



(12)发明专利申请

(10)申请公布号 CN 109564458 A

(43)申请公布日 2019.04.02

(21)申请号 201780048084.8

(51)Int.Cl.

(22)申请日 2017.07.05

G06F 1/3206(2019.01)

(30)优先权数据

G06F 9/50(2006.01)

15/224,834 2016.08.01 US

(85)PCT国际申请进入国家阶段日

2019.02.01

(86)PCT国际申请的申请数据

PCT/CN2017/091865 2017.07.05

(87)PCT国际申请的公布数据

W02018/024071 EN 2018.02.08

(71)申请人 华为技术有限公司

地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72)发明人 王军 卡西克·拉奥 叶寒栋

权利要求书2页 说明书9页 附图8页

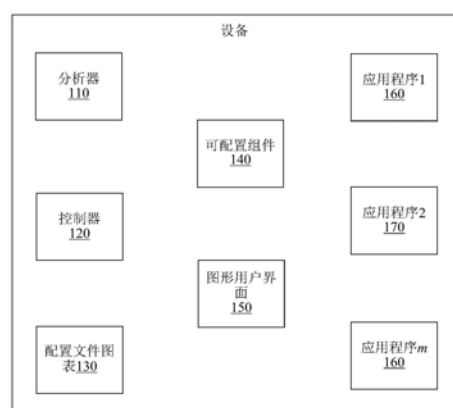
(54)发明名称

应用程序特定的,性能感知的能量优化

(57)摘要

装置包括:非瞬时性存储器,该非瞬时性存储器包括应用程序;控制器,该控制器耦接至所述存储器并且用于调整在所述装置中与独立于其它应用程序的所述应用程序相关联的配置,其中配置是所述装置的资源指配;以及分析器,该分析器耦合至所述存储器,并且用于:测量与所述配置对应的测量性能;以及测量与所述配置对应的测量功耗。

100
↓



1. 一种装置,其特征在于,包括:
非瞬时性存储器,所述非瞬时性存储器包括应用程序;
控制器,所述控制器耦合至所述存储器,并且用于调整在所述装置中与独立于其它应用程序的所述应用程序相关联的配置,其中所述配置是所述装置的资源指配;以及
分析器,所述分析器耦合至所述存储器,并且用于:
测量与所述配置对应的测量性能;以及
测量与所述配置对应的测量功耗。
2. 根据权利要求1所述的装置,其特征在于,所述配置包括与所述应用程序相关联的处理器频率、与所述应用程序相关联的存储器带宽、与所述应用程序相关联的线程迁移、与所述应用程序相关联的图形处理器 (graphics processing unit, GPU) 频率或与所述应用程序相关联的网络报文速率中的一个。
3. 根据权利要求2所述的装置,其特征在于,所述配置包括与所述应用程序相关联的处理器频率、与所述应用程序相关联的存储器带宽、与所述应用程序相关联的线程迁移、与所述应用程序相关联的GPU频率或与所述应用程序相关联的网络报文速率中的至少两个。
4. 根据权利要求1所述的装置,其特征在于,所述配置适用于所述装置的硬件组件。
5. 根据权利要求1所述的装置,其特征在于,所述配置适用于所述装置的软件组件。
6. 一种方法,其特征在于,包括:
确定与应用程序相关联的性能;
执行与所述应用程序相关联的配置,以便在降低与所述应用程序相关联的能耗的同时,大体上保持所述性能;
测量由所述执行产生的测量性能;以及
调整所述配置以响应所述测量性能。
7. 根据权利要求6所述的方法,其特征在于,所述执行包括指示可配置组件实现所述配置。
8. 根据权利要求7所述的方法,其特征在于,所述可配置组件是处理器,并且其中所述执行还包括指示所述处理器在可变频率下运行。
9. 根据权利要求7所述的方法,其特征在于,所述可配置组件是总线、存储器、图形处理器 (graphics processing unit, GPU)、接收器或发送器。
10. 根据权利要求6所述的方法,其特征在于,所述执行包括指示多个可配置组件实现所述配置。
11. 一种装置,其特征在于,包括:
非瞬时性存储器,所述非瞬时性存储器包括应用程序;
图形用户界面 (graphical user interface, GUI), 用于:
显示第一选择器用于用户选择所述应用程序的目标性能;以及
接收指示所述目标性能的第一用户输入;以及
控制器,所述控制器耦合至所述存储器和所述GUI,并且用于执行所述目标性能以响应所述第一用户输入。
12. 根据权利要求11所述的装置,其特征在于,所述GUI还用于:
显示第二选择器用于所述用户选择目标功耗;以及

接收指示所述目标功耗的第二用户输入；

其中所述控制器还用于：

关联所述目标性能与所述目标功耗；以及

执行所述目标性能以响应所述关联。

13. 根据权利要求11所述的装置，其特征在于，所述GUI还用于：

显示第二选择器用于所述用户选择所述应用程序是关键的还是非关键的；以及

接收指示关键或非关键的第二用户输入。

14. 根据权利要求13所述的装置，其特征在于，所述控制器还用于：

当所述第二用户输入指示所述应用程序是关键的时，执行所述应用程序的默认性能，其中所述默认性能是由所述应用程序的开发者、所述装置的制造商或所述用户设置的所述应用程序的性能；以及

当所述应用程序不关键时，执行所述目标性能。

15. 根据权利要求11所述的装置，其特征在于，所述控制器还用于在所述应用程序运行时执行所述目标性能。

16. 一种计算机程序产品，所述计算机程序产品包括存储于非瞬时性介质中的计算机可执行指令，在由处理器执行所述指令时使得装置：

调整在所述装置中与独立于其它应用程序的所述应用程序相关联的配置；

测量与所述配置对应的测量性能；

以及测量与所述配置对应的测量功耗。

17. 根据权利要求16所述的计算机程序产品，其特征在于，所述指令还使得所述装置在所述应用程序运行时调整所述配置。

18. 根据权利要求16所述的计算机程序产品，其特征在于，所述指令还使得所述装置：

确定与所述应用程序相关联的性能；以及

调整所述配置以便在降低与所述应用程序相关联的能耗的同时大体上保持所述性能。

19. 根据权利要求16所述的计算机程序产品，其特征在于，所述指令还使得所述装置：

显示第一选择器用于用户选择与所述应用程序相关联的目标性能；以及

接收指示所述目标性能的用户输入；以及

执行所述目标性能以响应所述用户输入。

20. 根据权利要求16所述的计算机程序产品，其特征在于，所述指令还使得所述装置：

当所述应用程序是关键的时，执行所述应用程序的默认性能；以及

当所述应用程序不关键时，执行所述应用程序的目标性能。

应用程序特定的,性能感知的能量优化

交叉申请

[0001] 本申请要求于2016年8月1日提交的、发明名称为“应用程序特定的,性能感知的能量优化(Application-Specific,Performance-Aware Energy Optimization)”的美国非临时专利申请第15/224,834号的优先权,其全部内容以引用的方式并入本文中。

