



(12) 发明专利

(10) 授权公告号 CN 111480336 B

(45) 授权公告日 2021.08.13

(21) 申请号 201880080730.3

(22) 申请日 2018.03.26

(65) 同一申请的已公布的文献号
申请公布号 CN 111480336 A

(43) 申请公布日 2020.07.31

(85) PCT国际申请进入国家阶段日
2020.06.17

(86) PCT国际申请的申请数据
PCT/CN2018/080576 2018.03.26

(87) PCT国际申请的公布数据
W02019/183785 ZH 2019.10.03

(73) 专利权人 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 王绪 杜维 周未来 杜松

(74) 专利代理机构 北京中博世达专利商标代理有限公司 11274

代理人 申健

(51) Int.Cl.
H04N 7/01 (2006.01)
G06F 1/3234 (2019.01)

(56) 对比文件
CN 104881104 A, 2015.09.02
US 2015294647 A1, 2015.10.15
CN 105045367 A, 2015.11.11
CN 105913868 A, 2016.08.31
CN 106933526 A, 2017.07.07
US 2016078846 A1, 2016.03.17

审查员 芦祎

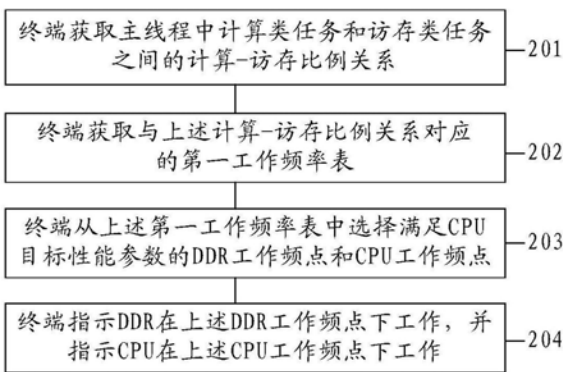
权利要求书4页 说明书16页 附图4页

(54) 发明名称

一种帧率调整方法及终端

(57) 摘要

本申请的实施例提供一种帧率调整方法及终端,涉及通信技术领域,可降低显示过程中的帧率抖动现象。该方法包括:终端获取当前的计算-访存比例关系为第一比例关系,该计算-访存比例关系用于反映终端执行的计算类任务和访存类任务之间的比例;终端获取与第一比例关系对应的第一工作频率表,第一工作频率表中包括DDR和CPU在不同工作频率组合下工作时的CPU性能参数;终端根据CPU性能参数在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合;终端指示该DDR在目标CPU-DDR工作频率组合中的DDR目标工作频点下工作,并指示CPU在目标CPU-DDR工作频率组合中的CPU目标工作频点下工作。



1. 一种帧率调整方法,其特征在于,包括:

终端获取当前的计算-访存比例关系为第一比例关系,所述计算-访存比例关系用于反映所述终端执行的计算类任务和访存类任务之间的比例;

所述终端获取与所述第一比例关系对应的第一工作频率表,所述第一工作频率表中包括当计算-访存比例关系为所述第一比例关系时,双倍速率同步动态随机存储器DDR和中央处理器CPU在不同工作频率组合下工作时的CPU性能参数,所述CPU性能参数用于反映所述CPU处理一帧显示画面所需的帧处理时长;

所述终端根据所述CPU性能参数,在所述第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,所述目标CPU-DDR工作频率组合中包括一组CPU目标工作频点和DDR目标工作频点;

所述终端指示所述DDR在所述DDR目标工作频点下工作,并指示所述CPU在所述CPU目标工作频点下工作。

2. 根据权利要求1所述的方法,其特征在于,终端获取当前的计算-访存比例关系为第一比例关系,包括:

所述终端从所述CPU的性能监测器中获取用于反映所述CPU计算性能的第一参数以及用于反映所述DDR访存性能的第二参数;

所述终端将所述第一参数与所述第二参数的比值作为所述第一比例关系。

3. 根据权利要求1所述的方法,其特征在于,所述终端根据所述CPU性能参数,在所述第一工作频率表中选择满足目标帧处理时长的目标CPU-DDR工作频率组合,包括:

所述终端根据所述第一工作频率表中的CPU性能参数,选择帧处理时长小于所述目标帧处理时长的CPU-DDR工作频率组合作为所述目标CPU-DDR工作频率组合。

4. 根据权利要求1所述的方法,其特征在于,在所述终端根据所述CPU性能参数,在所述第一工作频率表中选择满足目标帧处理时长的目标CPU-DDR工作频率组合之前,还包括:

所述终端检测所述CPU处理当前帧的显示画面时花费的第一实际帧处理时长;

所述终端获取所述CPU和所述DDR处理当前帧的显示画面时的实际CPU-DDR工作频率组合。

5. 根据权利要求4所述的方法,其特征在于,所述终端根据所述CPU性能参数,在所述第一工作频率表中选择满足目标帧处理时长的目标CPU-DDR工作频率组合,包括:

所述终端在所述第一工作频率表中查找与所述实际CPU-DDR工作频率组合对应的实际CPU性能参数;

所述终端在所述第一工作频率表中确定满足第一预设关系的目标CPU性能参数,其中,所述第一预设关系为:所述目标CPU性能参数 \geq (所述第一实际帧处理时长/所述目标帧处理时长)*所述实际CPU性能参数;

所述终端将所述第一工作频率表中与所述目标CPU性能参数对应的CPU-DDR工作频率组合作为所述目标CPU-DDR工作频率组合。

6. 根据权利要求4所述的方法,其特征在于,所述第一工作频率表中还包括所述DDR和所述CPU在不同工作频率组合下工作时的功耗开销;

其中,所述目标CPU-DDR工作频率组合是指满足所述目标帧处理时长且功耗开销最低的CPU-DDR工作频率组合。

7. 根据权利要求1-6中任一项所述的方法,其特征在于,所述方法还包括:

所述终端在预设的第二工作频率表中选择满足所述目标帧处理时长的GPU目标工作频点,所述第二工作频率表中包括GPU在不同工作频点下工作时的GPU性能参数,所述GPU性能参数用于反映GPU处理一帧显示画面所需的帧处理时长;

所述终端指示所述GPU在所述GPU目标工作频点下工作。

8. 根据权利要求7所述的方法,其特征在于,在所述终端在预设的第二工作频率表中选择满足所述目标帧处理时长的GPU目标工作频点之前,还包括:

所述终端检测所述GPU处理当前帧的显示画面时花费的第二实际帧处理时长;

所述终端获取所述GPU处理当前帧的显示画面时的实际GPU工作频点。

9. 根据权利要求8所述的方法,其特征在于,所述终端在预设的第二工作频率表中选择满足所述目标帧处理时长的GPU目标工作频点,包括:

所述终端在所述第二工作频率表中查找与所述实际GPU工作频点对应的实际GPU性能参数;

所述终端在所述第二工作频率表中确定满足第二预设关系的目标GPU性能参数,其中,所述第二预设关系为:所述目标GPU性能参数 \geq (所述第二实际帧处理时长/所述目标帧处理时长)*所述实际GPU性能参数;

所述终端将所述第二工作频率表中与目标GPU性能参数对应的GPU工作频点作为所述GPU目标工作频点。

10. 根据权利要求7所述的方法,其特征在于,所述第二工作频率表中还包括所述GPU在不同工作频点下工作时的功耗开销;

其中,所述GPU目标工作频点是指满足所述目标帧处理时长且功耗开销最低的GPU工作频点。

11. 一种终端,其特征在于,包括:中央处理器CPU、图形处理器GPU、双倍速率同步动态随机存储器DDR以及非易失性存储器;

所述非易失性存储器用于存储计算机执行指令,所述CPU与所述DDR和所述非易失性存储器均耦合,当所述终端运行时,所述CPU执行所述非易失性存储器中存储的所述计算机执行指令,以使所述终端执行以下步骤:

所述CPU获取当前的计算-访存比例关系为第一比例关系,所述计算-访存比例关系用于反映所述终端执行的计算类任务和访存类任务之间的比例;

所述CPU从所述非易失性存储器中获取与所述第一比例关系对应的第一工作频率表,所述第一工作频率表中包括当计算-访存比例关系为所述第一比例关系时,所述DDR和所述CPU在不同工作频率组合下工作时的CPU性能参数,所述CPU性能参数用于反映所述CPU处理一帧显示画面所需的帧处理时长;

所述CPU根据所述CPU性能参数,在所述第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,所述目标CPU-DDR工作频率组合中包括一组CPU目标工作频点和DDR目标工作频点;

所述CPU指示所述DDR在所述DDR目标工作频点下工作,并指示所述CPU在所述CPU目标工作频点下工作。

12. 根据权利要求11所述的终端,其特征在于,所述CPU获取当前的计算-访存比例关系

为第一比例关系,包括:

所述CPU从所述CPU的性能监测器中获取用于反映所述CPU计算性能的第一参数以及用于反映所述DDR访存性能的第二参数;

所述CPU将所述第一参数与所述第二参数的比值作为所述第一比例关系。

13. 根据权利要求11所述的终端,其特征在于,所述CPU根据所述CPU性能参数,在所述第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,包括:

所述CPU根据所述第一工作频率表中的CPU性能参数,选择帧处理时长小于所述目标帧处理时长的CPU-DDR工作频率组合作为所述目标CPU-DDR工作频率组合。

14. 根据权利要求11所述的终端,其特征在于,在所述CPU根据所述CPU性能参数,在所述第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合之前,还包括:

所述CPU检测所述CPU处理当前帧的显示画面时花费的第一实际帧处理时长;

所述CPU获取所述CPU和所述DDR处理当前帧的显示画面时的实际CPU-DDR工作频率组合。

15. 根据权利要求14所述的终端,其特征在于,所述CPU根据所述CPU性能参数,在所述第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,包括:

所述CPU在所述第一工作频率表中查找与所述实际CPU-DDR工作频率组合对应的实际CPU性能参数;

所述CPU在所述第一工作频率表中确定满足第一预设关系的目标CPU性能参数,其中,所述第一预设关系为:所述目标CPU性能参数 \geq (所述第一实际帧处理时长/所述目标帧处理时长)*所述实际CPU性能参数;

