

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 15/16 (2006.01)

G06F 9/46 (2006.01)



[12] 发明专利说明书

专利号 ZL 00816272.7

[45] 授权公告日 2006 年 7 月 12 日

[11] 授权公告号 CN 1264102C

[22] 申请日 2000.11.24 [21] 申请号 00816272.7

[30] 优先权

[32] 1999.11.25 [33] NO [31] 19995774

[86] 国际申请 PCT/NO2000/000397 2000.11.24

[87] 国际公布 WO2001/038998 英 2001.5.31

[85] 进入国家阶段日期 2002.5.24

[71] 专利权人 艾利森电话股份有限公司

地址 瑞典斯德哥尔摩

[72] 发明人 P·约翰森 P·O·克罗卡

审查员 丁文勃

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 罗 朋 李亚非

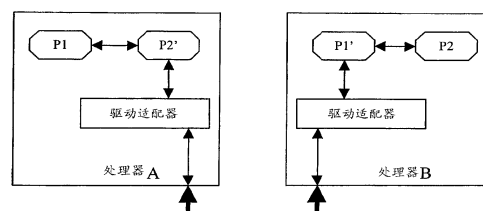
权利要求书 3 页 说明书 8 页 附图 13 页

[54] 发明名称

分布式处理系统中的进程间通信

[57] 摘要

在由两个或者多个互联的 CPUs 或者计算机机构成的分布式处理系统中，每个 CPUs 或者计算机具有一个操作系统，其中，在一个用于工作量分配的静态方式下，不同的进程用于特定的 CPUs，一个新的虚拟链接处理器 (VLH) 包括映象进程并且驱动适配器可以使进程在一个 CPU 或者计算机内驻留并运行，从而实现该进程与驻留并运行在其它 CPU 或者计算机内的进程之间的信号交换。



1、一种在含有两个或多个处理器或者计算机的分布式数
据处理系统中用于为处理器间或者计算机间提供进程信号交换的装
置，其中所述两个或多个处理器或者计算机各自包括相应的、独立的
5 操作系统，以及至少一个用于互连所述两个或多个处理器或者计算机
的互连装置，并且所述两个或多个处理器或者计算机适于各自的现行
进程驻留，并且至少一个相应的处理器间或计算机间的通信驱动器与
所述至少一个互连装置相关联，

其特征在於所述提供进程信号交换的装置包括一个虚拟链接处理器
10 装置，

所述虚拟链接处理器装置具有至少一个映象进程，该映象进程使每
个现行进程驻留在所述两个或多个处理器或者计算机的其中一个中，
并且要求可以与驻留在所述两个或多个处理器或者计算机的另一个中
的另一个现行进程进行通信，所述至少一个映象进程驻留在所述两个或
15 多个处理器或者计算机的所述一个中，以及包括：

多个驱动适配器，所述驱动适配器各自驻留在所述两个或多个处理
器或者计算机的相应一个中，多个所述驱动适配器中的每个驱动适配器
都适于与所述至少一个映象进程中共同驻留的一个进行通信，与所述通
信驱动器中共同驻留的一个通信驱动器进行通信，和与所述相应的、独
20 立的操作系统进行通信。

2、如权利要求 1 所述的装置，其特征在於，驻留在系统的所述两
个或多个处理器或者计算机中一个的一个映象进程被分配有一个进程
标识符来鉴别，该进程标识符与驻留在系统的所述两个或多个处理器
或者计算机的另一个中的对应的现行进程的进程标识符相同。

25 3、如权利要求 1 所述的装置，其特征在於，一个单独的映象进程
标识符被分别地分配到每个相应的映象进程中。

4、如权利要求 3 所述的装置，其特征在於，所述单独的映象进程
标识符本身在整个系统中是唯一的进程标识符。

5、如权利要求 3 所述的装置，其特征在於，所述映象进程标识符
30 与各自的处理器或者计算机地址相结合来构成在整个系统中唯一的进
程标识符。

6、如权利要求 1 或 2 所述的装置，其特征在於，每一个所述的现

行进程是与该系统的所述两个或多个处理器或者计算机中特定的一个静态相关的，或者静态地驻留在该系统的所述两个或多个处理器或者计算机的特定的一个中。

5 7、如权利要求 1 或 2 所述的装置，其特征在于，所述虚拟链接处理器装置包括所述现行进程与系统的所述两个或多个处理器或者计算机的相应的一些中的相关关系的参考目录。

8、如权利要求 7 所述的装置，其特征在于，位于系统的每个通信处理器或者计算机中的地址表由所述参考目录发展而来。

10 9、如权利要求 2 所述的装置，其特征在于，由一个处理器或者计算机中的一个进程传送到另一个处理器或者计算机中的另一个进程的信号包括一个临时修改，该临时修改至少包括接收映象进程的进程标识符。

15 10、如权利要求 9 所述的装置，其特征在于，由一个处理器或者计算机中的一个进程传送到另一个处理器或者计算机中的另一个进程的信号包括一个临时修改，该临时修改包括发送映象进程的进程标识符。

11、如权利要求 1 所述的装置，其特征在于，所述两个或多个处理器或者计算机的特定的一个所驻留的驱动适配器包括一个面对至少一个共同驻留的映象进程或操作系统的接口。

20 12、如权利要求 1 或者 11 所述的装置，其特征在于，所述两个或多个处理器或者计算机的特定的一个所驻留的驱动适配器包括一个面对与所述两个或多个处理器或者计算机的同一个特定处理器或者计算机相关的处理器或者计算机的互相连接的通信驱动器的接口。

25 13、一种在含有两个或多个处理器或者计算机的分布式数据处理系统中、用于为处理器间或者计算机间提供进程信号交换的方法，其中所述两个或多个处理器或者计算机各自包括独立的操作系统，至少一个处理器间或者计算机间通信驱动器，以及至少一个处理器间或者计算机间通信驱动器相关的互联装置，用于互联所述两个或多个处理器或者计算机，其特征在于，包括以下步骤：

30 -在一个处理器或者计算机上将意欲用于另一个处理器或者计算机中的另一个现行进程的预定信号，从一个初始现行进程发送到表示另一个现行进程的一个共同驻留的第一映象进程中，

- 通过第一映像进程和包含有所述另一个现行进程的系统地址的表来修正信号参数以产生修正信号，以指示作为该信号的下一个接收者的驻留在该另一个处理器或计算机中的一个第二映像进程，

5 - 通过驱动适配器、所述至少一个处理器间或者计算机间通信驱动器、以及所述至少一个处理器间或者计算机间通信驱动器相关的互联装置，将修正信号发送到驻留了第二映像进程的另一个处理器或者计算机，

 - 在另一个处理器或者计算机中通过一个代表信号初始进程的第二映像进程来接收修正信号，

10 - 利用由所接收信号所载的参数通过接收第二映像进程来修正所接收信号的参数而产生修正的接收信号，以指示作为信号的下一个接收者的驻留在该另一个处理器或计算机中的该另一个现行进程，并且从该第二映像进程将所述修正的接收信号传送到所要信号的该另一个现行进程中。

