说明书

发明名称：在 RDMA 中实现定时器处理的方法、系统、芯片及存储介质

相关申请的交叉引用

本申请要求于 2024 年 3 月 26 日提交中国专利局、申请号为 202410355510.4、发明名称为“在 RDMA 中实现定时器处理的方法、系统、芯片及存储介质”的中国专利申请的优先权，所述专利申请的全部内容通过引用结合在本申请中。

技术领域

本申请涉及数据存储技术领域，特别是涉及一种在 RDMA 中实现定时器处理的方法、系统、芯片及存储介质。

背景技术

在远程直接数据存取（Remote Direct Memory Acess，RDMA）技术的应用中，各功能（Function）中每个队列对（Queue Pair，QP）均会挂载时间节点（time node），用于计时唤醒 RDMA 装置进行相应的收发处理。在一些例子中，RDMA 引擎可以为每个 QP 提供涉及两类定时器的时间节点，分别是响应类型（Acknowledge，Ack）定时器和接收侧未准备好类型（Receiver not ready，Rnr）定时器。

一般地，通过时间轮来实现上述时间节点的定时功能，以响应类型（Ack）定时器为例，在一个应用场景中，RDMA 引擎提供多个时间轮（如 4 个），每个时间轮能够表示的时长分别为 2ms/32ms/512ms/8s，每个时间轮包含 16 个刻度，每个时间轮的刻度表示的时长分别为 128us/2ms/32ms/512ms。由于 RDMA 引擎支持的 QP 数很多，通常为 M（兆）级别，因此 RDMA 引擎最大需要提供 1M 个 ACK 类型的时间节点，RDMA 引擎采用链表的方式管理时间节点，RDMA 引擎为每个时间轮刻度提供一个链表，当某个时间轮刻度超时到时，RDMA 引擎会读取该刻度对应链表里面所有的时间节点供 RDMA 引擎用于超时判断。在极端情况下，会产生 1M 个时间节点并链入同一个时间轮的同一个刻度对应的链表内。由于

考虑功耗性能面积（Power performance area，PPA）的原因，片上通常不会提供很大的资源来存储这 1M 个链表节点，因此片上存储不下的链表节点会溢出到主机（Host）的 DDR 存储器里面，即 RDMA 引擎提供片上和片外同一条链表来管理一个时间轮刻度的所有时间节点。

相关技术中有两种方法，第一种是共用四个时间轮，其存在的缺点是当 RDMA 引擎需要支持虚拟化时，由于一个时间轮刻度对应链表的节点可能属于不同的功能（Function），因此在片外没办法只提供同一条链表来存储不同 Function 的时间节点，当芯片内部的空闲空间不够时，需要将定时节点链接到 Host 的链表中，这很容易引起 Function 干扰的问题。另一种方法为为每个 Function 都提供一套时间轮，这样每个 Function 的每个时间轮刻度都有独立的链表来管理各自的时间节点，即不会出现不同 Function 的时间节点链入一条链表的情况。

如图 1 所示，示出了相关技术中提供的支持多个 Function 时间节点虚化管理的原理示意图；从中可以看出，一套时间轮组（共 4 个时间轮）需要管理 64 条链表，其占用 PPA 较大，当 RDMA 引擎支持虚拟化时，RDMA 引擎为每个 Function 都提供一套时间轮，由于虚拟化时支持的 Function 数量很大，通常是 K 级别；RDMA 引擎共实现了 n*4 个时间轮，实现和管理片内 n*4*64 个链表，同时还需要管理链表在片外的部分，因此 RDMA 引擎为定时器付出的资源很大，会影响 PPA；

另外由于时间轮采用的链表分片内和片外两部分，让整体设计变得复杂。再次，RDMA 引擎实现的时间轮只能提供最大 8s 的超时，当 RDMA 引擎超时时长超过 8s 时，比如支持长距离的 RDMA 引擎需要支持更大的超时时间，那 RDMA 引擎需要增加时间轮来达到目的，这在增加 PPA（power performance area）同时，也使整体架构缺乏灵活性。

