



(21) 申请号 202210692955.2

(22) 申请日 2022.06.17

(65) 同一申请的已公布的文献号

申请公布号 CN 115220910 A

(43) 申请公布日 2022.10.21

(73) 专利权人 中科驭数(北京)科技有限公司

地址 100094 北京市海淀区北清路81号院
一区4号楼14层1401室

(72) 发明人 李明 孟繁毅

(74) 专利代理机构 北京开阳星知识产权代理有限公司 11710

专利代理师 袁义科

(51) Int. Cl.

G06F 9/50 (2006.01)

(56) 对比文件

CN 106649836 A, 2017.05.10

CN 106775758 A, 2017.05.31

CN 108241532 A, 2018.07.03

CN 102541663 A, 2012.07.04

CN 213876703 U, 2021.08.03

CN 107736065 A, 2018.02.23

CN 1866218 A, 2006.11.22

CN 110990161 A, 2020.04.10

US 2019384641 A1, 2019.12.19

R. Lavanya et al..Multi-attribute
resource discovery mechanism using
bitmaps on HPGRID system.《2012
International Conference on Recent Trends
in Information Technology》.2012,全文.

刘忠麟;艾中良;高泽;李常宝.面向司法专
题分析的大数据计算资源调度方法.计算机工程
与应用.2017, (20), 全文.

向鹏;黄海于.耦合分布式仿真中任务调度
的研究与实现.计算机技术与发展.2013, (12),
全文.

审查员 魏旭阳

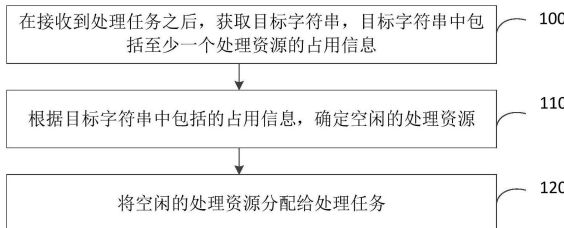
权利要求书2页 说明书9页 附图2页

(54) 发明名称

一种资源调度方法、装置及设备

(57) 摘要

本公开涉及一种资源调度方法、装置及设备,该资源调度方法通过在接收到处理任务之后,获取包括至少一个处理资源的占用信息的目标字符串,根据目标字符串中包括的占用信息,确定空闲的处理资源,并将空闲的处理资源分配给对应的处理任务,从而快速准确的实现资源的调度,提升了工作效率。



1. 一种资源调度方法,其特征在于,包括:

在接收到处理任务之后,获取目标字符串,所述目标字符串中包括至少一个处理资源的占用信息;

根据所述目标字符串中包括的所述占用信息,确定空闲的处理资源;

将所述空闲的处理资源分配给所述处理任务,所述目标字符串为二进制字符串;

其中,所述根据所述目标字符串中包括的所述占用信息,确定空闲的处理资源,包括:对所述目标字符串进行取反处理,得到第一字符串;将所述目标字符串与数值1进行求和处理,得到第二字符串;对所述第一字符串和所述第二字符串进行与处理,得到第三字符串;将所述第三字符串与所述处理资源进行映射,根据映射关系确定所述第三字符串中特殊字符所对应的处理资源为空闲的处理资源。

2. 根据权利要求1所述的资源调度方法,其特征在于,所述根据所述目标字符串中包括的所述占用信息,确定空闲的处理资源,包括:

根据所述目标字符串中包含的占用信息,确定目标占用信息在所述目标字符串中的位置,所述目标占用信息用于指示资源已被占用;

根据所述目标占用信息的所述位置与处理资源之间的映射关系,将所述位置对应的处理资源确定为空闲的处理资源。

3. 根据权利要求1所述的资源调度方法,其特征在于,所述将所述空闲的处理资源分配给所述处理任务,包括:

在所述空闲的处理资源有多个时,根据所述处理任务的任务类型,从所述多个空闲的处理资源中,确定和所述任务类型匹配的处理资源分配给所述处理任务。

4. 根据权利要求1-3任一项所述的资源调度方法,其特征在于,所述将所述空闲的处理资源分配给所述处理任务之后,还包括:

在所述目标字符串中更新所述分配给所述处理任务的处理资源的占用信息。

5. 一种资源调度装置,其特征在于,包括:

信息获取模块,用于在接收到处理任务之后,获取目标字符串,所述目标字符串中包括至少一个处理资源的占用信息;

