



(12) 发明专利

(10) 授权公告号 CN 101727313 B

(45) 授权公告日 2013.05.01

(21) 申请号 200910209873.2

(22) 申请日 2007.03.06

(30) 优先权数据

11/370723 2006.03.07 US

(62) 分案原申请数据

200710087737.1 2007.03.06

(73) 专利权人 英特尔公司

地址 美国加利福尼亚州

(72) 发明人 E·克里默 G·萨夫兰斯基

I·蒙亚克 J·多维克

(74) 专利代理机构 中国专利代理(香港)有限公

司 72001

代理人 朱海煜 徐予红

(51) Int. Cl.

G06F 9/38 (2006.01)

G06F 9/318 (2006.01)

(56) 对比文件

US 5715421 A, 1998.02.03,

US 6694424 B1, 2004.02.17,

CN 1209706 C, 2005.07.06,

CN 1589431 A, 2005.03.02,

George Z. Chrysos and Joel S. Emer.

Memory Dependence Prediction using Store Sets. 《Computer Architecture》. 1998,

Glenn Reinman, Brad Calder. Predictive Techniques for Aggressive Load Speculation. 《Microarchitecture》. 1998,

审查员 姜雪

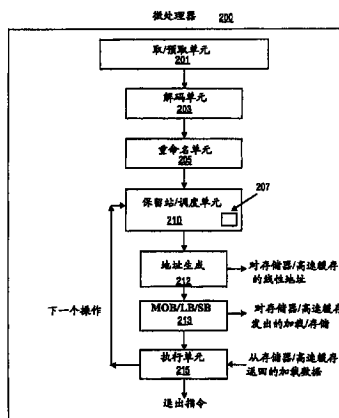
权利要求书1页 说明书9页 附图8页

(54) 发明名称

执行存储器消歧的方法

(57) 摘要

本发明的名称是“执行存储器消歧的技术”。一种存储器存取管理技术。更具体来说,本发明的至少一个实施例涉及在对应于相同目标地址的较旧的存储操作之前对存储器发出加载的技术。



1. 一种用于执行存储器消歧的方法,包括:

如果对应于加载操作的饱和计数器处于门限值并且没有出现最大的误预测率,则把所述加载操作预测为与较旧的挂起存储操作没有冲突;

如果所述加载操作的饱和计数器没有处于门限值,则使所述饱和计数器递增;

如果所述预测不正确,则使看门狗刷新计数器递减;

如果所述预测正确,则使看门狗消歧计数器递增;以及

如果消歧计数器值与刷新计数器值的比率达到最小值,则禁用所述预测。

2. 如权利要求 1 所述的方法,还包括在所述加载操作被预测为与所述较旧的挂起存储操作没有冲突时,在所述较旧的挂起存储操作之前加载与所述加载操作的目标地址对应的数据。

3. 如权利要求 2 所述的方法,还包括通过将比在执行加载操作之后解决的加载操作更旧的存储的目标地址与加载操作的目标地址进行比较,确定所述预测是否正确。

4. 如权利要求 3 所述的方法,还包括在所述预测不正确时,复位所述饱和计数器。

5. 如权利要求 3 所述的方法,还包括在所述预测不正确时,销毁所述加载操作。

6. 如权利要求 3 所述的方法,还包括在所述预测不正确时,与所述较旧的挂起存储操作一起按照程序顺序再发出所述加载操作。

执行存储器消歧的方法

[0001] 本申请是申请日为 2007 年 3 月 6 日、申请号为 200710087737.1、发明名称为“执行存储器消歧的技术”的申请的分案申请。

技术领域

[0002] 本公开涉及信息处理领域,以及更具体来说,涉及存储器存取管理领域。

背景技术

[0003] 在一些现有技术微处理器或处理系统中,信息(数据或指令)可由微处理器采用诸如“加载”操作或“存储”操作之类的操作来存取。此外,加载和存储操作可响应由处理器执行的指令(或者子指令,例如微操作或“uop”)而执行。在一些处理体系结构中,加载指令可被解码为一个 uop,而存储指令则可被解码为两个或两个以上 uop,其中包括存储地址(STA)uop 和存储数据(STD)uop。为了本公开的目的,存储 uop 以及指令将称作“存储操作”或“存储”,以及加载 uop 和指令将称作“加载操作”或“加载”。

[0004] 在一些处理器或处理系统中,多个加载和存储操作可同时执行或者挂起。例如,在包含其中的每个可同时对不同操作起作用的多个处理级的流水线式处理器中,可能有若干加载和存储操作被同时进行,每个在流水线中的不同级上进行。但是,在各个流水线级上,通过加载指令从其中加载数据或者通过存储指令向其中存储数据的地址(统称“目标地址”)是未知的或者“不明确的”。这是因为加载和存储指令或 uop 的目标地址有时在已经开始执行加载或存储之后才确定。

[0005] 图 1 说明一种流水线式处理器的一部分,其中具有取/预取级、对适当指令或 uop 分配寄存器的一个或多个重命名单元、以及调度和存储指令或 uop、例如与加载和存储对应的 uop、直到确定它们相应的目标地址为止的一个或多个调度单元/保留站单元。

[0006] 当加载和存储(例如 STA uop)从保留站分发时,它们可被发送到地址生成单元,地址生存单元生成要发送到存储器或高速缓存的加载和存储的相应线性地址。加载操作通常从保留站分发到存储器有序缓冲器(MOB)内的加载缓冲器中,在其中检验加载是否有冲突以及与其它存储操作的相关性。如果不存在冲突或者与存储的相关性,则加载可分发给存储器/高速缓存群集。否则,在被分发到存储器/高速缓存之前,加载可能需要在 MOB 等到解决了相关性和/或冲突。

[0007] 一旦加载被分发给存储器/高速缓存,存储器/高速缓存可能把作为加载的目标的数据返回给执行单元保留站,执行单元保留站可采用加载数据来生成下一个的地址,用作将从调度器/保留站分发的某个后续 uop 的操作数。

[0008] 可能包括 STA uop 的存储操作可遵照与加载相似的途径。但是,通常不允许存储不按程序顺序分发给存储器/高速缓存,而加载则可在加载与其它存储操作之间不存在相关性/冲突时的任何时间被分发给存储器/高速缓存。

[0009] 在一些现有技术处理器中,MOB 用于以适当顺序存储加载和存储操作,使得向存储单元写信息的所有存储操作被分发并且允许它们在可采用来自相同地址的信息的加载操

作之前将其信息写入存储器。按照程序顺序在相应的加载操作（即加载操作具有与先前的存储操作相同的目标地址）之前出现的存储操作可称作“较旧的”存储操作，以及相应的加载操作可称作比按程序顺序的先前存储操作“较新的”加载操作。

[0010] 如果不存在加载与存储之间的相关性 / 冲突，则加载相对于存储可不按程序顺序来存取存储器。在现有技术的一部分中，在较旧的挂起存储之前所处理的加载被假定为始终对应于相同的目标存储器地址，以便阻止先前处理的加载可能加载将由较旧的存储所较新的数据，因而通过返回废弃的信息而在它们所对应的无论什么程序中产生不正确的结果。

[0011] 但是，由于并非在处理按照程序顺序的较旧的挂起存储之前所处理的所有加载都对应于相同的存储器地址，所以这个假设可能证明是过于保守的。因此，加载可被延迟许多周期向存储器发出，直到相应的较旧的挂起存储被处理并以适当顺序存储在 MOB 中为止。这又可能引起存储器存取时间的不必要延迟，它可能过度侵蚀处理器和系统性能。

