



[12] 发明专利申请公开说明书

[21] 申请号 03800717.7

[43] 公开日 2004年10月13日

[11] 公开号 CN 1537274A

[22] 申请日 2003.4.17 [21] 申请号 03800717.7

[30] 优先权

[32] 2002.5.28 [33] JP [31] 154313/2002

[86] 国际申请 PCT/JP2003/004886 2003.4.17

[87] 国际公布 WO2003/100613 日 2003.12.4

[85] 进入国家阶段日期 2004.1.20

[71] 申请人 索尼株式会社

地址 日本东京都

[72] 发明人 户川敦之

[74] 专利代理机构 北京市柳沈律师事务所

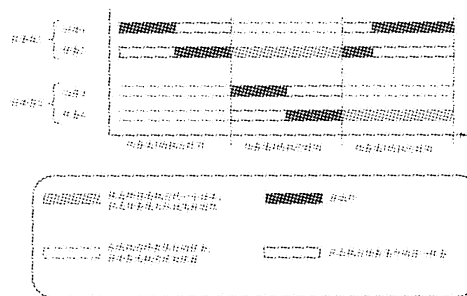
代理人 郭定辉 黄小临

权利要求书 3 页 说明书 12 页 附图 6 页
按照条约第 19 条的修改 4 页

[54] 发明名称 处理系统或计算机系统的任务控制方法和计算机程序

[57] 摘要

一种具有记录这样一种定时的机制的处理系统，在该定时具有高紧急度的处理被启动。如果该处理系统在具有低紧急度的处理期间进入临界区，就记录和检查是否在临界区的执行期间启动具有高紧急度的处理。如果具有高紧急度的处理不被启动，则该处理系统进入临界区，如果具有高紧急度的处理被启动，则该处理系统进行控制，以推迟进入临界区，直至具有高紧急度的处理结束。在执行多个任务的环境中，最好能够执行临界区中的专有存取控制。



1、一种在多个处理之间执行专有存取控制的处理系统，其特征在于，该处理系统包括：

- 5 开始定时记录装置，用于记录具有高紧急度的处理将要开始的定时；
 检验装置，用于一旦在具有低紧急度的处理的中间进入不能被多个处理同时引用的临界区，就通过引用上述的开始定时，检验是否将在临界区的执行期间开始具有高紧急度的处理；和
 控制装置，用于根据检验结果控制临界区的执行。

- 10 2、根据权利要求1所述的处理系统，其特征在于，如果在临界区的执行期间将不开始具有高紧急度的处理，则所述控制装置允许进入临界区，如果在临界区的执行期间开始具有高紧急度的处理，则延迟进入临界区。

3、一种存在多个任务执行环境的处理系统，根据不同策略执行任务执行环境的调度，其特征在于，该处理系统包括：

- 15 执行环境交换装置，用于交换任务执行环境；
 调度时间管理装置，用于管理调度时间，在此调度时间将接着进行任务执行环境的交换；
 任务执行管理装置，在当前任务执行环境下，根据上述调度时间管理任务的执行。

- 20 4、根据权利要求3所述的处理系统，其特征在于，当目前任务执行环境下运行的任务进入不能被多个任务同时引用的临界区时，所述任务执行环境管理装置根据在相对于调度时间的临界区的执行时间是否有空间，来确定是否进入所述临界区。

- 25 5、根据权利要求3所述的处理系统，其特征在于，当在相对于调度时间的临界区的执行时间有空间时，任务执行环境管理装置允许进入临界区，但是当没有空间时，向所述执行环境交换装置发出交换任务执行环境的指令。

6、根据权利要求3所述的处理系统，其特征在于，还包括上下文保存装置，当所述执行环境交换装置交换任务执行环境时，保存交换时的任务执行状态。

- 30 7、一种计算机系统上的任务控制方法，在该计算机系统中存在根据不同策略执行调度的多个任务执行环境，其特征在于，计算机系统的任务控制方

法包括:

执行环境交换步骤, 用于交换任务执行环境;

调度时间管理步骤, 用于管理调度时间, 在此调度时间将接着进行任务执行环境的交换;

- 5 任务执行管理步骤, 在当前任务执行环境下, 根据上述调度时间管理任务的执行。

- 8、根据权利要求 7 所述的计算机系统的任务控制方法, 其特征在于, 在所述任务执行环境管理步骤中, 当目前任务执行环境下运行的任务进入不能被多个任务同时引用的临界区时, 根据在相对于调度时间的临界区的执行时间是否有空间, 来确定是否进入所述临界区。
- 10

9、根据权利要求 8 所述的计算机系统的任务控制方法, 其特征在于, 在所述任务执行环境管理步骤中, 当在相对于调度时间的临界区的执行时间有空间时, 允许进入临界区, 但是当没有空间时, 向所述执行环境交换装置发出交换任务执行环境的指令。

- 15 10、根据权利要求 7 所述的计算机系统的任务控制方法, 其特征在于, 还包括上下文保存步骤, 当所述执行环境交换步骤交换任务执行环境时, 保存交换时的任务执行状态。

- 11、一种以计算机可读格式写入的计算机程序, 以便在计算机系统上执行在多个处理之间执行专有存取控制的处理, 其特征在于, 该计算机程序包
- 20 括:

开始定时记录步骤, 用于记录具有高紧急度的处理将要开始的定时;

检验步骤, 用于一旦在具有低紧急度的处理的中间进入不能被多个处理同时引用的临界区, 就通过引用上述的开始定时, 检验是否在临界区的执行期间将开始具有高紧急度的处理; 和

- 25 控制步骤, 用于根据检验结果控制临界区的执行。

12、一种计算机程序, 在该程序中以计算机可读格式写入对计算机系统的任务控制的过程, 并且在该计算机系统存在根据不同策略执行其调度的多个任务执行环境, 其特征在于, 所述计算机程序包括:

执行环境交换步骤, 用于交换任务执行环境;

- 30 调度时间管理步骤, 用于管理调度时间, 在此调度时间将接着进行任务执行环境的交换;

任务执行管理步骤，用于在当前任务执行环境下，根据所述调度时间管理任务的执行。

处理系统或计算机系统的任务控制方法和计算机程序

5 技术领域

本发明涉及一种处理系统或计算机系统的任务控制方法和计算机程序，该处理系统或计算机系统通过执行程序提供预定处理业务。特别是涉及一种关于分类处理系统或分类计算机系统的任务控制方法和计算机程序，在该分类处理系统或分类计算机系统中，存在多个控制流（例如，中断处理程序和标准处理程序）。

具体而言，本发明涉及一种处理系统或计算机系统的任务控制方法和计算机程序，其中所述处理系统或计算机系统执行程序的一些部分（临界区）的专有存取控制，而该临界区不能由多个任务同时引用。特别是，本发明涉及在存在多个任务执行环境的情况下，执行程序的临界区的专有存取控制的

15 处理系统或计算机系统的任务控制方法和计算机程序。

背景技术

随着现代 LSI（大规模集成电路）技术的创新推进，各种类型的信息处理装置和信息通信装置已经被开发和出售，并且已经渗透到我们日常生活中。借助这些装置可以提供各类处理装置，这些处理装置具有在操作系统提供的

20 执行环境下，由 CPU（中央处理单元）或者某些其它处理器执行的预定程序代码。

然而，在程序设计中，让多个控制流（有时称作“任务”）存在于程序中往往是有用的。这里所用的术语“多个控制流”是指这样的事实：如图 6 所示，多个“当前运行时的诸多点”存在于程序处理流中，换言之，存在于流程中。

