



(12) 发明专利

(10) 授权公告号 CN 108139976 B

(45) 授权公告日 2020.10.27

(21) 申请号 201580080376.0

(73) 专利权人 安培计算有限责任公司

(22) 申请日 2015.07.30

地址 美国加利福尼亚州

(65) 同一申请的已公布的文献号

(72) 发明人 谢尔·斯文森

申请公布号 CN 108139976 A

(74) 专利代理机构 北京柏杉松知识产权代理事  
务所(普通合伙) 11413

(43) 申请公布日 2018.06.08

代理人 谢攀 刘继富

(30) 优先权数据

(51) Int.CI.

14/710,837 2015.05.13 US

G06F 12/0862 (2016.01)

(85) PCT国际申请进入国家阶段日

G06F 12/126 (2016.01)

2017.11.24

审查员 刘曼

(86) PCT国际申请的申请数据

PCT/US2015/042792 2015.07.30

(87) PCT国际申请的公布数据

W02016/182588 EN 2016.11.17

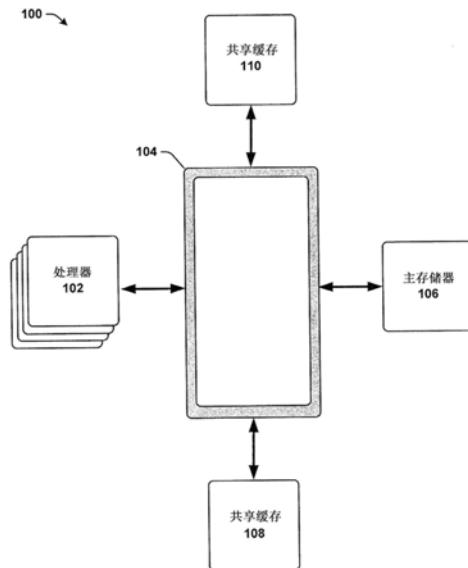
权利要求书2页 说明书6页 附图8页

(54) 发明名称

用于促进移出的预取标记

(57) 摘要

描述了一种将数据从主存储器中预取到缓存，然后将不使用的数据移出到较低级别的缓存的系统。该预取系统将数据从主存储器中预取到缓存，并且不立即使用、或者是太大而不能放入缓存的数据集的一部分的数据能够被标记以移出到较低级别的缓存，与必须从主存储器中再次加载数据相比，使得能够用较短的延迟得到数据。这大大降低了预取可用数据的成本，并且防止缓存垃圾。



1.一种缓存预取系统,包括:

互连,其被配置为可通信地耦接多个处理器、共享缓存和主存储器,其中,所述共享缓存由所述多个处理器中的两个或更多个共享;

处理器缓存预取器,其被配置为经由所述互连从所述主存储器中预取数据集以存储在处理器缓存中,其中,所述处理器缓存与所述多个处理器中的至少一个处理器相关联,并且其中,所述处理器缓存预取器基于所述数据集的第一部分数据被处理器第一将来使用的第一可能性,不将所述数据集的第一部分数据标记为用于移出,并且标记与所述数据集相关联的第二部分数据以指示所述第二部分数据是不使用的,具有被处理器第二将来使用的第一可能性,并且要从处理器缓存移出,其中,所述第一将来使用的第一可能性基于处理器将在第一预定时间段内使用所述数据集的第一部分数据的可能性,并且所述第二将来使用的第一可能性基于处理器将在第二预定时间段内使用所述第二部分数据的可能性;以及

处理器缓存移出器,其被配置为基于对所述第二部分数据的标记,将所述第二部分数据从处理器缓存移出。

2.根据权利要求1所述的缓存预取系统,其中,

所述互连包括环形互连。

3.根据权利要求1所述的缓存预取系统,其中,所述处理器缓存包括2级缓存,并且所述共享缓存包括3级缓存。

4.根据权利要求1所述的缓存预取系统,其中,所述数据集包括与处理器当前请求以由所述处理器执行的数据有关的数据。

5.根据权利要求4所述的缓存预取系统,其中,基于确定与所述数据集相关联的第一地址空间与主存储器中的处理器当前请求的数据相关联的第二地址空间在预定接近度内,确定所述数据集与处理器当前请求的数据有关。

6.根据权利要求1所述的缓存预取系统,其中,所述处理器缓存预取器还被配置为响应于没有在处理器缓存中识别到与所述数据集的至少一部分相关联的存储器位置,经由互连从主存储器中预取所述数据集以存储在处理器缓存中。

7.一种缓存预取系统,包括:

多个处理器;以及

存储器,其存储可执行指令,当所述可执行指令由所述多个处理器中的至少一个处理器执行时,促进操作的执行,所述操作包括:

从系统存储器中预取数据集,以存储在与所述多个处理器中的至少一个处理器相关联的处理器缓存中;

基于所述数据集的第一部分数据被处理器在第一预定时间段内使用的第一可能性,不将所述数据集的第一部分数据标记为用于移出并且不从处理器缓存发送,利用第一指示来标记与所述数据集相关联的第一部分数据,其中,所述数据集的第一部分数据被确定为具有由处理器在所述第一预定时间段内执行的第一预定可能性;

利用第二指示将所述数据集的第二部分数据标记为不使用,所述第二部分数据被预测为要在第二预定时间段内被处理器执行并且要从处理器缓存发送,其中,所述第二部分数据被确定为具有在所述第二预定时间段内被处理器执行的第二预定可能性;以及

基于对所述第二部分数据的标记,将所述第二部分数据从处理器缓存移出到共享缓存

中,其中,所述共享缓存由所述多个处理器中的两个或更多个共享。

8. 根据权利要求7所述的缓存预取系统,其中,

当由处理器执行的第一预定可能性满足预定可能性阈值时,所述第一部分数据不被标记。

9. 根据权利要求7所述的缓存预取系统,其中,所述操作还包括:响应于从所述处理器接收到执行所述第二部分数据的请求,将移出的第二部分数据从所述共享缓存预取到所述处理器缓存。

10. 根据权利要求7所述的缓存预取系统,其中,

所述处理器、所述系统存储器和所述共享缓存经由环形互连可通信地耦接。

11. 根据权利要求7所述的缓存预取系统,其中,从共享缓存取回数据的第一延迟小于从系统存储器取回数据的第二延迟。

12. 根据权利要求7所述的缓存预取系统,其中,所述数据集包括与处理器当前请求以由处理器执行的数据有关的数据,其中,基于确定所述数据集在系统存储器中与处理器当前请求的数据在逻辑上或物理上接近,确定所述数据集与处理器当前请求的数据有关。

13. 一种缓存方法,包括:

从系统存储器中预取数据集,以存储在处理器缓存中,所述处理器缓存与多个处理器中的至少一个处理器相关联;

基于所述数据集的第一部分数据的使用的第一可能性,不将所述数据集的第一部分数据标记为用于移出,所述第一部分数据被预测为要被处理器在第一预定时间段内使用并且不被从处理器缓存传送,其中,所述数据集的第一部分数据被确定为具有被处理器在所述第一时间段内使用的第一预定可能性;

利用第二指示来标记所述数据集的第二部分数据,所述第二指示表示所述数据集的第二部分数据是不使用的,并且被预测为被处理器在第二预定时间段内使用并且要从处理器缓存传送,其中,所述第二部分数据被确定为具有在第二预定时间段内被处理器使用的第二预定可能性;以及