发明内容

[0012] 本发明公开了一种用于执行存储器消歧的方法的实施例，包括：

[0013] 如果对应于加载操作的饱和计数器处于门限值并且没有出现最大的误预测率，则把所述加载操作预测为与较旧的挂起存储操作没有冲突；

[0014] 如果所述加载操作的饱和计数器没有处于门限，则使所述饱和计数器递增；

[0015] 如果所述预测不正确，则使看门狗刷新计数器递减；

[0016] 如果所述预测正确，则使看门狗消歧计数器递增；以及

[0017] 如果消歧计数器值与刷新计数器值的比率达到最小值，则禁用所述预测。

附图说明

[0018] 通过附图、作为示例而不是限制来说明本发明。

[0019] 图 1 是现有技术处理器的简图，其中，加载和存储仅当它们在由处理器运行之后处于程序顺序时才向存储器发出。

[0020] 图 2 是根据本发明的一个实施例的处理器简图，其中，允许在处理器中处理的加载在相应的较旧的存储操作通过处理器处理之前来存取存储器。

[0021] 图 3 说明根据本发明的一个实施例的加载缓冲器，它可存储将在相应的较旧的存储操作之前向存储器发出的加载操作。

[0022] 图 4 说明根据一个实施例、预测加载缓冲器条目中存储的加载操作可在相应的较旧的存储之前向存储器发出。

[0023] 图 5 是状态图，说明根据一个实施例的看门狗单元的功能。

[0024] 图 6 是流程图，说明根据一个实施例、加载操作是否可在相应的存储操作之前向存储器发出。

[0025] 图 7 是在其中可采用一个实施例的共享总线系统。

[0026] 图 8 说明在其中可采用一个实施例的点对点总线系统。

具体实施方式

[0027] 本发明的实施例涉及计算机系统。更具体来说,本发明的至少一个实施例涉及允许加载操作在较旧的挂起存储操作之前向存储器发出的技术。此外,本发明的一个实施例涉及根据加载和相应的较旧的存储操作相互没有冲突的频率来禁用或启用在较旧的挂起存储操作之前向存储器发出加载操作的技术。在整个公开中,对于存储,术语“挂起”表示目标地址(加载和存储操作将存取的存储器中的地址)仍未确定的事实。关于加载的术语“挂起”表示其目标地址被确定但仍未向存储器/高速缓存发出的加载。

[0028] 不是始终等待让加载相对于 MOB 中的挂起存储操作按程序顺序来处理,本发明的实施例而是通过带有/没有预测地推测挂起加载(即仍将要退出的加载)是否将从不是对应于任何较旧的挂起存储操作的存储器地址中加载信息,来改进对存储器的加载存取的吞吐量。在至少一个实施例中,加载缓冲器的条目各对应于散列预测器阵列条目,散列预测器阵列条目可保持关于是否将允许相应加载缓冲器条目中的挂起加载在挂起的较旧的存储操作之前来存取存储器的预测。在一个实施例中,挂起加载操作是否可在挂起的较旧的存储操作之前存取存储器的预测取决于与特定预测器表条目对应的先前挂起加载操作的成功(先前的挂起加载是否已存取存储器而没有与较旧的存储操作冲突)。

[0029] 在一个实施例中,饱和计数器可用来保持挂起加载是否将与较旧的挂起存储操作冲突(即,试图存取与较旧的挂起存储操作对应的存储器地址)的预测。如果已经允许在较旧的挂起存储操作之前来存取存储器的加载最终与较旧的存储操作冲突,则至少一个实施例从导致冲突的推测加载重新开始。但是,至少一个实施例中,误预测可能充分地不频繁出现,从而帮助对存储器的加载/存储存取的整体改进。

[0030] 在较旧的挂起存储操作之前存取存储器的推测加载的成功率下降到低于某个门限时,则至少一个实施例可包括“看门狗”单元(逻辑和/或软件)来禁止挂起加载在较旧的挂起存储操作之前推测性地存取存储器。在本公开中,“存储器”可用来表示高速缓存、DRAM 或者将由加载和存储操作存取的其它任何存储器结构。

[0031] 图 2 说明根据一个实施例的处理器的一部分,其中,可按照预测算法、如散列函数,在其它较旧的挂起存储操作之前向存储器地址推测性地发出挂起加载。具体来说,图 2 说明一种流水线式处理器 200 的一部分,其中具有取/预取级 201、解码器级 203、对适当指令或 uop 分配寄存器的一个或多个重命名单元 205、以及存储与加载和存储操作(例如 STA uop)对应的 uop、直到确定它们相应的目标地址源操作数为止的一个或多个调度/保留站单元 210。图 2 还说明生成与加载和存储对应的目标线性地址的地址生成单元 212 以及根据通过向存储器/高速缓存分发加载操作所返回的加载数据来生成对于将从调度器/保留站 210 分发的下一个操作的指针。图 2 还说明 MOB213,它可包含按照程序顺序存储加载和存储并检验加载与存储之间的相关性/冲突的加载和存储缓冲器。

[0032] 在一个实施例中,可在向存储器/高速缓存发出较旧的存储之前向存储器/高速缓存发出加载,而无需等待确定加载是否与较旧的挂起存储相关或冲突。这样,本发明的至少一个实施例可由于以下事实而改进处理器性能:可无需等待确定较旧的存储的目标地址而发出加载,来自其中的数据可能比某些现有技术体系结构更快地由执行单元用来分发来自调度器/保留站的后续操作。

[0033] 在一个实施例中,散列表 207 可与例如存储在加载缓冲器中的加载或者在处理器的任何位置挂起的加载对应的多个条目配合使用。在一个实施例中,散列表存储在以逻辑

电路实现的和 / 或以软件实现的存储器中。在一个实施例中, 散列预测表的各条目可包含至少一个饱和计数器来保持关于是否可在处理器中挂起的较旧的挂起存储操作之前向存储器发出各加载的预测。有利的是, 散列预测条目可根据指令指针 (EIP) 或者它的某种派生 (例如 EIP 的散列形式) 来索引。

[0034] 图 3 说明根据本发明的一个实施例的加载缓冲器, 其中, 各条目可包含可能允许在较旧的挂起存储操作之前以及在加载退出并存储于 MOB 之前来存取存储器的加载操作。除了与加载操作 305 对应的操作码、数据和其它信息之外, 包括例如加载缓冲器条目 301 在内的加载缓冲器条目可包含跟踪与加载缓冲器条目对应的加载操作是否可在较旧的挂起存储操作之前存取存储器的其它字段。

[0035] 例如, 在一个实施例中, 各条目可包含存储表明是否允许加载在挂起的较旧的存储操作之前来存取存储器的位的字段 (MDA 字段 307)、表明相应的加载操作是否在较旧的挂起存储操作之前实际上已存取存储器的字段 (MDD 字段 309)、表明相应的加载操作是否应当使相应的预测器条目被更新 (例如根据具有未决定地址的较旧的存储操作的存在) 的字段 (MDU 字段 311)、表明相应的加载操作是否应当使预测器复位 (例如在加载与存储之间的冲突的情况下) 的字段 (MDR 字段 313) 以及存储表明比特定加载缓冲器条目内的加载较旧的存储缓冲器中的最新的存储的存储缓冲器条目的存储色值的字段 315。类似地, 图 2 的处理器中的存储缓冲器可在各条目中包含表明作为比加载色字段所涉及的存储操作更新的加载缓冲器中的最旧的加载操作的加载色的字段。在各加载缓冲器条目中还可存在其它字段, 其中包括用于对是否允许相应的加载操作在其它挂起的较旧的存储操作之前来存取存储器执行散列预测的饱和计数器字段。图 3 所示的字段的相对位置和大小不代表所有实施例。在其它实施例中, 这些字段可能处于不同的位置, 并且可能具有不同的大小。