发明内容

根据本申请的各种实施例，提供一种在 RDMA 中实现定时器处理的方法、系统、芯片及存储介质。

作为本申请的一方面，提供一种在 RDMA 中实现定时器处理的方法，应用于 RDMA 引擎中，其至少包括如下步骤：

接收主机所生成的 RDMA 报文，为该报文生成一对应的超时时间节点，在其超时时间未超出预先设定的阈值时，为所述超时时间节点确定共享时间轮组中对应的时间轮以及超时时间在所述时间轮中对应的超时时间刻度；

判断共享时间轮组对应的存储单元中是否存在空闲节点；

在所述存储单元中存在空闲节点时，申请一个空闲节点以存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，由 RDMA 引擎进行超时判断；

在所述超时时间超出预先设定的阈值，或所述存储单元中不存在空闲节点时，将超时时间节点以事件队列的方式告知主机，由主机管理所述超时时间。

其中，进一步包括：在 RDMA 引擎中配置一个共享时间轮组，所述时间轮组包括预定第一数目个不同类别的时间轮，每个时间轮对应预定第二数目个刻度值，每个刻度值关联有一链表。

其中，在所述存储单元中存在空闲节点时，申请一个空闲节点存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，进一步包括：

在存储单元中存在空闲节点时，进行空闲节点申请，获得一空闲的存储地址；

将所述 RDMA 报文的 Function ID 和 QPN 写入至所述空闲的存储地址对应的存储空间中；

在所确定的超时时间刻度对应的链表中建立与所述空闲的存储地址之间的关联。

其中，由 RDMA 引擎进行超时判断，进一步包括：

当时间轮刻度到期时，遍历对应的链表，取出所述链表的所有超时时间

节点；

将所取出的超时时间节点发送给 RDMA 引擎进行超时判断，并执行相应超时时间节点中的任务。

其中，在存储单元中不存在空闲节点时，将所述超时时间节点以事件队列的方式告知主机，由主机管理所述超时，进一步包括：

在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点对应的 RDMA 的 Function ID 和 QPN 信息通过该 Function 对应的事件队列通知主机。

相应地，本申请的另一方面，还提供一种在 RDMA 中实现定时器处理的系统，其至少包括：

RDMA 报文接收处理，用于接收主机所生成的 RDMA 报文，为该报文生成一对应的超时时间节点，在其超时时间未超出预先设定的阈值时，为所述超时时间节点确定共享时间轮组中对应的时间轮以及超时时间在所述时间轮中对应的超时时间刻度

条件判断单元，用于判断共享时间轮组对应的存储单元中是否存在空闲节点；

第一处理单元，用于在所述存储单元中存在空闲节点时，申请一个空闲节点以存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，由 RDMA 引擎进行超时判断；

第二处理单元，用于在所述超时时间超出预先设定的阈值，或所述存储单元中不存在空闲节点时，将超时时间节点以事件队列的方式告知主机，由主机管理所述超时时间。

其中，进一步包括：

配置单元，用于在 RDMA 引擎中配置一个共享时间轮组，所述时间轮组包括预定第一数目个不同类别的时间轮，每个时间轮对应预定第二数目个刻度值，每个刻度值关联有一链表。

其中，第一处理单元进一步包括：

空闲节点申请单元，用于在存储单元中存在空闲节点时，进行空闲节点申请，获得一空闲的存储地址；

写入单元，用于将所述 RDMA 报文的 Function ID 和 QPN 写入至所述空闲的存储地址对应的存储空间中；

链接单元，用于在所确定的超时时间刻度对应的链表中建立与所述空闲的存储地址之间的关联；

遍历处理单元，用于当时间轮刻度到期时，遍历对应的链表，取出所述链表的所有超时时间节点；

第一超时处理单元，用于将所取出的超时时间节点发送给 RDMA 引擎进行超时判断，并执行相应超时时间节点中的任务。

其中，第二处理单元进一步包括：

通知单元，用于在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点对应的 RDMA 的 Function ID 和 QPN 信息通过该 Function 对应的事件队列通知主机。