15

分布式处理系统中的进程间通信

发明领域

- 5 本发明涉及计算机系统，该系统包括两个或者多个处理单元（CPU），尤其是该系统中的进程间进程间通信，其中通过静态方式向可以独立操作的不同的 CPUs 分配不同的进程以及该系统中用于这样的进程间进程间通信的不包括公知的传统网络性能的操作系统（OS）来实现工作量共享。

10

背景技术

- 在计算机系统中，需要满足效率和速度的需求，由于一个标准计算机系统的单一 CPU 每次仅仅能够实现一条信号指令。对于这个问题，通常的解决方法是：采用包括两个或者多个现行的 CPUs（或者计算机）增加计算机系统的容量。对于不止一个 CPU 的上述系统，多项任务，典型的是，作为进程给每个 CPU 分配唯一的符号处理标识符（PID），该标识符包括由 CPUs 操作的计算机进程，可以被分布在 CPUs 中用于共享工作量并且允许在同一时刻可以执行不止一个指令。典型地，在多数系统中，为了不同的目的，例如信息交换或者信号接收和发送，需要进行通信，并且当不同的进程被存储在不同的 CPUs 时，这样的系统还必须包括一些装置，通过这些装置，不同的进程之间可以相互交流而不需要限制进程间进程间通信就可以处理同一 CPU 的分配。可以发现，在这样的系统中 OS 或者任何中断例程也可以发送信号，但是只有进程可以接收信号。当这些系统的 CPUs 或者计算机独立操作并且它们的 OS 不包括用于在不同的 CPUs 或者计算机之间用于系统内“透明”进程间进程间信号交换的网络能力，需要提供用于在不同的 CPUs 或者计算机进程之间完成进程间进程间通信的其它方法。

- 30 为了给进程间进程间通信提供方便，在上述提到类型的多处理器或者多个计算机系统中提到过的解决方法是公知的。其中一些解决方法建议使用普通存储器进行这样的通信，而其它的方法则建议在该系统的处理器之间附加其它类型的连接。

美国专利 US5062040 揭露了一种用于确保信号不丢失而不对复制信号进行处理的多处理装置，并且多处理系统被“透明”的用于现有进程中。一个信号处理码片(signal-handling code fragment)总是在本地处理器中运行从而扩展处理的用户部分。用户核心进程必须发送当前信号结果（也就是，一个信号是否已经被接收到了，而没有被忽略）到一个含有占位包的占位进程，以及任意修改信号结果必须被发送到一个有效的占位进程中，如果任意非忽略信号在占位进程响应之前随后被用户进程接收。则种情况被定义为伪信号状态。每个核心进程用于维护一个表格链接目录，该表格包括前向发送信号，并且一个系统进程运行在基本水平，可以从列表中删除信号并且通过连接到每个处理器的通信信道前向发送到正确的目的地。

发明内容

因此，本发明的一个目的是通过一个分布式处理的多处理系统提供一种用于进程间通信的方案，该方案可以维护处理器和 OS 的独立并且其中 OS 一般不需要控制元件或者信息。

本发明的上述目的是通过提供一个新的虚拟链接处理器（VLH）实现的，该处理器包括用于系统内的现行处理通信的映象进程（shadow process,SP）以及用于 CPUs 之间通信的驱动适配器（DA）。在本发明的一个方案中，因为需要与其它系统的 CPUs 或者计算机驻留进程进行通信，CPUs 或者计算机的现行驻留处理的符号 PIDs 也被定义在其它的通信 CPUs 或者计算机中。因此，一个 CPU 或者计算机的驻留 SP 的符号 PID 与别的 CPU 或者计算机的驻留 SP 的符号 PID 相同。从而，一个特定 CPU 的驻留进程将被保存在 CPU 中，其中该进程驻留查找一个与“远程”目标进程的符号 PID 具有相同符号 PID 的“本地”SP，该驻留进程由它的 PID 所定义并且可以传送一个信号到具有另一个 PID 的“远程”CPU 的“远程”驻留目标进程。为了信号交换，本地 SP 到一个本地通信现行进程的接口与连接现行“远程”目标进程的接口相同。本地“SP”，然而，实际并没有对该信号起作用，但是却服务于到另一个 CPU 前向接收信号的操作，“远程”目标进程被分配给该 CPU 并且对它进行操作。在接收端 CPU，通过一个类似的 SP 接收

信号消息，该接收方法轮流将信号转发到现行目标进程。因此，本发明所述系统的 CPUs 或者计算机之间的信号传送无需具有任何关于 VLH 存在信息的现行进程就很容易通过新的 VLH 装置实现。从上述内容可以直接看出，该 SP 可以用于从一个进程接收一个信号并将该信号转发到另一个 CPU 或者计算机，也可以向一个进程发送一个信号并且该信号曾经从另一个 CPU 或者计算机转发。为了在 CPUs 之间实现信号传送，每个 CPU 的映象进程都具有到驱动适配器的接口，该驱动适配器轮流具有到驱动器的接口用于发送装置，该发送装置用于提供从一个 CPU 或者计算机到另一个之间传送信息的系统信道。

- 10 根据本发明的一个方面，一种在含有两个或多个处理器或者计算机的分布式数据处理系统中用于为处理器间或者计算机间提供进程信号交换的装置，其中所述两个或多个处理器或者计算机包括一个独立的操作系统，至少一个内部处理器或者计算机间通信驱动器以及至少一个在所述两个或多个处理器或者计算机之间的相关的互相连接驱动器，其特征在一个虚拟连接处理器装置包括：

至少一个用于每个现行进程的映象进程，该映象进程要求可以与驻留在另一个处理器或者计算机中的另一个现行进程进行通信，和

- 多个驱动适配器，每个都适于与所述至少一个映象进程中至少一个共同驻留的映象进程和所述通信驱动器中至少一个共同驻留的通信驱动器 20 和所述操作系统进行通信。

- 根据本发明的另一个方面，一种在含有两个或多个处理器或者计算机的分布式数据处理系统中，用于为处理器间或者计算机间提供进程信号交换的方法，其中所述两个或多个处理器或者计算机包括独立的操作系统，至少一个处理器间或者计算机间通信驱动器以及至少一个 25 在所述两个或多个处理器或者计算机之间的相关的互连驱动器，其特征在于，包括以下步骤：

- 在一个处理器或者计算机上将意欲用于另一个处理器或者计算机中的另一个现行进程的预定信号，从一个初始现行进程发送到表示另一个现行进程的一个共同驻留的第一映象进程中，
- 30 - 通过映象进程和包含有驻留在所述另一个现行进程的系统地址的表来修正信号参数，以指示作为该信号的下一个接收者的驻留在该另一个处理器或计算机中的一个第二映象进程，