[0036] 加载缓冲器或其它结构中存储的加载可与预测器表条目关联, 预测器表条目在一个实施例中可采用饱和计数器 (例如 415) 来实现散列函数, 以便记录无冲突加载和存储的成功预测的历史。在其它实施例中可采用其它预测技术。图 4 说明根据本发明的预测器表 400, 其中的条目对应于例如图 3 所示的至少一个加载缓冲器的条目。在一个实施例中, 预测表可能是一个或多个加载缓冲器的组成部分。在其它实施例中, 预测表可能处于来自一个或多个加载缓冲器的独立电路中。

[0037] 图 4 的预测表采用与加载操作对应的 EIP 405 的散列形式 401 来索引。在一个实施例中, 加载的 EIP 的六个最低有效位经由索引逻辑 410 用来索引 64 个条目 (标记为 0 至 63) 的预测表。在一个实施例中, 各预测器条目相当于 16 种状态的饱和计数器, 其中包括复位状态, 以 4 位来实现。在其它实施例中, EIP 的更多或更少的位或者 EIP 的某个函数可用来索引更多或更少的预测器表条目。同样地, 在其它实施例中, 各条目可能用作具有采用或多或少位的更多或更少状态的饱和计数器。此外, 在其它实施例中可采用其它历史跟踪技术。

[0038] 在一个实施例中, 与将在挂起的较旧的存储操作之前对存储器发出的加载对应的加载饱和计数器 (例如 415) 在加载退出期间被递增或者重置为零值 (或者另外的开始值)。在其它实施例中, 在其它情况下可修改饱和计数器。在一个实施例中, 如果加载退出而没有与在发出加载时挂起的较旧的存储操作“冲突” (即, 从挂起的较旧的存储操作将对其中写信息的存储器地址中加载数据), 则与在其它较旧的挂起存储之前对存储器发出的

加载对应的饱和计数器递增。在一个实施例中,如果加载与在发出加载(例如当加载退出时所确定)时挂起的较旧的存储操作冲突,则与将在较旧的挂起存储器之前对存储器发出的加载对应的饱和计数器被复位。

[0039] 在一个实施例中,如果与加载对应的预测器表中的饱和计数器达到门限值,则在挂起的较旧的存储操作之前从一个或多个处理器保留站(而不是从 MOB)中发出加载。否则,加载可能不是从保留站发出,而是必须相对于较旧的挂起存储按照程序顺序从 MOB 中发出。在一个实施例中,饱和计数器的至少一个的门限值对应于 15 个连续的无冲突加载发布,在一个实施例中以包括复位在内的计数器的 16 种状态中的 15 种来反映。在其它实施例中,可选择较少保守的门限,例如 10 个连续的无冲突加载发布。

[0040] 在一个实施例中,加载可在调度器中进行无序调度,并且稍后发送到 MOB。如果例不存在挂起的先前存储,或者存在挂起的先前存储,但是预测器已经达到门限值,加载则可从将被立即执行的 MOB(“MOB 旁路”)转发。MOB 或者可判定加载仍然无法继续送往执行单元(例如,当存在仍未解决的较旧的存储并且相应的预测器计数器未饱和时),在这些情况下,在解决有问题的存储时,MOB 将在稍后重新调度此加载)。

[0041] 在一个实施例中,在与来自存储器地址的信息的加载对应的饱和计数器反映来自那个存储器地址的加载的 15 个连续的无冲突发布之后,可不按程序顺序对此存储器地址发出后续加载(即,在较旧的存储之前,从保留站,经由 MOB 旁路),而不是关于较旧的挂起存储按程序顺序等待从 MOB 发出。但是,在一个实施例中,如果在退出时确定已经对于较旧的存储将在其中存储信息的存储单元发出加载(即在加载与较旧的存储之间发生冲突),则与那个加载对应的(即与加载的 EIP 对应的)预测表中的饱和计数器复位到初始状态、如零值,或者递减到较小的值,并且没有来自那个地址的其它加载可被预测为无冲突(因而在那个地址的较旧的挂起存储之前发出到存储器),直到饱和计数器再次达到门限值为止(例如,在一个实施例中为 15 个连续无冲突加载之后)。如果加载被误预测、对存储器发出,并且实际上被确定为与较旧的存储冲突,则此加载必须按照程序顺序(即,在已确定较旧的存储操作的目标地址、并且存储按照程序顺序被存储在具有加载的 MOB 中之后)从 MOB 中再发出(例如在执行流水线刷新/重新开始操作之后)。

[0042] 在一个实施例中,与将对存储器发出的加载对应的预测表中的饱和计数器根据散列 EIP 来索引。读取计数器值的结果可存储在与可能位于加载缓冲器或保留站或者另外的某种结构中的加载关联的 MDA 位中(例如“1”表示正预测或“允许”,“0”表示负预测或“不允许”)。如果计数器值饱和(即在门限值处或者高于门限值),则认为加载与仍未对存储器发出的任何较旧的存储没有冲突,以及可发出加载来存取目标地址上的存储器。如果稍后发现加载与较旧的存储冲突(例如在已确定较旧的存储目标地址之后),则使加载刷新/作废,并从 MOB 对存储器再发出(即对于较旧的存储按照程序顺序)。

[0043] 如果对于给定加载读取预测器并且相应的计数器未饱和(即包含低于门限的值),则加载将存储在 MOB 中,并相对挂起的较旧的存储按照程序顺序对存储器发出。在一个实施例中,如果加载与较旧的存储冲突,则设置 MDU 位。如果加载实际上在较旧的挂起存储之前从存储器加载信息(即预测加载不会冲突),则不管加载是否实际上与较旧的存储冲突,均设置 MDD 位。可参考这个位来确定是否需要验证预测是正确的。同样,如果预测不正确,并且实际上存在与较旧的存储的冲突,则将使加载刷新/作废,并与所有后续指令一

起再发出。

[0044] 根据一个实施例,为了检测误码预测,比在执行加载之后决定的加载较旧的存储的目标地址与包括误码预测加载在内的目标地址较新加载进行比较。一旦找到加载和较旧的存储的匹配目标地址,可设置 MDR 位来反映那个加载的计数器必须复位。此外,在一个实施例中,可设置 MDD 位来表明将使误预测加载刷新 / 作废。

[0045] 在一个实施例中,与 MOB 关联的逻辑采用 MDU 和 MDR 位(存储于与在挂起的较旧的存储之前对存储器发出的加载对应的加载缓冲器条目中)来确定将如何更新预测 / 误预测加载的预测器表条目。例如,在一个实施例中,如果未设置 MDU 位,则不更新相应的饱和和计数器,而如果设置了 MDU 位并且未设置 MDR 位,则计数器递增。但是,如果 MDU 和 MDR 位均被设置,则计数器复位。下表概述根据一个实施例、在确定如何或者是否更新给定加载的预测器表条目时的 MDU 和 MDR 位的使用:

[0046]

