



# [12] 发明专利申请公开说明书

[21] 申请号 03147463.2

[43] 公开日 2004年8月18日

[11] 公开号 CN 1521625A

[22] 申请日 2003.7.11 [21] 申请号 03147463.2

[30] 优先权

[32] 2002.7.12 [33] JP [31] 2002-204167

[71] 申请人 日本电气株式会社

地址 日本东京都

[72] 发明人 山崎茂雄 爱野茂幸

[74] 专利代理机构 中科专利商标代理有限责任公

司

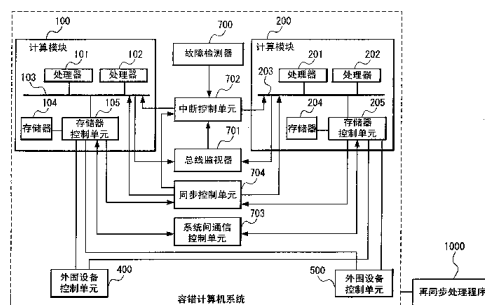
代理人 朱进桂

权利要求书 8 页 说明书 20 页 附图 10 页

[54] 发明名称 容错计算机系统，其再同步方法，及其再同步程序

[57] 摘要

在包括多个具有一个处理器和一个存储器的计算模块并且每个计算模块彼此同步地执行同一指令串的锁步同步容错计算机系统中，当检测到每个计算模块中的相应处理器之间对一个外部总线的接入状态不一致时，如果在包括每个计算模块的系统中没有检测到故障，那么在对所有处理器产生一个中断以便在执行使计算模块之间的指令执行状态一致的延迟调节之后，相对于每个计算模块执行恢复同步操作的处理。



- 1.一种锁步同步容错计算机系统，该系统包括多个具有处理器和存储  
5 器的计算模块、每个计算模块彼此同步地处理同一指令串，其中  
当检测到每个所述计算模块中的所述处理器之间对外部总线的接入  
状态不抑制时，如果在包括每个所述计算模块的系统中没有检测到故障，  
那么在对所有所述处理器产生一个中断以执行用于使所述计算模块之间的  
指令执行状态一致的延迟调节之后，相对于每个所述计算模块执行恢  
10 复同步操作的处理。
- 2.根据权利要求1所述的容错计算机系统，进一步包括：  
故障检测器，用于监测系统中故障存在/不存在；  
总线监测器，用于监测每个所述计算模块中的每个所述处理器的对  
外部总线的接入状态；  
15 中断控制装置，用于在所述总线监测器检测到每个所述计算模块中的  
每个所述处理器对外部总线的接入状态不同步的情况下，当所述故障  
检测器没有检测到故障时，产生用于将检测结果通知每个所述处理器的  
中断；  
连接到每个所述计算模块的系统间通信控制装置，用于在每个所述  
20 计算模块中的所述处理器之间通知指令执行状态；和  
连接到每个所述计算模块的同步控制装置，用于在进行了使每个所  
述计算模块中的指令执行状态一致的延迟调节之后，产生一个用于恢复  
所有所述计算模块的同步操作的重置信号。
- 3.根据权利要求2所述的容错计算机系统，其中  
25 所有所述处理器中的每一个包括用于计算处理器中执行的指令的数  
量的指令数计数器，当所述处理器接收到来自所述中断控制装置的中断  
并且转移到再同步的处理器管理模式时，所述指令数计数器不能操作；  
每个所述处理器将它自己的指令数计数器的值与从其它计算模块中  
每个对应处理器接收的指令数计数器的值比较；

不包括指示所有计数器中最大值的指令数计数器的所述计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送通知，和

- 5 包括指示最大值的指令数计数器的所述计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，指令所述同步控制装置产生用于致使所有计算模块恢复彼此同步操作的重置信号。

4.根据权利要求3所述的容错计算机系统，其中

- 在其中所述指令数计数器值不是最大的所述计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中，

所述处理器设置在用于在执行了一个指令之后转移到所述处理器管理模式的逐步执行模式，和

所述处理器重复逐步执行模式的处理，直到指令数计数器值与最大指令数计数器值一致。

- 15 5.根据权利要求2所述的容错计算机系统，其中

所有接收到来自所述中断控制装置的中断的所述处理器中的每一个转移到一种其中用于计算处理器中执行指令的数量的指令数计数器不能操作并且保存和存储了一个程序计数器值的再同步处理的处理器管理模式；

- 20 每个所述处理器读出每个处理器的指令数计数器值和所述保存的程序计数器值，并且将值发送到所有其它所述计算模块；

每个所述处理器将指令数计数器值与从其它计算模块中的每个处理器接收的指令数计数器值比较；

- 所述指令数计数器值不是最大的所述计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送通知，和

包括指示最大值的指令数计数器的所述计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，指令所述同步控制装置产生用于致使所有计算模块恢复彼此同步操作的重置信号。

6.根据权利要求 5 所述的容错计算机系统，其中

在其中所述指令数计数器值不是最大的所述计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中，

5 将所述处理器设置在用于在执行到一个指定指令串中的一个特定位置的指令之后，转移到所述处理器管理模式的断点指定执行模式；

将由从包括指示最大值的指令数计数器的所述计算模块接收的程序计数器值指示的指令位置指定为指令串中的所述特定位置，和

在执行指令串达到一个指定指令串中的所述特定位置之后，所述处理器转移到处理器管理模式。

10 7.根据权利要求 3 所述的容错计算机系统，其中

在读出了一个程序执行状态之后，包括指示最大值的指令数计数器的所述计算模块等待来自其它计算模块的所述通知；

15 在执行延迟调节处理之后，所有不包括指示最大值的指令数计数器的其它计算模块读出程序执行状态，并且将程序执行状态与延迟调节处理完成的通知一起发送到包括指示最大值的指令数计数器的计算模块，和

20 包括指示最大值的指令数计数器的计算模块比较所有计算模块的程序执行状态，并且当所有程序执行状态彼此一致时，发出产生用于恢复同步操作的重置信号的指令，并且当检测到其程序执行状态不一致的计算模块时，在执行了切断和无效计算模块的处理之后，发出产生用于恢复同步操作的重置信号的指令。

8.根据权利要求 2 所述的容错计算机系统，其中

提供了多对所述故障检测器，所述总线监测器，所述中断控制装置，所述系统间通信控制装置，和所述同步控制装置。

25 9.一种锁步系统容错计算机系统中的再同步方法，该锁步系统容错计算机系统包括多个具有一个处理器和一个存储器的计算模块、每个计算模块彼此同步地处理同一指令，该方法包括步骤：

当检测到每个所述计算模块中所述处理器之间对外部总线接入状态不一致时，如果在包括每个所述计算模块的系统中没有检测到故障，那

么对所有所述处理器产生中断，和

在执行了使所述计算模块之间指令执行状态一致的延迟调节之后，相对于每个所述计算模块执行恢复同步操作的处理。

5 10.根据权利要求 9 所述的容错计算机系统的再同步方法，进一步包括步骤：

监测系统中存在/不存在故障；

监测每个所述计算模块中的每个所述处理器对外部总线的接入状态；

10 当检测到每个所述计算模块中每个所述处理器对外部总线的接入状态不一致时，如果没有检测到故障，那么产生中断，以便将检测结果通知所有所述处理器，和

在执行了使所述计算模块之间的指令执行状态一致的延迟调节之后，产生用于执行恢复所有所述计算模块的同步操作的处理的重置信号。

11.根据权利要求 10 所述的再同步方法，进一步包括步骤：

15 所有接收到所述中断的所述处理器中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作的再同步处理的处理器管理模式；

每个所述处理器将读出的指令数计数器值与从其它计算模块中每个处理器接收的指令数计数器值比较；

20 不包括指示所有计数器之间最大值的指令数计数器的所述计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送通知，和

25 包括指示最大值的指令数计数器的所述计算模块，等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，产生用于致使所有计算模块恢复彼此同步操作的重置信号。

12.根据权利要求 11 所述的再同步方法，其中

在其中所述指令数计数器值不是最大的所述计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将所述处理器设置在用于在执行了一个指令之后转移到所述处理器管理模式的逐步执行模式，和

所述处理器重复逐步执行模式的处理，直到指令数计数器值与最大指令数计数器值一致。

5 13.根据权利要求 10 所述的再同步方法，其中

所有接收到所述中断的所述处理器中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作并且保存和存储程序计数器值的再同步处理的处理器管理模式；

10 每个所述处理器读出每个处理器的指令数计数器值和所述保存的程序计数器值，并且将值发送到所有其它所述计算模块；

每个所述处理器将指令数计数器值与从其它计算模块中的每个处理器接收的指令数计数器值比较；

15 所述指令数计数器值不是最大的所述计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送通知，和

包括指示最大值的指令数计数器的所述计算模块等待接收来自所有其它计算模块的通知，当接收到所有通知时，产生用于致使所有计算模块恢复彼此同步操作的重置信号。

14.根据权利要求 13 所述的再同步方法，其中

20 在其中所述指令数计数器值不是最大的所述计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将所述处理器设置在用于在执行到一个指定指令串中的一个特定位置的指令之后转移到所述处理器管理模式的断点指定执行模式；

25 将由从包括指示最大值的指令数计数器的所述计算模块接收的程序计数器值指示的指令位置指定为指令串中所述特定位置，和

在执行指令串达到指定指令串中所述特定位置之后，所述处理器转移到处理器管理模式。

15.根据权利要求 11 所述的再同步方法，其中

在读出程序执行状态之后，包括指示最大值的指令数计数器的所述

计算模块等待来自其它计算模块的所述通知；

在执行了延迟调节处理之后，不包括指示最大值的指令数计数器的所有其它计算模块读出程序执行状态，并且将程序执行状态与延迟调节处理完成的通知一起发送到包括指示最大值的指令数计数器的计算模块，和

5

包括指示最大值的指令数计数器的计算模块比较所有计算模块的程序执行状态，并且当所有程序执行状态彼此一致时，发出产生恢复同步操作的重置信号的指令，并且在检测到程序执行状态不一致的计算模块时，在执行了切断和无效计算模块的处理之后，发出产生用于恢复同步操作的重置信号的指令。

10

16.一种用于执行锁步同步容错计算机系统的再同步处理的再同步程序，该锁步同步容错计算机系统包括多个具有一个处理器和一个存储器的计算模块、每个计算模块彼此时钟同步地处理同一指令串，包括功能：

15

当检测到每个所述计算模块中的所述处理器之间的对外部总线接入状态不一致时，如果在包括每个所述计算模块的系统中没有检测到故障，对所有所述处理器产生一个中断，和

在执行了用于使所述计算模块之间的指令执行状态一致的延迟调节之后，致使每个所述处理器恢复同步操作。

20

17.根据权利要求 16 所述的再同步程序，进一步包括功能：

监测系统中存在/不存在故障；

监测每个所述计算模块中的每个处理器对外部总线的接入状态，

当检测到每个所述计算模块中的每个所述处理器对外部总线的接入状态中的不一致时，如果在所述故障监测中没有检测到故障，那么产生用于将检测结果通知每个所述处理器的中断，和

