



[12] 发明专利说明书

[21] ZL 专利号 95196678.2

[45] 授权公告日 2003 年 1 月 1 日

[11] 授权公告号 CN 1097786C

[22] 申请日 1995. 12. 8 [21] 申请号 95196678.2

[30] 优先权

[32] 1994. 12. 9 [33] SE [31] 9404296 - 7

[86] 国际申请 PCT/SE95/01481 1995. 12. 8

[87] 国际公布 WO96/18145 英 1996. 6. 13

[85] 进入国家阶段日期 1997. 6. 9

[73] 专利权人 艾利森电话股份有限公司

地址 瑞典斯德哥尔摩

[72] 发明人 L·G·史文森 L·U·詹森

A·N·I·霍尔特·罗斯特

B·M·塞缪尔森

[56] 参考文献

JP1101735A 1995. 4. 19 G06F15/20, 15/02

JP1101735A 1995. 4. 19 G06F15/20, G06F15/02

US5193187A 1993. 3. 9 G06F9/46

US5193187A 1993. 3. 9 G06F9/46

WO9207331A1 1992. 4. 30 G06F15/40

WO9207331A1 1992. 4. 30 G06F15/40

审查员 张 蕾

[74] 专利代理机构 中国专利代理(香港)有限公司

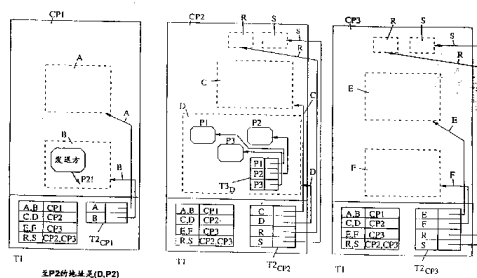
代理人 王 勇 王忠忠

权利要求书 3 页 说明书 18 页 附图 6 页

[54] 发明名称 一种计算机控制系统

[57] 摘要

一种计算机控制系统包括一个处理体系结构。此处理体系结构保证应用程序以协作进程方式被执行, 进程相互间、与一个控制系统间、与此体系结构外的外部世界间交互作用。进程实例(发送方, P1, P2, P3)借助于分布单元(A-F, R, S)分布于控制系统处理器(CP1, CP2, CP3)上。



1. 一种计算机控制系统，其特征在于包括

5 一个处理体系结构，它包括一个控制系统平台，该控制系统平台包括多个用来处理一个数据库的不同部分的控制系统处理器，所述处理体系结构还包括管理由处理器提供的资源和实施由控制系统提供的抽象概念的一个操作系统，还包括进程资源的表示；

所述处理体系结构保证应用程序以协作进程方式被执行，进程相互间、与该控制系统间、与此体系结构外的外部世界间交互作用；

10 进程资源表示的实例借助分布单元分布于控制系统处理器上，每一个分布单元形成一组进程实例；

该分布单元属于特定分布单元类型，其中的每一分布单元类型描述属于此种类型的分布单元的共同属性。

2. 根据权利要求1的系统，其特征在于，其中进程实例包括执行程序代码的物理进程实例，和

15 一种抽象概念形式的逻辑进程实例，在一个系统中的逻辑进程实例的种类由存在于系统中的进程类型、和它们的属性确定，以及在硬件设备方面由系统的配置确定。

3. 根据权利要求2的系统，其特征在于，其中分布单元既组合逻辑的也组合物理的进程实例，组合逻辑实例，因为实例和分布单元间的关系是用逻辑实例来表述的；组合物理实例，因为当一个分布单元被分配给一个处理器时，与分布单元有关的所有的物理进程实例是在那个特定处理器上生成的。

4. 根据权利要求3的系统，其特征在于，其中进程包括中央的进程类型，对每一种此类进程类型，有一个逻辑进程实例，

25

具有由一个键标识的实例的进程类型，对每一键值，有一个逻辑进程实例，不同的进程类型对它们相应的键，可具有不同的类型，因而具有许多不同的逻辑进程实例，

30 具有由系统本身中的软件安装的实例的进程类型，有一个与此进程类型一起紧密地开发的数据库对象类型，该数据库对象类型的一个实例对应一个逻辑进程实例。

5. 根据权利要求4的系统，其特征在于，其中，对进程类型，

分布单元与进程实例间的关系是以如下所描述的：

- 对于一个中央进程类型，它是微不足道的，因为仅有一个逻辑实例，所以可仅有一个分布单元，

5 - 对于实例是由一个键来标识的进程类型，分布单元的数量与此进程规范一起规定为一个常量，而此关系表述为一个编程的算法，此算法把每一键值映射到分布单元之一，

- 对于实例是由系统本身中的软件安装的进程类型，进程实例在分布单元被组合，这由数据库对象间的关系表述。

6. 根据权利要求 5 的系统，其特征在于，其中，进程包括
10 静态的进程类型，对静态的进程类型，控制系统保证对于每一逻辑实例总有一个物理实例，

动态的进程类型，对动态的进程类型，控制系统仅保证在其它的一些进程要与它通信时才生成物理实例。

7. 根据权利要求 6 的系统，其特征在于，其中，进程包括
15 单一的进程类型，对于单一的进程类型，除了在某些系统活动期间当两个实例协作，但对系统中的其它的进程却作为一个进程之外，对每一个逻辑实例，都有一个物理实例；

复制的进程类型，对于复制的进程类型，对一个逻辑实例，可存在若干个物理实例。

20 8. 根据权利要求 7 的系统，其特征在于，其中对单一的进程类型，把一个分布单元唯一地分配给单个处理器，对一个复制的进程类型，一个分布单元可能在若干处理器上被复制。

9. 根据权利要求 6 或 8 的系统，其特征在于，其中多个进程类
25 型可以共享同一分布单元类型，假定这些类型是某一通用种类的，这些种类包括：

中央的、静态的和单一的进程类型，
键标的、动态的和单一的进程类型，
中央的、静态的和复制的进程类型，
30 中央的、动态的和复制的进程类型，
安装的、静态的和单一的进程类型。

10. 根据权利要求 9 的系统，其特征在于，其中键标的、动态的

和单一的进程类型的实例和一个具有与此进程实例相同的主键的数据库对象实例被用来引用此相同分布单元类型，用于在必要时迫使这样一个进程实例在同一处理器上执行，此同一处理器保持具有对应键的这样一个数据库对象的主复制件。

5

一种计算机控制系统

发明的技术领域

5 本发明总的来说涉及一种计算机控制系统，例如包括在电信网中的一个网元内的此种计算机控制系统。

一个电信网包括网元，即系统，网元协作以把网络业务提供给网络用户。由一个网元支持的多种不同的业务，在较大程度上是由软件实现的，软件由在网元中的一个计算机系统存储和处理。计算机系统存储与
10 网络和业务相关的数据，控制业务，与用户驻地设备和与其它网元交互作用，以及控制本地的传输和交换硬件。

从物理设备的角度看此计算机系统，此计算机系统硬件主要包括多个互连的通信装置和协作处理器以及大容量存储设备，它们由多个不同的设备模块分开。

15

相关领域描述

EP 405,829 涉及一种电信系统，在这种电信系统中软件是借助于相互独立的软件部件以对象的形式实现的。一个在运行系统（**runtime system**）中的功能运行链接器（**function runtime linker**）记录对象并
20 存储一个指向这些对象的数据的数据指针。为与另一个对象通信，源对象传送一个消息到运行系统。该消息包括宿对象的方法的名称和标识。

运行系统仅服务一单个处理器或一组对象，且如果宿对象属于由该运行系统服务的一组对象时，借助于此方法的标识和数据指针来调用宿对象。在一个宿对象是位于另一个处理器中的情况下，运行系统广播此
25 消息到其它处理器。在每个接收处理器中，运行系统搜索它的链接器表寻找此消息的宿对象的符号名，如果找到的话，在消息中的方法标识和运行链接器的数据指针信息的基础上调用宿对象。处理器间消息包括一个源处理器名称，并且当接收到一个处理器间消息时，每一处理器的运行系统存储此源处理器的名称和源对象的符号名。

30 在 US 4,901,231 中，描述了一种在多个处理器上执行的多处理器系统。在一个处理器中的一个用户进程可访问其它处理器中的系统资源。

当一个用户进程访问一个本地文件时，该访问是借助一个用户文件表来完成的。当用户进程访问一个远端文件时，该访问是在一个借助于到一个部分进程的端口表所标识的虚通路上，通过一个端口表，然后通过此部分进程的用户文件表和系统文件表来完成的。

