



(12) 发明专利

(10) 授权公告号 CN 101038567 B

(45) 授权公告日 2012. 05. 23

(21) 申请号 200710088308. 6

(22) 申请日 2007. 03. 15

(30) 优先权数据

11/377, 506 2006. 03. 16 US

(73) 专利权人 国际商业机器公司

地址 美国纽约阿芒克

(72) 发明人 C·R·约翰斯

(74) 专利代理机构 北京市金杜律师事务所

11256

代理人 朱海波

(51) Int. Cl.

G06F 12/08 (2006. 01)

G06F 13/28 (2006. 01)

(56) 对比文件

同上.

US 5611074 A, 1997. 03. 11, 第 2 栏第 62

行 - 第 3 栏第 19 行、权利要求 1-2、图 1, 3.

Maged M. Michael, Michael L. Scott.

Implementation of Atomic Primitives on Distributed Shared Memory

Multiprocessors. High-Performance Computer

Architecture, 1995. Proceedings. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=386540&isnumber=8763?tag=1>. 1995, 223-224.

审查员 尹春梅

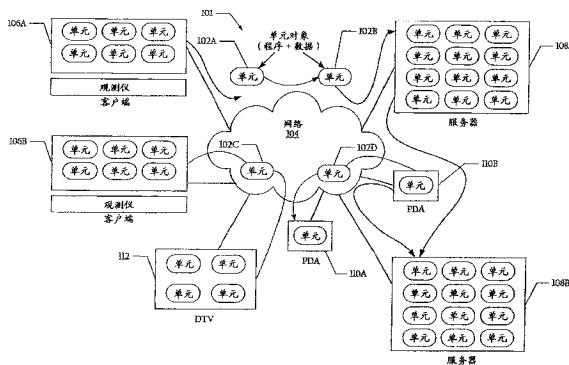
权利要求书 2 页 说明书 10 页 附图 6 页

(54) 发明名称

执行高速缓存线轮询操作的方法、系统和设备

(57) 摘要

在此公开了一种利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询的方法、系统、设备和产品。在一个实施例中,提供了一种方法,其包括在第一可高速缓冲存储区域中存储一个缓冲器标记占用指示符数据值,并且经由存储和预留指令在所述第一可高速缓冲存储区域上设置加载/存储操作预留。在所述实施例中,响应于已经在第一可高速缓冲存储区域上复位加载/存储操作预留的判定,经由条件加载指令存取存储在所述第一可高速缓冲存储区域中的数据值。相反地,响应于没有在第一可高速缓冲存储区域上复位加载/存储操作预留的判定,停止执行条件加载指令。



1. 一种在信息处理系统中执行高速缓存线轮询操作的方法,所述方法包括:

由处理器在数据缓冲器中存储数据,使得所述数据缓冲器被布置在包括多个可高速缓冲存储区域的存储器中,使得所述存储器数据缓冲器能够被外部设备存取以便所述外部设备从所述数据缓冲器取回所述数据,并且使得所述多个可高速缓冲存储区域能够被所述外部设备存取以更新其中存储的状态数据;

由所述处理器在所述存储器内的第一可高速缓冲存储区域中存储缓冲器标记占用指示符状态数据值,并由所述处理器通过执行存储和预留指令 STAR 设置与所述第一可高速缓冲存储区域相关联的加载/存储操作预留,使得由所述处理器通过执行所述 STAR 指令在所述第一可高速缓冲存储区域中存储所述缓冲器标记占用指示符状态数据值,以便通知所述外部设备能够存取所述存储器数据缓冲器,其中所述数据已经被存储在所述存储器数据缓冲器中并且能够被所述外部设备存取以取回所述数据;

由所述处理器执行预留丢失时加载指令 LDRL 以判定所述外部设备是否已经复位所述加载/存储操作预留,其中所述复位对应于所述外部设备对存储在所述第一可高速缓冲存储区域中的所述状态数据进行修改以表明在所述外部设备已经完成从所述存储器数据缓冲器取回所述数据之后的“未占用”状态指示符;

响应于所述外部设备对所述第一可高速缓冲存储区域的存取,将存储于所述第一可高速缓冲存储区域中的当前状态数据值与所述缓冲器标记占用指示符状态数据值进行比较;

停止执行所述预留丢失时加载指令 LDRL 直到所述处理器判定所述外部设备已经复位与所述第一可高速缓冲存储区域相关联的所述加载/存储操作预留,其中所述外部设备是通过如下方式来进行所述复位的,即所述外部设备在所述第一可高速缓冲存储区域中存储“未占用”指示符状态数据值以表明所述外部设备已经完成从所述存储器数据缓冲器取回所述数据;以及

响应于所述外部设备在所述第一可高速缓冲存储区域中存储所述“未占用”指示符状态数据值,由所述处理器在所述存储器数据缓冲器中存储附加数据以便由所述外部设备取回。

2. 根据权利要求 1 所述的方法,所述方法还包括:

检测进程上下文切换;以及

响应于检测到所述进程上下文切换,复位与所述第一可高速缓冲存储区域相关联的所述加载/存储操作预留。

3. 根据权利要求 1 所述的方法,其中经由所述存储和预留指令 STAR 设置与所述第一可高速缓冲存储区域相关联的所述加载/存储操作预留包括:

复位与所述存储器内的第二可高速缓冲存储区域相关联的先前的加载/存储操作预留;以及

响应于已经复位与所述第二可高速缓冲存储区域相关联的所述先前的加载/存储操作预留的判定,设置与所述第一可高速缓冲存储区域相关联的所述加载/存储操作预留。

4. 一种用于在信息处理系统中执行高速缓存线轮询操作的设备,所述设备包括:

用于由处理器在数据缓冲器中存储数据、使得所述数据缓冲器被布置在包括多个可高速缓冲存储区域的存储器中、使得所述存储器数据缓冲器能够被外部设备存取以便所述外

部设备从所述数据缓冲器取回所述数据、并且使得所述多个可高速缓冲存储区域能够被所述外部设备存取以更新其中存储的状态数据的装置；

用于由所述处理器在所述存储器内的第一可高速缓冲存储区域中存储缓冲器标记占用指示符状态数据值、并由所述处理器通过执行存储和预留指令 STAR 设置与所述第一可高速缓冲存储区域相关联的加载 / 存储操作预留、使得由所述处理器通过执行所述 STAR 指令在所述第一可高速缓冲存储区域中存储所述缓冲器标记占用指示符状态数据值的装置，以便通知所述外部设备能够存取所述存储器数据缓冲器，其中所述数据已经被存储在所述存储器数据缓冲器中并且能够被所述外部设备存取以取回所述数据；

用于由所述处理器执行预留丢失时加载指令 LDRL 以判定所述外部设备是否已经复位所述加载 / 存储操作预留的装置，其中所述复位对应于所述外部设备对存储在所述第一可高速缓冲存储区域中的所述状态数据进行修改以表明在所述外部设备已经完成从所述存储器数据缓冲器取回所述数据之后的“未占用”状态指示符；

用于响应于所述外部设备对所述第一可高速缓冲存储区域的存取，将存储于所述第一可高速缓冲存储区域中的当前状态数据值与所述缓冲器标记占用指示符状态数据值进行比较的装置；

用于停止执行所述预留丢失时加载指令 LDRL 直到所述处理器判定所述外部设备已经复位与所述第一可高速缓冲存储区域相关联的所述加载 / 存储操作预留的装置，其中所述外部设备是通过如下方式来进行所述复位的，即所述外部设备在所述第一可高速缓冲存储区域中存储“未占用”指示符状态数据值以表明所述外部设备已经完成从所述存储器数据缓冲器取回所述数据；以及

用于响应于所述外部设备在所述第一可高速缓冲存储区域中存储所述“未占用”指示符状态数据值、由所述处理器在所述存储器数据缓冲器中存储附加数据以便由所述外部设备取回的装置。

5. 根据权利要求 4 所述的设备，还包括：

用于检测进程上下文切换的装置；以及

用于响应于检测到所述进程上下文切换，复位与所述第一可高速缓冲存储区域相关联的所述加载 / 存储操作预留的装置。

6. 根据权利要求 4 所述的设备，其中用于经由所述存储和预留指令 STAR 设置与所述第一可高速缓冲存储区域相关联的所述加载 / 存储操作预留的装置包括：

用于复位与所述存储器内的第二可高速缓冲存储区域相关联的先前的加载 / 存储操作预留的装置；以及

用于响应于已经复位与所述第二可高速缓冲存储区域相关联的所述先前的加载 / 存储操作预留的判定，设置与所述第一可高速缓冲存储区域相关联的所述加载 / 存储操作预留的装置。

