 由以上可见，本申请实施例通过创建监测线程，根据所述监测线程向所述应用进程指示或者派发任务，记录所述任务指示或者派发的时间戳，监测所述时间戳是否发生变化，若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态，从而可以帮助开发人员准确查找、定位出现ANR问题的原因。
- [124] 请参阅图6，对应上述一种进程保护方法，本申请实施例还提出一种进程保护装置，该装置100包括：生成单元101、建立单元102、第一控制单元103、第二控制单元104。
- [125] 其中，所述生成单元101，用于若检测到一应用进程正在运行，生成与所述应

用进程相对应的服务进程。在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

- [126] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

- [127] 建立单元102，用于建立所述服务进程与所述应用进程之间的连接关系。在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。

- [128] 第一控制单元103，用于控制所述服务进程启动针对所述应用进程的保护程序

。在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[129] 第二控制单元104，用于若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[130] 请参阅图7，对应上述一种进程保护方法，本申请实施例还提出一种进程保护装置，该终端200包括：生成单元201、建立单元202、第一获取单元203、创建单元204、第一设置单元205、第三控制单元206、第一控制单元207、第二控制单元208。

[131] 其中，所述生成单元201，用于若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信

、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

- [132] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

- [133] 建立单元202，用于建立所述服务进程与所述应用进程之间的连接关系。在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。

- [134] 第一获取单元203，用于获取与所述服务进程相对应的分叉子进程。在本申请实施例中，将所述服务进程fork出来的子进程作为所述分叉子进程，将分叉子进程的上一级进程称为父进程，在分叉子进程与父进程之间具体某种特性，即当父进程被杀掉后，与其相关的分叉子进程仍然可以存活，并且不受父进程的影

响。

- [135] 创建单元204，用于在所述分叉子进程中创建定时器。在本申请实施例中，在获得所述分叉子进程后，针对所获得的分叉子进程添加一个定时器，所述定时器可以是反复性定时器，也可以是一次性定时器，具体的定时器类型可以根据用户的需要自行选择，在此不作限定。
- [136] 第一设置单元205，用于设置所述定时器的时间间隔。在本申请实施例中，通过设置定时器时间间隔，决定分叉子进程所运行的两个间隔时间，如设置10秒，则10秒内分叉子进程只可运行一次。
- [137] 第三控制单元206，用于根据所述时间间隔控制所述应用进程显示运行信息的操作。在本申请实施例中，分叉子进程的运行用于显示与其对应的应用进程的运行信息，根据所述时间间隔控制所述应用进程显示运行信息的操作，例如，可以将定时器的设置改为手动设置，分叉子进程运行一次之后必须响应某些操作才可再次运行。通过设置定时器以控制应用进程显示运行信息的操作，可以达到动态显示当前应用进程信息的效果。
- [138] 第一控制单元207，用于控制所述服务进程启动针对所述应用进程的保护程序。在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。
- [139] 第二控制单元208，用于若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程

并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[140] 请参阅图8，对应上述一种进程保护方法，本申请实施例还提出一种进程保护装置，该装置300包括：生成单元301、建立单元302、第二获取单元303、第二设置单元304、调用单元305、第三获取单元306、第一控制单元307、第二控制单元308。

[141] 其中，所述生成单元301，用于若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。

[142] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming

Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函

数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。

- [143] 建立单元302，用于建立所述服务进程与所述应用进程之间的连接关系。在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。
- [144] 第二获取单元303，用于获取与所述服务进程相对应的分叉子进程。在本申请实施例中，将所述服务进程fork出来的子进程作为所述分叉子进程，将分叉子进程的上一级进程称为父进程，在分叉子进程与父进程之间具体某种特性，即当父进程被杀掉后，与其相关的分叉子进程仍然可以存活，并且不受父进程的影响。
- [145] 第二设置单元304，用于在所述分叉子进程中设置无障碍服务类。在本申请实施例中，通过在分叉子进程设置无障碍服务类，并回调无障碍服务类中的方法，可以达到动态显示当前运行的应用进程信息的效果。可选的，所设置的无障碍类可以是AccessibilityService类。
- [146] 调用单元305，用于调用所述无障碍服务类中预设的方法项。在本申请实施例中，在的无障碍服务类中的预设的方法项可以是onAccessibilityEvent(AccessibilityEvent event)方法。
- [147] 第三获取单元306，用于根据所述预设的方法项获取所述应用进程的运行信息。在本申请实施例中，通过回调无障碍服务类中的方法，可以动态显示当前运行的应用进程信息。
- [148] 第一控制单元307，用于控制所述服务进程启动针对所述应用进程的保护程序。在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应

用程序（Application Programming

Interface, API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

- [149] 第二控制单元308，用于若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

- [150] 如图9所示，所述调用单元305，包括：

- [151] 调用子单元3051，用于若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项。在本申请实施例中，当系统检测到一个符合无障碍服务设定的无障碍事件过滤参数的事件时，预设的方法项被系统回调。

- [152] 反馈单元3052，用于传递关联的无障碍事件并向用户反馈。在本申请实施例中，例如，当用户在应用程序中点击一个按钮或聚焦一个用户界面控件时，无障碍服务会为此应用程序提供反馈，此时，系统调用所预设的方法项，传递相关联的无障碍事件，无障碍服务就可以翻译和使用该事件为用户提供反馈。需要说明的是，在无障碍服务的生命周期中，所预设的方法项会被调用多次。

