(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117931391 A
(43)申请公布日 2024.04.26

(21)申请号 202311721650.0

(22)申请日 2023.12.14

(71)申请人 天翼云科技有限公司
地址 100007 北京市东城区青龙胡同甲1号、3号2幢2层205-32室

(72)发明人 廖怡 樊小平 符权 刘禄仁

(74)专利代理机构 北京轻创知识产权代理有限公司 11212
专利代理师 王澎

(51)Int.Cl.
G06F 9/48(2006.01)
G06F 9/50(2006.01)
G06F 13/40(2006.01)
G06F 13/42(2006.01)
G06F 13/28(2006.01)

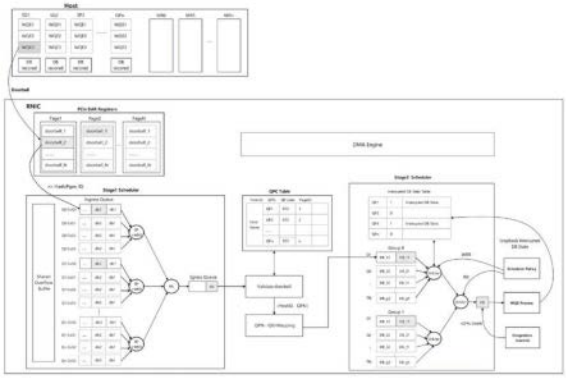
权利要求书4页 说明书9页 附图1页

(54)发明名称

一种基于RMDA的无损高效的数据处理方法及网络接口卡

(57)摘要

本发明公开了一种基于RMDA的无损高效的数据处理方法及网络接口卡,属于数据中心的数据通信领域,在RMDA中当大规模QP并发时,解决不同QP的请求的高效调度处理问题,保证请求信号不丢失、QP请求不乱序、调度周期不浪费,同时结合拥塞控制有效解决不同QP的调度公平性问题,避免大消息阻塞其他的QP处理以及单次调度需要处理的请求数过多,解决头阻问题。



CN 117931391 A

1.一种基于RDMA的网络接口卡,其特征在于:包含PCIe BAR寄存器处理模块、第一阶段调度模块、QPC状态表、DB验证模块、QPN-QID映射模块、第二阶段调度模块、调度策略配置模块、拥塞控制模块、DMA引擎:

其中,PCIe BAR寄存器处理模块,用于负责软硬交互中门铃信号DB的解析和处理;

第一阶段调度模块,用于负责第一阶段门铃信号DB基于PageID和优先级的调度,保证门铃信号DB不丢失;

QPC状态表,缓存QPC状态信息;

DB验证模块,用于验证DB的合法性和对应QP的状态信息是否正确;

QPN-QID映射模块,用于为DB分配进入第二阶段调度模块的入队ID;

第二阶段调度模块,用于负责第二阶段DB基于hostID和QPN的调度,保证不同的Host的不同QP可以得到公平调度;

调度策略配置模块,用于配置第二阶段调度模块中调度级数、每个调度器规模、调度算法等策略的配置;

拥塞控制模块,用于为每个QP分配Credit,控制每个QP在一轮调度周期可以发出的消息大小;

WQE处理模块,用于WQE的预取和WQE的处理,若WQE不能完整处理,则向第二阶段调度模块返回DB处理的断点信息

DMA引擎,用于RNIC和Host直接的数据搬运;

其中,PageID为页ID,表示软件敲DB对应的BAR空间地址的ID;QPC表示QP的上下文信息,用于缓存QP地址信息;QP表示RDMA的连接队列;hostID表示host的序列号ID;QPN表示QP的序列号ID;Credit为信用;WQE表示一个RDMA请求;RNIC表示RDMA网卡。

2.一种基于权利要求1所述的RDMA的网络接口卡的RDMA的无损高效的数据流处理方法,其特征在于:具体包含如下步骤:

步骤1,当目标主机中的SQ中有新的WQE产生,主机会产生一个门铃信号DB发送给RNIC;即通过PCIe接口将DB信息写入RNIC为该QP分配的Doorbell空间;

其中,SQ表示发送端的队列;

步骤2,RNIC根据Doorbell寄存器的地址解析该地址的PageID以及Doorbell的QPN、优先级CoS信息,将Doorbell加入第一阶段的调度器中;第一阶段调度器采用层次调度的结构,支持M个Group,每个Group又可进一步分为4个优先级队列,在每个Group中采用SP+WRR调度算法,不同的Group之间采用RR调度算法;SP为严格优先级调度算法;WRR为加权轮询调度算法;

步骤3,第一阶段调度模块的第一层调度器采用配置好的调度算法选择待调度队列,将待处理的DB加入调度器的输出队列中;

步骤4,第一阶段调度模块的第二级调度器采用RR调度将第一级调度器输出的DB写入输出队列中;

步骤5,DB验证模块从第一级调度器第二级输出队列取出队头的DB进行合法性验证,判断DB的PageID与DB的QPN绑定的QPC中PageID是否一致以及QP的状态是否正常;若一致则将DB输入QPN-QID映射模块,若不一致则将该DB丢弃并返回错误给目标主机;其中,QPN表示QP的序列号ID;QID表示在RNIC中为QP分配的本地ID,在RDMA系统中是唯一的;QPN-QID映射模

块用于QPN和本地QID的映射查找,通过QID作为本地QP Context的索引;

步骤6,DB输入QPN-QID映射模块后,根据DB中的HostID查找其对应的GroupID,根据QPN映射到对应GroupID的输入队列中,保证同一个HostID的QP和DB放在同一个调度组中;

其中,GroupID表示调度组的ID,不同的HostID可以放在不同的调度Goup;

其中,HostID表示主机的ID,若在支持虚拟化的场景中,一个VM对应一个独立的HostID;

步骤7,第二级调度模块中各个层级调度器将DB输出至模块最后的输出队列;

