



(12) 发明专利申请

(10) 申请公布号 CN 112685156 A

(43) 申请公布日 2021.04.20

(21) 申请号 202011578565.X

(22) 申请日 2020.12.28

(71) 申请人 北京五八信息技术有限公司
地址 100080 北京市海淀区学清路甲18号
中关村东升科技园学院园三层301室

(72) 发明人 董亮亮

(74) 专利代理机构 北京卫智畅科专利代理事务
所(普通合伙) 11557
代理人 陈佳

(51) Int.Cl.
G06F 9/48 (2006.01)

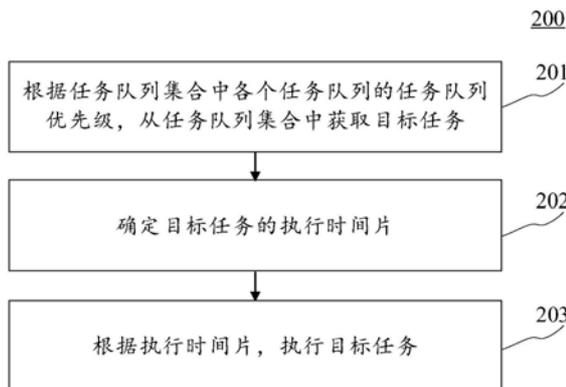
权利要求书1页 说明书9页 附图3页

(54) 发明名称

任务执行方法、装置、电子设备和计算机可读介质

(57) 摘要

本公开的实施例公开了任务执行方法、装置、电子设备和计算机可读介质。该方法的一具体实施方式包括：根据任务队列集合中各个任务队列的任务队列优先级，从任务队列集合中获取目标任务；确定上述目标任务的执行时间片；根据上述执行时间片，执行上述目标任务。该实施方式实现了任务执行所需的时间片动态、灵活的生成，提高了任务执行效率，降低了内存资源的消耗。



1. 一种任务执行方法,包括:
根据任务队列集合中各个任务队列的任务队列优先级,从所述任务队列集合中获取目标任务;
确定所述目标任务的执行时间片;
根据所述执行时间片,执行所述目标任务。
2. 根据权利要求1所述的方法,其中,所述确定所述目标任务的执行时间片,包括:
响应于确定不存在与所述目标任务对应的基础时间片,确定是否存在所述目标任务的历史执行记录;
响应于确定存在所述历史执行记录,根据所述历史执行记录,确定所述基础时间片。
3. 根据权利要求2所述的方法,其中,所述确定所述目标任务的执行时间片,还包括:
根据所述基础时间片,确定所述执行时间片。
4. 根据权利要求3所述的方法,其中,所述根据所述基础时间片,确定所述执行时间片,包括:
对所述基础时间片进行调整,得到调整后的基础时间片;
根据所述调整后的基础时间片和所述目标任务对应的任务队列的任务队列优先级,确定所述执行时间片。
5. 根据权利要求2所述的方法,其中,所述确定所述目标任务的执行时间片,还包括:
响应于确定不存在所述历史执行记录,将预设基础时间片确定为所述基础时间片。
6. 根据权利要求1所述的方法,其中,所述从所述任务队列集合中获取目标任务,包括:
从所述任务队列集合中选择满足预设条件的任务队列作为第一目标任务队列;
从所述第一目标任务队列中获取目标任务。
7. 根据权利要求1所述的方法,其中,所述任务队列中的任务具有任务优先级;所述任务是根据所述任务优先级加入所述任务队列集合中对应的任务队列的。
8. 根据权利要求1所述的方法,其中,所述方法还包括:
响应于确定所述目标任务在所述执行时间片内未执行完成,终止所述目标任务的执行,以及将所述目标任务加入第二目标任务队列。
9. 一种任务执行装置,包括:
获取单元,被配置成根据任务队列集合中各个任务队列的任务队列优先级,从所述任务队列集合中获取目标任务;
确定单元,被配置成确定所述目标任务的执行时间片;
执行单元,被配置成根据所述执行时间片,执行所述目标任务。
10. 一种电子设备,包括:
一个或多个处理器;
存储装置,其上存储有一个或多个程序;
当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如权利要求1-8中任一所述的方法。
11. 一种计算机可读介质,其上存储有计算机程序,其中,所述程序被处理器执行时实现如权利要求1-8中任一所述的方法。

任务执行方法、装置、电子设备和计算机可读介质

技术领域

[0001] 本公开的实施例涉及计算机技术领域，具体涉及任务执行方法、装置、电子设备和计算机可读介质。

背景技术

[0002] 任务执行，是一种将任务从任务队列中取出并执行该任务的方法。目前，在执行任务时，通常采用的方式为：根据预设的时间片执行任务。例如，预先为任务队列设置对应的时间片，当该任务队列中的任务需要执行时，在预设的时间片内执行该任务队列中的任务。

[0003] 然而，当采用上述方式执行任务时，经常会存在如下技术问题：

[0004] 需要预先人为的为任务队列设置固定的时间片，难以动态的、灵活的确定任务执行所需的时间片。从而，可能导致所需执行时间较长的任务在预设的时间片内未执行完毕而被强制中断。进而，导致不同任务之间较为频繁的切换，降低任务执行效率，增加内存资源的消耗。

