



(12) 发明专利

(10) 授权公告号 CN 117331858 B

(45) 授权公告日 2024. 01. 30

(21) 申请号 202311628285.9

(22) 申请日 2023.11.30

(65) 同一申请的已公布的文献号
申请公布号 CN 117331858 A

(43) 申请公布日 2024.01.02

(73) 专利权人 摩尔线程智能科技(北京)有限责
任公司

地址 100080 北京市海淀区翠微中里14号
楼四层B655

(72) 发明人 请求不公布姓名 请求不公布姓名
请求不公布姓名 请求不公布姓名

(74) 专利代理机构 北京林达刘知识产权代理事
务所(普通合伙) 11277

专利代理师 李茂家 周蕾

(51) Int.Cl.

G06F 12/0871 (2016.01)

G06F 12/0893 (2016.01)

G06T 1/20 (2006.01)

G06T 1/60 (2006.01)

(56) 对比文件

CN 112559049 A, 2021.03.26

KR 20090119032 A, 2009.11.19

王一达;赵长海;李超;张建磊;晏海华;张威
毅.异构计算环境下的三维Kirchhoff叠前深度
偏移混合域并行算法.石油地球物理勘探.2018,
(第03期),全文.

审查员 刘肇荣

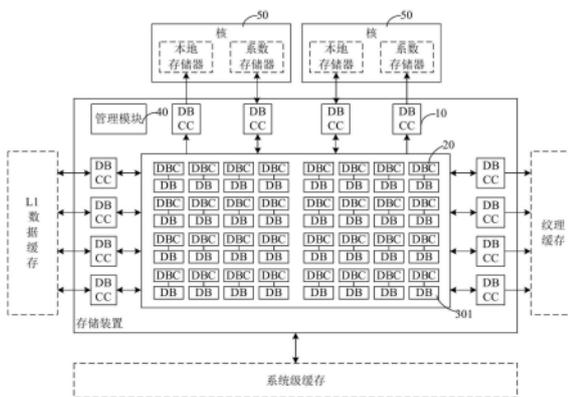
权利要求书5页 说明书24页 附图9页

(54) 发明名称

存储装置及数据处理系统

(57) 摘要

本公开涉及数据存储技术领域,尤其涉及一
种存储装置及数据处理系统。存储装置设置在处
理器的各核的外部,包括:多个数据块、多个控制
模块、至少一个控制中心模块和管理模块,各控
制模块与对应的一个数据块连接并控制该数据
块的访问;各控制中心模块与被分配的数据块所
连接的控制模块通信连接,以通过控制模块对被
分配的数据块的访问进行控制;管理模块若接收
到的资源管理请求为配置请求,则为配置请求指
示的各第一控制中心模块分配数据块,并根据访
问类型对第一控制中心模块、被分配的数据块所
连接的控制模块的访问控制逻辑进行配置。实现
存储资源的统一管理,可以根据不同的需要灵
活、动态分配存储资源,且使得各类存储空间都
可得到有效利用。



1. 一种存储装置,其特征在于,所述装置设置在处理器的各核的外部,所述装置包括:多个数据块、多个控制模块、至少一个控制中心模块和管理模块,

各所述数据块用于存储数据,各所述控制模块与对应的一个数据块连接并控制该数据块的访问;

各控制中心模块与被分配的数据块所连接的控制模块通信连接,以通过实现通信连接的控制模块对被分配的数据块的访问进行控制;

所述管理模块,用于接收资源管理请求,在所述资源管理请求为配置请求的情况下,为所述配置请求指示的各第一控制中心模块分配数据块,并根据所述配置请求指示的访问类型对所述第一控制中心模块、被分配的数据块所连接的控制模块的访问控制逻辑进行配置;

其中,所述第一控制中心模块为所述至少一个控制中心模块中的部分或全部。

2. 根据权利要求1所述的装置,其特征在于,不同访问类型对应的访问控制逻辑不同,所述访问类型包括缓存行访问和静态存储访问;所述缓存行访问对应的访问控制逻辑为以缓存行形式对数据块进行访问,所述静态存储访问对应的访问控制逻辑为以粒度单元的形式对数据块进行访问,所述第一控制中心模块所分配的数据块作为核内存储器或核外缓存。

3. 根据权利要求1或2所述的装置,其特征在于,

所述管理模块,还用于在所述资源管理请求为释放请求的情况下,释放所述释放请求指示的各第二控制中心模块及其所分配的数据块;

其中,所述第二控制中心模块为所述第一控制中心模块中的部分或全部;被释放的各所述第二控制中心模块、各所述第二控制中心模块所分配的数据块连接的控制模块均被关闭,以等待再次分配。

4. 根据权利要求2所述的装置,其特征在于,各所述控制中心模块包括控制单元、统计单元、重构单元和总命中测试单元,各所述控制模块包括命中测试单元、信息存储单元、块控制单元、替换单元和读写单元;

在所述访问类型为缓存行访问的情况下,所述第一控制中心模块和被分配的数据块的控制模块中的各单元均被配置为开启;

在所述访问类型为静态存储访问的情况下,所述第一控制中心模块中的控制单元、被分配的数据块所连接的控制模块中的块控制单元和读写单元被配置为开启。

5. 根据权利要求4所述的装置,其特征在于,

所述控制单元,用于根据接收到的访问类型为缓存行访问的访问申请生成第一请求,并将所述第一请求发送至总命中测试单元,所述第一请求中携带有缓存行标签;

总命中测试单元,用于将所述第一请求广播至各所述块控制单元;

各所述块控制单元,用于在确定所述第一请求的访问类型为缓存行访问的情况下,将所述第一请求发送至所在控制模块中的命中测试单元;

各所述命中测试单元,用于根据接收到的第一请求对所控制的数据块进行缓存行的命中测试,并将测试结果返回所述统计单元;

统计单元,用于根据接收到的所述控制模块返回的测试结果生成总结果,并将所述总结果发送至控制单元;其中,所述总结果包括命中结果和命中失败结果,所述命中结果是接

收到任意第一个结果为命中的测试结果的情况下生成的；所述命中失败结果是接收到来自各控制模块的测试结果均为未命中的情况下生成的；

所述测试结果中携带有以下信息的至少一种：指示命中测试是否命中的状态信息、进行命中测试的数据块的数据块ID、所述缓存行标签；结果为命中的测试结果中还携带有命中缓存行的缓存行ID。

6. 根据权利要求5所述的装置，其特征在于，

所述统计单元，还用于在生成的总结果为所述命中结果的情况下，生成停止信号请求，并控制所述总命中测试单元将所述停止信号请求发广播至还未返回测试结果的各命中测试单元；

各所述命中测试单元，用于响应于接收到的所述停止信号请求，停止针对所述第一请求的命中测试。

7. 根据权利要求5所述的装置，其特征在于，

所述控制单元，还用于在所述总结果为表示全部未命中的失败结果的情况下，向所述重构单元发送替换请求；

所述重构单元，用于在接收到替换请求的情况下，根据预先构建的管理组织结构从数据块分区中确定出待查数据块，并向管理所述待查数据块的控制模块中的替换单元发送查找请求；

所述替换单元，用于响应于所述查找请求，采用PLRU算法查找出所述待查数据块中的可替换缓存行，并生成替换结果返回所述重构单元；

所述重构单元，还用于根据所述替换结果生成替换结果并返回所述控制单元，所述替换结果携带有所述可替换缓存行的缓存行ID、所述可替换缓存行所在的数据块的数据块ID；

其中，所述数据块分区包括所述重构单元所在的控制中心模块所分配的所有数据块，所述管理组织结构包括对应于所述数据块分区的二叉树结构和所述二叉树结构中各节点的节点信息。

8. 根据权利要求4所述的装置，其特征在于，

所述控制单元，用于根据接收到的访问类型为静态存储访问的访问申请生成第二请求，并将所述第二请求发送至对应的块控制单元，所述第二请求中携带有地址信息；

所述块控制单元，用于在接收到第二请求后，在确定所述第二请求的访问类型为静态存储访问的情况下，控制所述读写单元根据所述地址信息从所述块控制单元所控制的数据块的中读取数据或写入数据。

9. 根据权利要求1所述的装置，其特征在于，所述装置还包括：

数据块复制模块，用于接收来自第一控制模块的复制请求，并根据所述复制请求确定所要复制的待复制数据所在的第二数据块以及控制所述第二数据块的第二控制模块，从所述第二控制模块请求到的待复制数据并写入所述第一控制模块所控制的第一数据块中；

其中，所述第一控制模块和所述第二控制模块不同，且所述第一控制模块和所述第二控制模块分别为所述存储装置中已经被分配的数据块对应的控制模块中的任意一个。

10. 根据权利要求9所述的装置，其特征在于，所述数据块复制模块，包括：

请求缓冲器，用于接收来自所述第一控制模块的复制请求并存储；

控制器,用于从所述请求缓冲器中获取到当前需要执行的复制请求,将所述复制请求拆分为多个循环请求,依次向根据各所述循环请求确定出的第二控制模块发出数据请求或根据各所述循环请求确定出的第二控制模块所控制的第二数据块对应的上一级缓存发出数据请求;

数据缓冲器,用于接收来自所述第二控制模块或上一级缓存的待复制数据,并将所述待复制数据写入所述第一控制模块所控制的第一数据块中。

11. 根据权利要求7所述的装置,其特征在于,所述节点信息包括:处理状态、左分区计数器和右分区计数器,

其中,根据预先构建的管理组织结构从数据块分区中确定出待查数据块,包括:

在接收到替换请求的情况下,将数据块分区对应的二叉树结构中的根节点确定为第一个目标节点并获取所述目标节点的节点信息;

在所述目标节点为非叶节点的情况下,将当前所述目标节点的处理状态所指示的替换方向上的所述目标节点的子节点确定为下一目标节点以及更新所述目标节点的对应于替换方向的目标计数器,直至确定出的目标节点为叶节点,则基于所述叶节点确定出待查数据块;所述目标计数器为所述目标节点的左分区计数器和右分区计数器中对应于所述目标节点的处理状态所指示的替换方向的计数器,各节点的处理状态所指示的替换方向为节点的左侧分支或右侧分支;

所述信息存储单元用于存储所控制数据块中各缓存行的基本信息,所述基本信息包括各缓存行的缓存行标签和缓存行ID。

12. 根据权利要求11所述的装置,其特征在于,基于所述叶节点确定出待查数据块,包括:

将所述叶节点的处理状态所指示的替换方向对应的数据块确定为待查数据块;或者将所述叶节点对应的数据块确定为待查数据块。

13. 根据权利要求11所述的装置,其特征在于,所述节点信息还包括左分区权重和右分区权重,

所述重构单元,还用于在所述目标计数器的数值为预设值的情况下,重置所述目标计数器为初始值以及更改所述目标节点的处理状态所指示的替换方向;

其中,所述初始值为对应于所述目标计数器所在分支方向的分区权重、所述预设值为0;或者,所述初始值为0、所述预设值为对应于所述目标计数器所在分支方向的分区权重;左分区计数器所在分支方向的分区权重为左分区权重,右分区计数器所在分支方向的分区权重为右分区权重。

14. 根据权利要求13所述的装置,其特征在于,

所述重构单元,还用于在确定所述目标节点的目标计数器为无效和/或所述目标节点对应于处理状态所指示的替换方向的分区权重为0的情况下,确定所述目标节点为叶节点;

其中,所述目标节点对应于处理状态所指示的替换方向的分区权重为左分区权重或右分区权重。

15. 根据权利要求11所述的装置,其特征在于,所述节点信息还包括左指针和右指针,

所述重构单元,还用于根据当前所述目标节点的处理状态所指示的替换方向对应的目标指针查找到下一目标节点;

其中,若当前所述目标节点的处理状态所指示的替换方向为左侧分支,则所述目标指针为左指针;若当前所述目标节点的处理状态所指示的替换方向为右侧分支,则所述目标指针为右指针。

16. 根据权利要求12所述的装置,其特征在于,所述叶节点的节点信息还包括:右地址索引、右有效索引标识、左地址索引和左有效索引标识,

其中,将所述叶节点的处理状态所指示的替换方向对应的数据块确定为待查数据块,包括:在所述叶节点的处理状态所指示的替换方向对应的有效索引标识为有效的情况下,将所述叶节点的处理状态所指示的替换方向对应的目标地址索引所指向的数据块确定为待查数据块;

其中,若当前所述目标节点的处理状态所指示的替换方向为左侧分支,则所述目标地址索引为所述左地址索引;若当前所述目标节点的处理状态所指示的替换方向为右侧分支,则所述目标地址索引为所述右地址索引。

17. 根据权利要求12所述的装置,其特征在于,所述叶节点的节点信息还包括左地址索引、左有效索引标识,或者所述叶节点的节点信息还包括右地址索引和右有效索引标识,

其中,将所述叶节点对应的数据块确定为待查数据块,包括:在所述叶节点的节点信息中存在的左有效索引标识或右有效索引标识为有效的情况下,将所述叶节点的节点信息中存在的地址索引所指向的数据块确定为待查数据块。

18. 根据权利要求13所述的装置,其特征在于,各节点的左分区权重、右分区权重与所述节点的左侧分支中数据块的左侧数量、所述节点的右侧分支中数据块的右侧数量相关,

在所述节点的左侧数量和右侧数量均不为零的情况下,所述节点的左分区权重、右分区权重的值,根据节点的左侧数量和右侧数量的最简整数比确定;

在所述节点的左侧数量和/或右侧数量为零的情况下,所述节点的左分区权重、右分区权重的值分别设置为节点的左侧数量和右侧数量。

19. 根据权利要求11-18任意一项所述的装置,其特征在于,

所述重构单元,还用于响应于重构请求,根据配置信息设置所述数据块分区中的数据块的块数量以及各所述数据块中缓存行的行数量,并设置各所述数据块的数据块ID、缓存行的缓存行ID;根据所述数据块分区中数据块的块数量,创建对应所述数据块分区的二叉树结构,以及确定所述二叉树结构中各节点的节点信息,完成所述数据块分区的管理组织结构配置。

20. 根据权利要求1所述的装置,其特征在于,各所述数据块中包括 2^n 个缓存行, n 为正整数。

21. 一种数据处理系统,其特征在于,包括:

处理器,所述处理器中设置有位于所述处理器的各核外部的存储装置;

所述存储装置包括:多个数据块、多个控制模块、至少一个控制中心模块和管理模块;

各所述数据块用于存储数据,各所述控制模块与对应的一个数据块连接并控制该数据块的访问;

各控制中心模块与被分配的数据块所连接的控制模块通信连接,以实现通信连接的控制模块对被分配的数据块的访问进行控制;

所述管理模块,用于接收资源管理请求,在所述资源管理请求为配置请求的情况下,为

所述配置请求指示的各第一控制中心模块分配数据块,并根据所述配置请求指示的访问类型对所述第一控制中心模块、被分配的数据块所连接的控制模块的访问控制逻辑进行配置;