25 在图 6 所示的实例中，在某一点 T1，步骤 S1 和步骤 S3 分别在流 I 和流 II 中执行。与此相对，在经过某些时间后的下一个点 T2，步骤 S2 和步骤 S3 分别在流 I 和流 II 中执行。

一般来说，当存在多个流以及每个流操作每个流共有的数据时，不能保持数据的一致性，除非这些数据是同步的。这里所用的“共用数据”包括任务

30 列表、条件变量等。条件变量是一个任务具有的诸多条件被抽象的概念，该

概念被用作当任务应当转换到等待状态或者当任务应当返回到运行状态时的通信方式(手段)。

例如,在控制 B 和 C 的两个流存在的情况下,将考虑每个控制流执行下列处理的情况。

5 过程 1: 读出变量 x 的值;

 过程 2: 把一个读出值加 1 后的值代入变量 x。

当两个流 B 和 C 的每个执行一次上述处理时,相同处理在执行两次后结束。因此,变量 x 的值应当增加 2。然而,当流 B 和流 C 如下所述完全相同时,变量 x 的值仅仅增加 1。

10 (1) 流 B 执行过程 1;

 (2) 流 C 执行过程 1;

 (3) 流 B 执行过程 2;

 (4) 流 C 执行过程 2。

15 为了避免这样的操作错误,需要在某一确定流中执行的一系列引用和更新操作(事务处理)期间,禁止引用和更新来自其它流的数据。

在上述实例中,由于在流 B 的操作序列之前,流 C 完成了引用和更新变量 x,即过程 1 和过程 2,因此出现了丧失数据一致性的问题。

20 如上述过程 1 到过程 2 这种操作还可以被认为是不能被多个任务同时引用的一个程序的多个部分,以下将其称之为“临界区(critical section)”。此外,为解决临界区中数据一致性问题,禁止其它任务引用和更新数据,也可以称之为“专有存取控制”。换言之,在对于控制流中某些数据执行一系列处理时,延迟由控制的另一个流对相同数据的操作动作,也就是,单独执行特定数据的操作。

本发明认为专有存取控制机制最好具有以下特点。

25 (1) 不存在具有高紧急度的处理被具有低紧急度的处理(发生优先级颠倒)延迟的可能性。

 (2) 甚至可以在多个任务集之间执行专有存取控制,其中根据不同策略执行这些任务集的调度。

30 在上述特点之中,(1)是必不可少的,其理由是本领域熟练技术人员显而易见的。此外,在一个计算机系统上同时运行多个操作系统时,或者在利用多个任务集(每个任务集具有特有的特征)之每个的特有调度方法时,(2)

也是必不可少的。

例如，对于任务之间的专有存取控制，使用了“互斥机制”和“信标机制”。然而，在这些方法中，利用这种专有存取控制存在以下问题：存在高优先级处理被低优先级处理延迟的可能性，也就是存在出现优先级颠倒的可能性，

5 因而不能满足特征（1）。

作为减轻优先级颠倒问题的方法，优先级继承协议（例如，参见 Lui Sha, Rangunathan Rajkumar 和 John P. Lehoczky 的论文：“Priority Inheritance Protocol: An Approach to Real-Time Synchronization”，IEEE Transaction on Computers, 第 39 卷第 9 期，第 1179-1185 页，1990 年 9 月出版）被提出。

10 优先级事务处理协议涉及这样一种方法，其中，在低优先级任务正在执行一系列操作的同时，在高优先级任务正在试图操作相同数据的情况下，低优先级任务的优先级暂时上升到与该高优先级任务相同的优先级。

图 7 中示出了优先级继承协议的操作。在此情况下，如果在低优先级任务 A 正在操作相同数据的同时，高优先级任务 B 就试图开始操作某些数据，

15 则必然发生延迟。此时，任务 A 的优先级暂时上升到与任务 B 相同的等级。此后，尽管其优先级低于任务 B 但高于任务 A 的任务 C 试图开始运行，但由于任务 A 的优先级被提升到高于任务 C，因此不中断任务 A 的运行。然后，在结束任务 A 后，在保持数据一致性的同时，任务 B 能够开始操作数据，而不会被优先级低于自己的任务 C 中断。

20 此外，该优先级继承协议是以根据共同标准即优先级执行所有任务的调度的思想为基础的。因此，很难将该优先级继承协议应用于多个调度共存的系统（特别是，存在不能根据优先级执行调度的任务集的一个系统），所述的多个调度共存例如是多个操作系统同时在单一计算机系统上操作的任务执行环境。换言之，不能满足上述的特征（2）。

25 作为一个没有带来这些问题的方法，调度器自觉同步方法可以被提供一个实例（例如，参见 Leonidas I. Kontothanassis, Robert W. Wisniewski, Michael L Scott 等人的论文：“Scheduler Conscious Synchronization”，ACM Transaction on Computer Systems, 第 15 卷第 1 期，1997 出版）。该方法通过禁止其任务在临界区的执行期间进行调度，限制了具有低紧急度的处理具有

30 具有高紧急度的处理的效果。特别是，具有高紧急度的处理的延迟时间被抑制到最大临界区执行时间以下。此外，该方法仅仅以禁止调度的机制存在为

基础。

然而，由于该方法未考虑到应用于多个调度共存的系统，因此在这样一种任务执行环境下，仍然遗留具有高紧急度的处理将被具有低紧急度的处理延迟的可能性。换言之，该方法不能满足上述的特征（2）。

- 5 此外，作为满足上述特征（1）和（2）的一个方法，无阻塞同步方法可以被提供为一个实例（例如，参见 Michael Barry Greenwald 的论文：“Non-blocking Synchronization and System Design”，斯坦福大学的博士论文）。然而，为了应用该方法，需要一个特殊硬件，从而导致成本增加。

10 发明内容

本发明的目的是提供一个良好的处理系统或计算机系统的任务控制方法和计算机程序，在该处理系统和计算机系统中，多个控制流（例如一个中断处理程序和一个标准程序）存在于一个程序中。

- 15 本发明另一个目的是提供一个良好的处理系统或计算机系统的任务控制方法和计算机程序，该处理系统或计算机系统能够适合于在一个程序部分（临界区）中执行专有存取控制，而且该程序部分不会同时被多个任务引用。

本发明的再一个目的是提供一种良好的处理系统或计算机系统的任务控制方法和计算机程序，该处理系统或计算机系统能够适合于在具有不同调度策略的多任务执行环境存在的情况下，在临界区中执行专有存取控制。

- 20 本发明的再一个目的是提供一种良好的处理系统或计算机系统的任务控制方法和计算机程序，该处理系统或计算机系统能够适合于在多个调度策略共存的系统中执行专有存取控制，而且不需要特定硬件。

本发明是考虑了上述问题作出的，本发明第一方面是一种在多个处理之间执行专有存取控制的处理系统，并且是一种包括以下特征的处理系统：

- 25 开始定时记录装置，用于记录开始具有高紧急度的处理的定时；
检验装置，用于一旦在具有低紧急度的处理的中间进入不能被多个处理同时引用的临界区，就通过引用上述的开始定时，检验是否在临界区的执行期间将开始具有高紧急度的处理；和
控制装置，用于根据检验结果控制临界区的执行。

- 30 然而，这里所用的“系统”是指多个设备（或者实现特定功能的功能模块）的逻辑聚集，并且与每个设备和功能模块是否存在于单一壳体中无关（这适

用于下文)。

为了在临界区的执行期间维持数据的一致性，如果在临界区的执行期间将不开始具有高紧急度的处理，则上述的控制装置进行接受进入临界区，如果在临界区的执行期间开始具有高紧急度的处理，则控制装置执行专有存取