发明内容

[0005] 本公开的内容部分用于以简要的形式介绍构思，这些构思将在后面的具体实施方式部分被详细描述。本公开的内容部分并不旨在标识要求保护的技术方案的关键特征或必要特征，也不旨在用于限制所要求的保护的技术方案的范围。

[0006] 本公开的一些实施例提出了任务执行方法、装置、电子设备和计算机可读介质，来解决以上背景技术部分提到的技术问题中的一项或多项。

[0007] 第一方面，本公开的一些实施例提供了一种任务执行方法，该方法包括：根据任务队列集合中各个任务队列的任务队列优先级，从上述任务队列集合中获取目标任务；确定上述目标任务的执行时间片；根据上述执行时间片，执行上述目标任务。

[0008] 第二方面，本公开的一些实施例提供了一种任务执行装置，装置包括：获取单元，被配置成根据任务队列集合中各个任务队列的任务队列优先级，从上述任务队列集合中获取目标任务；确定单元，被配置成确定上述目标任务的执行时间片；执行单元，被配置成根据上述执行时间片，执行上述目标任务。

[0009] 第三方面，本公开的一些实施例提供了一种电子设备，包括：一个或多个处理器；存储装置，其上存储有一个或多个程序，当一个或多个程序被一个或多个处理器执行，使得一个或多个处理器实现上述第一方面任一实现方式所描述的方法。

[0010] 第四方面，本公开的一些实施例提供了一种计算机可读介质，其上存储有计算机程序，其中，程序被处理器执行时实现上述第一方面任一实现方式所描述的方法。

[0011] 本公开的上述各个实施例具有如下有益效果：首先，根据任务队列集合中各个任务队列的任务队列优先级，从上述任务队列集合中获取目标任务。从而，在获取目标任务的过程中考虑到了任务队列优先级的不同，可以优先处理所在的任务队列优先级较高的任务，确保高优先级的任务优先执行。然后，确定上述目标任务的执行时间片。从而，在目标任

务被取出执行时,实时确定该目标任务所需的执行时间片,达到动态、灵活的确定任务执行所需的时间片的效果。最后,根据上述执行时间片,执行上述目标任务。从而,可以在一定程度上缓解根据预设的固定时间片执行任务时,所需执行时间较长的目标任务在预设的时间片内无法执行完毕而被强制中断所导致的任务的频繁切换的情况。由此,本公开可以在一定程度上避免任务的频繁切换、提高任务执行效率以及降低内存资源的消耗。

附图说明

[0012] 结合附图并参考以下具体实施方式,本公开各实施例的上述和其他特征、优点及方面将变得更加明显。贯穿附图中,相同或相似的附图标记表示相同或相似的元素。应当理解附图是示意性的,元件和元素不一定按照比例绘制。

[0013] 图1是根据本公开的一些实施例的任务执行方法的一个应用场景示意图;

[0014] 图2是根据本公开的任务执行方法的一些实施例的流程图;

[0015] 图3是根据本公开的任务执行方法的另一些实施例的流程图;

[0016] 图4是根据本公开的任务执行方法的另一些实施例中的历史执行记录的存储形式的示意图;

[0017] 图5是根据本公开的任务执行装置的一些实施例的结构示意图;

[0018] 图6是适于用来实现本公开的一些实施例的电子设备的结构示意图。

具体实施方式

[0019] 下面将参照附图更详细地描述本公开的实施例。虽然附图中显示了本公开的某些实施例,然而应当理解的是,本公开可以通过各种形式来实现,而且不应该被解释为限于这里阐述的实施例。相反,提供这些实施例是为了更加透彻和完整地理解本公开。应当理解的是,本公开的附图及实施例仅用于示例性作用,并非用于限制本公开的保护范围。

[0020] 另外还需要说明的是,为了便于描述,附图中仅示出了与有关发明相关的部分。在不冲突的情况下,本公开中的实施例及实施例中的特征可以相互组合。

[0021] 需要注意,本公开中提及的“第一”、“第二”等概念仅用于对不同的装置、模块或单元进行区分,并非用于限定这些装置、模块或单元所执行的功能的顺序或者相互依存关系。

[0022] 需要注意,本公开中提及的“一个”、“多个”的修饰是示意性而非限制性的,本领域技术人员应当理解,除非在上下文另有明确指出,否则应该理解为“一个或多个”。

[0023] 本公开实施方式中的多个装置之间所交互的消息或者信息的名称仅用于说明性的目的,而并不是用于对这些消息或信息的范围进行限制。

[0024] 下面将参考附图并结合实施例来详细说明本公开。

[0025] 图1是本公开的一些实施例的任务执行方法的一个应用场景的示意图。

[0026] 在图1的应用场景中,首先,计算设备101可以根据任务队列集合102中各个任务队列的任务队列优先级,从上述任务队列集合102中获取目标任务103。然后,计算设备101可以确定上述目标任务103的执行时间片104。最后,计算设备101可以根据上述执行时间片104,执行上述目标任务103。

[0027] 需要说明的是,上述计算设备101可以是硬件,也可以是软件。当计算设备为硬件时,可以实现成多个服务器或终端设备组成的分布式集群,也可以实现成单个服务器或单