背景技术

[0002] 移动电话、笔记本电脑、平板电脑和其它智能设备越来越普及。这些智能设备的应用用同样也越来越普及。智能设备制造商必须增加处理功率以适应来自应用增加的处理需求。但是,处理功率的增加需要电功率的增加,而电功率的增加导致电池消耗的增加。为了解决增加的电池消耗,智能设备制造商寻求增加电池寿命,并且寻求通过减少功耗以改进电功率管理,使得智能电话用户继续使用延长的电池寿命。

发明内容

[0003] 当前的功率管理技术是与应用程序无关的。因此,这些技术不能为许多应用程序提供最佳的功率管理。根据本公开的各种实施例,提供了应用程序特定的,性能感知的能量优化。控制器以应用程序特定的方式来执行配置。换句话说,所述控制器针对独立于其它应用程序的应用程序执行所述配置。在此情况下,“独立地”可指所述控制器在所述应用程序运行时执行所述配置,并且不管其它应用程序是否在运行。所述控制器以性能感知的方式来执行所述配置。举例来说,所述控制器可以区分关键应用程序和非关键应用程序,保持或大体上保持关键应用程序的默认性能,以及保持或大体上保持非关键应用程序的性能。所述实施例大体上保持了应用程序的性能,同时降低了应用程序的能耗。能量等于功率乘以时间,因此,当时间段固定时,减小功耗就是减少能耗。减少能耗延长了电池寿命。

[0004] 在第一方面中,本申请提供一种装置,所述装置包括:非瞬时性存储器,所述非瞬时性存储器包括应用程序;控制器,所述控制器耦合至所述存储器,并且用于调整在所述装置中与独立于其它应用程序的所述应用程序相关联的配置,其中所述配置是所述装置的资源的指配;以及分析器,所述分析器耦合至所述存储器,并且用于:测量与所述配置对应的测量性能;以及测量与所述配置对应的测量功耗。

[0005] 根据第一方面的装置的第一实施方式,其中所述配置包括与所述应用程序相关联的处理器频率、与所述应用程序相关联的存储器带宽、与所述应用程序相关联的线程迁移、与所述应用程序相关联的GPU频率或与所述应用程序相关联的网络报文速率中的一个。

[0006] 根据第一方面或所述第一方面的任一前述实施方式的所述装置的第二实施方式,其中所述配置包括与所述应用程序相关联的处理器频率、与所述应用程序相关联的存储器带宽、与所述应用程序相关联的线程迁移、与所述应用程序相关联的GPU频率或与所述应用程序相关联的网络报文速率中的至少两个。

[0007] 根据第一方面或所述第一方面的任一前述实施方式的所述装置的第三实施方式,其中所述配置适用于所述装置的硬件组件。

[0008] 根据第一方面或所述第一方面的任一前述实施方式的所述装置的第四实施方式,其中所述配置适用于所述装置的软件组件。

[0009] 在第二方面中,本申请提供一种方法,包括:确定与应用程序相关联的性能;执行与所述应用程序相关联的配置,以便在降低与所述应用程序相关联的能耗的同时大体上保持所述性能;测量由所述执行产生的测量性能;以及调整所述配置以响应所述测量性能。

[0010] 根据第二方面的所述方法的第一实施方式,其中所述执行包括指示可配置组件实现所述配置。

[0011] 根据第二方面或所述第二方面的任一前述实施方式的所述方法的第二实施方式,其中所述可配置组件是处理器,并且其中所述执行还包括指示所述处理器在可变频率下运行。

[0012] 根据第二方面或所述第二方面的任一前述实施方式的所述方法的第三实施方式,其中所述可配置组件是总线、存储器、GPU、接收器或发送器。

[0013] 根据第二方面或所述第二方面的任一前述实施方式的所述方法的第四实施方式,其中所述执行包括指示多个可配置组件实现所述配置。

[0014] 在第三方面中,本申请提供一种装置,包括:非瞬时性存储器,所述非瞬时性存储器包括应用程序;GUI,用于:显示第一选择器用于用户选择所述应用程序的目标性能,以及接收指示所述目标性能的第一用户输入;以及控制器,所述控制器耦合至所述存储器和所述GUI,并且用于执行所述目标性能以响应所述第一用户输入。

[0015] 根据第三方面的所述装置的第一实施方式,所述GUI还用于:显示第二选择器用于所述用户选择目标功耗;以及接收指示所述目标功耗的第二用户输入,其中所述控制器还用于:将所述目标性能与所述目标功耗;以及执行所述目标性能以响应所述关联。

[0016] 根据第三方面或所述第三方面的任一前述实施方式的所述装置的第二实施方式,所述GUI还用于:显示第二选择器用于所述用户选择所述应用程序是关键的还是非关键的;以及接收指示关键或非关键的第二用户输入。

[0017] 根据第三方面或所述第三方面的任一前述实施方式的所述装置的第三实施方式,所述控制器还用于:当所述第二用户输入指示所述应用程序是关键的时,执行所述应用程序的默认性能,其中所述默认性能是由所述应用程序的开发者、所述装置的制造商或所述用户设置的所述应用程序的性能;以及当所述应用程序不关键时,执行所述目标性能。

[0018] 根据第三方面或所述第三方面的任一前述实施方式的所述装置的第四实施方式,其中所述控制器还用于在所述应用程序运行时执行所述目标性能。

[0019] 在第四方面中,本申请提供一种计算机程序产品,所述计算机程序产品包括:存储于非瞬时性介质上的计算机可执行指令,在由处理器执行所述指令时使得装置:调整在所述装置中与独立于其它应用程序的所述应用程序相关联的配置;测量与所述配置对应的测量性能;以及测量与所述配置对应的测量功耗。

[0020] 根据第四方面的所述装置的第一实施方式,其中所述指令还使得所述装置在所述应用程序运行时调整所述配置。

[0021] 根据第四方面或所述第四方面的任一前述实施方式的所述装置的第二实施方式,其中所述指令还使得所述装置:确定与所述应用程序相关联的性能;以及调整所述配置以便在降低与所述应用程序相关联的能耗的同时大体上保持所述性能。

[0022] 根据第四方面或所述第四方面的任一前述实施方式的所述装置的第三实施方式,其中所述指令还使得所述装置:显示第一选择器用于用户选择与所述应用程序相关联的目标性能;接收指示所述目标性能的用户输入;以及执行所述目标性能以响应所述用户输入。

[0023] 根据第四方面或所述第四方面的任一前述实施方式的所述装置的第四实施方式,其中所述指令还使得所述装置:当所述应用程序是关键的时候,执行所述应用程序的默认性能;以及当所述应用程序不关键时,执行所述应用程序的目标性能。

[0024] 以上实施例中的任一个可以与以上其它实施例中的任一个组合以创建新的实施例。根据以下结合附图以及权利要求书进行的详细描述,将更清楚地理解这些特征以及其它特征。

