



(12)发明专利

(10)授权公告号 CN 104007806 B

(45)授权公告日 2017. 10. 24

(21)申请号 201410280928.X

(22)申请日 2014.06.20

(65)同一申请的已公布的文献号

申请公布号 CN 104007806 A

(43)申请公布日 2014.08.27

(73)专利权人 中科创达软件股份有限公司

地址 100191 北京市海淀区龙翔路甲1号泰翔商务楼4层401-409

(72)发明人 朱勇 赵鸿飞

(74)专利代理机构 北京集佳知识产权代理有限公司

公司 11227

代理人 王宝筠

(51)Int. Cl.

G06F 1/32(2006.01)

(56)对比文件

CN 102687098 A, 2012.09.19, 说明书第0025-0084段及附图1-14.

US 7451334 B2, 2008.11.11, 全文.

CN 101593015 A, 2009.12.02, 全文.

CN 103500003 A, 2014.01.08, 说明书第0026-0102段及附图1-2.

审查员 徐生芹

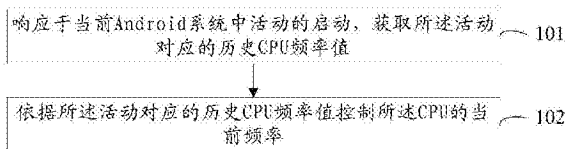
权利要求书2页 说明书6页 附图1页

(54)发明名称

一种安卓系统中的CPU频率的控制方法、装置和设备

(57)摘要

本申请提供了一种安卓系统中的CPU频率的控制方法、装置和设备,该方法包括:响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值;依据所述活动对应的历史CPU频率值控制所述CPU的当前频率。该装置包括:获取历史频率值模块,用于响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值;控制模块,用于依据所述历史CPU频率值控制所述CPU的当前频率。采用本申请的方法、装置和系统,能够有效的降低移动设备的CPU的工作频率,进而能够有效降低移动设备的功耗,不仅能够节约移动设备的资源,还能够提升用户的使用体验。



1. 一种Android系统中的CPU频率的控制方法,其特征在于,包括:

响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值;所述获取所述活动对应的历史CPU频率值,包括:获取所述活动的活动名称;依据所述活动名称从数据管理器中获取该活动对应的历史CPU频率值;

依据所述活动对应的历史CPU频率值控制所述CPU的当前频率;所述依据所述历史CPU频率值控制所述CPU的当前频率,包括:判断所述活动的执行次数是否满足预设次数阈值,如果是,则依据所述活动对应的历史CPU频率值确定所述CPU的当前频率,如果否,则按照按需策略确定所述CPU的频率,所述按需策略为:由所述当前Android系统的内核态来确定CPU的当前频率的策略;

其中,所述依据所述活动对应的历史CPU频率值确定所述CPU的频率,包括:

判断所述活动的历史CPU频率值是否满足频率波动阈值,如果满足,则继续判断是否存在所占比例大于预设比例阈值的CPU频率值,如果存在,则将所占比例大于预设比例阈值的CPU频率值确定为CPU的当前频率;

如果不满足或者不存在,则按照按需策略确定所述CPU的频率,所述按需策略为:由所述当前Android系统的内核态来确定CPU的当前频率的策略。

2. 根据权利要求1所述的方法,其特征在于,在所述活动启动之后,还包括:

按照预设的采集周期采集所述活动运行的CPU频率值,并将采集到的CPU频率值存储至所述数据管理器。

3. 一种Android系统中的CPU频率的控制装置,其特征在于,包括:

获取历史频率值模块,用于响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值;所述获取历史频率值模块包括:获取活动名称子模块,用于获取所述活动的活动名称;获取历史频率值子模块,用于依据所述活动名称从数据管理器中获取该活动对应的历史CPU频率值;

控制模块,用于依据所述历史CPU频率值控制所述CPU的当前频率;所述控制模块包括:第一判断子模块,用于判断所述活动的执行次数是否满足预设次数阈值;第一确定子模块,用于在所述第一判断子模块的结果为是的情况下,则依据所述活动对应的历史CPU频率值确定所述CPU的当前频率;第二确定子模块,用于在所述第一判断子模块的结果为否的情况下,按照按需策略确定所述CPU的频率,所述按需策略为:由所述当前Android系统的内核态来确定CPU的当前频率的策略;

其中,所述第一确定子模块包括:

第二判断子模块,用于判断所述活动的历史CPU频率值是否满足频率波动阈值;