基于利用所述第二指示对所述第二部分数据的标记,将所述第二部分数据从处理器缓存移出到共享缓存中,其中,所述共享缓存由所述多个处理器中的两个或更多个共享。

14. 根据权利要求13所述的缓存方法,其中,当由处理器使用的第一预定可能性满足预定可能性阈值时,不标记所述数据集的第一部分数据。

15. 根据权利要求13所述的缓存方法,还包括:响应于从所述处理器接收到执行所述第二部分数据的请求,将移出的第二部分数据从所述共享缓存预取到所述处理器缓存。

16. 根据权利要求13所述的缓存方法,其中,经由耦接所述处理器、所述系统存储器和所述共享缓存的环形互连来执行所述预取和移出。

17. 根据权利要求13所述的缓存方法,还包括:

将第二指示位设置为第二值以便利用所述第二指示来标记所述第二部分数据。

## 用于促进移出的预取标记

### 技术领域

[0001] 本公开涉及将数据从主存储器中预取到缓存，并移出数据以防止缓存垃圾。

### 背景技术

[0002] 目前的微处理器比存储有程序的存储器要快得多，这意味着如果从主存储器中读取指令，则程序指令不能被快速读取以保持微处理器繁忙。向每个处理器增加一个缓存（小容量的非常快速的存储器）能够通过提供对所需指令的更快速的访问来加快处理时间。

[0003] 预取是将预期的指令在被处理器请求之前加载到缓存中的过程。现代系统已发展成多缓存系统，其中每个处理器或处理器的每个核心可以具有专用于每个核心/处理器的一级或两级缓存，以及在核心/处理器之间共享的一级或多级附加缓存。远离核心/处理器的每个连续级别的缓存可以比先前级别的缓存更大但是却更慢。从主存储器预取到处理器缓存（例如1级或2级缓存）能够提供对数据的低延迟访问，但是由于处理器缓存的大小很小，所以没有立即使用的数据、或者大于缓存大小的数据集能够引起资源冲突和干扰（缓存垃圾）。

### 发明内容

[0004] 在一个实施例中，缓存预取系统包括互连，其被配置为可通信地耦接处理器、共享缓存和主存储器。缓存预取系统能够包括处理器缓存预取器，其被配置为经由互连从所述主存储器预取数据集到处理器缓存，其中，所述处理器缓存与所述处理器相关联，并且其中所述处理器缓存将该数据集的第一部分数据标记为不使用的，并且将该数据集的第二部分数据标记为已使用的。缓存预取系统能够包括处理器缓存移出器，其被配置为经由互连将第一部分数据移出到共享缓存，并且经由互连将第二部分数据移出到主存储器。

[0005] 在另一实施例中，缓存预取系统能够包括处理器；以及存储可执行指令的存储器，可执行指令在被处理器执行时促进操作的性能。操作能够包括从系统存储器中预取数据集到与处理器相关联的处理器缓存，并且利用第一部分数据是已执行的数据的指示来标记该数据集的第一部分数据。操作能够包括利用第二部分数据是可能要执行的数据的指示来标记该数据集的第二部分数据，并且将第二部分数据移出到共享缓存。

[0006] 在另一实施例中，提供了一种方法，其包括从系统存储器中预取数据集到与处理器相关联的处理器缓存。方法能够包括利用第一部分数据是已使用的数据的指示来标记该数据集的第一部分数据。方法能够包括利用第二部分数据是不使用的数据的指示来标记该数据集的第二部分数据。方法能够包括将第二部分数据移出到共享缓存。

### 附图说明

[0007] 图1是示出了用于利用环形架构互连来预取的系统的实施例的框图。

[0008] 图2是示出了预取系统的实施例的框图。

[0009] 图3是示出了预取系统的实施例的框图。

- [0010] 图4是示出了预取系统的实施例的框图。
- [0011] 图5是示出了预取系统的实施例的框图。
- [0012] 图6是示出了具有预取缓存系统的处理器的实施例的框图,所述预取缓存系统移出数据以防止缓存垃圾。
- [0013] 图7是示出了具有预取缓存系统的处理器的实施例的框图,所述预取缓存系统移出数据以防止缓存垃圾。
- [0014] 图8示出了用于预取和移出不使用数据以防止缓存垃圾的方法的实施例的流程图。
- [0015] 图9示出了示例性电子计算环境的框图。
- [0016] 图10示出了示例性数据通信网络的框图。

## 具体实施方式

[0017] 各个实施例提供了一种系统,其将数据从主存储器中预取到缓存,并且然后将不使用的数据移出到较低级别的缓存。该预取系统将数据从主存储器中预取到缓存,并且不能立即使用的数据、或者太大而不能放入缓存的数据集中的一部分数据能够被标记以移出到较低级别的缓存,与数据必须从主存储器中再次加载相比,使得能够用较短的延迟得到数据。这大大降低了预取可用数据的成本,并且防止缓存垃圾。

[0018] 由于从较慢的存储器(DDR、磁盘)中的加载在本质上往往是空间和时间的,因此能够在数据被使用之前预取数据。预取利用了加载的空间性质,因为相关数据可能在物理上或逻辑上位于处理器正在请求的数据的附近。因此,对于地址流而言,对存储器地址A进行加载之后,将很大可能很快对存储器地址A+N进行加载,其中N能够是任意的整数,并且从而可能发展到发生对地址A+N\*m进行加载,其中m是另一任意的带符号整数。

[0019] 通过预取相关数据以及当前所请求的数据,能够避免以后从主存储器或从存储器中加载数据。由于处理器级缓存(例如2级缓存)不是非常的大,所以不应将数据存储在那里,除非所述数据下一步就被执行或即将很快地被处理。因此,不立即在下一步执行的数据能够被标记为不使用的数据,并且能够被移出到较低级别的缓存,例如三级缓存。3级缓存可能比2级缓存慢,但是其可能更大,并且能够更加可行地保留稍后可能使用的数据。当移出到3级缓存的数据之后被处理器请求时,相比于从主存储器中预取数据,2级缓存能够从3级缓存中更快地预取数据。

[0020] 图1是示出了用于利用环形架构互连来预取的系统100的实施例的框图。处理器102能够包含一个或多个核心(图1中示出了4个核心)。处理器102还能够包含处理器级缓存,或与处理器102的每个核心相关联的缓存。在一些实施例中,该缓存能够是2级缓存。处理器102还能够经由环形互连104可通信地耦接到一个或多个共享缓存108和110以及主存储器106。在具有环形互连(例如环形互连104)的系统中,共享缓存108和110能够在一个或多个处理器(例如,处理器102等)之间共享,而无需直接绑定到每个处理器。在一些实施例中,共享缓存、3级缓存因此能够分布在多个核心和/或处理器之中。

[0021] 在一个实施例中,处理器102能够包含用于处理器102中的每个核心或每对核心的处理器级缓存或2级缓存。由处理器102执行的数据能够从2级缓存中取出到1级缓存,或者甚至直接取出到与处理器102相关联的寄存器。为了确保处理器102上的处理器级缓存具有

处理器102将执行所需的数据，并且为了避免处理器在缓存中未找到存储位置的缓存缺失，处理器缓存能够经由环形互连104从主存储器106中预取数据。使用预测性启发法，处理器缓存中的预取器能够使用上述A+N\*m算法来预测接下来可能会使用哪个数据，并且将该数据预加载到处理器缓存中，使得当处理器102请求该数据时该数据是可用的。