附图说明

[0025] 为了更透彻地理解本发明,现参阅结合附图和具体实施方式而描述的以下简要说明,其中的相同参考标号表示相同部分。

[0026] 图1是根据本公开的实施例的设备的示意图。

[0027] 图2是图1中的配置文件图表的图式。

[0028] 图3是根据本公开的实施例的设备的示意图。

[0029] 图4是根据本公开的实施例的能量优化方法的流程图。

[0030] 图5是根据本公开的实施例的应用程序分析方法的流程图。

[0031] 图6是根据本公开的另一实施例的能量优化方法的流程图。

[0032] 图7是说明由图4中的方法产生的实验节能的图表。

[0033] 图8是根据本公开的又一个实施例的能量优化方法的流程图。

具体实施方式

[0034] 首先应理解,尽管下文提供一项或多项实施例的说明性实施方案,但所公开的系统 and/或方法可使用任何数目的技术来实施,无论该技术是当前已知还是现有的。本发明决不应限于下文所说明的说明性实施方案、附图和技术,包括本文所说明并描述的示例性设计和实施方案,而是可在所附权利要求书的范围以及其等效物的完整范围内修改。

[0035] 以下首字母缩略词和首字母缩写词适用于:

ASIC: 专用集成电路

CPU: 中央处理器

DSP: 数位信号处理器

DVFS: 动态电压频率调整

EO: 电-光

FPGA: 现场可编程门阵列

Gb/s: 千兆比特/秒

GHz: 吉兆赫

GPU: 图形处理器

GUI: 图形用户界面

MHz:兆赫兹

OE:光-电

OS:操作系统

PMU:性能监视单元

RAM:随机存取存储器

ROM:只读存储器

RX:接收单元

s:秒

SRAM:静态RAM

TCAM:三态内容寻址存储器

TX:发送单元

W:瓦

%:百分比。

[0036] 智能设备制造商可以通过各种技术来执行电功率管理。一种被称为调速器的此类技术使用核心模块以调整使用DVFS的电路级的智能设备组件。调速器基于系统负载调整处理器频率或调整其它组件和这些组件的功能。调速器和其它技术基于功率与电压和频率成比例的实情以执行电功率管理。因此,功率随电压、频率或两者的降低而降低。然而,调速器和其它技术是基于全系统负载,因此是独立于应用程序调整整个系统。换句话说,这些技术是与应用程序无关的。因此,这些技术不提供最佳的功率管理。

[0037] 在此公开的是应用程序特定的,性能感知的能量优化的实施例。控制器以应用程序特定的方式执行配置。换句话说,控制器针对独立于其它应用程序的应用程序执行配置。在此情况下,“独立地”指控制器在应用程序运行时执行配置,并且不管其它应用程序是否处于运行中。控制器以性能感知的方式执行配置。举例来说,控制器可以区分关键应用程序和非关键应用程序,保持或大体上保持关键应用程序的默认性能,以及保持或大体上保持非关键应用程序的性能。实施例大体上保持了应用程序的性能,同时降低了应用程序的能耗。能量等于功率乘以时间,因此,当时间段固定时,减小功耗就是减少能耗。减少能耗延长了电池寿命。

[0038] 图1是根据本公开的实施例的设备100的示意图。设备100可以为智能设备如移动电话,笔记本电脑或平板电脑。设备100包括分析器110、控制器120、配置文件图表130、可配置组件140、GUI 150,以及m个应用程序160-180,其中m是正整数。设备100的组件彼此耦合。设备100可用于经由如蜂窝网络或因特网之类的网络与其它设备通信。

[0039] 分析器110对应用程序160-180进行分析,以为各种配置确定应用程序160-180的性能和功耗。分析器110可以以脱机方式或联机方式执行此操作。联机是指当控制器120控制指定应用程序160-180,离线是指控制器120不控制指定应用程序160-180。分析器110将配置,性能和功耗存储于配置文件图表130中。

[0040] 配置,性能和功耗构成了应用程序160-180的特性。配置指的是用于设备100的与应用程序160-180相关联的资源指配。例如,配置指的是处理器频率、总线带宽、存储器带宽、线程迁移、GPU频率、网络报文速率或其它度量的任意组合。配置适用于硬件组件、软件组件或硬件组件和软件组件两者。性能指的是应用程序160-180在一个时间段内执行的多

个指令,或者指其它度量。功耗指的是每个优化时段应用程序160-180消耗的功率量,或者指其它度量。优化周期还可被称作优化周期,控制时段或控制周期。

[0041] 控制器120还可被称作优化器。控制器120保持或大体上保持关键应用程序160-180的默认特征,并且控制器120在降低或优化能耗时,保持或大体上保持非关键应用程序的目标特征。应用程序160-180的开发者可在开发应用程序160-180时或在其它适当时间指定默认特征,设备100的制造商和设备100的用户可在安装应用程序160-180时或在其它适当时间指定默认特征。用户可以确定他或她期望针对每个应用程序160-180的目标性能,并且控制器120指示设备100的其它组件执行与这些目标性能相关联的目标配置。开发者、制造商或用户指定哪些应用程序160-180是关键的以及哪些应用程序160-180是非关键的,并且可以在任何时间改变这些指定。关键应用程序可以是不牺牲其性能的应用程序160-180。非关键应用程序可以是所有其它应用程序160-180。当应用程序160-180运行时,控制器120可以控制应用程序160-180。控制器120可以指示分析器130以脱机方式对应用程序160-180进行分析,并且控制器120可以联机方式控制应用程序160-180。

[0042] 为执行配置,控制器120指示可配置组件140以相对于应用程序160-180的方式运行。例如,如果可配置组件140是处理器并且控制器120正在控制应用程序160,则控制器120指示处理器在应用程序160运行时,以目标频率运行设备100的所有组件,并且在应用程序160终止之后,以默认频率运行设备100的所有组件。或者,控制器120指示可配置组件140以特定于应用程序160-180中的每一个的方式运行。或者,控制器120指示应用程序160-180以特定方式与可配置组件140一起运行。控制器120可以以类似的方式执行性能和功耗。

[0043] 图2是图1中的配置文件图表130的图式。配置文件图表130包括与应用程序160-180中的一个,例如应用程序160,相关联的n个配置文件,其中n是正整数。设备100包括应用程序170-180的类似配置文件图表。每个配置文件包括配置、性能和功耗。

[0044] 作为一示例,配置文件1包括300MHz的配置,1.0000的性能和1.383W的功耗。300MHz的配置表示应用程序160在频率为300MHz时使用可配置组件140,比如,处理器。或者,300MHz的配置表示处理器140在一段时间内以300MHz的频率运行所有应用程序160-180。1.0000的性能是归一化值,该归一化值表示配置文件1具有最低性能。换句话说,控制器120收集所有配置文件,将配置文件从具有最低性能的第一配置文件排序到具有最高性能的最后一个配置文件,将第一配置文件的性能指定为1.0000,并且将其余配置文件的性能指定为第一配置文件的性能的倍数。举例来说,若配置文件1包括1,000指令/s的性能,则配置文件2包括1,015.5指令/秒的性能,而配置文件n包括6,460.7指令/秒的性能。或者,性能是非归一化值。1.383W的功耗表示应用程序160在一段时间内消耗1.383W。

[0045] 回到图1,可配置组件140是处理器或多个处理器,总线、存储器、GPU、接收器、发送器或其配置可被控制器120更改的另一组件的任意组合。举例来说,控制器120可以更改处理器或GPU的处理速度、总线或存储器的带宽、处理器之间的线程迁移、或接收器或发送器相对于应用程序160-180的网络报文速率。尽管示出了一个可配置组件140,但是设备100可以具有任意数量的可配置组件,并且控制器120可以同时控制任意数量的可配置组件。

