(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 113050874 A
(43)申请公布日 2021.06.29

(21)申请号 201911369136.9

(22)申请日 2019.12.26

(71)申请人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72)发明人 严雪过 冯宇波 谭海波 陈晓雨 董伟伟

(74)专利代理机构 北京同达信恒知识产权代理有限公司 11291
代理人 黄冠雄

(51)Int.Cl.
G06F 3/06(2006.01)

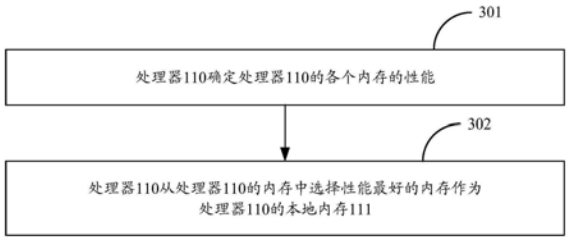
权利要求书2页 说明书14页 附图6页

(54)发明名称

一种内存设置方法以及装置

(57)摘要

一种内存设置方法以及装置，用以在内存混插多种不同性能的内存的情况下，为节点分配本地内存。本申请中，处理器根据处理器的内存的性能设置本地内存及远端内存，将性能较好的内存设置为本地内存，可以使得处理器能够优先访问性能优的内存，提高处理器从本地内存读写数据的效率，提高整个系统的性能。



CN 113050874 A

1.一种内存设置方法,其特征在于,该方法由非统一内存访问架构NUMA系统中的处理器执行,所述处理器包括至少两个内存,所述方法包括:

当所述处理器启动时,获取所述至少两个内存的性能;

根据所述至少两个内存的性能设置所述至少两个内存中的至少一个为本地内存,设置所述至少两个内存中的至少一个为远端内存,其中,所述本地内存的性能优于所述远端内存的性能。

2.如权利要求1所述的方法,其特征在于,所述方法还包括:

将所述远端内存中数据读写频率不低于第一预设值的数据迁移至所述本地内存。

3.如权利要求2所述的方法,其特征在于,所述方法还包括:

确定需要存储在所述本地内存中的内存页的数量N,所述需要存储在所述本地内存中的内存页的数量N为所述内存中数据读写频率由高至低排列的内存页中的前N个;

确定第N个内存页的数据读写频率为所述第一预设值。

4.如权利要求2所述的方法,其特征在于,所述方法还包括:

根据所述内存中内存页的数据读写频率对所述内存中的内存页划分优先级,每个优先级对应一个数据读写频率范围,不同优先级对应的数据读写频率范围不同;

确定需要存储在所述本地内存中的内存页的数量N,所述需要存储在所述本地内存中的内存页的数量N为所述内存中优先级由高至低排列的内存页中的前N个;

确定第N个内存页的数据读写频率为所述第一预设值。

5.如权利要求3或4所述的方法,其特征在于,所述确定需要存储在所述本地内存中的内存页的数量N,包括:

分别确定所述本地内存及所述远端内存中数据读写频率大于第二预设值的内存页;

确定所述本地内存中数据读写频率大于所述第二预设值的内存页占所述内存中数据读写频率大于第二预设值的内存页的比例;

用所述比例乘以所述内存中被使用的内存页的总数量得到所述需要存储在所述本地内存中的内存页的数量N。

6.如权利要求1～5任一所述的方法,其特征在于,所述本地内存和所述远端内存均为动态随机存取存储器DRAM。

7.如权利要求1～5任一所述的方法,其特征在于,所述本地内存为DRAM,所述远端内存为非DRAM存储器。

8.一种内存设置装置,其特征在于,该装置用于设置NUMA系统中的处理器包括的至少两个内存,所述装置包括:

获取模块,用于当所述处理器启动时,获取所述至少两个内存的性能;

设置模块,用于根据所述至少两个内存的性能设置所述至少两个内存中的至少一个为本地内存,设置所述至少两个内存中的至少一个为远端内存,其中,所述本地内存的性能优于所述远端内存的性能。

9.如权利要求8所述的装置,其特征在于,所述装置还包括迁移模块:

所述迁移模块,用于将所述远端内存中数据读写频率不低于第一预设值的数据迁移至所述本地内存。

10.如权利要求9所述的装置,其特征在于,所述装置还包括确定模块:

所述确定模块，用于确定需要存储在所述本地内存中的内存页的数量N，所述需要存储在所述本地内存中的内存页的数量N为所述内存中数据读写频率由高至低排列的内存页中的前N个；确定第N个内存页的数据读写频率为所述第一预设值。

11.如权利要求9所述的装置，其特征在于，所述确定模块，还用于：

根据所述内存中内存页的数据读写频率对所述内存中的内存页划分优先级，每个优先级对应一个数据读写频率范围，不同优先级对应的数据读写频率范围不同；

确定需要存储在所述本地内存中的内存页的数量N，所述需要存储在所述本地内存中的内存页的数量N为所述内存中优先级由高至低排列的内存页中的前N个；

确定第N个内存页的数据读写频率为所述第一预设值。

12.如权利要求10或11所述的装置，其特征在于，所述确定模块在确定需要存储在所述本地内存中的内存页的数量N，具体用于：

分别确定所述本地内存及所述远端内存中数据读写频率大于第二预设值的内存页；

确定所述本地内存中数据读写频率大于所述第二预设值的内存页占所述内存中数据读写频率大于第二预设值的内存页的比例；

用所述比例乘以所述内存中被使用的内存页的总数量得到所述需要存储在所述本地内存中的内存页的数量N。

13.如权利要求8～12任一所述的装置，其特征在于，所述本地内存和所述远端内存均为动态随机存取存储器DRAM。

14.如权利要求8～12任一所述的装置，其特征在于，所述本地内存为DRAM，所述远端内存为非DRAM存储器。

一种内存设置方法以及装置

技术领域

[0001] 本申请涉及存储技术领域,尤其涉及一种内存设置方法以及装置。

背景技术

[0002] 非统一内存访问架构(non-uniform memory access architecture,NUMA)是一种多处理器的计算机架构。对于具有NUMA结构的计算设备,每个处理器配备有内存,除了访问自身配备的内存,每个处理器还可以访问其他处理器的内存。在计算设备启动时,会按照计算设备内的内存离处理器的距离,将离处理器最近的内存(也就是为处理器配备的内存)设置为本地内存,而将离处理器比较远的内存(例如其他处理器的内存)设置为远端内存。在现有的NUMA中,由于本地内存距离处理器比较近,访问速度快,所以会将本地内存设置为优先访问的内存,以提升数据的访问速率。

[0003] 但是在计算设备中包括不同性能的内存时,如果性能比较差但离处理器近的内存被设置为本地内存,则可能不会提升处理器的访问速率。

发明内容

[0004] 本申请提供一种内存设置方法以及装置,用以在内存混插多种不同性能的内存的情况下,为节点分配本地内存。

[0005] 第一方面,本申请提供了一种内存设置方法,该方法由NUMA系统中的处理器执行,所述处理器包括至少两个内存,方法包括:当处理器启动时,处理器可以先获取至少两个内存的性能,例如处理器可以读取SPD所检测的信息获取该至少两个内存的性能。之后,处理器依据该至少两个内存的性能设置本地内存和远端内存,本地内存的性能可以优于远端内存的性能。例如,处理器可以将该至少两个内存中选择性能最好的至少一个内存设置为本地内存,将该至少两个内存的剩余内存设置为本地内存。

[0006] 通过上述方法,处理器根据处理器的内存的性能设置本地内存及远端内存,将性能较好的内存设置为本地内存,可以使得处理器能够优先访问性能优的内存,提高处理器从本地内存读写数据的效率,提高整个系统的性能。

[0007] 在一种可能的实现方式中,在设置了本地内存和远端内存后,处理器还可以进行数据迁移,处理器可以从远端内存中将数据读写频率最高的数据迁移到本地内存。例如,处理器可以将远端内存中数据读写频率高于第一预设值(如第一预设值为本申请实施例中的目标数据读写频率)的所有数据迁移至本地内存。处理器还可以将数据读写频率等于第一预设值的部分数据迁移至本地内存。

[0008] 通过上述方法,数据读写频率最高的数据保存在本地内存,处理器能够较为高效的本地内存中获取这些数据。

[0009] 在一种可能的实现方式中,第一预设值可以是经验值,也可以是处理器根据处理器的内存中各个内存页的数据读写频率确定的。

[0010] 例如,处理器可以确定处理器的至少两个内存中数据读写频率由高至低排列的内

存页前N个内存页为需要存储在本地内存中的内存页，第N个内存页的数据读写频率可以作为第一预设值。

[0011] 又例如，处理器可以根据内存中内存页的数据读写频率对内存中的内存页划分优先级，每个优先级对应一个数据读写频率范围，不同优先级对应的数据读写频率范围不同；将内存中优先级由高至低排列的内存页中的前N个内存页作为确定需要存储在本地内存中的内存页，第N个内存页的数据读写频率为第一预设值。

[0012] 通过上述方法，第一预设值的设置方式较为灵活，而根据处理器的内存中各个内存页的数据读写频率确定的第一预设值更加准确，便于后续可以将远端内存中数据读写频率排序最靠前的一些数据迁移到本地内存中。

[0013] 在一种可能的实现方式中，处理器还可以确定需要存储在本地内存中的内存页的数量N，确定方式如下：处理器可以分别确定本地内存及远端内存中数据读写频率大于第二预设值(如第二预设值为本申请实施例中的阈值)的内存页；之后，确定本地内存中数据读写频率大于第二预设值的内存页占内存中数据读写频率大于第二预设值的内存页的比例；该比例与内存中被使用的内存页的总数量的乘积可以作为数量N。

