

(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



(10) 国际公布号

WO 2019/201011 A1

(43) 国际公布日
2019 年 10 月 24 日 (24.10.2019)

(51) 国际专利分类号:

G06F 12/02 (2006.01)

国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(21) 国际申请号: PCT/CN2019/075599

(22) 国际申请日: 2019 年 2 月 20 日 (20.02.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:

201810351166.6 2018 年 4 月 18 日 (18.04.2018) CN

(71) 申请人: 华为技术有限公司 (HUAWEI TECHNOLOGIES CO., LTD.) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(72) 发明人: 臧磊真 (ZANG, Leizhen); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。童朝柱 (TONG, Chaozhu); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。薛军 (XUE, Jun); 中

(74) 代理人: 北京同立钧成知识产权代理有限公司 (LEADER PATENT & TRADEMARK FIRM); 中国北京市海淀区西直门北大街 32 号枫蓝国际 A 座 8F-6, Beijing 100082 (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, 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,

(54) Title: SYSTEM STARTUP METHOD AND APPARATUS, ELECTRONIC DEVICE AND STORAGE MEDIUM

(54) 发明名称: 系统的启动方法、装置、电子设备和存储介质

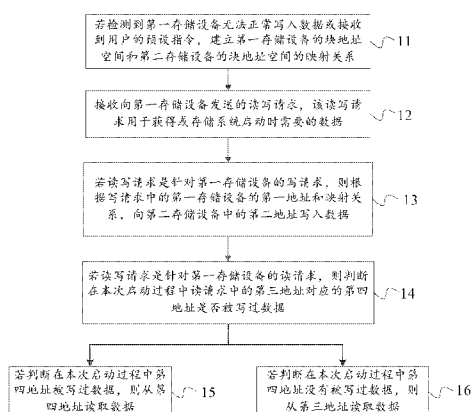


图 2

- 11 If detected that a first storage device cannot normally write in data or if a preset instruction of a user is received, establish mapping between an address space of the first storage device and an address space of a second storage device
- 12 Receive a read-write request sent to the first storage device, the read-write request being used for obtaining or storing data required when a system is started up
- 13 If the read-write request is a write request for the first storage device, write the data into a second address in the second storage device according to a first address of the first storage device in the write request and the mapping
- 14 If the read-write request is a read request for the first storage device, determine whether a fourth address corresponding to a third address in the read request during the current startup process has data written in or not
- 15 If determined that the fourth address during the current startup process has data written in, read the data from the fourth address
- 16 If determined that the fourth address during the current startup process does not have data written in, read the data from the third address

(57) Abstract: A system startup method and apparatus, an electronic device and a storage medium, the method comprising: if detected that a first storage device cannot normally write in data or if a preset instruction of a user is received, establishing mapping between an address space of the first storage device and an address space of a second storage device (11); receiving a read-write request sent to the first storage device, the read-write request being used for obtaining or storing data required when a system is started up (12); if the read-write request is a write request for the first storage device, writing the data into a second address in the second storage device according to a first address of the first storage device in the write request and the mapping (13); if the read-write request is a read request for the first storage device, determining whether a fourth address corresponding to a third address in the read request has data written in or not (14); if the fourth address has data written in, reading the data from the fourth address (15); and if the fourth address does not have data written in, reading the data from the third address (16). With the described method, normal startup of the system may be ensured.

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: 一种系统的启动方法、装置、电子设备和存储介质, 若检测到第一存储设备无法正常写入数据或接收到用户的预设指令, 建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系(11); 接收向第一存储设备发送的读写请求, 读写请求用于获得或存储系统启动时需要的数据(12); 若读写请求是针对第一存储设备的写请求, 则根据写请求中的第一存储设备的第一地址和映射关系, 向第二存储设备中的第二地址写入数据(13); 若读写请求是针对第一存储设备的读请求, 则判断读请求中的第三地址对应的第四地址是否被写过数据(14), 若第四地址被写过数据, 则从第四地址读取数据(15); 若第四地址没有被写过数据, 则从第三地址读取数据(16)。该方法可以保证系统的正常启动。

系统的启动方法、装置、电子设备和存储介质

本申请要求于 2018 年 04 月 18 日提交中国专利局、申请号为 201810351166.6、申请名称为“系统的启动方法、装置、电子设备和存储介质”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。

技术领域

本申请实施例涉及通信技术领域，尤其涉及一种系统的启动方法、装置、电子设备和存储介质。

背景技术

智能手机、电视或平板等电子设备使用嵌入式多媒体卡(embedded multimedia card; eMMC)/通用闪存存储(universal flash storage; UFS)存储器作为存储设备，eMMC/UFS 内部由控制器和 NAND 闪存(NAND Flash Storage)组成。

由于工艺结构的原因，NAND 闪存具有一定的编程擦写次数(program-erase cycle; PE)，当 NAND 闪存达到预定的 PE 次数时，NAND 闪存的寿命会耗尽，将不能再进行编程或擦除。此时，eMMC/UFS 会变为只读，或者虽能写入数据但无法保证数据的正确性。因此，当 NAND 闪存寿命耗尽时，电子设备系统将不能正常启动。对电子设备的用户来说，系统不能正常启动会导致用户无法对其重要数据进行转移。尤其是诸如通讯录，照片等，这些数据的丢失对用户的生活带来巨大的麻烦。对此现有技术中也有一些解决方案，但往往都需要将存储设备拆除后，通过第三方设备读取其中的用户数据，还需要对用户数据进行解密。这些操作对普通用户来说实施困难，需要寻求专业技术人员的帮助，这显然增加了大量的时间成本和金钱成本，影响用户体验。

发明内容

本申请提供一种系统的启动方法、装置、电子设备和存储介质，能够在存储设备发生问题时，用户不需要进行复杂的操作就能获得转移重要的用户数据的机会，提升了用户体验。

第一方面，本申请实施例提供一种系统的启动方法，包括：

若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读

请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址被写过数据，则从所述第四地址读取数据；

若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

在本申请实施例，以及下文提到的各个实施例中，所述“建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系”可以是在第一存储设备发生无法写入的情况时，临时为第一地址和第二地址建立映射关系。也可以是预先就设定好第一地址和第二地址的映射关系，然后在所述第一存储设备发生无法写入的情况时，从软件或者硬件中读取并启用该映射关系，也就是说，在本情况下，建立也包括“启用”的含义。

在上述方案中，系统启动时需要的数据包括可执行代码、系统数据和用户数据等。

在本方案中，由于在接收到向第一存储设备发送的读写请求时，若该读写请求为针对第一存储设备的写请求，则会将写请求映射至第二存储设备的第二地址，并在第二地址写入数据，这样，将会保证在系统启动过程中能够正常写入数据；若该读写请求为针对第一存储设备的读请求，则在判断出在本次启动过程中该读请求中的第三地址对应的第二存储设备的第四地址被写过数据时，从第四地址读取数据，若第四地址没有被写过数据，则将从第三地址读取数据，这样，将会保证在系统启动过程中能够读取到存储设备的数据，由于不仅能够正常的写入数据，而且能够读取到正确的数据，从而可以保证系统的正常启动，进而能够在存储设备发生问题时，用户不需要进行复杂的操作就能获得转移重要的用户数据的机会，提升了用户体验。

在一种可能的实现方式中，所述方法还包括：

建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据，包括：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

在上述方案中，在启动系统时，会在块映射设备中分配内存空间，并在该内存空间中建立位图信息，其中，在初始状态时，该位图信息中的所有标志位可以初始化为全0或者全1，位图信息中的标志位用于表示因映射第一存储设备的写请求而引起的第二存储设备各地址数据的写状态。当第二存储设备的某个第四地址被写入数据时，则块映射设备会对该第四地址对应的标志位的值进行更新。

由于位图信息用于标识与第一存储设备的地址空间对应的第二存储设备的地址空间中的各地址是否被写过数据，因此，可以通过位图信息实现快速查找和判断读写请求中第二地址是否被写过数据，这样可以快速确定出是从第一存储设备读取数据还是从第二存储设备读取数据，从而提高了数据读取的效率。

在一种可能的实现方式中，所述第一存储设备包括通用闪存存储 UFS 或嵌入式多媒体卡 eMMC。

在一种可能的实现方式中，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

在本方案中，内存可以是片内随机存取存储器，当然，也可以是其他的存储设备。比如，只要在系统支持的情况下，第二存储设备也可以是外部存储设备通过外部接口，诸如通用串行总线（universal serial bus，USB）连接到电子设备中形成的存储模块。

在一种可能的实现方式中，所述第一存储设备用于存储系统启动时需要的数据。

5 在本方案中，该系统启动时需要的数据包括可执行代码、系统数据和用户数据。

在一种可能的实现方式中，所述第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围。

在本方案中，第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围，这样，可以更容易建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

10 在一种可能的实现方式中，所述若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系，包括：

15 在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

在一种可能的实现方式中，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

20 在上述方案中，可以通过内核启动时读取并保存在系统文件 sysfs 里的 eMMC/UFS NAND 闪存的状态寄存器值判断 eMMC/UFS 是否处于只读状态，若状态寄存器值表征 NAND 闪存只读或异常，则说明 eMMC/UFS 处于只读状态，即第一存储设备无法正常写入数据；另外，若数据分区文件系统不能正常写入数据或读取值不正确，则用户数据分区的文件系统处于只读状态，即第一存储设备无法正常写入数据。

25 在一种可能的实现方式中，所述方法还包括：

根据所述数据启动卷原生服务 vold、其它原生服务、虚拟机、系统服务和锁屏应用；接收用户根据所述锁屏应用输入的密码、图案或指纹数据；

根据所述密码、图案或指纹数据，通过所述卷原生服务 vold 访问用户数据分区文件系统，使所述系统中的用户数据可解密，用户数据可访问以启动系统，启动完毕正常运行。

30 在一种可能的实现方式中，所述方法还包括：

在启动虚拟机和系统服务时，禁止对程序包进行优化和编译。

在一种可能的实现方式中，所述方法还包括：

输出提示信息，所述提示信息用于提醒用户备份数据。

第二方面，本申请实施例提供一种系统的启动装置，包括：

35 处理单元，用于在检测到第一存储设备无法正常写入数据或接收到用户的预设指令时，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

所述处理单元，用于接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

所述处理单元，还用于在所述读写请求是针对所述第一存储设备的写请求时，根据所

述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

所述处理单元，还用于在所述读写请求是针对所述第一存储设备的读请求时，判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