## 执行高速缓存线轮询操作的方法、系统和设备

### 技术领域

[0001] 本发明的实施例一般地涉及数据处理系统的操作,并且更特别地涉及一种利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询的方法、系统、设备和产品。

### 背景技术

[0002] 在现代计算技术出现之初,信息处理(例如:计算机)系统包括有限数量的组件,包括:单处理器、系统存储器和少量的输入/输出(I/O)装置,例如显示装置、键盘以及与图形用户接口的创建有关的指针控制装置(例如鼠标、轨迹球或者类似装置)。然而,随着信息处理系统的发展,经由通信和对于系统共享资源的竞争而互相连接的系统组件数量显著增加。因此,现代的常规信息处理系统可能包括大量的系统组件(例如,利用 SMP(对称多处理器)、ASMP(非对称多处理器)、NUMA(非一致性存储器存取)或者类似配置的多个处理器,协处理器,直接存储器存取控制器,以及每个都有可能包含附属的处理器、寄存器和存储器的 I/O 装置。)

[0003] 为了在现代信息处理系统中协调系统组件的活动,已经采用了多种技术。信息处理系统组件可以利用与中断服务例程或处理器结合的中断来传达和/或指示一个事件的发生。同样,可利用存储器映射的 I/O 和端口或“端口映射”的 I/O 在系统组件(例如处理器和 I/O 装置)之间提供通信。

[0004] 在用于执行输入/输出(I/O)操作目的的单元间的数据传送中,信息处理系统的单元之间的活动的协调是特别重要的。例如,在信息处理系统处理器向缓冲器中存放了旨在由 I/O 装置或者多处理器系统中的另一处理器进行处理的数据之后,数据提供处理器通常将通知 I/O 装置或数据接收处理器到缓存器的数据传送已完成。在常规信息处理系统中,通常通过向 I/O 装置或数据接收处理器中的存储器映射输入/输出(MIMO)寄存器写入特定数据值来完成这样的通知。检测到对关联 MIMO 寄存器的写操作之后,I/O 装置或数据接收处理器可以经由直接存储器存取(DMA)从缓冲器中取回数据。

[0005] 在一些常规信息处理系统中,可以经由 MMIO 寄存器轮询或经由中断来检测数据的 DMA 取回的完成。然而因为对于相对小的缓冲器来说中断开销通常太大,以及 MMIO 寄存器轮询低效地使用总线带宽(否则该总线带宽将可以用于 DMA 传送,从而提高整个系统的吞吐量),所以 MMIO 寄存器轮询和中断都不是检测 DMA 完成的有效机制。

[0006] 在另一种被称为“高速缓存线轮询”的用于检测 DMA 完成的常规技术中,在通知 I/O 装置(例如:经由 MMIO)有关缓冲器的可用性之前,将预先确定的“占用”指示符数据值写入可高速缓冲存储区域,通常将其称为缓冲器标记或信号(semaphore)。然后处理器针对预先确定的“未占用”指示符数据值轮询缓冲器标记,以检测相应 DMA 的完成。由于数据已经在处理器的高速缓冲存储器中修改,高速缓存线轮询不会产生任何附加的总线活动。在完成从缓冲器中取回(DMA)数据后,I/O 装置或接收处理器将“未占用”完成数据值写入缓冲器标记。然后,数据提供处理器可以经由标准的高速缓冲一致性协议存取新缓冲器标记值,其间高速缓冲存储器中“占用”指示缓冲器标记数据失效或由新完成值代替。

[0007] 从系统的观点来看,高速缓存线轮询是一种有效的轮询机制。然而,为了实现高速缓存线轮询,数据提供处理器重复执行“轮询”指令集直到 DMA 传送完成且缓冲器标记值更新,这样浪费了有价值的系统资源(例如:处理器周期、总线周期,电量,指令或线程分派时隙,或类似的资源)。

### 发明内容

[0008] 在此公开了一种用于利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询的方法、系统,设备和产品。在一个实施例中,提供了一种方法,其包括在第一可高速缓冲存储区域中存储缓冲器标记占用指示符数据值,并且经由存储和预留指令在所述第一可高速缓冲存储区域设置加载/存储操作预留。在所述实施例中,响应于已经在第一可高速缓冲存储区域上复位加载/存储操作预留的判定,经由预留丢失时加载指令存取存储于第一可高速缓冲存储区域的数据值。相反地,响应于没有在第一可高速缓冲存储区域上复位加载/存储操作预留的判定,停止执行预留丢失时加载指令。

[0009] 前述是一个概要并因而必然包括对细节的简化、概括和省略;所以,本领域的普通技术人员应该明白此概要仅是说明性的,并且不旨在进行任何限制。同样,本领域的普通技术人员将明白,能以许多方式实现在此公开的操作,包括硬件形式、软件形式或其组合形式的实现,而且在不脱离本发明及其广泛方面的情况下,可进行这些改变和修正。仅由权利要求限定的本发明的其他方面、发明特征以及优点在后面阐述的非限制性详细描述中将变得显而易见。

### 附图说明

[0010] 参考附图,本发明将更容易理解,并且对本领域的普通技术人员来说,其多数特征和优点将变得显而易见,其中:

[0011] 图 1 示出了包括根据本发明的一个实施例的信息处理系统的通信网络;

[0012] 图 2 示出了根据本发明的一个实施例的信息处理系统的高级框图;

[0013] 图 3 示出了表示根据本发明的一个实施例的能够利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询的信息处理系统选定部分的框图;

[0014] 图 4 示出了根据本发明的一个实施例的用于针对存储和预留指令以及预留丢失时加载指令而管理预留的状态机的状态图;

[0015] 图 5 示出了根据本发明的一个实施例的利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询的过程的流程图;以及

[0016] 图 6 示出了根据本发明的一个实施例的用于经由存储和预留指令以及预留丢失时加载指令利用进程间通信执行高速缓存线轮询的过程的执行流程图。

[0017] 附图中使用相同或相似的参考符号来指示相似或同样的项目。

### 具体实施方式

[0018] 以下内容详细描述了至少是完成这里描述的一个或多个系统、装置和/或过程最好的预期方式。这些描述用于说明并且不应该看作限制。

[0019] 在以下详细描述中,给出了许多特定细节,比如特定的方法顺序、结构、单元和连

接。然而应该明白,不需要使用这些和其他特定细节来实现本发明的实施例。在其他情况下,省略了或者没有特别详细地描述众所周知的结构、单元或者连接,目的是避免不必要地使描述不清楚。

[0020] 说明书中引用“一个实施例”、“实施例”或者“多个实施例”旨在有关于实施例而描述的特定特征、结构或特性包括在本发明的至少一个实施例中。在说明书中不同地方出现这样的短语不一定是指同一实施例,也不一定是指与其他实施例相互排斥的独立或可选实施例。此外,描述的多种特征可能由某些实施例展示,而不由其他实施例展示。同样,描述的多种需求可能是一些实施例所需要的,而不是另一些实施例所需要的。

[0021] 本发明实施例提供了可用于执行高速缓存线轮询的存储和预留指令以及预留丢失时加载指令,其包含在这里所述的方法、设备、信息处理系统和机器可读介质产品中。图 1 示出了包括根据本发明的一个实施例的信息处理系统的通信网络。在本发明的一个或更多实施例中,在设置加载预留的先前设置的加载预留已经“丢失”或复位后,在所述存储和预留 (STAR) 指令之后执行的常规加载 (LD) 指令将完成执行并返回数据,否则通常将停止。同样,在相应的加载 / 存储操作预留丢失或复位后,预留丢失时加载 (LDRL) 指令可能完成执行并返回数据。在一个实施例中,STAR 和 LDRL 指令可组合在单一的高速缓存线轮询例程循环中,以便在进程上下文切换期间正确地管理预留的设置和复位。与常规加载相比,增加了 LDRL 指令允许程序或处理器检查缓冲器标记数据值的状态,而不会引起处理器线程不必要的停止。