步骤8,WQE处理模块从输出队列的队头取出DB,并根据QPN读取处理WQE所需的QPC状态;在QPC中,包含Max_Burst_Size、Max_Batch_WQE_count,分别表示单个WQE可以发送的最大消息数和一次可以处理的WQE的最大数量;Max_Burst_Size表示单次调度周期允许发送的最大字节数;Max_Batch_WQE_count表示单次调度周期允许获取的WQE最大数量,也即单次调度处理周期允许处理的最大消息个数;

步骤9,WQE处理模块向拥塞控制模块请求Credit;其中,Credit是由拥塞控制算法为每个QP分配的可发送消息的大小;

步骤10,WQE处理模块根据Max_Batch_WQE_count和当前可缓存的WQE数量WQE_Available_Count,通过DMA引擎向SQ获取不超过不超过N个WQE,其中,N＝min(WQE_Available_Count,Max_Batch_WQE_count);WQE_Available_Count表示最大可用的WQE数量;N表示单次调度周期可以处理的WQE个数,其中,N取WQE_Available_Count和Max_Batch_WQE_count的最小值;

步骤11,WQE处理模块逐个处理缓存的WQE,每处理一个WQE即更新QPC中WQE的消费指针,并更新Credit值,Credit值为当前的值减去WQE消费的Credit值;若在处理WQE的过程中,剩余的Credit不足以处理一个完整的WQE,将在剩余Credit消费后,将WQE处理的中断的状态返回,缓存在第二调度模块中的缓存在Interupted DB State表中,并将该队列对应的bitmap置1;其中,所述Interupted DB State包括HostID、QPN和Produce_Index、Target_WQE_Index和Walk_Offset,Produce_Index表示当前处理的WQE_Index,Walk_offset表示当前未处理完的WQE已经发送的数据指针,Target_WQE_Index表示当前的DB需要处理到的WQE位置;Credit为令牌值,表示当前该QP发送的字节数;Interupted DB State为中断DB信息表,用来缓存DB调度的中断信息;bitmap为位图,该bitmap中每一个bit位与QPN对应,若该QPN信息有效,则将对应的bit位置1,若无效则置0;

步骤12,当第二阶段调度模块在下一个周期调度到某个队列时,先通过Bitmap判断该队列中是否有中断的DB状态,也即该队列对应的bitmap是否为1;若bitmap＝0,则从调度输入队列中读取一个新的DB处理;若bitmap＝1,则优先读取DB中断状态,并将DB中断状态信息组成一个新的DB发送到WQE处理模块;

步骤13,WQE处理模块根据该DB的信息,从Produce_Index指向的WQE的Walk_Offset处开始继续处理MR的数据,依据Credit判断是否可以完全将WQE处理,重复步骤11-步骤13,直到DB不再中断可以全部发出去后将bitmap置0;其中,Produce_Index表示消费指针,表示当前的SQ处理到的位置;Walk_Offset表示单个WQE处理的中断位置的虚拟地址。

3.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:Doorbell加入第一阶段调度器的基本步骤包括:

步骤2.1:第一阶段调度模块根据Hash(PageID)得到目标队列的GroupID,并根据DB中的CoS选择将Doorbell加入对应的优先级队列中;若队列未满,直接将Doorbell写入队尾,若该队列中存在该QPN的Doorbell,可以合并成1个DB;若该队列满了,转至步骤2.2;

其中,CoS表示Channel of Sevice表示优先级通道,通常支持8种优先级;

步骤2.2:第一阶段调度模块将Doorbell信息记录在Overflow Buffer中;Overflow Buffer是所有第一阶段调度队列的共享缓存;在Overflow Buffer中为每个GroupID的每个优先级分配一个entry,DB信息以链表方式按序缓存;若Doorbell加入Overflow Buffer中,Overflow buffer中已经存在该Doorbell所在的QPN的信息,则用新的Doorbell代替旧的Doorbell,也即在buffer中,对于同一个QP只会缓存一个最新的Doorbell;在Overflow Buffer中通过bitmap记录GroupID的各个优先级是否有Doorbell,若有Doorbell信息,则将比特位置1,否则为0。

4.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:在Overflow Buffer中只缓存DB的QPN,当调度到该QPN时,通过读取缓存在目标主机的DB Record获取最新的WQE_Index,再生成新的DB加入到调度器的第一层调度输出队列中,执行下一层调度;其中,DB Reecord是缓存在host主机侧的内容,用于记录SQ的消费指针produce index和生产指针consumer index。

5.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:第一层仲裁器或者调度器Arbiter对于同一个Group内不同CoS队列的调度方法如下:

步骤3.1,若当前轮询到GroupID=n,CoS=m的队列,首先判断该队列是否为空;若不为空则将队头的DB取出,加入调度器的输出队列中;若该队列为空,则转至步骤3.2;

步骤3.2,从Overflow Buffer中读取该队列的bitmap,判断是否缓存该队列的DB信息;若存在,则从链表头部将DB信息取出,加入对应的调度输出队列中,并将该DB信息从链表中删除,本次调度结束,等待下一个调度周期;若不存在,则跳过该队列,轮询到下一队列,本轮调度结束;Overflow Buffer为溢出缓存,当调度队列满了时,将DB信息缓存在溢出缓存中,该buffer是所有调度队列共享的缓存。

6.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:HostID可以用PF+VF ID表示;其中,PF ID表示physical function id,表示PCIE中物理通道的ID;VF ID表示virtual function id,表示PCIE虚拟通道的ID,一个虚通道代表一个虚拟机。

7.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:第二阶段调度模块也是采用层次化多级调度方式,其调度层级、每一层的Group大小、每一层调度器的调度算法可通过调度策略模块配置。

8.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:调度算法包括但不局限于SP、RR、WRR、DWRR调度算法;

其中,SP为严格优先级调度算法;

RR为轮询算法;

WRR为加权轮询调度算法;

DWRR为差分加权轮询算法。

9.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:第二阶段调度模块的第一级输入队列包含2个缓存行Entry用于缓存DB,其中队头的Entry表示正