第三判断子模块,用于在所述第二判断子模块的结果为是的情况下,判断是否存在所占比例大于预设比例阈值的CPU频率值;

第三确定子模块,用于在所述第三判断子模块的结果为是的情况下,将所占比例大于预设比例阈值的CPU频率值确定为CPU的当前频率;和,第四确定子模块,用于在所述第二判断子模块或者所述第三判断子模块的结果为否的情况下,按照按需策略确定所述CPU的频率。

4. 根据权利要求3所述的装置,其特征在于,还包括:

采集模块,用于按照预设的采集周期采集所述活动运行的CPU频率值;

存储模块,用于将采集到的CPU频率值存储至所述数据管理器。

一种安卓系统中的CPU频率的控制方法、装置和设备

技术领域

[0001] 本申请涉及智能手机的Android系统领域,特别涉及Android系统中的CPU频率的控制方法、装置和设备。

背景技术

[0002] 移动设备也被称为行动装置(Mobile device)、流动装置、手持装置(handheld device)等,是一种口袋大小的计算设备,通常有一个小的显示萤幕,触控输入,或是小型的键盘。例如,掌上游戏机、移动电话、智能手机、平板电脑等。移动设备由于使用比较频繁,所以对电量要求越来越高,用户也越来越倾向于较为省电的移动设备。

[0003] 在现有技术中,移动设备的电量很大一部分都消耗在CPU上。CPU功耗的公式为:功率=c*总栅电容*电压*电压*频率。其中,c为与电路有关的一个调整参数,总栅电容对于一个CPU是固定的,可见CPU的功率与工作电压的平方成正比,与工作频率成正比。而工作频率和工作电压对于一个CPU都有固定的配置关系。因此,如果能有效降低CPU的工作频率就能有效降低移动设备的功耗。

[0004] 发明人在研究过程中发现,现有技术中的移动设备并没有一种能有效降低CPU的工作频率的技术方案。

发明内容

[0005] 本申请所要解决的技术问题是提供一种Android系统中的CPU频率的控制方法,从而能够有效的降低移动设备的CPU的工作频率,进而能够有效降低移动设备的功耗,不仅能够节约移动设备的资源,还能够提升用户的使用体验。

[0006] 本申请还提供了一种Android系统中的CPU频率的控制装置和设备,用以保证上述方法在实际中的实现及应用。

[0007] 为了解决上述问题,本申请公开了一种Android系统中的CPU频率的控制方法,包括:

[0008] 响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值;

[0009] 依据所述历史CPU频率值控制所述CPU的当前频率。

[0010] 可选的,所述获取所述活动对应的历史CPU频率值,包括:

[0011] 获取所述活动的活动名称;

[0012] 依据所述活动名称从数据管理器中获取该活动对应的历史CPU频率值。

[0013] 可选的,在所述活动启动之后,还包括:

[0014] 按照预设的采集周期采集所述活动运行的CPU频率值,并将采集到的CPU频率值存储至所述数据管理器。

[0015] 可选的,所述依据所述活动对应的历史CPU频率值控制所述CPU的当前频率,包括:

[0016] 判断所述活动的执行次数是否满足预设次数阈值,如果满足,则继续判断是否存在所占比例大于阈值比例阈值的CPU频率值,如果存在,则所占比例大于预设比例阈值的

CPU频率值确定为CPU的当前频率,如果不满足或者不存在,则按照按需策略确定所述CPU的频率。

[0017] 可选的,所述依据所述活动对应的历史CPU频率值确定所述CPU的频率,包括:

[0018] 判断所述活动的历史CPU频率值是否满足频率波动阈值,如果是,则将所述CPU的频率确定为所述历史CPU频率值的平均值,如果不是,则按照按需策略确定所述CPU的频率。

[0019] 本申请还公开了一种Android系统中的CPU频率的控制装置,包括:

[0020] 获取历史频率值模块,用于响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值;

[0021] 控制模块,用于依据所述历史CPU频率值控制所述CPU的当前频率。

[0022] 可选的,所述获取历史频率值模块包括:

[0023] 获取活动名称子模块,用于获取所述活动的活动名称;

[0024] 获取历史频率值子模块,用于依据所述活动名称从数据管理器中获取该活动对应的历史CPU频率值。

[0025] 可选的,还包括:

[0026] 采集模块,用于按照预设的采集周期采集所述活动运行的CPU频率值;

[0027] 存储模块,用于将采集到的CPU频率值存储至所述数据管理器。

[0028] 可选的,所述控制模块包括:

[0029] 第一判断子模块,用于判断所述活动的执行次数是否满足预设次数阈值;

[0030] 第一确定子模块,用于在所述判断子模块的结果为是的情况下,则依据所述活动对应的历史CPU频率值确定所述CPU的当前频率;

[0031] 第二确定子模块,用于在所述判断子模块的结果为否的情况下,按照按需策略确定所述CPU的频率。

[0032] 可选的,所述第一确定子模块包括:

[0033] 第二判断子模块,用于判断所述活动的历史CPU频率值是否满足频率波动阈值;

[0034] 第三判断子模块,用于在所述第二判断子模块的结果为是的情况下,判断是否存在所占比例大于阈值比例阈值的CPU频率值;

[0035] 第三确定子模块,用于在所述第三判断子模块的结果为是的情况下,将所占比例大于预设比例阈值的CPU频率值确定为CPU的当前频率;

[0036] 第四确定子模块,用于在所述第二判断子模块或者所述第三判断子模块的结果为否的情况下,按照按需策略确定所述CPU的频率。

[0037] 与现有技术相比,本申请包括以下优点:

[0038] 在本申请中,通过以系统中的活动为参考,并根据活动对应的历史CPU频率值来控制CPU的当前频率,可以使CPU的工作频率能够有效地得到降低,进而能够有效降低移动设备的功耗,不仅能够节约移动设备的资源,还能够提升用户的使用体验。

附图说明

[0039] 为了更清楚地说明本申请实施例中的技术方案,下面将对实施例描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其

他的附图。

[0040] 图1是本申请的一种Android系统中的CPU频率的控制方法实施例1的流程图；

[0041] 图2是本申请的一种Android系统中的CPU频率的控制装置实施例结构框图。

具体实施方式

[0042] 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,所描述的实施例仅仅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。

[0043] 本申请可用于众多通用或专用的计算装置环境或配置中。例如:个人计算机、服务器计算机、手持设备或便携式设备、平板型设备、多处理器装置、包括以上任何装置或设备的分布式计算环境等等。

[0044] 本申请可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本申请,在这些分布式计算环境中,由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[0045] 为了方便本领域技术人员更清楚的了解本申请的相关技术背景,申请人首先对现有的安卓系统的4种CPU调频策略进行简单的介绍。Android系统主要有4种CPU调频策略:performance(性能策略),powersave(节电策略),ondemand(按需策略),以及,userspace(用户态策略)。

[0046] 其中,“performance”表示在这种策略下CPU固定运行在最高工作频率。但是这样会导致CPU的功耗太大。“powersave”表示在这种策略下CPU固定运行在最低工作频率。但是这种策略下CPU的性能最差,很多情况满足不了用户对性能的要求。“ondemand”表示在这种策略下CPU运行在较低工作频率,当系统负载增大时会提高CPU的工作频率。由于这个策略是基于一个定时器,不断地计算系统在一个timer期间的负载情况,得出负载值后,根据一些固定的算法,判定是否调节CPU的工作频率以及如何调节。但是因为一个周期内的平均值不能真实地反应出系统的频率需求,尤其是使用当前时刻之前的一个周期的结果,来决定下一个周期系统的频率,所以这种调节也不准确。“userspace”表示在这种策略下CPU工作频率由用户态控制。这种策略的问题是用户态很难准确知道系统需要多高的频率。

[0047] 参考图1,示出了本申请一种Android系统中的CPU频率的控制方法实施例的流程图,可以包括以下步骤:

[0048] 步骤101:响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值。

[0049] 在本申请实施例中,可以设置一个调频管理器,来监听Android系统中活动的启动或停止。如果监听到当前Android系统中活动启动了,则可以获取到该活动的历史CPU频率值,即是该活动曾经在运行时对应的CPU频率值是多少。具体的,可以设置一个决策器来实现步骤101和步骤102,当调频管理器监听到活动启动的时候,就调用决策器来获取到活动对应的历史CPU频率值。

[0050] 其中,获取所述活动对应的历史CPU频率值的步骤,具体可以包括:

[0051] 步骤A1:获取所述活动的活动名称。

[0052] 因为Android系统中的各个活动都有对应的名称,首先获取到已经启动的活动的活动名称。

[0053] 步骤A2:依据所述活动名称从数据管理器中获取该活动对应的历史CPU频率值。

[0054] 在本实施例中,可以设置一个数据管理器用来存放各个活动对应的历史CPU频率值,那么在得到活动名称之后,就可以从数据管理器中对应查找到匹配的历史CPU频率值。