[0014] 通过上述方法，根据该比例与内存中被使用的内存页的总数量的乘积确定的数量N为本地内存当前所允许存储的数据读写频率最高的内存页的数量，是一种上限，根据该数量N进行的数据迁移后，可以保证本地内存和远端内存中数据读写频率大于第二预设值的内存页的分布比例不变，但本地内存中所存储数据读写频率大于第二预设值的内存页为处理器的内存中数据读写频率由高至低排列的内存页前N个内存页，最终达到本地内存中存储有数据读写频率最高的N个内存页的效果。

[0015] 在一种可能的实现方式中，本地内存和远端内存均为DRAM。

[0016] 通过上述方法，当处理器的内存中存在多种不同性能DRAM时，可以依据性能设置本地内存和远端内存，提升处理器的访问速率。

[0017] 在一种可能的实现方式中，本地内存为DRAM，远端内存为非DRAM存储器。

[0018] 通过上述方法，当处理器的内存中除了DRAM之外，还有其他类型的内存，可以选取性能较高的DRAM作为本地内存，保证处理器可以从DRAM中较为高效的访问数据。

[0019] 第二方面，本申请实施例还提供了一种内存设置装置，有益效果可以参见第一方面的描述此处不再赘述。该设备具有实现上述第一方面的方法实例中行为的功能。该功能可以通过硬件实现，也可以通过硬件执行相应的软件实现。硬件或软件包括一个或多个与上述功能相对应的模块。在一个可能的设计中，该设备的结构中包括获取模块以及设置模块。可选的，还可以包括迁移模块和确定模块。这些单元可以执行上述第一方面方法示例中的相应功能，具体参见方法示例中的详细描述，此处不做赘述。

[0020] 第三方面，本申请实施例还提供了一种服务器，有益效果可以参见第一方面的描述此处不再赘述。服务器的结构中包括处理器和至少两个内存，处理器被配置为支持执行上述第一方面方法中相应的功能。所述至少两个内存与处理器耦合，其保存服务器必要的程序指令和数据。服务器的结构中还包括通信接口，用于与其他设备进行通信。

[0021] 第四方面，本申请还提供一种计算机可读存储介质，计算机可读存储介质中存储有指令，当其在计算机上运行时，使得计算机执行上述各方面的方法。

[0022] 第五方面，本申请还提供一种包含指令的计算机程序产品，当其在计算机上运行

时，使得计算机执行上述各方面的方法。

[0023] 第六方面，本申请还提供一种计算机芯片，芯片与存储器相连，芯片用于读取并执行存储器中存储的软件程序，执行上述各方面的方法。

附图说明

[0024] 图1为本申请提供的一种服务器架构示意图；

[0025] 图2为本申请提供的另一种服务器架构示意图；

[0026] 图3为本申请提供的一种内存设置方法示意图；

[0027] 图4为本申请提供的一种数据迁移方法示意图；

[0028] 图5为本申请提供的一种链表结构示意图；

[0029] 图6为本申请提供的一种列表结构示意图；

[0030] 图7为本申请提供的一种确定目标内存单元的方法示意图；

[0031] 图8为本申请提供的另一种数据迁移方法示意图；

[0032] 图9为本申请提供的一种优先级划分示意图；

[0033] 图10为本申请提供的另一种确定目标内存单元的方法示意图；

[0034] 图11为本申请提供的目标内存单元的分布示意图；

[0035] 图12为本申请提供的一种内存设置装置的结构示意图。

具体实施方式

[0036] 如图1所示，为本申请实施例提供的NUMA系统中的服务器100的一种架构示意图。服务器100中包括一个或多个处理器，对于任一处理器，都配置有自己内存，处理器与自己的内存通过系统总线连接。每个处理器的内存可分为两种，一种为本地内存，一种为远端内存。本地内存和远端内存用于存储该处理器运行所需的数据。

[0037] 图1中以服务器100中包括2个处理器，分别为处理器110和处理器120为例，处理器110的内存A分为本地内存111和远端内存112。其中，本地内存111的性能优于远端内存112的性能。

[0038] 处理器120的内存B分为本地内存121和远端内存122。其中，本地内存121的性能优于远端内存122的性能。

[0039] 现有技术中，一般将为处理器配置的内存设置为本地内存，而将处理器能够访问的其他处理器的内存设置为远端内存。但在本发明实施例中，根据处理器的内存的性能设置本地内存及远端内存，以使处理器优先访问性能优的内存。

[0040] 如图2所示，为本申请实施例提供的NUMA系统中的一个服务器100的另一种架构示意图。服务器100中包括一个或多个处理器，一个处理器可以从另一个处理器的内存中获取数据。也就是说，一个处理器也可以连接另一个处理器的内存。对于任一处理器，该处理器连接的内存(处理器连接的内存包括该处理器的内存和其他处理器的内存)分为本地内存和远端内存。本地内存的性能优于远端内存的性能。本地内存和远端内存用于存储该处理器运行所需的数据。在图2的架构中，是根据处理器的所能访问的所有内存的性能设置本地内存及远端内存，以使处理器优先访问性能优的内存。

[0041] 图2中以服务器中包括2个处理器，分别为处理器110和处理器120为例，处理器110

连接处理器120的内存B，处理器120连接处理器110的内存A。

[0042] 对于处理器110，处理器110连接的内存(也即处理器110的内存A和处理器120的内存B)可以分为本地内存111和远端内存112。

[0043] 对于处理器120，处理器120连接的内存(也即处理器110的内存A和处理器120的内存B)可以分为本地内存121和远端内存122。

[0044] 在当前的NUMA系统中，在服务器启动时，每个处理器会侦测系统中所有内存距离自己的距离，并把距离最近的处理器设置为本地内存，而将其他内存设置为远端内存。但在本申请实施例中，在服务器启动时，会检测系统中所有内存的性能或该处理器的内存的性能，并将性能最好的内存设置为本地内存，而将其他内存设置为远端内存。例如，图1和图2中本地内存121的性能优于远端内存122。关于如何根据内存性能设置本地内存和远端内存的方法请参考图3的描述。

[0045] 下面以本地内存121和远端内存122为例对本地内存和远端内存的类型进行说明，一般有以下几种情况：

[0046] 情况一、本地内存111和远端内存112的类型相同，但本地内存111的性能优于远端内存112。

[0047] 对于如图1所示的服务器架构，若处理器110的内存的类型相同，本地内存111为处理器110的内存中性能最高的内存，其余内存为远端内存112。对于如图2所示的服务器架构，若处理器110连接的内存的类型相同，本地内存111为处理器110连接的内存中性能最高的内存，其余内存为远端内存112。

[0048] 例如，处理器110的内存或处理器110连接的内存均为动态随机存取存储器(dynamicrandom access memory，DRAM)。但即便同一种类型的内存，内存在性能上也存在差异。例如，双倍速率同步动态随机存储器3(double data rate，DDR3)和DDR4均为DRAM，但DDR4性能上通常优于DDR3。又例如，相较于不具备错误检查和纠正(error correcting code，ECC)功能的DRAM，带有ECC功能的DRAM能够保证数据的完整性，安全性更高。又例如，内存频率更高的DRAM性能更优。又例如，制造日期越接近当前日期的内存的性能越优。又例如制作厂商为主流制作厂商的内存的性能比普通制作厂商的内存的性能好。

[0049] 在这种情况下，本地内存111和远端内存112均为DRAM，本地内存111可以为处理器110的内存中性能最好的DRAM，其余DRAM可以作为远端内存112(如图1所示的服务器架构)。本地内存111可以为处理器110连接的内存中性能最好的DRAM，其余DRAM可以作为远端内存112(如图2所示的服务器架构)。

[0050] 情况二、本地内存111和远端内存112的类型不同，但本地内存111的性能优于远端内存112。

[0051] 对于如图1所示的服务器架构，若处理器110的内存的类型不同，本地内存111为处理器110的内存中性能最高的内存，其余内存为远端内存112。对于如图2所示的服务器架构，若处理器110连接的内存的类型不同，本地内存111为处理器110连接的内存中性能最高的内存，其余内存为远端内存112。

[0052] 例如，处理器110的内存或处理器110连接的内存除了DRAM，还有其他类型的内存，如数据中心级持久性内存(data center persistent memory，DCPMM)。

[0053] 其中，DCPMM是一种特殊的存储器，在不同的模式下，可以作为非易失性内存或易

失性内存。举例来说,DCPMM存在三种不同的模式,分为内存模式(memory mode,MM)、应用模式(app direct,AD)以及混合模式(MIX)。其中,内存模式下的DCPMM可作为易失性内存,应用模式下的DCPMM可作为非易失性内存,能够实现掉电数据不丢失;混合模式下的DCPMM部分存储空间可以作为非易失性内存,部分存储空间可以作为易失性内存。

[0054] DCPMM仅是举例,本申请实施例中不限定其他类型内存的具体类型,凡是能够用于存储处理器110运行所需的数据的内存均适用于本申请实施例。需要说明的是,本申请中涉及的内存是指能够实现字节级访问的存储器。

[0055] 在这种情况下,本地内存111和远端内存112的类型不同,本地内存111可以为处理器110的内存中的DRAM,其余类型的内存可以作为远端内存112(如图1所示的服务器架构)。本地内存111可以为处理器110连接的内存中的DRAM,其余类型的内存可以作为远端内存112(如图2所示的服务器架构)。