5 控制，从而延迟进入临界区。

因此，根据本发明第一方面的处理系统，可以在多个处理之间执行专有存取控制，而不延迟具有高紧急度的处理的开始。

此外，根据本发明第一方面的处理系统，与不执行优先级继承的互斥和信标相比，通过降低上下文交换的频率减少了开销。

10 此外，根据本发明第一方面的处理系统，甚至在根据不同策略执行调度的多个任务集之间，也可以恰当地执行专有存取控制。

此外，根据本发明第一方面的处理系统，在多个操作系统同时操作的一个系统中，这些操作系统上操作的任务之间的专有存取控制以及这些操作系统之间的专有存取控制成为可能。

15 此外，根据本发明第一方面的处理系统，专有存取控制适合于在多个调度策略共存的一个系统中执行，并且不需要特定的硬件。

此外，本发明第二方面是一种处理系统或计算机系统的任务控制的方法，在该处理系统或计算机系统中，存在根据不同策略执行调度的多个任务执行环境，并且是一个特征在于包括以下步骤的处理系统或计算机系统的任务控制方法：

20

执行环境交换装置或步骤，用于交换任务执行环境；

调度时间管理装置或步骤，用于管理调度时间，在此调度时间将接着进行任务执行环境的交换；

25 任务执行管理装置或步骤，在当前任务执行环境下，根据上述调度时间管理任务的执行。

这里，任务执行管理装置或步骤仅需要通过以下方式执行任务的专有存取控制：当目前任务执行环境下运行的任务将要进入不能被多个任务同时引用的临界区时，根据在相对于调度时间的临界区的执行时间是否存在空间，来判断是否进入临界区。

30 具体而言，当在相对于调度时间的临界区的执行时间有空间时，任务执行管理装置或步骤允许进入临界区，但是当没有空间时，可以指令执行环境

交换装置交换任务执行环境。

这里，还提供了一个上下文保存装置，用于在任务执行被上述执行环境交换装置或步骤交换的时候，保存该交换时候的任务执行的状态。

根据本发明第二方面的处理系统或计算机系统的任务控制方法，甚至在
5 根据不同策略执行调度的多个任务集之间，也可以恰当地执行专有存取控制。

此外，根据本发明第二方面的处理系统或计算机系统的任务控制方法，在多个操作系统同时操作的一个系统中，这些操作系统上操作的任务之间的专有存取控制以及这些操作系统之间的专有存取控制成为可能。

此外，根据本发明第二方面的处理系统或计算机系统的任务控制方法，
10 与互斥和信标相比，通过降低上下文交换的频率减少了开销。

此外，根据本发明第二方面的处理系统或计算机系统的任务控制方法，专有存取控制适合于在多个调度策略共存的一个系统中执行，并且不需要特定的硬件。

此外，本发明第三方面是一种以计算机可读格式写入的计算机程序，以便在一个计算机系统中执行一个在多个处理之间执行专有存取控制的处理，
15 并且是其特征在于包括以下步骤的计算机程序：

开始定时记录步骤，记录开始具有高紧急度的处理的定时；

检验步骤，一旦在具有低紧急度的处理的中间进入一个不能被多个处理同时引用的临界区，就通过引用上述的开始定时，检验是否在临界区的执行
20 期间将开始具有高紧急度的处理；和

控制步骤，根据检验结果控制临界区的执行。

此外，本发明第四方面是一种计算机程序，在该程序中以计算机可读格式写入对计算机系统的任务控制的过程，在该计算机系统中存在根据不同策略执行其调度的多个任务执行环境，并且是其特征在于包括以下步骤的计
25 算机程序：

执行环境交换步骤，用于交换任务执行环境；

调度时间管理步骤，用于管理调度时间，在此调度时间将接着进行任务执行环境的交换；

任务执行管理步骤，用于在当前任务执行环境下，根据上述调度时间管
30 理一个任务的执行。

涉及本发明第三和第四方面之每个的计算机程序定义了以计算机可读格

式写入的计算机程序，以便实现计算机系统的预定处理。换言之，通过在计算机系统上安装涉及本发明第三和第四方面之每个的计算机程序，可以在计算机系统上演示协作效果，并且可以实现与涉及本发明第一和第二方面的处理系统或者计算机系统的任务控制方法的相同效果。

- 5 通过下面的基于本发明模式和附图的详细说明，将会使本发明的其它目的、特点和优点变得更加清楚。

附图说明

- 图 1 是用于本发明实施的处理系统 10 的硬件配置的示意图；
10 图 2 是显示同时运行的多个操作系统的示意图；
图 3 是涉及本发明模式的任务执行环境的配置的示意图；
图 4 是显示任务集交换软件交换任务集的一个过程的流程图；
图 5 是显示当属于任务集的任务进入临界区时执行的一个过程的流程图；
15 图 6 是显示一个程序的处理流即流程的示意图，在该程序中存在正在运行的多个点；以及
图 7 是显示优先级继承协议操作的示意图。

具体实施方式

- 20 下面将结合附图给出对本发明的模式的详细说明。

图 1 示意地显示了用于本发明实施的处理系统 10 的硬件配置。如图所示，处理系统 10 包括：处理器 11，RAM（随机存取存储器）12，ROM（只读存储器）13，多个输入/输出装置 14-1、14-2……，以及定时器 15。

- 25 处理器 11 是处理系统 10 的主控制器，并且在操作系统（OS）控制下运行各种程序代码。

通常，将操作系统管理和控制程序运行的单位称作“任务”。在涉及本模式的处理系统中，运行多个任务存在于一个程序中。因此，存在比处理器 11 的数量更多的任务，处理器 11 是实际进行计算的部件。

- 30 操作系统实际上通过频繁地交换由处理器 11 处理的任務，并行运行每个任务。每个任务具有分配的任务 ID，借助该任务 ID 可以将一个任务与其它任务相区别。此外，每个任务执行一系列操作，换言之，执行关于利用存储

栈上的数据区域的数据的事务处理。

此外，在本模式中，多个操作系统可以被同时运行。这些系统分别构成具有不同特征的任务集，并且根据不同策略执行任务的调度。下面将说明涉及本模式的处理系统 10 的任务执行环境。

5 处理器 11 通过总线 16 与其它装置（下面将进行说明）互连。系统总线 16 上的每个装置被给予特有的存储器地址或者 I/O 地址，并且处理器 11 通过指定这些地址能够存取访问预定装置。系统总线 16 是包括地址总线、数据总线和控制总线的公共信号传输路径。

10 RAM 12 是可写存储器，并且可以被用来装载由处理器 11 运行的程序代码或者暂时存储一个运行程序的工作数据。程序代码可以包括：例如，BIOS（基本输入/输出系统），操作外围装置的硬件的装置驱动器、操作系统、应用程序等。

ROM 13 是非易失性存储器，用于永久存储预定码和数据，并且存储例如 BIOS、自诊断程序（通电自检：POST）等。

15 输入/输出装置 14 包括：连接显示器 21 的显示接口 14-1；用户输入装置接口 14-2，连接诸如键盘 22 和鼠标 23 的用户输入装置；外部存储装置接口 14-3，连接诸如硬盘 24 和媒体驱动器 25；网络接口卡（NIC）14-4，与外部网络相连接；等等。

20 显示接口 14-1 是一个专用接口控制器，用于实际处理由处理器 11 发出的绘画命令。显示接口 14-1 上处理的绘画数据一旦被写入帧缓存器（未示出）之后，就例如通过显示器 12 在显示屏上输出。

25 HDD 24 是一个外部存储装置（已知），在该装置中固定安装了一个作为存储器载体的磁盘。HDD 24 在存储容量、数据传送速率等方面优于其它外部存储装置。通常，在 HDD 24 中以非易失性方式存储由处理器 11 运行的操作系统的程序代码、应用程序、装置驱动器等。在 HDD 24 上以可运行状态安装软件程序将被称作向一个系统安装程序。例如，可以在 HDD 24 上安装使多个任务存在而设计的应用程序以及实现本发明的操作系统。