个终端设备。当计算设备体现为软件时,可以安装在上述所列举的硬件设备中。其可以实现成例如用来提供分布式服务的多个软件或软件模块,也可以实现成单个软件或软件模块。在此不做具体限定。

[0028] 应该理解,图1中的计算设备的数目仅仅是示意性的。根据实现需要,可以具有任意数目的计算设备。

[0029] 具体来说,造成任务频繁切换,任务执行效率降低,内存资源消耗增加的原因在于:根据预先人为设置的固定时间片执行任务,难以动态的、灵活的对时间片进行调整。基于此,可以通过本公开的任务执行方法的一些实施例的流程200或300中的步骤解决上述问题。

[0030] 继续参考图2,示出了根据本公开的任务执行方法的一些实施例的流程200。该任务执行方法,包括以下步骤:

[0031] 步骤201,根据任务队列集合中各个任务队列的任务队列优先级,从任务队列集合中获取目标任务。

[0032] 在一些实施例中,任务执行方法的执行主体(如图1所示的计算设备101)可以根据任务队列集合中各个任务队列的任务队列优先级,从上述任务队列集合中获取目标任务。其中,上述任务队列集合中的各个任务队列具有不同的任务队列优先级。可以用数字表示上述任务队列的任务队列优先级。可以预先约定数字越大,表示的任务队列的优先级就越高。上述任务队列的任务队列优先级可以是预先设置的。上述任务队列中的任务可以是计算机进程。上述目标任务可以是任务队列优先级与任务队列中包括的任务数目的乘积值最大的任务队列中队头的任务。

[0033] 作为示例,上述任务队列集合中可以包括三个任务队列。这三个任务队列的任务队列优先级可以分别是1,2和3。则任务队列优先级为3的任务队列的优先级最高。任务队列优先级为1的任务队列的优先级最低。

[0034] 在一些实施例的一些可选的实现方式中,上述执行主体从上述任务队列集合中获取目标任务,可以包括以下步骤:

[0035] 第一步,从上述任务队列集合中选择满足预设条件的任务队列作为第一目标任务队列。其中,上述预设条件可以是:任务队列是上述任务队列集合中所有非空任务队列中任务队列优先级最高的任务队列。

[0036] 第二步,从上述第一目标任务队列中获取目标任务。其中,可以将上述第一目标任务队列中队头的任务作为目标任务。

[0037] 在上述实现方式中,在获取目标任务的过程中考虑到了任务队列优先级的不同,可以优先处理所在的任务队列优先级较高的任务,确保高优先级的任务优先执行。

[0038] 在一些实施例的一些可选的实现方式中,上述任务队列中的任务具可以有任务优先级。上述任务可以是根据上述任务优先级加入上述任务队列集合中对应的任务队列的。其中,任务的优先级可以是人为预先设定的。任务优先级的表示方法和任务队列优先级相同。则可以根据上述任务的优先级,将上述任务加入上述任务队列集合中任务队列优先级与上述任务的优先级相同的任务队列中。

[0039] 步骤202,确定目标任务的执行时间片。

[0040] 在一些实施例中,上述执行主体可以将上述第一目标任务队列中包括的任务的数

目与平均时间片的乘积值确定为上述目标任务的执行时间片。其中,上述平均时间片可以是上述任务队列集合中所存储过的各个历史任务的平均执行时间。

[0041] 在上述方式中,根据第一目标任务队列中包括的任务的数目动态的确定目标任务的执行时间片,可以在任务较多时,尽量保证任务在一个完整的时间片中能够执行完毕。可以在一定程度上避免所需执行时间较长的任务被频繁终止。减少过于频繁的任务切换对于内存等资源的消耗。

[0042] 步骤203,根据执行时间片,执行目标任务。

[0043] 在一些实施例中,上述执行主体可以根据上述执行时间片为上述目标任务分配CPU(Central Processing Unit,中央处理器)占用时长,以供上述目标任务在上述CPU占用时长内执行。其中,上述CPU占用时长与上述执行时间片所表示的时长等长。

[0044] 作为示例,上述执行时间片可以是100秒。则可以为上述目标任务分配100秒的CPU占用时长,以供上述目标任务在100秒的CPU占用时长内执行。

[0045] 在上述实现方式中,根据实时动态确定的执行时间片,执行上述目标任务。可以在一定程度上缓解根据预设的固定时间片执行任务时,所需执行时间较长的目标任务在预设的时间片内无法执行完毕而被强制中断所导致的任务的频繁切换。

[0046] 本公开的上述各个实施例具有如下有益效果:首先,根据任务队列集合中各个任务队列的任务队列优先级,从上述任务队列集合中获取目标任务。从而,在获取目标任务的过程中考虑到了任务队列优先级的不同,可以优先处理所在的任务队列优先级较高的任务,确保高优先级的任务优先执行。然后,确定上述目标任务的执行时间片。从而,在目标任务被取出执行时,实时确定该目标任务所需的执行时间片,达到动态、灵活的确定任务执行所需的时间片的效果。最后,根据上述执行时间片,执行上述目标任务。从而,可以在一定程度上缓解根据预设的固定时间片执行任务时,所需执行时间较长的目标任务在预设的时间片内无法执行完毕而被强制中断所导致的任务的频繁切换的情况。由此,本公开可以在一定程度上避免任务的频繁切换、提高任务执行效率以及降低内存资源的消耗。