[0022] 由于相关数据可能在物理上或在逻辑上接近所请求的数据，或在地址空间上是相关的，所以除了所请求的数据之外，还能够预取相关数据，因为处理器可能会稍后请求该相关数据。如果数据集对于处理器上的2级缓存太大，或者在预定的时间段内可能不被使用，则2级缓存能够将该数据标记为“仅预取”或“不使用”，这将会促进该数据移出到共享缓存108或110。可能完全不相关的数据能够被移回到主存储器106，但是如果数据可能在不久的将来会被使用，或者在预定的时间段内会被使用，则数据能够被标记以移出到共享缓存108或110。基于数据是否要被移出到共享缓存108或110，能够通过设置指示位打开或关闭来标记数据。在一个实施例中，能够基于处理器将来使用的可能性的指示来标记数据。基于将来使用的可能性，3级缓存或共享缓存108或110能够在数据移出到主存储器106之前将数据保持预定的时间长度。类似地，如果在预定的时间长度内的将来使用的可能性非常高，则处理器缓存可以不将数据标记为用于移出。

[0023] 当处理器稍后请求该数据时，从共享缓存108或110中取回数据的延迟将比从主存储器106中取回数据的延迟短得多。

[0024] 应当理解，尽管已经参考了2级和3级缓存，但是本文的描述标记预取数据以移出到分级较低的缓存系统的预取系统的概念可以在其他实施例中应用。例如，数据能够被预取到1级缓存，然后移出到2级缓存，并且在其他实施例中，能够使用3级和4级缓存或其各种组合。

[0025] 图2示出了预取系统200的示例性实施例的框图。处理器202能够包含一个或多个核心(图2中示出了4个核心)。处理器202还能够包括处理器级缓存204，并且在一些实施例中，每个核心能够包括例如缓存204的缓存。在一些实施例中，该缓存204能够是2级缓存。处理器202还能够可通信地耦接到互连210并且经由互连210耦接到共享缓存208和主存储器206。在实施例中，共享缓存208能够在一个或多个处理器(例如，处理器202等)和/或核心之间共享，而无需直接绑定到每个处理器。在一些实施例中，共享缓存208、3级缓存因此能够分布在多个核心和/或处理器中。

[0026] 由处理器202执行的数据能够从2级缓存中被取出到1级缓存，或者甚至直接取出到与处理器202相关联的寄存器。为了确保处理器202上的处理器级缓存204具有处理器202将执行所需的数据，并且为了避免处理器在缓存中未找到存储位置的缓存缺失，处理器缓存能够经由互连210从主存储器206中预取数据。使用预测性启发法，处理器缓存中的预取器能够使用上述A+N\*m算法来预测接下来可能会使用哪个数据，并且经由互连210向主存储器206发送请求以将该数据预加载到处理器缓存204中，使得当处理器202请求该数据时该数据是可用的。

[0027] 图3示出了预取系统300的实施例的框图。在处理器302和缓存304中的预取器从主存储器306中请求要预加载的数据之后，该数据能够经由互连310从主存储器306被发送到处理器302和缓存304。在一些实施例中，能够跳过共享缓存308(3级缓存)将数据直接预取到2级缓存304，并且在其他实施例中，能够首先将数据预取到共享缓存308，然后从共享缓

存308中预取到缓存304。由缓存304预取的数据能够是处理器302正在请求的数据以及可能与被请求的数据相关的其他数据。相关的数据能够在地址空间上相关的数据，或者在物理上或逻辑上位于所请求的数据附近的数据，并且通过预取所有可能相关的数据能够比在多个操作中预取所请求的数据更高效。

[0028] 图4示出了预取系统400的实施例的另一框图。在处理器402和缓存404已经从主存储器406中预取相关数据的集合之后，缓存404能够经由互连410将数据的一部分移出到共享缓存408。由于相关的数据可能在物理上或在逻辑上接近所请求的数据，或在地址空间上是相关的，所以除了所请求的数据之外，还能够预取相关的数据，因为处理器可能会稍后请求该相关的数据。如果数据集对于处理器402上的缓存404太大，或者在预定的时间段内可能不被使用，则缓存404能够将该数据标记为“仅预取”或“不使用”，这将会促进该数据移出到共享缓存408。可能完全不相关的数据能够被移回到主存储器406，但是如果数据可能在不久的将来会被使用，或者在预定的时间段内会被使用，则数据能够被标记以移出到共享缓存408。基于数据是否要被移出到共享缓存408，能够通过设置指示位打开或关闭来标记数据。在实施例中，能够基于处理器402将来使用的可能性的指示来标记数据。基于将来使用的可能性，3级缓存或共享缓存408能够在数据移出到主存储器406之前将数据保持预定的时间长度。类似地，如果在预定时间段内的将来使用的可能性非常高或高于预定可能性，则处理器缓存404可以不将数据标记为用于移出。

[0029] 图5示出了预取系统500的实施例的框图。在标记数据已被移出到共享缓存508之后，如果处理器502请求被移出的数据，或者缓存504上的预取器确定处理器502将很快地处理/执行相关数据，则能够经由互连510从共享缓存508中预取该数据。在实施例中，如果数据已经被移回到主存储器506，则缓存504能够经由互连510从主存储器506中加载数据。

[0030] 图6示出了具有预取缓存系统600的处理器的实施例的框图，所述预取缓存系统600移出数据以防止缓存垃圾。处理器602能够包含一个或多个核心以及缓存604。在一些实施例中，每个核心或每对核心可以具有相应的缓存。在一些实施例中，缓存604能够是2级缓存。处理器602还能够经由互连610可通信地耦接到共享缓存612和主存储器614(DDR、磁盘等)。在具有环形互连(例如互连610)的系统中，共享缓存612能够在一个或多个处理器(例如，处理器602等)之间共享，而无需直接绑定到每个处理器。在一些实施例中，共享缓存612、3级缓存因此能够分布在多个核心和/或处理器之中。

[0031] 在实施例中，缓存604能够包括预取组件606(预取器)和移出组件608(移出器)。预取组件能够被配置为经由互连610将数据集从主存储器614中预取到缓存604，其中处理器缓存604与处理器602相关联。预取组件606能够将来自该数据集的第一部分数据标记为不使用的，并将来自该数据集的第二部分数据标记为已使用的。

[0032] 移出组件608能够被配置为经由互连610将第一部分数据移出到共享缓存612，并且经由互连610将第二部分数据移出到主存储器614。

[0033] 图7示出了具有预取缓存系统700的处理器的实施例的框图，所述预取缓存系统700移出数据以防止缓存垃圾。处理器702能够包括具有标记组件706和预测组件708的缓存704。标记组件706能够标记由缓存704预取的数据以指示该数据是不使用的或将来可能会使用的。

[0034] 由于与处理器所请求的数据相关的数据可能在物理上或在逻辑上接近所请求的

数据,或在地址空间上是相关的,所以除了所请求的数据之外,还能够预取相关的数据,因为处理器702可能会稍后请求相关的数据。如果数据集对于处理器上的缓存704太大,或者在预定的时间段内可能不被使用,则缓存704能够将该数据标记为“仅预取”或“不使用”,这将会促进该数据移出到3级缓存或更低级的缓存。可能完全不相关的数据能够被移回到主存储器,但是如果数据可能在不久的将来会被使用,或者在预定的时间段内会被使用,则数据能够被标记以移出到共享缓存。基于数据是否要被移出到共享缓存,能够由标记组件706通过设置指示位打开或关闭来标记数据。在一个实施例中,能够基于处理器将来使用的可能性的指示来标记数据。预测组件708能够确定在预定时间段内处理器702要使用数据的可能性,并且基于将来使用的可能性,3级缓存或共享缓存能够在数据移出到主存储器之前将数据保存预定的时间长度。标记组件706还能够在数据上设置指示将来使用的相对或绝对可能性的指示符。