[0046] GUI 150向用户呈现用户界面。GUI 150可以提供选择器以提示用户指定哪些应用程序160-180是关键的并且哪个应用程序160-180是非关键的。GUI 150还可提示用户指定将哪些目标性能适用于应用程序160-180。应用程序160-180是通信应用程序、社交媒体应

用程序、娱乐应用程序或其它应用程序。

[0047] 尽管设备100的组件被示为分立组件,但是设备100可以以任意合适的方式组合或分离它们的功能。作为第一示例,分析器110和控制器120是单个部件。作为第二示例,分析器110和控制器120在设备100中,并且控制器120控制第二设备中的应用程序160-180。作为第三示例,设备100是,或者设备100的组件是,更大设备的组件或者更大设备的逻辑分区。

[0048] 图3是根据本公开的实施例的设备300的示意图。设备300适用于实现设备100。设备300包括:用于接收数据的入端口310和用于接收数据的RX 320处理器、逻辑单元、或处理数据的CPU 330;用于发送数据的TX 340和出端口350;以及用于存储数据的存储器360。设备300还可包括耦接至入端口310、RX 320、TX 340以及出端口350的用于光或电信号进出的OE组件和EO组件。

[0049] 处理器330是硬件、中间件、固件和软件的任意合适的组合。处理器330包括一个或多个CPU芯片、芯、FPGA、ASIC或DSP的任意组合。处理器330与入端口310、RX 320、TX 340、出端口350和/或存储器360通信。处理器330包括优化器370,该优化器370实施公开的实施例。因此,优化器370的存在对设备300的功能性有显著改进,且使设备300转变到不同状态。或者,存储器360将优化器370存储为指令,并且处理器330执行这些指令。

[0050] 存储器360包括一个或多个磁盘、磁带机和固态驱动。当设备300选择执行这些程序时,设备300可以使用存储器360作为溢流数据存储设备来存储程序,并且在这些程序的执行期间存储设备300读取的指令和数据。存储器360可以是易失性或非易失性的,并且可以是ROM、RAM、TCAM、SRAM或其任意组合。

[0051] 图4是根据本公开的实施例的能量优化方法400的流程图。当用户安装应用程序160-180时或在其它适当的时间,设备100实现方法400。在步骤405,分析器110和控制器120执行分析。控制器120指示分析器110测量,举例来说应用程序160,的脱机性能脱机功耗以进行不同配置。在步骤410,分析器110从这些配置、性能和功耗生成配置文件图表130。

[0052] 在判定菱形415中,控制器120确定应用程序160是否关键。举例来说,GUI 150提示用户指示应用程序160是否关键。若应用程序160是关键的,则方法400转到步骤420。在步骤420,控制器120执行默认配置,如直至应用程序160终止。分析器110从开发者,制造商或用户获得默认配置,并在配置文件图表130中指示默认配置。然后,控制器120根据配置文件图表130确定默认配置并执行默认配置。例如,在配置文件图表130中,默认配置是2.457MHz或2.457GHz。控制器120指示可配置组件140(在此情况中为处理器)在应用程序160运行时,在2.457GHz的频率下处理。若应用程序160是非关键的,则该方法转到步骤425。

[0053] 在步骤425,控制器120确定目标性能。举例来说,GUI 150为用户提示目标性能并为此目的显示选择器。举例来说,利用配置文件图表130,GUI 150显示用于目标性能的滑动标度和用于目标功耗的相应滑动标度。用于目标性能的滑动标度可以表示慢速,中等,快速或其它类似的单词,而用于目标功耗的滑动标度可以表示标准,改进,加速或其它类似单词。滑动标度可以表示来自配置文件图表130的值或其它合适的值,而不是使用单词,这对于用户来说是更容易理解的。用户将滑动标度提供到用户输入。当目标性能的滑动标度移动以响应用户输入时,举例来说,当用于目标性能的滑动标度从快移动到慢时,用于目标功耗的滑动标度刻度以相应的方式移动,举例来说,从标准移动到加速。GUI 150从用户输入确定目标性能,并将目标性能发送至控制器120。举例来说,GUI 150将来自配置文件图表130

的目标性能1.0155发送至控制器120。或者,用户为目标功耗或目标性能和目标功耗两者提供用户输入。

[0054] 在步骤430,控制器120执行目标配置。控制器120参考配置文件图表130,并确定对应于目标性能1.0155的500MHz的目标配置。控制器120执行目标配置。举例来说,控制器120指示可配置组件140以500MHz的频率运行。

[0055] 在步骤435,分析器110测量应用程序160的测量性能。举例来说,分析器110测得测量性能1.0100。分析器110将测量性能发送至控制器120。

[0056] 在判定菱形440中,控制器120确定所测量性能是否与目标性能匹配。控制器120可以在,举例来说,0.5%的性能余量内这样做。若所测量性能与目标性能匹配,则方法400转到步骤445。在步骤445,控制器120继续执行目标配置,直至应用程序160终止。若所测量性能与目标性能不匹配,则方法转到步骤450。举例来说,测量性能1.0100比指定性能1.0155小大约0.54%,并且0.54%大于0.5%的性能余量。由于应用程序170-180的并发运行、设备100的老化、环境温度或可在分析应用程序160与执行目标配置之间变化的其它因素,可能出现差异。

[0057] 在步骤450,控制器120确定调整的配置。控制器120可使用比例计算、回归分析计算或相对于目标配置、目标性能和目标功耗的其它计算来确定调整的配置。举例来说,控制器120确定调整的配置503MHz。或者,调整的配置是目标配置的增量。举例来说,调整的配置是503MHz-500MHz或3MHz。在步骤455,控制器120执行调整的配置。举例来说,控制器120指示可配置组件140在503MHz的频率内运行。

[0058] 在步骤460,分析器110测应用程序160的调整的性能。举例来说,分析器110测量调整的性能1.0101。分析器110将调整的性能发送至控制器120。

[0059] 在判定菱形465中,控制器120确定调整的性能是否与目标性能匹配。控制器120可以在,举例来说,0.5%的性能余量内这样做。若所调整的性能与目标性能匹配,则方法400转到步骤470。举例来说,调整的性能1.0150比指定性能1.0155小大约0.05%,并且0.05%小于0.5%的性能余量。若测量性能与目标性能不匹配,则方法返回到步骤450。举例来说,测量性能是1.0101。

[0060] 任选地,在步骤470,控制器120更新配置文件图表130。控制器120更新配置文件图表130,从而使得图表130与测量性能匹配。举例来说,对于配置文件2,控制器120将配置从500MHz变更至503MHz。最后,在步骤475,控制器120继续执行调整优化直至应用程序160终止。

[0061] 控制器120可在应用程序160运行时控制应用程序160。控制器120可以在下一个可用优化周期内这样做。举例来说,若优化周期是2s,则分析器110测量从0s到1s的测量性能,并且分析器110在1.5s向控制器120报告测量性能,然后控制器120在2s到4s内使用测量性能控制应用程序160。

