

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G06F 9/44

G06F 9/445



[12] 发明专利申请公开说明书

[21] 申请号 01819423.0

[43] 公开日 2004 年 9 月 1 日

[11] 公开号 CN 1526093A

[22] 申请日 2001.10.4 [21] 申请号 01819423.0

[30] 优先权

[32] 2000.10.4 [33] US [31] 60/237,910

[86] 国际申请 PCT/US2001/031058 2001.10.4

[87] 国际公布 WO2002/029550 英 2002.4.11

[85] 进入国家阶段日期 2003.5.23

[71] 申请人 西门子能量及自动化公司

地址 美国佐治亚州

[72] 发明人 R·胡伯

[74] 专利代理机构 中国专利代理(香港)有限公司

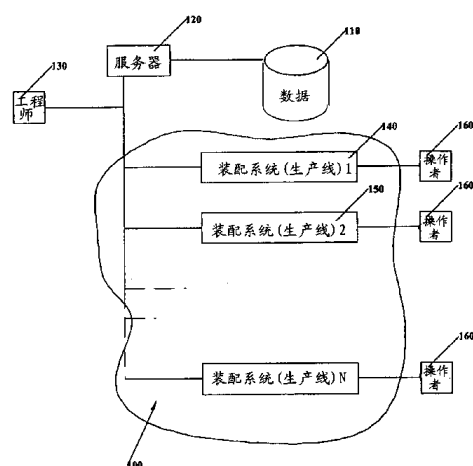
代理人 王 岳 王 勇

权利要求书 3 页 说明书 10 页 附图 7 页

[54] 发明名称 制造系统软件版本管理

[57] 摘要

制造系统软件的版本管理涉及一个两部分数据结构，具有两个识别符，它们分别依据对制造系统程序的顶层和低层子集所做的改变或修改而被独立自动地修改。依据本发明的其它各个方面，管理版本的方法涉及在预先选择的条件下自动分配一个版本标志符，自动通知需要知道这种改变的人员，以及通过修订比较功能的方式管理修订。



1. 用于对制造系统编程的计算机操作的系统中的一种管理程序版本修订的方法，其中一个程序的版本被下载到制造系统，并与一个版本标志符关联，该方法包括步骤：

在制造系统上运行该程序；

当该程序在制造系统上运行时，如果依据预先选择的标准执行，则修订该程序的版本标志符。

2. 依据权利要求 1 的方法，其中版本标志符包括一个版本标号、一个版本识别符和一个状态识别符。

3. 如权利要求 2 的方法，其中版本识别符包括一个顶层版本识别符和一个低层版本识别符。

4. 如权利要求 1 的方法，其中版本识别符包括一个号码，而指定程序修订版本标志符的步骤包括增加版本识别符的步骤。

5. 依据权利要求 2 的方法，其中状态识别符包括一个批准指示符。

6. 依据权利要求 5 的方法，其中制造系统包括至少一条生产线，预先选择的标准包括检查该程序是否满意地运行在该生产线上，且其中，指定程序修订的状态识别符的步骤包括激活该批准指示符。

7. 如权利要求 6 的方法，其中批准指示符是作为不用手工干预在生产线上运行程序的条件而被检测的。

8. 如权利要求 2 的方法，其中状态识别符包括一个发布指示符。

9. 依据权利要求 8 的方法，其中制造系统包括多条生产线，预先选择的标准包括确定该程序满意地运行在这些生产线上，而发布指示符将该程序识别为在多条生产线上使用而发布的。

10. 依据权利要求 8 的方法，其中发布指示符是作为在制造系统任何一条生产线上运行程序不用手工干预的条件而被检测的。

11. 依据权利要求 1 的方法，其中制造系统包括一个电子装配系统。

12. 一种计算机数据结构，用于识别计算机控制的制造系统的程序，其中这些程序包括由分级形式组织的子集，这些子集包括一个顶层子集以及与顶层子集分级相关和互相分级相关的多个低层子集，该数据结构包括：

第一部分，用于指示对程序顶层子集的修订；

第二部分，用于指示对程序任何一个子集的修订。

13. 依据权利要求 12 的计算机数据结构，其中，第一和第二部分中的任一个包括一个号码，该号码增加表示所述修订。

14. 依据权利要求 12 的计算机数据结构，其中第一和第二部分在位置 5 上互相邻近。

15. 依据权利要求 13 的计算机数据结构，其中第一和第二部分位于小数点的分别一边。

16. 一种实现数据结构的方法，该数据结构用于识别计算机控制的制造系统的程序，其中这些程序包括以分级形式组织的子集，这些子集 10 包括一个顶层子集以及与顶层子集分级相关或互相分级相关的多个低层子集，数方法包括：