- 5 US 5,187,790 涉及一种具有多个同时运行的进程的计算机系统，包括至少一个服务器进程和多个客户机进程。每一进程有一个表示对象访问权的标识的列表。每一对象有一个访问校验列表，此列表具有用来确定可以访问对象的那些处理器的标识。

10

发明概述

任何计算机控制系统具有一个在至少一个处理器上执行的控制程序。目前对此种系统作出了大量的要求，这些要求有，应用开发应容易，系统应具有极高的可用性，以及应容易升级它的容量，等等。满足这些要求的一种方法是采用分层体系结构来设计系统，在这一分层结构中，

15 在控制系统平台上对应用编程。而这一控制系统平台应该关心的是最普遍的问题，而仅把应用领域的问题留给应用编程者。

在这种体系结构中，应用程序，以及许多操作系统软件，以协作进程的方式执行。

把处理器放置到进程体系结构是一件麻烦事。当把处理器仅看作资源提供者时，要求操作系统对应用程序掩盖各单个的处理器，并给出仅

20 有一个较大的处理器的印象。另外，存在不同处理器的事实 - 各自具有自己的特性和控制把系统连到外部世界的那些硬件设备的特定能力 - 在把应用程序模型化为多个不同进程方面是非常重要的条件。当考虑处理器时，以由处理器提供的资源尽可能得到充分利用的方式来配置系统

25 也是必须的。但是，这应该尽可能不限制控制系统对应用编程者隐藏各单个处理器。

因而，从应用编程者的角度，控制系统能够最好被看作一个同类的平台，此平台不但给进程提供执行和内存资源，还提供不同业务，例如

- 用于数据永久存储的一个数据库，
- 30 - 用于实施管理对象的一个框架，
- 普通操作系统业务，象定时器和时钟，
- 应用程序的管理，

- 启动和重新启动过程、差错恢复等的控制。

遵照本发明的计算机控制系统包括一个处理体系结构，此体系结构包括一个控制系统平台，此平台包括用于处理所述数据库的不同部分的多个控制系统处理器，包括一个管理由处理器提供的资源以及实施由控制系统提供的抽象概念的操作系统，以及进程资源的表示。此处理体系结构保证应用程序以协作进程方式被执行，进程相互间、与一个控制系统间、与此体系结构外的外部世界间交互作用。进程资源表示的实例借助于分布单元分布在多个控制系统处理器上，每一分布单元形成一组进程实例。分布单元属于特定分布单元类型，其中每一分布单元类型描述属于该类型的分布单元的共同属性。

这里一个进程表示一个独立地执行的单元，独立地执行的单元表示它执行程序代码而不管系统中的其它进程在这同一时间正在干什么。每一进程具有它自己的专有状态，即，一个进程不能直接对属于其它进程的数据进行操作。

一个进程可为形成它的进程类型的一个实例的一特定类型。进程类型决定进程具有什么样的功能，不同类型的进程作不同的事情，而相同类型的实例作相同的事情但对不同的数据或不同的硬件单元进行操作。

进程实例可包括执行程序代码的物理进程实例，以及包括抽象概念的形式的逻辑进程实例。系统中的逻辑进程实例的种类由存在于系统中的进程类型、以及它们的属性确定，以及在硬件设备方面由系统的配置确定。

分布单元可组合逻辑的和物理的进程实例。逻辑实例被组合是因为实例和分布单元间的关系是用逻辑实例来表述的，而物理实例被组合是因为当一个分布单元被分配给一个处理器时，所有的与此分布单元相关的物理进程实例在那一特定处理器上生成。

进程还可包括中央的进程类型，对每一此种类型的进程类型有一个逻辑进程实例。还可有具有由一个键标识的实例的进程类型，对每一键值有一个逻辑进程实例。不同的进程类型在它们的各自的键上能够具有不同的类型，因而具有不同数目的逻辑进程实例。具有由系统本身中的软件安装的实例的进程类型，可具有一个与此进程类型一起紧密地开发的数据库对象类型。此数据库对象类型的一个实例对应一个逻辑进程实

例。

对进程类型来说，分布单元和刚描述的此种进程实例间的关系可表述如下。对一个中央的进程类型，它是微不足道的，因为仅有一个逻辑实例，因而可仅仅有一个分布单元。对一个它的实例是由一个键来标识的进程类型，分布单元的数量与此进程规范一起规定为一个常量，而此关系表述为一个编程的算法，此算法把每一键值映射到分布单元之一。对于它的实例是由系统本身中的软件所安装的这种进程类型，进程实例在由数据库对象间的关系所表述的分布单元中被组合。

10 进程还可包括静态的进程类型，和动态的进程类型，对静态的进程类型，控制系统保证对于每一逻辑实例总有一个物理实例。对动态的进程类型，控制系统仅保证在其它的一些进程要与它进行通信时才生成物理实例。

15 还有的进程类型为单一的进程类型以及为复制的进程类型，对单一的进程类型，每一逻辑实例有一个物理实例，除了在某些系统活动期间当两个实例协作时，但对系统中的其它的进程却作为一个进程；对复制的进程类型，每一逻辑实例可能有若干个物理实例。

对单一的进程，一个分布单元被唯一地分配给一单个处理器，而对复制的进程，一个分布单元可在若干个处理器上被复制。

20 若干个进程类型还可允许共享同一分布单元类型，假定此类型是某一通用种类。这些种类包括中央的、静态的和单一的进程类型，键标的、动态的和单一的进程类型，中央的、静态的和复制的进程类型，中央的、动态的和复制的进程类型，以及安装的、静态的和单一的进程类型。

25 一个键标的、动态的和单一的进程类型的实例和一个具有与此进程实例相同的主键的数据库对象实例被优选地作出以引用此相同分布单元类型，用于在必要时迫使这样一个进程实例在同一处理器上执行，此同一处理器保持具有对应键的这样一个数据库对象的主复制件。

附图简述

本发明的实施方案，参考附图，现在描述如下

30 图 1 简略示意了一个电信网，

图 2 是一个简略示意一个网元（如图 1 所示）的软件结构的视图，

图 3 是一个简略示意一个包含在遵照图 2 的网元中的控制系统平台

的结构视图，

图 4 - 8 是用来示意不同进程类型的处理器群的简略图，

图 9 是一个网元的硬件的简略视图，

图 10 是遵照图 9 的网元的管理的简略视图，

5 图 11 是遵照图 9 的网元的资源的简略视图，

图 12 简略示意了想要在包含在遵照图 9 的网元中的控制处理器上执行的进程实例的分布，

图 13 和 14 示意了从一个位于一个控制处理器中的进程实例到在另一个控制处理器中的进程实例的访问，

10 图 13 是具有包含在其中的分布单元的三个处理器的简图，和

图 14 是示意一个寻找过程的流程图。

实施方案的详述

图 1 简略示意了遵照一个基本的 TMN 方案的一种电信网。此网络，
15 在 102 示出，包括网元 104，网元 104 协作以把网络业务提供给用户。由一个网元支持的多种不同的业务，在较大程度上是由软件实现的，软件由在网元中的一个计算机系统存储和处理。计算机系统存储与网络和业务相关的数据，控制业务，与用户驻地设备（在 106 和 108 由电话用户设备示出）和其它网元交互作用/通信，以及控制本地的传输和交换硬
20 件。网络操作系统在 110 和 112、以及用于本地网元 116 的管理的操作系统 114 示出。

从物理设备的角度看计算机系统，计算机系统硬件主要包括多个互连的通信装置和协作处理器和大量的存储设备，它们由多个不同设备模块分开。

25 任何计算机控制系统具有一个在至少一个处理器上执行的控制程序。目前对此种系统作出了大量的要求，这些要求有，应用开发应容易，系统应具有极高的可用性，以及应容易升级它的容量，等等。满足这些要求的一种方法是采用分层体系结构来设计系统，在这一分层结构中，在控制系统平台上对应用编程。而这一控制系统平台应该关心的是最普
30 遍的问题，而仅把应用领域的问题留给应用编程者。

在这种体系结构中，应用程序，以及许多操作系统软件，包括在进程中执行的代码，这些进程相互间，进程与控制系统，以及通过硬件设

备与系统外的其它设备交互作用。这在图 2 概括性地予以示意，图 2 是在网元 202 中软件执行环境的一个逻辑视图。所示的网元 202 接收线路 204，例如用户线，用户线终止于入局硬件设备 206，例如以用于用户线的连接的用户线电路的形式的硬件设备。出局硬件设备在 208 示出。