[0056] 又例如,处理器110的内存或处理器110连接的内存中有多种性能不同的DRAM,且除了DRAM之外,还包括其他类型存储器。

[0057] 在这种情况下,本地内存111和远端内存112的类型不同,本地内存111可以为处理器110的内存中性能最好的DRAM,剩余内存可以作为远端内存112(如图1所示的服务器架构)。本地内存111可以为处理器110连接的内存中性能最好的DRAM,剩余内存可以作为远端内存112(如图2所示的服务器架构)。

[0058] 下面结合附图3,以如图1所示的服务器架构为例,对本申请实施例提供内存的分配方式进行说明,如图3所示,该方法包括:

[0059] 步骤301:处理器110确定处理器110的各个内存的性能。

[0060] 处理器110可以读取串行存在检测(serial presence detect,SPD)芯片所检测的信息,根据SPD芯片读取的信息确定各个内存的性能。其中,在系统启动阶段,SPD芯片能够对服务器中每个内存插槽中插入的内存进行检测,在对各个内存检测后,可以将检测到的信息保存在处理器110的内存中,便于处理器110后续读取SPD芯片所检测的信息。

[0061] SPD所检测的信息包括每个内存的信息。每个内存的信息包括但不限于:内存的类型,是否具备ECC功能,内存频率,制造日期(该内存的生产日期),制造厂商(制造该内存的厂商的名称)等信息。

[0062] 其中,内存的类型可以指示该内存为DRAM(如DDR3、DDR4),还是除DRAM外的其他类型的内存。

[0063] 若处理器110的各个内存的类型相同,均为DRAM。

[0064] 服务器根据SPD芯片所检测的信息确定各个内存的性能时,可以比较各个内存的信息,根据内存之间的差异信息确定内存的性能。其中内存之间的差异信息是指SPD所检测的信息中内存之间存在差异的信息。

[0065] 例如,SPD所检测的信息中记录内存1的类型为DDR3,内存2的类型为DDR4,内存的类型即为差异信息,处理器110确定内存2的性能优于内存1。又例如,SPD所检测的信息中记录内存1和内存2的类型均为DDR4,但内存1具备ECC功能,内存2不具备ECC功能,是否具备ECC功能的信息即为差异信息,处理器110确定内存1的性能优于内存2。又例如,SPD所检测的信息中记录内存1和内存2的类型均为DDR4,但内存1的频率高于内存2的频率,内存频率即为差异信息,处理器110确定内存1的性能优于内存2。又例如,SPD所检测的信息中记录内

存1和内存2的类型均为DDR4，但内存1的频率和内存2的频率均属于高频，制造厂商即为差异信息，内存1的制造厂商为主流厂商，内存2的制作厂商属于普通的制作厂商，处理器110确定内存1的性能优于内存2。

[0066] 若处理器110的各个内存的类型不同，除了DRAM之外，还包括其他类型的内存。

[0067] 这种情况下，处理器110可以默认DRAM的性能优于其他类型的内存。

[0068] 作为一种可能的实施方式，当处理器110的多个内存包括多种不同的DRAM，处理器110可以采用前述方法确定多种不同的DRAM的性能。

[0069] 步骤302：处理器110从处理器110的内存中选择性能最好的内存作为处理器110的本地内存111。

[0070] 处理器110在确定了处理器110的各个内存的性能之后，可以优先选择性能最好的内存做为本地内存111，剩余的内存作为远端内存112。

[0071] 在NUMA系统中，在系统启动阶段，可以调用acpi_numa_memory_affinity_init函数设置远端内存112相应的NUMA type字段为numa_nodes_pmem，本地内存111相应的NUMA type字段为numa_nodes_dram。

[0072] 本申请实施例并不限定本地内存111的大小，服务器可以根据处理器110所运行的进程预估在运行过程中所需存储的数据量，根据该数据量确定本地内存111的大小。例如，处理器110所运行的进程用于维护某一数据库，所需的存储的数据量较大，可以根据维护的数据库中经常需要被读写的数据的数据量确定该本地内存111的大小，进而从处理器110的内存中选择大小接近于该数据量、且性能最好的内存作为本地内存111，其中，数据库中经常需要被读写的数据的数据量可以由该数据库的输入输出(input output，IO)模型进行评估确定。

[0073] (1)、在处理器110的各个内存的类型相同，均为DRAM的情况下，处理器110可以选择性能最好的DRAM作为处理器110的本地内存111。

[0074] (2)、在处理器110的内存中除了DRAM之外，还包括其他类型的内存，处理器110可以选择DRAM作为处理器110的本地内存111。进一步的，若处理器的内存中有多种性能不同的DRAM，处理器110可以选择DRAM中性能最好的DRAM作为处理器110的本地内存111。

[0075] 服务器100中的各个处理器可以按照如图3所示的方法为设置本地内存111。如图3所示的方法也可以应用于如图2所示的服务器架构中，也即处理器110需要确定处理连接的内存的性能，选择性能最好的内存作为处理器110的本地内存111，具体实施方式可参见前述内容，此处不再赘述。

[0076] 对于任一处理器，该处理器的内存分为本地内存和远端内存，本地内存和远端内存可以用于存储处理器运行所需的数据。但由于处理器从性能较好的本地内存中进行数据读写效率较高，为了能够使处理器具备较佳的数据读写效率，可以将处理器的内存中读写频率最高的数据均存储在本地内存中，也就是需要从远端内存中读写效率较高的数据迁移至本地内存中。

[0077] 下面结合如图4，基于如图1所述的服务器架构，对处理器110的本地内存111和远端内存112进行数据迁移的方法进行说明，参见图4，该方法包括：

[0078] 步骤401：处理器110确定处理器110的内存中内存单元的数据读写频率。

[0079] 由于数据在处理器110的内存中存储时，通常是以内存单元(如内存页)为粒度进

行存储的。也就是说，内存可以包括多个内存单元，每个内存单元可以存储等量的数据。处理器110可以确定各个内存单元的数据读写频率。

[0080] 处理器110在执行步骤401时，可分为如下两个步骤：

[0081] 步骤1、处理器110多次读取扩展页表(extend page table，EPT)中的信息，确定处理器110的内存中各个内存单元的数据被读取的次数以及被写入的次数。

[0082] 其中，EPT记录了各个内存单元的读写状态。每个内存单元在EPT中对应两个字段，分别为dirty bit(为方便说明，简称为D字段)和access bit(为方便说明，简称为A字段)。

[0083] D字段用于指示是否有数据写入该内存单元，例如，0指示有数据写入，1指示无数据写入。A字段用于指示是否读取该内存单元的数据，例如，0指示未读取，1指示读取。

[0084] 针对处理器110的内存中任一内存单元，每当读取内存单元中的数据或在该内存单元中写入数据时，EPT中的相应字段将会被更新。

[0085] 例如，当读取一个内存单元中的数据时，EPT中该内存单元对应的D字段变为0，A字段变为1。当在该内存单元中写入数据时，EPT中该内存单元对应的D字段变为1，A字段变为1。

[0086] 处理器110多次读取EPT中的信息时，可以在一个时间段内，间隔一定时间读取EPT中的信息，读取次数可以为设定值。对于一个内存单元，若EPT中的信息记录该内存单元的数据被读取，则该内存单元的数据被读取的次数加一；若EPT中的信息记录该内存单元的数据被写入，则该内存单元的数据被写入的次数加一。在读取EPT中的信息的次数达到设定值后，确定处理器110记录的该处理器110的内存中各个内存单元的数据被读取的次数以及被写入的次数。

[0087] 需要说明的是，这里读取EPT的具体次数本申请实施例并不限定。由上可知，处理器110通过多次读取EPT中的信息确定的处理器110的内存中各个内存单元的数据被读取的次数以及被写入的次数，并不一定是准确的在该时间段内，各个内存单元的数据真实被读取的次数以及被写入的次数，但可以在一定程度上反映出各个内存单元的数据被读取的次数以及被写入的次数的相对值。

[0088] 步骤2、处理器110根据各个内存单元的数据被读取的次数以及被写入的次数确定各个内存单元的数据读写频率。

[0089] 处理器110在计算各个内存单元的数据读写频率时，内存单元的数据读写频率可以是根据该内存单元中数据被读取的次数以及被写入的次数确定的。例如，对于任一内存单元，该内存单元的数据读写频率可以等于内存单元中数据的被读取的次数与被写入的次数之和。又例如，可以分别设置读权重和写权重，计算内存单元中数据的被读取的次数和读权重的乘积1、以及内存单元中数据的被写入的次数和写权重的乘积2。该内存单元的数据读写频率可以等于乘积1与乘积2之和。读权重和写权重的具体数值本申请实施例并不限定，可以根据具体应用场景设置。

[0090] 由此，处理器110可以计算出各个内存单元的数据读写频率，处理器110可以保存各个内存单元的数据读写频率。处理器110在保存各个内存单元的数据读写频率时，可以构建链表，以记录各个内存单元的数据读写频率。如图5所示，为处理器110构建的一种链表示意图，针对每个一个内存单元，对应一个数组，该数组中包括该内存单元的地址、内存单元的访问总量(内存单元中数据的被读取的次数与被写入的次数之和)、内存单元的数据读写

频率。

[0091] 步骤402:处理器110统计具有各数据读写频率的内存单元的数量。

[0092] 处理器110在计算了各个内存单元的数据读写频率后,可以统计具有相同的数据读写频率的内存单元的数量,保存具有各个数据读写频率的内存单元的数量。具有各个数据读写频率的内存单元的数量可以构成一个列表保存在处理器110中,如图6所示,为处理器110保存的各个数据读写频率的内存单元的数量的列表,该列表中记录了不同数据读写频率的内存单元的数量,图6所示的数值仅是举例。

