

1. 一种由缓冲器管理器执行的将去往存储器缓冲器的数据排队的方法,所述方法包括:

接收以下请求,所述请求针对由所述缓冲器管理器分配的、用于存储从多个队列中选择的一个队列的数据令牌集合的地址的指针和地址,其中所述数据令牌集合包括两个以上数据令牌;

访问存储器列表,并且在所述存储器列表中生成指向所分配的地址的指针和地址,其中所述存储器列表包括两个以上并联链接的存储器列表;

向所述存储器列表中写入用于所分配的地址的指针,以形成由两个以上链接的存储器位置构成的集合,其中由两个以上链接的存储器位置构成的所述集合中的每个存储器位置被分配所述两个以上并联链接的存储器列表中的不同的并联链接的存储器列表,其中所述指针将由两个以上链接的存储器位置构成的所述集合中的每个存储器位置外部地链接在一起,并且其中所述两个以上并联链接的存储器列表中的每个存储器列表都被分成两个以上存储器位置;以及

在从所述队列接收到另一个数据令牌集合时向另一个由两个以上链接的存储器位置构成的集合移植附加地址分配,并且生成将所述另一个由两个以上链接的存储器位置构成的集合中的存储器位置链接在一起的附加指针。

2. 根据权利要求1所述的方法,还包括:

生成尾指针和头指针,所述尾指针和头指针跨所述两个以上并联链接的存储器列表链接所述数据令牌集合的所分配的两个以上地址。

3. 根据权利要求1所述的方法,其中

所述缓冲器管理器被耦合到入队处理器,所述入队处理器可操作用于将所述数据令牌集合向所述两个以上并联链接的存储器列表的所分配的地址中入队,并且

所述缓冲器管理器还被耦合到出队模块,所述出队模块可操作用于将所述数据令牌集合从所述两个以上并联链接的存储器列表的所分配的地址出队。

4. 根据权利要求3所述的方法,还包括:

接收来自所述入队处理器的请求以用于将所述两个以上数据令牌入队,其中所述两个以上数据令牌具有通过指针跨所述两个以上并联链接的存储器列表链接的地址。

5. 根据权利要求3所述的方法,还包括:

接收来自所述出队模块的请求以用于将所述两个以上数据令牌出队,其中所述两个以上数据令牌具有通过指针跨所述两个以上并联链接的存储器列表链接的地址。

6. 根据权利要求1所述的方法,还包括:

根据地址可用性填入用于所述附加地址分配的所述两个以上并联链接的存储器列表。

7. 根据权利要求6所述的方法,其中按照连续跨越所述不同的并联链接的存储器列表中的每一个存储器列表的循环方式执行所述填入,直至所有请求的地址被分配。

8. 一种计算机系统,包括:

存储器缓冲器;和

耦合到所述存储器缓冲器的缓冲器管理器,所述缓冲器管理器被配置为:

接收以下请求,所述请求针对由所述缓冲器管理器分配的、用于存储从多个队列中选择的一个队列的数据令牌集合的地址的指针和地址,其中所述数据令牌集合包括两个以上

数据令牌；

访问存储器列表，并且在所述存储器列表中生成指向所分配的地址的指针和地址，其中所述存储器列表包括两个以上并联链接的存储器列表；

向所述存储器列表中写入用于所分配的地址的指针，以形成由两个以上链接的存储器位置构成的集合，其中由两个以上链接的存储器位置构成的所述集合中的每个存储器位置被分配所述两个以上并联链接的存储器列表中的不同的并联链接的存储器列表，其中所述指针将由两个以上链接的存储器位置构成的所述集合中的每个存储器位置外部地链接在一起，并且其中所述两个以上并联链接的存储器列表中的每个存储器列表都被分成两个以上存储器位置；以及

在从所述队列接收到另一个数据令牌集合时向另一个由两个以上链接的存储器位置构成的集合移植附加地址分配，并且生成将所述另一个由两个以上链接的存储器位置构成的集合中的存储器位置链接在一起的附加指针。

9. 根据权利要求8所述的计算机系统，其中所述两个以上并联链接的存储器列表包括并联链接的第一存储器列表、第二存储器列表、第三存储器列表和第四存储器列表。

10. 根据权利要求8所述的计算机系统，其中所述数据令牌集合的所分配的两个以上地址通过尾指针和头指针跨所述两个以上并联链接的存储器列表被链接。

11. 根据权利要求8所述的计算机系统，还包括入队处理器，所述入队处理器被配置为：

将所述数据令牌集合向所述两个以上并联链接的存储器列表的所分配的地址中入队，并且向所述缓冲器管理器生成请求以用于将所述两个以上数据令牌入队，并且其中所述两个以上数据令牌具有通过指针跨所述两个以上并联链接的存储器列表链接的地址。

12. 根据权利要求8所述的计算机系统，还包括出队模块，所述出队模块被配置为：

将所述数据令牌集合从所述两个以上并联链接的存储器列表的所分配的地址出队，并且向所述缓冲器管理器生成请求以用于将所述两个以上数据令牌出队，并且其中所述两个以上数据令牌具有通过指针跨所述两个以上并联链接的存储器列表链接的地址。

13. 根据权利要求8所述的计算机系统，其中所述用于所述附加地址分配的所述两个以上并联链接的存储器列表根据地址可用性被动态地填入。

14. 根据权利要求13所述的计算机系统，其中按照连续跨越所述不同的并联链接的存储器列表中的每一个存储器列表的循环方式执行所述两个以上并联链接的存储器列表的填入，直至所有请求的地址被分配。

15. 一种数据处理系统，包括：

至少一个处理器，可操作用于将数据令牌集合向存储器列表入队和用于将所述数据令牌集合从并联链接的存储器列表的所分配的地址出队，其中所述数据令牌集合包括两个以上数据令牌，并且其中所述存储器列表包括两个以上并联链接的存储器列表；和

耦合到所述至少一个处理器的缓冲器管理器，所述缓冲器管理器可操作用于执行一种将去往存储器缓冲器的数据排队的方法，所述方法包括：

接收以下请求，所述请求针对由所述缓冲器管理器分配的、用于存储所述数据令牌集合的地址的指针和地址；

访问存储器列表，并且在所述存储器列表中生成指向所分配的地址的指针和地址；

向所述存储器列表中写入用于所分配的地址的指针，以形成由两个以上链接的存储器

位置构成的集合,其中由两个以上链接的存储器位置构成的所述集合中的每个存储器位置被分配所述两个以上并联链接的存储器列表中的不同的并联链接的存储器列表,其中所述指针将由两个以上链接的存储器位置构成的所述集合中的每个存储器位置外部地链接在一起,并且其中所述两个以上并联链接的存储器列表中的每个存储器列表都被分成两个以上存储器位置;以及

在接收到另一个数据令牌集合时向另一个由两个以上链接的存储器位置构成的集合移植附加地址分配,并且生成将所述另一个由两个以上链接的存储器位置构成的集合中的存储器位置链接在一起的附加指针。

16. 根据权利要求15所述的数据处理系统,其中所述两个以上并联链接的存储器列表包括并联链接的第一存储器列表、第二存储器列表、第三存储器列表和第四存储器列表。

17. 根据权利要求15所述的数据处理系统,其中所述数据令牌集合的所分配的两个以上地址通过尾指针和头指针跨所述两个以上并联链接的存储器列表被链接。

18. 根据权利要求15所述的数据处理系统,还包括入队处理器,所述入队处理器被配置为:

将所述数据令牌集合向所述两个以上并联链接的存储器列表的所分配的地址中入队,并且向所述缓冲器管理器生成请求以用于将所述两个以上数据令牌入队,并且其中所述两个以上数据令牌具有通过指针跨所述两个以上并联链接的存储器列表链接的地址。

19. 根据权利要求15所述的数据处理系统,还包括出队模块,所述出队模块被配置为:

将所述数据令牌集合从所述两个以上并联链接的存储器列表的所分配的地址出队,并且向所述缓冲器管理器生成请求以用于将所述两个以上数据令牌出队,并且其中所述两个以上数据令牌具有通过指针跨所述两个以上并联链接的存储器列表链接的地址。

20. 根据权利要求15所述的数据处理系统,其中所述用于所述附加地址分配的所述两个以上并联链接的存储器列表根据地址可用性被动态地填入,并且其中按照连续跨越所述不同的并联链接的存储器列表中的每一个存储器列表的循环方式执行所述两个以上并联链接的存储器列表的填入,直至所有请求的地址被分配。

用于使用多个链接的存储器列表的方法及装置

[0001] 本申请是申请号为201510290966.8、申请日为2015年05月29日、发明名称为“用于使用多个链接的存储器列表的方法及装置”的中国发明专利申请的分案申请。

技术领域

[0002] 本公开内容总体上涉及网络数据传输领域，并且更具体地涉及一种用于在将数据令牌从所选择的队列出队和入队期间链接多个存储器列表以便增加交换网络在传输数据分组时的吞吐量的系统和方法。

