



(12)发明专利

(10)授权公告号 CN 106980595 B

(45)授权公告日 2019.08.13

(21)申请号 201710207715.8

(22)申请日 2014.12.05

(65)同一申请的已公布的文献号
申请公布号 CN 106980595 A

(43)申请公布日 2017.07.25

(62)分案原申请数据
201410734202.9 2014.12.05

(73)专利权人 三星半导体(中国)研究开发有限
公司

地址 215021 江苏省苏州市工业园区凤里
街337号

专利权人 三星电子株式会社

(72)发明人 唐红飞

(74)专利代理机构 北京铭硕知识产权代理有限
公司 11286

代理人 曾世骁 郭鸿禧

(51)Int.Cl.
G06F 15/167(2006.01)

(56)对比文件
CN 102567275 A,2012.07.11,
CN 101178701 A,2008.05.14,
审查员 孙丹

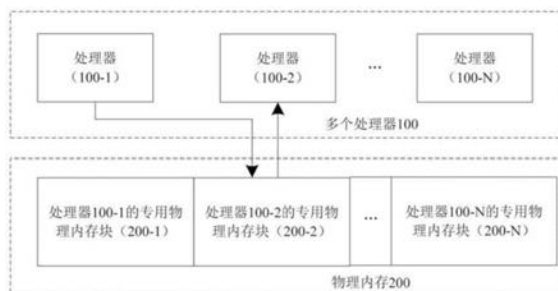
权利要求书2页 说明书11页 附图8页

(54)发明名称

共享物理内存的多处理器通信系统及其通信方法

(57)摘要

提供一种共享物理内存的多处理器通信系统及其通信方法,所述多处理器通信系统包括:多个处理器,其中,所述多个处理器互相发送和接收数据;物理内存,被划分为多个物理内存块,使得每个处理器具有一专用物理内存块;其中,所述多个处理器中的将发送数据的发送处理器将数据发送到所述多个处理器中的将接收数据的接收处理器的专用物理内存块,并且,所述接收处理器从它的专用物理内存块读取所述数据。根据所述多处理器通信系统及其通信方法,多个处理器互相传递数据时,可减少内存空间的占用,提高内存的使用效率,从而提高数据的传输速度。



1. 一种共享物理内存的多处理器通信系统,包括:
 - 多个处理器,其中,所述多个处理器互相发送和接收数据;
 - 物理内存,被划分为多个物理内存块,使得每个处理器具有一专用物理内存块;
 - 其中,所述多个处理器中的将发送数据的发送处理器将数据发送到所述多个处理器中的将接收数据的接收处理器的专用物理内存块,并且,所述接收处理器从它的专用物理内存块读取所述数据,
 - 其中,发送处理器请求接收处理器在它的专用物理内存块中分配用于缓存数据的临时数据缓存区,并且,接收处理器响应于发送处理器的请求而分配临时数据缓存区,
 - 其中,所述系统还包括:消息盒,其中,所述消息盒用于暂存若干条具备完整语义的消息,其中,所述发送处理器和所述接收处理器通过所述消息盒进行消息交互,
 - 其中,所述临时数据缓存区使用完毕后释放。
2. 如权利要求1所述的系统,其中,所述数据为突发数据,其中,用于缓存突发数据的临时数据缓存区包括单个缓存区。
3. 如权利要求1所述的系统,其中,所述数据为流数据,其中,用于缓存流数据的临时数据缓存区包括由多个分片组成的循环缓存区。
4. 如权利要求2所述的系统,其中,消息盒用于暂存突发数据消息,其中,所述突发数据消息包括指示突发数据传输事件的类型信息。
5. 如权利要求3所述的系统,其中,消息盒用于暂存流数据消息,其中,所述流数据消息包括指示流数据传输事件的类型信息。
6. 如权利要求4或5所述的系统,其中,消息盒为从物理内存中划分出的专用区域,或者,消息盒为多处理器通信系统中设置的专用硬件寄存器。
7. 如权利要求6所述的系统,其中,消息盒还暂存常驻消息,其中,所述常驻消息适用于频繁进行消息传递的场景。
8. 如权利要求1所述的系统,其中,所述多处理器通信系统集成在单芯片系统中,或者,所述多处理器通信系统集成在共享多端口物理内存的多处理器系统中。
9. 一种共享物理内存的多处理器通信系统的通信方法,其中,所述多处理器通信系统包括多个处理器和物理内存,其中,所述物理内存还包括消息盒,其中,所述消息盒用于暂存若干条具备完整语义的消息,所述多个处理器互相发送和接收数据,所述多个处理器通过所述消息盒进行消息交互,所述方法包括:
 - (a) 将物理内存划分为多个物理内存块,使得每个处理器具有一专用物理内存块;
 - (b) 由所述多个处理器中的将发送数据的发送处理器将数据发送到所述多个处理器中的将接收数据的接收处理器的专用物理内存块;以及
 - (c) 由所述接收处理器从它的专用物理内存块读取所述数据,其中,在步骤(a)和(b)之间,还包括:
 - (d) 发送处理器请求接收处理器在它的专用物理内存块中分配用于缓存数据的临时数据缓存区;
 - (e) 接收处理器响应于发送处理器的请求而分配临时数据缓存区;其中,在步骤(b)中,发送处理器将数据发送到分配的临时数据缓存区;在步骤(c)中,接收处理器从分配的临时数据缓存区读取所述数据,

其中,所述临时数据缓存区使用完毕后释放。

共享物理内存的多处理器通信系统及其通信方法

[0001] 本申请是向中国知识产权局提交的申请日为2014年12月05日、申请号为201410734202.9、发明名称为“共享物理内存的多处理器通信系统及其通信方法”的申请的分案申请。

技术领域

[0002] 本发明总体说来涉及多处理器通信系统的设计。更具体地讲,涉及一种通过共享物理内存来实现多处理器间通信的通信系统及其通信方法。

背景技术

[0003] 在现有的多处理器通信系统中,多处理器间通过串行通信接口(例如,UART、USB、SPI等)进行数据通信,利用串行通信接口进行通信时外围电路复杂,传输速度缓慢,难以在多处理器间实现数据量大的通信或对速度要求高的通信。

[0004] 而在现有的共享物理内存的多处理器通信系统中,多处理器间进行数据通信时,不需要外围电路,存取速度快,实时性强。具体说来,在通过共享物理内存进行多处理器间通信的系统中,将物理内存划分为多个专用物理内存块,这样,每个处理器独占一个专用物理内存块以用于运行其专有的软件系统,此外,针对处理器间的数据通信,还会在物理内存中预留一个物理内存块,该物理内存块是全局内存共享区域,包括保存多处理器间通信数据的数据共享缓存区和用于保存并更新数据缓存区状态的状态共享缓存区。

[0005] 通常,为了在现有的共享物理内存的多处理器通信系统进行通信,一般需要预留比较大的全局内存共享区域,而这部分内存空间需要根据特定的硬件系统进行构造(例如,产品的设计需求、处理器的实际数目、每个处理器的应用场景等)来制定相应的使用规划。这种方式存在几个明显的问题:

[0006] 第一、全局内存共享区域长期预留,比较浪费内存(全局内存共享区域一旦静态规划好,确定的数据共享缓存区和状态共享缓存区即无法被动态地释放或回收做其它用途);

[0007] 第二、全局内存共享区域的使用规划不灵活(在因某些原因而要改变内存布局的情况下,需要计算并修改各个处理器上的软件系统中的参数,较为麻烦且容易出错);

[0008] 第三、多处理器间数据通信过程需要两次数据复制的操作(数据需要从发送处理器复制到全局内存共享区域,再从全局共享内存区域复制到接收处理器),效率不高。

[0009] 因此,在现有的通过共享物理内存存在多处理器间进行通信的系统中传递数据时,物理内存的使用效率较低,数据传输的速度较慢。

发明内容

[0010] 本发明的示例性实施例在于提供一种共享物理内存的多处理器通信系统及其通信方法,以解决现有多处理器通信系统中内存使用效率低、数据传输速度慢的问题。

[0011] 根据本发明示例性实施例的一方面,提供一种共享物理内存的多处理器通信系统,包括:多个处理器,其中,所述多个处理器互相发送和接收数据;物理内存,被划分为多

个物理内存块,使得每个处理器具有一专用物理内存块;其中,所述多个处理器中的将发送数据的发送处理器将数据发送到所述多个处理器中的将接收数据的接收处理器的专用物理内存块,并且,所述接收处理器从它的专用物理内存块读取所述数据。

