

# [12] 发明专利说明书

[21] ZL 专利号 98119278.5

[45] 授权公告日 2001 年 8 月 29 日

[11] 授权公告号 CN 1070103C

[22] 申请日 1998.9.17 [24] 颁证日 2001.6.16

[21] 申请号 98119278.5

[30] 优先权

[32] 1997.9.17 [33] DE [31] 19740775.7

[73] 专利权人 福克有限公司

地址 联邦德国弗尔登

[72] 发明人 黑茨·福克 托马斯·库恩

克里斯汀·布莱滕斯坦

[56] 参考文献

EP0178328 1986. 4. 23 G05B19/02

US5032975 1991. 7. 16 G05B15/00

审查员 22 52

[74] 专利代理机构 中国国际贸易促进委员会专利商标事务所

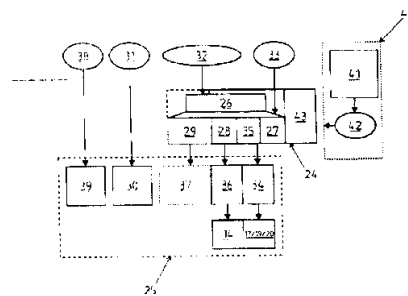
代理人 马 浩

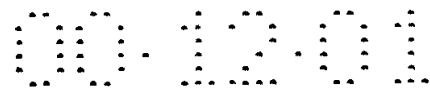
权利要求书 2 页 说明书 6 页 附图页数 6 页

[54] 发明名称 用于具有机器人的托板设备的控制系统

[57] 摘要

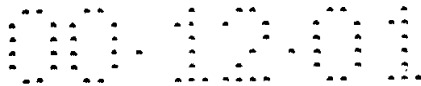
本发明涉及具有机器人的托板设备的控制系统。在已知的控制系统中,机器人通过一数据转换装置被连接到一控制单元。已知控制系统的控制单元包括各种硬件组件。必须花费大量的时间和金钱将其用线连接在一起。各种硬件组件必须以各种方式编程,维护及维修。这是高强度劳动。此外,由于其有限的存储容量,已知控制系统非常不灵活。根据本发明在控制系统中,控制单元被配置为形同一工控机。它实现所有控制单元功能,即涉及机器人过程控制,SPC 和控制台的功能。布线和启动程序数量至最低限度。另外,根据本发明的控制系统易于维修和维护。





## 权 利 要 求 书

1. 一种托板设备，配有至少一个含有多个自动操纵的轴的机器人（11），具有
  - a) 一个控制单元（13）和
  - b) 连接机器人（11）、控制单元（13）和托板设备（17, 19, 20）的数据转换单元，
  - c) 控制单元（13）被配置为一个微处理器系统（23），
  - d) 控制单元（13）包括一个机器人过程控制、一个存储过程控制（SPC）和一个控制台，
  - e) 包括机器人过程控制功能，SPS 功能和控制台功能的用于控制托板设备的控制单元（13）的所有功能由微处理器系统实现，
  - f) 控制单元（13）具有多任务操作系统，其中多个 SPC 任务以并行方式运行。
2. 根据权利要求 1 所述的托板设备，其特征在于机器人过程控制和控制台的功能一起作为第一控制组件（24）在微处理器系统中实现。
3. 根据权利要求 2 所述的托板设备，其特征在于 SPC 功能作为第二控制组件（25）的部分在微处理器系统中实现。
4. 根据权利要求 3 所述的托板设备，其特征在于第一控制组件（24）和第二控制组件（25）在微处理器系统中相互独立实现。
5. 根据权利要求 3 所述的托板设备，其特征在于第二控制组件（25）在一个实时循环中实现。
6. 根据权利要求 3 所述的托板设备，其特征在于第一控制组件（24）通过接口与第二控制组件（25）通信，第二控制组件（25）生成用于机器人（11）的实时兼容控制指令。
7. 根据权利要求 3 所述的托板设备，其特征在于第一控制组件（24）和第二控制组件（25）可以通过文件（31, 32, 33, 38）适应于具有被控机器人的托板设备。
8. 根据权利要求 2 所述的托板设备，其特征在于第三控制组件