[0047] 进一步参考图3,其示出了任务执行方法的另一些实施例的流程300。该任务执行方法的流程300,包括以下步骤:

[0048] 步骤301,根据任务队列集合中各个任务队列的任务队列优先级,从任务队列集合中获取目标任务。

[0049] 在一些实施例中,步骤301的具体实现方式及所带来的技术效果可以参考图2对应的那些实施例中的步骤201,在此不再赘述。

[0050] 步骤302,响应于确定不存在与目标任务对应的基础时间片,确定是否存在目标任务的历史执行记录。

[0051] 在一些实施例中,上述执行主体可以响应于确定不存在与上述目标任务对应的基础时间片,确定是否存在上述目标任务的历史执行记录。其中,上述基础时间片可以是在上述目标任务加入上述任务队列集合之后,第一次被取出执行时所确定的。上述基础时间片在确定后可以被记录在缓存中,直到上述目标任务执行完毕后从缓存中清除。上述历史执行记录可以包括上述目标任务的各个历史执行时长。上述历史执行时长可以是上述目标任务完整的执行一次所占用的时间片之和。可以在数据库中查找目标任务的历史执行记录。这种查询方式,需要在数据库中预先存储目标任务的历史执行时长。参考图4,上述历史

执行记录可以以表格的形式存储在数据库中。

[0052] 步骤303,响应于确定存在历史执行记录,根据历史执行记录,确定基础时间片。

[0053] 在一些实施例中,上述执行主体可以响应于确定存在上述历史执行记录,根据上述历史执行记录,确定上述基础时间片。

[0054] 作为示例,可以将上述历史执行记录中包括的各个历史执行时长的算数平均值确定为上述基础时间片。

[0055] 作为又一示例,对上述历史执行记录中包括的各个历史执行时长进行线性拟合处理,得到上述基础时间片。可以将上述各个历史执行时长输入时间序列预测模型中进行线性拟合处理,以得到上述基础时间片。上述时间序列预测模型可以是线性趋势模型、对数线性趋势模型或者fbprophet (Facebook Prophet) 等等。这种线性拟合方式,需要预先剔除历史执行记录中异常的历史执行时长。上述异常的历史执行时长可以指由任务执行异常所导致的过长或过短的执行时长。可以通过异常值分析方法确定上述历史执行记录中异常的历史执行时长。上述异常值分析方法可以是Interquartile Range (四分位距法)、z-score (z分数法) 和Isolation Forest (孤立森林法) 等等。

[0056] 在一些实施例的一些可选的实现方式中,上述执行主体可以响应于确定不存在上述历史执行记录,将预设基础时间片确定为上述基础时间片。其中,上述预设基础时间片可以是预先人为设置的。若在数据库中查找不到上述目标任务的历史执行记录,则将预设基础时间片确定为上述基础时间片。

[0057] 步骤304,根据基础时间片,确定执行时间片。

[0058] 在一些实施例中,上述执行主体可以将上述第一目标任务队列中包括的任务的数目与上述基础时间片的乘积值确定为上述执行时间片。

[0059] 在一些实施例的一些可选的实现方式中,上述执行主体根据上述基础时间片,确定上述执行时间片,可以包括以下步骤:

[0060] 第一步,通过以下公式,对上述基础时间片进行调整,得到调整后的基础时间片:

$$[0061] \quad T_1 = T_0 + T_0 \times \left(\frac{R}{\frac{M}{m}} \right)$$

[0062] 其中, T_1 表示上述调整后的基础时间片。 T_0 表示上述基础时间片。 M 表示上述执行主体的总内存。 m 表示上述执行主体为上述目标任务分配的基础内存。 R 表示上述执行主体中当前正在执行的任务的总数。

[0063] 第二步,根据上述调整后的基础时间片和上述目标任务对应的任务队列的任务队列优先级,确定上述执行时间片。其中,上述与上述目标任务对应的任务队列可以是上述目标任务被取出执行之前所在的任务队列。

[0064] 作为示例,可以将上述调整后的基础时间片和上述目标任务对应的任务队列的任务队列优先级的乘积值确定为上述执行时间片。

[0065] 上述步骤302-304作为本公开的实施例的一个发明点,进一步解决了背景技术提及的技术问题“频繁的任务切换,降低任务执行效率,增加内存资源的消耗”。导致较为频繁的任务切换,降低任务执行效率,增加内存资源消耗的因素往往如下:现有的任务执行方法