[0022] 如图 1 所示,系统 101 包括网络 104,多个信息处理系统(例如计算机和计算装置)和它相连。在本发明的多种实施例中,网络 104 可以包括局域网、全球网(例如因特网)或任何其他通信网络。在图 1 所示的实施例中,连接到网络 104 的信息处理系统包括客户端计算机 106、服务器计算机 108、个人数字助理 (PDA) 110、数字电视 (DTV) 112 并且还可以包括其他没有显示的有线或无线计算机和计算装置。在所示的实施例中,网络 104 的成员信息处理系统使用的处理部件由通用的计算模块构成。这些处理部件优选地还都具有相同的指令集体系结构 (ISA),并根据通用的处理器指令集完成处理。

[0023] 在图 1 所示的实施例中,包含在任意特定处理部件内的计算模块数量依赖于该处理部件要完成的信息处理需要的处理能力。例如,由于系统 101 的服务器 108 比客户端 106 完成更多的数据和应用处理,所以服务器 108 比客户端 106 包含更多的计算模块。另外,PDA110 完成相对少量的处理。在所示的实施例中,每个计算模块包含处理控制器和多个同样的处理单元,用以完成网络 104 上传的数据和应用的并行处理。

[0024] 系统 101 的同类配置提高了适应性、处理速度和处理效率。因为系统 101 的每一成员利用一个或多个(或一些部分)同样的计算模块完成处理,所以完成数据和应用实际处理的特定计算机或计算装置与在常规系统中相比具有更少的相关性。而且,特定应用和数据的处理可以在网络成员间共享。通过在全系统内唯一地标识包含由系统 101 的处理数据和应用的单元,可以将处理结果传送给请求处理的计算机或计算装置,而不管处理发生在哪里。因为完成该处理的模块具有通用的结构和使用通用的 ISA,所以避免了用于达到处理部件之间兼容的附加软件层的计算负担。这种体系结构和编程模型提高了例如实时执行多媒体应用所必须的处理速度。

[0025] 为了更好的利用由系统 101 提高的处理速度和效率,将该系统处理的数据和应用

打包为唯一识别的、统一格式的软件单元 102。每个软件单元 102 包括,或者可以包括,应用和数据。每个软件单元 102 还包括一个 ID(标识符),以便在网络 104 和系统 101 中全局地识别该单元。软件单元的这种结构统一性和软件单元在整个网络上的唯一标识有利于在网络 104 的任何计算机和计算装置上处理应用和数据。例如,客户端 106 可以设计一个软件单元 102,但是因为客户端 106 的处理能力的限制,它将该软件单元传送给服务器 108 进行处理。因此,软件单元可以在整个网络 104 内转移,以便根据网络上处理资源的可用性来进行处理。

[0026] 系统 101 的处理部件和软件单元的同类结构也避免了目前异构网络的许多问题。例如,避免了效率低下的编程模型,其寻求允许在使用例如虚拟机(诸如 JAVA 虚拟机之类)的任何 ISA 上处理应用。因此,系统 101 可以实现远比常规网络更有效且效率更高的宽带处理。

[0027] 图 2 示出了根据本发明实施例的信息处理系统的高级框图。虽然针对图 2 的信息处理系统示出了特定数量和排列的部件,但应该意识到,本发明的实施例不限于具有任何特定数量、类型或者排列的组件的数据处理系统,而且可以包括多种数据处理系统类型、体系结构和形式要素(例如网络部件或节点、个人计算机、工作站、服务器等)。

[0028] 图 2 描述的信息处理系统是单元宽带引擎(Cell BroadbandEngine:CBE)体系结构的一个例子,其中可以实现本发明的示例性方面。如图 2 所示,CBE 200 包括一个主处理器单元(PPE)210 和多个协处理器单元(SPE)220-234,它们通过高速宽带内部单元互连总线(EIB)286 通信互连并和这里还将描述的附加系统单元通信连接。如图所示,所示出的实施例的 CBE200 还包括一个或多个外部总线或装置 290,它通过总线接口控制器(BIC)287 连接到 EIB286;还包括共享存储器 289,它通过存储器接口控制器(MIC)288 连接到 EIB286。

[0029] CBE200 可以是一个芯片级系统,从而图 2 描述的每个单元可以提供于单个微处理器芯片上。而且,在一个实施例中,CBE200 可以提供为异构处理环境,其中系统中 SPE220-234 中每一个可以从每个其他的 SPE 接收不同的指令;并且,每个 SPE 的指令集不同于 PPE210 的指令集,例如 PPE210 可以执行基于精简指令集计算机(RISC)的指令,而 SPE200-234 执行向量指令。

[0030] 在图 2 所示的实施例中,SPE220-234 通过 EIB286 彼此相连并连接到 PPE210。另外,SPE220-234 的每一个通过 EIB286 分别连接到 MIC288 和 BIC287。MIC288 提供到共享存储器 289 的通信接口。共享存储器 289 可包括多个系统存储器类型的存储单元(例如随机存取存储器(RAM)、只读存储器(ROM)、闪存等)中的任何一个。BIC287 提供 CBE200 与其他外部总线和装置 290 之间的通信接口。示例性外部装置可包括:传统的 I/O 装置,比如键盘、显示器、打印机、指针控制装置(例如轨迹球、鼠标、手写板等)、扬声器以及麦克风;存储装置,比如固定的或“硬”磁媒体存储装置、光存储装置(例如 CD 或者 DVD ROM)、固态存储装置(例如 USB、安全数字 SD™ 卡、压缩闪存 CF™ 卡、MMC 等)、可移动磁介质存储装置(比如软盘和磁带),或者其他存储装置或介质;以及有线的或无线的通信装置或媒体(例如通过调制解调器或直接网络接口存取的通信网络)。

[0031] 在本发明的一个实施例中,PPE210 是双线程处理单元。该双线程 PPE210 和 8 个 SPE220-234 的结合使 CBE200 能够处理 10 个并行线程以及超过 228 个未完成的存储器请求。在通常操作环境下,PPE210 充当 8 个 SPE220-234 的控制器,这些 SPE 处理多数的计算

工作量,例如,PPE210 可用于执行一个或多个传统的操作系统,而 SPE220-234 完成向量浮点编码的执行。

[0032] 在一个实施例中,PPE210 包括主处理器单元 (PPU) 或核心以及相关的一级 (L1) 和二级 (L2) 高速缓冲存储器 (未示出);并且 SPE220-234 中的每个 SPE 包括协处理单元 (SPU)、存储器流控制单元、本地存储器以及包括直接存储器存取 (DMA) 控制器、存储器管理单元 (MMU) 和总线接口单元 (未示出) 的组的总线接口单元。在一个示例性实施例中,所述本地存储器包括 256KB 的指令和数据存储器,其对于 PPE210 是可见的并且可由软件直接寻址。

[0033] PPE210 可用小程序或线程加载 SPE220-234,将 SPE 链接在一起处理复杂操作中的每个步骤。例如,装有 CBE200 的机顶盒可加载用于读 DVD、进行视频和音频解码并显示的程序,并且数据从 SPE 传到 SPE 直到最终显示在输出显示器上。在 4GHZ,每个 SPE220-234 给出理论上 32GFLOPS 的性能,PPE210 也有相似性能级别。在操作中,PPE210 也可以执行指令并将从共享存储器 289 取回的数据通过 MIC288 处理到其本地寄存器或高速缓存。同样,外部装置 290 也可例如通过 BIC287 和 SPE220-234 中一个或多个 DMA 控制器访问共享存储器 289。

[0034] 图 3 示出了表示根据本发明实施例的信息处理系统的选定部分的框图,其能够利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询。在当前描述中,类似的参考标号用以表示图 2 和图 3 的信息处理系统之间相应的系统单元。例如,图 3 所示实施例中的 PPE310 对应于图 2 中的 PPE210。图 3 中的信息处理系统 300 包括 PPE310,其通过 EIB386 分别利用 MIC388 和 BIC387 连接到共享存储器 389 和外部装置 390。

[0035] 在图 3 所示的实施例中,共享存储器 389 包括可高速缓冲存储区域 336,如图所示其包含指定缓冲器标记数据值的数据。PPE310 包含主处理单元 (PPU) 316,如图所示其分级连接到 L1 高速缓冲存储器 312 和 L2 高速缓冲存储器 314。在图 3 的实施例中,PPU316 包括多个功能单元和数据存储单元。更具体地,PPU316 包括:加载/存储单元 318,用来执行存储器存取指令(例如,从存储器中加载和存入存储器);和条件寄存器 320,其以位或标记的形式存储数据,其指示 PPU316 的当前状态,反映某些数据处理或信息处理操作的结果(例如,数据溢出或下溢,肯定的或否定的结果,或类似情况)。