其中,所述第一控制中心模块为所述至少一个控制中心模块中的部分或全部。

存储装置及数据处理系统

技术领域

[0001] 本公开涉及数据存储技术领域,尤其涉及一种存储装置及数据处理系统。

背景技术

[0002] 随着不同领域对处理器不同需求的变化,现阶段处理器的种类也多种多样,例如,CPU(Central Processing Unit,中央处理单元/器)、GPU(Graphic Process Unit,图形处理器,又称显示核心、视觉处理器、显示芯片),等等。以GPU为例,GPU运算工作相关的数据可以存储在片上或片外的存储单元中。其中,外部全局存储器(Global Memory,也称外部内存、全局内存)作为片外存储单元,设置在离GPU的core(核心处理器,也称核、运算核心、内核等)较远的地方,访问速度慢。为了解决外部全局存储器访问速度慢的问题,通常在处理器的core内部增加片上静态存储单元(如本地存储器(Local Memory)、系数存储器(coeffcient memory)等等)以及处于GPU外的缓存(如L1数据缓存(L1 data cache)、纹理缓存(texture cache)等等),其先对读写数据进行缓存,方便后续GPU读写时访问到缓存,可快速获得,无需再从外部中获取。但由于不同处理器在应用到不同的场景中对其缓存和片上存储器的访问需求不同,不同处理器的缓存和片上存储器的使用频率等使用情况也存在差异,比如,GPU可用于图形渲染和AI等通用计算领域。在GPU实际应用中,L1数据缓存在这两个不同领域均会使用到;但是Local Memory通常在通用计算领域广泛使用,在图形渲染领域很少使用。如何解决相关技术中这种不同应用场景下处理器的存储单元使用不均衡的问题,是值得关注的。

发明内容

[0003] 有鉴于此,本公开提出了一种存储装置及数据处理系统。

[0004] 根据本公开的一方面,提供了一种存储装置,所述装置设置在所述处理器的各核的外部,所述装置包括:多个数据块、多个控制模块、至少一个控制中心模块和管理模块;

[0005] 各所述数据块用于存储数据,各所述控制模块与对应的一个数据块连接并控制该数据块的访问;

[0006] 各控制中心模块与被分配的数据块所连接的控制模块通信连接,以通过实现通信连接的控制模块对被分配的数据块的访问进行控制;

[0007] 所述管理模块,用于接收资源管理请求,在所述资源管理请求为配置请求的情况下,为所述配置请求指示的各第一控制中心模块分配数据块,并根据所述配置请求指示的访问类型对所述第一控制中心模块、被分配的数据块所连接的控制模块的访问控制逻辑进行配置;

[0008] 其中,所述第一控制中心模块为所述至少一个控制中心模块中的部分或全部。

[0009] 在一种可能的实现方式中,不同访问类型对应的访问控制逻辑不同,所述访问类型包括缓存行访问和静态存储访问;所述缓存行访问对应的访问控制逻辑为以缓存行形式对数据块进行访问,所述静态存储访问对应的访问控制逻辑为以粒度单元的形式对数据块

进行访问,所述第一控制中心模块所分配的数据块作为核内存储器或核外缓存。

[0010] 在一种可能的实现方式中,所述管理模块,还用于在所述资源管理请求为释放请求的情况下,释放所述释放请求指示的各第二控制中心模块及其所分配的数据块;

[0011] 其中,所述第二控制中心模块为所述第一控制中心模块中的部分或全部;被释放的各所述第二控制中心模块、各所述第二控制中心模块所分配的数据块连接的控制模块均被关闭,以等待再次分配。

[0012] 在一种可能的实现方式中,各所述控制中心模块包括控制单元、统计单元、重构单元和总命中测试单元,各所述控制模块包括命中测试单元、信息存储单元、块控制单元、替换单元和读写单元;

[0013] 在所述访问类型为缓存行访问的情况下,所述第一控制中心模块和被分配的数据块的控制模块中的各单元均被配置为开启;

[0014] 在所述访问类型为静态存储访问的情况下,所述第一控制中心模块中的控制单元、被分配的数据块所连接的控制模块中的块控制单元和读写单元被配置为开启。

[0015] 在一种可能的实现方式中,所述控制单元,用于根据接收到的访问类型为缓存行访问的访问申请生成第一请求,并将所述第一请求发送至总命中测试单元,所述第一请求中携带有缓存行标签;

[0016] 总命中测试单元,用于将所述第一请求广播至各所述块控制单元;

[0017] 各所述块控制单元,用于在确定所述第一请求的访问类型为缓存行访问的情况下,将所述第一请求发送至所在控制模块中的命中测试单元;

[0018] 各所述命中测试单元,用于根据接收到的第一请求对所控制的数据块进行缓存行的命中测试,并将测试结果返回所述统计单元;

[0019] 统计单元,用于根据接收到的所述控制模块返回的测试结果生成总结果,并将所述总结果发送至控制单元;其中,所述总结果包括命中结果和命中失败结果,所述命中结果是接收到任意第一个结果为命中的测试结果的情况下生成的;所述命中失败结果是接收到来自各控制模块的测试结果均为未命中的情况下生成的;

[0020] 所述测试结果中携带有以下信息的至少一种:指示命中测试是否命中的状态信息、进行命中测试的数据块的数据块ID、所述缓存行标签;结果为命中的测试结果中还携带有命中缓存行的缓存行ID。

[0021] 在一种可能的实现方式中,所述统计单元,还用于在生成的总结果为所述命中结果的情况下,生成停止信号请求,并控制所述总命中测试单元将所述停止信号请求发广播至还未返回测试结果的各命中测试单元;

[0022] 各所述命中测试单元,用于响应于接收到的所述停止信号请求,停止针对所述第一请求的命中测试。

[0023] 在一种可能的实现方式中,所述控制单元,还用于在所述总结果为表示全部未命中的失败结果的情况下,向所述重构单元发送替换请求;

[0024] 所述重构单元,用于在接收到替换请求的情况下,根据预先构建的管理组织结构从数据块分区中确定出待查数据块,并向管理所述待查数据块的控制模块中的替换单元发送查找请求;

[0025] 所述替换单元,用于响应于所述查找请求,采用PLRU算法查找出所述待查数据块

中的可替换缓存行,并生成替换结果返回所述重构单元;

[0026] 所述重构单元,还用于根据所述替换结果生成替换结果并返回所述控制单元,所述替换结果携带有所述可替换缓存行的缓存行ID、所述可替换缓存行所在的数据块的数据块ID;

[0027] 其中,所述数据块分区包括所述重构单元所在的控制中心模块所分配的所有数据块,所述管理组织结构包括对应于所述数据块分区的二叉树结构和所述二叉树结构中各节点的节点信息。

[0028] 在一种可能的实现方式中,所述控制单元,用于根据接收到的访问类型为静态存储访问的访问申请生成第二请求,并将所述第二请求发送至对应的块控制单元,所述第二请求中携带有地址信息;

[0029] 所述块控制单元,用于在接收到第二请求后,在确定所述第二请求的访问类型为静态存储访问的情况下,控制所述读写单元根据所述地址信息从所述块控制单元所控制的数据块的中读取数据或写入数据。

[0030] 在一种可能的实现方式中,所述装置还包括:

[0031] 数据块复制模块,用于接收来自第一控制模块的复制请求,并根据所述复制请求确定所要复制的待复制数据所在的第二数据块以及控制所述第二数据块的第二控制模块,从所述第二控制模块请求到的待复制数据并写入所述第一控制模块所控制的第一数据块中;

[0032] 其中,所述第一控制模块和所述第二控制模块不同,且所述第一控制模块和所述第二控制模块分别为所述存储装置中已经被分配的数据块对应的控制模块中的任意一个。

[0033] 在一种可能的实现方式中,所述数据块复制模块,包括:

[0034] 请求缓冲器,用于接收来自所述第一控制模块的复制请求并存储;

[0035] 控制器,用于从所述请求缓冲器中获取到当前需要执行的复制请求,将所述复制请求拆分为多个循环请求,依次向根据各所述循环请求确定出的第二控制模块发出数据请求或根据各所述循环请求确定出的第二控制模块所控制的第二数据块对应的上一级缓存发出数据请求;

[0036] 数据缓冲器,用于接收来自所述第二控制模块或上一级缓存的待复制数据,并将所述待复制数据写入所述第一控制模块所控制的第一数据块中。

[0037] 在一种可能的实现方式中,所述节点信息包括:处理状态、左分区计数器和右分区计数器;

[0038] 其中,根据预先构建的管理组织结构从数据块分区中确定出待查数据块,包括:

[0039] 在接收到替换请求的情况下,将数据块分区对应的二叉树结构中的根节点确定为第一个目标节点并获取所述目标节点的节点信息;

[0040] 在所述目标节点为非叶节点的情况下,将当前所述目标节点的处理状态所指示的替换方向上的所述目标节点的子节点确定为下一目标节点以及更新所述目标节点的对应于替换方向的目标计数器,直至确定出的目标节点为叶节点,则基于所述叶节点确定出待查数据块;所述目标计数器为所述目标节点的左分区计数器和右分区计数器中对应于所述目标节点的处理状态所指示的替换方向的计数器,各节点的处理状态所指示的替换方向为节点的左侧分支或右侧分支;

[0041] 所述信息存储单元用于存储所控制数据块中各缓存行的基本信息,所述基本信息包括各缓存行的缓存行标签和缓存行ID。

[0042] 在一种可能的实现方式中,基于所述叶节点确定出待查数据块,包括:

[0043] 将所述叶节点的处理状态所指示的替换方向对应的数据块确定为待查数据块;或者

[0044] 将所述叶节点对应的数据块确定为待查数据块。

[0045] 在一种可能的实现方式中,所述节点信息还包括左分区权重和右分区权重;

[0046] 所述重构单元,还用于在所述目标计数器的数值为预设值的情况下,重置所述目标计数器为初始值以及更改所述目标节点的处理状态所指示的替换方向;

[0047] 其中,所述初始值为对应于所述目标计数器所在分支方向的分区权重、所述预设值为0;或者,所述初始值为0、所述预设值为对应于所述目标计数器所在分支方向的分区权重;左分区计数器所在分支方向的分区权重为左分区权重,右分区计数器所在分支方向的分区权重为右分区权重。

[0048] 在一种可能的实现方式中,所述重构单元,还用于在确定所述目标节点的目标计数器为无效和/或所述目标节点对应于处理状态所指示的替换方向的分区权重为0的情况下,确定所述目标节点为叶节点;

[0049] 其中,所述目标节点对应于处理状态所指示的替换方向的分区权重为左分区权重或右分区权重。

[0050] 在一种可能的实现方式中,所述节点信息还包括左指针和右指针;

[0051] 所述重构单元,还用于根据当前所述目标节点的处理状态所指示的替换方向对应的目标指针查找到下一目标节点;

[0052] 其中,若当前所述目标节点的处理状态所指示的替换方向为左侧分支,则所述目标指针为左指针;若当前所述目标节点的处理状态所指示的替换方向为右侧分支,则所述目标指针为右指针。

[0053] 在一种可能的实现方式中,所述叶节点的节点信息还包括:右地址索引、右有效索引标识、左地址索引和左有效索引标识;

[0054] 其中,将所述叶节点的处理状态所指示的替换方向对应的数据块确定为待查数据块,包括:在所述叶节点的处理状态所指示的替换方向对应的有效索引标识为有效的情况下,将所述叶节点的处理状态所指示的替换方向对应的目标地址索引所指向的数据块确定为待查数据块;

[0055] 其中,若当前所述目标节点的处理状态所指示的替换方向为左侧分支,则所述目标地址索引为所述左地址索引;若当前所述目标节点的处理状态所指示的替换方向为右侧分支,则所述目标地址索引为所述右地址索引。

[0056] 在一种可能的实现方式中,所述叶节点的节点信息还包括左地址索引、左有效索引标识,或者所述叶节点的节点信息还包括右地址索引和右有效索引标识;

[0057] 其中,将所述叶节点对应的数据块确定为待查数据块,包括:在所述叶节点的节点信息中存在的左有效索引标识或右有效索引标识为有效的情况下,将所述叶节点的节点信息中存在的地址索引所指向的数据块确定为待查数据块。

[0058] 在一种可能的实现方式中,各节点的左分区权重、右分区权重与所述节点的左侧

分支中数据块的左侧数量、所述节点的右侧分支中数据块的右侧数量相关；

[0059] 在所述节点的左侧数量和右侧数量均不为零的情况下,所述节点的左分区权重、右分区权重的值,根据节点的左侧数量和右侧数量的最简整数比确定；

[0060] 在所述节点的左侧数量和/或右侧数量为零的情况下,所述节点的左分区权重、右分区权重的值分别设置为节点的左侧数量和右侧数量。

[0061] 在一种可能的实现方式中,所述重构单元,还用于响应于重构请求,根据配置信息设置所述数据块分区中的数据块的块数量以及各所述数据块中缓存行的行数量,并设置各所述数据块的数据块ID、缓存行的缓存行ID;根据所述数据块分区中数据块的块数量,创建对应所述数据块分区的二叉树结构,以及确定所述二叉树结构中各节点的节点信息,完成所述数据块分区的管理组织结构配置。

[0062] 在一种可能的实现方式中,各所述数据块中包括 2^n 个缓存行,n为正整数。

[0063] 根据本公开的另一方面,提供了一种数据处理装置,包括:

[0064] 处理器,所述处理器中设置有位于所述处理器的各核外部的存储装置;

[0065] 所述存储装置包括:多个数据块、多个控制模块、至少一个控制中心模块和管理模块;

[0066] 各所述数据块用于存储数据,各所述控制模块与对应的一个数据块连接并控制该数据块的访问;

[0067] 各控制中心模块与被分配的数据块所连接的控制模块通信连接,以通过实现通信连接的控制模块对被分配的数据块的访问进行控制;

[0068] 所述管理模块,用于接收资源管理请求,在所述资源管理请求为配置请求的情况下,为所述配置请求指示的各第一控制中心模块分配数据块,并根据所述配置请求指示的访问类型对所述第一控制中心模块、被分配的数据块所连接的控制模块的访问控制逻辑进行配置;

[0069] 其中,所述第一控制中心模块为所述至少一个控制中心模块中的部分或全部。

[0070] 通过本公开实施例提供的一种存储装置及数据处理系统,可以实现存储资源的统一管理且控制逻辑统一,无需断电就可以根据不同的资源需求灵活、动态、均衡分配各类型存储空间所使用的数据块,使得各类存储空间都可拥有足量数据块,保证了不同类型的存储空间可以有效利用的同时提升程序的执行速度和效率,且实现了数据块的复用。