背景技术

[0003] 在数据交换网络中，数据业务被分类为在缓冲器中的大量队列中存储的各种流。在路由器或者其他网络元件中，所存储的队列通常竞争公共的外向通信链路或者出站端口（例如物理通信链路或者伪线）。因此，所缓冲的队列需要在出站侧被调度。因此，需要由路由器中输出侧的令牌调度设备在传输之前执行处理以选择所排队的分组中的哪个分组将为队列中的为了外向传输的下一个分组。

[0004] 通常，在路由过程中，来自不同源端口的数据令牌基于其源端口和目的端口以及业务类型而被分类。它们随后被分类成缓冲器中的不同队列。通过路由器网络和/或由路由器网络处理的数据令牌至少暂时被维持在缓冲器存储器中。通常，被称为链表队列的存储器数据结构被维持与该缓冲器存储器相关联。该链表队列包含分别指向该缓冲器存储器中存储有与每个数据令牌关联的数据的每个存储位置的指针的列表。常规的队列结构通常在一个连续的顺序列表中存储与在该缓冲器存储器中当前存储的每个数据令牌关联的每个指针。当数据令牌进入队列中时，每个队列的状态被更新并且相关链表（linklist）指针被更新。基于输出端口的可用性，调度器选择正确的调度算法以将数据令牌从队列中出队。当数据令牌移出队列时，每个队列的状态和链表指针将然后再次被更新。

[0005] 由于需要基于源端口和目的端口以及业务类型来对令牌分类的步骤，所以在多播业务数据分组的处理中产生了附加延迟时间。首先，与数据分组关联的数据令牌随后被分类到不同的队列中，并且然后在进入队列中时，该队列的状态被更新。然后，基于端口可用性，调度器选择调度算法来将令牌从缓冲器存储器中的对应队列中出队。这一步骤等待端口可用性必须重复用于待发送的数据令牌中的每个数据令牌。另外，如果需要数据令牌的副本被重复并且发送至额外的端口，则需要进行服务质量（QoS）算法检查。此外，当为了例如高优先级业务（诸如控制业务或者语音/视频延迟敏感业务、传输）需要保证优先服务时，必须在传输该数据令牌时进行一些额外的检查。

[0006] 缓冲器管理器的性能受到入队和出队处理操作中的数据令牌的流水线流量的限制。当缓冲器充满时，该缓冲器管理器必须等待直到数据令牌被出队才能将附加数据令牌入队。为了降低这里的延迟时间，引入并联缓冲器以使得附加数据令牌能够入队。然而，这样的附加并联缓冲器使用附加资源和处理功率。此外，每当缓冲器存储器要在由路由器执行的某种操作中处理时，必须访问或者寻址该队列以便获得与该令牌关联的指针。

[0007] 随着与路由器或者其他分组交换机关联的处理速度增加(例如每秒10吉比特或者更高),与用来维持队列的存储器关联的输入和输出带宽和访问延迟变得至关重要。也就是说,鉴于每次需要令牌指针时都必须访问常规的队列的事实,队列存储器可能变为显著瓶颈。

发明内容

[0008] 因此,有利的是,提供一种使用地址链表的缓冲器存储器管理,其通过使用虚拟队列技术、基于提供多个入队和出队令牌缓冲器操作的链表能够增加在当前设备中可以执行的入队和出队操作的当前限制数目。

[0009] 本公开内容的实施例提供了一种用于使用多个列表类型子队列形成虚拟队列的链表的技术,该子队列列表的实现使得能够并联调度多个数据令牌传输,从而显著增加路由器中的交换机制的吞吐量。

[0010] 因此,本公开内容的实施例采用将去往存储器缓冲器的数据排队的方法,所述方法包括从多个队列中选择队列并且从所选择的队列接收数据令牌。队列模块从缓冲器管理器请求用于由缓冲器管理器分配的用于存储数据令牌的地址的指针和地址。然后,通过缓冲器管理器访问存储器列表,并且在存储器列表中生成指向所分配的地址的指针和地址。所访问的存储器列表包括用于附加地址分配的多个链接的存储器列表。此外,向所访问的存储器列表中写入用于所分配的地址的指针。该指针将所分配的地址链接在一起。该方法还包括:在从所述队列接收到随后的数据令牌时向其他存储器列表移植附加地址分配,并且生成将所述其他存储器列表中所分配的地址链接在一起的附加指针。

[0011] 根据本公开内容的另一实施例采用将去往存储器缓冲器的数据排队的装置,所述装置包括:用于从多个队列中选择队列以及从所选择的队列接收数据令牌的选择器模块;响应于所述模块以用于请求由所述管理器分配的用于存储所述数据令牌的地址的指针和地址的管理器;响应于所述管理器以访问存储器列表并且在所述存储器列表中生成指向所分配的地址的指针和地址的分配模块。所访问的存储器列表包括用于附加地址分配的多个链接的存储器列表。所述分配模块向所访问的存储器列表中写入用于所分配的地址的指针,其中所述指针将所分配的地址链接在一起,并且所述分配模块在从所述队列接收到随后的数据令牌时向其他存储器列表移植附加地址分配,并且生成将所述其他存储器列表中的所分配的地址链接在一起的附加指针。

[0012] 根据本公开内容的又一实施例,所述队列模块包括所述队列模块包括用于将数据令牌向所链接的存储器列表的所分配的地址中入队的入队模块以及用于将数据令牌从所链接的存储器列表的所分配的地址出队的出队模块。所述入队模块被配置成向所述缓冲器管理器生成请求以用于将多个数据令牌一次性入队,其中所述多个数据令牌具有通过指针跨所述多个存储器列表链接的地址。所述出队模块被配置成向所述缓冲器管理器生成请求以用于将多个数据令牌一次性出队,其中所述多个数据令牌具有通过指针跨所述多个存储器列表链接的地址。

[0013] 本发明内容必然包含对细节的简化、概括和省略。因此,本领域技术人员应当理解,本发明内容仅是说明性的,而非意在以任何方式限制。仅由权利要求来限定的本公开内容的其他方面、发明特征和优点将在以下给出的非限制性详细描述中变得清楚。

附图说明

[0014] 通过阅读以下详细描述并且结合附图,将能更好地理解本公开内容的实施例,在附图中相似的附图标记指代相似的元件,并且其中:

[0015] 图1是描绘根据本公开内容的实施例的示例性路由器中的缓冲器管理器的配置的框图;

[0016] 图2是根据本公开内容的实施例的缓冲器管理器的框图,该缓冲器管理器描绘示例性路由器中的队列尾指针和队列头指针的配置;

[0017] 图3图示根据本公开内容的实施例的示例性虚拟队列链表配置;

[0018] 图4图示根据本公开内容的实施例的示例性队列链表配置;

[0019] 图5A、图5B图示根据本公开内容的实施例的具有链表结构的示例性头指针和尾指针配置的框图;

[0020] 图6A、图6B图示了根据本公开内容的实施例的将数据令牌在多个存储器列表结构中链接在一起的框图;

[0021] 图7是示出根据本公开内容的实施例的具有链表结构的缓冲器管理器操作的流程图;

[0022] 图8是根据本公开内容的实施例的缓冲器管理器的框图,该缓冲器管理器描绘示例性路由器中的队列尾指针和队列头指针的配置;以及

[0023] 图9是描绘根据本公开内容的示例性路由器中的缓冲器管理器模块的配置的框图。

具体实施方式

[0024] 现在将详细参考本发明的优选实施例,其示例在附图中图示。虽然本发明结合优选实施例来描述,然而应当理解,其并非意在将本发明限于这些实施例。相反,本发明意在覆盖备选、修改和等同方案,这些备选、修改和等同方案可以被包括在如所附权利要求所限定的本发明的精神和范围内。此外,在对本发明的实施例的以下详细描述中,提出了大量具体细节以便提供对本发明的透彻理解。然而,本领域普通技术人员应当认识到,可以在没有这些具体细节的情况下实践本发明。在其他情况下,没有详细描述公知的方法、过程、部件和电路,以避免不必要地混淆本发明的实施例的各个方面。示出本发明的实施例的附图是半图解性的并且并未按比例绘制,特别地,其中一些尺度是为了清楚地呈现并且在附图中被放大示出。类似地,虽然附图中的视图出于便于理解的目的而通常示出了相似的方位,然而附图中的这一描绘在极大程度上是任意的。通常,本发明可能在任何方位中操作。