[0012] 在所述系统中,发送处理器可请求接收处理器在它的专用物理内存块中分配用于缓存数据的临时数据缓存区,并且,接收处理器可响应于发送处理器的请求而分配临时数据缓存区。

[0013] 在所述系统中,所述数据可以为突发数据,其中,用于缓存突发数据的临时数据缓存区可包括单个缓存区。

[0014] 在所述系统中,所述数据可以为流数据,其中,用于缓存流数据的临时数据缓存区可包括由多个分片组成的循环缓存区。

[0015] 所述系统可还包括:消息盒,暂存突发数据消息,其中,所述突发数据消息可包括指示突发数据传输事件的类型信息。

[0016] 在所述系统中,所述突发数据传输事件可包括以下事件中的至少一项:请求分配临时数据缓存区、临时数据缓存区已分配、临时数据缓存区存有待读取数据、临时数据缓存区中的数据已被读取。

[0017] 在所述系统中,当突发数据传输事件为请求分配临时数据缓存区时,所述突发数据消息可还包括关于请求分配临时数据缓存区的请求参数,其中,所述关于请求分配临时数据缓存区的请求参数包括请求的单个缓存区的大小以及应用标识、发送处理器和接收处理器的标识;当突发数据传输事件为临时数据缓存区已分配时,所述突发数据消息可还包括关于分配临时数据缓存区的分配参数,其中,所述关于分配临时数据缓存区的分配参数可包括实际分配的单个缓存区的大小和位置以及应用标识、发送处理器和接收处理器的标识;当突发传输事件为临时数据缓存区存有待读取数据时,所述突发数据消息可还包括关于临时数据缓存区存有待读取数据的参数,其中,所述关于临时数据缓存区存有待读取数据的参数可包括单个缓存区的地址、待读取数据的长度以及应用标识、发送处理器和接收处理器的标识;当突发传输事件为临时数据缓存区中的数据已被读取时,所述突发数据消息还包括关于临时数据缓存区中的数据已被读取的参数,其中,所述关于临时数据缓存区中的数据已被读取的参数可包括单个缓存区的地址以及应用标识、发送处理器和接收处理器的标识。

[0018] 所述系统可还包括:消息盒,暂存流数据消息,其中,所述流数据消息可包括指示流数据传输事件的类型信息。

[0019] 在所述系统中,所述流数据传输事件可包括以下事件中的至少一项:请求建立流连接、流连接已建立、分片状态发生改变、请求断开流连接、流连接已断开。

[0020] 在所述系统中,当流数据传输事件为请求建立流连接时,所述流数据消息还可包括关于请求建立流连接的请求参数,其中,所述关于请求建立流连接的请求参数可包括组成请求的循环缓存区的分片的数目和每个分片的大小以及应用标识、发送处理器和接收处理器的标识;当流数据传输事件为流连接已建立时,所述流数据消息可还包括关于流连接的唯一标识和实际分配临时数据缓存区的分配参数,其中,所述关于流连接的唯一标识用于全局唯一标识已建立的流连接;所述关于实际分配临时数据缓存区的分配参数可包括组成分配的循环缓存区的分片的数目以及每个分片的大小、位置和编号以及应用标识、发送

处理器和接收处理器的标识;当流数据传输事件为请求断开流连接时,所述流数据消息可还包括关于断开流连接的请求参数,其中,所述关于断开流连接的请求参数可包括关于流连接的唯一标识以及应用标识、发送处理器和接收处理器的标识;当流数据传输事件为流连接已断开时,所述流数据消息可还包括关于流连接已断开的参数,其中,所述关于流连接已断开的参数可包括关于流连接的唯一标识以及应用标识、发送处理器和接收处理器的标识。

[0021] 在所述系统中,消息盒可以为从物理内存中划分出的专用区域,或者,消息盒可以为多处理器通信系统中设置的专用硬件寄存器。

[0022] 在所述系统中,消息盒还可暂存常驻消息,其中,所述常驻消息适用于频繁进行消息传递的场景。

[0023] 在所述系统中,所述多处理器通信系统可集成在单芯片系统中,或者,所述多处理器通信系统可集成在共享多端口物理内存的多处理器系统中。

[0024] 根据本发明示例性实施例的另一方面,提供一种共享物理内存的多处理器通信系统的通信方法,其中,所述多处理器通信系统包括多个处理器和物理内存,所述多个处理器互相发送和接收数据,所述方法包括:(a)将物理内存划分为多个物理内存块,使得每个处理器具有一专用物理内存块;(b)由所述多个处理器中的将发送数据的发送处理器将数据发送到所述多个处理器中的将接收数据的接收处理器的专用物理内存块;以及(c)由所述接收处理器从它的专用物理内存块读取所述数据。

[0025] 在所述方法中,在步骤(a)和(b)之间,可还包括:(d)发送处理器请求接收处理器在它的专用物理内存块中分配用于缓存数据的临时数据缓存区;(e)接收处理器响应于发送处理器的请求而分配临时数据缓存区;其中,在步骤(b)中,发送处理器将数据发送到分配的临时数据缓存区;在步骤(c)中,接收处理器从分配的临时数据缓存区读取所述数据。

[0026] 在所述方法中,所述数据可以为突发数据,其中,用于缓存突发数据的临时数据缓存区可包括单个缓存区。

[0027] 在所述方法中,所述数据可以为流数据,其中,用于缓存流数据的临时数据缓存区可包括由多个分片组成的循环缓存区。

[0028] 在根据本发明示例性实施例的共享物理内存的多处理器通信系统及其通信方法中,不需要预留全局内存共享区域,多个处理器可通过各自的专用物理内存块互相传递数据,从而提高内存的使用效率,并加快数据的传输速度。

附图说明

[0029] 通过下面结合示例性地示出实施例的附图进行的详细描述,本发明示例性实施例的上述和其他目的将会变得更加清楚,其中:

[0030] 图1示出根据本发明示例性实施例的共享物理内存的多处理器通信系统的框图;

[0031] 图2示出根据本发明示例性实施例的共享物理内存的多处理器通信系统的通信方法的流程图;

[0032] 图3示出根据本发明示例性实施例的共享物理内存的多处理器进行通信的示例;

[0033] 图4示出根据本发明另一示例性实施例的共享物理内存的多处理器通信系统的通信方法的流程图;

[0034] 图5示出根据本发明示例性实施例的消息盒的示例；

[0035] 图6示出根据本发明示例性实施例的共享物理内存的多处理器针对突发数据进行通信的示例；

[0036] 图7A示出根据本发明示例性实施例的多处理器通信系统中的发送处理器针对突发数据进行通信的方法的流程图；

[0037] 图7B示出根据本发明示例性实施例的多处理器通信系统中的接收处理器针对突发数据进行通信的方法的流程图；

[0038] 图8示出根据本发明示例性实施例的共享物理内存的多处理器针对流数据进行通信的示例；

[0039] 图9A示出根据本发明示例性实施例的多处理器通信系统中的发送处理器针对流数据进行通信的方法的流程图；

[0040] 图9B示出根据本发明示例性实施例的多处理器通信系统中的接收处理器针对流数据进行通信的方法的流程图。

具体实施方式

[0041] 现将详细参照本发明的示例性实施例,所述实施例的示例在附图中示出,其中,相同的标号始终指的是相同的部件。

[0042] 图1示出根据本发明示例性实施例的共享物理内存的多处理器通信系统的框图。

[0043] 如图1所示,根据本发明示例性实施例的共享物理内存的多处理器通信系统(以下简称:“多处理器通信系统”)包括:多个处理器100、物理内存200。这里,作为示例,所述多处理器通信系统可集成在单芯片系统中,或者,所述多处理器通信系统可集成在共享多端口物理内存的多处理器系统中。所述处理器100可以是中央处理器、微处理器、物理处理器、数字信号处理器等处理器。所述物理内存200可以是随机存取存储器(RAM)。

[0044] 这里,所述多个处理器100(例如,图1所示的处理器100-1、处理器100-2、•••••、处理器100-N)互相发送和接收数据。

[0045] 此外,物理内存200被划分为多个物理内存块,使得每个处理器具有一专用物理内存块。如图1所示,物理内存200可被划分为:处理器100-1的专用物理内存块200-1、处理器100-2的专用物理内存块200-2、•••••、处理器100-N的专用物理内存块200-N。