[0035] 鉴于上述示例性系统,参照图8的流程图可以更好地理解可以根据所描述的主题实现的方法。尽管该方法被示出和描述为一系列的块,但所要求保护的主题不受块的顺序的限制,因为一些块可以以不同的顺序发生和/或与本文所描绘的其他块同时发生。并不是所有示出的块都要求实现下文所描述的方法。

[0036] 图8是用于预取和移出不使用的数据以防止缓存垃圾的方法的实施例的流程图。方法800能够起始于802,其中该方法包括从系统存储器中预取数据集合到与处理器相关联的处理器缓存。在一些实施例中,预取能够由处理器缓存中的预取器来执行,该处理器缓存能够是2级缓存。预取不仅加载由处理器指示的存储地址处的数据,还加载与所请求的数据相关的存储地址处的数据。使用预测性启发法,预取器能够预测将来可能使用哪些数据,这些数据能够包括在相似的地址空间上的数据,或在物理上或逻辑上接近所请求数据的数据。该相关的数据能够被预加载到缓存中以允许处理器更快地访问数据。

[0037] 在804处,该方法能够包括利用第一部分数据是已使用的数据的指示来标记该数据集的第一部分数据。如果数据被处理器使用,并且在不久的将来不太可能被再次使用,则数据能够被标记用于移回到主存储器或磁盘。类似地,在806处,该方法能够包括利用第二部分数据是可能要执行的数据的指示来标记该数据集的第二部分数据,并且在808处,该方法包括将第二部分数据移出到共享缓存。可能不立即使用、但在不久的将来可能会使用的数据能够被发送到共享缓存(例如,3级缓存)。该共享缓存相比主存储器提供对数据的更快访问,并且移出数据允许处理器处的缓存(2级缓存)保留可能比被移出的数据更快或更早使用的数据。

[0038] 本文所描述的技术能够应用于期望促进执行预取和移出以避免缓存垃圾的任何装置。考虑到了用于结合各种实施例使用的各种手持式、便携式和其他计算设备以及计算对象,即,设备可能希望与多个客户设备或虚拟机共享计算资源的任何地方。因此,以下在图9中描述的通用远程计算机是一个示例,并且能够利用具有网络/总线互操作性和交互的任何客户端来实现所公开的主题。因此,能够在网络化托管服务的环境中实现所公开的主题,其中涉及很少或最少的客户端资源,例如,其中客户端设备仅用作到网络/总线的接口的网络环境,诸如放置在设备中的对象。

[0039] 图9示出了其中能够实现所公开的主题的一些方面的计算系统环境900的示例,尽管计算系统环境900仅是设备的计算环境的一个示例。

[0040] 图9是用于实现所公开的主题的示例性设备,其包括计算机910形式的通用计算设备。计算机910的组件可以包括处理单元920、系统存储器930以及系统总线921,所述系统总线将包括系统存储器在内的各种系统组件耦接到处理单元920。

[0041] 计算机910包括各种计算机可读介质。计算机可读介质能够是计算机910能够访问的任何可用介质。系统存储器930可以包括易失性和/或非易失性存储器(诸如只读存储器(ROM)和/或随机存取存储器(RAM))形式的计算机存储介质。

[0042] 计算机910能够使用到一个或多个其他远程计算机(例如,远程计算机970)的逻辑连接在网络或分布式环境中操作,该远程计算机又能够具有与设备910不同的媒介能力。图9中所描绘的逻辑连接包括网络971,其例如是局域网(LAN)或广域网(WAN),但也能够包括有线或无线的其他网络/总线。当在LAN网络环境中使用时,计算机910能够通过网络接口或适配器连接到LAN971。当在WAN网络环境中使用时,计算机910通常能够包括通信组件,例如,调制解调器,或用于在诸如因特网之类的WAN上建立通信的其他装置。

[0043] 图10是示例性网络或分布式计算环境的示意图。分布式计算环境包括计算对象1010、1012等,以及计算对象或设备1020、1022、1024、1026、1028等,其可以包括程序、方法、数据存储、可编程逻辑等,如由应用程序1030、1032、1034、1036、1038和数据存储1040所表示的。

[0044] 每个计算对象1010、1012等以及计算对象或设备1020、1022、1024、1026、1028等能够通过通信网络1042与一个或多个其他计算对象1010、1012等以及计算对象或设备1020、1022、1024、1026、1028等进行通信。通信网络1042可以包括向图10的系统提供服务的其他计算对象和计算设备,和/或可以表示多个互连的网络。能够使用大量网络拓扑和网络基础设施,例如客户端/服务器,端对端或混合架构。在客户端/服务器架构中,特别是网络系统中,客户端通常是访问由另一台计算机(例如服务器)提供的共享网络资源的计算机。

[0045] 由存储器和处理器执行的进程都能够是一个组件。作为另一个示例,架构能够包括电子硬件(例如,并行或串行晶体管)、处理指令和处理器的布置,其以适合于电子硬件的布置的方式来实现处理指令。

[0046] 所公开的主题能够被实现为方法、装置或制品,使用常规的制造方法、编程或工程技术来生产硬件、固件、软件或其任何合适的组合以控制电子设备来实现所公开的主题。计算机可读介质能够包括硬件介质、软件介质、非临时性介质或传输介质。