[0093] 步骤403:处理器110根据各个内存单元的数据读写频率确定处理器110的内存中数据读写频率不小于预设值的目标内存单元,目标内存单元的数量等于目标值N,目标值N可以是经验值,也可以是根据分布比例S与处理器110的内存中内存单元的数量的乘积确定的,其中分布比例S等于本地内存111中数据读写频率高于阈值的内存单元的数量与处理器110的内存中数据读写频率大于阈值的内存单元的数量的比值,关于确定目标内存单元的具体方法请参考图7的描述。

[0094] 步骤404:处理器110将位于远端内存112的目标内存单元中的数据迁移至本地内存111。

[0095] 标注了目标内存单元后,处理器110确定远端内存112中的目标内存单元,处理器110确定目标内存单元位于本地内存111或远端内存112的方式与确定内存单元位于本地内存111或远端内存112的方式相同,具体可参见图7所示的实施例中步骤701的相关说明,此处不再赘述。之后,将远端内存112中的目标内存单元迁移到本地内存111中。

[0096] 作为一种可能的实施方式,处理器110在执行步骤404时,可以用远端内存112的目标内存单元中的数据替换本地内存111中未被标注的内存单元中的数据,将本地内存111中被替换出的数据存储到远端内存112中。

[0097] 如图7所示,为本申请实施例提供的一种确定目标内存单元的方法,该方法包括:

[0098] 步骤701:处理器110可以先确定当前处理器110的内存中数据读写频率大于阈值的内存单元的分布情况。

[0099] 处理器110可以遍历处理器110内存中各个内存单元,当遍历的内存单元的数据读写频率大于阈值时,处理器110可以调用函数move-page(),输入该内存单元的虚拟地址,根据函数move-page()返回的参数确定该内存单元在本地内存111,还是远端内存112,直至将处理器110的内存中的所有内存单元遍历完成,可以计算出本地内存111中数据读写频率大于阈值的内存单元的数量和远端内存112中数据读写频率大于阈值的内存单元的数量。

[0100] 需要说明的是,函数move-page()可以根据输入的内存单元的虚拟地址输出参数,该参数可以指示该内存单元作为本地内存中的内存单元时,该本地内存所属的处理器。在本申请实施例中,由于本地内存111和远端内存112本质上均为处理器110的内存。为了能够区分本地内存111和远端内存112,处理器110可以将远端内存112可以设置一个虚拟处理器的本地内存111,该虚拟处理器可以不执行任何处理操作。当函数move-page()返回的参数指示处理器110时,说明该内存单元位于本地内存111中,当返回的参数指示虚拟处理器时,说明该内存单元位于远端内存112中。

[0101] 假设处理器110确定本地内存111中数据读写频率大于阈值的内存单元的数量为

第一值以及远端内存112中数据读写频率大于阈值的内存单元的数量为第二值。

[0102] 若第二值与第一值相差较小，说明在远端内存112中数据读写频率大于阈值的内存单元的数量较多，该处理器110从远端内存112中读写数据的频次较高，导致该处理器110读写数据的效率并不高，需要将远端内存112中读写频率较高的数据迁移到本地内存111中。

[0103] 若第二值与第一值相差较大，第二值较小，说明在远端内存112中数据读写频率大于阈值的内存单元的数量较少，远端内存112中读写频率较高的数据也较少，该处理器110从第二数据中读写数据的频次较低，这种情况下可以不进行数据迁移。

[0104] 需要说明的是，本申请实施例并不限定阈值的具体数值。例如，该阈值可以为零，处理器110可以统计本地内存111中非冷页的数量和远端内存112中非冷页的数量。冷页是指内存中很少被读写的内存页，非冷页是指除冷页之外的内存页。

[0105] 步骤702、处理器110根据本地内存111中数据读写频率大于阈值的内存单元的数量(第一值)和远端内存112中数据读写频率大于阈值的内存单元的数量(第二值)可以计算本地内存111中数据读写频率高于阈值的内存单元在所述处理器110的内存中数据读写频率大于阈值的内存单元的分布比例S。以第一值为T1，第二值为T2，分布比例S＝T1/(T1+T2)。

[0106] 步骤703、处理器110可以根据分布比例S确定是否需要进行数据迁移，例如分布比例S接近于100％，示例性的，分布比例S处于90％～100％间，说明本地内存111中存储了大部分需要频繁读或写的数据。若分布比例S低于90％，说明有一部分需要频繁读或写的数据存储在远端内存112中，需要进行数据迁移。

[0107] 处理器110也可以不根据分布比例S确定是否有需要进行数据迁移(也即不执行步骤703)，直接进行数据迁移，处理器110在进行数据迁移之前，需要先根据分布比例S确定目标内存单元的数量(步骤704)，之后根据目标内存单元的数量从处理器110内存中标注目标内存单元(步骤705)。

[0108] 步骤704、处理器110将分布比例S与处理器110内存中内存单元的总数的乘积T作为目标值N，该目标值N为处理器110内存中数据读写频率排序在前S的内存单元的数量。

[0109] 在本申请实施例中允许目标值N存在小范围的波动。例如，处理器110可以在计算了目标值N之后，更新该目标值N，如在该目标值N的基础上，减少特定值。又例如，处理器110也可以选择比分布比例S小的值S1，将S1与处理器110内存中内存单元的总数的乘积作为目标值N。处理器110选择S1的方式本申请实施例并不限定，例如处理器110可以在分布比例S的基础上减去设定值后，作为S1。

[0110] 步骤705、处理器110在确定了目标值N之后，根据内存单元的数据读写频率从处理器110内存中标注目标内存单元。

[0111] 由前述内容可知，分布比例S可以反映出本地内存111中所能存储的数据读写频率大于阈值的内存单元(需要被频繁读取的数据)的数量。举例来说，处理器110统计计算的第一值为40，第二值也为60，计算的分布比例为40％，说明当前本地内存111中存储了处理器110内存中40％的数据读写频率大于阈值的内存单元中的数据。但在未进行数据迁移之前，本地内存111中该数据读写频率大于阈值的内存单元的数据并不一定包括处理器110中内存中数据读写频率最高的数据。

[0112] 为了能够保证本地内存111中存储的40％的数据读写频率大于阈值的内存单元中的数据为处理器110的内存中数据读写频率最大的、排序在前40％的内存单元的数据。处理器110可以先计算出数据读写频率最大的、排序在前40％的内存单元的数量N。之后，再根据内存单元的数据读写频率标注出数量等于N的目标内存单元。这样，标注出的目标内存单元即为数据读写频率最大的、排序在前40％的内存单元。

[0113] 若不采用分布比例S与处理器110内存中内存单元的总数的乘积T作为目标值N，当目标值N过大，会导致本地内存111和远端内存112进行大量的数据迁移，本地内存111与远端内存112之间的数据需要频繁的迁入迁出，会降低整体系统的性能。当目标值N过小，本地内存111和远端内存112仅进行少量的数据迁移，数据迁移后，本地内存111所存储的数据中也仅有少部分是需要被处理器110频繁读写的，并不能提高处理器110的数据读写效率。可见由分布比例S确定的目标值N规定了数据迁移时，本地内存110所需要存储数据读写频率较高的内存单元的一个上限值，能够保证在不改变分布比例S的前提下，使本地内存111中能够存储较多的需要被频繁读写的数据。

[0114] 下面对处理器110内存单元的数据读写频率从处理器110内存中标注目标内存单元的方式进行说明。

[0115] 处理器110可以先确定目标数据读写频率，该处理器110的内存中数据读写频率大于目标数据读写频率的内存单元的数量小于目标值N，处理器110的内存中数据读写频率不小于目标数据读写频率的内存单元的数量不小于目标值N。

[0116] 示例性的，处理器110可以利用预先保存的各个数据读写频率的内存单元的数量，从数据读写频率最高的内存单元的数量开始，按照数据读写频率由大到小的顺序，依次累加，记累加值为D，直至累加值D最接近目标值N，但不大于目标值N，将还未累加的最大的数据读写频率作为目标数据读写频率。

[0117] 以目标值N为80，预先保存的各个数据读写频率的内存单元的数量如图6所示为例，处理器110可以从数据读写频率为100的内存单元开始累加，当累加到数据读写频率为60的内存单元时，累加值为70，70最接近目标值80，且小于目标值N(累加到数据读写频率为50的内存单元时，累加值为100，已超过目标值N)。数据读写频率50即为目标数据读写频率。

[0118] 之后，处理器110标注目标内存单元。示例性的，处理器110标注处理器110内存中数据读写频率大于目标数据读写频率的内存单元，还可以标注目标数据读写频率的内存单元中的部分内存单元，该部分内存单元的数量等于目标值N与累加值的差值。

[0119] 仍以目标值N为80，预先保存的各个数据读写频率的内存单元的数量如图6所示为例，数据读写频率50即为目标数据读写频率。处理器110标注处理器110内存中数据读写频率大于50的内存单元，之后从数据读写频率为50的内存单元中标注10(目标值80与累加值70的差值)个内存单元。

[0120] 处理器110标注的目标内存单元中的数据即为处理器110内存中读取频率排序在前S的数据，包括了处理器110内存中数据读写频率不小于预设值(也即目标数据读写频率)的内存单元中的数据。这样，当处理器110进行数据读写时，大部分数据读写操作发生在本地内存111中，能够有效的提高处理器110的数据读写效率。下面结合如图8，基于如图1所述的服务器架构，对处理器110的本地内存111和远端内存112中另一种数据迁移的方式进行说明，参见图8，该方法包括：