[0046] 相应地,所述多个处理器100中的将发送数据的发送处理器将数据发送到所述多个处理器100中的将接收数据的接收处理器的专用物理内存块,并且,所述接收处理器从它的专用物理内存块读取所述数据。这里,在以下的示例性实施例中,为了便于描述,假设处理器100-1指示将发送数据的发送处理器,处理器100-2指示将接收数据的接收处理器。这里,应理解,发送处理器和接收处理器不局限于处理器100-1和处理器100-2,还可以是多个处理器100中的其他处理器。此外,单个处理器既可用于发送数据,也可用于接收数据。也就是说,处理器在发送数据时被用作发送处理器,而在接收数据时被用作接收处理器。

[0047] 具体说来,处理器100-1可将数据发送到处理器100-2的专用物理内存块200-2,并且,处理器100-2从所述处理器100-2的专用物理内存块200-2读取所述数据。

[0048] 在上述根据本发明示例性实施例的共享物理内存的多处理器通信系统中,不需要预留全局内存共享区域,多个处理器可通过各自的专用物理内存块互相传递数据,从而提

高内存的使用效率,并加快数据的传输速度。

[0049] 图2示出根据本发明示例性实施例的共享物理内存的多处理器通信系统的通信方法(以下,称为“多处理器通信方法”)的流程图。这里,所述多处理器通信系统包括多个处理器和物理内存,并且,所述多个处理器互相发送和接收数据。作为示例,所述处理器可以是中央处理器、微处理器、物理处理器、数字信号处理器等处理器。所述物理内存可以是随机存取存储器(RAM)。此外,作为示例,所述多处理器通信系统可集成在单芯片系统中,或者,所述多处理器通信系统可集成在共享多端口物理内存的多处理器系统中。

[0050] 如图2所示,在步骤S100,将物理内存划分为多个物理内存块,使得每个处理器具有一专用物理内存块。

[0051] 在步骤S200,由所述多个处理器中的将发送数据的发送处理器将数据发送到所述多个处理器中的将接收数据的接收处理器的专用物理内存块。在步骤S300,由所述接收处理器从它的专用物理内存块读取所述数据。这里,作为示例,单个处理器既可用于发送数据,也可用于接收数据。也就是说,处理器在发送数据时被用作发送处理器,而在接收数据时被用作接收处理器。

[0052] 在上述根据本发明示例性实施例的共享物理内存的多处理器通信系统的通信方法中,不需要预留全局内存共享区域,多个处理器可通过各自的专用物理内存块互相传递数据,从而提高内存的使用效率,并加快数据的传输速度。

[0053] 图3示出根据本发明示例性实施例的共享物理内存的多处理器进行通信的示例。

[0054] 作为示例,如图3所示,多个处理器100可包括发送处理器(即,处理器100-1)和接收处理器(即,处理器100-2),其中,处理器100-1请求处理器100-2在它的专用物理内存块200-2中分配用于缓存数据的临时数据缓存区B,并且,处理器100-2响应于处理器100-1的请求而分配临时数据缓存区B。应理解,所述缓存区B被用于暂时存放发送处理器期望发送到接收处理器的数据。

[0055] 图4示出根据本发明另一示例性实施例的共享物理内存的多处理器通信系统的通信方法的流程图。

[0056] 如图4所示,在步骤S100,将物理内存划分为多个物理内存块,使得每个处理器具有一专用物理内存块。在步骤S400,发送处理器可请求接收处理器在它的专用物理内存块中分配用于缓存数据的临时数据缓存区。在步骤S500,接收处理器可响应于发送处理器的请求而分配临时数据缓存区。在步骤S200,发送处理器可将数据发送到分配的临时数据缓存区。在步骤S300,接收处理器可从分配的临时数据缓存区读取所述数据。

[0057] 这里,根据实际的应用场景的不同以及临时数据缓存区的分配和释放的策略的不同,多处理器通信系统中的多处理器通信的数据可包括突发数据(Burst Data)和流数据(Stream Data)。

[0058] 此外,作为附加部件,所述多处理器通信系统可还包括消息盒(Message Box),其中,所述消息盒可用于暂存若干条具备完整语义的消息。根据本发明示例性实施例的多处理器通信系统中的每个处理器都可以访问消息盒。作为示例,消息盒可以从物理内存中划分出的专用区域(即,物理内存的一部分),或者,消息盒可以是多处理器通信系统中设置的专用硬件寄存器。应注意,在图3所示的示例性实施例中,所述消息盒210为从物理内存中划分出的专用区域(即,物理内存200的一部分)。

[0059] 下面,将结合图5来描述根据本发明示例性实施例的消息盒210的示例。

[0060] 具体说来,如图5所示,消息盒210可被划分为X个单元槽(SLOT),其中,所述单元槽可被设计成固定的相同大小。一般地,依据实际应用,X可取16、32、64或其它整数。每一个单元槽可以承载一条具备完整语意的消息。这里,所述消息可以是数据通信的一方请求另一方执行某操作(例如,发送处理器请求接收处理器分配临时数据缓存区)的命令,或者,所述消息可以是命令执行(例如,接收处理器响应于发送处理器的命令而分配临时数据缓存区)之后的响应。这里,作为示例,每一条消息都包含了通信双方的处理器标识,用于指示消息的发出者和消息的接收者。此外,所述单元槽还可承载常驻消息,所述承载常驻消息的单元槽将在较长时间内被占用,不会被设置为空闲可用状态,从而通信双方将优先使用该单元槽,通过这种方式,可提高消息传递的效率。

[0061] 图6示出根据本发明示例性实施例的共享物理内存的多处理器针对突发数据进行通信的示例。

[0062] 如图6所示,作为示例,在发送处理器期望发送到接收处理器的数据为突发数据的情况下,用于缓存突发数据的临时数据缓存区B可包括单个缓存区B1。应理解,突发数据可指示数据通信中以突发方式传输的数据。在这种情况下,消息盒210中可暂存包括指示突发数据传输事件的类型信息的突发数据消息,这里,作为示例,所述突发数据传输事件可包括以下事件中的至少一项:请求分配临时数据缓存区、临时数据缓存区已分配、临时数据缓存区存有待读取数据、临时数据缓存区中的数据已被读取。此外,作为示例,当突发数据传输事件为请求分配临时数据缓存区时,所述突发数据消息还包括关于请求分配临时数据缓存区的请求参数,其中,所述关于请求分配临时数据缓存区的请求参数包括请求的单个缓存区B1的大小以及应用标识(用于区分同一个处理器上的不同的请求)、发送处理器和接收处理器的标识;当突发数据传输事件为临时数据缓存区已分配时,所述突发数据消息还包括关于分配临时数据缓存区的分配参数,其中,所述关于分配临时数据缓存区的分配参数包括实际分配的单个缓存区B1的大小和位置以及应用标识、发送处理器和接收处理器的标识;当突发传输事件为临时数据缓存区存有待读取数据时,所述突发数据消息还包括关于临时数据缓存区存有待读取数据的参数,其中,所述关于临时数据缓存区存有待读取数据的参数包括单个缓存区的地址、待读取数据的长度以及应用标识、发送处理器和接收处理器的标识;当突发传输事件为临时数据缓存区中的数据已被读取时,所述突发数据消息还包括关于临时数据缓存区中的数据已被读取的参数,其中,所述关于临时数据缓存区中的数据已被读取的参数包括单个缓存区的地址以及应用标识、发送处理器和接收处理器的标识。

[0063] 以下将结合图7A和图7B来描述根据本发明示例性实施例的多处理器通信系统针对突发数据进行通信的方法。其中,图7A示出根据本发明示例性实施例的多处理器通信系统中的发送处理器针对突发数据进行通信的方法的流程图;图7B示出根据本发明示例性实施例的多处理器通信系统中的接收处理器针对突发数据进行通信的方法的流程图。

[0064] 具体说来,如图7A所示,在步骤S101,发送处理器(即,处理器100-1)进行初始化。如图7B所示,在步骤S201,接收处理器(即,处理器100-2)进行初始化。

[0065] 再次参照图7A,在步骤S102,处理器100-1构造一条突发数据消息,该突发数据消息可包括指示“请求分配临时数据缓存区”这一突发数据传输事件的类型信息(这里,类型

信息用于指示突发数据传输事件)以及相应的请求参数(这里,请求参数可包括请求分配的缓存区的大小、应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的突发数据消息写入消息盒中的某一空闲槽中。在步骤S103,处理器100-1向处理器100-2发送中断。