[0071] 根据下面参考附图对示例性实施例的详细说明,本公开的其它特征及方面将变得清楚。

附图说明

[0072] 包含在说明书中并且构成说明书的一部分的附图与说明书一起示出了本公开的示例性实施例、特征和方面,并且用于解释本公开的原理。

[0073] 图1示出本公开实施例的一种GPU存储器结构示意图。

[0074] 图2示出本公开实施例的一种存储装置的框图。

[0075] 图3示出本公开实施例的一种数据块对的示意图。

[0076] 图4示出本公开实施例的一种控制中心模块和控制模块的框图。

[0077] 图5示出本公开实施例的一种示例一的管理组织结构的构建示意图。

- [0078] 图6示出本公开实施例的一种示例二数据块分区对应的二叉树结构的示意图。
- [0079] 图7示出本公开实施例的一种替换步骤的流程图。
- [0080] 图8示出本公开实施例的一种存储装置的框图。
- [0081] 图9示出本公开实施例的一种数据复制模块的工作流程示意图。
- [0082] 图10示出本公开实施例的一种工作组列表的示意图。
- [0083] 图11示出本公开实施例的一种数据复制流程图。
- [0084] 图12示出本公开实施例的一种数据复制流程图。

具体实施方式

[0085] 以下将参考附图详细说明本公开的各种示例性实施例、特征和方面。附图中相同的附图标记表示功能相同或相似的元件。尽管在附图中示出了实施例的各种方面,但是除非特别指出,不必按比例绘制附图。

[0086] 在这里专用的词“示例性”意为“用作例子、实施例或说明性”。这里作为“示例性”所说明的任何实施例不必解释为优于或好于其它实施例。

[0087] 另外,为了更好的说明本公开,在下文的具体实施方式中给出了众多的具体细节。本领域技术人员应当理解,没有某些具体细节,本公开同样可以实施。在一些实例中,对于本领域技术人员熟知的方法、手段、元件和电路未作详细描述,以便于凸显本公开的主旨。

[0088] 相关技术中,处理器在被应用于到不同的处理领域执行不同类型的处理任务时,对其缓存的使用存在区别,以下以GPU为例进行说明。

[0089] GPU中本地存储器Local Memory作为工作组(work group)内部的所有线程之间共享数据的空间,通常是GPU中每个核(core)内单独存放,同时按照板块(Bank)进行访问。系数存储器作为插值系数存储的单元,也是通过板块进行访问。

[0090] 指令缓存单元、纹理缓存单元、通用数据缓存均作为离核最近的1级缓存,通常可在1个或多个核旁边进行数据缓存,方便后续使用过程中再次访问到该资源时,快速访问到。1级缓存等缓存单元通常被组织为缓存行(Cache line)形式,访问时通过缓存行地址(line address)进行缓存行的访问,具体缓存行内数据根据偏移地址选取获得。

[0091] 但是,这些数据存储器和缓存均作为特定存储单元专用,数据分开存储。GPU可用于图形渲染和AI等通用计算领域,在实际应用中,GPU的L1数据缓存在应用于图形渲染和通用计算领域(如AI)均会被使用到,但是GPU的本地存储器通常在通用计算领域广泛使用,在图形渲染领域很少使用。如图1所示,该GPU中设置有两个核,且每个核内部分别设置有供该核使用的本地存储器,两个核共同访问同一个L1数据缓存。但如果只是用很少的工作组且使用很小的本地存储器空间甚至不用时,两个本地存储器空间使用率不高,很浪费。同时访问全局内存的数据缓存量比较多时,仅有一块比较小的L1数据缓存,会导致缓存行不够用,缓存行会频繁被替换,使得程序执行速度下降,GPU整体性能也不高。由于已有的GPU中本地存储器、系数存储器、纹理缓存和L1数据缓存等通用数据缓存,全部都是分开进行存储管理的,且大小固定,各自使用一套控制逻辑,控制复杂,验证困难,难以复用。GPU在通用计算领域和图形渲染领域所需使用是数据缓存不同,导致部分资源处于饱和甚至不够用的状态,而另一部分资源空闲无法使用的不均衡现象。

[0092] 为解决上述技术问题,本公开实施例提供了一种存储装置及数据处理系统,可以

实现存储资源的统一管理且控制逻辑统一,无需断电就可以根据不同的资源需求灵活、动态、均衡分配各类型存储空间的所使用的数据块,使得各类存储空间都可拥有足量数据块,保证了不同类型的存储空间可以有效利用的同时提升程序的执行速度和效率,且实现了数据块的复用。

[0093] 如图2所示,该存储装置应用于处理器(图中未示出),且该存储装置可以设置在所述处理器的各核50的外部,该存储装置包括:多个数据块(Data Block,DB) 301、多个控制模块(Data Block Control,DBC) 20、至少一个控制中心模块(Data Block Control Center,DBCC) 10和管理模块(Data Block Management,DBM) 40。

[0094] 各所述数据块301用于存储数据,各所述控制模块20与对应的一个数据块301连接并控制该数据块301的访问。每一个数据块301与控制模块20分别一一对应连接。

[0095] 各控制中心模块10与被分配的数据块301所连接的控制模块20通信连接,以通过实现通信连接的控制模块20对被分配的数据块301的访问进行控制。

[0096] 所述管理模块40,用于接收资源管理请求,在所述资源管理请求为配置请求的情况下,为所述配置请求指示的各第一控制中心模块分配数据块,并根据所述配置请求指示的访问类型对所述第一控制中心模块、被分配的数据块301所连接的控制模块20的访问控制逻辑进行配置。其中,所述第一控制中心模块为所述至少一个控制中心模块10中的部分或全部。

[0097] 在本实施例中,管理模块40若确定当前未被分配的数据块不能满足配置请求对应的资源需求(也即,当前空闲未被分配的数据块的数量少于配置请求指示的需要使用的数据块数量),则管理模块40可以继续等待直至未被分配的数据块数量满足资源需求之后,再继续执行配置请求。

[0098] 在本实施例中,不同访问类型对应的访问控制逻辑不同,满足不同访问类型的控制逻辑需求。访问类型可以包括缓存行访问和静态存储访问。缓存行访问对应的访问控制逻辑为以缓存行形式对数据块301进行访问,静态存储访问对应的访问控制逻辑为以粒度单元的形式对数据块301进行访问。例如,可以以板块为粒度单元,则静态存储访问对应的访问控制逻辑可以为以板块的形式对数据块301进行访问。

[0099] 在本实施例中,每个数据块301可以是一个或多个SRAM,每个数据块301可以具有支持多个板块单次访问的能力,可以减少数据冲突和访问延迟。数据块301还可以支持多个缓存行(cache line)的访问,例如每个数据块301可以包括 2^n (n 为正整数)个缓存行。如图3所示,两个数据块301可以作为一个数据块对(Data Block Pair,DBP)进行分配。其中,图3中一个数据块有128行,每一行有16个板块,每个板块存储1个双字(Double Word,DW)的数据。例如,如下表1所示,配置完成后可以得到如下配置信息。

[0100] 表 1配置信息示例

名称	数值	描述信息
TOTAL_DATA_SRAM_SIZE	192KB	总的存储空间大小
DATA_BLOCK_SIZE	8KB	数据块的大小
DATA_BLOCK_NUMBER	24 个	数据块的数量
DATA_BLOCK_Pair	2 个 DB	数据块对, 每个数据块对包括 2 个数据块
DATA_BLOCK_Pair_NUMBER	12 个	数据块对的数量

[0101]

[0102] 在本实施例中,配置请求可以指示以下信息:需要配置的第一控制中心模块、需要为各第一控制中心模块分配的数据块的数量、各第一控制中心模块对应的访问类型。其中,处理器在运行不同的任务之前,可以根据所需执行任务的任务需求生成配置请求,不同的任务可以属于相同或不同的应用领域(渲染领域或通用计算领域等),针对不同的任务处理器需利用不同的工作组以完成对应任务的执行。任务需求可以指示以下信息:需要设置哪些空间类型的核内存储器、核外缓存;对核内存储器和核外缓存的空间大小需求;访问类型的设置等。这样,处理器可以根据任务需求确定需要配置哪些控制中心模块(也即第一控制中心模块)、为该控制中心模块需要分配多少个数据块以及设置对应的访问类型。

[0103] 在一些实施例中,如图2所示,各所述第一控制中心模块所分配的数据块可以作为核内存储器或核外缓存等空间类型的供处理器使用的存储空间,实现了核内存储器和核外缓存的合并存储管理,一个数据块可以根据需要在不同的时间段分配为不同空间类型的存储空间,实现了数据块的复用。例如,对于任意一个第一控制中心模块所分配的全部数据块可以作为本地存储器、系数存储器、L1数据缓存、纹理缓存、系统级缓存中的任意一种使用。换言之,可以根据实际需要通过对各数据块具体作为何种存储空间进行灵活配置,且可以根据程序等对存储资源使用量的不同动态配置核内存储器、核外缓存的大小(通过设置数据块的数量以及数据块中缓存行的数量来改变大小),使程序等在执行过程中核内存储器、核外缓存资源充足且无闲置、均可以得到有效利用,且可以提升程序的执行速度,使用更为灵活。

[0104] 在一些实施例中,在存储装置的预先设置中,可以预先根据处理器可能会处理的不同任务的任务需求,确定不同空间类型的核内存储器和核外缓存所需要的最大数据块数量,而后根据最大数据块数量设置控制中心模块10所连接的控制模块20的数量,以便于后续管理模块40可以基于配置请求的指示使能第一控制中心模块及其被分配的数据块301所连接的控制模块20,使得第一控制中心模块与被分配的数据块301所连接的控制模块20之间能够实现通信连接。在存储装置的预先设置中,同一个控制模块20可以连接到一个或多个控制中心模块10,但在基于配置请求完成配置之后,同一个控制模块20仅会与一个第一控制中心模块10通信连接。也即,对于执行配置请求完成的任意一次配置中同一个数据块仅会被分配给一个第一控制中心模块进行管控,同一个数据块在执行不同配置请求完成的不同配置中可以分配给不同的第一控制中心模块进行管控。这样,可以适应不同任务的配置需求,灵活分配第一控制中心模块所分配的数据块的数量,保证核内存储器、核外缓存都具有足够的数据块可供使用,实现存储空间的均衡配置。

[0105] 在一种可能的实现方式中,如图4所示,各所述控制中心模块10可以包括控制单元101、统计单元103、重构单元104和总命中测试单元102,各所述控制模块20可以包括命中测试单元201、信息存储单元202、块控制单元204、替换单元203和读写单元205。

[0106] 其中,在所述访问类型为缓存行访问的情况下,所述第一控制中心模块中的各单元和被分配的数据块301所连接的控制模块20中的各单元均被配置为开启。也即,在所述访问类型为缓存行访问的情况下,各第一控制中心模块10中的控制单元101、统计单元103、重构单元104和总命中测试单元102均被配置为开启,被分配的数据块301的控制模块20中的命中测试单元201、信息存储单元202、块控制单元204、替换单元203和读写单元205均被配置为开启。

[0107] 在所述访问类型为静态存储访问的情况下,所述第一控制中心模块中的控制单元101、被分配的数据块301的控制模块20中的块控制单元204和读写单元205被配置为开启。

[0108] 在一种可能的实现方式中,所述管理模块40,还用于在所述资源管理请求为释放请求的情况下,释放所述释放请求指示的各第二控制中心模块及其所分配的数据块;其中,所述第二控制中心模块为所述第一控制中心模块中的部分或全部。被释放的各所述第二控制中心模块、各所述第二控制中心模块所分配的数据块连接的控制模块均被关闭(也即关闭之前开启的控制中心模块中的各单元、关闭控制模块中被开启的各单元),以等待再次分配。

[0109] 通过上述方式,仅在控制中心模块10和控制模块20需要工作使用时开启、被释放停止使用时关闭,可以降低功耗且可以及时增加处理器可分配的数据块数量。而且,根据访问类型对第一控制中心模块10及其通信连接的控制模块20中的单元进行选择性地开启(关闭)也能够降低功耗。且对于完成使用的数据块及时进行释放,可以降低资源的占用,提高资源的有效使用率,实现数据块的复用。

[0110] 在本实施例中,管理模块40中可以记录有各控制中心模块10、控制模块20的使用状态并根据当前各模块的实际使用情况及时更新其使用状态,使用状态包括已使用(表示被标记的模块已经被占用)、未使用(表示被标记的模块处于空闲状态可以配置使用)两种,可以通过不同的标识标记这两种状态,例如,可以对每个控制中心模块10、控制模块20利用1 bit进行使用状态记录,“1”表示已使用、“0”表示未使用。在一些实施例中,管理模块40在接收到配置请求之后,可以根据配置请求指示的访问类型执行以下任意一种操作以实现第一控制中心模块和被分配的数据块的控制模块中的相应单元的开启:向第一控制中心模块10的控制单元101发送启动请求,以使得控制单元101可以基于启动请求开启第一控制中心模块10中需要开启的其他单元、开启通信连接到的控制模块20中需要开启的单元。向第一控制中心模块10的控制单元101及其通信连接的控制模块20中的块控制单元204发送启动请求,以使得控制单元101可以基于启动请求开启第一控制中心模块10中需要开启的单元、块控制单元204可以基于启动请求开启控制模块20中需要开启的单元。向第一控制中心模块10的控制单元101和控制模块20中需要开启的单元发送使能信号(enable)。向第一控制中心模块10和控制模块20中需要开启的单元发送使能信号(enable)、并向第一控制中心模块10和控制模块20需要关闭的单元发送关闭信号(disable)。本领域技术人员可以根据实际需要开启第一控制中心模块10和控制模块20中相应单元的实现方式进行设置,本公开对此不作限制。

[0111] 以下分别对不同资源访问类型下,存储装置的工作过程进行示意性说明。

[0112] 缓存行访问:

[0113] 所述控制单元101根据接收到的访问类型为缓存行访问的申请生成第一请求,并将所述第一请求发送至总命中测试单元102。总命中测试单元102用于将第一请求广播至各所述控制模块20中的块控制单元204。块控制单元204在确定所述第一请求的访问类型为缓存行访问的情况下,将所述第一请求发送至所在控制模块20中的命中测试单元201。各所述命中测试单元201根据接收到的第一请求对所控制的数据块301进行缓存行的命中测试(Hit/MissTest),并将测试结果返回所述统计单元103。统计单元103根据接收到的所述控制模块20返回的测试结果生成总结果,并将所述总结果发送至控制单元101。

[0114] 其中,第一请求可以为针对数据进行读取或写入的访问请求,该第一请求中可以包括指示进行数据读取或写入的缓存行的缓存行标签(Cacheline Tag),例如,第一请求可以为缓存行标签。其中,控制单元101可以接收外部装置的进行数据读取或写入的缓存行访问申请并根据该访问申请生成第一请求,该访问申请中包括缓存行标签或者可以将缓存行标签作为该访问申请。