相应地，本申请的又一方面，还提供一种芯片，其部署有如前述的系统。

相应地，本申请的再一方面，还提供一种计算机可读存储介质，其上存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现如前述的方法的步骤。

附图说明

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仅是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动性的前提下，根据这些附图获得其他的附图仍属于本申请的范畴。

图 1 为现有技术中提供的支持多个 Function 时间节点虚化管理的原理示意图；

图 2 为本申请提供的一种在 RDMA 中实现定时器处理的方法的一个实施例的主流程示意图；

图 3 为申请涉及的应用环境示意图；

图 4 为本申请提供的方法中时间节点虚化管理的原理示意图；

图 5 为本申请提供的一种在 RDMA 中实现定时器处理的系统的一个实施例的结构示意图；

图 6 为图 5 中第一处理单元的结构示意图；

图 7 为图 5 中第二处理单元的结构示意。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请作进一步地详细描述。

如图 1 所示，示出了本申请提供的一种在 RDMA 中实现定时器处理的方法的一个实施例的主流程示意图。一并结合图 2 至图 4 所示，在本实施例中，所述方法应用于 RDMA 通信系统中，一般地，如图 2 所示，在一个 RDMA 通信系统中，至少包括位于本端的本端主机、本端外部存储空间（DDR）、本端 RDMA 引擎、本端 PCIe 设备，以及位于远端的远端 PCIe 设备、远端 RDMA 引擎、远端主机、远端外部存储空间（DDR）。其中，本端 PCIe 设备和远端 PCIe 设备可以是诸如：网络适配器、存储控制器、加速卡、硬件加密卡等。

具体如图 1 所示，在本申请实施例中，所述在 RDMA 中实现定时器处理的方法应用于本端 RDMA 中，具体地，所述方法进一步包括如下的步骤：

步骤 S10，接收主机所生成的 RDMA 报文，为该报文生成一对应的超时时间节点，判断其超时时间是否超过一设定的超时时间阈值；在其超时时间未超出预先设定的阈值时，为所述超时时间节点确定共享时间轮组中对应的时间轮以及超时时间在所述时间轮中对应的超时时间刻度；

可以理解的是，在本申请的实施例中，需预先包括如下步骤：

在 RDMA 引擎中配置一个共享时间轮组，所述时间轮组包括预定第一数目个不同类别的时间轮，每个时间轮对应预定第二数目个刻度值，每个刻度值对应有一个链表；如图 4 所示，在本申请的实施例中，RDMA 引擎只实现 1 个时间轮组，内部提供和典型在线流个数相当的时间节点。具体地，在一个例子中，所述第一数目为 4，所述第二类目为 16。故在图 4 的例子中，RDMA 引擎共实现了 4 个时间轮，实现和管理 4*64 个片上链表，且链表无需设置片外部分，在具体的例子中，采用一个存储单元对所述共享时间轮组的链表进行存储和管理。

在一个例子中，假设超时时间为 2s，设定的超时时间阈值为 8s，由于 2s 小于 8s，没有超过阈值，那么根据超时时间 2s 选择对应的时间轮（假设对应的时间轮组中的四个时间轮分别为 16s（时间轮 0）、16*16s（时间轮 1）、256*16s（时间轮 2）、256*16*16s（时间轮 3）），则选择时间轮 0，假设时间轮 0 当前所在的刻度为刻度 5，那么得出超时时间刻度为 5+2=7，即时间轮 0 中的刻度 7 为超时时间刻度。

步骤 S11，对所述超时时间及当前共享时间轮组的空闲空间进行判断，判断共享时间轮组对应的存储单元中是否存在空闲节点；

步骤 S12，如果判断结果为在所述存储单元中存在空闲节点，则申请一个空闲节点以存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，由 RDMA 引擎进行超时判断；

在具体的例子中，所述步骤 S12 进一步包括：

在存储单元中存在空闲节点时，进行空闲节点申请，获得一空闲的存储地址；

将所述 RDMA 报文的 Function ID 和 QPN 写入至所述空闲的存储地址对应的存储空间中；

在所确定的超时时间刻度对应的链表中建立与所述空闲的存储地址之间的关联；在具体的例子中，可以利用链表或者 bitmap 来管理空闲节点，每个时间刻度都对应一条链表，通过对每一条链表的首、尾指针来实现对链表