30 媒体驱动器 25 是装入便携媒体并用于访问（存取）其数据记录面的装置，所述的便携媒体比如是 CD（光盘）、MO（磁光盘）、DVD（数字多能光盘）等。

便携媒体主要用来备份软件程序、数据文件等，作为计算机可读格式的

数据，或者用于在系统之间传递数据的目的（换言之，包括出售、流通和散发）。例如，能够物理地流通和散发使多个任务存在而设计的应用程序以及在使用这些便携媒体的多个装置之间实现本发明的操作系统。

5 根据预定通信协议如以太网（已注册商标）等的网络接口 14-1 可以将系统 10 连接到局部网络如 LAN（局域网），或者还连接到广域网如互联网。

在网络中，多个主机终端（未示出）以透明状态连接，并且建立了分布式计算环境。在网络中，可以执行软件程序和数据内容的散发。例如，可以经由网络下载使多个任务存在而设计的应用程序以及实现本发明的操作系统。

10 中断等级被分配给输入/输出装置 14-1、14-2.....的每一个，并且响应预定事件的出现（例如，键盘输入的 GUI 处理，鼠标点击等，或者硬盘中数据传递的完成），经由中断请求信号线 19 通告处理器 11。处理器 11 响应这种中断请求，执行对应的中断处理程序。

15 定时器 15 是一个以预定周期生成定时器信号的装置。也把一个中断等级分配给定时器 15，并且经由中断请求信号线 19 生成关于处理器 11 的周期性中断。

此外，图 1 所示的这样一种处理系统 10 的实例是用于 IBM 个人计算机“PC/AT（个人计算机/高级技术）”的兼容机器或者后继机器。当然，也能够应用具有与涉及本模式的处理系统 10 不同结构的计算机。

20 在本模式中，多个操作系统同时运行。这些操作系统中的每个构成一个具有不同特征的任务集，并且根据不同策略执行任务的调度。

图 2 示意地示出了同时运行的多个操作系统。在该图所示的实例中，在单一处理系统 10 上同时运行两个操作系统 OS1 和 OS2。在 OS 1 提供的任务执行环境下，执行任务 1 和任务 2。此外，在 OS 2 提供的任务执行环境下，
25 执行任务 3 和任务 4。OS 1 上执行的任务 1 和任务 2 构成任务集 1，OS 2 上执行的任务 3 和任务 4 构成任务集 2。

在处理系统 10 中，例如通过上下文交换的方式，交替地给予 OS 1 和 OS 2 系统使用权。上下文被交换到 OS 1 期间的周期对应于任务集 1 的执行周期。在任务集 1 的执行周期期间，根据 SO 1 具有的调度，在任务 1 与任务 2 之间
30 执行专有存取控制，换言之，执行任务集 1 的专有控制。如同专有存取控制的方法那样，也可以应用互斥、信标、优先级继承协议或者其它已知或特有

方法。

同样，上下文被交换到 OS 2 期间的周期对应于任务集 2 的执行周期。在任务集 2 的执行周期期间，根据 SO 2 具有的调度，执行任务 3 与任务 4 之间的专有存取控制，换言之，执行任务集 2 的专有控制。如同专有存取控制的方法那样，也可以应用互斥、信标、优先级继承协议或者其它已知或特有方法。

本发明实现了如图 2 所示的多个任务共存的系统中的专有存取控制，其中，所述多个任务的调度根据不同策略实现。

在图 3 中，示意地显示了涉及本发明的模式的任务执行环境的配置。如图 10 所示，其调度根据不容策略实现的多个任务同时存在。

任务集 1 的管理模块和任务集 2 的管理模块分别对应于 OS 1 和 OS 2，并且在授予系统使用权的周期内，根据预定调度策略执行任务集 1 和任务集 2 的专有存取控制。任务交换软件是执行操作系统的所有管理的软件模块，比如多个操作系统间的上下文交换，并且能够进行从任务集选择一个执行任务的管理。任务集交换软件使用从定时器 15 获取时间信息的定时模块以适当的定时完成系统使用权的重新分配（换言之，任务集交换）。

在本模式中，任务集交换软件保持以下三个变量，以便适当地管理任务集的交流。

(1) 当前任务集：

20 存储标识任务集的值变量，其中把处理器使用权正在分配给该任务集。

(2) 上下文保存地址：

在重新分配使用权中，需要保存此时刻的处理器状态。变量“上下文保存地址”是保持该保存区域的地址的变量。

(3) 使用权转移的调度时间：

25 一个变量，用于保持使用权将接着被转移的时间。该变量是每个任务集的管理模块能够读出一个值的变量。

在图 4 中，以流程图的形式示出了任务集交换软件交换任务集的过程。

在交换任务集时，首先在上述的变量“上下文保存地址”中保存当前状态（步骤 S1）。更具体地说，所述的保存状态涉及处理器 11 的寄存器值以及 30 RAM 12 的存储器图像的消除。

然后，计算任务集的交换将接着要执行的时间（步骤 S2）。接着，将已

计算的时间代入上述的变量“使用权转移的调度时间”（步骤 S3）。

此外，将使用权将要转移的任务集的标志符代入上述的变量“当前任务集”（步骤 S4）。

接着，将定时器调整到在步骤 S2 中计算的随后执行任务交换的时间（步骤 S5）。该定时器在设定的时间造成中断发生。通过改中断，激活下一个任务集交换处理。

然后，把使用权的转移通知给将转移使用权的任务集管理模块（步骤 S6）。

在图 5 中，以流程图形式显示了进入临界区时，由属于某个任务集的一个任务完成的过程。

10 管理当前任务集的一个任务执行的任务管理模块引用由任务集交换软件保持的变量“使用权转移的调度时间”，并确定存在用于任务执行的空间（步骤 S11）。这里所用的空间对应于通过从使用权转移的调度时间减去一个将临界区执行时间加到当前时间的值所获得的值。

接着，确定空间是否大于零（步骤 S12）。

15 如果空间大于零，则执行专用于每个任务集的临界区进入处理（步骤 S13）。

另一方面，当空间等于或小于零时，如果进入临界区，则出现其它任务操作相同数据被拒绝的矛盾，而在此期间，临界区的处理不能在下一个任务集交换时间到来之前结束。为此，任务集管理模块将处理器使用权的放弃通知给任务集交换软件。

20 通过该通知，在处理（使其为“处理 A”）的中间，将控制转移到不同任务集的一个任务上。当把处理器 11 的使用权再次授予正在执行处理 A 的任务归属的任务集时，处理 A 达到完成。

25 例如，当在当前任务集中执行的情况下，在具有低紧急度的处理的中间进入临界区时，引用下一个任务集交换将要执行的调度时间，并且检验是否将在临界区的执行期间开始具有高紧急度的处理。如果将不开始该具有高紧急度的处理，则进入临界区，但是如果将开始该具有高紧急度的处理，则执行控制，推迟进入临界区，直至具有高紧急度的处理完成。

30 因此，可以在多个处理之间恰当地执行专有存取控制，而不会在根据不同策略执行其调度的多个任务集之间，延迟具有高紧急度的处理的开始。

补充

至此，已经参照特定模式详细说明了本发明。然而，本领域熟练技术人员将会明白，在不背离本发明的范围条件下，可以对上述模式作出修改和替代。换言之，本发明是以实例方式公开的，但是不能按限制方式解释本发明的内容。为了确定本发明的范围，应当考虑下面所述的权利要求的范围。