- [153] 请参阅图10，对应上述一种进程保护方法，本申请实施例还提出一种进程保护装置，该装置400包括：生成单元401、创建单元402、执行单元403、记录单元404、监测单元405、判定单元406、建立单元407、第一控制单元408、第二控制单元409。
- [154] 其中，所述生成单元401，用于若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程。在本申请实施例中，在终端开启时，用户可以选择相对应的应用程序的开启，在开启应用程序时，与应用程序对应的应用进程便开始运行。其中，应用程序是指用户可以看得见并感觉得到的程序，例如微信、word、photoshop等，应用进程指的是一般用户可能感觉不到它的存在，并作为应用程序的执行实体而存在，例如一些服务、windows内核等计算机程序，他们在后台中默默的运行着。具体的，一个“应用程序”一定可以找到一个“应用进程”与其对应，但一个“应用进程”不一定会有一个“应用程序”与其对应，有些应用进程可能是后台的运行进程，用户无法看到，也无法感觉到。
- [155] 需要说明的是，若应用进程保持运行，则与该应用进程相对应的应用程序可以被使用，若应用进程被杀掉，即被停止运行，则与该应用进程相对应的应用程序将被关闭。若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程，可选的，可以利用终端系统的漏洞来生成一个服务进程，所生成的服务进程不会在终端系统的通知栏处出现，服务进程的运行就如同运行着一个后台service进程一样，生成的方法如下：（1）若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数；（2）若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理。
- [156] 创建单元402，用于创建监测线程。在本申请实施例中，一般实时监控功能放在一个单独开辟的线程比较合适，因为这样可以大大减轻主线程的负担。创建监测线程，可选的，通过在终端的平台上新建一个对话框工程，如Test，并在BOOL C TestDlg::OnInitDialog() 函数里创建所述监测线程。
- [157] 执行单元403，用于根据所述监测线程向所述应用进程指示或者派发任务。在

本申请实施例中，在所述监测线程内的Looper（有消息循环的线程）使用Handler（异步消息的处理）不断发消息往应用进程中的UI线程，并派发一个任务由系统去处理。

- [158] 记录单元404，用于记录所述任务指示或者派发的时间戳。在本申请实施例中，用户可以记录下所派发任务的时间戳，而这个任务会修改派发任务时记录下的时间戳。派发完任务后，线程睡眠固定时间（一般为5s），并等待线程重新开启。
- [159] 监测单元405，用于监测所述时间戳是否发生变化。在本申请实施例中，待线程重新开启后，通过判断记录的时间是否有改变来检测之前派发的任务是否执行了，如果任务未被执行，证明线程阻塞了，成功监测到应用程序无响应（Application Not Responding，ANR）。
- [160] 判定单元406，用于若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。
- [161] 建立单元407，用于建立所述服务进程与所述应用进程之间的连接关系。在本申请实施例中，当为应用程序创建服务进程之后，终端需要建立服务进程和应用进程之间的连接，通过监控该连接的状态，确定是否需要与服务进程相对应的应用程序进行保护。其中，终端为服务进程创建了套接口，监测服务进程和应用进程之间的连接即为监测服务进程的套接口和应用进程的套接口之间的连接。通过建立服务进程和应用进程之间的连接关系，终端可以建立应用程序的服务进程和应用进程之间的连接，并且通过该连接来实现数据传输，并通过监听该连接是否中断来判断应用程序的状态，对应用程序的状态确定过程方便快捷，系统时延小，系统资源消耗小。
- [162] 第一控制单元408，用于控制所述服务进程启动针对所述应用进程的保护程序。在本申请实施例中，启动针对所述应用进程的保护程序，可选的，若当前应用程序（Application Programming Interface，API）编程接口数量小于18，在前台Service中直接传入Notification()函数，设置定时唤醒应用程序的时间间隔，并通过发送唤醒广播来促使已挂掉的应用进程重新启动，其中，该时间间隔可以是5分钟、10分钟，或者其他时间，

具体的时间间隔在此不作限定；若API编程接口数量大于或者等于18，同时启动两个id相同的前台Service，然后再将后启动的Service做stop处理，并通过使用Hack手段对正在被杀掉的应用进程作保活，即通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[163] 第二控制单元409，用于若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。在本申请实施例中，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行，可选的，若应用进程中断时，应用进程和服务进程之间的连接便会中断，服务进程会接收到应用进程中断通知消息，服务进程一旦接收到中断通知消息，即确定应用程序可能已被中断，将重新启动应用程序的应用进程中的部分或者全部功能；可选的，若应用进程将被中断时，通过使用Hack手段对正在被杀掉的应用进程作保活，通过提高被保护的应用进程的优先级，在应用进程将要面临被杀掉时，利用被保护的应用进程的优先级高于其他进程而不被杀掉以达到保活的目的。

[164] 由以上可见，本申请实施例通过创建监测线程，根据所述监测线程向所述应用进程指示或者派发任务，记录所述任务指示或者派发的时间戳，监测所述时间戳是否发生变化，若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态，从而可以帮助开发人员准确查找、定位出现ANR问题的原因。

[165] 在硬件实现上，以上生成单元101、建立单元102、第一控制单元103、第二控制单元104等可以以硬件形式内嵌于或独立于数据处理的装置中，也可以以软件形式存储于数据处理装置的存储器中，以便处理器调用执行以上各个单元对应的操作。该处理器可以为中央处理单元（CPU）、微处理器、单片机等。

[166] 上述进程保护装置可以实现为一种计算机程序的形式，计算机程序可以在如图11所示的计算机设备上运行。

[167] 图11为本申请一种进程保护设备的结构组成示意图。该设备可以是终端，也可以是服务器，其中，终端可以是智能手机、平板电脑、笔记本电脑、台式电脑、个人数字助理和穿戴式设备等具有通信功能的电子设备。服务器可以是独立

的服务器，也可以是多个服务器组成的服务器集群。参照图11，该计算机设备500包括通过系统总线501连接的处理器502、非易失性存储介质503、内存储器504和网络接口505。其中，该计算机设备500的非易失性存储介质503可存储操作系统5031和计算机程序5032，该计算机程序5032被执行时，可使得处理器502执行一种进程保护方法。该计算机设备500的处理器502用于提供计算和控制能力，支撑整个计算机设备500的运行。该内存储器504为非易失性存储介质503中的计算机程序5032的运行提供环境，该计算机程序被处理器执行时，可使得处理器502执行一种进程保护方法。计算机设备500的网络接口505用于进行网络通信，如发送分配的任务等。本领域技术人员可以理解，图11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。