(40), 第三控制组件(40)通过一个接口与第一控制组件(24)通信。

9. 根据权利要求8所述的托板设备,其特征在于第一控制组件(24)可以借助于第三控制组件(40)适应于由被控制托板设备处理的包样本或类似物。

10. 根据权利要求1所述的托板设备,其特征在于机器人过程控制功能作为第一控制组件(24)在微处理器系统中实现并且控制台功能作为第四控制组件(47)实现。

11. 根据权利要求10所述的托板设备,其特征在于SPC功能作为第二控制组件(25)部分在微处理器系统中实现并且控制组件(24, 25, 47)相互独立实现。

12. 根据权利要求1所述的托板设备,其特征在于使用一标准软件接口(46)。

# 说明书

## 用于具有机器人的托板设备的控制系统

本发明涉及用于机器人，尤其是用于具有机器人的托板设备的控制系统，其中，机器人和垫以托板的组件通过一数据转换单元连接到控制单元。

专门用于带有机器人的托板设备的控制系统已为人所熟知，在已知的控制系统中，控制单元包括三个不同的结构组件，即一个机器人过程控制系统，一个存储程序控制单元和一个外部控制台。这三个组件间及它们与数据转换单元的连接需要大量费用和相应人员，因为必须小心地将上述三组件的硬件用线连接在一起。这三个组件（硬件）间的不断数据交换特别耗时，因而限制了托板设备的效率，另一缺点是必须掌握不同类型的组件和编程语言以及维护这些组件。已知控制系统的还有一个缺点是，由于其有限的存储容量，仅能处理一定数量的由编程机器人控制的包样本(pack sample)。用于托板设备的公知控制系统需要很高费用用以启动过程，维护及维修，且由于其有限的存储容量，总体上说非常不灵活。

由此出发，本发明基于创建在启动，维护和维修方面不复杂的控制系统，而且在可处理的包样本数量方面，其显示更大的灵活性。

基于解决上述问题，本发明控制系统的特征在于控制单元配置为一微处理器系统，尤其是配置为一工控机，其中微处理器系统实现控制单元的所有功能，尤其是涉及用于控制托板设备的机器人过程控制，一个存储程序控制单元和一个外部控制台的控制功能。

通过使用统一的微处理器系统作为控制单元，由于存储容量不再受单个硬件组件有限存储量的限制，因此控制系统的存储空间有相当大地增加。因而，实际上可以处理任意数量的包样本。这增强了控制系统的灵活性。此外，因为微处理器系统利用软件实现控制单元的所有功能，尤其是涉及机器人过程控制，存储程序控制单元和控制台的功能，只有一个组件，即微处理器系统，必须用线连到数据电路的终端设备上。由此，仅用少量的人力或支出就可启动，维护及维修该控制系统。

在一个有利的实施例中，共同作为第一控制组件的机器人过程控制和

控制台功能由微处理器系统的软件实现。作为第二控制组件部分的存储程序控制单元的功能同样地由微处理器系统的软件实现。因此，只有两个控制组件必须由微处理器系统软件控制和维护。这也简化控制系统的启动操作，维护和维修。

本发明的最佳改进陈述于从属权利要求和说明书中。本发明的具体实施例参照附图予以说明，其中：

图 1 为包括控制单元，带抓爪器的垫以托板的机器人及带倾卸装置的供给桌的托板设备的简图，

图 2 为用于包括垫以托板的机器人的托板设备的控制系统的模块模拟简图，

图 3 控制系统配置方框图，

图 4 控制系统运行时序特性方框图，

图 5 为用于包括垫以托板的机器人的托板设备的控制系统的另一个模块模拟简图，

图 6 所示为两个 SPC 任务及实时核心的模块模拟简图，

图 1 所示为具有机器人托板设备的控制系统 10 的硬件设计视图，例如 US 5 232 332 所述。垫以托板的机器人 11 通过数据转换单元 12 连接到控制单元 13。根据图 1 垫以托板的机器人 11，即其马达驱动控制 14 通过一 I/O 模块（输入/输出模块）15 连接到数据转换单元 12。控制单元 13 其部件也通过一 I/O 模块 16 连接到数据转换单元 12。除垫以托板的机器人 11 的马达驱动控制 14 外，垫以托板的机器人 11 的抓爪器 17 也通过一相应的 I/O 模块 18 连接到数据转换单元 12。除马达驱动控制 14 和垫以托板的机器人 11 的抓爪器 17 之外，用于垫以托板的机器人处理包样本的供给桌 19 及设备在供给桌 19 之上的倾卸装置 20 也通过相应的 I/O 模块 21，22 连接到数据转换单元 12。同样方法，托板设备的其它组件，如板式运送机，托板升降桌也可被连接。数据转换单元 12 最好被配置为光纤总线系统。

依据本发明，控制系统 10 的控制单元 13 形同一微处理器系统，即工控机 23。工控机实现控制具有垫以托板的机器人 11 的托板设备的所有控制单元功能。这包括机器人过程控制，存储程序控制（SPC）和控制台的功能。所有这些功能可用工控机的软件实现。因而，依据本发明利用控制

系统 10 对托板设备的操作控制仅需要工控机 23 作为控制单元 13 被连接到数据转换单元 12 即可。这使布线最少。工控机 23 因此替代目前公知所需的机器人过程控制，SPC 单元和外部控制台这三个独立组件。

图 2 所示为由工控机 23 上软件实现的控制托板设备的功能的模块模拟简图。机器人过程控制和控制台的功能在工控机上共同作为第一控制组件 24 实现。另一方面，SPC 功能在工控机 23 上作为第二控制组件 25 部分实现。工控机上的第一控制组件 24 和第二控制组件 25 是完全相互独立的。

第一控制组件 24 的原理部件是一机器人编程语言解释器，特别是一个 IRL 解释器 26。IRL 解释器（工业机器人语言解释器）用软件实现机器人过程控制和控制台的功能。为实现控制台功能，IRL 解释器有一控制台模块 27。机器人过程控制功能通过一定位模块和用于辅助功能的模块 29 实现。在用于辅助功能的模块 29 的帮助下，垫以托板的机器人 11 可以，例如，指定到一时序最佳位置。