所述CPU将所述第一工作频率表中与所述目标CPU性能参数对应的CPU-DDR工作频率组合作为所述目标CPU-DDR工作频率组合。

16. 根据权利要求11-15中任一项所述的终端,其特征在于,

所述CPU在预设的第二工作频率表中选择满足所述目标帧处理时长的GPU目标工作频点,所述第二工作频率表中包括GPU在不同工作频点下工作时的GPU性能参数,所述GPU性能参数用于反映GPU处理一帧显示画面所需的帧处理时长;

所述CPU指示所述GPU在所述GPU目标工作频点下工作。

17. 根据权利要求16所述的终端,其特征在于,在所述CPU在预设的第二工作频率表中选择满足所述目标帧处理时长的GPU目标工作频点之前,还包括:

所述CPU检测所述GPU处理当前帧的显示画面时花费的第二实际帧处理时长;

所述CPU获取所述GPU处理当前帧的显示画面时的实际GPU工作频点。

18. 根据权利要求17所述的终端,其特征在于,所述CPU在预设的第二工作频率表中选择满足所述目标帧处理时长的GPU目标工作频点,包括:

所述CPU在所述第二工作频率表中查找与所述实际GPU工作频点对应的实际GPU性能参数;

所述CPU在所述第二工作频率表中确定满足第二预设关系的目标GPU性能参数,其中,所述第二预设关系为:所述目标GPU性能参数 \geq (所述第二实际帧处理时长/所述目标帧处理时长)*所述实际GPU性能参数;

所述CPU将所述第二工作频率表中与所述目标GPU性能参数对应的GPU工作频点作为所述GPU目标工作频点。

19. 一种芯片,其特征在于,包括:中央处理器CPU和存储器;

所述存储器用于存储计算机执行指令,所述CPU与所述存储器连接,当所述芯片运行时,所述CPU执行所述存储器存储的所述计算机执行指令,以使所述芯片执行如权利要求1-10中任一项所述的帧率调整方法。

20. 一种计算机可读存储介质,所述计算机可读存储介质中存储有指令,其特征在于,当所述指令在终端上运行时,使得所述终端执行如权利要求1-10中任一项所述的帧率调整方法。

一种帧率调整方法及终端

技术领域

[0001] 本申请实施例涉及通信技术领域,尤其涉及一种帧率调整方法及终端。

背景技术

[0002] 帧率(frame per second,FPS)是指显示器每秒钟刷新图像的帧数,当帧率越高时,人眼观看到的变化图像的连续性越好。目前,手机等终端进行显示时一般使用固定的帧率(例如60Hz)刷新显示画面的内容,也就是说,手机中中央处理器(central processing unit,CPU)每处理一帧的时长(本申请实施例中称为帧处理时长)为16.7ms(即1s/60Hz)左右。而当CPU的处理能力不足或手机过热时会导致帧率下降,如果CPU处理一帧的时长超过16.7ms,则会发生丢帧行为,此时用户使用手机时会感受到卡顿。

[0003] 那么,为了使帧率能够稳定在60Hz左右以保证显示画面的流畅度,可检测当前显示画面的实时帧率,当实时帧率超过预设范围的上限时,手机内的调频器可降低CPU的工作频率,而当实时帧率低于预设范围的下限时,调频器可提高CPU的工作频率。

[0004] 但是,CPU处理每一帧显示画面的时间快慢是由CPU的计算能力和双倍速率同步动态随机存储器(Double Data Rate,DDR)的访存能力共同决定的。在计算密集型应用场景下,提高CPU的工作频率能够显著降低显示画面的处理时间,而在访存密集型应用场景(例如游戏场景)下,提高CPU的工作频率对降低显示画面的处理时间的影响较小,此时一味地提高CPU的工作频率不仅无法解决帧率抖动的问题,还会增加手机的功耗。

发明内容

[0005] 本申请的实施例提供一种帧率调整方法及终端,可降低显示过程中的帧率抖动现象,同时降低终端功耗。

[0006] 为达到上述目的,本申请的实施例采用如下技术方案:

[0007] 第一方面,本申请的实施例提供一种帧率调整方法,包括:终端获取当前的计算-访存比例关系为第一比例关系,该计算-访存比例关系可反映出反映终端执行的计算类任务和访存类任务之间的比例;终端获取与该第一比例关系对应的第一工作频率表,第一工作频率表中包括当计算-访存比例关系为第一比例关系时,DDR和CPU在不同工作频率组合下工作时的CPU性能参数,该CPU性能参数用于反映CPU处理一帧显示画面所需的帧处理时长;这样,终端可以根据上述CPU性能参数,在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,该目标CPU-DDR工作频率组合中包括一组CPU目标工作频点和DDR目标工作频点;后续,终端可指示DDR在上述DDR目标工作频点下工作,并指示CPU在上述CPU目标工作频点下工作,使得CPU后续处理每一帧显示画面时时满足上述目标帧处理时长。

[0008] 也就是说,由于CPU主要用于执行计算类型的任务,而DDR主要用于执行访存类型的任务,因此,在本申请中,如果当前计算类型的任务较为密集,则终端可设置CPU在较高的工作频率下工作,使得CPU处理这些计算类型的任务时的速度加快;相应的,如果当前访存

类型的任务较为密集,则终端可设置DDR在较高的工作频率下工作,使得DDR处理这些访存类型的任务时的速度加快。这样终端可以根据当前计算类任务与访存类任务的分布情况动态、合理地设置DDR和CPU的工作频率,使得每一帧显示画面的帧处理时长稳定在目标帧处理时长内,以提高显示画面的流畅度。

[0009] 在一种可能的设计方法中,终端获取当前的计算-访存比例关系为第一比例关系,具体包括:终端从CPU的性能监测器中获取用于反映CPU计算性能的第一参数以及用于反映该DDR访存性能的第二参数;终端将第一参数与第二参数的比值作为第一比例关系,该第一比例关系可反映出当前为访存密集型显示场景或计算密集型显示场景。

[0010] 在一种可能的设计方法中,终端在第一工作频率表中选择满足目标帧处理时长的目标CPU-DDR工作频率组合时,由于上述第一工作频率表中的CPU性能参数用于反映CPU处理一帧显示画面所需的帧处理时长,因此,终端可根据上述第一工作频率表中的CPU性能参数,选择帧处理时长小于该目标帧处理时长的CPU-DDR工作频率组合作为该目标CPU-DDR工作频率组合。

[0011] 在一种可能的设计方法中,在终端在第一工作频率表中选择满足目标帧处理时长的目标CPU-DDR工作频率组合之前,还可以包括:终端检测CPU处理当前帧的显示画面时花费的第一实际帧处理时长;终端获取CPU和该DDR处理当前帧的显示画面时的实际CPU-DDR工作频率组合。

[0012] 此时,终端在上述第一工作频率表中选择满足目标帧处理时长的目标CPU-DDR工作频率组合,具体包括:终端在第一工作频率表中查找与上述实际CPU-DDR工作频率组合对应的实际CPU性能参数;进而,终端可在第一工作频率表中确定满足第一预设关系的目标CPU性能参数,其中,第一预设关系为:该目标CPU性能参数 \geq (第一实际帧处理时长/该目标帧处理时长)*该实际CPU性能参数;这样,终端可将第一工作频率表中与该目标CPU性能参数对应的CPU-DDR工作频率组合作为该目标CPU-DDR工作频率组合。

[0013] 在一种可能的设计方法中,第一工作频率表中还可以包括该DDR和CPU在不同工作频率组合下工作时的功耗开销;其中,该目标CPU-DDR工作频率组合是指满足该目标帧处理时长且功耗开销最低的CPU-DDR工作频率组合,这样不仅可以保证CPU的帧处理时长帧率稳定在目标帧处理时长内,还可以最大程度的降低DDR和CPU工作时产生的功耗开销。

[0014] 在一种可能的设计方法中,上述帧率调整方法还包括:终端在预设的第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点,该第二工作频率表中包括GPU在不同工作频点下工作时的GPU性能参数,该GPU性能参数用于反映GPU处理一帧显示画面所需的帧处理时长;进而,终端可指示该GPU在该GPU目标工作频点下工作。这样,在处理每一帧显示画面时还可使得GPU的帧处理时长稳定在上述目标帧处理时长内,可进一步提升显示画面的流畅度。

[0015] 在一种可能的设计方法中,在终端在上述第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点之前,还包括:终端检测该GPU处理当前帧的显示画面时花费的第二实际帧处理时长;终端获取该GPU处理当前帧的显示画面时的实际GPU工作频点。

[0016] 此时,终端在上述第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点,具体包括:终端在上述第二工作频率表中查找与该实际GPU工作频点对应的实际GPU性能参数;进而,终端可在第二工作频率表中确定满足第二预设关系的目标GPU性能参数,

其中,第二预设关系为:该目标GPU性能参数 \geq (第二实际帧处理时长/该目标帧处理时长)*该实际GPU性能参数;这样,终端可将第二工作频率表中与该目标GPU性能参数对应的GPU工作频点作为该GPU目标工作频点。

[0017] 在一种可能的设计方法中,第二工作频率表中还包括该GPU在不同工作频点下工作时的功耗开销;其中,该GPU目标工作频点是指满足该目标帧处理时长且功耗开销最低的GPU工作频点,这样不仅可以保证GPU的帧处理时长帧率稳定在目标帧处理时长内,还可以最大程度的降低GPU工作时产生的功耗开销。

[0018] 第二方面,本申请实施例提供一种终端,包括:获取单元,用于获取当前的计算-访存比例关系为第一比例关系,该计算-访存比例关系用于反映终端执行的计算类任务和访存类任务之间的比例;以及获取与第一比例关系对应的第一工作频率表,第一工作频率表中包括当计算-访存比例关系为第一比例关系时,DDR和CPU在不同工作频率组合下工作时的CPU性能参数,CPU性能参数用于反映CPU处理一帧显示画面所需的帧处理时长;确定单元,用于根据上述CPU性能参数,在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,该目标CPU-DDR工作频率组合中包括一组CPU目标工作频点和DDR目标工作频点;指示单元,用于指示该DDR在该DDR目标工作频点下工作,并指示CPU在CPU目标工作频点下工作。