信息处理模块,用于根据所述目标字符串中包括的所述占用信息,确定空闲的处理资源;

资源调度模块,用于将所述空闲的处理资源分配给所述处理任务,所述目标字符串为二进制字符串;

其中,所述信息处理模块包括:

第一字符串获取子模块,用于对所述目标字符串进行取反处理,得到第一字符串;

第二字符串获取子模块,用于将所述目标字符串与数值1进行求和处理,得到第二字符串;

计算子模块,用于对所述第一字符串和所述第二字符串进行与处理,得到第三字符串;

第二空闲资源确定子模块,用于将所述第三字符串与所述处理资源进行映射,根据映射关系确定所述第三字符串中特殊字符所对应的处理资源为空闲的处理资源。

6. 根据权利要求5所述的资源调度装置,其特征在于,所述信息处理模块包括:

占用信息位置确定子模块,用于根据所述目标字符串中包含的占用信息,确定目标占

用信息在所述目标字符串中的位置,所述目标占用信息用于指示资源已被占用;

第一空闲资源确定子模块,用于根据所述目标占用信息的所述位置与处理资源之间的映射关系,将所述位置对应的处理资源确定为空闲的处理资源。

7.一种计算机设备,其特征在于,包括:

存储器和处理器,其中,所述存储器中存储有计算机程序,当所述计算机程序被所述处理器执行时,实现如权利要求1-4中任一项所述的方法。

8.一种计算机可读存储介质,其特征在于,所述存储介质中存储有计算机程序,当所述计算机程序被处理器执行时,实现如权利要求1-4中任一项所述的方法。

一种资源调度方法、装置及设备

技术领域

[0001] 本公开涉及计算机信息管理技术领域,尤其涉及一种资源调度方法、装置及设备。

背景技术

[0002] 在计算机的使用过程中和硬件设计的过程中存在着大量资源选择和分配的工作负载,包括计算机接口(I/O)、内存、中央处理器(central processing unit,CPU)分配等,在这种情况下,如何快速准确的实现资源的调度是需要解决的问题。

发明内容

[0003] 因此,本公开提供了一种资源调度方法、装置及设备。

[0004] 本公开实施例的第一方面提供了一种资源调度方法,该方法包括:在接收到处理任务之后,获取目标字符串,目标字符串中包括至少一个处理资源的占用信息;根据目标字符串中包括的占用信息,确定空闲的处理资源;将空闲的处理资源分配给处理任务。

[0005] 在一种实施方式中,根据目标字符串中包括的占用信息,确定空闲的处理资源,可以包括:根据目标字符串中包含的占用信息,确定目标占用信息在目标字符串中的位置,目标占用信息用于指示资源已被占用;根据目标占用信息的位置与处理资源之间的映射关系,将位置对应的处理资源确定为空闲的处理资源。

[0006] 在又一种实施方式中,目标字符串为二进制字符串;根据目标字符串中包括的占用信息,确定空闲的处理资源,可以包括:对目标字符串进行取反处理,得到第一字符串;将目标字符串与数值1进行求和处理,得到第二字符串;对第一字符串和第二字符串进行与处理,得到第三字符串;将第三字符串与处理资源进行映射,根据映射关系确定第三字符串中特殊字符所对应的处理资源为空闲的处理资源。

[0007] 在又一种实施方式中,将空闲的处理资源分配给处理任务,可以包括:在空闲的处理资源有多个时,根据处理任务的任务类型,从多个空闲的处理资源中,确定和任务类型匹配的处理资源分配给处理任务。

[0008] 在一种实施方式中,将空闲的处理资源分配给处理任务之后,还可以包括:在目标字符串中更新分配给处理任务的处理资源的占用信息。

[0009] 本公开实施例的第二方面提供了一种资源调度装置,该装置包括:

[0010] 信息获取模块,用于在接收到处理任务之后,获取目标字符串,目标字符串中包括至少一个处理资源的占用信息;

[0011] 信息处理模块,用于根据目标字符串中包括的占用信息,确定空闲的处理资源;

[0012] 资源调度模块,用于将空闲的处理资源分配给处理任务。

[0013] 在一种实施方式中,信息处理模块,包括:

[0014] 占用信息位置确定子模块,用于根据目标字符串中包含的占用信息,确定目标占用信息在目标字符串中的位置,目标占用信息用于指示资源已被占用;

[0015] 第一空闲资源确定子模块,用于根据目标占用信息的位置与处理资源之间的映射

关系,将位置对应的处理资源确定为空闲的处理资源。

[0016] 在另一种实施方式中,信息处理模块,还包括:

[0017] 第一字符串获取子模块,用于对目标字符串进行取反处理,得到第一字符串;

[0018] 第二字符串获取子模块,用于将目标字符串与数值1进行求和处理,得到第二字符串;

[0019] 计算子模块,用于对第一字符串和第二字符串进行与处理,得到第三字符串;

[0020] 第二空闲资源确定子模块,用于将第三字符串与处理资源进行映射,根据映射关系确定第三字符串中特殊字符所对应的处理资源为空闲的处理资源。

[0021] 在一种实施方式中,资源调度模块,用于:

[0022] 在空闲的处理资源有多个时,根据处理任务的任务类型,从多个空闲的处理资源中,确定和任务类型匹配的处理资源分配给处理任务。

[0023] 在一种实施方式中,本公开实施例第二方面提供的装置还可以包括:

[0024] 信息更新模块,用于在目标字符串中更新分配给处理任务的处理资源的占用信息。

[0025] 本公开实施例的第三方面提供了一种计算机设备,该计算机设备包括存储器和处理器,其中,存储器中存储有计算机程序,当该计算机程序被处理器执行时,处理器可以实现上述第一方面的方法。

[0026] 本公开实施例的第四方面提供了一种计算机可读存储介质,该存储介质中存储有计算机程序,当该计算机程序被处理器执行时,处理器可以实现上述第一方面的方法。

[0027] 本公开实施例提供的技术方案与现有技术相比具有如下优点:

[0028] 本公开实施例,通过在接收到处理任务之后,获取包括至少一个处理资源的占用信息的目标字符串,根据目标字符串中包括的占用信息,确定空闲的处理资源,并将空闲的处理资源分配给对应的处理任务,从而快速准确的实现资源的调度,提升了工作效率。

附图说明

[0029] 此处的附图被并入说明书中并构成本说明书的一部分,示出了符合本公开的实施例,并与说明书一起用于解释本公开的原理。

[0030] 为了更清楚地说明本公开实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,对于本领域普通技术人员而言,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。

[0031] 图1是本公开实施例提供的一种资源调度方法的流程图;

[0032] 图2是本公开实施例提供的一种资源调度方法中确定空闲的处理资源的流程图;

[0033] 图3是本公开实施例提供的一种资源调度方法中的位图信息示例图;

[0034] 图4是本公开实施例提供的一种资源调度方法中位运算的示例图;

[0035] 图5是本公开实施例提供的一种资源调度装置的结构示意图;

[0036] 图6是本公开实施例中的一种计算机设备的结构示意图。

具体实施方式

[0037] 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。

[0038] 应当理解,本公开的方法实施方式中记载的各个步骤可以按照不同的顺序执行,和/或并行执行。此外,方法实施方式可以包括附加的步骤和/或省略执行示出的步骤。本公开的范围在此方面不受限制。

[0039] 本文使用的术语“包括”及其变形是开放性包括,即“包括但不限于”。术语“基于”是“至少部分地基于”。术语“一个实施例”表示“至少一个实施例”;术语“另一实施例”表示“至少一个另外的实施例”;术语“一些实施例”表示“至少一些实施例”。其他术语的相关定义将在下文描述中给出。

[0040] 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。

[0041] 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”。

[0042] 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。

[0043] 在现有的资源调度方法中,当电子设备接收到处理任务进行资源调度时,可能会出现如下问题:

[0044] 1、在进行资源调度时,需要从头到尾遍历处理资源,对每一个资源进行分析判断,以此得到可以进行调度的处理资源,导致在进行资源调度时需要进行大量的分析判断,进行任务处理的时间较长,降低了工作效率。

[0045] 2、在进行资源调度时,需要创建资源池,将处理资源中未分配使用的资源在该资源池中统一管理,这样导致在进行资源调度的过程中需要一个额外的管理空间处理未分配使用的资源,提高了资源调度成本,降低了工作效率。

[0046] 针对上述问题,本公开实施例提供了一种资源调度方法、装置及设备。下面结合图1-图4对本公开实施例提供的资源调度方法进行说明。

[0047] 在本公开实施例中,该资源调度方法可以由电子设备执行。其中,电子设备可以包括但不限于诸如移动电话、笔记本电脑、数字广播接收器、PDA(个人数字助理)、PAD(平板电脑)、PMP(便携式多媒体播放器)、车载终端(例如车载导航终端)、可穿戴设备等等的移动终端以及诸如数字TV、台式计算机、智能家居设备等等的固定终端。