在指示对程序顶层子集修订的第一部分中，插入指示已经做过这样修订的第一记号；

15 在指示对程序任何一个子集修订的第二部分中，插入指示已经做过这样修订的第二记号。

17. 依据权利要求 16 的方法，其中第一部分包括一个顶层子集的识别符，而第二部分包括一个低层子集标识符。

18. 依据权利要求 16 的方法，其中顶层子集包括一个主对象。

19. 依据权利要求 18 的方法，其中主对象包括电路板的表示。

20. 依据权利要求 18 的方法，其中低层子集包括与主对象相关的子 20 对象。

21. 如权利要求 20 的方法，其中主对象包括电路板的表示，而子对象包括要在电路板上放置的元件的表示。

22. 一种计算机实施的方法，用于管理对用在制造系统控制中的程序的修订，该方法包括步骤： 25

识别已经对程序做了修订；

如果修订，识别该程序是否满足预先选择的标准；如果修订，如果该程序满足预先选择的标准，则依据预先选择的方案自动地选择一个版本标志符；和

30 自动地使被选择的版本指示符与程序代码相关。

23. 依据权利要求 22 的方法，其中的制造系统包括一个电子装配系统。

24. 一种计算机实施的方法，用于管理对用在由至少一个操作者监督的制造系统控制中的程序的修订，其中制造系统通过网络与连接在包含该程序的数据库的服务器进行通信，该制造系统和服务器还通过网络与至少一个客户设备通信，该至少一个客户设备允许被授权这样做的人员与5 该服务器通信，以访问该程序，通过与制造系统相关的接口，该程序还可以经服务器被至少一个操作者访问，该方法包括步骤：

检测发生了对程序的修订；

确定对程序的修订是否由至少一个被授权人中特定的一个人来做；

10 如果确定对程序的修订不是由至少一个被授权人中特定的一个人来做，则通过网络从服务器向客户设备发送一条消息，通知特定人已经做了修订。

25. 依据权利要求 24 的方法，其中至少一个被授权人是一个处理工程师。

26. 如权利要求 24 的方法，其中的制造系统包括一个电子装配系统。

15 27. 在规划具有多条生产线的制造系统的计算机控制系统中的一种管理对程序修订的方法，其中程序的版本被下载到制造系统，该程序与一个版本标志符相关，并且其中程序与制造的商品有关，系统的结构使得根据程序中的信息以图形方式表示制造的商品，该方法包括步骤：

20 接收一个请求，以将与第一版本标志符相关程序的第一版本同与第二版本标志符相关程序第二版本比较，第一和第二版本标志符被分配在多条生产线中使用的各个版本上。

检索与第一和第二版本标志符相关的程序版本；

比较程序版本以识别一组差异；

以图形方式表示商品；

25 在商品的图形表示中包含该组差异的表示，同商品图形表示对比，显示该组差异具有可视的特点。

28. 依据权利要求 27 的方法，其中的制造系统包括一个电子装配系统。

制造系统软件版本管理

- 5 本申请根据 35 U.S.C ¶ 119(e) 要求在 2000 年 10 月 4 日提交的美国临时专利申请 No. 60/237,910 的优先日期利益，该专利的内容整体地在此引作参考。

发明领域

- 10 本发明总体上涉及制造控制系统领域，具体地说涉及一种用于管理制造控制系统的计算机程序版本的系统和方法。

发明背景

- 在制造系统中，如电子组件的制造系统，与制造相关的商品的制造
15 和加工由计算机程序来表示。在电子组件的制造中，在组件设计和在代表这些组件的计算机程序中都可能发生很多改变。一个产品通常包括本身可能改变的元件。而且，可能开发的第二产品只是在一定方面与第一产品不同。计算机程序的改变，不管起因于产品的改变还是起因于软件重新计算，都必须认真管理以避免用不正确的指令运行生产线，因而浪费生产时间和材料。因此，重要的是给每个程序分配适当的版本以跟踪
20 各种版本，并且，只批准和发布那些已经正式授权的程序的版本。

- 当一个产品或产品的派生产品，不仅是在负责的工程师的监督下在一条生产线上制造的，而且是在一个工厂的多条生产线上，或者在地理上分布的几个工厂的多条生产线上制造时，管理程序的各种授权版本的问题就变得更困难了。在这种情况下，计算机程序版本的建立就是一个
25 很大的挑战，更不用说这些版本的批准和发布。

因此需要一种实用和有规则的方法来建立和管理表示制造的商品的计算机程序版本，以及它们的批准和发布。

发明概述

- 30 本发明专注于上面所提出的一种方法的需要，这种方法就是要在计算机操作的制造系统中，管理对控制商品制造的程序版本的修订。依据

本发明系统和方法的应用，可以消除制造生产装配线上可能遇到的时间和零件的浪费。这些问题可能是由对于装配零件而运行的程序的最新状态理解上的不一致所引起的。该理解不一致又可能是由于缺少一种正确的装置来跟踪控制生产线运行的程序的最新版本而导致的。即使使用跟踪单一数字版本号的方法和装置，例如电子板(如后面所称的电路板，印刷电路板)，示例地表示为标志符“V1”，对于跟踪一类程序是满意的，但用来跟踪程序的改变则是不够的。依据本发明的一个方面，版本号按照这样的方式来指定，使得对相同程序所做的任何改变都将反映在版本号中。这通过提供一个附加的字段来完成，例如对于版本号(如 V1.00)中的十进制数位置上，这种表示除了可以反映对主对象本身所做的改变外，还可以反映在程序主要部分的子部分或子对象中所做的改变。下面将参照实施例来说明本发明的各个方面，在该实施例中，带有电子元件的印刷电路板由一个面向对象的计算机程序所代表。但是本发明的这些方面也可以应用到其它商品的制造中，还有实际上分级的其它形式的计算机表示法中。

对代表制造商品的计算机程序的主对象可以改变，或者对该程序的一个或多个子对象做出改变。依据本发明的版本管理系统还可以区分不同的子对象。本发明提供一种反映特定子对象中改变的方法，例如，包含与产品相关的设置指令的对象。这可以通过识别由其唯一名称设置的每个程序的配方(Recipe)或指令集来完成，使得对于在主配方改变之下的子配方，这些改变会反映在标志符或其它状态识别符的一个部分中。