[0121] 步骤801:同步骤401,具体可参见前述内容,此处不再赘述。

[0122] 步骤802:同步骤402,具体可参见前述内容,此处不再赘述。

[0123] 步骤803:处理器110根据内存单元的数据读写频率对处理器110的内存中内存单元划分优先级。

[0124] 数据读写频率高的内存单元优先级高。本申请实施例并不限定优先级的划分方式,例如,处理器110可以基于最低的数据读写频率,以20为步长,划分优先级。例如最低数据读写频率为0,数据读写频率从0到20的内存单元处于一个优先级,记为优先级1。数据读写频率从30到50的内存单元处于一个优先级,记为优先级2。读写频率从60到80的内存单元处于一个优先级,记为优先级3。读写频率从90到100的内存单元处于一个优先级,记为优先级4。

[0125] 处理器110可以保存各个内存单元的优先级,处理器110可以采用队列的方式保存各个内存单元的优先级,如图9所示,处理器110可以保存每个优先级队列,属于同一队列的优先级相同,每个优先级队列中记录了该优先级队列的优先级,以及该优先级中所包括的内存单元的信息(如内存单元的标识、虚拟地址等)。

[0126] 步骤804:处理器110根据处理器110的内存中各个内存单元的优先级确定处理器110的内存中数据读写频率不小于预设值的目标内存单元,目标内存单元的数量等于目标值N,目标值N的说明可参见前述内容,此处不再赘述。关于确定目标内存单元的具体方法请参考图10的描述。

[0127] 步骤805:同步骤404,具体可参见前述内容,此处不再赘述。

[0128] 如图10所示,为本申请实施例提供的另一种确定目标内存单元的方法,该方法包括:

[0129] 步骤1001:同步骤701,具体可参见前述内容,此处不再赘述。

[0130] 步骤1002:同步骤702,具体可参见前述内容,此处不再赘述。

[0131] 步骤1003:同步骤703,具体可参见前述内容,此处不再赘述。

[0132] 步骤1004:同步骤704,具体可参见前述内容,此处不再赘述。

[0133] 步骤1005、处理器110在确定了目标值N之后,根据内存单元的优先级从处理器110内存中标注目标内存单元。

[0134] 处理器110可以先确定处理器110内存中的内存单元的目标优先级,目标优先级需要满足如下条件:处理器110的内存中优先级大于目标优先级的内存单元的总量小于目标值N,处理器110的内存中优先级不小于目标优先级的内存单元的总量不小于目标值N。

[0135] 处理器110确定目标优先级的方式有很多种,下面列举其中两种:

[0136] (1)、处理器110可以利用预先保存的各个数据读写频率的内存单元的数量,从数据读写频率最高的内存单元的数量开始,按照读写频率由大到小的顺序,依次累加,记累加值为D,直至累加值D最接近目标值N,但不大于目标值N,将还未累加的内存单元的最大优先级作为目标优先级,该目标优先级也为当前还未累加的最大数据读写频率的内存单元所属的优先级。

[0137] 以预先保存的各个读写频率的内存单元的数量如图6所示、优先级的划分如图9所示、目标值N为80为例,处理器110可以从读写频率为100的内存单元开始累加,当累加到读写频率为60的内存单元时,累加值为70,70最接近目标值80,且小于目标值N(累加到读写频

率为50的内存单元时，累加值为100，已超过目标值N)。读写频率50所属的优先级2即为目标优先级。

[0138] (2)、处理器110可以利用预先保存的各个读写频率的内存单元的数量以及优先级对应的读写频率范围，从优先级最高的内存单元的数量开始，按照优先级由大到小的顺序，依次累加，记累加值为D，直至累加值D最接近目标值N，但不大于目标值N，将还未累加的内存单元的最大优先级作为目标优先级。

[0139] 仍以预先保存的各个数据读写频率的数据的内存单元的数量如图6所示、优先级的划分如图9所示、目标值N为80为例，处理器110可以从优先级4的内存单元开始累加，当累加到优先级为3的内存单元时，累加值为45，45最接近目标值80，且小于目标值N(累加到优先级为2的内存单元时，累加值为145，已超过目标值N)。还未累加的最大优先级为优先级2即为目标优先级。

[0140] 处理器110标注处理器110内存中优先级大于目标优先级的内存单元，还可以标注目标优先级的内存单元中的部分内存单元，该部分内存单元的数量等于目标值N与累加值的差值，且部分内存单元的读写频率不小于目标数据读写频率。

[0141] 仍以目标值N为80，预先保存的各个数据读写频率的内存单元的数量如图6所示为例，优先级3即为目标优先级。处理器110标注处理器110内存中优先级大于2的内存单元，之后优先级为2的内存单元中标注读写频率为50的内存单元，处理器110需要标注10个读写频率为50的内存单元，最终标注的内存单元的数量才能达到目标值N。如图11所示，为处理器110所标注的目标内存单元，其中，底色为灰色的内存单元即为目标内存单元。

[0142] 处理器110标注的目标内存单元中的数据即为处理器110内存中数据读写频率排序在前S的数据，包括了处理器110内存中数据读写频率大于预设值内存单元中的数据。

[0143] 另外，处理器110也可以将处理器110内存中数据读写频率最低的，且位于本地内存111的数据迁移到远端内存112，本申请实施例并不限定数据从本地内存111到远端内存112的迁移方法，处理器110可以将本地内存111中数据读写频率小于阈值的数据迁移到远端内存112中。

[0144] 基于与方法实施例同一发明构思，本申请实施例还提供了一种内存设置装置，用于执行上述方法实施例中处理器110执行的方法，相关特征可参见上述方法实施例，此处不再赘述，如图12所示，该装置用于对处理器的至少两个内存进行设置，该装置包括获取模块1201以及设置模块1202。可选的，还包括迁移模块1203以及确定模块1204。

[0145] 获取模块1201，用于当处理器启动时，获取至少两个内存的性能。获取模块用于执行如图3所示的实施例中步骤301。

[0146] 设置模块1202，用于根据至少两个内存的性能设置至少两个内存中的至少一个为本地内存；设置至少两个内存中的至少一个为远端内存，其中，本地内存的性能优于远端内存的性能。设置模块用于执行如图3所示的实施例中步骤302。

[0147] 在一种可能的实施方式中，该装置还可以对本地内存和远端内存中的数据进行迁移。其中，迁移模块1203可以将远端内存中数据读写频率不低于第一预设值(如前述方法实施例中的目标数据读写频率)的数据迁移至本地内存。迁移模块1203用于执行如图4或8所示的实施例。

[0148] 在一种可能的实施方式中，确定模块1204可以用于确定第一预设值，该第一预设

值可以是经验值，也可以是根据处理器的内存中各个内存页的数据读写频率确定的。

[0149] 例如，确定模块1204可以将内存中数据读写频率由高至低排列的内存页中的前N个内存页，作为需要存储在本地内存中的内存页确定模块1204可以将内存中数据读写频率由高至低排列的内存页中的第N个内存页的数据读写频率为第一预设值。确定模块1204用于执行如图7所示的实施例。

[0150] 又例如，确定模块1204可以根据内存中内存页的数据读写频率对内存中的内存页划分优先级，每个优先级对应一个数据读写频率范围，不同优先级对应的数据读写频率范围不同；将内存中优先级由高至低排列的内存页中的前N个内存页作为确定需要存储在本地内存中的内存页，第N个内存页的数据读写频率为第一预设值。确定模块1204用于执行如图10所示的实施例。

[0151] 在一种可能的实施方式中，确定模块1204在确定需要存储在本地内存中的内存页的数量N时，可以分别确定本地内存及远端内存中数据读写频率大于第二预设值的内存页；再确定本地内存中数据读写频率大于第二预设值的内存页占内存中数据读写频率大于第二预设值的内存页的比例；将该比例乘以内存中被使用的内存页的总数量的结果值作为数量N。

[0152] 在一种可能的实施方式中，本地内存和远端内存均为DRAM。

[0153] 在一种可能的实施方式中，本地内存为DRAM，远端内存为非DRAM。

[0154] 在一个简单的实施例中，本领域的技术人员可以想到上述实施例中的处理器所在的服务器可以如图1或2所示。具体的，图12中的获取模块1201、设置模块1202、迁移模块1203以及确定模块1204的功能/实现过程均可以通过图1或图2中的处理器110调用处理器的内存中存储的计算机执行指令来实现。

[0155] 本领域内的技术人员应明白，本申请的实施例可提供为方法、系统、或计算机程序产品。本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0156] 本申请是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器，使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0157] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中，使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品，该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0158] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上，使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理，从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0159] 显然，本领域的技术人员可以对本申请实施例进行各种改动和变型而不脱离本申

请实施例范围。这样，倘若本申请实施例的这些修改和变型属于本申请权利要求及其等同技术的范围之内，则本申请也意图包含这些改动和变型在内。

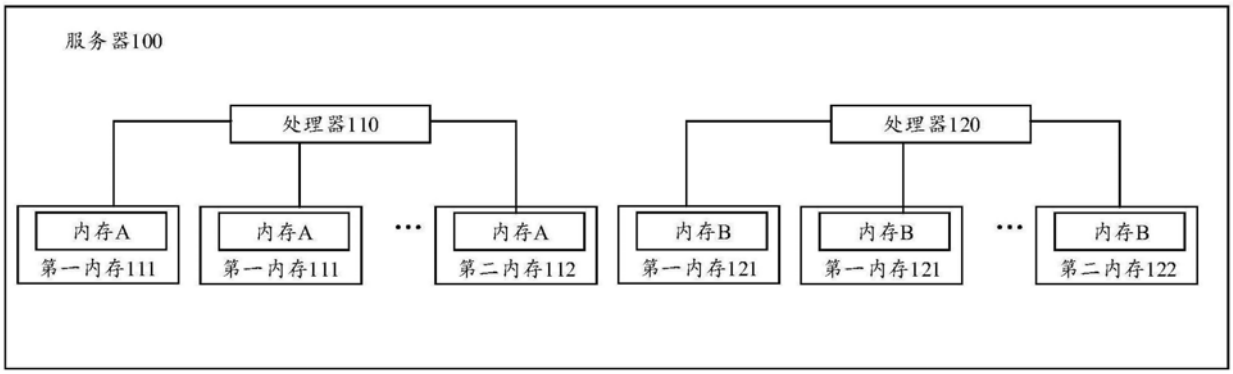

服务器100
处理器110
处理器120
内存A
第一内存111
内存A
第一内存111
内存A
第二内存112
内存B
第一内存121
内存B
第一内存121
内存B
第二内存122

