(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 112987692 B
(45)授权公告日 2024.08.06

(21)申请号 202110231594.7
(22)申请日 2016.10.07
(65)同一申请的已公布的文献号
申请公布号 CN 112987692 A
(43)申请公布日 2021.06.18
(30)优先权数据
62/239,657 2015.10.09 US
(62)分案原申请数据
201680059057.6 2016.10.07
(73)专利权人 费希尔-罗斯蒙特系统公司
地址 美国德克萨斯州
(72)发明人 G·K·劳 G·R·谢里夫
(74)专利代理机构 永新专利商标代理有限公司 72002
专利代理师 戚英豪 丁燕
(51)Int.Cl.
G05B 23/02(2006.01)
G05B 19/05(2006.01)
G05B 19/042(2006.01)
G06F 3/0482(2013.01)
G06F 3/04847(2022.01)
(56)对比文件
CN 103097973 A,2013.05.08
CN 104536436 A,2015.04.22
审查员 严雪莹

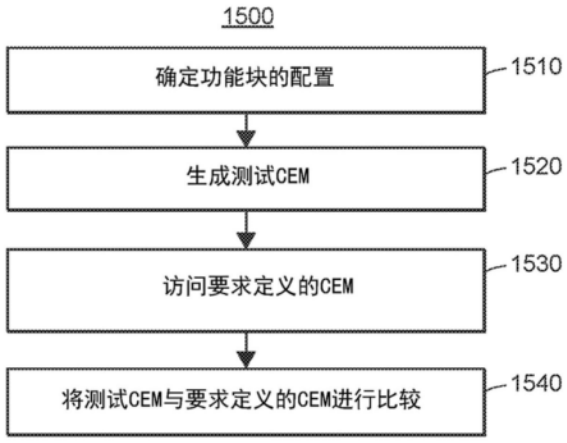
权利要求书2页 说明书23页 附图17页

(54)发明名称

用于验证因果矩阵的安全逻辑的系统和方法

(57)摘要

用于确定过程工厂的过程控制系统的配置的系统和方法,该过程控制系统被实现为功能块集合,该系统和方法包括:针对功能块集合中的每一个功能块,基于以下项来确定功能块的配置:(i)功能块的输出集合,(ii)功能块的逻辑,以及(iii)功能块的输入集合。该系统和方法还包括:基于功能块集合的配置集合来生成具有测试原因集合和测试结果集合的测试因果矩阵(CEM)。该系统和方法还包括:访问具有原因集合和结果集合的要求定义的CEM。该系统和方法还可以包括:将测试CEM与要求定义的CEM进行比较以确定是否存在差异集合。



CN 112987692 B

1.一种计算机实现的方法,其验证过程控制系统的安全逻辑,所述方法包括:

访问所述过程控制系统的安全控制逻辑,所述安全控制逻辑基于原因集合和结果集合进行控制,其中,所述原因集合中的每一个原因表示过程工厂内的状况,并且所述结果集合中的每一个结果表示将在所述过程工厂内执行的结果,并且其中,所述原因集合和所述结果集合中的至少一些是作为因果对而相关的,由此对应的结果响应于对应的原因中的一个或多个原因的发生而激活,并且其中,所述安全控制逻辑实现为多个互连的功能块,所述多个互连的功能块中的每一个具有输入集合、输出集合、以及将所述输入集合与所述输出集合相关联的功能块逻辑,并且其中,所述多个功能块中的一个功能块的至少一个输出作为输入连接到所述多个功能块中的至少一个其它功能块,并且其中,所述原因中的每一个是对所述多个互连的功能块中的至少一个功能块的输入并且所述结果中的每一个是所述多个互连的功能块中的至少一个功能块的输出;

基于所述安全控制逻辑的所述多个功能块来生成具有测试原因集合和测试结果集合的测试因果矩阵,其中,所述测试原因集合和所述测试结果集合中的至少一些是作为测试因果对而相关的;

访问要求定义的因果矩阵;以及

将所述测试因果矩阵与所述要求定义的因果矩阵进行比较以确定是否存在差异集合。

2.根据权利要求1所述的计算机实现的方法,其中,所述一个或多个功能块包括一个或多个监控块和一个或多个结果块,其中,所述一个或多个监控块中的每一个包括第一输入集合和第一输出集合以及监控块逻辑,其中,所述一个或多个结果块中的每一个包括第二输入集合和第二输出集合以及结果块逻辑,并且其中,所述安全控制逻辑分布在至少一个监控块的监控块逻辑和所述结果块中的至少一个结果块的结果块逻辑内。

3.根据权利要求2所述的计算机实现的方法,其中,所述监控块或结果块中的一个或多个是互连的。

4.根据权利要求3所述的计算机实现的方法,其中,所述一个或多个监控块或结果块中的一个或多个是经由分层、嵌套、循环、或链连接来互连的。

5.根据权利要求4所述的计算机实现的方法,还包括:

遍历所述一个或多个互连的监控块或结果块以生成所述测试因果矩阵。

6.根据权利要求5所述的计算机实现的方法,其中,遍历所述一个或多个互连的监控块或结果块以生成所述测试因果矩阵包括:

识别对第一监控块的第一输入;

识别所述第一监控块的第一输出,所述第一输出逻辑地耦接到所述第一监控块的所述第一输入;

识别对第一结果块的第二输入,其中,对所述第一结果块的所述第二输入被逻辑地耦接到所述第一监控块的所述第一输出;

识别所述第一结果块的第二输出,所述第二输出逻辑地耦接到所述第一结果块的所述第二输出;

确定将所述第一监控块的所述第一输入与所述第一监控块的所述第二输出相关联的逻辑表达式以生成组合的逻辑,以及

使用所述组合的逻辑来确定所述测试因果矩阵的一个或多个因果对。

7.根据权利要求1所述的计算机实现的方法,其中,所述要求定义的因果矩阵是定义所述过程工厂所需的所述安全控制逻辑的准确表示。

8.根据权利要求1所述的计算机实现的方法,还包括:

对于所述差异集合中的每一个差异:

识别所述多个功能块中与所述测试因果矩阵的因果对相对应的一个或多个功能块,所述测试因果矩阵的所述因果对不同于所述要求定义的因果矩阵的对应因果对;以及

显示所识别的一个或多个功能块。

用于验证因果矩阵的安全逻辑的系统和方法

[0001] 本申请是名称为“用于验证因果矩阵的安全逻辑的系统和方法”、国际申请日为2016年10月7日、国际申请号为PCT/US2016/056024、国家申请号为201680059057.6的发明专利申请的分案申请。

[0002] 相关申请

[0003] 本专利申请是常规专利申请,其要求于2015年10月9日提交的、标题为“A System and Method for Configuring Separated Monitor and Effect Blocks of a Process Control System”的美国临时专利申请序列号62/239,657的申请日的优先权和权益,该临时专利申请在此通过引用被明确地并入本文。

技术领域

[0004] 本公开内容总体上涉及管理过程工厂内的过程控制系统,并且更具体地,涉及配置与过程控制系统相关联的因果矩阵(cause and effect matrix,CEM)以及创建与其相关的监控块和结果块。

背景技术

[0005] 过程控制系统,如在化学、石油或其它过程中使用的过程控制系统,通常包括一个或多个过程控制器,所述过程控制器通信地耦合到至少一个主机或操作员工作站并经由模拟、数字或组合的模拟/数字总线或线路耦合到一个或多个现场设备。现场设备(可以是例如阀、阀定位器、开关和变送器(例如,温度、压力和流率传感器))在过程工厂内执行功能,诸如打开或关闭阀以及测量过程参数。过程控制器接收指示由现场设备进行的过程测量的信号和/或与现场设备有关的其它信息,使用该信息来实现控制例程,随后生成控制信号,控制信号通过总线或线路被发送到现场设备以控制过程的操作。来自现场设备和控制器的信息通常可用于由操作员工作站执行的一个或多个应用,以使操作员能够执行关于过程的任何所期望的功能,例如配置过程、查看过程的当前状态、修改过程的操作等。

[0006] 另外,在许多过程中,提供单独的安全系统以检测过程工厂内的显著安全相关问题,并且当出现可能引起或导致工厂内的严重危险的问题(诸如有毒化学品溢出、爆炸等)时,自动关闭阀门、从设备移除电力、切换工厂内的流程等。这些安全系统通常具有除了标准过程控制器之外的一个或多个单独的控制器,称为逻辑解算器,逻辑解算器经由安装在过程工厂内的单独的总线或通信线路连接到安全现场设备。逻辑解算器使用安全现场设备来检测与显著事件相关联的过程状况,诸如某些安全开关或关断阀的位置、过程中的上溢或下溢、重要发电或控制设备的操作、故障操作检测设备的操作等,从而检测过程工厂内的“事件”。当检测到可能是单一状况或同时发生两个或更多个状况的事件(通常称为“原因(cause)”)时,安全控制器采取一些动作(通常称为“结果(effect)”)来限制事件的有害性质,诸如关闭阀、使设备断开(turn off)、从工厂的各部分移除电力等。通常,这些动作或结果包括将安全设备切换到触动(trip)或“安全”操作模式,该操作模式被设计为防止过程工厂内的严重或危险状况。

[0007] 过程工厂的操作员(诸如管理者和工程师)通常维护一数据结构,该数据结构存储相关原因和结果。例如,矩阵可以具有多个行和列,其中每行对应于原因,每列对应于结果并且矩阵的每个单元对应于特定的因果关系。这些单元可以由各种触发(trigger)填充,这些触发指示每个原因与结果之间的关系。一般根据为控制系统或工厂定义安全设计的要求文档来配置所谓的因果矩阵(CEM)。控制工程师可以利用CEM来对控制系统进行工程设计(engineer),从而相应地实现安全设计。然而,这样的CEM受限于矩阵的定义的大小,并且经常不足够大以处理所有期望的原因/结果数据关系。此外,这样的CEM不能处理更复杂/精细的原因/结果,诸如链连接、链接、分级(level)、循环等。此外,大的CEM实现到控制逻辑中是繁琐的,并因此在实现期间容易出错。在安全系统中,维持准确的CEM是必要的,因为CEM中的差错可能是严重的,这是由于正常操作的安全系统的故障可能导致工厂人员的严重受伤甚至死亡并且导致工厂内设备和材料的可能数百万美元的损坏。

发明内容

[0008] 过程工厂的过程控制系统可以具有安全系统,该安全系统可以被实现或设计为影响(effect)在因果矩阵(CEM)中定义的控制逻辑,其中CEM是以可视表示显示的、用于过程工厂的安全动作的汇总(summary)。一般而言,CEM定义针对过程工厂内的各种安全协议或过程的基本因果关系。通常,CEM可以包括输入集合和输出集合,其中输入集合中的每一个表示过程工厂内的状况,并且输出集合中的每一个表示将在过程工厂内执行的结果或动作。此外,输入集合和输出集合中的至少一些是作为因果对而相关的,由此对应的结果响应于对应的状况或原因的发生而激活。