5 包括许多传输信道的中继线 210 通过硬件设备 208 离开网元 202。在 212 示意了一个交换机。网元 202 中的一个软件系统由用虚线框定的方框 214 示出。所示的方框 214 包括一个由方框 216 示出的控制系统平台，以及在 218、220 和 222 示出的相互间交互作用的进程。例如可以想象进程 218 控制用户线电路 206（它由箭头 224 示出）中的一个，进程 222 控制

10 用户线电路 208（它由箭头 226 示出）中的一个。进程 220 控制（由指向交换机 212 的箭头 228 示出的）此两用户线电路间的直连接。

参照图 3，控制系统平台 214 可包括若干控制处理器 302，它提供执行和内存资源，以及包括操作系统软件 304，它管理由处理器 302 提供的资源以及实施由控制系统提供的抽象化。在 306 示出的进程是一种

15 公用资源。

因而在下文，当讨论操作系统时，指的是图 3 所示出的那种操作系统，并且称之为“所考虑的操作系统”。进程可为形成它的进程类型的一个实例的某一特定类型。进程类型决定进程具有什么样的功能，不同类型的进程作不同的事情，而相同类型的实例作相同的事情但对不同的

20 数据或不同的硬件单元进行操作。

为了描述进程概念的不同方面，进程实例可表征为逻辑的或物理的进程实例。一个逻辑进程实例与进程本身之外的一个特定的实体有关，如一个“功能”、一个用户线电路、一个用户等等。一个物理进程实例表示一个逻辑的实例，它在内存具有一个状态，以及执行程序代码。

25 术语“逻辑进程实例”是一个抽象概念。在一个系统中的逻辑进程实例的种类由存在于此系统中的进程类型、和它们的属性确定，以及在硬件设备方面由系统的配置确定。这是一个恒定状态的问题，不管系统的执行状态如何。一个物理实例是一个逻辑实例的“物质化”，但在一个系统中有什么样的物理实例不仅取决于有什么样的逻辑实例，还取决于

30 于当前的运行状态。在一段时间上可有表示相同逻辑实例的不同的物理实例，而对具有某些属性的进程类型，还可在同一时间上表示同一逻辑实例的多个物理实例。

一个进程类型还可是静态的或动态的。对静态的进程，控制系统保证对于每一逻辑实例总有一个物理实例，对动态的进程，控制系统仅保证在其它的一些进程要与它进行通信时才生成物理实例。

此外，一个进程类型还可以是“中央的”。对这种中央的类型的进程，对每一此类进程类型只有一个逻辑进程实例。

10 进程的另一种类型具有由一个键标识的实例。对此类进程类型，对每一键值有一个逻辑进程实例。不同的进程类型对它们的相应的键可具有不同的类型，因而具有不同数目的逻辑进程实例。对于一种键标的和动态的进程类型，还可以是数据库和进程的“共同分配”。“共同分配”的概念具有下列的背景。

当对一个数据库对象更新时，必须对包含此实例的数据库对象分布单元的某些形式的所有拷贝进行更新。这意味着信息必须传送到若干个控制处理器。如果进行更新的进程实例在与数据库分布单元拷贝之一相同的控制处理器上执行时，信息的传送是最少的。

15 如果一个进程实例是一个键标的进程类型，而且它频繁地更新一个具有与此进程实例相同的主键的数据库对象实例，可通过从进程类型和数据库对象类型两者中参考此相同的分布单元类型来在保持主复制件的控制处理器上迫使进程实例被执行。这一共同分配要求这些类型共享此相同的键类型，而且此进程类型是键标的、动态的那种。

20 还有另一种进程类型可有与数据库对象相关的实例。对一种进程类型，它的实例是通过在控制系统本身中的软件安装的，因而有一个与此进程类型一起紧密地开发的数据库对象类型。该数据库对象类型的一个实例就对应一个逻辑进程实例。

25 最后，一个进程实例可是单一的，或具有复制的实例。当一类型具有“单一的”属性，除了在某些系统活动期间当两个实例协作，但对系统中的其它的进程却作用为一个进程之外，对每一个逻辑实例，最多有一个物理实例。当一类型具有“复制的”属性，在同一时间，每一个逻辑实例，可存在若干个不同位的物理实例。这一属性可仅和“中央的”属性结合。对一个静态的此类进程类型，它含有这样的意思，即一个物理实例在许多处理器中的每一个上生成。对一个动态的此类进程类型，它含有这样的意思，即每次访问一个逻辑实例时，生成一个新的物理实例。

30

当一个进程要请求一个在另一个进程上的远端操作时，它或通过标识一特定的逻辑进程实例或通过标识一特定的物理进程实例，来访问其它的进程。

5 为能够标识一个逻辑进程实例，就必须知道进程的类型和不同实例间的区别，某些类型的一个键值，或数据库对象的一个引用。这通常不是一个问题，因为一种类型的进程都被设计与一特定的其它类型的进程协作。但是也有可能通过一个通用逻辑实例标识符来标识一个“未知”的逻辑实例，例如通过从数据库检索一个这样的标识符。然后将从对要标识的实例比较了解的某些其它进程中写入这一标识符。

10 为标识一个物理进程实例，对通用物理进程标识符，存在一个数据类型。这些标识符，或进程实例引用，仅当物理实例存在时才有效，以及当物理实例本身使引用可用的情况下，通过把它存储在数据库，或通过经远端操作把它送到其它进程，它才可被其它进程使用。

15 一个“中央的”进程实例，或一个用键来标识的进程实例，可从另一个网元中的一个进程被访问到，被访问的进程在此运行的这一网元必须提供一个用来标识此网元的网络地址。缺省情况，仅仅在同一网元中的进程可被访问。由数据库对象标识的进程实例仅在一个网元内才可被访问。

20 上面讨论的进程类型的属性应在进程类型规范中在设计语言中规定。这一规范应规定：

- 类型的名称。
- 它是静态的还是动态的。
- 主键的类型。
- 实例能被复制与否。
- 25 - 对于具有一个主键的类型：一个分布的限制或一个对分布单元规范的引用。
- 对于一个安装的类型：一个对分布单元规范的引用。

下面给出规定三种不同进程类型的进程类型规范的例子。

```

30  PROCESS  TYPE ResourceHand ler IS
        STATIC;
        END  PROCESS  TYPE  ResourceHandler ;

```

```

PROCESS TYPE Traffic Control IS
    DYNAMIC;
    PRIMARY KEY key: Subscriber Number;
    DISTRIBUTION LIMIT 1000;
5  END PROCESS TYPE TrafficControl;

PROCESS TYPE LIC_Handler IS
    STATIC;
    PRIMARY KEY key: REFERENCE TO LIC_Data;
10  DISTRIBUTED AS LIC_Board;
    END PROCESS TYPE LIC_Handler ;

```

一个物理进程实例总是在一个，且仅仅在一个处理器上执行。当一个实例要生成时，控制系统必须以这样的方式即处理器被均匀地利用，来决定在哪一个处理器上此实例要生成。控制系统还必须始终监视生成的实例来把远端的操作等指向合适的进程实例。一个进程接着一个进程地这样做将是不可能的，仅因为存在太多的实例。

上面的问题借助分布单元可以解决，分布单元是用来把一组进程实例组合在同一处理器上。分布单元既组合逻辑的进程实例也组合物理的进程实例。它们组合逻辑的实例，因为实例和分布单元间的关系是用逻辑实例来表述的，但是它们也组合物理实例，因为当一个分布单元被分配给一个处理器时，所实际发生的是与分布单元有关的所有物理进程实例都在此特定处理器上生成。

分布单元和进程实例间的关系的表述方式取决于进程类型的属性。

- 对于一个中央的进程类型，它是微不足道的。仅有一个逻辑实例，所以可仅有一个分布单元。

- 对于它的实例是由一个键来标识的进程类型，分布单元的数量应与此进程规范一起规定为一个常量，而此关系表述为一个编程的算法，此算法把每一键值映射到分布单元之一。

- 对于它的实例是安装的这种进程类型，分布单元也被安装，且进程实例与分布单元实例间的关系在安装期间建立。

分布单元在多个处理器上是否被复制，取决于它们组合的进程的属性。对单一的进程，一个分布单元也被唯一地分配到一个单一处理器，而对复制的进程类型，一个分布单元可在多个处理器上被复制。

5 所考虑的操作系统上的一个应用包括执行代码、在进程的内存中具有一个状态、以及在数据库中永久地存储数据的多个协作进程。一个进程实例仅在一个控制处理器上执行，但它可和其它进程通信而不管它们在其上执行的那些控制处理器。

10 为获取功能性，当硬件设备（包括设备处理器）要被运用时，除了物理限制外，所考虑的操作系统上的一个应用的设计者，无需考虑任何分配问题。进程间的通信由所考虑的操作系统透明地处理。