25

在执行了使所述计算模块之间的指令执行状态一致的延迟调节之后，产生用于恢复所有所述计算模块的同步操作的重置信号。

18.根据权利要求 17 所述的再同步程序，包括：

所有接收到所述中断的所述处理器中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作的再同步处理的处

理器管理模式的功能；

每个所述处理器将读出的指令数计数器值与从其它计算模块中的每个处理器接收的指令数计数器值比较的功能；

5 不包括指示所有计数器中最大值的指令数计数器的所述计算模块进行执行指令的延迟调节直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送通知的功能，和

10 包括指示最大值的指令数计数器的所述计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，发出产生用于致使所有计算模块恢复同步操作的重置信号的指令的功能。

19.根据权利要求 18 所述的再同步程序，包括：

在其中所述指令数计数器不是最大的所述计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

15 将所述处理器设置在用于在执行了一个指令之后转移到所述处理器管理模式的逐步执行模式的功能；

所述处理器重复逐步执行模式的处理，直到指令数计数器值与最大指令数计数器值一致的功能。

20.根据权利要求 18 所述的再同步程序，包括：

20 接收到所述中断的所有所述处理器中的每一个转移到其中用于计算处理器中执行的指令数量的指令数计数器不能操作并且保存和存储程序计数器值的再同步处理的处理器管理模式的功能；

每个所述处理器读出每个处理器的指令数计数器值和所述保存的程序计数器值，并且将值发送到所有其它所述计算模块的功能；

25 每个所述处理器将读出的指令数计数器值与从其它计算模块中的每个处理器接收的指令数计数器值比较的功能；

所述指令数计数器值不是最大的所述计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送通知的功能，和

包括指示最大值的指令数计数器的所述计算模块等待接收来自所有

其它计算模块的通知，并且当接收到所有通知时，发出产生用于致使所有计算模块恢复同步操作的重置信号的指令的功能。

21.根据权利要求 20 所述的再同步程序，包括：

在其中所述指令数计数器值不是最大的所述计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将所述处理器设置在用于在执行所述处理器到达一个指定指令串中的一个特定位置的指令之后，转移到所述处理器管理模式的断点指定执行模式的功能；

将由从包括指示最大值的指令数计数器的所述计算模块接收的程序计数器值指示的一个指令位置指定为指令串中的所述特定位置的功能，和

在所述处理器执行指令串到达指定指令串中的所述特定位置之后，将所述处理器转移到处理器管理模式的功能。

22.根据权利要求 20 所述的再同步程序，包括：

包括指示最大值的指令数计数器的所述计算模块，在读出程序执行状态之后，等待来自其它计算模块的所述通知的功能，

所有不包括指示最大值的指令数计数器的其它计算模块读出程序执行状态，并且在执行了延迟调节处理之后，把程序执行状态与延迟调节处理完成的通知一起发送到包括指示最大值的指令数计数器的计算模块的功能，和

包括指示最大值的指令数计数器的计算模块比较所有计算模块的程序执行状态，并且当所有程序执行状态彼此一致时，发出产生用于恢复同步操作的重置信号的指令，并且当检测到程序执行状态不一致的计算模块时，在执行了切断和无效计算模块的处理之后，发出产生用于恢复同步操作的重置信号的指令的功能。

## 容错计算机系统，其再同步方法，及其再同步程序

### 5 技术领域

本发明涉及一种通过彼此时钟同步的多个计算模块、以完全相同的方式处理同一指令串的锁步同步容错计算机系统。更具体地讲，涉及一种在计算模块之间发生同步错误时（锁步脱离），实现快速再同步处理的容错计算机系统 and 高速再同步控制方法。

10

### 背景技术

利用惯用锁步同步容错计算机系统，当多个计算模块之间同时执行同一指令串时，检测到一个计算模块由于故障或其它外部或内部因素而具有与其它计算模块不同的输出时，采取以下的对策。在下面，将检测到的未能与其它计算模块同步操作的计算模块称为失步状态的计算模块。

更具体地讲，采取切断锁步从操作状态脱离的计算模块，并且在需要时根据造成失步的因素替换该计算模块，或当不需要替换时，根据将计算模块重整到操作状态的需要，执行重新初始化处理之类的处理中的一个的措施。

在惯用锁步同步容错计算机系统中，在上述重整到操作状态时，由于无论是否替换失步状态的计算模块，为了使该计算模块与其它继续操作的计算模块同步以再执行同一处理，要把操作状态的计算模块占有的所有存储器数据，复制到要在其重整时重整的计算模块占有的存储器中。

在惯用锁步同步容错计算机系统中，在执行了失步状态的计算模块的替换，根据造成失步等的部分进行重新初始化处理等等之后，当所述计算模块再次重整到操作状态时，在操作状态的其它计算模块暂停了很长时间。

更具体地讲，惯用锁步同步容错计算机系统存在以下的问题：在对

失步状态的计算模块进行重整处理的同时，整个容错计算机系统使它的操作暂停一个很长的时间（一般为3至5秒，或数分钟）。

原因是，为了将失步状态的计算模块整合到操作状态，始终要把所有的存储内容从继续操作的计算模块复制到要重整的计算模块中。

- 5 当在复制处理过程中一个正常计算模块的操作继续进行时，正常计算模块的存储器内容也有可能在复制处理过程中改变，从而不可能适当地执行复制。为了避免这种情况，暂时停止操作状态的计算模块，以防止其存储器内容的更新。

10 由于一个当今的计算模块中的存储容量高达数千兆字节，复制整个存储区需要很长的时间。

在一个锁步同步容错计算机系统中，计算模块之间的失步状态是由于各种原因产生的。

- 15 第一种情况是，发生在计算模块内的固定故障。在这种情况下，应当更换具有故障的计算模块，并且当把被替换的计算模块重整到操作系统中时，需要复制在操作状态的计算模块的存储器中的所有数据。

在一个锁步同步容错计算机系统中，除了上述固定故障之外，因为尽管一个计算模块操作是正常的，但是由于计算模块中的每个单元的制造中的误差使得该计算模块操作在不同的定时，或因为 $\alpha$ -射线之类的效应造成的存储器的可自动修复间歇性故障，也可能发生失步状态。

- 20 在这些情况中，由于固定故障没有发生在计算模块本身中，因而不需要从根本上更换模块，并且通过使其处理再次与操作中的其它计算模块的处理同步，以重整所述计算模块，从而可以将整个容错计算机系统恢复到正常操作状态。

## 25 发明内容

本发明的一个目的是提供一种使因不是固定故障的原因造成了锁步脱离的计算模块能够以比惯用系统更高的速度再次整合到操作状态，从而大大减少整合处理造成的系统操作的暂停时间的容错计算机系统，其再同步方法，及其再同步程序。

本发明的另一个目的是要提供一种通过减少上述重整处理的时间实现提高系统可用性的容错计算机系统，其再同步方法，及其再同步程序。

根据本发明的第一方面，一种锁步同步容错计算机系统包括多个具有处理器和存储器的计算模块，在计算机系统中每个计算机模块彼此同步地处理同一指令串，其中当检测到每个计算模块中的处理器之间对外部总线的接入状态不一致时，如果在包括每个计算模块的系统中没有检测到故障，那么在对所有处理器产生一个中断以执行使计算模块之间的指令执行状态一致的延迟调节之后，相对于每个计算模块执行同步的恢复操作的处理。

10 在本优选的结构中，容错计算机系统进一步包括监测系统中故障存在/不存在的故障检测器；

总线监测器，用于监测每个计算模块中的每个处理器对外部总线的接入状态；

15 中断控制装置，用于在总线监测器检测到每个计算模块中的每个处理器对外部总线的接入状态不同步的情况下，当故障检测器没有检测到故障时，产生一个将检测结果通知每个处理器的中断；

20 连接到每个计算模块的系统间通信控制装置，用于在每个计算模块中的处理器之间通知指令执行状态，和连接到每个计算模块的同步控制装置，用于在进行了使每个计算模块中的指令执行状态一致的延迟调节之后，产生一个恢复所有计算模块同步操作的重置信号。

在另一个优选结构中，所有处理器中的每一个包括一个用于计算处理器中执行的指令的数量的指令数计数器，当处理器接收到来自中断控制装置的中断时指令数计数器不能操作，并且转移到一个用于再同步的处理器管理模式；

25 每个处理器将它本身的指令数计数器的值与从其它计算模块中每个对应的处理器接收的指令数计数器的值比较；

不包括指示所有计数器中最大值的指令数计数器的计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送一

个通知，并且包括指示最大值的指令数计数器的计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，指令同步控制装置产生一个致使所有计算模块恢复彼此同步的操作的重置信号。

5 在另一种优选结构中，在指令数计数器值不是最大的计算模块执行指令直到指令数计数器值与最大指令数计数器值一致的延迟调节中，将处理器设置在用于在执行了一个指令之后转移到处理器管理模式的逐步执行模式，并且处理器重复逐步执行模式的处理，直到指令数计数器值与最大指令数计数器值一致。

10 在另一个优选结构中，从中断控制装置接收到中断的所有处理器中的每一个转移到进行再同步处理的处理器管理模式，在处理器管理模式中，用于计算处理器中执行的指令数的指令数计数器不能操作，并且保存和存储一个程序计数器值；

每个处理器读出每个处理器的指令数计数器值和保存的程序计数器值，并且把值发送到所有其它计算模块；

15 每个处理器将指令数计数器值与一个从其它计算模块中每个处理器接收的指令数计数器值比较；

指令数计数器值不是最大的计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送一个通知，并且包括指示最大值的指令数计数器的计算模块等待接收来自所有其它计算模块的通知，当接收到所有通知的时，指令同步控制装置产生一个用于致使所有计算模块恢复彼此同步的操作的重置信号。

25 在另一个优选结构中，在指令数计数器值不是最大的计算模块执行指令直到指令数计数器值与最大指令数计数器值一致的延迟调节中，处理器被设置在用于在执行到一个在指定的指令串中的一个特定位置的指令之后转移到处理器管理模式的断点指定执行模式；

将由从包括指示最大值的指令数计数器的计算模块接收的程序计数器值指示的指令位置指定为指令串中的特定位置，并且在执行指令串到达指定的指令串中的特定位置之后，处理器转移到处理器管理模式。

在另一个优选结构中，在读出了一个程序执行状态后，包括指示最大值的指令数计数器的计算模块等待来自其它计算模块的通知；

在执行了延迟调节处理之后，所有不包括指示最大值的指令数计数器的其它计算模块读出一个程序执行状态，并且将程序执行状态与延迟调节处理完成通知一起发送到包括指示最大值的指令数计数器的计算模块，并且包括指示最大值的指令数计数器的计算模块比较所有计算模块的程序执行状态，并且当所有计算模块的程序执行状态一致时，发出产生用于恢复同步操作的重置信号的指令，并且当检测到程序执行状态不一致的一个计算模块时，在执行了切断和使该计算模块无效的处理之后，发出产生用于恢复同步操作的重置信号的指令。