在调度的DB,第二个Entry表示下一个待调度的DB。

10.根据权利要求1所述的基于RDMA的无损高效的数据流处理方法,其特征在于:DB加入输入队列时遵循以下方法:若队列为空,则将新的DB加入队头;若队列只有1个DB,则将新的DB加入队尾;若队尾也有DB,则将新的DB替换旧的DB缓存在队列中。

一种基于RMDA的无损高效的数据处理方法及网络接口卡

技术领域

[0001] 本发明属于数据中心的数据通信领域,尤其涉及一种基于RMDA的无损高效的数据处理方法及网络接口卡。

背景技术

[0002] RDMA技术在数据中心中得到越来越广泛的部署,以实现“高带宽、低时延、高吞吐、零丢包”的无损网络。基于RDMA技术的网络接口卡(RNIC)在处理大规模IO并发请求时,粗粒度的IO请求调度易导致严重的头阻问题,不同QP的请求互相阻塞;对于大消息的IO请求需要占用较长的资源,影响小消息的处理,无法保证多QP间的公平性;同时,在大规模请求IO并发条件下,若请求处理不及时,易造成IO丢失,对上层应用性能产生严重影响。

发明内容

[0003] 本发明所要解决的技术问题是针对背景技术的不足提供本发明提供了一种基于RDMA技术的网络接口卡和数据处理方法,当主机中存在大规模QP并发时,可高效解决不同QP的请求的调度处理问题,保证请求信号不丢失、QP请求不乱序、调度周期不浪费,同时结合拥塞控制有效保证不同QP的调度公平性问题,解决头阻问题。

[0004] 本发明为解决上述技术问题采用以下技术方案:

[0005] 一种基于RDMA的网络接口卡,包含PCIe BAR寄存器处理模块、第一阶段调度模块、QPC状态表、DB验证模块、QPN-QID映射模块、第二阶段调度模块、调度策略配置模块、拥塞控制模块、DMA引擎:

[0006] 其中,PCIe BAR寄存器处理模块,用于负责软硬交互中门铃信号DB的解析和处理;

[0007] 第一阶段调度模块,用于负责第一阶段门铃信号DB基于PageID和优先级的调度,保证门铃信号DB不丢失;

[0008] QPC状态表,缓存QPC状态信息;

[0009] DB验证模块,用于验证DB的合法性和对应QP的状态信息是否正确;

[0010] QPN-QID映射模块,用于为DB分配进入第二阶段调度模块的入队ID;

[0011] 第二阶段调度模块,用于负责第二阶段DB基于hostID和QPN的调度,保证不同的Host的不同QP可以得到公平调度;

[0012] 调度策略配置模块,用于配置第二阶段调度模块中调度级数、每个调度器规模、调度算法等策略的配置;

[0013] 拥塞控制模块,用于为每个QP分配Credit,控制每个QP在一轮调度周期可以发出的消息大小;

[0014] WQE处理模块,用于WQE的预取和WQE的处理,若WQE不能完整处理,则向第二阶段调度模块返回DB处理的断点信息

[0015] DMA引擎,用于RNIC和Host直接的数据搬运;

[0016] 其中,PageID为页ID,表示软件敲DB对应的BAR空间地址的ID;QPC表示QP的上下文

信息,用于缓存QP地址信息;QP表示RDMA的连接队列;hostID表示host的序列号ID;QPN表示QP的序列号ID;Credit为信用;WQE表示一个RDMA请求;RNIC表示RDMA网卡;

[0017] 一种基于RDMA的网络接口卡的RDMA的无损高效的数据流处理方法,具体包含如下步骤:

[0018] 步骤1,当目标主机中的SQ中有新的WQE产生,主机会产生一个门铃信号DB发送给RNIC;即通过PCIe接口将DB信息写入RNIC为该QP分配的Doorbell空间;

[0019] 其中,SQ表示发送端的队列;

[0020] 步骤2,RNIC根据Doorbell寄存器的地址解析该地址的PageID以及Doorbell的QPN、优先级CoS信息,将Doorbell加入第一阶段的调度器中;第一阶段调度器采用层次调度的结构,支持M个Group,每个Group又可进一步分为4个优先级队列,在每个Group中采用SP+WRR调度算法,不同的Group之间采用RR调度算法;SP为严格优先级调度算法;WRR为加权轮询调度算法;

[0021] 步骤3,第一阶段调度模块的第一层调度器采用配置好的调度算法选择待调度队列,将待处理的DB加入调度器的输出队列中;

[0022] 步骤4,第一阶段调度模块的第二级调度器采用RR调度将第一级调度器输出的DB写入输出队列中;

[0023] 步骤5,DB验证模块从第一级调度器第二级输出队列取出队头的DB进行合法性验证,判断DB的PageID与DB的QPN绑定的QPC中PageID是否一致以及QP的状态是否正常;若一致则将DB输入QPN-QID映射模块,若不一致则将该DB丢弃并返回错误给目标主机;其中,QPN表示QP的序列号ID;QID表示在RNIC中为QP分配的本地ID,在RDMA系统中是唯一的;QPN-QID映射模块用于QPN和本地QID的映射查找,通过QID作为本地QP Context的索引;

[0024] 步骤6,DB输入QPN-QID映射模块后,根据DB中的HostID查找其对应的GroupID,根据QPN映射到对应GroupID的输入队列中,保证同一个HostID的QP和DB放在同一个调度组中;

[0025] 其中,GroupID表示调度组的ID,不同的HostID可以放在不同的调度Goup;

[0026] 其中,HostID表示主机的ID,若在支持虚拟化的场景中,一个VM对应一个独立的HostID;

[0027] 步骤7,第二级调度模块中各个层级调度器将DB输出至模块最后的输出队列;