- 通过驱动适配器, 所述至少一个驱动器以及所述至少一个与互连相关的驱动器来将修正信号发送到驻留了第二映象进程的另一个处理器或者计算机,

5 - 在另一个处理器或者计算机中通过一个代表信号初始进程的第二映象进程来接收修正信号,

- 利用由所接收信号所载的参数通过接收第二映象进程来修正所接收信号参数, 以指示作为信号的下一个接收者的驻留在该另一个处理器或计算机中的该另一个现行进程, 并且

10 - 从该第二映象进程将所述修正的接收信号传送到所要信号的该另一个现行进程中。

附图说明

图 1 是一个包括两个 CPUs 或者计算机的系统的问题概况的示意图, 每个 CPUs 或者计算机包括一个需要相互通信的不同驻留进程。

15 图 2 是本发明用于解决图 1 所示问题的一个 VLH 解决方案范例示意图。

图 3 是本发明的 DA 进程的一个实施例的示意图。

图 4 是本发明的驱动适配器的一个过程的实施例的示意图。

图 5 是本发明的驱动适配器另一个过程的实施例的示意图。

20 图 6 是本发明的映象进程的是实例的示意图。

图 7 是本发明含有 VLH 解决方案的系统范例的示意图。

图 8 表示图 7 中所示系统范例的现行进程的 PIDs 之间关系, 各个处理器的 PIDs 映象进程以及它们各自的位置。

图 9a-c 表示图 7 中所示系统范例的查询表格的发展。

25 图 10 表示在图 7 所示系统的一个方向传送一个信号。

图 11 表示在图 7 所示系统的另一个方向传送一个信号。

具体实施方式

30 如下所述, 通过实施例以及相关的附图, 将对本发明作更详细的介绍。

图 1 表示一个具有两个处理器并且进程 P1 和 P2 在处理器 A 和 B 中驻留和运行的系统的典型问题情况, 分别地, 其中进程之间希望可以

进行通信。每个处理器的操作系统（OS）可以对同一个处理器内驻留和运行的进程进行相互通信，但是不能给驻留和运行在不同处理器中的进程之间提供通信。因此，图 1 所示的典型情况，虽然处理器通过物理连接进行信号传送，但是进程 P1 和 P2 却不能相互通信。

- 5 图 2 表示一个类似图 1 所示情况的示意图，但是现在提供了两个新的 P1'和 P2'，本发明的驱动适配器，可以使进程之间进行信号传递。在本发明系统中已经实现并且其中利用了处理器的不同类型，系统设计者必须考虑多种处理器的格式，例如字节排序和填充。

参见图 3，本发明所示的一个 DA 进程实现范例的符号表达如图所
10 示。在开始并且初始化之后，进程位于空闲等待（idle）状态直到出现一个信号事件。在信号事件时刻，如果信号来自一个驱动器，那么进程通过下述分支标记（1）表示，或者，如果信号将要被发送到一个驱动器，那么进程通过分支标记（2）表示。

参见图 4，本发明所示的一个 DA 实现的过程范例的符号表达如图
15 所示。该过程用于发送一个信号到 DA。

参见图 5，本发明所示的一个 DA 实现的过程范例的符号表达如图
所示。该过程用于根据信号的初始进程从 DA 发送一个信号到 SP。该过程的映象进程 PID（SPID）取决于参数发送端并且利用 GetPID（）函数转换到本地 PID。SHADOW_SIGNAL 可以是唯一的信号 ID，仅
20 仅用于从 DA 发送到任意 SP。

参见图 6，本发明所示的一个（一般）SP 执行中的符号表达如图所
示。在开始并且初始化之后，进程位于空闲等待状态直到出现一个信号事件。在信号事件时刻，如果信号来自一个 DA，进程过程通过下述
分支标记（1）表示，对于所有其它信号，进程则通过下述分支标记（2）
25 表示：

（1）当一个映象进程接收到一个 SHADOW_SIGNAL，从接收到的信号中提取信号 ID 和初始信号。然后从信号中提取预定的接收 SPID 并且利用 GetPID（）函数转换到本地 PID，并且信号被发送到预定的进程。随后，SP 恢复到 IDLE 状态；

30 （2）一旦任意一个其它的信号被接收，该信号就被修改：SP 将接收端的 SPID 和发送端的 SPID 加到信号上。信号接收端的处理器地址决定信号发送到哪一个处理器上。然后信号被发送到驱动适配器。SP

恢复到 IDLE 状态。

对于上述两段所描述的进程，下面适用于：GetPID（）函数在本地处理器上查找一个基于 SPID 的 PID。GetSPID（）函数在本地处理器上查找一个基于 PID 的 SPID。GetAddress（）函数查找一个基于 SPID 的 PID。处理器地址用于识别不同的处理器。

为了避免模糊或者不确定，在本发明的最佳实施例中系统中仅仅有唯一的一个 SPID。另一个方案是使用含有 CPU 或者计算机的地址的 SPID 作为进程唯一的标识符。Receiver（）函数以及 Sender（）函数是 OS 函数并且各自返回接收端和发送端的 PID。

另一个系统范例如图 7 所示。假定系统需要已经被满足，那么所需要的范例的参数就不再通过一个含有可用信号处理器的系统来实现了。因此，为了使系统具有合适的可用空间以及处理功率，在所示的系统范例中总共含有 7 个现行进程，现行进程已经被分为两组（每一组用于一个处理器）。一组包括进程 P1，P2 和 P3，另一组包括进程 P4，P5，P6 和 P7，这两个组分别在处理器 A 和 B 中驻留和运行。

现在两个组被分为两个不同的处理器，这就需要对在进程之间曾经有的通信连接重新建立通信。如图 8 所示，这已经通过在每个处理器中引入和放置“映象进程”实现了，每个“映象进程”用于遗失的“现行”进程。映象进程包括新的 VLH，该 VLH 使得系统中的进程信号交换变得更容易。如范例所示，需要在第一组的进程 P1，P2，P3 以及第二组的 P4，P5，P6 以及第二组的 P7 之间进行通信，然而不需要第一组的任何进程之间的信号交换。因此不必对 P7 采用映象进程。

在引入映象进程之后，在处理器内就建立了响应系统。通过这样的安排，第一组的现行进程可以查询一个地址用于通过响应于第二组现行进程的映象进程进行信号交换，反之亦然。

为了使不同 CPUs 之间的信息能够交流，两个处理器之间还需要有通信装置。这样的通信装置应该包括驱动器以及相关的物理连接。为了使映象进程可以使用这样的通信装置，还要有驱动适配器。这些设备如图 7 所示。

最后，还需要一个用于识别在处理器之间的进程的方法。在所示系统中，假定处理器不具备常规存储器。根据本发明，识别处理器的方法参见图 8 所示的表格范例，对应于图 7 的系统范例。

参见图 8，可见范例中的进程 P7 在处理器 A 中并不包括映象进程 PID 或者相应的 PID，而处理器 P7 也不需要与任何驻留在处理器 A 中进程进行信号交换。

在图 9a-9c 中揭示了表格的发展过程。假定系统不具有通常的存储器，并且图 8 的参考表格被分为两张表格，如图 9 所示，并且这两张表格分别放置在相应的处理器中。从图 9b 的表格中可以看到：一旦在处理器 A 的进程和进程 P7 之间不再通信，就从表格中删除进程 P7。现在我们具有一个参考表格，该表格可以指明信号来自哪个进程以及将被发送到哪个进程。