[0009] 过程控制系统的管理员可以将CEM实现为各种功能块的集合。然而,取决于过程工厂的规模和/或复杂度,给定的CEM可能包含众多原因、结果和因果对,并且可能因此需要对应众多数量的功能块来实现。因此,这种实现会变得耗时、复杂且繁琐,从而导致潜在的实现差错。根据所描述的系统和方法,提供了用于将过程控制系统内的CEM实现为被描述成监控功能块和结果功能块的、分开的但互连的功能块集合以实现CEM逻辑的技术。

[0010] 在一个实施例中,所述系统和方法可以识别CEM内的模式和分组(grouping),并且可以根据所识别的模式和分组来实现监控块集合和结果块集合,因此降低了CEM的实现的复杂度。在一个实现方式中,CEM内的数据的分组(例如,CEM的列)可以被定义为由该CEM的该部分定义的逻辑的数值表示(numerical representation),以提供一种简单并且较不复杂的方式来理解和验证CEM的逻辑是在用于实现CEM逻辑的功能块(例如,监控和结果块)内实现的。此外,可以使用工具来分析和重新排序或重新排列CEM(例如,CEM的行和/或列)以提供更好、更有逻辑性、更容易实现的等等的CEM逻辑的分组以实现为一个或多个因果块集合。

[0011] 本公开内容提供了另外的用于管理CEM的技术。具体地,本文描述的系统和方法可以用于配置CEM以包括交互功能。例如,配置的CEM可以包括用以访问详述构成CEM的因果关系的安全协议的一个或多个文档的连接或选择、描绘CEM的一个或多个结果的当前和/或过去状态以使用户能够更容易地理解关于在工厂中实现的特定结果的先前状况或操作的图表、以及包括与CEM的原因和结果相关的设备的过程工厂的示图。

[0012] 另外,由于包含在CEM中的通常大量的信息,因此工程师可能难以识别过程控制系

统中包含的任何差异或差错。本文提供的系统和方法进一步使得能够实现逆向工程技术和系统以自动创建测试CEM,该测试CEM定义由过程工厂中的设备和控制逻辑实际实现的CEM逻辑(或者,在一些实现方式中,过程控制系统内的监控块和结果块)以及特定过程工厂所需的安全协议。因此,本文描述的系统和方法可以将测试CEM与现有CEM进行比较,以识别工厂操作的实际配置与可以在设计文档中详述的工厂操作的配置之间的任何差异或差错。

附图说明

[0013] 下面描述的附图描绘了其中公开的系统和方法的各个方面。应该理解的是,每个附图描绘了所公开的系统和方法的特定方面的实施例,并且每个附图旨在符合其可能的实施例。此外,尽可能地,以下描述引用包括在以下附图中的附图标记,其中在多个附图中描绘的特征利用一致的附图标记来表示。

[0014] 在附图中示出了当前讨论的布置,然而,应该理解的是,当前实施例不限于所示出的精确布置和手段,其中:

[0015] 图1是示例性过程工厂的框图;

[0016] 图2是图1中示意性示出的示例性工作站的框图;

[0017] 图3是示例性因果矩阵的图示;

[0018] 图4是示例性的监控块和结果块集合的图示;

[0019] 图5是可以用于实现监控块和结果块的功能块集合的第一示例的图示;

[0020] 图6是可以用于实现监控块和结果块的功能块集合的第二示例的图示;

[0021] 图7是配置与过程工厂相关联的监控块和结果块的示例性方法的流程图。

[0022] 图8是因果矩阵的第二示例的图示;

[0023] 图9是图8的因果矩阵的第二示例的图示,该因果矩阵已被重新组织并且被配置成单独的逻辑块;

[0024] 图10是重新组织因果矩阵的示例性方法的流程图;

[0025] 图11是具有示例性数值表示的因果矩阵的第二示例的图示;

[0026] 图12是计算因果矩阵逻辑的数值表示的示例性方法的流程图;

[0027] 图13是对应于因果矩阵的安全逻辑的各种互连的用户界面的示例性图示;

[0028] 图14是用于在图13的互连的用户界面之间进行导航的示例性方法的流程图;

[0029] 图15是用于对测试因果矩阵进行逆向工程的示例性方法的流程图;

[0030] 图16A-图16D是显示监控的安全事件的示例性用户界面的图示;

[0031] 图17是用于显示监控的安全事件的示例性方法的流程图;

[0032] 图18是包括许可(permissive)和时间延迟的触发的示例性因果矩阵的图示;

[0033] 附图仅出于说明目的描绘了优选实施例。本领域技术人员根据以下讨论将容易认识到,可以采用本文所示出的系统和方法的替代实施例而不脱离本文所描述的本发明的原理。

具体实施方式

[0034] 图1是包括一个或多个节点12、16、18和20的示例性过程工厂10的框图。在图1的示例性过程工厂10中,节点12和16中的每一个包括过程控制器12a、16a,过程控制器12a、16a

经由输入/输出(I/O)设备24连接到一个或多个现场设备22和23,输入/输出(I/O)设备24可以是例如Foundation现场总线(Fieldbus)接口、HART接口等。控制器12a和16a还经由网络30耦合到节点18和20中的一个或多个主机或操作员工作站18a和20a,网络30可以包括例如总线、诸如以太网LAN的有线局域网(LAN)、无线LAN、广域网(WAN)、内联网等中的一个或多个。虽然控制器节点12、16以及与其相关联的I/O设备24和现场设备22、23通常位于有时恶劣的工厂环境内并分布于整个有时恶劣的工厂环境中,但是操作员工作站节点18和20通常位于控制室或可由控制人员容易访问的其它不太恶劣的环境中。

[0035] 一般而言,节点18和20的工作站18a和20a可以用于储存和执行用于配置和监控过程工厂10的应用,和/或管理过程工厂10中的设备22、23、24和控制器12a、16a。例如,工作站18a和/或20a可以包含诸如系统导航器应用15、因果分析器工具17、过程控制配置应用19和安全配置应用21之类的工具,安全配置应用21可以被实现为管理过程工厂10的安全要求。系统导航器应用15可以被实现为提供互连的用户界面群组,用户界面提供关于过程工厂中的安全要求和设备的信息。因果分析器工具可以被实现为管理因果矩阵(CEM)和/或通过从已知的安全要求和/或功能块进行逆向工程来创建因果矩阵。此外,过程控制配置应用19和安全配置应用21向用户提供通过工作站18a和/或20a管理过程工厂的设备的能力。配置数据库32可以连接到网络30并且可以作为数据历史库和/或配置数据库来操作,当过程工厂10的当前配置被下载到节点12、16、18、20和/或储存在节点12、16、18、20内时,该数据历史库和/或配置数据库储存过程工厂10的当前配置。配置数据库还可以包含用于重新排列CEM的规则31和/或数值表示33。

[0036] 控制器12a和16a中的每一个(举例来说,其可以是由艾默生过程管理公司出售的DeltaV™控制器)可以储存并执行控制器应用,该控制器应用使用若干个不同的、独立执行的控制模块或块来实现控制策略。控制模块可以各自由通常所称的功能块组成,其中每个功能块是总体控制例程的部分或子例程,并且结合其它功能块(经由称为链路的通信)来操作以实现过程工厂10内的过程控制回路。如所公知的,功能块通常执行输入功能(诸如与变送器、传感器或其它过程参数测量设备相关联的输入功能)、控制功能(诸如与执行PID、模糊逻辑等控制的控制例程相关联的控制功能)或控制某些设备(例如阀)的操作的输出功能中的一个,以执行过程工厂10内的一些物理功能。当然,混合和其它类型的功能块存在并可以被利用。尽管现场总线协议和DeltaV™系统协议可以使用以面向对象编程协议设计和实现的控制模块和功能块,但是控制模块可以使用任何期望的包括例如顺序功能块、梯形逻辑等等的控制编程方案来设计,并且不限于使用功能块或任何其它特定的编程技术来设计。典型地,如在过程控制节点12和16内所储存的控制模块的配置可以储存在配置数据库32中,配置数据库32可由工作站18a和20a执行的应用访问。功能块可以储存在例如控制器12a、16a中并且由其执行,这通常是当这些功能块是用于或关联于标准4-20mA设备和诸如HART设备之类的某些类型的智能现场设备时的情况,或者可以储存在现场设备本身中并由其实现,这可以是现场总线设备的情况。

[0037] 在图1所示出的系统中,耦合到控制器12a和16a的现场设备22和23可以是标准4-20mA设备,或者可以是智能现场设备,诸如HART、Profibus或Foundation现场总线现场设备,其包括处理器和存储器。这些设备中的一些,诸如Foundation现场总线现场设备(在图1中用附图标记23来标示),可以储存并执行与在控制器12a和16a中实现的控制策略相关联

的模块或子模块(诸如功能块)。当然,现场设备22、23可以是任何类型的设备,诸如传感器、阀、变送器、定位器等,并且I/O设备24可以是遵循任何期望的通信或者控制器协议(诸如HART、Foundation现场总线、Profibus等)的任何类型的I/O设备。

[0038] 控制器12a和16a各自包括实现或监督储存在存储器中的一个或多个过程控制例程的处理器,该存储器可以包括储存在其中或以其它方式与其相关联的控制回路。控制器12a和16a与现场设备22、23、工作站18a、20a和数据库32通信以便以任何期望的方式来控制过程。控制器12a和16a各自可以被配置为以任何期望的方式实现控制策略或控制例程。

[0039] 过程工厂10还可以包括与过程控制节点12和16集成的安全系统14(由虚线指示)。安全系统14通常可以作为安全仪表系统(SIS)来操作以监控和超越(override)由过程控制节点12和16提供的控制,从而最大化过程工厂10的可能的安全操作。

[0040] 节点12和16中的每一个可以包括一个或多个安全系统逻辑解算器50。每个逻辑解算器50是具有处理器和存储器的I/O设备,并且被配置为执行储存在存储器中的安全逻辑模块。每个逻辑解算器50被通信地耦合以向安全系统现场设备60和62提供控制信号和/或从安全系统现场设备60和62接收信号。另外,节点12和16中的每一个可以包括至少一个消息传播设备(MPD)70,该MPD 70经由环形或总线连接74(仅在图1中示出其部分)通信地耦合到其它MPD 70。安全系统逻辑解算器50、安全系统现场设备60和62、MPD 70以及总线74通常构成图1的安全系统14。