[0025] 然而应当铭记,所有这些术语和相似的术语要与适当的物理量关联并且仅是适用于这些物理量的方便的标记。除非清楚地指出或者根据以下讨论能够清楚地看出,否则应当理解,遍及本发明,使用诸如“处理”或“访问”或“执行”或“存储”或“渲染”等术语的讨论是指将被表示为计算机系统的寄存器和存储器以及其他计算机可读介质内的物理(电子)量的数据操纵和转换成被表示为计算机系统存储器或寄存器或者其他这样的信息存储、传输或客户端设备内的物理量的数据的计算机系统或类似的电子计算设备的动作和处理。当若干实施例中出现某个部件时,相同附图标记的使用表示该部件与原始实施例中图示的部件相同。

[0026] 入队和出队缓冲器存储器管理

[0027] 图1示出了在其中可以实现本公开内容的数据令牌处理系统100。该系统100包括处理器110, 该处理器110具有复制、QoS及入队处理器120、缓冲器管理器130、调度出队140和缓冲器存储器150。该处理器110被配置成在从其接收数据令牌和其他数据令牌的网络与去往由调度出队140控制的输出端口160的网络之间提供接口。具有复制、QoS及入队处理器120、缓冲器管理器130、调度出队140和缓冲器存储器150的处理器110可以被实现为例如安装在路由器或交换机的线路或端口上的一个或多个集成电路。该数据令牌可以被认为但是不限于待传输的数据分组的代表部分。例如, 数据分组可以具有头部和净荷部二者。该数据令牌将通常包含数据分组的头部或者元数据的收集或者与头部关联的各方面或者与整个数据分组关联的各方面。另外, 如果该头部在64位至128位之间, 则所需的数据令牌取决于网络需要可以是45-50位。该数据令牌可以与指针所指的数据分组关联, 并且该数据分组可以被存储在由耦合至该数据分组的关联的指针所指的外部存储器块中。

[0028] 应当理解, 图1所示的系统元件的特定布置仅是说明性示例。更特别地, 如先前所指出的, 本公开内容可以根据任何类型的分组或令牌处理器来实现并且不限于任何特定分组处理应用。

[0029] 图2示出了根据本公开内容的实施例的缓冲存储器管理器200以及缓冲管理操作210中的队列存储器中的队列尾指针240、队列头指针250和空指针260。应当理解, 链表指针240至260可以在处理器120和140(图1)的控制下驻留并且被维持在缓冲器管理器130(图1)中。每个队列具有队列头指针250和队列尾指针240二者。该队列尾指针240规定写入操作的位置并且该队列头指针250规定读取操作的位置。在执行读取操作时, 该队列头指针250将沿着每个新的读取操作的队列移位。同样地, 该队列尾指针240将沿着每个新的写入操作的队列移位。因此, 上一队列尾指针240标识周期中的上一写入操作, 上一队列头指针250标识周期中的上一读取操作。

[0030] 如图2所示, 在根据队列尾指针请求220访问特定队列时, 读取新的队列头指针250和新的队列尾指针240。接收队列的令牌。在接收该令牌之后, 可以访问以读取和处理关于下一头指针的信息。

[0031] 图3示出了根据令牌被接收的顺序被填入的所选择的 Q_0 的并联链表配置。初始请求的指针可以是队列头指针或者队列尾指针。每个 Q_0 将被分成4个链接的链表 Q_{00} 至 Q_{03} 。因此, Q_{00} 可以被分成4个存储器位置 Q_{00} 至 Q_{03} , 其是图3中示出的列表 Q'_{000} 至 Q'_{002} 。 Q_{00} 至 Q_{03} 中的单个链表被进一步再分成4个子链表 Q_{00} 至 Q_{03} 。这使得例如在从 Q_0 到现在的一个读取周期中包括4个存储位置读数 Q_{00} 至 Q_{03} 或者每个 Q_{00} 读数 $N/4$ 个读取周期。换言之, 将每个 Q_0 的附加的 Q_{00} 至 Q_{03} 链接在一起实现了在单个钟控周期 Q_0 内接收附加数据令牌集合的能力, 这些附加数据令牌集合初始在最高达4个周期的时间内在没有对附加链表的链接的情况下已经被读入。

[0032] 图4图示了对于 Q_{00} 而言为 $N/4$ 的每个 Q_0 的读取周期。第一钟控周期可以包括 Q_{00} 、 Q_{01} 、 Q_{12} 和 Q_{00} 。下一钟控周期移位 $N/4$ 在 Q_{01} 、 Q_{12} 、 Q_{00} 和 Q_{01} 处开始。随后的周期以 Q_{12} 开始每个移位 $N/4$ 。因此, 每个随后的读取和写入周期被执行 $N/4$ 次移位。因此, 在特定的 Q_0 钟控时间内, 存在大约最多达 $N/4$ 的4个钟控寄存器移位以缩短每个寄存器移位的周期距离。

[0033] 在图5A中, 在选择队列之后, 执行读取请求并且需要将数据令牌505出队。调度器

响应于该读取请求弹出所选择的队列中的数据令牌。所请求的数据令牌由缓冲器管理器放置在存储器组的集合的第一可用存储器位置中,在这一实例中,存储器组₀是被表示为所请求的数据令牌的所分配的地址。头指针520由该缓冲器存储器的链表类型管理器应用选择以指向所请求的令牌505.该存储器位置由该缓冲器管理器标记为Q₀₀₀。该缓冲器管理器的内部时钟应用针对正被请求的数据令牌分配4元组的存储器位置。在接收到数据505的第一令牌之后,可以查明第一钟控周期Q₀的开始点。这一开始点是初始头指针510被指向的地方并且在初始令牌请求505被读取和处理之后被再次识别。

[0034] 每个存储器位置的集合Q_{00N}被分配上至4个存储器位置以用于从所选择的队列接收的数据令牌。下一存储器位置经由尾指针515被链接至Q₀₀₀。缓冲器管理器发送下一读取请求并且从队列接收第二数据令牌。该第二数据令牌被动态分配至被表示为Q₀₀₁的组₁中的可用存储器位置。从该缓冲器管理器接收的下一指针515变成存储器位置Q₀₀₁的随后的头指针。这一将头指针和尾指针耦合的菊花链实现了数据令牌从跨存储器组的离队操作的两步过程。例如在开始515和550处的两个数据令牌是从队列请求的初始请求数据令牌Q₀₀₀ 515和Q₀₀₁ 550。令牌Q₀₀₀ 515和Q₀₀₁ 550二者在涉及Q₀头指针510的初始读取的两步过程中出队并且处理来自Q₀₀₀位置的数据令牌。在处理Q₀₀₀之后,接着可以读取下一尾指针515并且由缓冲器管理器标识该下一尾指针515。即,在请求Q₀时,读取Q₀₀₀,并且接收Q₀₀₀以及然后接收与Q₀₀₀关联的数据,并且标识到下一Q₀₀₁的路径。还标识从Q₀₀₁下一尾指针550至Q₀₀₂和随后的下一指针565的路径。另外,在请求Q₀被公开(transpire)时,用于发送与Q₀关联的对象集合的位置已知。另外,图5示出了用于数据令牌525和570的附加集合的链路的其他路径菊花链。存储器分配Q₀₃₀被链接至Q₀₃₁和随后的指向组₂的地址Q₀₃₂的尾指针535。地址Q₀₂₀的存储器列表分配经由其尾指针被连接至Q₀₂₁ 540。附加分配Q_{01N} 545在随后周期中的存储器列表中呈现附加分配。

[0035] 随着缓冲器管理器继续发送读取请求,创建4个链接的存储器位置的元组或者集合。在完成4个链接的位置的第一集合时,缓冲器管理器开始下一个4个链接的位置的集合并且继续创建4个存储器位置的元组或者集合的这一过程直至所有数据令牌从所选择的队列中被弹出。如果有奇数个数据令牌或者数据令牌不足以填充在上一集合中可能发生的待填充的4个链接的位置的集合,则在列表中将有空位置。

[0036] 接着,第三存储器位置通过尾指针550类似地耦合至第二存储器位置Q₀₀₁。该尾指针550现在是第三存储器位置Q₀₀₂的头指针。第三存储器位置在组₂中通过Q₀₀₂来表示,并且存储器位置中的每个存储器位置被动态地表示并且在跨多个存储器组迁移的循环方式中分配。这一循环规定可以将数据在给定时间中转发到所标识的队列外部的带宽输出或者按照网络交换算法可以从给定队列出队的数据的频率。例如,在图5A中,从Q₀₀₀ 515到Q₀₀₁ 550到Q₀₀₂ 565的循环配置被示出为链接存储器组列表1和2的集合。每个存储器组列表510、530和555包括在路径中被链接在一起的令牌列表Q₀₀₀至Q_{03N},使得每个令牌能够与Q₀的集合相关联。

[0037] 图5B用框图图示了队列尾指针585以及队列尾指针590的相关子块Q₀₀至Q₀₃。还示出了队列头指针586以及队列头指针595的相关子块Q₀₀至Q₀₃。队列尾指针590的堆叠的子块和队列头指针595的堆叠的子块中的每个子块在图5的组1和2所示的并联链表中被链接在一起。