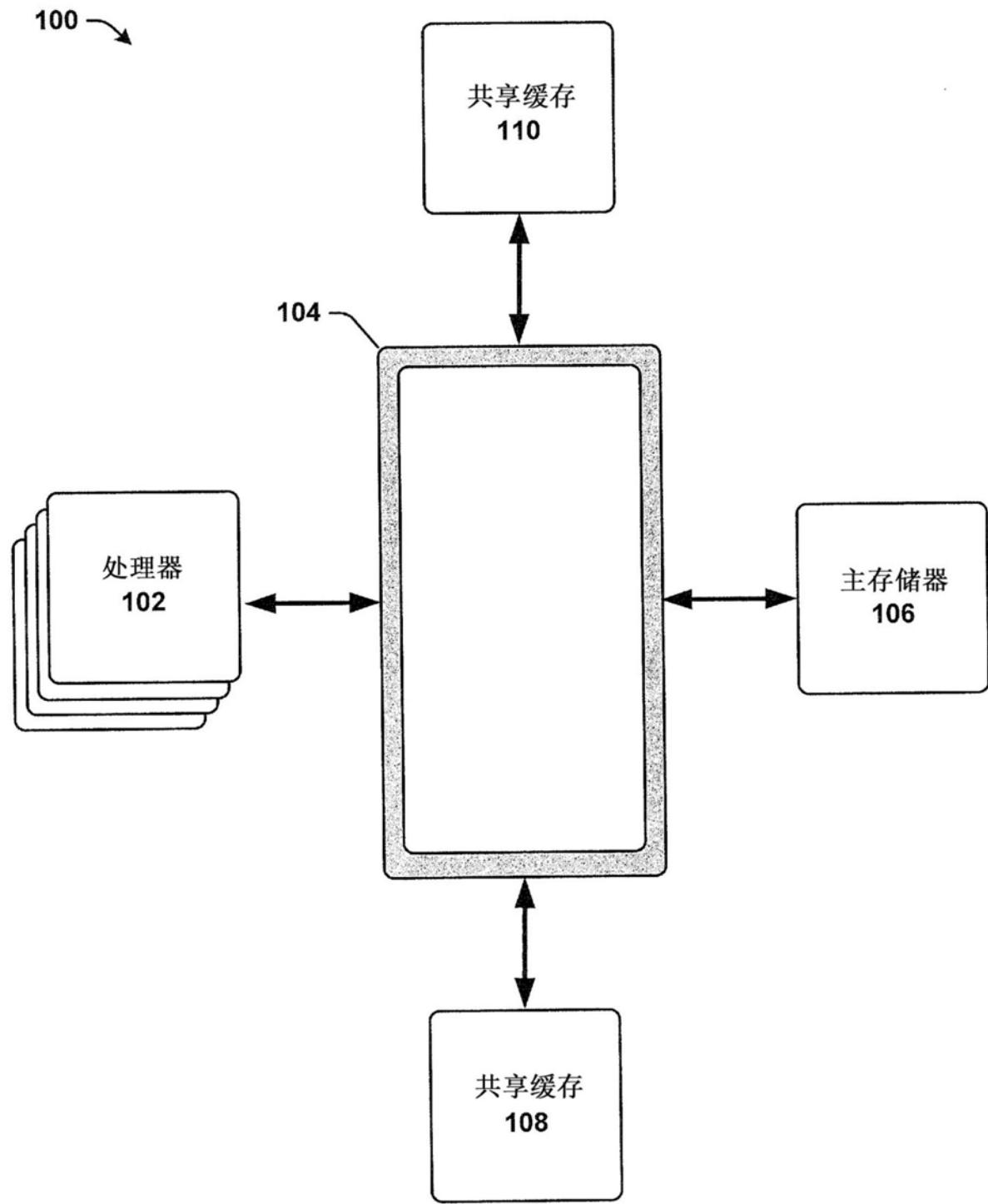


图1

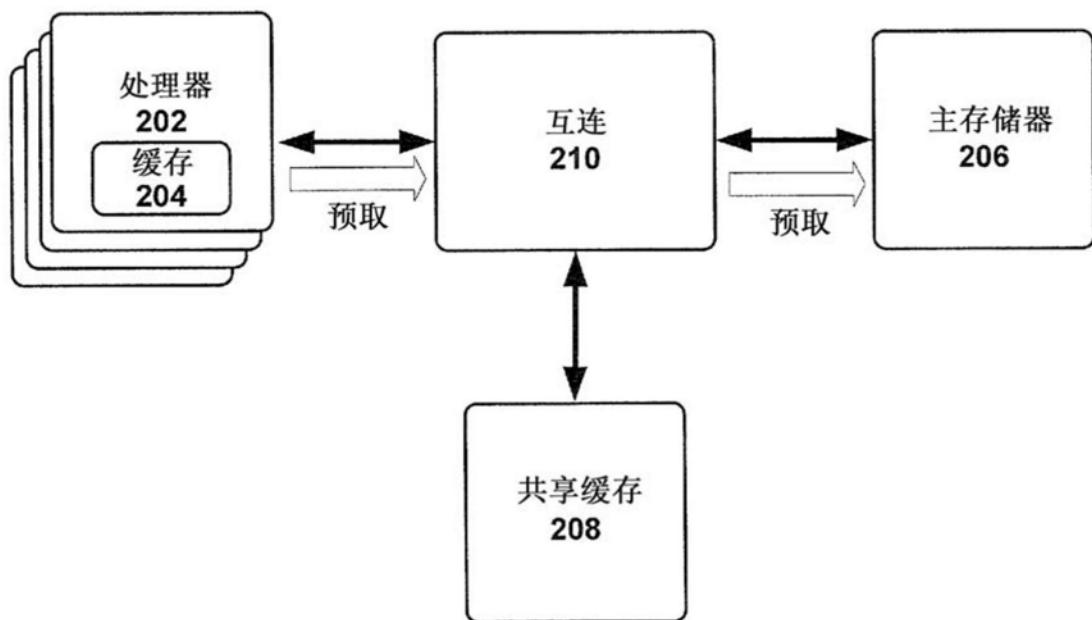


图2

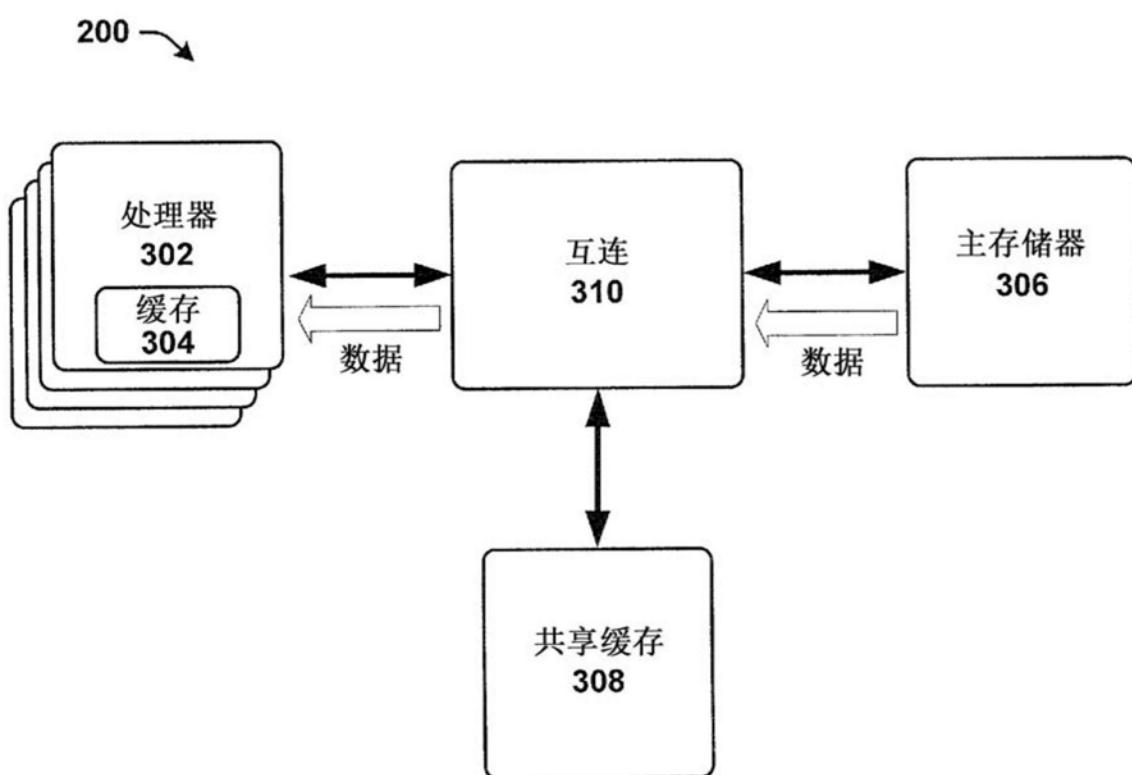


