



(12)发明专利

(10)授权公告号 CN 105511806 B

(45)授权公告日 2018.09.07

(21)申请号 201510856803.1

(56)对比文件

(22)申请日 2015.11.30

CN 102129353 A, 2011.07.20,

(65)同一申请的已公布的文献号

CN 1648878 A, 2005.08.03,

申请公布号 CN 105511806 A

CN 101218566 A, 2008.07.09,

(43)申请公布日 2016.04.20

CN 101382927 A, 2009.03.11,

(73)专利权人 华为技术有限公司

US 2010082879 A1, 2010.04.01,

地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

审查员 刘念

(72)发明人 薛春 石亮 高聪明

(74)专利代理机构 北京龙双利达知识产权代理有限公司 11329

代理人 王君 张欣

(51)Int.Cl.

G06F 3/06(2006.01)

权利要求书2页 说明书9页 附图3页

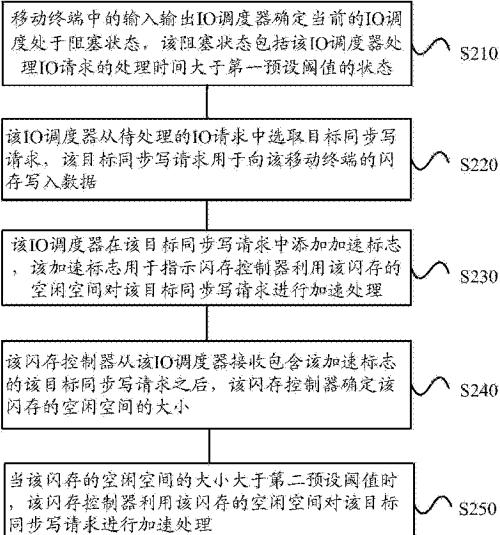
(54)发明名称

处理写请求的方法和移动终端

(57)摘要

本发明实施例公开了一种处理写请求的方法和移动终端，该方法包括：移动终端中的IO调度器确定当前的IO调度处于阻塞状态；该IO调度器从待处理的IO请求中选取目标同步写请求；该IO调度器在该目标同步写请求中添加加速标志，该加速标志用于指示闪存控制器利用该闪存的空闲空间利用该闪存的空闲空间对该目标同步写请求进行加速处理；该闪存控制器从该IO调度器接收包含该加速标志的该目标同步写请求之后，该闪存控制器确定该闪存的空闲空间的大小；当该闪存的空闲空间的大小大于第二预设阈值时，该闪存控制器对该目标同步写请求进行加速处理。本发明实施例的处理写请求的方法和移动终端能够减少移动终端中系统卡顿的情况。

200



1. 一种处理写请求的方法,其特征在于,包括:

移动终端中的输入输出IO调度器确定当前的IO调度处于阻塞状态,所述阻塞状态包括所述IO调度器处理IO请求的处理时间大于第一预设阈值的状态;

所述IO调度器从待处理的IO请求中选取目标同步写请求,所述目标同步写请求用于向所述移动终端的闪存写入数据;

所述IO调度器在所述目标同步写请求中添加加速标志,所述加速标志用于指示闪存控制器利用所述闪存的空闲空间对所述目标同步写请求进行加速处理;

所述闪存控制器从所述IO调度器接收包含所述加速标志的所述目标同步写请求之后,所述闪存控制器确定所述闪存的空闲空间的大小;

当所述闪存的空闲空间的大小大于第二预设阈值时,所述闪存控制器利用所述闪存的空闲空间对所述目标同步写请求进行加速处理。

2. 如权利要求1所述的方法,其特征在于,所述IO调度器从待处理的IO请求中选取目标同步写请求,包括:

所述IO调度器从IO请求队列中选取排序最靠前的同步写请求;

所述IO调度器将所述排序最靠前的同步写请求确定为所述目标同步写请求。

3. 如权利要求1或2所述的方法,其特征在于,所述移动终端中的IO调度器确定当前的IO调度处于阻塞状态,包括:

所述IO调度器确定IO请求队列中的待处理的IO请求的个数;

当所述待处理的IO请求的个数大于第一阈值时,所述IO调度器确定当前的IO调度处于阻塞状态。

4. 如权利要求1或2所述的方法,其特征在于,所述移动终端中的IO调度器确定IO调度处于阻塞状态,包括:

所述IO调度器确定当前正在处理的写请求或读请求的处理时间;

当所述处理时间大于第二阈值时,所述IO调度器确定当前IO调度处于阻塞状态。

5. 如权利要求1或2所述的方法,其特征在于,所述闪存控制器利用所述闪存的空闲空间对所述目标同步写请求进行加速处理,包括:

所述闪存控制器为所述目标同步写请求分配快速编程页;

所述闪存控制器使用所述快速编程页,对所述目标同步写请求进行加速处理。

6. 如权利要求5所述的方法,其特征在于,所述方法还包括:

在所述闪存控制器利用所述闪存的空闲空间对所述目标同步写请求进行加速处理之后,确定所述闪存的空闲空间的大小;

当所述闪存的空闲空间的大小小于第三阈值时,所述闪存控制器回收为所述目标同步写请求分配所述快速编程页时产生的损失页。

7. 一种处理写请求的移动终端,其特征在于,包括:

处理器,用于利用输入输出IO调度器,确定当前的IO调度处于阻塞状态,所述阻塞状态包括所述IO调度器处理IO请求的处理时间大于第一预设阈值的状态;从待处理的IO请求中选取目标同步写请求,所述目标同步写请求用于向所述移动终端的闪存写入数据,在所述目标同步写请求中添加加速标志,所述加速标志用于指示闪存控制器利用所述闪存的空闲空间对所述目标同步写请求进行加速处理;

所述闪存控制器,用于从所述处理器接收包含所述加速标志的所述目标同步写请求之后,确定所述闪存的空闲空间的大小,当所述闪存的空闲空间的大小大于第二预设阈值时,利用所述闪存的空闲空间对所述目标同步写请求进行加速处理。

8. 如权利要求7所述的移动终端,其特征在于,所述处理器具体用于:利用所述I/O调度器从I/O请求队列中选取排序最靠前的同步写请求;将所述排序最靠前的同步写请求确定为所述目标同步写请求。

9. 如权利要求7或8所述的移动终端,其特征在于,所述处理器具体用于:利用所述I/O调度器确定I/O请求队列中的待处理的I/O请求的个数;当所述待处理的I/O请求的个数大于第一阈值时,确定当前的I/O调度处于阻塞状态。

10. 如权利要求7或8所述的移动终端,其特征在于,所述处理器具体用于:利用所述I/O调度器确定当前正在处理的写请求或读请求的处理时间;当所述处理时间大于第二阈值时,确定当前I/O调度处于阻塞状态。

11. 如权利要求7或8所述的移动终端,其特征在于,所述闪存控制器具体用于:为所述目标同步写请求分配快速编程页;使用所述快速编程页,对所述目标同步写请求进行加速处理。

12. 如权利要求11所述的移动终端,其特征在于,所述闪存控制器具体用于:在利用所述闪存的空闲空间对所述目标同步写请求进行加速处理之后,确定所述闪存的空闲空间的大小;当所述闪存的空闲空间的大小小于第三阈值时,回收为所述目标同步写请求分配所述快速编程页时产生的损失页。

处理写请求的方法和移动终端

技术领域

[0001] 本发明实施例涉及移动终端领域,尤其涉及处理写请求的方法和移动终端。

背景技术

[0002] 近年来,移动终端已经成为人们生活中不可或缺的电子产品。移动终端的操作系统可视作一系列软硬件构成的层级结构。例如,在安卓(Android)操作系统中进行数据管理和维护的层级架构从上到下主要包括:数据库管理系统、文件系统、块设备驱动以及底层存储设备。其中,块设备驱动主要采用输入输出(Input Output, IO)调度器进行IO请求的调度。而底层存储设备通常是基于闪存的存储设备,例如,底层存储设备可以是嵌入式多媒体卡(Embedded Multi Media Card,eMMC)存储设备,eMMC存储设备包括闪存芯片以及控制闪存芯片进行读取操作的eMMC控制器。

[0003] 在移动终端的文件系统从底层存储设备读写数据的过程中,IO调度器中将产生相应的IO请求。其中,IO请求包括同步请求和异步请求。当IO调度器处理同步请求时,所述请求的数据必须全部写入存储设备或者全部从存储设备读出之后才能向系统返回确认值,从而所述同步请求才能够执行完毕。如果同步请求的处理时间过长或待处理的同步请求过多,即IO调度阻塞时,会造成移动终端中的系统卡顿、性能下降等问题,影响用户体验。

发明内容

[0004] 本发明实施例公开了一种处理写请求的方法和移动终端,以减少移动终端中的系统卡顿的情况。

[0005] 第一方面,提供一种处理写请求的方法,所述方法包括:移动终端中的输入输出IO调度器确定当前的IO调度处于阻塞状态,该阻塞状态包括该IO调度器处理IO请求的处理时间大于第一预设阈值的状态;该IO调度器从待处理的IO请求中选取目标同步写请求,该目标同步写请求用于向该移动终端的闪存写入数据;该IO调度器在该目标同步写请求中添加加速标志,该加速标志用于指示闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理;该闪存控制器从该IO调度器接收包含该加速标志的该目标同步写请求之后,该闪存控制器确定该闪存的空闲空间的大小;当该闪存的空闲空间的大小大于第二预设阈值时,该闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理。

[0006] 结合第一方面,在第一方面的第一种可能的实现方式中,该方法还包括:该IO调度器从待处理的IO请求中选取目标同步写请求,包括:该IO调度器从IO请求队列中选取排序最靠前的同步写请求;该IO调度器将该排序最靠前的同步写请求确定为该目标同步写请求。

[0007] 结合第一方面或第一方面的第一种可能的实现方式,在第一方面的第二种可能的实现方式中,该方法还包括:该移动终端中的IO调度器确定当前的IO调度处于阻塞状态,包括:该IO调度器确定IO请求队列中的待处理的IO请求的个数;当该待处理的IO请求的个数大于第一阈值时,该IO调度器确定当前的IO调度处于阻塞状态。

[0008] 结合第一方面或第一方面的第一种可能的实现方式,在第一方面的第三种可能的实现方式中,该方法还包括:该移动终端中的I0调度器确定I0调度处于阻塞状态,包括:该I0调度器确定当前正在处理的写请求或读请求的处理时间;当该处理时间大于第二阈值时,该I0调度器确定当前I0调度处于阻塞状态。

[0009] 结合第一方面、第一方面的第一种至第三种可能的实现方式中的任一种可能的实现方式,在第一方面的第四种可能的实现方式中,该闪存控制器对该目标同步写请求进行加速处理,包括:该闪存控制器为该目标同步写请求分配快速编程页;该闪存控制器使用该快速编程页,对该目标同步写请求进行加速处理。