[0038] 图6A和图6B图示了针对由4个并联链表组成的单独队列的队列100并联链表的示例的框图。应该注意,如果从图6A的动态链表的示例性实施例添加或者去除一个或多个并联链表,则本公开内容的示例性实施例的概念将不改变。

[0039] 图6A在框图600中图示了针对由4个并联链表组成的单独队列的队列100并联链表的示例:Q100-0,第一并联链表605;Q100-1,第二并联链表610;Q100-2,第三并联链表615;和Q100-3,第四并联链表620。块的子列表之间的外部虚拟耦合625、630图示了链表Q100-0₁到Q100-1₁到Q100-2₁到Q100-3₁并且回到Q100-0₁,以及Q100-0₂到Q100-1₂到Q100-2₂到Q100-3₂并且回到Q100-0₂的循环配置中的第一和第二耦合集合。该耦合包括一起链接至4个数据令牌的集合的尾指针和头指针,其中上一耦合625、630共同用作集合中的上一数据令牌的尾指针和下一数据令牌的头指针。内部耦合Q100-0₁到Q100-0₂到Q100-0₃表示按照连续顺序的各个数据令牌集合的各个并联列表。因此,数据令牌集合1的上一尾指针被连接至下一数据令牌集合2,并且集合1的上一尾指针用作数据令牌集合2的第一头指针。在图6A的示例中,因为有链表Q100-0₁到Q100-1₁到Q100-2₁到Q100-3₁并且回到Q100-0₁,以及Q100-0₂到Q100-1₂到Q100-2₂到Q100-3₂并且回到Q100-0₂的数据令牌的两个完整集合,所以有由尾指针525和630生成的、指向数据令牌集合2和3的两个头指针。

[0040] 另外,在图6A中示出了将存储器列表分配入队和弹出并且将4个数据令牌集合的存储器分配的多个集合入队的整个链表循环类型周期或者循环过程。子列表605、610、615和620之间的数据令牌的第一集合的耦合在图6A中被表示为“1-1-1-1”。缓冲器管理器依赖于外部计数器应用来跟踪4个集合中的每个数据令牌集合,并且在缓冲器存储器中创建虚拟表以存储具有关联的头指针和尾指针的所有计数值。另外,数据令牌可以具有与用于子列表中的4个集合中的每个数据令牌集合的附加数据集合关联的附加指针,从而允许在外部位置处的数据存储。在完成初始的4个集合时,缓冲器管理器增加其内部计数并且表示为下一集合“2-2-2-2”。重复这一过程直至所有数据令牌从所选择的队列入队或者出队。缓冲器管理器使用内部计数器应用跟随4个集合的块的FIFO类型排序,并且每个集合还按照如图6A的示例中示出的1-2-3集合的数字顺序耦合。

[0041] 图6B在框图中图示了针对队列100的业务类别4和端口4中的特定目的(未示出)生成的队列列表的示例。示出了并联存储器子列表635、640、645和650。例如,用于Q100的队列尾指针存储器是Q100-0,其在被给予当前指针1000的存储器子列表635中。另外,从存储器读取的空指针是1000并且被写入至用于队列100-0的队列尾存储器。因此,用于Q100-0的队列尾存储器具有当前指针1000和下一指针1100。下表1.2中还示出了具有当前指针和下一指针的“尾请求”的每个的列表。这些队列尾请求、当前指针和下一指针中的每个构成内部并联链表。存储器分配在Q100-0中是随机的并且是按照可用性动态选择的。因此,存储器列表中的排序根据存储器位置的可用性而改变。下表1.2中示出了生成的动态列表,其中在入队过程期间,缓冲器管理器向4个数据令牌的第一集合分配当前指针1000和然后的下一指针1100用于4个数据令牌的下一集合。图6B中示出了尾指针1000的内部计数器和计数器0。由从1000-0至2000-1的尾指针的耦合示出了第一集合的数据令牌的下一指针。计数器被更新到1并且子列表Q100-1中的当前指针是2000。在子列表Q100-2中,当前指针是3000伴随着计数器2,并且在子列表Q100-3中,当前指针是4000伴随着计数器3。子列表635-650中的每个子列表被分配有当前指针,因为缓冲器管理器迁移至每个列表并且更新其内部计数器,

从而创建用于4个数据令牌的每个集合的头指针和尾指针的集合。在通过子列表中的每个子列表迁移之后,缓冲器管理器将空指针返回655、660作为数据令牌的下一集合的下一当前指针。使用这一循环方式,所有子列表被用于将数据令牌存储在4个集合中,并且每个集合的上一尾指针边长下一集合的随后的头指针。因此,初始Q100-0链表635的计数器值被表示在0、4、8等的4个集合中,其对应于向数据令牌的每个集合分配的指针的集合。

[0042] 表1.1

[0043] 入队与缓冲器管理器交互

[0044] Qtail Teq-100-0->当前指针-1000,下一指针-1100

[0045] Qtail Teq-100-1->当前指针-2000,下一指针-2100

[0046] Qtail Teq-100-2->当前指针-3000,下一指针-3100

[0047] Qtail Teq-100-3->当前指针-4000,下一指针-4100

[0048] Qtail Teq-100-4->当前指针-1100,下一指针-1200

[0049] Qtail Teq-100-5->当前指针-2100,下一指针-2200

[0050] Qtail Teq-100-6->当前指针-3100,下一指针-3200

[0051] Qtail Teq-100-7->当前指针-4100,下一指针-4200

[0052] Qtail Teq-100-8->当前指针-1200,下一指针-1300

[0053] 图7是图示由图2的缓冲器管理器操作执行的示例性步骤的流程图。另外,示出了根据图7的流程图的步骤的用于队列100的连续的分组业务的表示。在步骤705,基于目的端口和与数据关联的业务的类别来选择队列。在步骤710中,(图1的)入队模块120向(图1的)缓冲器管理器130请求指针以在(图1的)缓冲器存储器150中存储数据。例如,关于图6A和6B中的所标识的队列100,入队模块可以请求用于队列100的指针位置。在步骤715,缓冲器管理器检查队列尾指针存储器以找到所识别的队列并且将当前指针设置为其位置。另外,缓冲器管理器从空指针存储器弹出附加的一个空指针并且将其写入队列尾指针存储器并且还将这一空指针标识为到入队模块的下一指针。

[0054] 在步骤720中,入队模块使用当前指针来将令牌和下一位置写入缓冲器存储器中。该入队模块使用位置1000来写入令牌和1100作为如表1.1所示的下一存储位置。此外,对于下一队列尾请求100-1,入队模块使用在存储位置3000处的当前指针和在存储位置3100处的下一指针。表1.1图示随后的队列尾请求以及当前指针和下一指针位置。

[0055] 在步骤725,在将队列调度为要出队之后,出队模块向缓冲器管理器请求队列的队列头指针。作为示例,当队列100被调度为要出队时,该出队模块请求待发送至缓冲器管理器的队列100的头指针1000。如图表1.2所示,在步骤730中,缓冲器管理器向所请求的队列分配头指针值1000。

[0056] 步骤735,图示了在出队模块从缓冲器存储器读取分组和下一指针并且然后向缓冲器管理器转发下一指针作为头指针更新时的出队模块操作。该出队模块从缓冲器中的其接收令牌和下一指针1100的位置1000读取。该出队模块发送1100作为队列100的头指针更新请求。指针1000现在为空指针,并且该出队模块向缓冲器管理器转发空指针。(出队操作参考下面的图表1.2)

[0057] 出队与缓冲器管理器交互

[0058] 表1.2

- [0059] Head Req-100->头指针-1000
- [0060] Head update-1100
- [0061] Head Req-101->头指针-2000
- [0062] Head update-2100
- [0063] Head Req-102->头指针-3000
- [0064] Head update-3100
- [0065] Head Req-103->头指针-4000
- [0066] Head update-4100
- [0067] Head Req-100->头指针-1100
- [0068] Head update-1200
- [0069] Head Req-101->头指针-2100
- [0070] Head update-2200
- [0071] Head Req-102->头指针-3100
- [0072] Head update-3200
- [0073] Head Req-103->头指针-4100
- [0074] Head update-4200
- [0075] Head Req-101->头指针-1200
- [0076] Head update-1300

[0077] 当出队模块开始队列100的调度的队列操作时,这一操作由调度器在跨多个队列同样地分离的每个 Q_0 在4个钟控周期内执行。该调度器依次请求队列100、101、102和103的头指针信息,使得按照调度器请求的优先级而执行出队操作。当由调度器每4个时钟在调度的队列100中执行将队列出队时,整个 Q_0 可以被视为访问通过4个不同的并联链表被链接在一起的数据的 Q_0 。因此,对于每个单个并联链表,该出队操作可以在周期内出现直至16个时钟的最大速度。对于吞吐量和延迟限制的请求,并联链表出队可以从4增加至8和以上。