[0028] 步骤8,WQE处理模块从输出队列的队头取出DB,并根据QPN读取处理WQE所需的QPC状态;在QPC中,包含Max_Burst_Size、Max_Batch_WQE_count,分别表示单个WQE可以发送的最大消息数和一次可以处理的WQE的最大数量;Max_Burst_Size表示单次调度周期允许发送的最大字节数;Max_Batch_WQE_count表示单次调度周期允许获取的WQE最大数量,也即单次调度处理周期允许处理的最大消息个数;

[0029] 步骤9,WQE处理模块向拥塞控制模块请求Credit;其中,Credit是由拥塞控制算法为每个QP分配的可发送消息的大小;

[0030] 步骤10,WQE处理模块根据Max_Batch_WQE_count和当前可缓存的WQE数量WQE_Available_Count,通过DMA引擎向SQ获取不超过不超过N个WQE,其中,N＝min(WQE_Available_Count,Max_Batch_WQE_count);WQE_Available_Count表示最大可用的WQE数量;N表示单次调度周期可以处理的WQE个数,其中,N取WQE_Available_Count和Max_Batch_

WQE_count的最小值;

[0031] 步骤11,WQE处理模块逐个处理缓存的WQE,每处理一个WQE即更新QPC中WQE的消费指针,并更新Credit值,Credit值为当前的值减去WQE消费的Credit值;若在处理WQE的过程中,剩余的Credit不足以处理一个完整的WQE,将在剩余Credit消费后,将WQE处理的中断的状态返回,缓存在第二调度模块中的缓存在Interupted DB State表中,并将该队列对应的bitmap置1;其中,所述Interupted DB State包括HostID、QPN和Produce_Index、Target_WQE_Index和Walk_Offset,Produce_Index表示当前处理的WQE_Index,Walk_offset表示当前未处理完的WQE已经发送的数据指针,Target_WQE_Index表示当前的DB需要处理到的WQE位置;Credit为令牌值,表示当前该QP发送的字节数;Interupted DB State为中断DB信息表,用来缓存DB调度的中断信息;bitmap为位图,该bitmap中每一个bit位与QPN对应,若该QPN信息有效,则将对应的bit位置1,若无效则置0;

[0032] 步骤12,当第二阶段调度模块在下一个周期调度到某个队列时,先通过Bitmap判断该队列中是否有中断的DB状态,也即该队列对应的bitmap是否为1;若bitmap＝0,则从调度输入队列中读取一个新的DB处理;若bitmap＝1,则优先读取DB中断状态,并将DB中断状态信息组成一个新的DB发送到WQE处理模块;

[0033] 步骤13,WQE处理模块根据该DB的信息,从Produce_Index指向的WQE的Walk_Offset处开始继续处理MR的数据,依据Credit判断是否可以完全将WQE处理,重复步骤11-步骤13,直到DB不再中断可以全部发出去后将bitmap置0;其中,Produce_Index表示消费指针,表示当前的SQ处理到的位置;Walk_Offset表示单个WQE处理的中断位置的虚拟地址。

[0034] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,Doorbell加入第一阶段调度器的基本步骤包括:

[0035] 步骤2.1:第一阶段调度模块根据Hash(PageID)得到目标队列的GroupID,并根据DB中的CoS选择将Doorbell加入对应的优先级队列中;若队列未满,直接将Doorbell写入队尾,若该队列中存在该QPN的Doorbell,可以合并成1个DB;若该队列满了,转至步骤2.2;

[0036] 其中,CoS表示Channel of Sevice表示优先级通道,通常支持8种优先级;

[0037] 步骤2.2:第一阶段调度模块将Doorbell信息记录在Overflow Buffer中;Overflow Buffer是所有第一阶段调度队列的共享缓存;在Overflow Buffer中为每个GroupID的每个优先级分配一个entry,DB信息以链表方式按序缓存;若Doorbell加入Overflow Buffer中,Overflow buffer中已经存在该Doorbell所在的QPN的信息,则用新的Doorbell代替旧的Doorbell,也即在buffer中,对于同一个QP只会缓存一个最新的Doorbell;在Overflow Buffer中通过bitmap记录GroupID的各个优先级是否有Doorbell,若有Doorbell信息,则将比特位置1,否则为0。

[0038] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,在Overflow Buffer中只缓存DB的QPN,当调度到该QPN时,通过读取缓存在目标主机的DB Record获取最新的WQE_Index,再生成新的DB加入到调度器的第一层调度输出队列中,执行下一层调度;其中,DB Reecord是缓存在host主机侧的内容,用于记录SQ的消费指针produce index和生产指针consumer index。

[0039] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,第一层仲裁器或者调度器Arbiter对于同一个Group内不同CoS队列的调度方法如下:

[0040] 步骤3.1,若当前轮询到GroupID＝n,CoS＝m的队列,首先判断该队列是否为空;若不为空则将队头的DB取出,加入调度器的输出队列中;若该队列为空,则转至步骤3.2;

[0041] 步骤3.2,从Overflow Buffer中读取该队列的bitmap,判断是否缓存该队列的DB信息;若存在,则从链表头部将DB信息取出,加入对应的调度输出队列中,并将该DB信息从链表中删除,本次调度结束,等待下一个调度周期;若不存在,则跳过该队列,轮询到下一队列,本轮调度结束;Overflow Buffer为溢出缓存,当调度队列满了时,将DB信息缓存在溢出缓存中,该buffer是所有调度队列共享的缓存。

[0042] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,HostID可以用PF+VF ID表示;其中,PF ID表示physical function id,表示PCIE中物理通道的ID;VF ID表示virtual function id,表示PCIE虚拟通道的ID,一个虚通道代表一个虚拟机。

[0043] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,第二阶段调度模块也是采用层次化多级调度方式,其调度层级、每一层的Group大小、每一层调度器的调度算法可通过调度策略模块配置。

[0044] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,调度算法包括但不局限于SP、RR、WRR、DWRR调度算法;

[0045] 其中,SP为严格优先级调度算法;

[0046] RR为轮询算法;

[0047] WRR为加权轮询调度算法;