图1

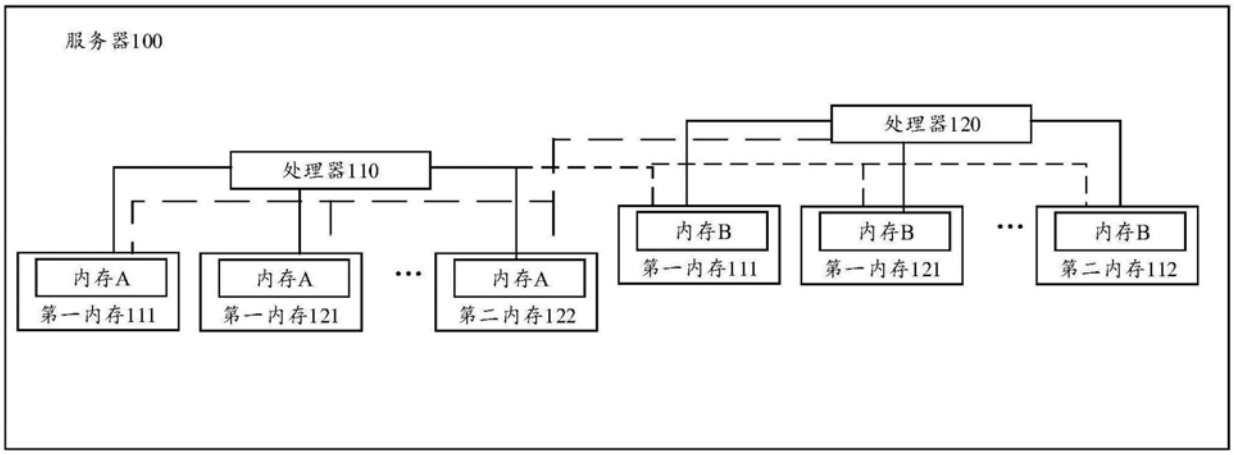

服务器100
处理器120
处理器110
内存A
第一内存111
内存A
第一内存121
内存A
第二内存122
内存B
第一内存111
内存B
第一内存121
内存B
第二内存112

图2

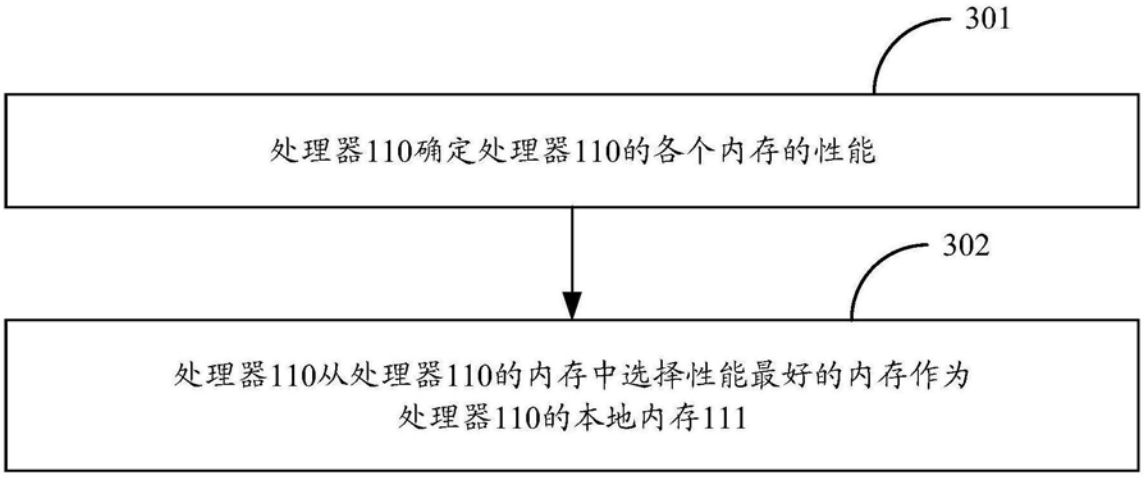

301
处理器110确定处理器110的各个内存的性能
302
处理器110从处理器110的内存中选择性能最好的内存作为
处理器110的本地内存111

图3

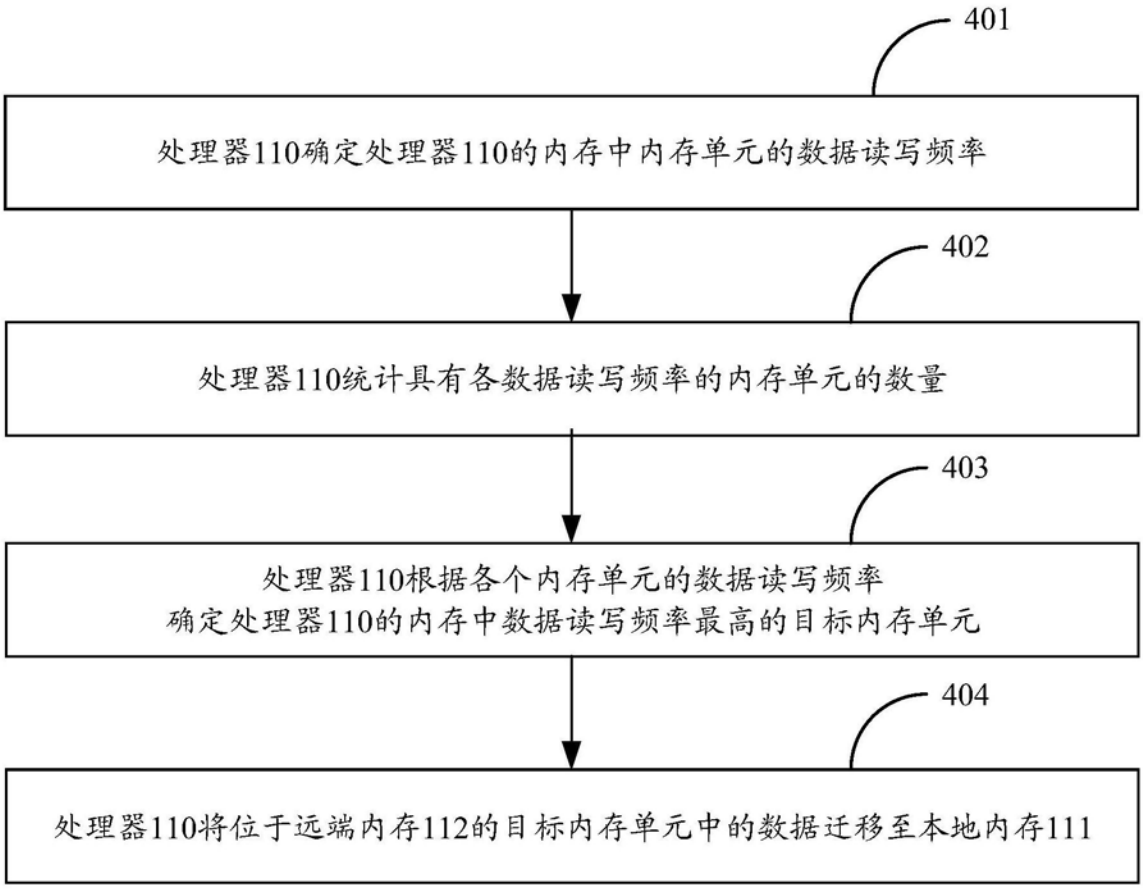

401
处理器110确定处理器110的内存中内存单元的数据读写频率
402
处理器110统计具有各数据读写频率的内存单元的数量
403
处理器110根据各个内存单元的数据读写频率
确定处理器110的内存中数据读写频率最高的目标内存单元
404
处理器110将位于远端内存112的目标内存单元中的数据迁移至本地内存111