[0010] 结合第一方面的第四种可能的实现方式中的任一种可能的实现方式,在第一方面的第五种可能的实现方式中,在该闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理之后,确定该闪存的空闲空间的大小;当该闪存的空闲空间的大小小于第三阈值时,该闪存控制器回收为该目标同步写请求分配该快速编程页时产生的损失页。

[0011] 第二方面,提供了一种处理写请求的移动终端,该移动终端包括:处理器,用于利用输入输出I0调度器,确定当前的I0调度处于阻塞状态,该阻塞状态包括该I0调度器处理I0请求的处理时间大于第一预设阈值的状态;从待处理的I0请求中选取目标同步写请求,该目标同步写请求用于向该移动终端的闪存写入数据,在该目标同步写请求中添加加速标志,该加速标志用于指示闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理;闪存控制器,用于从该处理器接收包含该加速标志的该目标同步写请求之后,确定该闪存的空闲空间的大小,当该闪存的空闲空间的大小大于第二预设阈值时,利用该闪存的空闲空间对该目标同步写请求进行加速处理。

[0012] 结合第二方面,在第二方面的第一种可能的实现方式中,该处理器具体用于:利用该I0调度器从I0请求队列中选取排序最靠前的同步写请求;将该排序最靠前的同步写请求确定为该目标同步写请求。

[0013] 结合第二方面或第二方面的第一种可能的实现方式,在第二方面的第二种可能的实现方式中,该处理器具体用于:利用该I0调度器确定I0请求队列中的待处理的I0请求的个数;当该待处理的I0请求的个数大于第一阈值时,确定当前的I0调度处于阻塞状态。

[0014] 结合第二方面或第二方面的第一种可能的实现方式,在第二方面的第三种可能的实现方式中,该处理器具体用于:利用该I0调度器确定当前正在处理的写请求或读请求的处理时间;当该处理时间大于第二阈值时,确定当前I0调度处于阻塞状态。

[0015] 结合第二方面、第二方面的第一种至第三种可能的实现方式中的任一种可能的实现方式,在第二方面的第四种可能的实现方式中,该闪存控制器具体用于:为该目标同步写请求分配快速编程页;使用该快速编程页,对该目标同步写请求进行加速处理。

[0016] 结合第二方面的第四种可能的实现方式,在第二方面的第五种可能的实现方式中,该闪存控制器具体用于:在利用该闪存的空闲空间对该目标同步写请求进行加速处理之后,确定该闪存的空闲空间的大小;当该闪存的空闲空间的大小小于第三阈值时,回收为该目标同步写请求分配该快速编程页时产生的损失页。

[0017] 在本发明实施例中,移动终端中的I0调度器在确定当前的I0调度处于阻塞状态之后,从待处理的I0请求中选取目标同步写请求,并在该目标同步写请求中添加加速标志,以使闪存控制器在接收到包含该加速标志的目标同步写请求、并确定闪存的空闲空间的大小

大于第二预设阈值之后,对该目标同步写请求进行加速处理,从而能够减少移动终端中的由于处理同步请求造成的系统卡顿的情况。

附图说明

[0018] 为了更清楚地说明本发明实施例的技术方案,下面将对本发明实施例中所需要使用的附图作简单地介绍,显而易见地,下面所描述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。

[0019] 图1是根据本发明实施例的移动终端的示意性结构图。

[0020] 图2是根据本发明实施例的处理写请求的方法的示意性流程图。

[0021] 图3是根据本发明实施例的处理写请求的移动终端300的示意图。

[0022] 图4是根据本发明实施例的处理写请求的移动终端400的示意图。

具体实施方式

[0023] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明的一部分实施例,而不是全部实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动的前提下所获得的所有其他实施例,都应属于本发明保护的范围。

[0024] 应理解,本发明实施例的处理写请求的方法可以应用于移动终端(Mobile Terminal),所述移动终端可以称之为终端(Terminal)、移动台(Mobile Station,简称为“MS”)或用户设备(User Equipment,简称为“UE”)等。所述移动终端可以经无线接入网(Radio Access Network,简称为“RAN”)与一个或多个核心网进行通信,例如,移动终端可以是移动电话(或称为“蜂窝”电话)。例如,所述移动终端还可以是便携式、袖珍式、手持式、计算机内置的或者车载的移动装置。

[0025] 应理解,本发明实施例中的移动终端可以包括主机、闪存以及闪存控制器,主机通过闪存控制器读写闪存中的数据。例如,图1示出了本发明实施例中的移动终端的示意性结构图。如图1所示,主机中的IO调度器可以通过闪存控制器控制闪存的读写操作。其中,IO调度器可以通过接口与闪存控制器相连,闪存控制器可以通过闪存接口与闪存连接,以实现对闪存的控制。

[0026] 应理解,本发明实施例中的闪存可以是嵌入式多媒体卡(Embedded Multi Media Card,eMMC)存储设备中的闪存芯片,闪存控制器可以是eMMC存储设备中的eMMC控制器。闪存也可以是其它基于闪存的存储设备,闪存控制器也可以是其它能够实现控制闪存读写操作的控制器,本发明实施例对此不作具体限定。

[0027] 图2示出了本发明实施例的处理写请求的方法200的示意性流程图,如图2所示,所述方法200包括:

[0028] S210,移动终端中的IO调度器确定当前的IO调度处于阻塞状态,该阻塞状态包括该IO调度器处理IO请求的处理时间大于第一预设阈值的状态;

[0029] S220,该IO调度器从待处理的IO请求中选取目标同步写请求,该目标同步写请求用于向该移动终端的闪存写入数据;

[0030] S230，该I0调度器在该目标同步写请求中添加加速标志，该加速标志用于指示闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理；

[0031] S240，该闪存控制器从该I0调度器接收包含该加速标志的该目标同步写请求之后，该闪存控制器确定该闪存的空闲空间的大小；

[0032] S250，当该闪存的空闲空间的大小大于第二预设阈值时，该闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理。

[0033] 在本发明实施例中，移动终端中的I0调度器在确定当前的I0调度处于阻塞状态之后，从待处理的I0请求中选取目标同步写请求，并在所述目标同步写请求中添加加速标志，以使闪存控制器在接收到包含所述加速标志的目标同步写请求、并确定闪存的空闲空间的大小大于第二预设阈值之后，对所述目标同步写请求进行加速处理，从而能够减少移动终端中的由于处理同步请求造成的系统卡顿的情况。

[0034] 应理解，本发明实施例中，上述I0调度器处理I0请求的处理时间，可以包括I0调度器在产生I0请求（包括读请求和写请求）之后到所述I0请求被处理并向系统返回确认值所使用的时间。或者，上述处理I0请求的处理时间，可以包括I0请求等待被处理的等待时间或执行I0请求的执行时间。另外，上述处理I0请求可以包括根据I0请求，将I0请求写入闪存或从闪存中读取出来的过程。

[0035] 应理解，本发明实施例中的阻塞状态，可以指I0调度器当前处理的I0请求的处理时间超过第一预设阈值，也可以指I0调度器处理I0请求队列中的所有I0请求的处理时间超过第一预设阈值。或者，阻塞状态可以包括I0调度器中待处理的I0请求的等待时间超过第一预设阈值的情况。本发明实施例中对第一预设阈值的取值不作具体限定，可以是根据实际操作时I0调度器的处理能力确定，也可以根据经验确定。

[0036] 应理解，本发明实施例对I0调度器确定当前的I0调度处于阻塞状态所采用的方法不作限定，可以是先确定当前I0调度器处理I0请求的处理时间超过某个阈值，从而确定当前的I0调度处于阻塞状态。也可以是先确定当前I0调度器中待处理的I0请求个数超过一定个数，从而确定当前的I0调度处于阻塞状态。或者也可以包括其它本领域技术人员定义的确定I0调度处于阻塞状态的方法。

[0037] 可选地，从I0请求中选取目标同步写请求，可以是选取待处理的I0请求中的一个或几个同步写请求。上述I0调度器从待处理的I0请求中选取目标同步写请求，可以理解为，I0调度器从待处理的I0请求中的同步写请求中，选出至少一个同步写请求，并将所述至少一个同步写请求确定为目标同步写请求。所述至少一个目标同步写请求可以是待处理的同步写请求中的其中一个或几个，也可以是待处理的同步写请求中的全部。

[0038] 可选地，本发明实施例对I0调度器添加加速标志的方法不作限定，例如，I0调度器可以在同步写请求之中增加一个标志位，所述标志位可以为1个比特或几个比特。当I0调度器确定目标同步写请求之后，可以在目标同步写请求中的标志位放置加速标志。或者，在同步写请求中增加标志位，也可以理解为，在同步写请求所对应的数据参数中增加一个加速标志位参数，如果确定同步写请求为目标同步写请求之后，可以将所述加速标志位参数置为预先规定的数值。例如，当标志位为1个比特位时，可以设定标志位的符号为1时，表示对所述目标同步写请求进行加速处理。

[0039] 可选地，作为一个实施例，闪存控制器在从I0调度器接收到目标同步写请求之后，

可以确定所述目标同步写请求中是否包括加速标志。以所述闪存控制器是eMMC控制器为例,可以对现有技术中的eMMC控制器的功能进行扩展,使eMMC控制器能够识别在I0调度器中被添加加速标志的目标同步写请求。若eMMC控制器确定收到的同步写请求中不包含加速标志时,则可以按照普通模式处理所述同步写请求。可选地,当eMMC控制器接收到目标同步写请求之后,可以通过加速标志确定所述目标同步写请求,然后确定闪存的空闲空间的大小,以确定是否对所述目标同步写请求进行加速处理。若闪存的空闲空间的大小大于第二预设阈值时,对所述目标同步写请求进行加速处理,若闪存的空闲空间的大小小于第二预设阈值时,则可以按照普通模式处理所述目标同步写请求。

[0040] 应理解,闪存控制器对目标同步写请求进行加速处理时,将对闪存的空间产生一定的浪费,即跟普通处理模式相比,闪存控制器对目标同步写请求进行加速处理时,会占用更多的空间。所以在本发明实施例中,当闪存控制器接收到包含加速标志的目标同步写请求时,可以确定当前的空闲空间的大小是否能够进行加速处理,当空闲空间的大小大于第二预设阈值时,表示空闲空间的大小能够进行加速处理。当空闲空间的大小小于第二预设阈值时,表示空闲空间的大小不能够进行加速处理。应理解,对于第二预设阈值的设定,可以以不影响对闪存读写数据为准,本发明实施例对此并不作具体限定。例如,第二预设阈值可以设定为略大于闪存的垃圾回收阈值。