[168] 其中，所述处理器502执行如下操作：

[169] 若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程；

[170] 建立所述服务进程与所述应用进程之间的连接关系；

[171] 控制所述服务进程启动针对所述应用进程的保护程序；

[172] 若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。

[173] 在一些实施例中，所述处理器502还执行如下操作：

[174] 获取与所述服务进程相对应的分叉子进程；

[175] 在所述分叉子进程中创建定时器；

[176] 设置所述定时器的时间间隔；

[177] 根据所述时间间隔控制所述应用进程显示运行信息的操作。

[178] 在一些实施例中，所述处理器502还执行如下操作：

[179] 获取与所述服务进程相对应的分叉子进程；

[180] 在所述分叉子进程中设置无障碍服务类；

[181] 调用所述无障碍服务类中预设的方法项；

[182] 根据所述预设的方法项获取所述应用进程的运行信息。

- [183] 在一些实施例中，所述调用的无障碍服务类中预设的方法项，包括：
- [184] 若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项；
- [185] 传递关联的无障碍事件并向用户反馈。
- [186] 在一些实施例中，所述处理器502还执行如下操作：
- [187] 创建监测线程；
- [188] 根据所述监测线程向所述应用进程指示或者派发任务；
- [189] 记录所述任务指示或者派发的时间戳；
- [190] 监测所述时间戳是否发生变化；
- [191] 若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。
- [192] 本申请提供了一种计算机可读存储介质，所述计算机可读存储介质存储有一个或者一个以上程序，所述一个或者一个以上程序可被一个或者一个以上的处理器执行，以实现以下步骤：
- [193] 若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程；
- [194] 建立所述服务进程与所述应用进程之间的连接关系；
- [195] 控制所述服务进程启动针对所述应用进程的保护程序；
- [196] 若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [197] 计算机可读存储介质被所述处理器执行时，使得一个或多个处理器还执行以下步骤：
- [198] 获取与所述服务进程相对应的分叉子进程；
- [199] 在所述分叉子进程中创建定时器；
- [200] 设置所述定时器的时间间隔；
- [201] 根据所述时间间隔控制所述应用进程显示运行信息的操作。
- [202] 计算机可读存储介质被所述处理器执行时，使得一个或多个处理器还执行以下步骤：
- [203] 获取与所述服务进程相对应的分叉子进程；
- [204] 在所述分叉子进程中设置无障碍服务类；

- [205] 调用所述无障碍服务类中预设的方法项；
- [206] 根据所述预设的方法项获取所述应用进程的运行信息。
- [207] 计算机可读存储介质被所述处理器执行时，使得一个或多个处理器还执行以下步骤：所述调用的无障碍服务类中预设的方法项，包括：
- [208] 若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项；
- [209] 传递关联的无障碍事件并向用户反馈。
- [210] 计算机可读存储介质被所述处理器执行时，使得一个或多个处理器还执行以下步骤：
- [211] 创建监测线程；
- [212] 根据所述监测线程向所述应用进程指示或者派发任务；
- [213] 记录所述任务指示或者派发的时间戳；
- [214] 监测所述时间戳是否发生变化；
- [215] 若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。
- [216] 本申请前述的存储介质可以是非易失性存储介质，例如：磁碟、光盘、只读存储记忆体（Read-Only Memory，ROM）等各种可以存储程序代码的介质。
- [217] 本申请所有实施例中的单元可以通过通用集成电路，例如CPU（Central Processing Unit，中央处理器），或通过ASIC（Application Specific Integrated
- [218] Circuit，专用集成电路）来实现。
- [219] 本申请实施例进程保护方法中的步骤可以根据实际需要进行顺序调整、合并和删减。
- [220] 本申请实施例进程保护装置中的单元可以根据实际需要进行合并、划分和删减。
- [221] 以上所述，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到各种等效的修改或替换，这些修改或替换都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以权利要求的保护范围为准。

权利要求书

- [权利要求 1] 一种进程保护方法，其特征在于，所述方法包括：
若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程；
建立所述服务进程与所述应用进程之间的连接关系；
控制所述服务进程启动针对所述应用进程的保护程序；
若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [权利要求 2] 如权利要求1所述的方法，其特征在于，在所述建立所述服务进程与所述应用进程之间的连接关系之后，所述方法还包括：
获取与所述服务进程相对应的分叉子进程；
在所述分叉子进程中创建定时器；
设置所述定时器的时间间隔；
根据所述时间间隔控制所述应用进程显示运行信息的操作。
- [权利要求 3] 如权利要求1所述的方法，其特征在于，在所述建立所述服务进程与所述应用进程之间的连接关系之后，所述方法还包括：
获取与所述服务进程相对应的分叉子进程；
在所述分叉子进程中设置无障碍服务类；
调用所述无障碍服务类中预设的方法项；
根据所述预设的方法项获取所述应用进程的运行信息。
- [权利要求 4] 如权利要求3所述的方法，其特征在于，所述调用的无障碍服务类中预设的方法项，包括：
若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项；
传递关联的无障碍事件并向用户反馈。
- [权利要求 5] 如权利要求1所述的方法，其特征在于，在所述若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程之后，所

述方法包括：

创建监测线程；

根据所述监测线程向所述应用进程指示或者派发任务；

记录所述任务指示或者派发的时间戳；

监测所述时间戳是否发生变化；

若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。

[权利要求 6]

一种进程保护装置，其特征在于，所述装置包括：

生成单元，用于若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程；

建立单元，用于建立所述服务进程与所述应用进程之间的连接关系；

第一控制单元，用于控制所述服务进程启动针对所述应用进程的保护程序；

第二控制单元，用于若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。

[权利要求 7]

如权利要求6所述的装置，其特征在于，所述装置还包括：

第一获取单元，用于获取与所述服务进程相对应的分叉子进程；

创建单元，用于在所述分叉子进程中创建定时器；

第一设置单元，用于设置所述定时器的时间间隔；

第三控制单元，用于根据所述时间间隔控制所述应用进程显示运行信息的操作。

[权利要求 8]