[0019] 在一种可能的设计方法中,终端还包括计算单元,上述获取单元,还用于从CPU的性能监测器中获取用于反映CPU计算性能的第一参数以及用于反映该DDR访存性能的第二参数;该计算单元,用于将第一参数与第二参数的比值作为第一比例关系。

[0020] 在一种可能的设计方法中,该确定单元,具体用于根据第一工作频率表中的CPU性能参数,选择帧处理时长小于该目标帧处理时长的CPU-DDR工作频率组合作为该目标CPU-DDR工作频率组合。

[0021] 在一种可能的设计方法中,该获取单元,还用于检测CPU处理当前帧的显示画面时花费的第一实际帧处理时长;获取CPU和该DDR处理当前帧的显示画面时的实际CPU-DDR工作频率组合。

[0022] 此时,该确定单元,具体用于:在第一工作频率表中查找与该实际CPU-DDR工作频率组合对应的实际CPU性能参数;在第一工作频率表中确定满足第一预设关系的目标CPU性能参数,其中,第一预设关系为:该目标CPU性能参数 \geq (第一实际帧处理时长/该目标帧处理时长)*该实际CPU性能参数;将第一工作频率表中与该目标CPU性能参数对应的CPU-DDR工作频率组合作为该目标CPU-DDR工作频率组合。

[0023] 在一种可能的设计方法中,该确定单元,还用于在预设的第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点,第二工作频率表中包括GPU在不同工作频点下工作时的GPU性能参数,该GPU性能参数用于反映GPU处理一帧显示画面所需的帧处理时长;该指示单元,还用于指示该GPU在该GPU目标工作频点下工作。

[0024] 在一种可能的设计方法中,该获取单元,还用于:检测该GPU处理当前帧的显示画面时花费的第二实际帧处理时长;端获取该GPU处理当前帧的显示画面时的实际GPU工作频点。

[0025] 此时,该确定单元,具体用于:在第二工作频率表中查找与该实际GPU工作频点对应的实际GPU性能参数;在第二工作频率表中确定满足第二预设关系的目标GPU性能参数,

其中,第二预设关系为:该目标GPU性能参数 \geq (第二实际帧处理时长/该目标帧处理时长)*该实际GPU性能参数;将第二工作频率表中与该目标GPU性能参数对应的GPU工作频点作为该GPU目标工作频点。

[0026] 第三方面,本申请实施例提供一种终端,包括:CPU、GPU、双倍速率同步动态随机存储器DDR以及非易失性存储器;该非易失性存储器用于存储计算机执行指令,CPU与该DDR和该非易失性存储器均耦合,当终端运行时,CPU执行该非易失性存储器中存储的该计算机执行指令,以使终端执行以下步骤:

[0027] CPU获取当前的计算-访存比例关系为第一比例关系,该计算-访存比例关系用于反映终端执行的计算类任务和访存类任务之间的比例;CPU从该非易失性存储器中获取与第一比例关系对应的第一工作频率表,第一工作频率表中包括当计算-访存比例关系为第一比例关系时,该DDR和CPU在不同工作频率组合下工作时的CPU性能参数,CPU性能参数用于反映CPU处理一帧显示画面所需的帧处理时长;CPU在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,该目标CPU-DDR工作频率组合中包括一组CPU目标工作频点和DDR目标工作频点;CPU指示该DDR在该DDR目标工作频点下工作,并指示CPU在CPU目标工作频点下工作。

[0028] 在一种可能的设计方法中,CPU获取当前的计算-访存比例关系为第一比例关系,包括:CPU从CPU的性能监测器中获取用于反映CPU计算性能的第一参数以及用于反映该DDR访存性能的第二参数;CPU将第一参数与第二参数的比值作为第一比例关系。

[0029] 在一种可能的设计方法中,CPU在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,包括:CPU根据第一工作频率表中的CPU性能参数,选择帧处理时长小于该目标帧处理时长的CPU-DDR工作频率组合作为该目标CPU-DDR工作频率组合。

[0030] 在一种可能的设计方法中,在CPU在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合之前,还包括:CPU检测CPU处理当前帧的显示画面时花费的第一实际帧处理时长;CPU获取CPU和该DDR处理当前帧的显示画面时的实际CPU-DDR工作频率组合。

[0031] 在一种可能的设计方法中,CPU在第一工作频率表中选择满足预设的目标帧处理时长的目标CPU-DDR工作频率组合,包括:CPU在第一工作频率表中查找与该实际CPU-DDR工作频率组合对应的实际CPU性能参数;CPU在第一工作频率表中确定满足第一预设关系的目标CPU性能参数,其中,第一预设关系为:该目标CPU性能参数 \geq (第一实际帧处理时长/该目标帧处理时长)*该实际CPU性能参数;CPU将第一工作频率表中与该目标CPU性能参数对应的CPU-DDR工作频率组合作为该目标CPU-DDR工作频率组合。

[0032] 在一种可能的设计方法中,该方法还包括:CPU在预设的第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点,第二工作频率表中包括GPU在不同工作频点下工作时的GPU性能参数,该GPU性能参数用于反映GPU处理一帧显示画面所需的帧处理时长;CPU指示该GPU在该GPU目标工作频点下工作。

[0033] 在一种可能的设计方法中,在CPU在预设的第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点之前,还包括:CPU检测该GPU处理当前帧的显示画面时花费的第二实际帧处理时长;CPU获取该GPU处理当前帧的显示画面时的实际GPU工作频点。

[0034] 在一种可能的设计方法中,CPU在预设的第二工作频率表中选择满足该目标帧处理时长的GPU目标工作频点,包括:CPU在第二工作频率表中查找与该实际GPU工作频点对应的实际GPU性能参数;CPU在第二工作频率表中确定满足第二预设关系的目标GPU性能参数,其中,第二预设关系为:该目标GPU性能参数 \geq (第二实际帧处理时长/该目标帧处理时长)*该实际GPU性能参数;CPU将第二工作频率表中与该目标GPU性能参数对应的GPU工作频点作为该GPU目标工作频点。

[0035] 第四方面,本申请实施例提供一种芯片,包括CPU和存储器;该存储器用于存储计算机执行指令,CPU与该存储器连接,当该芯片运行时,CPU执行该存储器存储的该计算机执行指令,以使该芯片执行上述任一项帧率调整方法。

[0036] 第五方面,本申请实施例提供一种计算机可读存储介质,该计算机可读存储介质中存储有指令,当该指令在上述任一项终端上运行时,使得终端执行上述任一项帧率调整方法。

[0037] 第六方面,本申请实施例提供一种包含指令的计算机程序产品,当其在上述任一项终端上运行时,使得终端执行上述任一项帧率调整方法。

[0038] 本申请的实施例中,上述终端内各部件的名字对设备本身不构成限定,在实际实现中,这些部件可以以其他名称出现。只要各个部件的功能和本申请的实施例类似,即属于本申请权利要求及其等同技术的范围之内。

[0039] 另外,第二方面至第六方面中任一种设计方式所带来的技术效果可参见上述第一方面中不同设计方法所带来的技术效果,此处不再赘述。

附图说明

[0040] 图1为本申请实施例提供的一种终端的结构示意图一;

[0041] 图2为本申请实施例提供的一种帧率调整方法的流程示意图一;

[0042] 图3为本申请实施例提供的一种帧率调整方法的原理示意图一;

[0043] 图4为本申请实施例提供的一种帧率调整方法的原理示意图二;

[0044] 图5为本申请实施例提供的一种帧率调整方法的原理示意图三;

[0045] 图6为本申请实施例提供的一种帧率调整方法的流程示意图二;

[0046] 图7为本申请实施例提供的一种帧率调整方法的原理示意图四;

[0047] 图8为本申请实施例提供的一种帧率调整方法的效果示意图;

[0048] 图9为本申请实施例提供的一种终端的结构示意图二;

[0049] 图10为本申请实施例提供的一种终端的结构示意图三。

具体实施方式

[0050] 以下,术语“第一”、“第二”仅用于描述目的,而不能理解为指示或暗示相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或者隐含地包括一个或者更多个该特征。在本申请实施例的描述中,除非另有说明,“多个”的含义是两个或两个以上。

[0051] 本申请实施例提供一种帧率调整方法可应用于手机、平板电脑、可穿戴设备、车载设备、增强现实(augmented reality,AR)\虚拟现实(virtual reality,VR)设备、笔记本

电脑、车载设备、超级移动个人计算机(ultra-mobile personal computer,UMPC)、上网本、个人数字助理(personal digital assistant,PDA)等具有显示功能的任意终端上,本申请实施例对此不作任何限制。

[0052] 如图1所示,本申请实施例中的终端可以为手机100。下面以手机100为例对实施例进行具体说明。应该理解的是,图示手机100仅是上述终端的一个范例,并且手机100可以具有比图中所示出的更多的或者更少的部件,可以组合两个或更多的部件,或者可以具有不同的部件配置。

[0053] 如图1所示,手机100具体可以包括:处理器101、射频(radio frequency,RF)电路102、存储器103、触摸屏104、蓝牙装置105、一个或多个传感器106、Wi-Fi装置107、定位装置108、音频电路109、外设接口110以及电源系统111等部件。这些部件可通过一根或多根通信总线或信号线(图1中未示出)进行通信。本领域技术人员可以理解,图1中示出的硬件结构并不构成对手机的限定,手机100可以包括比图示更多或更少的部件,或者组合某些部件,或者不同的部件布置。

[0054] 下面结合图1对手机100的各个部件进行具体的介绍:

[0055] 处理器101(也可称为CPU)是手机100的控制中心,利用各种接口和线路连接手机100的各个部分,通过运行或执行存储在存储器103内的应用程序,以及调用存储在存储器103内的数据,执行手机100的各种功能和处理数据。在一些实施例中,处理器101可包括一个或多个处理单元;举例来说,处理器101可以是华为技术有限公司制造的麒麟960芯片。在本申请一些实施例中,上述处理器101还可以包括指纹验证芯片,用于对采集到的指纹进行验证。

[0056] 另外,处理器101中还可以设置性能监视器(performance monitor),性能监视器可以实时监测CPU在运行过程中的各项运行参数,例如CPU周期(CPU cycle,又称为机器周期)、内存占用率以及处于空闲(idle)状态的时长等。