需要在预先人为预设的时间片内执行任务队列中的任务,而这种方法难以动态的、灵活的确定任务执行所需的时间片。如果解决了上述因素,就能在一定程度上达到避免任务的频繁切换,提高任务执行效率,降低内存资源消耗的效果。为了达到这一效果,本公开首先引入了目标任务的历史执行记录,并根据历史执行记录确定目标任务的基础时间片。历史执行记录为基础时间片的确定提供了历史参考依据。从而在确定最终的执行时间片时考虑到历史执行时长的参考价值。使得确定的执行时间片更符合任务执行的实际规律。接着,根据当前执行主体中的任务执行情况,对基础时间片进行调整。当执行主体中正在执行的任务的总数较多、内存资源占用量较高时,执行主体中的负载和压力较大,任务执行速度较慢。此时,适当的增加基础时间片,可保证任务的顺利执行,避免频繁的切换任务对内存资源的过多消耗。最后,再根据任务队列优先级对调整后的基础时间片进行进一步的调整,进一步增加优先级较高的任务队列中任务的执行时间片,提高其执行效率。从而,本公开进一步提高了任务执行效率,降低了内存资源的消耗。

[0066] 步骤305,根据执行时间片,执行目标任务。

[0067] 在一些实施例中,步骤305的具体实现方式及所带来的技术效果可以参考图2对应的那些实施例中的步骤203,在此不再赘述。

[0068] 步骤306,响应于确定目标任务在执行时间片内未执行完成,终止目标任务的执行,以及将目标任务加入第二目标任务队列。

[0069] 在一些实施例中,上述执行主体可以响应于确定上述目标任务在上述执行时间片内未执行完成,终止上述目标任务的执行,以及将上述目标任务加入第二目标任务队列。当上述任务队列集合中存在任务队列优先级比上述第一目标任务队列的任务队列优先级低的任务队列时,将上述任务队列集合中任务队列优先级仅次于上述第一目标任务队列的任务队列优先级的任务队列作为第二目标任务队列。当上述第一目标任务队列的任务队列优先级是上述任务队列集合中所有任务队列中任务队列优先级最低的任务队列时,将上述第一目标任务队列作为第二目标任务队列。

[0070] 从图3中可以看出,与图2对应的一些实施例的描述相比,图4对应的一些实施例中的任务执行方法的流程300体现了对生成执行时间片进行扩展的步骤。由此,这些实施例描述的方案引入了目标任务的历史执行记录,并根据历史执行记录确定目标任务的基础时间片。历史执行记录为基础时间片的确定提供了历史参考依据。从而在确定最终的执行时间片时考虑到历史执行时长的参考价值。使得确定的执行时间片更符合任务执行的实际规律。接着,根据当前执行主体中的任务执行情况,对基础时间片进行调整。当执行主体中正在执行的任务的总数较多、内存资源占用量较高时,执行主体中的负载和压力较大,任务执行速度较慢。此时,适当的增加基础时间片,可保证任务的顺利执行,避免频繁的切换任务对内存资源的过多消耗。最后,再根据任务队列优先级对调整后的基础时间片进行进一步的调整,进一步增加优先级较高的任务队列中任务的执行时间片,提高其执行效率。从而,本公开进一步提高了任务执行效率,降低了内存资源的消耗。

[0071] 进一步参考图5,作为对上述各图所示方法的实现,本公开提供了一种任务执行装置的一些实施例,这些装置实施例与图2所示的那些方法实施例相对应,该装置具体可以应用于各种电子设备中。

[0072] 如图5所示,一些实施例的任务执行装置500包括:获取单元501、确定单元502和执

行单元503。其中,获取单元501,被配置成根据任务队列集合中各个任务队列的任务队列优先级,从上述任务队列集合中获取目标任务。确定单元502,被配置成确定上述目标任务的执行时间片。执行单元503,被配置成根据上述执行时间片,执行上述目标任务。

[0073] 在一些实施例的可选实现方式中,装置500的确定单元502可以被进一步配置成:响应于确定不存在与上述目标任务对应的基础时间片,确定是否存在上述目标任务的历史执行记录;响应于确定存在上述历史执行记录,根据上述历史执行记录,确定上述基础时间片。

[0074] 在一些实施例的可选实现方式中,装置500的确定单元502还可以被进一步配置成:根据上述基础时间片,确定上述执行时间片。

[0075] 在一些实施例的可选实现方式中,装置500的确定单元502又可以被进一步配置成:对上述基础时间片进行调整,得到调整后的基础时间片;根据上述调整后的基础时间片和上述目标任务对应的任务队列的任务队列优先级,确定上述执行时间片。

[0076] 在一些实施例的可选实现方式中,装置500的确定单元502又可以被更进一步配置成:响应于确定不存在上述历史执行记录,将预设基础时间片确定为上述基础时间片。

[0077] 在一些实施例的可选实现方式中,装置500的获取单元501可以被进一步配置成:从上述任务队列集合中选择满足预设条件的任务队列作为第一目标任务队列。从上述第一目标任务队列中获取目标任务。

[0078] 在一些实施例的可选实现方式中,上述任务队列中的任务具有任务优先级。上述任务是根据上述任务优先级加入上述任务队列集合中对应的任务队列的。

[0079] 在一些实施例的可选实现方式中,装置500还可以包括:响应于确定上述目标任务在上述执行时间片内未执行完成,终止上述目标任务的执行,以及将上述目标任务加入第二目标任务队列。

[0080] 可以理解的是,该装置500中记载的诸单元与参考图2描述的方法中的各个步骤相对应。由此,上文针对方法描述的操作、特征以及产生的有益效果同样适用于装置500及其中包含的单元,在此不再赘述。