[0048] 图1示出了本公开实施例提供的一种资源调度方法的流程示意图。

[0049] 如图1所示,该资源调度方法可以包括如下步骤:

[0050] S100、在接收到处理任务之后,获取目标字符串,目标字符串中包括至少一个处理资源的占用信息。

[0051] 在本公开实施例中处理任务可以为电子设备接收到的对资源的分配需求。该资源的分配需求可以为电子设备对计算机接口(I/O)的分配需求,可以为电子设备对内存的分配需求,也可以为电子设备对中央处理器(central processing unit,CPU)的分配需求。

[0052] 在本公开实施例中,处理资源至少包括两种情况:已经分配过和还未分配过。其中,已经分配过的处理资源对应的占用信息为已占用,还未分配过的处理资源对应的占用

信息为未占用。

[0053] 本实施例的电子设备在接收到处理任务之后,获取对应处理资源的资源信息,并至少提取该资源信息中的一个占用信息,根据占用信息将处理资源用字符串进行标识,得到目标字符串。

[0054] 在本实施例中处理资源的占用信息至少包括两种:处理资源已占用和处理资源未占用。其中,处理资源已占用表示对应的处理资源已分配至对应的硬件设备中,处理资源未占用表示对应的处理资源还未进行分配。

[0055] 可选地,对应处理资源的资源信息可以示例性的理解为对应处理资源至少包括是否被占用以及该资源在总体资源中所处的位置信息,并根据对应处理资源是否被占用的资源信息将对应处理资源使用二进制进行标记,将对应处理资源已占用使用二进制标记为“1”,将对应处理资源未占用标记为“0”,根据对应处理资源的上述资源信息构建位图模型,将处理资源中所包含的信息以bit位进行标识,将对应的资源信息映射至位图模型中,其中,bit位的位置顺序表示对应处理资源在总体资源中的位置顺序。

[0056] 步骤S110、根据目标字符串中包括的占用信息,确定空闲的处理资源。

[0057] 在本公开实施例中,可以根据标记好的目标字符串所对应的资源信息得到未占用的资源信息,以此确定空闲的处理资源。再本公开另一种实施方式中,可以根据位图模型获取目标字符串对应的占用信息以及对应的位置信息,并根据获取到的信息确定已被占用的处理资源的位置信息,根据已被占用的处理资源的位置信息以及位图模型与处理资源之间的映射关系,并根据该映射关系确定位图模型中剩余位置所对应的处理资源为空闲资源。

[0058] 步骤S120、将空闲的处理资源分配给处理任务。

[0059] 在本公开实施例中,获取空闲的处理资源的资源信息,根据该资源信息将空闲的处理资源分配至对应的硬件设备中,完成分配任务。

[0060] 在一种示例性的分配方法中,获取到的空闲的处理资源可以为计算机接口资源,根据该计算机接口资源的资源信息进行匹配,匹配到对应的硬件设备,根据处理任务的任務要求,将计算机接口资源分配至硬件设备,完成处理任务,在将空闲的处理资源分配完成后,将对应于目标字符串中的占用信息进行更新,将其更新为已占用。

[0061] 在另一种示例性的分配方法中,获取到的空闲的处理资源可以为内存资源,根据该内存资源的资源信息进行匹配,匹配到对应的硬件设备,根据处理任务的任務要求,将内存资源分配至硬件设备,完成处理任务在将空闲的处理资源分配完成后,将对应于目标字符串中的占用信息进行更新,将其更新为已占用。

[0062] 在另一种示例性的分配方法中,获取到的空闲的处理资源可以为中央处理器资源,根据该中央处理器资源的资源信息进行匹配,匹配到对应的硬件设备,根据处理任务的任務要求,将中央处理器资源分配至硬件设备,完成处理任务在将空闲的处理资源分配完成后,将对应于目标字符串中的占用信息进行更新,将其更新为已占用。

[0063] 本公开实施例,通过在接收到处理任务之后,获取包括至少一个处理资源的占用信息的目标字符串,根据目标字符串中包括的占用信息,确定空闲的处理资源,并将空闲的处理资源分配给对应的处理任务,从而快速准确的实现资源的调度,提升了工作效率。

[0064] 图2是本公开实施例提供的一种资源调度方法中确定空闲的处理资源的流程图,如图2所示,在本公开实施例的一种实施方式中,可以通过如下方法确定空闲的处理资源:

[0065] 步骤S210、对目标字符串进行取反处理,得到第一字符串。

[0066] 步骤S220、将目标字符串与数值1进行求和处理,得到第二字符串。

[0067] 步骤S230、对第一字符串和第二字符串进行与处理,得到第三字符串。

[0068] 步骤S240、将第三字符串与处理资源进行映射,根据映射关系确定第三字符串中特殊字符所对应的处理资源为空闲的处理资源。

[0069] 具体地,可以通过公式 $result = (\sim state) \& (state + 1)$ 确定空闲的处理资源,根据计算结果得到第三字符串,根据第三字符串确定空闲的处理资源。其中,result表示为第三字符串,该第三字符串中至少包括一个未占用的资源信息; $\sim state$ 表示为第一字符串; $state + 1$ 表示为第二字符串。

[0070] 在本公开实施例中,state表示为根据处理资源所构建的位图模型所包含的目标字符串,将该目标字符串进行取反操作,得到对应于目标字符串的掩码,将未占用的资源信息对应的字符串在掩码中凸显出来,以此获得第一字符串。

[0071] 在本公开实施例中,state表示为根据处理资源所构建的位图模型所包含的目标字符串,通过二进制加法,对目标字符串进行求和出来,将目标字符串与数值1相加,得到第二字符串。

[0072] 本公开实施例,通过对目标字符串进行数据处理,得到第一字符串和第二字符串,而后对第一字符串和第二字符串进行位运算的与处理,得到第三字符串,根据位图模型与处理资源标记信息之间的映射关系,确定第三字符串与位图模型之间的映射关系,根据该映射关系得到处理资源中未占用资源对应于第三字符串中特殊字符的位置信息,从而确定空闲资源,提升了在资源调度过程中确定空闲资源的效率,并且通过位运算得到相应的数值,在将对应位置的资源进行分配后,能够及时更新相应位置的占用信息,准确性较高。

[0073] 在本公开另一些实施例中,为更加快速的得到空闲的处理资源,该资源调度方法还可以包括:在空闲的处理资源有多个时,根据处理任务的类型,从多个空闲的处理资源中,确定和任务类型匹配的处理资源分配给处理任务。

[0074] 可选地,可以根据公式 $result = (\sim state) \& (state + 1)$ 得到的计算结果,将计算结果对应的第三字符串与处理资源进行映射,根据映射关系将结果中数值1的位置所对应的处理资源确定为空闲资源,而后将该空闲资源中根据处理任务的类型匹配的处理资源分配给处理任务。

[0075] 本公开实施例,通过在多个空闲的处理资源中,利用公式得到最低位所对应的空闲资源,从而能够快速将空闲资源分配至对应的硬件设备中,提升了工作效率。

[0076] 下面,以图3、图4为例,对本公开实施例的一种资源调度方法进行说明。

[0077] 本公开实施例,示出了四个同类型的资源,假设资源1和资源3的占用信息为已占用,资源2和资源4的占用信息为未占用,将上述四个资源利用二进制字符串进行标识,将资源1和资源3标识为“1”,资源2和资源4标识为“0”,则上述四个资源的目标字符串为“1010”,利用位图模型的到上述四个资源的位图信息,即为图3所示“0101”,使用公式进行与处理,即为图4所示的“ $\sim (0101) \& (0101 + 1)$ ”,得到第三字符串“0010”,根据第三字符串与位图模型之间的映射关系,确定第三字符串中数值1的位置所对应的处理资源为空闲的处理资源,将其分配至对应的处理任务,同时对完成分配后的处理资源与位图模型进行或处理,更新分配后处理资源对应的标识信息。

[0078] 本实施例通过将四个同类型的资源利用二进制字符串进行标识,并将标识后的字符串进行计算,从而快速计算得到了四个同类型字符串中最低位的未占用的资源信息,提升了工作效率。

[0079] 需要说明的是,在进行资源的调度过程中直接确定空闲的处理资源的过程至少部分内容的方法与上述实施例相似,在此不做赘述。

[0080] 图5是本公开实施例提供的一种资源调度装置的结构示意图。该检测装置可以理解为上述实施例中的电子设备中的部分功能模块。如图5所示,资源调度装置50包括:

[0081] 信息获取模块51,用于在接收到处理任务之后,获取目标字符串,目标字符串中包括至少一个处理资源的占用信息;