图 7 所示的系统范例是一个仅仅具有两个 CPUs 或者计算机的系统。因此，在本发明仅仅具有两个 CPUs 或者计算机的系统中，地址域几乎可以忽略不计，而上述表格可以被进一步分为如图 9c 的范例中所示的表格。这是因为一个从现行进程发送到 SP 的信号应该总是被发送到另一个“远程”CPU 或者计算机，并且一个从 DA 发送到 SP 的信号应该被发送到一个现行“本地”进程。

参见图 10，从处理器 A 的一个进程到处理器 B 的一个进程进行信号交换的范例如图所示。假定有一信号，S1，从现行进程 P1 发送到现行进程 P5。由于映象进程 P5' 与现行进程 P5 具有相同的符号 PID，信号的第一接收者将成为响应 P5，即 P5'，的映象进程。P5' 将查询发送者的本地 PID，如果有必要，还查询它自己的 PID。映象进程利用这些数据查询相应处理器的独立的 PID，该 PID 被称作映象进程 PID。所有这些数据都通过较低电平（level）发送，例如驱动适配器以及驱动器。在驱动级，含有一个 ID 的数据被发送到另一个处理器，从而通知另一个处理器中的驱动器这是一个“映象进程”信号。为了数据传输，可以采用重新传输以及 / 或者帧编码方法，这取决于本发明的解决方案。当信号到达另一个处理器的驱动适配器时，在表格中执行一个查询，查找响应于信号发送者的那个映象进程。信号 S3（仍然带有发送者和接收者信息）接着被发送到该进程。该进程处于情况 P1'。当信号到达 P1' 时，发送者和接收者的信息被删除，而初始信号被发送到预定的接收进程 P5。

上文已经揭示了如何从 P1 向 P5 发送一个信号。如图 11 所示，范例的进程 P5 可以响应进程 P1 的信号，或者可以从 P5 向 P1 发送一个

新的信号。上述两种情况的过程都类似于上文所述的过程，仅仅是信号流向相反。

可以发现信号 S2 和 S5 也可以是一个产生 DA 的函数调用，但是也可看作一个表示信息流的信号。

- 5 还可以发现，在本发明的解决方案中，上述的信号发送响应并不是强制性的。前面（P1 到 P5 以及 P5 到 P1）描述的两个信号顺序因此可以看作两个独立的信号顺序。

优点

- 10 进程所采用的信号（有时候也称作消息）是为了向其它的进程提供一些情况。OS 或者任何中断例程（没有阐释）也可能发送信号，但是只有进程可以接收信号。这也可以在微处理器中完成，分配系统中的每个处理器的 OS 并不提供不同进程之间的通信，以至于被驻留在静态方式下并且运行于不同的处理器之间。

- 15 通过在静态方式下给不同的 CPUs 分配不同的进程来实现工作量分配，其中上述 CPUs 相互独立运行并且其操作系统（OS）不包括传统的网络功能以及相关的系统开销。