[0062] 控制器120针对测量性能提示分析器110,分析器110将测量性能作为响应提供给控制器120。或者,分析器110响应另一事件向控制器120提供测量性能。分析器110可以直接从应用程序160-180获得测量性能或从如PMU的另一组件获得测量性能,PMU可以是独立组件或可以是可配置组件140或另一组件的一部分。

[0063] 控制器120可以指示可配置组件140在可变频率下运行,以便在大体上满足性能的

同时降低能耗。大体上可指在1%、5%、10%或其它合适的度量内。举例来说,在步骤425, GUI 150从用户输入确定目标性能1.0155,并将目标性能发送至控制器120。在下一优化周期的前0.8s内,控制器120指示可配置组件140在525MHz的频率下运行,该频率比配置文件图表130中与目标性能1.0155对应的目标配置500MHz大25MHz。在优化周期剩余的1.2s内,控制器120指示可配置组件140在480MHz的频率下运行,该频率比配置文件图表130中与目标性能1.0155对应的目标配置500MHz小20MHz。因此,在下一个优化周期期间的平均频率可以是大约500MHz,即目标配置。设备100可以使用与方法400类似的迭代方法来实现该变频方法。控制器120可以是智能的,从而使得随着时间推移,控制器120提高了频率可变性,以便在满足性能的同时进一步减少能耗。

[0064] 在一些示例中,控制器120可以在没有用户输入的情况下指示应用程序160的配置。具体地说,控制器120确定应用程序160的默认性能,并且指示可配置组件140在如上所述的可变频率下运行。举例来说,默认性能6.4607对应于2.457GHz的默认配置2.457GHz。在下一优化周期的前0.8s内,控制器120指示可配置组件140在2.6GHz的频率下运行,该频率比默认配置2.457MHz大143MHz。在下一优化周期的剩余的1.2s内,控制器120指示可配置组件140在2.3GHz的频率下运行,该频率比默认配置2.457GHz小157MHz。2.6GHz和2.3GHz的可变频率构成优化配置。因此,在下一个优化周期期间的平均频率可以是大约2.457GHz,即目标配置。

[0065] 图5是根据本公开的实施例的应用程序分析方法500的流程图。当用户安装应用程序160-180中的一个(举例来说,应用程序160)或在其它适当的时间时,设备100实现方法500。设备100执行方法500,以便生成配置文件图表130,控制器120使用配置文件图表130以实现能量优化。

[0066] 在步骤510,分析器110执行分析。分析器110可以为各种配置测量性能和功耗。在步骤520,分析器110从这些配置、性能和功耗生成配置文件图表130。分析器110指定一个配置为默认配置、性能中的一个为默认性能、功耗中的一个为默认功耗。

[0067] 在判定菱形530中,开发者、制造商或用户确定应用程序160是否关键。若应用程序在判定菱形530中被确定为是关键的,则方法500转到步骤540。在步骤540,控制器120选择应用程序160的默认性能。应用程序160的默认性能可为从配置文件图表130中选出的数字。控制器120根据配置文件图表130或根据开发者、制造商或用户,确定默认性能。若在判定菱形530中确定应用程序是非关键的,则方法500转到步骤560。在步骤560,控制器120选择多个性能以供用户进一步选择。

[0068] 图6是根据本公开的另一实施例的能量优化方法600的流程图。设备100实现方法600,并且可以在图5中的步骤540或步骤550之后进行。在步骤610,控制器120接收性能。当应用程序160是关键的时,该性能是步骤540中的默认性能,或者当应用程序160是非关键的时,该性能是从步骤550中的多个性能中选择的目标性能。

[0069] 在判定菱形620中,控制器120确定应用程序160是否已终止。举例来说,用户可以终止应用程序160。若应用程序160已终止,则方法600结束。若应用程序160没有终止,则方法600转到判定菱形630。

[0070] 在判定菱形630中,控制器120确定应用程序是否处于运行中。举例来说,控制器120确定用户是否正在主动使用应用程序160或用户是否已经最小化应用程序160。若应用

程序160不处于运行中,则方法600返回到判定菱形620。若应用程序处于运行中,则方法600转到步骤640。

[0071] 在步骤640,分析器110测量应用程序160的测量性能。在步骤650,控制器120确定满足目标性能的优化配置。控制器120通过参考配置文件图表130并确定满足目标性能的配置来进行,且还将能耗减到最少。最后,在步骤660,控制器120执行优化的配置并且返回到判定菱形620。设备100可以周期性地执行判定菱形620-630和步骤640-660。制造商可以设置周期性。

[0072] 图7是说明由图6中的方法600产生的实验节能的图表700。在Google Nexus 6手机上进行实验,使用Android 6.0.0S并运行Linux内核版本3.10。图表700示出了两个应用程序的目标性能、测量性能、性能改进和节能,即视频转换器应用程序和MobileBench浏览器基准应用程序。

[0073] 对于视频转换器应用程序,方法600提供18.95%的节能,同时保持大致相同的性能。对于MobileBench浏览器基准应用程序,方法600提供12.1%的节能,同时保持大致相同的性能。实际上,对于这两种情形,测量性能略高于目标性能。这些节能提高了电池寿命。

[0074] 图8是根据本公开的又一个实施例的能量优化方法800的流程图。当用户安装应用程序160-180时或在其它适当时间时,设备100实现方法800。在步骤810,确定与应用程序相关联的性能。举例来说,性能是目标性能,在没有用户输入的情况下,分析器110确定该目标性能。或者,GUI 150接收指示与应用程序160相关联的性能的用户输入。

[0075] 在步骤820,执行与应用程序相关联的配置,以便在降低与应用程序相关联的能耗的同时大体上保持性能。举例来说,控制器120执行配置,该配置包括可变频率。或者,控制器120参考配置文件图表130,确定对应于性能的配置,并执行配置。

[0076] 在步骤830,测量由执行产生的测量性能。举例来说,分析器110测量由执行应用程序产生的应用程序160的测量性能。最后,在步骤840,响应测量性能调整配置。举例来说,控制器120确定优化的性能与性能余量内的优化性能不匹配,并确定调整的配置。

[0077] 在示例实施例中,装置包括:非瞬时性存储器元件,该非瞬时性存储器元件包括应用程序;控制器元件,该控制器元件耦接至存储器并且用于调整在装置中与独立于其它应用程序的应用程序相关联的配置,其中配置是装置的资源指配;以及分析器元件,该分析器元件耦合至存储器,并且用于:测量与配置对应的测量性能;以及测量与配置对应的测量功耗。

[0078] 虽然本发明多个具体实施例,但应当理解,所公开的系统和方法也可通过其它多种具体形式体现,而不会脱离本发明的精神或范围。本发明的实例应被视为说明性而非限制性的,且本发明并不限于本文中所给出的细节。例如,各种元件或组件可以在另一系统中组合或整合,或者某些特征可以省略或不实施。

[0079] 此外,在不脱离本发明的范围的情况下,各种实施例中描述和说明为离散或单独的技术、系统、子系统和方法可以与其它系统、组件、技术或方法进行组合或合并。展示或论述为彼此耦合或直接耦合或通信的其它项也可以采用电方式、机械方式或其它方式经由某一接口、设备或中间组件间接地耦合或通信。其它变更、替换、更替示例对本领域技术人员而言是显而易见的,均不脱离此处公开的精神和范围。