第二控制组件 25 是一实时平台，因而，它以实时方式运行。第二控制组件 25 的部件是一实现 SPC 功能的 SPC 编译器 30。

IRL 解释器 26 和 SPC 编译器 30 均可通过软件文件调整以适应受控的实际垫以托板的机器人 11。对于 SPC 编译器，它含有 SPC 文件 31。对于 IRL 解释器，是文件 32，33。文件 32 也被称为 IRL 程序文件，而文件 33 被称为控制台文件。

第一控制组件 24 和第二控制组件 25 通过相应软件接口通信。由 SPC 编译器 30 生成的 SPC 任务 34 被 IRL 解释器 26 的一个独立 SPC 模块 35 调用。第二控制组件 25 的 SPC 任务 34 和第一控制组件 24 的 SPC 模块 35 在进程中交换数据。第一控制组件 24 和第二控制组件 25 间更进一步的数据交换发生在第一控制组件 24 的定位模块 28 和第二控制组件 25 的定位任务 36 之间。此外，数据通过第一控制组件 24 用以辅助功能的模块 29 和第二控制组件 25 的模块 37 之间的相应接口往复传送。

托板设备的所有外围组件，即抓爪器 17，供给桌 19 和倾卸装置 20 等，通过由 SPC 编译器 30 生成的 SPC 任务 34 来控制。进而，例如，监视供给桌 19 和抓爪器 17 的紧急关闭电路。这里 SPC 任务 34 在一可调实时循环中循环。另一方面，IRL 解释器 26 在其它事件间实现机器人过程控制。与

SPC 编译器 30 相对照， IRL 解释器 26 不在实时方式下运行。 IRL 解释器 26 也不执行任何循环迭代。 IRL 解释器生成用于 SPC 任务 34 和定位任务 36 的功能调用语句。控制台功能由 IRL 解释器作为控制台模块 27 来实现。在每个 IRL 指令后找寻控制台地址。

控制系统的运行时序特性显示在图 4 上。关于图 4 应再次强调 SPC 功能一直于由 IRL 解释器 26 实现的功能之外而独立运行。第一控制组件 24 的 IRL 解释器 26 通过定位模块 28 和辅助功能模块 29 不断处理所有 IRL 指令。它调用定位任务 36 或模块 37，它假定指令由 IRL 解释器 26 生成并生成用于垫以托板的机器人 11 的实时兼容控制指令。定位指令通过 I/O 模块（输入/输出模块） 15 从运行在实时方式下的定位任务 36 处发送到垫以托板的机器人 11 的马达驱动控制 14。在类似情况下， SPC 任务 34 生成用于垫以托板的机器人 11 的抓爪器 17，供给桌 19 和倾卸装置 20 的实时兼容控制指令。数据通过 I/O 模块 18， 21 和 22 传送。

控制系统的配置依据图 3 实施。第一步，受控托板设备机械数据，包括机器人 11，其抓爪器 17，供给桌 19 及倾卸装置 20 的数据，被设置在文件 38 中，文件 38 也被称为配置文件。第二步，一相应 SPC 程序被写入 SPC 文件 31。文件 31， 38 用于运行在实时循环中第二控制组件 25 的配置。第三步，在文件 32 中，即在 IRL 程序文件中，生成一相应 IRL 程序。第四步，用于控制台的配置被写入文件 33。垫以托板的机器人 11 的机械数据及已被设置在文件 38 中的设备外围组件数据通过一相应接口被传送到第二控制组件 25 并准备处理同样的相应模块 39。

由可控垫以托板的机器人 11 处理的包样本控制系统的构造通常由第三控制组件 40 执行。它最好是一个外部个人计算机。第三控制组件 40 包含托板系统 41，在其帮助下可生成用于包样本的文件 42。第三控制组件是一带有脱机托板系统的外部个人计算机。用于包样本的文件 42 由分配到 IRL 解释器 26 的数据存储单元 43 读取。因为 IRL 解释器 26，此前已多次提及，是在工控机上实现的，数据存储单元 43 有相当大的容量，因此实际上包样本任意数量的文件 42 均可被存储及处理。

另外，也可直接在 IRL 解释器 26 处生成作为示教(teach - in)程序的文件 42。

本文描述的机器人控制系统，尤其是图 2 中所示的模块模拟简图，自然地未被设计成多任务操作系统（如 DOS）而是被设计成一单任务操作系统。单任务操作系统的取向使得具有有限变化的单片软件结构和满足某种需求的软件尺度应运而生。

图 5 用于描述多任务操作系统的机器人控制，如 Windows NT。多任务操作系统类型允许必要的模块软件设计。若下面没有其它特别说明，则图 2 的下半部由虚线包围的矩形框中的模块作为第二控制组件 25。