本发明的 VLH 既不需要修改为专用操作系统的处理器也不需要增加系统的开销和复杂程度就可以对驻留和运行在不同处理器之间的进程进行信号交换。

- 20 并且，利用识别映象进程，本发明可以维持系统的简单性。映象进程看似相同，然而却不同。

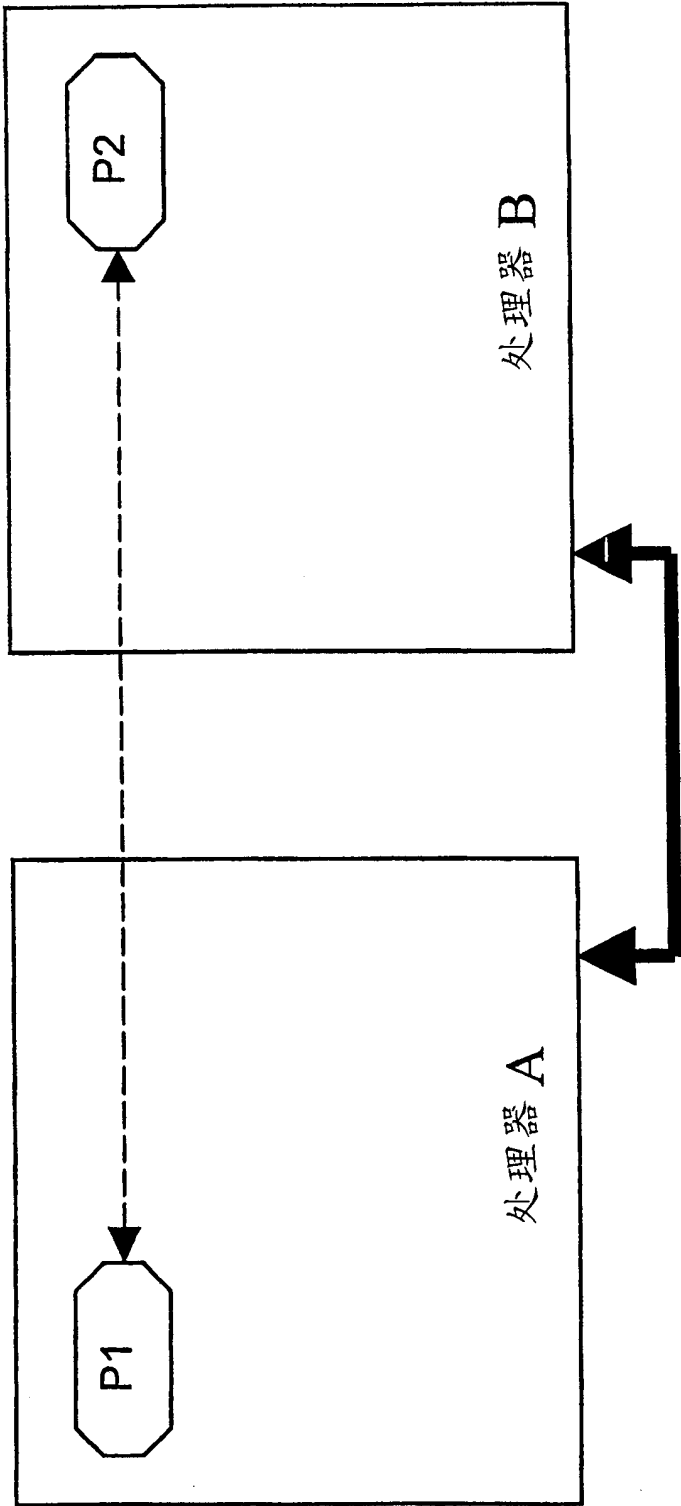


图 1

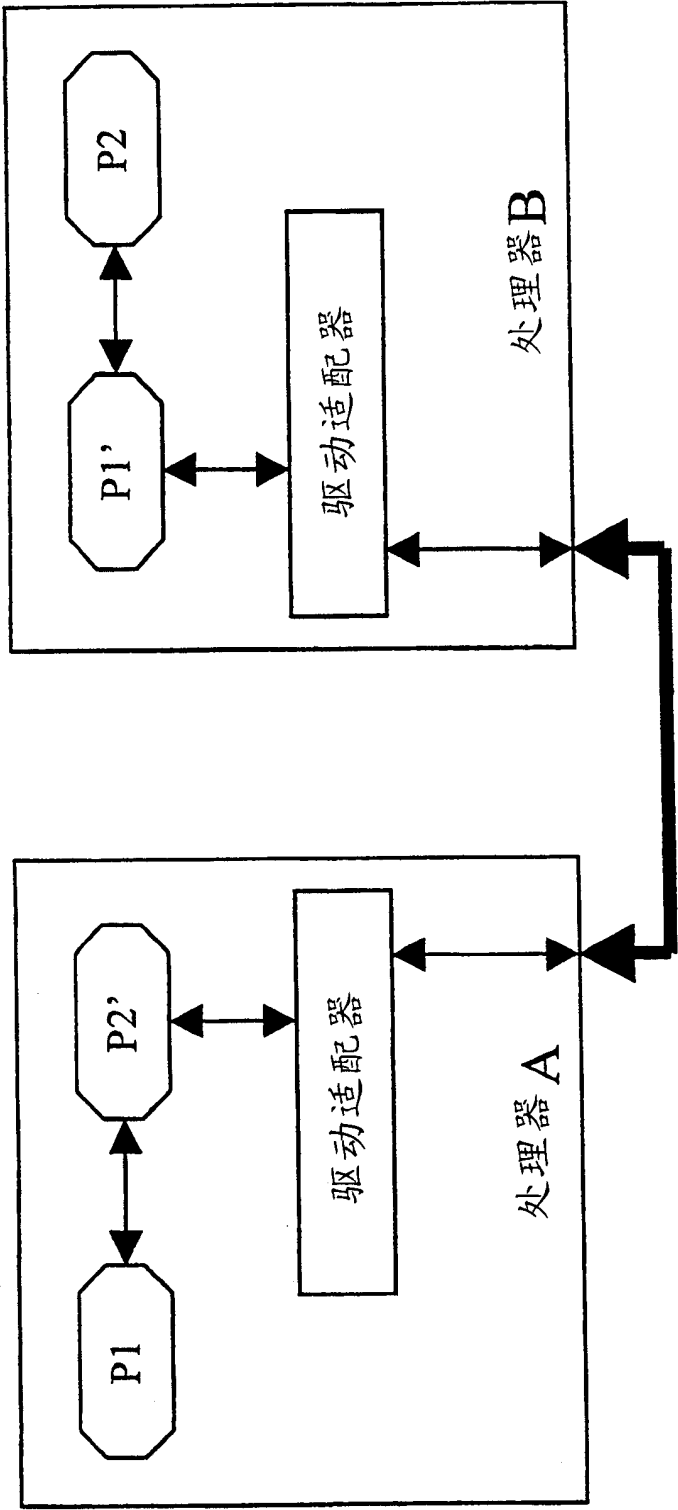


图 2

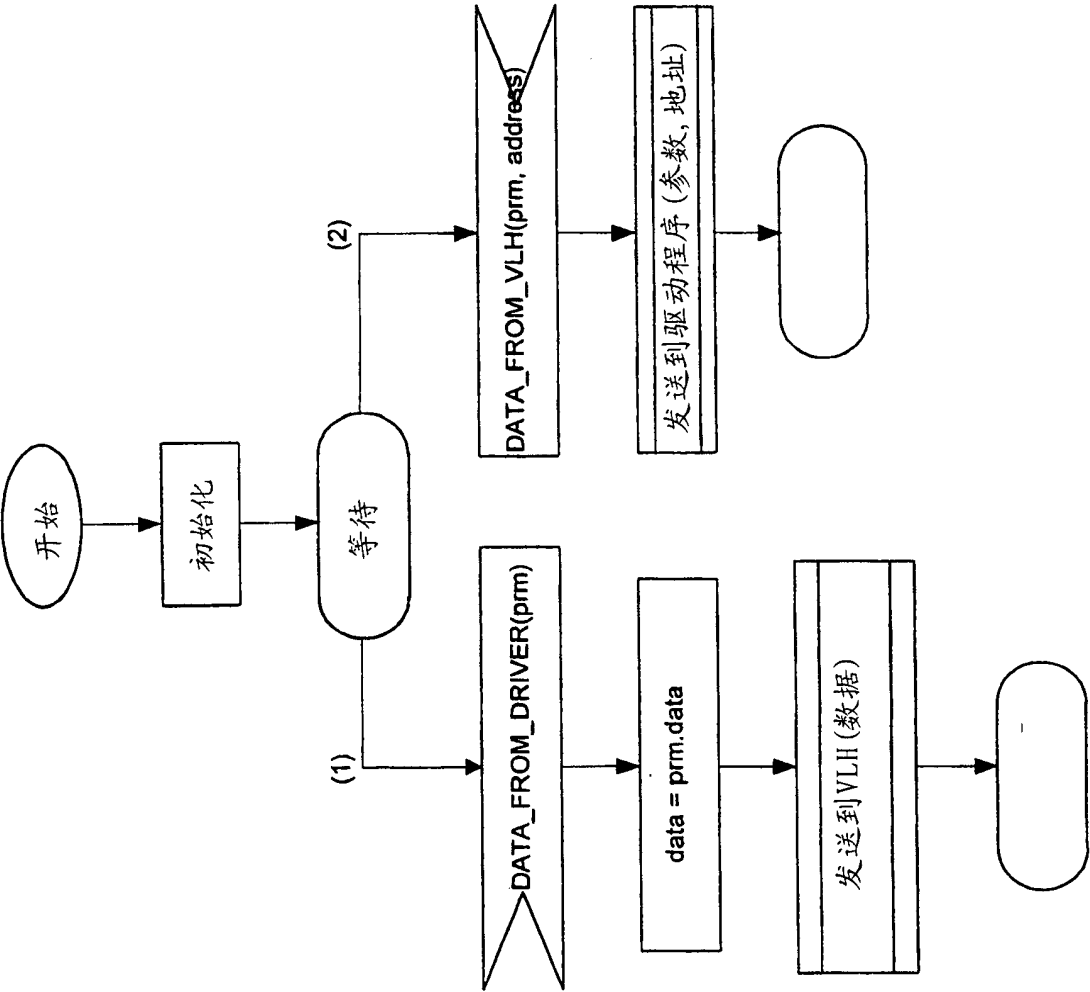


图 3

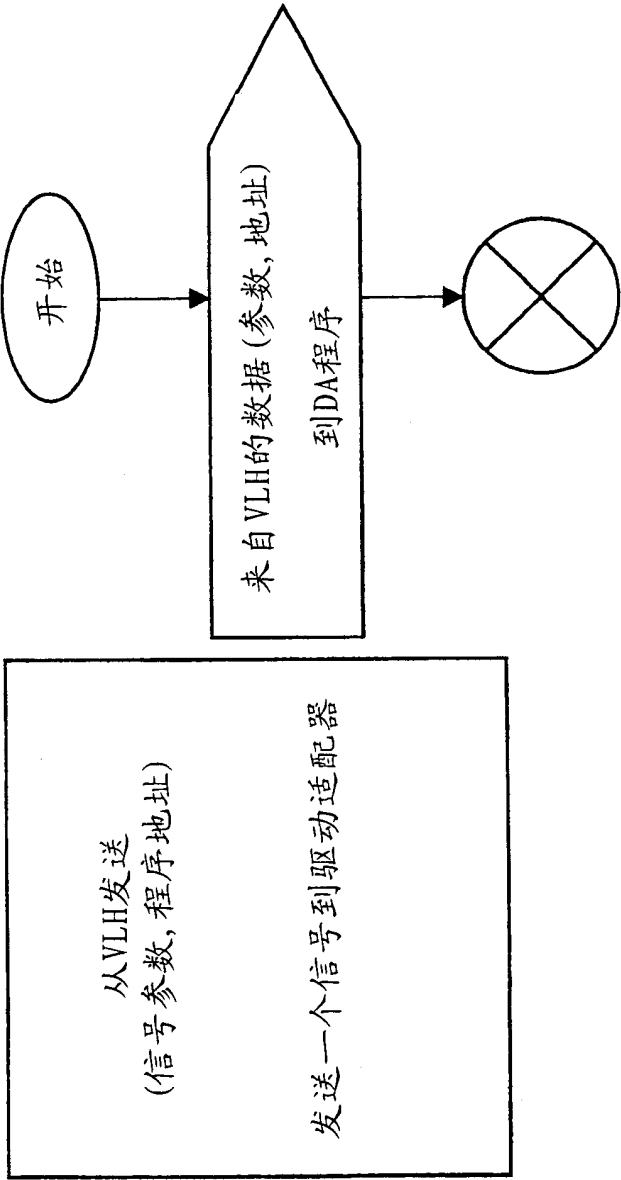


图 4

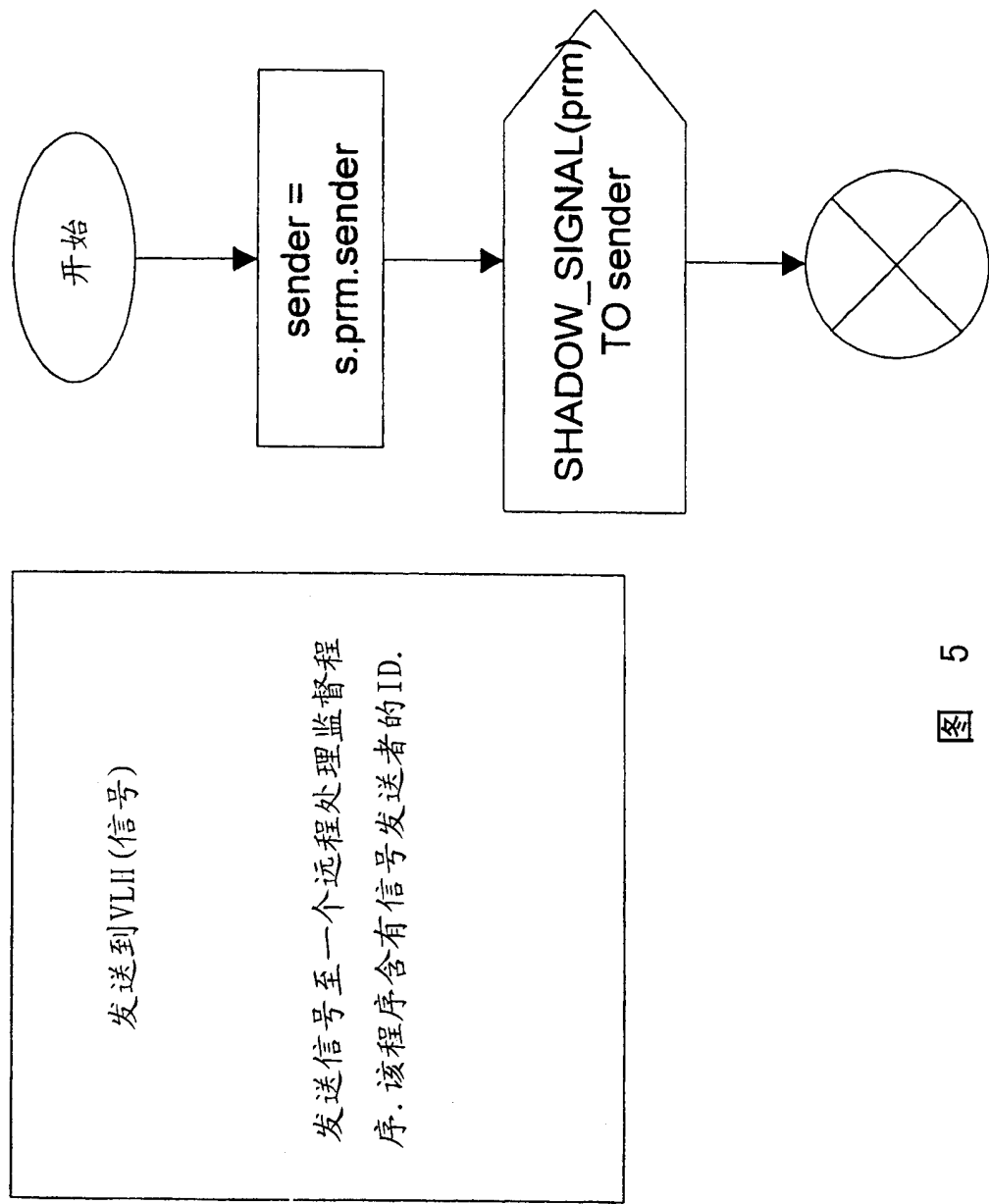