[0055] 可以理解的是,在本申请实施例中,可以设置一个采样器,用来在活动启动期间,采集CPU的工作频率。那么在所述活动启动之后,还包括:

[0056] 步骤B:按照预设的采集周期采集所述活动运行的CPU频率值,并将采集到的CPU频率值存储至数据管理器。

[0057] 在实际应用中,可以预先设置好采样器的采集周期,则活动启动之后,就可以由采样器根据预设的采集周期采集活动运行时的CPU频率值,并将采集到的CPU频率值和活动名称对应存储至数据管理器,以便后续需要从数据管理器中查询得到。其中,采集周期可以以毫秒为单位,例如数百毫秒等。

[0058] 步骤102:依据所述活动对应的历史CPU频率值控制所述CPU的当前频率。

[0059] 在得到该活动对应的历史CPU频率值的时候,就参考历史CPU频率值来控制CPU的当前频率。

[0060] 步骤102在具体实现的时候,可以包括:

[0061] 步骤C1:判断所述活动的执行次数是否满足预设次数阈值,如果是,进入步骤C2,如果不是,则进入步骤C3。

[0062] 在实际应用中,因为可能存在活动第一次运行时并没有对应的CPU历史频率值的情况,所以可以先判断该活动的执行次数是否满足预设次数阈值。例如,将预设次数阈值设置为1,表示如果活动已经执行过一次,就进入步骤C2,如果不满足则说明活动还未被执行过,就进入步骤C3。

[0063] 步骤C2:依据所述活动对应的历史CPU频率值确定所述CPU的当前频率。

[0064] 如果活动已经执行过一次,那么数据管理器中就会存储有采样器采集到的历史CPU频率值,此时可以参考着历史CPU频率值来确定CPU的当前频率。

[0065] 步骤C3:按照按需策略确定所述CPU的当前频率。

[0066] 而如果活动未被执行过,则可以按照之前介绍过的按需策略的方式确定CPU的频率,也即是由Android系统的内核态来确定CPU的当前频率。因为由内核态来确定CPU的当前频率属于现有技术,在此不再赘述。

[0067] 在实际应用中,步骤C2在具体实现的时候,可以包括:

[0068] 步骤D1:判断所述活动的历史CPU频率值是否满足频率波动阈值,如果是,则进入步骤D2,如果不是,则进入步骤D4。

[0069] 可以理解的是,在依据历史CPU的频率值来确定当前CPU频率的时候,可以先判断历史CPU频率值是否满足频率波动阈值,例如频率波动阈值为1,则说明对应记录的多个历史CPU频率值都相等,即是只有一个频率值,此时说明活动启动的时候CPU运转比较正常,频率比较稳定,则可以进入步骤D2中直接将历史CPU频率值确定为CPU的当前频率。当然,频率

波动阈值也可以设置为2,即是数据管理器中与该活动对应的有两个历史CPU频率值,此时可以判断一下是不是某一个频率值所占的比例大于预设的一个比例阈值,例如90%,如果存在这样的频率值,则可以将该频率值确定为CPU的当前工作频率,反之,则还需要进入步骤D3,按照按需策略确定CPU的当前工作频率。当然,本领域技术人员还可以根据实际需求来设置其他数值。

[0070] 步骤D2:判断是否存在所占比例大于阈值比例阈值的CPU频率值,如果存在,则进入步骤D3,如果不存在,则进入步骤D4。

[0071] 在满足频率波动阈值的情况下,判断是不是有所占比例大于阈值比例阈值的CPU频率值,如果有,也说明CPU的工作频率相对比较稳定。

[0072] 步骤D3:将所占比例大于预设比例阈值的CPU频率值确定为CPU的当前工作频率。

[0073] 步骤D4:按照按需策略确定所述CPU的频率。

[0074] 如果没有满足频率波动阈值,或者虽然满足频率波动阈值,但是不存在所占比例大于预设比例阈值的CPU频率值,都说明数据管理器中记录的CPU的频率波动较大,那么此时就可以还按照按需策略来确定CPU的频率。

[0075] 采用本发明实施例,通过以系统中的活动为参考,并根据活动对应的历史CPU频率值来控制CPU的当前频率,可以使CPU的工作频率能够有效地得到降低,进而能够有效降低移动设备的功耗,延长移动设备的电池续航力,这样不仅能够节约移动设备的资源,还能够提升用户的使用体验。