但是，在程序开发期间，或在程序或生产线本身调试期间的每一种改变最好不反映在版本标志符中，否则，将很难管理可能发生的许多改变。而且不必要跟踪对各个产品没有任何影响的改变，直到发现这些改变是正确的并被用在了生产线上。批准和发布的特征还包括运行生产线的工厂的类型。

于是，在一条生产线的工厂中，对装配机器编程以制造产品的工程师还可以是设置和运行该生产线的人员。因此，工程师/操作者对任何改变很熟知，并且可以创造、批准和发布与他/她认为合适的这些改变相一致的程序版本。

在具有几条生产线工厂，或者具有地理分布的制造操作，或者具有几个工厂的企业的情況下，做出代表一个产品的程序的工程师可能与运

行和设置生产线的任何人员不同。在这种设置中，产生的问题是谁负责发行用于在生产线上运行的程序的正确版本，如何发行正确版本，以及谁来批准运行生产线。本发明提供了一种发布程序，有助于确保只发布最新工作的和批准的程序。即使对程序做了很小的改变，都将通知使用者(如处理工程师)对程序做了修改，而该修改需要批准。在本发明该方面的实施
5 实施例中，允许发布一个程序的特定版本是依赖于生产线种类的：对于一条生产线的发布，可以允许所有人员(操作者、处理工程师、管理者)；对一个工厂的发布，允许处理工程师；而对于一个企业的发布，则允许企业总部的特定工程师。就在本发明中揭示的程序还根据在哪里和如何
10 发布版本来划分使用者应负的责任级别。

在本发明的第一实施例中，揭示了一种管理对制造系统程序代码版本的修订的方法，其中特定的程序版本被下载并运行在制造系统中，如果程序依据预先选择的标准运行，那么就指定修订状态标志符。

本发明的另一个方面是关于计算机数据结构的，用来识别计算机控制的制造系统的程序。在本发明这个方面的实施例中，程序包括由彼此
15 以分级形式而组织起来的子集，这些子集包括一个顶层子集和多个与顶层子集分级地相关和彼此相关的低层子集。第一部分表示对程序顶层子集的修订，而第二部分表示对程序任何低层子集的修订。在此将说明实现数据结构的方法，在表示对程序顶层子集修订的第一部分中，插入第一
20 记号来表示已经做了这样的修订，相似地，在表示对程序顶层子集修订的第二部分中，插入第二记号来表示已经做了这样的修订。

在本发明的另一方面，说明了一种计算机实现方法，用于管理对使用在制造系统控制中的程序的修订。该方法涉及识别已经对程序做了修订；如果修改，识别该程序是否满足预先选择的标准；如果修改，并满
25 足预先选择的标准，则将依据预先选择的方案自动地选择一个版本标志符；并且自动地使被选择的版本标志符与程序代码相关。

另外揭示了管理对程序修订的方法，该程序使用在由至少一个操作者监督的制造系统的控制中，以及使用在多条生产线的控制中。在前一种方法中，制造系统通过网络同服务器通信，该服务器连接到包含该程
30 序的数据库中，制造系统和服务器还通过网络与至少一个客户设备通信，为了访问该程序，通过被授权这样做的人，至少一个客户设备允许与服务器通信，由至少一个操作者，通过与制造系统相关接口的，该程序也

是可以经服务器访问的。通过网络检测发生的对程序的修订，并且确定对程序的修订是否由至少一个被授权人中特定一个人所做。如果修订不是由至少一个被授权人中的特定一个人所做的，那么，通过网络从服务器向客户设备发送一条消息，以通知该特定的人已经做了修订。

- 5 在后一种方法中，程序的一个版本被下载到制造系统中，以测试与状态识别符相关的程序，其中该程序与制造的商品相关，制造的商品可以根据程序中的信息以图形表示。然后，接收一个请求，以将与第一状态识别符相关程序的第一版本同与第二状态识别符相关程序的第二版本进行比较，第一和第二状态识别符被分配在使用在多条生产线的各个版本上。接着，检索与第一和第二状态识别符相关的程序版本，比较程序版本以识别一组差异，这些差异显示为具有同商品图形表示相反的可视特性。

- 15 因此，本发明提供一种计算机实现的方法，管理对使用在制造系统控制中程序的修订。这通过提供修订/版本管理两部分的数据结构来实现；一种完成两部分数据结构的方法；一种管理修订的方法，涉及在预先选择的条件下自动分配版本识别符；一种管理修订的方法，涉及自动通知需要知道这种改变的人员；以及一种管理修订的方法，涉及修订比较功能。本发明的一个目的是减少时间和零件的浪费，并且因此促进生产线的生产率。从下面的说明中，本领域的普通技术人员将会清楚本发明的其它目的和优点。