[0041] 图1的逻辑解算器50可以是任何期望类型的安全系统控制设备,其包括处理器和存储器,该存储器储存适于在处理器上执行的安全逻辑模块,以使用现场设备60和62来提供与安全系统14相关联的控制功能。当然,安全现场设备60和62可以是遵循或使用任何已知或期望的通信协议的任何期望类型的现场设备,诸如上面提到的那些。特别地,现场设备60和62可以是常规地由单独的、专用安全相关控制系统控制的类型的安全相关的现场设备。在图1所示出的过程工厂10中,安全现场设备60被描绘为使用专用的或点对点通信协议,诸如HART或4-20mA协议,而安全现场设备62被示出为使用总线通信协议,诸如现场总线协议。安全现场设备60可以执行任何期望的功能,诸如关闭阀、关闭开关等。然而,安全系统现场设备60和62可以是其它类型的设备,并且可以使用用以与逻辑解算器50通信的其它类型的通信协议,包括任何期望的有线或无线通信协议。

[0042] 可以在节点12和16中的每一个中使用公共背板(未示出)以将控制器12a和16a通信地耦合到过程控制I/O卡24、安全逻辑解算器50以及MPD 70。控制器12a和16a还通信地耦合到网络30。控制器12a和16a、I/O设备24、逻辑解算器50、MPD 70可以经由网络30来与节点18和20通信。

[0043] 如本领域普通技术人员将理解的,节点12、16中的背板(未示出)使得逻辑解算器50能够彼此本地通信以协调由这些设备实现的安全功能,向彼此传送数据,和/或执行其它集成的功能。类似地,节点16中的背板(未示出)使得逻辑解算器50能够彼此本地通信以协调由这些设备实现的安全功能,向彼此传送数据,和/或执行其它集成的功能。另一方面,MPD 70进行操作以使得设置在工厂10的大不相同位置中的安全系统14的部分仍然彼此通信以提供过程工厂10的不同节点处的协调的安全操作。特别地,MPD 70结合总线74使得与过程工厂10的不同节点12和16相关联的逻辑解算器50能够通信地级联在一起,以允许根据分配的优先级来进行过程工厂10内的安全相关功能的级联。MPD 70和总线74向安全系统提

供通信链路,该通信链路作为网络30的替代。

[0044] 替代地,过程工厂10内的不同位置处的两个或更多个安全相关功能可以互锁或互连,而不必通过使用MPD 70和通信线路74运行至工厂10的单独区域或节点内的各个安全现场设备的专用线路。换而言之,MPD 70和总线74的使用使得安全工程师能够设计和配置安全系统14,该安全系统14本质上分布在整个过程工厂10中,但是其具有通信地互连的不同部件,以使得相异的安全相关硬件能够根据需要彼此通信。该特征还提供了安全系统14的可扩展性,因为其使得当需要另外的安全逻辑解算器时或者当新的过程控制节点被添加到过程工厂10时能够将另外的安全逻辑解算器添加到安全系统14中。将理解的是,逻辑解算器50通常包括控制逻辑,该控制逻辑实现由一个或多个因果矩阵(CEM)定义的安全逻辑。

[0045] 图2是示意性地示出示例性工作站18a的结构的框图(工作站20a可以包括相同或类似的设备)。工作站18a可以包括至少一个处理器100、易失性存储器104和非易失性存储器108。易失性存储器104可以包括例如随机存取存储器(RAM)。在一些实施例中,RAM可以由一个或多个电池作为后备电源,以便在电力故障的情况下数据不会丢失。非易失性存储器108可以包括例如硬盘、只读存储器(ROM)、压缩盘ROM(CD-ROM)、可编程ROM(PROM)、可擦除可编程ROM(EPROM)、电可擦除可编程ROM(EEPROM)、数字多功能盘(DVD)、闪存等中的一个或多个。工作站18a还可以包括工作站I/O设备112。处理器100、易失性存储器104、非易失性存储器108和工作站I/O设备112可以经由地址/数据总线116互连。工作站18a还可以包括至少一个显示设备120和至少一个用户输入设备124,其可以是例如键盘、小键盘、鼠标、轨迹球、触摸屏、光笔等中的一个或多个。在一些实施例中,易失性存储器104、非易失性存储器108和工作站I/O设备112中的一个或多个可以经由与地址/数据总线116分开的总线(未示出)耦合到处理器100,或者可以直接耦合到处理器100。

[0046] 显示设备120和用户输入设备124与工作站I/O设备112相耦合。另外,工作站18a经由工作站I/O设备112耦合到网络30。尽管工作站I/O设备112在图2中被示出为一个设备,但是其可以包括几个设备。另外,在一些实施例中,显示设备120和用户输入设备124中的一个或多个可以直接耦合到地址/数据总线116或耦合到处理器100。

[0047] 现在参考图1和图2,与控制节点12、16中的一个或多个相关联的过程控制配置应用19可以储存在工作站18a和20a中的一个或多个上并由其执行。例如,过程控制配置应用19可以储存在非易失性存储器108和/或易失性存储器104上,并由处理器100执行。然而,如果需要,可以在与过程工厂10相关联的其它计算机中储存并执行该应用。一般而言,过程控制配置应用19允许编程人员、控制工程师或其他人员创建和配置将由控制器12a、16a、I/O设备24和/或现场设备22、23实现的控制例程、控制模块、功能块、程序、逻辑等。这些控制例程、控制模块、功能块、程序、逻辑等可以随后经由网络30下载到适当的控制器12a、16a、I/O设备24和/或现场设备22、23。

[0048] 类似地,与安全系统14相关联的安全系统配置应用21可以储存在工作站18a和20a中的一个或多个上并由其执行。例如,安全系统配置应用21可以储存在非易失性存储器108和/或易失性存储器104上,并由处理器100执行。然而,如果需要,可以在与过程工厂10相关联的其它计算机中存储并执行该应用。一般而言,安全系统配置应用允许编程人员、安全工程师或其他人员创建和配置将由逻辑解算器50和/或设备60、62实现的安全相关的控制例程、安全逻辑模块、功能块、程序、逻辑等。这些控制例程、安全模块、功能块、程序、逻辑等可

以随后经由网络30下载到适当的控制器12a、16a、逻辑解算器50和/或设备60、62。

[0049] 安全系统通常以由国际电工委员会(IEC)61131-3标准定义的几种语言中的一种语言来编程,并且在一些情况下,安全逻辑可以由一系列互连的功能块或其它例程组成。无论编程语言如何,起点通常是指定对控制和/或安全动作的要求的叙述性文档。在安全系统中,安全要求被记录(document)在安全要求规范(SRS)中。SRS(下面更详细描述)可以提供可以由纯文本、逻辑图或因果图(也称为因果矩阵)表示的逻辑描述。因果矩阵(CEM)是由安全系统以简单的可视表示提供的安全动作的汇总。因此,CEM定义由安全逻辑实现的基本因果关系,并且是用于安全逻辑的配置的基础。

[0050] 图3示出了可以经由任何类型的显示设备显示的CEM 300的一个示例性表示。特别地,显示设备可以是与安全配置应用21相关联的用户界面的一部分,并且显示可以例如经由工作站18a的显示设备120被呈现给编程人员或管理员。作为可以在传统过程控制系统中使用的CEM的示例性CEM 300包含多个原因和多个结果。CEM的原因通常由安全要求规范来定义,并且涉及整个过程工厂10中由逻辑解算器50、现场设备22、23、60和62等指示、测量或检测的状况或在其处的状况。在CEM 300中定义的不同的原因C1、C2等与CEM 300的每一行相关联。例如,一个原因可以是传感器的读数为:工厂的特定区域的温度在安全或预定义的范围之外。

[0051] 当发生对应于原因的状况时,可以触发结果,其中结果可以是将在工厂中执行的动作。CEM 300的不同结果E1、E2等是针对CEM 300的每一列而定义的并且与其相关联。例如,CEM 300的一个结果(例如,E3)可以涉及将在工厂中执行的安全动作,诸如关闭阀、发出警报声等。当特定原因(例如,C2或C6)触发特定结果(E3)时,则存在对应的因果对或关系。

[0052] 在CEM 300中,因果关系在每个单元中由“X”表示,指示与单元列相关联的结果由与单元行相关联的原因触发。这些关系在本文中可以被称为因果对。在替代的实现方式中,单元可以由各种“触发”填充,这些“触发”更精确地指示相关联的原因和结果可以如何相关。例如,触发可以具有以下形式:“X”,其指示如果接收到原因,则结果将被立即激活,“T”,其意味着如果接收到原因,则该结果将以时间延迟进行激活,“P”,其指示如果接收到原因,则结果将是许可的,等等。此外,空的单元可以指示特定的因果对在矩阵中当前不是相关的,并且因此可能在工厂中不是活动的(即,原因的发生与结果没有触发关系)。

[0053] 示例性CEM 300是7×7矩阵,其可能小于过程工厂的典型因果矩阵,但为了说明起见以简单形式示出。示例性CEM 300包括在对应单元集合中的每一个单元中由“X”表示的10个因果关系。例如,原因2(C2)在分别对应于结果3、4和5(E3、E4和E5)的每个单元中包括“X”。因此,如果原因2(C2)的相关联事件发生,则可以由工厂中的安全逻辑模块在过程工厂内触发结果3、4和5(E3、E4和E5)的相应动作。然而,在一些实施例中,结果3、4和5中的每一个在被触发之前也可能需要发生其它相关联的原因。例如,取决于系统中使用的逻辑,在结果4被触发之前,结果4可能需要原因2、3、4和/或5中的一个或多个被激活(即,因为针对原因2、3、4和5中的每一个,结果4具有“X”)。因此,由CEM定义的逻辑可以基于“或”逻辑(即,结果列中任何一个原因的发生将导致结果的启动),或者可以基于“与”逻辑(即,结果列中的每个原因必须在结果由安全逻辑触发之前存在)。

[0054] 在另一个实施例中,取决于发生哪些原因,可以以不同方式触发结果(诸如结果4)。例如,如果发生一个相关联的原因,则可以在延迟上触发结果(诸如结果4),而如果发生

两个或更多个相关联的原因,则可以立即触发结果(诸如结果4)。此外,一些相关联的原因可以激活自动触发,而其它原因可以激活针对结果(诸如结果4)的延迟触发。此外,一些相关联的原因可以独立于其它相关联的原因而触发结果,而其它相关联的原因可以仅当它们结合一个或多个其它原因而存在时触发结果。所提供的示例并非旨在是限制性的,并且逻辑和/或延迟的任何组合可以通过对应的因果对来实现。