如权利要求6所述的装置，其特征在于，所述装置还包括：

第二获取单元，用于获取与所述服务进程相对应的分叉子进程；

第二设置单元，用于在所述分叉子进程中设置无障碍服务类；

调用单元，用于调用所述无障碍服务类中预设的方法项；

第三获取单元，用于根据所述预设的方法项获取所述应用进程的

运行信息。

- [权利要求 9] 如权利要求8所述的装置，其特征在于，所述调用单元，包括：
调用子单元，用于若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项；
反馈单元，用于传递关联的无障碍事件并向用户反馈。
- [权利要求 10] 如权利要求1所述装置，其特征在于，所述装置还包括：
创建单元，用于创建监测线程；
执行单元，用于根据所述监测线程向所述应用进程指示或者派发任务；
记录单元，用于记录所述任务指示或者派发的时间戳；
监测单元，用于监测所述时间戳是否发生变化；
判定单元，用于若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。
- [权利要求 11] 一种进程保护设备，其特征在于，包括：
存储器，用于存储实现进程保护的程序；以及
处理器，用于运行所述存储器中存储的进程保护的程序，以执行以下操作：
若检测到一应用进程正在运行，生成与所述应用进程相对应的服务进程；
建立所述服务进程与所述应用进程之间的连接关系；
控制所述服务进程启动针对所述应用进程的保护程序；
若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。
- [权利要求 12] 如权利要求11所述的设备，其特征在于，所述处理器还执行以下操作：
获取与所述服务进程相对应的分叉子进程；
在所述分叉子进程中创建定时器；

设置所述定时器的时间间隔；

根据所述时间间隔控制所述应用进程显示运行信息的操作。

[权利要求 13]

如权利要求11所述的设备，其特征在于，所述处理器还执行以下操作：

获取与所述服务进程相对应的分叉子进程；

在所述分叉子进程中设置无障碍服务类；

调用所述无障碍服务类中预设的方法项；

根据所述预设的方法项获取所述应用进程的运行信息。

[权利要求 14]

如权利要求13所述的设备，其特征在于，所述调用的无障碍服务类中预设的方法项，包括：

若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项；

传递关联的无障碍事件并向用户反馈。

[权利要求 15]

如权利要求11所述的设备，其特征在于，所述处理器还执行以下操作：

创建监测线程；

根据所述监测线程向所述应用进程指示或者派发任务；

记录所述任务指示或者派发的时间戳；

监测所述时间戳是否发生变化；

若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。

[权利要求 16]