附图简述

图 1a 是依据本发明，表示版本标志符数据结构的实施例的符号表示。

- 25 图 1b 表示依据本发明版本标志符的实施例，该实施例表示对于主对象和子对象的版本号两级表示。

图 2a 表示在构建代表移动电话处理流程的环境下，一个依据本发明的主对象和子对象创建例子。

- 30 图 2b 表示依据本发明，构建移动电话处理流程，以及为具有对象和子对象程序来制定版本方法的实施例。

图 2c 表示依据本发明，构建移动电话处理流程，以及子配方中的改变对主配方版本标志符的实施例的影响。

图 3 是依据本发明, 在单条生产线上表示版本跟踪方法的实施例的图表。

图 4 是依据本发明, 在多条生产线的工厂中表示版本跟踪方法的实施例的图表。

5 图 5 表示依据本发明, 在与数据库、服务器和处理工程师相通信的网络上, 计算机控制的制造系统的示意图。

本发明的详细说明

图 1a 表示用来识别一个程序版本以及程序版本状态的数据结构标志符, 该程序包括主对象和子对象, 主对象和子对象一起表示制造的商品, 还有与该商品相关的制造程序步骤。在说明的实施例中表示为大写的“X”(01)的第一字符是版本标号, 可以是 V, 它是“Version”的第一字母。在版本标号之后的顶层版本识别符(02)表示主对象被改变的次数, 即主对象的修订号, 而在小数点之后的低层版本识别符表示子对象的修订号。15 在该号之后的大写“Y”表示特定版本的状态。因此, 使用在电子元件生产线上处理的印刷电路板为例, 在图 1b 中的记号代表该板的第二版本, 在该电路板上带有子对象元件的第三个修订, 并且, V2.03R 是发布的版本, 这里的“R”代表“Released”。如果它是一个下面详细说明了批准版本, 那么该记号就应该表示为 V2.03V。本领域的普通技术人员应该理解, 一般地, 在图 1a 和 1b 中这样的标号可以表示一个程序的任何主部分(例如对象)和子部分(子对象的)的状态和修订版本。而且, 在制定版本方法的该实施例中(虽然不是一般地), 版本号至少有两个增加的数字, 中间由小数点分开。小数点左边的第一数字代表主对象修订号, 而第二数字代表与主对象相关的任何子对象修改的修订号。20

25 给对象如电路板分配版本号, 是从初始编程开始。图 2a 是一个示意图, 表示为移动电话电路板装配而对计算机操作的系统编程的初始步骤。该电路板可能是具有要求的布线层的多层板, 并在其上提供有安装点, 其它元件如电阻器、电容器、晶体管、集成芯片和小外形晶体管(SOT)等在这些安装点使用如球窝接合焊接、布线焊接和本领域熟知的其它技术来焊接到电路板表面上。还应当理解, 这些各种元件将通过各自的送入器被送入定位机中。而且, 该系统编程为一个具体的生产线。这样, 通过30 使用特定的生产线的机器设置, 将将对计算机控制的系统编程为在电

路板上布置特定元件，在这种情况下是指移动电话电路板。因此很显然，最终装配的电路板具有不同的特性或特征，这些特性或特征不仅依赖于布置在电路板上的元件类型，而且依赖于生产线上使用的特定机器设置。换句话说，被编程到计算机控制的系统中的全部指令或配方必须包括一个子配方，该配方有元件列表、这些元件位置列表，同时还包含另一个配方，描述生产线设置特性，以图 2 所示的情况为例，包含送入器特性。

当重新对具有新对象的生产线编程时，重要的是要注意到新对象被自动地分配版本 V1.00，而不管子对象的版本，这将在下面实施例的说明中以较大篇幅说明。于是，在构建图 2a 表示的移动电话处理流程中，首先在程序中创建子对象：也就是包形式(10)，即区域的几何形状，包括晶体管元件将被放置进去的空间，元件(20)本身，位置列表(30)，也就是元件将被安放在板上的位置；电路板(40)。每个子目标都具有版本 V1.00。相似地，在移动电话主配方的子配方一侧，子对象也被分配版本 V1.00：也就是用于将元件送到电路板位置上的送入器(55)，以及移动电话线路的设置。最后，主对象(70)，也就是对于子配方中所有指令的主程序(70)也得到版本 V1.00。即使子对象是不同版本，主对象还是得到版本 V1.00。此时的版本还不是发布的制造版本。

但是，一旦主配方(70)首先被创建版本 V1.00，在子对象中的任何后续改变和修改都将反映在从改变后的子对象向上，直到并包括主配方的所有子对象的修订号中。于是，如果利用具有带有包形式因数(10)的元件(20)的配方 V1.00 而构建的移动电话电路板由于该形式因数中所规定的不充分的公差而有缺陷，就可以分配一个不同的形式因数，从而将包形式版本改变为 V2.00。任何不包括其它子对象，因而也不受其影响的子对象，如形式因数情况下那样，都将使其版本号的第二组号码恒定在.00 上。但是，如以上所指出的，任何被其它子对象中的改变影响的子对象都使它们的版本号增加.01。因此，在图 2b 中表示的版本序号为 V1.01。

在图 2b 右边的任何改变，也就是在同一个图右边(80)的全体子配方中的任何改变，其特征可以是电路板配方，确实还影响左边(90)的子配方，其特征可以是设置或生产线配方。由于包形式现在是 V2.00，所以包含该包形式的送入器具有版本 V1.01，并且设置子对象也是这样。例如，如果由于某些原因修改了送入器，并变成版本 2.00，这将影响子对象配方，并且，主配方或对象(70)也将使其版本增加.01 到 V1.02。这表示在

图 2c 中。从而主对象，作为设置配方的上方，在其子对象中表示出两种改变。在其中产生改变的子配方最好应该给定唯一的名称，例如在图 2b 左边的“设置”或“生产线”配方，以及同一个图右边的“电路板”配方，这样，用户可以跟踪改变或修改发生的地方。