[0066] 再次参照图7B,在步骤S202,处理器100-2等待处理器100-1发送的中断。在处理器100-2从处理器100-1接收到中断时,在步骤S203,从消息盒中读取突发数据消息并获取类型信息。在步骤S204,判断所述类型信息用于指示哪种突发数据传输事件。在根据类型信息判断突发数据传输事件为请求分配临时数据缓存区的情况下,在步骤S205,获取关于分配临时数据缓存区的请求参数。在步骤S206,根据请求参数分配单个缓存区。在步骤S207,处理器100-2构造一条突发数据消息,该突发数据消息可包括指示“临时数据缓存区已分配”这一突发数据传输事件的类型信息以及相应的分配参数(这里,分配参数可包括实际分配的单个缓存区的大小和位置、应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的突发数据消息写入消息盒中的某一空闲槽中。在步骤S208,处理器100-2向处理器100-1发送中断。然后处理器100-2返回执行步骤S202。

[0067] 返回再次参照图7A,在步骤S104,处理器100-1等待处理器100-2发送的中断。在处理器100-1从处理器100-2接收到中断时,在步骤S105,从消息盒中读取突发数据消息并获取类型信息。在步骤S106,判断所述类型信息用于指示哪种突发数据传输事件。在根据类型信息判断突发数据传输事件为临时数据缓存区已分配的情况下,在步骤S107,获取关于分配的临时数据缓存区的分配参数。在步骤S108,根据分配参数将数据写入单个缓存区。在步骤S109,处理器100-1构造一条突发数据消息,该突发数据消息包括指示“临时数据缓存区存有待读取数据”这一突发数据传输事件的类型信息以及相应的参数(这里,参数可包括单个缓存区的地址、待读取数据的长度以及应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的突发数据消息写入消息盒中的某一空闲槽中。在步骤S110,处理器100-1向处理器100-2发送中断。然后处理器100-1返回执行步骤S104。

[0068] 返回再次参照图7B,在处理器100-2从处理器100-1接收到中断时,在步骤S203,从消息盒中读取突发数据消息并获取类型信息。在步骤S204,判断所述类型信息用于指示哪种突发数据传输事件。在根据类型信息判断突发数据传输事件为临时数据缓存区存有待读取数据的情况下,在步骤S209,从单个缓存区中读取处理器100-1写入的数据。在步骤S210,释放单个缓存区。在步骤S211,处理器100-2构造一条突发数据消息,该突发数据消息可包括指示“临时数据缓存区中的数据已被读取”的类型信息以及相应的参数(这里,参数包括单个缓存区的地址以及应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的突发数据消息写入消息盒中的某一空闲槽中。在步骤S212,处理器100-2向处理器100-1发送中断。在步骤S213,判断是否退出本次数据通信,若确定退出本次数据通信,则结束本次数据通信;若确定不退出本次数据通信,则返回执行步骤S202。

[0069] 返回再次参照图7A,在处理器100-1从处理器100-2接收到中断时,在步骤S105,从消息盒中读取突发数据消息并获取类型信息。在步骤S106,判断所述类型信息用于指示哪种突发数据传输事件。在根据类型信息判断突发数据传输事件为临时数据缓存区中的数据已被读取的情况下,在步骤S111,判断数据是否全部已发送,若数据全部已发送,则结束本次数据通信;若数据未全部发送,则返回执行步骤S102。

[0070] 通过上述方式发送突发数据时,传输方式简单,突发数据所需的单个缓存区可以即用即分配,用完即释放,防止了内存的泄露。由于单个缓存区的分配和释放需要一定的系统开销,因此,优选地,可将这种传输方式适用于少量的、偶发的数据通信情景。

[0071] 图8示出根据本发明示例性实施例的共享物理内存的多处理器针对流数据进行通信的示例。

[0072] 如图8所示,作为另一示例,在发送处理器期望发送到接收处理器的数据为流数据的情况下,用于缓存流数据的临时数据缓存区B可包括循环缓存区B2,其中,所述循环缓存区B2可由多个分片(例如,B2-1、B2-2、•••••、B2-K)组成。应理解,流数据可指示连续传输的数据序列。在这种情况下,消息盒210中可暂存包括指示流数据传输事件的类型信息的流数据消息,这里,作为示例,所述流数据传输事件可包括以下事件中的至少一项:请求建立流连接、流连接已建立、分片状态发生改变、请求断开流连接、流连接已断开。此外,作为示例,当流数据传输事件为请求建立流连接时,所述流数据消息可还包括关于请求建立流连接的请求参数,其中,所述关于请求建立流连接的请求参数可包括组成请求的循环缓存区B2的分片的数目和每个分片的大小以及应用标识、发送处理器和接收处理器的标识;当流数据传输事件为流连接已建立时,所述流数据消息可还包括关于流连接的唯一标识和实际分配临时数据缓存区的分配参数,其中,所述关于流连接的唯一标识用于全局唯一标识已建立的流连接;所述关于实际分配临时数据缓存区的分配参数可包括组成分配的循环缓存区B2的分片的数目以及每个分片的大小、位置和编号以及应用标识、发送处理器和接收处理器的标识;当流数据传输事件为请求断开流连接时,所述流数据消息可还包括关于断开流连接的请求参数,其中,所述关于断开流连接的请求参数可包括流连接的唯一标识以及应用标识、发送处理器和接收处理器的标识;当流数据传输事件为流连接已断开时,所述流数据消息可还包括关于流连接已断开的参数,其中,所述关于流连接已断开的参数包括流连接的唯一标识以及应用标识、发送处理器和接收处理器的标识。此外,作为示例,消息盒210还暂存常驻消息,其中,所述常驻消息可指示循环缓存区B2的每个分片的状态(例如,可指示循环缓存区B2的每个分片中是否存有待读取的数据以及待读取数据的长度),换句话说,常驻消息可保存指示循环缓存区B2的每个分片中是否缓存有数据的信息(例如,“有数据”或者“无数据”)以及“有数据”时待读取数据的长度。

[0073] 以下将结合图9A和图9B来描述根据本发明示例性实施例的多处理器通信系统针对流数据进行通信的方法。其中,图9A示出根据本发明示例性实施例的多处理器通信系统中的发送处理器针对流数据进行通信的方法的流程图;图9B示出根据本发明示例性实施例的多处理器通信系统中的接收处理器针对流数据进行通信的方法的流程图。

[0074] 具体说来,如图9A所示,在步骤S301,发送处理器(即,处理器100-1)进行初始化。如图9B所示,在步骤S401,接收处理器(即,处理器100-2)进行初始化。

[0075] 再次参照图9A,在步骤S302,处理器100-1构造一条流数据消息,该流数据消息可包括指示“请求建立流连接”这一流数据传输事件的类型信息(这里,类型信息用于指示流数据传输事件)以及相应的请求参数(这里,请求参数可包括请求的循环缓存区的分片的数目和每个分片的大小以及应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的流数据消息写入消息盒中的某一空闲槽中。在步骤S303,处理器100-1向处理器100-2发送中断。

[0076] 再次参照图9B,在步骤S402,处理器100-2等待处理器100-1发送的中断。在处理器100-2从处理器100-1接收到中断时,在步骤S403,从消息盒中读取流数据消息并获取类型信息。在步骤S404,判断所述类型信息用于指示哪种流数据传输事件。在根据类型信息判断流数据传输事件为请求建立流连接的情况下,在步骤S405,获取关于请求建立流连接的请求参数。在步骤S406,根据请求参数分配包括多个分片的循环缓存区。在步骤S407,处理器100-2构造一条流数据消息,该流数据消息可包括关于流连接的唯一标识、指示“流连接已建立”的类型信息、应用标识、处理器100-1和处理器100-2的标识等信息以及相应的分配参数(这里,分配参数可包括实际分配的循环缓存区的分片的数目以及每个分片的大小、位置和编号),然后将所构造的流数据消息写入消息盒中的某一空闲槽中。在步骤S408,处理器100-2向处理器100-1发送中断,并返回执行步骤S402。