[0041] 应理解,本发明实施例对闪存控制器对目标同步写请求进行加速处理的具体方法不作限制。例如,可以使用现有技术中的加速方法对目标同步写请求进行加速处理。

[0042] 可选地,作为一个实施例,本发明实施例的处理写请求的方法200中,该I0调度器从待处理的I0请求中选取目标同步写请求,包括:该I0调度器从I0请求队列中选取排序最靠前的同步写请求;该I0调度器将该排序最靠前的同步写请求确定为该目标同步写请求。

[0043] 应理解,本发明实施例对I0调度器选取目标同步写请求的方法不作限定,例如,I0调度器可以采取某种遍历的方法遍历I0请求队列,从I0请求队列中选取排序最靠前的同步写请求,并将所述排序最靠前的同步写请求确定为目标同步写请求。

[0044] 应理解,通常情况下,I0调度器根据I0请求队列的排序从前至后依次处理I0请求。换句话说,I0请求在I0请求队列的排序越靠前,就越先被处理。当I0调度处于阻塞状态时,选取在I0请求队列中最靠前的同步写请求为目标同步写请求,并在所述目标同步写请求中添加加速标志,因为所述目标同步写请求排序在前,相比排序在后的同步写请求,可以更早地被I0调度器发送至闪存控制器,从而由闪存控制器执行加速处理,进而可以在更短的时间内缓解I0阻塞状态,减少系统卡顿的情况。

[0045] 可选地,作为一个实施例,本发明实施例的处理写请求的方法200中,该移动终端中的I0调度器确定当前的I0调度处于阻塞状态,包括:该I0调度器确定I0请求队列中的待处理的I0请求的个数;当该待处理的I0请求的个数大于第一阈值时,该I0调度器确定当前的I0调度处于阻塞状态。

[0046] 应理解,在本发明实施例中,当I0请求队列中的待处理的I0请求的个数较多时,代表着I0请求队列中的待处理的I0请求要等待较长的时间才能被写入闪存,由此可以确定当前I0调度器处于阻塞状态。应理解,对于第一阈值的设定并不作具体限定。例如,可以根据当前系统需求来设置第一阈值,也可以根据经验设置。

[0047] 可选地,作为一个实施例,本发明实施例的处理写请求的方法200中,该移动终端

中的I0调度器确定I0调度处于阻塞状态,包括:该I0调度器确定当前正在处理的写请求或读请求的处理时间;当该处理时间大于第二阈值时,该I0调度器确定当前I0调度处于阻塞状态。

[0048] 应理解,在本发明实施例中,根据I0调度器当前在处理的I0请求的处理时间,可以获知当前I0调度器处理I0请求的能力,进而确定当前I0调度器是否处于阻塞状态。其中,本发明实施例中的写请求可以包括同步写请求,也可以包括异步写请求。应理解,移动终端中的I0调度器在读写操作过程中产生的所有读请求都属于同步请求。本发明实施例对于第二阈值的设定并不作具体限定。例如,可以根据当前系统需求来设置第二阈值,也可以根据经验设置。

[0049] 可选地,作为一个实施例,本发明实施例中的处理写请求的方法200中,该闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理,包括:该闪存控制器为该目标同步写请求分配快速编程页;该闪存控制器使用该快速编程页,对该目标同步写请求进行加速处理。

[0050] 应理解,在本发明实施例中,对闪存控制器使用快速编程页进行加速处理的具体方法不作限制。例如,以闪存是多级存储单元(Multiple Level Cell,MLC)闪存为例,MLC闪存中的每个存储单元可以编程两个比特位,分别为最高有效位(Most Significant Bit,MSB)和最低有效位(Least Significant,LSB),对应的MLC闪存中存在MSB页和LSB页,MSB页全部由MSB组成,LSB页全部由LSB组成。在编程过程中,MSB页的编程速度要快于LSB页。在为目标同步写请求写分配快速编程页时,可以设定为数据只写入MSB页,而跳过LSB页,从而加快了闪存控制器处理目标同步写请求的速度。

[0051] 可选地,作为一个实施例,本发明实施例中的处理写请求的方法200中,还包括:在该闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理之后,确定该闪存的空闲空间的大小;当该闪存的空闲空间的大小小于第三阈值时,该闪存控制器回收为该目标同步写请求分配该快速编程页时产生的损失页。

[0052] 应理解,本发明实施例中,对所述第三阈值的设定不作具体限定。所述第三阈值可以为所述闪存的垃圾回收阈值,当闪存的空闲空间小于垃圾回收阈值时,闪存控制器需要对闪存的空间进行清理回收,从而回收对目标同步写请求分配快速编程页时产生的损失页。

[0053] 例如,前例中的MLC闪存在对目标同步写请求进行加速处理时,闪存中的LSB页并未写入数据,但是占据了闪存的存储空间,LSB页即是对目标同步写请求分配快速编程页时产生的损失页。优先回收损失页,能够及时减少对闪存空间的浪费,提高对闪存空间的利用率。以闪存控制器是eMMC控制器为例,可以对eMMC控制器的中的功能机制进行设置,将损失页设置为无效页,则在垃圾回收过程中,即可将加速处理时产生的损失页回收,以提高闪存空间的利用率。