5 所述处理单元，还用于在判断出本次启动过程中所述第四地址被写过数据时，从所述第四地址读取数据；

所述处理单元，还用于在判断出本次启动过程中所述第四地址没有被写过数据时，从所述第三地址读取数据。

在一种可能的实现方式中，所述处理单元，具体用于：

10 建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述处理单元，还用于：

15 根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

在一种可能的实现方式中，所述第一存储设备包括通用闪存存储 UFS 或嵌入式多媒体卡 eMMC。

在一种可能的实现方式中，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

20 在一种可能的实现方式中，所述第一存储设备用于存储系统启动时需要的数据。

在一种可能的实现方式中，所述第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围。

在一种可能的实现方式中，所述处理单元，具体用于：

25 在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

在一种可能的实现方式中，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

30 在一种可能的实现方式中，所述处理单元，具体用于：

根据所述数据启动卷原生服务 vold、其它原生服务、虚拟机、系统服务和锁屏应用；接收用户根据所述锁屏应用输入的密码、图案或指纹数据；

根据所述密码、图案或指纹数据，通过所述卷原生服务 vold 访问用户数据分区文件系统，使所述系统中的用户数据可解密，用户数据可访问以启动系统，启动完毕正常运行。

35 在一种可能的实现方式中，所述处理单元，还用于在启动虚拟机和系统服务时，禁止对程序包进行优化和编译。

在一种可能的实现方式中，所述处理单元，还用于输出提示信息，所述提示信息用于提醒用户备份数据。

第三方面，本申请实施例还提供一种电子设备，该电子设备可以包括处理器，用于耦

合指令存储器，并执行所述指令存储器中的指令，所述处理器用于在所述电子设备启动的过程中基于所述指令执行下述动作：

若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

5 接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

10 若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址被写过数据，则从所述第四地址读取数据；

若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

在一种可能的实现方式中，所述处理器还用于在所述电子设备启动的过程中基于所述指令执行下述动作：

15 建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

20 所述判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据，包括：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

在一种可能的实现方式中，所述第一存储设备包括通用闪存存储 UFS 或嵌入式多媒体卡 eMMC。

25 在一种可能的实现方式中，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

在一种可能的实现方式中，所述第一存储设备用于存储系统启动时需要的数据。

在一种可能的实现方式中，所述第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围。

30 在一种可能的实现方式中，所述处理器还用于在所述电子设备启动的过程中基于所述指令执行下述动作：

在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

35 在一种可能的实现方式中，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

第四方面，本申请实施例还提供一种可读存储介质，保存有指令，所述指令用于指示所述电子设备在启动过程中执行上述第一方面中任一项所述的系统的启动方法。

第五方面，本申请实施例还提供了一种包含指令的计算机程序产品，当其在计算机上

运行时，使得计算机执行本申请实施例的第一方面提供的系统的启动方法。

第六方面，本申请实施例还提供一种芯片，芯片上存储有计算机程序，在计算机程序被处理器执行时，执行上述第一方面中提供的系统的启动方法。

在以上方面中，本申请实施例中提到的第一存储设备和第二存储设备的地址可以指的是块地址。

在以上方面中，本发明实施例提供的方法和启动装置，可以被应用于包括芯片，终端产品或服务器设备等的实体产品中。所述实体产品具有实现上述各方面或其任意可能的方式中的系统的启动方法的功能。所述功能可以通过硬件实现，也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的单元。而以上各方面提到的电子设备可以是芯片，终端产品或服务器设备等实体产品，也可以是实体产品中提供相应功能的功能模块或功能模块集合。本申请提供的系统的启动方法、装置、电子设备和存储介质，在第一存储设备无法写入数据时，通过映射的方式将本需要写入所述第一存储设备的数据写入第二存储设备，满足了系统启动的需要。对用户来说，在第一存储无法写入的情况下仍然可以正常启动系统，使得用户可以通过电子设备的系统来直接转移用户数据，保证了用户体验。

附图说明

图 1 为本申请实施例涉及电子设备的系统的示意图；

图 2 为本申请实施例提供的系统的启动方法实施例一的流程示意图；

图 3 为系统启动过程的系统模块示意图；

图 4 为启动过程的另一系统模块示意图；

图 5 为提示信息的界面示意图；

图 6 为本申请实施例提供的一种电子设备的结构示意图。

具体实施方式

以下，对本申请中的部分用语进行解释说明，以便于本领域技术人员理解。

本申请实施例以基于 Linux 内核的 Android 系统为例进行说明，但其实施并不受限于 Android 系统或 Linux 内核中，任何存在因存储器失效而无法启动的系统都可以采用本发明实施例的方法。

在基于 Linux 内核的 Android 系统中：

1) 存储设备通常为块设备，其中，块设备包括物理设备以及操作系统下该物理设备对应的数据结构(块设备结构体)。需要说明的是，本申请中的块设备包括但不限于 Android 或 Linux 的块设备，其也可以为其他系统下的块设备。

2) 块映射设备：块映射设备是操作系统中在不同块设备之间重定向和映射读写请求的块设备，在 Linux 下，块映射设备作为 Device-Mapper 设备类的一个子类型实现，Device-Mapper 是 Linux 内核中支持逻辑卷管理的通用设备映射机制，为实现用于存储资源管理的块设备驱动提供了一个高度模块化的内核架构。Device-Mapper 在内核中通过模块化的目标驱动插件实现了对块设备输入输出(input/output; I/O)请求的过滤或者重定向等

功能。

本申请实施例提供的系统的启动方法，可以适用于电子设备中，如智能手机、电视、平板和车载智能终端等，尤其适用于在电子设备的内部存储设备无法正常写入数据，如何使得系统能够正常启动的应用场景中。在现有技术中，由于工艺结构的原因，电子设备的内部存储设备具有一定的编程擦写次数，当内部存储设备达到预定的编程擦写次数时，内部存储设备的寿命会耗尽，将不能再进行编程或擦除。此时，内部存储设备会变为只读，或者虽能写入数据但无法保证数据的正确性。这样，电子设备的系统将不能正常启动，使用者无法进入系统，导致用户数据的丢失。

在Android系统，用户无法进入系统主要有两个原因：1) Android系统内核Linux启动后的用户态第一进程init会启动很多系统原生服务，原生服务写数据时，由于存储设备写入失败，文件系统检测到错误码，导致系统重启；或者原生服务检测到写失败错误码而退出，系统不能正常启动而重启；或者原生服务读取到写入的不正确数据而产生逻辑错误，使这些原生服务异常退出，导致系统会重启；2) 现在Android设备的用户数据(userdata)分区通常采用基于文件的加密(File Based Encryption, FBE)加密方式，在用户不输入屏幕解锁口令或图案或指纹的情况下，卷服务vold(是原生服务的其中一个)只能使系统设备数据可解密，但却无法使得用户数据可解密。

针对上述两个问题，1) 虽然可以通过挂载tmpfs到/data目录使系统原生服务能够写入数据，启动起系统，但却使解密相关服务vold和模块e4crypt无法访问位于原userdata分区上的密钥和数据；2) 由于数据安全原因，在用户没有解锁屏幕的情况下，无法解密用户数据，因为用户数据加密密钥的生成或访问有用户锁屏口令或图案或指纹数据参与。因此，在电子设备的内部存储设备变为只读状态，或者在用户数据分区的文件系统处于只读状态时，如何使系统能够正常启动，以保证用户的数据不会丢失，是需要解决的问题。

本申请实施例考虑到这种情况，提出一种系统的启动方法。图1所示为本申请实施例涉及电子设备的系统的示意图。所述系统包括中央处理器(central processing unit, CPU)，存储控制器，第一存储设备和内存。所述第一存储设备用于存储系统启动时需要的数据，诸如程序指令和程序工作时的参数等，通常为eMMC和UFS。所述内存为所述CPU运行时要使用的数据提供缓存，通常为各种随机存储器(random-access memory, RAM)，在当前的个人电脑或移动终端上通常会使用双倍速同步动态随机存储器(double data rate synchronous dynamic RAM, DDR SDRAM)。存储控制器则用于解码中央处理器发出的各种存储器的读写请求，其可以是如图1中所示的与所述第一存储设备和内存分离的独立器件，也可以集成在所述第一存储设备或所述内存的内部，用于专门管理对应的存储器的访问请求。本申请实施例要实现的目的就是：在启动过程中，在所述第一存储器异常时，或者说所述第一存储器为只读状态时，使得系统仍然能够被正常地启动。

如图2所示，本申请实施例提供一种系统的启动方法，该方法可以由任意执行系统的启动方法的装置来执行，该装置可以通过软件和/或硬件实现。本实施例中，该装置可以集成在电子设备中。在图1所示的电子设备的系统的基础上，如图2所示，本实施例的方法可以包括：

步骤11、若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

在本实施例中，在系统启动过程中，在进行数据的读写时，电子设备需要访问第一存储设备，并从第一存储设备中读写数据，因此，在系统启动时，电子设备中的处理器将会检测存储器件和系统状态，当检测到第一存储设备无法正常写入数据时，或者检测到接收到用户输入的预设指令时，将会建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

其中，该第一存储设备可以用于存储系统启动时需要的数据，其中，该系统启动时需要的数据包括可执行代码、系统数据和用户数据等。在终端设备中，中央处理器通常被设置在处理器芯片中，而可执行的代码、系统数据和用户数据则通常被存储在芯片外的存储器（即片外存储器）中，所述片外存储器通常为UFS或eMMC。当然，随着技术的发展，也不排除使用其他种类的存储设备作为所述第一存储设备。

另外，第二存储设备可以为图1中所示的内存虚拟出的块设备，如ramdisk。

在一种可能的方式中，第一存储设备无法正常写入数据包括：第一存储设备处于只读状态和/或用户数据分区的文件系统处于只读状态。

具体的，可以通过内核启动时读取并保存在系统文件sysfs里的eMMC/UFS NAND闪存的状态寄存器值判断eMMC/UFS是否处于只读状态，若状态寄存器值表征NAND闪存只读或异常，则说明eMMC/UFS处于只读状态，即第一存储设备无法正常写入数据；另外，若数据分区文件系统不能正常写入数据或读取值不正确，则用户数据分区的文件系统处于只读状态，即第一存储设备无法正常写入数据，此时，电子设备中的处理器将会建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

另外，启动加载程序传递给内核并由内核保存在设备文件系统devfs里的命令行，该命令行中有参数指示接收到用户的预设指令，此时，电子设备中的处理器也会建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。其中，用户的预设指令可以包括用户通过按压某个物理按键或者多个物理按键的组合触发的指令，也可以包括用户通过触摸屏幕绘制预设的图形或者字母等方式触发的指令，当然，还可以包括用户通过语音方式触发的指令等，对于用户的预设指令的具体触发方式，本申请实施例在此不做限制。

此外，本申请实施例提供的是一种启动方法，用于解决存储器失效导致系统无法启动的问题。但是在实际产品使用过程中，使用本申请的启动方法会使得第二存储设备被作为第一存储设备的替代写目标，这也带来了一个结果就是在系统启动后的用户操作都不会在第一存储设备中留下痕迹，也就是无法篡改第一存储设备的数据。这相当于让系统进入了一种“状态自清除”的状态，用户的操作痕迹会在系统再次关闭后被清除，这样做的结果是可以将第一存储器的设备从用户的“危险操作”中保护起来，在未来的可能应用环境中，通过用户主动发起的预设指令来启动本申请是有现实意义的。

进一步地，在本发明实施例中，在第一存储设备无法被写入时，所述第二存储设备作为第一存储设备替代设备来存储数据，其可以为不同于所述第一存储设备的各种存储设备，比如内存虚拟出的块设备或外部存储设备，其中，内存虚拟出的块设备为使用内存模拟出的块设备。

其中，内存例如可以包括片内随机存取存储器（random access memory，RAM）或双倍低功耗速率同步动态随机存储器（low power double data rate，LPDDR），当然，也可以是其他的内存。另外，在可能的实现方式中，只要在系统支持的情况下，第二存储设备也