[0077] 返回再次参照图9A,在步骤S304,处理器100-1等待处理器100-2发送的中断。在处理器100-1从处理器100-2接收到中断时,在步骤S305,从消息盒中读取流数据消息并获取类型信息。在步骤S306,判断所述类型信息用于指示哪种流数据传输事件。在根据类型信息判断流数据传输事件为流连接已建立的情况下,在步骤S307,获取流连接的唯一标识、应用标识以及关于实际分配临时数据缓存区的分配参数,并根据分配参数得到分片的数目以及每个分片的大小、位置和编号。处理器100-1再构造一条常驻消息,其中,该常驻消息包含了循环缓冲区各分片的状态信息,然后处理器100-1将所构造的常驻消息写入消息盒中的某一空闲槽中(常驻消息并非必须,而是可选地,它在流连接的整个数据传输周期内始终驻留在其所在的槽中,借助于该常驻消息,处理器100-1和处理器100-2可以在数据传输时快速交换循环缓冲区各分片的状态信息)。在步骤S308,处理器100-1等待有数据发送。在步骤S309,判断数据是否全部已发送。在数据未全部已发送的情况下,在步骤S310,从常驻消息中依次读取分片的状态信息,确定是否存在“无数据”分片。在存在“无数据”分片的情况下,在步骤S311,依次将数据写入循环缓存区的“无数据”分片。在步骤S312,将状态发生改变的分片的状态(包括该分片中待读取的数据长度)以及将指示“分片状态发生改变”的类型信息更新到消息盒中的常驻消息中,以表明所述分片存有待读取的数据。在步骤S313,处理器100-1向处理器100-2发送中断,返回执行步骤S309。另一方面,在没有“无数据”分片的情况下,返回执行步骤S304。

[0078] 具体说来,在步骤S311,当处理器100-1向循环缓存区中的一个“无数据”分片B2-1写入数据后,在步骤S312,处理器100-1可将常驻消息中暂存的分片B2-1的状态信息改为“有数据”状态(包括暂存该分片中待读取的数据长度),同时处理器100-1在步骤S313向处理器100-2发送中断。其中,在步骤S309和S310,若处理器100-1还有数据需要向分片写入且存在“无数据”分片,则可继续向分片写入数据。若处理器100-1还有数据需要向分片写入但所有分片均为“有数据”,则中止处理器100-1向分片写入数据的处理,并返回步骤S304,等待处理器100-2向处理器100-1发送的中断。

[0079] 另一方面,在数据全部已发送的情况下,在步骤S314,处理器100-1判断是否断开流连接。在断开流连接的情况下,在步骤S315,处理器100-1将构造一条流数据消息,该流数据消息可包括指示“请求断开流连接”的类型信息以及相应的请求参数(这里,请求参数可包括流连接的唯一标识、应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的流数据消息写入消息盒中的某一空闲槽中(如果传输数据过程中使用了常驻消息,

则此时常驻消息已不再需要,可以直接使用该常驻消息所在的槽来暂存待发送的消息)。在步骤S316,处理器100-1向处理器100-2发送中断,返回执行步骤S304。

[0080] 返回再次参照图9B,在步骤S402,处理器100-2等待处理器100-1发送的中断。在处理器100-2从处理器100-1接收到中断时,在步骤S403,从消息盒中读取流数据消息并获取类型信息。在步骤S404,判断所述类型信息用于指示哪种流数据传输事件。在根据类型信息判断流数据传输事件为分片状态发生改变(即,分片存有待读取的数据)的情况下,在步骤S409,依次从消息盒中的常驻消息中确定是否存在有待读取的数据的分片,这里,处理器100-2从常驻消息中确定是否存在有待读取的数据的分片时,可从上一次读取的分片开始确定后续分片的状态,而不是从头确定每个分片的状态。在存在“有数据”分片的情况下,在步骤S410,从“有数据”分片中读取数据并清空该分片。在步骤S411,将指示“分片状态发生改变”的类型信息、以及状态发生改变的分片的状态更新到消息盒中的常驻消息中,以表明所述分片中无数据。在步骤S412,处理器100-2向处理器100-1发送中断,返回执行步骤S409。也就是说,每当处理器100-2从处理器100-1接收到用于通知分片存有待读取的数据的中断时,则处理器100-2将依次从“有数据”分片读取数据,直到不存在可读取的数据(即,读取到“无数据”分片),在这种情况下,处理器100-2停止读取数据,并返回步骤S402以等待处理器100-1之后再次发送中断。

[0081] 另一方面,在根据类型信息判断流数据传输事件为请求断开流连接的情况下,在步骤S413,处理器100-2获取关于断开流连接的请求参数。在步骤S414,根据所述关于断开流连接的请求参数释放分配的循环缓存区,在步骤S415,处理器100-2构造一条流数据消息,该流数据消息可包括指示“流连接已断开”的类型信息以及相应的参数(这里,参数包括流连接的唯一标识、应用标识、处理器100-1和处理器100-2的标识等信息),然后将所构造的流数据消息写入消息盒中的某一空闲槽中,在步骤S416,处理器100-2向处理器100-1发送中断。最后结束本次数据通信。

[0082] 返回再次参照图9A,在步骤S304,处理器100-1等待处理器100-2发送的中断。在处理器100-1从处理器100-2接收到中断时,在步骤S305,从消息盒中读取流数据消息并获取类型信息。在步骤S306,判断所述类型信息用于指示哪种流数据传输事件。在根据类型信息判断流数据传输事件为分片中的数据已被读取的情况下,执行步骤S309。另一方面,在根据类型信息判断流数据传输事件为流连接已断开的情况下,结束本次数据通信。

[0083] 通过上述方式发送流数据时,可以循环重复利用循环缓存区,不需要频繁的执行缓存区的分配与释放操作,提高了通信效率,因此,这种传输方式适合大量、频繁传输数据的数据通信情景。流连接的管理策略可以有效地防止内存的泄露,例如,当通信的一方不需要进行通信时,可以通知通信的另一方断开流连接,使得负责分配缓冲区的数据接收处理器能够及时地释放循环缓存区,防止内存泄露,避免引起数据接收处理器的错误(例如,系统崩溃)。

[0084] 综上所述,在根据本发明示例性实施例的共享物理内存的多处理器通信系统及其通信方法中,不需要预留全局内存共享区域,多个处理器可通过各自的专用物理内存块互相传递数据,从而提高内存的使用效率,并且,数据传输中减少了数据的复制次数,从而加快数据的传输速度。