工业应用性

根据本发明，可以提供对高级处理系统或者计算机系统的任务控制方法和计算机程序，在此类高级处理系统和计算机系统中，多个控制流（例如，中断处理程序和标准处理程序，或者多个任务，和类似事情）存在于一个程序中。

此外，根据本发明，可以提供对高级处理系统或者计算机系统的任务控制方法和计算机程序，在该高级处理系统和计算机系统中，可以在不能被多个任务同时引用的一部分程序（临界区）中恰当地执行专有存取控制。

此外，根据本发明，可以提供对高级处理系统或者计算机系统的任务控制方法和计算机程序，在该高级处理系统和计算机系统中，可以在具有不同调度策略的多个任务执行环境存在的条件下，恰当地执行临界区中的专有存取控制。

此外，根据本发明，可以提供对高级处理系统或者计算机系统的任务控制方法和计算机程序，在该高级处理系统和计算机系统中，可以在多个调度策略共存的系统中恰当地执行专有存取控制，而不需要任何特殊硬件。

根据本发明，专有存取控制可以在多个处理之间执行，而且不延迟具有高紧急度的处理的开始。

此外，根据本发明，通过减少上下文交换的频度，较之不执行优先级继承的互斥和信标，可以减少开销。

此外，根据本发明，甚至可以在根据不同策略执行调度的多个任务集之间，恰当地执行专有存取控制。

此外，根据本发明，在多个操作系统同时操作的一个系统中，在这些操作系统上操作的诸多任务之间的专有存取控制以及操作系统之间的专有存取控制成为可能。

与无阻塞同步相反，本发明能够上述效果，而又不使用特殊硬件。

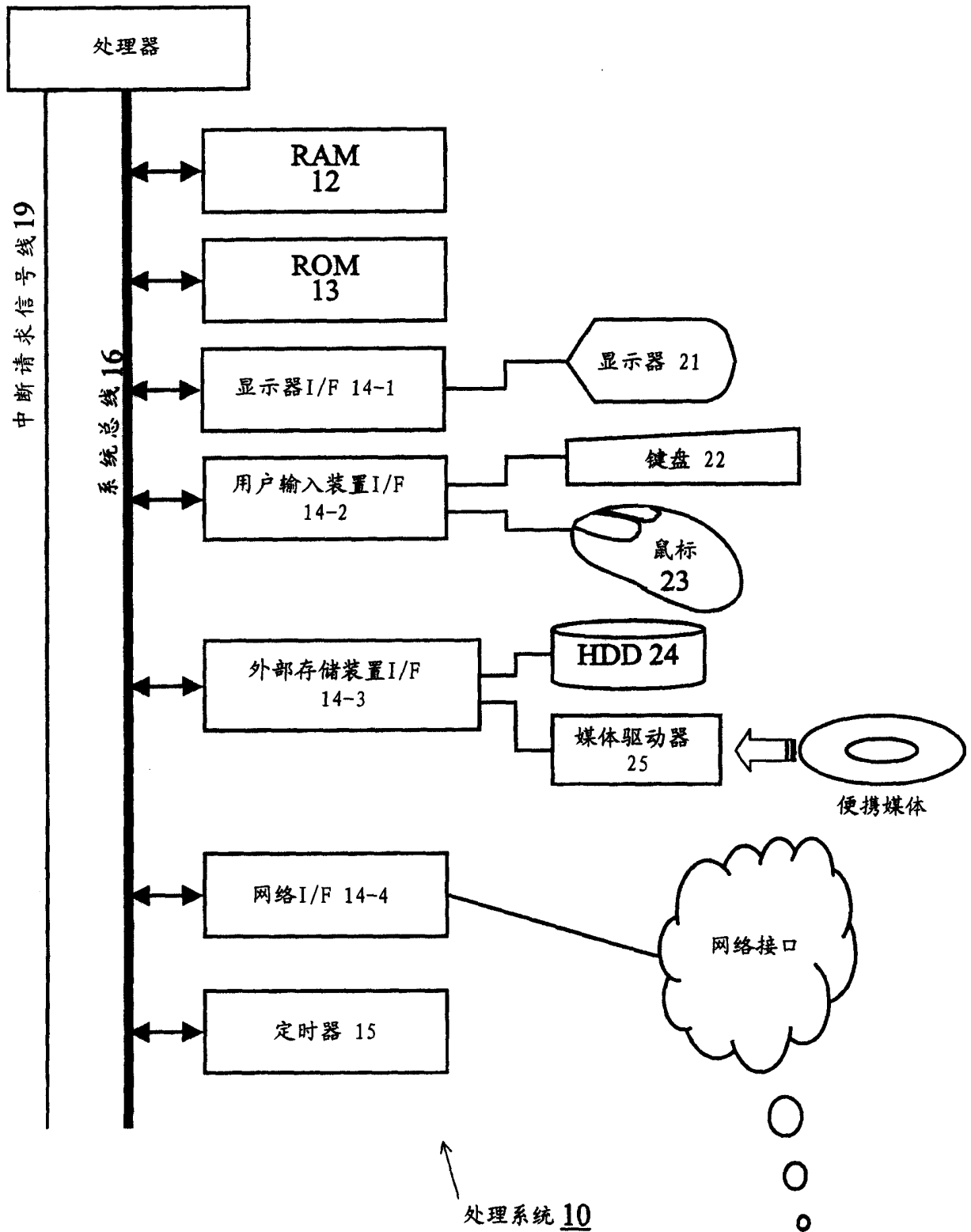


图 1

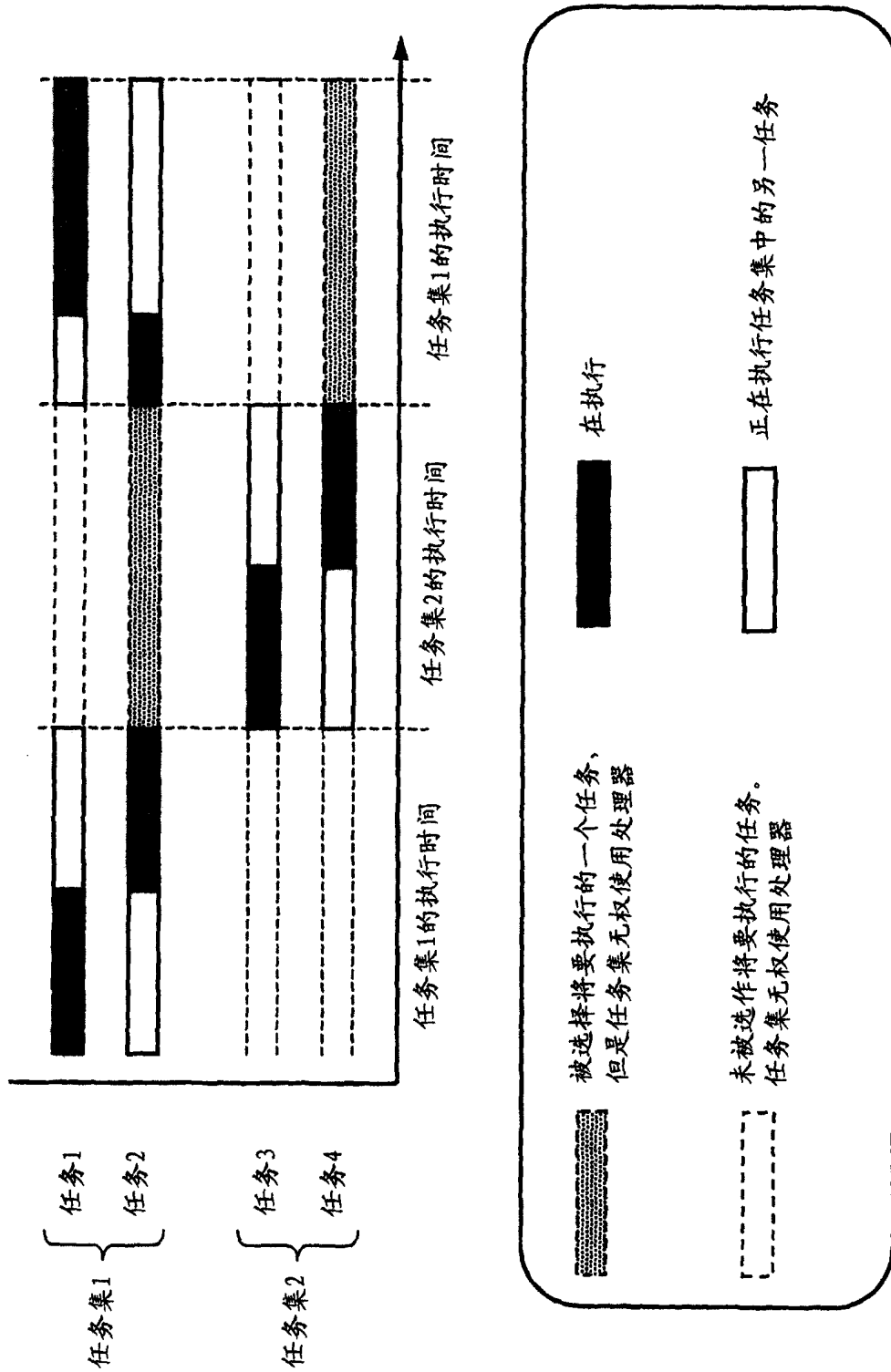