[0115] 其中,测试结果包括命中(hit)和未命中(miss)两种,测试结果中携带有指示是否命中的状态信息,若状态信息为hit状态即表示命中、状态信息为miss状态即表示未命中。其中,命中(hit)表示在该命中测试单元201所在的控制模块20所控制的数据块301中存在对应于第一请求的缓存行。未命中(miss)表示在该命中测试单元201所在的控制模块20所控制的数据块301中不存在对应于第一请求的缓存行。在一些实施例中,测试结果中还可以携带有命中测试单元201所测试的数据块的数据块ID和/或命中缓存行的缓存行标签。其中,对于为命中(hit)的测试结果,其中还可以携带有被命中的命中缓存行的缓存行ID。

[0116] 其中,所述总结果包括命中结果和命中失败结果,所述命中结果是接收到任意第一个结果为命中的测试结果的情况下生成的,表示某一个命中测试单元201已经找到满足第一请求的命中缓存行(也即当前已经接收到的测试结果中有一个为命中(hit)),命中结果中可以包括命中缓存行所在的数据块的数据块ID、该命中缓存行的缓存行ID和缓存行标签等;所述命中失败结果是接收到来自各控制模块的测试结果均为未命中的情况下生成的,通过各命中测试单元201的测试均未找到满足第一请求的命中缓存行,也即所有测试结果均为未命中(miss)。

[0117] 所述测试结果中携带有以下信息的至少一种:指示命中测试是否命中的状态信息、进行命中测试的数据块的数据块ID(DB id)、所述缓存行标签;结果为命中的测试结果中还携带有命中缓存行的缓存行ID(cacheline id)。

[0118] 这样,由于第一请求被广播到每个命中测试单元201,各命中测试单元201可以同步进行命中测试,相比于相关技术中的命中测试,整体的命中测试的等待时间降低至原来的 $1/x$ (x 为数据块的数量)。则在数据块数量 $x > 1$ 时,本公开所提供的方式能够减少命中测试的等待时间,且 x 越大等待时间所能减少的就越长,等待时间的缩短效果就越明显。

[0119] 在一种可能的实现方式中,信息存储单元202用于存储其所在控制模块20所控制的数据块301中各缓存行的缓存行标签、缓存行ID、有效位(valid),脏数据标记(dirty mask)、上下文标识(context id)等基本信息进行存储管理,信息存储单元202可以为随机访问存储器。各命中测试单元201可以基于信息存储单元202中存储的各缓存行的缓存行标签,进行命中测试。

[0120] 其中,若命中测试单元201的命中测试命中对应于第一请求的命中缓存行,则命中测试单元201向所述统计单元103返回为命中(hit)的测试结果。若命中测试确认真实未命中对应于第一请求的缓存行,则命中测试单元201向统计单元103返回为未命中(miss)的测试结果。

[0121] 其中,统计单元103还用于在接收各测试结果的过程中,在确定接收到的测试结果为命中(hit)的情况下,生成停止信号请求并发送至总命中测试单元102。该停止信号请求携带有被命中的命中缓存行的缓存行标签,该停止信号请求用于指示各命中测试单元201停止针对“停止信号请求中携带的缓存行标签”的命中测试。总命中测试单元102还用于将

停止信号请求广播至并未返回测试结果的各命中测试单元201。各命中测试单元201则根据接收到的停止信号请求停止正在进行的(或者即将进行的)针对“停止信号请求中携带的缓存行标签”的命中测试。

[0122] 在该实现方式中,所述控制单元101,还用于在所述总结果为表示全部未命中的失败结果的情况下,向所述重构单元104发送替换请求。该替换请求用于指示所述重构单元104基于预先配置的针对其管理控制的数据块分区的管理组织结构,寻找到该数据块分区中可以作为替换的缓存行。重构单元104还用于响应于替换请求执行下述“替换步骤”进行替换查找,并将得到的替换结果发送至控制单元101。其中,替换结果可以包括可替换缓存行所在的数据块的数据块ID、该可替换缓存行的缓存行ID等。

[0123] 其中,重构单元104响应于替换请求所查找的数据块分区包括所述重构单元104所在的控制中心模块10所分配的所有数据块301。所述管理组织结构包括对应于所述数据块分区的二叉树结构和所述二叉树结构中各节点的节点信息,所述二叉树结构为满二叉树结构或非满二叉树结构。

[0124] 在本实施例中,各数据块分区对应的二叉树结构中节点的层数和数量与该数据块分区中数据块的块数量相关,二叉树结构中各叶节点最多能够对应该数据块分区中的一个数据块或两个数据块。叶节点最多所能够对应的数据块的数量不同,二叉树结构存在两种不同的实现方式。

[0125] 示例一:若叶节点最多所能够对应的数据块的数量为2。如图5所示,在该示例中数据块分区1包括6个数据块,对应的二叉树有3层、5个节点,其中叶节点有3个,每个叶节点对应两个数据块。每个节点确定有对应该节点的节点信息。

[0126] 示例二:若叶节点最多所能够对应的数据块的数量为1。如图6所示,在该示例中数据块分区1包括6个数据块,对应的二叉树有4层、11个节点,其中叶节点有6个,每个叶节点对应一个数据块。每个节点确定有对应该节点的节点信息。

[0127] 在本实施例中,所述节点信息可以包括:处理状态(Processing Status,S)、左分区权重(Left Partition weight,LW)、左分区计数器(Left Partition counter,LC)、左指针(Left Pointer,LP)、左有效索引标识(Left VldIdx_flag,LVF)、左地址索引(Left AddrIndex,LAI)、右分区权重(Right Partition weight,RW)、右分区计数器(Right Partition counter,RC)、右指针(Right Pointer,RP)、右有效索引标识(Right VldIdx_flag,RVF)、右地址索引(Right AddrIndex,RAI)。

[0128] 其中,处理状态S用于指示当前该对节点左右两侧分支中对应的数据块进行处理的的状态,指示替换方向,各节点的处理状态所指示的替换方向为该节点的左侧分支或右侧分支。在一些实施例中,处理状态可以指示当前可以替换该节点的左侧分支中数据块的缓存行还是右侧分支中数据块的缓存行。可以设置不同的处理状态标识以表示不同的替换方向,例如,可以设置“0”表示处理状态S为替换左侧分支,设置“1”表示处理状态S为替换右侧分支。其中,示例一中各节点的节点信息中均设置有处理状态。示例二中,仅非叶节点的节点信息中设置有处理状态,叶节点的节点信息中不含有处理状态或处理状态位为无效。

[0129] 左分区权重LW用于表示该节点左侧分支的对应权重。右分区权重RW用于表示该节点右侧分支的对应权重。左分区权重LW、右分区权重RW是与该节点左右两侧分支中数据块的数量相关的。左分区权重LW:右分区权重RW=该节点左侧分支中数据块的数量LN:该节点

右侧分支中数据块的数量RN。在一些示例中,可以根据LN/RN的最简整数比确定LW、RW。以图5、图6所示的示例为例,其根节点的 $LN/RN = 4/2 = 2/1$,则根节点的LW、RW可以分别设置为2、1。其中,在示例一中若叶节点的LN和/或RN为0,则节点的LW、RW分别设置为LN、RN。在示例二中叶节点的LW、RW均设置为0、或者也可以设置为其他可以指示该节点为叶节点的特定数值,本公开对此不作限制。

[0130] 左分区计数器LC,用于进行计数。不断根据节点左侧分支中数据块的使用情况对左分区计数器LC的数值进行更新。在一些实施例中,可根据左分区计数器LC表示的含义设置左分区计数器LC的初始值、以及结束计数的预设值。若左分区计数器LC当前的数值与该节点左侧分支中剩余未被替换使用的数据块的比例存在对应关系(也即, $LC/LW = \text{节点左侧未被替换使用的数据块的数量} / \text{节点左侧全部数据块的数量}$),则该左分区计数器LC的初始值等于左分区权重LW、最小值为0(也即预设值为0)。或者,若左分区计数器LC当前的数值与该节点左侧分支中已经被替换使用的数据块的比例存在对应关系(也即, $LC/LW = \text{该节点左侧已经被替换使用的数据块的数量} / \text{该节点左侧全部数据块的数量}$),则该左分区计数器LC的初始值等于0、最大值为左分区权重LW(也即预设值为左分区权重LW)。同理,右分区计数器RC也用于进行计数,其使用方式参见左分区计数器LC,为免冗余不予赘述。

[0131] 其中,示例二中,叶节点的节点信息中并不包括左分区计数器LC和右分区计数器RC,或者,叶节点的节点信息中左分区计数器LC和右分区计数器RC为无效。

[0132] 左地址索引LAI用于指向对应的数据块。左有效索引标识LVF用于表示左地址索引LAI是否有效(例如,1表示有效,0表示无效)。其中,在示例一中,若当前节点的左分区权重LW为0且左有效索引标识LVF为有效时,则该节点为叶节点且该节点的左地址索引LAI指向该叶节点对应的某一个数据块。若当前节点的左分区权重LW为非0和/或左有效索引标识LVF为无效时,则该节点为非叶节点且该节点的左地址索引LAI无效。同理,右地址索引RAI用于指向对应的数据块。右有效索引标识RVF用于表示右地址索引RAI是否有效(例如,1表示有效,0表示无效),其使用方式参见左地址索引LAI、左有效索引标识LVF,为免冗余不予赘述。在示例二中,若当前节点的左分区权重LW为0且左有效索引标识LVF为有效时,则该节点为叶节点且该节点的左地址索引LAI指向该叶节点一一对应的一个数据块。若当前节点的左分区权重LW为非0和/或左有效索引标识LVF为无效时,则该节点为非叶节点且该节点的左地址索引LAI无效。在示例二中,若当前节点的右分区权重RW为0且右有效索引标识RVF为有效时,则该节点为叶节点且该节点的右地址索引RAI指向该叶节点一一对应的一个数据块。若当前节点的右分区权重RW为非0和/或右有效索引标识RVF为无效时,则该节点为非叶节点且该节点的右地址索引RAI无效。其中,在示例二中,叶节点的节点信息仅有“右地址索引RAI、右有效索引标识RVF”或“左有效索引标识LVF、左地址索引LAI”。

[0133] 左指针LP指向该节点左侧的子节点的位置,右指针RP表示该节点右侧的子节点的位置,硬件用硬连线表示,如图5中左指针LP为图中指向左节点箭头表示,如图5中右指针RP为图中指向右节点箭头表示。其中,非叶节点的节点信息中包括左指针LP,叶节点的节点信息中不包括左指针LP、或节点信息中该左指针LP位为无效。非叶节点的节点信息中包括右指针RP,叶节点的节点信息中不包括右指针RP、或节点信息中该右指针RP位为无效。

[0134] 这样,通过上述管理组织结构(二叉树结构及其节点信息)的设置,在进行替换的过程中,由于对应于数据块分区的二叉树结构使得替换中各数据块之间的替换概率相同,

而进一步对于每一个数据块中的缓存行采用PLRU算法进行替换查找能够实现同数据块中缓存行之间的替换概率相同,这样从整体上就可以使得各数据块分区中各缓存行具有相同的替换概率。

[0135] 在本实施例中,重构单元104所执行“替换步骤”可以包括:在接收到替换请求的情况下,将数据块分区对应的二叉树结构中的根节点确定为第一个目标节点并获取所述目标节点的节点信息;在所述目标节点为非叶节点的情况下,将当前所述目标节点的处理状态所指示的替换方向上的目标节点的子节点确定为下一目标节点以及更新目标节点的对应于替换方向的目标计数器,直至确定出的目标节点为叶节点。其中,若确定目标节点为叶节点则基于所述叶节点确定出待查数据块;采用PLRU算法查找出所述待查数据块中的可替换缓存行;根据所述待查数据块的数据块ID和所述可替换缓存行的缓存行ID生成查找结果。

[0136] 而因管理组织结构的差异,替换步骤也存在一定区别,以下结合图7对基于“上述示例一所提供的管理组织结构”的“替换步骤”进行说明。

[0137] 举例来说,如图7所示,重构单元104所执行的“替换步骤”是基于“上述示例一所提供的管理组织结构”进行缓存行替换查找的步骤,“替换步骤”可以包括步骤S201-步骤S213。

[0138] 如图7所示,在步骤S201中,接收替换请求。并将其所控制的数据块分区所对应的二叉树结构中的根节点确定为第一个目标节点。在步骤S202中,在确定出目标节点后,获取到该目标节点的节点信息。

[0139] 如图7所示,在步骤S203中,在得到目标节点的节点信息之后,根据目标节点的节点信息中的处理状态S的指示将目标节点的左侧分支或右侧分支确定为替换方向。例如,若“0”表示左侧分支、“1”表示右侧分支,则在处理状态S为0时,替换方向即为目标节点的左侧分支;在处理状态为1时,替换方向即为目标节点的右侧分支。

[0140] 如图7所示,在步骤S204中,确定出替换方向之后,确定该替换方向对应的权重(例如,若替换方向为目标节点的左侧分支,则替换方向对应的权重即为左分区权重LW)。其中,若替换方向对应的权重为0和/或者对应的目标计数器为无效,则继续执行步骤S205。若替换方向对应的权重不为0且目标计数器有效,则继续执行步骤S209。其中,目标计数器为左分区计数器右分区计数器中与替换方向对应的计数器。

[0141] 如图7所示,在步骤S205中,确定出替换方向对应的有效索引标识(例如,若替换方向为目标节点的左侧分支,则替换方向对应的有效索引标识即为左有效索引标识LVF),并进一步确定替换方向对应的有效索引标识是否为有效。其中,若替换方向对应的有效索引标识为有效,则继续执行步骤S207。若替换方向对应的有效索引标识为无效,则继续执行步骤S206。其中,如图7所示,在步骤S206中,若替换方向对应的有效索引标识为无效,则可以确定目标节点的替换方向上的各数据块均不符合查找条件,丢弃该替换请求、停止查找,并生成替换失败异常并上报异常寄存器,以使得后续可以针对该异常进行处理。

[0142] 如图7所示,在步骤S207中,若替换方向对应的有效索引标识为有效,则可以确定当前该目标节点为叶节点,则将该叶节点的与替换方向对应的右地址索引RAI或左地址索引LAI确定为目标地址索引。例如,若替换方向为目标节点的左侧分支,则替换方向对应的有效索引标识即为左有效索引标识LVF,此时若左有效索引标识LVF有效就可以将左地址索引LAI确定为目标地址索引。

[0143] 如图7所示,在步骤S208中,在确定出目标地址索引之后,重构单元104将所述目标地址索引所指向的数据块,确定为待查数据块,而后向管理所述待查数据块的控制模块20中的替换单元203发出查找请求(S2081)。

[0144] 其中,替换单元203接收到查找请求之后,会基于PLRU算法查找出待查数据块中能够作为替换的缓存行,替换单元203若能够查找到能够作为替换的缓存行,则将该缓存行作为可替换缓存行、并向重构单元104返回查找结果(该查找结果可以指示可替换缓存行的缓存行ID、以及该可替换缓存行所在的数据块的数据块ID),进而使得重构单元104可以基于接收到的查找结果生成替换结果(S2082)。