[0085] 应注意,本发明的以上各个实施例仅仅是示例性的,而本发明并不受限于此。本领域

域技术人员应该理解：在不脱离本发明的原理和精神的情况下，可对这些实施例进行改变，其中，本发明的范围在权利要求及其等同物中限定。

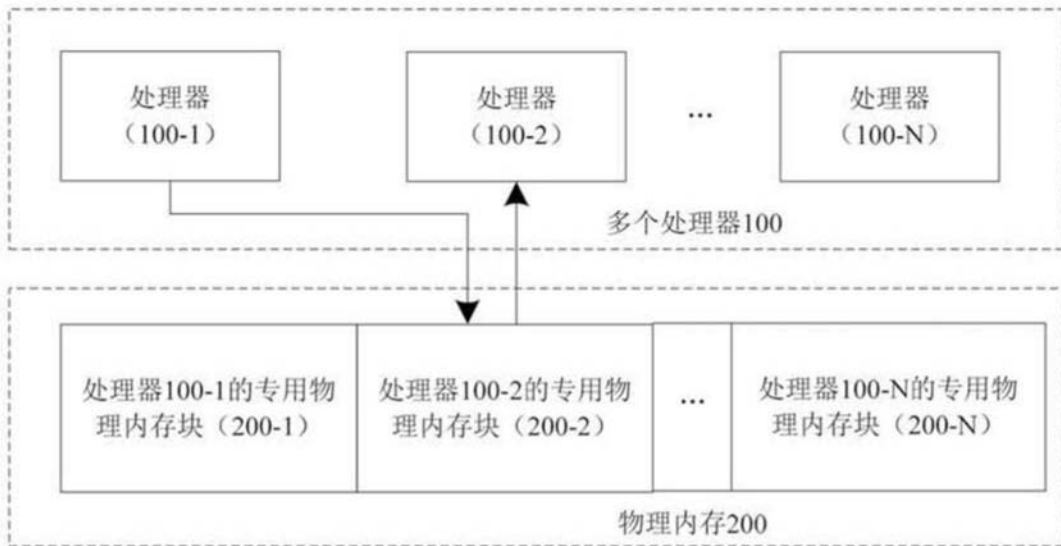


图1

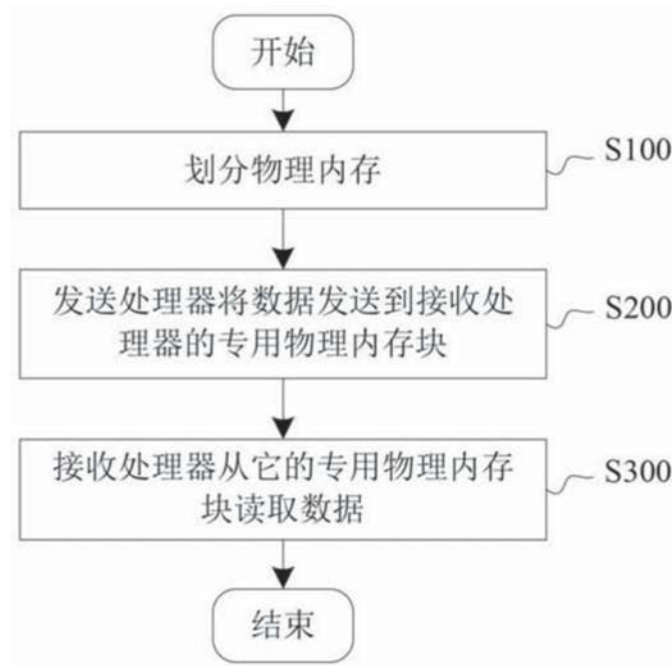