[0054] 上文结合图1和图2详细阐述了本发明实施例的处理写请求的方法的具体实施例,下文将结合图3和图4,详细阐述本发明实施例的处理写请求的移动终端。

[0055] 图3示出了根据本发明实施例的处理写请求的移动终端300的示意图,应理解,本发明实施例的移动终端300中的各个模块的下述和其他操作和/或功能分别为了实现图2中的各个方法的相应流程,为了简洁,在此不再赘述,如图3所示,所述移动终端300包括:

[0056] 处理器310，用于利用I/O调度器，确定当前的I/O调度处于阻塞状态，该阻塞状态包括该I/O调度器处理I/O请求的处理时间大于第一预设阈值的状态；从待处理的I/O请求中选取目标同步写请求，该目标同步写请求用于向该移动终端的闪存写入数据，在该目标同步写请求中添加加速标志，该加速标志用于指示闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理；

[0057] 闪存控制器320，用于从该处理器接收包含该加速标志的该目标同步写请求之后，确定该闪存的空闲空间的大小，当该闪存的空闲空间的大小大于第二预设阈值时，利用该闪存的空闲空间对该目标同步写请求进行加速处理。

[0058] 在本发明实施例中，移动终端中的处理器利用I/O调度器在确定当前的I/O调度处于阻塞状态之后，从待处理的I/O请求中选取目标同步写请求，并在所述目标同步写请求中添加加速标志，以使闪存控制器在接收到包含所述加速标志的目标同步写请求、并确定闪存的空闲空间的大小大于第二预设阈值之后，对所述目标同步写请求进行加速处理，从而能够减少移动终端中的由于处理同步请求造成的系统卡顿的情况。

[0059] 如图3所示，处理器310和闪存控制器320之间可以通过总线系统连接。

[0060] 应理解，在本发明实施例中，所述处理器310可以是中央处理单元(Central Processing Unit, CPU)，所述处理器310还可以是其他通用处理器、数字信号处理器(Digital Signal Processor, DSP)、专用集成电路(Application Specific Integrated Circuit, ASIC)、现成可编程门阵列(Field Programmable Gate Array, FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。通用处理器可以是微处理器或者所述处理器也可以是任何常规的处理器等。

[0061] 可选地，作为一个实施例，该处理器310具体用于：利用该I/O调度器从I/O请求队列中选取排序最靠前的同步写请求；将该排序最靠前的同步写请求确定为该目标同步写请求。

[0062] 可选地，作为一个实施例，该处理器310具体用于：利用该I/O调度器确定I/O请求队列中的待处理的I/O请求的个数；当该待处理的I/O请求的个数大于第一阈值时，确定当前的I/O调度处于阻塞状态。

[0063] 可选地，作为一个实施例，该处理器310具体用于：利用该I/O调度器确定当前正在处理的写请求或读请求的处理时间；当该处理时间大于第二阈值时，确定当前I/O调度处于阻塞状态。

[0064] 可选地，作为一个实施例，该闪存控制器320具体用于：为该目标同步写请求分配快速编程页；使用该快速编程页，对该目标同步写请求进行加速处理。

[0065] 可选地，作为一个实施例，该闪存控制器320具体用于：在利用该闪存的空闲空间对该目标同步写请求进行加速处理之后，确定该闪存的空闲空间的大小；当该闪存的空闲空间的大小小于第三阈值时，回收为该目标同步写请求分配该快速编程页时产生的损失页。

[0066] 图4示出了根据本发明实施例的处理写请求的移动终端400的示意图。所述移动终端400可以为一种处理写请求的装置。应理解，本发明实施例的移动终端400中的各个模块的下述和其他操作和/或功能分别为了实现图2中的各个方法的相应流程，为了简洁，在此不再赘述，所述移动终端400可以为移动终端，如图4所示，所述移动终端400包括：

[0067] I/O调度模块410，用于确定当前的I/O调度处于阻塞状态，该阻塞状态包括该I/O调度器处理I/O请求的处理时间大于第一预设阈值；从待处理的I/O请求中选取目标同步写请求，该目标同步写请求用于向该移动终端的闪存写入数据，在该目标同步写请求中添加加速标志，该加速标志用于指示闪存控制器利用该闪存的空闲空间对该目标同步写请求进行加速处理；

[0068] 闪存控制模块420，用于从该处理器接收包含该加速标志的该目标同步写请求之后，确定该闪存的空闲空间的大小，当该闪存的空闲空间的大小大于第二预设阈值时，利用该闪存的空闲空间对该目标同步写请求进行加速处理。

[0069] 在本发明实施例中，移动终端中的I/O调度模块在确定当前的I/O调度处于阻塞状态之后，从待处理的I/O请求中选取目标同步写请求，并在所述目标同步写请求中添加加速标志，以使闪存控制模块在接收到包含所述加速标志的目标同步写请求、并确定闪存的空闲空间的大小大于第二预设阈值之后，对所述目标同步写请求进行加速处理，从而能够减少移动终端中的由于处理同步请求造成的系统卡顿的情况。

[0070] 可选地，作为一个实施例，该I/O调度模块410具体用于：从I/O请求队列中选取排序最靠前的同步写请求；将该排序最靠前的同步写请求确定为该目标同步写请求。

[0071] 可选地，作为一个实施例，该I/O调度模块410具体用于：确定I/O请求队列中的待处理的I/O请求的个数；当该待处理的I/O请求的个数大于第一阈值时，确定当前的I/O调度处于阻塞状态。

[0072] 可选地，作为一个实施例，该I/O调度模块410具体用于：确定当前正在处理的写请求或读请求的处理时间；当该处理时间大于第二阈值时，确定当前I/O调度处于阻塞状态。