[0145] 在一些实施例中,叶节点的右地址索引RAI、左地址索引LAI还可以设置为指向对应的数据块的控制模块20,则右地址索引RAI、左地址索引LAI可以设置为对应的数据块的控制模块20的控制模块标识(DBC id),以便于重构单元104基于目标地址索引直接确定出对应的控制模块20并向其中的替换单元203发出查找请求。

[0146] 如图7所示,在步骤S209中,继续确定出替换方向对应的目标计数器(例如,若替换方向为目标节点的左侧分支,则替换方向对应的目标计数器即为左分区计数器LC)当前的数值,若目标计数器当前的数值为预设值(如0),则继续执行步骤S211。若目标计数器当前的数值不是预设值,则执行步骤S210。在一些实施例中,也可以先更新目标计数器的值后,再确定目标计数器当前的数值是否为预设值,若目标计数器当前的数值为预设值(如0),则继续执行步骤S211;若不是预设值则执行步骤S212,本公开对此不作限制。

[0147] 如图7所示,在步骤S210中,更新目标计数器的值,而后执行步骤S212。其中,若目标节点的目标计数器(也即目标节点的左分区计数器LC或右分区计数器RC)的初始值为对该目标节点的左分区权重LW或右分区权重RW、预设值为0,则“更新目标计数器的值”就可以为使得目标计数器的值减1。若目标节点的目标计数器(也即目标节点的左分区计数器LC或右分区计数器RC)的初始值为0、预设值为该目标节点的左分区权重LW或右分区权重RW,则“更新目标计数器的值”就可以为使得目标计数器的值加1。

[0148] 如图7所示,在步骤S211中,重置目标计数器的数值,并更改该目标节点的处理状态S。而后执行步骤S212。其中,若目标节点的目标计数器的初始值为对该目标节点的左分区权重LW或右分区权重RW、预设值为0,则“重置目标计数器的数值”就可以为使得目标计数器的值重新设置为左分区权重LW或右分区权重RW。若目标节点的目标计数器的初始值为0、预设值为该目标节点的左分区权重LW或右分区权重RW,则“重置目标计数器的数值”就可以为使得目标计数器的值重新设置为0。其中,“更改该目标节点的处理状态S”可以为:若目标节点的处理状态S指示替换方向为左侧分支,则将目标节点的处理状态S修改为指示替换方向为右侧分支,例如,将处理状态S由“0”修改为“1”。或者,若目标节点的处理状态S指示替换方向为右侧分支,则将目标节点的处理状态S修改为指示替换方向为左侧分支,例如,将处理状态S由“1”修改为“0”。

[0149] 如图7所示,在步骤S212中,更新所存储的目标节点的节点信息。更新所存储的目标节点的节点信息可以包括:将目标计数器的值从原数值替换为当前数值、更新目标节点的处理状态S;或者,将目标计数器的值从原数值替换为当前数值。以保证所存储的各节点的节点信息始终与数据块的访问情况对应,保证缓存行替换的均衡性。

[0150] 如图7所示,在步骤S213中,确定出指示下一目标节点的目标指针,其中,该目标指

针可以为目标节点的替换方向所对应的指针,为左指针LP或右指针RP。例如,若替换方向为目标节点的左侧分支,则替换方向对应的目标指针即为左指针LP。而后执行步骤S202,进一步根据目标指针确定出下一目标节点的节点信息的存储地址,进而根据存储地址获取到下一目标节点的节点信息。

[0151] 其中,步骤S212和步骤S213的执行顺序可如图7所示,也可以步骤S212在步骤S213之后、或者同步执行,本公开对此不作限制。

[0152] 在本实施例中,若“替换步骤”是基于“上述示例二所提供的管理组织结构”进行缓存行替换查找的步骤,则基于“示例二所提供的管理组织结构”的“替换步骤”与图8所示的替换步骤仅部分步骤存在区别,为简明仅对区别步骤进行说明。其中,因为基于“示例二所提供的管理组织结构”中叶节点与数据块是一一对应的,则基于“示例二所提供的管理组织结构”的“替换步骤”与图8所示的替换步骤的区别在于:

[0153] 在步骤S206之后,在确定目标节点为叶节点的情况下,直接将该目标节点中存储的左地址索引或右地址索引确定为目标地址索引,而后继续执行步骤S208。

[0154] 为简明示意“替换步骤”的实现方式,以下通过简单示例进行说明。其中,处理状态指示的替换方向均为先左后右。

[0155] 假设“替换步骤”是如图5所示的基于“示例一所提供的管理组织结构”或如图6所示的基于“示例二所提供的管理组织结构”进行缓存行替换查找,若根节点的左分区权重和右分区权重分别为4和2、根节点的左子节点的左分区权重和右分区权重均为2,则会按照“数据块DB_a、数据块DB_b、数据块DB_c、数据块DB_d、数据块DB_e、数据块DB_f、数据块DB_a……”这样的循环顺序确定出待查数据块。

[0156] 假设“替换步骤”是如图5所示的基于“示例一所提供的管理组织结构”或如图6所示的基于“示例二所提供的管理组织结构”进行缓存行替换查找,若根节点的左分区权重和右分区权重分别为2和1、根节点的左子节点的左分区权重和右分区权重均为1,则会按照“数据块DB_a、数据块DB_c、数据块DB_e、数据块DB_b、数据块DB_d、数据块DB_f、数据块DB_a……”这样的循环顺序确定出待查数据块。

[0157] 其中,控制单元101,还用于在接收到来自重构单元104的替换结果、或接收到来自所述统计单元103的命中结果的情况下,基于所述替换结果或命中结果生成请求结果发送至发出第一请求的外部装置。以使得外部装置可以基于请求结果进行执行相应操作。

[0158] 在本实施例中,第一控制中心模块的管理模块40在完成第一控制中心模块10的数据块301分配后,在确定已经完成数据块分配且访问类型为缓存行访问的情况下,生成重构请求,以通过重构请求控制重构单元104将所在第一控制中心模块所分配的所有数据块确定为该第一控制中心模块所控制的数据块分区,并执行“初始化配置步骤”完成所述数据块分区的如图5所示的管理组织结构配置。

[0159] 在一些实施例中,重构单元104通过执行“初始化配置步骤”完成所述数据块分区的管理组织结构配置。“初始化配置步骤”包括:响应于重构请求(重构请求还可以是用户发出的;也可以是在检测到满足预设条件后产生的,如检测到程序启动则可以形成重构请求),根据配置信息对数据块分区进行使用配置。该配置信息可以被携带在重构请求中,用于指示以下信息中的部分或全部:数据块分区当前可用的存储量(也可以为当前可用缓存行的数量)、数据块分区中数据块的块数量(Data Block Partition number)、每个数据块

的大小(size)。对数据块分区进行使用配置可以包括:根据各数据块分区对应的块数量,为各数据块分区配置数据块以及确定出各数据块DB的数据块ID(DB id);根据各数据块对应的行数量,为各数据块分配缓存行并确定各缓存行的缓存行ID(cacheline id)和缓存行标签(cacheline tag)。而后根据数据块分区中数据块301的块数量,创建对应各数据块分区的二叉树结构,以及确定二叉树结构中各节点的节点信息,完成数据块分区的管理组织结构配置。而后存储二叉树结构及其各节点的节点信息。在一些实施例中,可以将二叉树结构及其各节点的节点信息存储至对应的存储器中,如对应的缓冲器。

[0160] 在本实施例中,针对第一请求针对数据所需进行的操作(读取或写入)的不同,装置1所执行的存在一定区别,以下对不同情况进行示意性解释。

[0161] 情况一:

[0162] 若外部装置发来的是针对数据进行读取的访问申请,则第一请求还可以指示需要对数据进行的操作是读取。则命中测试单元201在确定出命中缓存行之后,确定信息存储单元202中存储的该命中缓存行的有效位valid和脏数据标记dirty mask。

[0163] 命中测试单元201若基于有效位valid和脏数据标记dirty mask确定该命中缓存行中存储的数据为无效且为非脏数据,则命中测试单元201可以向所述统计单元103返回为命中的测试结果。统计单元103继而可以向控制单元101发送命中结果,以使得控制单元101基于数据块ID和缓存行ID控制对应的控制模块20读取到命中缓存行中的数据,并将该数据作为请求结果返回外部装置。

[0164] 命中测试单元201若基于有效位valid和脏数据标记dirty mask确定该命中缓存行中存储的数据为有效且为脏数据,则可以先将该命中缓存行中存储的数据存储到下一级缓存,而后命中测试单元201向所述统计单元103返回为命中的测试结果。统计单元103继而可以向控制单元101发送命中结果,以使得控制单元101基于数据块ID和缓存行ID控制对应的控制模块20读取到命中缓存行中的数据,并将该数据作为请求结果返回外部装置。

[0165] 在一些实施例中,控制单元101可以将数据块ID和缓存行ID发送至对应的控制模块20的读写单元205,读写单元205中的读端口(read port)会基于数据块ID和缓存行ID读取到数据之后返回给控制单元101。

[0166] 情况二:

[0167] 若外部装置发来的是针对数据进行读取的访问申请,则第一请求还可以指示需要对数据进行的操作是读取。则命中测试单元201在未命中之后,重构单元104查找到可替换缓存行之后,还用于根据该可替换缓存行对应数据块ID从对应的信息存储单元202获取到该可替换缓存行的有效位valid和脏数据标记dirty mask。

[0168] 重构单元104若基于有效位valid和脏数据标记dirty mask确定该可替换缓存行中存储的数据为无效且为非脏数据,则可以向控制单元101返回替换结果(携带有可替换缓存行的所在的数据块的数据块ID和可替换缓存行的缓存行ID)。以使得控制单元101基于数据块ID和缓存行ID控制对应的控制模块20读取到可替换缓存行中的数据,并将该数据作为请求结果返回外部装置。

[0169] 重构单元104若基于有效位valid和脏数据标记dirty mask确定该可替换缓存行中存储的数据为有效且为脏数据,则可以先将该可替换缓存行中存储的数据存储到下一级缓存,而后向控制单元101返回替换结果(携带有可替换缓存行的所在的数据块的数据块ID

和可替换缓存行的缓存行ID)。以使得控制单元101基于数据块ID和缓存行ID读取到可替换缓存行中的数据,并将该数据作为请求结果返回外部装置。

[0170] 在一些实施例中,控制单元101可以将数据块ID和缓存行ID发送至对应的控制模块20的读写单元205,读写单元205中的读端口基于数据块ID和缓存行ID读取到数据之后返回给控制单元101。

[0171] 情况三:

[0172] 若外部装置发来的是针对数据进行写入的访问申请,访问申请中可以携带有需要写入的数据,则第一请求还可以指示需要对数据进行的操作是写入。则命中测试单元201在确定出命中缓存行之后,确定信息存储单元202中存储的该命中缓存行的有效位valid和脏数据标记dirty mask。

[0173] 命中测试单元201若基于有效位valid和脏数据标记dirty mask确定该命中缓存行中存储的数据为无效且为非脏数据,则可以命中测试单元201向所述统计单元103返回为命中的测试结果。统计单元103继而可以向控制单元101发送命中结果,以使得控制单元101基于数据块ID和缓存行ID将数据写入到命中缓存行,并将写入成功作为请求结果返回外部装置。其中,还需要将命中缓存行的valid和dirty mask分别更新为有效、脏数据状态。

[0174] 命中测试单元201若基于有效位valid和脏数据标记dirty mask确定该命中缓存行中存储的数据为有效且为脏数据,则可以先将该命中缓存行中存储的数据存储到下一级缓存,而后命中测试单元201向所述统计单元103返回为命中的测试结果。统计单元103继而可以向控制单元101发送命中结果,以使得控制单元101基于数据块ID和缓存行ID将数据写入到命中缓存行,并将写入成功作为请求结果返回外部装置。

[0175] 在一些实施例中,控制单元101可以将待写入的数据、数据块ID和缓存行ID发送至对应的控制模块20的读写单元205,读写单元205中的写端口(write port)基于数据块ID和缓存行ID将数据写入命中缓存行之后报告控制单元101。

[0176] 情况四:

[0177] 若外部装置发来的是针对数据进行写入的访问申请,访问申请中可以携带有需要写入的数据,则第一请求还可以指示需要对数据进行的操作是写入。则命中测试单元201在未命中之后,重构单元104查找到可替换缓存行之后,还用于根据该可替换缓存行对应数据块ID从对应的信息存储单元202获取到该可替换缓存行的有效位valid和脏数据标记dirty mask。

[0178] 重构单元104若基于有效位valid和脏数据标记dirty mask确定该可替换缓存行中存储的数据为无效且为非脏数据,则可以向控制单元101返回替换结果(携带有可替换缓存行的所在的数据块的数据块ID和可替换缓存行的缓存行ID)。以使得控制单元101基于数据块ID和缓存行ID将数据写入到可替换缓存行,并将写入成功作为请求结果返回外部装置。其中,还需要将命中缓存行的valid和dirty mask分别更新为有效、脏数据状态。

[0179] 重构单元104若基于有效位valid和脏数据标记dirty mask确定该可替换缓存行中存储的数据为有效且为脏数据,则可以先将该可替换缓存行中存储的数据存储到下一级缓存,而后向控制单元101返回替换结果(携带有可替换缓存行的所在的数据块的数据块ID和可替换缓存行的缓存行ID)。以使得控制单元101基于数据块ID和缓存行ID将数据写入到可替换缓存行,并将写入成功作为请求结果返回外部装置。

[0180] 在一些实施例中,控制单元101可以将待写入的数据、数据块ID和缓存行ID发送至对应的控制模块20的读写单元205,读写单元205中的写端口基于数据块ID和缓存行ID将数据写入命中缓存行之后报告控制单元101。

[0181] 在一种可能的实现方式中,控制模块20的读写单元205包括上述写端口和读端口。读写单元205用于在控制单元101的控制下从对应的数据块存储空间(DB SRAM)中进行缓存行ID所对应的缓存行的数据读取、数据写入。

[0182] 在本实施例中,控制单元101在接收到上下文刷新/无效请求(Context flush/invalid)的情况下,将上下文刷新/无效请求广播至其通信连接的各块控制单元204。其中,上下文刷新/无效请求中携带有上下文标识(context id)和请求类型标识,该请求类型标识用于指示操作类型,操作类型包括刷新操作(flush)和无效操作(invalid)。各块控制单元204接收到上下文刷新/无效请求之后,将请求中携带的上下文标识与信息存储单元202中存储的上下文标识进行比对,若存在相同的上下文标识且该上下文标识对应的有效位为有效valid、脏数据标记为脏数据dirty,则将该上下文标识对应的缓存行中的数据存储到下一级缓存中。其中,块控制单元204若进一步确定上下文刷新/无效请求中请求类型标识为无效操作invalid,则将该缓存行的有效位设置为无效invalid;若进一步确定上下文刷新/无效请求中请求类型标识为刷新操作flush,则将不再进行进一步操作确定执行完针对该缓存行的操作,完成上下文刷新/无效请求。

