



(12) 发明专利

(10) 授权公告号 CN 102016790 B

(45) 授权公告日 2014. 08. 06

(21) 申请号 200980115765. 7

(22) 申请日 2009. 02. 23

(30) 优先权数据

12/112, 502 2008. 04. 30 US

(85) PCT国际申请进入国家阶段日

2010. 11. 01

(86) PCT国际申请的申请数据

PCT/US2009/034866 2009. 02. 23

(87) PCT国际申请的公布数据

W02009/134517 EN 2009. 11. 05

(73) 专利权人 飞思卡尔半导体公司

地址 美国得克萨斯

(72) 发明人 威廉·C·莫耶

(74) 专利代理机构 中原信达知识产权代理有限

责任公司 11219

代理人 刘光明 穆德骏

(51) Int. Cl.

G06F 9/06 (2006. 01)

G06F 12/08 (2006. 01)

G06F 9/00 (2006. 01)

(56) 对比文件

US 2008091884 A1, 2008. 04. 17, 说明书第 [0041]–[0058]、附图 1A, 1B, 2, 3.

US 2005251626 A1, 2005. 11. 10, 说明书第 [0012] 段.

US 2008091884 A1, 2008. 04. 17, 说明书第 [0041]–[0058]、附图 1A, 1B, 2, 3.

审查员 李伟华

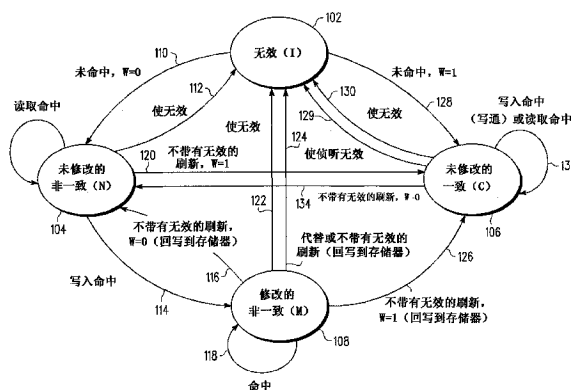
权利要求书3页 说明书14页 附图9页

(54) 发明名称

数据处理系统中的高速缓存一致性协议

(57) 摘要

一种数据处理系统 (10) 包括：具有高速缓存 (28) 的第一主设备 (14)；第二主设备 (16 或 22)；经由系统互连而可操作地耦接到第一主设备和第二主设备的存储器 (20)。所述高速缓存包括为该高速缓存的数据单元实现一组高速缓存一致性状态 (102, 104, 108, 106) 的高速缓存控制器 (29)。所述高速缓存一致性状态包括：无效状态 (102)；未修改的非一致状态 (104)，该状态指示没有修改所述高速缓存的数据单元的数据，并且不保证所述高速缓存的数据单元的数据与数据处理系统的至少一个其他存储设备中的数据是一致的；以及未修改的一致状态 (106)，该状态指示没有修改该数据单元的数据，并且该数据单元的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的。



1. 一种数据处理系统,包括:

第一主设备,所述第一主设备包括高速缓存;

第二主设备;

存储器,所述存储器经由系统互连可操作地耦接到所述第一主设备和所述第二主设备;

其中,所述高速缓存包括高速缓存控制器,所述高速缓存控制器为所述高速缓存的数据单元实现一组高速缓存一致性状态,所述高速缓存一致性状态包括:

无效状态;

未修改的非一致状态,所述未修改的非一致状态指示没有修改所述高速缓存的数据单元中的数据,并且不保证所述高速缓存的数据单元中的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的;和

未修改的一致状态,所述未修改的一致状态指示没有修改所述数据单元的数据,并且所述数据单元的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的。

2. 如权利要求1所述的数据处理系统,其中,响应于所述高速缓存的读取未命中,进行从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换,其中,所述读取未命中的数据被写入到所述数据单元,其中,响应于所述读取未命中不对所述数据处理系统的任何其他高速缓存执行高速缓存一致性操作。

3. 如权利要求1所述的数据处理系统,其中:

响应于写入高速缓存未命中,从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换不产生在所述数据处理系统的其他高速缓存中的高速缓存一致性操作,其中,写入未命中的数据被写入到上述高速缓存的数据单元中。

4. 如权利要求1所述的数据处理系统,其中,从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换不会引起所述数据处理系统的其他高速缓存内的查找操作,所述查找操作用于确定那些高速缓存是否包含对应于正被写入到所述数据单元的数据的存储器地址。

5. 如权利要求1所述的数据处理系统,其中,当所述高速缓存的数据单元处于所述未修改的非一致状态时,指向那个数据单元的侦听事务不会导致那个数据单元的一致性状态的改变。

6. 如权利要求1所述的数据处理系统,其中,响应于在其中高速缓存未命中的数据被写入到所述高速缓存的数据单元的高速缓存未命中:

响应于指示写入到所述数据单元的数据不需要写通的写通需求属性,进行从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换;

响应于指示写入到所述高速缓存的所述数据单元的数据需要写通的写通需求属性,进行从所述高速缓存的数据单元的所述无效状态到所述未修改的一致状态的转换。

7. 如权利要求6所述的数据处理系统,其中:

所述高速缓存未命中是写入未命中;

如果所述写通需求属性指示所述写入未命中的数据需要写通,则所述写入未命中的数据不在后续的回写总线事务中被写入到存储器。

8. 如权利要求1所述的数据处理系统,其中,所述高速缓存一致性状态进一步包括:

修改的非一致状态,所述修改的非一致状态指示修改了所述高速缓存的数据单元中的数据,并且不保证所述高速缓存的数据单元中的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的。

9. 如权利要求 8 所述的数据处理系统,其中,响应于所述数据单元的数据的写入命中,进行从所述高速缓存的数据单元的所述未修改的非一致状态到所述修改的非一致状态的转换。

10. 如权利要求 9 所述的数据处理系统,其中:

当处于所述未修改的一致状态时,所述高速缓存的数据单元的数据的写入命中产生到所述存储器的数据的写通;

当处于所述未修改的非一致状态时,所述高速缓存的数据单元的数据的写入命中不产生数据的写通操作。

11. 如权利要求 8 所述的数据处理系统,其中,响应于不带有对于所述数据单元的无效高速缓存操作命令的刷新高速缓存操作命令以及写通需求属性指示所述数据单元的数据不需要写通,进行从所述高速缓存的数据单元的所述修改的非一致状态到所述未修改的非一致状态的转换。

12. 如权利要求 8 所述的数据处理系统,其中,响应于所述数据单元的刷新高速缓存操作命令以及数据的写通需求属性指示所述数据需要写通,进行从所述高速缓存的数据单元的所述修改的非一致状态到所述未修改的一致状态的转换。

13. 如权利要求 8 所述的数据处理系统,其中,当数据单元的数据的写通需求属性指示所述数据不需要写通时,从所述高速缓存的所述数据单元的所述修改的非一致状态到所述无效状态的转换响应于下述命令而进行:

不带有对于所述数据单元的无效高速缓存操作命令的对所述数据单元的刷新高速缓存操作命令,其中,响应于所述刷新高速缓存操作命令,通过回写总线事务将所述数据写入存储器;或者

对于所述数据单元的无效高速缓存操作命令。

14. 如权利要求 1 所述的数据处理系统,其中,当写通需求属性指示所述高速缓存的数据单元的数据需要写通时,响应于不带有对于所述数据单元的无效高速缓存操作命令的对于所述数据单元的刷新高速缓存操作命令,进行从所述高速缓存的所述数据单元的所述未修改的非一致状态到所述未修改的一致状态的转换。

15. 如权利要求 1 所述的数据处理系统,其中,当写通需求属性指示所述数据不需要写通时,响应于不带有对于所述数据单元的无效高速缓存操作命令的对于所述数据单元的刷新高速缓存操作命令,进行从所述高速缓存的数据单元的所述未修改的一致状态到所述未修改的非一致状态的转换。

16. 如权利要求 1 所述的数据处理系统,其中,响应于对于所述数据单元的无效高速缓存操作命令,进行从所述高速缓存的数据单元的所述未修改的非一致状态和所述未修改的一致状态中的每一个到所述无效状态的转换。

17. 一种操作数据处理系统的方法,所述方法包括:

由第一主设备执行数据的写入事务,所述写入事务产生对所述数据处理系统的高速缓存的写入未命中,所述第一主设备可操作地耦接到系统互连,所述数据处理系统包括第二

主设备和可操作地耦接到所述系统互连的存储器；

响应于所述写入未命中，向所述高速缓存写入数据；

如果所述写入事务的所述数据的写通需求属性指示需要所述数据的写通，则执行到所述存储器的所述数据的写通，其中，如果所述写通需求属性指示不需要所述数据的写通，则不执行所述数据的写通；

如果所述写通需求属性指示需要所述数据的写通，则在所述数据处理系统的至少一个其他高速缓存中进行搜索，以确定所述至少一个其他高速缓存是否包含与所述写入事务相对应的存储器地址，其中，如果所述写通需求属性指示不需要所述数据的写通，则不在所述至少一个其他高速缓存中执行搜索。

数据处理系统中的高速缓存一致性协议

技术领域

[0001] 本公开大体上涉及数据处理系统,并且更具体地涉及数据处理系统中的高速缓存一致性协议。

背景技术

[0002] 高速缓存一致性协议通常用于多处理器系统中。一个这样的协议是 MESI 协议。然而,为了实现 MESI 协议,多处理器系统的总线协议需要支持特定的总线事务 (transaction) 类型。此外,对于低端系统来说 MESI 协议可能是太复杂或太昂贵。

附图说明

[0003] 通过示例方式来描述本发明,并且本发明不限于所附附图,其中相同的附图标记表示相同的元素。出于简单明了表示图中元素的目的,图中元素并不一定按比例进行绘制。

[0004] 图 1 描述了依照本发明一个实施例的数据处理系统。

[0005] 图 2 依照本发明一个实施例,描述了图 1 中数据处理系统的处理器的一部分。

[0006] 图 3 依照本发明一个实施例,描述了由图 2 的处理器执行的高速缓存一致性协议的状态图。

[0007] 图 4 依照一种总线协议,描述了提供 HBURST 总线信号编码的表。

[0008] 图 5 依照一种总线协议,描述了提供 HPROT 总线信号编码的表。

[0009] 图 6-9 以流程图的形式,描述了用于确定是否执行高速缓存一致性操作 (例如侦听 (snooping)) 的不同实施例。

具体实施方式

[0010] 在一个实施例中,高速缓存一致性协议可以将分离的转换和状态用于一致性和非一致性写入。在一个实施例中,该高速缓存一致性协议使用由存储器管理单元 (MMU) 提供的写通 (write-through) 页属性作为用于限定一致性状态之间的一个或多个转换的控制值。在一个实施例中,该高速缓存一致性协议依赖于用于一致性业务 (coherent traffic) 的写通操作,并且支持用于非一致性数据的回写事务。进一步,该高速缓存一致性协议可以和当前不支持高速缓存一致性的标准总线协议一起使用。并且,在一个实施例中,可以选择高速缓存一致性所需要的总线事务的子集。也就是,基于一个或多个因素,一个特定的总线事务可能会或可能不会导致高速缓存一致性操作 (例如,可能会或可能不会导致产生侦听请求)。并且,在一个实施例中,该用于执行选择性侦听的一个或多个因素存在于现有的工业标准总线协议中,从而使得不需要额外的信令机制或对于标准总线协议的其他修改。