可以是外部存储设备通过外部接口，诸如通用串行总线（universal serial bus, USB），连接到电子设备中形成的存储模块。

在一种可能的实现方式中，在第二存储设备是内存虚拟出的块设备，如ramdisk时，在检测到第一存储设备无法正常写入数据或接收到用户的预设指令时，Android系统中的进程init将通过内核创建第二存储设备，其中，创建的第二存储设备的块地址空间的范围大于或等于第一存储设备的块地址空间的范围。

可选的，第二存储设备是内存ramdisk块设备时，可以是Linux内核中设备名为“ramdisk”的设备实例，对应设备驱动“brd”。另外，由于第二存储设备是在有数据写入时才真正分配内存页面，而且设备驱动“brd”使用基树管理内存页面，因此页面是稀疏的，不需要为整个块地址空间连续分配内存，故不会耗尽系统内存，真正占用的内存数量是实际写入的内存页面数量。

在第二存储设备包括外部存储设备时，该外部存储设备可以包括外部SD卡、通过通用串行总线（universal serial bus; USB）接入的存储设备等。

本领域技术人员可以理解，init是Linux内核启动后运行的第一个进程，先是在内核态运行，后转换为用户态运行，本申请实施例中是指在用户态运行的进程。

在一种可能的实现方式中，在系统上电后，在启动加载程序或内核程序或初始化程序运行时，电子设备将会检测第一存储设备是否能正常写入数据或是否接收到用户的预设指令；若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

另外，可以理解的是，若检测到第一存储设备能够正常写入数据或没有接收到用户的预设指令，则将会按照现有的方式进行系统的启动。

步骤12、接收向第一存储设备发送的读写请求，该读写请求用于获得或存储系统启动时需要的数据。

在本步骤中，在系统启动过程中，电子设备中的处理器会发送读写请求，重定向逻辑模块在接收到读写请求后，将根据读写请求中携带的目标设备的设备标识判断该读写请求是发送给哪一个存储设备的，在确定出该读写请求是向第一存储设备发送的读写请求时，重定向逻辑模块会将该读写请求重定向到块映射设备。其中，该读写请求用于获取或存储系统启动时需要的数据，其中，该系统启动时需要的数据可以包括可执行代码、系统数据和用户数据等。另外，该读写请求中还携带有读写数据的目标设备的设备标识、地址信息以及所要读写的数据缓存。

步骤13、若读写请求是针对第一存储设备的写请求，则根据写请求中的第一存储设备的第一地址和映射关系，向第二存储设备中的第二地址写入数据。

在本步骤中，在接收到向第一存储设备发送的读写请求后，根据读写请求中的读写类型字段判断出该读写请求是针对第一存储设备的写请求，由于第一存储设备无法正常写入数据，则需要将写请求进行重定向。在具体的实现过程中，该写请求中包括有第一存储设备的第一地址，则根据该第一地址和建立的映射关系，通过至少一个块映射设备（device-mapper; DM）将写请求映射至第二存储设备的第二地址，并在第二存储设备的第二地址写入数据。

举例来说，若第一存储设备的块地址10、块地址20、块地址30和块地址40分别和第二

存储设备的块地址15、块地址25、块地址35和块地址45之间存在映射关系，若接收到针对第一存储设备的写请求，且该写请求中的第一地址为块地址20，则将通过至少一个块映射设备将写请求映射至第二存储设备的块地址25，并在块地址25写入数据。

5 另外，在实际应用中，进程init需要通过内核创建块映射设备，并加载块映射设备的映射功能表，块映射设备会将针对第一存储设备的写请求，映射至第二存储设备的相应地址。

在本实施例中，该块映射设备作为Device-Mapper设备类的一个子类型实现，上述块映射设备的映射功能表指示块映射设备把读写请求克隆、拆分为页面大小数据块的小读写请求，并将针对第一存储设备的写请求映射至第二存储设备。

10 值得注意的是，当第二存储设备为内存虚拟出来的块设备时，第二存储设备也可以实现在块映射设备中，作为块映射设备的一个子功能模块，来实现数据写入读取的功能。

需要进行说明的是，若第二存储设备为外部存储设备，将写请求映射至外部存储设备的第二地址，在第二地址写入数据，可以是将块设备写请求数据映射到外部存储设备的指定存储位置、指定分区或指定文件中，另外，为了提高安全性，通常需要对写入到外部存储设备的数据进行加密处理。其中，第二存储设备的功能也可以用下列方式实现：具体实
15 现可能用整个外部存储设备做第二存储设备，也有可能用其上的某个分区做第二存储设备，还有可能直接指定其上的某段地址空间做第二存储设备，还有可能用外部存储某个分区文件系统上的某个大文件做映射目标。

步骤14、若读写请求是针对第一存储设备的读请求，则判断在本次启动过程中读请求中的第三地址对应的第四地址是否被写过数据。

20 其中，若判断出在本次启动过程中第四地址被写过数据，则执行步骤15，否则，执行步骤16。

在本步骤中，在接收到向第一存储设备发送的读写请求后，若判断出该读写请求是针对第一存储设备的读请求，由于在系统启动过程中，若存在针对第一存储设备的写请求时，会将数据写入到第二存储设备，因此，在读取数据时，将需要判断与读请求中包括的第三
25 地址存在映射关系的第四地址是否被写入过数据。

在一种可能的实现方式中，电子设备中的处理器将会根据块映射设备中建立的位图信息，判断第四地址是否被写过数据，其中，该位图信息用于标识与第一存储设备的地址空间对应的第二存储设备的地址空间中的各地址是否被写过数据。

30 具体的，在启动系统时，会在块映射设备中分配内存空间，并在该内存空间中建立位图信息，其中，在初始状态时，该位图信息中的各标志位可以被初始化为全0或者全1，各标志位分别用于表示因映射第一存储设备的写请求而引起的第二存储设备各地址数据的写状态。当第二存储设备的某个第四地址被写入数据时，则块映射设备会对该第四地址对应的标志位的值进行更新。

例如：若第二存储设备包括有地址1-地址10，则位图信息的初始值可以为“0000000000”，
35 其中，位图信息中的各标志位分别用于表示地址1-地址10中数据的写状态，当标志位的值为0时，表示对应的地址没有被写入数据。若第二存储设备的地址3被写入了数据，则块映射设备会将位图信息中的第3个标志位的值更新为1，即位图信息被更新为“0010000000”。值得注意的是，位图信息的初始值也可以为“1111111111”，若第二存储设备的某个第四地址被写入了数据，则块映射设备会将位图信息中第四地址的对应标志位的值由1更新为0。

进一步地，在判断在本次启动过程中读写请求中的第三地址对应的第四地址是否被写过数据时，可以根据位图信息中第四地址的对应标志位，判断第四地址是否被写过数据。

具体的，可以根据位图信息中各标志位的值，确定第四地址是否被写入过数据。例如：若第二存储设备包括有地址1-地址10，且位图信息为“1010010001”，若假设0代表第二存储设备的第四地址没有被写过数据，1代表第二存储设备的第四地址被写过数据，则根据该位图信息，可以判断出地址1、地址3、地址6和地址10被写入过数据。

在本实施例中，由于位图信息用于标识与第一存储设备的地址空间对应的第二存储设备的地址空间中的各地址是否被写过数据，因此，可以通过位图信息实现快速查找和判断读写请求中第四地址是否被写过数据，这样可以快速确定出是从第一存储设备读取数据还是从第二存储设备读取数据，从而提高了数据读取的效率。

步骤15、若判断在本次启动过程中第四地址被写过数据，则从第四地址读取数据。

在本步骤中，若判断出在本次启动过程中第二存储设备的第四地址被写过数据，为了保证读取的数据的正确性，将从第二存储设备的第四地址读取数据。

在一种可能的实现方式中，可以通过至少一个块映射设备从第四地址读取数据。

步骤16、若判断在本次启动过程中第四地址没有被写过数据，则从第三地址读取数据。

在本步骤中，若判断出在本次启动过程中第二存储设备的第四地址没有被写过数据，则将从第一存储设备的第三地址读取数据，这样可以实现对第一存储设备的只读不写。

在一种可能的实现方式中，通过至少一个块映射设备从第一地址读取数据。通过上述各步骤，上层文件系统发送到处于物理只读状态的第一存储设备的读写请求可以正常完成，从而对文件系统及以上层透明，这样，在第一存储设备只读时，让写数据操作仍能通过磁盘或分区对应的原内核数据结构（块设备结构体）完成，从而能够保证应用读写数据成功，系统继续按正常流程启动。

在该方法中，由于在接收到向第一存储设备发送的读写请求时，若该读写请求为针对第一存储设备的写请求，则会将写请求映射至第二存储设备的第二地址，以在第二地址写入数据，这样，将会保证在系统启动过程中能够正常写入数据，若该读写请求为针对第一存储设备的读请求，则在判断出第二地址被写过数据时，从第二地址读取数据，若第二地址没有被写过数据，则将从第一地址读取数据，这样，将会保证在系统启动过程中能够读取到存储设备的数据，并使系统能够写入数据成功。由于不仅能够读取到存储设备上的数据，包括解密后的数据，而且还能够正常的写入数据并正确读取，从而可以保证系统的正常启动，使得用户能够获得可以通过电子设备的系统本身就能够转移数据，保证了用户体验。

下面，在图1所示的电子设备的系统结构的基础上，对图2中所示的数据的读取和写入的过程，进行详细说明。

图3为系统启动过程的系统模块示意图，如图3所示，电子设备在启动时，电子设备创建块映射设备（device-mapper；DM）和第二存储设备，其中，系统的主要模块包括进程init 201，原生服务 202，系统服务和应用 203，紧急备份应用 204，卷原生服务 vold 205，系统目录 206，文件系统 207，通用块层 208，bio 重定向逻辑模块 209，块映射设备 210，第一存储设备 211，第二存储设备 212 和位图信息 213。其中，上述各模块为在系统启动过程中可能会用到的模块。

其中，进程 init 201 是 Android 系统内核启动后的第一个用户态进程；块映射设备 210 是虚拟化的软件模块，其执行主体为处理器，如中央处理器 (central processing unit; CPU)，块映射设备 210 包含 bio 重定向逻辑模块 209 和位图信息 213，其中，位图信息 213 用于标识与第一存储设备 211 的地址空间对应的第二存储设备 212 的地址空间中的各地址是否被写过数据，也即位图信息 213 用于标识因映射第一存储设备 211 的写请求而引起的第二存储设备 212 各地址是否被写过数据。

系统目录 206 包括 /data 目录，还有 /system、/persist、/cache 等目录，其中，/system 目录用来挂载 system 分区，system 分区主要存储系统可执行程序。/persist 目录用来挂载 persist 分区，persist 分区主要存储电子设备的配置参数等。/cache 目录用于挂载 cache 分区，cache 分区主要用来存储系统或应用程序的缓存数据。块映射设备 210、第一存储设备 211 和第二存储设备 212 都属于块设备，位次上是平等的，没有从属关系，其中，第一存储设备 211 包括 UFS 或 eMMC，第二存储设备包括内存虚拟出的块设备或外部存储设备。文件系统 207 通过通用块层 208 访问这些块设备。