[0076] 对于前述的各方法实施例,为了简单描述,故将其都表述为一系列的动作组合,但是本领域技术人员应该知悉,本申请并不受所描述的动作顺序的限制,因为依据本申请,某些步骤可以采用其他顺序或者同时进行。其次,本领域技术人员也应该知悉,说明书中所描述的实施例均属于优选实施例,所涉及的动作和模块并不一定是本申请所必须的。

[0077] 与上述本申请一种Android系统中的CPU频率的控制方法实施例所提供的方法相对应,参见图2,本申请还提供了一种Android系统中的CPU频率的控制装置实施例,在本实施例中,可以包括:

[0078] 获取历史频率值模块201,用于响应于当前Android系统中活动的启动,获取所述活动对应的历史CPU频率值。

[0079] 其中,所述获取历史频率值模块201具体可以包括:

[0080] 获取活动名称子模块,用于获取所述活动的活动名称;和,获取历史频率值子模块,用于依据所述活动名称从数据管理器中获取该活动对应的历史CPU频率值。

[0081] 控制模块202,用于依据所述历史CPU频率值控制所述CPU的当前频率。

[0082] 其中,所述控制模块202具体可以包括:

[0083] 第一判断子模块,用于判断所述活动的执行次数是否满足预设次数阈值;第一确定子模块,用于在所述判断子模块的结果为是的情况下,则依据所述活动对应的历史CPU频率值确定所述CPU的当前频率;和,第二确定子模块,用于在所述判断子模块的结果为否的情况下,按照按需策略确定所述CPU的频率。

[0084] 其中,所述第一确定子模块又可以包括:

[0085] 第二判断子模块,用于判断所述活动的历史CPU频率值是否满足频率波动阈值;第三判断子模块,用于在所述第二判断子模块的结果为是的情况下,判断是否存在所占比例

大于阈值比例阈值的CPU频率值;第三确定子模块,用于在所述第三判断子模块的结果为是的情况下,将所占比例大于预设比例阈值的CPU频率值确定为CPU的当前频率;和,第四确定子模块,用于在所述第二判断子模块或者所述第三判断子模块的结果为否的情况下,按照按需策略确定所述CPU的频率。

[0086] 其中,在不同的实施例中,该装置还可以包括:

[0087] 采集模块,用于按照预设的采集周期采集所述活动运行的CPU频率值;和,存储模块,用于将采集到的CPU频率值存储至所述数据管理器。

[0088] 采用本发明实施例,通过以系统中的活动为参考,并根据活动对应的历史CPU频率值来控制CPU的当前频率,可以使CPU的工作频率能够有效地得到降低,进而能够有效降低移动设备的功耗,延长移动设备的电池续航力,这样不仅能够节约移动设备的资源,还能够提升用户的使用体验。

[0089] 需要说明的是,本说明书中的各个实施例均采用递进的方式描述,每个实施例重点说明的都是与其他实施例的不同之处,各个实施例之间相同相似的部分互相参见即可。对于装置类实施例而言,由于其与方法实施例基本相似,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

[0090] 最后,还需要说明的是,在本文中,诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。

[0091] 以上对本申请所提供的一种Android系统中的CPU频率的控制方法、装置和系统进行了详细介绍,本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想;同时,对于本领域的一般技术人员,依据本申请的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本申请的限制。

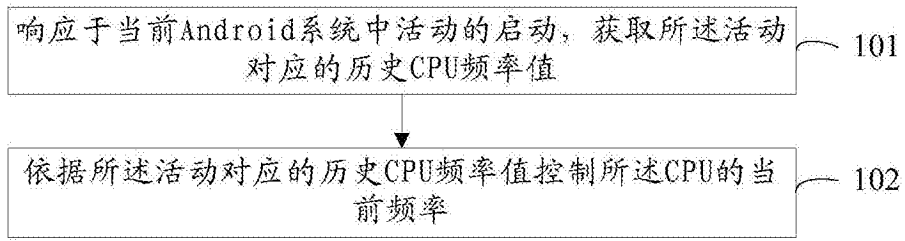


图1

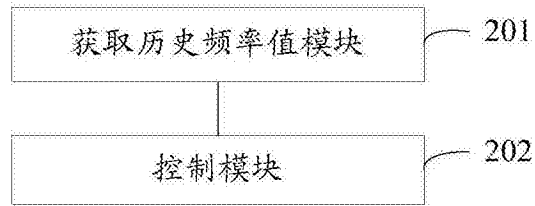


图2