的管理；

当时间轮刻度到期时，遍历对应的链表，取出所述链表的所有超时时间节点；

将所取出的超时时间节点发送给 RDMA 引擎进行超时判断，并执行相应超时时间节点中的任务。

步骤 S13，在判断结果为所述超时时间超出预先设定的阈值，或在存储单元中不存在空闲节点时，将所述超时时间节点以事件队列的方式告知主机，由主机管理所述超时。

在具体的例子中，所述步骤 S13 进一步包括：

在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点对应的 RDMA 的 Function ID 和 QPN 信息通过该 Function 对应的事件队列通知主机。

在具体的例子中，主机侧会利用其系统定时器来为每一接收到的时间节点提供定时功能；

当系统定时器到达时，主机通过 doorbell 告知 RDMA 引擎，以对相应的时间节点进行超时判断，并执行相应时间节点中的任务。

结合图 4 所示，以一个具体的例子进行说明。在本实施例中，RDMA 引擎只实现 1 个时间轮组，内部提供和典型在线流个数相当的时间节点，当时间节点资源耗尽时，或者定时器的超时值超过配置时，RDMA 引擎通过每个 function 的事件队列告诉 HOST 侧驱动，由驱动利用系统提供的定时器来实现定时功能，当系统定时器超时，HOST 侧驱动通过 doorbell 告诉 RDMA 引擎供 RDMA 引擎用于超时判断。RDMA 引擎共实现了 4 个时间轮，实现和管理 4*64 个片上链表，链表无需设置片外部分。

本方案中 RDMA 引擎使用定时器的过程如下：

步骤 1，RDMA 引擎发送一个报文出去，需要监控报文在某个时间内是否完成，因此在其定时器管理模块中启动一个定时器，定时器管理模块根据定时器的超时值来确定时间轮和时间轮内的刻度。

步骤 2，定时器管理模块从共享时间节点资源内分配时间节点。

步骤 3，定时器管理模块将时间节点链入到对应的时间轮刻度链表里面。

步骤 4，当时间轮刻度到期时，定时器管理模块会遍历对应的链表将时间节点都取出来并送给 RDMA 引擎用于超时判断，对于超时的时间节点，则执行时间节点中所携带的任务。

步骤 5，当共享时间节点资源不足时，或者定时器超时值超过预定阈值时，将时间节点通过该时间节点对应的 Function 的事件队列通知给主机侧（如主机驱动）。

步骤 6，主机驱动利用系统提供的定时器来为 RDMA 引擎提供定时功能。

步骤 7，当系统定时器超时，主机驱动通过 doorbell 的方式通知 RDMA 引擎用于超时判断。

可以理解的是，实施本申请提供的方法，通过软硬结合的方式实现 RDMA 引擎虚拟化定时器，片上只保留典型在线流的时间节点，大部分场景下由于时间节点需求不大并不需要驱动干涉，当发生片上时间节点不够时或者定时器超时值很大时，通过事件队列通知主机，由主机驱动利用系统提供的定时器来实现定时功能。

本技术方案可以让 RDMA 引擎在支持虚拟化时，无需要增加额外的资源来实现定时器，节省了芯片的 PPA，同时简化了芯片设计，另外由于将超时时间比较大的定时器，直接送往主机驱动由系统定时器来实现，可以让 RDMA 引擎支持的超时值范围更广。

如图 5 所示，示出了本申请提供的一种在 RDMA 中实现定时器处理的系统的一个实施例的结构示意图，其在如图 3 的应用环境中实现。一并结合图 6 至图 7 所示，在本实施例中，所述这系统 1 至少包括：