MDU	MDR	要采取的动作
0	随意	不更新计数器
1	0	计数器递增
1	1	计数器复位

[0047] 在一个实施例中,如果误预测的数量或比率变得过高,则是否在较旧的存储之前对存储器发出加载的预测可暂时禁用,以便保持处理器性能。预测技术则可在一定时间量之后或者在满足某个标准之后重新启用。在一个实施例中,逻辑、软件或者它们的某种组合可用来实现“看门狗”单元,它控制启用还是禁用在较旧的挂起存储之前对存储器发出加载的预测。在一个实施例中,看门狗可概念化为实现两种不同状态(例如经由逻辑状态机),各采用两个计数器来跟踪在较旧的挂起存储之前对存储器发出的任一个或全部加载的预测的成功。

[0048] 例如,在一个实施例中,当上述预测机制启用时,两个计数器 - 消歧计数器和刷新计数器 - 用来跟踪加载预测的成功率。具体来说,每当加载被成功地预测为与较旧的挂起存储操作没有冲突、因而从存储器加载目标信息而无需按照程序顺序对存储器发出(例如从 MOB)时,消歧计数器可递增。另一方面,每当加载被误预测为与较旧的挂起存储操作没有冲突(在一个实施例中可在加载退出之后来确定)时,刷新计数器可递减。另外,刷新计数器可增加来自消歧计数器的进位值。在刷新计数器递减到低于零、由此创建消歧计数器值与刷新计数器值之间的负比率之后,预测机制被禁用,并且对于较旧的存储从 MOB 中按照程序顺序对存储器发出加载。

[0049] 是否启用预测机制的确定也可依靠计数器、例如计算成功的将要变成的预测的数量的预测计数器以及另一个计数器、如计算将要变成的误预测的数量的误预测计数器的使用。在一个实施例中,在退出原本在启用预测机制时会被成功地预测为与较旧的挂起存储没有冲突的加载之后,预测计数器递减。在退出原本在启用预测机制时会被误预测为与较旧的挂起存储没有冲突的加载之后,预测计数器复位到某个初始值(例如“0”),并且误预测计数器递增。在预测计数器达到饱和值(例如计数器的最大数量)并且误预测计数器没

有超过最大门限之后,可启用加载的预测机制。饱和值可取决于实现者认为存在足够的成功加载预测来保证重新启用预测机制的时间。在一个实施例中,这可通过对于每 1 个(例如对应于一个或多个位的误预测计数器)将要变成的误预测的 256 个(例如对应于 8 位的预测计数器)将要变成的成功预测的比率来反映。

[0050] 类似地,预测机制可在满足成功对不成功预测的预期比率之后被禁用。例如,在一个实施例中,如果对于每 1024 个(例如对应于 16 位消歧计数器)成功预测出现 4 个(例如对应于 2 位刷新计数器)或 4 个以上误预测,则禁用预测机制。在其它实施例中,其它技术可用来跟踪预测的成功率,例如时间相关计数器,以便确定启用或禁用预测机制的时间。

[0051] 例如以上所述的看门狗单元可防止无冲突加载的不合理误预测基本上影响处理器性能。图 5 给出状态图,说明根据一个实施例的看门狗单元的功能。在活动状态 501,在各成功预测和刷新计数器根据高于其最大允许值的消歧进位来递增之后,消歧计数器将递增,而刷新计数器则每当加载被误预测时递减。在一个实施例中,如果刷新计数器递减到低于零,则清除所有计数器,并且状态图转变为禁用状态 505。在禁用状态中,如果预测已经成功(即,如果处于启用状态),则刷新计数器递增,而如果误预测已经发生(即,如果处于启用状态),则刷新计数器被清除到其初始状态。在一个实施例中,在刷新计数器达到其最大值或者另外某个门限值之后,或者如果出现来自计数器的进位,则启用预测机制。

[0052] 在一个实施例中,加载缓冲器条目也可包括存储表明相应的加载是否在较旧的存储操作之前已经分发给存储器的存储器消歧推测(MDS)位的存储单元,但用于处于禁用状态的看门狗单元。MDS 位或者相似功能性的位可帮助确定各种加载操作的“命中率”。

[0053] 图 6 是流程图,说明根据一个实施例、在执行本发明的各个方面时可执行的操作。在操作 601,如果加载缓冲器中的特定加载的饱和计数器处于门限,则在操作 610 在看门狗处于启用状态时加载被预测为与任何较旧的挂起存储没有冲突,因而可在操作 615 对存储器无序地发出。如果加载的饱和计数器没有处于门限,则饱和计数器在操作 605 递增。如果看门狗不是处于启用状态,则在操作 611 按照程序顺序发出该加载。然后在操作 612,检验加载是否与较旧的挂起存储冲突。如果加载有冲突,则将要变成的误预测计数器在操作 613 递减。如果没有冲突,则将要变成的成功预测计数器在操作 614 递增。如果加载被预测为与任何较旧的存储没有冲突,并且在操作 620,预测在退出加载时原来是正确的,则成功消歧计数器在操作 625 递增。但是,如果加载被误预测为与任何较旧的挂起存储没有冲突,则在操作 630,饱和计数器复位,以及刷新计数器递减。

[0054] 图 7 说明在其中可使用本发明的一个实施例的前端总线(FSB)计算机系统。处理器 705 从第一级(L1)高速缓冲存储器 710 和主存储器 715 中存取数据。在本发明的其它实施例中,高速缓冲存储器可能是第二级(L2)高速缓存或者是计算机系统存储器分层结构中的其它存储器。此外,在一些实施例中,图 7 的计算机系统可包含 L1 高速缓存以及 L2 高速缓存。

[0055] 图 7 的处理器中所示的是机器状态的存储区 706。在一个实施例中,存储区可能是一组寄存器,而在其它实施例中,存储区可能是其它存储器结构。图 7 中还表示的是根据一个实施例、用于保持区段的存储区 707。在其它实施例中,保持区段可能位于其它装置或存储器结构中。处理器可具有任何数量的处理核。但是,本发明的其它实施例可在系统的其它装置、如独立总线代理中实现,或者通过硬件、软件或它们的某种组合分布于整个系统。

[0056] 主存储器可通过各种存储源来实现,例如动态随机存取存储器(DRAM)、硬盘驱动器(HDD)720或者经由网络接口730远离计算机系统设置的包含各种存储装置和技术的存储源。高速缓冲存储器可设置在处理器内或者设置在处理器附近,例如设置在处理器的本地总线707上。

[0057] 此外,高速缓冲存储器可包含较快的存储单元、如六晶体管(6T)单元,或者接近相等或更快的存取速度的其它存储单元。图7的计算机系统可能是例如微处理器等的总线代理的点对点(PtP)网络,它们经由专用于PtP网络上的各代理的总线信号进行通信。图8说明以点对点(PtP)配置来设置的计算机系统。具体来说,图8说明其中的处理器、存储器和输入/输出装置通过多个点对点接口互连的系统。

[0058] 图8的系统还可包括若干处理器,为了简洁起见仅示出其中两个处理器870、880。处理器870、880均可包括与存储器22、24连接的本地存储控制器集线器(MCH)872、882。处理器870、880可采用PtP接口电路878、888经由点对点(PtP)接口850来交换数据。处理器870、880均可采用点对点接口电路876、894、886、898经由各个PtP接口852、854与芯片组890交换数据。芯片组890还可经由高性能图形接口839与高性能图形电路838交换数据。本发明的实施例可设置在具有任何数量的处理核的任何处理器中或者设置在图8的PtP总线代理的每个中。