[0057] 其中,性能监视器监测到的一些参数,例如一段时间内执行指令(instruction)的数目,可以反映出CPU的计算性能;相应的,性能监视器监测到的另一些参数,例如,缓存失效(cachemiss)数目,可以反映出CPU与存储器之间的访存性能。

[0058] 那么,通过性能监视器监测到的这些参数,可以计算出CPU在运行过程实时执行的计算类任务与访存类任务之间的关系,从而判断出当前的运行场景为计算密集型场景还是访存密集型场景。

[0059] 射频电路102可用于在收发信息或通话过程中,无线信号的接收和发送。特别地,射频电路102可以将基站的下行数据接收后,给处理器101处理;另外,将涉及上行的数据发送给基站。通常,射频电路包括但不限于天线、至少一个放大器、收发信机、耦合器、低噪声放大器、双工器等。此外,射频电路102还可以通过无线通信和其他设备通信。所述无线通信可以使用任一通信标准或协议,包括但不限于全球移动通讯系统、通用分组无线服务、码分多址、宽带码分多址、长期演进、电子邮件、短消息服务等。

[0060] 存储器103用于存储应用程序以及数据,处理器101通过运行存储在存储器103的应用程序以及数据,执行手机100的各种功能以及数据处理。存储器103主要包括存储程序区以及存储数据区,其中,存储程序区可存储操作系统、至少一个功能所需的应用程序(比如声音播放功能、图像播放功能等);存储数据区可以存储根据使用手机100时所创建的数据

据(比如音频数据、电话本等)。此外,存储器103可以包括高速随机存取存储器(random access memory, RAM),还可以包括非易失存储器,例如磁盘存储器件、闪存器件或其他易失性固态存储器件等。存储器103可以存储各种操作系统,例如,苹果公司所开发的iOS®操作系统,谷歌公司所开发的Android®操作系统等。上述存储器103可以是独立的,通过上述通信总线与处理器101相连接;存储器103也可以和处理器101集成在一起。

[0061] 在本申请实施例中,存储器103中一般包括直接与CPU通信的内存,例如,该内存可以是DDR存储器(也可称为DDR SDRAM或DDR)等任意类型的存储介质,本申请实施例对此不做任何限制。

[0062] 以DDR为例,当CPU在访存密集型(memory intensive)场景下运行时,说明CPU需要频繁对DDR进行读写操作,此时,如果提高DDR的工作频率则可以缩短CPU处理当前帧的显示画面时花费的帧处理时长,从而提高终端的帧速率。

[0063] 另外,在本申请实施例中,手机100还可以包括图形处理器(Graphics Processing Unit, GPU) 115。其中,GPU 115是一种专门在个人电脑、工作站、游戏机和一些移动设备(如平板电脑、智能手机等)上进行图像运算工作的微处理器。它可将手机100所需要的显示信息进行转换驱动,并向显示器104-2提供行扫描信号,控制显示器104-2的正确显示。

[0064] 具体的,在显示过程中,CPU 101可将相应的绘图命令发送给GPU 115,例如,该绘图命令可以为“在坐标位置(x,y)处画个长和宽为 $a \times b$ 大小的长方形”,那么,GPU 115根据该绘图指令便可以迅速计算出该图形的所有像素,并在显示器104-2上指定位置画出相应的图形。

[0065] 可以看出,每一帧显示画面的处理速度除了与CPU 101的处理能力相关外,还与GPU 115的处理能力相关。CPU 101将绘图指令发送给GPU 115之后,如果GPU 115的处理速度过慢,则绘图指令会堆积在GPU 115处无法及时处理,此时GPU成为绘帧瓶颈。因此,提高GPU 115的处理能力(例如提高GPU 115的工作频率)也可以提高手机的帧率。

[0066] 需要说明的是,GPU 115可以以功能模块的形式集成在处理器101内,也可以以独立的实体形态(例如,显卡)设置在手机100内,本发明实施例对此不作任何限制。

[0067] 触摸屏104具体可以包括触控板104-1和显示器104-2。

[0068] 其中,触控板104-1可采集手机100的用户在其上或附近的触摸操作(比如用户使用手指、触控笔等任何适合的物体在触控板104-1上或在触控板104-1附近的操作),并将采集到的触摸信息发送给其他器件(例如处理器101)。其中,用户在触控板104-1附近的触摸操作可以称之为悬浮触控;悬浮触控可以是指,用户无需为了选择、移动或拖动目标(例如图标等)而直接接触触控板,而只需用户位于终端附近以便执行所想要的功能。此外,可以采用电阻式、电容式、红外线以及表面声波等多种类型来实现触控板104-1。

[0069] 显示器(也可称为显示屏)104-2可用于显示由用户输入的信息或提供给用户的信息以及手机100的各种菜单。可以采用液晶显示器、有机发光二极管等形式来配置显示器104-2。触控板104-1可以覆盖在显示器104-2之上,当触控板104-1检测到在其上或附近的触摸操作后,传送给处理器101以确定触摸操作的类型,随后处理器101可以根据触摸操作的类型在显示器104-2上提供相应的视觉输出。虽然在图1中,触控板104-1与显示屏104-2是作为两个独立的部件来实现手机100的输入和输出功能,但是在某些实施例中,可以将触

控板104-1与显示屏104-2集成而实现手机100的输入和输出功能。

[0070] 手机100还可以包括蓝牙装置105,用于实现手机100与其他短距离的终端(例如手机、智能手表等)之间的数据交换。本申请实施例中的蓝牙装置可以是集成电路或者蓝牙芯片等器件。

[0071] 手机100还可以包括至少一种传感器106,比如指纹采集器件、光传感器、运动传感器以及其他传感器。具体地,可以在手机100的背面(例如后置摄像头的下方)配置指纹采集器件,或者在手机100的正面(例如触摸屏104的下方)配置指纹采集器件。又例如,可以在触摸屏104中配置指纹采集器件来实现指纹识别功能,即指纹采集器件可以与触摸屏104集成在一起来实现手机100的指纹识别功能;光传感器可包括环境光传感器及接近传感器,其中,环境光传感器可根据环境光线的明暗来调节触摸屏104的显示器的亮度,接近传感器可在手机100移动到耳边时,关闭显示器的电源。作为运动传感器的一种,加速计传感器可检测各个方向上(一般为三轴)加速度的大小,静止时可检测出重力的大小及方向,可用于识别手机姿态的应用(比如横竖屏切换、相关游戏、磁力计姿态校准)、振动识别相关功能(比如计步器、敲击)等;至于手机100还可配置的陀螺仪、气压计、湿度计、温度计、红外线传感器等其他传感器,在此不再赘述。

[0072] Wi-Fi装置107,用于为手机100提供遵循Wi-Fi相关标准协议的网络接入,手机100可以通过Wi-Fi装置107接入到Wi-Fi接入点,进而帮助用户收发电子邮件、浏览网页和访问流媒体等,它为用户提供了无线的宽带互联网访问。在其他一些实施例中,该Wi-Fi装置107也可以作为Wi-Fi无线接入点,可以为其他终端提供Wi-Fi网络接入。

[0073] 定位装置108,用于为手机100提供地理位置。可以理解的是,该定位装置108具体可以是全球定位系统(global positioning system,GPS)或北斗卫星导航系统、俄罗斯GLONASS等定位系统的接收器。定位装置108在接收到上述定位系统发送的地理位置后,将该信息发送给处理器101进行处理,或者发送给存储器103进行保存。在另外的一些实施例中,该定位装置108还可以是辅助全球卫星定位系统(assisted global positioning system,AGPS)的接收器,AGPS系统通过作为辅助服务器来协助定位装置108完成测距和定位服务,在这种情况下,辅助定位服务器通过无线通信网络与终端例如手机100的定位装置108(即GPS接收器)通信而提供定位协助。在另外的一些实施例中,该定位装置108也可以是基于Wi-Fi接入点的定位技术。由于每一个Wi-Fi接入点都有一个全球唯一的媒体接入控制(media access control,MAC)地址,终端在开启Wi-Fi的情况下即可扫描并收集周围的Wi-Fi接入点的广播信号,因此可以获取到Wi-Fi接入点广播出来的MAC地址;终端将这些能够标示Wi-Fi接入点的数据(例如MAC地址)通过无线通信网络发送给位置服务器,由位置服务器检索出每一个Wi-Fi接入点的地理位置,并结合Wi-Fi广播信号的强弱程度,计算出该终端的地理位置并发送到该终端的定位装置108中。

[0074] 音频电路109、扬声器113、麦克风114可提供用户与手机100之间的音频接口。音频电路109可将接收到的音频数据转换后的电信号,传输到扬声器113,由扬声器113转换为声音信号输出;另一方面,麦克风114将收集的声音信号转换为电信号,由音频电路109接收后转换为音频数据,再将音频数据输出至RF电路102以发送给比如另一手机,或者将音频数据输出至存储器103以便进一步处理。

[0075] 外设接口110,用于为外部的输入/输出设备(例如键盘、鼠标、外接显示器、外部存

储器、用户识别模块卡等)提供各种接口。例如通过通用串行总线(universal serial bus, USB)接口与鼠标连接,通过用户识别模块卡卡槽上的金属触点与电信运营商提供的用户识别模块卡(subscriber identification module, SIM)卡进行连接。外设接口110可以被用来将上述外部的输入/输出外围设备耦接到处理器101和存储器103。

[0076] 手机100还可以包括给各个部件供电的电源装置111(比如电池和电源管理芯片),电池可以通过电源管理芯片与处理器101逻辑相连,从而通过电源装置111实现管理充电、放电、以及功耗管理等功能。

[0077] 尽管图1未示出,手机100还可以包括摄像头(前置摄像头和/或后置摄像头)、闪光灯、微型投影装置、近场通信(NFC near field communication, NFC)装置等,在此不予赘述。

[0078] 手机100进行显示时,上述显示器104-2可按照GPU 115提供的扫描信号显示GPU 115内缓存的每一帧显示画面。由于显示器104-2显示时的帧率一般设置在某一固定数值(例如60Hz),即CPU或GPU生成每一帧显示画面花费的帧处理时长需要小于在16.7ms左右。那么,如果CPU或GPU处理一帧显示画面的帧处理时长远大于16.7ms,则会出现掉帧现象,此时显示器104-2显示时的帧率会出现抖动,影响了显示画面的流畅度。