RDMA 报文接收处理单元 10，用于接收主机所生成的 RDMA 报文，为该报文生成一对应的超时时间节点，在其超时时间未超出预先设定的阈值

时，为所述超时时间节点确定共享时间轮组中对应的时间轮以及超时时间在所述时间轮中对应的超时时间刻度；

条件判断单元 11，用于判断共享时间轮组对应的存储单元中是否存在空闲节点；

第一处理单元 12，用于在所述存储单元中存在空闲节点时，申请一个空闲节点以存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，由 RDMA 引擎进行超时判断；

第二处理单元 13，用于在所述超时时间超出预先设定的阈值，或所述存储单元中不存在空闲节点时，将超时时间节点以事件队列的方式告知主机，由主机管理所述超时时间；

配置单元 14，用于在 RDMA 引擎中配置一个共享时间轮组，所述时间轮组包括预定第一数目个不同类别的时间轮，每个时间轮对应预定第二数目个刻度值，每个刻度值关联有一链表；

如图 6 所示，在具体的例子中，所述第一处理单元 12 进一步包括：

空闲节点申请单元 120，用于在存储单元中存在空闲节点时，进行空闲节点申请，获得一空闲的存储地址；

写入单元 121，用于将所述 RDMA 报文的 Function ID 和 QPN 写入至所述空闲的存储地址对应的存储空间中；

链接单元 122，用于在所确定的超时时间刻度对应的链表中建立与所述空闲的存储地址之间的关联；

遍历处理单元 123，用于当时间轮刻度到期时，遍历对应的链表，取出所述链表的所有超时时间节点；

第一超时处理单元 124，用于将所取出的超时时间节点发送给 RDMA 引擎进行超时判断，并执行相应超时时间节点中的任务。

如图 7 所示，在具体的例子中，所述第二处理单元 13 进一步包括：

通知单元 130，用在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点对应的 RDMA 的 Function ID 和

QPN 信息通过该 Function 对应的事件队列通知主机。

相应地，本申请的又一方面，还提供一种芯片，其部署有如前述图 5 至图 87 所描述的系统。更多细节，请一并参考并结合前述对图 5 至图 7 的描述，在此不进行赘述。

相应地，本申请的再一方面，还提供一种计算机可读存储介质，其上存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现如前述图 2 至图 4 所描述的方法的步骤。更多细节，请一并参考并结合前述对图 2 至图 4 的描述，在此不进行赘述。

实施本申请实施例，具有如下有益效果：

本申请提供了一种在 RDMA 中实现定时器处理的方法、系统、芯片及存储介质。通过软硬结合的方式实现 RDMA 引擎虚拟化定时器，片上只保留典型在线流的时间节点，大部分场景下由于时间节点需求不大并不需要驱动干涉，当发生片上时间节点不够时或者定时器超时值很大时，通过事件队列通知主机，由主机驱动利用系统提供的定时器来实现定时功能。

本技术方案可以让 RDMA 引擎在支持虚拟化时，无需要增加额外的资源来实现定时器，节省了芯片的 PPA，同时简化了芯片设计，另外由于将超时时间比较大的定时器，直接送往主机驱动由系统定时器来实现，可以让 RDMA 引擎支持的超时值范围更广，从而提高了架构的灵活性。

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、装置、或计算机程序产品。因此，本申请可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且，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。

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。

以上所揭露的仅为本申请一种较佳实施例而已，当然不能以此来限定本申请之权利范围，因此依本申请权利要求所作的等同变化，仍属本申请所涵盖的范围。

权利要求书

1、一种在RDMA中实现定时器处理的方法，应用于RDMA引擎中，其特征在于，至少包括如下步骤：

接收主机所生成的RDMA报文，为该报文生成一对应的超时时间节点，在其超时时间未超出预先设定的阈值时，为所述超时时间节点确定共享时间轮组中对应的时间轮以及超时时间在所述时间轮中对应的超时时间刻度；

判断共享时间轮组对应的存储单元中是否存在空闲节点；

在所述存储单元中存在空闲节点时，申请一个空闲节点以存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，由RDMA引擎进行超时判断；