100

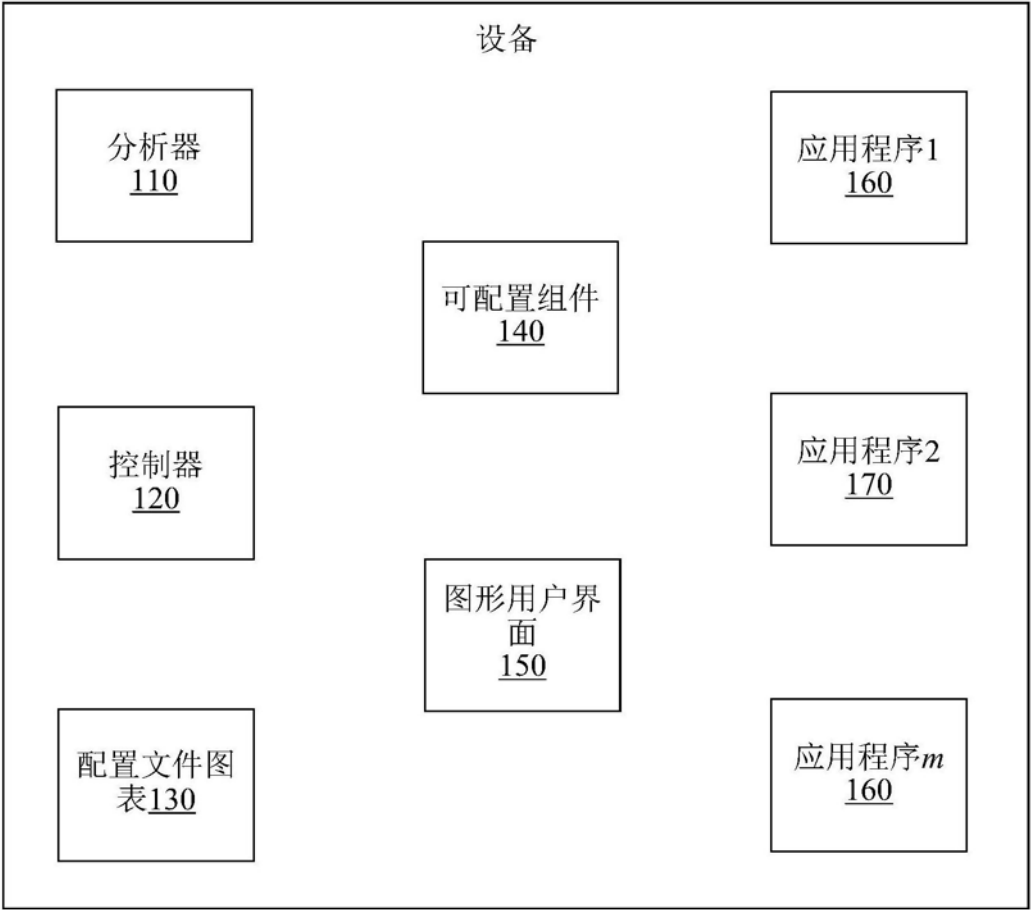


图1

130
↓

	配置 (处理器频率, 以MHz为单位)	性能 (归一化的)	功耗 (W)
配置文件1	300	1.0000	1.383
配置文件2	500	1.0155	1.067
...	...	...	...
配置文件 <i>n</i>	2,457	6.4607	5.601