[0011] 就像这里使用的,术语“总线”用于表示可以用于传输诸如数据、地址、控制、或状态之类的一个或多个不同类型信息的多个信号或导体。这里讨论的导体可以被示例或者描述为一个单独的导体、多个导体、单向性导体或者双向性导体。然而,不同的实施例中可以改变导体的实施方式。例如,可以使用分离的单向性导体而不是双向性导体,反之亦然。并

且,多个导体可以由串行地或者以时间复用的方式传输多个信号的单导体代替。同样的,承载多个信号的单导体也可以被分成承载这些信号的子集的多个不同的导体。因此,对于传输信号而言存在很多选择。

[0012] 当提到使一个信号、状态位、或类似装置成为其逻辑真或逻辑假状态时,在这里分别使用术语“断言”或“设置”和“无效”(或者“取消断言”或“清除”)。如果逻辑真状态是逻辑电平 1,则逻辑假状态就是逻辑电平 0。并且如果逻辑真状态是逻辑电平 0,则逻辑假状态就是逻辑电平 1。

[0013] 这里描述的每个信号可以被设计成正逻辑或负逻辑,其中,负逻辑可以用信号名称上面的一个小横杠或者名称后面的星号(*)来表示。在负逻辑情况下,信号低电平有效,其中,逻辑真状态对应于逻辑电平 0。在正逻辑情况下,信号高电平有效,其中,逻辑真状态对应于逻辑电平 1。注意,本文描述的任何信号都能够以正逻辑或负逻辑信号二者之一来设计,因此,在可替换的实施例中,那些以正逻辑信号方式描述的信号可以以负逻辑信号的方式实现,那些以负逻辑信号方式描述的信号可以以正逻辑信号的方式实现。

[0014] 这里使用方括号表示总线的导体或者值的位的位置。例如,“总线 60[7:0]”或者“总线 60 的导体 [7:0]”表示总线 60 的 8 个低阶导体,并且“地址位 [7:0]”或者“地址 [7:0]”表示地址值的 8 个低阶位。置于数字之前的符号“\$”表示该数以十六进制或基数十六的形式表示。置于数字之前的符号“%”表示该数以二进制或基数二的形式表示。

[0015] 图 1 描述了依照本发明的一个实施例的数据处理系统 10 的方框图。系统 10 可以是片上系统。系统 10 可以在一个单独的集成电路或多个集成电路上实现。系统 10 包括系统互连 12,其例如可以是系统总线。在一个实施例中,该系统互连 12 是依据 AMBA AHB 或 AXI 协议(AMBA、AXI、和 AHB 是 ARM 公司的商标)运行的系统总线。系统 10 包括:任意数量的处理器,例如处理器 14 和 16;系统事务仲裁器 32;高速缓存一致性管理器 18;存储器 20;2 级高速缓存 41;其他主设备 22;其他从设备 24;输入/输出(I/O)模块 26;和直接存储器存取(DMA)模块 40,每一个均双向耦接到系统互连 12。处理器 14 包括高速缓存 28,处理器 16 包括高速缓存 30。高速缓存一致性管理器 18 包括一致性主设备控制寄存器 19。高速缓存一致性管理器 18 也可以通过独立于系统互连 12 的侦听控制接口而双向耦接到 L2 级高速缓存 41、处理器 14、和处理器 16 中的每一个。另外,高速缓存一致性管理器 18 可以双向耦接到系统 10 中的其他主设备。系统 10 的替代实施例可以包括图 1 中没有描述的额外的电路,或者可以包括比所描述出的电路还少的电路。在某些实施例中,高速缓存一致性管理器 18 可以分布在处理器 14 和 16 之内或者不存在。

[0016] 图 2 是与图 1 中的数据处理系统 10 相关的处理器 14 的框图。处理器 14 可以实施为以流水线形式执行操作,并且可以包括指令通道(instruction pipe)23、执行单元 25、指令获取单元 27、控制电路 17、通用寄存器 31、加载/存储单元 33、总线接口单元(BIU)48、存储器管理单元(MMU)42、和高速缓存 28。在一个实施例中,控制电路 17 双向耦接到 BIU 48、高速缓存 28、MMU 42、指令获取单元 27、指令通道 23、执行单元 25、和加载/存储单元 33 中的每一个。指令通道 23 双向耦接到指令获取单元 27、执行单元 25 和加载/存储单元 33。执行单元 25 双向耦接至 MMU 42、高速缓存 28 和通用寄存器 31 中的每一个。加载/存储单元双向耦接到通用寄存器 31、MMU 42、和高速缓存 28。处理器 14 可以通过耦接到 BIU 48 的系统互连 12 而与数据处理系统 10 的其他部件通信。高速缓存 28 包括控制电路

29, 在一个实施例中, 其至少部分地基于由 MMU 42 提供的存储器页属性而为高速缓存 28 实现高速缓存一致性协议。高速缓存 28 包括多个高速缓存条目 (entry) (也被称为高速缓存行), 每个高速缓存条目包括地址标签、对应数据、和状态信息, 其包括了涉及当前高速缓存一致性协议状态的信息。注意, 除了控制电路 29 和任何用于实现高速缓存一致性协议的电路之外, 处理器 14 的运行是本领域技术人员熟知的, 因此此处将不会进一步详细讨论。

[0017] 如本领域中已知的那样, MMU 42 将虚拟地址 (其可以是从, 比如执行单元 25、加载 / 存储单元 33、和指令获取单元 27 接收到的) 转换为物理地址。MMU 42 将转换过的地址提供到高速缓存 28。即, MMU 42 提供与该物理地址相对应的存储器的页的物理地址和页属性。例如, 高速缓存 28 中的条目可以和存储器 20 的存储位置相对应, 其中, 存储器 20 包括称为页的存储单元。因此, MMU 42 提供的页属性和物理地址转换信息可以编程到 MMU 42 的存储电路 (或者处理器 14 中的其他地方) 中, 并且对应于包含了提供的物理地址的特定页。在一个实施例中, 这些页属性包括写通需求属性 (W) 和如下属性之一: 高速缓存禁止 (I), 存储器一致性需求 (M) (其也可以被称为全局共享 (SG))、受防护 (G)、字节序 (Endianness) (E)、和用户可定义 (例如 U0, U1, U2, U3) 属性。例如, 在一个实施例中, MMU 42 包括转译旁观缓冲区 (TLB), TLB 的条目存储了地址转换和以上这些属性, 从而控制处理器 14 在每个 TLB 条目相关的页中进行存储访问的方式。在一个实施例中, 每一页的页属性包括 W 和 M 两个, 每个 W 属性和 M 属性可以互相独立地设置。

[0018] 典型地, 为了允许多主设备系统中的存储器一致性而支持高速缓存一致性。可通过使用高速缓存一致性操作来保持一致性。在一个实施例中, 这些高速缓存一致性操作包括提供到诸如处理器 14 的处理器器的侦听请求 (即, 侦听无效请求)。在一个实施例中, 这些侦听请求由高速缓存一致性管理器 18 提供到处理器 14 和处理器 16 中的每一个。这些侦听无效请求由处理器接收, 以便处理器能够确定与侦听无效请求相对应的地址是否存储在高速缓存中。因此, 高速缓存一致性操作可以是指侦听查找, 其中, 在高速缓存中执行查找以确定高速缓存中是否包含与侦听事务相关的数据。即, 高速缓存一致性操作可以包括搜索该数据处理系统 (例如, 数据处理系统 10) 的至少一个高速缓存, 以确定至少一个高速缓存是否包含与侦听总线事务的存储器地址相关的数据。接着, 如果高速缓存包含与侦听事务相关的数据 (包括同与侦听请求相对应的存储器地址匹配的条目), 那么该处理器能够为该条目执行合适的一致性操作或者覆写该条目。在一个实施例中, 该高速缓存一致性协议中使用侦听无效一致性操作, 并且由于由另一总线主设备执行的侦听事务很可能正在修改存储在那个地址的数据, 所以该匹配的条目被无效。替代地, 注意高速缓存一致性操作能够涉及高速缓存条目的状态变化。

[0019] 图 3 描述了高速缓存一致性协议 100 具有四个状态: 修改的非一致状态 (M)、无效状态 (I)、未修改的非一致状态 (N)、和未修改的一致性状态 (C)。在这个四状态协议中, 单独的转换和状态用于一致性或非一致性的写入。在一个实施例中, 高速缓存一致性协议 100 使用 MMU 42 提供的 W 页属性作为限定一致性状态之间的一个或多个转换的控制值。在一个实施例中, 该高速缓存一致性协议依赖于用于一致性业务的写通操作, 并且支持用于非一致性数据的回写事务。例如, 正像下面将详细描述, 一致性写入使用 I 和 C 状态, 并且以写通式写入的方式来执行。非一致性写入使用 I、N 和 M 状态, 并且以回写式写入的方式来执行。

[0020] 将参照处理器 14 来描述图 3 的状态图。即,在一个实施例中,高速缓存 28 中的控制电路 29 执行图 3 中的状态图。高速缓存 28 中的每个高速缓存条目包括一个或多个状态指示符,用于指示高速缓存条目的状态 (M、I、N 或 C) (注意,高速缓存条目也可以称作高速缓存行)。如图 3 中所描述的,状态 102 对应于 I 状态,状态 104 对应于 N 状态,状态 108 对应于 M 状态,并且状态 106 对应于 C 状态。如关于高速缓存一致性协议 100 中所使用的,当高速缓存条目是“未修改”时,处理器 14 的高速缓存 28 中的高速缓存条目的数据自其被分配开始相对于存储器 20 中的副本没有被修改。当高速缓存条目是“修改的”时,处理器 14 的高速缓存 28 中的高速缓存条目的数据自其被分配开始已经相对于存储器 20 中的副本而被修改。当高速缓存条目是“一致的”,对应于相同的地址位置,高速缓存 28 中的高速缓存条目与存储器 20 中的,以及与系统 10 的任何高速缓存 (例如高速缓存 30) 的任何有效条目中的数据是相同的。“不一致”是指与其他高速缓存是不一致的。即,当高速缓存条目“不一致”时,对应于存储器 20 中同一地址位置,高速缓存 28 的高速缓存条目的数据可能具有与诸如高速缓存 30 之类的其他高速缓存的有效条目不同的值。注意,由于“不一致”是指与其他高速缓存不一致,因此当高速缓存条目“不一致”时,高速缓存条目的数据和存储器 20 中的可能是相同的。当对于存储器中具体地址位置没有执行高速缓存一致性操作时,可能出现非一致性。