[0048] DWRR为差分加权轮询算法。

[0049] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,第二阶段调度模块的第一级输入队列包含2个缓存行Entry用于缓存DB,其中队头的Entry表示正在调度的DB,第二个Entry表示下一个待调度的DB。

[0050] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,DB加入输入队列时遵循以下方法:若队列为空,则将新的DB加入队头;若队列只有1个DB,则将新的DB加入队尾;若队尾也有DB,则将新的DB替换旧的DB缓存在队列中。

[0051] 作为本发明基于RDMA的无损高效的数据流处理方法的进一步优选方案,在步骤11,所述Interupted DB State包括HostID、QPN和Produce_Index、Target_WQE_Index和Walk_Offset,Produce_Index表示当前处理的WQE_Index,Walk_offset表示当前未处理完的WQE已经发送的数据指针,Target_WQE_Index表示当前的DB需要处理到的WQE位置。

[0052] 本发明采用以上技术方案与现有技术相比,具有以下技术效果:

[0053] 1、多阶段式调度,基于不同的调度粒度设计调度队列,避免所有DB放在一起串行执行,造成严重的头阻问题;

[0054] 2、第一阶段调度器模块,基于进程Page ID的通用DB分配方法,区别于传统的DB类型与地址绑定的方法,将DB类型与地址解耦,提高了DB空间的利用率;

[0055] 3、第一阶段调度器模块中,基于Host、进程和优先级等综合因素设计DB队列管理和调度方法,可以实现不同粒度的调度,在保证同一个QP调度不乱序的情况下,高优先级的请求可以有效调度,有效地解决大规模QP并发条件下的头阻问题;

[0056] 4、第一阶段调度器中,共享Overflow Buffer的DB无损处理方法,保证在高并发条件下不丢失DB信息,实现请求的无损处理;同时只缓存DB状态信息,通过读取DB Record得

到丢失的DB信息,有效节约缓存;

[0057] 5、第二阶段调度模块中,通过Credit+WQE个数+消息大小等约束,限制每个调度周期可以处理的WQE生成的最大数据量,保证多QP调度的公平性,避免大消息一直占用资源,阻塞其他的QP,缓解头阻问题;

[0058] 6、第二阶段调度模块中,通过对每个调度队列增加Interupted DB state缓存DB处理中断状态,在调度时支持WQE断点续传能力,实现精细化QP调度,保证高效无损地处理请求。

附图说明

[0059] 为了更清楚地说明本申请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,下面描述中的附图仅是本申请的一些实施例,对于本领域技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0060] 图1是本发明实施例1提供的应用于裸金属场景的一种无损高效的数据处理方法处理方法示意图;

[0061] 图2是本发明实施例2提供的应用于虚拟机场景一种无损高效的数据处理方法处理方法示意图。

具体实施方式

[0062] 下面结合附图对本发明的技术方案做进一步的详细说明:

[0063] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。下面根据附图和优选实施例详细描述本发明,本发明的目的和效果将变得更加明白,应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。

[0064] 本公开实施例提供的方法应用于数据存储设备中的网络接口卡,该网络接口卡(RDMA network interface card,RNIC)基于远程直接内存访问(remote direct memory access,RDMA)实现。

[0065] 在目标主机中支持RDMA协议,通过队列对(queue pair,QP)实现数据的发送和接收。每个QP包含一个发送队列(send queue,SQ)和一个接收队列(receive queue,RQ),其中,SQ负责消息的发送,RQ负责消息的接收,每个QP的SQ和RQ可以分别关联一个完成队列(completion queue,CQ)。每个QP都有一个本地唯一的QP号(QP Number,QPN)。QP的一些状态信息,包括QPN、CQN、QP地址、QP长度等信息都存在QP Context(QPC)里,在RNIC中会维护一个QPC表,缓存注册的QPC信息。

[0066] 在目标主机中可包括多个发送队列,例如SQ1、SQ2、SQ3,当目标主机下发一个工作请求(Work Request,WR)时,会向SQ中可用空间写入一个工作队列元素(Work Queue Element,WQE),并通过驱动向与该SQ绑定的RNIC发送门铃信号(Doorbell,DB)通知网卡。当RNIC收到DB之后,会通过DMA读取DB携带的QPN、WQE_index等信息读取QPC Table,计算得到

SQ中WQE的地址,再发起DMA请求读取WQE的内容再处理。

[0067] RNIC与目标主机之间采用通信总线连接(PCIe),DB写入PCIe的BAR寄存器。在RNIC中,PCIe BAR空间以页(Page)为单位管理,不同的进程分配不同的BAR空间寄存器,BAR空间之间相互隔离。每个Page中分配有N个通用的DB,若目标主机向RNIC发送Doorbell,则通过PCIe接口将Doorbell写入与该进程绑定的BAR空间中可用的Doorbell。在Doorbell中有一个2bit的DB_type字段标识Doorbell类型,其中,DB_type=00是保留值,DB_type=01表示CQ doorbell,DB_type=10表示SQ Doorbell,DB_type=11表示EQ_AEQ Doorbell。

[0068] DB的本质是目标主机下发的请求通知信号,RNIC根据DB下发顺序和携带的参数去处理对应的WR或WQE。对DB的处理实质是对WR的处理。为解决大规模QP并发条件下的请求调度和处理问题,有效缓解不同的QP调度造成的头阻问题,保证不同QP的调度公平性,同时保证在请求突发条件下也DB不丢失.