[0055] 如将更详细地讨论的,由CEM定义的逻辑可以被分解为在由CEM定义的原因和结果的子集上实现的多个逻辑集合或逻辑群组,并且这些不同的逻辑子集可以通过安全逻辑实现中的特定功能块来实现。例如,可以使用功能块来实现由CEM 300中示出的选定逻辑块305和310定义的逻辑。在该情况下,逻辑块305将包括两个原因输入(C2和C3)并且将对应于三个结果输出(E3、E4和E5)。在该示例性实施例中,通过简单地识别CEM 300的填充单元的集群(cluster)来识别将由逻辑块305和310实现的逻辑子集。这里,逻辑块305和310虽然仅覆盖CEM 300的49个单元中的12个,但是包含由CEM 300指示的大部分重要信息(因/果关系)。在其它实施例中,可以使逻辑块305和310更大和/或可以添加或识别另一个逻辑块以包括CEM 300的未被包括在逻辑块305和310中的剩余填充单元。如将在下面更详细讨论的,CEM可以被重新排列以更好或更有效的方式聚集填充单元,从而帮助识别逻辑块,并且转而创建功能块。虽然对于给定CEM 300这种聚集可能看起来像轻松的事,但是人可能几乎不可能有效地识别具有数百(或数千)个单元的CEM中的模式(pattern)。

[0056] 在传统系统中,CEM由状态机功能块表示,其中原因是输入并且结果是输出。通常,为CEM中的每个结果创建状态机功能块。因此,状态机功能块的使用受其定义的大小限制,因此可以大量地猛增。然而,与传统系统不同,当前系统将CEM组织成两种类型的功能块:监控块和结果块,其用于在实现复杂或大的CEM时降低逻辑复杂度并增加安全系统内的逻辑实现的优化。

[0057] 更具体而言,使用分开的监控块和结果块来实现在对应的CEM中定义的任何逻辑模式或逻辑群组,通过将原因与结果分成两种不同类别的块来解决传统系统的缺点。通常,监控块(MB)是原因的抽象表示,结果块(EB)是结果的抽象表示。这样,系统可以通过链接或以其它方式连接到一个或多个结果块的一个或多个监控块来表示大的CEM及其原因和结果。例如,监控块集合的输出可以用作到一个或多个结果块的输入,并且相应地每个结果块的输入可以源于来自一个或多个监控块的输入。在一实施例中,监控块的输出可以替代地或另外地用作到一个或多个其它监控块的输入。因此,监控块和/或结果块可以根据需要被链连接、嵌套、分层和/或分级以便最佳地实现任何期望的CEM逻辑。此外,将CEM表示(以及实现)为多个MB和EB使得安全系统的实现和维护更容易,并且还允许容易地表示和配置更复杂的CEM关系。

[0058] 创建单独的监控块和结果块有许多优点。具体而言,MB和EB可以根据需要设计大小,这导致不易出错的更快且更简单的实现。此外,由于对因果关系的透明描述,使用这些较小大小的MB和EB作为功能块来实现CEM逻辑的控制或安全系统可以更容易地进行测试和故障排除(或一般地,反向工程)。此外,大的CEM可以分解为更可管理的大小的逻辑块。此外,通过使用单独的MB和EB来更容易地表示复杂的因果关系。例如,分层、循环、嵌套、链连接等都可以使用单独的监控块和结果块来描绘。

[0059] 图4是互连的监控块和结果块集合的示意图400。图4的监控块和结果块集合包括

(实现)在图3的CEM 300中提供的或由图3的CEM 300定义的所有信息或逻辑。这里,监控块405和410通常对应于逻辑块305和310的原因(C2-C5),而结果块415和420通常对应于逻辑块305和310的结果(E3、E4、E5、E6)或与逻辑块305和310相关联的结果(E3、E4、E5、E6)。例如,监控块1(MB1)405包括CEM 300的原因2和原因3作为输入。然而,MB1 405的输出不直接对应于CEM 300的结果3、4和5(如同逻辑块305)。与通常被创建以实现逻辑块305和310(逻辑块305和310实现直接因果关系)的逻辑的状态机相比,监控块可以包括输入(诸如来自其它监控块的原因和输出)和输出(其可以被发送到其它监控块或结果块),但其不直接对应于结果。例如,MB1 405的输出被发送到集合400中的各种其它监控块和结果块。具体地,监控块MB1(405)的输出401被发送到结果块1(EB1)415,监控块MB1的输出402被发送到监控块2(MB2)410,并且监控块MB1的输出403被发送到结果块2(EB2)420。

[0060] MB的输出通常提供关于对应输入的信息。例如,输出401提供关于原因2的信息(MB1 405的对应单元中的'X'表示这种关系)。类似地,输出402提供关于原因2和/或原因3的信息。例如,如果原因C2或C3中的任一个存在(例如,逻辑上为真),则输出402可以为高(逻辑1),或者仅当原因C2和C3都存在时输出402可以为高。当然,可以关于原因C2和C3执行其它逻辑操作以确定输出402,诸如异或等。以类似的方式,MB2 410的输出411提供关于三个输入(MB1 405的输出402、原因4和原因5)的信息。换而言之,输出411提供关于原因2和3(如由产生原因块MB1的输出402的逻辑所定义的)以及原因4和5的信息。

[0061] 现在参考结果块415和420,结果块1(EB1)415接收两个输入,即来自监控块MB1的输出401(其取决于原因2的状态)和原因6(对应于来自CEM 300的原因6)。结果块EB1(415)仅对应于一个结果,即结果3。因此,与CEM 300类似,结果块EB1(415)将原因2和6相关联以创建结果3,结果3是结果块EB1(415)的输出。如将理解的,结果块EB1(415)可以基于输出401的状态(其又与原因2的状态有关)和原因6(C6)的状态来实现任何期望的逻辑和延迟。

[0062] 类似地,结果块EB2(420)对应于并实现逻辑,该逻辑创建或定义CEM 300的结果4、5和6的状态。追溯结果块EB2(420)的输入回到对应的监控块,可以看出针对CEM 300的结果4、5和6的因果关系受结果块EB2(420)影响。特别地,结果块EB2 420接收输出411,输出411基于输入到监控块MB2的原因4和5并基于监控块MB1的输出402。因此,输出411具有从用于触发结果块EB2中的结果4的原因2、3、4和5导出的值或状态。此外,结果块EB2接收输出403和412,输出403和412由原因2和4逻辑地定义,并且在某些逻辑表达式中被用来触发结果5。此外,结果块EB2接收对应于或被定义为基于原因4和5的逻辑值的输出413,并且使用输出413来触发结果6。转而,图4的监控块和结果块集合400包括在图3的CEM 300中先前提供或针对图3的CEM 300定义的所有关系信息(和逻辑)。虽然在该示例中通过将CEM 300分解成监控块和结果块集合400而提供的优点可能不是明显的,但是在分解更大的CEM时,优点更明显。应该注意的是,如图4所描绘的监控块和结果块集合400仅意味着作为示例,并且监控块和结果块可以以无数的大小和配置来创建和组织,以实现由CEM定义的逻辑。

[0063] 图5是配置屏幕500的图示的一个示例,配置屏幕500可以由显示设备显示并且表示或描绘实现CEM或CEM的一部分的逻辑的监控块和结果功能块集合。配置屏幕500表示监控块和结果块的更详细的功能块实现,相比于监控块和结果块集合400,监控块和结果块集合400旨在作为与监控块和结果块相关联的逻辑的示意表示。在图5的示例中,配置屏幕500包括输入(原因508、原因掩码512和逻辑类型506)、对应于图4的监控块MB1(405)的监控块

502、以及对应于图4的结果块EB1(415)的结果块504。

[0064] 监控块502接收分别对应于两个原因508、原因掩码输入512和逻辑类型506的四个输入(IN_D1和IN_D2、IN_MASK和LOGIC_TYPE)。逻辑类型506定义在当前的监控块和结果块集合中正在实现什么类型的逻辑。在一实施例中,逻辑类型可以是正的或负的。正逻辑可以指示所有原因最初以“假”状态开始,并且如果触动变为“真”。因此,如果一个或多个原因是“真”,则对应的输出可以是“真”。转而,对应的结果块可以接收一个或多个“真”输入,这可以提升结果块的状态和/或触发结果块。负逻辑可以是相似的,但原因最初以“真”开始,并且如果原因发生则设置为“假”。该示例性逻辑并不旨在是限制性的,并且逻辑类型506还可以包括可能有助于实现监控块和结果块的“与”逻辑、“或”逻辑或任何其它逻辑。

[0065] 原因掩码输入512可以表示用于过滤由监控块502接收的原因508的初始参数。监控块502还包括用于配置监控块502的三个配置掩码CFG_MASK 1、CFG_MASK 2和CFG_MASK 3 510,其中每个掩码表示哪些原因对应于每个输出,并且在一些情况下,表示用于从非掩蔽的输入生成输出的逻辑。配置掩码510可以是从CEM导出的数值表示,如下面更详细描述的。

[0066] 监控块502还包括五个输出(OUT_D1至OUT_D3 514、RAW_VAL 516和MASK_VAL 518),其中输出514中的一个(OUT_D1)用作对结果块504的输入(如图4的配置中所标识的)。原始值516可以简单地输出接收到的原因508的值,而掩蔽值518可以在应用原因掩码512之后输出原因508的值。此外,OUT_D1至OUT_D3对应于图4的MB1 405的输出401-403。配置掩码510指示哪些原因对应于每个输出。例如,(配置掩码510的)CFG_MASK1被设置为‘A’,其可以指示仅(原因508的)原因2对应于(输出514的)OUT_D1。此外,(配置掩码510的)CFG_MASK2被设置为‘B’,其可以指示(原因508的)原因2和原因3对应于(输出514的)OUT_D2。此外,在一些情况下,配置掩码510可以是数值表达式,诸如十六进制数,表示哪个监控块输入驱动或影响特定监控块输出、和/或用于从块输入生成块输出的实际逻辑。

[0067] 如图5所示,结果块504可以包括四个输入(IN_D1和IN_D2 520、重置522和LOGIC_TYPE 506)以及两个输出(状态526和OUT_D 524)。结果块504的输入520包括监控块502的输出514以及图3的CEM 300的原因6。结果块504的状态526可以对应于与结果块504对应的设备的操作状态。换而言之,如果没有接收到对应的“真”原因,则状态526可能是正常的。然而,如果接收到例如设置为“真”值的一个或多个原因,则状态可以改变以指示新的状态(例如,“警告”、“危险”、“触发”)。当结果块504处于非正常状态时,一旦必要的动作已经被执行,重置输入522就可以允许用户自动地将结果块504的状态重置为正常。当然,也可以向结果块提供其它状态改变输入,例如允许重置(例如,在原因输入或监控块输入改变状态等时)。此外,在该情况下,如果接收到的输入520中的一个或多个是“真”,则对应于CEM 300的结果3的输出OUT_D可以是触发的响应,因为在该示例中的逻辑类型506被设置为正。