根据图 2 第一控制组件 24 在图 5 中被分解成一个解释器模块 26，一个用于辅助功能 29 的模块，一个定位模块 28 和 SPC 模块 35，及一个机器人控制模块 44。解释器 26 仅用于处理 IRL 程序及处理非运行功能。而除辅助模块 28，29 和 35 之外，解释器模块 26 有与其它程序通信的一个标准软件接口，例如按照 COM 标准（COM=组件目标模型）。移动指令，用户输出等通过此接口发送。通过使用此标准软件接口 46，可以很容易地将独立控制组件添加到控制系统中。人们可按需扩充系统。

控制模块 44 被设计成由运行或启动机器的机械员使用的工具。它用于显示用户输出数据和程序错误。在单任务系统中这些任务由图 2 所示的 IRL 模块执行。另外，由 IRL 模块 26 发送的赋值和移动预处理指令发生在图 5 所示的控制模块 44 中，在单任务系统中它发生在图 2 所示的第二控制组件 25 中。移动指令的赋值用于准备移动并代表排列在控制模块 44 中的块 44a，它通过接口 46 接收数据。也排列在控制模块 44 中的块 44b 代表用于控制模块 44 和接口 46 间交换数据的用户/系统接口。此外，块 44c 集成在控制模块 44 中，它代表传送数据到 SPC 任务 34 的 SPC 模块。

另外，机器人控制模块 44 是与实时任务通信的唯一模块（除显示模块或控制台 47）。一个实时任务是，例如，一个定位任务 36 或 SPC 任务 34。

显示模块或控制台 47 也作为一独立程序运行。因而，不用依赖于与单任务控制系统共用的特殊专门配置的控制台软件。然而可以检索任意（标准）软件包。显示模块或控制台 47 有来自文件 33 的配置数据。与图 2 中显示的系统相对照，它从第一控制组件 24 中分离开并作为第四控制组件通过软件接口 46 实现与第一控制组件 24 的通信。

实时软件被用于多任务机器人控制。这使得以并行方式运行多个 SPC

任务成为可能。与图 2 中表示的相比较实时软件的修改结构可能改变显示在图 2 中的定位模块结构。而图 2 所示的单任务系统中的定位模块 28 被植入实时核心且仅能通过改变和更新整个实时软件才可更改，图 6 所示的用于多任务系统的模块根据图 5 只能以通用基本函数形式被植入实时核心 48。在此方式下，用于机器人轴的特殊程序（或移动）类型的函数，如线性程序 49，PTP（针对针）程序 50 或其它程序 51，作为辅助 SPC 任务 52 实现。它让开发者可能只通过简单地改动 SPC 程序而不修改实时核心 48 就能修改并补充定位功能。因为轴定位在其任务中作为一独立的 SPC 程序运行，所以不再有控制设备所需的来自“正常”SPC 程序的封装。

此外，图 6 示出用于设备控制 54 的第二 SPC 任务 53。设备控制 54 与实时核心 48，特别是与实时模块 55 通信。

与图 2 中所示的系统相对照，图 5 和图 6 所示的系统其优点在于 SPC 任务 52，53 不直接运行于实时核心 55 中。这意味着 SPC 任务 52,53 有任何一变化发生时，对于实时核心不需另行更改。这有效地减少故障源的数量。

新控制系统的优点是显而易见的。使用工控机 23 作为控制单元 13 使得垫以托板的机器人 11 的功能增强，运行速度提高从而取代从前的三个独立硬件组件。因此，整个托板设备可由一个中央控制单元编程及控制。此外，通过使用工控机 23 作为控制单元 13，实际上可有无限存储容量。这使得直接存储包样本上的大量文件成为可能。这使得控制系统非常灵活。另外，可简易快速地将工控机连接到数据转换单元 12，即光纤总线系统上。布线及启动控制系统 10 的费用因此降至最低。