一种计算机可读存储介质，其特征在于，所述计算机可读存储介质存储有一个或者一个以上程序，所述一个或者一个以上程序可被一个或者一个以上的处理器执行，以实现以下步骤：

若检测到某一应用进程正在运行，生成与所述应用进程相对应的服务进程；

建立所述服务进程与所述应用进程之间的连接关系；

控制所述服务进程启动针对所述应用进程的保护程序；

若检测到用于停止所述应用进程的停止事件时，控制所述服务进程利用所述保护程序保护所述应用进程并使所述应用进程继续运行。

[权利要求 17] 如权利要求16所述的计算机可读存储介质，其特征在于，所述步骤还包括：

获取与所述服务进程相对应的分叉子进程；

在所述分叉子进程中创建定时器；

设置所述定时器的时间间隔；

根据所述时间间隔控制所述应用进程显示运行信息的操作。

[权利要求 18] 如权利要求16所述的计算机可读存储介质，其特征在于，所述步骤还包括：

获取与所述服务进程相对应的分叉子进程；

在所述分叉子进程中设置无障碍服务类；

调用所述无障碍服务类中预设的方法项；

根据所述预设的方法项获取所述应用进程的运行信息。

[权利要求 19] 如权利要求18所述的计算机可读存储介质，其特征在于，所述调用的无障碍服务类中预设的方法项，包括：

若检测到符合无障碍服务设定的无障碍事件过滤参数事件时，调用所述预设方法项；

传递关联的无障碍事件并向用户反馈。

[权利要求 20] 如权利要求16所述的计算机可读存储介质，其特征在于，所述步骤还包括：

创建监测线程；

根据所述监测线程向所述应用进程指示或者派发任务；

记录所述任务指示或者派发的时间戳；

监测所述时间戳是否发生变化；

若检测到所述时间戳未发生变化，判定所述应用进程处于ANR状态。

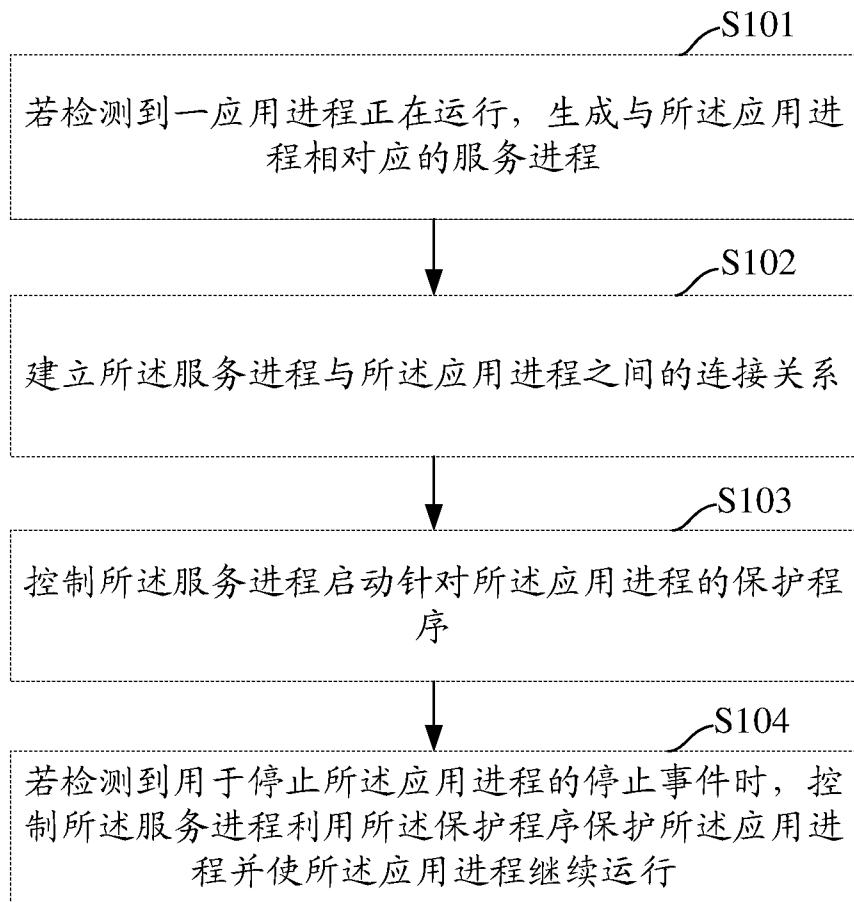


图 1

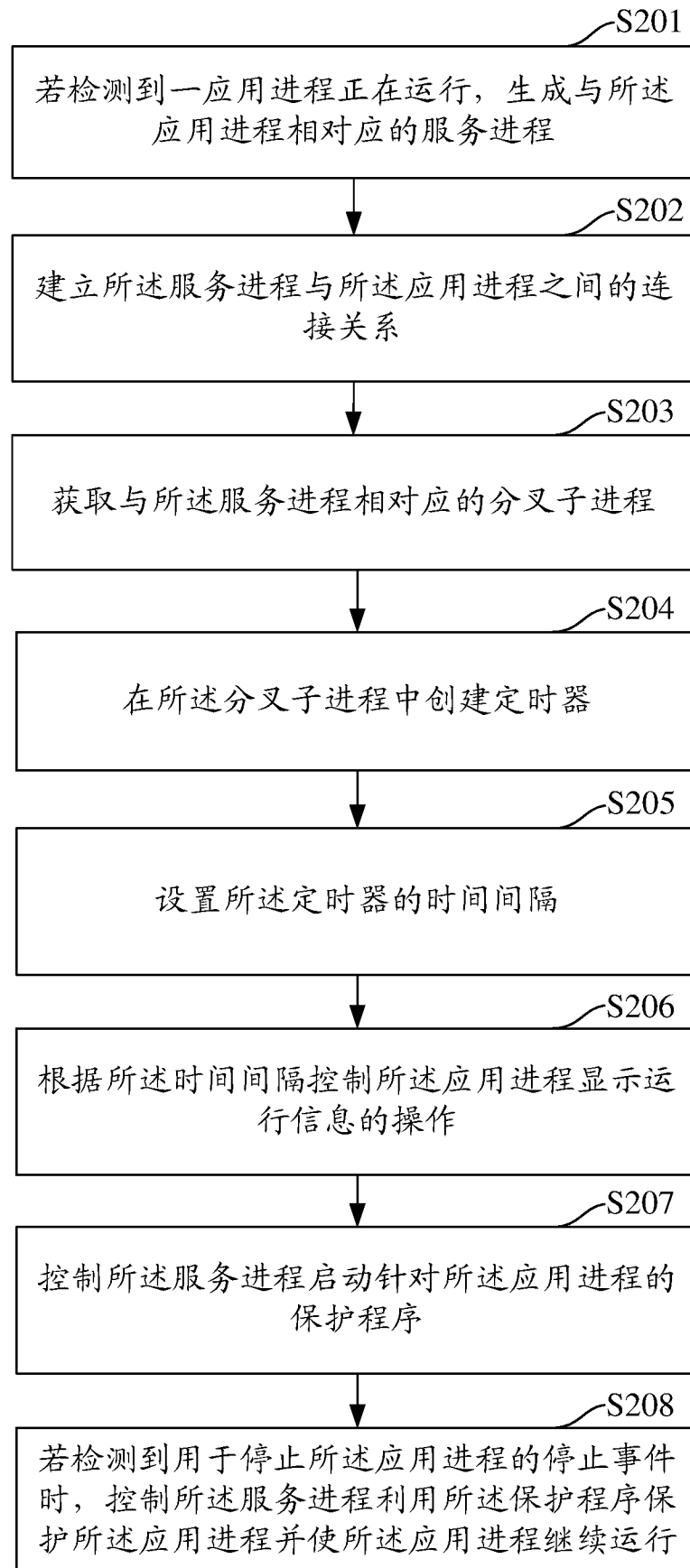


图 2

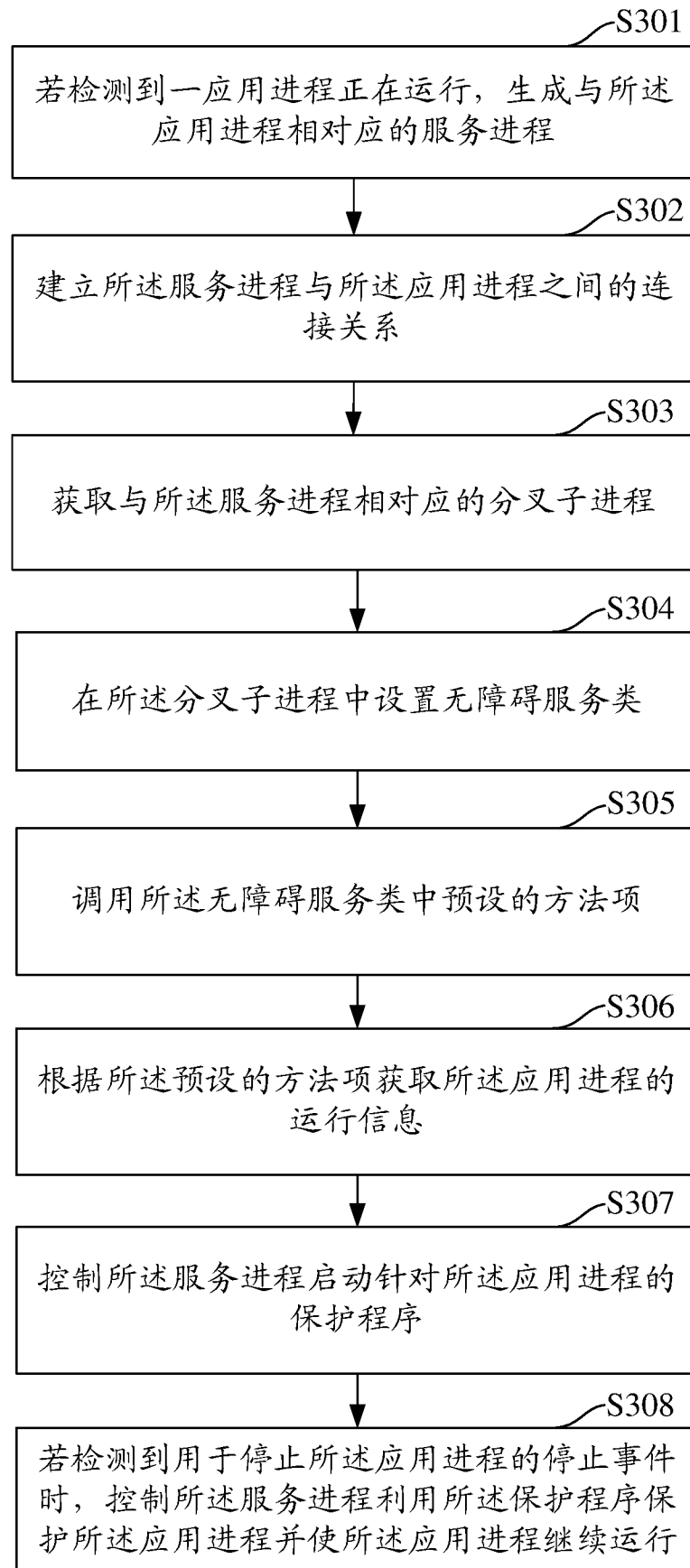


图 3

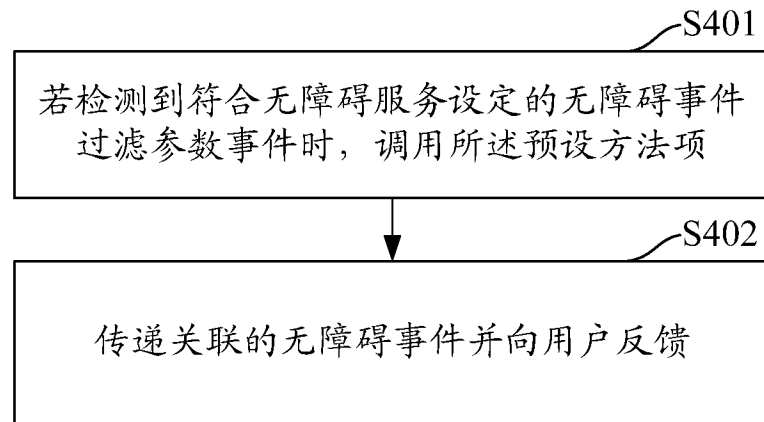


图 4

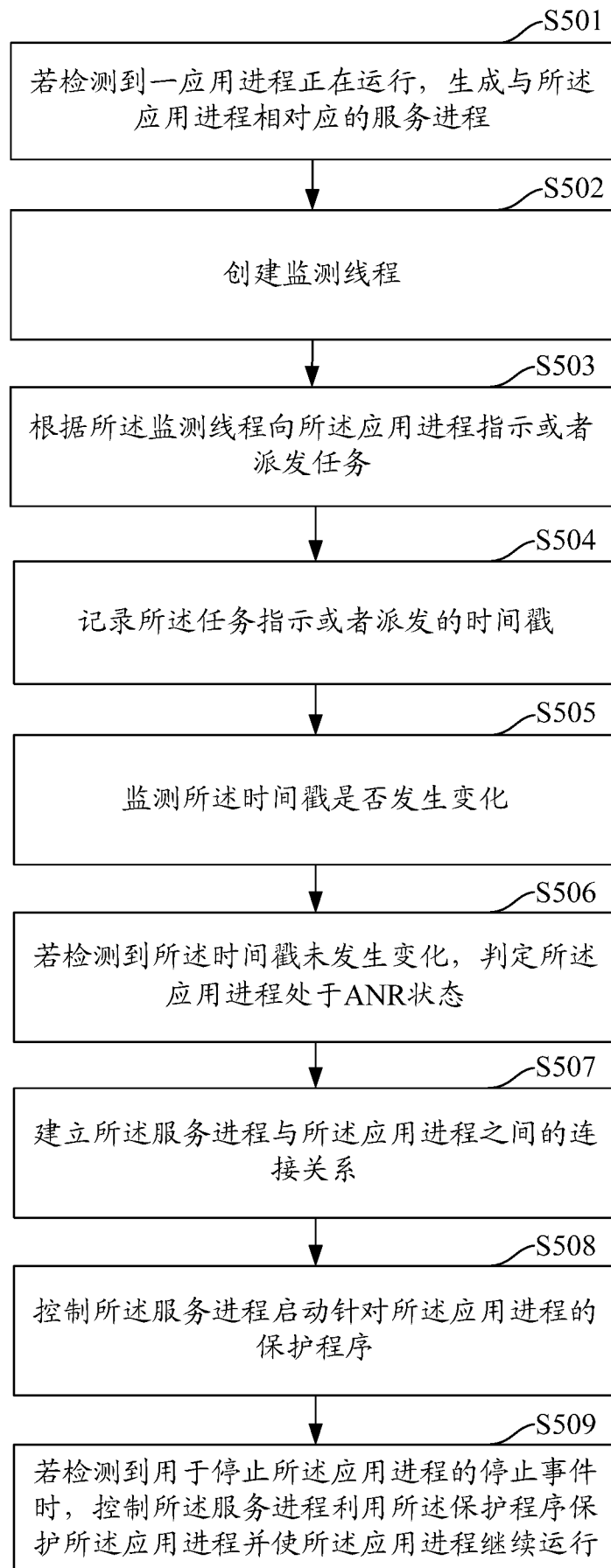


图 5

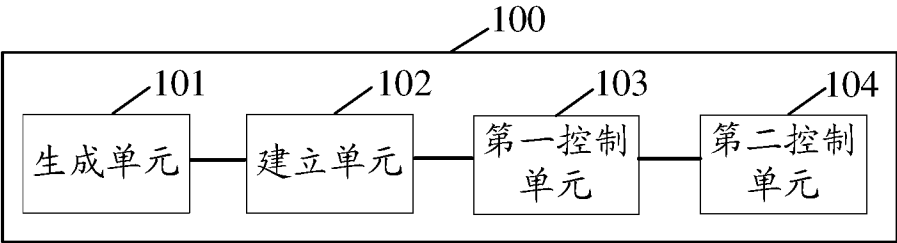


图 6

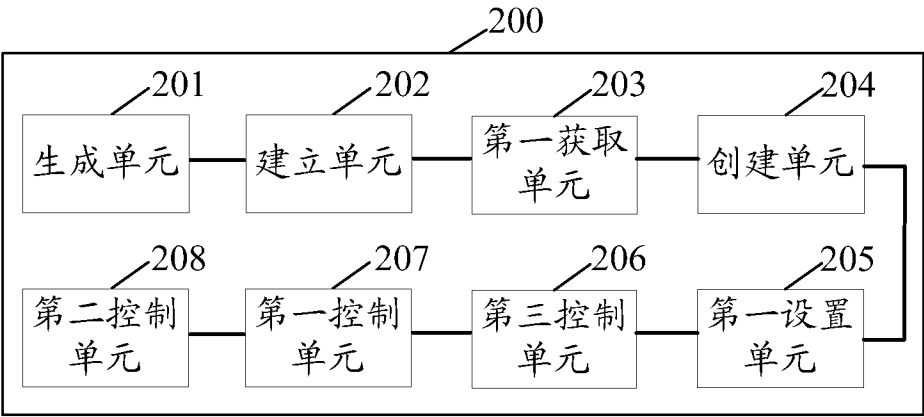


图 7

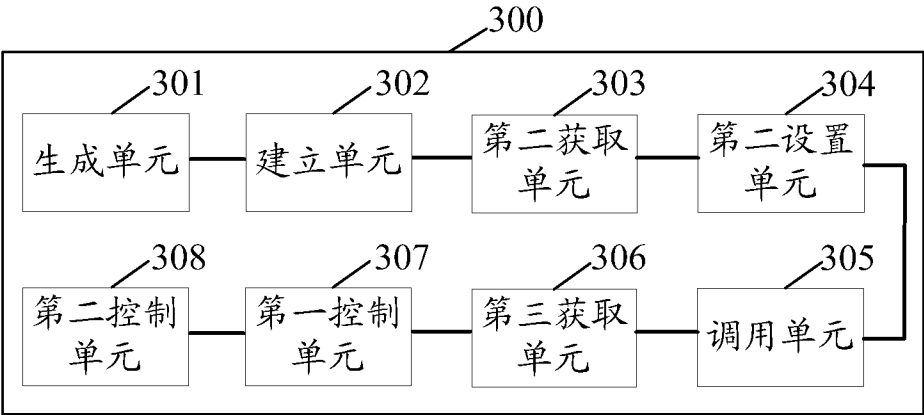


图 8

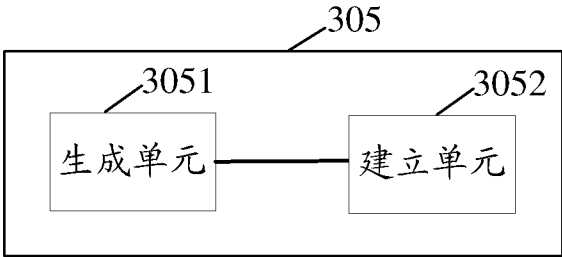


图 9

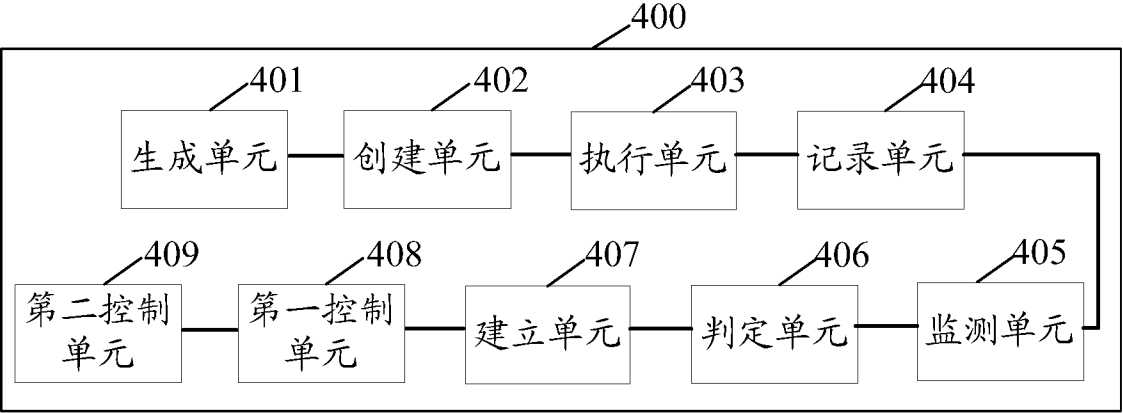


图 10

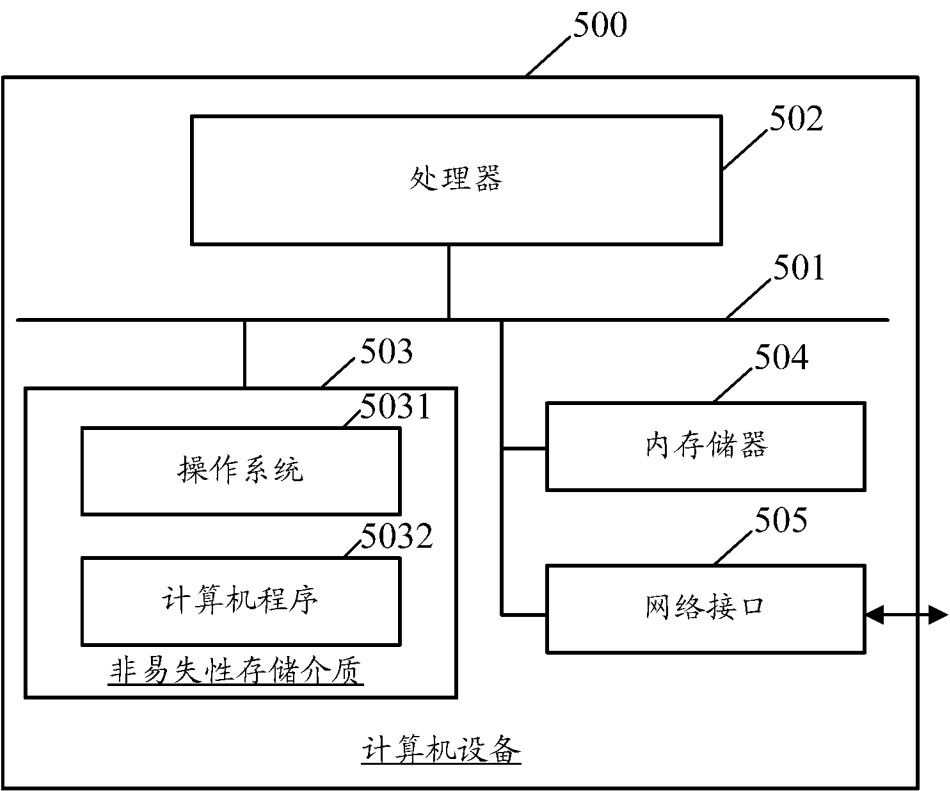


图 11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2017/108238