图4

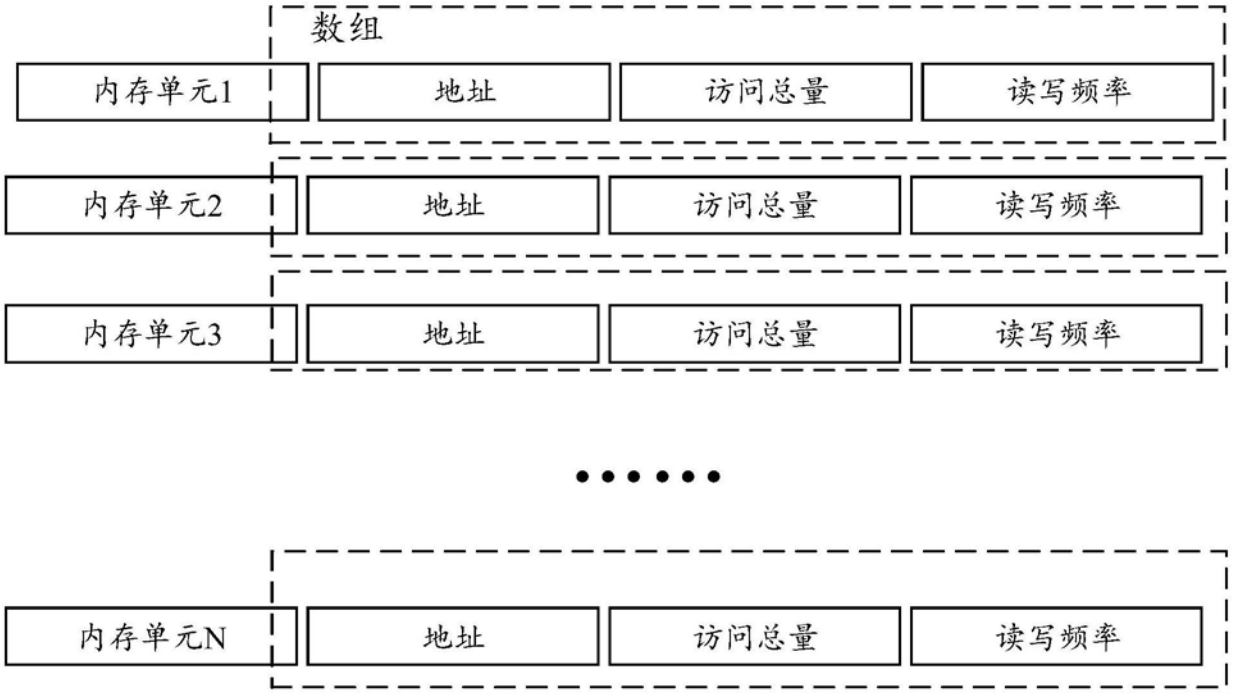

数组
内存单元1
地址
访问总量
读写频率
内存单元2
地址
访问总量
读写频率
内存单元3
地址
访问总量
读写频率
内存单元N
地址
访问总量
读写频率

图5

数据读写频率	0	10	20	30	40	50	60	70	80	90	100
内存单元的数量	5	7	13	20	25	30	25	20	12	8	5

图6

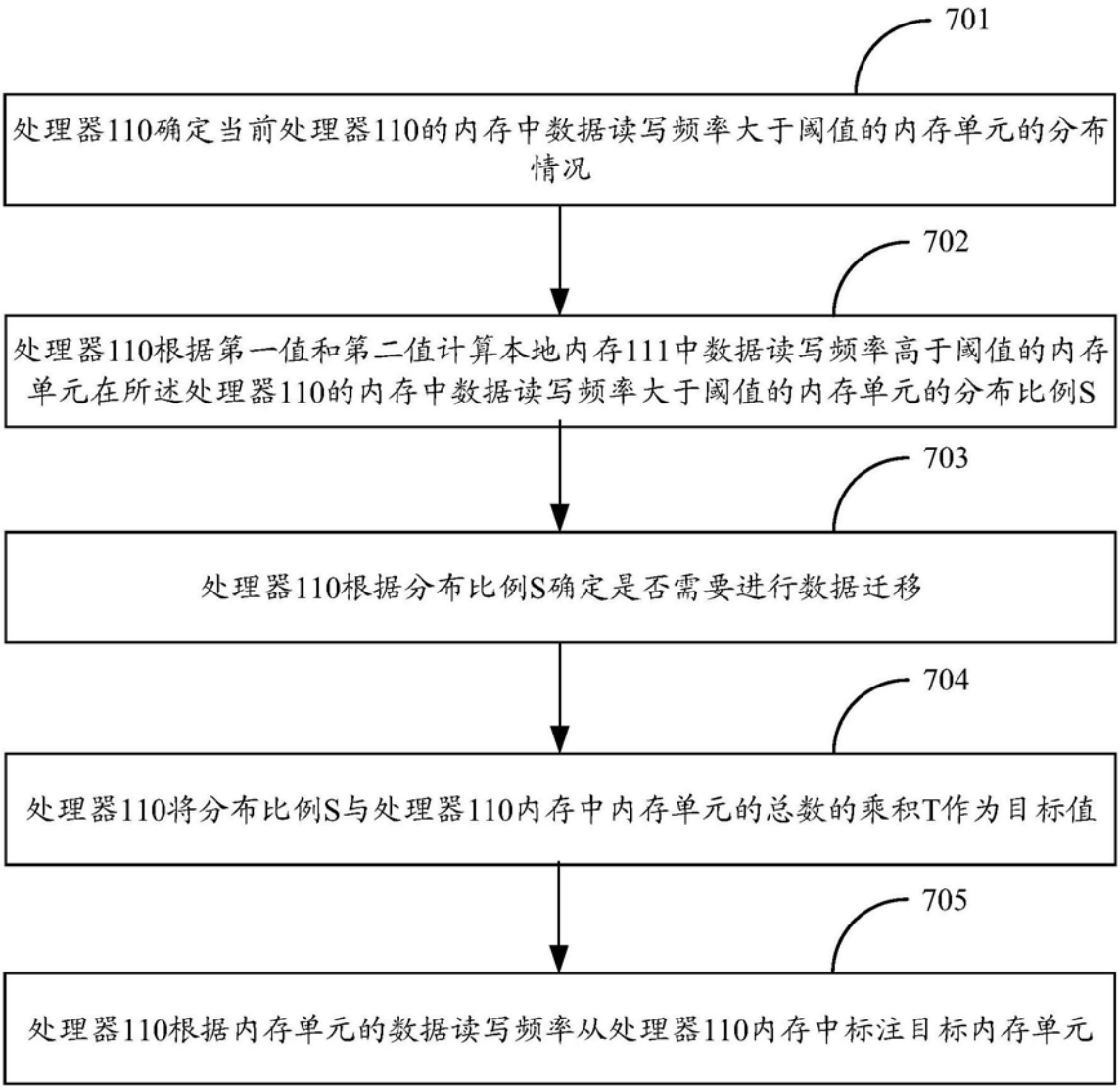


图7

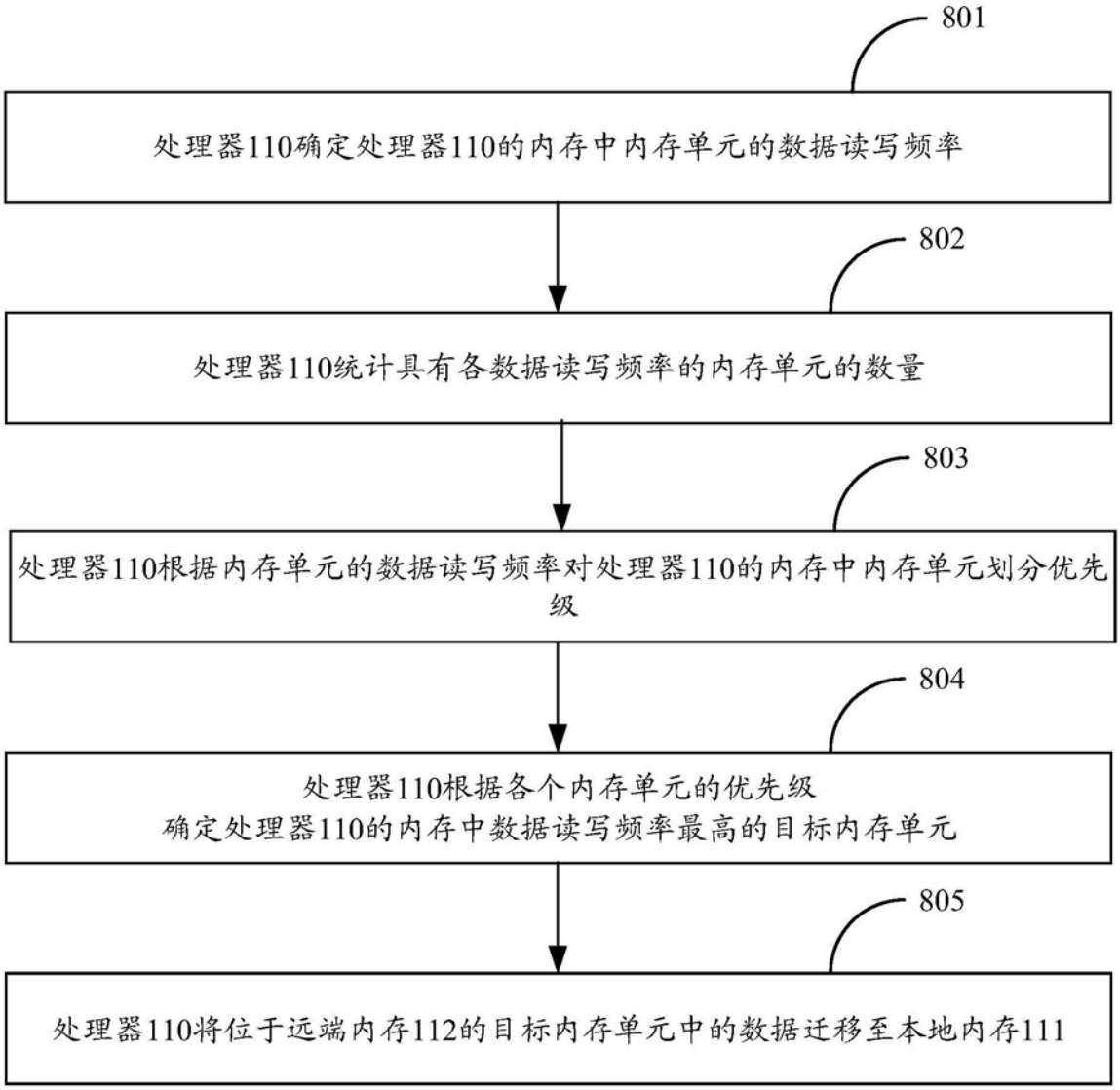

801
处理器110确定处理器110的内存中内存单元的数据读写频率
802
处理器110统计具有各数据读写频率的内存单元的数量
803
处理器110根据内存单元的数据读写频率对处理器110的内存中内存单元划分优先级
804
处理器110根据各个内存单元的优先级
确定处理器110的内存中数据读写频率最高的目标内存单元
805
处理器110将位于远端内存112的目标内存单元中的数据迁移至本地内存111