[0059] 但是,本发明的其它实施例可存在于图8的系统的其它电路、逻辑单元或装置中。此外,本发明的其它实施例可分布于图8所示的若干电路、逻辑单元或装置上。

[0060] 本文涉及的处理器或者根据本发明的一个实施例设计的其它任何组件可在从创建到模拟到制造的各个阶段中进行设计。表示设计的数据可通过多种方式来表示设计。首先,如在模拟中可用的那样,硬件可采用硬件描述语言或者另一种功能描述语言来表示。作为补充或替代,采用逻辑和/或晶体管门电路的电路级模型可在设计过程的部分阶段产生。此外,在某个阶段,大部分设计达到在其中它们可采用表示各种装置的物理设置的数据来建模的等级。在使用传统半导体制造技术的情况中,表示装置设置模型的数据可能是指在用于生产集成电路的掩模的不同掩模层上的各种特征是否存在的数据。

[0061] 在设计的任何表示中,数据可存储在任何形式的机器可读媒体中。经调制或者以其它方式产生以便传送这种信息的光或电波、存储器或者磁或光存储装置、如盘可能是机器可读媒体。这些媒体的任一个可“承载”或“表明”该设计或者在本发明的实施例中使用的其它信息、如差错恢复例程中的指令。传送表明或承载信息的电载波以便执行电信号的复制、缓冲或重传时,制作新的副本。因此,通信提供商或网络提供商的动作可能是制作体现本发明的技术的产品、如载波的副本。

[0062] 因此,公开用于引导存储器存取、如加载或存储的技术。虽然在附图中描述和表示了某些实施例,但是要理解,这些实施例只是对广泛的发明的说明而不是限制,以及本发明不限于所示及所述的具体构造和配置,因为本领域的技术人员在研究本公开之后可能会想到其它各种修改。在其中的增长迅速并且不易预见其它进步的例如这样的技术的领域中,通过实现技术进步进行促进,所公开的实施例在配置和细节上可易于修改,而没有背离本公开的原理或所附权利要求书的范围。

[0063] 本发明的一个或多个实施例的各个方面可能在可使用本发明的一个或多个实施例的处理器或计算机系统的广告中描述、论述或引用。这类广告可包括但不限于新闻纸、杂

志、广告牌或其它报纸或有形媒体。具体来说,本发明的一个或多个实施例的各个方面可经由网站、弹出式广告或其它基于万维网的媒体在互联网上做广告,而不管包含生成网站或弹出广告的程序是位于美国还是其领区。

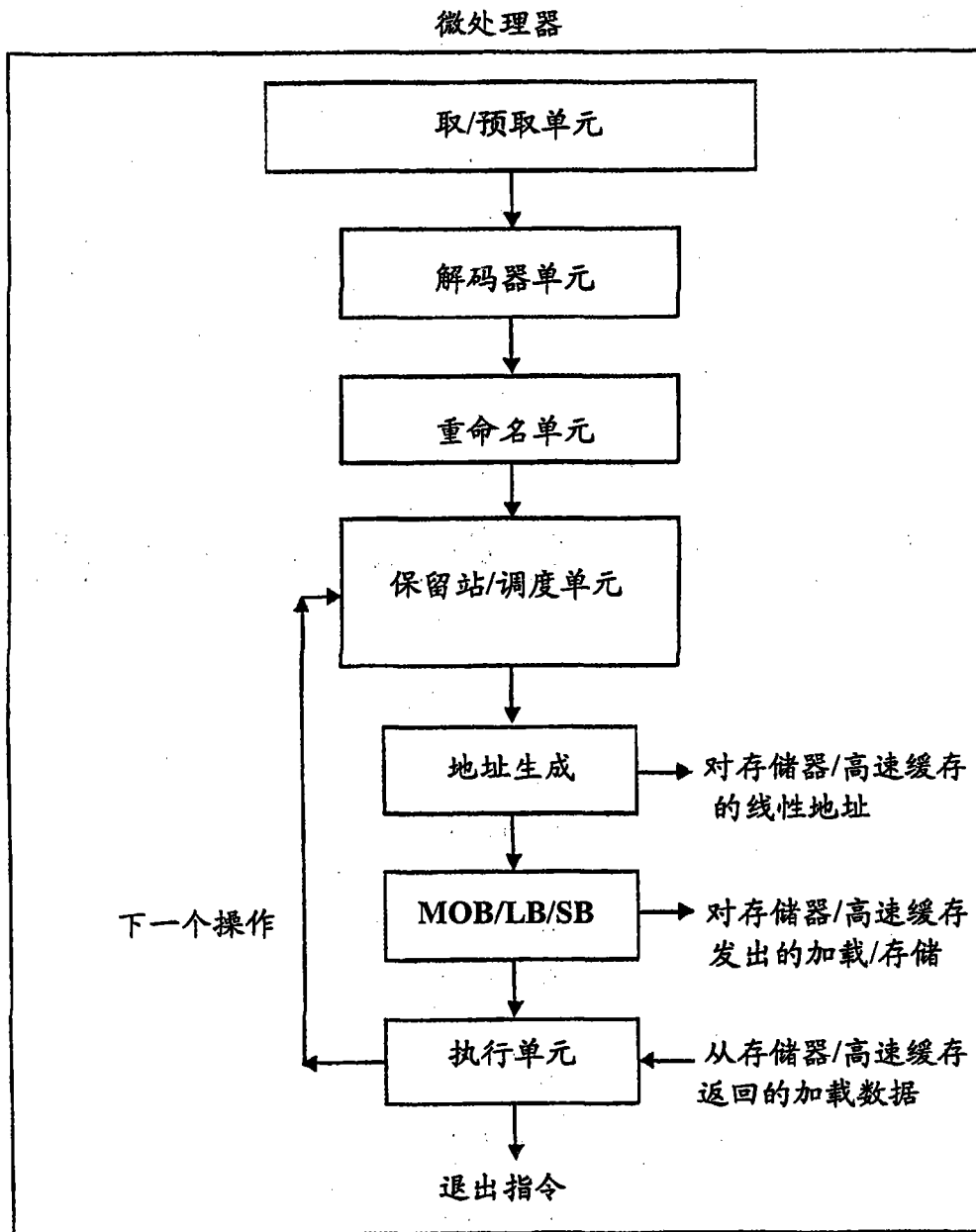


图1 (现有技术)