[0082] 信息处理模块52,用于根据目标字符串中包括的占用信息,确定空闲的处理资源;

[0083] 资源调度模块53,用于将空闲的处理资源分配给处理任务。

[0084] 在一种实施方式中,信息处理模块52,可以包括:

[0085] 占用信息位置确定模块,用于根据目标字符串中包含的占用信息,确定目标占用信息在目标字符串中的位置,目标占用信息用于指示资源已被占用;

[0086] 第一空闲资源确定子模块,用于根据目标占用信息的位置与处理资源之间的映射关系,将位置对应的处理资源确定为空闲的处理资源。

[0087] 在另一种实施方式中,信息处理模块52,还可以包括:

[0088] 第一字符串获取子模块,用于对目标字符串进行取反处理,得到第一字符串;

[0089] 第二字符串获取子模块,用于将目标字符串与数值1进行求和处理,得到第二字符串;

[0090] 计算子模块,用于对第一字符串和第二字符串进行与处理,得到第三字符串;

[0091] 第二空闲资源确定子模块,用于将第三字符串与处理资源进行映射,根据映射关系确定第三字符串中特殊字符所对应的处理资源为空闲的处理资源。

[0092] 在另一种实施方式中,资源调度模块53,用于:

[0093] 在空闲的处理资源有多个时,根据处理任务的任务类型,从多个空闲的处理资源中,确定和任务类型匹配的处理资源分配给处理任务。

[0094] 在另一种实施方式中,该资源调度装置50,还可以包括:

[0095] 信息更新模块,用于在目标字符串中更新分配给处理任务的处理资源的占用信息。

[0096] 本公开实施例提供的装置能够执行图1-图4中任一实施例的方法,其执行方式和有益效果类似,在这里不再赘述。

[0097] 本公开实施例还提供一种计算机设备,该计算机设备包括处理器和存储器,其中,所述存储器中存储有计算机程序,当所述计算机程序被所述处理器执行时可以实现上述图1-图4中任一实施例的方法。

[0098] 示例的,图6是本公开实施例中的一种计算机设备的结构示意图。下面具体参考图6,其示出了适于用来实现本公开实施例中的计算机设备1000的结构示意图。本公开实施例中的计算机设备1000可以包括但不限于诸如移动电话、笔记本电脑、数字广播接收器、PDA(个人数字助理)、PAD(平板电脑)、PMP(便携式多媒体播放器)、车载终端(例如车载导航终端)等等的移动终端以及诸如数字TV、台式计算机等等可以利用移动网络传输数据的固定

终端。图6示出的移计算机设备仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0099] 如图6所示,计算机设备1000可以包括处理装置(例如中央处理器、图形处理器等)1001,其可以根据存储在只读存储器(ROM)1002中的程序或者从存储装置1008加载到随机访问存储器(RAM)1003中的程序而执行各种适当的动作和处理。在RAM 1003中,还存储有计算机设备1000操作所需的各种程序和数据。处理装置1001、ROM 1002以及RAM 1003通过总线1004彼此相连。输入/输出(I/O)接口1005也连接至总线1004。

[0100] 通常,以下装置可以连接至I/O接口1005: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1006;包括例如液晶显示器(LCD)、扬声器、振动器等的输出装置1007;包括例如磁带、硬盘等的存储装置1008;以及通信装置1009。通信装置1009可以允许计算机设备1000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计算机设备1000,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。

[0101] 特别地,根据本公开的实施例,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,本公开的实施例包括一种计算机程序产品,其包括承载在非暂态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。在这样的实施例中,该计算机程序可以通过通信装置1009从网络上被下载和安装,或者从存储装置1008被安装,或者从ROM 1002被安装。在该计算机程序被处理装置1001执行时,执行本公开实施例的方法中限定的上述功能。

[0102] 需要说明的是,本公开上述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一——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(EPROM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。

[0103] 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。

[0104] 上述计算机可读介质可以是上述计算机设备中所包含的;也可以是单独存在,而未装配入该计算机设备中。

[0105] 上述计算机可读介质承载有一个或者多个计算机程序,当上述一个或者多个计算机程序被该计算机设备执行时,使得该计算机设备:在计算机设备的移动数据链路处于活跃状态时,检测移动数据链路的连通性;若检测到移动数据链路断流,则对移动数据链路进行重置。

[0106] 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操作的计算机程序代码,上述程序设计语言包括但不限于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)。