[0081] 下面参考图6,其示出了适于用来实现本公开的一些实施例的电子设备600的结构示意图。图6示出的电子设备仅仅是一个示例,不应对本公开的实施例的功能和使用范围带来任何限制。

[0082] 如图6所示,电子设备600可以包括处理装置(例如中央处理器、图形处理器等)601,其可以根据存储在只读存储器(ROM)602中的程序或者从存储装置608加载到随机访问存储器(RAM)603中的程序而执行各种适当的动作和处理。在RAM 603中,还存储有电子设备600操作所需的各种程序和数据。处理装置601、ROM 602以及RAM603通过总线604彼此相连。输入/输出(I/O)接口605也连接至总线604。

[0083] 通常,以下装置可以连接至I/O接口605:包括例如触摸屏、触摸板、键盘、鼠标、摄像头、麦克风、加速度计、陀螺仪等的输入装置606;包括例如液晶显示器(LCD)、扬声器、振动器等的输出装置607;以及通信装置609。通信装置609可以允许电子设备600与其他设备进行无线或有线通信以交换数据。虽然图6示出了具有各种装置的电子设备600,但是应理解的是,并不要求实施或具备所有示出的装置。可以替代地实施或具备更多或更少的装置。图6中示出的每个方框可以代表一个装置,也可以根据需要代表多个装置。

[0084] 特别地,根据本公开的一些实施例,上文参考流程图描述的过程可以被实现为计算机软件程序。例如,本公开的一些实施例包括一种计算机程序产品,其包括承载在计算机可读介质上的计算机程序,该计算机程序包含用于执行流程图所示的方法的程序代码。在这样的一些实施例中,该计算机程序可以通过通信装置609从网络上被下载和安装,或者从存储装置608被安装,或者从ROM 602被安装。在该计算机程序被处理装置601执行时,执行本公开的一些实施例的方法中限定的上述功能。

[0085] 需要说明的是,本公开的一些实施例中记载的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPR0M或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本公开的一些实施例中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本公开的一些实施例中,计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读信号介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于:电线、光缆、RF(射频)等等,或者上述的任意合适的组合。

[0086] 在一些实施方式中,客户端、服务器可以利用诸如HTTP(HyperText Transfer Protocol,超文本传输协议)之类的任何当前已知或未来研发的网络协议进行通信,并且可以与任意形式或介质的数字数据通信(例如,通信网络)互连。通信网络的示例包括局域网(“LAN”),广域网(“WAN”),网际网(例如,互联网)以及端对端网络(例如,ad hoc端对端网络),以及任何当前已知或未来研发的网络。

[0087] 上述计算机可读介质可以是上述电子设备中所包含的;也可以是单独存在,而未装配入该电子设备中。上述计算机可读介质承载有一个或者多个程序,当上述一个或者多个程序被该电子设备执行时,使得该电子设备:根据任务队列集合中各个任务队列的任务队列优先级,从上述任务队列集合中获取目标任务;确定上述目标任务的执行时间片;根据上述执行时间片,执行上述目标任务。

[0088] 可以以一种或多种程序设计语言或其组合来编写用于执行本公开的一些实施例的操作的计算机程序代码,上述程序设计语言包括面向对象的程序设计语言——诸如Java、Smalltalk、C++,还包括常规的过程式程序设计语言——诸如“C”语言或类似的设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络——包括局域网(LAN)或广域网(WAN)——连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0089] 附图中的流程图和框图,图示了按照本公开各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,该模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0090] 描述于本公开的一些实施例中的单元可以通过软件的方式实现,也可以通过硬件的方式来实现。所描述的单元也可以设置在处理器中,例如,可以描述为:一种处理器包括获取单元、确定单元和执行单元。其中,这些单元的名称在某种情况下并不构成对该单元本身的限定,例如,获取单元还可以被描述为“根据任务队列集合中各个任务队列的任务队列优先级,从上述任务队列集合中获取目标任务的单元”。

[0091] 本文中以上描述的功能可以至少部分地由一个或多个硬件逻辑部件来执行。例如,非限制性地,可以使用的示范类型的硬件逻辑部件包括:现场可编程门阵列(FPGA)、专用集成电路(ASIC)、专用标准产品(ASSP)、片上系统(SOC)、复杂可编程逻辑设备(CPLD)等等。