[0068] 作为示例,图5示出了当原因2和原因3输入是“假”并且块502和504两者的逻辑类型被设置为正时(块502的)OUT_D1、(块504的)IN_D2和Out_D中的每一个的状态。现在,如果(原因508的)原因2将在过程工厂中发生,则原因2的状态可以从“假”改变为“真”。因此,MB1 502将接收输入IN_D1为真。基于Logic_Type输入处的正逻辑类型和用于OUT_D1(输出1)的配置掩码(即,CFG_MASK1),对应的输出OUT_1将随后变为真。在该示例中,在应用配置掩码510之后,IN_D1驱动或影响输出514(OUT_D1至OUT_D3)中的每一个的值。特别地,基于IN_D1被设置为“真”,OUT_D1可以被设置为“真”。因此,EB1 504的状态526随后将接收至少一个

“真”输入(IN_D1)。因此,EB1 504的状态526将被改变为“触发”,并且结果块504的输出524OUT_D将被设置为“真”,这意味着图4的结果3将被触发或为“真”。因此,该结果已经被触发,并且过程控制工厂中的任何对应的动作和/或警报可以开始。

[0069] 图6是配置屏幕600的图示的另一示例,配置屏幕600可以由显示设备显示并且可以表示监控块和结果块的配置。图6的示例关注于来自图4的监控块MB2和结果块EB2。出于说明的目的,逻辑类型606被设置为负,这意味着当处于正常状态时,所有原因都是“真”。在负逻辑类型中,当发生一个或多个输入时,状态切换到“假”,监控块的对应输出可以被设置为“假”,结果块可以接收为“假”的输出并随后触发将结果设置为“真”的结果。

[0070] 此外,结果块EB 604包括时间延迟输入608。在该示例中,当“20”被输入到结果块604的Delay_Time 1输入时,结果块604的输入1(IN_D1)可以使得输出OUT_D以20秒的延迟(DELAY_TIME1)被触发。然而,因为时间延迟(DELAY_TIME2)被设置为零,所以该示例的输入2(IN_D2)可以使得输出(OUT_D)被立即触发。该示例并不旨在是限制性的,并且可以为特定的结果块设置任意数量的延迟和延迟时间。

[0071] 作为示例,如果原因4将发生,原因4的状态(并且因此IN_D2)将改变为“假”。参考回到图4,可以看到原因4对应于或影响监控块410的输出411-413中的每一个。因此,在该示例中,原因4(和IN_D2)可以根据由监控块602的配置掩码实现的逻辑来驱动或影响MB2的所有输出(OUT_D1至OUT_D3)。特别地,MB2 602的OUT_D1可以被设置为“假”。转而,EB2 604的IN_D2将被接收为“假”。因此,当对应于EB2604的IN_D2的时间延迟(DELAY_TIME2)被设置为零时间延迟时,EB2 604的对应结果可以被立即触发。作为示例,如果图4的MB2 410的输出411对应于MB2 602的OUT_D1,则这些输出仅驱动(CEM 300的)结果4,其可以被表示为EB 604中的OUT_D1。因此,EB 604的OUT_D1现在可以被触发并设置为“真”。

[0072] 图5和图6中提供的示例性监控块和结果块旨在是简化的以用于演示目的。例如,虽然监控块502在图5中示出为具有四个输入和五个输出,但是其它实施例可以包括基于监控块的功能所需的任何期望数量的输入和输出。在一个实施例中,输入IN_Dx和输出OUT_Dx的数量通常对应于重新组织的CEM的每个逻辑块中的输入和输出的数量。此外,该系统可以配置掩码,使得可以实现一个监控块以驱动或影响多个结果块和另外的监控块。转而,CEM可以通过分层、循环、嵌套、链连接等分成多个监控块和结果块,这可以为系统提供比传统状态机实现方式更大的灵活性来配置过程控制工厂的系统。

[0073] 图7是配置与过程工厂相关联的监控块和结果块的示例性方法700的流程图。方法700可以周期性地实现和/或响应于触发事件(例如配置工程师或其他用户或其他安全逻辑设计者的指示或启动信号)来实现。方法700可以由电子设备(例如,因果分析器工具17)执行,该电子设备可以包括过程工厂(诸如参考图1所讨论的过程工厂10)的一个或多个部件。

[0074] 在框710处,电子设备可以接收或以其它方式访问CEM。在某些实施例中,重新排列CEM以在识别逻辑块之前去除稀疏并且以其它方式收集群组的集群中的信息可能是有益的。在框715处,电子设备可以自动重新排列CEM和/或使用户重新排列CEM。下面更详细地讨论用于自动重新排列CEM的方法。在框720处,电子设备可以识别并创建监控块和结果块集合以实现CEM的逻辑。在框730处,电子设备可以监控块和结果块显示给用户,诸如可以设计用于实现CEM的安全或控制逻辑的配置或安全逻辑工程师。特别地,电子设备可以使得显示设备显示图形用户界面(GUI),其中GUI可以指示第一监控块、第二监控块和结果块。此外,

第一监控块、第二监控块和结果块中的每一个可以指示以以第一维度和第二维度的矩阵排列的多个单元,其中沿着第一维度的位置可以指示输出,并且沿着第二维度的位置可以对应于输入,以使得多个单元可以基于该多个单元相对于第一维度和第二维度的位置来定义输入/输出对。

[0075] 在框740中,电子设备可以配置或者可以使用户配置监控块和结果块以实现CEM的逻辑。在一实施例中,电子设备可以使用户经由输入设备来输入配置数据。在另一个实施例中,电子设备可以通过解析CEM来自动确定或生成配置数据。根据实现方式,电子设备可以配置第一监控块的输出中的一个输出以用作第二监控块的输入中的一个输入,可以配置第一监控块的输出中的另外一个输出以及第二监控块中的输出中的一个输出以用作对结果块的输入,和/或可以将第一监控块、第二监控块和结果块中的每一个的多个单元中的至少一个单元指定为触发,该触发与相应单元的相应输入/输出对相关联并且对应于过程工厂中的状况。

[0076] 在一实施例中,为了配置监控块和结果块,电子设备可以并入(incorporate)具有定义另外输入/输出对的另外多个单元的至少一个另外监控块,配置该另外监控块的至少一个输出以用作对第一监控块、第二监控块和结果块中的至少一个的输入,并且将该另外多个单元中的至少一个单元指定为另外触发,该另外触发与相应的另外单元的相应的另外输入/输出对相关联并且对应于过程工厂中的另外状况。在另一个实施例中,为了配置监控块和结果块,电子设备可以并入具有定义另外输入/输出对的另外多个单元的至少一个另外结果块,配置另外结果块的至少一个输入以对应于第一监控块或第二监控块中的一个的输出,并且将该另外多个单元中的至少一个单元指定为另外触发,该另外触发与相应的另外单元的相应的另外输入/输出对相关联并且对应于过程工厂中的另外状况。

[0077] 另外,在一实施例中,为了配置监控块和结果块,电子设备可以为第一监控块和第二监控块中的每一个配置输入,可以为第一监控块和第二监控块中的至少一个配置输入掩码,输入掩码与第一监控块和第二监控块中的至少一个的输入在逻辑上相关联,可以将至少一个触发指定为时间延迟触发以使得相关联的结果以时间延迟进行激活,和/或可以将至少一个触发指定为许可触发。

[0078] 在框750中,电子设备可以储存配置的监控块和结果块。特别地,电子设备可以将配置数据储存在与第一监控块、第二监控块和结果块相关联的计算机可读介质上。在一实施例中,电子设备还可以在显示设备上显示第一监控块、第二监控块和结果块中的每一个的多个单元,并且可以指示相应的多个单元内的相应触发。

[0079] 当然,方法700可以创建任意数量的监控块和结果块,这些监控块和结果块以任意数量的方式连接在一起以使用这些互连的监控块和结果块来实现CEM的逻辑。每个监控块可以包括CEM的任何数量的原因或原因的任何子集作为对其的输入,并且可以包括绑定到其它监控块的输出的输入,由此影响级联的监控块。此外,任何结果块可以从输入集合确定一个或多个结果,并且可以接收监控块的任何输出和/或任何原因输入作为输入。此外,方法700可以互连或使用户互连各个监控块和其它监控块以及结果块(即,定义各个监控块和其它监控块以及结果块之间的连接)。这样,每个监控块包括逻辑,该逻辑基于一个或多个原因信号(直接输入到监控块或者以由输入到另一个上游监控块的原因信号形成的另一个中间逻辑信号的形式输入到监控块)来确定一个或多个中间逻辑条件或信号。类似地,每个

结果块基于对其的输入集合来产生一个或多个结果信号,其中这样的输入是从一个或多个监控块输出的原因信号和/或中间逻辑信号。以此方式,方法700使中间逻辑信号能够在表示原因信号的某些逻辑组合的一个或多个监控块中形成,并提供或使用该中间逻辑信号作为对一个或多个结果块的输入,从而简化由结果块实现的配置、大小和逻辑来创建结果信号。

[0080] 对于较小的CEM,有可能的是,安全工程师在方法700的框715处手动重新排列和/或配置CEM,诸如通过识别模式或通过尝试对相关的原因和结果进行分组。这样的重新排列可以由用户经由图形用户界面手动实现,该用户移动或重新排列CEM的各个行和/或列以将定义因果关系的单元(例如,标记有X的单元)分组为彼此接近或形成更密集的分组。然而,重新组织大的CEM有多种方式,并且确定用于重新组织的最佳选项是有益的。因此,有机会动态地和自动地分析并重新组织与过程控制系统相关联的CEM。

[0081] 在一实施例中,系统(即,图1的计算机系统)可以实现因果分析器工具17以基于规则集合来自动地重新组织大的CEM。在一实施例中,规则31可以储存在图1的配置数据库32中和/或可以经由工作站18a和/或20a的用户界面来接收。分析器工具17可以分析CEM以在给定规则集合31的情况下确定CEM的最适当的或优化的配置(即,重新排列CEM以产生监控块和结果块集合的最佳方式)。规则集合31可以由工程师指定,或者基于特定过程工厂的当前需求或配置由计算机(例如分析器工具17)自动生成。例如,规则集合31可指示CEM应当被组织成群组,其中某些原因和/或结果基于对应的逻辑解算器50、MPD 70和/或现场设备22、23、24、60和62而分组在一起,在逻辑解算器、MDP和/或现场设备中将实现该逻辑。此外,规则集合31可以指示:基于特定模式、基于系统的效率和/或基于其它标准,CEM应该被重新组织以去除稀疏性。在另一个实施例中,该规则集合31可以指示某些原因和/或结果(或者原因和/或结果的群组)不被移动。在另一个实施例中,规则集合31可以指示需要重新组织的特定原因和/或结果的权重,当尝试应用导致不同结果的多个规则时,所述权重用来解决冲突。