- 5 在计算机控制的制造系统初始编程后，任何后续的修改，即编辑对象，依据被编辑的版本是否发布，就具有不同的意义。首先，当版本 1.00 下载到生产线上时，生产线上的机器只允许一个电路板穿过生产线，但是只限于在“试车”模式下。第二电路板被允许移动到机器的输入传送机中，但是要求使用者在该电路板可以被装配之前按下开始按钮。使用者将被要求检查整个电路板并确定该板是否没有任何错误地装配。为了
- 10 有助于检查各个修改，被修改参数图形方式的打印输出以及结构打印输出会显示在站上，或者被打印出来以有助于发现对早期发布版本的修改。使用者一同意该程序，根据如下将进一步说明的允许级别，该版本就被发布或批准，并在版本的最后分别接收到一个“R”或“V”，然后该版本
- 15 被冻结，按下开始按钮使每个电路板通过生产线之后，生产线就不再需要手工干预了。这样，当正在编辑最后发布的版本时，该版本会自动地加 1 并将最后的数字设置为.00。相似地，当正在编辑较早的版本时，版本的第一数字将增加 1，而最后两个数字设置为.00。当正在编辑未发布的版本时，版本号不增加。

- 20 当正在编辑未发布版本的子对象(发布的或未发布的)时，主对象的版本不增加。另一方面，当正在编辑发布版本的子对象时，版本加 1 并将被标注为未发布。然后，当发布相同的子对象，甚至作为不同配方的部分而发布时，对于子对象发布的区域，将自动发布主对象。

- 25 本领域的普通技术人员将认识到最重要的是使最后批准和工作的程序运行在生产线上。如果对工作的程序做了稍微改变，那么，依据本发明，使用者自动被通知检查或批准那些修改。依据使用者具备的“允许”或批准的权限，可以允许使用者来使生产线的产品批准，或者发布工厂或全球的产品。

- 30 通过授予使用者批准和发布的允许级别，区分批准和发布处理。发布的对象(例如配方)对于具有多条生产线的整个工厂，或者甚至多个工厂是有效的，而批准对象仅对一条生产线有效。发布只能通过具有较高允许级别的被授权使用者给予对象。对比之下，批准可以由具有较低允

许级别的人员来做，例如操作者，允许这些人员修改产生问题的程序，然后继续运行该生产线。但是，操作者所做的任何改变仅仅对该生产线有效，它要求专家(例如被授权的处理工程师)同意，来将该改变传播到该工厂的其余的地方。同样地，在批准的对象可以下载到其它生产线上之前，它们必须首先由具有适当允许级别的人来选择。一般地，发布的版本是将要下载的版本。就象较高级别对象的发布会自动发布较低级别的对象那样，较高级别的批准将会自动地使较低级别的对象批准。

依据本发明的另一个方面，发布一个版本的处理在不减少生产线生产率的情况下，提供了最大的灵活性。也就是说，任何使用者可以发布与他或她的责任相称的对象。在一个实施例中，在每个机器上，该机器上运行未发布的版本，在每个产品(如电路板是在未发布版本下运行的)之后出现一个发布对话，请求发布数据。例如，问题可能是“你想发布配方电话 386 吗？”，弹出的对话框分别具有标记为“是”，“否”和“细节”的三个按钮，在“细节”的按钮之下，可以显示将要发布的准确信息。最好地，在“细节”按钮下还有附加特性，以便能够选择和发布部分配方，而不是要求较高级别允许的整个配方。这样，与他们的责任级别相称的任何人员都可以发布带有或不带有使用者责任的对象。

但是发布带有使用者责任的对象要求记录有关哪个对象被谁记录的信息。这在多于一个操作者工作的生产线上尤为重要，以便知道谁发布了一个特定的对象。在这种情况下，在发布对话上按下“是”按钮之后，出现另一个弹出的对话，在这里要输入使用者的姓名和口令。只有当口令与使用者的姓名一致时，并且具体的使用者具有所要求的较低允许级别时，该对象如配方才被发布，并且使用者的信息和它一起保存起来。

当操作者装备有移动操作者套件，如带有条码扫描器的袖珍电脑(Pocket PC[®])，或其它适合的手持计算机，同固定计算机站对比，操作者可以实现不同的功能，包括闭环元件批准，性能最大化和其它功能。在使用者可以使用该制造控制系统之前，必须登录到其上。由于每个操作者拥有一个操作者套件，所以可以使用登录信息来检测允许级别，以便发布对象和发布包含使用者信息的信息。

在本发明的另一方面，计算机控制的制造系统为用户提供帮助来确定他/她是否想下载最新的发布版本，一个具体版本或一个对象最新未发布的版本，如配方。这将会通过给使用者选择下载版本的机会来完成，

但是要与他/她的允许级别相称。如果具有最小被授予的权限，使用者将仅能选择该生产线最新发布或批准的版本来下载。这对操作者来说是正常情况。而且，当选择了一个配方时，缺省的版本最好是最新发布或批准的版本。较新的、未发布的或部分发布的版本对使用者来讲也是可用的。部分发布的版本是对另一条生产线批准的版本，或为另一个工厂发布的版本。然后，使用者选择确定选择哪个版本。如果需要的话还可以选择前一个版本。