说明书附图

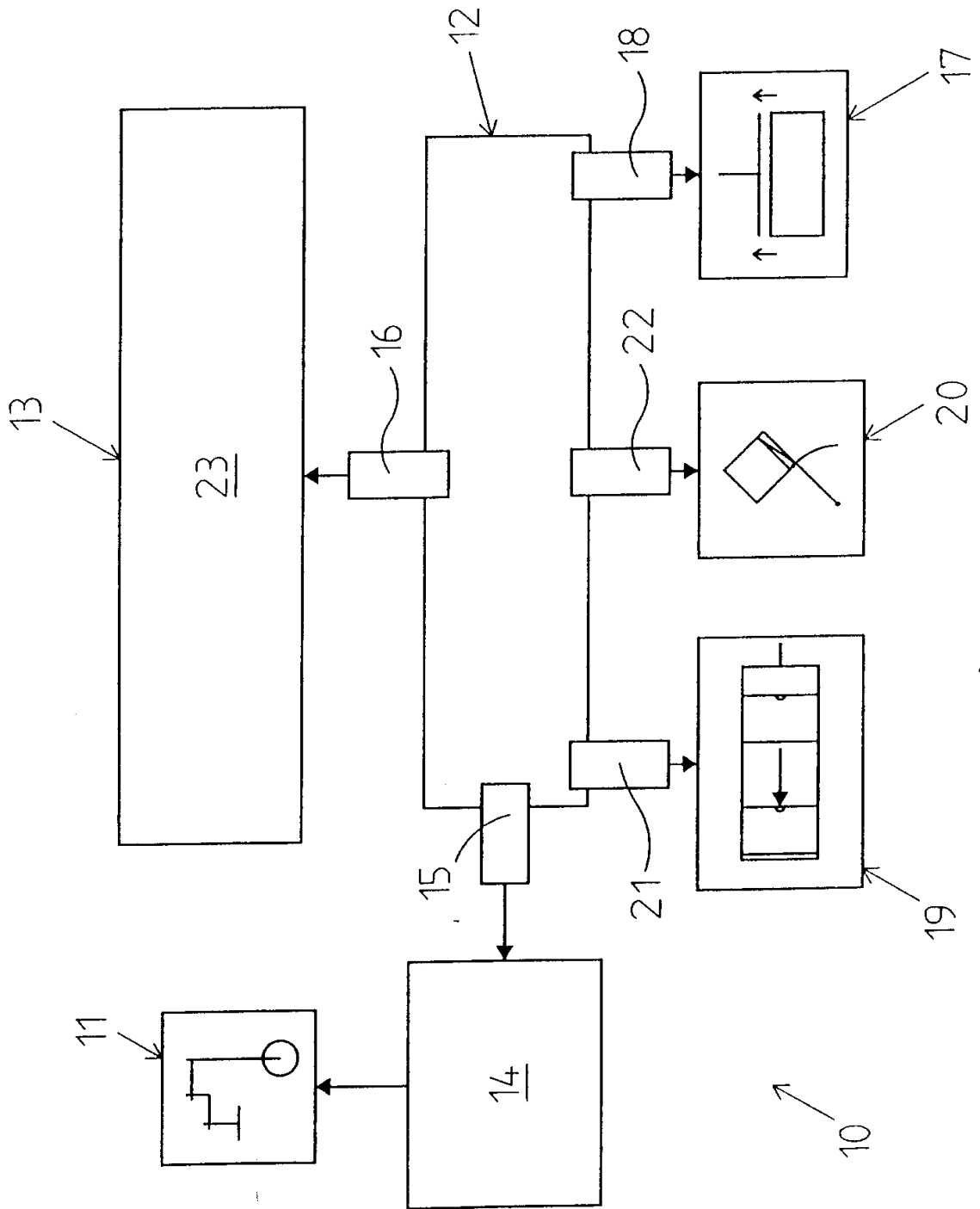


图1

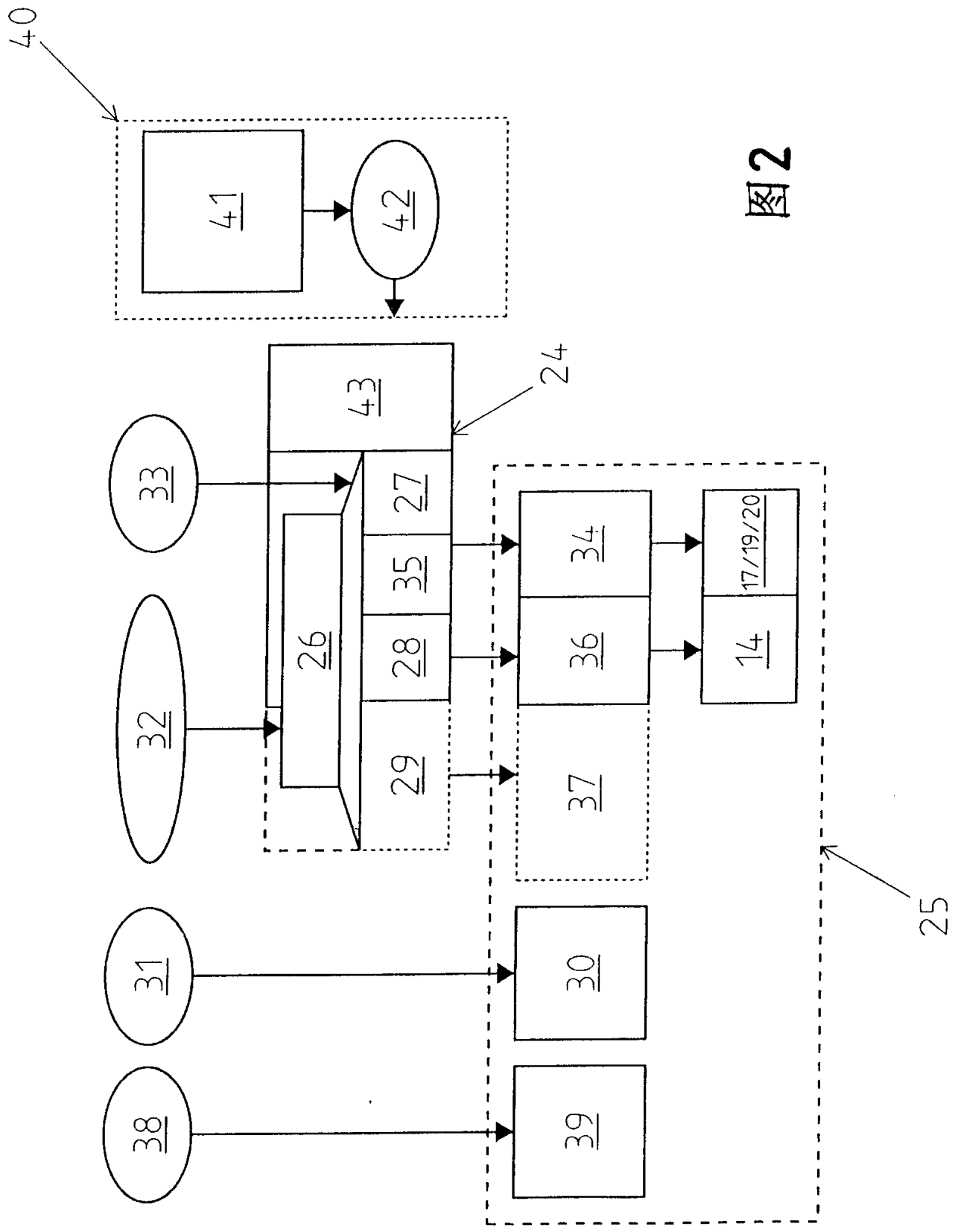