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/44 (2018.01) i; G06F 9/48 (2006.01) i; G06F 21/54 (2013.01) 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; CNKI; DWPI; EPODOC: 进程, 应用进程, 服务进程, service 进程, 停止, 中断, 终止, 保护, 分叉, 监控, 时间, 间隔, 状态, process, application process, service process, protect, stop, fork, monitor, interval, state

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 105303103 A (BEIJING QIHOO TECHNOLOGY CO., LTD. et al.) 03 February 2016 (03.02.2016), description, paragraphs [0024]-[0026]	1-20
A	CN 104809400 A (UNION MOBILE PAY TECHNOLOGY CO., LTD.) 29 July 2015 (29.07.2015), entire document	1-20
A	CN 104598313 A (GUANGZHOU JIUBANG DIGITAL TECHNOLOGY CO., LTD. et al.) 06 May 2015 (06.05.2015), entire document	1-20
A	CN 102193825 A (STRONGUNION TECHNOLOGY CO., LTD.) 21 September 2011 (21.09.2011), entire document	1-20
A	CN 105589757 A (SHANGHAI TIANMAI JUYUAN CULTURE MEDIA CO., LTD.) 18 May 2016 (18.05.2016), entire document	1-20

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

* Special categories of cited documents:	"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
"A" document defining the general state of the art which is not considered to be of particular relevance	"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
"E" earlier application or patent but published on or after the international filing date	"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
"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)	"&" document member of the same patent family
"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	

Date of the actual completion of the international search 03 June 2018	Date of mailing of the international search report 13 June 2018