图2

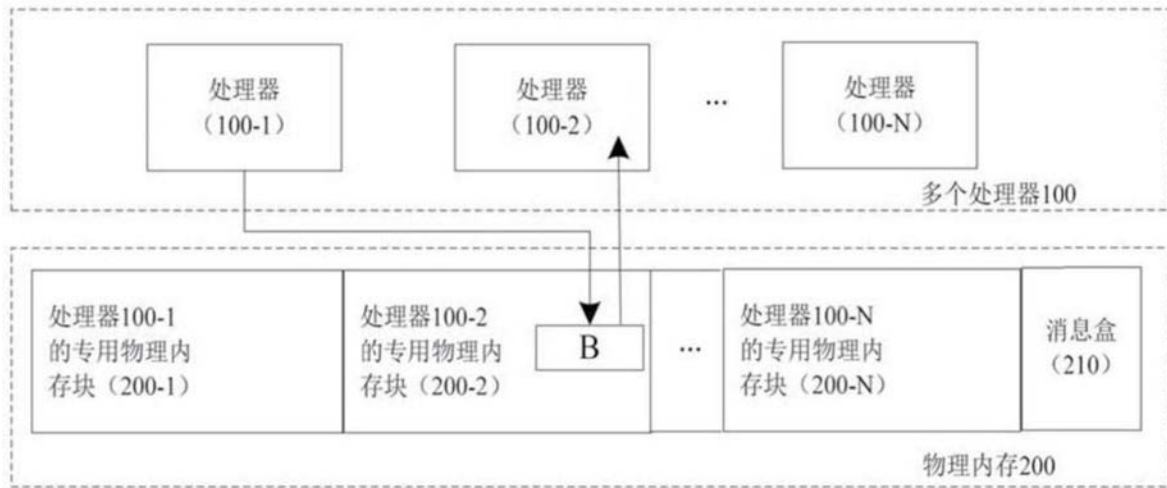


图3

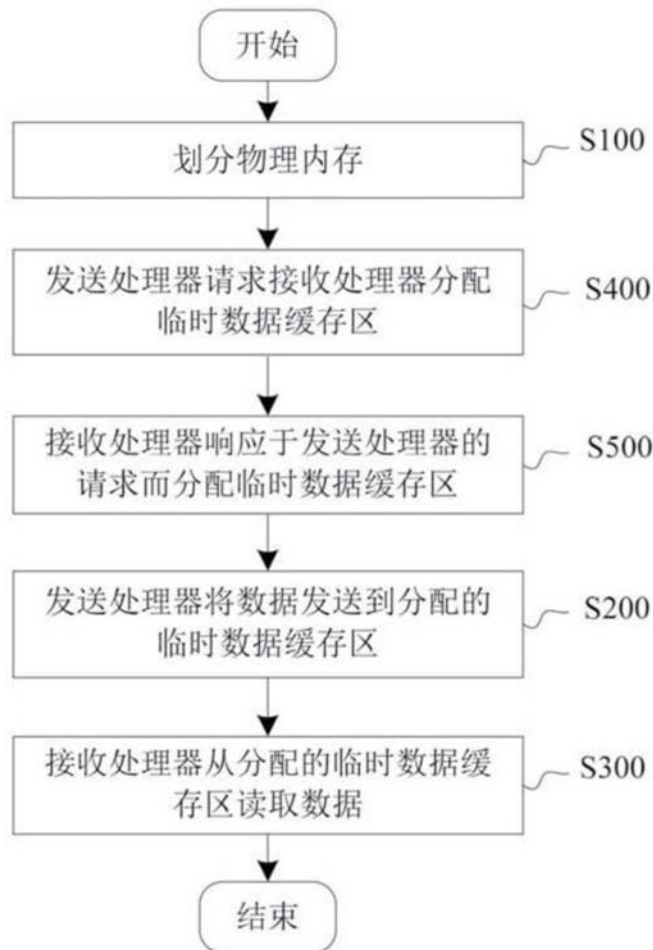


图4



图5

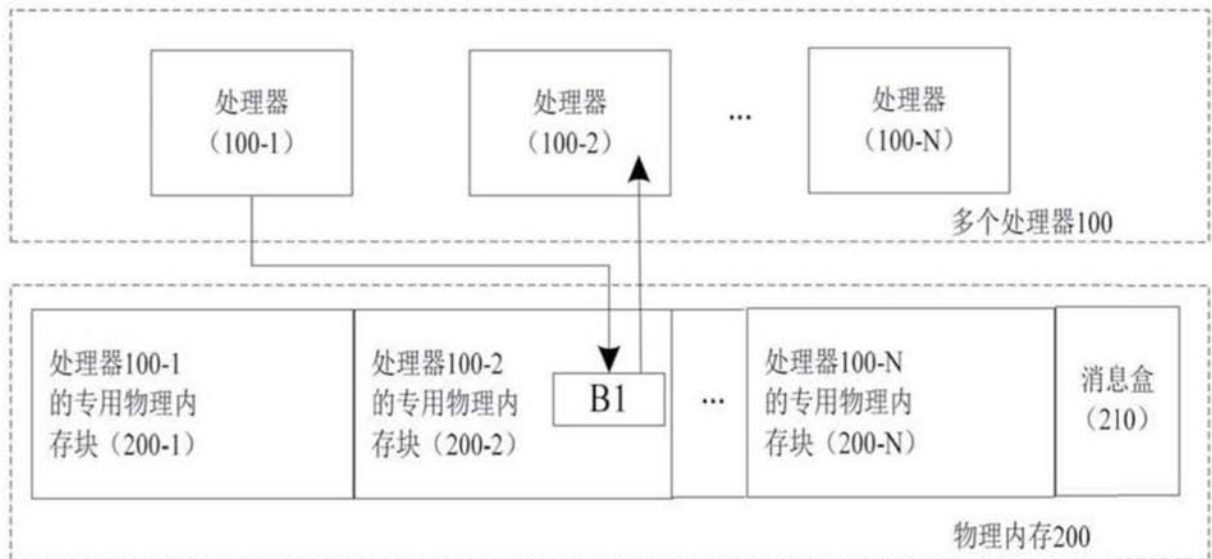


图6

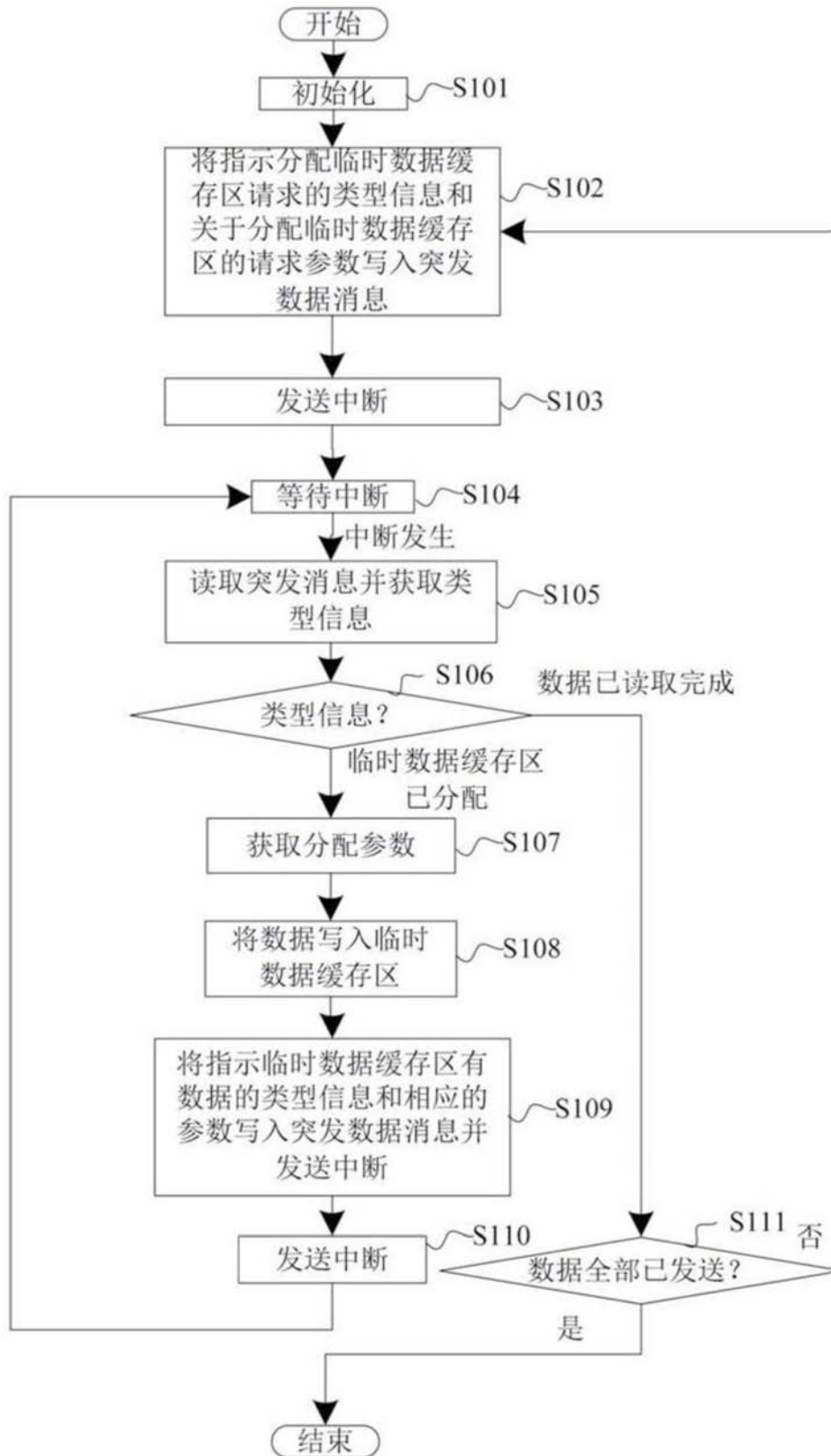