在另一个优选结构中，提供了多对故障检测器，总线监测器，中断控制装置，系统间通信控制装置，和同步控制装置。

根据本发明的另一方面，一种锁步同步容错计算机系统中的再同步方法，该锁步同步容错计算机系统包括多个具有一个处理器和一个存储器的计算模块并且每个计算模块彼此同步地处理同一指令串，该方法包括步骤：当检测到每个计算模块中的处理器之间对一个外部总线的接入状态不一致时，如果在包括每个计算模块的系统中没有检测到故障，那么对所有处理器产生中断，并且在执行了使计算模块之间的指令执行状态一致的执行延迟调节之后，执行相对于每个计算模块的恢复同步操作的处理。

在优选的结构中，容错计算机系统的再同步方法进一步包括步骤：监测系统中故障存在/不存在；

监测每个计算模块中每个处理器对外部总线的接入状态；

当检测到每个计算模块中的每个处理器对外部总线的接入状态不一致时，如果没有检测到故障，那么产生一个将检测结果通知所有处理器的中断，并且在执行了使计算模块之间的指令执行状态一致的延迟调节之后，产生用于执行恢复所有计算模块的同步操作的处理的重置信号。

在另一个优选结构中，再同步方法进一步包括步骤：接收到中断的所有处理器中的每一个转移到其中用于计算处理器中执行的指令的数量

的指令数计数器不能操作的再同步处理的处理器管理模式；

每个处理器将读出的指令数计数器值与从其它计算模块中的每个处理器接收的指令数计数器值比较；

5 不包括指示所有计数器中最大值的指令数计数器的计算模块进行执行指令的延迟调节，直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送一个通知，并且包括指示最大值的指令数计数器的计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，产生用于致使所有计算模块恢复彼此同步操作的重置信号。

10 在另一个优选结构中，在指令数计数器值不是最大的计算模块执行指令，直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将处理器设置在一个用于在执行了一个指令之后转移到处理器管理模式的逐步执行模式，并且处理器重复逐步执行模式的处理，直到指令数计数器值与最大指令数计数器值一致。

15 在另一个优选结构中，所有接收到中断的处理器中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作，并且保存和存储一个程序计数器值的再同步处理的处理器管理模式；

每个处理器读出每个处理器的指令数计数器值和保存的程序计数器值，并且将该值发送到所有其它计算模块；

20 每个处理器将指令数计数器值与一个从其它计算模块中的每个处理器接收的指令数计数器值比较；

指令数计数器值不是最大的计算模块进行执行指令的延迟调节直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送一个通知，并且包括指示最大值的指令数计数器的计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，产生用于致使所有计算模块恢复彼此同步操作的重置信号。

25

在另一个优选结构中，在指令数计数器值不是最大的计算模块执行指令直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将处理器设置到用于在执行到在一个指定的指令串中的特定位置的指令之后转移到处理器管理模式的断点指定执行模式；

将由从包括指示最大值的指令数计数器的计算模块接收的程序计数器值指示的指令位置指定为指令串中的特定位置，并且在执行指令串到达一个指定指令串中的特定位置之后，处理器转移到处理器管理模式。

在另一个优选结构中，在读出了一个程序执行状态之后，包括指示最大值的指令数计数器的计算模块等待来自其它计算模块的通知；

在执行了延迟调节处理之后，不包括指示最大值的指令数计数器的所有其它计算模块读出一个程序执行状态，并且将程序执行状态与延迟调节处理完成的通知一起发送到包括指示最大值的指令数计数器的计算模块，并且包括指示最大值的指令数计数器的计算模块比较所有计算模块的程序执行状态，并且当所有状态彼此一致时，指令产生用于恢复同步操作的重置信号，并且当检测到程序执行状态不一致的计算模块时，在执行了切断和无效计算模块的处理之后，指令产生恢复同步操作的重置信号。

根据本发明的另一方面，一种用于执行锁步同步容错计算机系统的再同步处理的再同步程序，锁步同步容错计算机系统包括多个具有一个处理器和一个存储器的计算模块、每个计算模块彼此时钟同步地处理同一指令串，包括以下功能：当检测到每个计算模块中的处理器之间对一个外部总线的接入状态不一致时，如果在包括每个计算模块的系统中没有检测到故障，那么对所有处理器产生中断，并且在执行了用于使计算模块之间的指令执行状态一致的延迟调节之后，致使每个处理器恢复同步操作。

在优选的结构中，再同步程序进一步包括以下功能：监测系统中故障存在/不存在；

监测每个计算模块中的每个处理器到一个外部总线的接入状态，当检测到每个计算模块中的每个处理器到外部总线的接入状态不一致时，如果在故障监测中没有检测到故障，那么产生一个将检测结果通知每个处理器的中断，并且在执行了使计算模块之间的指令执行状态一致的延

迟调节之后，产生一个用于恢复所有计算模块的同步操作的重置信号。

在另一个优选结构中，再同步程序包括接收到中断的所有处理器中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作的再同步处理的处理器管理模式的功能；

- 5 每个处理器将读出的指令数计数器值与一个从其它计算模块中每个处理器接收的指令数计数器值比较的功能；

不包括所有计数器中指示最大值的指令数计数器的计算模块进行执行指令的延迟调节直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括指示最大值的指令数计数器的计算模块发送一个通知的功能，和包括指示最大值的指令数计数器的计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，发出产生用于致使所有计算模块恢复同步操作的重置信号的指令的功能。

- 10 在另一个优选结构中，再同步程序包括，在指令数计数器不是最大的计算模块执行指令直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将处理器设置在一种用于在执行了一个指令之后转移到处理器管理模式的逐步执行模式的功能；

处理器重复逐步执行模式的处理直到指令数计数器值与最大指令数计数器值一致的功能。

- 20 在另一个优选结构中，再同步程序包括，所有接收到中断的处理器中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作并且保存和存储一个程序计数器值的再同步处理的处理器管理模式的功能；

每个处理器读出每个处理器的指令数计数器值和保存的程序计数器值，并且将值发送到所有其它计算模块的功能；

每个处理器将读出的指令数计数器值与从其它计算模块中的每个处理器接收的指令数计数器值比较的功能；

指令数计数器值不是最大的计算模块进行执行指令的延迟调节直到指令数计数器值与最大指令数计数器值一致，并且当值一致时，向包括

指示最大值的指令数计数器的计算模块发送一个通知的功能，和包括指示最大值的指令数计数器的计算模块等待接收来自所有其它计算模块的通知，并且当接收到所有通知时，发出产生致使所有计算模块恢复同步操作的重置信号的指令的功能。

- 5 在另一个优选结构中，再同步程序包括，在其中指令数计数器值不是最大的计算模块执行指令直到指令数计数器值与最大指令数计数器值一致的延迟调节中；

将处理器设置在用于在执行处理器到达在一个指定指令串中的特定位置的指令之后，转移到处理器管理模式的断点指定执行模式的功能；

- 10 将由从包括指示最大值的指令数计数器的计算模块接收的程序计数器值指示的一个指令位置指定为指令串中的特定位置的功能，和在处理器执行指令串达到一个指定指令串中的特定位置之后，将处理器转移到处理器管理模式的功能。

- 15 在另一个优选结构中，再同步程序包括，包括指示最大值的指令数计数器的计算模块在读出一个程序执行状态之后等待来自其它计算模块的通知的功能，所有不包括指示最大值的指令数计数器的其它计算模块读出一个程序执行状态，并且在执行了延迟调节处理之后把程序执行状态与延迟调节处理完成的通知一起发送到包括指示最大值的指令数计数器的计算模块的功能，和包括指示最大值的指令数计数器的计算模块比  
20 较所有计算模块的程序执行状态，并且当所有执行状态彼此一致时，发出产生用于恢复同步操作的重置信号的指令，和当检测到程序执行状态不一致的计算模块时，在执行了切断和无效计算模块的处理之后，发出产生用于恢复同步操作的重置信号的指令的功能。

- 25 从下面给出的详细说明中，可以对本发明的其它目的、特征、和优点有更清楚的了解。

#### 附图说明

从下面给出的详细说明以及本发明的优选实施例的附图，可以更充分地理解本发明，但是，不能将其作为对本发明的限制，而仅是为了解

释和理解本发明。

在附图中：

图 1 是显示根据本发明的第一实现模式的一个容错计算机系统的结构的方框图；

5 图 2 是用于解释根据第一模式的容错计算机系统的的第一操作的流程图；

图 3 是用于解释容错计算机系统的的第一操作的流程图；

图 4 是用于解释容错计算机系统的第二操作的流程图；

图 5 是用于解释容错计算机系统的第二操作的流程图；

10 图 6 是用于解释容错计算机系统的第三操作的流程图；

图 7 是显示根据第二实现模式的一个容错计算机系统的结构的方框图；

图 8 是用于解释根据第二模式的容错计算机系统的操作的流程图；

图 9 是用于解释根据第二模式的容错计算机系统的操作的流程图；

15 和

图 10 是显示根据第三实现模式的一个容错计算机系统的结构的方框图。

### 具体实施方式

20 以下参考附图详细讨论本发明的优选实施例。在以下的说明中，提出了许多特定的细节以便提供对本发明的全面了解。但是，熟悉本领域的人员应当知道，本发明可以不用这些细节实现。在其它的例子中，那些众所周知的结构没有详细的示出，以避免不必要地使本发明不明确。

以下参考附图详细说明本发明的实现模式。

25 图 1 中示出了根据本发明的第一实现模式的容错计算机系统的结构。在图 1 中，容错计算机系统包括多个计算模块 100 和 200，每个计算模块 100 和 200 彼此时钟同步地处理同一指令串。容错计算机系统将每个计算模块的处理结果相互比较。当一个计算模块展示出错误时，处理可以由剩下的计算模块继续。

计算模块包括多个处理器 101 和 102，处理器外部总线 103，存储器 104，和存储器控制单元 105。计算模块 200 也包括与计算模块 100 相同的组件。

此外，计算模块 100 和 200 通过存储器控制单元 105 和 205 连接到  
5 用于控制外围设备的外围设备控制单元 400 和 500。

上述容错计算机系统进一步包括故障检测器 700，总线监测器 701，中断控制单元 702，系统间通信控制单元 703，和同步控制单元 704。

故障检测器 700 监测包括相应计算模块 100 和 200 的整个系统中存在/不存在故障。

10 总线监测器 701 监测每个计算模块 100 和 200 中的处理器对外部总线 103 和 203 的接入状态。

在总线监测器 701 检测到相应计算模块 100 和 200 中的每个处理器对外部总线的输出中的不一致的情况下，当故障检测器 700 没有检测到故障时，中断控制单元 702 产生一个中断，以把影响通知所有处理器。

15 连接到相应计算模块 100 和 200 的系统间通信控制单元 703 提供了一个用于由相应计算模块 100 和 200 中的处理器检查操作中不一致的状态的通信路径。