图 5

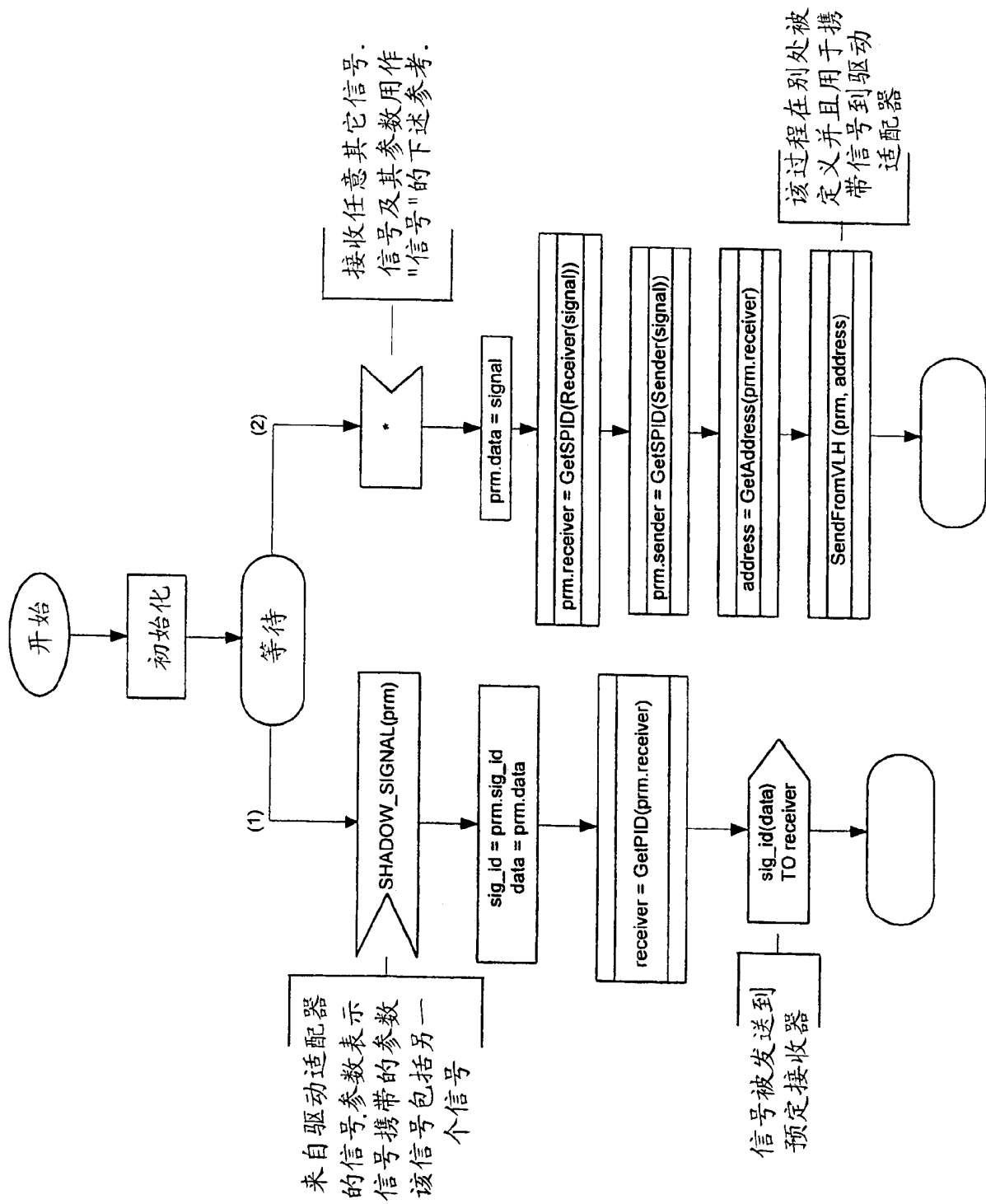


图 6

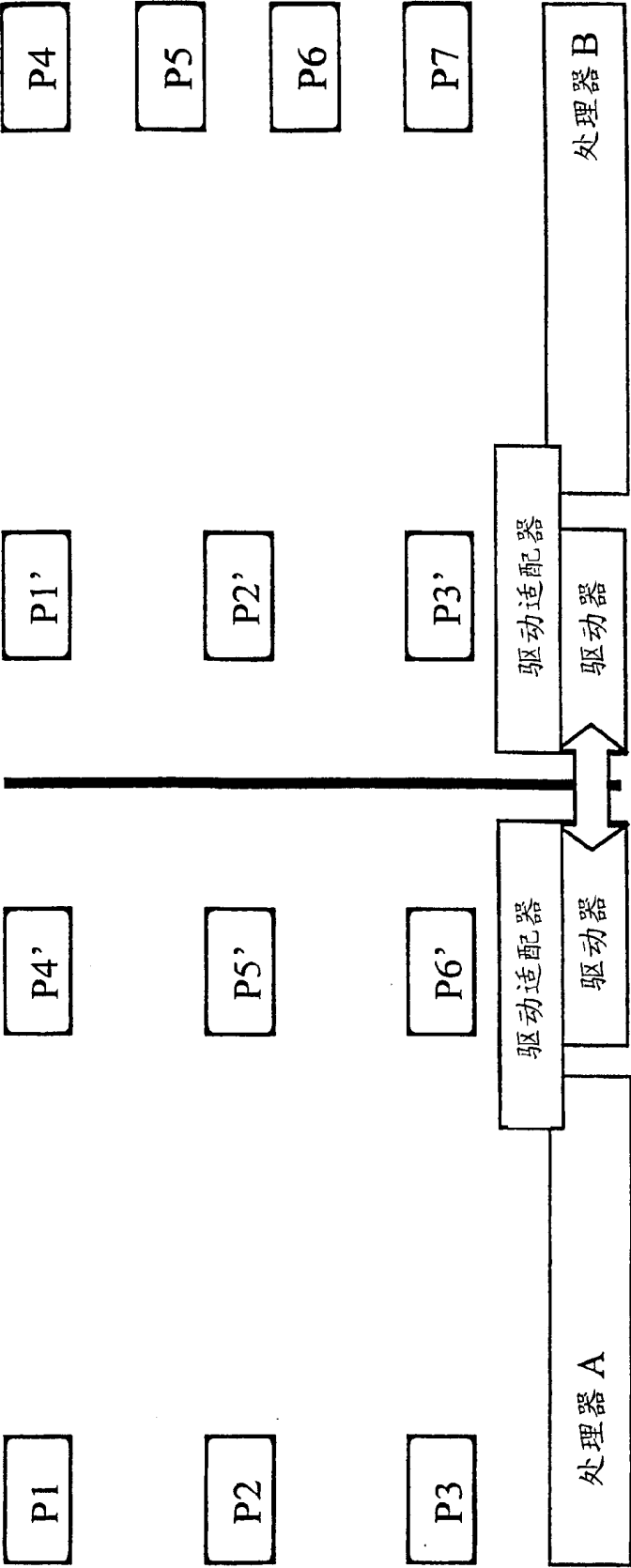


图 7

处理器 A PIDs	阴影 PIDs	地址	处理器 B PIDs
P1	sP1	A	P1'
P2	sP2	A	P2'
P3	sP3	A	P3'
P4'	sP4	B	P4
P5'	sP5	B	P5
P6'	sP6	B	P6
-	-	B	P7

图 8

处理器 A		
PIDs	阴影 PIDs	地址
P1	sP1	A
P2	sP2	A
P3	sP3	A
P4'	sP4	B
P5'	sP5	B
P6'	sP6	B
-	-	-

处理器 B		
地址	阴影 PIDs	PIDs
A	sP1	P1'
A	sP2	P2'
A	sP3	P3'
B	sP4	P4
B	sP5	P5
B	sP6	P6
B	-	P7

图 9a

处理器 A		
PIDs	阴影 PIDs	地址
P1	sP1	A
P2	sP2	A
P3	sP3	A
P4'	sP4	B
P5'	sP5	B
P6'	sP6	B

处理器B		
地址	阴影 PIDs	PIDs
A	sP1	P1'
A	sP2	P2'
A	sP3	P3'
B	sP4	P4
B	sP5	P5
B	sP6	P6

图 9b

处理器 A		处理器 B	
PIDs	阴影 PIDs	阴影 PIDs	PIDs
P1	sP1	sP1	P1'
P2	sP2	sP2	P2'
P3	sP3	sP3	P3'
P4'	sP4	sP4	P4
P5'	sP5	sP5	P5
P6'	sP6	sP6	P6