[0079] 由于CPU在处理每一帧显示画面时需要与DDR交互共同完成,因此当手机100显示时的帧率发生抖动时,仅仅通过调整CPU工作频率的方式并不能保证每一帧显示画面的帧处理时长能够在16.7ms之内完成。

[0080] 对此,在本申请实施例中,终端可实时检测当前显示场景中计算类型的任务与访存类型的任务之间的关系。由于CPU主要用于执行计算类型的任务,而DDR主要用于执行访存类型的任务,那么,当帧率发生抖动时,如果当前显示场景中计算类型的任务较为密集,则终端可设置CPU在较高的工作频率下工作,使得CPU处理这些计算类型的任务时的速度加快;相应的,如果当前显示场景中访存类型的任务较为密集,则终端可设置DDR在较高的工作频率下工作,使得DDR处理这些访存类型的任务时的速度加快。这样,终端可以根据当前显示场景中计算类任务与访存类任务的分布情况动态、合理地设置DDR和CPU的工作频率,使得每一帧显示画面的帧处理时长稳定在16.7ms左右,以提高显示画面的流畅度。

[0081] 以下,将结合具体实施例详细阐述本申请实施例提供的一种帧率调整方法,如图2所示,该方法包括:

[0082] S201、终端获取主线程中计算类任务和访存类任务之间的计算-访存比例关系。

[0083] 其中,当计算-访存比例的取值越大时,说明当前的计算类任务较多,CPU需要频繁执行计算类任务才能保证每一帧显示画面能够在预设的帧处理时长内(例如16.7ms)处理完成。相应的,当计算-访存比例的取值越小时,说明当前的访存类任务较多,CPU需要频繁地与DDR交互执行访存类任务,才能保证每一帧显示画面能够在预设的帧处理时长内处理完成。

[0084] 在步骤S201中,当一个程序启动时,终端的操作系统(OS)会创建一个主线程(mainthread)开始运行,那么,如图3所示,终端可以获取该主线程的标识(例如主线程的ID),进而,终端使用主线程的ID在CPU的性能监测器(performance monitor)中获取CPU在运行该主线程时的目标参数。

[0085] 其中,上述目标参数包括用于反映CPU计算性能的第一参数(例如instruction数

目等),以及用于反映DDR访存性能的第二参数(例如cache miss数目等)。

[0086] 这样,仍如图3所示,终端根据获取到的第一参数和第二参数可计算出主线程运行时计算类任务与访存类任务的比例关系,即计算-访存比例关系。例如,可以设置比例参数R用于反映计算-访存比例关系,其中 $R = \text{第一参数} / \text{第二参数} = \text{instruction数目} / \text{cache miss数目}$ 。

[0087] 当然,除了上述instruction数目和cache miss数目之外,终端还可以根据性能监测器监测到的其他参数,例如CPU运行周期等,计算上述计算-访存比例关系。并且,计算上述计算-访存比例关系的方式也不限于使用第一参数除以第二参数这种计算方式,本申请实施例对此不做任何限制。

[0088] 另外,终端获取到主线程的ID后,可以周期性的从性能监测器中获取上述目标参数,并计算主线程中计算类任务和访存类任务之间的计算-访存比例关系,进而执行下述步骤S202-S204,这样终端可以实时地根据当前的计算-访存比例关系调整CPU和DDR的工作频率,使得显示帧率稳定在目标频率(例如60Hz)左右。

[0089] 示例性的,在周期性获取上述目标参数时,终端可将该周期设置为与目标频率(例如60Hz)所对应的目标帧处理时长(例如16.7ms)。也就是说,理论上终端可以获取到CPU处理每一帧显示画面时产生的目标参数,这样,终端可以每次根据当前这一帧中的计算-访存比例关系调整下一帧中CPU和DDR的工作频率,使得CPU和DDR可以快速、及时的处理帧率抖动的问题,使帧率更加稳定。

[0090] 在本申请的另一些实施例中,终端内还可以预先存储不同应用运行时的计算-访存比例关系。例如,可预先通过测试等手段确定游戏应用A运行时为访存密集型应用,反映其计算-访存比例关系的比例参数 $R = 0.1$,而解压应用B运行时为计算密集型应用,反映其计算-访存比例关系的比例参数 $R = 3$ 。

[0091] 那么,终端可以通过获取正在运行的主线程的ID或正在运行的应用的包名(packname)确定当前运行的具体应用,例如游戏应用A。进而,终端可以在预先存储的不同应用运行时的计算-访存比例关系中查找与游戏应用A对应的比例参数R为0.1,从而获取到当前运行游戏应用A时计算类任务和访存类任务之间的计算-访存比例关系。

[0092] 类似的,终端还可以预先存储终端在不同运行场景(例如支付场景、运动场景等)或提供不同服务(例如位置服务、输入法服务等)时的计算-访存比例,那么,终端可以根据当前实时检测出的运行场景或服务确定出当前计算类任务和访存类任务之间的计算-访存比例关系,本申请实施例对此不做任何限制。

[0093] S202、终端获取与上述计算-访存比例关系对应的第一工作频率表,该第一工作频率表中包括在上述比例关系下DDR和CPU在不同工作频率组合工作时的性能参数。

[0094] 其中,如图4所示,终端内预先存储有与不同计算-访存比例对应的CPU-DDR工作频率表401。其中,每一个CPU-DDR工作频率表401是指终端在运行某一计算-访存比例的程序时,DDR和CPU分别在不同工作频率组合下运行时产生的性能参数(performance)A(也可称为CPU性能参数),该性能参数A可反映出CPU处理一帧显示画面所需的帧处理时长T,例如,该性能参数A可以是单位时长内CPU处理的指令数目。一般,当性能参数A的取值越大时,CPU处理一帧显示画面所需的帧处理时长T越小,即性能参数A与帧处理时长T之间成反比例关系。

[0095] 示例性的,在图4所示的CPU-DDR工作频率表401中,以CPU处理一帧显示画面所需的帧处理时长 T 作为性能参数 A 进行举例说明。可以理解的是,图4中帧处理时长 T 这一选项也可以是与参数帧处理时长 T 相关联的任意参数 $f(T)$ 代替,即性能参数 $A=f(T)$,本申请实施例对此不做任何限制。

[0096] 另外,仍如图4所示,CPU-DDR工作频率表401中还可以包括DDR和CPU分别在不同工作频率组合下运行时DDR和CPU产生的功耗 P 等参数,本申请实施例对此不做任何限制。

[0097] 那么,终端可以根据步骤S201中得到的计算-访存比例关系,查询与该计算-访存比例关系对应的CPU-DDR工作频率表(即第一工作频率表)。例如,步骤S201中得到的比例参数 $R=1.2$,那么,仍如图4所示,终端可以在上述CPU-DDR工作频率表401中查找到与 $R=1.2$ 最接近的计算-访存比例为1,进而,终端可以将计算-访存比例为1时的CPU-DDR工作频率表作为上述第一工作频率表。该第一工作频率表中记录了当计算-访存比例为1时,DDR和CPU分别在不同工作频率组合下运行时产生的功耗 P 以及每一帧的帧处理时长 T 。

[0098] 需要说明的是,在本申请实施例中,与上述计算-访存比例关系对应的第一工作频率表,是指预先存储的多个CPU-DDR工作频率表中计算-访存比例关系(例如上述比例参数 R)与步骤S201中计算的计算-访存比例关系相同或最接近CPU-DDR工作频率表。

[0099] 另外,除了以表格的形式存储上述DDR和CPU分别在不同工作频率组合下运行时产生的性能参数外,终端还可以通过其他数据结构(例如索引)等方式建立不同计算-访存比例关系、DDR和CPU的工作频率组合以及性能参数 A 之间的对应关系,本申请实施例对此不做任何限制。

[0100] S203、终端从上述第一工作频率表中选择满足CPU目标性能参数的DDR工作频点和CPU工作频点。

[0101] 在本申请的一些实施例,以上述CPU目标性能参数 A 为小于或等于16.7ms的目标帧处理时长举例。在步骤S203中,终端可以从图4所示的第一工作频率表中,选择帧处理时长 T 小于或等于16.7ms时对应的DDR和CPU的工作频率组合,该工作频率组合中的DDR工作频点即可设置为下一帧DDR工作时的的工作频点,该工作频率组合中的CPU工作频点即可设置为下一帧CPU工作时的的工作频点。

[0102] 另外,当第一工作频率表中帧处理时长 T 小于或等于16.7ms的工作频率组合有多个时,终端可以选择功耗 P 最低的工作频率组合,这样,终端在该工作频率组合中的DDR工作频点和CPU工作频点下工作时,不仅可以保证帧率稳定在60Hz,还可以最大程度的降低DDR和CPU工作时产生的功耗开销。

[0103] 在本申请的另一些实施例中,以上述CPU目标性能参数 A 为帧处理时长 T 的函数举例,即 $A=f(T)$,其中性能参数 A 与帧处理时长 T 之间成反比例关系。此时,终端还可以检测CPU在实际处理每一帧显示画面时花费的实际帧处理时长,进而根据该实际帧处理时长在上述第一工作频率表中选择满足目标帧处理时长的DDR工作频点和CPU工作频点。

[0104] 示例性的,如图5所示,在终端从CPU的性能监测器中获取上述目标参数的同时,终端还可以获取此时CPU处理一帧显示画面的实际帧处理时长 T' 。以安卓操作系统为例,在显示每一帧显示画面时,终端可启动安卓操作系统内的surfaceflinger服务,surfaceflinger服务可通过调用一系列函数计算出每个图形(Surface)在最终合成图像中的位置,然后交由HwComposer或开放图形库(open graphics library,OpenGL)生成最终的

显示缓存(buffer),最终由显示器显示buffer中的缓存的显示内容。

[0105] 那么,终端可以对surfaceflinger服务中开始处理每一帧显示画面时执行的函数(例如enqueue buffer函数)打桩,并且对surfaceflinger服务中每一帧显示画面处理结束时执行的函数(例如release buffer函数)打桩。这样,两次打桩时的时间差即可作为CPU处理一帧显示画面的实际帧处理时长 T' 。