图2

300

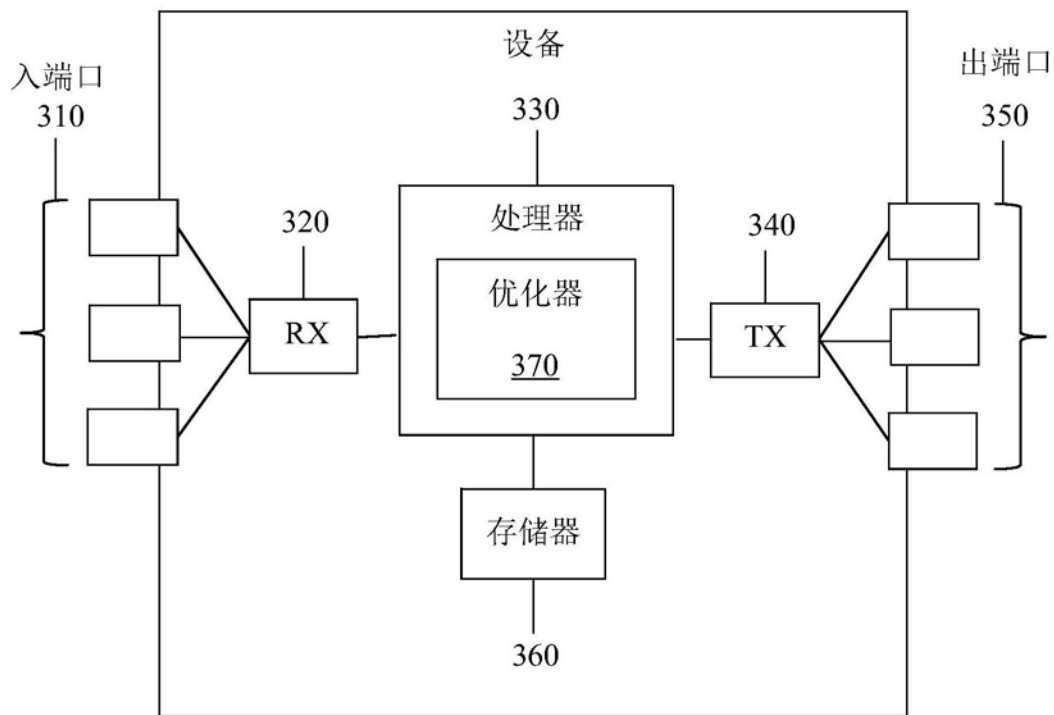


图3

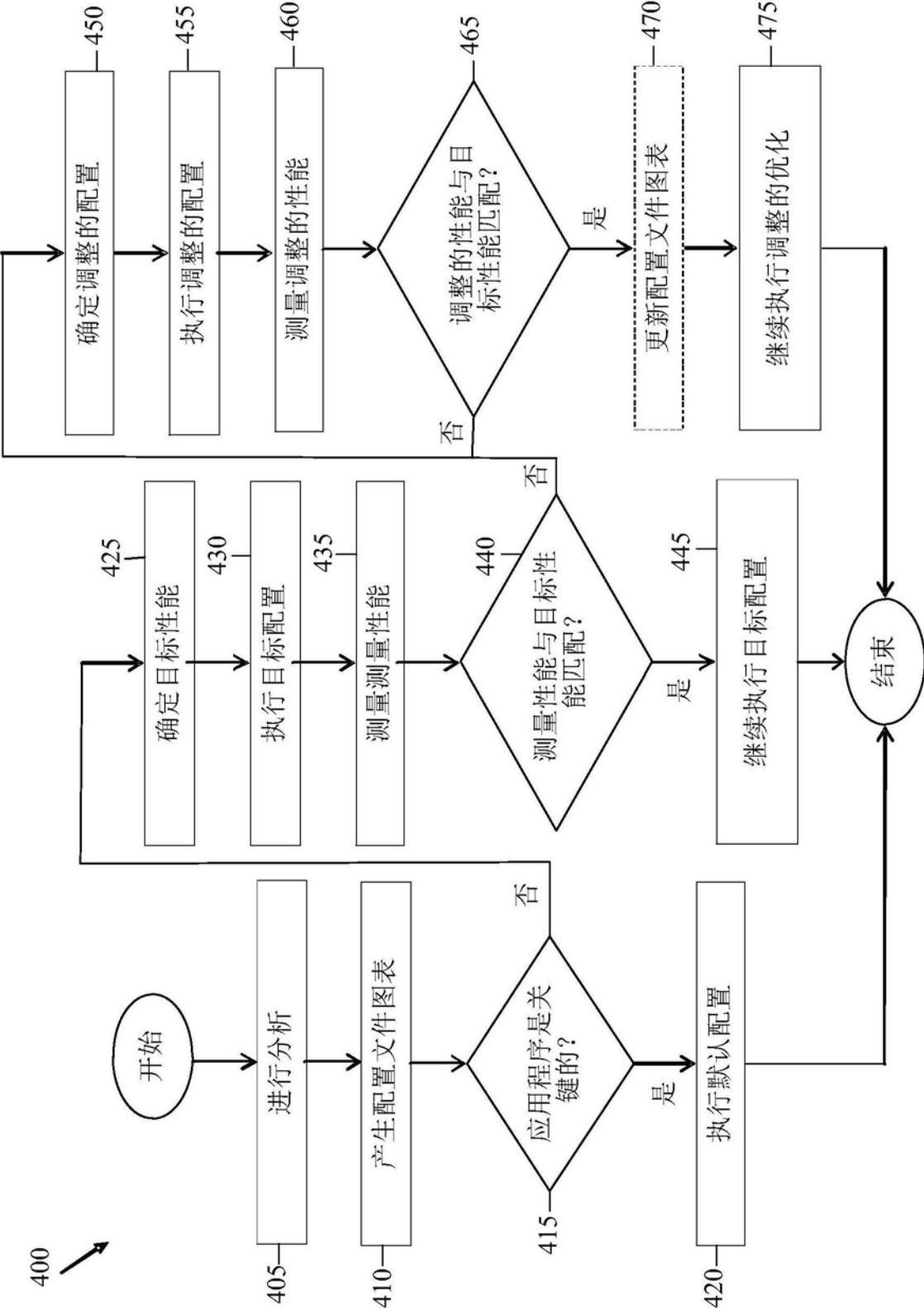


图4

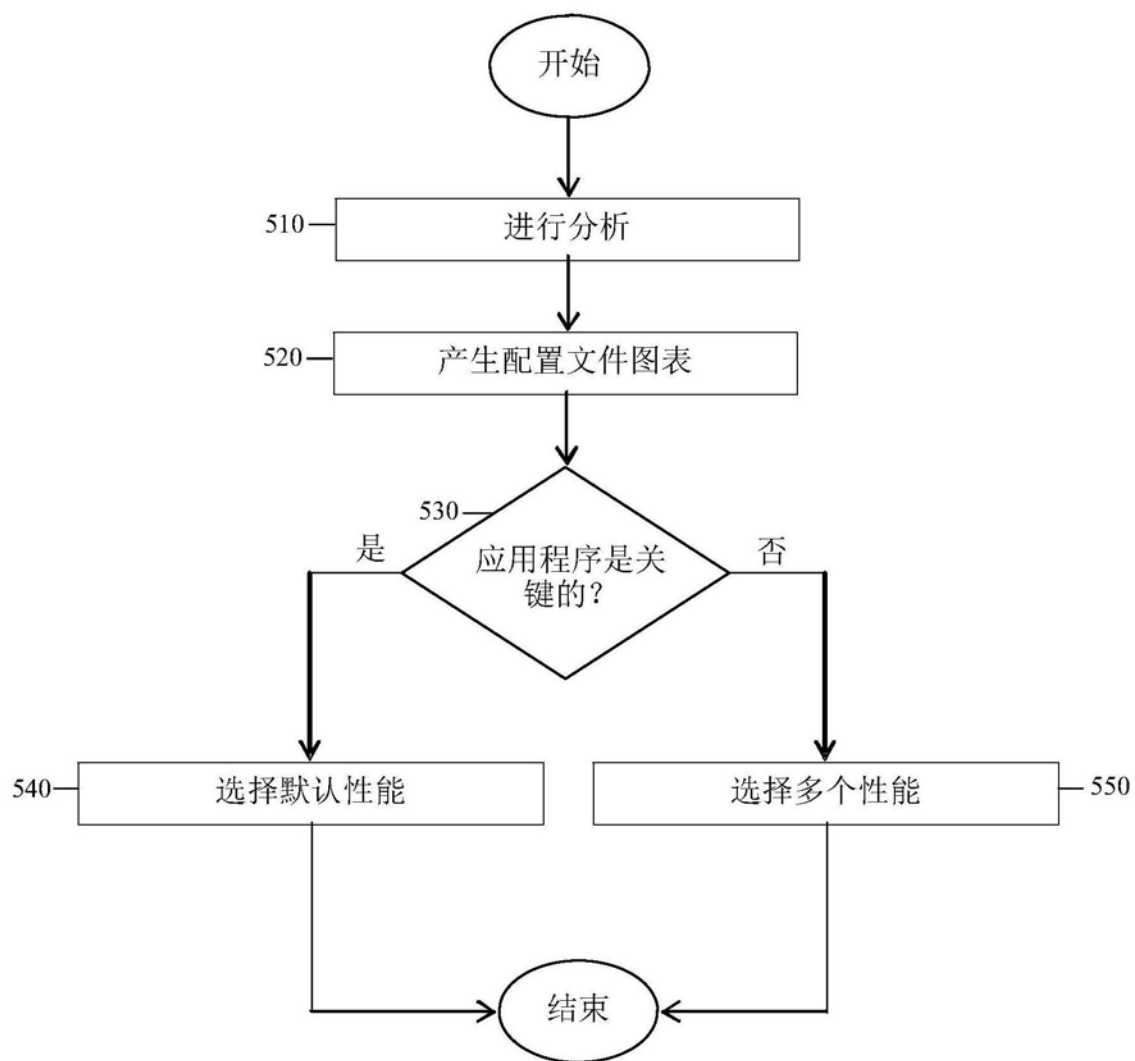
500
↓

图5

600

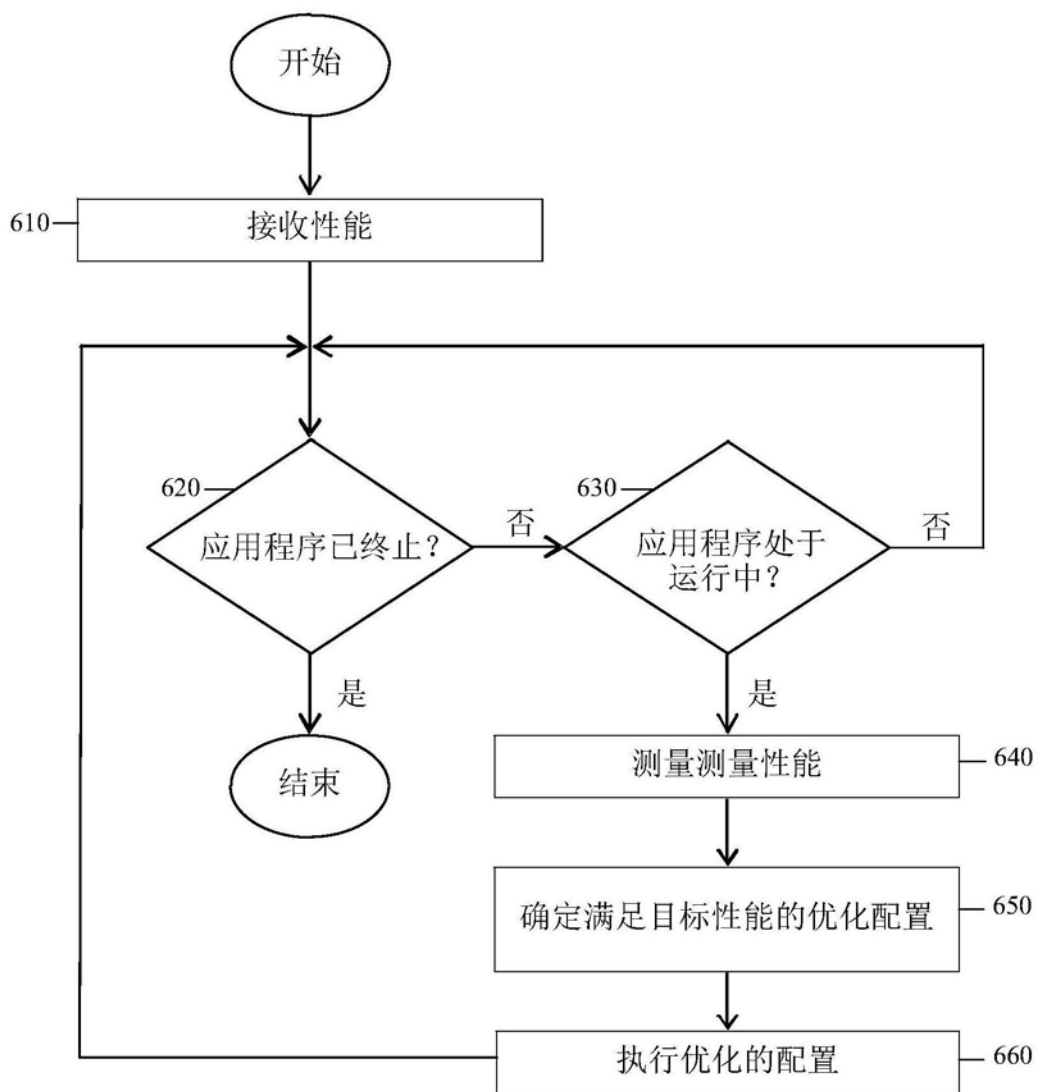