[0107] 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。

[0108] 描述于本公开实施例中所涉及到的模块可以通过软件的方式实现,也可以通过硬件的方式来实现。其中,模块的名称在某种情况下并不构成对该模块本身的限定。

[0109] 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。

[0110] 在本公开的上下文中,机器可读介质可以是有形的介质,其可以包含或存储以供指令执行系统、装置或设备使用或与指令执行系统、装置或设备结合地使用的程序。机器可读介质可以是机器可读信号介质或机器可读储存介质。机器可读介质可以包括但不限于电子的、磁性的、光学的、电磁的、红外的、或半导体系统、装置或设备,或者上述内容的任何合适组合。机器可读存储介质的更具体示例会包括基于一个或多个线的电气连接、便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦除可编程只读存储器(EPROM或快闪存储器)、光纤、便捷式紧凑盘只读存储器(CD-ROM)、光学储存设备、磁储存设备、或上述内容的任何合适组合。

[0111] 本公开实施例还提供一种计算机可读存储介质,所述存储介质中存储有计算机程序,当所述计算机程序被处理器执行时可以实现上述图1-图4中任一实施例的方法,其执行

方式和有益效果类似,在这里不再赘述。

[0112] 本公开实施例还提供一种计算机程序产品,该计算机程序产品中包括计算机程序,当该计算机程序被处理器执行时,使得处理器可以执行上述图1-图4中任一实施例的方法。

[0113] 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。

[0114] 以上所述仅是本公开的具体实施方式,使本领域技术人员能够理解或实现本公开。对这些实施例的多种修改对本领域的技术人员来说将是显而易见的,本文中所定义的一般原理可以在不脱离本公开的精神或范围的情况下,在其它实施例中实现。因此,本公开将不会被限制于本文所述的这些实施例,而是要符合与本文所公开的原理和新颖特点相一致的最宽的范围。