[0021] 一致的高速缓存条目可能处于 C 状态或者 I 状态。不需要一致性的高速缓存条目可以假设为 I、N、或 M 状态。注意到,对于一致的高速缓存条目不支持修改或共享状态 (就像 MESI 协议中所使用的一样),因而不需要侦听回写或介入操作。不一致的高速缓存条目可能仍然假定为修改的状态。在一个实施例中,需要一致性操作的存储器区域 (例如在存储器 20 中) 被标记为“存储器一致性需求” (其 M 位被设置为 1) 和“写通需求” (其 W 位被设置为 1)。注意,协议 100 可以与不包含或支持高速缓存一致性操作的总线协议,比如例如 AMBA AHB 或 AXI 协议一起使用。对于不支持高速缓存一致性的总线协议,无法实现 MESI 协议。例如,无法发信号来表示 MESI 协议的从 S 状态到 M 状态的转换,其中, MESI 协议使用 AMBA 和 AHB 协议中不存在的“只涉及地址的事务 (address only transaction)”。此外,不支持 MESI 高速缓存一致性事务的该总线协议不支持 MESI 协议中用于实现从 I 状态到 E 状态转换的“读且试图修改”事务。AMBA AHB 或 AXI 总线协议没有定义需要这种协议的其他事务类型。因此,在一个实施例中,注意高速缓存一致性协议 100 可以覆盖当前不支持高速缓存一致性的现有总线协议。

[0022] 参考图 3,当高速缓存条目处于无效状态,并且出现导致对无效的高速缓存条目的分配的高速缓存未命中 (miss) 时,使用 W 页属性来确定转换到状态 N (非一致状态) 还是状态 C (一致状态)。当出现高速缓存未命中,并且 $W = 0$,指示对于该地址不执行写通时,高速缓存条目的状态通过转换 110 从状态 I 转换到状态 N。由于没有对该地址执行写通,所以不能保证高速缓存条目的数据是一致的。由于没有执行写通,向该高速缓存条目的写入事务不在系统互连 12 上进行,因而该事务就无法被高速缓存一致性管理器 19 (或其他高速缓存) 侦听。因此,由于不会执行一致性操作,所以认为是不一致的。由于高速缓存条目是新分配的,所以其数据被认为是未修改的。

[0023] 然而,当发生高速缓存未命中,并且 $W = 1$,指示执行写通以向该地址写入时,则高速缓存条目的状态通过转换 128 从状态 I 转换到状态 C。由于执行写通以向该地址写入,因

此保持了一致性。例如,对于写入未命中进行的写通,在转换 128 之后重新分配的高速缓存条目中存储的数据也经由系统互连 12 提供到存储器 20。对于读取未命中进行的写通,要存入新分配的高速缓存条目中的数据由存储器 20 经由系统互连 12 提供到处理器 14。因此,在执行写通的过程中,新分配的高速缓存条目中存储的数据经由高速缓存一致性管理器 18 监控的系统互连 12 来提供。因此高速缓存一致性管理器 18 能够通过侦听与写通事务相对应的系统互连 12 上的地址,从而执行高速缓存一致性操作,并且执行高速缓存一致性操作。例如,高速缓存一致性管理器 18 能够向处理器 16 提供侦听无效请求,从而使处理器 16 能够确定该侦听地址是否在高速缓存 30 内命中 (hit),如果命中,则将其无效。

[0024] 一旦处于状态 C,对高速缓存条目的任何读取或写入命中将致使该高速缓存条目保持在状态 C 中。在写入命中的情况下,执行写通 (由于用于该地址的 W 页属性被设置) 以便允许保持一致性。即,在写通之后,高速缓存一致性操作能够得以执行。例如,正像上面所描述的,高速缓存一致性管理器 18 能向处理器 16 发出侦听无效请求,以便保持一致性。对高速缓存条目的高速缓存无效操作经由转换 130 将高速缓存条目返回状态 I。还要注意,在状态 C 中,当与高速缓存条目的地址相关的数据被诸如处理器 16 的另一个处理器修改时,处理器 14 可能从高速缓存一致性管理器 18 接收高速缓存一致性操作,例如,侦听无效请求。在这种情况下,侦听无效请求会命中处理器 14 的高速缓存条目,并致使高速缓存条目的状态经由转换 129 转换回到状态 I。

[0025] 也可以由处理器 14 或 16 对存储器 20 中的位置执行高速缓存刷新 (flush) 操作 (其也可以被称为刷新高速缓存操作命令),用于分别采用高速缓存 28 或 30 中现有的任何修改数据来更新存储器 20。在一个实施例中,这些高速缓存刷新操作可以被指定用于将任何修改的数据刷新至存储器 20,然后将高速缓存条目置于有效状态 (即,不带有无效的高速缓存刷新操作),或者可替代地,将任何修改的数据刷新至存储器 20,并且执行刷新之后,将高速缓存条目置于无效状态 (即,带有无效的高速缓存刷新操作)。高速缓存刷新操作提供地址用于在高速缓存中查找,而不用事先知道与高速缓存中存在的地址相对应的数据的存在,或者事先知道包含了数据的高速缓存条目的状态。因此,在一个实施例中,刷新操作是如下的操作,其中,如果高速缓存条目具有任何修改的数据,则匹配的高速缓存条目被写回到存储器。即,如果高速缓存行或条目是脏的,那么它将被写回到存储器 (例如存储器 20)。如果没有修改的数据,则不需要写回到存储器。也要注意,可以带有或不带有无效地执行刷新。

[0026] 仍然参考状态 C,执行不带有无效的高速缓存刷新操作之后,再一次检查 W 页属性以确定如何在状态之间转换。如果,在不带有无效的高速缓存刷新操作之后,W 页属性目前被清零 (例如,作为对用于存储器 20 中对应页的 W 页属性进行修改的软件的结果),则高速缓存条目的状态经由转换 134 转换到状态 N。即,高速缓存条目没有被无效 (由于执行的刷新不带有无效),但是现在,也不再执行写通。这样,又一次无法保证一致性。另外,由于请求刷新,因此知道在完成刷新操作时数据未被修改。如果执行带有无效的刷新,则执行刷新操作之后,高速缓存条目被无效,因此转换回状态 I (经由转换 130)。如果执行不带有无效的刷新,在刷新操作之后高速缓存条目保持有效,并且基于 W 页属性确定状态 (是否保持状态 C 或者转换到状态 N)。

[0027] 一旦在状态 N,高速缓存条目的任何读取命中都将致使高速缓存条目保持在状态

N。对高速缓存条目的高速缓存无效操作将高速缓存条目经由转换 112 返回到状态 I。取决于 W 页属性的值,不带有无效的高速缓存刷新操作可能引起转换到状态 C。例如,如果执行不带有无效的刷新操作,并且 W 页属性的值仍然是零,则高速缓存条目的状态保持为状态 N。如果执行不带有无效的高速缓存刷新操作,并且 W 页属性目前是 1 (例如,其通过软件而被改变),则高速缓存条目的状态由状态 N 转换到状态 C,其中,由于该刷新,在该转换中数据保持未被修改,并且由于目前执行了写通,从而保持了一致性。如果执行带有无效的刷新,则在刷新操作之后,高速缓存条目的状态返回到 I (经由转换 112)。在处于状态 N 的高速缓存条目写入命中之后,高速缓存条目的状态经由转换 114 从状态 N 转换到状态 M。因此,由于没有执行写通,所以高速缓存条目仍然是不一致的;然而,由于其在从状态 I 到状态 N 的转换 110 之后被分配因而被修改,所以认为其现在经过了修改。

[0028] 一旦处于状态 M 中,对高速缓存条目的任何命中都将致使高速缓存条目保持在状态 M。如果执行了不带有无效的高速缓存刷新操作,则像上面所描述的,执行向存储器 (例如存储器 20) 的回写,并且不无效高速缓存条目。然后,基于 W 页属性,高速缓存条目从状态 M 转换到状态 N 或 C。在 $W = 0$ 的情况下,高速缓存条目的状态经由转换 116 返回到状态 N,其中,目前数据再一次未被修改 (由于刷新操作),并且未执行写通 (因此不保证一致性)。在 $W = 1$ 的情况下,高速缓存条目的状态经由转换 126 转换回到状态 C,其中,目前数据再一次未被修改 (由于刷新操作),并且执行写通 (因此保持一致性)。在状态 M,如果执行带有无效的刷新,则执行向存储器的回写,正如上面所描述的,并且无效高速缓存条目,经由转换 124 将高速缓存条目的状态返回到状态 I。如果对处于状态 M 的高速缓存条目执行替换操作,修改数据首先被回写到存储器 20。然后高速缓存条目中的数据被首先无效 (经由转换 124 转换到状态 I),然后被覆盖写入 (由于实际上发生了分配,因此取决于 W 页属性的值,导致转换到状态 N 或 C)。并且,如果执行了无效操作,高速缓存条目的状态经由转换 122 从状态 M 转换到状态 I,并且不执行回写操作。

[0029] 因此,可以理解,高速缓存一致性协议如何将分离的转换和状态用于一致性和非一致性写入。在一个实施例中,高速缓存一致性协议 100 使用 MMU 42 提供的 W 页属性作为用于限定一致性状态之间的一个或多个转换的控制值。在一个实施例中,高速缓存一致性协议依赖于用于一致性业务的写通操作,并且支持用于非一致数据的回写事务。因此,高速缓存一致性协议 100 可以和不提供高速缓存一致性操作的标准总线协议一起使用。

[0030] 在一个实施例中,高速缓存一致性管理器 18 可以选择高速缓存一致性需要的总线事务的子集。即,基于一个或多个因素,正如下面将要参考图 4-9 讨论的,特定的事务类型可能会或可能不会被侦听,从而将会或不会导致高速缓存一致性操作。此外,在一个实施例中,用于执行选择性侦听的一个或多个因素存在于现有的工业标准总线协议,例如 AMBA AHB 或 AXI 中。这样,就不需要额外的信令机制或对于标准总线协议的其他修改。