图 2

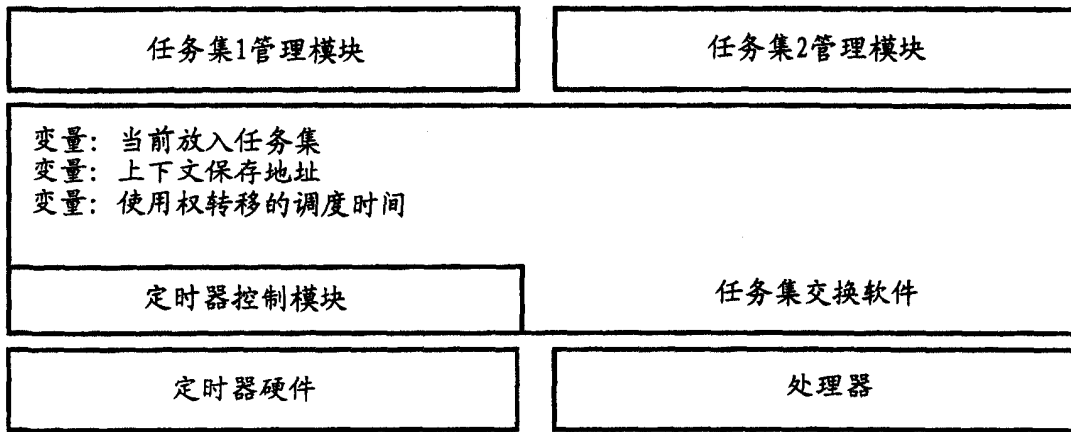


图 3

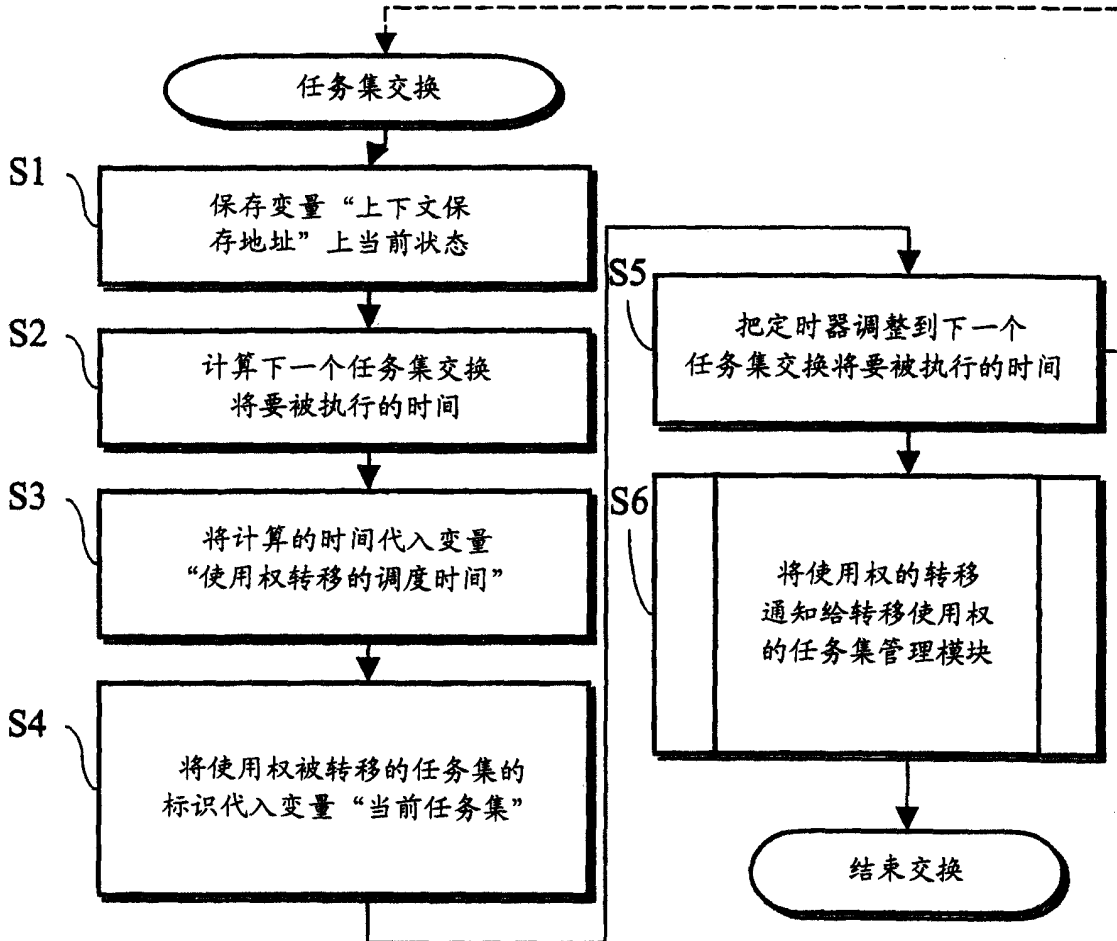


图 4

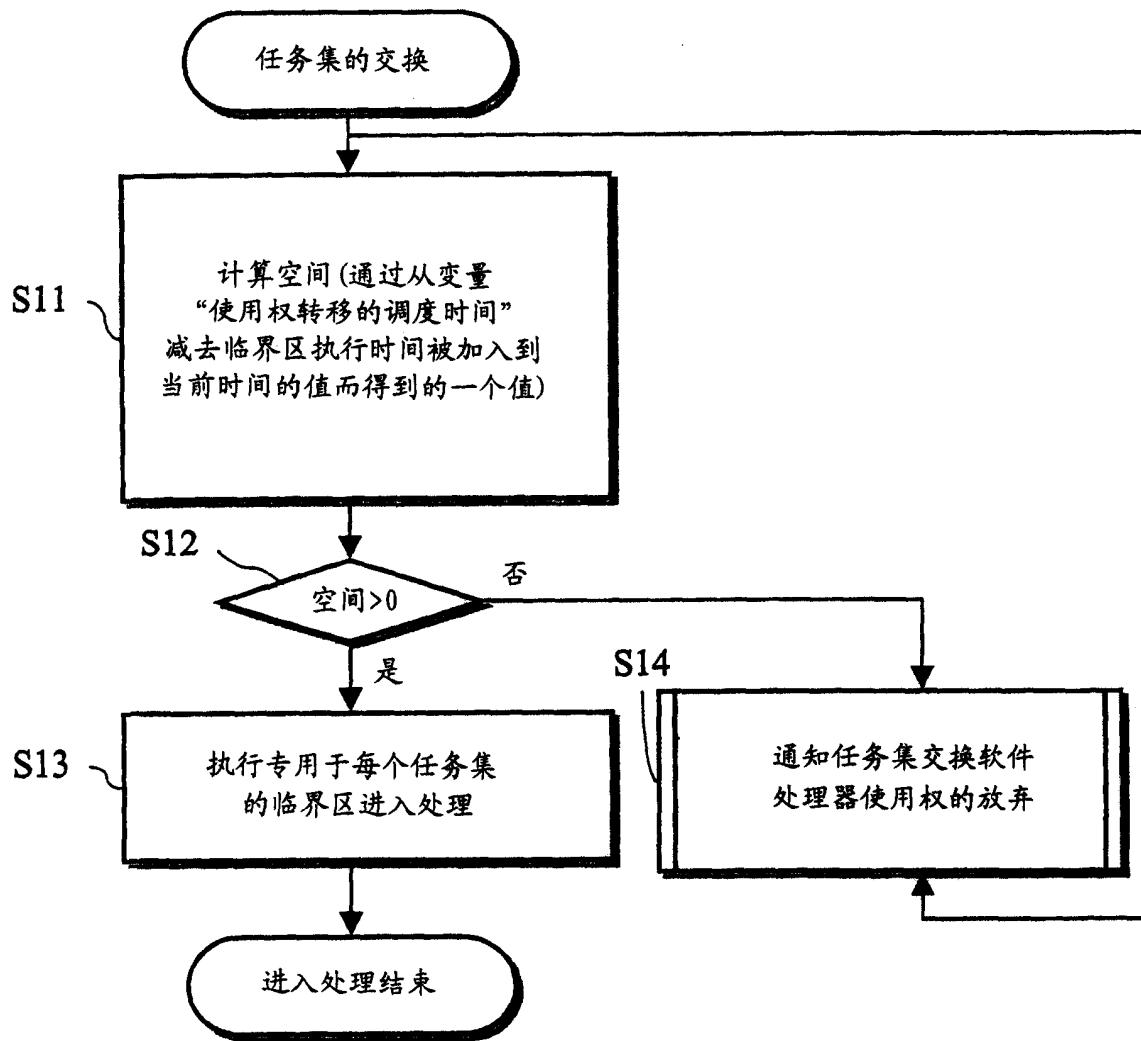


图 5

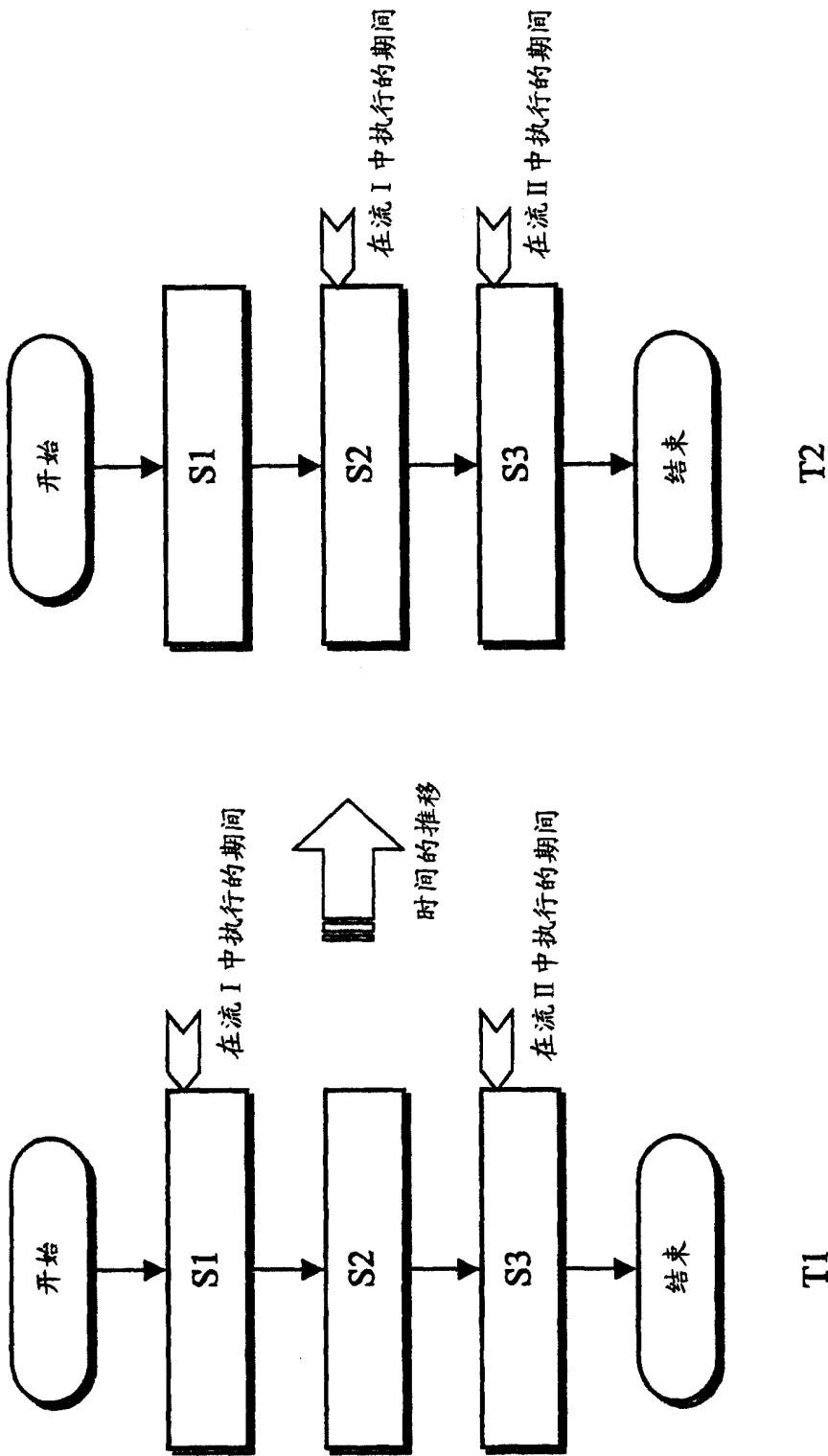


图 6

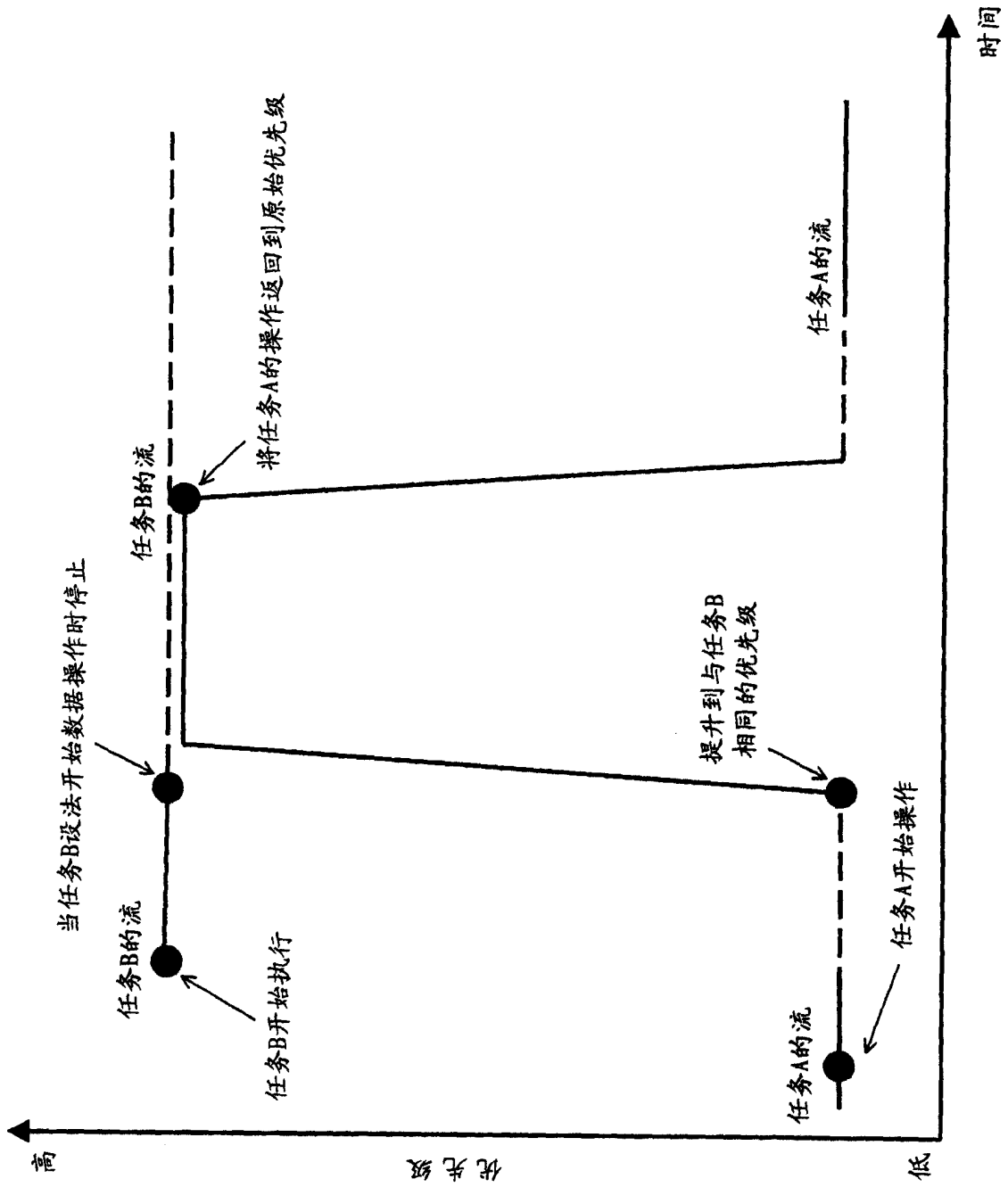


图 7

基于条约 19 条(1)的说明书

本发明如另外提交的手续修改书的权利要求范围第 3 项中所记载的那样,是一种“存在于多个任务执行环境的运算处理系统,根据不同策略执行任务调度,其特征在于,该运算处理系统包括:

执行环境切换装置,用于切换任务执行环境;

预定时间管理装置,用于管理预定时间,在此预定时间进行对下一个任务的执行环境的切换;以及

任务执行管理装置,根据上述预定时间管理在当前任务执行环境下的任务。”

根据本发明的运算处理系统,任务集切换软件参照作为保持的变量的使用权转变预定时间,通过从使用权转变预定时间中减掉当前时间与临界区执行时间相加所得的值,来判断任务执行的裕度。并且,可根据这样求出的裕度,控制任务执行环境的切换(例如,请参照本发明说明书中第 16 页第 1 行至同页第 24 行)。

这里所说的任务执行环境是由操作系统提供的,“根据不同策略执行调度的多个任务执行环境”相当于多个操作系统同时工作的系统环境。因此,根据本发明,在这样的计算环境中,根据任务执行的裕度,进行任务间的排他控制,同时也能够实现操作系统间的排他控制(例如,请参照本发明说明书中第 8 页第 6 行至同页第 16 行)。