图2

图 3

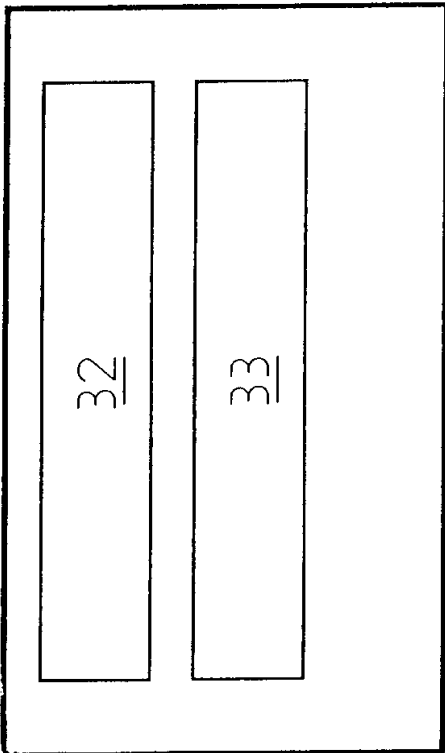
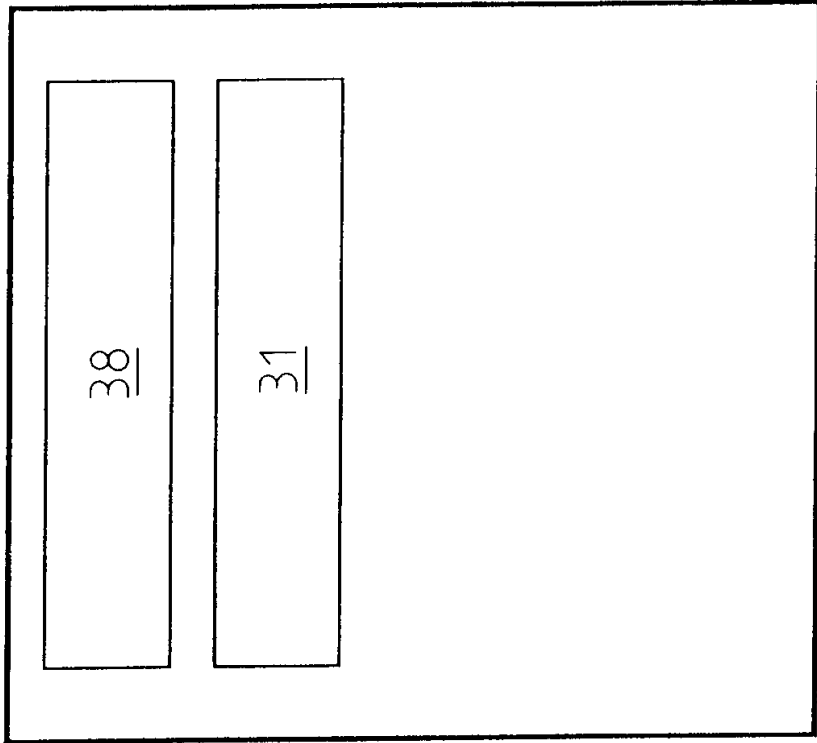