[0082] 在一实施例中,规则集合31可以指示CEM将被重新组织为特定数量的群组和/或特定大小的群组。规则集合31还可以指示组织群组的方式。例如,规则集合31可以指示每个群组应该包含特定数量、特定最大数量或特定最小数量的原因和/或结果。在一实施例中,规则集合31可以指示这些群组不应该包含重叠的原因和/或结果。规则31还可以指定某些原因或结果应该被分组在一起,因为例如这些原因将被特定逻辑解算器检测到或在特定节点中被检测到,或者结果可能需要由特定节点处的特定逻辑解算器实现。无论如何,一旦CEM被重新组织,规则集合31还可以使工程师能够手动配置CEM中的特定原因和/或结果。应该理解的是,可以设想替代的或另外的规则。

[0083] 在一实现方式中,分析器工具可以接收或生成规则集合31,该规则集合31指示:仅应该重新组织与过程工厂的某些区域相对应的某些原因和/或结果,或者应该将这些原因和结果重新组织在一起或者重新组织为群组。类似地,规则集合31可以指示仅CEM的原因和结果的某个子集应该被重新组织。分析器工具还可以接收或生成规则集合31,该规则集合31“锁定”某些行和/或列以防止在重新组织期间移动所接收的行和/或列。此外,分析器工具可以接收或生成规则集合31,该规则集合31指示:对应于正逻辑的原因(即,如果原因是“开启(on)”,则结果被激活)应该被分组在一起并且对应于负逻辑的原因(即,如果原因是

“开启”,则结果不被激活)应该被分组在一起。也可以使用基于CEM单元中定义的逻辑的类型(即,要实现的逻辑的类型)对CEM行和列进行分组或重新组织的其它方式。

[0084] 相应地,CEM的重新组织可能需要多部分分析(multi-part analysis),该多部分分析可以由计算机实现并且可以基于该规则集合31。计算机可以基于对应的逻辑解算器50、MPD 70和/或现场设备22、23、24、60和62,或者通过最适合于实现规则集合31的任何其它元件,按照行、按照列、按照群组、按照触发对CEM进行分析。例如,图8是比图3的先前示例性CEM 300明显更大的示例性CEM 800。CEM 800包含分散在整个矩阵中的多个填充单元。虽然CEM 800仅比CEM 300略大,但是很显然,识别将由监控块和结果块集合实现的CEM 800中的逻辑块或逻辑群组的问题越来越复杂。此外,CEM 800包括从较大分组分散的填充单元,这增加了有效选择要用于生成监控块和结果块的逻辑块的困难。对于大小越来越大的CEM,手动选择或定义逻辑块的困难变得非常大。

[0085] 图9示出了描绘CEM 900的显示的一个示例,CEM 900是来自图8的CEM 800的重新组织的版本。如图9所示,CEM 900已被组织为包括三个基本(primary)群组或逻辑块:901、902和903。在示例性实施例中,逻辑块901、902和903中的每一个可以对应于过程工厂内的特定逻辑解算器50。在另一个实施例中,计算机可能基于规则集合31内定义的标准而已识别逻辑块901-903。

[0086] 例如,图9的逻辑块901可以对应于全部属于过程工厂的特定物理位置(例如,特定加热段)中或将由工厂控制系统中的相同控制器或逻辑解算器实现的结果集合。此外,逻辑块902可以由分析器工具通过识别其中所有原因与所有结果是相关的群组来从CEM去除稀疏性而得到。特别地,在逻辑块902中,原因4-10中的每一个与结果3-5中的每一个配对。逻辑块903可以对应于负逻辑因果关系的群组。虽然示例性CEM 900包含三个逻辑块,但是CEM可以被分解成任何数量的逻辑块,并且基于上述任何规则31、或者任何组合或规则31或者可以由分析器工具17在分析和重新组织CEM时使用的任何其它上面未提及的规则。如上面关于图3和图4所描述的,逻辑块901、902和903可以各自用于定义互连的监控块和结果块集合以实现CEM的这些部分的逻辑。

[0087] 图10是重新排列因果矩阵以及定义和/或管理要用于开发过程控制系统的安全或控制逻辑的CEM的逻辑块的示例性方法1000的流程图。方法1000可以周期性地实现和/或响应于触发事件来实现,例如,诸如在工厂的配置期间,每当逻辑的CEM被改变或更新时等。方法1000可以由电子设备(例如,图1的分析器工具17)执行,该电子设备可以包括过程工厂(例如参考图1所讨论的过程工厂10)的一个或多个部件。在框1010处,电子设备可以访问具有输入集合和输出集合(即,原因集合和结果集合)的初始因果矩阵。在实施例中,输入集合中的每一个输入可以表示过程工厂内的状况,并且输出集合中的每一个输出可以表示将在过程工厂内执行的结果。此外,输入集合和输出集合中的至少一些可以是作为因果对而相关的,由此对应的结果可以响应于对应状况的发生而激活。初始因果矩阵(CEM)可以储存在过程控制工厂中的数据储存库中,或者可以由用户在电子设备处生成以用于配置工厂中的新过程。还可以从过程控制系统之外的数据库接收初始CEM。在一些实施例中,初始CEM可以仅由具有适当的凭证的工程师访问,并且因此可能需要登录或其它密码来授权对初始CEM的访问。

[0088] 电子设备可以在初始CEM内定义相关群组集合中的每一个群组。特别地,在框1020

处,电子设备可以访问与相关群组集合相关联的规则集合31。特别地,电子设备可以通过过程控制系统内部或外部的一个或多个数据库来访问规则集合31。电子设备还可以将规则集合31作为由过程控制工厂的工程师提供的输入来接收。此外,规则集合31可以是通过各种数据库和/或输入而访问的各种规则的组合。如上面详细讨论的,规则集合31的目的是可以以有效和高效的方式重新组织CEM。

[0089] 在一个实施例中,规则可以指定输出集合中的指定部分必须在相同的相关群组内。在另一个实施例中,规则可以指定输入集合的部分必须编号一定数量。在又一实施例中,规则可以指定输入集合和输出集合都不应该在相关群组集合之中重叠。当然,可以使用任何其它期望的规则。

[0090] 在框1030处,电子设备可以根据如由在CEM中定义的对应因果对所定义的规则集合来识别与输出(结果)集合的一部分相关的输入(原因)集合的一部分。此外,在框1040处,电子设备可以重新排列输入集合的该部分和输出集合的该部分,使得对应的因果对的部分被重新排列。框1040可以执行这种重新排列以定义将使用监控块和结果块集合(如上面所定义的)来实现的一个或多个功能块逻辑单元。框1050可以分析重新排列的CEM并且判定过程是否完成,并且如果没有完成,则提供控制到框1030以识别将被用于重新排列CEM的其它规则,进一步力图基于重新排列的CEM来优化监控块和结果块的创建。此外,当重新排列完成时,框1050可以在重新排列的CEM内定义逻辑块或逻辑群组,诸如图9的三个逻辑分组901、902和903。

[0091] 在一实现方式中,电子设备可以根据由框1050定义的相关群组集合来进一步为过程控制系统配置一个或多个功能块逻辑单元。另外地或替代地,针对相关群组集合的每个相关群组,电子设备可以根据重新排列的因果对来自动计算相关群组或相关群组的一部分的数值表示,诸如通过计算下面参考图11-图12更详细地讨论的相关群组的十六进制表示。

[0092] 一旦分析器工具已经重新组织CEM 900,系统可以进一步将CEM 900分解成单独的逻辑群组,以进一步提高在创建实现那些逻辑群组的监控块和结果块时的效率。图11描绘了图9的CEM 900的另外表示。特别地,系统可以分析图9的CEM 900以产生各种数值表示1101、1102和1103,系统可以使用这些表示来配置功能块作为互连的监控块和结果块集合。在一实施例中,数值表示1101-1103中每一个可以基于由重新排列的CEM 900所定义的原因和结果对定义的逻辑关系的配置,将输出或结果表示或定义为诸如十六进制值之类的值。该数值与将每列表示为逻辑表达式的传统系统形成对照。然而,由于实现或理解逻辑表达式的困难,这样的传统系统是低效的。

[0093] 在一实施例中,系统可以通过为矩阵中的每个单元分配两个值中的一个(例如,开启(ON)或关闭(OFF),1或0等)并随后将CEM的行或列的每个比特群组(例如,四个二进制数字)转换为十六进制数字来指定(devise)数值表示。例如,如图11所示,输出14的数值表示1101是与输出14相关联的单元的十六进制表示(FE08),其中单元中的X被视为二进制“1”并且空单元被视为二进制“0”。该计算可以通过将输出14分解成4比特群组(其通过单元之间的较粗线划分并且组成从上到下的四个比特数字:1111、1110、0000、1000)来演示,随后将每个比特群组转换成十六进制数字。在该情况下,输出14的数值表示1101是FE08,因为在十六进制数字中,F=1111,E=1110,0=0000和8=1000。以相同的方式,数值表示1102(07E0)对应于输出5,因为从上到下,输出5可以被分解成比特0000、0111、1110、0000,它们转换成

十六进制数字07E0。以类似的方式,输出17可以表示为十六进制数字0072(数值表示1103)。示例性数值表示并不旨在是限制性的,并且列和/或行中的一些或全部可以被分配数值表示。此外,数值表示不一定必须是十六进制转换,并且可以以任何其它合适的形式来进行。

[0094] 为CEM内的逻辑单元的某些分组制定数值表示有许多益处。特别地,与传统系统相比,列到十六进制转换更简单,不需要额外的门电路(gate)或编程来生成表达式,占用更少的存储空间来储存,并且占用更少的带宽来与功能块输入进行通信。此外,十六进制值输入可以根据需要更容易地进行纠错以确保准确性,下面参考图15的测试矩阵进行讨论。

[0095] 数值表示可以进一步使系统能够配置安全系统配置环境的因果关系。特别地,数值表示可以使系统能够定义大量原因和结果之间的关系。此外,数值表示可以通过将整个行和/或列提炼(distill)成单个数值来帮助消除配置差错。此外,数值表示可以提供简单而有效的方式来识别因/果关系的变化,并进一步减少管理CEM中的变化所需的努力。