20 连接到相应计算模块 100 和 200 的同步控制单元 701 响应来自任何一个处理器的指令对所有计算模块 100 和 200 产生一个用于控制再同步的重置信号。

在这里，包括在每个计算模块 100 和 200 中的处理器的数量不限于本实现模式中所示的两个，每个计算模块可以安装一个，或三个，或更多的处理器。

25 此外，计算模块的数量不限于本实现模式中所示的两个，而是可以包括三个或更多的计算模块，以具有与本发明相同的效果。

接下来，参考图 1，2 和 3 详细地说明根据第一实现模式的如此构造的容错计算机系统的操作。

故障检测器 70 监测包括相应计算模块 100 和 200 以及相应外围设备控制单元 400 和 500 的整个容错计算机系统中存在/不存在故障，并且将

结果通知中断控制单元 702。

总线监测器 701 连接到相应计算模块 100 和 200 的处理器外部总线 103 和 203。总线监测器 701 比较处理器 101、102、201 和 202 对外部总线的接入状态（总线协议），以监测相应处理器 101、102、201 和 202 是否彼此同步地在同一定时接入处理器外部总线 103 和 203。当检测到处理器 101、102、201 和 202 的操作中存在不一致时，总线监测器 701 将结果通知中断控制单元 702。

当总线监测器 701 通知检测到操作中存在不同步时，如果故障检测器 700 检测到容错计算机系统中没有故障时，中断控制单元 702 产生一个中断，以便将结果通知每个计算模块中的所有处理器。

中断控制单元 702 也在中断产生后抑制产生一个新的中断，直到从同步控制单元 704 发出一个指令。抑制产生新的中断防止了将每个计算模块的再同步处理造成的处理器对外部总线的接入的操作中的不同步认识为普通操作中的不同步。

接收到来自中断控制单元 702 的中断的每个处理器 101、102、201 和 202 通过执行用于恢复彼此时钟同步地执行同一指令串的状态的再同步处理，纠正相应计算模块 100 和 200 之间的操作中的偏差。

以下参考图 2 和 3 中所示的流程图说明有关的再同步处理。

一旦接收到一个中断，处理器 101、102、201 和 202 中的每一个转移到其中用于计算处理器中执行的指令的数量的指令数计数器不能操作的再同步处理的处理器管理模式（步骤 201）。

然后，处理器 101、102、201 和 202 中的每一个读出自己的指令数计数器的值（步骤 202），以将读出的指令数计数器的值通过存储器控制单元 105 和 205 以及系统间通信控制单元 703 发送到其它计算模块（步骤 203）。

处理器 101、102、201 和 202 中的每一个也读出通过系统间通信控制单元 703 以及存储器控制单元 105 和 205 发送的其它计算模块中的处理器的指令数计数器的值（步骤 204）。每个处理器比较所有计算模块每个中的相应处理器的指令数计数器的值（步骤 205）。

在这里，包括一个其指令数计数器指示所有计数器中最大值的处理器的计算模块（此后，将其称为具有最大值的指令数计数器的计算模块），在其它计算模块由于指令数计数器值比较结果而执行等价于一个延迟的指令时，进行排队等待执行的处理（步骤 206）。当所有计算模块 100 和 5 200 中执行的指令的数量彼此一致时，具有最大值的指令数计数器的计算模块通过存储器控制单元 105 或 205 指令同步控制单元 704 对所有计算模块 100 和 200 产生一个用于再同步处理的重置指令（步骤 207）。

另一方面，所有不包括指示最大值的指令数计数器的计算模块进行执行指令的延迟逐步调节处理，直到进入与执行的指令的数量（指令数 10 计数器值）是最大的计算模块的指令串执行状态相同的状态（步骤 208）。指令数计数器值不是最大的每个计算模块将延迟逐步调节处理的完成通知指令数计数器值是最大的计算模块（步骤 209），然后等待用于再同步处理的重置指令（步骤 210）。

一旦接收到来自具有指示最大值的指令数计数器的计算模块的指令，同步控制单元 704 对所有计算模块 100 和 200 同时产生一个用于恢复同步操作的重置信号。此外，与此同时，单元 704 向中断控制单元 702 发出一个指令，以允许产生一个中断，从而准备恢复到常规同步操作。 15

图 3 示出了图 2 中所示延迟逐步调节处理的详细功能。

在延迟逐步调节处理中（图 2 中的步骤 208），将处理器设置到其中 20 处理器刚好在执行了一个指令之后转移到处理器管理模式的逐步执行模式（步骤 301）。此后，处理器转移到正常执行模式（步骤 302）。

结果，在仅执行了一个指令之后（步骤 303），处理器转移到处理器管理模式（步骤 304）。

处理器读出自己的指令数计数器的值（步骤 305），并且将读出的值 25 与已经从具有最大值的计算模块接收的指令数计数器的值比较（步骤 306）。当它们未能彼此一致时，处理器再重复逐步执行模式的处理（步骤 301），并且当它们彼此一致时，完成延迟逐步调节处理。

上述功能使得一个由于不是固定故障原因造成锁步脱离的计算模块的状态能够与其它计算模块的状态一致。通常，当第一次通过监测外部

总线检测到不同步时，即使在失步状态，它的失步程度也低到足以通过执行上述的几步处理恢复到执行同一指令串的状态。因此，与所有存储区都要复制的惯用系统相比，它的处理时间可以大大减少。

图 4 和 5 是解释图 1 中所示构造的容错计算机系统中的再同步处理的第二操作的流程图。

在这个第二操作中，接收到来自中断控制单元 702 的中断的处理器 101、102、201 和 202 中的每一个执行图 4 中所示的再同步处理，以纠正相应计算模块 100 和 200 之间的操作偏差（失步），和恢复同步地执行同一指令串的状态。

一旦接收到中断，处理器 101、102、201 和 202 中的每一个转移到其中处理器中用于计算执行的指令数量的指令数计数器不能操作，并且保存和存储正常操作中的程序计数器（PC）的值的再同步处理的处理器管理模式（步骤 401）。

处理器 101、102、201 和 202 中的每一个读出它自己的指令数计数器值和保存的程序计数器的值（步骤 402），并且通过存储器控制单元 105 和 205 以及系统间通信控制单元 703 将读出的值发送到其它计算模块（步骤 403）。

处理器 101、102、201 和 202 中的每一个也读出通过系统间通信控制单元 703 以及存储器控制单元 105 和 205 从其它计算模块发送的其它计算模块的处理器指令数计数器的值和程序计数器值（步骤 404）。每个处理器将它自己的指令数计数器的值与所有计算模块中每个的对应处理器中的指令数计数器的值比较（步骤 405）。

在这里，包括其指令数计数器指示所有处理器中最大值的处理器的计算模块，在其它计算模块执行等价于作为比较结果的延迟的指令时，进行排队等待执行的处理（步骤 406）。当所有计算模块中执行的指令数量一致时，具有最大值的指令数计数器的计算模块通过存储器控制单元 105 或 205 向同步控制单元 704 发出指令，以产生对所有计算模块的再同步处理的重置指令（步骤 407）。

另一方面，所有其指令数计数器值不是最大的计算模块进行执行指

令的延迟逐步调节处理，直到执行了与具有指示最大值的指令数计数器的计算模块相同数量的指令（步骤 408）。每个不存在最大值的指令数计数器的计算模块，将延迟逐步调节处理的完成通知具有最大值的计算模块（步骤 409），此后，等待再同步处理的重置指令（步骤 410）。

5 图 5 示出了图 4 中延迟逐步调节处理的详细步骤。

在上述延迟逐步调节处理中（图 4 中的步骤 408），将处理器设置到在执行到达一个指定指令串中的一个特定位置的指令之后，将处理器再转移到处理器管理模式的断点指定执行模式。此时，将由从具有最大值的指令数计数器的计算模块接收的程序计数器值指示的指令位置作为指令串  
10 的特定位置（步骤 501）。此后，处理器转移到正常执行模式（步骤 502）。

结果，在执行指令达到指定的指令串中特定位置之后，即，达到指令串中与具有最大值的指令数计数器的计算模块中的位置相同的位置之后（步骤 503），处理器完成了延迟逐步调节处理，并且再次转移到处理器  
15 管理模式（步骤 504）。

图 6 是解释根据第一实现模式的容错计算机系统的第三操作的流程图，特别示出了遵循图 2 和 4 中所示的操作中指令数计数器比较处理（图 2 中的步骤 205 和图 4 中的步骤 405）的处理的另一个例子。

在图 6 所示的处理中，在读出了在具有最大值的指令数计数器的计算模块的程序执行状态之后（步骤 601），具有最大值的计算模块等待来自其它计算模块的延迟逐步调节处理完成的通知（步骤 602）。

另一方面，在通过任何上述方法或类似方法执行了延迟逐步调节处理之后（步骤 701），所有其指令数计数器值不是最大的其它计算模块各读出它的程序执行状态（步骤 702）。每个其它计算模块将程序执行状态  
25 与延迟逐步调节处理完成的通知一起发送到指令数计数器值是最大的计算模块（步骤 703），此后，等待再同步处理的重置指令（步骤 704）。

已经接收到程序执行状态的具有指示最大值的指令数计数器的计算模块比较所有计算模块的程序执行状态（步骤 603）。在状态全都一致的情况下，具有指示最大值的指令数计数器的计算模块执行立即恢复处理

的同步操作。当检测到其状态未能一致的计算机模块时，具有指示最大值的指令数计数器的计算机模块，在执行仅切断有问题计算机模块以使其无效的处理之后（步骤 604），执行恢复所有其它计算机模块的处理的操作（步骤 605）。

- 5       图 6 中所示处理内容使得即使锁步同步容错计算机系统采用了一个在逐步执行和断点执行中不总是可靠的处理器，也能够没有错误处理结果地实现本发明。

接下来，参考图 7，说明根据本发明的第二实现模式的容错计算机系统。

- 10       参考图 7，根据本实现模式的容错计算机系统包括多个各具有一个处理器和一个存储器的计算机模块 100，200 和 300，和多个分别具有与外围设备连接的机构 803 和 804 的外围设备控制单元 400 和 500。每个计算机模块 100，200 和 300 彼此同步地处理同一指令串，并且相互比较每个计算机模块的处理结果。即使当一个计算机模块出现故障，处理也可以通过其余  
15       计算机模块继续。

- 在根据第二实现模式的容错计算机系统中，每个外围设备控制单元 400 和 500 包括图 1 中所示的故障检测器 700，总线监测器 701，中断控制单元 702，系统间通信控制单元 703，和同步控制单元 704，并且进一步包括用于控制每个计算机模块和外围设备中每个控制单元与外围设备连  
20       接机构 803 和 804 之间的连接的 PCI 桥路 705。