图7A

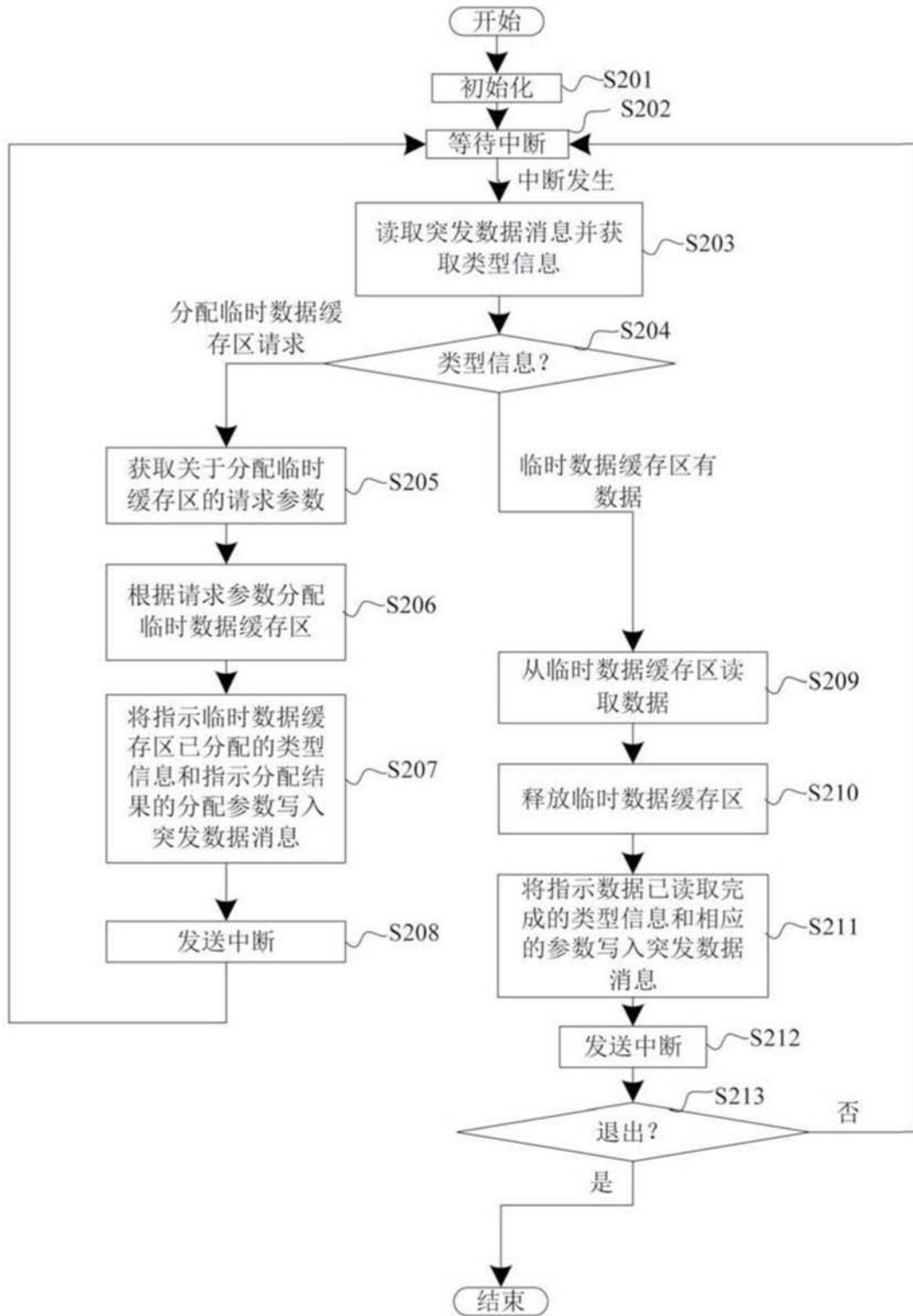


图7B

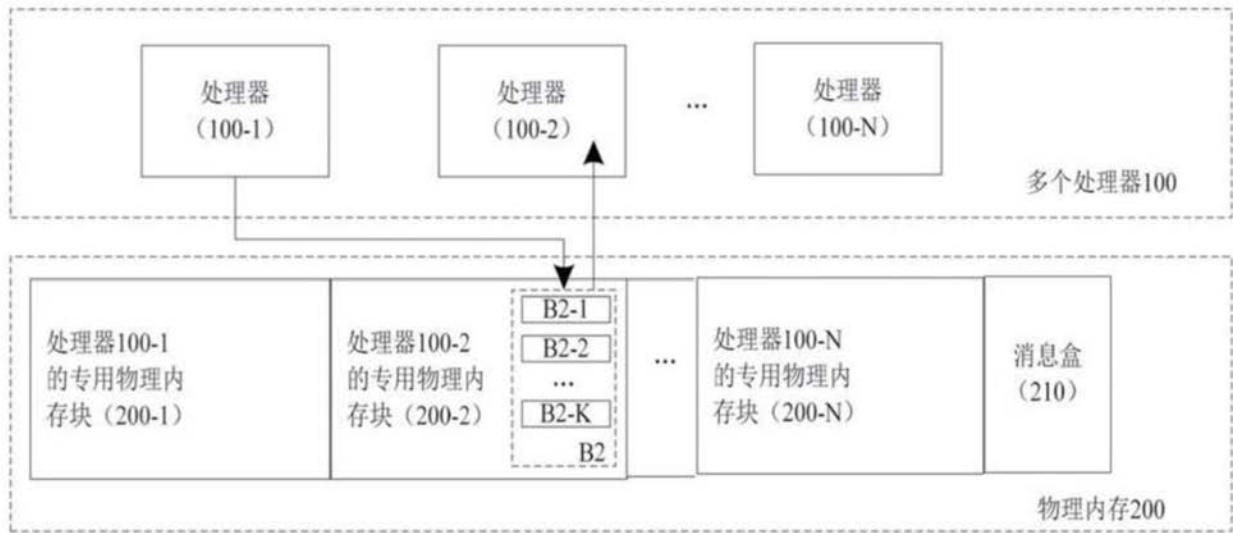


图8

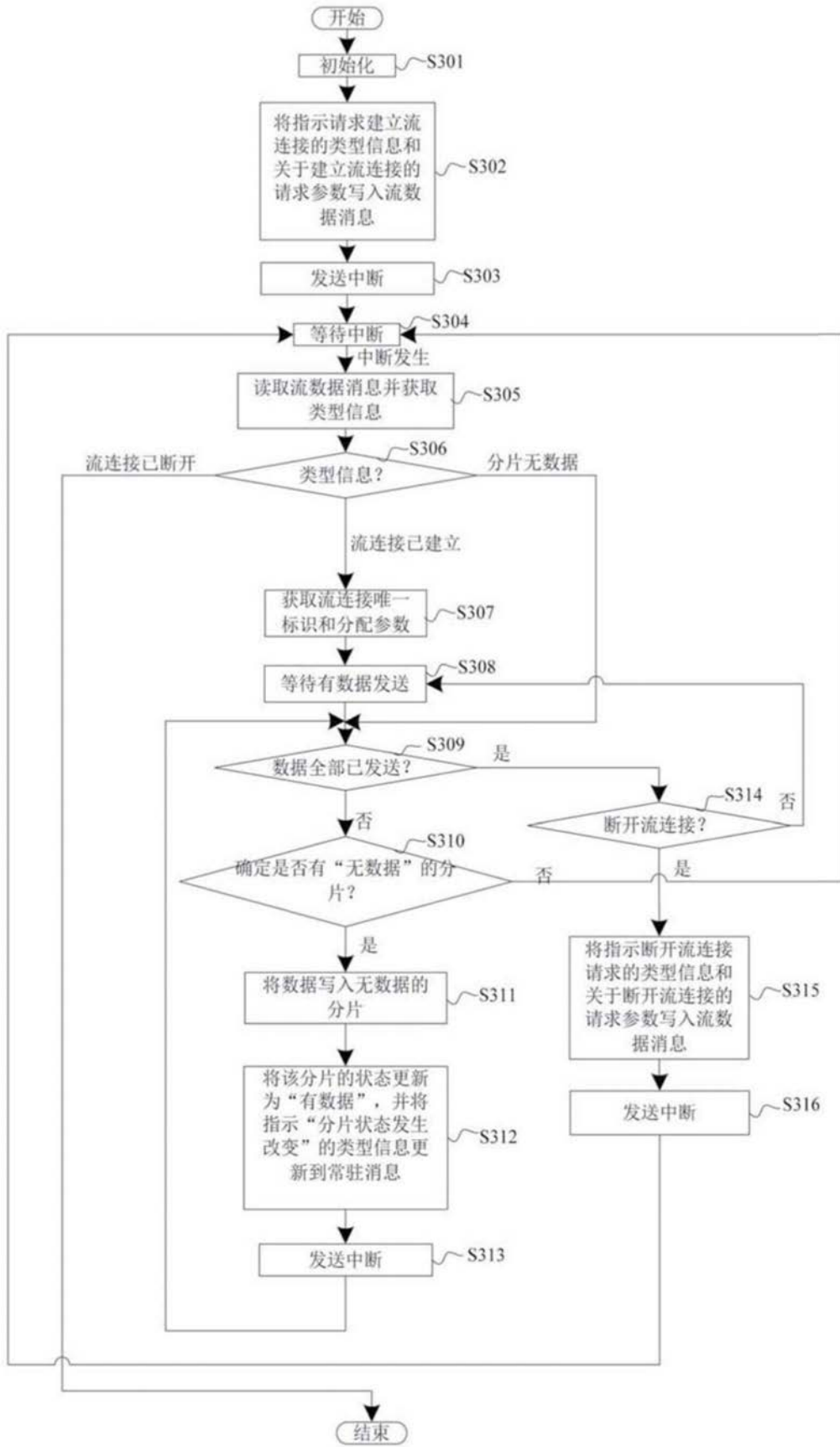


图9A

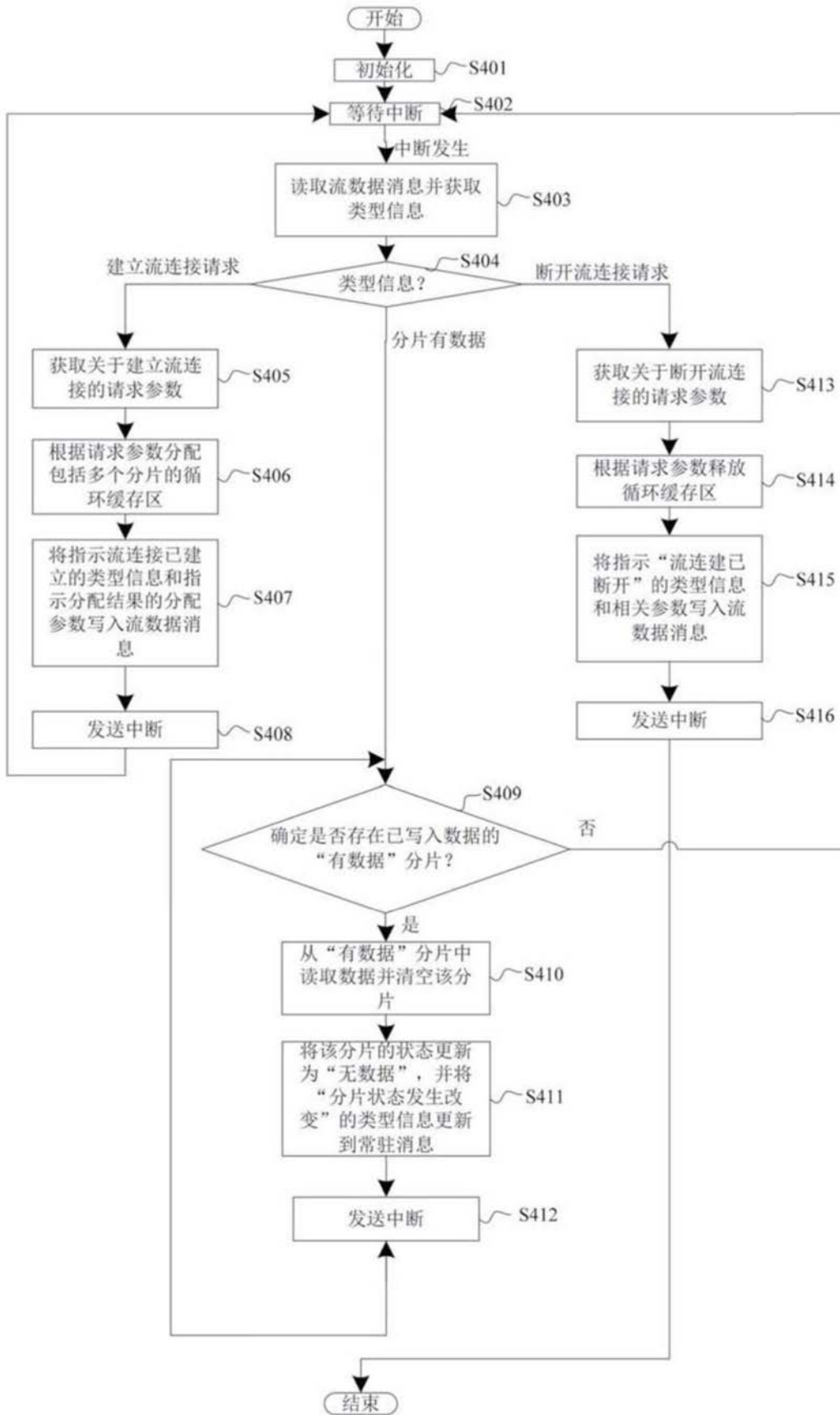


图9B