---	--

Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451	Authorized officer LIU, Ke Telephone No. (86-10) 53961302
---	---

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2017/108238

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 105303103 A	03 February 2016	None	
CN 104809400 A	29 July 2015	None	
CN 104598313 A	06 May 2015	None	
CN 102193825 A	21 September 2011	None	
CN 105589757 A	18 May 2016	None	

国际检索报告

国际申请号

PCT/CN2017/108238

A. 主题的分类 G06F 9/44(2018.01)i; G06F 9/48(2006.01)i; G06F 21/54(2013.01)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;CNKI;DWPI;EPDOC:进程, 应用进程, 服务进程, service进程, 停止, 中断, 终止, 保护, 分叉, 监控, 时间, 间隔, 状态, process, application process, service process, protect, stop, fork, monitor, interval, state																				
C. 相关文件 <table border="1"> <thead> <tr> <th>类 型*</th> <th>引用文件, 必要时, 指明相关段落</th> <th>相关的权利要求</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>CN 105303103 A (北京奇虎科技有限公司 等) 2016年 2月 3日 (2016 - 02 - 03) 说明书第[0024]-[0026]段</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 104809400 A (联动优势科技有限公司) 2015年 7月 29日 (2015 - 07 - 29) 全文</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 104598313 A (广州市久邦数码科技有限公司 等) 2015年 5月 6日 (2015 - 05 - 06) 全文</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 102193825 A (北京思创银联科技股份有限公司) 2011年 9月 21日 (2011 - 09 - 21) 全文</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 105589757 A (上海天脉聚源文化传媒有限公司) 2016年 5月 18日 (2016 - 05 - 18) 