图3

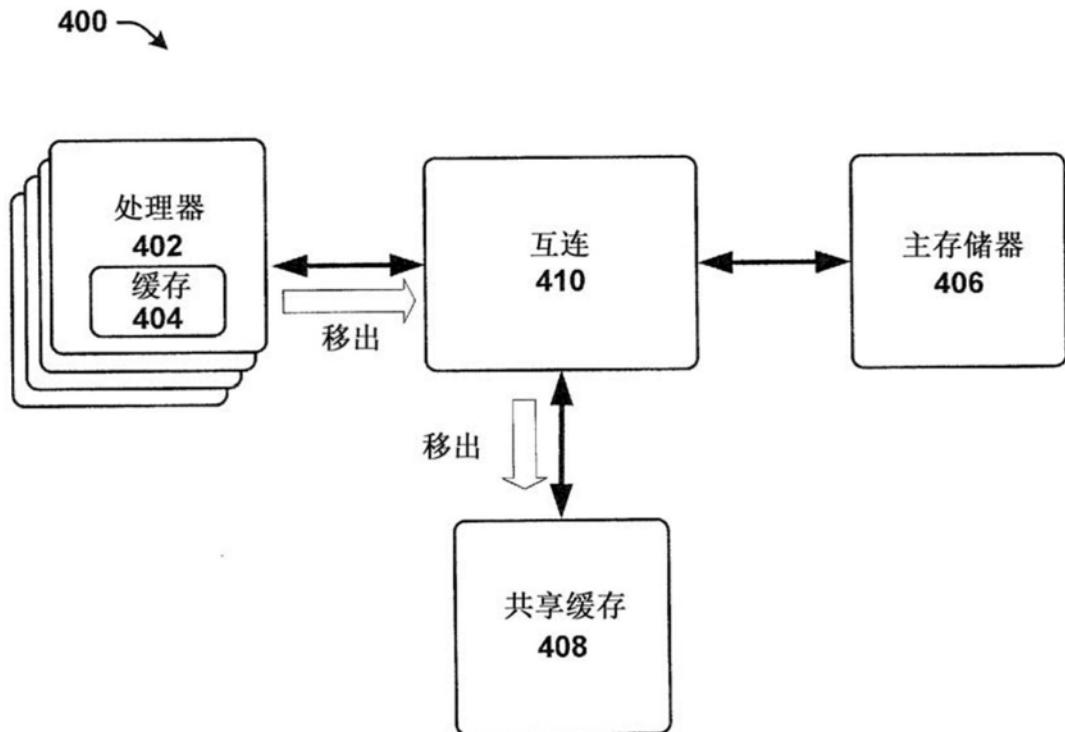


图4

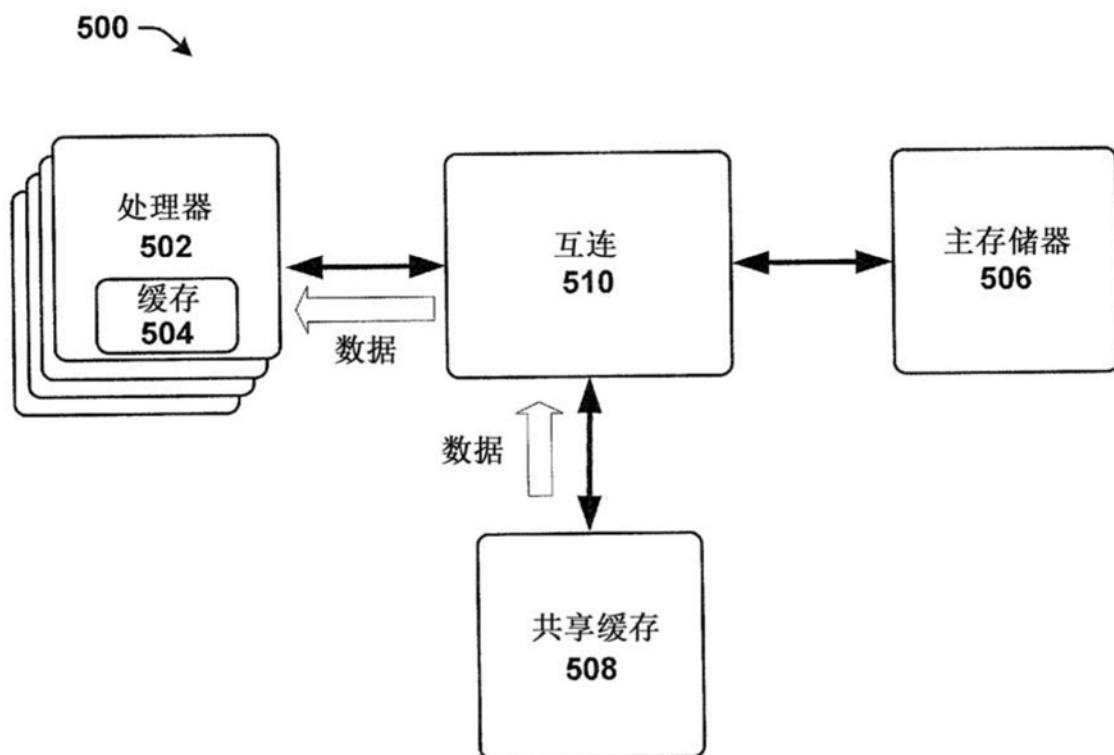


图5