15 为了获取较好的性能，当设计进程结构和数据库用法时，以及当提供给网络工程师有关软件配置的指南时，应用设计者必须考虑硬件分布。因为信息必须在多个通信信道上传送，不同子网上的多个控制处理器的功能间的协作，当然比此子网内或一个控制处理器内的协作，费用高且花费的时间长。

所考虑的操作系统，通过掌握在哪一控制处理器上可找到进程实例和从应用程序中掩盖哪一控制处理器，以及把进程间的通信和数据库操作指向合适的控制处理器，来实施上述的分布透明性。

20 如上所述的，如果所考虑的操作系统始终监视每个单一进程实例在哪里被分配，就会有許多开销。通过上述的分布单元概念可降低开销。这一方法的好处就是所考虑的操作系统仅需要始终监视非常少的分布单元。

25 一个分布单元是关于一种特定的分布单元类型。一个分布单元类型描述此类型（即进程实例和共同分配的数据库对象实例的类型，如果有的话，将它们分组）的分布单元的共同属性，以及少数其它特性，在大多数情况下描述存在于系统中的此类型的分布单元的确切数目。

30 当根据把一个分布单元类型分给控制处理器的一个分布池来配置系统时，阐述了分布单元如何分布于控制处理器上。分布池理论上是一个控制处理器的任意组合。在一个分布池中可以有任意数量的控制处理器，而一个控制处理器可以是任意数量的分布池的一部分。

对于每一分布单元类型，应该有一个对应的规范，在规范中应以与进程类型规范相同的设计语言来规定分布单元的属性。这一规范应规

定:

- 类型的名称。
 - 进程实例的类型。
 - 共同分配的数据库对象实例的类型。
- 5 - 该类型的分布单元的数量。

下面给出三个不同分布单元类型的例子。

中央进程类型:

```

DISTRIBUTION UNIT TYPE Central_DUT IS
    PROCESS TYPE Central_PT;
10 END;

```

具有键的进程类型:

```

DISTRIBUTION UNIT TYPE Keyed_DUT IS
    PRIMARY KEY key:KeyType;
15 DISTRIBUTION LIMIT 100;
    PROCESS TYPE Keyed_PT;
END;

```

共同分配数据库和进程:

```

20 DISTRIBUTION UNIT TYPE Coalloacted_DUT IS
    PRIMARY KEY key:KeyType;
    DISTRIBUTION LIMIT 100;
    PROCESS TYPE Coallocated_PT;
    OBJECT TYPE Coallocated_OT;
25 END;

```

和上面规范共存的规范:

```

PROCESS TYPE Central_PT IS
    STATIC;
30 DISTRIBUTED AS Central_DUT;
END;

```

```

PROCESS TYPE Keyed_PT IS
    DYNAMIC;
    PRIMARY KEY key:KeyType;
    DISTRIBUTED AS Keyed_DUT;
5  END;

```

```

PROCESS TYPE Coallocated_PT IS
    DYNAMIC;
    PRIMARY KEY key:KeyType;
10  DISTRIBUTED AS Coallocated_DUT;
    END;

```

```

OBJECT TYPE Coallocated_OT IS;
    PRIMARY KEY key: KeyType;
15  DISTRIBUTED AS Coallocated_DUT;
    END;

```

可能有多个进程类型，假定这些类型是同一种类的，它们共享同一分布单元类型。对于每一种进程类型，把分布单元类型分到一个分布池，表示稍微有点不同的情况。对前面提到的几种进程类型，这一点将参考图 4 - 8 更详细地进行描述。

第一种，是中央的、静态的和单一的进程类型，在图 4 中示出。图 4 简略地示意了属于一个分布池的三个控制处理器 402、404 和 406。对于这种类型，仅有一个进程实例 408，因而仅有一个分布单元 410。所考虑的操作系统选择池中的控制处理器之一，即例中示意的处理器 404，分布单元 410 被分给控制处理器 404，且进程实例 408 在控制处理器 404 上执行。如果此处理器 404 发生故障，所考虑的操作系统将从池中选择另一个控制处理器，即例中的处理器 402 和 406 之一，且在该处理器上重新启动进程。

第二种，是键标的、动态的和单一的进程类型，在图 5 中示意。图 5 简略地示意了属于一个分布池的三个处理器 502、504 和 506。如图所示，处理器 502 包括两个分布单元 508 和 510。处理器 504 包括两个

分布单元 512 和 514。处理器 506 包括两个分布单元 516 和 518。

5 每一分布单元包括三个逻辑进程实例，它们中某些可能被“物质化”为一个物理实例，因为某些其它进程要与它通信以接触到它的业务。这在图 5 中已按以下方式示例。对于分布单元 508，三个“纯”的逻辑实例示意在 508₁₁₋₁₃。对于分布单元 510，两个逻辑实例示意在 510_{11, 12}，以及一个物理实例在 510_{ph1}。分布单元 512 示意是包括一个逻辑实例 512₁₁和两个物理实例 512_{ph1, ph2}。分布单元 514 示意是包括两个逻辑实例 514_{11, 12}和一个物理实例 514_{ph1}。分布单元 516 示意是包括两个逻辑实例 516_{11, 12}和一个物理实例 516_{ph1}。分布单元 518 示意是包
10 括两个逻辑实例 518_{11, 12}和一个物理实例 518_{ph1}。

对于这种类型，应用编程者必须提供一个确定分布单元的数量的分布极限，在例中此分布极限为 6，以及提供一个把标识进程实例的键翻译为一个标识一特定分布单元的号码的功能。所考虑的操作系统，如图 5 所示，在池中的控制处理器上均匀地扩展分布单元，以及当一个控制
15 处理器失效时，在此控制处理器中的分布单元移到池中的剩下的控制处理器上，而当此控制处理器功能恢复后再移回来。因为进程实例是动态的，当一个进程实例通过建立一个与它的对话被访问时，所考虑的操作系统将仅仅启动一个物理实例。一个在失效的控制处理器上执行的实例不会自动地重新启动。

20 第三种，是中央的、静态的和复制的进程类型，在图 6 中示出。对于这样一种进程类型，仅仅有一个分布单元，它被所考虑的操作系统复制到分布池中的所有控制处理器，且在每一个控制处理器上启动一个进程实例。这在图 6 中示意。在图 6 中有属于同一池的三个控制处理器 602、604 和 606，每一个包含一个分别复制的分布单元 608、610 和
25 612，它们的各自的进程分别示于 614、616 和 618。示于 620 的第四个处理器不属于其它处理器池，因而不接收任何复制的分布单元。

30 控制处理器 602、604 或 606 之一失效时，在它上面执行的实例就消失直到此控制处理器的功能恢复。当一个复制的进程类型被访问，发送方可仅明确地指出照此复制的进程类型，不是实例之一。所考虑的操作系统选择在与发送方“最近的”控制处理器上的实例，即，如果有一个“最近的”控制处理器的话，在这同一个的控制处理器上，不然在同一个子网中，否则在任何地方。如果有多个控制处理器一样“近”，所

考虑的操作系统随机选择。注意如此复制的进程的实例的内部状态之间相互独立。所考虑的操作系统将不保持它们相互一致。

5 第四种，是中央的、动态的和复制的进程类型，在图7中示出。图7示意了三个控制处理器702、704和706，它们是同一池的仅有的控制处理器。所考虑的操作系统在此分布池中的所有控制处理器上复制一个
10 单个的分布单元，这在708、710和712示出，并且每当访问这样一个进程时，在此池中的某一个控制处理器上启动一个新的进程实例。进程实例714和716在复制的分布单元708中示出，而复制的分布单元710，如图所示，包括三个进程实例718、720和722。复制的分布单元706
15 包括一个单独的进程实例724。

所挑选的控制处理器取决于以与静态的、复制的进程类型的一样的方式发送的进程的位置。

20 第五种，是安装的、静态的和单一的进程类型，在图8中示出。图8示意了属于同一个池的三个控制处理器802、804和806。控制处理器804和806形成一个组。把分布单元类型分给一个分布池只意味着连接到此分布单元类型的那一类型的进程实例可被安装在该池中的任何
25 控制处理器上。但直到在运行时它们被安装后，才有分布单元和进程实例。

30 为了安装一个分布单元，必须选择一个处理器组，例如在图8所示的。一个处理器组是一组从配置数据构筑的控制处理器。当一个分布单元（示于808）被安装时，所考虑的操作系统在所选择的处理器组中，选择控制处理器中的一个，例子中为处理器804，所选处理器组也是分布池的一部分，分布单元类型被分配给此分布池。在这样安装的分布单元808中，在例中示为810和812的进程实例，也在运行时被增加。所
35 考虑的操作系统在所选择的控制处理器上启动在分布单元中的进程实例，如果当前所选择的控制处理器失效的话就选择另一个控制处理器并在其上重新启动进程。