全文</td> <td>1-20</td> </tr> </tbody> </table>			类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求	X	CN 105303103 A (北京奇虎科技有限公司 等) 2016年 2月 3日 (2016 - 02 - 03) 说明书第[0024]-[0026]段	1-20	A	CN 104809400 A (联动优势科技有限公司) 2015年 7月 29日 (2015 - 07 - 29) 全文	1-20	A	CN 104598313 A (广州市久邦数码科技有限公司 等) 2015年 5月 6日 (2015 - 05 - 06) 全文	1-20	A	CN 102193825 A (北京思创银联科技股份有限公司) 2011年 9月 21日 (2011 - 09 - 21) 全文	1-20	A	CN 105589757 A (上海天脉聚源文化传媒有限公司) 2016年 5月 18日 (2016 - 05 - 18) 全文	1-20
类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求																		
X	CN 105303103 A (北京奇虎科技有限公司 等) 2016年 2月 3日 (2016 - 02 - 03) 说明书第[0024]-[0026]段	1-20																		
A	CN 104809400 A (联动优势科技有限公司) 2015年 7月 29日 (2015 - 07 - 29) 全文	1-20																		
A	CN 104598313 A (广州市久邦数码科技有限公司 等) 2015年 5月 6日 (2015 - 05 - 06) 全文	1-20																		
A	CN 102193825 A (北京思创银联科技股份有限公司) 2011年 9月 21日 (2011 - 09 - 21) 全文	1-20																		
A	CN 105589757 A (上海天脉聚源文化传媒有限公司) 2016年 5月 18日 (2016 - 05 - 18) 全文	1-20																		
☐ 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 2018年 6月 3日		国际检索报告邮寄日期 2018年 6月 13日																		
ISA/CN的名称和邮寄地址 中华人民共和国国家知识产权局(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 刘可 电话号码 86-(10)-53961302																		

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2017/108238

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	105303103	A	2016年 2月 3日	无	
CN	104809400	A	2015年 7月 29日	无	
CN	104598313	A	2015年 5月 6日	无	
CN	102193825	A	2011年 9月 21日	无	
CN	105589757	A	2016年 5月 18日	无	