相反,在(日本)特开平 8-95807 号公报中,根据从优先级低的任务的临界区的结束为止的时间和优先级高的任务的执行预定时间

的关系，来控制执行优先级低的任务。另外，在(日本)特开2003-131892号公报中，通过控制任务的优先级，来避免由于中断处理而使优先级逆转的现象。但是，无论在上述的哪一篇文献中都完全没有提到根据任务的执行环境的切换预定时间来进行任务的管理这一点。另外，无论在上述的哪一篇文献中，也完全没有提到根据不同的策略来进行任务调度的多个任务集之间(即，多个操作系统之间)的排他控制这一点。

- 1、一种存在多个任务执行环境的处理系统，根据不同策略执行任务执行环境的调度，其特征在于，该处理系统包括：
- 5 执行环境交换装置，用于交换任务执行环境；
调度时间管理装置，用于管理调度时间，在此调度时间将接着进行任务执行环境的交换；
任务执行管理装置，在当前任务执行环境下，根据上述调度时间管理任务的执行。
- 10 2、根据权利要求1所述的处理系统，其特征在于，当目前任务执行环境下运行的任务进入不能被多个任务同时引用的临界区时，所述任务执行管理装置根据在相对于调度时间的临界区的执行时间是否有空间，来确定是否进入所述临界区。
- 15 3、根据权利要求1所述的处理系统，其特征在于，当在相对于调度时间的临界区的执行时间有空间时，任务执行管理装置允许进入临界区，但是没有空间时，向所述执行环境交换装置发出交换任务执行环境的指令。
- 4、根据权利要求1所述的处理系统，其特征在于，还包括上下文保存装置，当所述执行环境交换装置交换任务执行环境时，保存交换时的任务执行状态。