生成恰当的处理组对应用设计者和网络工程师是一件重要的事，这是因为所考虑的操作系统不知道实际的物理限制。

30 安装的进程实例通常与表示硬件资源的管理对象实例紧密相关，且当一个硬件资源被加入到一个系统时，通常需要一个进程实例来控制或处理此资源。硬件资源可能需要以某种方式被管理，因而生成表示此资

源的一个管理对象实例。进程的安装和它的分布单元因而常常与管理对象的生成一起进行。从表示硬件资源的管理对象与表示其它硬件资源的管理对象间的关系，可获得合适的要选择的处理器组。

按照上面刚说的，以及参考图 9 - 12，现在描述一个示例性的安装例子，图 9 示意一个网元的硬件视图，图 10 示意一个网元的管理视图，图 11 示意一个网元的资源视图，而图 12 简略地示意打算在包括此网元中的控制处理器上执行的进程实例的分布。

为了解决在硬件配置中的物理限制问题，一个处理器组的目的是使安装的进程实例在哪些控制处理器上执行成为可能。在图 9 的网元中，如图中所示的，设备 902 和 904 分别包含在方框 D1 和 D2 中。方框 D1 和 D2 是用来分别表示连接到交换机 906 的多个设备。这些包括设备 902 和 904 的设备，仅能从连接到与这些设备一样的交换机 906 的控制处理器 908 和 910 通达它们。另一个交换机 912 连接到两个控制处理器 914 和 916 以及如图所示的分别包含在方框 D3 和 D4 中的两个设备 918 和 920。方框 D3 和 D4 用来分别表示连接到交换机 912 的多个设备，包括设备 918 和 920。

在图 10 的对应管理视图中，图 9 中的处理器 908、910、914 和 916，设备 902、918 和 920，以及交换机 906 和 912，用管理对象表示，这些管理对象被赋给与图 10 中对应单元相同的参考数字，但带有下标 MO。相似的，在图 11 的对应资源视图中，与图 10 中用管理对象所表示的相同的处理器、设备和交换机，用资源对象表示，这些资源对象被赋给与图 10 中对应单元相同的参考数字，但带有下标 RO。

图 10 示意了一个表示分布池的管理对象 1002 如何组合处理器管理对象 908_{MO} 、 910_{MO} 和 914_{MO} 、 916_{MO} 。一个分布单元类型被分配给分布池，将在下面参考图 12 中进一步详细描述。管理对象 908_{MO} 、 910_{MO} 连接到交换机管理对象 906_{MO} ， 906_{MO} 还连接到设备管理对象 902_{MO} 。管理对象 914_{MO} 、 916_{MO} 连接到交换机管理对象 912_{MO} ， 912_{MO} 还连接到设备管理对象 918_{MO} 和 920_{MO} 。

在图 11 的对应资源视图中，表示图 11 的相应资源的资源对象以与图 10 的管理对象相似的方式组织。但图 11 的资源视图，还包括表示作为一个组的控制处理器 908 和 910 的资源对象 922_{RO} ，以及表示作为一个组的控制处理器 918 和 920 的资源对象 924_{RO} 。

一个分布单元类型，它在图 12 中用 1202 简略示意，被分配给由管理对象 1002 表示的分布池，作为一个例子，它包括该类型的分布单元，用方框 1204 - 1218 表示。每个分布单元包括由在各方框中的矩形表示的多个进程实例。

- 5 因此，分布单元 1204 和 1206，每个形成用于处理设备 D1 的一组进程实例，并被安装供在控制处理器 908 上执行。一个用于处理设备 902 的进程实例在 1220 示出。分布单元 1208 和 1210，每个形成用于处理设备 D2 的一组进程实例，并被安装供在控制处理器 910 上执行。一个用于处理设备 904 的进程实例在 1222 示出。
- 10 分布单元 1212 和 1214，每个形成用于处理设备 D3 的一组进程实例，并被安装供在控制处理器 914 上执行。一个用于处理设备 918 的进程实例在 1224 示出。分布单元 1216 和 1218，每个形成用于处理设备 D4 的一组进程实例，并被安装供在控制处理器 916 上执行。一个用于处理设备 920 的进程实例在 1226 示出。
- 15 处理连接到交换机 906 或 912 任一个的一个设备的一个进程实例，因为如上面所描述的、由资源对象 922_{RO} 和 924_{RO} 支持的控制处理器的成组，必须在连接到该交换机的各控制处理器 908、910 或 918、920 上分别运行。因而如果一个新的设备 928 增加到交换机 906，一个处理此新的设备的进程实例应相类似地被安装、被选择以在控制处理器 908
- 20 和 910 的任一个上执行。因此一个表示此新设备的资源对象 928_{RO} 应被安装、连接至资源对象 922_{RO} ，如箭头 930 所示出的，并且此新的进程实例应分配给形成一组意欲在控制处理器 908 和 910 上执行的进程实例的一个分布单元。作为一个例子，这样一个新的进程实例如图 12 在 1228 所示，被安装用于在处理器 910 上执行。
- 25 从位于另一个控制处理器中的一个进程实例，访问一个控制处理器中的一个进程实例，现在将参照图 13 和 14 进行描述。

图 13 示意了三个控制处理器 CP1、CP2 和 CP3。所示的控制处理器 CP1 包含两个分布单元 A 和 B。所示的控制处理器 CP2 包含分布单元 C、D、R 和 S。所示的控制处理器 CP3 包含分布单元 E、F、R 和 S。假定分布单元 R 和 S 具有分配给它们的复制的第三和第四种的任何一种的进程实例。

对于控制处理器中的一个进程，为了能够请求一个对另一个进程的

操作，需要地址信息，此另一个进程可位于另一个控制处理器中。这一地址信息在图下方的表中提供。

5 每一个控制处理器有一个与此有关的第一表，它对所有的控制处理器都公用，表示在此控制处理器上的所有分布单元的分布，还有一个指向它自己的分布单元的第二表。对于每一个处理器 CP1、CP2 和 CP3，第一表表示为 T1，第二表分别表示为 T2_{CP1}、T2_{CP2} 和 T2_{CP3}。在图中可看到，第一表 T1 中的第一列中的每一行都表示分布单元，因而，从第二列表示在第一列中的分布单元所处的那个处理器。而表 T1 的第二行，可以看到分布单元 C 和 D 位于控制处理器 CP2 中。第二表 T2 在第一列的每行表示一个分布单元，而第二列在同一行提供一个指向该分布单元的指针。从表 2 的每行指向一个对应的分布单元的箭头，就可很容易地实现这一点，每一个箭头由一个与它所指向的分布单元相同的标识字母表示。

15 此外，每一个分布单元有它自己的表 T3，T3 包含指向它的进程实例的指针。在图 13 中，此表仅示意了在处理器 CP2 上的分布单元 D，此表表示为 T3_D。可以看到，分布单元 D 包含三个进程实例 P1、P2 和 P3。表 3 的第一列包括三行，这三行表示这些进程实例中相应的一个，而第二列各行有一个表示指向对应进程实例的指针的箭头。以相同的方式，还有很多与分布单元 R 和 S 相关的表，在图中未示出，这些表标识分配给这些分布单元 R 和 S 的进程实例。表 T1、T2 和 T3 形成了操作系统的一部分。

25 现在描述一些访问的例子。假定图 13 中的控制处理器 CP1 上的分布单元 B 中的一个进程实例发送方想要请求一个远端操作，该远程操作或在存在于控制处理器 CP2 中的进程实例中的一个上，如在进程实例 P2 上，或在分配给处理器 CP2 和 CP3 中的分布单元 R 和 S 之一的一个进程实例上。图 14 是一个流程图，它示意了一个由操作系统完成的搜寻过程，根据该过程可以找到所考虑的进程实例。

30 来自位于图 13 中的分布单元 B 的进程实例发送方的消息，寻找要转发的进程实例，出现在流程图的启动处 1402。该搜寻借助于表 T1 在步骤 1404 启动。如果要搜寻的进程实例是位于分布单元 D 的进程实例 P2 的话，表 T1 的第二行可揭示此分布单元可在处理器 CP2 找到。据此，步骤 1406 消除了进程实例已位于自己处理器上的分布单元这一情形。