图 9c

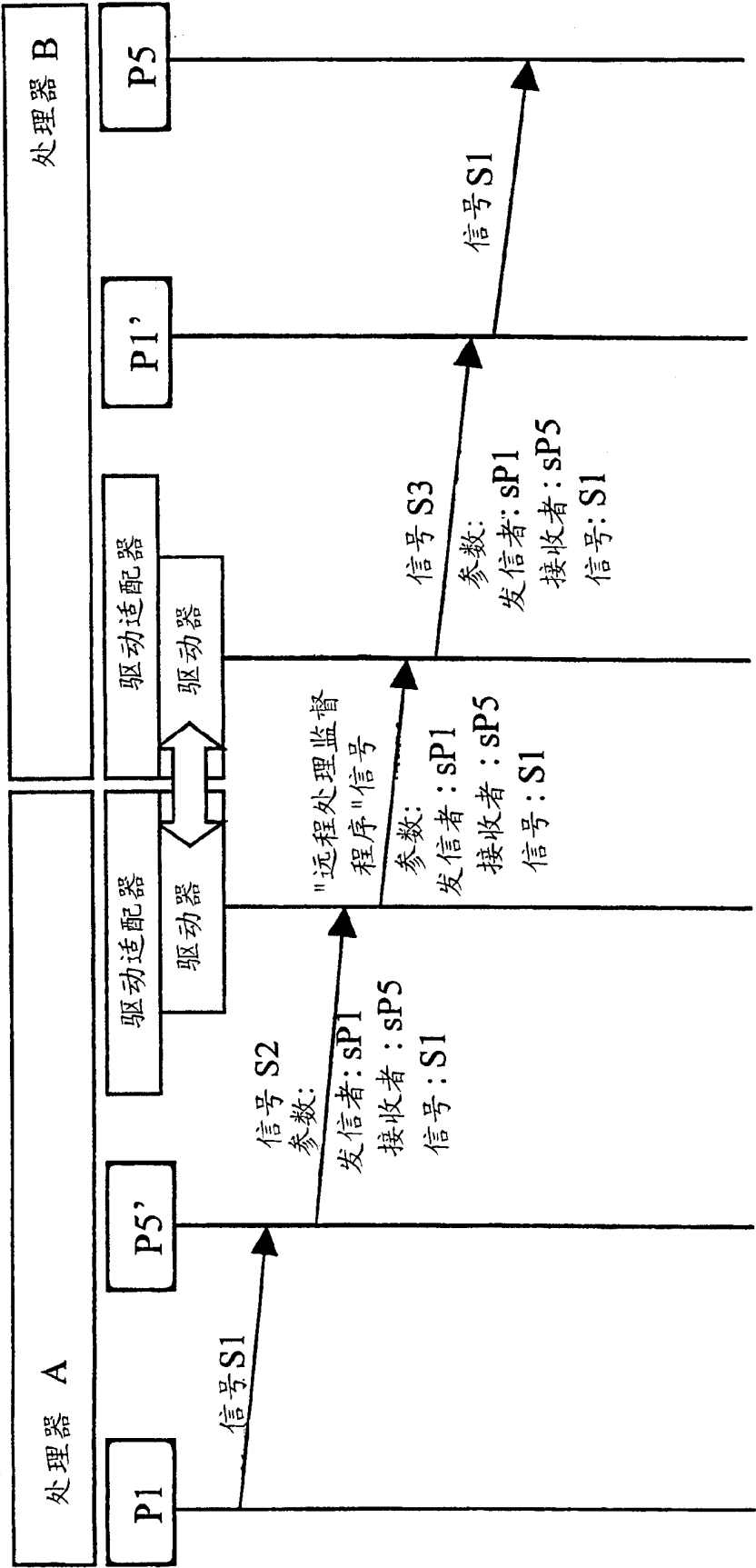


图 10

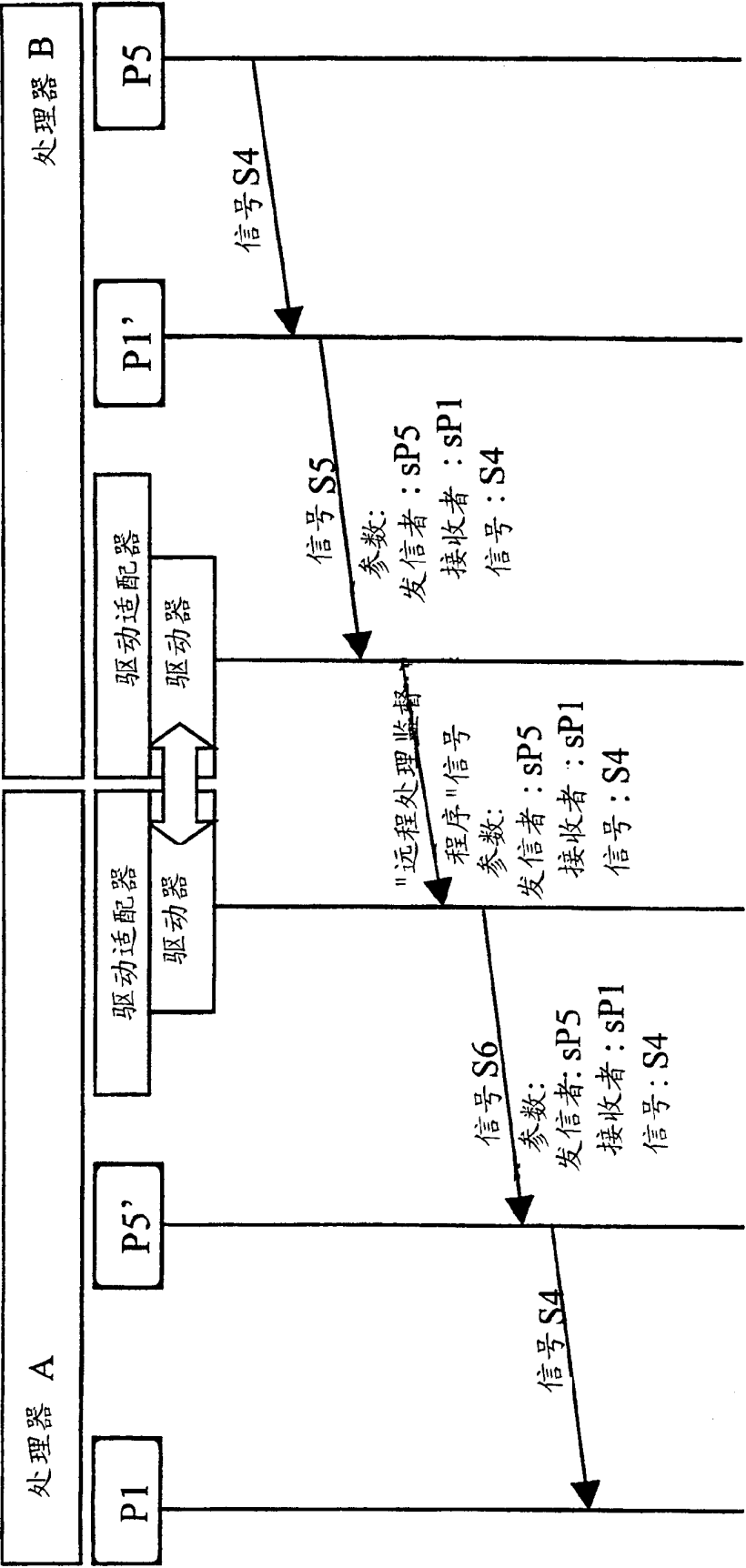


图 11