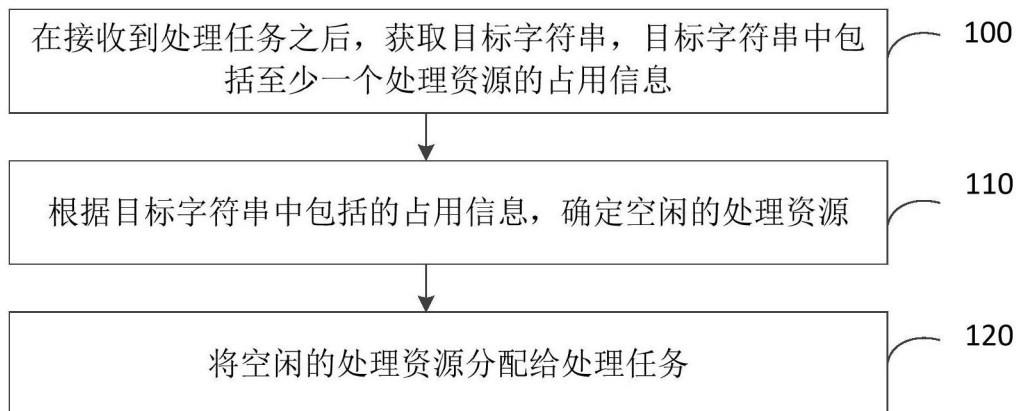


图1

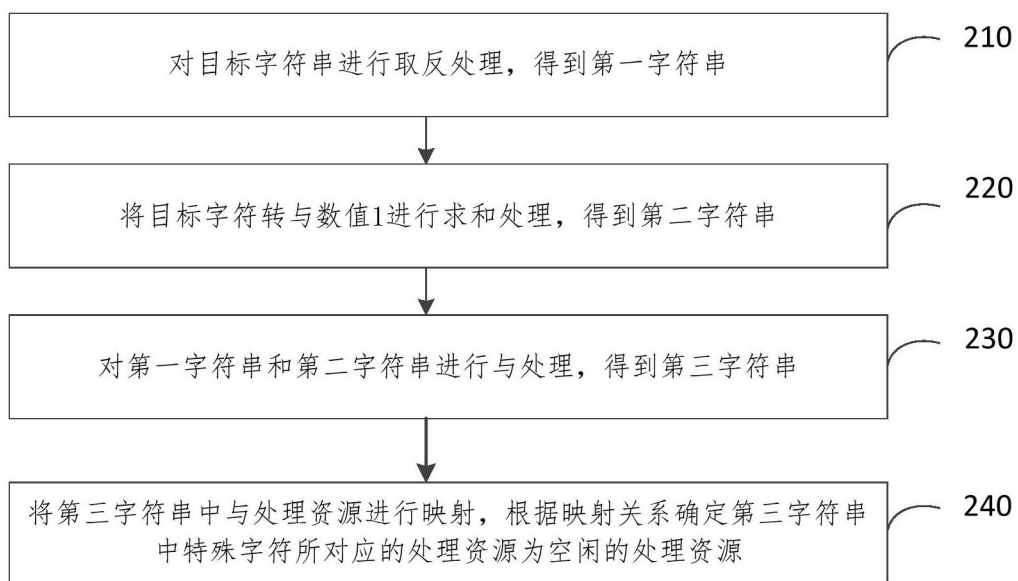


图2



图3

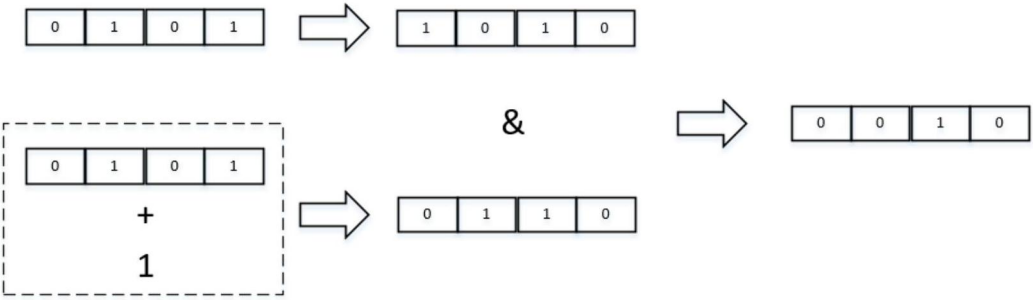


图4

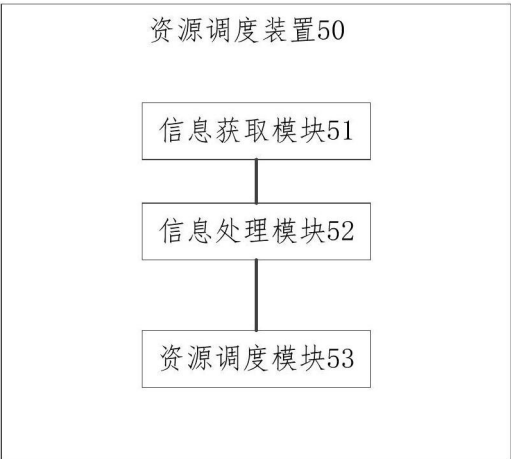


图5

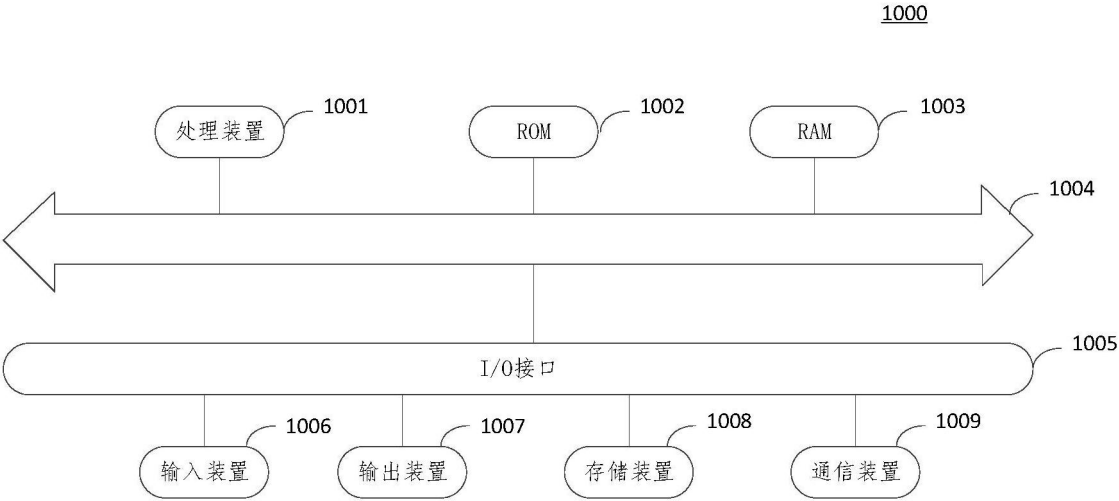


图6