[0031] 在一个实施例中,高速缓存一致性管理器 18 基于总线事务是单拍事务还是突发事务来进行侦听和执行高速缓存一致性操作。例如,在一个实施例中,如果总线事务是单拍事务而不是突发事务,则执行侦听和高速缓存一致性操作。这可能是由于处理器发起的突发通常进行的是回写事务,因此这些突发事务不需要高速缓存一致性操作。然而,在一个实施例中,如果事务是突发事务,但是由 DMA 40 而非处理器 (例如处理器 14 或 16) 发起,仍然执行侦听,这是因为通常由 DMA 发起的突发事务不执行回写,意味着仍然需要高速缓存

一致性操作。因此,在一个实施例中,是否侦听的判定可以基于总线事务的类型和发起总线事务的主设备或主设备的类型。正如下面将描述地,系统 10 中可以例如由高速缓存一致性管理器 18 或其他高速缓存一致性控制电路使用其他因素或因素组合,以确定是否要执行侦听。注意,在一个实施例中,为每一个总线事务提供主设备标识符(主设备 ID),用于识别出发起事务的主设备。主权(mastership)的确定可以通过发起主设备指示作为传输地址属性作为总线事务的一部分,或者可能以其他方式指示或信号通知。在一个实施例中,如果突发写入访问被发起,执行一致性操作的主设备的确定可以由系统 10 的用户以编程方式选择。在一个实施例中,可以在系统 10 内提供控制寄存器,以基于传输类型(突发或非突发,或在一组特定突发类型中选择的类型)指示应该执行一致性操作的一个或多个主设备。

[0032] 使用 AMBA AHB 协议作为例子,使用编码出 8 个可能的事务类型的 3 位总线信号 HBURST[2:0] 来提供突发信息。图 4 提供了这些编码。例如,HBURST 的值“000”表示单个传输(即,单拍传输),而其他 7 个值表示不同的突发事务类型。在一个实施例中,当高速缓存一致性管理器 18 确定 HBURST 的值为 000(表示单个传输),将执行侦听;然而,如果确定 HBURST 的值为 010(表示 4 拍环回突发(wrappingburst)),011(表示 4 拍递增突发),100(表示 8 拍环回突发),101(表示 8 拍递增突发),110(表示 16 拍环回突发)或者 111(表示 16 拍递增突发),则不执行侦听。依据实施例,当 HBURST 的值为 001(表示未定义长度的递增突发),可能会也可能不会执行侦听。正如上面所讨论的,是否侦听的判定可能进一步基于发起总线事务或传输的主设备或主设备类型。

[0033] 在一个实施例中,是否侦听的判定可能进一步取决于当前的访问(当前的总线事务)是否是可高速缓存的。仍然使用 AMBA AHB 协议为例,HPROT[3:0] 信号被提供了总线事务访问地址和指示,例如,传输是操作码获取还是数据访问、权限模式访问还是用户模式访问、以及对于带有 MMU 的主设备,指示当前访问是否是可高速缓存的。图 5 提供了 HPROT[3:0] 信号的编码方式。例如,当 HPROT[3] 具有值 0 时,由于总线事务的访问不可高速缓存所以不执行侦听,而当 HPROT[3] 具有值 1 时,由于总线事务的访问是可高速缓存的所以执行侦听。因此,在使用上述描述的因素之外或者代替上述描述的因素,HPROT 信号还可以用于确定是否会执行侦听。在另一个实施例中,当 HPROT[2] 具有值 0 时,由于总线事务的访问是不可缓冲的所以不执行侦听,而当 HPROT[2] 具有值 1 时,由于总线事务的访问是可缓冲的所以执行侦听。在一些实施例中,依据发起总线访问的特定主设备,HPROT 位的值可以以编程的方式用于控制是否执行一致性操作。因此,在使用上述描述的因素之外或者代替上述描述的因素,HPROT 信号还可以用于确定是否会执行侦听,可选地还可确定发起事务的主设备的身份。例如,可能的是,以实现当 HPROT[3] = 1 时为主设备 X 执行的一致性操作,当 HPROT[2] = 1 时为主设备 Y 执行的一致性操作,以及当 HPROT[3] = 1 并且 HPROT[2] = 0 时为主设备 Z 执行的一致性操作的方式,系统 10 的用户对一个或多个控制寄存器或寄存器字段进行编程。可替代地,也可以使用其他可实现的控制组合。

[0034] 图 6-9 描述了确定是否执行高速缓存一致性操作(例如侦听)的流程图的各种示例。正如上面讨论的,如果在系统 10 中存在其他高速缓存的话,高速缓存一致性操作可以包括对在系统 10 中的其他高速缓存产生侦听请求,在系统 10 中,那些高速缓存然后处理该侦听请求。在一个实施例中,高速缓存一致性操作可以包括高速缓存条目状态的改变。在

一个实施例中,每个流程可以由高速缓存管理器 18 执行。或者,其他高速缓存一致性控制电路可以执行这些功能的任意一个,并且存在于系统 10 的任意位置。在一个实施例中,每个单个的高速缓存或者高速缓存控制电路可以执行图 6-9 中描述的功能。图 6 描述了起始于框 502 的流程 500,在框 502 中,由总线主设备,例如处理器 14、处理器 16、DMA40 或其他主设备 22 发起总线事务。流程继续到确定是否执行写入事务的判定菱形框 504。如果否,流程继续到不执行高速缓存一致性操作的框 514。即,在图 6 描述的实施例中,不侦听读取事务。如果在判定菱形框 504 中,确定总线事务是写入事务,流程继续到确定总线事务是否是单拍(及,单个传输)事务的判定菱形框 506。如果否,则流程继续到框 514。如果总线事务是单拍事务,流程继续到对写入事务的数据执行高速缓存一致性操作的框 512。因此,在流程 500 中,只有总线事务是单拍写入事务才执行高速缓存一致性操作。

[0035] 图 7 描述了起始于框 602 的流程 600,在框 602 中,由总线主设备,诸如例如处理器 14、处理器 16、DMA40 或其他主设备 22 发起总线事务。流程继续到确定是否执行写入事务的判定菱形框 604。如果否,流程继续到不执行高速缓存一致性操作的框 614。即,在图 7 描述的实施例中,不侦听读取事务。如果在判定菱形框 604 中确定总线事务是写入事务,流程继续到确定总线事务是否是单拍(即,单个传输)事务的判定菱形框 606。如果是,流程继续到对写入事务的数据执行高速缓存一致性操作的框 612。如果否,流程继续到确定事务主设备 ID 的框 608。即,正如上面所描述的,在一个实施例中,为每一个总线事务提供主设备 ID 用于识别该发起主设备。然后流程继续到确定主设备 ID 是否指示诸如 DMA 40 之类的 DMA 的判定菱形框 610。如果是,流程继续到对写入事务的数据执行高速缓存一致性操作的框 612。如果否,流程继续到框 614。即,即使写入事务是突发写入(即,判定菱形框 606 的“否”分支),如果发起主设备是 DMA 则仍然执行高速缓存一致性操作。然而,如果写入事务是突发写入并且发起主设备不是 DMA,则不执行高速缓存一致性操作,因为在这种情况下,该突发事务很可能是回写事务。

[0036] 图 8 描述了起始于框 702 的流程 700,在框 702 中,由总线主设备,诸如例如处理器 14、处理器 16、DMA40 或其他主设备 22 发起总线事务。流程继续到确定是否执行写入事务的判定菱形框 704。如果否,流程继续到不执行高速缓存一致性操作的框 714。即,在图 8 描述的实施例中,不侦听读取事务。如果在判定菱形框 704 确定总线事务是写入事务,流程继续到确定总线事务是否是单拍(即,单个传输)事务的判定菱形框 706。如果是,流程继续到对写入事务的数据执行高速缓存一致性操作的框 712。如果否,流程继续到确定事务主设备 ID 的框 708。即,正如上面所描述的,在一个实施例中,为每一个总线事务提供主设备 ID 用于识别该发起主设备。然后流程继续到确定主设备 ID 是否指示选择的主设备的判定菱形框 710。如果是,流程继续到对写入事务的数据执行高速缓存一致性操作的框 712。如果否,流程继续到框 714。在一个实施例中,高速缓存一致性管理器 18 包括指示需要一致性的主设备的一致性主设备控制寄存器 19。例如,控制寄存器 19 可以利用系统 10 中的那些要求或需要高速缓存一致性的主设备的 ID 进行编程。因此,在图 8 的判定菱形框 710 中,在框 702 中发起总线事务的主设备的主设备 ID 可以与控制寄存器 19 比较,以确定发起主设备的主设备 ID 是否对应于控制寄存器 19 指示的选择的主设备。可替代地,其他电路或方法可以用来识别应该执行高速缓存一致性操作的发起主设备。因此,即使写入事务是突发写入(即,判定菱形框 706 的“否”分支),如果发起主设备是由一致性主设备控制寄存

器 19 选择的主设备,则仍然执行高速缓存一致性操作。在替代的实施例中,确定步骤 710 可以使用其他在系统 10 中预先确定的选择标准来执行。在一个实施例中,该选择标准可以由系统 10 的用户编程。

[0037] 图 9 描述了起始于框 802 的流程 800,在框 802 中,由总线主设备,诸如例如处理器 14、处理器 16、DMA40 或其他主设备 22 发起总线事务。流程继续到确定是否执行写入事务的判定菱形框 804。如果否,流程继续到不执行高速缓存一致性操作的框 814。即,在图 9 描述的实施例中,不侦听读取事务。如果在判定菱形框 804 中确定总线事务是写入事务,流程继续到读取总线事务的可高速缓存字段或信号的框 806。例如,这里可以参考前面描述过的 HPROT[3] 或 HPROT[2]。流程继续到确定可高速缓存字段或信号是否指示可高速缓存事务的判定菱形框 808。如果否,流程继续到框 814。如果是,流程继续到确定该写入事务是否是单拍写入的判定菱形框 810。如果否,流程继续到框 814。如果是,流程继续到为写入事务执行高速缓存一致性操作的框 812。即,在图 9 的实施例中,只对可高速缓存的单个写入事务执行高速缓存一致性操作。即,突发事务不会导致高速缓存一致性操作,如例如由依据系统互连 12 的总线协议提供的信号指示为非可高速缓存事务的任何事务不会导致高速缓存一致性操作一样。在替换的实施例中,判定步骤 808 接下来可以是以与图 8 中步骤 708 和 710 执行的确定的相似的方式识别发起该事务的主设备的确定步骤。在一个实施例中,该确定的结果可以作为决定是否总线事务执行高速缓存一致性操作的进一步的因素。