- 尽管在图中未示出，外围设备控制单元 500 的结构与外围设备控制单元 400 的完全相同，并且要使用外围设备控制单元 400 或外围设备控制单元 500 中哪一个控制装置，可以通过在计算机模块中的处理器上运行的程序任意确定，从而，即使当与再同步有关的外围设备控制单元 400  
25       或 500 出现故障时，也可以通过利用没有故障的外围设备控制单元 400 或 500 提高可用性。

除了各计算机模块 100、200 和 300 与各外围设备控制单元 400 和 500 之间的连接是利用作为一个代表性外围设备连接接口的 PCI 桥路 705 建立的，并且向和从系统间通信控制单元 703 的数据发送和接收以及对同

步控制单元 704 的重置处理指令是利用 PCI 协议执行的之外，本实现模式通过与图 1 中所示操作相同的操作取得了本发明的效果。

此外，本实现模式示出提供了三个计算模块的结构，和图 8 和 9 中所示的利用这种结构的再同步处理的详细操作。图 8 示出了在锁步同步容错计算机系统包括三个或更多的计算模块的情况下，从上述图 2 中的操作推广出的操作的流程图。除了计算模块的数量不同之外，处理的内容与图 2 和 3 中所示的相同。

更具体地讲，在图 8 和 9 所示的流程图中，一旦接收到一个中断，每个处理器转移到再同步的处理器管理模式，在这种模式中用于计算处理器中执行的指令的数量的指令数计数器未能操作（步骤 801）。

然后，每个处理器读出自己的指令数计数器的值（步骤 802），并且将读出的指令数计数器的值通过存储器控制单元 105 和 205、PCI 桥路 705、和系统间通信控制单元 703 发送到其它计算模块（步骤 803）。

每个处理器也读出通过系统间通信控制单元 703、PCI 桥路 705、和存储器控制单元 105 和 205 从所述其它计算模块发送的其它计算模块中处理器的指令数计数器的值（步骤 804）。每个处理器比较所有计算模块中每个中的对应处理器的指令数计数器的值（步骤 805）。

在这里，包括其指令数计数器指示所有计数器中最大值的处理器的计算模块，在其它计算模块执行等价于一个作为指令数计数器值比较结果的延迟的指令的同时，进行排队等待执行的处理（步骤 806）。当在所有计算模块 100，200 和 300 中执行的指令数量彼此一致时，具有最大值的指令数计数器的计算模块指令同步控制单元 704 通过存储器控制单元 105 或 205 和 PCI 桥路 705 产生对所有计算模块 100、200 和 300 的再同步处理的重置指令（步骤 807）。

另一方面，所有其指令数计数器值不是最大的计算模块进行执行指令的延迟逐步调节处理，直到进入与具有指示最大值的指令数计数器的计算模块相同的指令串执行状态（步骤 808）。每个没有最大值的指令数计数器的计算模块将延迟逐步调节处理的完成通知具有指示最大值的指令数计数器的计算模块（步骤 809），此后，等待再同步处理的重置指令

(步骤 810)。

一旦接收到指令，同步控制单元 704 对所有计算模块 100、200 和 300 同时产生用于恢复同步操作的重置信号。此外，单元 704 此时给中断控制单元 702 发出一个指令，以允许在操作不同步时产生一个中断，从而  
5 准备好了恢复到正常同步操作。

图 9 示出了图 8 中所示延迟逐步调节处理的详细功能。

在延迟逐步调节处理中（图 8 中的步骤 808），将处理器设置到一种逐步执行模式，在这种模式中，刚好在执行了一个指令之后，处理器转移到处理器管理模式（步骤 901）。此后，处理器转移到正常执行模式（步  
10 骤 902）。

结果，在仅执行了一个指令之后（步骤 903），处理器转移到处理器管理模式（步骤 904）。

处理器读出它自己的指令数计数器的值（步骤 905），并且将读出的值与从具有指示最大值的指令数计数器的计算模块接收的指令数计数器的值比较（步骤 906）。当它们未能彼此一致时，处理器再次重复逐步执行模式的处理（步骤 901），并且当它们彼此一致时，完成延迟逐步调节处理。  
15

图 10 示出了根据本发明的第三实现模式的容错计算机系统的结构。参考图 10，根据本实现模式的容错计算机系统，除了总线监测器 701 和中断控制单元 702 向和从每个计算模块发送和接收信息也通过 PCI 桥路 705 执行之外，与图 7 中所示第二实现模式的相同，并且通过图 8 和 9 中所示的相同操作获得了相同的效果。  
20

在本发明的容错计算机系统中，执行再同步处理的每个单元的功能不仅能够通过硬件实现，而且也可以通过将执行上述每个单元的功能的再同步处理程序 1000 装载到计算机处理设备的存储器中控制计算机处理设备而实现。再同步处理程序 1000 存储在磁盘或半导体存储器之类的记录介质中，并且从记录介质装载到计算机处理设备中，以控制计算机处理设备的操作，从而实现每种上述功能。  
25

尽管本发明是参考上述优选实现模式说明的，但是本发明不必限于

上述实现模式，而是可以通过在其技术思想范围内的各种不同形式实现。

无论计算模块的数量是两个、三个、还是更多，本发明通过上述结构和操作取得了完全相同的效果。

此外，对每个计算模块中的处理器的数量也没有限制，并且具有一个处理器的结构和具有三个或更多处理器的结构都以完全相同的方式发挥作用。此外，尽管示出的上述每种实现模式是相应处理器共享一个外部总线并且连接在同一总线上的情况，但是，例如，多个处理器以星形形式连接到一个存储器的结构，或形成在一个计算模块的处理器物理分割地形成在多个板上的结构，都不影响本发明的效果。

另外，本发明可以包括一对计算模块和一个诸如总线监测器之类的相应的再同步单元，或多个这些组件对。此外，这些组件可以提供在外围设备控制单元中，或可以形成在一个专用板上。作为选择，计算模块和诸如总线监测器之类的相应的再同步单元可以通过一个 PCI 或诸如 PCI-X 之类的其它标准的接口装置，或通过一个没有标准化的专用接口装置连接，以获得上述相同效果。

如上所述，本发明得到了以下效果。

第一个效果是，使得一个容错计算机系统中的一个特定计算模块，在它由于不是固定故障的其它原因从锁步状态脱离时，能够在极短的时间内恢复到锁步状态。

原因在于，在失步产生的最初阶段，当每个计算模块的处理中的偏差很小时，通过对一个处理器产生中断，并且在调节一个其处理具有延迟的计算模块的延迟以恢复同步操作之后，可以不用复制所有存储器而恢复同步操作状态。

第二个效果是，提高了容错计算机系统的可用性。原因在于，通过在锁步脱离时显著加快再同步处理速度，可以大大减少整个系统的暂停时间。

尽管本发明是通过参考其示例实施例显示和说明的，但是应当理解，本领域的技术人员可以对其进行上述和各种不同的改变、省略和增加，而不脱离本发明的精神和范围。因此，不应当将本发明理解为限于上述