如果要搜寻的进程实例是复制类型,表 T1 最后一行揭示可在处理器 CP2 和 CP3 任何一个上找到它。

在步骤 1408, 确定进程实例是否是复制的。如果是, 操作系统在步骤 1410 选择与发送进程的发送方最近的控制处理器 CP1 或 CP2 中的一个, 或当它们一样近时, 选择它们中的任何一个, 正如前面所描述的第三和第四种进程。如果没有, 步骤 1410 被旁路。

如果要搜寻的进程实例是 P2, 步骤 1412 转发此消息到步骤 1414, 在这里与处理器 CP2 相关的表 T2_{CP2} 第二行标识一个指向分布单元 D 的指针。步骤 1416 转发此消息到步骤 1418, 在这里表 T3_D 第二行标识一个指向进程实例 P2 的指针。

在步骤 1420 确定进程实例是动态的还是静态的。如果是动态的, 步骤 1422 确定进程实例是键标的还是复制的。如果是键标的, 该进程类型确定为键标的、动态的和单一的。在步骤 1424, 询问“启动?”, 将确定此逻辑实例是否物质化为一个物理的实例。如果不, 该进程实例在步骤 1426 启动, 且在步骤 1428 转发该消息。如果有一个物理实例, 流程转至消息转发步骤 1428, 如箭头 1430 所示。

如果在步骤 1420 发现是静态进程实例, 流程跳过步骤 1422、1424、1426, 转到步骤 1428, 如箭头 1432 所示。

如果在步骤 1422 的回答是“复制的”, 表示当前是一个中央的、动态的和复制的实例类型, 以及一个新的物理类型必须物质化。因而此流程跳过步骤 1424, 按照箭头 1434 转到步骤 1426, 即操作系统启动该访问的进程实例。