[0106] 仍如图5所示,除了CPU处理一帧显示画面的实际帧处理时长 T' ,终端还可以获取到当前CPU和DDR的实际工作频点。例如,终端可以检测当前CPU和DDR运行时的实际工作频点,又或者,由于CPU和DDR处理每一帧显示画面的工作频点都是终端通过步骤S201-S203确定的,那么终端可以缓存最近一次得到的CPU和DDR的工作频点,这样,终端缓存的最近一次得到的CPU和DDR的工作频点即为当前CPU和DDR运行时的实际工作频点。

[0107] 那么,终端可以在步骤S202得到的第一工作频率表中确定出与CPU和DDR运行时的实际工作频点对应的性能参数 A' (图5中所示的性能参数 A 为以帧处理时长 T 为自变量的函数,即性能参数 $A=f(T)$)。终端的目标帧处理时长 T'' 不大于16.7ms,而终端当前的实际帧处理时长为 T' ,那么,第一工作频率表中最优的目标性能参数 A'' 应满足以下关系:

$$[0108] \quad A'' * T'' \geq A' * T'$$

$$[0109] \quad \text{即 } A'' \geq A' * (T' / T'')$$

[0110] 这样,终端可以根据目标帧处理时长 T'' (16.7ms)、实际帧处理时长为 T' 以及CPU和DDR实际工作频点对应的性能参数 A' ,在第一工作频率表中确定出满足上述关系的一个或多个目标性能参数 A'' ,以及这一个或多个目标性能参数 A'' 对应的一组或多组DDR和CPU的工作频率组合。

[0111] 另外,当确定出的目标性能参数 A'' 有多个时,与这多个目标性能参数 A'' 对应的DDR和CPU的工作频率组合也有多组。那么,如果上述CPU-DDR工作频率表401中还包括DDR和CPU分别在不同工作频率组合下运行时产生的功耗 P ,则终端可以在与上述多组目标性能参数 A'' 对应的多组DDR和CPU的工作频率组合中,选择功耗 P 最低的工作频率组合作为DDR和CPU处理下一帧显示画面时的工作频点,以降低终端的功耗。

[0112] 又或者,由于终端可以周期性的不断重复步骤S201-S203得到满足CPU目标性能参数的DDR工作频点和CPU工作频点,那么,终端可以根据最近几次终端确定出的DDR工作频点和CPU工作频点,预测下一帧或多帧时DDR和CPU的工作频点,例如,将最近3次确定出的3个DDR工作频点中的最大值以及3个CPU工作频点中的最大值作为后续DDR和CPU的工作频点。

[0113] 至此,通过步骤S203,终端可从上述第一工作频率表中选择满足CPU目标性能参数的DDR工作频点和CPU工作频点,并将该DDR的工作频点和CPU的工作频点作为下一帧DDR和CPU工作时的的工作频点,使得CPU后续工作时处理一帧显示画面的帧处理时长稳定在16.7ms以内,提升显示画面的流畅度。

[0114] 需要说明的是,图5中终端获取实际帧处理时长 T_1 的过程也可以是周期性的,并且终端获取实际帧处理时长 T_1 的周期可以与步骤S201中终端获取性能监测器中目标参数的周期相同或不同,本申请实施例对此不做任何限制。

[0115] S204、终端指示DDR在上述DDR工作频点下工作,并指示CPU在上述CPU工作频点下工作。

[0116] 在步骤S204中,终端可以将步骤S203中确定出的DDR工作频点和CPU工作频点发送

给终端的调频器,由调频器分别调整DDR和CPU的工作频点,使得DDR和CPU均能够在符合当前计算-访存比例关系的工作频点下工作,从而稳定终端的显示帧率,降低终端的功耗和发热。

[0117] 其中,上述步骤S201-S204中终端的动作可以由上述图1中的处理器101根据存储器103中存储的计算机指令或数据来执行,本申请实施例对此不做任何限制。

[0118] 在本申请的另一些实施例中,每一帧显示画面的帧处理时长除了与CPU和DDR的工作频率相关外,还与GPU的工作频率相关。因此,在图4所示的帧率调整流程的基础上,终端还可以根据GPU处理一帧显示画面的实际帧处理时长,设置GPU在满足GPU目标性能参数的GPU工作频点下工作。

[0119] 此时,如图6所示,本申请实施例提供的一种帧率调整方法还包括下述步骤S601-S603:

[0120] S601、终端获取GPU处理一帧显示画面的实际帧处理时长。

[0121] 在图5所示的帧率调整流程的基础上,如图7所示,与终端获取CPU处理一帧显示画面的实际帧处理时长类似的,终端可以对GPU驱动(driver)在处理每一帧显示画面时执行的相关函数(例如fence函数)进行打桩,进而检测出GPU处理一帧显示画面时的实际帧处理时长 G' 。

[0122] S602、终端从第二工作频率表中选择满足GPU目标性能参数的GPU工作频点。

[0123] 与上述CPU-DDR工作频率表401类似的,如图7所示,终端内还可预先存储GPU工作频率表501(即第二工作频率表),用于指示GPU在不同工作频率下运行时产生的性能参数B(也可称为GPU性能参数),性能参数B用于反映GPU处理一帧显示画面所需的帧处理时长G,即性能参数 $B=f(G)$,与上述性能参数A类似的,性能参数B与GPU处理一帧显示画面所需的帧处理时长G呈反比例关系。

[0124] 与上述CPU-DDR工作频率表401类似的,GPU工作频率表501中还可以包括GPU在不同工作频率下运行时产生的功耗P等参数,本申请实施例对此不做任何限制。

[0125] 那么,与上述步骤S203类似的,终端可以从上述GPU工作频率表501中选择满足GPU目标性能参数的GPU工作频点。

[0126] 仍如图7所示,以上述GPU目标性能参数B为帧处理时长G的函数举例,除了GPU处理一帧显示画面的实际帧处理时长 G' ,终端还可以获取到当前GPU的实际工作频点。例如,终端可以检测当前GPU运行时的实际工作频点,又或者,由于GPU处理每一帧显示画面的工作频点都可以通过步骤S601-S602确定,那么终端可以缓存最近一次得到的GPU的工作频点,这样,终端缓存的最近一次得到的GPU的工作频点即为当前GPU运行时的实际工作频点。

[0127] 那么,终端可以在上述GPU工作频率表501中确定出与GPU运行时的实际工作频点对应的性能参数 B' 。由于GPU的目标帧处理时长 G'' 不大于16.7ms,而GPU当前的实际帧处理时长为 G' ,那么,GPU工作频率表501中最优的目标性能参数 B'' 应满足以下关系:

[0128] $B'' * G'' \geq B' * G'$

[0129] 即 $B'' \geq B' * (G' / G'')$

[0130] 这样,终端可以根据与GPU实际工作频点对应性能参数 B' 、目标帧处理时长 G'' (16.7ms)以及实际帧处理时长为 G' ,在GPU工作频率表501中确定出满足目标帧处理时长 G'' 的一个或多个目标性能参数 B'' ,以及这一个或多个目标性能参数 B'' 对应的一個或多个GPU

工作频率。

[0131] 另外,当确定出的目标性能参数B”有多个时,与这多个目标性能参数B”对应的GPU的工作频点也有多个。那么,如果上述GPU工作频率表501中还包括GPU在不同工作频率下运行时产生的功耗P,则终端可以在G在这多个目标性能参数B”对应的多个GPU的工作频率中选择功耗P最低的工作频率作为GPU处理下一帧显示画面时的工作频点,以降低终端的功耗。

[0132] 至此,通过步骤S602,终端可从上述GPU工作频率表501中选择满足GPU目标性能参数的GPU工作频点,预测出下一帧或多帧显示过程中的GPU工作频点,使得GPU后续工作时处理一帧显示画面的帧处理时长稳定在16.7ms以内,提升显示画面的流畅度。

[0133] 需要说明的是,图7中终端获取GPU实际帧处理时长G’的周期可与终端获取CPU实际帧处理时长T’的周期相同,本申请实施例对此不做任何限制。

[0134] S603、终端指示GPU在上述GPU工作频点下工作。

[0135] 与步骤S204类似的,终端还可以将步骤S602中确定出的GPU工作频点发送给终端的调频器,由调频器调整GPU的工作频点。这样,结合步骤S204,终端可以对DDR、CPU和GPU进行联合调频,使得DDR、CPU和GPU均能够在符合当前运行场景和自身处理能力的工作频点下工作,从而稳定终端的显示帧率,降低终端的功耗和发热。

[0136] 其中,上述步骤S601-S603中终端的动作可以由上述图1中的处理器101根据存储器103中存储的计算机指令或数据来执行,本申请实施例对此不做任何限制。

[0137] 另外,由于在本申请提供的帧率调整方法中终端可以实时检测每一帧显示画面的帧处理时长,因此一旦检测到某一帧的帧处理时长不满足目标帧处理时长(例如16.7ms)时,便可快速、及时的调整下一帧DDR、CPU和GPU的工作频率,提高下一帧显示画面的处理速度,从而有效避免因感知帧率抖动滞后而出现帧率出现大幅波动的现象。

[0138] 示例性的,如图8所示,为使用本申请提供的帧率调整方法运行游戏应用A时,与使用现有技术中帧率调整方法运行游戏应用A时的帧率波动图。从图8中可以看出,使用本申请提供的帧率调整方法可以将最低帧率从41Hz提升为52Hz,将小于45Hz的低帧率占比从0.33%降低为0,将显示画面的平均帧率从55.89Hz提升为56Hz,显示画面的流畅度也从1.68降低为1.03(流畅度取值越低画面越流畅),同时,经过测试,终端的功耗也从785毫安降低为754.8毫安。

[0139] 尤其是游戏应用A使用本申请提供的帧率调整方法运行数名玩家进行集体对抗的团战场景时(图8中圈出的位置),相比于现有技术可显著减低显示帧率的抖动现象,使得显示帧率一直稳定在60Hz左右。

[0140] 可以理解的是,上述终端等为了实现上述功能,其包含了执行各个功能相应的硬件结构和/或软件模块。本领域技术人员应该很容易意识到,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,本申请实施例能够以硬件或硬件和计算机软件的结合形式来实现。某个功能究竟以硬件还是计算机软件驱动硬件的方式来执行,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申请实施例的范围。

[0141] 本申请实施例可以根据上述方法示例对上述终端等进行功能模块的划分,例如,可以对应各个功能划分各个功能模块,也可以将两个或两个以上的功能集成在一个处理模