- 20 5、一种计算机系统上的任务控制方法，在该计算机系统中存在根据不同策略执行调度的多个任务执行环境，其特征在于，计算机系统的任务控制方法包括：
执行环境交换步骤，用于交换任务执行环境；
调度时间管理步骤，用于管理调度时间，在此调度时间将接着进行任务
25 执行环境的交换；
任务执行管理步骤，在当前任务执行环境下，根据上述调度时间管理任务的执行。
- 30 6、根据权利要求5所述的计算机系统的任务控制方法，其特征在于，在所述任务执行管理步骤中，当目前任务执行环境下运行的任务进入不能被多个任务同时引用的临界区时，根据在相对于调度时间的临界区的执行时间是否有空间，来确定是否进入所述临界区。

7、根据权利要求6所述的计算机系统的任务控制方法，其特征在于，在所述任务执行管理步骤中，当在相对于调度时间的临界区的执行时间有空间时，允许进入临界区，但是当没有空间时，向所述执行环境交换装置发出交换任务执行环境的指令。

5 8、根据权利要求5所述的计算机系统的任务控制方法，其特征在于，还包括上下文保存步骤，当所述执行环境交换步骤交换任务执行环境时，保存交换时的任务执行状态。

9、一种计算机程序，在该程序中以计算机可读格式写入对计算机系统的任务控制的过程，并且在该计算机系统中存在根据不同策略执行其调度的多个任务执行环境，其特征在于，所述计算机程序包括：

- 10 执行环境交换步骤，用于交换任务执行环境；
- 调度时间管理步骤，用于管理调度时间，在此调度时间将接着进行任务执行环境的交换；
- 任务执行管理步骤，用于在当前任务执行环境下，根据所述调度时间管
- 15 理任务的执行。