[0078] 图8为类似于图2的框图,但是图示了入队和出队操作的步骤。在图8中,接收输入令牌(1)以通过入队模块820进行分类。接着,输出(3)向缓冲器存储器830请求新的队列尾指针。用于并联链表(2)的队列尾存储器被发送至存储器(4)用于向缓冲器存储器850中的写入操作。该数据令牌被写入缓冲器存储器850。该调度器840挑选 $Q(Q_0)$ (5)并且请求并联链表的队列头指针(6)。该缓冲器管理器读取队列头指针并且使用这一头指针作为经更新的头指针并且然后向并联链表写入该经更新的头指针。

[0079] 在图9中,图示了根据本公开内容的实施例的当输入的令牌被接收和分类为 Q_0 时由缓冲器管理器执行的示例性过程。对新的队列的尾指针(3)的请求被发送至队列尾指针模块920。伴随着每个请求生成并联链表的队列尾指针,并且当接收到令牌时,对于该令牌执行向缓冲器存储器中的写入操作。对于调度器操作,在 Q_0 被选中时,请求要向队列头指针模块930发送的队列头指针(6)。对于并联链表的队列头指针被发送自并联链表中的每个并联链表。所请求的队列头指针被读取并且与队列已更新的头指针和现在的空旧指针一起转发(8)与头指针关联的令牌。

[0080] 因此,如本公开内容的示例性实施例所示,通过将4个并联的子链表链接在一起,使得所选择的 Q_0 的给定N周期内的入队和出队操作、读取写入操作最小化。此外,根据本公

开内容的路由过程有利地提供了入队和出队操作的有限的缓冲器空间分配的增加的吞吐量,从而降低了缓冲器管理器在这一瓶颈操作所浪费的延迟时间。通过将队列分配划分为多个虚拟子队列以用于入队和出队,在降低这样的入队和出队操作中产生的总体延迟时间时,可以更快地获取并且传输数据令牌而不对物理缓存器配置作出显著改变。

[0081] 应当理解,路由器中的电路可以被实现为专用集成电路 (ASIC)、专用标准零件 (ASSP)、片上系统 (SoC)、现场可编程门阵列 (FPGA) 等。另外,应当理解,路由器可以包括本领域公知的各种其他功能和部件。

[0082] 虽然本文中已经公开了某些优选实施例和方法,然而本领域技术人员根据以上公开内容应当理解,可以在不偏离本公开内容的精神和范围的情况下做出对这样的实施例和方法的变化和修改。意图在于,本公开内容应当仅限于所附权利要求以及适用法律细则和原理所要求的范围。