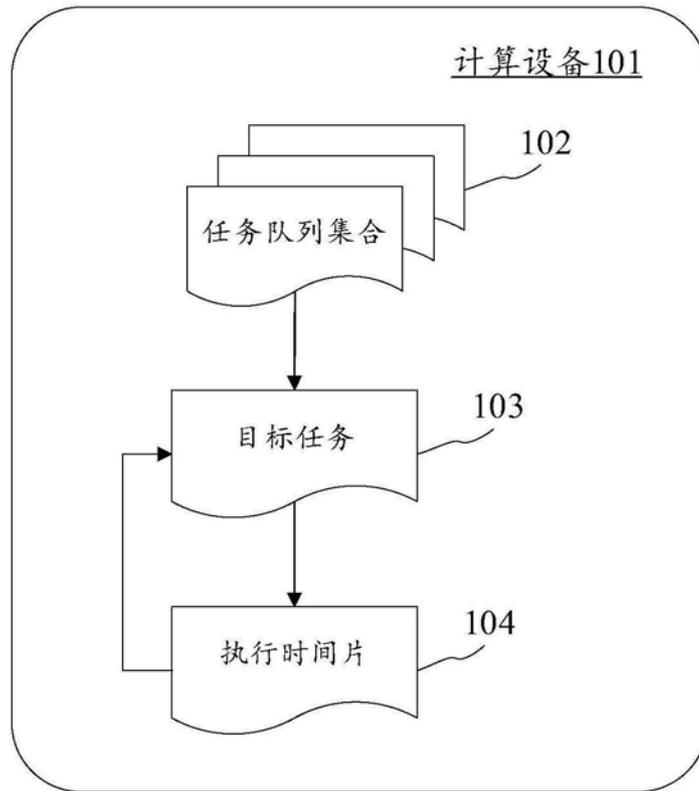


图1

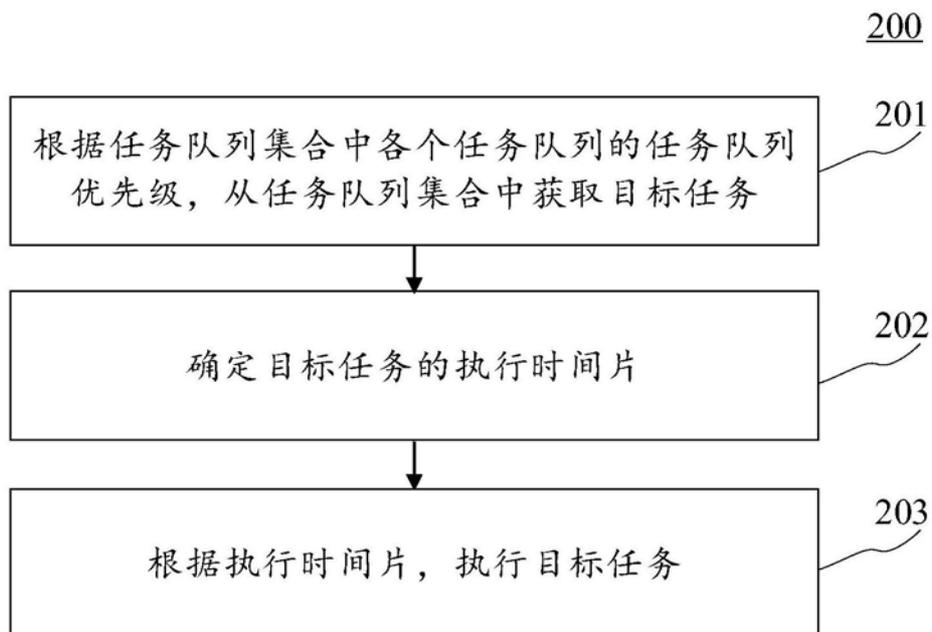


图2

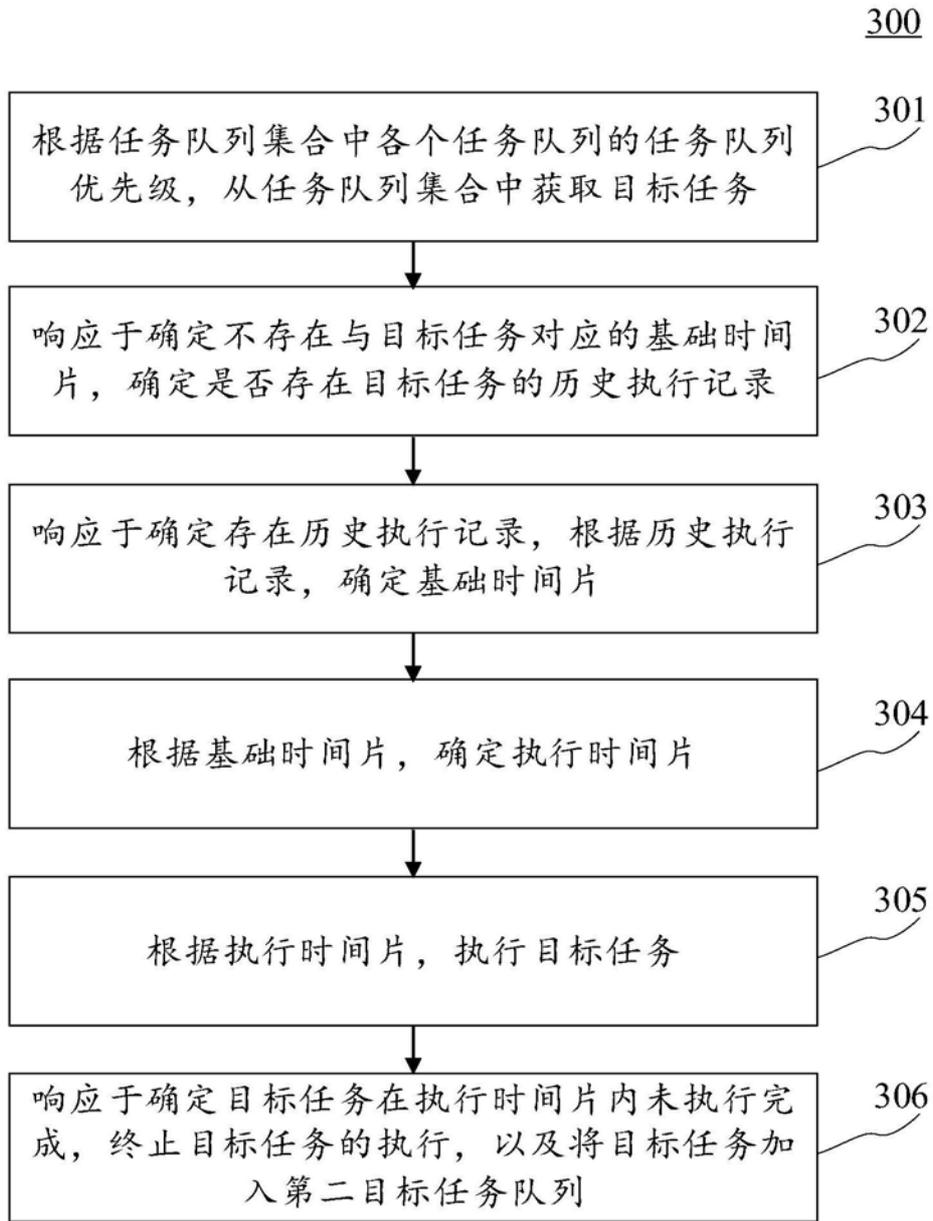


图3

执行开始时刻	执行时长/秒
2020-11-20-13:59:25	50