在所述超时时间超出预先设定的阈值，或所述存储单元中不存在空闲节点时，将超时时间节点以事件队列的方式告知主机。

2、如权利要求1所述的方法，其特征在于，进一步包括：

在RDMA引擎中配置一个共享时间轮组，所述时间轮组包括预定第一数目个不同类别的时间轮，每个时间轮对应预定第二数目个刻度值，每个刻度值关联有一链表。

3、如权利要求2所述的方法，其特征在于，在所述存储单元中存在空闲节点时，申请一个空闲节点存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，进一步包括：

在存储单元中存在空闲节点时，进行空闲节点申请，获得一空闲的存储地址；

将所述RDMA报文的Function ID 和QPN写入至所述空闲的存储地址对应的存储空间中；

在所确定的超时时间刻度对应的链表中建立与所述空闲的存储地址之间的关联。

4、如权利要求3所述的方法，其特征在于，每个时间刻度对应一条链表，每一条链表的首、尾指针用于管理该链表。

5、如权利要求3或4所述的方法，其特征在于，由RDMA引擎进行超时判断，进一步包括：

当时间轮刻度到期时，遍历对应的链表，取出所述链表的所有超时时间节点；

将所取出的超时时间节点发送给RDMA引擎进行超时判断，并执行相应超时时间节点中的任务。

6、如权利要求5所述的方法，其特征在于，在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点以事件队列的方式告知主机，由主机管理所述超时时间，进一步包括：

在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点对应的RDMA的Function ID和QPN信息通过该Function对应的事件队列通知主机。

7、如权利要求6所述的方法，其特征在于，所述超时时间通过系统定时器确定。

8、一种在RDMA中实现定时器处理的系统，其特征在于，至少包括：

RDMA报文接收处理，用于接收主机所生成的RDMA报文，为该报文生成一对应的超时时间节点，在其超时时间未超出预先设定的阈值时，为所述超时时间节点确定共享时间轮组中对应的时间轮以及超时时间在所述时间轮中对应的超时时间刻度；

条件判断单元，用于判断共享时间轮组对应的存储单元中是否存在空闲节点；

第一处理单元，用于在所述存储单元中存在空闲节点时，申请一个空闲节点以存储所述超时时间节点的节点内容，并链入至所确定的超时时间刻度对应的链表中，由RDMA引擎进行超时判断；

第二处理单元，用于在所述超时时间超出预先设定的阈值，或所述存储单元中不存在空闲节点时，将超时时间节点以事件队列的方式告知主机。

9、如权利要求8所述的系统，其特征在于，进一步包括：

配置单元，用于在RDMA引擎中配置一个共享时间轮组，所述时间轮组包括预定第一数目个不同类别的时间轮，每个时间轮对应预定第二数目个刻度值，每个刻度值关联有一链表。

10、如权利要求9所述的系统，其特征在于，第一处理单元进一步包括：

空闲节点申请单元，用于在存储单元中存在空闲节点时，进行空闲节点申请，获得一空闲的存储地址；

写入单元，用于将所述RDMA报文的Function ID 和QPN写入至所述空闲的存储地址对应的存储空间中；

链接单元，用于在所确定的超时时间刻度对应的链表中建立与所述空闲的存储地址之间的关联。

11、如权利要求10所述的系统，其特征在于，每个时间刻度对应一条链表，每一条链表的首、尾指针用于管理该链表。

12、如权利要求10或11所述的系统，其特征在于，第一处理单元进一步包括：

遍历处理单元，用于当时间轮刻度到期时，遍历对应的链表，取出所述链表的所有超时时间节点；

第一超时处理单元，用于将所取出的超时时间节点发送给RDMA引擎进行超时判断，并执行相应超时时间节点中的任务。

13、如权利要求8所述的系统，其特征在于，第二处理单元进一步包括：

通知单元，用于在所述超时时间超出预先设定的阈值，或存储单元中不存在空闲节点时，将所述超时时间节点对应的RDMA的Function ID和QPN信息通过该Function对应的事件队列通知主机。

14、如权利要求13所述的系统，其特征在于，所述超时时间通过系统定时器确定。

15、一种芯片，其特征在于，其部署有如权利要求8至14任一项所述的系统。

16、一种计算机可读存储介质，其上存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现如权利要求1至7中任一项所述的方法的步骤。