[0183] 静态存储访问:

[0184] 所述控制单元101根据接收到的访问类型为静态存储访问的访问申请生成第二请求,并将所述第二请求发送至对应的块控制单元204,所述第二请求中携带有地址信息。所述块控制单元204在接收到第二请求后,在确定所述第二请求的访问类型为静态存储访问的情况下,控制所述读写单元205根据地址信息从所述块控制单元204所控制的数据块301的中读取数据或写入数据。

[0185] 其中,控制单元101可以接收来自外部装置的进行数据读取或写入的静态存储访问申请并根据该访问申请生成第二请求。静态存储访问申请中可以携带有逻辑地址,控制单元101会将该逻辑地址转换为对应于板块的物理地址,以及计算出对应的控制模块标识DBC id、板块标识(bank id)并作为地址信息,进而形成第二请求并根据控制模块标识DBC id发送至对应的块控制单元204。

[0186] 其中,第二请求还可以指示针对数据所需执行的是读取还是写入操作。块控制单元204在接收到第二请求后将该第二请求添加到读写单元205的读取队列或写入队列,队列可以通过FIFO(First Input First Output)实现。读写单元205在根据仲裁确定需要执行该第二请求后,将板块标识发送至对应的读端口或写端口。读端口在接收到板块标识之后,基于板块标识及其对应的板块地址从对应的板块中读取数据后,块控制单元204将该数据作为请求结果返回外部装置。写端口在接收到板块标识之后,基于板块标识及其对应的板块地址将数据写入对应的板块后,块控制单元204将写入成功作为请求结果返回外部装置。

[0187] 在一种可能的实现方式中,如图8所示,存储装置还可以包括数据块复制模块(Data Block Copy Engine)60。该数据块复制模块60用于接收来自第一控制模块的复制请求,并根据所述复制请求确定所要复制的待复制数据所在的第二数据块以及控制所述第二

数据块的第二控制模块,从所述第二控制模块请求到的待复制数据并写入所述第一控制模块所控制的第一数据块中。所述第一控制模块和所述第二控制模块不同,且所述第一控制模块和所述第二控制模块分别为所述存储装置中已经被分配的数据块对应的控制模块中的任意一个。这样,实现了不同数据块之间数据的直接复制传输,保证了数据的一致性,减少了经过核内部进行数据传输的过程,减轻了核内通用数据寄存器和执行流水线的访问压力。

[0188] 如图9所示,数据块复制模块60可以包括请求缓冲器601、控制器602和数据缓冲器603。如图10所示,数据块复制模块60存储有工作组列表(Group List)和控制模块列表(DBC List)。

[0189] 其中,控制模块列表用于记录每个控制中心模块10所分配的数据块301对应的控制模块20的控制模块标识。

[0190] 工作组列表用于记录当前处理器利用存储装置正在执行的各工作组的工作组信息。每个工作组的工作组信息包括工作组有效信息、上下文标识、锁定计数器、共享有效信息、第一起始地址、第一范围、第二起始地址和第二范围。例如,如9中示出的vldgrp、contextid、lockcounter、vldshared、sharedbase、sharedrange、resbase、resrang依次表示工作组有效信息、上下文标识、锁定计数器、共享有效信息、第一起始地址、第一范围、第二起始地址和第二范围,可以根据实际需要对工作组信息中各信息的顺序进行设置,本公开对此不作限制。

[0191] 工作组有效信息,用于指示所对应的工作组当前是否正在执行,位宽可以为1bit。

[0192] 锁定计数器,用于标记需要传输的数据块计数信息,位宽可以为8bit。

[0193] 共享有效信息,用于标记当前工作组内部共享数据信息是否可被其他工作组拷贝使用,位宽可以为1bit。

[0194] 第一起始地址和第一范围分别用于表示工作组间可共享数据的所在空间的逻辑起始地址及其地址范围,位宽分别为12bit和20bit。第二起始地址和第二范围分别用于表示工作组内可共享数据的所在物理空间的物理起始地址及其地址范围,位宽分别为12bit和20bit。

[0195] 如图9所示,数据块复制模块60可以包括请求缓冲器601、控制器602和数据缓冲器603。

[0196] 请求缓冲器601,用于接收来自第一控制模块的复制请求,并存储该复制请求。其中,第一控制模块为存储装置中已经被分配的数据块对应的多个控制模块20中的任意一个。控制器602,用于从请求缓冲器601获取当前需要执行的复制请求,将所述复制请求拆分为多个循环请求,依次向根据各所述循环请求确定出的第二控制模块发出数据请求或根据各所述循环请求确定出的第二控制模块所控制的第二数据块对应的上一级缓存发出数据请求。

[0197] 数据缓冲器603,用于接收来自所述第二控制模块或上一级缓存的待复制数据,并将所述待复制数据写入所述第一控制模块所控制的第一数据块中。

[0198] 在本实施例中,第二控制模块是存储装置中已经被分配的数据块对应的控制模块20中的一个。其中,第二控制模块和第一控制模块可以如图9所示属于不同的控制中心模块10管控,且第二控制模块和第一控制模块所属的控制中心模块10被分配为不同的存储空

间,例如,第二控制模块所控制的数据块为LI数据缓存、第一控制模块所控制的数据块为本地存储器。第二控制模块和第一控制模块属于不同的控制中心模块10管控,且第二控制模块和第一控制模块所属的控制中心模块10被分配为同一存储空间但处于不同的工作组,例如,第二控制模块、第一控制模块所控制的数据块为本地存储器,但二者属于不同的工作组。

[0199] 其中,控制器602在获取到复制请求之后,其复制过程因第一控制模块和第二控制模块的情况不同,可能存在以下几种可能的复制实现过程:

[0200] 复制可能一:

[0201] 第二控制模块和第一控制模块属于不同的控制中心模块10管控,且第二控制模块和第一控制模块所属的控制中心模块10被分配为同一存储空间但处于不同的工作组,例如,第二控制模块、第一控制模块所控制的数据块为本地存储器,但二者属于不同的工作组。

[0202] 则数据块复制模块60执行复制请求的过程,可以结合图11所示出的数据块复制模块60实现数据复制所执行的步骤S701-步骤S711进行说明。

[0203] 在步骤S701中,请求缓冲器601接收来自第一控制模块的复制请求,并存储该复制请求。

[0204] 在步骤S702中,控制器602从请求缓冲器601中获取到复制请求,并根据复制请求所指示的目的地址范围进行循环拆分,生成多个循环请求。不同循环请求针对目的地址范围内的不同子地址范围中的数据复制。目的地址范围越大,所需复制的数据越多,需要循环的次数就越大,产生的循环请求就越多。而后,控制器602还用于从多个循环请求中确定出之后需要执行的当前循环请求。其中,复制请求可以指示目标地址范围、所需要复制的数据对应的目标工作组的目标工作组信息等复制数据所需的信息,本公开对此不作限制。各循环请求中携带复制请求中信息的部分内容。

[0205] 在步骤S703中,控制器602查找工作组列表,将当前循环请求指示的目标工作组信息与工作组列表中的各工作组信息进行比较,得到比较结果。该比较结果可以指示是否在工作组列表中找到与目标工作组信息一致的工作组信息。可以通过比较目标工作组信息和工作组列表中的各工作组信息的上下文标识进行工作组信息是否一致的比对。

[0206] 在步骤S704中,控制器602根据比较结果确定工作组列表中存在与目标工作组信息一致的工作组信息后,若进一步确定该与目标工作组信息一致的工作组信息的工作组有效信息为有效、锁定计数器有效且共享有效信息有效的情况下,确定第一控制模块所要申请的待复制数据处于被锁定的状态无法释放,可以进行数据复制(否则确定不能进行数据复制),则执行步骤S706,否则,根据比较结果确定工作组列表中不存在与目标工作组信息一致的工作组信息或者确定不能进行数据复制,则执行步骤S705。在步骤S705中,控制器602向第一控制模块返回复制失败报告,以使得第一控制模块可以基于复制失败报告进一步执行其他获取数据的操作。

[0207] 在步骤S706中,控制器602根据当前循环请求对应的子地址范围计算出当前循环请求所要复制的待复制数据的逻辑地址。

[0208] 在步骤S707中,控制器602确定待复制数据的逻辑地址是否超出工作组列表中与目标工作组信息一致的工作组信息中第一起始地址和第一范围所指示的边界范围。若确定

待复制数据的逻辑地址超过边界范围,则可以确定该当前循环请求所要复制的数据不能成功实现,继续执行步骤S702确定出新的当前循环请求后执行步骤S703。若确定逻辑地址没有超过边界范围,则继续执行步骤S708。

[0209] 在步骤S708中,控制器602将逻辑地址转换为物理地址。

[0210] 在步骤S709中,控制器602基于物理地址进一步确定该物理地址对应于控制模块列表中的哪一个控制模块标识,并将对应的控制模块标识所指示的控制模块确定为第二控制模块,以及进一步确定该物理地址对应的板块标识或缓存行ID。而后控制器602向第二控制模块发送第一数据请求,并发送完第一数据请求之后继续执行步骤S702确定出新的当前循环请求后执行步骤S703,直至所有循环请求均执行完毕。

[0211] 其中,控制器602可以直接将第一数据请求发送至第二控制模块中的读写单元205中,第二控制模块的读写单元205在接收到第一数据请求之后,会将第一数据请求中携带的板块标识或缓存行ID中的待复制数据返回给数据缓冲器603。数据缓冲器603在接收到第二控制模块返回的待复制数据之后存储该待复制数据(S710),并在确定满足条件的情况下将存储的待复制数据写入第一控制模块所控制的第一数据块中的指定位置(S711)。其中,满足条件可以是指数据缓冲器603已满无法再继续进行数据存储、或者已经完成针对复制请求的所有待复制数据的存储。

[0212] 复制可能二:

[0213] 第二控制模块和第一控制模块可以属于不同的控制中心模块10管控,且第二控制模块和第一控制模块所属的控制中心模块10被分配为不同的存储空间。例如,第二控制模块所控制的第二数据块为LI数据缓存、第一控制模块所控制的第一数据块为本地存储器。则数据块复制模块60执行复制请求的过程如图12所示包括步骤S701-步骤S704、步骤S705'、步骤S706-步骤S709、步骤S710'和步骤S711。其中,图12所示的“复制可能二”与图11所示“复制可能一”的区别在于,在“复制可能二”中步骤S705'和步骤S710'与“复制可能一”中步骤S705和步骤S710不同,以下仅对“复制可能二”与“复制可能一”的区别部分进行描述,其余过程参见上述“复制可能一”的相关描述。

[0214] 则在步骤S705'中,控制器602在步骤S704中,根据比较结果确定工作组列表中不存在与目标工作组信息一致的工作组信息或者确定不能进行数据复制,则进一步从下一级缓存(如第二控制模块所控制的第二数据块为LI数据缓存,则下一级缓存即为二级数据缓存(L2 cache))中请求待复制数据,该过程包括两种实现方式,控制器602根据待复制数据后续是否会被其他控制模块申请复制,确定该过程采用以下模式一(No Cachable)或模式二(Cachable)从下一级缓存中请求待复制数据。其中,若该待复制数据后续不会被其他控制模块申请复制,则采用以下模式一从下一级缓存中请求待复制数据;若该待复制数据后续会被其他控制模块申请复制,则采用以下模式二从下一级缓存中请求待复制数据。

[0215] 模式一:

[0216] 控制器602在根据比较结果确定工作组列表中不存在与目标工作组信息一致的工作组信息或者确定不能进行数据复制之后,在步骤S705'中进一步根据循环请求中的子地址范围确定出该待复制数据在下一级缓存中的复制地址,而后向该下一级缓存发送第二数据请求以及第二控制模块所在的控制中心模块10中的重构单元104发出替换请求。

[0217] 则,重构单元104接收到替换请求之后进行替换查找,确定出可替换缓存行,形成

查找结果,其中,重构单元104响应替换请求生成查找结果的过程可参考上文所述的“替换步骤”,为免冗余不予赘述。

[0218] 下一级缓存接收到第二数据请求之后,将第二数据请求中指示的复制地址对应的存储空间中存储的待复制数据发送至第二控制模块的读写单元205中,以使得该读写单元205将待复制数据写入重构单元104确定出的可替换缓存行中。

[0219] 下一级缓存接收到第二数据请求之后,还用于将第二数据请求中指示的复制地址对应的存储空间中存储的待复制数据发送至数据缓冲器603,数据缓冲器603存储该待复制数据(S710')。

[0220] 模式二:

[0221] 控制器602在根据比较结果确定工作组列表中不存在与目标工作组信息一致的工作组信息或者确定不能进行数据复制之后,在步骤S705'中进一步根据循环请求中的子地址范围确定出该待复制数据在下一级缓存中的复制地址,而后向该下一级缓存发送第三数据请求。该下一级缓存接收到第三数据请求之后,将第三数据请求中指示的复制地址对应的存储空间中存储的待复制数据发送至数据缓冲器603,数据缓冲器603存储该待复制数据(S710')。

[0222] 本公开还提供一种数据处理系统,包括:处理器,所述处理器中设置有位于所述处理器的各核外部的上述存储装置。

[0223] 需要说明的是,尽管以上述实施例作为示例介绍了存储装置及数据处理装置如上,但本领域技术人员能够理解,本公开应不限于此。事实上,用户完全可根据个人喜好和/或实际应用场景灵活设定各模块、单元,只要符合本公开的技术方案即可。

[0224] 本公开实施例还提出一种计算机可读存储介质,其上存储有计算机程序指令,所述计算机程序指令被处理器执行时实现上述方法。计算机可读存储介质可以是易失性或非易失性计算机可读存储介质。

[0225] 本公开实施例还提出一种电子设备,包括:处理器;用于存储处理器可执行指令的存储器;其中,所述处理器被配置为在执行所述存储器存储的指令时,实现上述方法。

[0226] 本公开实施例还提供了一种计算机程序产品,包括计算机可读代码,或者承载有计算机可读代码的非易失性计算机可读存储介质,当所述计算机可读代码在电子设备的处理器中运行时,所述电子设备中的处理器执行上述方法。

[0227] 本公开可以是系统、方法和/或计算机程序产品。计算机程序产品可以包括计算机可读存储介质,其上载有用于使处理器实现本公开的各个方面的计算机可读程序指令。