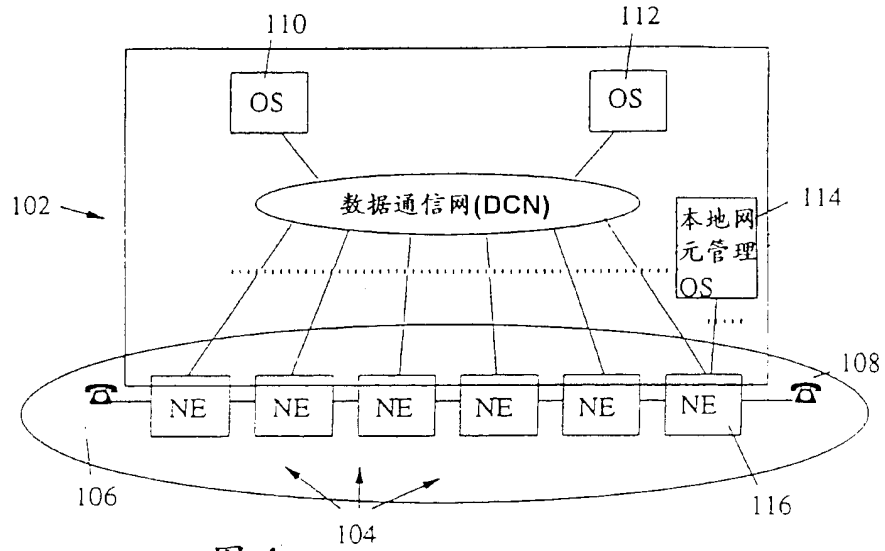


图 1

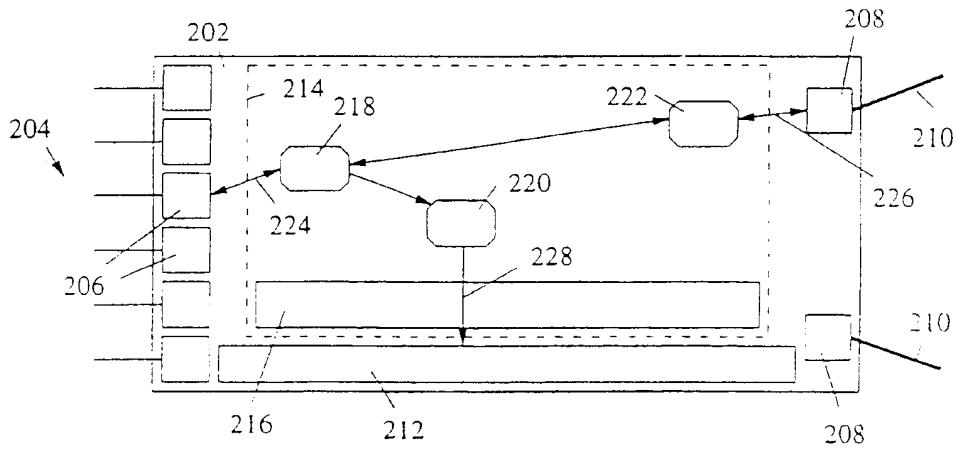


图 2

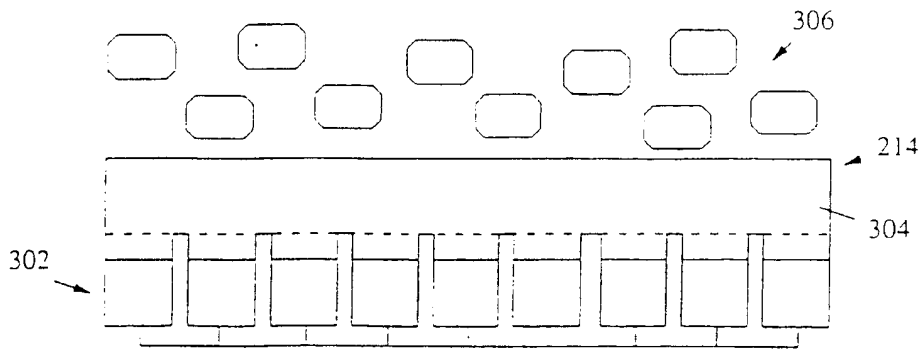


图 3

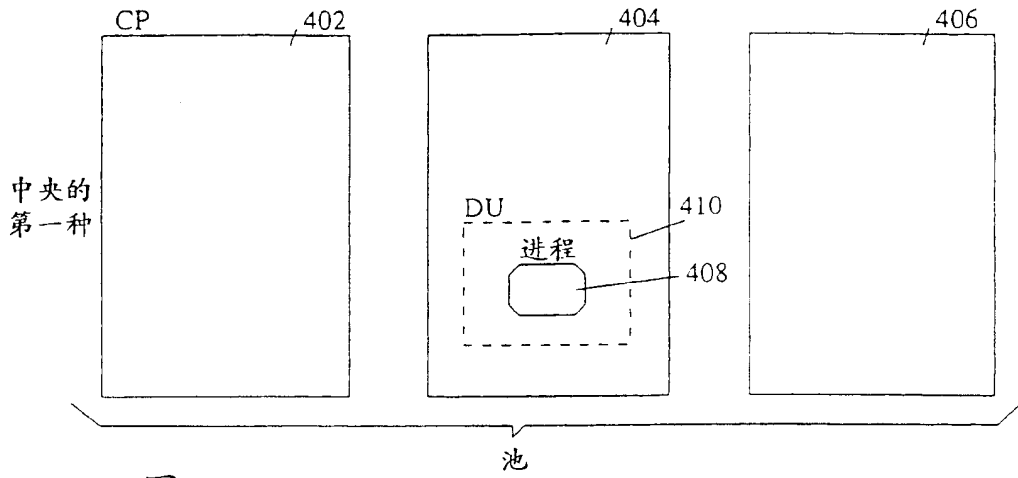


图 4

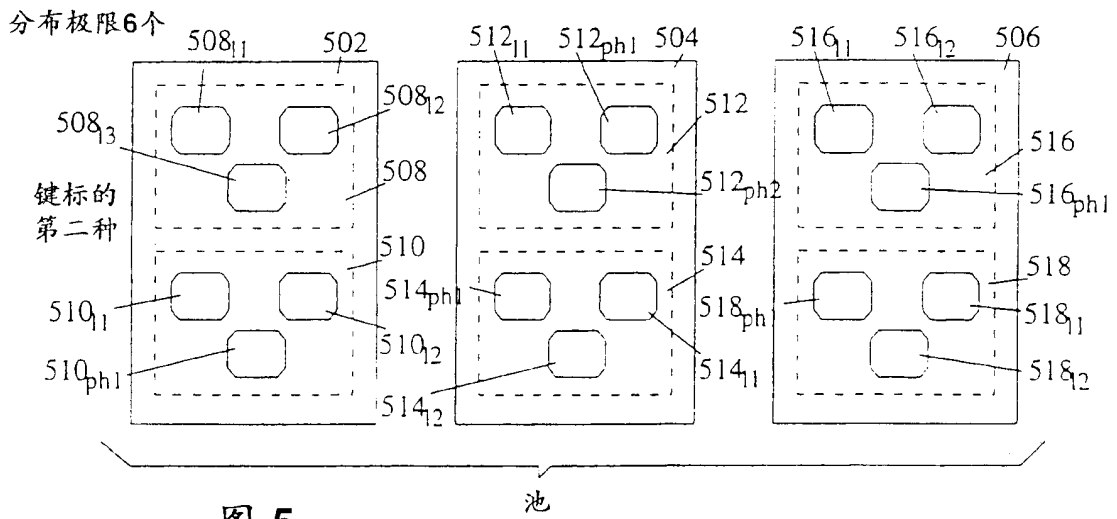


图 5

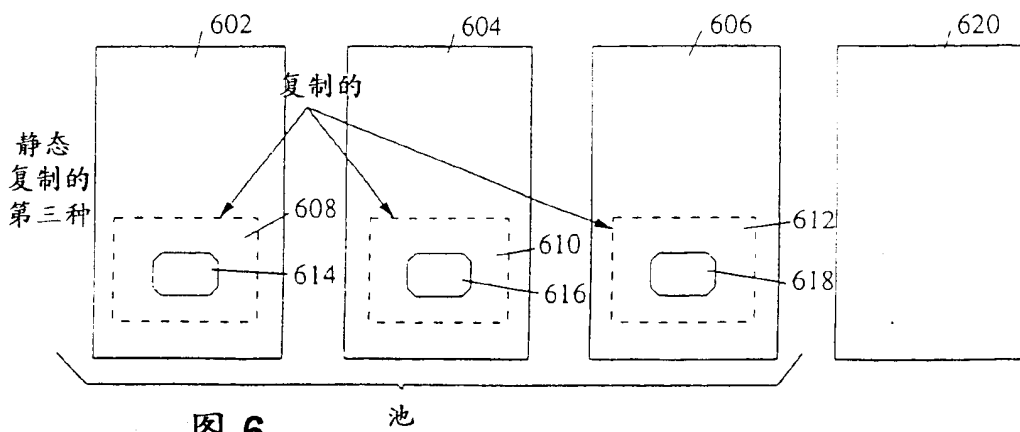
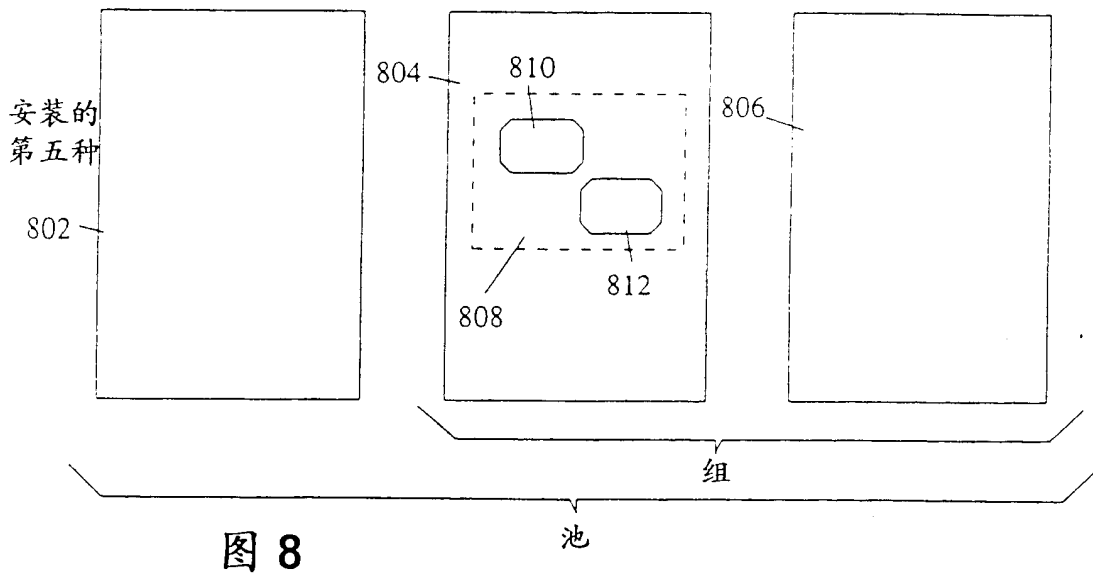
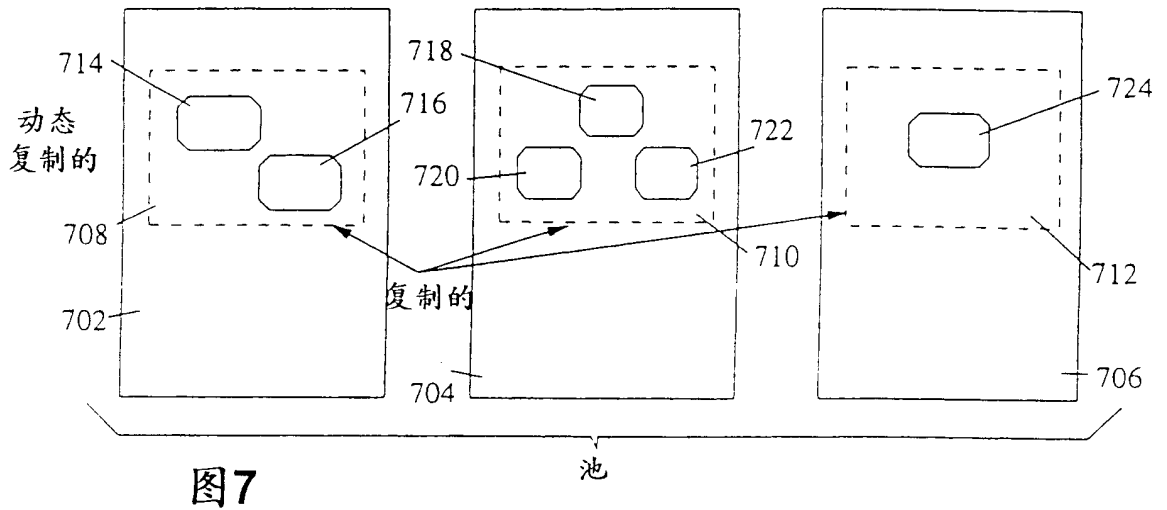