具有上述各种情况的本发明实施例所执行的例子如图 3 所示。在所示的实施例中，一条生产线的工厂是由一个或多个工程师编程的，没有限制。对这些产品(如电路板和装配的机器指令)编程的工程师常常也是运行该生产线的相同的人，或是为运行准备设置的人。因此，工程师/操作者熟知任何变化，并且相应于他/她认为合适的那些改变，可以做出程序批准或发布版本。还使用图 2a-2c，其中现在的配方 1 是指电路板配方，而配方 2 是指生产线或设置配方，可以看到带有电路板名称的对象和带有移动电话生产线的对象都采用版本 1.00，因为它们是新创建的对象，甚至带有在发布过程中对电路板做了改变的对象。然后，由于在图 3 事件 11 中对小外形晶体管(SOT)23 的形式因数所做的修改，对配方 1 和 2 的版本号均增加 0.01，而元件的版本号增加 1。它们在生产线请求发布之后，在事件 14 中发布，而工程师/操作者在确认该改变的有效性后发布。将会认识到，对于单条生产线而讲，批准和发布处理结合，因为没有该包可以发布到的其它生产线。作为另一个例子，图 3 表示当对不同产品的不同配方使用相同电路板被下载作为配方 3 时，于是，如果元件下游的形式因数被修改了，那么，用手持计算机代替移动电话，配方 3 的版本号按以前相同的方式增加 0.01。由于以前给出的原因，元件本身的版本号增加 1，成为 V3.0。在检查了电路板的完整性后，配方 3 和包形式被发布所叙述的版本号。由于在电路板上存在改变，所以系统以单条生产线企业级别来自动更新配方。

在多条生产线的工厂中版本跟踪情况如图 4 所示。这里，在有几条生产线的工厂中，或者在全球分布得工厂中，或者在具有几个工厂的企业中，对电路板和机器编程的工程师正常情况下是与运行生产线和设置不同的人。这样，在图 4 中工程师和操作者是两个不同的实体，只有处理工程师可以发布程序。但是，在发布前，操作者可以下载，修改和改

变对象并使它们生效,但是,这是仅对于他/她自己的生产线而言。当发生改变(见图 4 中的事件 37)时,工程师会被自动通知,并且只有他或她拥有向工厂中其余生产线,或者企业中全球分布的生产线发布具有新版本的修改包的权限。由于修改包形式因数,中级的递增变化为 0.01, 5 遵从图 3 所示情况中的相同过程。

图 5 所示系统涉及制造系统(100),包括几条生产线,通过网络与服务器(120)通信,服务器(120)又连接到包含程序的数据库(110)上。制造系统和服务器(120)还通过网络与至少一个客户设备通信,为了访问该程序,通过被授权这样做的人,至少一个客户设备允许与服务器(120)通信, 10 并且,由至少一个操作者通过与制造系统相关的接口,该程序还可以经服务器(120)被访问。通过网络检测发生的对程序的修订,并确定对程序的修订是否由至少一个被授权人员中特定一个人来做的。如果该修订不是由至少一个被授权人员中特定一个人来做的,通过网络从服务器向客户设备发送一条消息,以通知该特定的人员做了修订。

15 参照特定的实施例,表示和说明了本发明的各个方面,并且做了很多详细说明以便帮助理解它们。但是这些具体的细节对于实施本发明的这些方面不是必需使用的。而且,可以做出很多形式和细节上的变化,而不脱离本发明的精神和范围。例如一些情况下,不脱离本发明的这个方面的范围,方法步骤的特定排序可以是变化的,或者公开的数据结构 20 可以重新安排或补充,而又保留其内容。