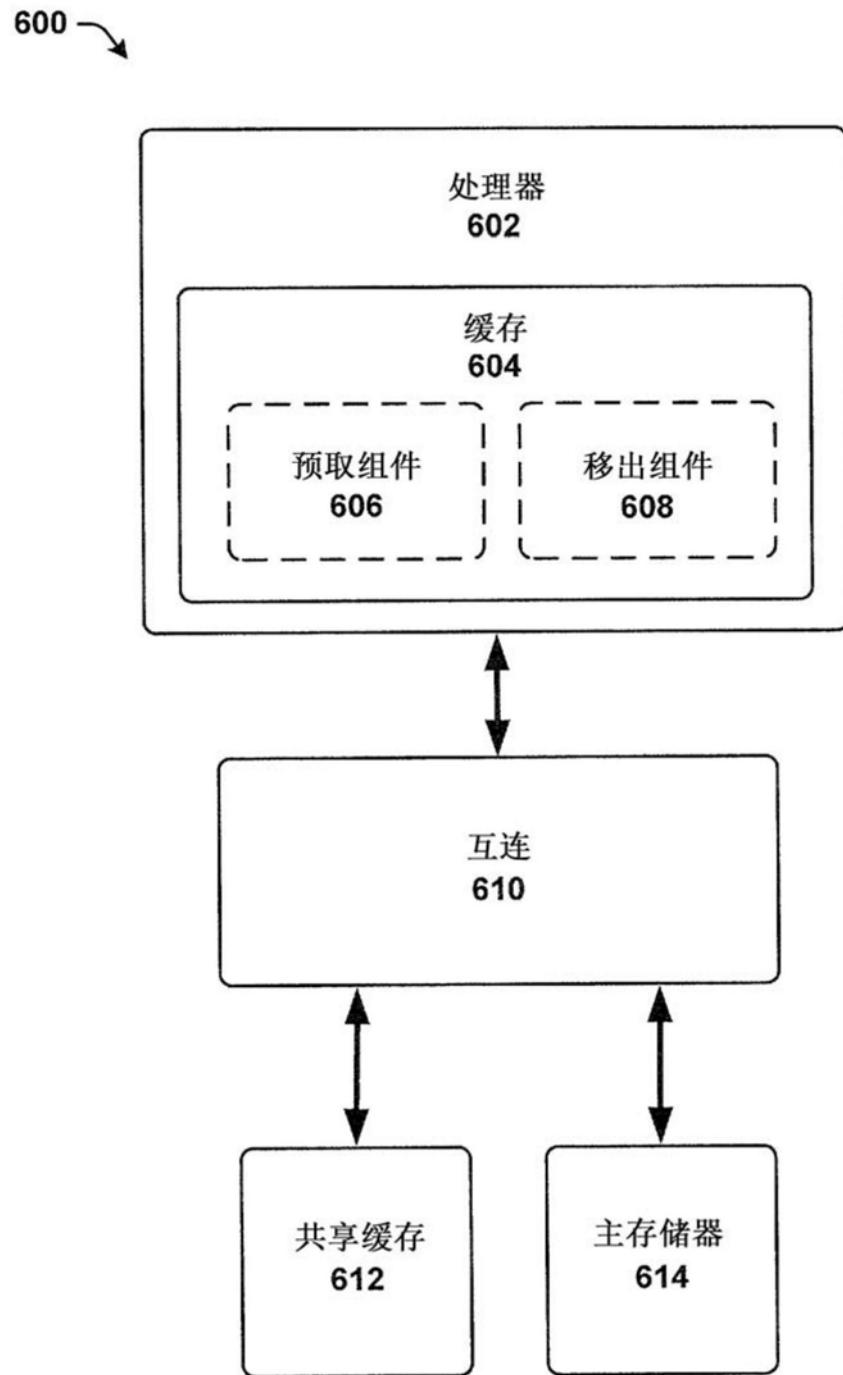


图6

700 ↘

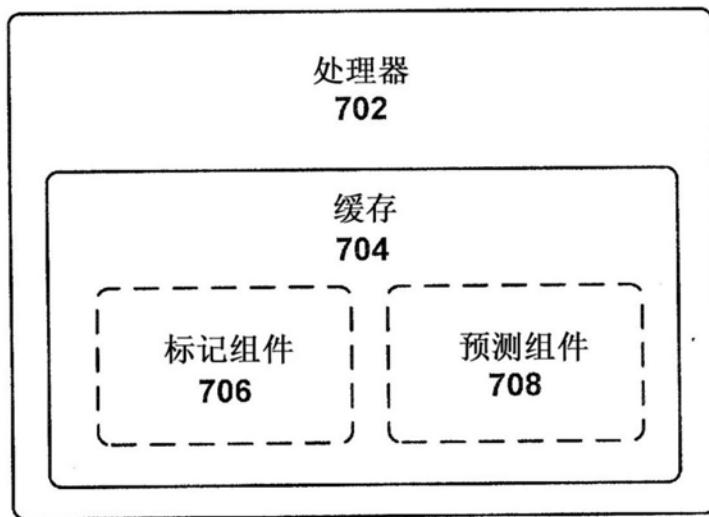


图7

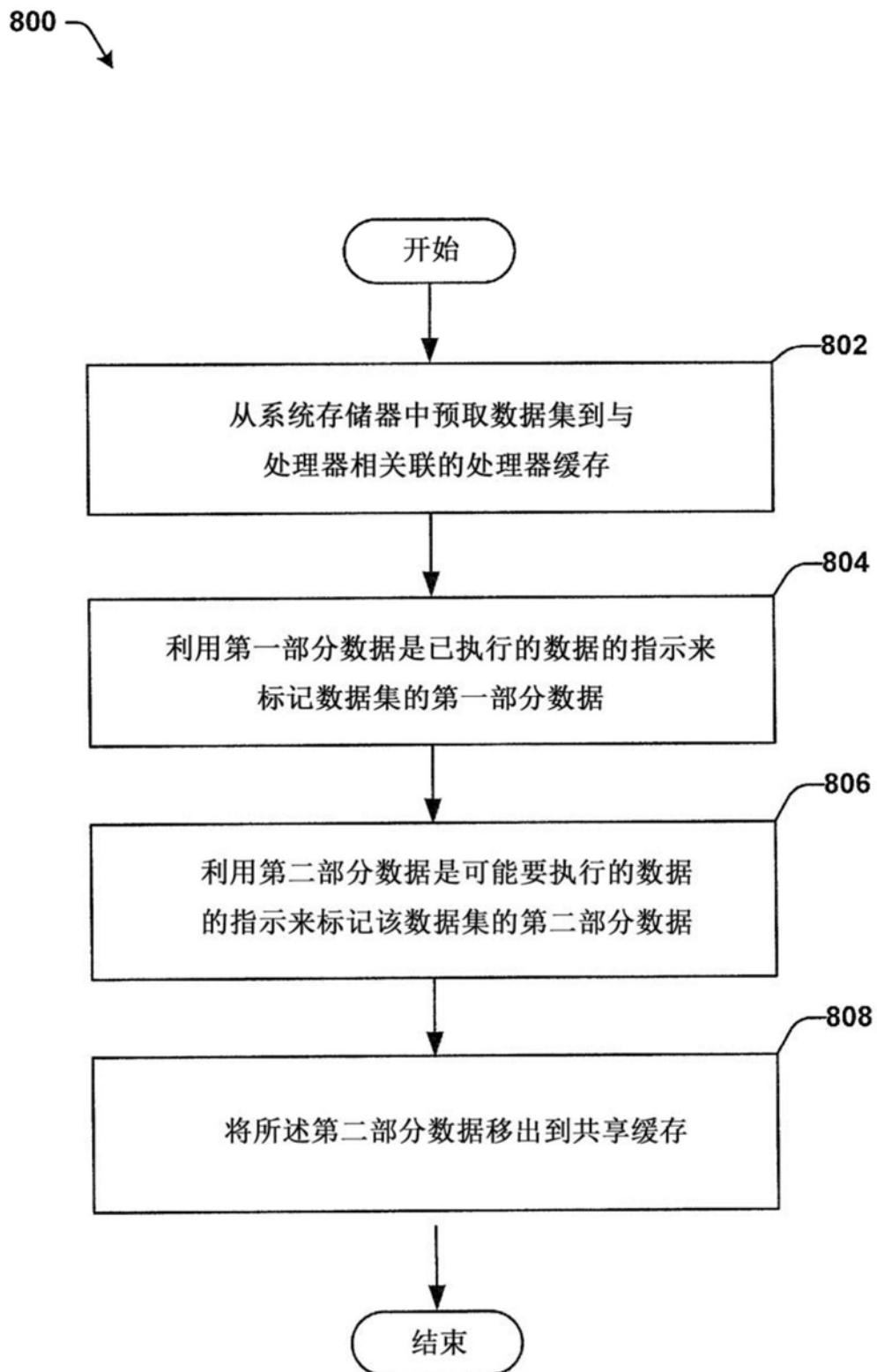