[0069] 如图1和图2所示,本发明中提出的一种无损高效的DB信号处理方法,其基本步骤包括:

[0070] 步骤1:当目标主机中的SQ中有新的WQE产生,主机会产生一个门铃信号DB发送给RNIC。即通过PCIe接口将DB信息写入RNIC为该QP分配的Doorbell空间。

[0071] 其中,SQ表示发送端的队列;

[0072] 步骤2:RNIC根据Doorbell寄存器的地址解析该地址的PageID以及Doorbell的QPN、优先级(CoS)等信息,将Doorbell加入第一阶段的调度器中。第一阶段调度器采用层次调度的结构,可支持M个Group,每个Group又可进一步分为4个优先级队列,在每个Group中采用SP+WRR调度算法,不同的Group之间采用RR调度算法。SP为严格优先级调度算法;WRR为加权轮询调度算法。

[0073] Doorbell加入第一阶段调度器的基本步骤包括:

[0074] 步骤2.1:第一阶段调度模块根据Hash(PageID)得到目标队列的GroupID,并根据DB中的CoS选择将Doorbell加入对应的优先级队列中。若队列未满,直接将Doorbell写入队尾,若该队列中存在该QPN的Doorbell,可以合并成1个DB。若该队列满了,转至步骤2.2。

[0075] 步骤2.2:第一阶段调度模块将Doorbell信息记录在Overflow Buffer中。Overflow Buffer是所有第一阶段调度队列的共享缓存。在Overflow Buffer中为每个GroupID的每个优先级分配一个entry,DB信息以链表方式按序缓存。若Doorbell加入Overflow Buffer中,Overflow buffer中已经存在该Doorbell所在的QPN的信息,则用新的Doorbell代替旧的Doorbell,也即在buffer中,对于同一个QP只会缓存一个最新的Doorbell。

[0076] 在Overflow Buffer中通过bitmap记录GroupID的各个优先级是否有Doorbell,若有Doorbell信息,则将比特位置1,否则为0。

[0077] 可选地,为节约缓存,在Overflow Buffer中只缓存DB的QPN,当调度到该QPN时,通过读取缓存在目标主机的DB Record获取最新的WQE_Index,再生成新的DB加入到调度器的第一层调度输出队列中,执行下一层调度。

[0078] 步骤3:第一阶段调度器的第一层调度器采用配置好的调度算法选择待调度队列,将待处理的DB加入调度器的输出队列中,其调度算法包括但不限于Round Robin(RR)调度等。

[0079] 第一层Arbiter对于同一个Group内不同CoS队列的调度方法如下:

[0080] 步骤3.1若当前轮询到GroupID＝n,CoS＝m的队列,首先判断该队列是否为空。若不为空则将队头的DB取出,加入调度器的输出队列中;若该队列为空,则转至步骤3.2;

[0081] 步骤3.2:从Overflow Buffer中读取该队列的bitmap,判断是否缓存该队列的DB信息。若存在,则从链表头部将DB信息取出,加入对应的调度输出队列中,并将该DB信息从链表中删除,本次调度结束,等待下一个调度周期。若不存在,则跳过该队列,轮询到下一队列,本轮调度结束。

[0082] 步骤4:第一阶段调度器的第二级Arbiter采用RR调度将第一级Arbiter输出的DB写入输出队列中。

[0083] 步骤5:DB验证模块从第一级调度器第二级Arbiter的输出队列中,取出队头的DB进行合法性验证,判断DB的PageID与DB的QPN绑定的QPC中PageID是否一致以及QP是否处于正常状态。若一致则将DB输入QPN-QID映射模块,若不一致则将该DB丢弃并返回错误给目标主机。其中,QPN表示QP的序列号ID;QID表示在RNIC中为QP分配的本地ID,在RDMA系统中是唯一的;QPN-QID映射模块用于QPN和本地QID的映射查找,通过QID作为本地QP Context的索引。

[0084] 步骤6:DB输入QPN-QID映射模块后,该模块根据DB中的HostID查找其对应的GroupID,再根据QPN映射到对应GroupID的输入队列中,保证同一个HostID的QP DB放在同一个调度组中。其中,GroupID表示调度组的ID,不同的HostID可以放在不同的调度Goup;

[0085] 其中,HostID表示主机的ID,若在支持虚拟化的场景中,一个VM对应一个独立的HostID。

[0086] 具体地,HostID可以用PF+VF ID表示。其中,PF ID表示physical function id,表示PCIE中物理通道的ID;VF ID表示virtual function id,表示PCIE虚拟通道的ID,一个虚通道代表一个虚拟机。

[0087] 第二阶段调度模块也是采用层次化多级调度方式,其调度层级、每一层的Group大小、每一层调度器的调度算法可通过调度策略模块配置。调度算法包括但不局限于SP、RR、WRR、DWRR等调度算法。

[0088] 第二阶段调度模块的第一级输入队列包含2个Entry用于缓存DB,其中队头的Entry表示正在调度的DB,第二个Entry表示下一个待调度的DB。DB加入输入队列时遵循以下方法:

[0089] 若队列为空,则将新的DB加入队头;若队列只有1个DB,则将新的DB加入队尾;若队尾也有DB,则将新的DB替换旧的DB缓存在队列中。

[0090] 步骤7:第二级调度模块中各个层级调度器将DB输出至模块最后的输出队列。

[0091] 步骤8:WQE处理模块从输出队列的队头取出DB,并根据QPN读取处理WQE所需的QPC状态。在QPC中,包含Max_Burst_Size、Max_Batch_WQE_count,分别表示单个WQE可以发送的最大消息数和一次可以处理的WQE的最大数量。Max_Burst_Size表示单次调度周期允许发送的最大字节数;Max_Batch_WQE_count表示单次调度周期允许获取的WQE最大数量,也即单次调度处理周期允许处理的最大消息个数。

[0092] 步骤9:WQE处理模块根据(HostID,QPN)向拥塞控制模块请求Credit,Credit是由拥塞控制算法为每个QP分配的可发送消息的大小。

[0093] 步骤10:WQE处理模块根据Max_Batch_WQE_count和当前可缓存的WQE数量WQE_Available_Count,通过DMA引擎向SQ获取不超过不超过N个WQE,其中,N＝min(WQE_Available_Count,Max_Batch_WQE_count)。其中,N＝min(WQE_Available_Count,Max_Batch_WQE_count);WQE_Available_Count表示最大可用的WQE数量;N表示单次调度周期可以处理的WQE个数,其中,N取WQE_Available_Count和Max_Batch_WQE_count的最小值。