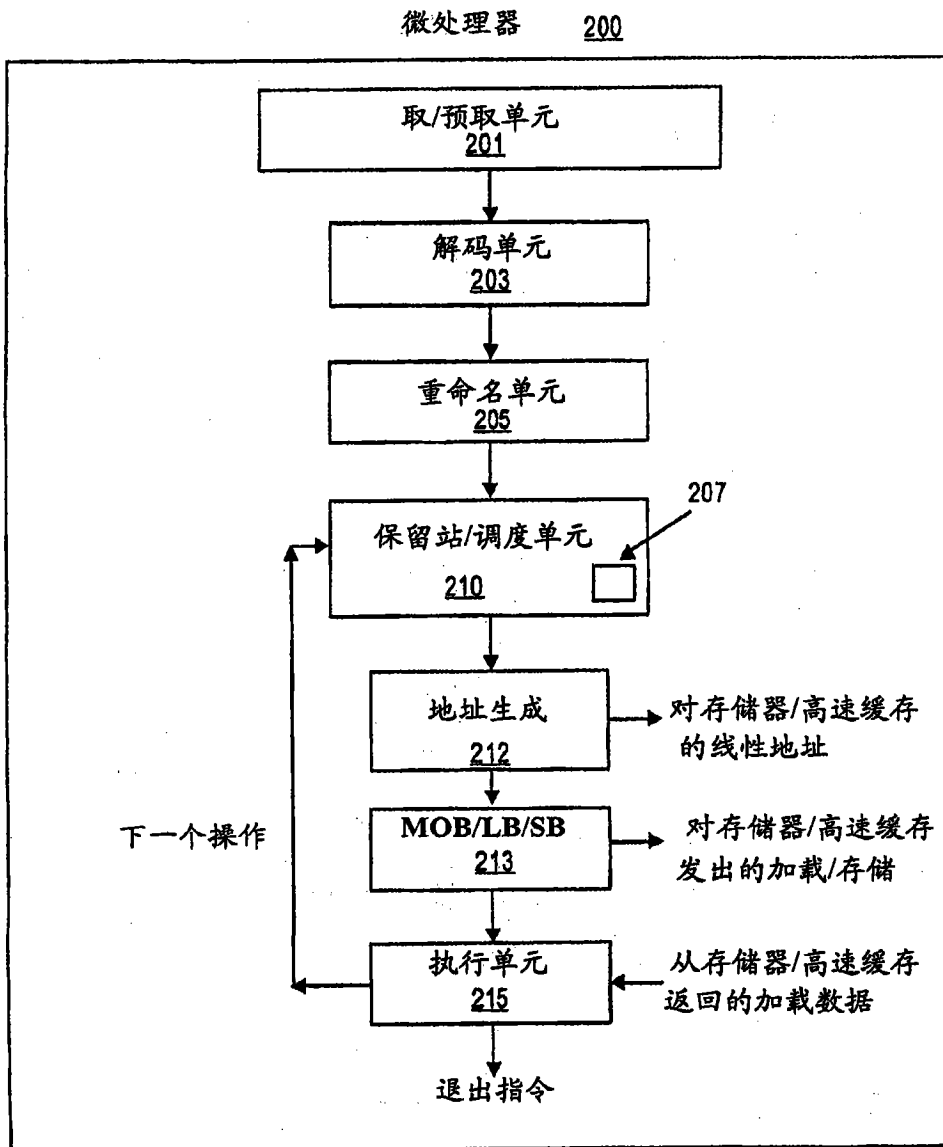


图 2

加载缓冲器
300

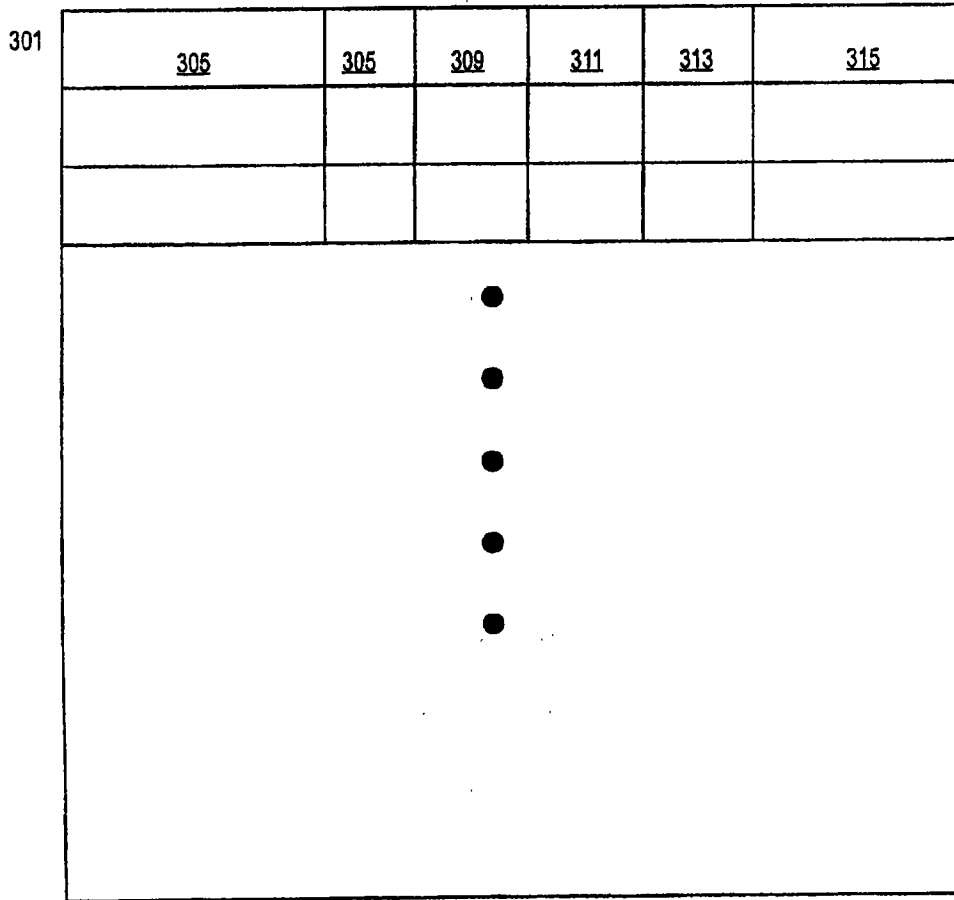


图 3