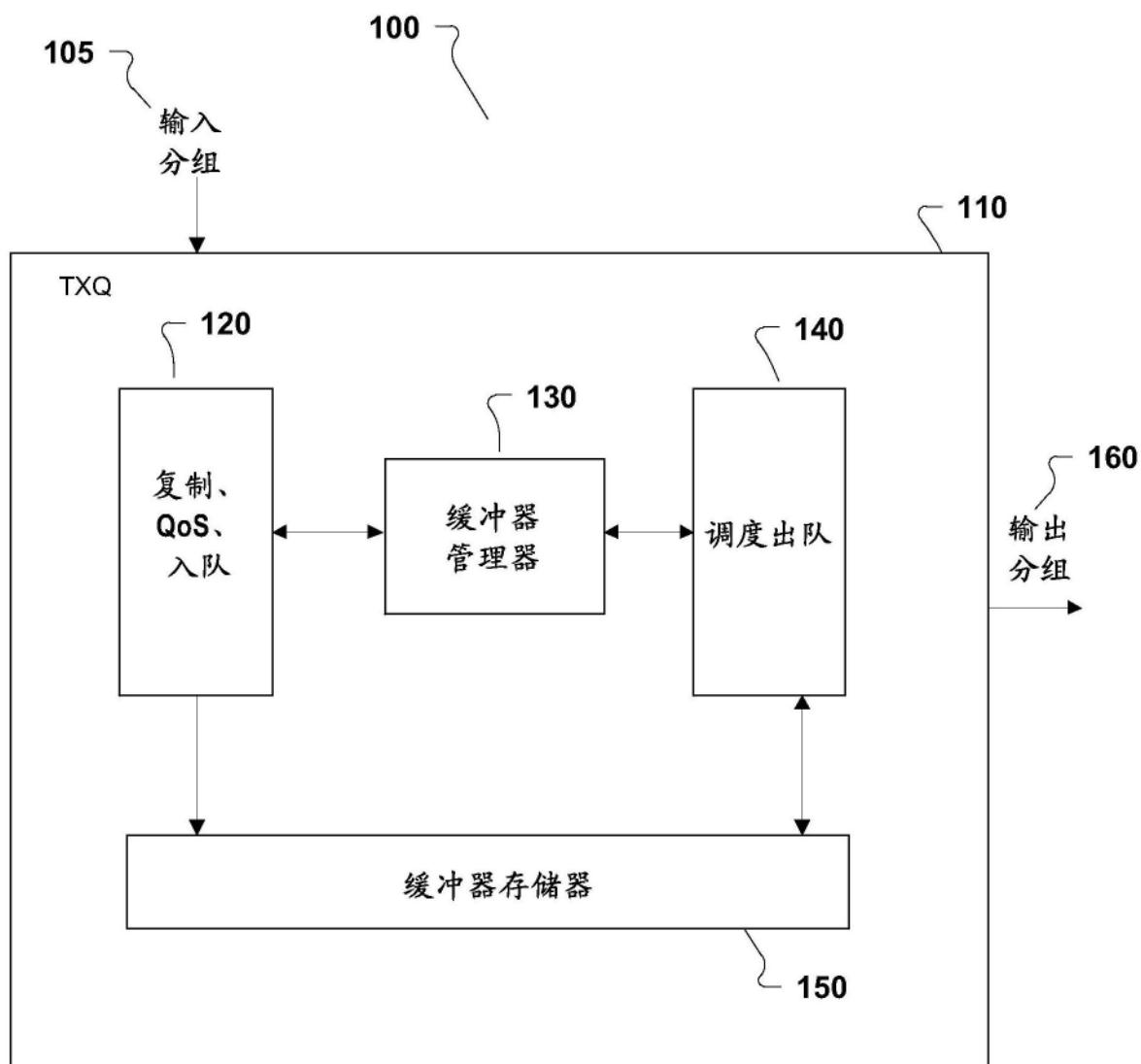


图1

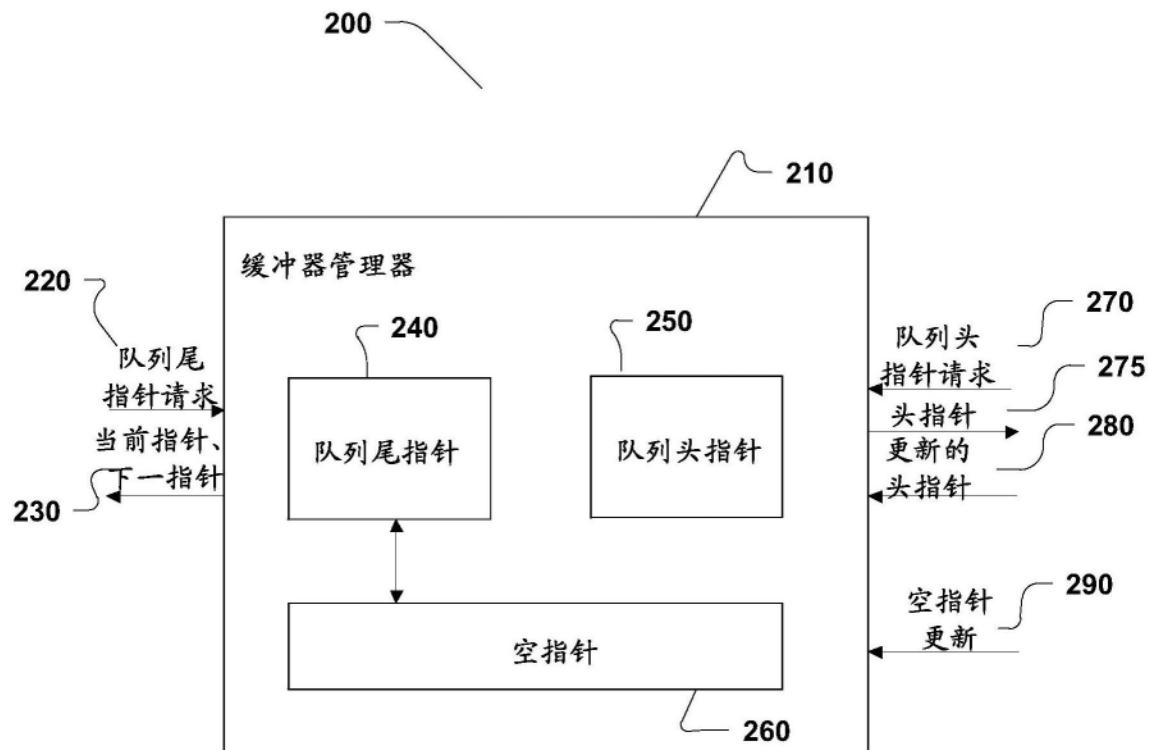


图2

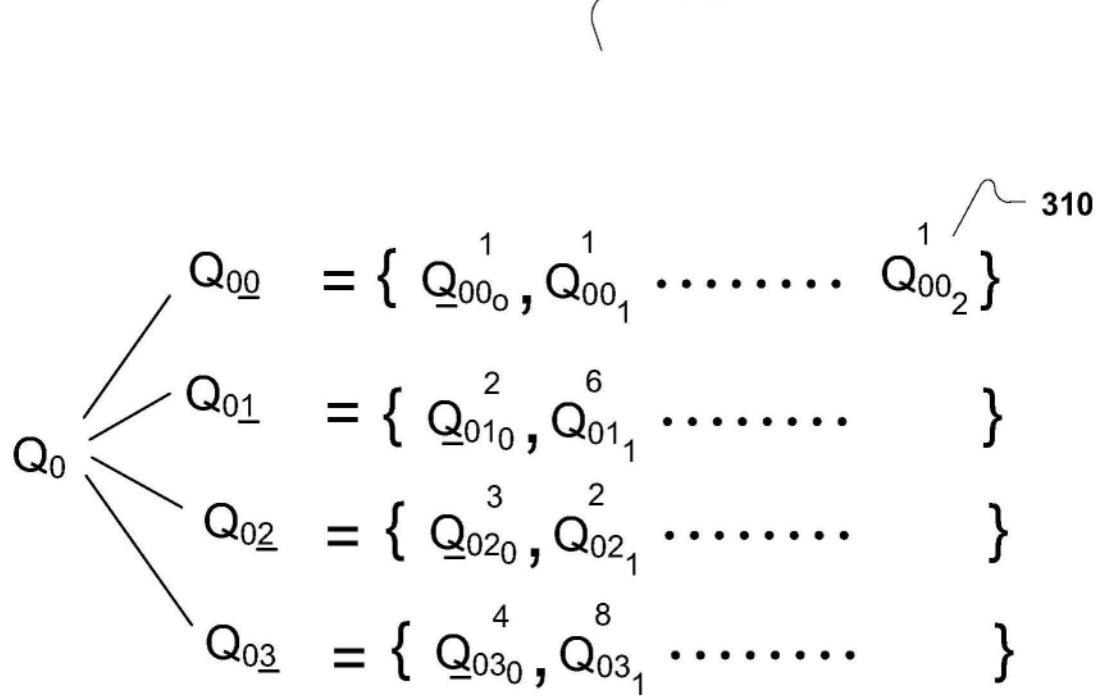


图3

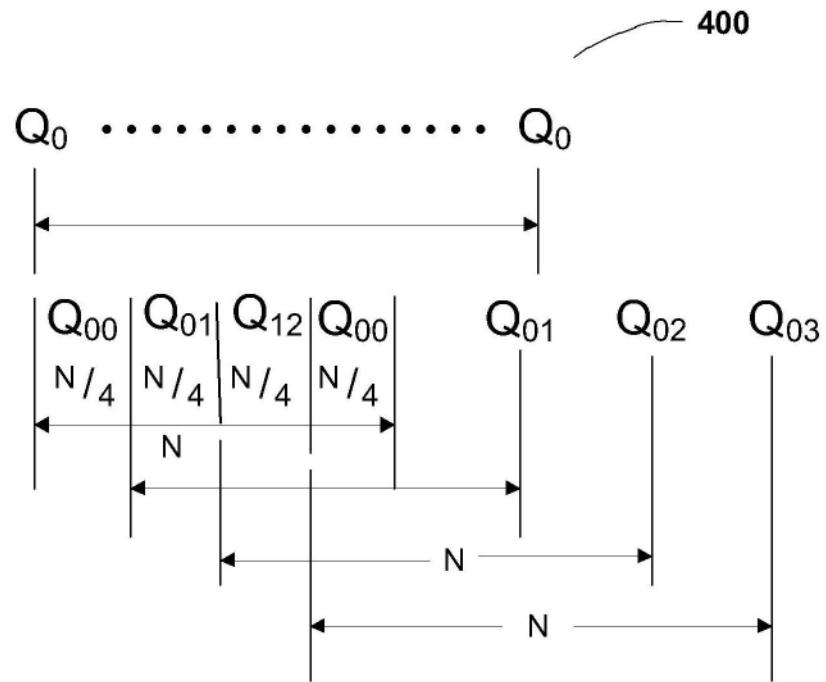