另外，在存储设备正常时，系统启动过程中需要从第一存储设备 211 中，比如 eMMC 或 UFS 中，读取数据。其中，读取的数据包括可执行代码、软硬件系统信息和系统配置等。另外，也需要在 eMMC 或 UFS 中写入数据，其中，写入的数据主要包括有系统启动状态信息、系统服务和用户程序的数据等。

在系统启动时，首先读取可执行程序二进制代码，由于可执行程序所在的存储分区是只读的，对应标志位为 0，且应用程序运行时需要使用内存中的系统数据或用户数据，而 CPU 不能直接读写第一存储设备 211 中的数据，所以只能先加载到内存中读，或将数据写到内存后，再由存储控制器将数据写到第一存储设备中，所以块映射设备指示存储控制器从第一存储设备里读取数据到内存，然后 CPU 执行内存中的程序。

当指示存储控制器从第一存储设备 211 读数据或者向第一存储设备 211 写数据时，首先会将读写请求重定向到块映射设备 210。具体的，在图 3 的基础上，若检测到第一存储设备 211 无法正常写入上述数据或接收到用户的预设指令，将建立第一存储设备 211 的块地址空间和第二存储设备 212 的块地址空间的映射关系，块映射设备 210 接收向第一存储设备 211 发送的读写请求，该读写请求用于获得或存储系统启动时需要的数据，若读写请求是针对第一存储设备 211 的写请求，即 CPU 欲向第一存储设备写入数据时，则根据写请求中的第一存储设备 211 的第一地址和映射关系，通过至少一个块映射设备 210 将写请求重定向和映射至第二存储设备 212 的第二地址，从而在第二地址写入数据。在第二地址写入数据时，块映射设备 210 会将位图信息 213 中该第二地址对应的标志位的值进行修改，以表示该第二地址被写过数据。

若读写请求是针对第一存储设备 211 的读请求，即 CPU 欲从第一存储设备 211 读取数据时，块映射设备 210 检查本次启动过程中是否有对对应块的地址发起过写操作，若对该对应块地址发起过写操作，则块映射设备 210 指示从该地址在第二存储设备 212 中的映射地址读取数据，若未对该对应块地址发起过写操作，则块映射设备 210 指示存储控制器从第一存储设备 211 的该块地址中读取数据到内存，具体的，可以通过判断在本次启动过程中读请求中的第一存储设备 211 的第三地址，以及位图信息 213 中对应标志位的值，判断第二存储设备 212 的第四地址是否被写过数据；若在本次启动过程中第四地址被写过数据，则从

第四地址读取数据；若在本次启动过程中第四地址没有被写过数据，则从第三地址读取数据。

进一步地，上述的读写请求可以为bio结构体或request结构体。bio结构体是描述读写请求从文件系统207派发到通用块层208的数据结构，request结构体是描述从通用块层208到块映射设备210的请求队列的数据结构。

其中，bio结构体是Linux通用块层中用来描述单一的I/O请求的数据结构，记录了一次输入输出（input/output；I/O）请求所必需的相关信息，主要包括该I/O请求是读请求还是写请求、I/O请求的目标块设备、I/O请求的数据缓存位置，I/O请求的块设备起始扇区等。

request结构体用于描述提交给块映射设备的I/O请求，一个request结构体通常会包含若干地址连续的bio结构体。

在一种可能的实现方式中，以读写请求为bio结构体为例，将读写请求重定向到块映射设备的过程，可以通过如下伪码描述实现：

```
if (bio->bi_bdev->bd_contains == dm->origin_bdev) {
    bio->bi_bdev = dm->bdev;
    generic_make_request(bio);
}
```

另外，通过至少一个块映射设备将读写请求映射至第二存储设备的第二地址的过程，可以通过如下伪码描述实现：

```
chunk_t chunk = sector_to_chunk(bio->bi_iter.bi_sector);
redow_fix_bio_cieinfo(bio);

if (test_bit(chunk, dm->wr_bitmap)) {
    bio->bi_bdev = dm->cow->bdev;
    return DM_MAPIO_REMAPPED;
}

if (bio_rw(bio) == WRITE) {
    set_bit(chunk, dm->wr_bitmap);
    bio->bi_bdev = dm->cow->bdev;
    return DM_MAPIO_REMAPPED;
} else {
    bio->bi_bdev = dm->origin->bdev;
    dm->origin_mrfn(bdev_get_queue(bio->bi_bdev), bio);
    return DM_MAPIO_SUBMITTED;
}
```

进一步地，下面，将分别针对读写请求为bio结构体或request结构体时，如何将写请求映射至第二存储设备的第二地址的过程进行详细说明。

在写请求为bio结构体时，继续参照图3所示。需要说明的是，在Linux内核中，第一存储设备211的请求队列位于通用块层208中，文件系统207通过通用块层208向第一存储设备

211提交bio结构体。块映射设备210接管第一存储设备211的请求队列的入口，以使内核发送到第一存储设备211的新的写请求重定向到块映射设备210自身，在新的bio结构体到达时，块映射设备210克隆并拆分带数据的bio结构体为页面大小的bio结构体，把bio写结构体映射到第二存储设备的第二地址。在实际应用中，由于4K是内存页面大小和绝大多数访问存储器的最小单位，块映射设备210会将带数据的bio结构体克隆拆分成一系列小的4K的bio结构体，每个4K的bio结构体对应位图信息中的一个位，即，一个位图信息位对应一个4K大小粒度的块数据。

在写请求为request结构体时，第二存储设备的驱动程序支持request结构体和相关接口，块映射设备和第一存储设备将通过request结构体和接口进行访问。需要说明的是，在Linux内核中，文件系统207通过通用块层208向块设备的请求队列提交request结构体请求。块映射设备210接管第一存储设备211的请求队列的后端接口，使用request结构体进行访问；请求队列的后端接口根据队列类型可以使request_fn函数或多队列操作结构体的queue_fn函数，以使内核发送到第一存储设备211的新的写请求映射至第二存储设备的第二地址。

值得注意的是，在读写请求为bio结构体时，首先检查bio结构体的数据扇区是否块对齐，并检查块映射设备自身是否有效和激活，若bio结构体的数据扇区块对齐，且块映射设备自身有效且被激活，则会分别处理REQ_FLUSH bio结构体和REQ_DISCARD bio结构体，其中，针对REQ_FLUSH bio结构体请求，直接重定向到第二存储设备进行处理；针对REQ_DISCARD bio结构体请求，将位图信息的对应标志位的值设置为1，把bio结构体请求映射到第二存储设备进行处理。

在读写请求为request结构体时，电子设备中的处理器检查request结构体的数据扇区是否块对齐，并检查块映射设备自身是否有效和激活，若request结构体的数据扇区块对齐，且块映射设备自身有效且被激活，则会分别处理REQ_FLUSH request结构体请求和REQ_DISCARD request结构体请求。其中，针对REQ_FLUSH request结构体请求，直接重定向到第二存储设备进行处理；针对REQ_DISCARD request结构体请求，将位图信息的对应标志位的值设置为1，把request结构体请求映射到第二存储设备进行处理。

值得注意的是，针对REQ_FLUSH和REQ_DISCARD bio结构体或request结构体的处理，在第二存储设备为外部存储设备时，需要将该bio结构体或request结构体映射给外部存储设备，由外部存储设备的内部控制器进行进一步处理。

下面，将分别针对读请求为bio结构体或request结构体时，对于不同结构体的读请求进行处理的过程进行详细说明。

对 bio 结构体读请求，若位图信息对应标志位的值为 1，则映射为从第二存储设备读取写过的数据，若位图信息中对应标志位的值为 0，则映射为从第一存储设备中读取原有的数据。

在读写请求为 request 结构体时，若判断出 request 结构体所要读取的数据块全部被修改过，即全部被写入过第二存储设备，则映射到第二存储设备读取写过的最新数据；若全部没有被修改过，即都没有被写入过第二存储设备，则映射到第一存储设备中读取原有数据；若部分修改过，即所要读取的数据中有一部分被写入到第二存储设备，有一部分没有被写过，则克隆并拆分 request 结构体，并发送到第一存储设备，以读取没有被写过的数据，并从第二存储设备读取修改过的块的最新数据，即读取写入到第二存储设备的数据。

更加详细地,对 request 结构体读请求,若该读请求中第一地址对应的第二地址的数据块在位图信息的对应标志位的值全为 1,表示整个读请求所要读取的数据全部被修改过,则映射到第二存储设备读取写过的数据;若位图信息的对应标志位的值全为 0,表示整个读请求所要读取的数据全部没有被修改过,则映射为从第一存储设备中读取原有的数据;

5 若位图信息的对应标志位的值部分为 0 部分为 1,则克隆并拆分 request 结构体请求为对应标志位为全 0 和全 1 的子 request 结构体请求,将对应标志位为全 0 的子 request 结构体请求映射到第一存储设备读取原有的数据,将对应标志位为全 1 的子 request 结构体请求从第二存储设备读取修改过的块的最新数据。

进一步说明的是,本实施例中第二存储设备可以不支持 request 结构体及其接口,而是

10 将处理 request 结构体的功能作为一个子功能包含在块映射设备的内部,通过 bio 结构体的 bio_vec 结构体数组或者逻辑扇区 sector 直接访问。

可选的,在进行数据的读写操作后,进程init201继续执行,并将根据读写的数据依次启动卷原生服务vold205、原生服务202、虚拟机、系统服务和应用203,在系统正常启动后,可以通过紧急备份应用204向用户输出提示信息,以提醒用户进行数据的备份。

15 进一步地,在块映射设备的数量为至少两个时,至少两个块映射设备分别与多个第一存储设备或第一存储设备的多个分区之间存在对应关系,与一个或多个第二存储设备之间存在对应关系。

具体的,图 4 为启动过程的另一系统模块示意图,图 4 与图 3 的区别在于,在块映射设备的数量为至少两个(如图 4 中的 DM-1、DM-2、DM-3……DM-N)时,各块映射设备

20 分别与第一存储设备中的多个可写分区之间存在对应关系。具体的,电子设备在启动时,系统的主要软件模块包括进程 init 301,原生服务 302,系统服务和应用 303,紧急备份应用 304,卷原生服务 vold 305,系统目录 306,文件系统 307,通用块层 308,块映射设备(块映射设备)多个,只以其中 4 个为例进行说明,分别为块设备 DM-1 309,块设备 DM-2 310,块设备 DM-3 311,块设备 DM-N 312,第二存储设备 317 以及第一存储设备(NAND 闪存)上的多个可写分区,只以其中 4 个为例进行说明,分别为 persist 分区 313,cache 分区 314,oeminfo 分区 315 和 userdata 分区 316。

25

在系统启动时,进程 init 301 把块映射设备挂载到系统目录 306 中的对应项上。如块设备 DM-1 309 挂载到/persist 上,块设备 DM-2 310 挂载到/cache 上,块设备 DM-3 311 不被挂载,直接以原始文件系统格式通过/dev/block/dm-3 设备访问,块设备 DM-N 挂载到/data

30 上。

块映射设备和系统的第一存储设备的可写分区对应,如块设备 DM-1 309 把读写请求映射到 persist 分区 313 和第二存储设备 317,块设备 DM-3 311 把读写请求映射到 oeminfo 分区 315 和第二存储设备 317,块设备 DM-N 312 把读写请求映射到 userdata 分区 316 和第二存储设备 317。