图8

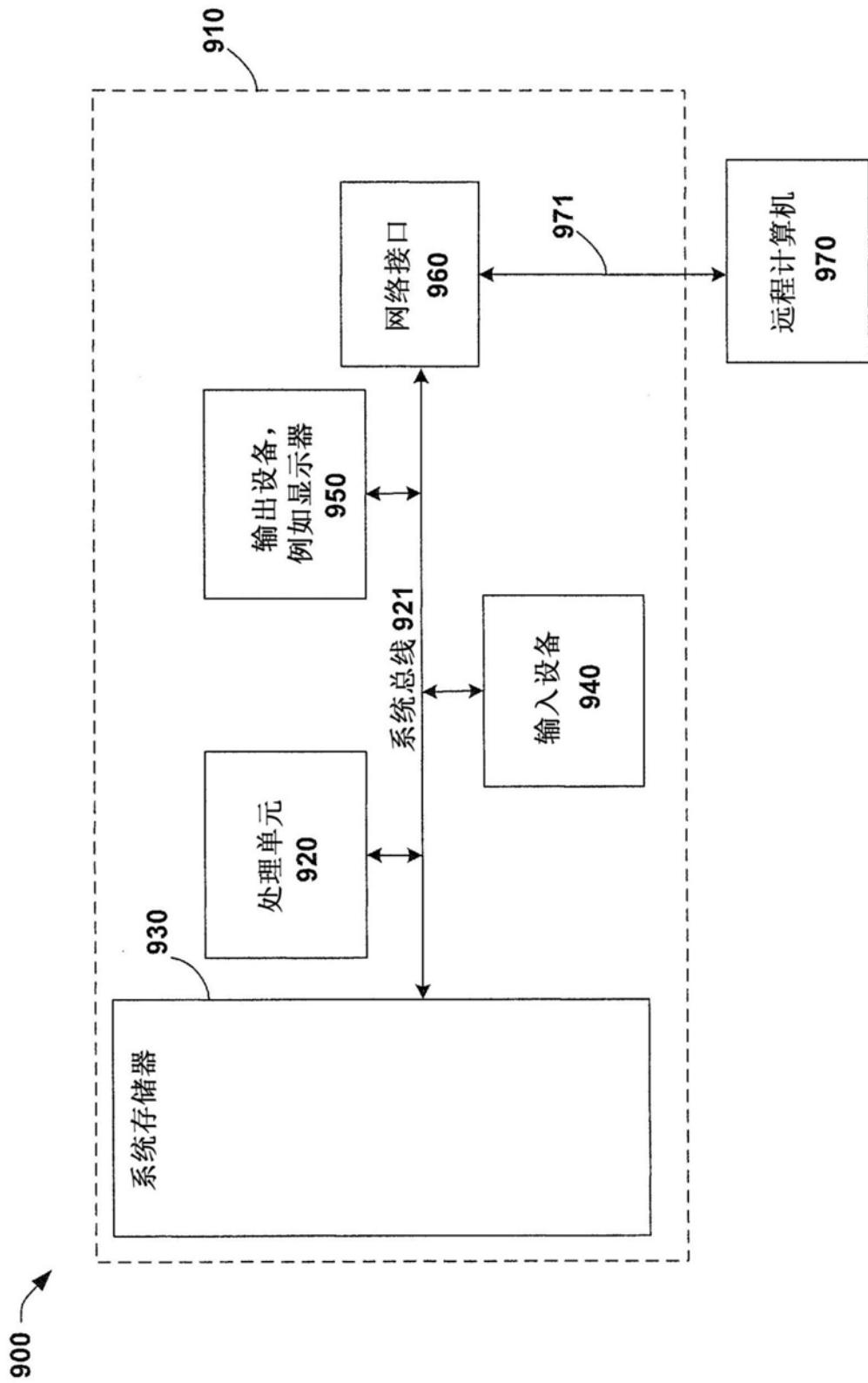


图9

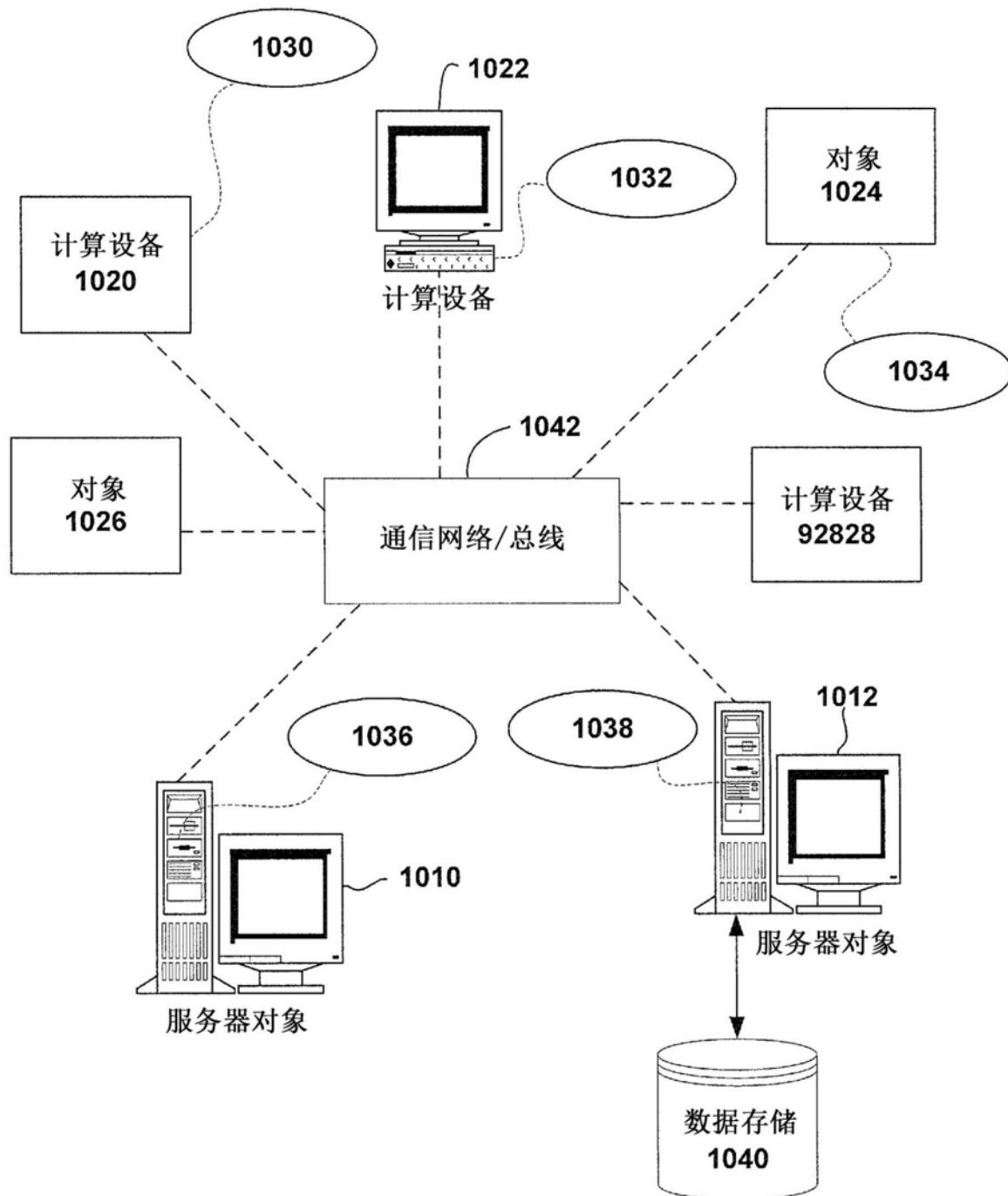


图10