[0038] 因此,可以理解如何单独或相互之间进行各种不同的组合地使用各种不同的因素,用于确定是否执行高速缓存一致性操作(例如,侦听)。高速缓存一致性操作的该种可选择性能可以例如使用现有的标准总线协议执行。

[0039] 因为实现本发明的装置在很大程度上是由本领域技术人员熟知的电子元件和电路组成,为了理解和领会本发明的实质精神和不使本发明的教导被模糊或分散,除了上文所描述的认为必要的电路细节以外将不对其它更多的电路细节进行说明。

[0040] 尽管已经相对于具体的传导类型或电势极性来描述了本发明,技术人员明白该传导类型和电势极性是可以颠倒的。

[0041] 而且,在说明书和权利要求书中如果有术语“前”、“后”、“顶”、“底”、“上”、“下”等等,则它们是用于描述性目的,而对于描述不变的相对位置并不是必要的。可以理解,这些术语的使用在适当的情况下是可以相互交换的,以致于这里描述的本发明的实施例例如能够以此处说明或描述之外的其他方式运行。

[0042] 在此所使用的术语“程序”被定义为用于在计算机系统上执行而设计的指令序列。程序或计算机程序可以包括子程序、函数、过程、对象方法、对象实现、可执行的应用程序、小程序(applet)、服务小程序(servlet)、源代码、目标代码,共享库/动态装载库和/或用于在计算机系统上执行而设计的其它指令序列。

[0043] 上述某些实施例,在适当时可使用各种不同的信息处理系统来实现。例如,尽管图 1 及其讨论描述了示例性的信息处理体系结构,但是该示例性的体系结构只是为了在本发明的各个方面的讨论中提供有用的参考而示出。当然,为了讨论的目的已经简化了对该体系结构的描述,并且它只是可以根据本发明来使用的许多不同类型的适合的体系结构之一。本领域技术人员将认识到在逻辑块之间的边界仅仅是说明性的,并且可替换的实施方

案可以合并逻辑块或电路元件,或者对各种逻辑块或电路元件实行可替换的功能分解。

[0044] 从而,应当了解在此所描述的体系结构只是示例性的,并且实际上能够实现获得相同功能的许多其它体系结构。在抽象但仍然明确的意义,用于实现相同功能的组件的任何布局是有效“关联的”,使得所希望的功能得以实现。因此,在不考虑体系结构或中间的组件的情况下,此处组合以实现特定功能的任何两个组件都能够被看作是彼此“关联的”,使得能够实现所希望的功能。类似地,这样关联的任何两个组件还能够被看作是彼此“可操作地连接”或者“可操作地耦接”地,以实现所希望的功能。

[0045] 还例如,在一个实施例中,所描述的系统 10 的元件是位于单个集成电路上或者位于相同的器件之内的电路。可选地,系统 10 可以包括彼此互连的任何数量的分离的集成电路或分离的器件。例如,存储器 20 可以和处理器 14 和 16 一样位于同样的集成电路上,或者位于分离的集成电路上,或者在与系统 10 的其他元件分离的其他外围设备内或从设备之内。其他从设备 24 和 I/O 电路 26 亦可位于单独的集成电路或设备上。还例如,系统 10 或其部分可以是物理电路的软件或代码表示,或者是可转换成物理电路的逻辑表示。这样,可以用任何适合类型的硬件描述语言来实现系统 10。

[0046] 而且,本领域技术人员会认识到在上述操作的功能之间的边界只是说明性的。多个操作的功能可以结合到单个操作中,和 / 或单个操作的功能可以分布于附加的操作中。而且,可替换的实施例可以包括具体操作的多种实例,并且在各种其它实施例中可以改变操作的顺序。

[0047] 在此所描述的所有或某些软件可以是诸如存储器 35 的计算机可读媒体或其它计算机系统上的其它媒体所接收的系统 10 的元素。该计算机可读媒体可以是与例如系统 10 的信息处理系统持久地、可移动地、或远程地耦接。计算机可读媒体可以包括,例如但不限于,任何数量的下列媒体:磁存储媒体,包括磁盘和磁带存储媒体;光存储媒体,例如压缩磁盘媒体(如,CD-ROM、CD-R 等)和数字视频磁盘存储媒体;非易失性存储器存储媒体,包括基于半导体的存储器单元,例如 FLASH 存储器、EEPROM、EPROM、ROM;铁磁数字存储器;MRAM;易失性存储媒体,包括寄存器、缓冲器、或高速缓存、主存、RAM 等;以及数据传输媒体,包括计算机网络、点对点电信设备、以及载波传输媒体,以上仅列举出其中几个。

[0048] 在一个实施例中,系统 10 在诸如个人计算机系统的计算机系统中实现。其他实施例可包括不同类型的计算机系统。计算机系统是可被设计为向一个或多个用户提供独立计算能力的信息处理系统。计算机系统可具有许多形式,包括但不限于:大型机、小型机、服务器、工作站、个人计算机、笔记型计算机、个人数字助理、电子游戏机、汽车和其他嵌入式系统、蜂窝电话和多种其他无线设备。典型的计算机系统包括至少一个处理单元、关联的存储器和若干输入 / 输出 (I/O) 设备。

[0049] 计算机系统根据程序处理信息,并且经由 I/O 设备产生最终的输出信息。程序是一系列指令,例如特定应用程序和 / 或操作系统。计算机程序通常内部存储在计算机可读存储介质上,或者经由计算机可读传输存储介质发送至计算机系统。计算机进程通常包括执行(运行)程序或者程序的一部分、当前程序值和状态信息,以及操作系统用来管理进程执行的资源。父进程可以生成其他子进程用于帮助执行该父进程的整体功能。由于父进程特定生成执行该父进程整体功能的一部分的子进程,所以由于子进程(以及孙子进程等)执行的功能有时可以说明为由该父进程执行。

[0050] 尽管本发明在此参考具体的实施例来描述,但是在没有脱离权利要求书所阐明的本发明的范围的情况下能够进行各种修改和改变。因此,说明书和附图应当被看作是说明性的而非限制性的,并且所有这些修改旨在被包括在本发明的范围内。在此针对具体的实施例所描述的任何益处、优点、或问题的解决方案并不旨在被认为是任何或所有的权利要求的关键的、必需的、或本质的特征或元素。

[0051] 如此处使用的,术语“耦接”不旨在局限于直接耦接或机械耦接。

[0052] 而且,如此处使用的,术语“一”定义为一个或多于一个。另外,权利要求中诸如“至少一个”和“一个或更多个”的引入性短语的使用不应被认为暗示着由不定冠词“一”引领的另一个权利要求元素将包含该引入的权利要求元素的任意特定权利要求局限于仅包含一个该元素的发明,即使当相同的权利要求包括引入性短语“一个或更多个”或“至少一个”及诸如不定冠词“一”时也是如此。这对定冠词的使用同样适用。

[0053] 除非另有说明,否则此处使用诸如“第一”和“第二”的术语用于区分这些术语所描述的元素。因而,这些术语不必旨在指示这些元素的时间上的或其它的优先次序。

[0054] 附加文本:

[0055] 1. 一种数据处理系统(例如,10),包括:

[0056] 第一主设备(例如,14),所述第一主设备包括高速缓存(例如,28);

[0057] 第二主设备(例如,16);

[0058] 存储器(例如,20),所述存储器经由系统互连(例如,12)可操作地耦接到第一主设备和第二主设备;

[0059] 其中,所述高速缓存包括高速缓存控制器(例如,29),所述高速缓存控制器为所述高速缓存的数据单元(例如,高速缓存条目或高速缓存行)实现一组高速缓存一致性状态(例如,I, N, C, M),所述高速缓存一致性状态包括:

[0060] 无效状态(例如,I);

[0061] 未修改的非一致状态(例如,N),所述未修改的非一致状态指示没有修改所述高速缓存的数据单元的数据,并且不保证所述高速缓存的数据单元的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的;以及

[0062] 未修改的一致状态(例如,C),所述未修改的一致状态指示没有修改所述数据单元的数据,并且所述数据单元的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的。

[0063] 2. 如项目1所述的数据处理系统,其中,响应于所述高速缓存的读取未命中,进行从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换,其中,所述读取未命中的数据被写入到所述数据单元,其中,响应于所述读取未命中,不对所述数据处理系统的任何其他高速缓存上执行高速缓存一致性操作。

[0064] 3. 如项目1所述的数据处理系统,其中:响应于写入高速缓存未命中,其中写入未命中的数据被写入的从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换不产生在所述数据处理系统的其他高速缓存中的高速缓存一致性操作。

[0065] 4. 如项目1所述的数据处理系统,其中,从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换不会引起所述数据处理系统的其他高速缓存内的查找操作,所述查找操作用于确定那些高速缓存是否包含对应于正被写入所述数据单元的

数据的存储器地址。

[0066] 5. 如项目 1 所述的数据处理系统,其中,当所述高速缓存的数据单元处于所述未修改的非一致状态时,指向那个数据单元的侦听事务不会导致那个数据单元的一致性状态的改变。

[0067] 6. 如项目 1 所述的数据处理系统,其中,响应于在其中高速缓存未命中的数据被写到所述高速缓存的数据单元中的所述高速缓存未命中:

[0068] 响应于指示写入到所述数据单元的数据不需要写通的写通需求属性,进行从所述高速缓存的数据单元的所述无效状态到所述未修改的非一致状态的转换(例如,转换 110);

[0069] 响应于指示写入到所述高速缓存的所述数据单元的数据需要写通的写通需求属性,进行从所述高速缓存的数据单元的所述无效状态到所述未修改的一致状态的转换(例如,转换 128)。

[0070] 7. 如项目 6 所述的数据处理系统,其中:

[0071] 所述高速缓存未命中是写入未命中;

[0072] 如果所述写通需求属性指示所述写入未命中的数据需要写通,则所述写入未命中的数据不在后续的回写总线事务中被写入到所述存储器。

[0073] 8. 如项目 1 所述的数据处理系统,其中,所述高速缓存一致性状态进一步包括:

[0074] 修改的非一致状态(例如,M),所述修改的非一致状态指示修改了所述高速缓存的数据单元的数据,并且不保证所述高速缓存的数据单元的数据与所述数据处理系统的至少一个其他存储设备中的数据是一致的。

[0075] 9. 如项目 8 所述的数据处理系统,其中,响应于所述数据单元的数据的写入命中,进行从所述高速缓存的数据单元的所述未修改的非一致状态到所述修改的非一致状态的转换(例如,转换 114)。

[0076] 10. 如项目 9 所述的数据处理系统,其中:

[0077] 当处于所述未修改的一致状态时,所述高速缓存的数据单元的数据的写入命中产生到所述存储器的数据写通(例如,转换 132);