块中。上述集成的模块既可以采用硬件的形式实现,也可以采用软件功能模块的形式实现。需要说明的是,本申请实施例中对模块的划分是示意性的,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。

[0142] 在采用对应各个功能划分各个功能模块的情况下,图9示出了上述实施例中所涉及的终端的一种可能的结构示意图,该终端用于实现以上各个方法实施例中记载的方法,其具体包括:第一获取单元901、第一计算单元902、第一确定单元903、第一指示单元904、第二获取单元905、第二确定单元906以及第二指示单元907。

[0143] 其中,第一获取单元901用于支持终端执行图2中的过程S202;第一计算单元902用于支持终端执行图2中的过程S201;第一确定单元903用于支持终端执行图2中的过程S203;第一指示单元904支持终端执行图2中的过程S204。第二获取单元905用于支持终端执行图6中的过程S601;第二确定单元906用于支持终端执行图6中的过程S602;第二指示单元907支持终端执行图6中的过程S603。其中,上述方法实施例涉及的各步骤的所有相关内容均可以援引到对应功能模块的功能描述,在此不再赘述。

[0144] 在采用集成的单元的情况下,可将上述第一获取单元901、第一计算单元902、第一确定单元903、第一指示单元904、第二获取单元905、第二确定单元906以及第二指示单元907可集成为处理模块。当然,终端还可以包括存储模块、通信模块以输入/输出模块等。此时,如图10所示,示出了上述实施例中所涉及的终端的一种可能的结构示意图,包括处理模块1001、通信模块1002、输入/输出模块1003以及存储模块1004。

[0145] 其中,处理模块1001用于对终端的动作进行控制管理。通信模块1002用于支持终端与其他网络实体的通信。输入/输出模块1003用于接收由用户输入的信息或输出提供给用户的信息以及终端的各种菜单。存储模块1004用于保存终端的程序代码和数据。

[0146] 示例性的,处理模块1001可以是处理器或控制器,例如可以是中央处理器(Central Processing Unit, CPU), GPU, 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), 现场可编程门阵列(Field Programmable Gate Array, FPGA)或者其他可编程逻辑器件、晶体管逻辑器件、硬件部件或者其任意组合。其可以实现或执行结合本申请公开内容所描述的各种示例性的逻辑方框, 模块和电路。所述处理器也可以是实现计算功能的组合, 例如包含一个或多个微处理器组合, DSP和微处理器的组合等等。

[0147] 通信模块1002可以是收发器、收发电路、输入输出设备或通信接口等。例如,通信模块1002具体可以是蓝牙装置、Wi-Fi装置、外设接口等等。

[0148] 存储模块1004可以是存储器,该存储器可以包括高速随机存取存储器(RAM)、DDR,还可以包括非易失存储器,例如磁盘存储器件、闪存器件或其他易失性固态存储器件等。

[0149] 输入/输出模块1003可以为触摸屏、键盘、麦克风以及显示器等输入输出设备。其中,显示器具体可以采用液晶显示器、有机发光二极管等形式来配置显示器。另外,显示器上还可以集成触控板,用于采集在其上或附近的触摸事件,并将采集到的触摸信息发送给其他器件(例如处理器等)。

[0150] 在上述实施例中,可以全部或部分的通过软件,硬件,固件或者其任意组合来实现。当使用软件程序实现时,可以全部或部分地以计算机程序产品的形式出现。所述计算机程序产品包括一个或多个计算机指令。在计算机上加载和执行所述计算机程序指令时,全

部或部分地产生按照本申请实施例所述的流程或功能。所述计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程装置。所述计算机指令可以存储在计算机可读存储介质中,或者从一个计算机可读存储介质向另一个计算机可读存储介质传输,例如,所述计算机指令可以从一个网站站点、计算机、服务器或数据中心通过有线(例如同轴电缆、光纤、数字用户线(DSL))或无线(例如红外、无线、微波等)方式向另一个网站站点、计算机、服务器或数据中心传输。所述计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包含一个或多个可用介质集成的服务器、数据中心等数据存储设备。该可用介质可以是磁性介质,(例如,软盘,硬盘、磁带)、光介质(例如,DVD)或者半导体介质(例如固态硬盘 Solid StateDisk(SSD))等。

[0151] 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,任何在本申请揭露的技术范围内的变化或替换,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所述权利要求的保护范围为准。