[0073] 可选地，作为一个实施例，该闪存控制模块420具体用于：为该目标同步写请求分配快速编程页；使用该快速编程页，对该目标同步写请求进行加速处理。

[0074] 可选地，作为一个实施例，该闪存控制模块420具体用于：在利用该闪存的空闲空间对该目标同步写请求进行加速处理之后，确定该闪存的空闲空间的大小；当该闪存的空闲空间的大小小于第三阈值时，回收为该目标同步写请求分配该快速编程页时产生的损失页。

[0075] 应理解，在本发明的各种实施例中，上述各过程的序号的大小并不意味着执行顺序的先后，各过程的执行顺序应以其功能和内在逻辑确定，而不应对本发明实施例的实施过程构成任何限定。

[0076] 本领域普通技术人员可以意识到，结合本文中所公开的实施例描述的各示例的单元及算法步骤，能够以电子硬件、计算机软件或者二者的结合来实现，为了清楚地说明硬件和软件的可互换性，在上述说明中已经按照功能一般性地描述了各示例的组成及步骤。这些功能究竟以硬件还是软件方式来执行，取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能，但是这种实现不应认为超出本发明的范围。

[0077] 所属领域的技术人员可以清楚地了解到，为了描述的方便和简洁，上述描述的系统、装置和单元的具体工作过程，可以参考前述方法实施例中的对应过程，在此不再赘述。

[0078] 在本申请所提供的几个实施例中，应所述理解到，所揭露的系统、移动终端和方法，可以通过其它的方式实现。例如，以上所描述的移动终端实施例仅仅是示意性的，例如，

所述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另外,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口、移动终端或单元的间接耦合或通信连接,也可以是电的,机械的或其它的形式连接。

[0079] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本发明实施例方案的目的。

[0080] 另外,在本发明各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以是两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。

[0081] 所述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时,可以存储在一个计算机可读取存储介质中。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分,或者所述技术方案的全部或部分可以以软件产品的形式体现出来,所述计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM, Read-Only Memory)、随机存取存储器(RAM, Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0082] 以上某一实施例中的技术特征和描述,为了使申请文件简洁清楚,可以理解适用于其他实施例,在其他实施例不再一一赘述。

[0083] 以上所述,仅为本发明的具体实施方式,但本发明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发明揭露的技术范围内,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发明的保护范围之内。因此,本发明的保护范围应以权利要求的保护范围为准。