---

特定实施例，而是包括可以在所附权利要求中定义的范围及其等价物内具体实现的所有可能的实施例。

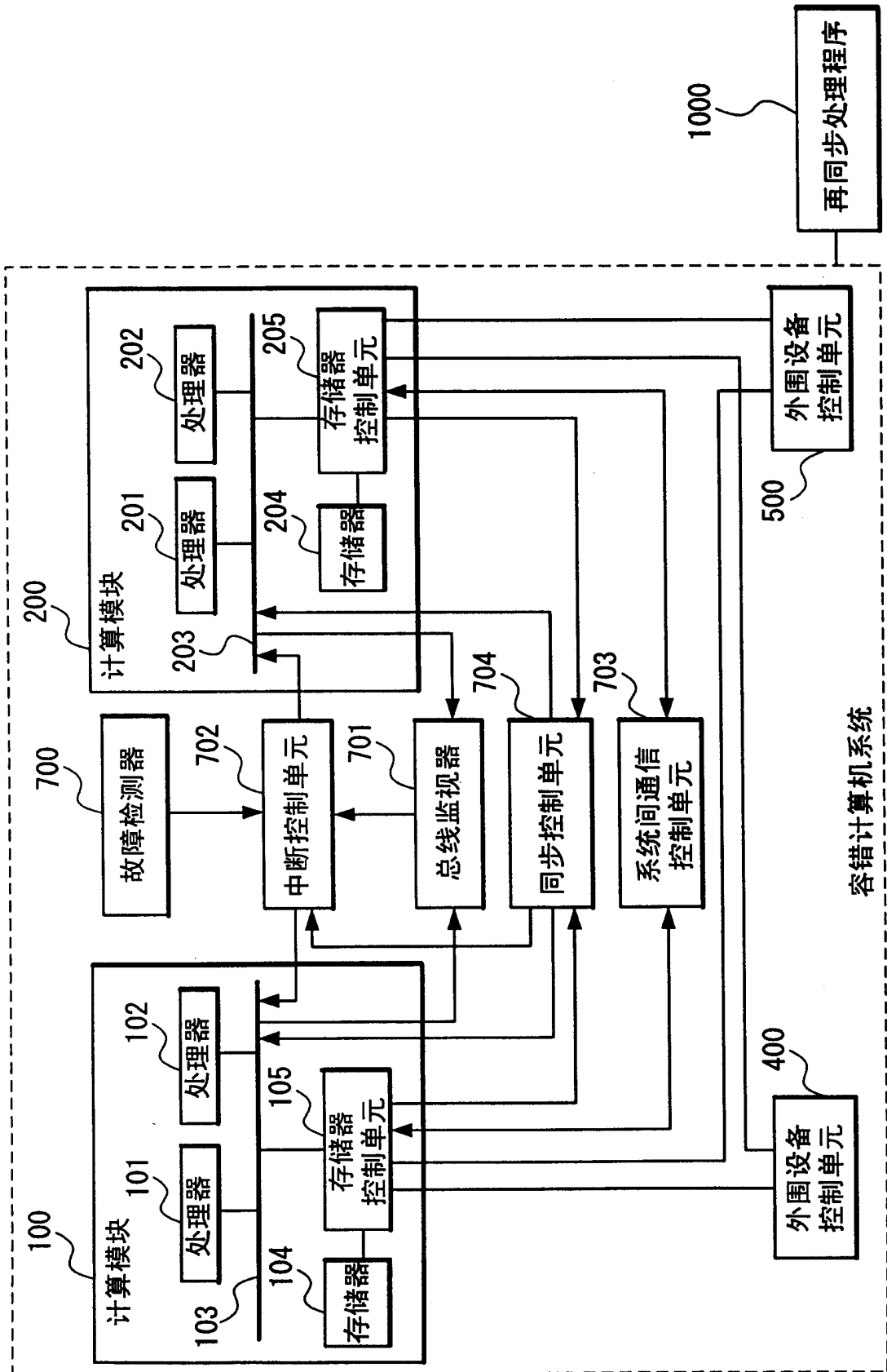


图 1

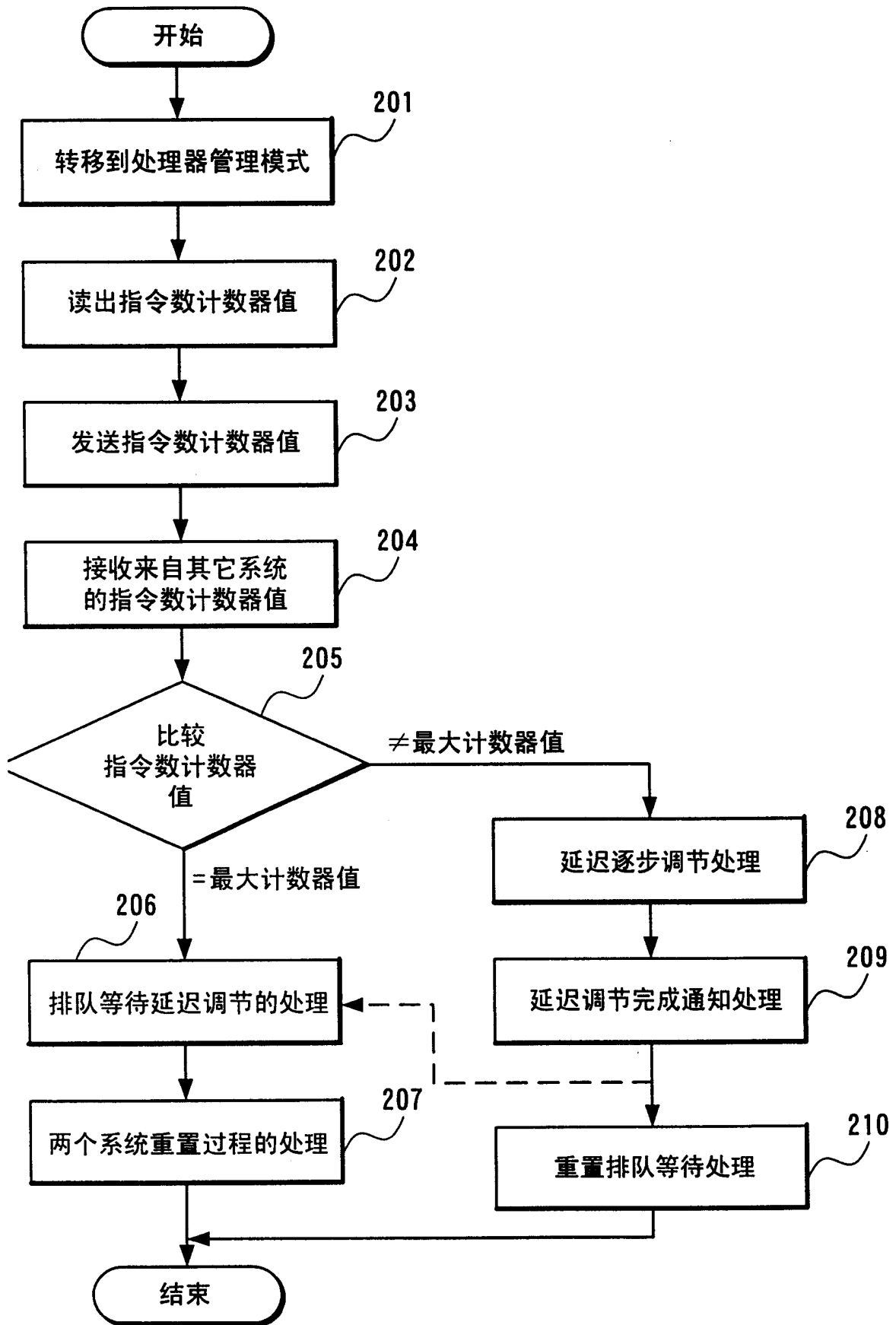


图 2

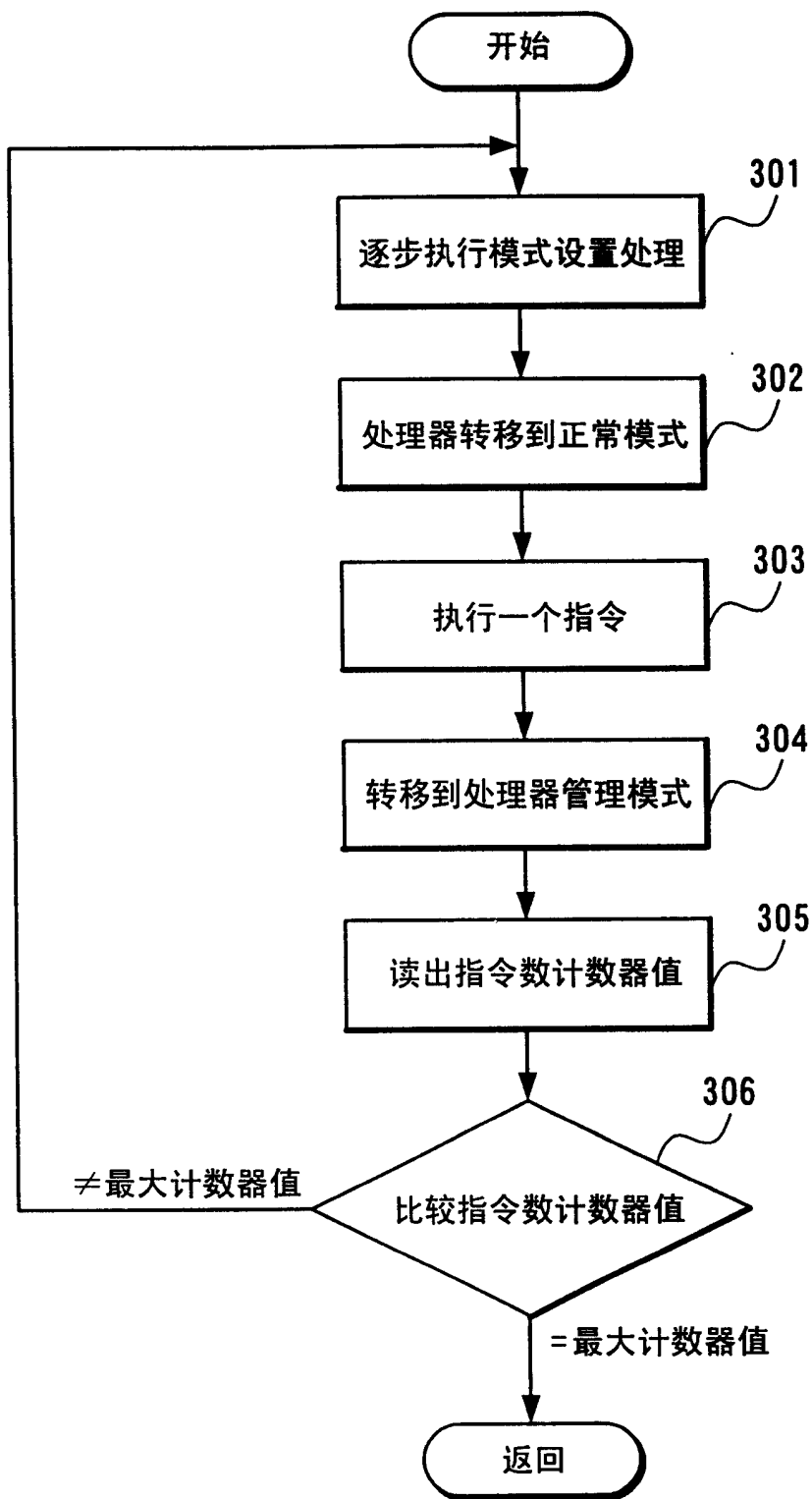