35 进一步说明的是,本实施例中的块映射设备映射读写请求到第一存储设备的各可写分区和第二存储设备 317 的映射方式,与只有一个块映射设备时,映射读写请求到第一存储设备和第二存储设备时的方式类似,不同之处在于,在创建每个块映射设备时,在其内部都会建立位图信息,每个块映射设备根据其内部的位图信息,按照对应第一存储设备的可写分区的线性地址映射到内存第二存储设备。

具体的，以块设备 DM-3 311 为例说明，把读写请求映射到 oeminfo 分区 315 和第二存储设备 317。当读写请求到达时，块设备 DM-3 311 克隆并拆分带数据的读写请求为页面大小的读写请求，并把写请求映射到第二存储设备 317，把读请求映射到第二存储设备 317 读取写过的最新数据，或从第一存储设备的 oeminfo 分区 315 读取原有数据，以实现对第一存储设备的只读不写。

可以理解的是，在读写数据后，电子设备将根据上述数据启动卷原生服务 vold、其它原生服务、虚拟机、系统服务和锁屏应用，并接收用户根据锁屏应用输入的密码、图案或指纹数据，然后根据密码、图案或指纹数据，通过卷原生服务 vold 访问用户数据分区文件系统，使系统中的用户数据可解密，用户数据可访问以启动系统，启动完毕正常运行。

具体的，在读写数据后，进程init继续执行，并按正常流程挂载各分区到文件系统节点，即系统目录中，然后按初始化脚本启动系统原生服务，卷原生服务vold和解密模块e4crypt按正常流程，首先从系统目录中的/data目录访问系统userdata分区，使系统和设备数据可解密，其他原生服务可以按正常流程从/data目录写入或读取数据。这样，系统仍然可以按正常流程启动，从而使设备数据可解密，使得进程init及其所启动的原生服务能正常读写和运行。

然后，会继续启动Java虚拟机和系统服务SystemServer，需要进行说明的是，在第一存储设备无法正常写入数据的状态下，会指示系统禁止对程序包进行优化和编译，以节省系统内存使用量。

接着，系统会启动锁屏应用，用户将通过锁屏应用输入密码、图案或指纹数据，电子设备在接收到用户根据锁屏应用输入的密码、图案或指纹数据后，将根据该密码、图案或指纹数据通过卷原生服务vold和解密模块e4crypt再次按正常流程从系统目录中的/data目录访问用户数据（userdata）分区文件系统，使系统中的用户数据可解密，用户数据可访问以启动系统，启动完毕正常运行。由于系统在用户输入输入密码、图案或指纹数据，解锁屏幕后，仍是按正常流程运行，从而让解密相关原生服务vold和解密模块e4crypt仍能从/data目录解密出位于用户数据（userdata）分区上的设备和用户数据。

进一步地，在系统正常启动后，电子设备将会输出提示信息，该提示信息用于提醒用户备份数据。

具体的，图 5 为提示信息的界面示意图，如图 5 所示，为了保证用户数据不丢失，在系统正常启动后，将会探测并准备或提示用户准备外部存储设备，以提醒用户系统存储异常，需要紧急备份或导出数据到外部存储，其中，紧急备份应用作为屏幕解锁后的第一界面，将会提醒用户进行数据备份，这样，用户将会根据自己的需求，操作紧急备份应用，以将数据导出到外部存储设备中，从而可以保证数据不丢失。

需要进一步说明的是，紧急备份应用实现为系统中的一个核心应用，在第一存储设备正常时，即系统正常运行时紧急备份应用不可见，其仅在第一存储设备无法正常写入数据或接收到用户的预设指令时才会启动运行。该紧急备份应用以在第一存储设备无法正常写入数据的情况下紧急导出数据为目的，还具有以下特征：1）不会将数据备份到第一存储设备，即 NAND 闪存或 eMMC 中，因为其已经处于异常状态；2）自带系统和用户应用程序管理模块，自带后台运行应用管理和内存清理的功能以保证系统有足够空闲内存，优先级设置高于普通应用。

在可选择的实施方式中，本申请实施例中的很多步骤都可以采用软件或硬件两种方式来实现，比如上文提到的第一存储设备和第二存储设备间的地址映射，位图信息的管理，以及各种条件上的判断。软件实现方案通过编程手段就能简单实现，在硬件实现方面，举例来说，位图信息在建立之初可以在内存中划定存储空间，然后通过硬件内部的长度寄存器和地址寄存器来记录该存储空间的起始地址和长度信息，从而在运行时硬件就可以直接通过地址寄存器和长度寄存器中获取位图信息的地址，并访问位图数据，而不需要通过软件调用；地址映射则可以通过为第一地址设置一个或多个寄存器，将这—个或多个寄存器的值组合起来映射对应的第二地址；而条件判断则可以通过 FPGA（field-programmable gate array，现场可编程阵列）来实现。也就是说，本申请实施例的方法或者下文提供的装置设备的方案、动作特征可以完全由软件来实现，也可以单独由硬件实体来实现，更或者是软件和硬件结合的方式。

本实施例提供的系统的启动方法，在检测到第一存储设备无法正常写入数据或接收到用户的预设指令时，将建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系，接收向第一存储设备发送的读写请求，读写请求用于获得或存储系统启动时需要的数据，若读写请求是针对第一存储设备的写请求，则根据写请求中的第一存储设备的第一地址和映射关系，向第二存储设备中的第二地址写入数据，若读写请求是针对第一存储设备的读请求，则判断在本次启动过程中读请求中的第三地址对应的第四地址是否被写过数据；若判断出在本次启动过程中第四地址被写过数据，则从第四地址读取数据；若判断出在本次启动过程中第四地址没有被写过数据，则从第三地址读取数据。由于在接收到向第一存储设备发送的读写请求时，若该读写请求为针对第一存储设备的写请求，则会将写请求映射至第二存储设备的第二地址，以在第二地址写入数据，这样，将会保证在系统启动过程中能够正常写入数据，若该读写请求为针对第一存储设备的读请求，则在判断出第四地址被写过数据时，从第四地址读取数据，若第四地址没有被写过数据，则从第三地址读取数据，这样，将会保证在系统启动过程中能够读取到存储设备的数据，并使系统能够写入数据成功。由于不仅能够读取到存储设备上的数据，包括解密后的数据，而且还能够正常的写入数据并正确读取，从而可以保证系统的正常启动，使得用户能够获得可以通过电子设备的系统本身就能够转移数据，保证了用户体验。

值得注意的是，本申请中的系统的启动方法，不限于 Android 系统或 Linux 内核下，在其他系统中，上述启动方法也同样适用。

为了实现图 3 所示的方法，本申请还提供一种系统的启动装置，该系统的启动装置可以包括：

处理单元，用于在检测到第一存储设备无法正常写入数据或接收到用户的预设指令时，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系；

所述处理单元，还用于接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

所述处理单元，还用于在所述读写请求是针对所述第一存储设备的写请求时，根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

所述处理单元，还用于在所述读写请求是针对所述第一存储设备的读请求时，判断在

本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

所述处理单元，还用于在判断出本次启动过程中所述第四地址被写过数据时，从所述第四地址读取数据；

5 所述处理单元，还用于在判断出本次启动过程中所述第四地址没有被写过数据时，从所述第三地址读取数据。

可选的，所述处理单元，具体用于：

建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

10 若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述处理单元，还用于：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

可选的，所述第一存储设备包括通用闪存存储UFS或嵌入式多媒体卡eMMC。

可选的，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

15 可选的，所述第一存储设备用于存储系统启动时需要的数据。

可选的，所述第二存储设备的块地址空间的范围大于或等于所述第一存储设备的块地址空间的范围。

可选的，在所述块映射设备的数量为至少两个时，各所述块映射设备分别与所述第一存储设备中的多个可写分区之间存在对应关系。

20 可选的，读写请求分别包括bio结构体或request结构体。

可选的，所述处理单元，具体用于：

在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

25 若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

可选的，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

可选的，所述处理单元，具体用于：

30 根据所述数据启动卷原生服务vold、其它原生服务、虚拟机、系统服务和锁屏应用；接收用户根据所述锁屏应用输入的密码、图案或指纹数据；

根据所述密码、图案或指纹数据，通过所述卷原生服务vold访问用户数据分区文件系统。

可选的，所述处理单元，还用于在启动虚拟机和系统服务时，禁止对程序包进行优化和编译。

35 可选的，所述处理单元，还用于输出提示信息，所述提示信息用于提醒用户备份数据。

本申请实施例所示的系统的启动装置，可以执行上述任一项实施例所示的系统的启动方法的技术方案，其实现原理以及有益效果类似，此处不再进行赘述。

需要说明的是，应理解以上装置的各个单元的划分仅仅是一种逻辑功能的划分，实际实现时可以全部或部分集成到一个物理实体上，也可以物理上分开。且这些单元可以全部

以软件通过处理元件调用的形式实现；也可以全部以硬件的形式实现；还可以部分单元通过软件通过处理元件调用的形式实现，部分单元通过硬件的形式实现。例如，处理单元可以为单独设立的处理元件，也可以集成在该电子设备的某一个芯片中实现，此外，也可以以程序的形式存储于电子设备的存储器中，由该电子设备的某一个处理元件调用并执行该处理单元的功能。其它单元的实现与之类似。此外这些单元全部或部分可以集成在一起，也可以独立实现。这里所述的处理元件可以是一种集成电路，具有信号的处理能力。在实现过程中，上述方法的各步骤或以上各个单元可以通过处理器元件中的硬件的集成逻辑电路或者软件形式的指令完成。

以上这些单元可以是被配置成实施以上方法的一个或多个集成电路，例如：一个或多个特定集成电路（application specific integrated circuit, ASIC），或，一个或多个微处理器（digital signal processor, DSP），或，一个或者多个现场可编程门阵列（field programmable gate array, FPGA）等。再如，当以上某个单元通过处理元件调度程序的形式实现时，该处理元件可以是通用处理器，例如中央处理器（central processing unit, CPU）或其它可以调用程序的处理器。再如，这些单元可以集成在一起，以片上系统（system-on-a-chip, SOC）的形式实现。

图6为本申请实施例提供的一种电子设备100的结构示意图。如图6所示，该电子设备100包括：处理器110和指令存储器120。所述处理器110用于耦合指令存储器120，并执行所述指令存储器120中的指令。所述处理器可以为图1中的CPU、所述指令存储器可以为图1中的内存等。

所述处理器110用于在所述电子设备100启动的过程中基于所述指令执行下述动作：

若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系；

接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址被写过数据，则从所述第四地址读取数据；

若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

可选的，所述处理器还用于在所述电子设备启动的过程中基于所述指令执行下述动作：

建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据，包括：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

可选的，所述第一存储设备包括通用闪存存储UFS或嵌入式多媒体卡eMMC。

可选的, 所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

可选的, 所述第一存储设备用于存储系统启动时需要的数据。

可选的, 所述第二存储设备的块地址空间的范围大于或等于所述第一存储设备的块地址空间的范围。

5 可选的, 在所述块映射设备的数量为至少两个时, 各所述块映射设备分别与所述第一存储设备中的多个可写分区之间存在对应关系。

可选的, 读写请求分别包括bio结构体或request结构体。

可选的, 所述处理器还用于在所述电子设备启动的过程中基于所述指令执行下述动作:

10 在系统上电后, 在启动加载程序或内核程序或初始化程序运行时, 检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令;

若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令, 建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

可选的, 所述第一存储设备无法正常写入数据包括: 所述第一存储设备处于只读状态; 和/或用户数据分区的文件系统处于只读状态。

15 可选的, 所述处理器110还用于在所述电子设备100启动的过程中基于所述指令执行下述动作:

根据所述数据启动卷原生服务vold、其它原生服务、虚拟机、系统服务和锁屏应用;

接收用户根据所述锁屏应用输入的密码、图案或指纹数据;

20 根据所述密码、图案或指纹数据, 通过所述卷原生服务vold访问用户数据分区文件系统。

可选的, 所述处理器110还用于在所述电子设备100启动的过程中基于所述指令执行下述动作:

在启动虚拟机和系统服务时, 禁止对程序包进行优化和编译。

可选的, 所述处理器110还用于执行下述动作:

25 输出提示信息, 所述提示信息用于提醒用户备份数据。

另外, 以上各个单元的部分或全部也可以通过集成电路的形式内嵌于该电子设备的某一个芯片上来实现。且它们可以单独实现, 也可以集成在一起。即以上这些单元可以被配置成实施以上方法的一个或多个集成电路, 例如: 一个或多个特定集成电路(application specific integrated circuit, ASIC), 或, 一个或多个微处理器(digital signal processor, DSP),
30 或, 一个或者多个现场可编程门阵列(field programmable gate array, FPGA)等。

在实际产品中, 第二存储设备可能会是内存虚拟出的块设备或外部存储设备, 而所述指令也通常会在执行时被写入内存中。也就是说, 第二存储设备和指令存储器在实际产品中可能会处于同一个硬件实体里, 也就是位于内存里。但这并不会发生问题, 因为在启动过程中需要向第一存储设备写入的数据, 往往是少量的改动字节, 即使映射到内存中也不会占用太多空间。而在本申请实施例中将所述内存中的地址空间拆分为第二存储设备和指令存储器也仅仅是为了描述方便, 其二者可以如上文所述那样处于同一个硬件实体中, 也可以处于不同的硬件实体中, 这都不应该构成本申请实施的限制。

在可选择的实施方案中, 指令存储器也可以被视作用于保存计算机程序指令的存储器, 也就是程序指令被写入内存之前所处的存储器, 也就是第一存储设备。也就是说, 指令存

储器和第一存储设备可以为统一的实体，虽然本申请实施例同时提到了指令存储器和第一存储设备，但这仅仅是为了从应用可能上说明的方便，二者属于同一实体并不会导致本领域技术人员搞不清楚本申请技术方案。

5 本申请还提供一种可读存储介质，保存有指令，所述指令用于指示所述电子设备在启动过程中执行下列动作：

若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系；

接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

10 若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址是否被写过数据，则从所述第四地址读取数据；

15 若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

可选的，所述指令还用于指示所述电子设备在启动过程中执行下列动作：

建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

20 若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据，包括：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

可选的，所述第一存储设备包括通用闪存存储UFS或嵌入式多媒体卡eMMC。

25 可选的，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

可选的，所述第一存储设备用于存储系统启动时需要的数据。

可选的，所述第二存储设备的块地址空间的范围大于或等于所述第一存储设备的块地址空间的范围。

30 可选的，在所述块映射设备的数量为至少两个时，各所述块映射设备分别与所述第一存储设备中的多个可写分区之间存在对应关系。

可选的，读写请求分别包括bio结构体或request结构体。

可选的，所述指令还用于指示所述电子设备在启动过程中执行下列动作：

在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

35 若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的块地址空间和第二存储设备的块地址空间的映射关系。

可选的，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

可选的，所述指令还用于指示所述电子设备在启动过程中执行下列动作：

根据所述数据启动卷原生服务vold、其它原生服务、虚拟机、系统服务和锁屏应用；
接收用户根据所述锁屏应用输入的密码、图案或指纹数据；
根据所述密码、图案或指纹数据，通过所述卷原生服务vold访问用户数据分区文件系统。

- 5 可选的，所述指令还用于指示所述电子设备在启动过程中执行下列动作：
在启动虚拟机和系统服务时，禁止对程序包进行优化和编译。
可选的，所述指令还用于指示所述电子设备在启动过程中执行下列动作：
输出提示信息，所述提示信息用于提醒用户备份数据。

- 10 本申请还提供一种程序产品，该程序产品包括计算机程序（即执行指令），该计算机程序存储在可读存储介质中。电子设备的至少一个处理器可以从可读存储介质读取该计算机程序，至少一个处理器执行该计算机程序使得电子设备实施前述各种实施方式提供的系统的启动方法。

- 15 本申请实施例还提供了一种系统的启动装置，包括至少一个存储元件和至少一个处理元件、所述至少一个存储元件用于存储程序，该程序被执行时，使得所述系统的启动装置执行上述任一实施例中的电子设备的操作。

- 20 实现上述各方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成。前述的程序可以存储于一可读取存储器中。该程序在执行时，执行包括上述各方法实施例的步骤；而前述的存储器（存储介质）包括：只读存储器（英文：read-only memory，ROM）、RAM、快闪存储器、硬盘、固态硬盘、磁带（magnetic tape）、软盘（floppy disk）、光盘（optical disc）及其任意组合。

权 利 要 求 书

1、一种系统的启动方法，其特征在于，包括：

若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

5 接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

10 若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址被写过数据，则从所述第四地址读取数据；

若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

2、根据权利要求1所述的方法，其特征在于，所述方法还包括：

15 建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据，包括：

20 根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

3、根据权利要求1或2所述的方法，其特征在于，所述第一存储设备包括通用闪存存储 UFS 或嵌入式多媒体卡 eMMC。

4、根据权利要求1-3任一项所述的方法，其特征在于，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

25 5、根据权利要求1所述的方法，其特征在于，所述第一存储设备用于存储系统启动时需要的数据。

6、根据权利要求1-5任一项所述的方法，其特征在于，所述第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围。

30 7、根据权利要求1-6任一项所述的方法，其特征在于，所述若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系，包括：

在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

35 若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

8、根据权利要求1-7任一项所述的方法，其特征在于，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

9、根据权利要求 1-8 任一项所述的方法，其特征在于，所述方法还包括：
根据所述数据启动卷原生服务 vold、其它原生服务、虚拟机、系统服务和锁屏应用；
接收用户根据所述锁屏应用输入的密码、图案或指纹数据；
根据所述密码、图案或指纹数据，通过所述卷原生服务 vold 访问用户数据分区文件系

5 统。

10、根据权利要求 1-9 任一项所述的方法，其特征在于，所述方法还包括：
输出提示信息，所述提示信息用于提醒用户备份数据。

11、一种系统的启动装置，其特征在于，包括：

处理单元，用于在检测到第一存储设备无法正常写入数据或接收到用户的预设指令时，
10 建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

所述处理单元，用于接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

所述处理单元，还用于在所述读写请求是针对所述第一存储设备的写请求时，根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；
15

所述处理单元，还用于在所述读写请求是针对所述第一存储设备的读请求时，判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

所述处理单元，还用于在判断出本次启动过程中所述第四地址被写过数据时，从所述第四地址读取数据；

20 所述处理单元，还用于在判断出本次启动过程中所述第四地址没有被写过数据时，从所述第三地址读取数据。

12、根据权利要求 11 所述的装置，其特征在于，所述处理单元，具体用于：

建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

25 若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述处理单元，还用于：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

30 13、根据权利要求 11 或 12 所述的装置，其特征在于，所述第一存储设备包括通用闪存存储 UFS 或嵌入式多媒体卡 eMMC。

14、根据权利要求 11-13 任一项所述的装置，其特征在于，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

15、根据权利要求 11 所述的装置，其特征在于，所述第一存储设备用于存储系统启动时需要的数据。

35 16、根据权利要求 11-15 任一项所述的装置，其特征在于，所述第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围。

17、根据权利要求 11-16 任一项所述的装置，其特征在于，所述处理单元，具体用于：

在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

18、根据权利要求 11-17 任一项所述的装置，其特征在于，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

19、根据权利要求 11-18 任一项所述的装置，其特征在于，所述处理单元，具体用于：根据所述数据启动卷原生服务 vold、其它原生服务、虚拟机、系统服务和锁屏应用；接收用户根据所述锁屏应用输入的密码、图案或指纹数据；根据所述密码、图案或指纹数据，通过所述卷原生服务 vold 访问用户数据分区文件系统。

20、根据权利要求 11-19 任一项所述的装置，其特征在于，所述处理单元，还用于输出提示信息，所述提示信息用于提醒用户备份数据。

21、一种电子设备，其特征在于，包括处理器，用于耦合指令存储器，并执行所述指令存储器中的指令，所述处理器用于在所述电子设备启动的过程中基于所述指令执行下述动作：

若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址被写过数据，则从所述第四地址读取数据；

若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

22、根据权利要求 21 所述的电子设备，其特征在于，所述处理器还用于在所述电子设备启动的过程中基于所述指令执行下述动作：

建立位图信息；所述位图信息用于标识与所述第一存储设备的地址空间对应的所述第二存储设备的地址空间中的各地址是否被写过数据；

若向所述第二存储设备的第四地址写入数据，则对所述位图信息中所述第四地址对应的标志位的值进行更新；

所述判断在本次启动过程中所述读写请求中的第三地址对应的第四地址是否被写过数据，包括：

根据所述位图信息中所述第四地址的对应标志位，判断所述第四地址是否被写过数据。

23、根据权利要求 21 或 22 所述的电子设备，其特征在于，所述第一存储设备包括通用闪存存储 UFS 或嵌入式多媒体卡 eMMC。

24、根据权利要求 21-23 任一项所述的电子设备，其特征在于，所述第二存储设备包括内存虚拟出的块设备或外部存储设备。

25、根据权利要求 21 所述的电子设备，其特征在于，所述第一存储设备用于存储系

统启动时需要的数据。

26、根据权利要求 21-25 任一项所述的电子设备，其特征在于，所述第二存储设备的地址空间的范围大于或等于所述第一存储设备的地址空间的范围。

5 27、根据权利要求 21-26 任一项所述的电子设备，其特征在于，所述处理器还用于在所述电子设备启动的过程中基于所述指令执行下述动作：

在系统上电后，在启动加载程序或内核程序或初始化程序运行时，检测所述第一存储设备是否正常写入数据或是否接收到用户的预设指令；

若检测到所述第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系。

10 28、根据权利要求 21-27 任一项所述的电子设备，其特征在于，所述第一存储设备无法正常写入数据包括：所述第一存储设备处于只读状态；和/或用户数据分区的文件系统处于只读状态。