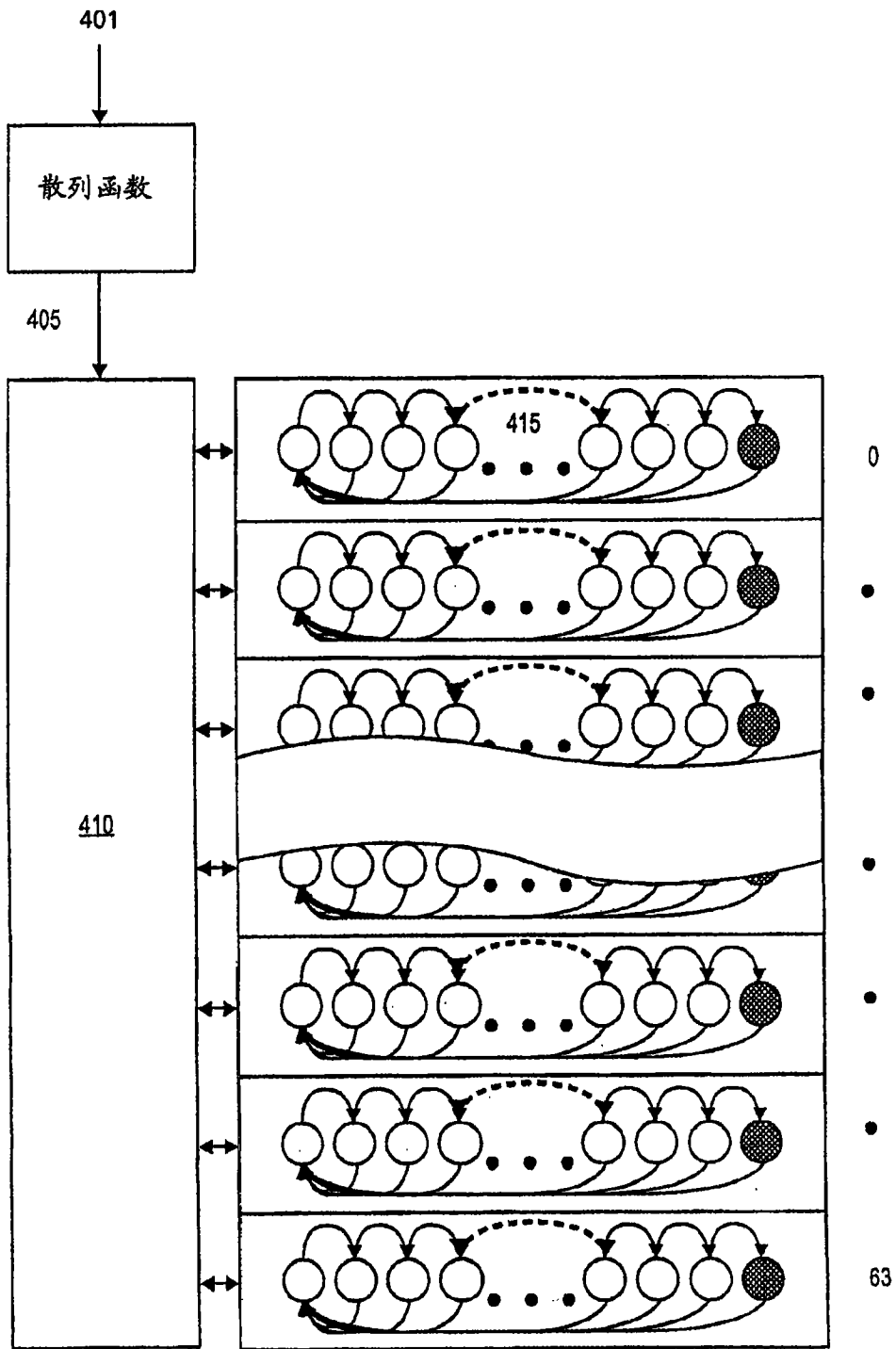


图 4

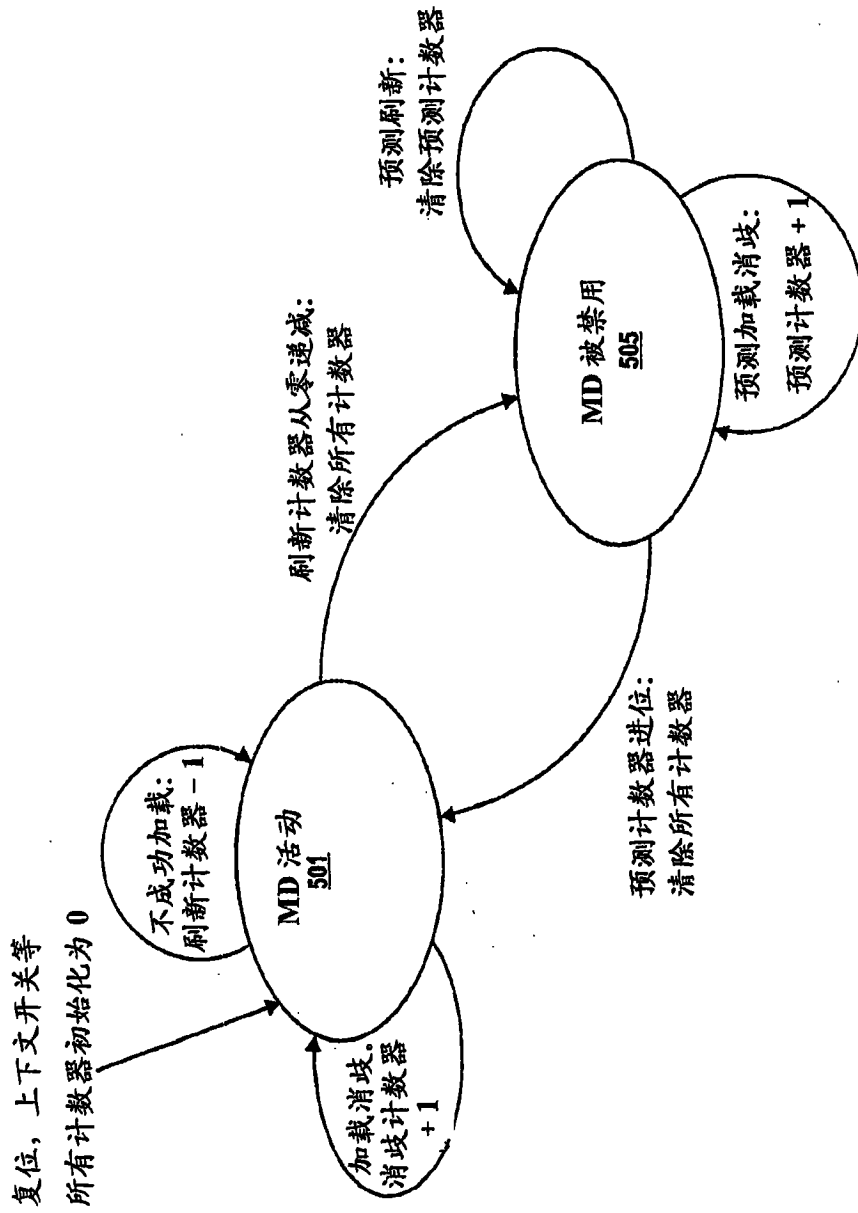


图 5

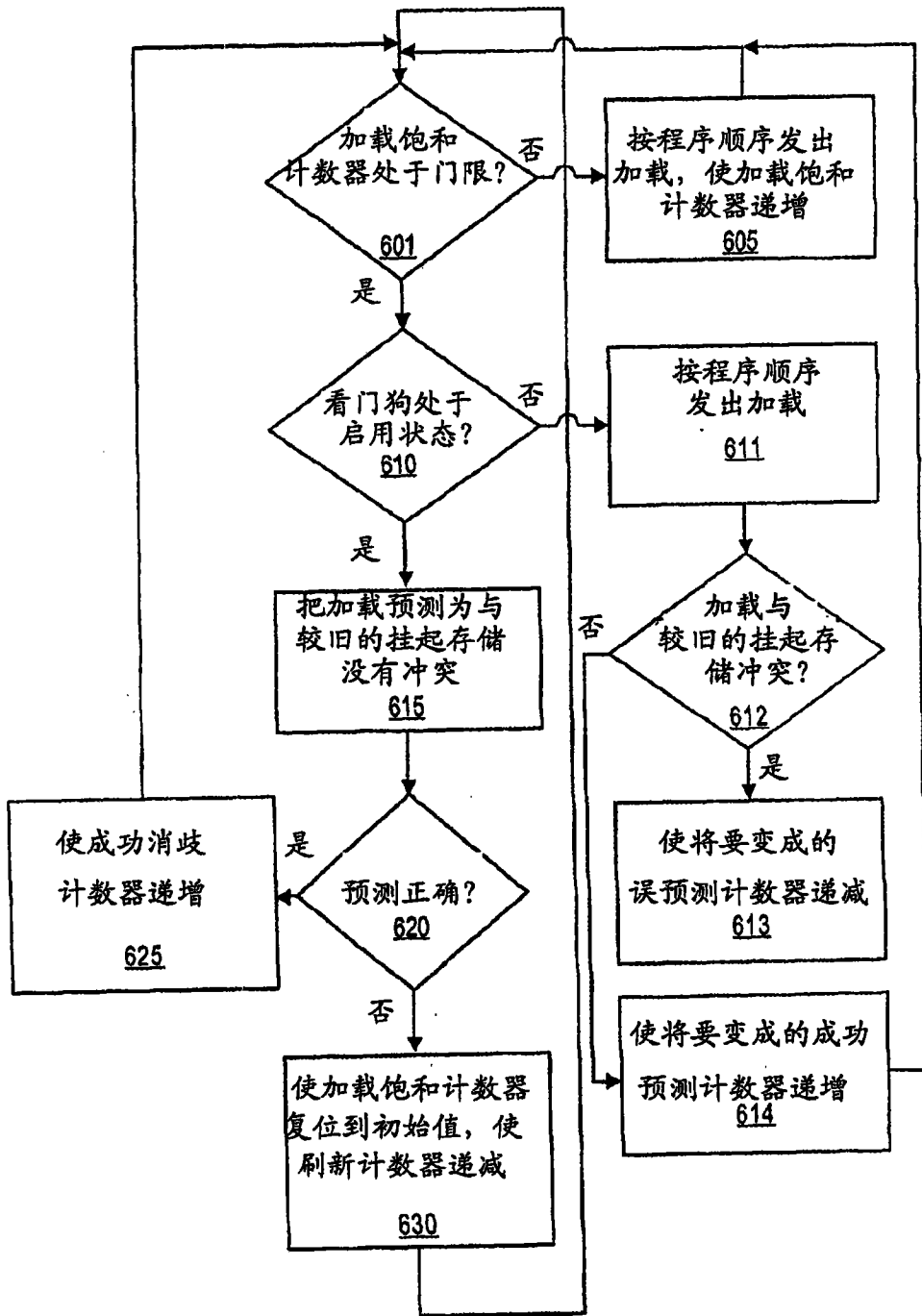