图 3

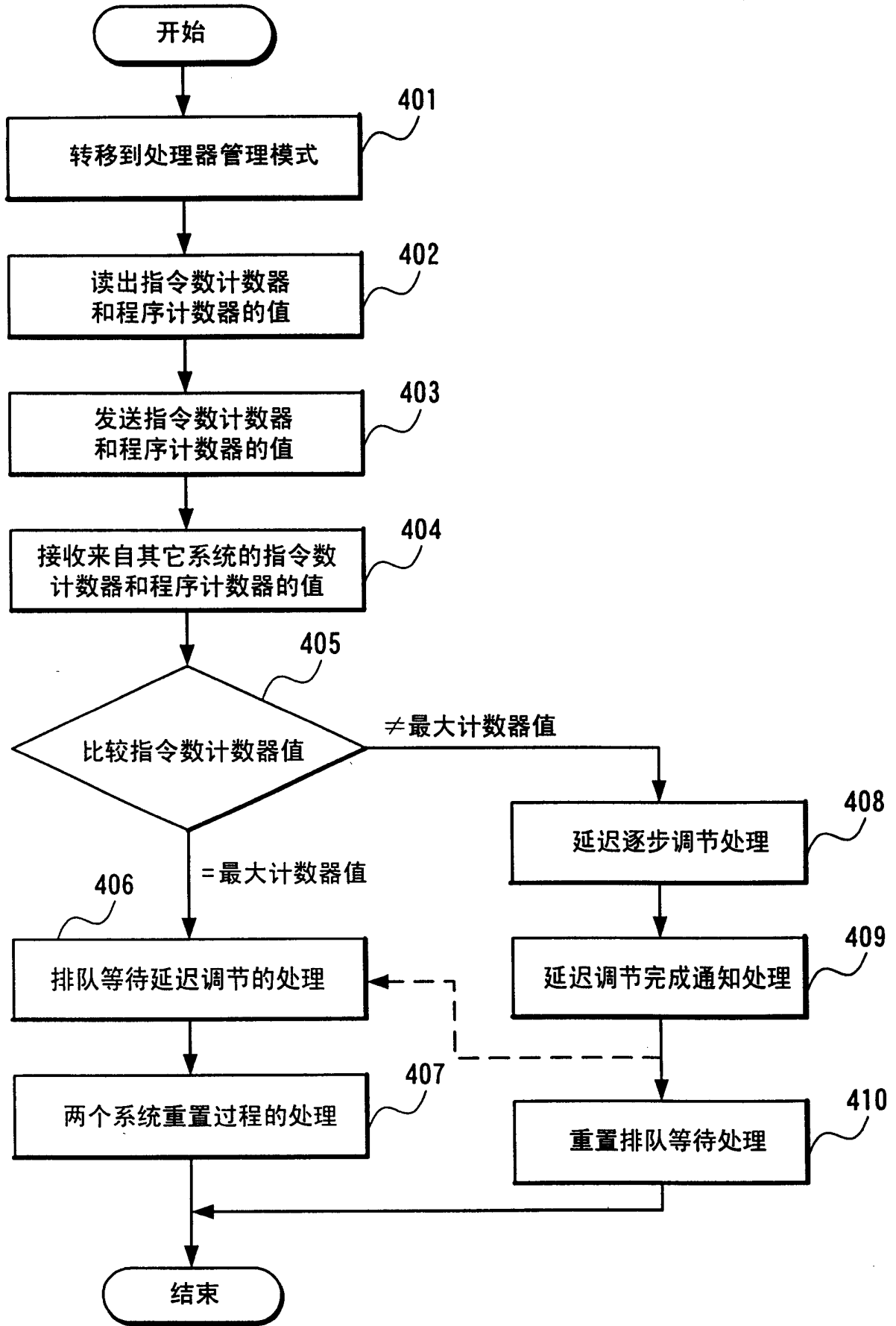


图 4

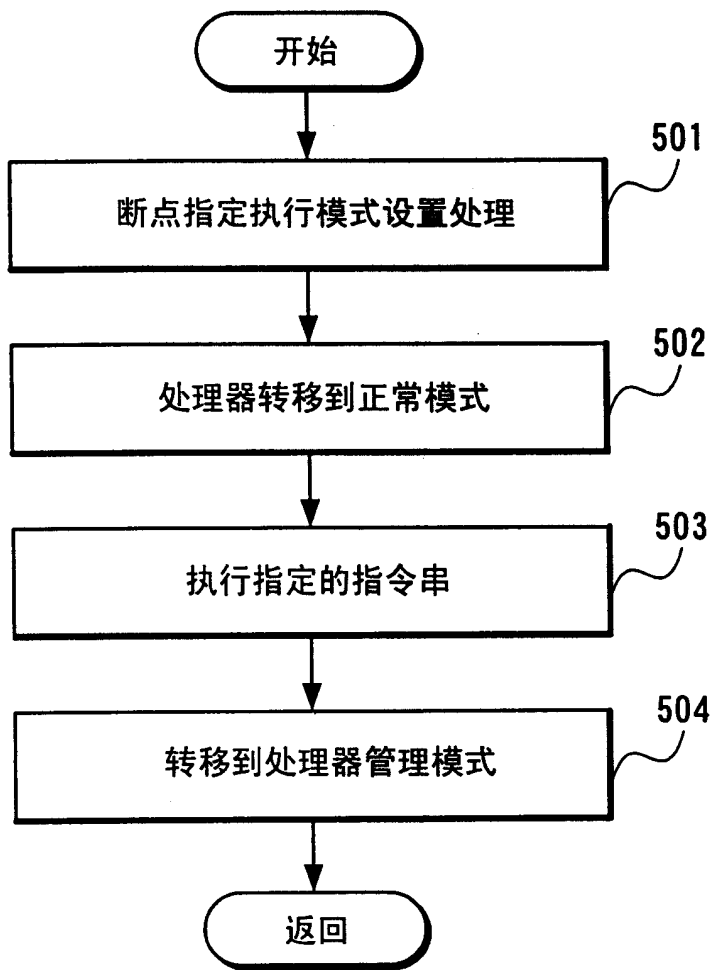


图 5

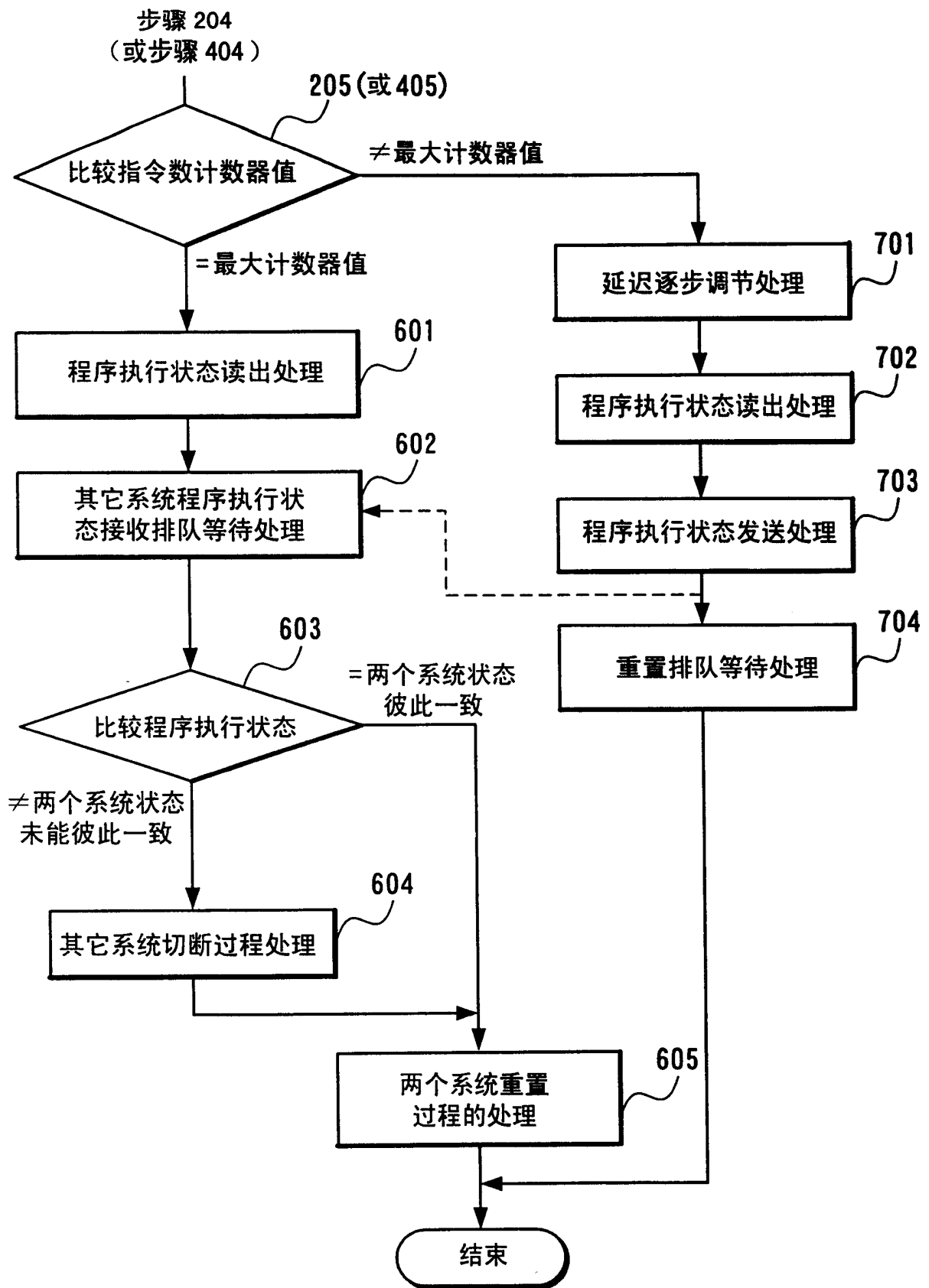


图 6