[0078] 当处于所述未修改的非一致状态时,所述高速缓存的数据单元的数据的写入命中不产生数据的写通操作。

[0079] 11. 如项目 8 所述的数据处理系统,其中:响应于不带有对于所述数据单元的无效高速缓存操作命令的对于所述数据单元的刷新高速缓存操作命令以及所述写通需求属性指示所述数据单元的数据不需要写通,进行从所述高速缓存的数据单元的所述修改的非一致状态到所述未修改的非一致状态的转换(例如,转换 116)。

[0080] 12. 如项目 8 所述的数据处理系统,其中,响应于对于所述数据单元的刷新高速缓存操作命令以及数据的所述写通需求属性指示所述数据需要写通,进行从所述高速缓存的数据单元的所述修改的非一致状态到所述未修改的一致状态的转换(例如,转换 126)。

[0081] 13. 如项目 8 所述的数据处理系统,其中,当数据单元的数据的写通需求属性指示所述数据不需要写通时,从所述高速缓存的所述数据单元的所述修改的非一致状态到所述无效状态的转换响应于下述命令而进行:

[0082] 不带有对于所述数据单元的无效高速缓存操作命令的对于所述数据单元的刷新

高速缓存操作命令,其中,响应于所述刷新高速缓存操作命令,通过回写总线事务将所述数据写入存储器(例如,转换 124);或者

[0083] 对于所述数据单元的无效高速缓存操作命令(例如,转换 122)。

[0084] 14. 如项目 1 所述的数据处理系统,其中,当写通需求属性指示所述高速缓存的数据单元的数据需要写通时,响应于不带有对于所述数据单元的无效高速缓存操作命令的对于所述数据单元的刷新高速缓存操作命令,进行从所述高速缓存的所述数据单元的所述未修改的非一致状态到所述未修改的一致状态的转换(例如,转换 120)。

[0085] 15. 如项目 1 所述的数据处理系统,其中,当写通需求属性指示所述数据不需要写通时,响应于不带有对于所述数据单元的无效高速缓存操作命令的对于所述数据单元的刷新高速缓存操作命令,进行从所述高速缓存的数据单元的所述未修改的一致状态到所述未修改的非一致状态的转换(例如,转换 134)。

[0086] 16. 如项目 1 所述的数据处理系统,其中,响应于对于所述数据单元的无效高速缓存操作命令,进行从所述高速缓存的数据单元的所述未修改的非一致状态和所述未修改的一致状态的中的每一个到所述无效状态的转换(例如,转换 112 或 130)。

[0087] 17. 一种操作数据处理系统(例如,10)的方法,所述方法包括:

[0088] 由第一主设备(例如,16)执行数据的写入事务,所述写入事务产生对所述数据处理系统的高速缓存(例如,30)的写入未命中,所述第一主设备可操作地耦接到系统互连(例如,12),所述数据处理系统包括第二主设备(例如,14)和可操作地耦接到所述系统互连的存储器(例如,20);

[0089] 响应于所述写入未命中,向所述高速缓存写入数据;

[0090] 如果所述写入事务的所述数据的写通需求属性(例如,W)指示需要所述数据的写通,则执行到所述存储器的所述数据的写通,其中,如果所述写通需求属性指示不需要所述数据的写通,则不执行所述数据的写通;

[0091] 如果所述写通需求属性指示需要所述数据的写通,则在所述数据处理系统的至少一个其他高速缓存(例如,28)中进行搜索以确定所述至少一个其他高速缓存是否包含与所述写入事务相对应的存储器地址,其中,如果所述写通需求属性指示不需要所述数据的写通,则不在所述至少一个其他高速缓存中执行搜索。

[0092] 18. 一种数据处理系统(例如,10),包括:

[0093] 处理器(例如,14),所述处理器包括高速缓存(例如,28);

[0094] 第二主设备(例如,16);

[0095] 存储器(例如,20),所述存储器经由系统互连(例如,12)而可操作地耦接到所述处理器和所述第二主设备;

[0096] 存储器属性逻辑(例如,在 MMU 42 内,或者处理器 14 内的其他位置或者系统 10 内的其他位置),用于提供与所述存储器的位置相关的多个属性,其中所述多个属性之一是写通需求属性(例如,W);

[0097] 其中,所述高速缓存包括高速缓存控制器(例如,29),所述高速缓存控制器包括为所述高速缓存的数据单元实现一组高速缓存一致性状态的电路,其中,所述高速缓存控制器使用所述写通需求属性,以确定所述存储器内的一个或多个位置是否需要存储器一致性。

[0098] 19. 如项目 18 所述的数据系统,进一步包括:

[0099] 可操作地耦接到所述系统互连的高速缓存一致性管理器(例如,18),其中,当所述写通需求属性指示写入事务需要所述存储器一致性时,所述高速缓存一致性管理器指示所述数据系统的其他高速缓存(例如,30,41),针对所述写入事务的存储器地址执行高速缓存一致性查找操作。

[0100] 20. 如项目 19 所述的数据系统,其中,当所述写通需求属性指示所述写入事务不需要所述存储器一致性时,在所述数据系统中不针对所述写入事务执行存储器一致性事务。