[0094] 步骤11:WQE处理模块逐个处理缓存的WQE,每处理一个WQE即更新QPC中WQE的消费指针,并更新Credit值,Credit值为当前的值减去WQE消费的Credit值。若在处理WQE的过程中,剩余的Credit不足以处理一个完整的WQE,将在剩余Credit消费后,将WQE处理的中断的状态返回,缓存在第二调度模块中的缓存在Interupted DB State表中,并将该队列对应的bitmap置1。其中,所述Interupted DB State包括HostID、QPN和Produce_Index、Target_WQE_Index和Walk_Offset,Produce_Index表示当前处理的WQE_Index,Walk_offset表示当前未处理完的WQE已经发送的数据指针,Target_WQE_Index表示当前的DB需要处理到的WQE位置;Credit为令牌值,表示当前该QP发送的字节数;Interupted DB State为中断DB信息表,用来缓存DB调度的中断信息;bitmap为位图,该bitmap中每一个bit位与QPN对应,若该QPN信息有效,则将对应的bit位置1,若无效则置0。

[0095] 步骤12:当第二阶段调度模块在下一个周期调度到某个队列时,先通过Bitmap判断该队列中是否有中断的DB状态,也即该队列对应的bitmap是否为1。若bitmap＝0,则从调度输入队列中读取一个新的DB处理;若bitmap＝1,则优先读取DB中断状态,并将DB中断状态信息组成一个新的DB发送到WQE处理模块。

[0096] 步骤13:WQE处理模块根据该DB的信息,从Produce_Index指向的WQE的Walk_Offset处开始继续处理MR的数据,再依据Credit判断是否可以完全将WQE处理,重复步骤11-步骤13,直到DB不再中断可以全部发出去后将bitmap置0。其中,Produce_Index表示消费指针,表示当前的SQ处理到的位置;Walk_Offset表示单个WQE处理的中断位置的虚拟地址。

[0097] 该方法主要针对一个WQE对应大消息或者一个DB中需要处理多个WQE的情况,需要消耗比较多的Credit,因而可能需要多轮调度周期才能处理完一个DB。

[0098] 至此,整个2阶段式的DB调度和处理方法结束。通过上述方法,可以保证DB不丢失,同时有效降低QP调度的头阻问题,保证多QP并发场景下WQE处理的公平。

[0099] 本发明提出的一种基于RDMA的无损高效的数据处理方法和网络接口卡提供了2种实施例,分别针对裸金属场景和云主机场景。在2种场景中,其基本模块和方法基本一致,如第五节的技术方案所示,但2种场景在RNIC中各个模块的处理又略有区别。

[0100] 无论是对于裸金属场景还是云主机场景,RNIC为每个Host绑定的BAR空间是唯一的,因而PageID是唯一的,两种条件下生成的DB加入第一阶段调度器的过程相同。

[0101] 对于裸金属场景,在宿主机中,只有一个Host创建QP下发请求,QPN是唯一的;在云主机场景中,在宿主机中会有多个虚拟机,可以看成多个host,每个虚拟机都可以创建QP,下发请求,QPN在VM当中是唯一的,不同的VM之间QPN可以相同。因而在QPC表中,需要按Host粒度维护和管理。

[0102] 在进行QPN-QID的映射时,若是裸金属场景只有一个host,则该host可以占用所有Group的调度资源;若是云主机场景有多个host,则需要基于host分配group。

[0103] 同样地,在拥塞控制模块中,需要为每个QP维护Credit,该Credit的管理和维护也要基于Host和QPN区分。其中,PageID为页ID,表示软件敲DB对应的BAR空间地址的ID;QPC表示QP的上下文信息,用于缓存QP地址信息;QP表示RDMA的连接队列;hostID表示host的序列号ID;QPN表示QP的序列号ID;Credit为信用;WQE表示一个RDMA请求;RNIC表示RDMA网卡。

[0104] 多阶段式调度,基于不同的调度粒度设计调度队列,避免所有DB放在一起串行执行,造成严重的头阻问题。

[0105] 第一阶段调度器模块,基于进程Page ID的通用DB分配方法,区别于传统的DB类型与地址绑定的方法,将DB类型与地址解耦,提高了DB空间的利用率。

[0106] 第一阶段调度器模块中,基于Host、进程和优先级等综合因素设计DB队列管理和调度方法,可以实现不同粒度的调度,在保证同一个QP调度不乱序的情况下,高优先级的请求可以有效调度,有效地解决大规模QP并发条件下的头阻问题

[0107] 第一阶段调度器中,共享Overflow Buffer的DB无损处理方法,保证在高并发条件下不丢失DB信息,实现请求的无损处理;同时只缓存DB状态信息,通过读取DB Record得到丢失的DB信息,有效节约缓存。

[0108] 第二阶段调度模块中,通过Credit+WQE个数约束,限制每个调度周期可以处理的WQE生成的数据包个数,保证多QP调度的公平性,避免大消息一直占用资源,阻塞其他的QP,缓解头阻问题。

[0109] 第二阶段调度模块中,通过对每个调度队列增加Interupted DB state缓存DB处理中断状态,在调度时支持WQE断点续传能力,实现精细化QP调度,保证高效无损地处理请求。

[0110] 本领域普通技术人员可以理解,以上所述仅为发明的优选实例而已,并不用于限制发明,尽管参照前述实例对发明进行了详细的说明,对于本领域的技术人员来说,其依然可以对前述各实例记载的技术方案进行修改,或者对其中部分技术特征进行等同替换。凡在发明的精神和原则之内,所做的修改、等同替换等均应包含在发明的保护范围之内本实施例中的所有技术特征均可根据实际需要而进行自由组合。

[0111] 最后应说明的是:以上所述仅为本发明的优选实施例而已,并不用于限制本发明,尽管参照前述实施例对本发明进行了详细的说明,对于本领域的技术人员来说,其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换,凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