[0036] L1 高速缓冲存储器 312 和 L2 高速缓冲存储器 314 中的每一个包括高速缓冲存储器管理单元 (CMU) (例如,L1 高速缓冲存储器 312 的 CMU322 和 L2 高速缓冲存储器 314 的 CMU328) 以及存储单元 (例如,L1 高速缓冲存储器 312 的存储单元 324 和 L2 高速缓冲存储器 314 的存储单元 332)。CMU322 和 CMU328 均用于控制相应存储单元 324 和 332 内的数据和/或指令的存储,例如执行高速缓存线替换算法、更新高速缓存线状态或状态元数据等等。存储单元 324 和 332 接着用于存储数据线或块,包括应用数据和/或指令以及伴随的元数据 (例如,高速缓冲存储器标签、状态位等)。虽然 CMU322 和 CMU328 描述为它们各自高速缓冲存储器的完整单元或模块,但在本发明的可选实施例中,CMU322 和/或 CMU328 或其功能可以以另外的配置提供 (例如:在 L1 高速缓冲存储器 312 和 L2 高速缓冲存储器 314 之一内,在 PPU316 内,作为独立单元或模块,或者单元或模块的组合)。

[0037] 根据本发明的一个实施例,PPU316 可以用于通过执行存储和预留 (STAR) 指令以及预留丢失时加载 (LDRL) 指令而完成高速缓存线轮询,现在将对此进行描述。在操作中,



PPU316 的加载 / 存储单元 318 可首先用于执行 STAR 指令, 以使得将指定“占用”缓冲器标记指示符数据值 (例如 :0xBB) 的数据存入共享存储器 389 的可高速缓冲存储区域 336, 如虚线 338 所指示的那样。在所示实施例中, 所述 STAR 指令用来信号通知相关外部装置 390 (例如 :图形装置) : 将由该装置取回的数据已经存储于相关缓冲器 (例如 :共享存储器 389 的专用部分或者信息处理系统 300 内部或外部的其他存储器) 中。在本发明的其他实施例中, 更多的操作 (例如 :异常或中断产生、信令、MMIO 写操作等等) 可用来通知外部装置 390 数据已写成存储于可高速缓冲存储器特定区域 336 中的缓冲器标记, 而且可以着手从所述缓冲器取回数据。

[0038] 一旦 PPU316 完成所述 STAR, 就可以利用 PPE310 继续执行高速缓存线轮询, 其中可以执行 LDRL 指令以使得将存储于共享存储器 389 的可高速缓冲存储区域 336 的数据存入 PPU316 的一个寄存器 (例如 :通用寄存器) 中 (未显示)。如这里进一步描述地, 所述 STAR 和 / 或 LDRL 指令可以使得通过在 L2 高速缓冲存储器 314 内的预留寄存器 330 内存储特定数据值而设定预留。在一个实施例中, 通过在预留寄存器 330 的预留位以及与预留相关的相应存储区域 (例如 :与共享存储器 389 的可高速缓冲存储区域 336 相关的特定存储器位置或区域) 的存储器地址中存储逻辑“1”来设置预留。在所述实施例中, 预留寄存器 330 中预留的设置和复位使得条件寄存器 320 内相应预留状态位被设置或复位。尽管预留寄存器 330 在 L2 高速缓冲存储器 314 的 CMU328 中示出, 但在本发明另外的实施例中, 这样的预留寄存器或数据也可存储于信息处理系统 300 的其他位置 (例如 :在 L1 高速缓冲存储器 312、PU316、独立的总线 /EIB 接口单元等内)。

[0039] 在已经设置预留后, 所述 LDRL 指令停止运行 (例如临时挂起而不执行或发行 / 完成), 直到在多个信息处理系统事件中的一个或多个事件发生后, 预留被清除或“复位”, 正如这里将更充分地描述并由条件寄存器 320 中相应预留状态位所指示的那样。在一个实施例中, 在检测到 (例如 :由外部装置 390) 尝试、请求或完成如虚线 340 所示的向可高速缓冲存储区域 336 的写操作后, 可以由 L2 高速缓冲存储器 314 的 CMU328 使用高速缓冲存储器“监听”操作复位预留。一旦外部装置 390 修改了存储于可高速缓冲存储区域 336 内的缓冲器标记数据, 就会使得预留被复位并因此“丢失”, 先前停止的 LDRL 指令可以重新开始, 导致缓冲器标记的数据值分别分级存储在 L1 高速缓冲存储器 312 和 L2 高速缓冲存储器 314 内的存储单元 324 和 332 中, 并最终分级存储在 PPU316 内的特定寄存器中。

[0040] 如这里将更充分地描述的那样, 一旦 LDRL 操作成功完成, 就可以将取回的缓冲器标记的数据值与已知的“占用”和 / 或“未占用”缓冲器标记指示符数据值进行比较。所述比较可以用于判定外部装置 390 是否已经完成取回 (例如通过 DMA 传输) 先前存储于相关缓冲器中的数据, 从而例如相关缓冲器可再用于到外部装置 390 的其他数据传输。

[0041] 图 4 示出了根据本发明的一个实施例的状态机的状态图, 其用于针对存储和预留指令以及预留丢失时加载指令来管理预留。因此所示“状态机”代表了将由信息处理系统 (例如, 图 3 中的信息处理系统 300) 的一个或多个单元完成的操作或并入到所述单元中的功能性。在一个实施例中, 这样的功能性被并入例如图 3 中的 PPU316 的处理部件或单元, 在其他的实施例中, 这样的功能性可包含在能够监视和控制相关信息处理系统的操作的独立的或附加的系统单元中。在本发明的一个实施例中, 针对发起或支持的每个硬件线程, 举例说明了图 4 状态图表示的功能性。

[0042] 如图所示,状态机 400 包括五个独立状态,即状态 S0、状态 S1、状态 S2、状态 S3 和状态 S4。在一个实施例中,使用存储指令来设置缓冲器标记“占用”指示符数据值并初始化要存取的缓冲器数据的状态。在另一个实施例中,所述存储指令还用于设置加载预留。

[0043] 在初始状态 S0,状态机 400 收到条件加载(例如:预留丢失时加载指令)后,记录针对缓冲器标记的高速缓存线地址,并进入状态 S1。所述条件加载的对象是相应于存储缓冲器标记的高速缓存线的地址。多个加载预留可同时存在。

[0044] 在高速缓冲存储器状态检查的状态 S1,检查高速缓冲存储器的状态。如果存储缓冲器标记的高速缓存线是无效的,则状态机 400 进入状态 S4。如果存储缓冲器标记的高速缓存线被修改或共享,则状态机 400 进入状态 S2。

[0045] 在等待预留丢失的状态 S2,当存在加载预留用于加载操作时,状态机 400 保持空闲。当加载预留丢失后,状态机 400 进入状态 S4。

[0046] 高速缓冲存储器具有一种机制可以检测是否另一处理器正在存取其高速缓存线中的一个。这种机制通常称为监听器。同样的处理可以由状态机 400 用来判定高速缓存线是否被其他处理器或装置修改。另外,状态机 400 监视同一处理器上的其他线程或共享高速缓冲存储器的其他处理器对缓冲器标记的存储指令。

[0047] 如果状态 S2 的唯一退出是由于存储缓冲器标记的高速缓存线被修改,那么处理器可能死锁(也就是不再进行任何操作)。在本发明的一个或更多可选实施例中,增加了其他的退出条件以使得即使加载预留没有丢失时状态机 400 也可以进入状态 S4,以避免可能死锁的情况。例如,可以使用中断来启动或导致状态 S2 到状态 S4 的转换。在所述实施例中,如果中断指向停止的处理器或处理器线程,则状态机 400 将退出到状态 S4,以使得中断被处理。如果中断不被处理,则处理器或其他装置就不更新缓冲器标记。