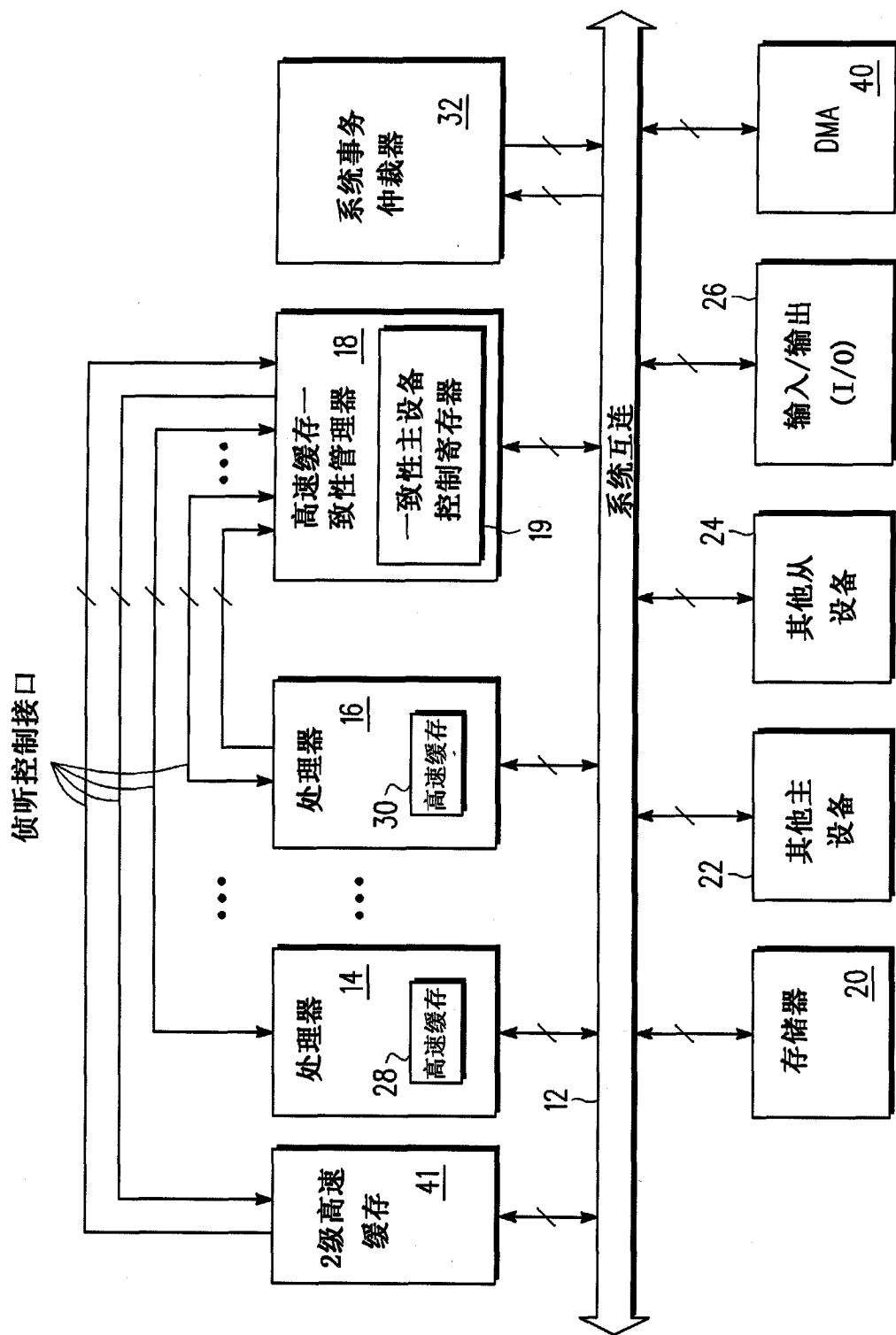
10

图 1

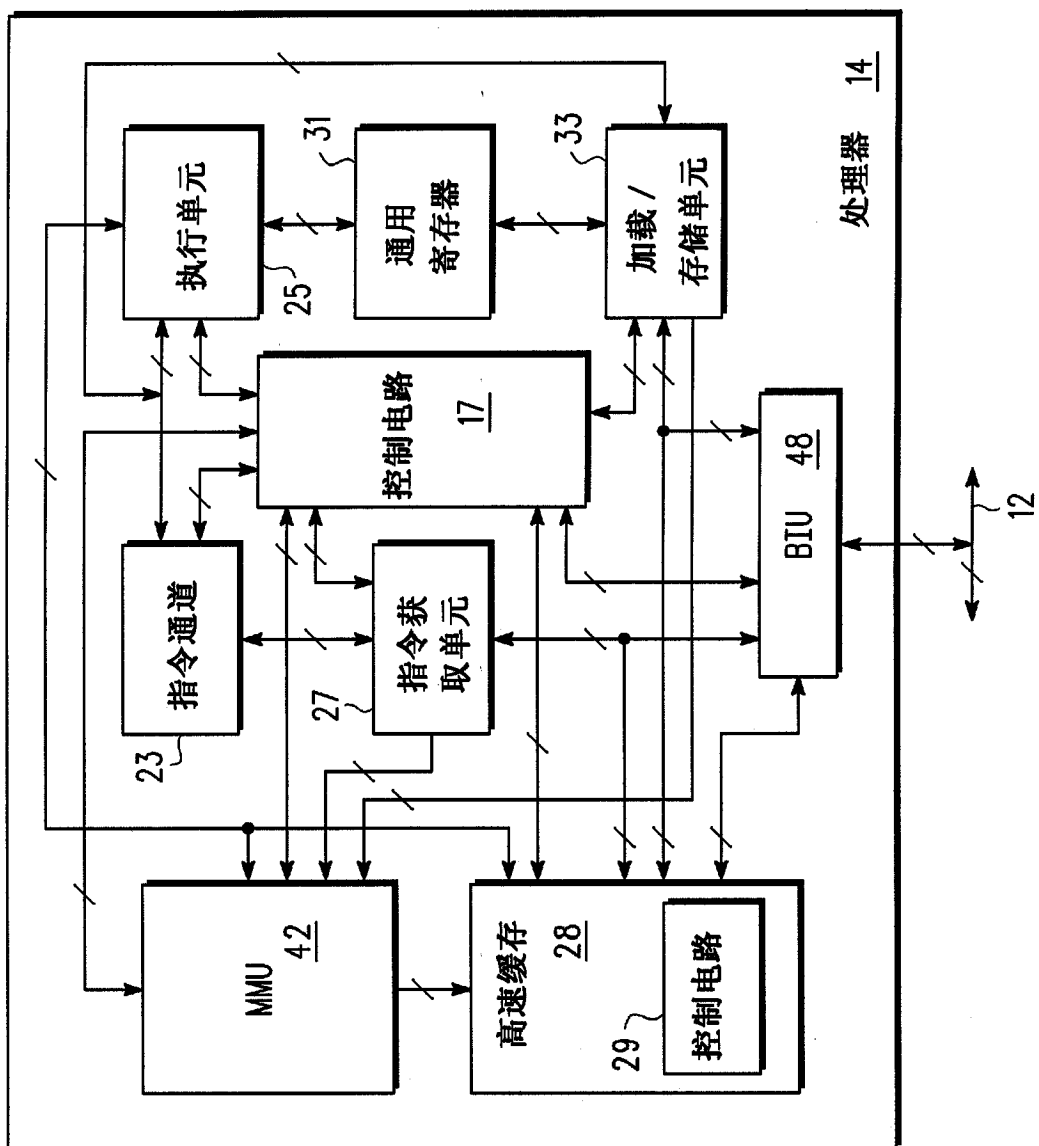


图 2

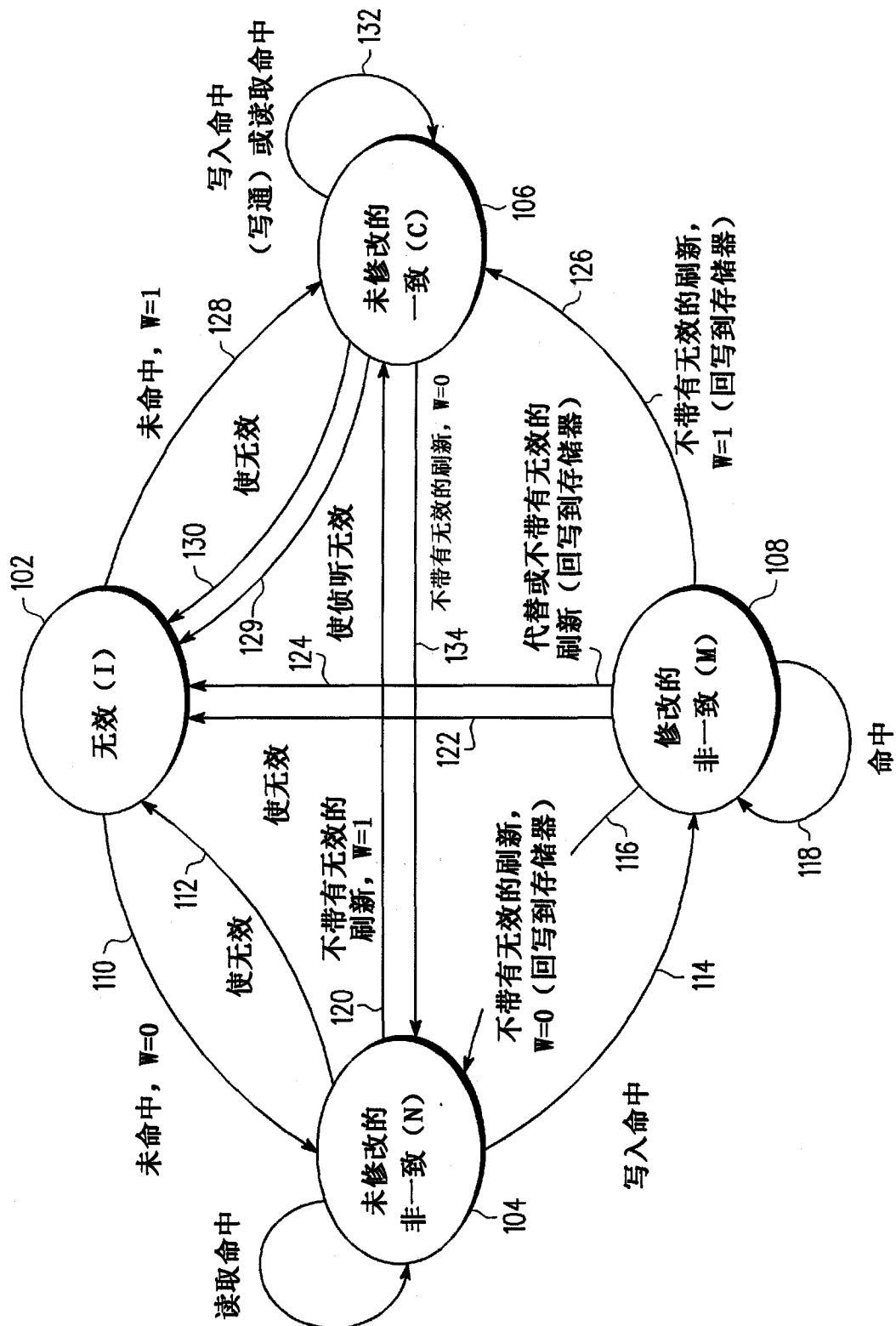


图 3

HBURST[2:0]	类型	说明
000	SINGLE	单个传输
001	INCR	未定义长度的递增突发
010	WRAP4	4拍环回突发
011	INCR4	4拍递增突发
100	WRAP8	8拍环回突发
101	INCR8	8拍递增突发
110	WRAP16	16拍环回突发
111	INCR16	16拍递增突发

对此进行侦听

对此不进行侦听

图 4

HPROT[3] 可高速缓存	HPROT[2] 可缓冲	HPROT[1] 有特权	HPROT[0] 数据/操作码	说明
-	-	-	0	操作码获取
-	-	-	1	数据访问
-	-	0	-	用户访问
-	-	1	-	特权访问
-	0	-	-	不可缓冲
-	1	-	-	可缓冲
0	-	-	-	不可高速缓存
1	-	-	-	可高速缓存