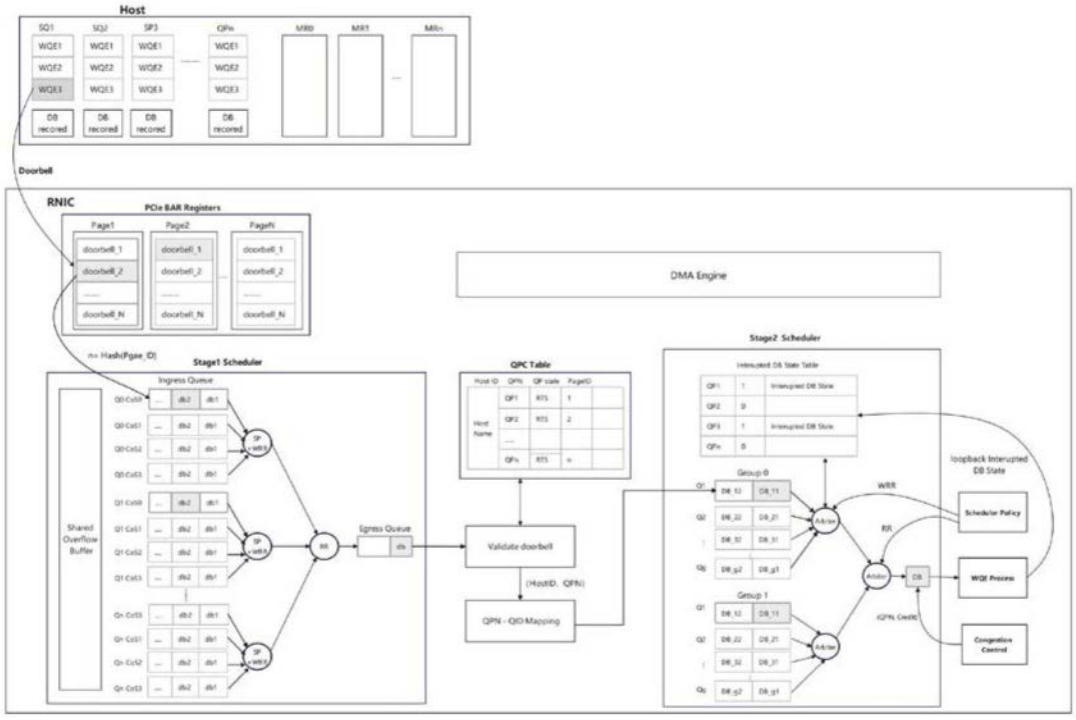

Host
Doorbell
RNIC
PCIe BAR Registers
DMA Engine
Stage1 Scheduler
Stage2 Scheduler
QPC Table
Ingress Queue
Egress Queue
Validate doorbell
QPN - QID Mapping
Shared Overflow Buffer
Group 0
Group 1
WRR
RR
Loopback Interupted DB State
Scheduler Policy
WQE Process
Congestion Control

图1

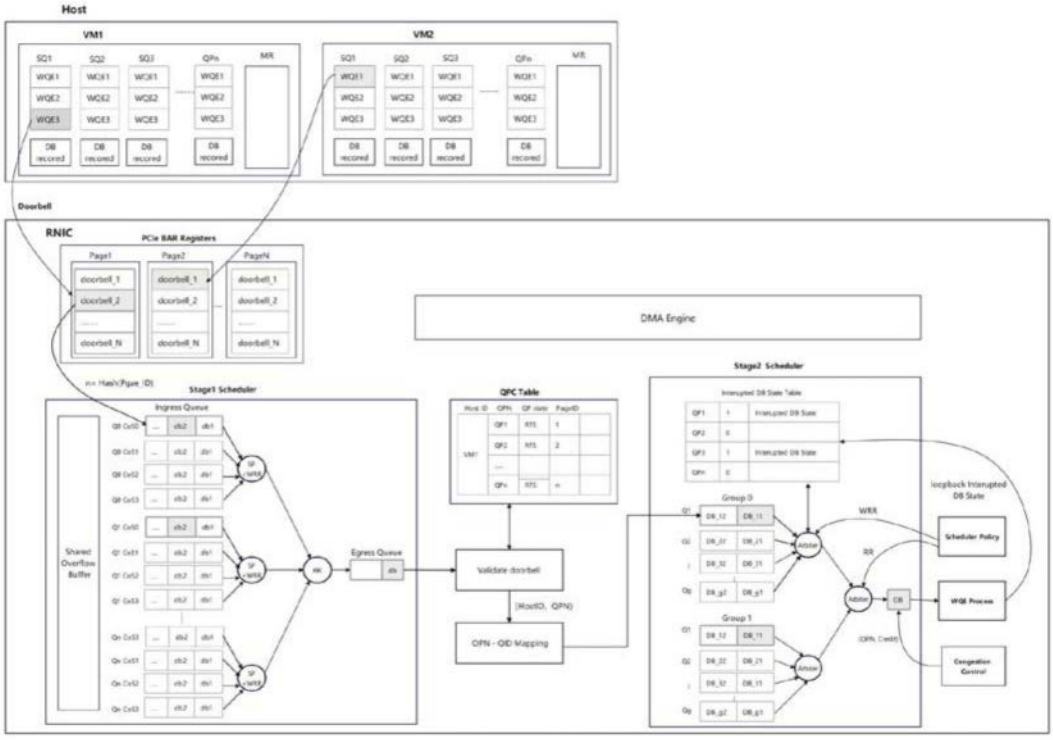

Host
VM1
VM2
Doorbell
RNIC
PCIe BAR Registers
DMA Engine
Stage1 Scheduler
Stage2 Scheduler
QPC Table
Ingress Queue
Egress Queue
Validate doorbell
QPN - QID Mapping
Shared Overflow Buffer
Group 0
Group 1
WRR
RR
Loopback Interupted DB State
Scheduler Policy
WQE Process
Congestion Control

图2