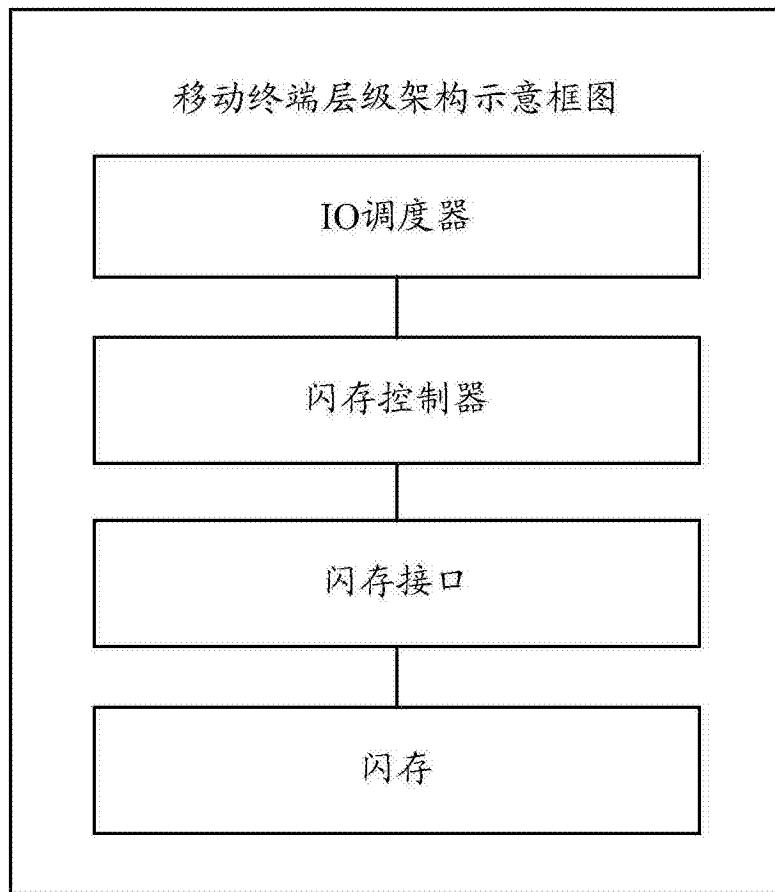


图1

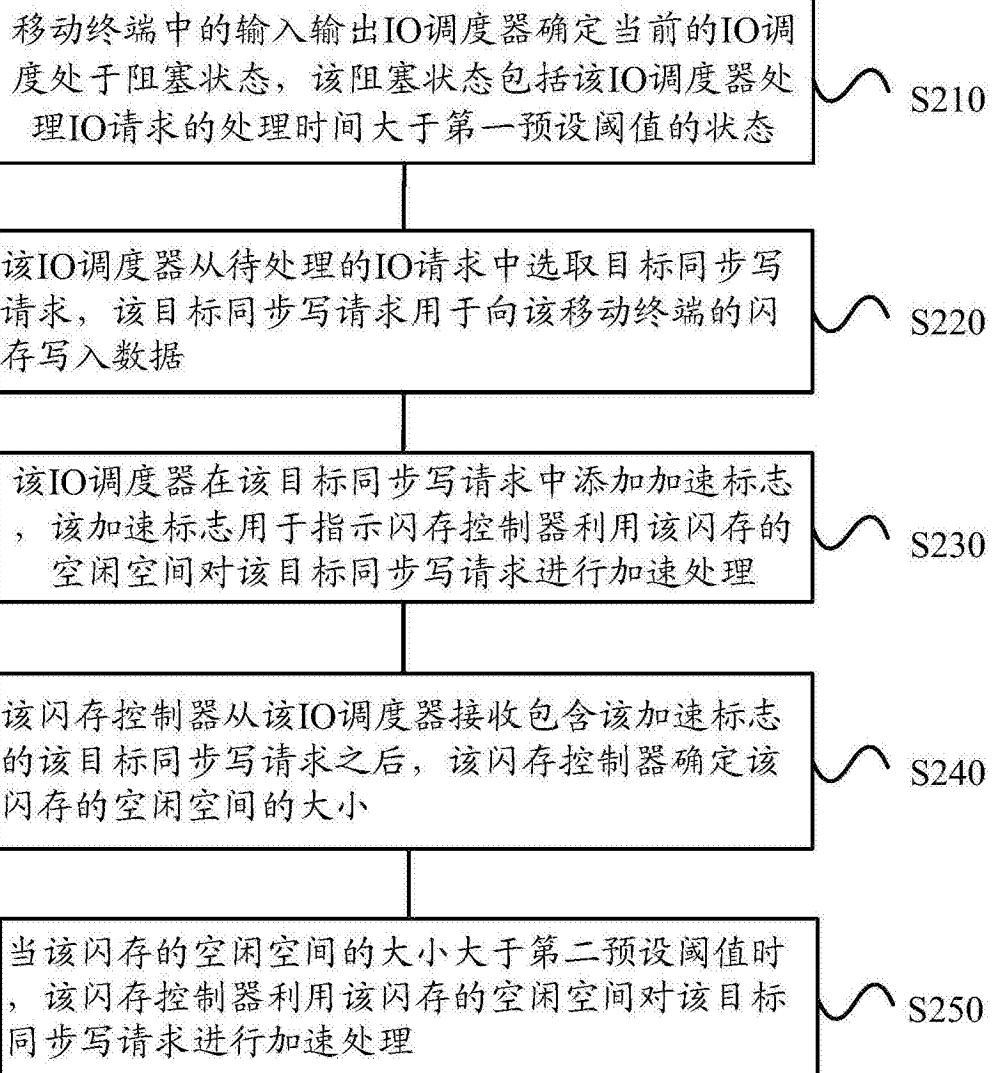
200

图2

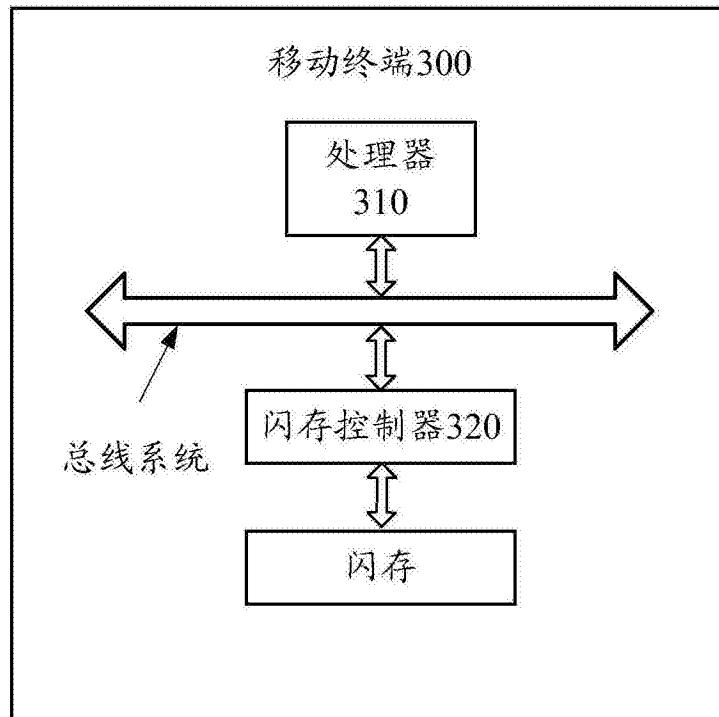


图3

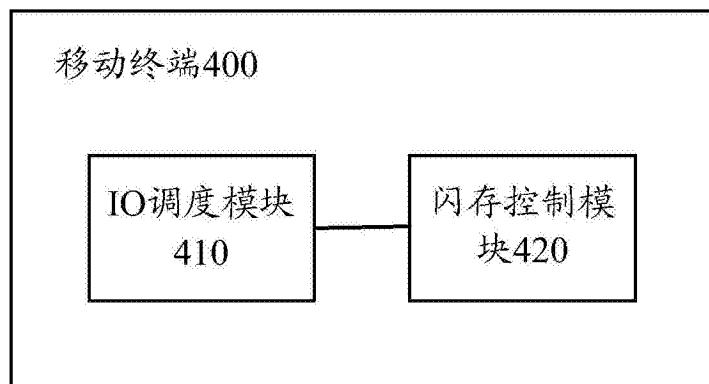


图4