图4

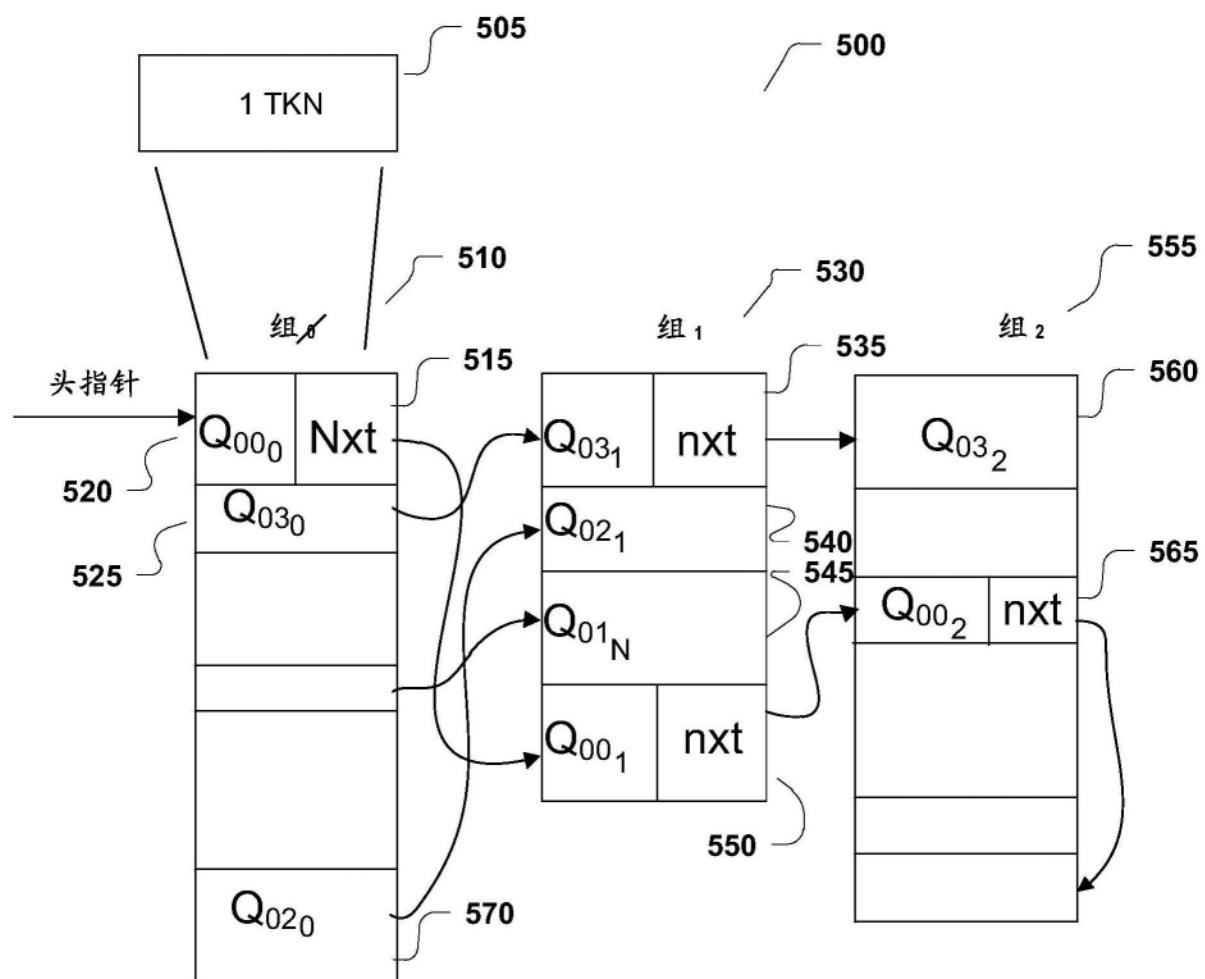


图5A

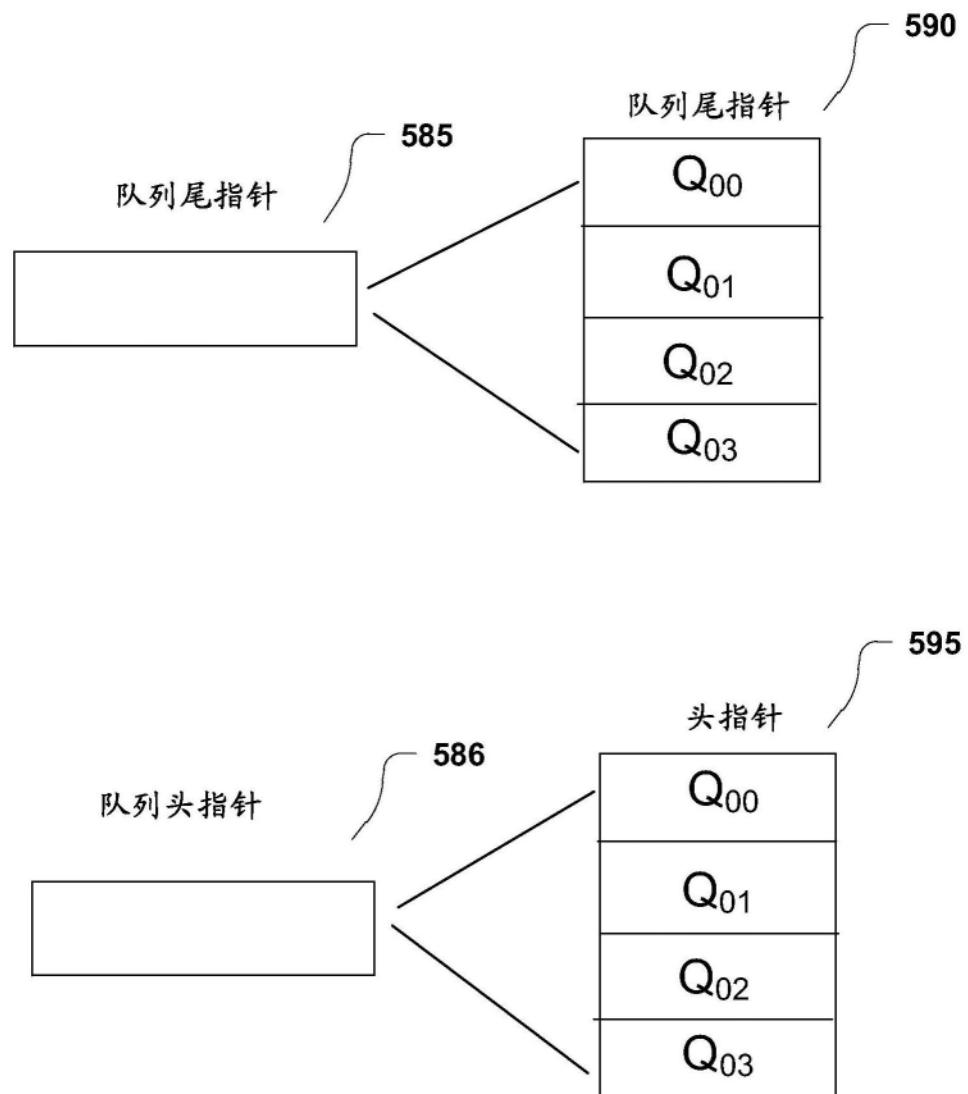


图5B

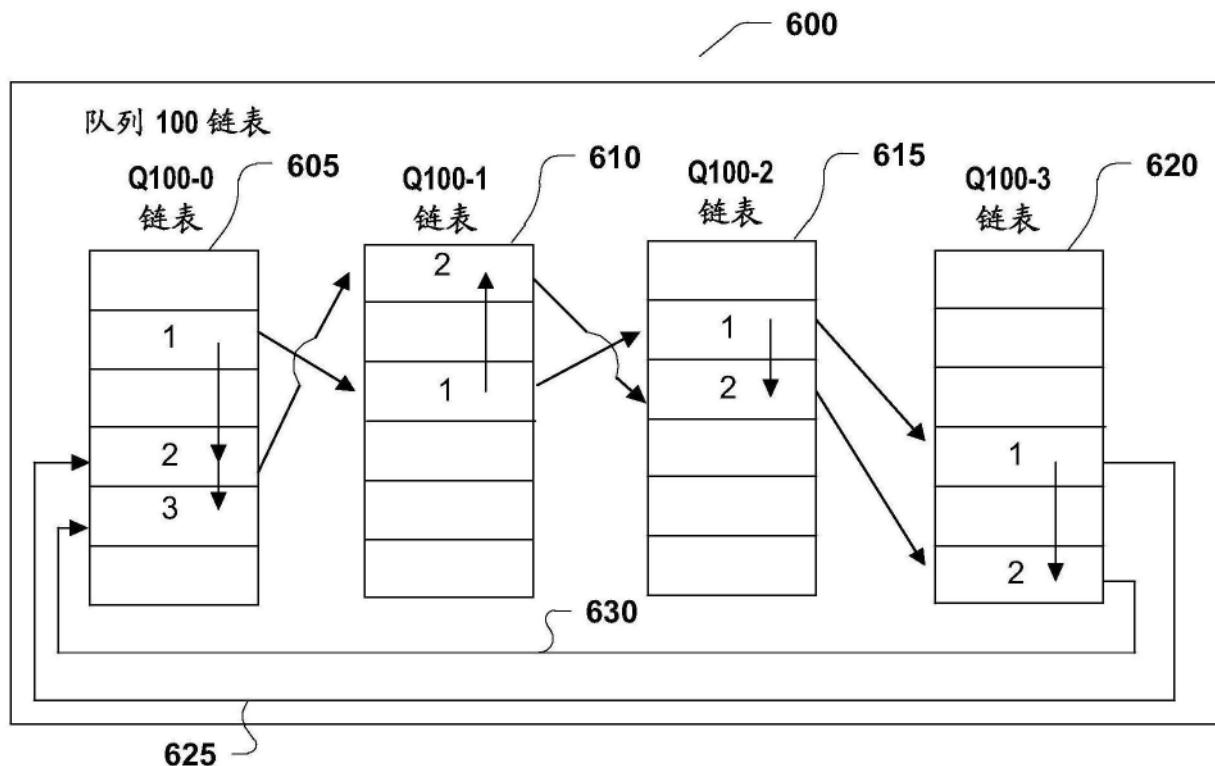


图6A