[0048] 在另一示例性实施例中,可以使用超时来引起状态 S2 到状态 S4 的转换。为了避免在不可接受的长时间内等待发生在状态 S2 和状态 S4 之间的转换,可以使用软件来触发针对高速缓存线轮询周期的超时。利用超时选项,状态机 400 经过特定时间量后将退出到状态 S4。在本发明的可选实施例中,可以事先设置超时值或者将超时值设置为预留丢失时加载指令的参数。

[0049] 在图 4 所示的状态图中,如果存储缓冲器标记的高速缓存线由于其他加载指令或需要同一高速缓存线的存储指令而被清除(castout),则可以进入等待预留丢失(缓冲器标记不在处理器的高速缓冲存储器内)状态 S3。除缓冲器标记没有存储于处理器的高速缓冲存储器中之外,状态 S3 和先前描述的状态 S2 基本上相同。状态 S3 存在和状态 S2 相同的退出条件,然而在状态 S3 中,在处理器的高速缓冲存储器中,将存储缓冲器标记的高速缓存线标示为不再有效(即被修改或共享)。在高速缓存线清除操作可能引起预留被复位的实施例中,可以移除状态 S3。在另一实施例中,如果高速缓冲存储器从系统总线预载数据(例如:高速缓冲存储器注入),则可能发生从状态 S3 返回状态 S2 的转换。

[0050] 在数据转发状态 S4,检查存储缓冲器标记的高速缓存线以判定存储缓冲器标记的高速缓存线是否包含有效数据(即共享的、修改的等等)。如果包含,则将数据转发给处理器并且状态机 400 重新进入状态 S0。这是缓冲器标记数据没有被处理器或装置修改的情况,因为加载预留的丢失是由其他事件(例如:中断或超时)引起的。在这种情况下,缓冲器标记包括指定“占用”指示符数据值的数据,并且然后执行比较指令。如果中断正待处理,

则接下来处理中断。当中断返回时,如果返回的数据是“占用”指示符数据值,则将采用分支。如果采用分支,则启动高速缓存线轮询例程的加载预留将再次开始加载预留过程。

[0051] 如果存储缓冲器标记的高速缓存线包含无效数据,则生成“加载失败 (load miss)”并且通过总线从共享存储器请求缓冲器标记数据。当缓冲器标记数据返回后,数据被转发给处理器,并且高速缓冲存储器的状态被更新。状态机 400 然后进入状态 S0。这个过程和高速缓冲存储器中失败的正常加载操作相同。在这种情况下,数据被装置修改为包含“未占用”指示符数据值(例如:0xBC 或者不等于 0xBB 的任何值)。

[0052] 其后,执行比较指令。如果返回的数据没有指定“占用”指示符数据值,则不会采用分支并且高速缓存线轮询例程退出,表明外部装置已经完成从相关缓冲器取回数据。

[0053] 有几种情况会引起预留丢失,包括:含有预留丢失时加载指令的地址所引用数据的高速缓存线的无效;同一处理器或共享同一高速缓冲存储器的另一处理器对与预留丢失时加载指令的地址相关联的数据的修改;对已停止的先前执行预留丢失时加载指令的线程提示中断;前述状态 S3 中,由同一处理器或共享高速缓冲存储器的另一处理器进行存储器操作引起的传统高速缓存线的替换/弹出/清除;或者一个或多个其他退出情况(例如:条件加载指令的超时)发生。

[0054] 在所述实施例中,高速缓存线的无效可以由一个装置或其他处理器读取高速缓存线以便修改其中存储的数据(RWITM)而引起,或由一个装置或其他处理器写高速缓存线(清洗性地写)而引起。

[0055] 尽管这里已经将缓冲器标记或信号描述为存储于可高速缓冲存储区域内,但在可选实施例中,存储区域不需要是可高速缓冲的。更明确地,本发明的方法或处理可以用于位于不可高速缓冲存储区域的缓冲器标记,只要处理器具有用于监听更新含有缓冲器标记的存储器地址的设备的装置。通过状态 S3 的存在,使得缓冲器标记位于不可高速缓冲存储区域变得明显。在状态 S3 中,处理器的高速缓冲存储器中的缓冲器标记是无效的,这和存在不可高速缓冲标记是同一状态。

[0056] 图 5 示出了根据本发明的实施例的利用存储和预留指令以及预留丢失时加载指令执行高速缓存线轮询的过程的流程图。为了说明的目的,所述过程实施例将针对图 3 的信息处理系统 300 的系统单元进行描述。首先在图 5 的所示过程实施例中,PPU316 的加载/存储单元 318 用数据(例如:将要由外部装置 390 操作或处理的数据)填充共享存储器 389 中的缓冲器(过程块 502)。在本发明的可选实施例中,向所述缓冲器传送数据可以利用一个或多个 SPE220-234 中提供的 DMA 控制器或引擎来完成。然后,加载/存储单元 318 执行存储和预留指令(过程块 503)。在所示过程实施例中,所述 STAR 指令的执行将指示或指定“占用”指示符数据值的数据存储在共享存储器 389 的可高速缓冲存储区域 336 内(过程块 506)并利用预留寄存器 330 “设置”预留(过程块 506)。

[0057] 其后,可以(例如:通过写 MMIO 寄存器、产生异常、中断、俘获等)通知外部(例如 I/O)装置 390:相关数据缓冲器已准备好被存取(未示出)。可选择地,这样的通知可以只通过如前所述地将缓冲器“占用”指示符数据值存储到高速缓冲存储区 336 内来完成。加载/存储单元 318 然后执行预留丢失时加载(LDRL)指令(过程块 507)。如图 5 所示,执行 LDRL 指令以判定预留是否已经被复位(过程块 508)(例如:通过检查条件寄存器 320 中标记的一位或多位);并且停止 LDRL 指令的进一步执行直到预留已经被复位(例如:通过

由外部装置 390 将“未占用”指示符数据值存储到可高速缓冲存储区域 336 的缓冲器标记中);或者一旦预留被复位,将存储于可高速缓冲存储区域 336 中的缓冲器标记数据加载到 PPU316 的一个寄存器(过程块 510)。尽管已经用一个连续循环来描述对何时预留被复位的判定,但是应该理解,处理器(在单线程的单处理器或多处理器系统内)或相关线程(在多线程的处理器系统内)不执行任何实际的指令,由此保存了有价值的处理能力和电量,而利用常规高速缓存线轮询技术则会浪费掉这些处理能力和电量。

[0058] 然后,PPU316 的定点执行单元(未显示)将已经加载可高速缓冲存储器 336 的缓冲器标记数据的 PPU316 的寄存器的内容与特定的“占用”指示符数据值相比较(过程块 512)。PPU316 的一个分支单元(未显示)随后利用比较结果,判定寄存器内容是否与“占用”指示符数据值相匹配(过程块 514)。然后,响应于寄存器内容不与已知“占用”指示符数据值相匹配的判定,所述过程实施例完全重新启动(过程块 502)或者在 LDRL 指令执行点(过程块 507)重新进入。

[0059] 尽管图 5 所示操作已经针对特定的系统单元进行了描述,但是用于完成这些操作的实际单元对于本发明的过程实施例来说并不重要。而且,在可选的实施例中,这些操作可由任何信息处理系统单元完成。同样,虽然图 5 所示流程图显示了特定的操作次序以及特定的过程操作粒度,但在可选的实施例中,所说明的次序可以变化(例如过程操作可按另一种次序完成,或者基本上并行地完成),并且一个或多个过程操作可能被合并或分解。同样,在本发明一些可选的实施例中,在必要的地方可增加额外的过程操作。

[0060] 本发明的实施例可包括软件、信息处理硬件和这里进一步描述的各种处理操作。各种发明实施例的特征和过程操作可包含在可执行的指令中,这些指令包含于机器可读的介质中,例如共享存储器 289、存储装置、通信装置或介质等。机器可读介质可以包括提供(即,存储和/或传输)机器(例如:CBE200)可读形式的数据的任何机制。

[0061] 例如,机器可读介质包括但不限于:随机存取存储器(RAM);只读存储器(ROM);磁存储媒体;光存储媒体;闪存装置;电的、光的和/或声的传播信号(例如:载波、红外信号、数字信号等);或者类似介质。所述可执行的指令可用于使得该指令编程的通用或专用处理器(比如 PPU316)完成本发明的操作、方法或过程。另外,本发明的特征或操作可由包含用于完成这些操作的硬连线逻辑的特定硬件组件完成,或由已编程的数据处理组件和定制硬件组件的任何组合完成。