图 5

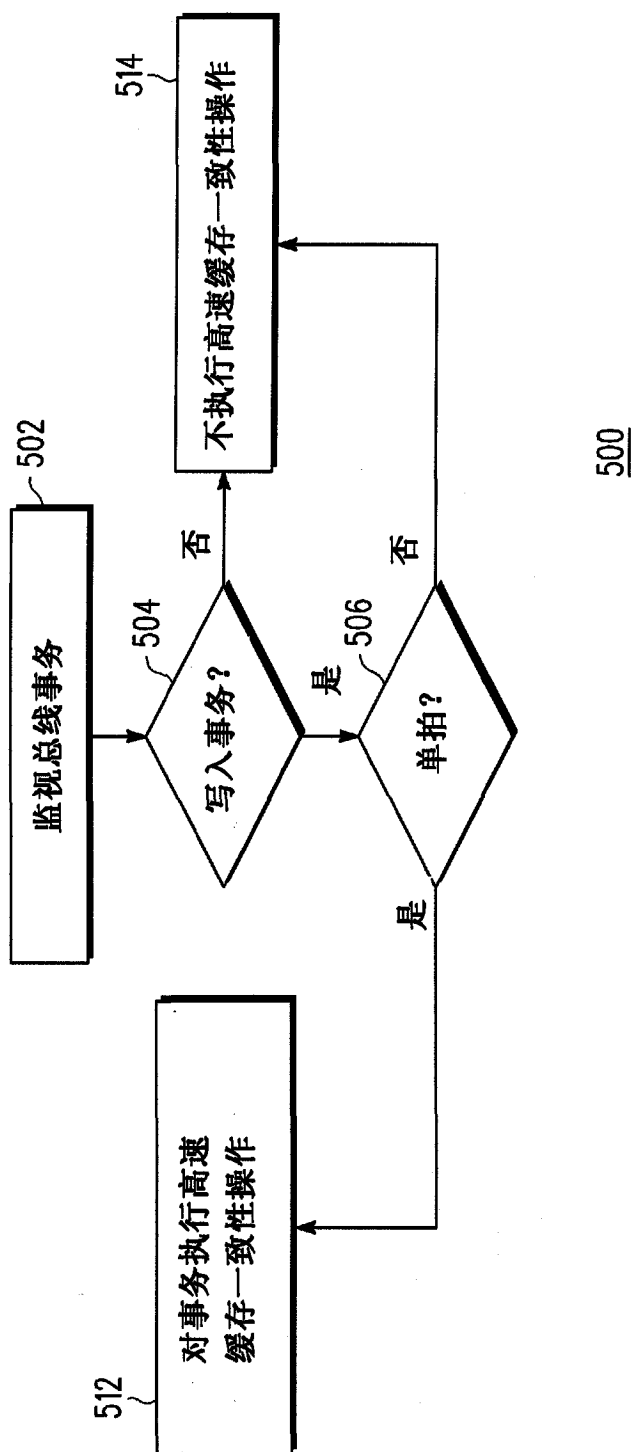


图 6

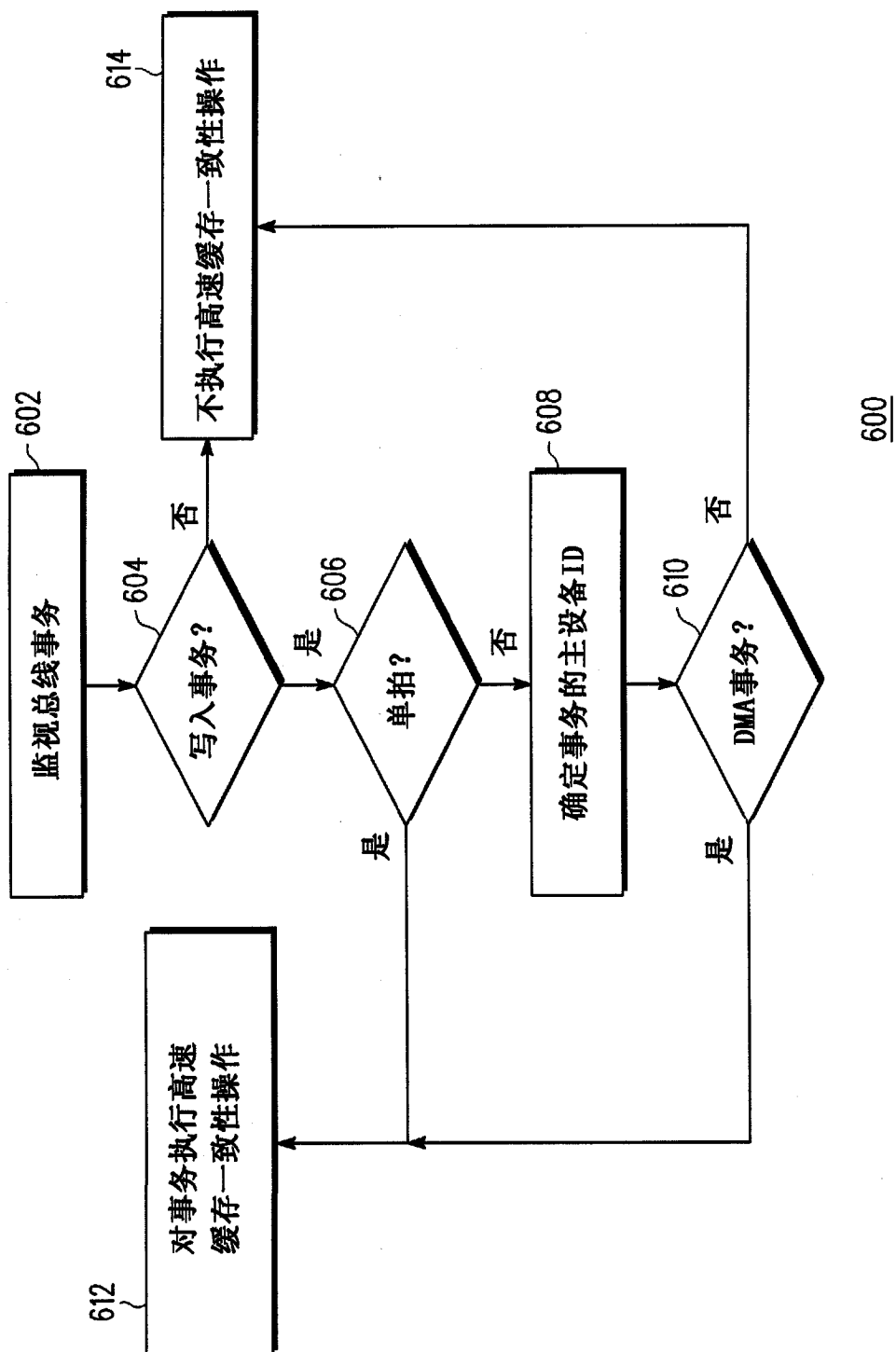


图 7

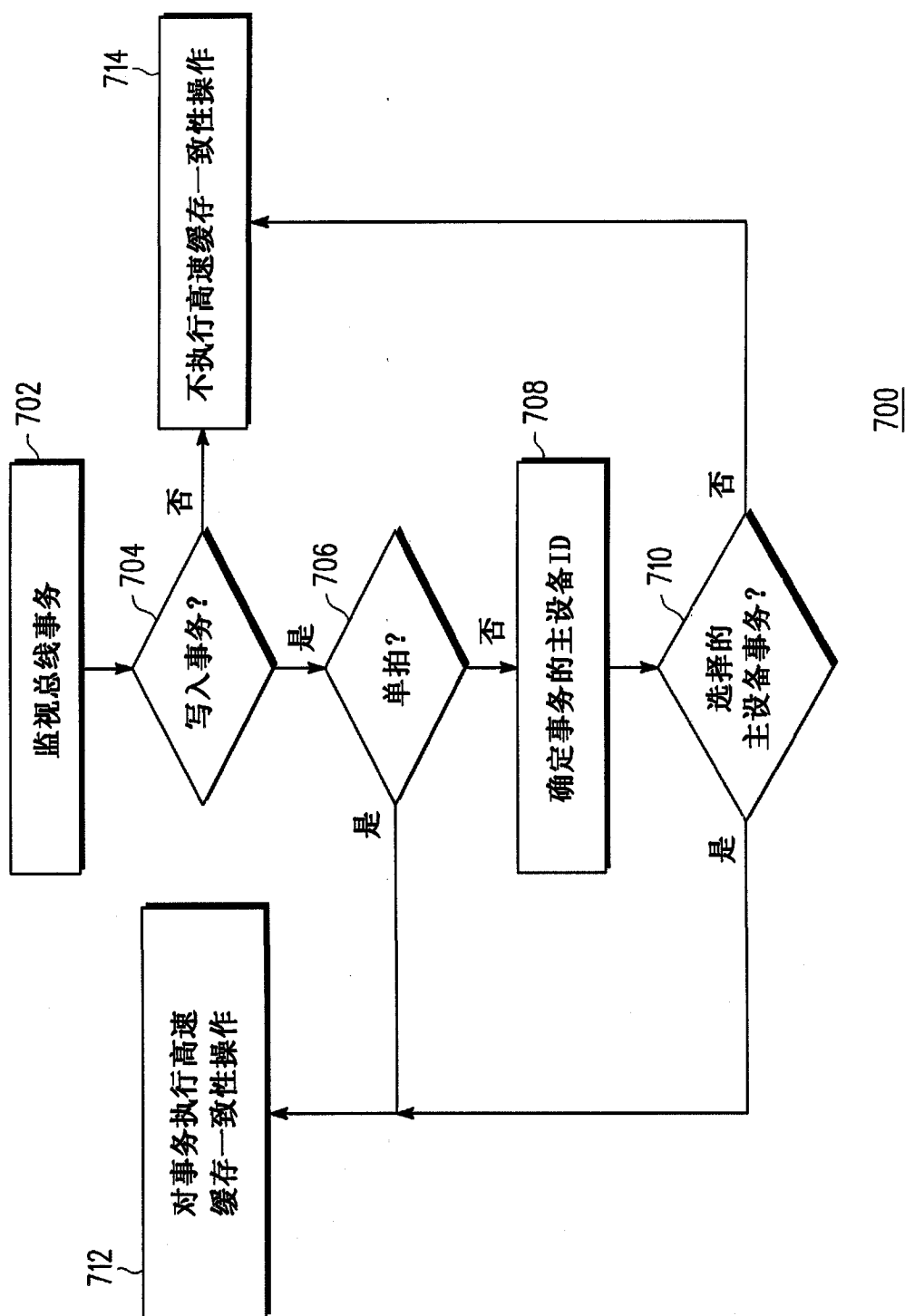


图 8

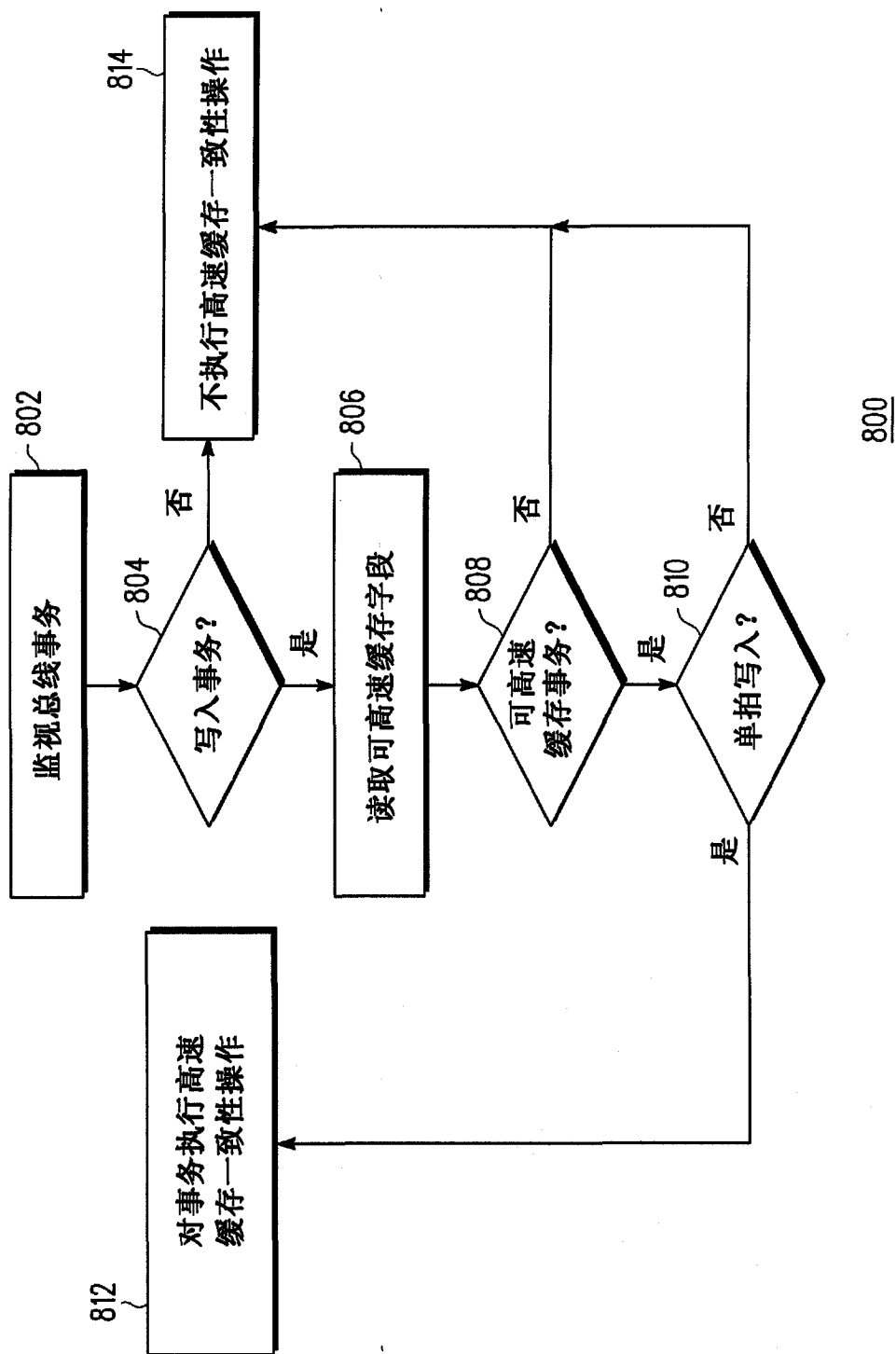


图 9