图 6

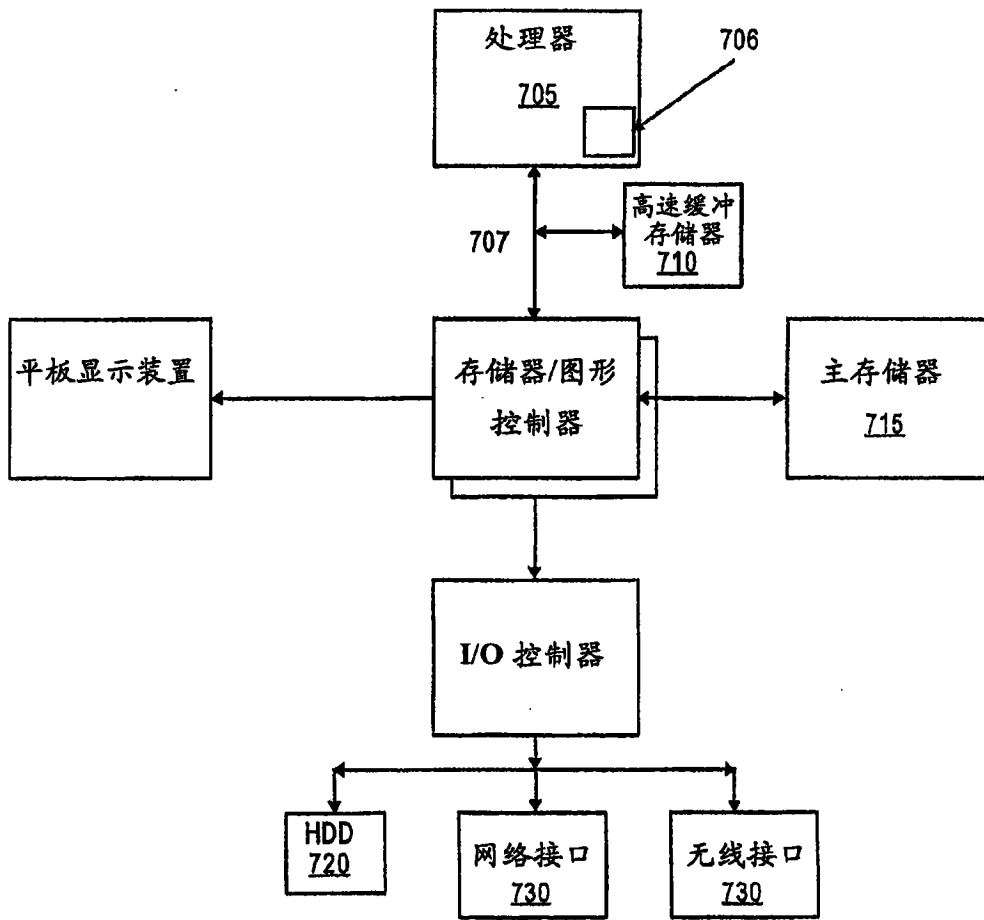


图 7

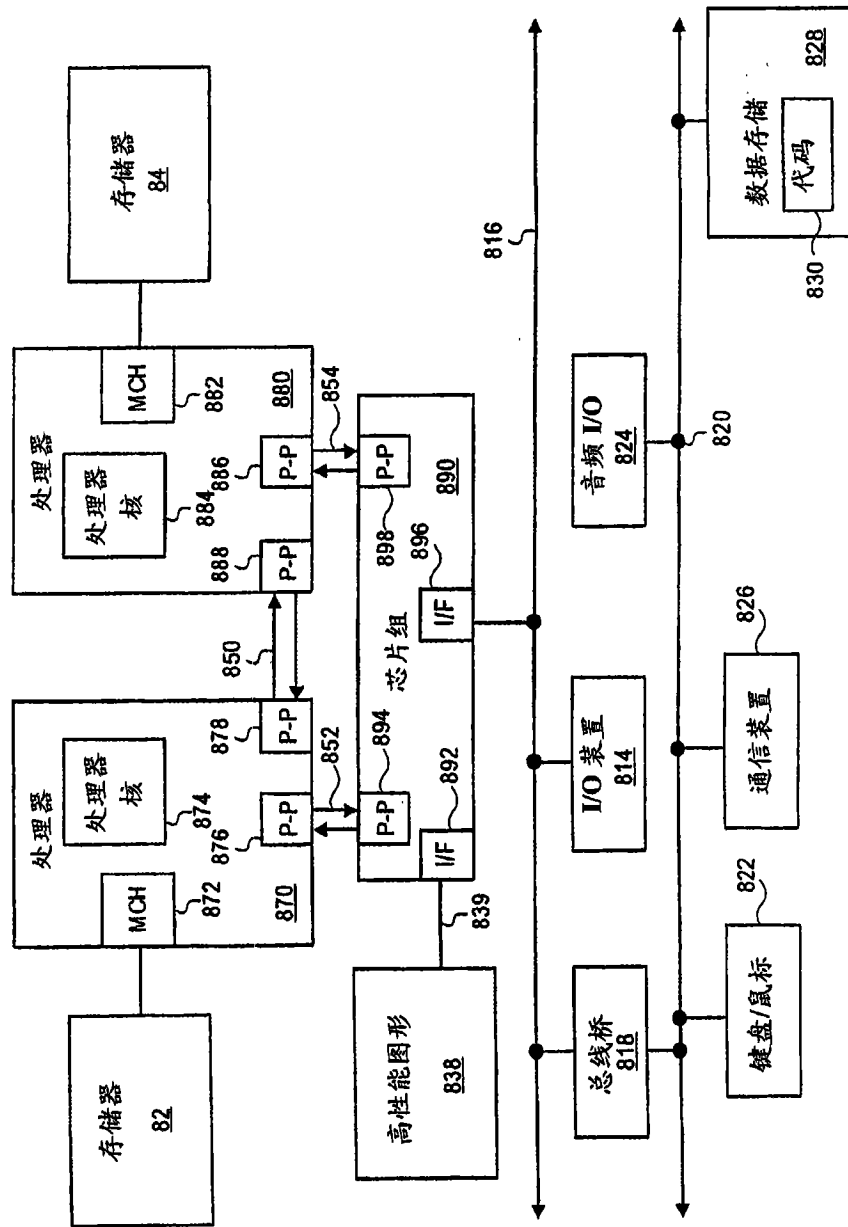


图 8