[0096] 例如,数值表示可以被实现为功能块(诸如图4-图6的监控块和结果块)中的配置掩码。因此,这些数值表示可以实际上识别在监控和/或结果块中针对特定结果要实现的逻辑。数值表示可以定义哪些输入对应于每个特定输出,并因此解关联(即,掩蔽)不对应于该特定输出的输入。例如,图11中的结果14的数值表示1101可以将所有原因6-13与结果14解关联。换而言之,监控块可以接收全部原因1-16,但是如果实现数值表示1101作为掩码,则仅将原因1-5和14-16与结果14相互关联。

[0097] 此外,当可能的单元值的范围大于2时(例如,当单元可以定义多个不同触发,诸如无值、X、T(指示时间延迟)、P(指示许可原因)等等时),系统可以调整数值表示。例如,对于四个可能的交叉点值的示例性范围,系统可以执行两个十六进制转换来生成所得到的数值表示。换而言之,每个单元的四个可能的不同值可以被表示为两比特数字的四个可能值中的一个值,这意味着每个单元将由两比特值而不是一比特值来定义,如图11所示。在该情况下,字符串的两个相邻单元的每个集合会形成四比特值,该四比特值可以被转换为十六进制数字。因此,该场景下的数值表示将是图11所示的数值表示两倍长,但会更加动态的,因为其可能表示将在实现CEM的逻辑中使用的更大数量的潜在逻辑表达式。替代地,系统可以使用除基16之外的适当的基来计算数值表示,并且随后可以可选地将数值表示转换(如果需要)为功能块(即,监控和结果块)的十六进制输入值。

[0098] 图12是用于创建/计算CEM内的值或元素的数值表示的示例性方法的流程图。在框1210处,因果分析器工具17可以访问CEM。在示例性实施例中,可以在继续之前重新排列CEM。在框1220处,工具17可以识别原因的子集。在一实施例中,如上所述,原因的子集可以属于特定的逻辑块和/或由规则集合定义。接下来,在框1230处,工具17可以为原因的子集定义单维度矩阵。单维度矩阵可以对应于CEM的特定结果。接下来,在框1240处,工具17可以计算单维矩阵的数值表示。如上所述,工具17可以将单维度矩阵转换为二进制字符串和/或多个二进制字符串。在一实施例中,工具可以随后去将一个或多个二进制串转换为十六进制表示或任何其它合适的数值表示。所计算得的数值表示可以作为数值表示33储存在储存库(诸如图1的配置数据库32)中。

[0099] 在框1250处,数值表示33可以随后用于配置功能块(例如,监控块和结果块)集合。例如,如上所述,数值表示33可以被实现为一个或多个监控块中的配置掩码。

[0100] 在本文描述的系统的另一方面中,系统导航器应用向用户提供在不同用户界面屏

幕之间快速导航的能力,所述用户界面屏幕为过程工厂提供相关的安全信息。这种信息可以在CEM、监控块和结果块、安全文档和系统配置显示中找到。在一些实施例中,这些不同的用户界面提供相同的安全逻辑的不同可视表示。因此,本发明提供(图1的)导航器工具15以在互链接的用户界面集合之间导航。例如,图13是互链接的用户界面集合的示例性图示1300。

[0101] 在一些示例性过程工厂中,安全协议以几种语言中的一种语言来进行编程。无论编程语言如何,安全协议的起点通常是指定对过程工厂的控制和/或安全动作的要求的叙述性文档。在其它示例性过程工厂(诸如安全仪表系统(SIS))中,安全要求被记录在称为安全要求规范(SRS)的文档中。

[0102] SRS的输入中的一个是识别的安全仪表功能(SIF)的列表。每个SIF防止特定危害,并提供定义的风险降低级别。SIS由一个或多个SIF构成。在一些实施例中,一些安全系统对SIS配置中的所有SIF进行组合而不区分每个单独的SIF。此外,一些安全系统遵循SIF方式并允许基于SIF的SIS配置。

[0103] SRS通常包括不同的部分。这些部分中一个部分是可以由纯文本、逻辑图或因果图(即,因果矩阵)表示的逻辑描述。如所提及的,一些安全系统对SIS配置中的所有SIF进行组合,并且CEM可视化在实现这样的实施例时可能非常方便。

[0104] 在一个实施例中,导航器应用15可允许工程师选择CEM内的给定原因(和/或结果)以导航到描述所选原因(和/或结果)的特定文档。例如,选择原因可以重定向到SRS中的特定SIF描述。该特征允许工程师查看与原因和/或结果相关联的特定安全逻辑。在一实施例中,工程师还能够选择与给定SIF相关联的安全模块(系统配置),随后可以被重定向到显示来自CEM的适当SIF的用户界面。此外,工程师可以选择CEM的元素并被重定向到与CEM的特定元素有关的突出显示(highlight)设备、逻辑块、功能块、监控块和结果块等的系统配置的显示。从安全或控制模块,用户也可以被重新定向到SRS或控制叙述中的适当部分。换而言之,本系统可以允许工程师在CEM、SRS或系统配置的视图之间无缝切换。

[0105] 例如,如果工程师在显示CEM 1310的图13的用户界面中选择原因和/或结果,则工程师可以被重定向到显示屏幕1320,该显示屏幕1320示出了包括与CEM的所选原因和/或结果有关的特定设备的系统配置。例如,系统配置1320可以包括与CEM的所选原因和/或结果有关的用于罐、阀、变送器、泵、管道、传感器等的符号。在该示例中,温度计图标1321被突出显示,指示所选原因和/或结果对应于温度传感器读数。

[0106] 此外,从CEM 1310或系统配置1320,工程师可以访问描述过程工厂的安全协议的文档(诸如SRS 1330)。图13示出了安全要求规范的一部分的示例性显示1330,其包括描述相关安全过程的图标1331和文本1332。导航器应用15允许工程师在显示之间切换,为工程师提供先前难以取得的信息和洞察。此外,从界面1310、1320和/或1330中的任何一个,用户可以访问显示监控块和结果块(例如,其它功能块或逻辑)集合的用户界面1340,该监控块和结果块集合包括实现(CEM 1310的)相关所选元素的逻辑。

[0107] 在示例性实施例中,工程师可以右击CEM 1310的元素(或SRS 1330、系统配置1320、或者监控块和结果块1340)来访问下拉菜单。下拉菜单可以向工程师提供包括访问其它显示视图(诸如1310、1320、1330和1340)中和/或其它视图(如关于下面的图16A-D所述)的一个的能力的选项。

[0108] 用户可以容易地从CEM 1310(或者从CEM 1310的各个单元、原因或结果)或系统配置1320导航到要求规范(SRS 1330)内的特定部分,诸如一般旁路理念(general bypass philosophy)、检验测试要求等的定义。

[0109] 该功能将提供在配置与设计文档之间的来回无缝转换,以便于配置验证、变更管理、故障排除和检验测试。

[0110] 图14描绘了实现对包括在由过程控制系统控制的过程工厂的安全要求规范(SRS)中的信息的访问的示例性方法1400的框图。方法1400可以由服务器或任何类型的电子设备来容易实现,其中服务器可以配备有或连接到被配置为显示内容的用户界面。SRS可以储存在服务器可以访问的存储器中。

[0111] 方法1400可以从框1410开始,在框1410处,服务器可以在用户界面中显示CEM。在实施例中,(CEM)可以包括含有原因集合和结果集合的元素集合,其中原因集合中的每一个原因可以表示过程工厂内的状况,并且结果集合中的每一个结果可以表示要在过程工厂内执行的结果。此外,原因集合和结果集合中的至少一些可以是作为因果对而相关的,由此对应的结果可以响应于对应状况的发生而激活。

[0112] 在框1420处,服务器可以经由用户界面接收对元素集合中的元素的选择。特别地,服务器可以接收对原因集合中的原因的选择或对结果集合的中的结果的选择。响应于接收到该选择,在框1430处,服务器可以从SRS访问与元素集合的元素相关联的信息集合。特别地,服务器可以从SRS访问与所选原因或所选结果相关联的信息集合。根据实施例,服务器可以从SRS访问与所选元素相关联的管道和仪表图(P&ID)、与所选元素相关联的安全仪表功能(SIF)描述或其它信息。

[0113] 在框1440处,服务器可以在用户界面中显示信息集合。在一实施例中,服务器还可以发起被配置为经由用户界面显示与所选元素相关联的安全逻辑的应用。此外,在一实施例中,服务器可以经由用户界面接收对在用户界面中显示的信息集合的一部分的另外选择,从SRS访问与信息集合的该部分相关联的另外信息集合,并在用户界面中显示该另外信息集合。此外,在一实施例中,服务器可以经由用户界面接收对在用户界面中显示的另外信息集合的选择,其中信息集合的该部分可以对应于CEM的元素集合中的另外元素,并且可以在用户界面中显示CEM和对该另外元素的指示。

[0114] 在一些实施例中,大的CEM可以包含数千个因果对。因此,这些大的CEM可以被分解为数百个监控块、结果块和数值表示。由于散布在众多数据结构上的大量信息,用户可能无法手动检查过程控制系统的安全逻辑是否在被准确地实现。先前的过程控制系统缺乏用于严格验证配置的过程控制系统是否满足所需安全协议的手段。换而言之,先前的系统没有对被实现为管理过程工厂的安全的CEM和功能块的准确性进行测试的方式。本公开内容提供了可以自动验证当前在过程工厂中实现的安全逻辑的工具(例如,因果分析器工具17)。

[0115] 在一个方面中,因果分析器工具17可以自动遍历过程工厂(或其一部分)的配置以生成已建立(as-built)或已配置(as-cnfigured)系统的一个或多个测试CEM。在一实施例中,工具17可以基于功能块(即,监控块和结果块)和表示当前实施的过程工厂的安全逻辑的数值表示,通过逆向工程来构建测试CEM。随后可以将测试CEM与定义CEM的要求(已知是过程工厂所需的安全逻辑的准确表示的CEM)进行比较。该比较可以揭示可随后呈现给用户的差异或其它差错。

[0116] 图15是用于验证因果矩阵的安全逻辑的示例性方法的框图。在框1510处,分析器工具17可以确定表示因果矩阵的安全逻辑的一个或多个功能块的配置。在一实施例中,如上所述,功能块是包括输入、输出和数值表示的监控块和结果块。因果分析器工具17可以遍历监控块和结果块(MEB)的输入和输出,同时考虑若干因素(诸如在MEB中实现的逻辑和/或MEB的数值表示),以确定MEB的配置。例如,工具17可以接收如上面的图5和图6所描述的监控块和结果块集合。工具17可以从结果块的输出开始,并遍历结果块的输入到达输入源(即,直接反馈入结果块的原因和/或监控块的输出)。工具17随后可以基于数值表示将监控块的输出跟踪到监控块的对应输入。工具17可以继续这种针对每个结果迭代遍历MES的过程,直到每个因果对的每个关系已被识别为止。