29、一种可读存储介质，保存有指令，所述指令用于指示电子设备在启动过程中执行下列动作：

15 若检测到第一存储设备无法正常写入数据或接收到用户的预设指令，建立第一存储设备的地址空间和第二存储设备的地址空间的映射关系；

接收向所述第一存储设备发送的读写请求，所述读写请求用于获得或存储系统启动时需要的数据；

20 若所述读写请求是针对所述第一存储设备的写请求，则根据所述写请求中的第一存储设备的第一地址和所述映射关系，向所述第二存储设备中的第二地址写入数据；

若所述读写请求是针对所述第一存储设备的读请求，则判断在本次启动过程中所述读请求中的第三地址对应的第四地址是否被写过数据；

若判断在本次启动过程中所述第四地址被写过数据，则从所述第四地址读取数据；

25 若判断在本次启动过程中所述第四地址没有被写过数据，则从所述第三地址读取数据。

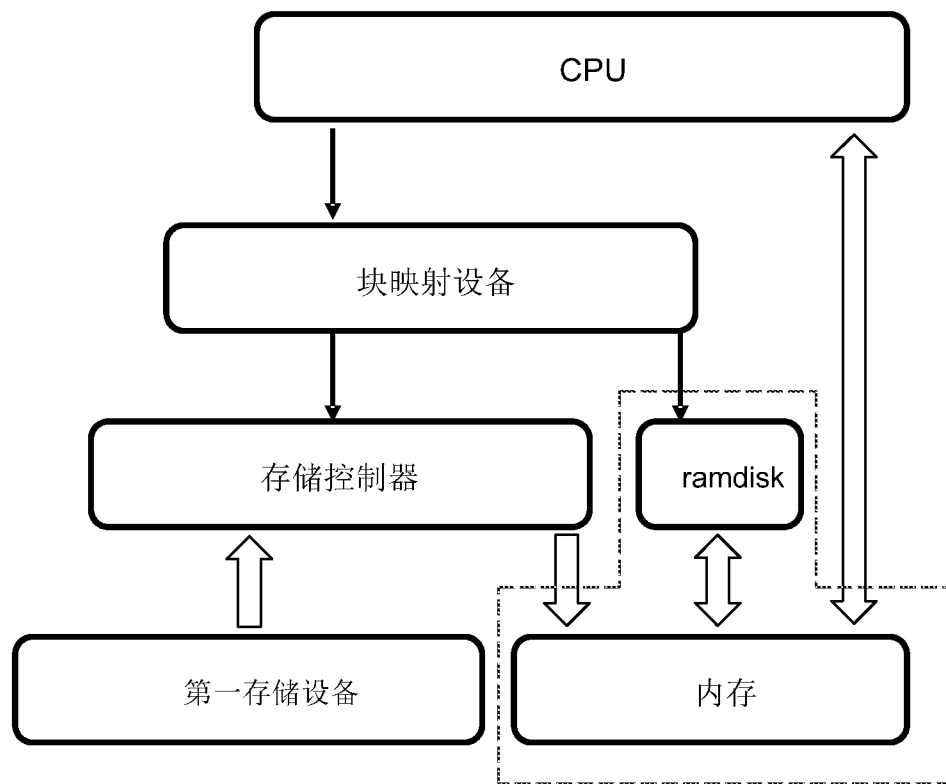


图 1

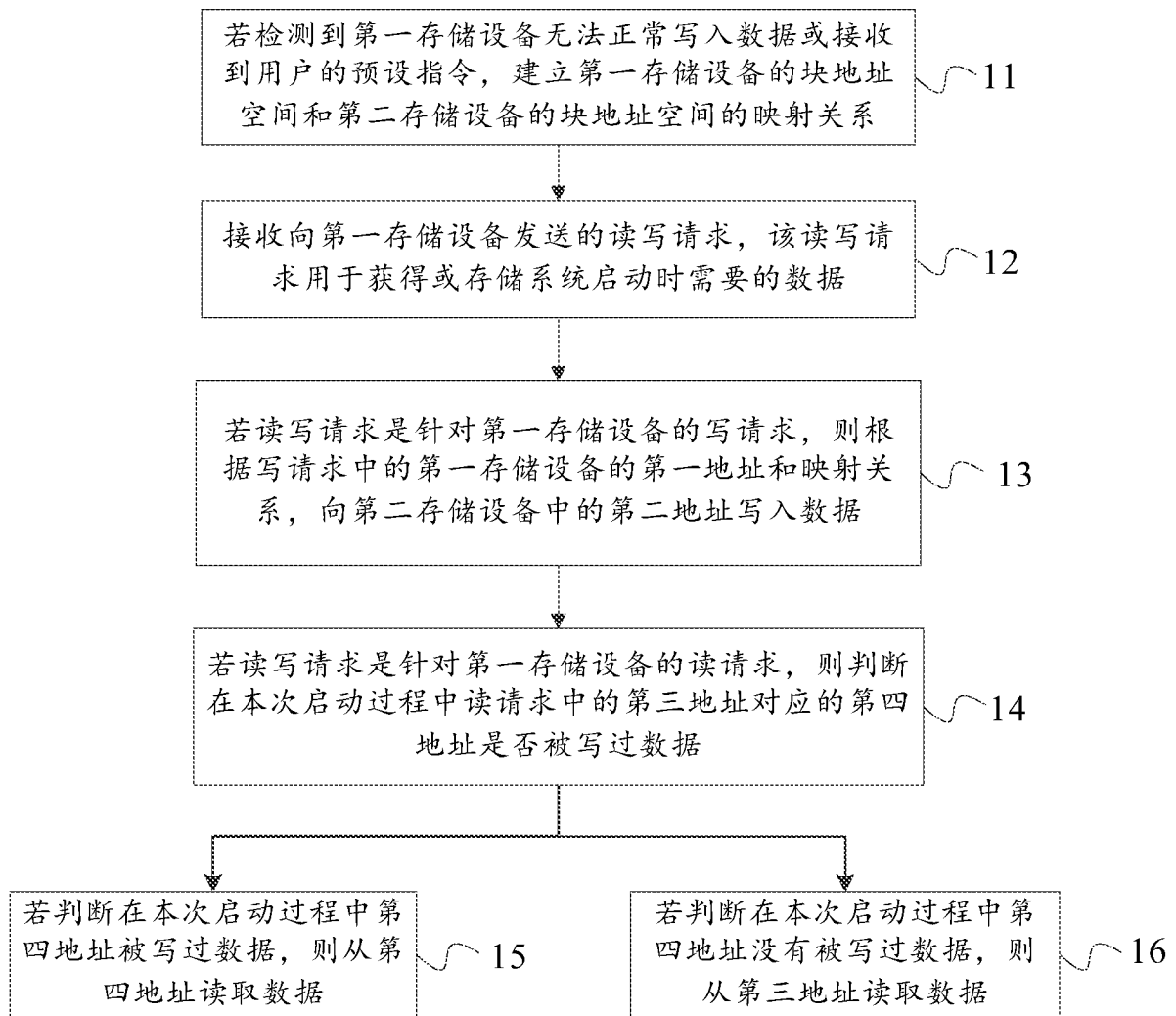


图 2

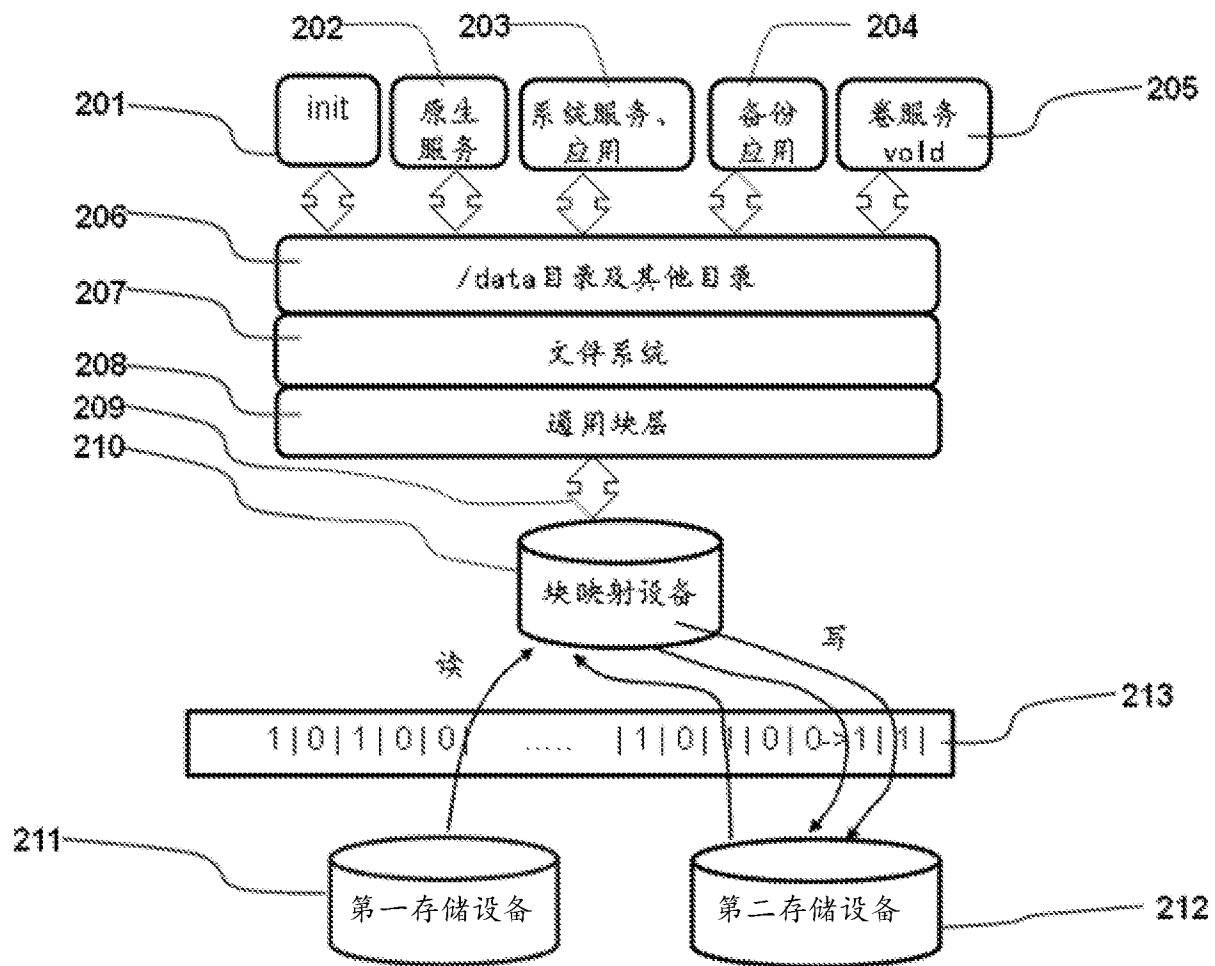


图 3

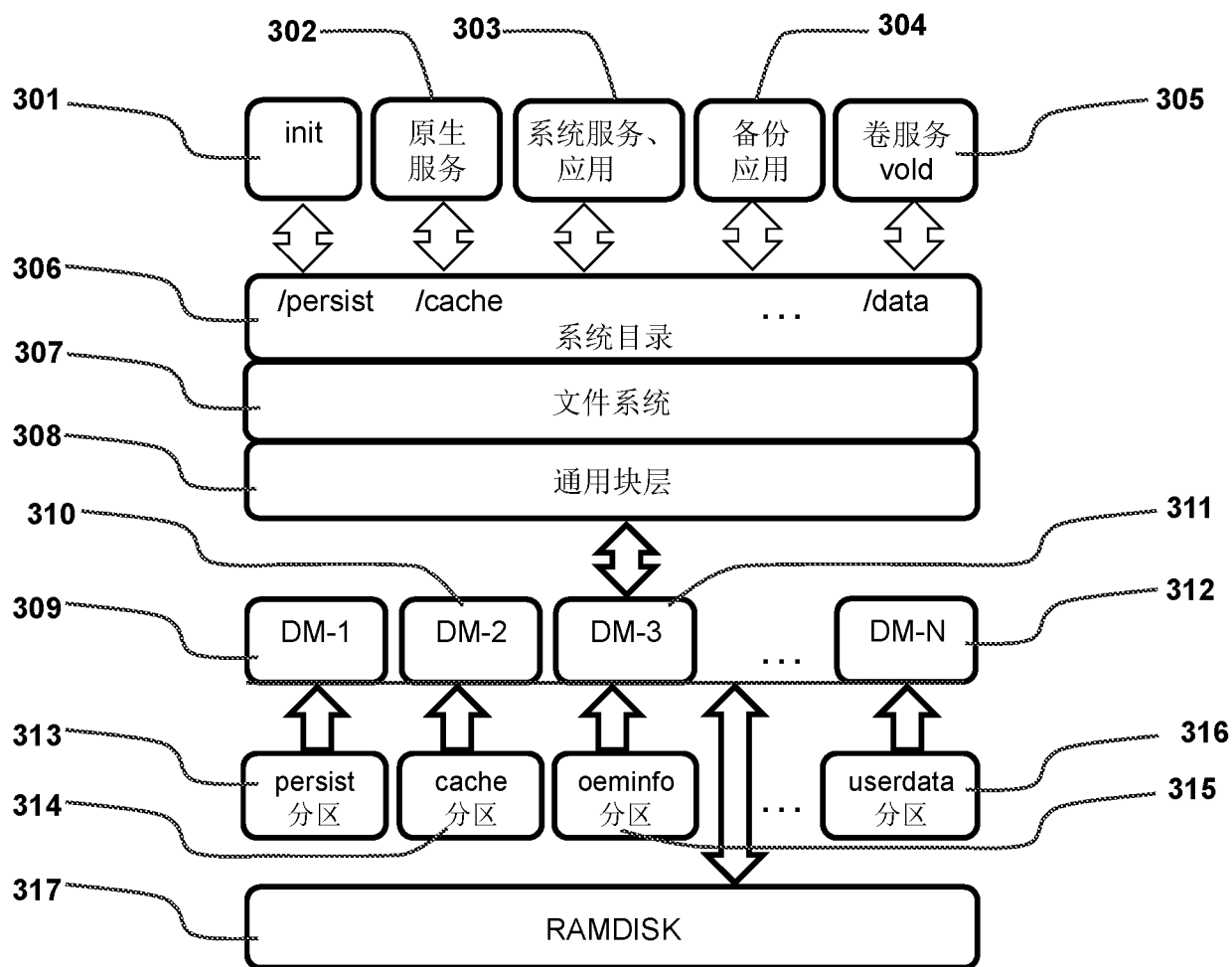


图 4

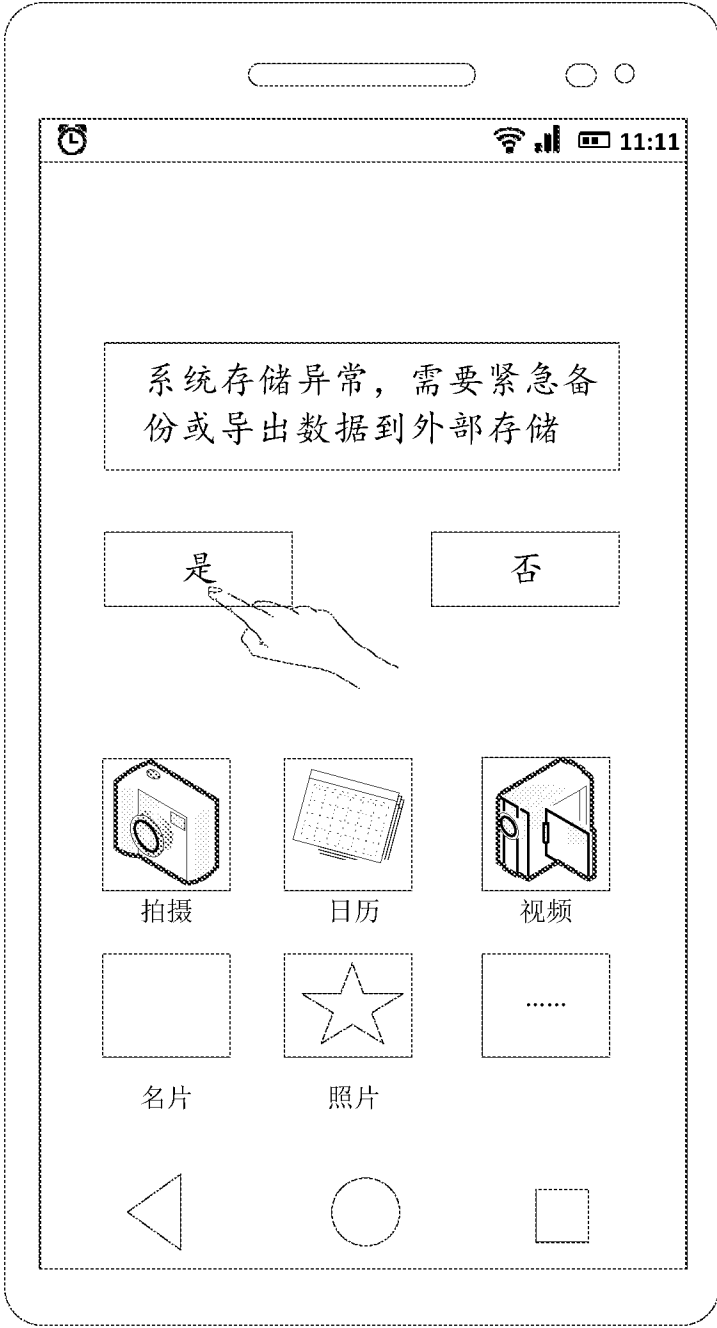


图 5

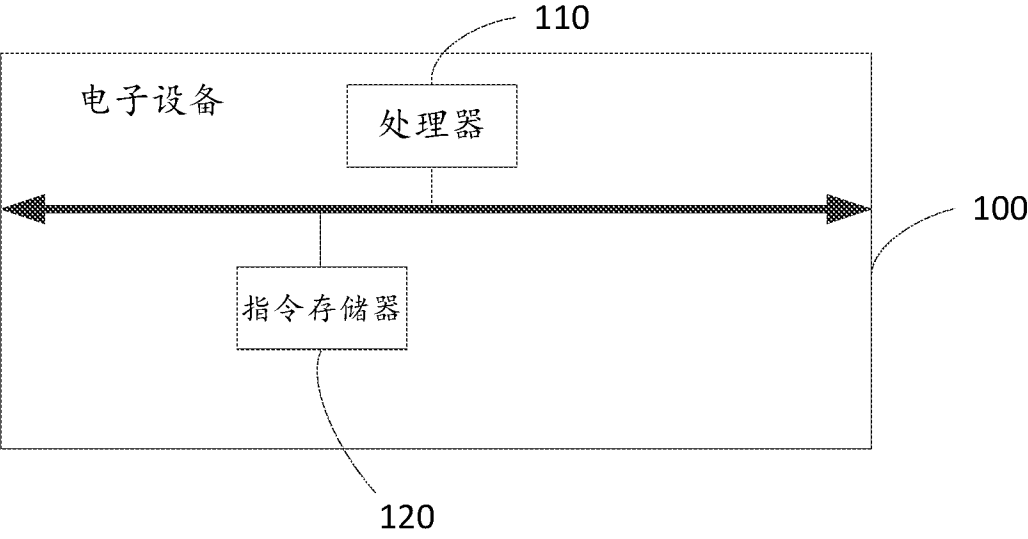


图 6

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/075599

A. CLASSIFICATION OF SUBJECT MATTER

G06F 12/02(2006.01)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)

WPI; EPODOC; CNPAT; CNKI; IEEE: 系统, 启动, 存储器, 写入, 指令, 预设, 地址, 映射, 读写, storage, system, starting, memory, write, primary, address, startup

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 108763099 A (HUAWEI TECHNOLOGIES CO., LTD.) 06 November 2018 (2018-11-06) claims 1-29	1-29
A	CN 102521013 A (HUAZHONG UNIVERSITY OF SCIENCE AND TECHNOLOGY) 27 June 2012 (2012-06-27) description, paragraphs [0002]-[0043]	1-29
A	CN 101923448 A (ANYKA (GUANGZHOU) MICROELECTRONICS TECHNOLOGY CO., LTD.) 22 December 2010 (2010-12-22) entire document	1-29
A	US 2015006795 A1 (CONTINENTAL AUTOMOTIVE GMBH) 01 January 2015 (2015-01-01) entire document	1-29



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“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)

“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

“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

“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

“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

“&” document member of the same patent family