[0062] 图 6 示出了根据本发明的一个实施例的用于利用进程间通信通过存储和预留以及预留后加载指令完成高速缓存线轮询的进程的执行流程图。如图 6 中的执行流程所描述的,至少两个进程(进程 A 和进程 B)在一段时间内轮流执行。在进程 A 的上下文中执行完 STAR 指令后,上下文切换到进程 B 的执行上下文。在进程 B 的上下文中,相应 STAR 指令的执行既复位进程 A 先前保持的已存在的预留,又设置了进程 B 的预留。然后在进程 B 的上下文中,当存在 LDRL 指令目标地址的预留(进程 B 的预留)时,LDRL 指令的执行停止,并且 LDRL 不返回任何数据。

[0063] 然后另一个上下文切换出现,使得对于进程 B 的 LDRL 指令返回数据,根据缓冲器标记数据值仍显示缓冲器占用状态的判定,进入循环。在上下文切换回进程 A 后,由其中重新进入的 LDRL 指令返回数据。在图 6 的示例性实施例中,返回的数据指示缓冲器占用状态,引起一个循环执行,包括 LDRL 指令的再次执行,引起针对进程 A 设置另一预留。进程 A 然

后在 LDRL 指令停止,直到预留被复位,在此点处,数据(指示缓冲器不占用状态)返回,并且如图所示退出针对进程 A 的高速缓存线轮询循环。

[0064] 在上下文切换返回进程 B 后,其中重新进入的 LDRL 指令再次返回数据,并且再次设置相应的预留。如图所示,在此点处,返回指示缓冲器占用状态的数据,引起再次进入循环。再次进入后,根据预留状态,LDRL 指令停止,直到缓冲器标记数据值被修改(例如:由外部装置);在此点处,返回指示缓冲器未占用状态的数据,并且退出进程 B 的高速缓存线轮询例程。尽管这里使用了术语“进程”和进程上下文切换,但应该理解也可使用其他执行上下文(例如:不重要的进程、线程、光纤等)。

[0065] 尽管已经在全功能数据处理系统的上下文中描述了本发明,但是本领域的普通技术人员应当理解本发明能作为各种形式的程序产品进行分发,并且无论用于实现该分发的信号承载媒体的具体类型如何,本发明都可以同样地应用。这样的信号承载媒体的例子包括类似软盘和 CD-ROM(光盘只读存储器)的可录媒体、类似数字和模拟通信链路的传输类型媒体、以及将来开发的媒体存储和分发系统。同样,可以利用用以完成确定操作或工作的软件模块实现本发明的实施例。所述软件模块可包括脚本、批处理或其他可执行文件,并且可存储于机器可读或计算机可读的介质上。这样,模块可存储在计算机系统的存储器中,以将数据处理或计算机系统配置为执行软件模块的一个或多个功能。其他新类型和各种类型的机器或计算机可读的存储介质可用来存储在此描述的模块。

[0066] 尽管已经示出和描述了本发明的特定实施例,但是对于本领域的普通技术人员来说,显然,根据这里的启示,在不脱离本发明及其广泛含义的情况下可以进行变化和修改。因此,所附权利要求将在其范围内涵盖落入本发明真实精神和范围内的所有这些改变和修改,并且在充分地认定各方面的等效形式的情况下,本发明的实施例倾向于仅受到所附权利要求的范围的限制。