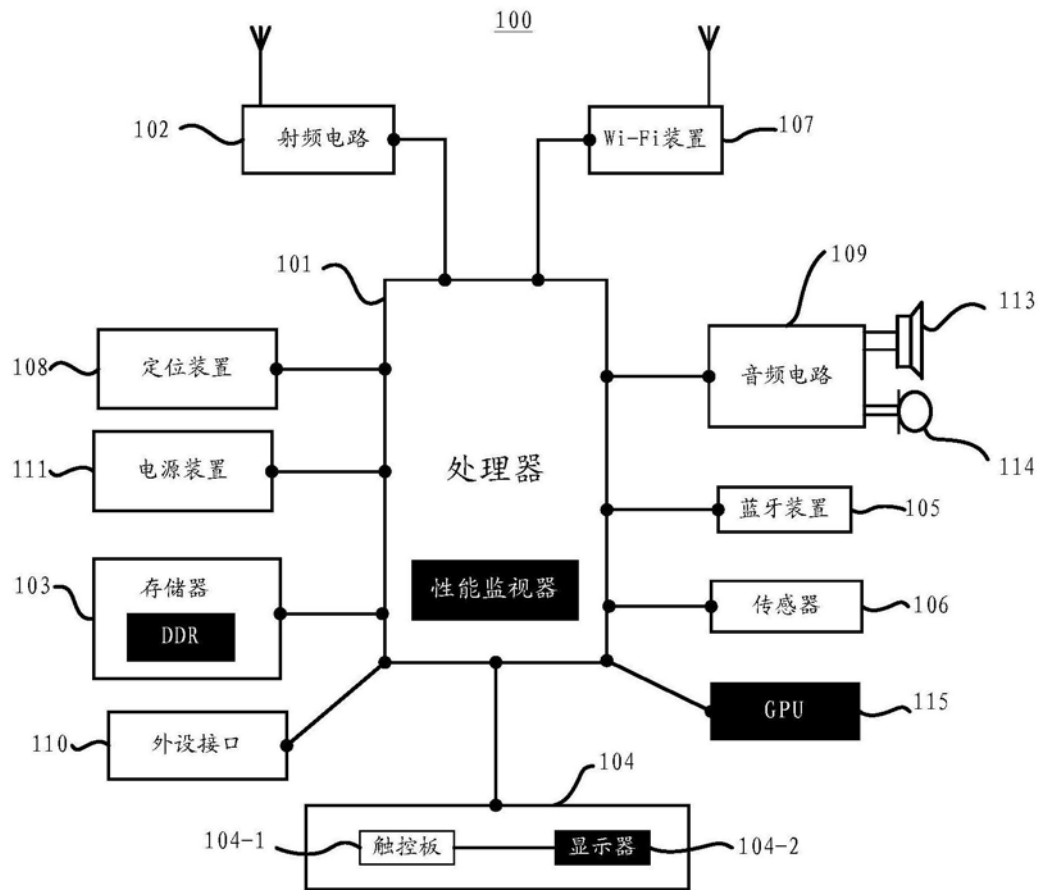


图1

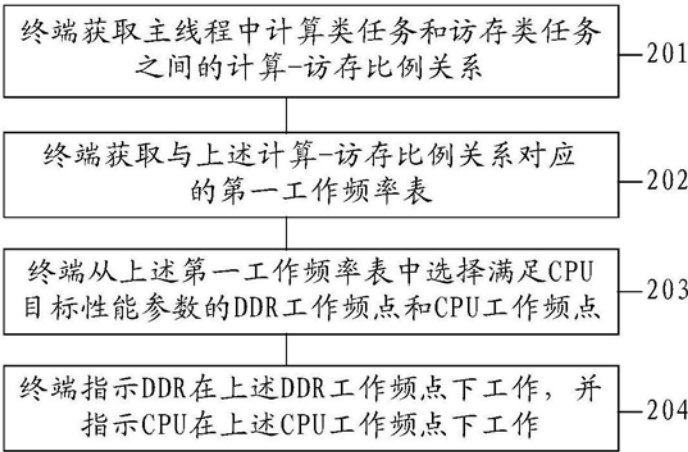


图2

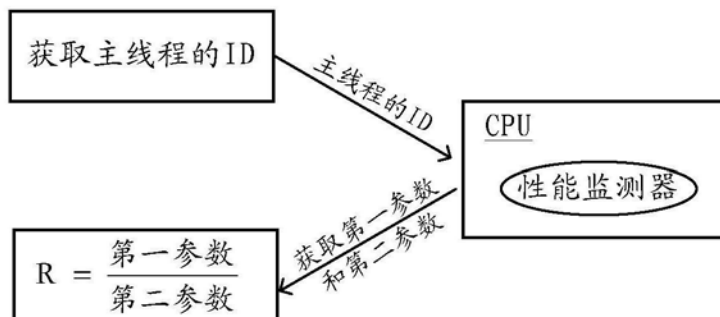


图3

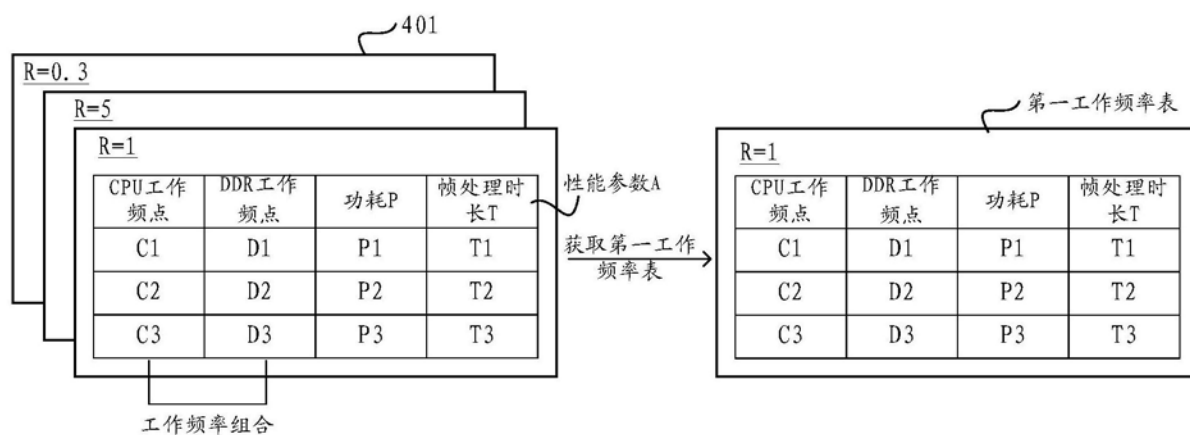


图4

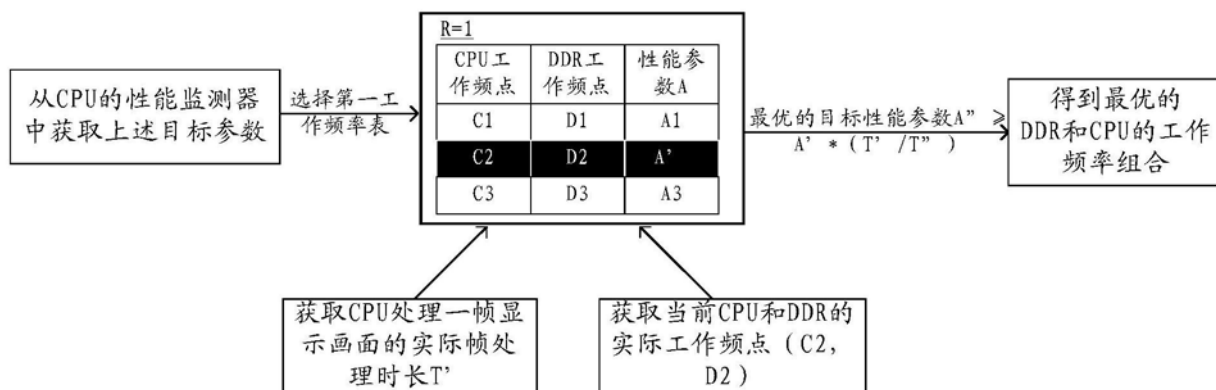


图5

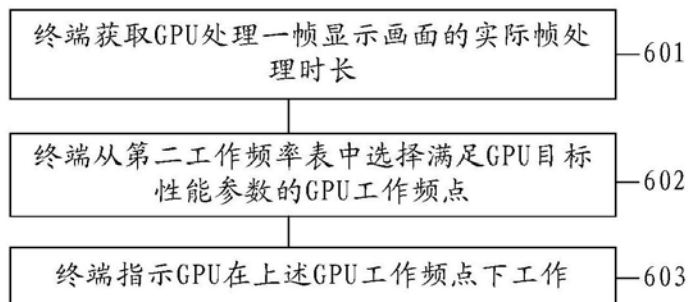


图6

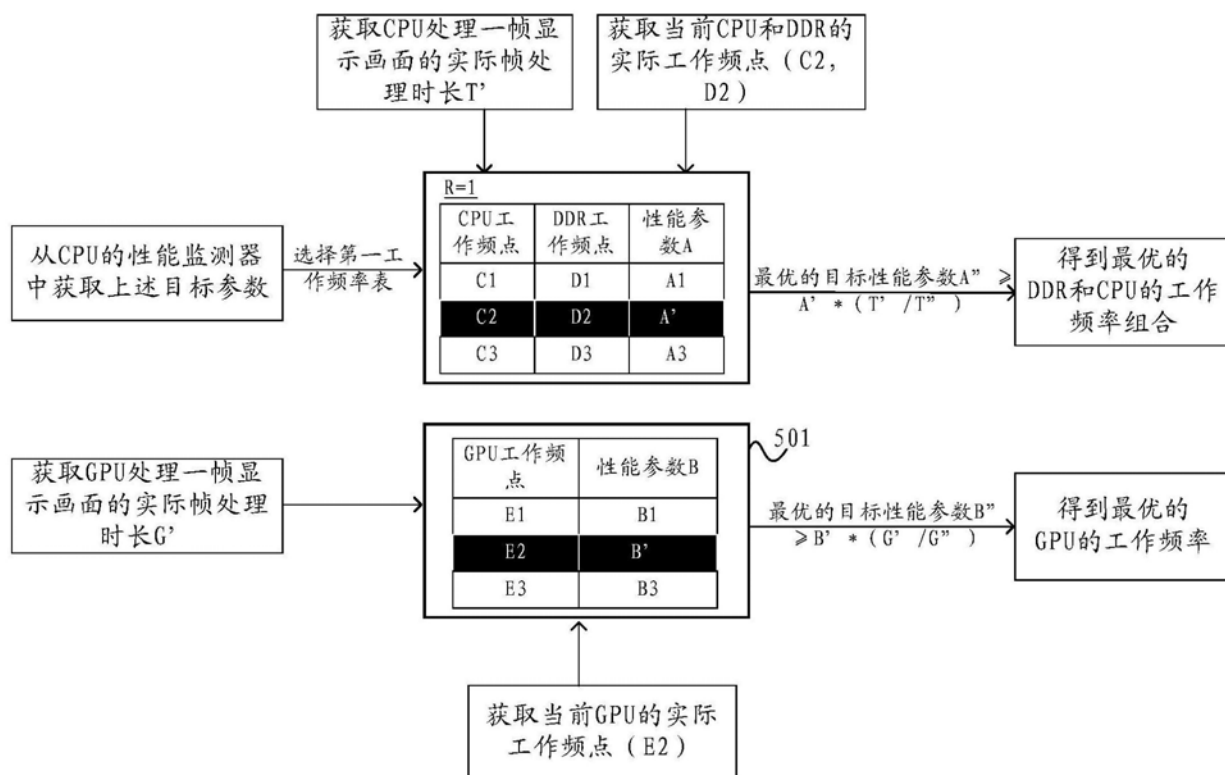


图7

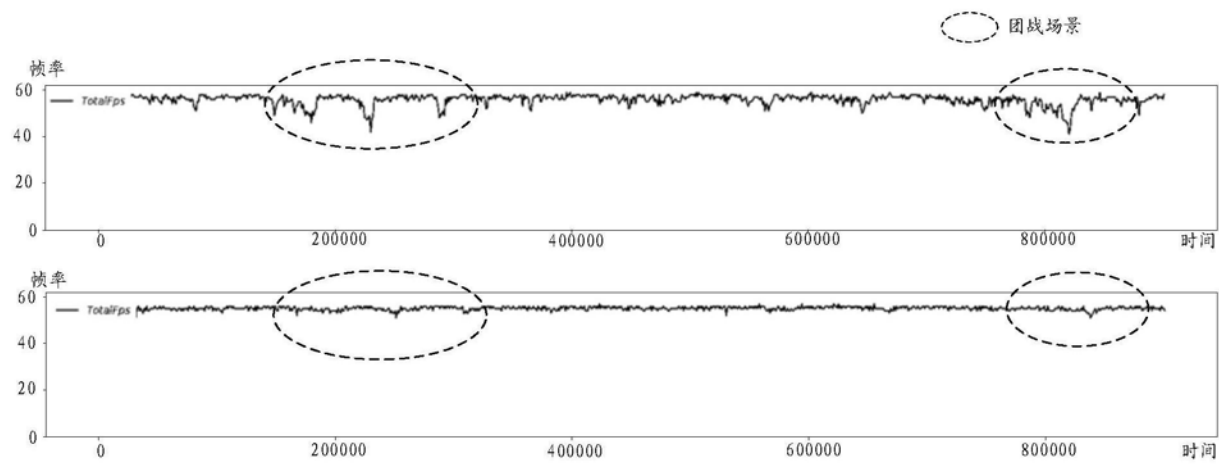


图8

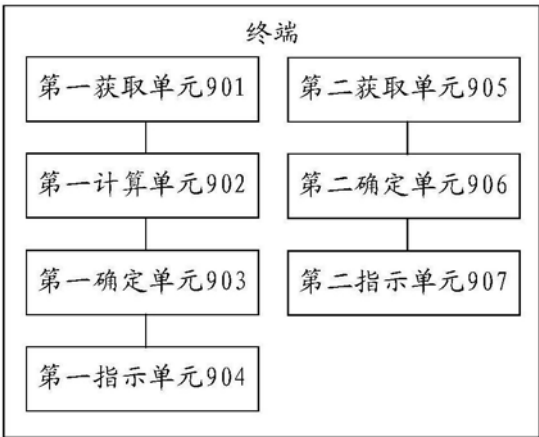


图9

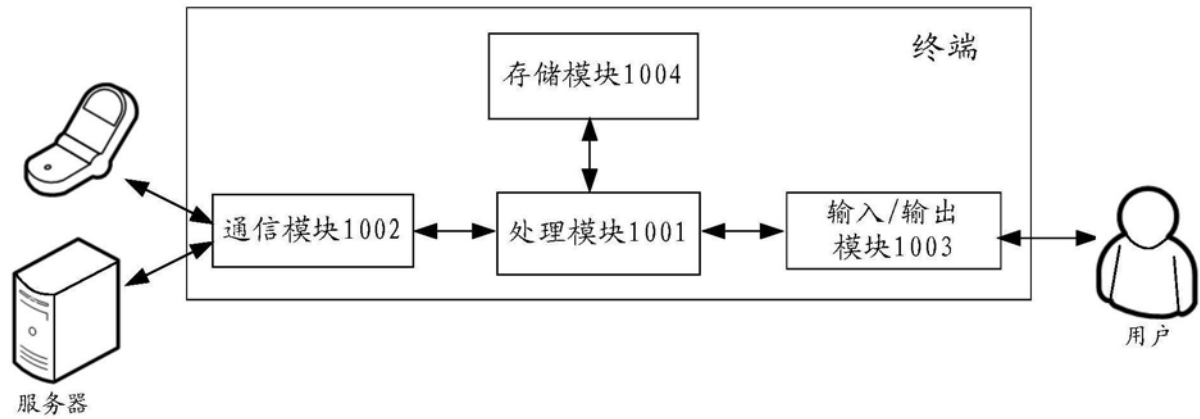


图10