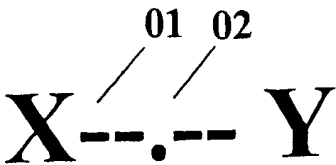


图 1a

V2.03R

图 1b

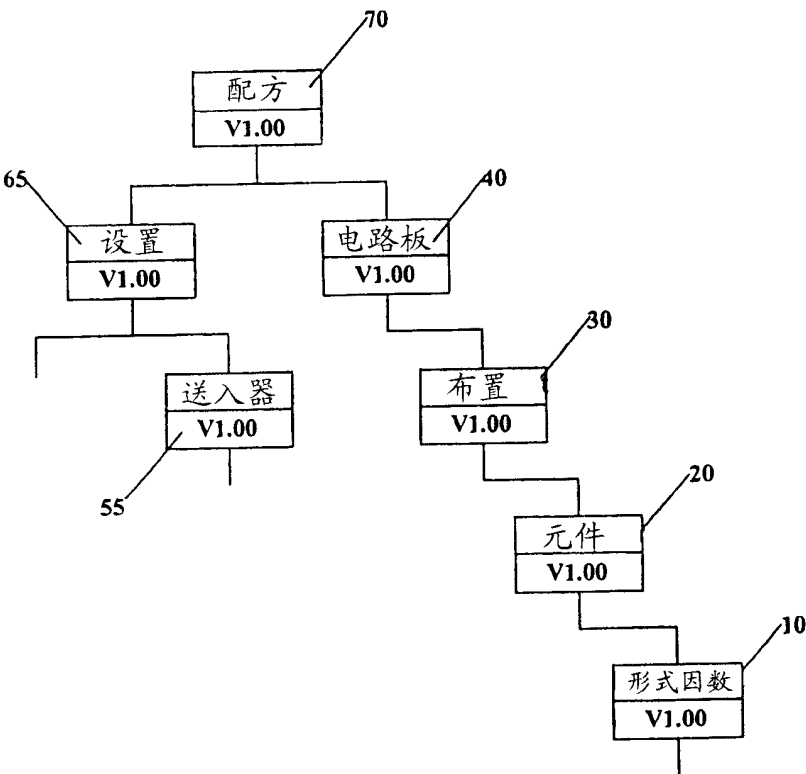


图 2a

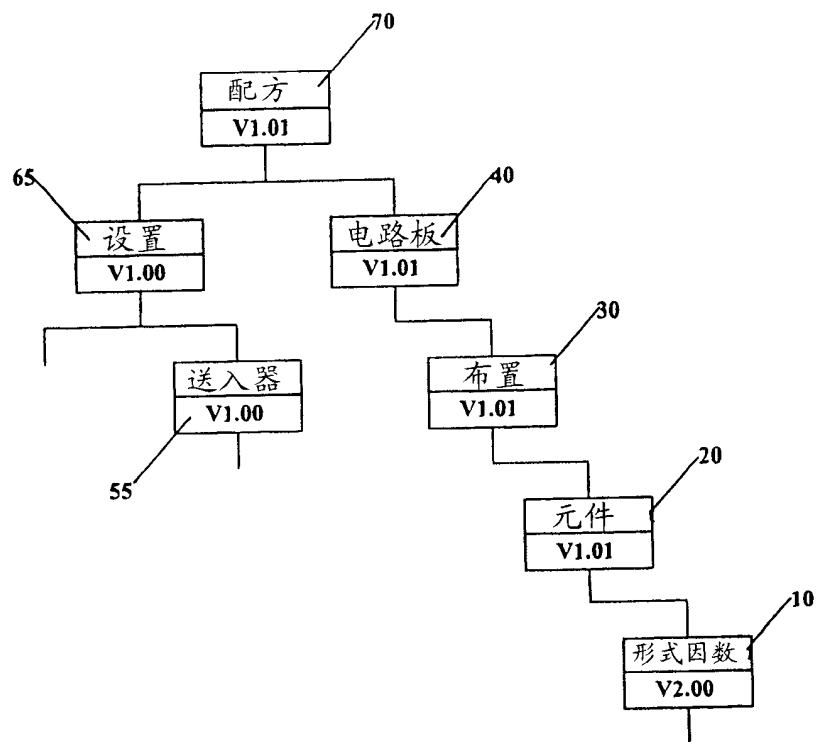


图 2b

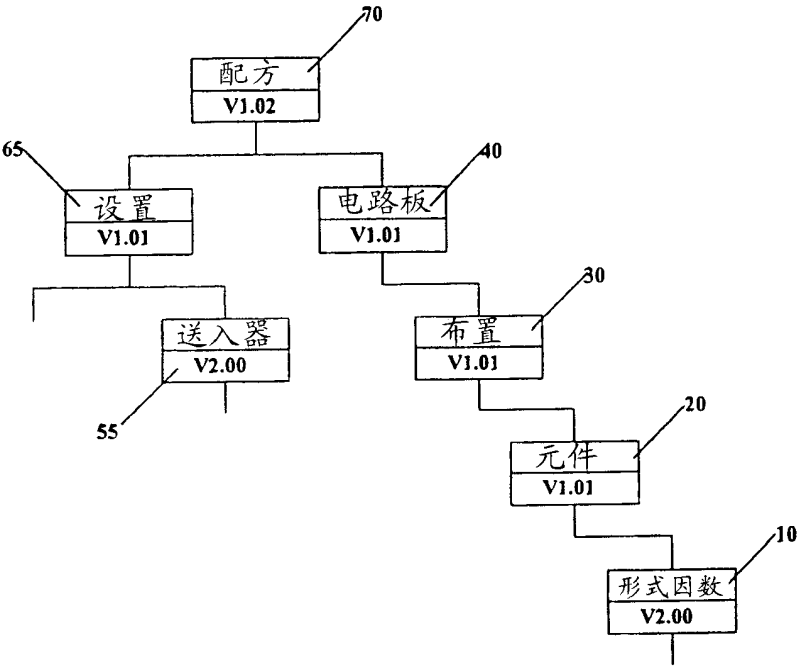


图 2c

在单条生产线上的版本跟踪

事件	使用者	行为	对象	姓名	版本
1	工程师/操作者	规划电路板	配方1	电路板	V1.00
2	工程师/操作者	修改同一个电路板	"	"	"
3	工程师/操作者	为智能电话线路下载配方2	配方1 配方2	电路板 SPLine	" "
4	生产线...	运行电路板和请求批准 validation	"	"	"
5	工程师/操作者	检查电路板和发现错误	配方1	电路板	
6	工程师/操作者	修正程序	"	"	"
7	工程师/操作者	下载配方	配方1 配方2	电路板 SPLine	" "
8	生产线...	运行电路板和请求批准	"	"	"
9	工程师/操作者	检查电路板和发布它	配方1 配方2	电路板 SPLine	V1.00R V1.00R
10	随后, 工程师/操作者想对子对象 SOT 23进行修改		包形式	SOT 23	V1.00R
11	工程师/操作者	修改SOT 23	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00 V1.01 V1.01
12	工程师/操作者	下载配方和运行电路板	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00 V1.01 V1.01
13	生产线...	运行电路板和请求发布配方 1/2 (生产线/电路板)	"	"	"
14	工程师/操作者	检查包形式并使它批准, 等于向单条生产线发布它	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00R V1.01R V1.01R
15	随后, 工程师/操作者下载程序		包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00R V1.01R V1.01R
16	生产线...	运行没有修改的电路板	"	"	"
17	随后, 工程师/操作者下载不同的配方 (3), 配方(3)具有与早期V1.01中修改 相同的SOT 23		配方3	掌上电脑	V1.00R
18	工程师/操作者	发现问题并再次不同地修改 SOT 23	配方3 包形式2	掌上电脑 SOT 23	V1.01 V3.0
19	工程师/操作者	下载程序到生产线上	"	"	"
20	生产线...	运行电路板并请求发布	"	"	"
21	工程师/操作者	检查电路板并发布它	配方3 包形式2	掌上电脑 SOT 23	V1.01R V3.0R
22	系统 M	在企业级别上自动更新配方	包形式 配方1 配方3	SOT 23 电路板 掌上电脑	V3.00R V1.02R V1.02R
图 3					