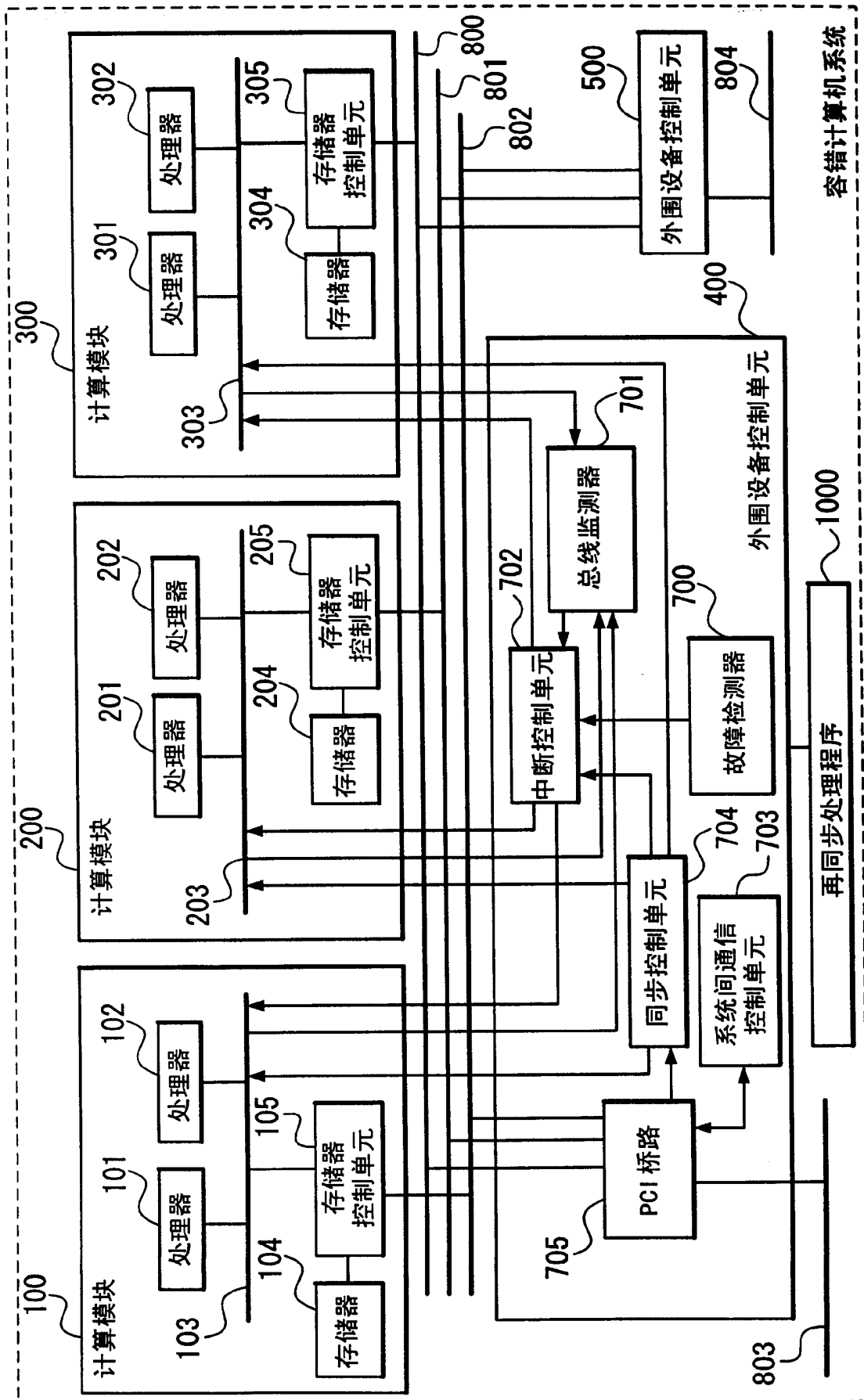


图 7

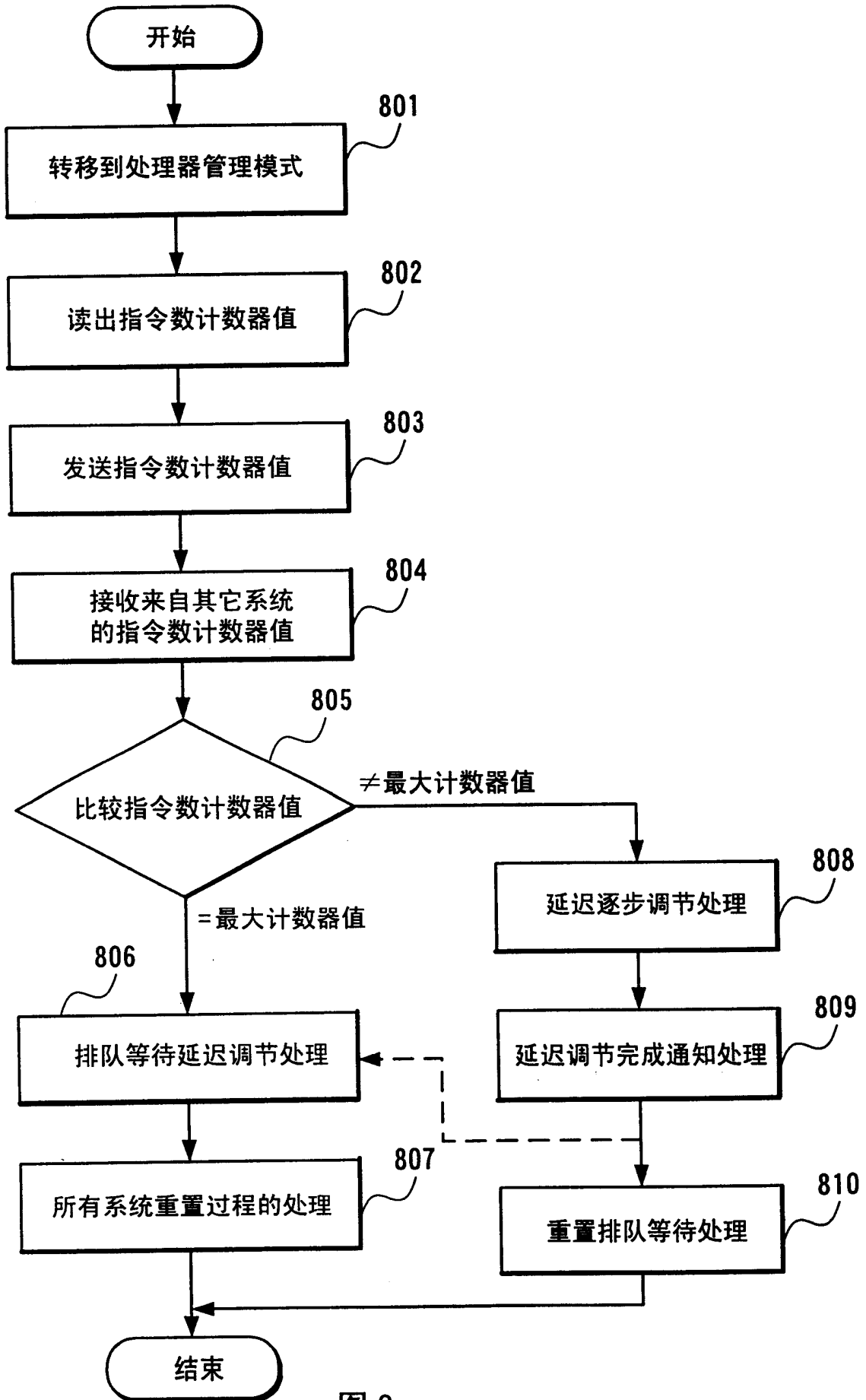


图 8

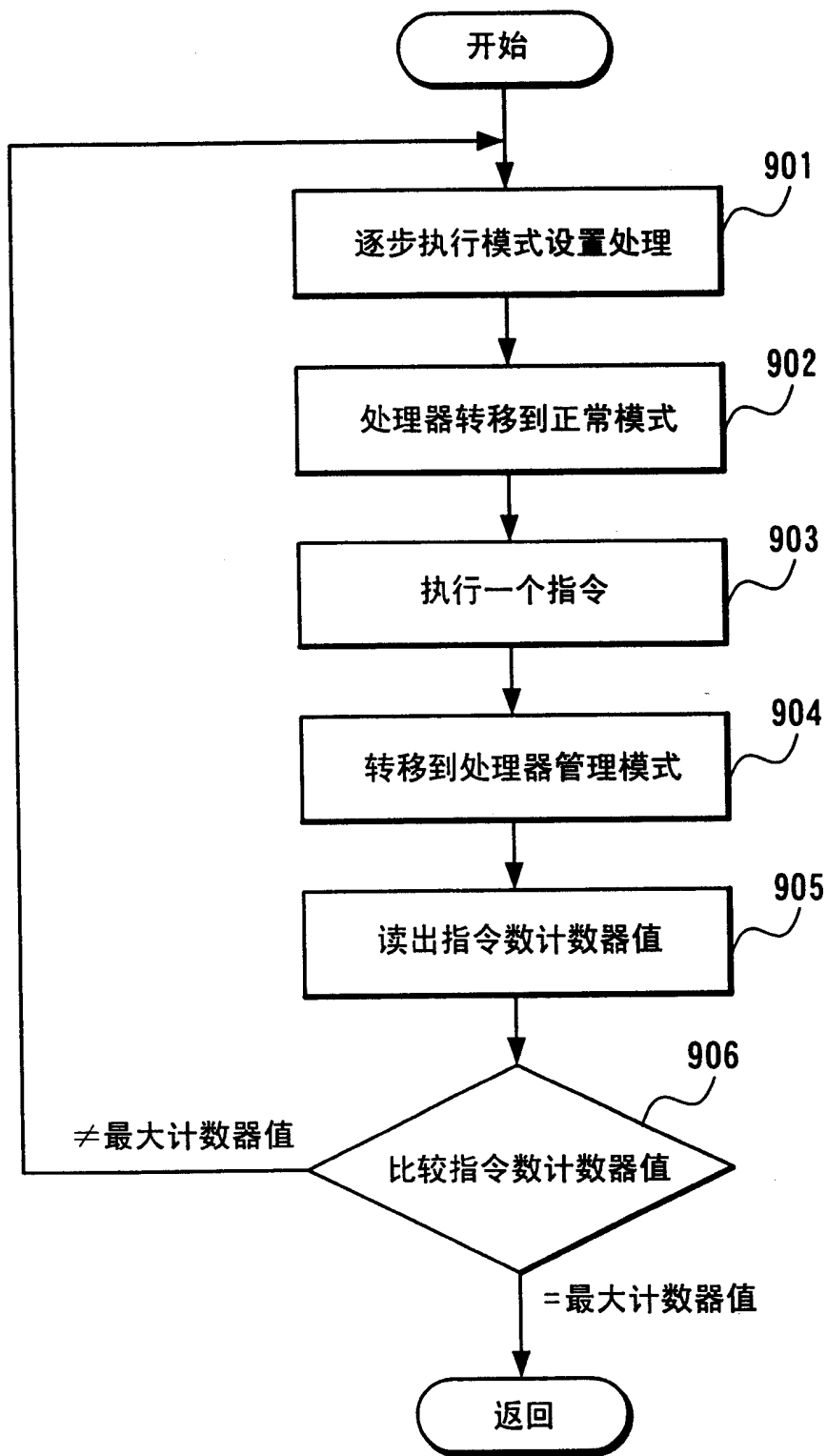


图 9

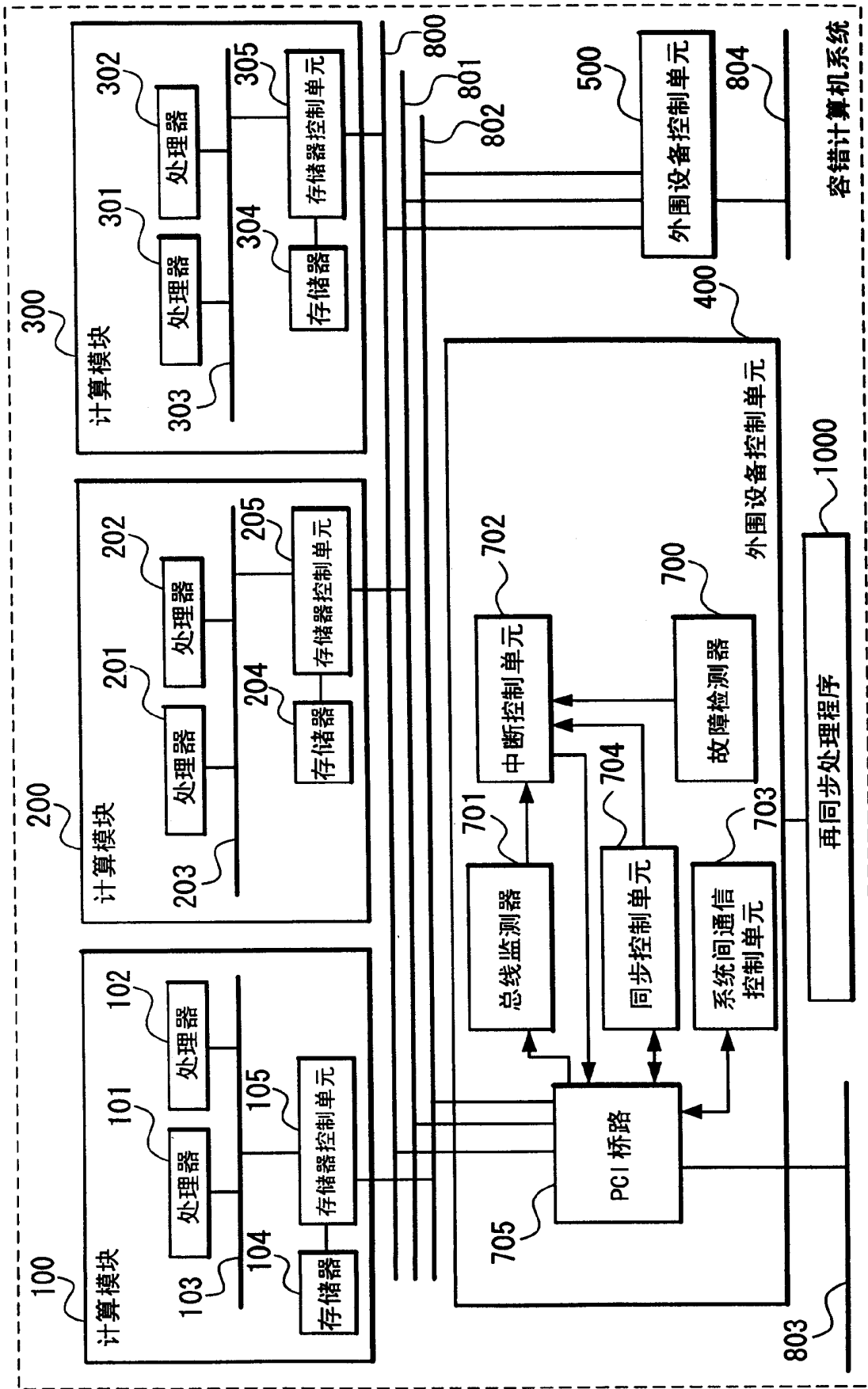


图 10