图6

700



应用程序	默认性能 (指令/s)	测量性能 (指令/s)	性能改进 (%)	节能
视频变频器	57	56.26	1.39	18.95
MobileBench浏览器基准	220	213.9	2.72	12.1

图7

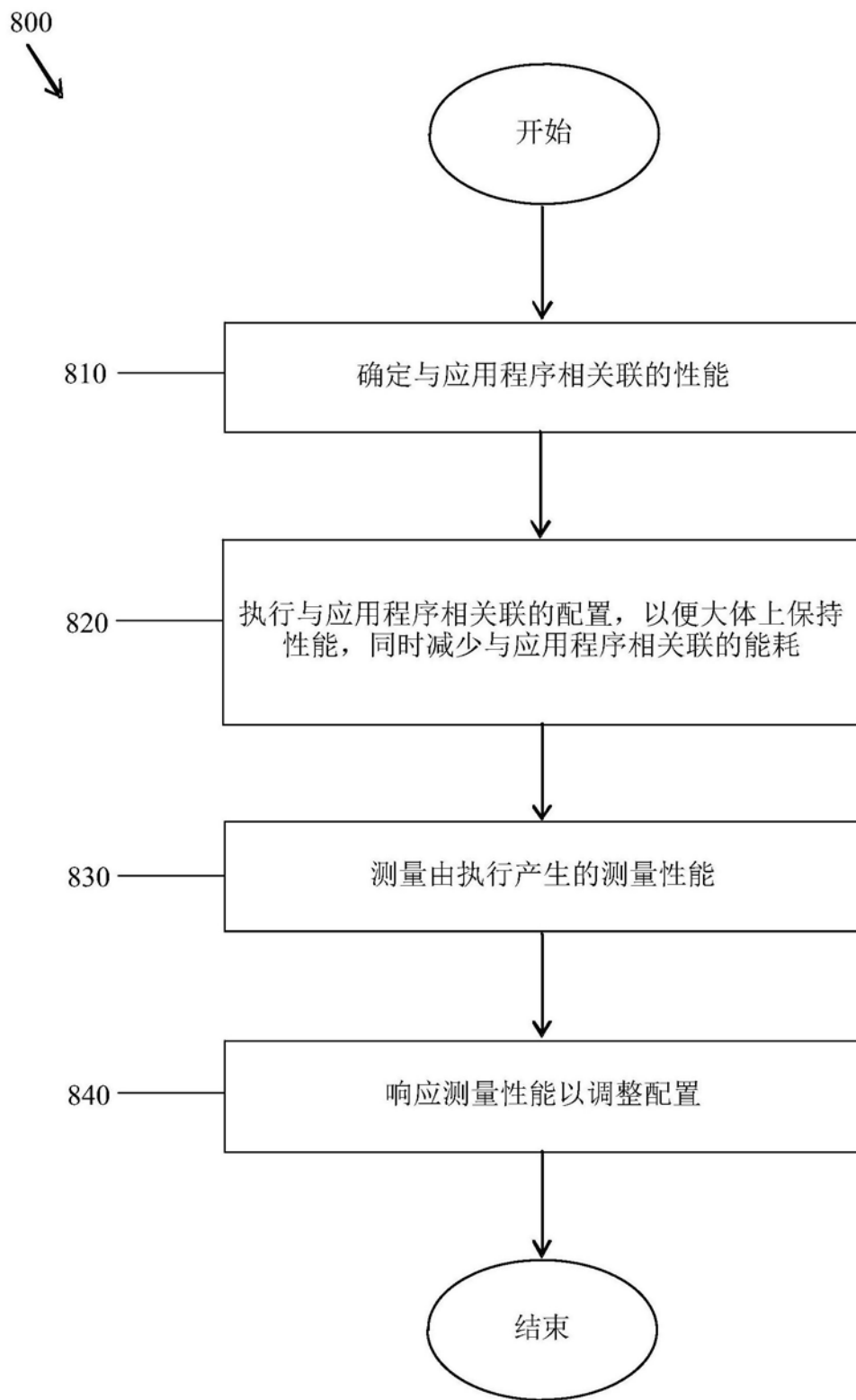


图8