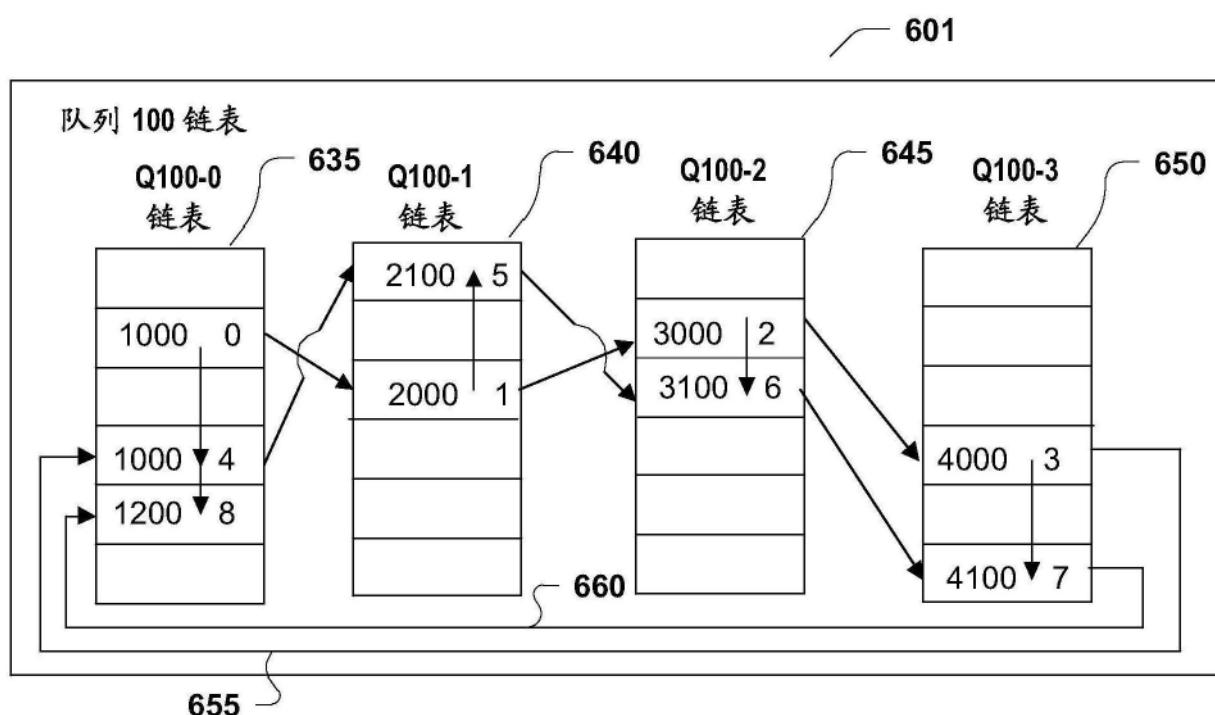


图6B

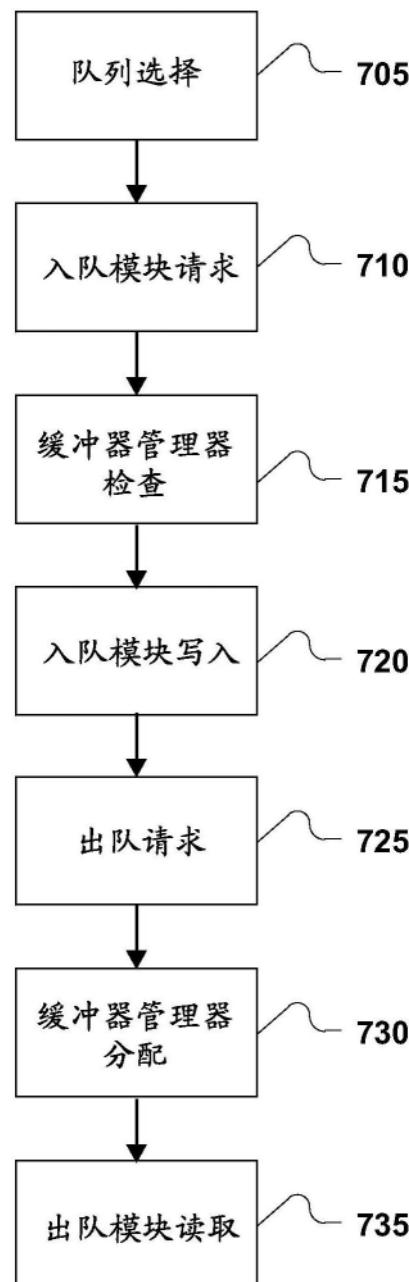


图7

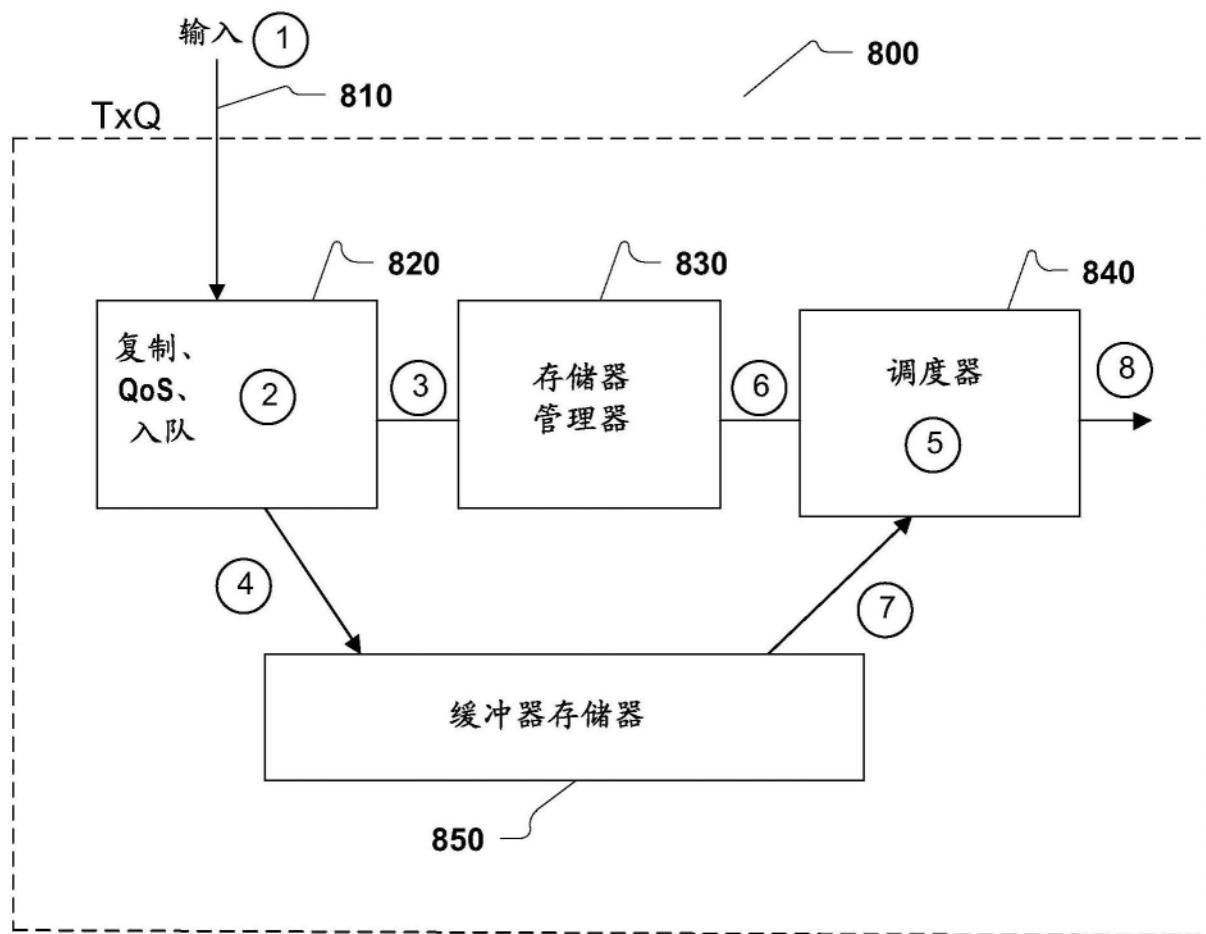


图8

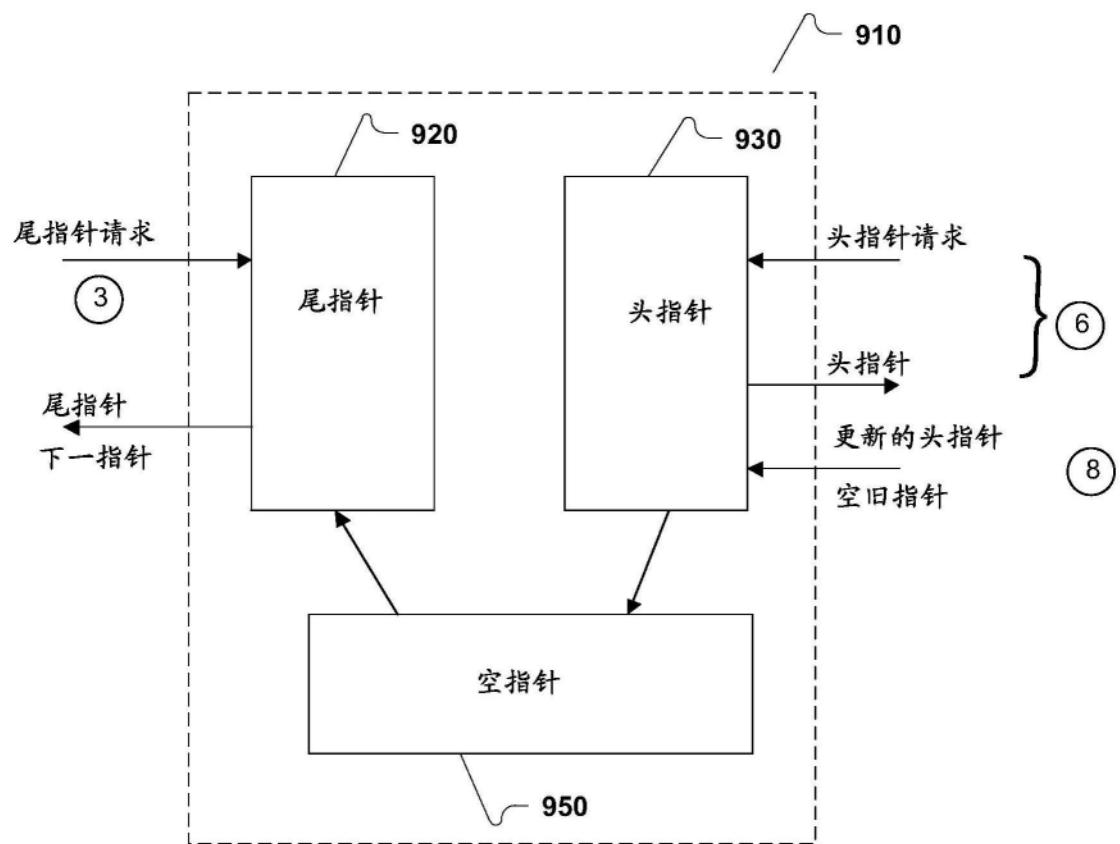


图9