在多条生产线的工厂中的版本跟踪

步骤	使用者	行为	对象	姓名	版本
23	工程师	规划电路板	配方1	电路板	V1.00
24	工程师	修改同一个电路板	"	"	"
25	工程师	为智能电话线路下载配方2	配方1 配方2	电路板 SPLine	" "
26	生产线..	运行电路板并请求操作者批准 配方/电路板	"	"	"
27	操作者	检查电路板并发现错误	配方1	电路板	
28	操作者	修正程序	"	"	"
29	操作者	下载配方并运行电路板	配方1 配方2	电路板 SPLine	" "
30	生产线..	运行电路板并请求操作者批准 配方/电路板	"	"	"
31	操作者	检查电路板并使它批准	配方1 配方2	电路板 SPLine	V1.00V V1.00V

32	随后, 工程师和操作者在SOT 23上发现 一些问题		包形式	SOT 23	V1.00R
33	操作者	修改SOT 23	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00 V1.01 V1.01
34	操作者	为智能电话线路下载配方	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00 V1.01 V1.01
35	生产线..	运行电路板并请求发布	"	"	"
36	操作者	检查电路板并使它批准	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00V V1.01V V1.01V

37	工程师被通知修改				
38	工程师	同意修改并发布SOT 23	包形式	SOT 23	V2.00R
39	操作者	下载到不同的生产线, 被发布的 对象还在不同的生产线上发布, 批准的对象必须再被批准	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00R V1.01 V1.01
40	生产线2	运行电路板并请求操作者使 配方/电路板批准			
41	操作者	检查程序并使它批准	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00R V1.01V V1.01V
42	工程师	检查程序并发布它	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00R V1.01R V1.01R
43	操作者	下载程序到第三条生产线上	包形式 配方1 配方2	SOT 23 电路板 SPLine	V2.00R V1.01R V1.01R
44	生产线3	运行电路板, 由于它是发布 的程序			

图 4

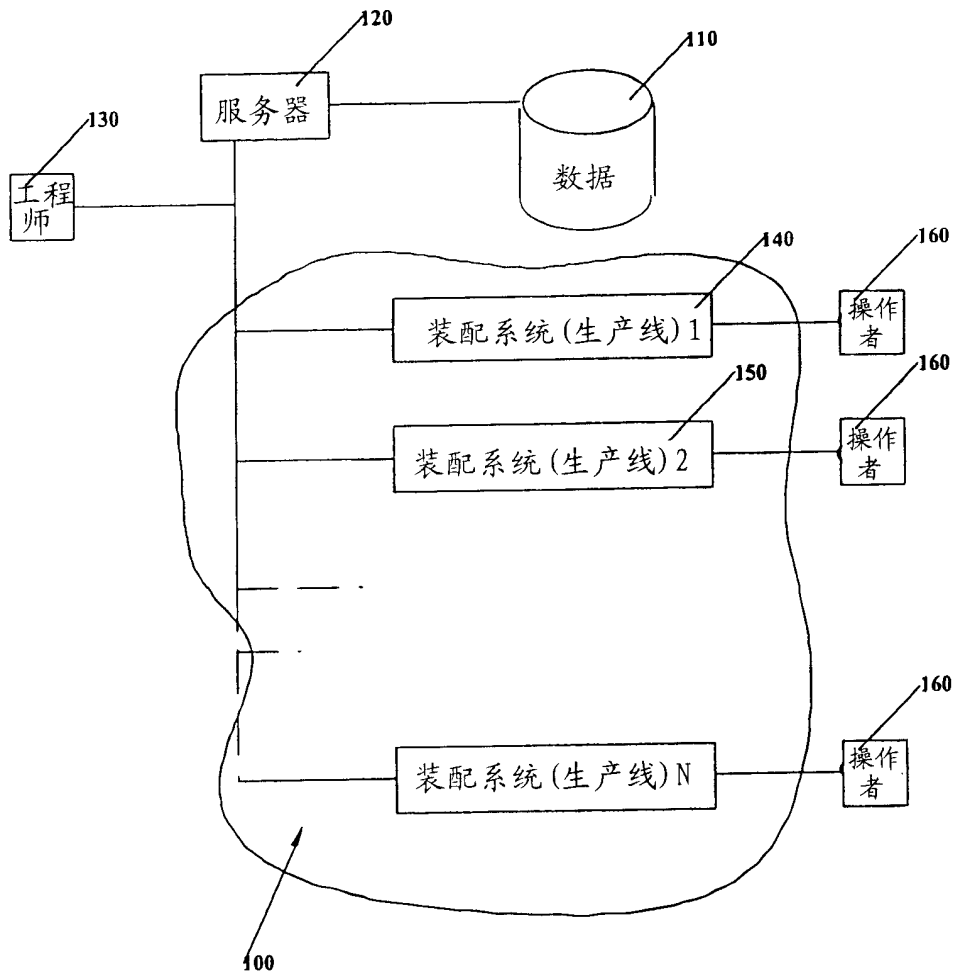


图 5