2020-11-20-15:48:15	48
2020-11-21-12:20:26	56
2020-11-22-06:52:55	2
2020-11-23-09:02:21	51
.....

图4

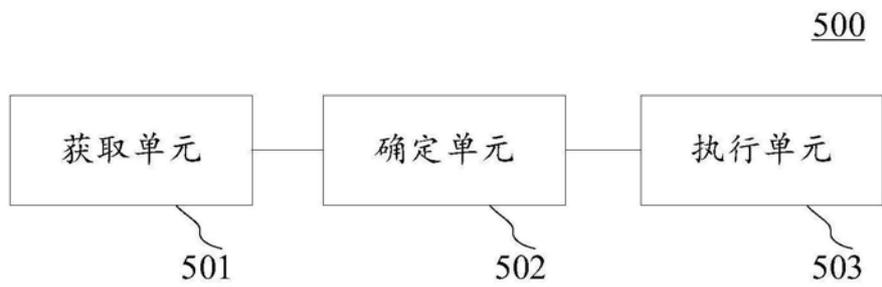


图5

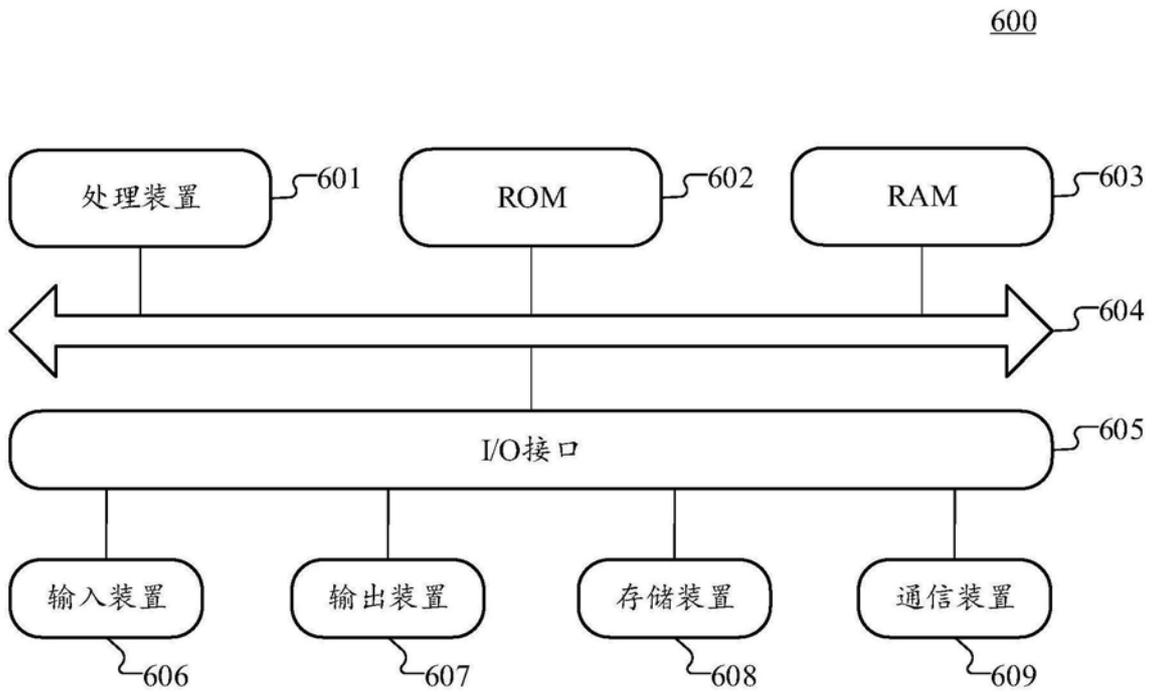


图6