[0228] 计算机可读存储介质可以是保持和存储由指令执行设备使用的指令的有形设备。计算机可读存储介质例如可以是一—但不限于——电存储设备、磁存储设备、光存储设备、电磁存储设备、半导体存储设备或者上述的任意合适的组合。计算机可读存储介质的更具体的例子(非穷举的列表)包括:便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、静态随机存取存储器(SRAM)、便携式压缩盘只读存储器(CD-ROM)、数字多功能盘(DVD)、记忆棒、软盘、机械编码设备、例如其上存储有指令的打孔卡或凹槽内凸起结构、以及上述的任意合适的组合。这里所使用的计算机可读存储介质不被解释为瞬时信号本身,诸如无线电波或者其他自由传播的电磁波、通过波导或其他传输媒介传播的电磁波(例如,通过光纤电缆的光脉冲)、或者通过电线传输

的电信号。

[0229] 这里所描述的计算机可读程序指令可以从计算机可读存储介质下载到各个计算/处理设备,或者通过网络、例如因特网、局域网、广域网和/或无线网下载到外部计算机或外部存储设备。网络可以包括铜传输电缆、光纤传输、无线传输、路由器、防火墙、交换机、网关计算机和/或边缘服务器。每个计算/处理设备中的网络适配卡或者网络接口从网络接收计算机可读程序指令,并转发该计算机可读程序指令,以供存储在各个计算/处理设备中的计算机可读存储介质中。

[0230] 用于执行本公开操作的计算机程序指令可以是汇编指令、指令集架构(ISA)指令、机器指令、机器相关指令、微代码、固件指令、状态设置数据、或者以一种或多种编程语言的任意组合编写的源代码或目标代码,所述编程语言包括面向对象的编程语言—诸如 Smalltalk、C++等,以及常规的过程式编程语言—诸如“C”语言或类似的编程语言。计算机可读程序指令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)。在一些实施例中,通过利用计算机可读程序指令的状态信息来个性化定制电子电路,例如可编程逻辑电路、现场可编程门阵列(FPGA)或可编程逻辑阵列(PLA),该电子电路可以执行计算机可读程序指令,从而实现本公开的各个方面。

[0231] 这里参照根据本公开实施例的方法、装置(系统)和计算机程序产品的流程图和/或框图描述了本公开的各个方面。应当理解,流程图和/或框图的每个方框以及流程图和/或框图中各方框的组合,都可以由计算机可读程序指令实现。

[0232] 这些计算机可读程序指令可以提供给通用计算机、专用计算机或其它可编程数据处理装置的处理器,从而生产出一种机器,使得这些指令在通过计算机或其它可编程数据处理装置的处理器执行时,产生了实现流程图和/或框图中的一个或多个方框中规定的功能/动作的装置。也可以把这些计算机可读程序指令存储在计算机可读存储介质中,这些指令使得计算机、可编程数据处理装置和/或其他设备以特定方式工作,从而,存储有指令的计算机可读介质则包括一个制品,其包括实现流程图和/或框图中的一个或多个方框中规定的功能/动作的各个方面的指令。

[0233] 也可以把计算机可读程序指令加载到计算机、其它可编程数据处理装置、或其它设备上,使得在计算机、其它可编程数据处理装置或其它设备上执行一系列操作步骤,以产生计算机实现的过程,从而使得在计算机、其它可编程数据处理装置、或其它设备上执行的指令实现流程图和/或框图中的一个或多个方框中规定的功能/动作。

[0234] 附图中的流程图和框图显示了根据本公开的多个实施例的系统、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、程序段或指令的一部分,所述模块、程序段或指令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。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,两个连续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。也要注意的,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动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。

[0235] 以上已经描述了本公开的各实施例,上述说明是示例性的,并非穷尽性的,并且也不限于所披露的各实施例。在不偏离所说明的各实施例的范围和精神的情况下,对于本技术领域的普通技术人员来说许多修改和变更都是显而易见的。本文中所用术语的选择,旨在最好地解释各实施例的原理、实际应用或对市场中的技术改进,或者使本技术领域的其它普通技术人员能理解本文披露的各实施例。