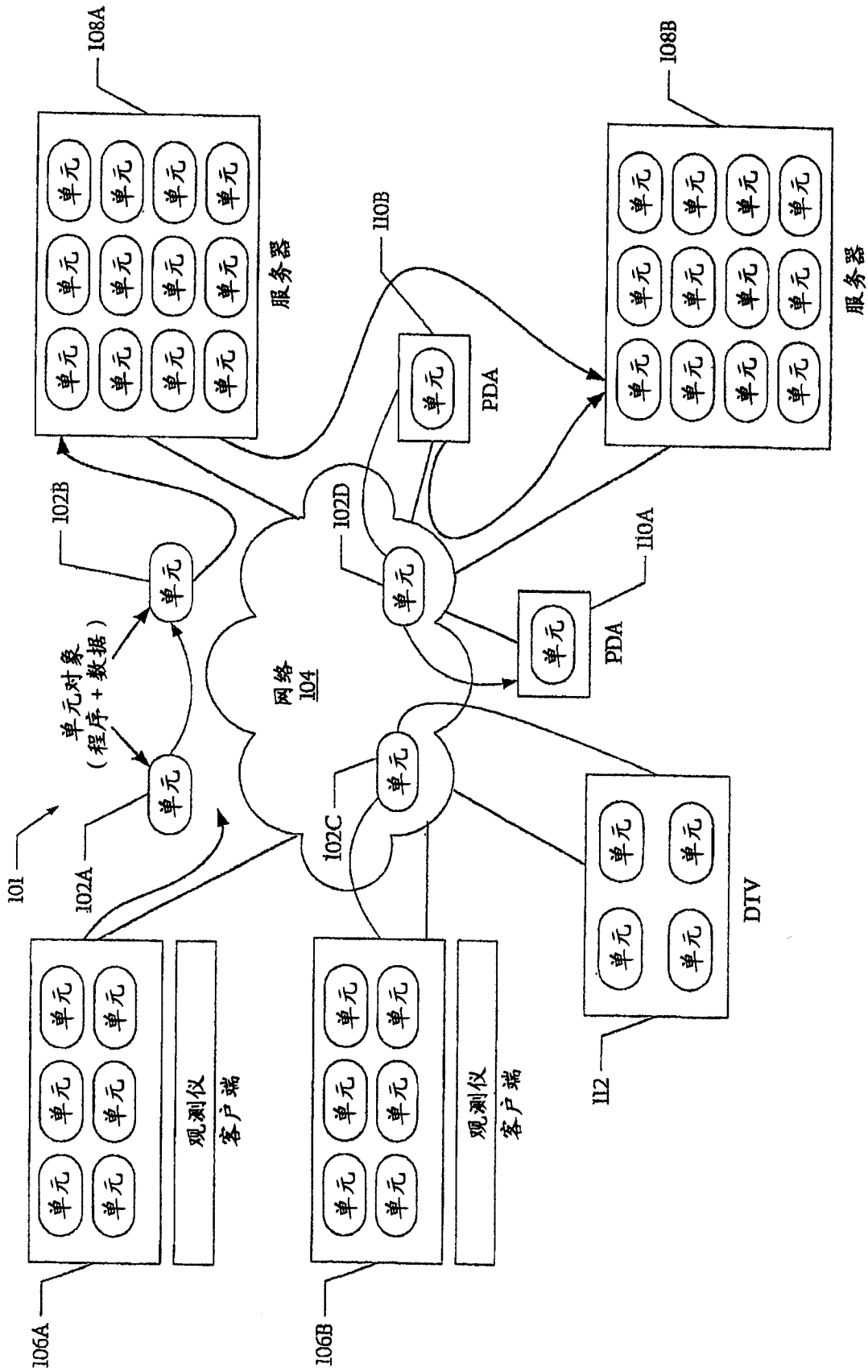


图 1

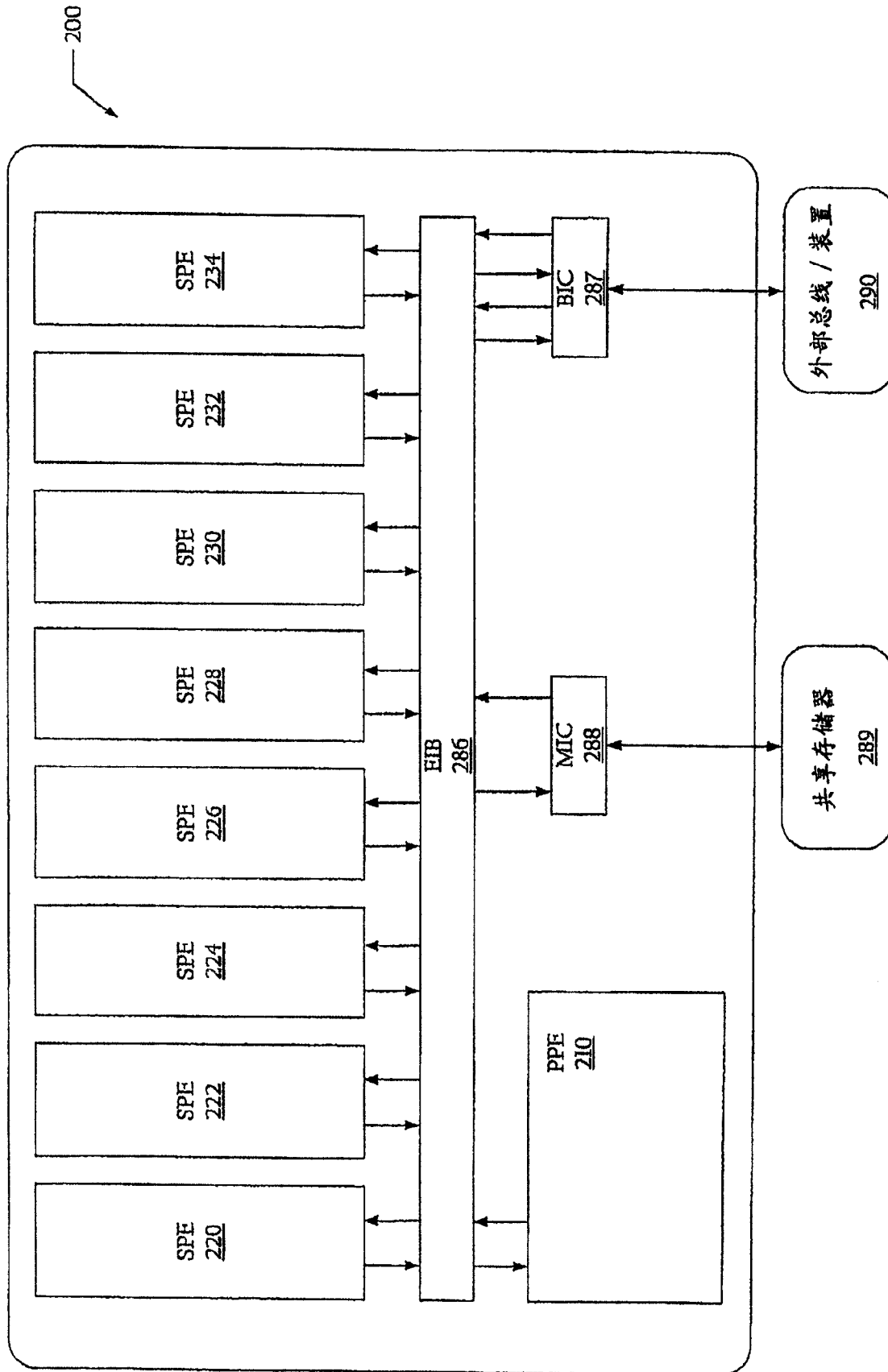


图 2

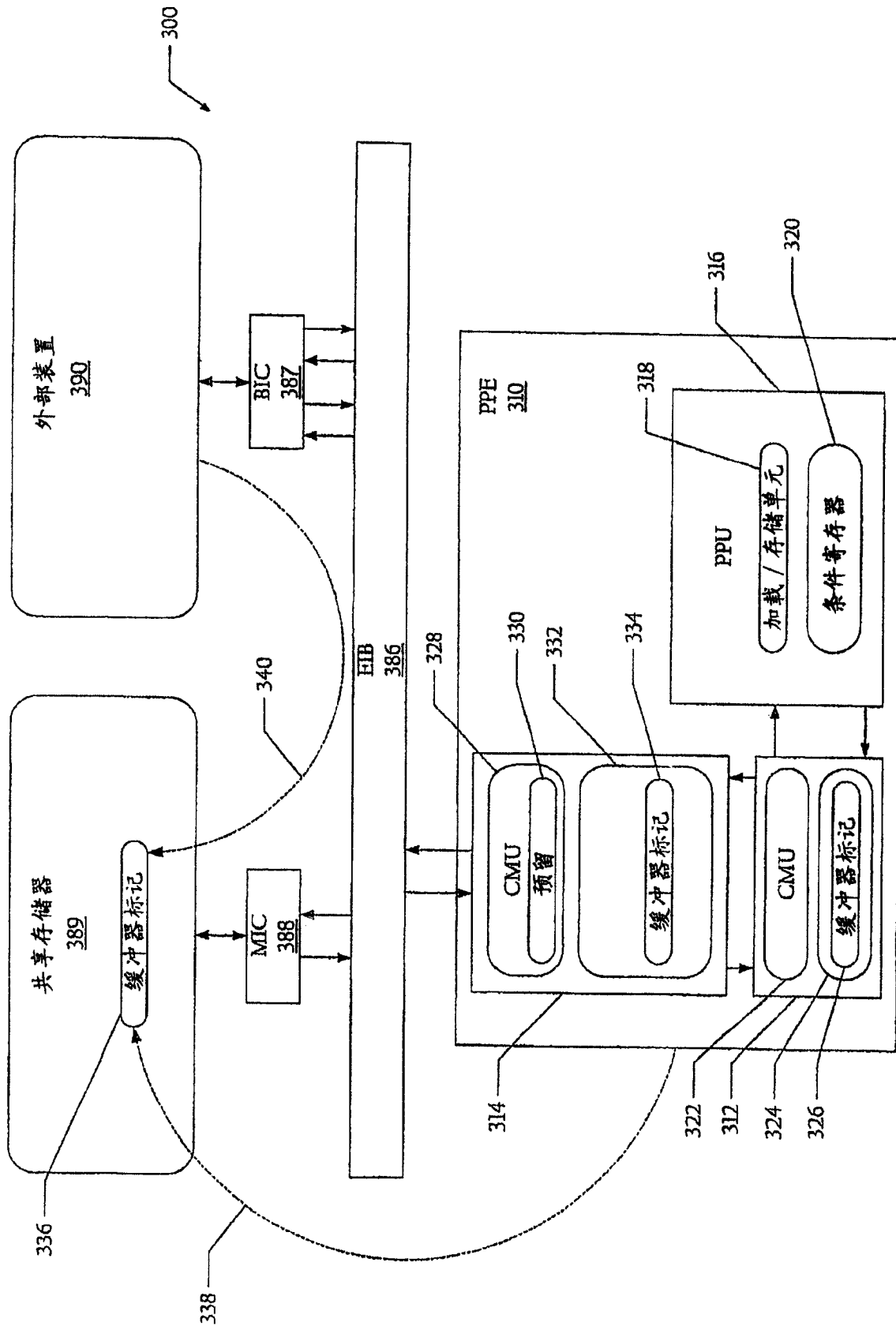


图 3