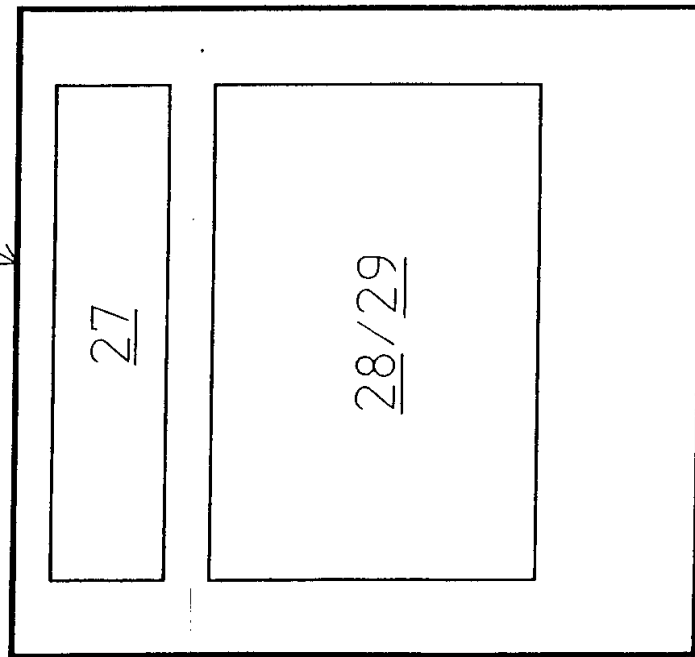
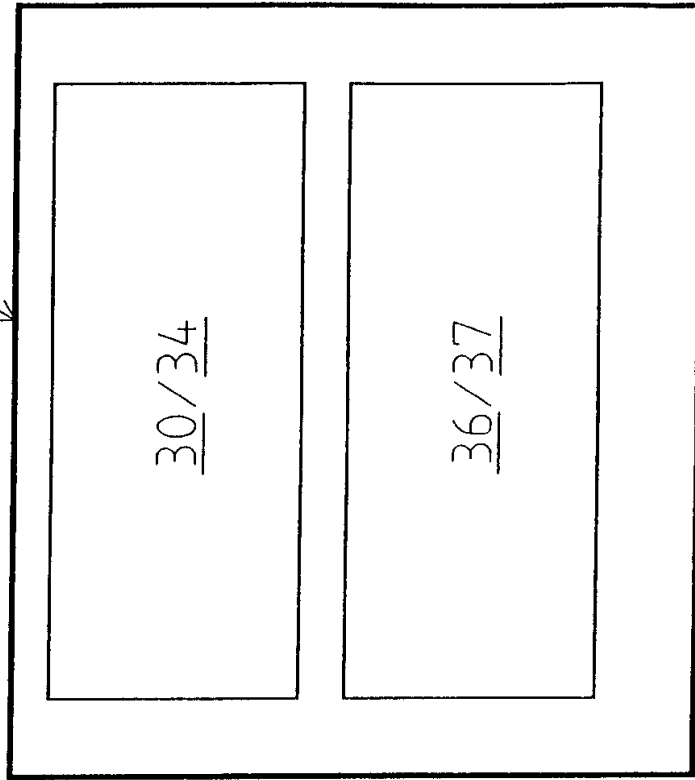