Date of the actual completion of the international search

25 April 2019

Date of mailing of the international search report

20 May 2019

Name and mailing address of the ISA/CN

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

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/075599

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	108763099	A	06 November 2018	None			
CN	102521013	A	27 June 2012	None			
CN	101923448	A	22 December 2010	None			
US	2015006795	A1	01 January 2015	CN	104081349	A	01 October 2014
				DE	102012201225	A1	01 August 2013
				EP	2807558	A1	03 December 2014
				WO	2013110736	A1	01 August 2013

国际检索报告

国际申请号

PCT/CN2019/075599

A. 主题的分类

G06F 12/02 (2006.01) 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))

WPI; EPODOC; CNPAT; CNKI; IEEE: 系统, 启动, 存储器, 写入, 指令, 预设, 地址, 映射, 读写, storage, system, starting, memory, write, primary, address, startup

C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
PX	CN 108763099 A (华为技术有限公司) 2018年 11月 6日 (2018 - 11 - 06) 权利要求1-29	1-29
A	CN 102521013 A (华中科技大学) 2012年 6月 27日 (2012 - 06 - 27) 说明书第[0002]-[0043]段	1-29
A	CN 101923448 A (安凯广州微电子有限公司) 2010年 12月 22日 (2010 - 12 - 22) 全文	1-29
A	US 2015006795 A1 (CONTINENTAL AUTOMOTIVE GMBH) 2015年 1月 1日 (2015 - 01 - 01) 全文	1-29

☐ 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

2019年 4月 25日

国际检索报告邮寄日期

2019年 5月 20日

ISA/CN的名称和邮寄地址

中国国家知识产权局 (ISA/CN)
中国北京市海淀区蓟门桥西土城路6号 100088

受权官员

李萌

传真号 (86-10) 62019451

电话号码 86-(10)-53961536

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2019/075599

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	108763099	A	2018年 11月 6日	无			
CN	102521013	A	2012年 6月 27日	无			
CN	101923448	A	2010年 12月 22日	无			
US	2015006795	A1	2015年 1月 1日	CN	104081349	A	2014年 10月 1日
				DE	102012201225	A1	2013年 8月 1日
				EP	2807558	A1	2014年 12月 3日
				WO	2013110736	A1	2013年 8月 1日

表 PCT/ISA/210 (同族专利附件) (2015年1月)