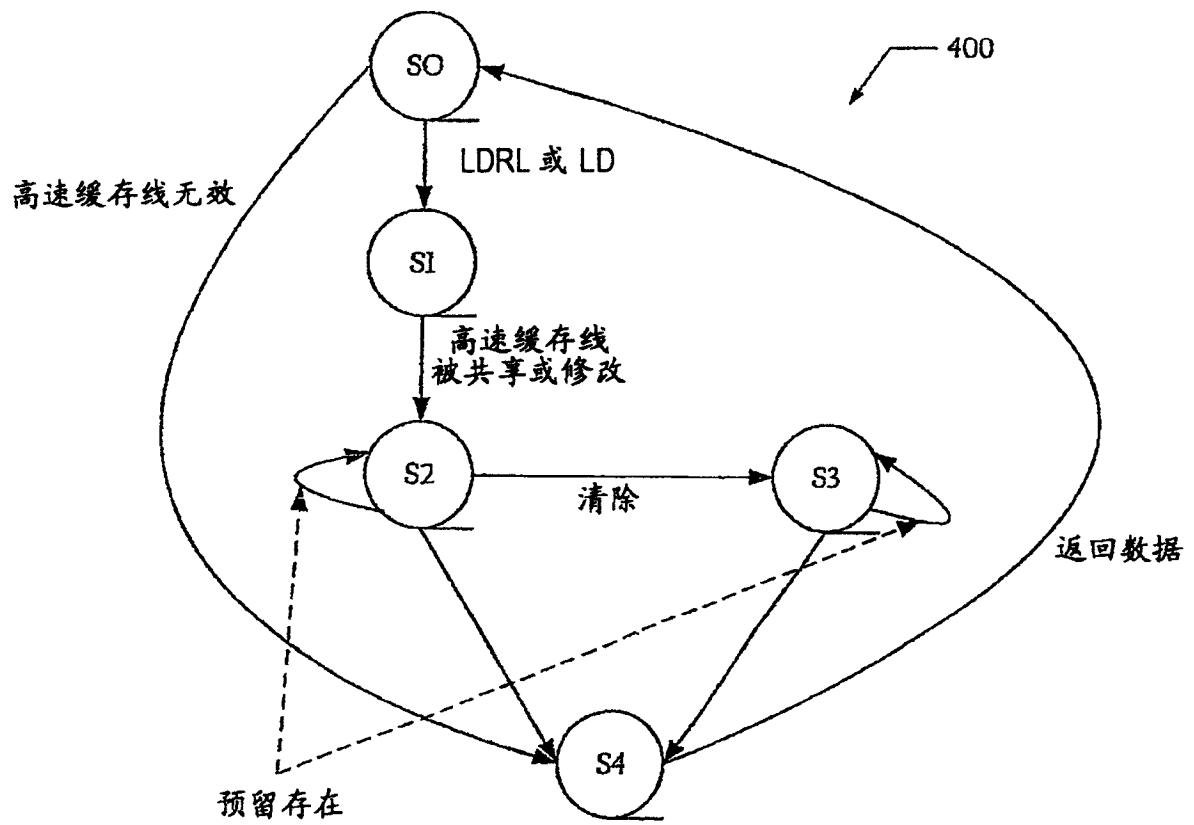


图 4

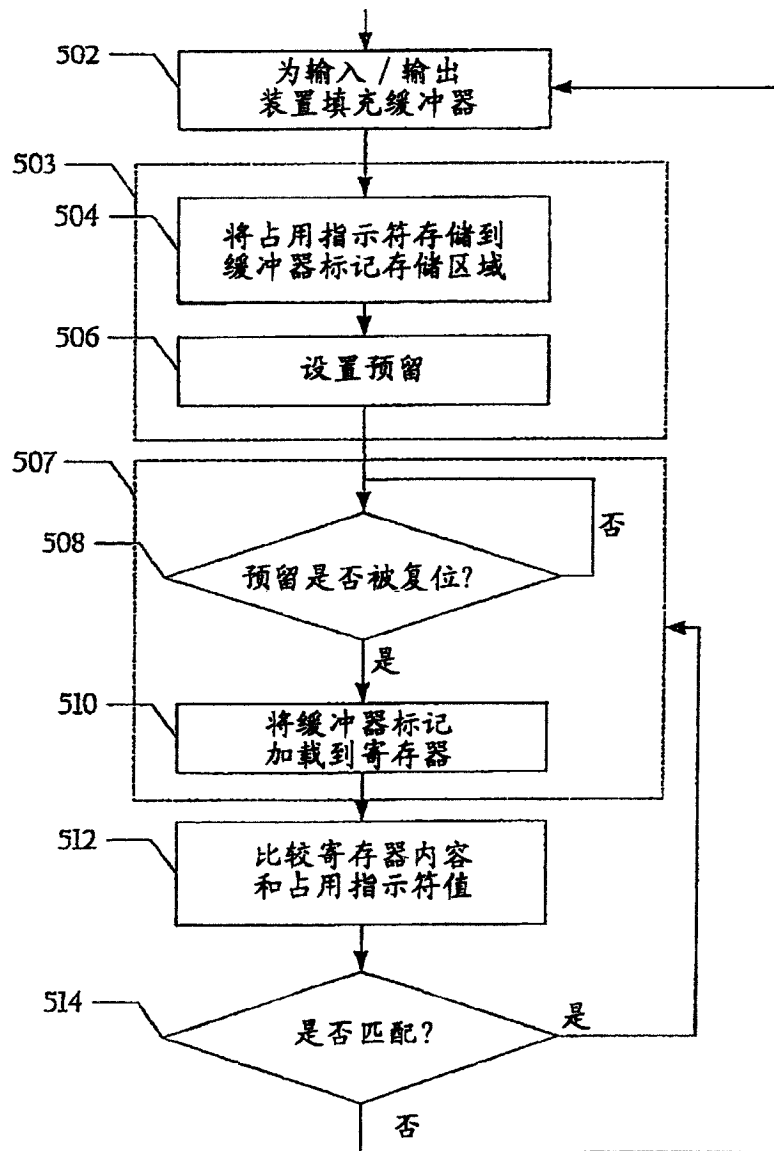


图 5

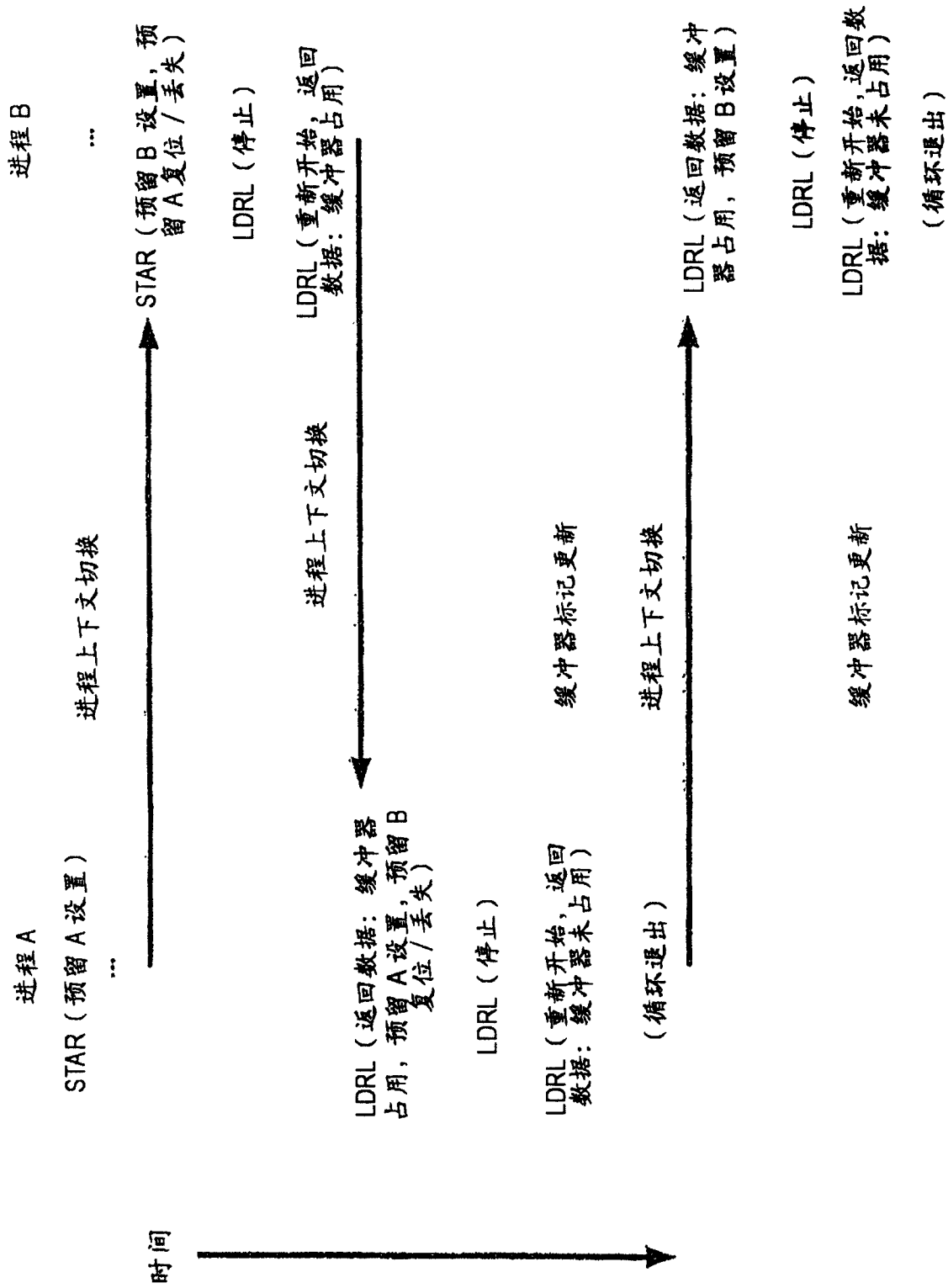


图 6