图4

24



25



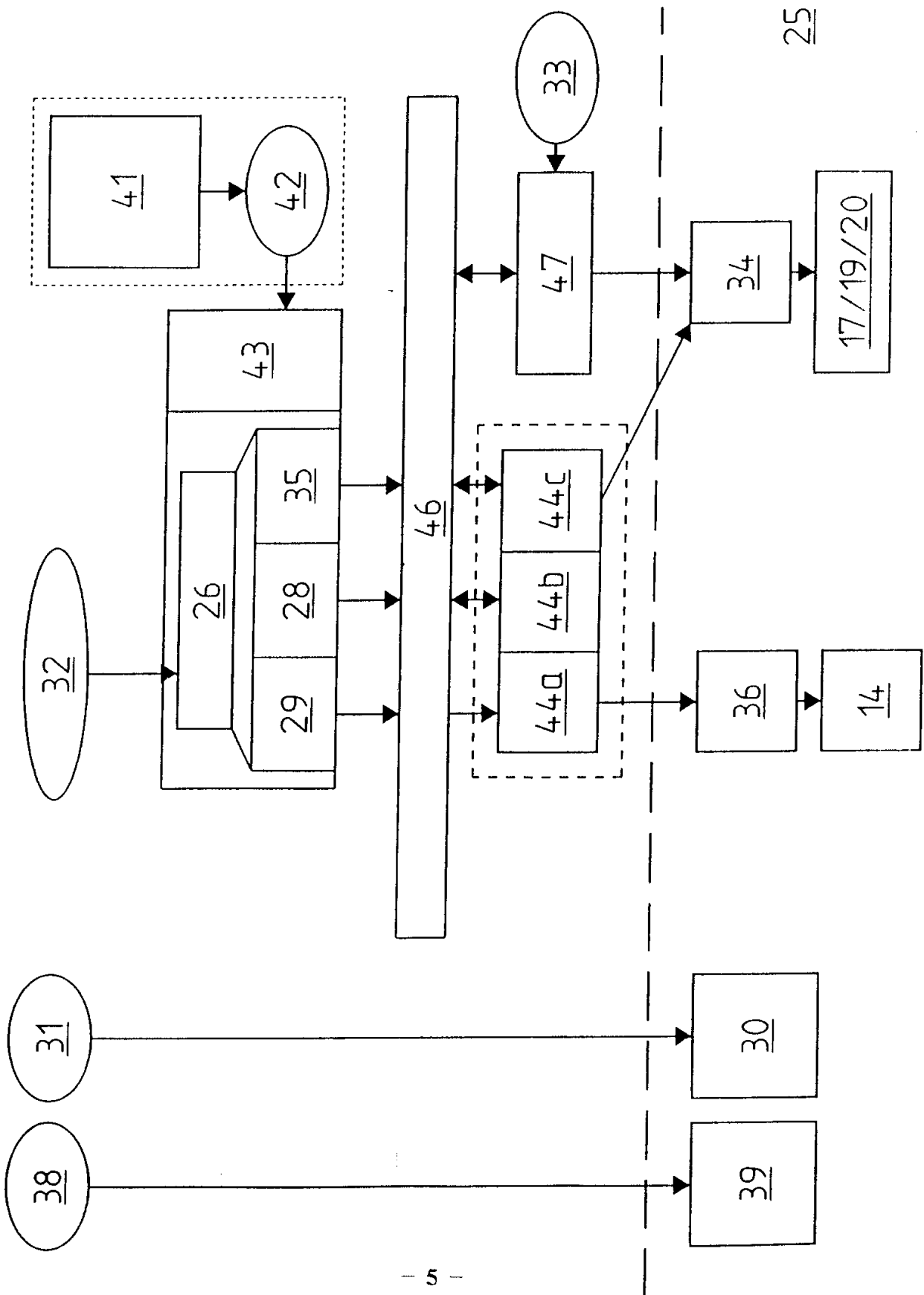


图 5

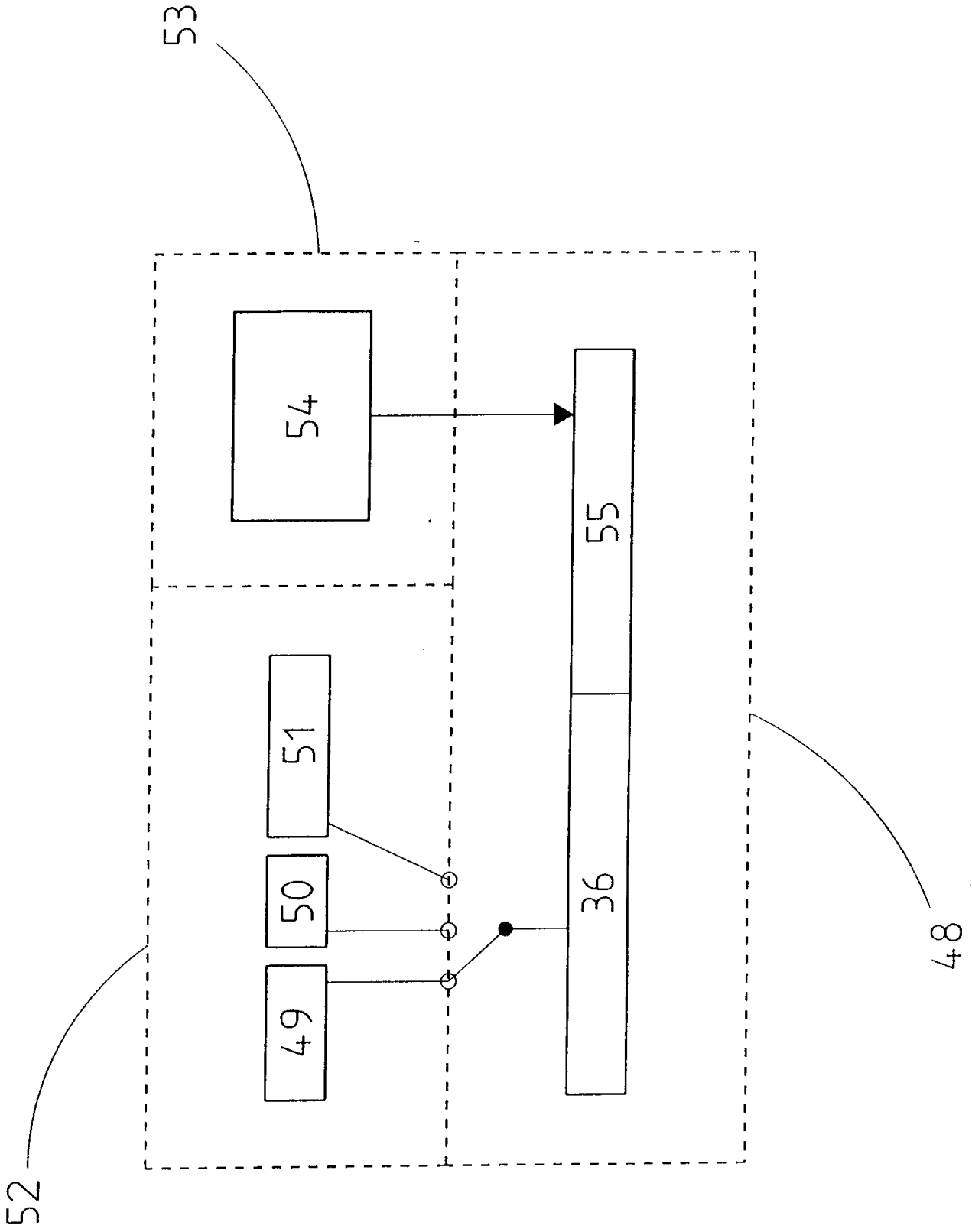


图6