图 6



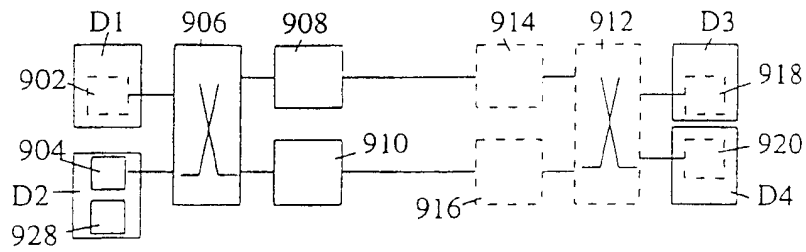


图 9

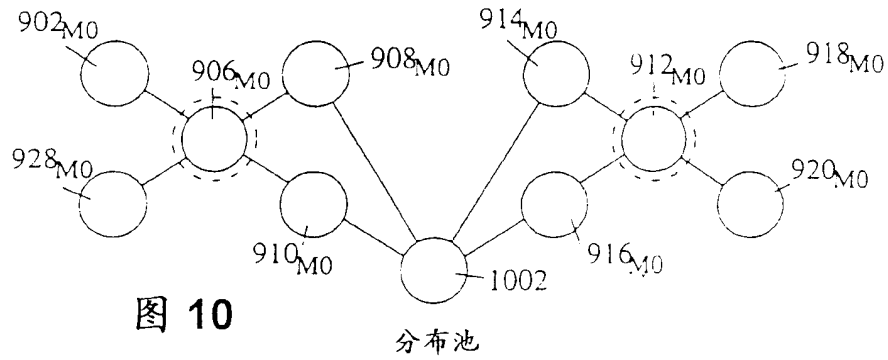


图 10

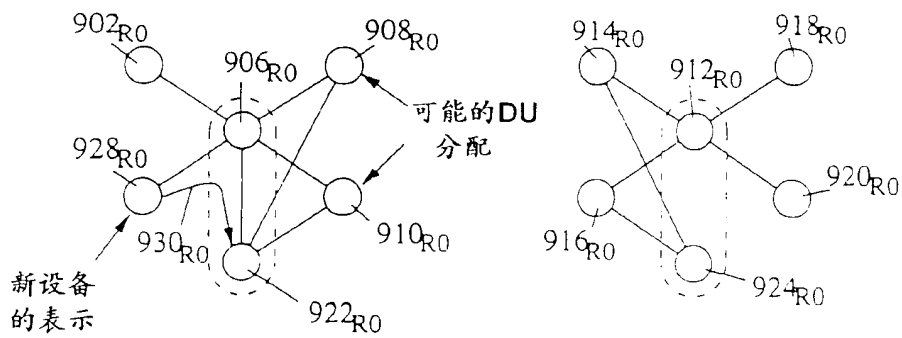


图 11

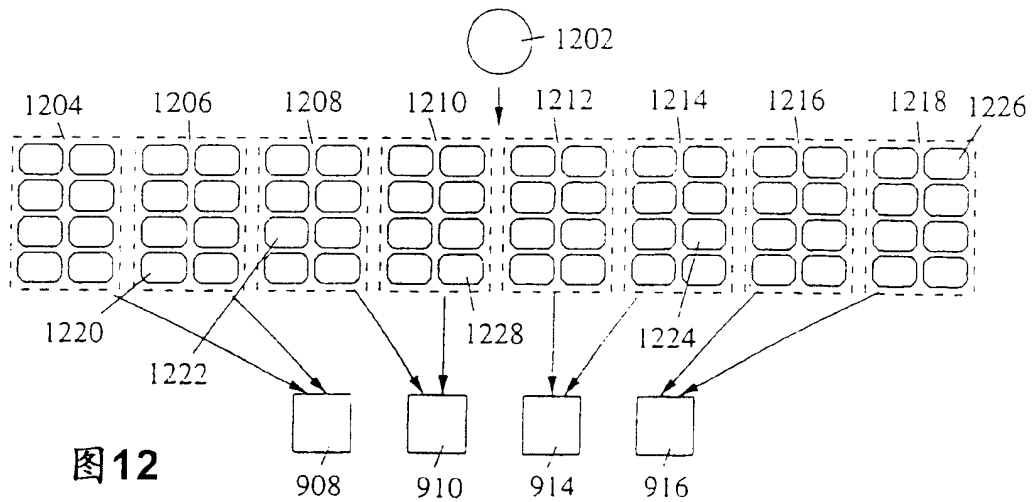


图 12

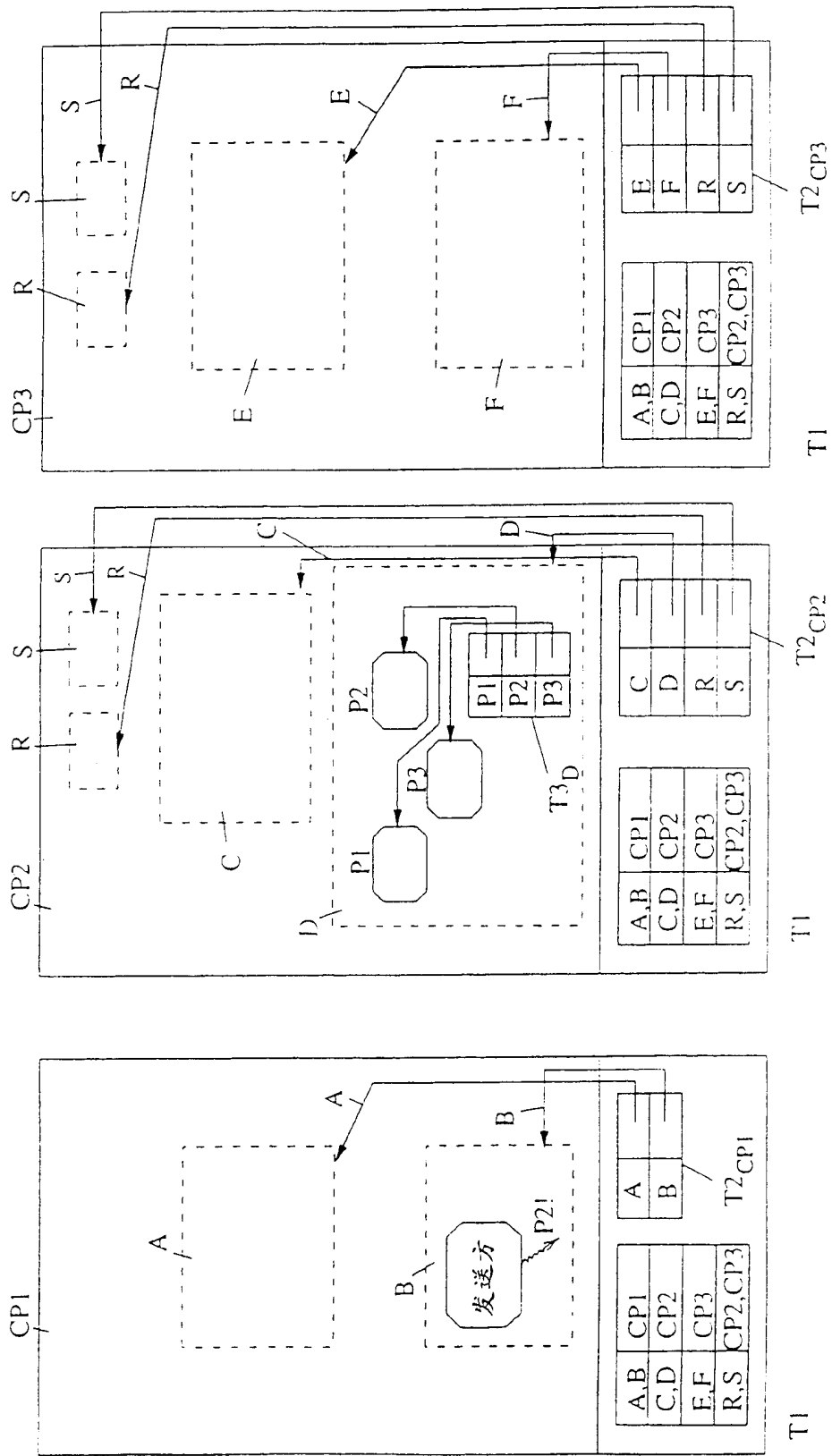


图 13

至P2的地址是(D,P2)

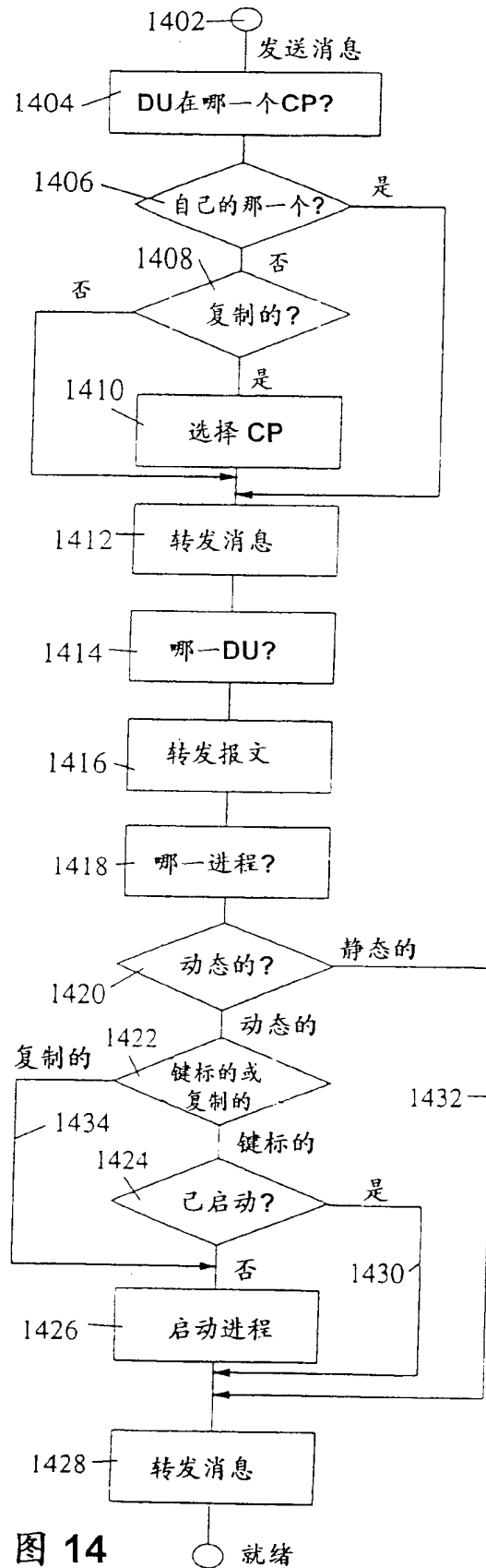


图 14