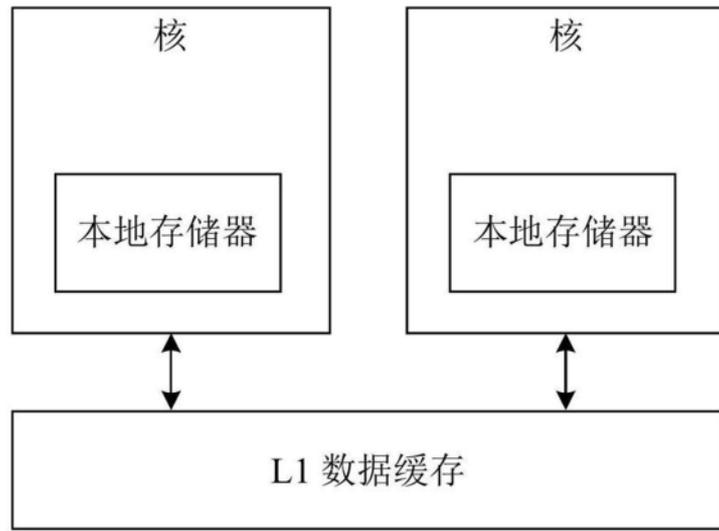


图 1

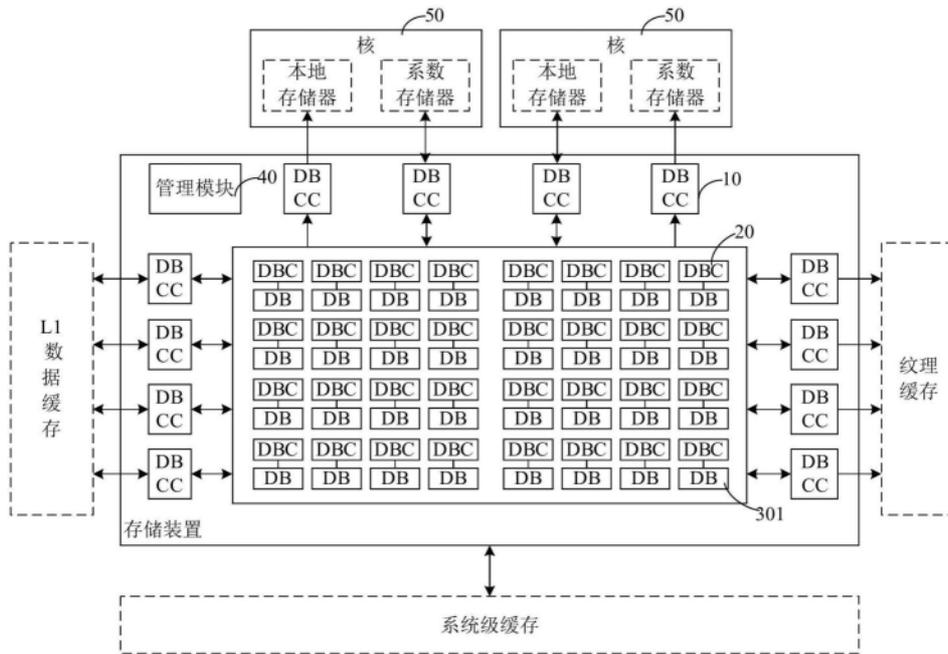


图 2

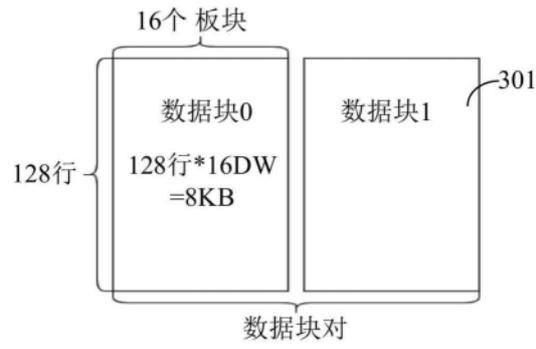


图 3

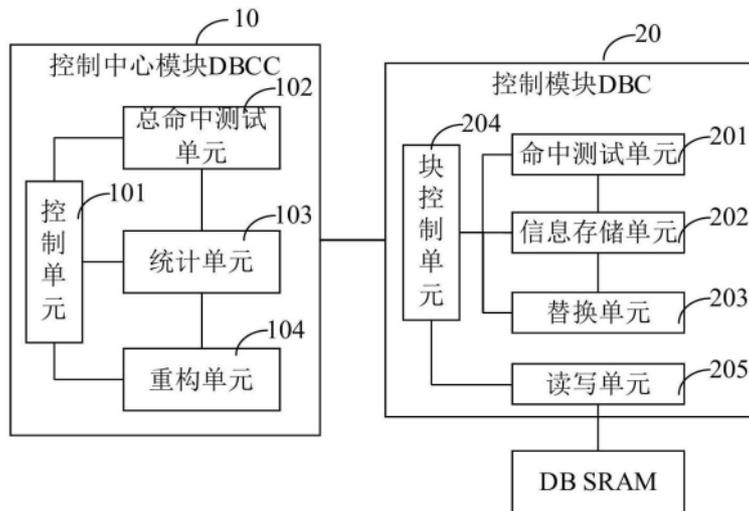


图 4

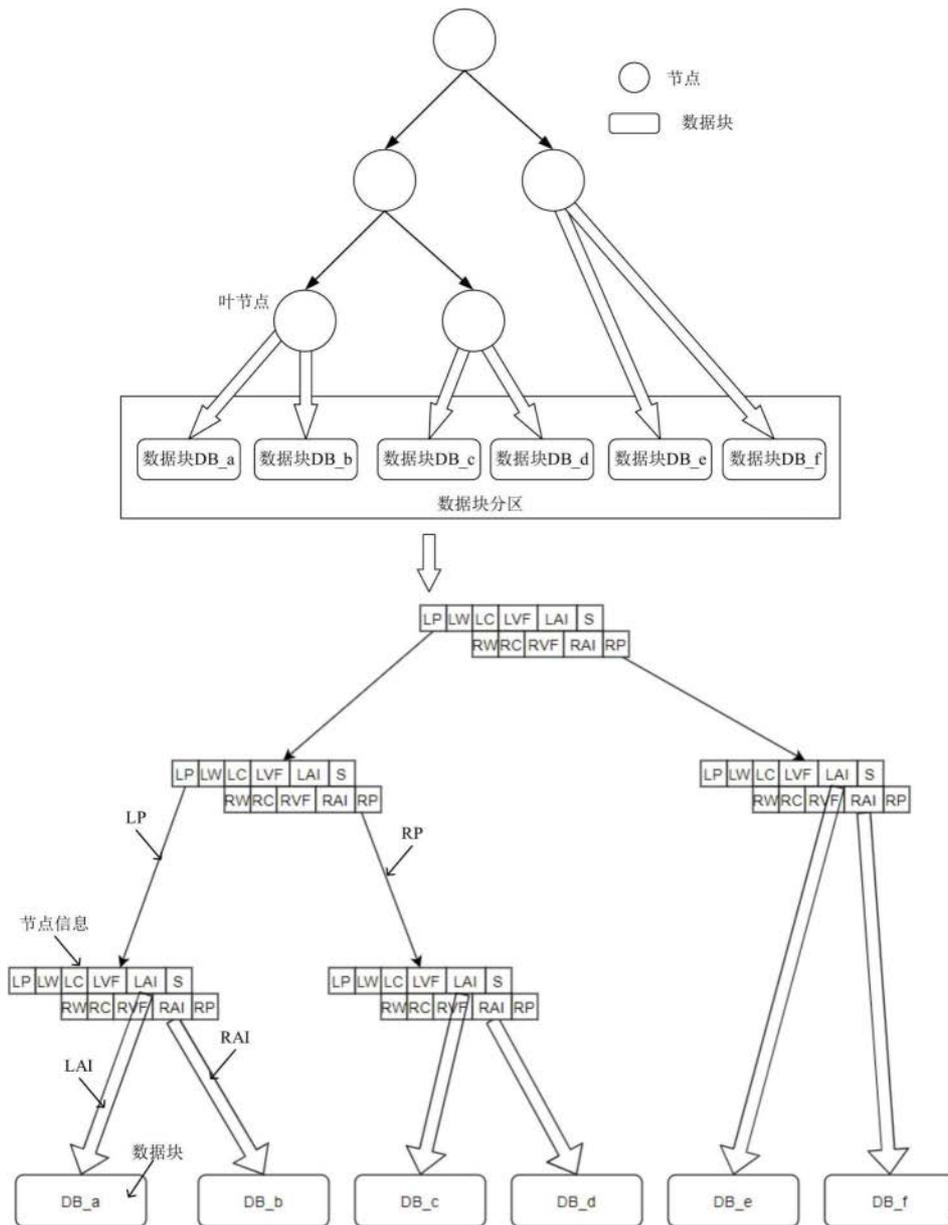


图 5

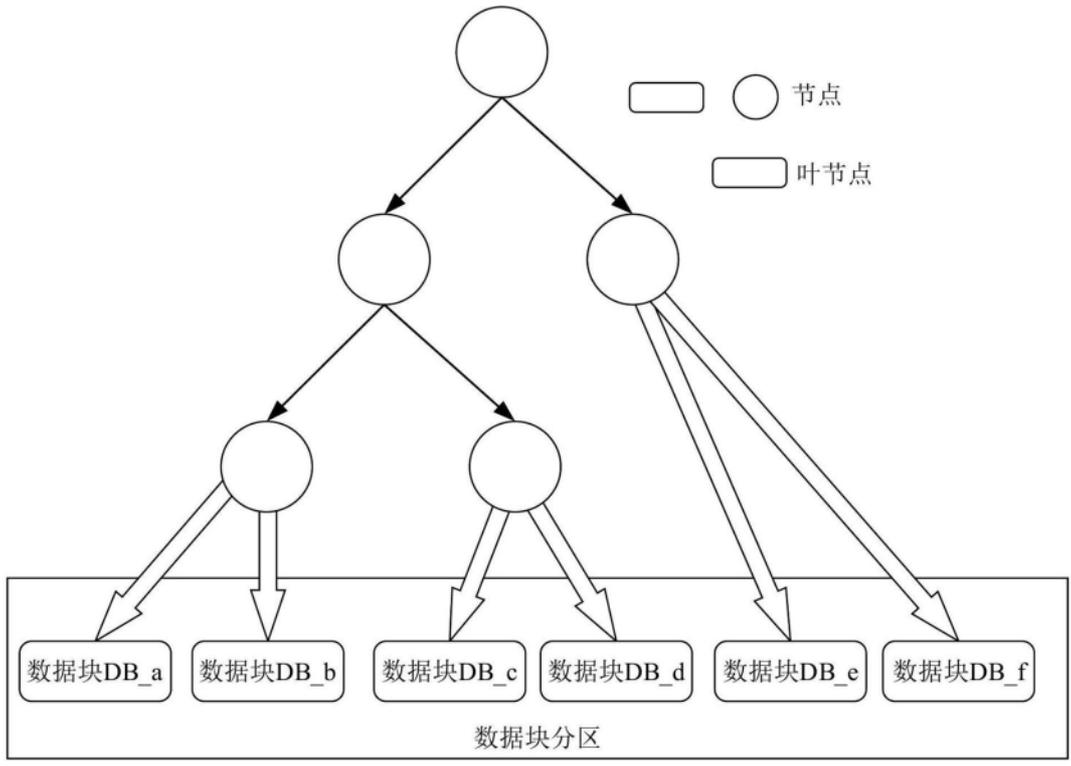


图 6

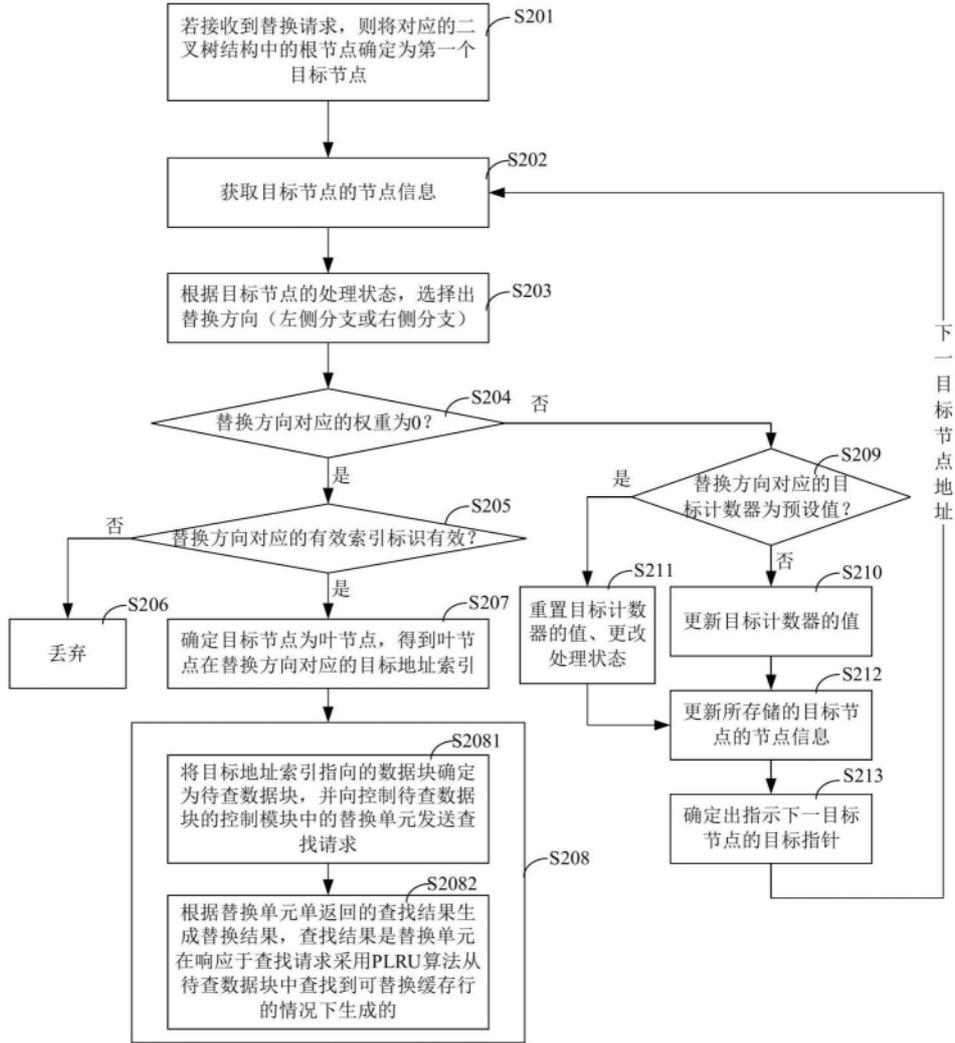


图 7

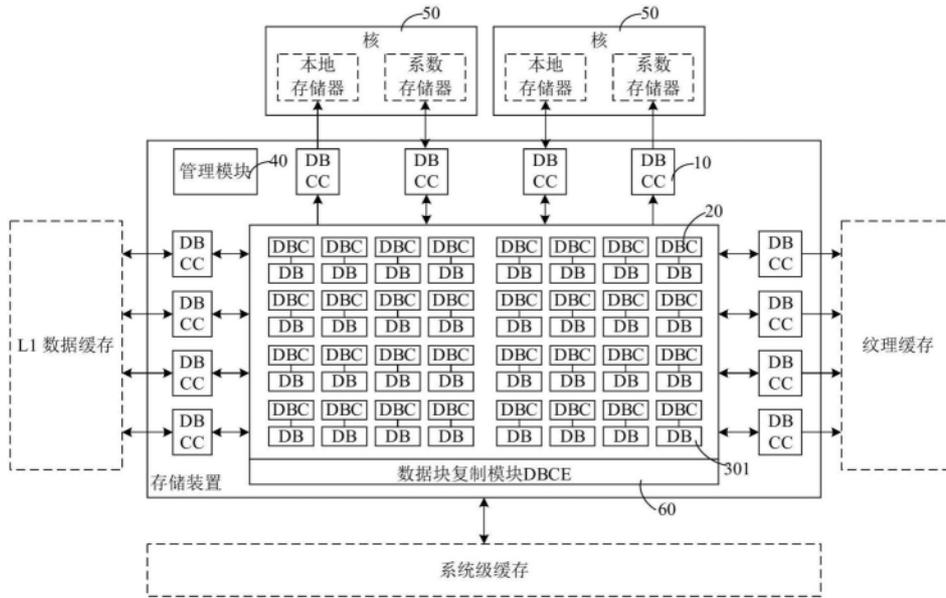


图 8

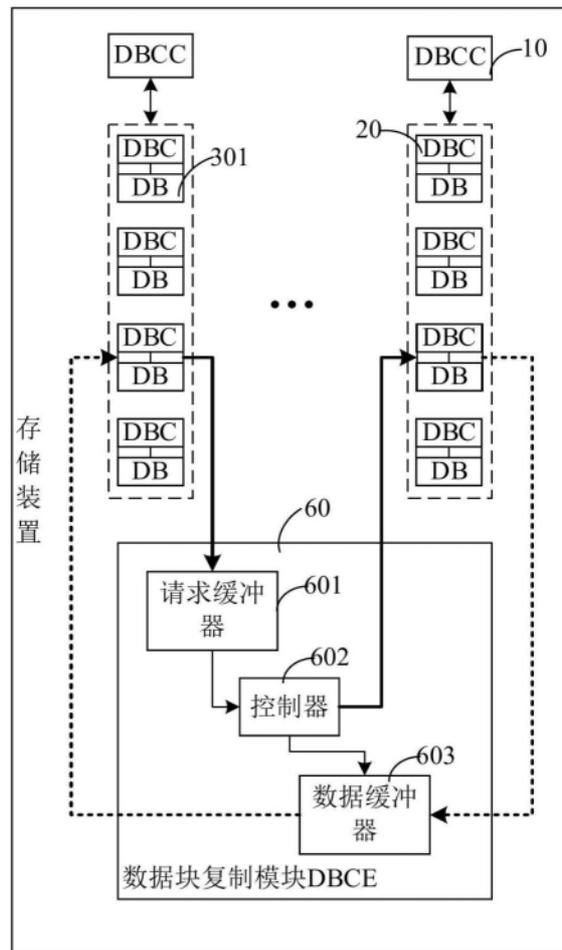


图 9

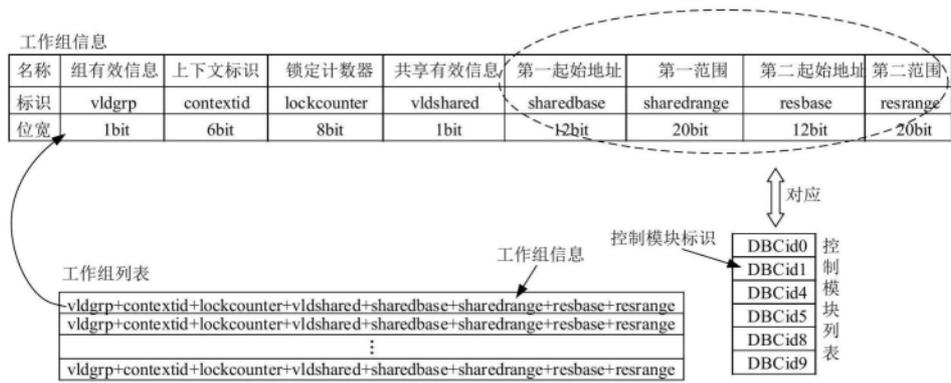


图 10

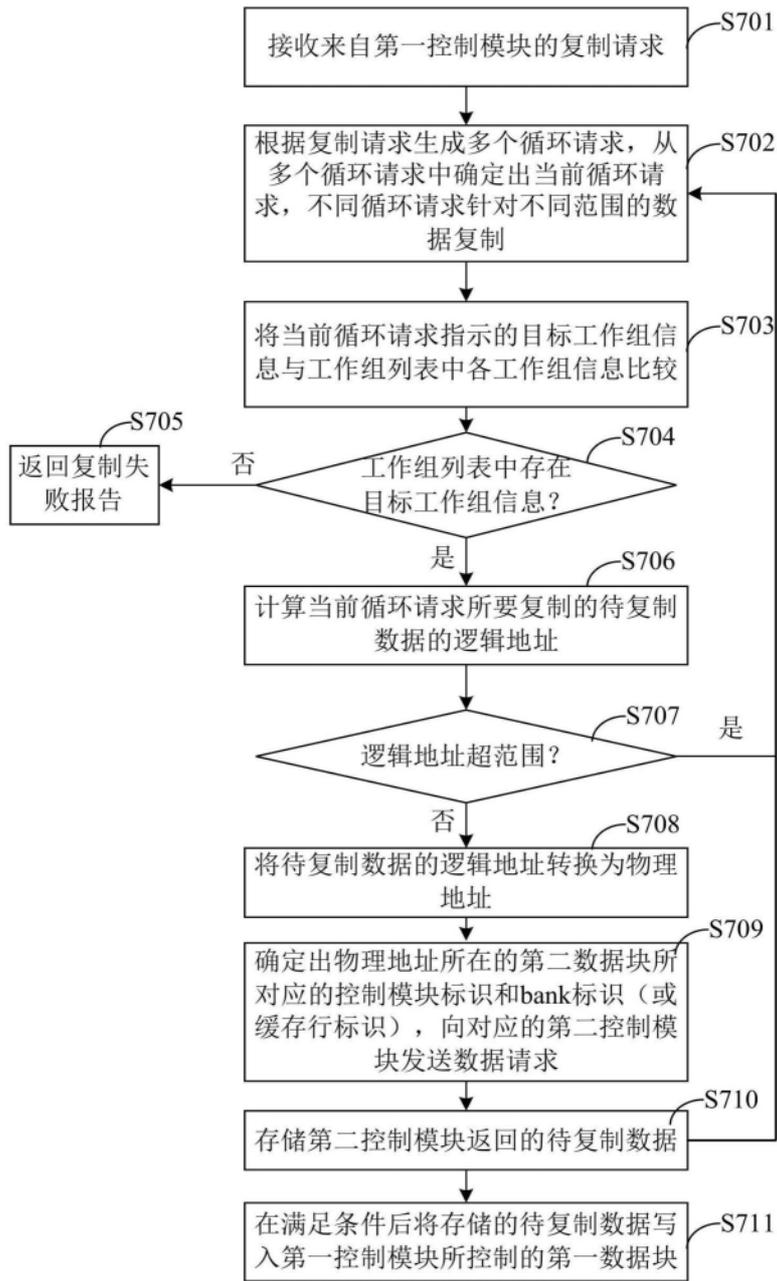


图 11

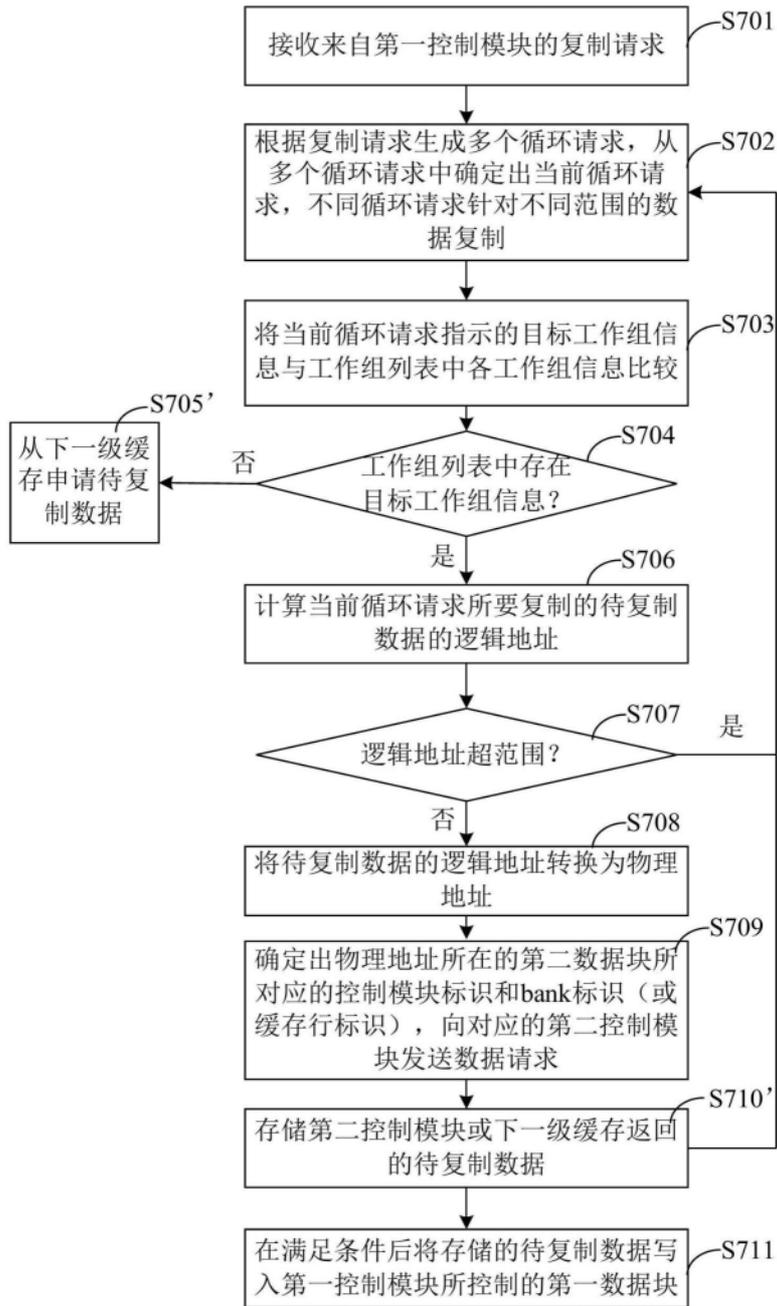


图 12