图8

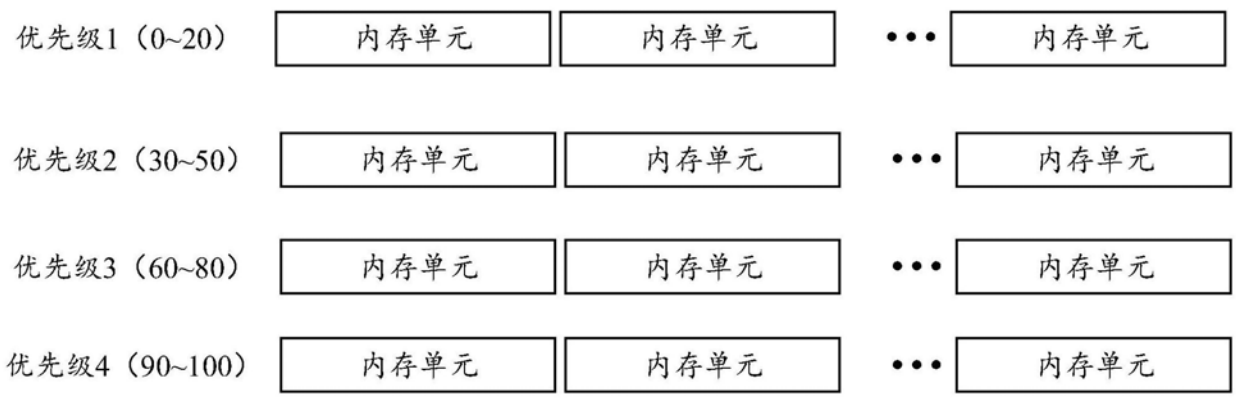

优先级1（0~20）
内存单元
内存单元
内存单元
优先级2（30~50）
内存单元
内存单元
内存单元
优先级3（60~80）
内存单元
内存单元
内存单元
优先级4（90~100）
内存单元
内存单元
内存单元

图9

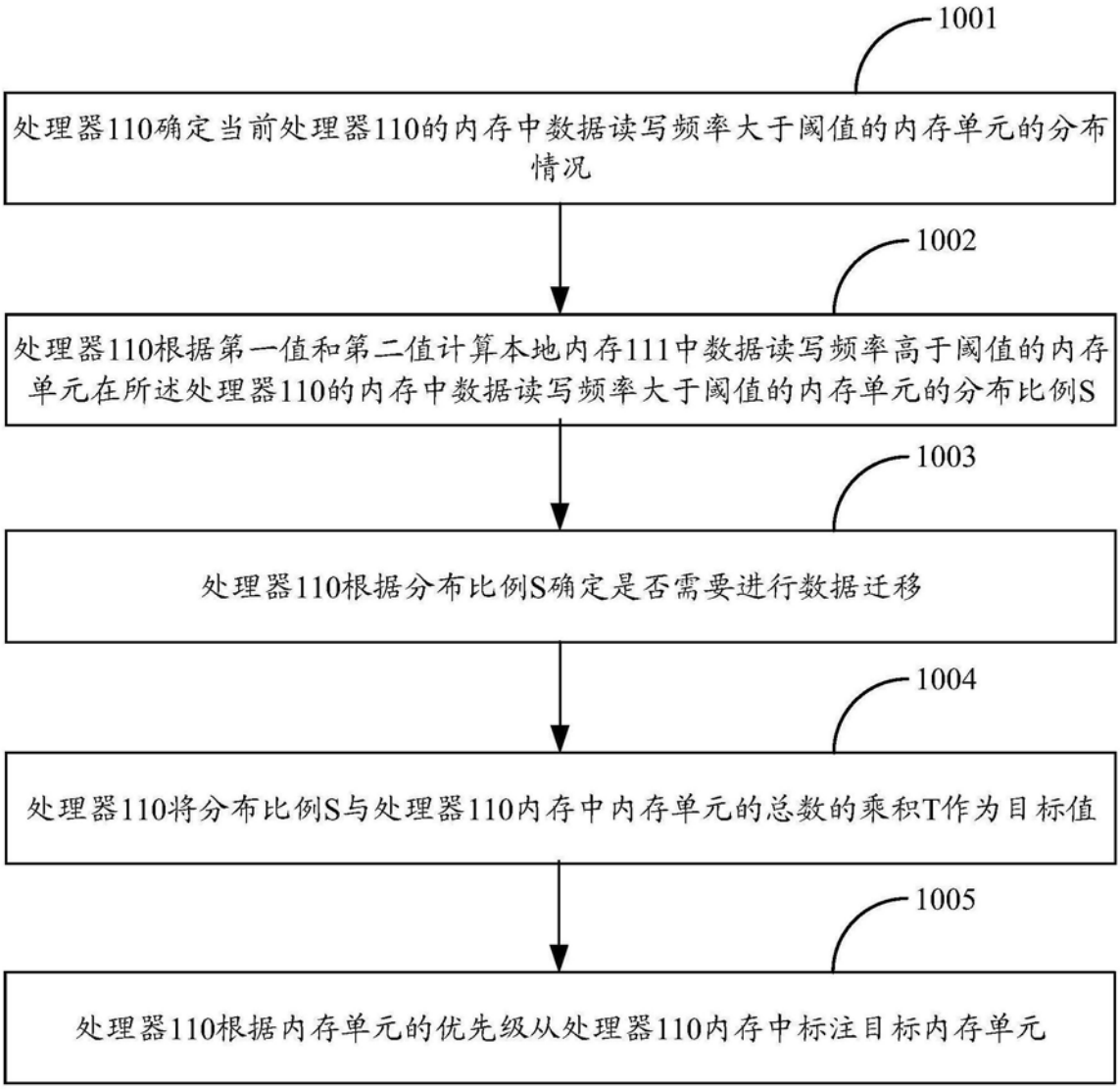

1001
处理器110确定当前处理器110的内存中数据读写频率大于阈值的内存单元的分布情况
1002
处理器110根据第一值和第二值计算本地内存111中数据读写频率高于阈值的内存单元在所述处理器110的内存中数据读写频率大于阈值的内存单元的分布比例S
1003
处理器110根据分布比例S确定是否需要进行数据迁移
1004
处理器110将分布比例S与处理器110内存中内存单元的总数的乘积T作为目标值
1005
处理器110根据内存单元的优先级从处理器110内存中标注目标内存单元

图10

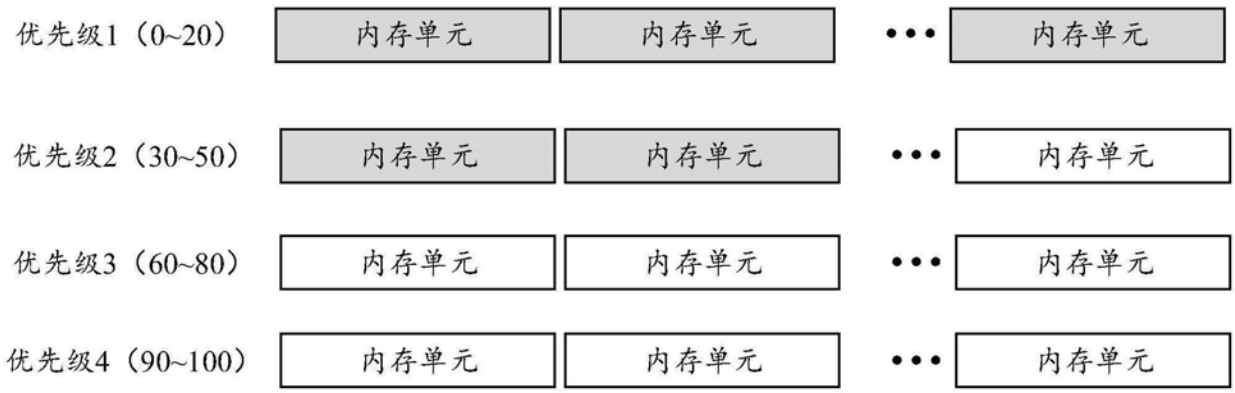

优先级1（0~20）
内存单元
内存单元
内存单元
优先级2（30~50）
内存单元
内存单元
内存单元
优先级3（60~80）
内存单元
内存单元
内存单元
优先级4（90~100）
内存单元
内存单元
内存单元

图11

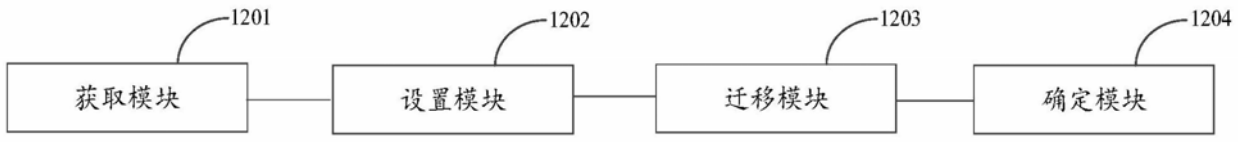


图12