[0117] 在框1520处,工具17可基于所确定的配置来生成测试CEM。基于所确定的监控块和结果块的配置,工具17可以利用所识别的因果对来填充测试CEM。一旦创建了测试CEM,工具17可以将测试CEM 37储存在数据储存库(诸如图1的配置数据库32)中。测试CEM 37可以实现为本文所描述的CEM中的任何CEM。

[0118] 在框1530处,工具17可以访问要求定义的CEM。在一实施例中,要求定义的CEM35可以储存在数据储存库(诸如图1的配置数据库32)中。在其它实施例中,工具17可以基于过程工厂中的设备的当前配置以及SRS和其它安全文档来创建要求定义的CEM。在一实施例中,要求定义的CEM 35可以包括原因集合和结果集合,其中因果对的关系是基于过程工厂的安全要求的。要求定义的CEM 35可以实现为本文所描述的CEM中的任何CEM。

[0119] 在框1540处,工具17可以将测试CEM 37与要求定义的CEM 35进行比较以确定是否存在任何差异。差异可以包括从测试CEM37到要求定义的CEM 35在因果对之间的任何差别。例如,因果对可能没有通过相同的触发类型(例如,许可的、立即的、延迟的)和/或相同的逻辑类型(和/或)而相互关联。

[0120] 工具17可以显示一个或多个所确定的差异中的任何差异。在一实施例中,工具17可以突出显示如图13和图16a-d中所描述的任何用户界面中的差异。换而言之,工具17可以突出显示不正确逻辑正被实现在CEM、监控块和结果块、SRS文档和/或系统配置用户界面中何处。

[0121] 关于上面关于图13和图14讨论的功能,查看过程工厂中一个或多个设备的状态历史、原因和/或结果也可能是有益的。本发明的另一方面提供了用于监控过程控制系统的安全系统状态的用户界面视图,其中状态通常基于或“关联到(keyed on)”物理设备和/或安全测试结果,并且可以描绘各种原因或结果的目前和过去状态,以向用户提供查看何时以及如何在工厂中实现CEM逻辑的某种能力。例如,工程师可以调出显示特定设备或某件装备或其群组的视图(如上面关于图13所讨论的),该视图随后可以用于进一步访问呈现每件被监控的设备/装备的当前和/或过去安全状态、在其上指示的原因信号、结果信号等(例如,图16A-D)。另外地或替代地,安全工程师可以对工厂中的特定物理设备/某件装备运行安全测试,并且结果可以显示在显示视图上(例如,图16A-D)。先前的系统不允许工程师快速监控和评估工厂(或工厂的期望区域的)的整体安全状态,而不必访问物理设备/装备的众多不同显示视图,或者通过必须运行特定的诊断来获得测试结果。先前的系统不仅对工程师不方便,而且在紧急情况期间,在工程师被迫费力通过多个视图或运行测试以便获取或找到他或她感兴趣的状态数据时会浪费宝贵的时间。

[0122] 本文描述的系统和方法提供了监控的安全事件的当前状态和/或状态变化的易于访问的显示视图(而不是特定设备、装备或测试结果的)。该系统将整个系统范围或区域范围的安全事件/输入状态聚合到单个显示视图或可视化中,随时间捕捉安全事件状态的变化,并将整个安全显示视图上的可视化安全事件链接到设备/装备/测试结果。

[0123] “安全事件”是监控的状况的逻辑表示。在示例性实施例中,CEM的每个监控的输入(原因)可以是监控的安全事件。另外,每个结果可以是监控的安全事件。期望要监控的每个安全事件的相应状态和/或相应的状态变化由安全事件可视化视图上的不同对象/项目/图形项目表示。例如,每个监控的事件可以由彩色点表示,其中不同的颜色表示不同的当前状态(例如,红-坏,蓝-注意,黑-OK)。另外地或替代地,可以例如通过不同的颜色或表示来表示当前状态的改变(二进制和/或变化程度)。这些状态和/或状态的变化可以随时间捕捉并保存。实际上,显示视图可以为监控的事件提供时间的滚动快照,并且可以包括针对以不同速率(例如,每2分钟、每20分钟、每2小时)监控的安全事件的不同部分。

[0124] 图16A-D是安全事件随时间的状态变化的示例性图示。在图16A-D中,事件E1是监控的安全事件。E1随时间的当前安全状态可以表示为沿着显示视图上的时间轴的运行线(running line)的形状,其中每个形状表示不同的状态(如图16A所描绘的)。在图表1600中,圆圈表示正常状态,正方形表示注意状态,三角形表示危险状态。替代地,可以基于发生变化的时间点来表示E1安全状态随时间的变化。例如,在图16B中,图表1610用“0”来描绘稳定状态(或没有状态改变),用“-”描绘安全状态的减少以及用“+”描绘安全状态的增加。应该理解的是,安全状态和其中的变化可以以任何类型的数值或图形形式来表示。图16C示出了将状态显示为数字的图表1620,其中从0起的每个负增量表示额外的状态降级。如果需要,可以表示安全状态的变化程度。例如,图表的y轴可以指示正常(normal)的降级的范围,并且随时间的安全状态可以看起来像线图或图16D的点状条形图1630。另外,监控的事件的彩色运行点线的单个线可以缓慢地从一种颜色改变/渐变为另一种颜色,从而表示安全状态的恶化和/或改善。

[0125] 以上示例不旨在是限制性的,并且可以显示数字、符号、颜色、图形和/或线的任何组合以使工程师能够对监控的事件的安全级别进行评估。此外,如果需要,可以储存各种事件的状态和/或状态的变化以用于后处理。

[0126] 在一实施例中,可以协调地(in concert)和/或一致地显示一个或多个图形。期望的监控的事件的分组可以靠近显示—例如,按照工厂区域、按照功能、按照对某些状况或因素的敏感性(例如,在批次过程的某些阶段期间)等等。工程师能够掩蔽显示视图,以一目了然能够查看感兴趣的特定安全事件。

[0127] 此外,可以为更抽象的安全事件提供可视化。如上所讨论的,监控的安全事件可以是一组监控的事件的抽象,诸如监控块和结果块。

[0128] 例如,参考图5和6,到结果块EB1和EB2中的使能器或直接输入可以是监控的安全事件和/或结果块EB1和EB2的每个结果E1、E2等可以是监控的安全事件。构成每个期望的监控事件的每个状况或输入可以对其状态的变化程度作出贡献。例如,如果监控的事件需要四个状况以便触发该事件,则当存在一个状况时,该监控的事件的状态可以是“-1”,当存在两个状况时,该监控的事件的状态可以是“-2”,当存在三个状况时,该监控的事件的状态可以是“-3”,当所有四个状况都存在时,状态可以是“X”或“触动”。因此,作为示例,图16A-图

16D的符号、数字、点等可以表示结果(或原因)的不同可能状态,或者可以表示在需要被设置为真状态或处于真状态以触动或启动结果信号的原因总数之中的被设置为真状态或处于真状态的原因的数量。

[0129] 此外,在特定安全状态或状态改变指示符上的点击或其它用户指示可以将用户自动链接到对应状况的细节。如上所述,工程师可以从安全事件可视化图访问SRS、系统配置和/或CEM显示。例如,并参考使监控的事件触动所需的四个状况的上述示例,如果安全可视化对于图表1620的上述示例性监控的事件指示“-1”并且用户点击“-1”,则可以显示包括导致对应于“-1”的安全状态的状况的设备或某件装备的系统配置的显示视图。

[0130] 图17描绘了使过程工厂内的安全事件可视化的示例性方法1700的框图。方法1700可以由服务器或者任何类型的电子设备来容易实现,其中服务器可以配备有或连接到被配置为显示内容的用户界面。

[0131] 方法1700可以开始于服务器访问(框1710)具有原因集合和结果集合的CEM。在实施例中,原因集合中的每一个原因可以表示过程工厂内的状况,并且结果集合中的每一个结果可以表示要在过程工厂内执行的结果。此外,原因集合和结果集合中的至少一些可以是作为因果对而相关的,由此对应的结果可以响应于对应的状况的发生而激活,并且原因集合和结果集合可以表示过程工厂内的监控的安全事件集合。

[0132] 服务器可以经由用户界面接收(框1720)对监控的安全事件集合中的监控的安全事件的选择。此外,服务器可以在用户界面中显示(框1730)对监控的安全事件的指示和监控的安全事件的当前状态。在实施例中,服务器可以将当前状态显示为一个或多个第一图形对象。

[0133] 服务器可以检测(框1740)针对监控的安全事件的状态变化。在一实施例中,服务器可以响应于时间段期满而检测状态的变化。根据状态的变化,服务器还可以在用户界面中显示(框1750)监控的安全事件的更新状态。在一实施例中,服务器可以将更新状态显示为可以不同于一个或多个第一图形对象的一个或多个第二图形对象。此外,在实施例中,服务器可以确定监控的安全事件的当前状态与更新状态之间的变化程度,并且可以在用户界面中显示变化程度。

[0134] 在一实施例中,服务器还可以响应于时间段期满,确定监控的安全事件的更新状态没有改变,并且可以在用户界面中显示监控的安全事件的更新状态。另外地或替代地,服务器可以经由用户界面接收对监控的安全事件的更新状态的选择,其中该监控的安全事件可以具有相关联的状况集合,并且可以在用户界面中显示针对相关联的状况集合中的每一个状况的状况状态。另外地或替代地,服务器可以经由用户界面接收对监控的安全事件的更新状态的选择,其中该监控的安全事件可以具有存在的相关联的状况,并且可以在用户界面中显示对过程设备内引起相关联的状况存在的设备的指示。

[0135] 另外地或替代地,服务器可以在存储器中储存表示监控的安全事件的数据、监控的安全事件的当前状态以及监控的安全事件的更新状态。此外,另外地或替代地,服务器可以在用户界面中显示(i)对监控的安全事件集合的另外监控的安全事件的另外指示,以及(ii)另外监控的安全事件的另外当前状态,检测针对另外监控的安全事件的另外状态变化,并且根据另外状态变化,在用户界面中显示另外监控的安全事件的另外更新状态。

[0136] 上面提供的示例性CEM是旨在用于说明目的的简化表示。图18示出了作为更复杂

的CEM的示例性CEM 1800,其包括时间延迟的触发、许可的触发、立即的触发和重置触发。CEM 1800是CEM的更本质的示例,其是对现实世界CEM的更准确表示。在如图18所示的CEM 1800中,仅包含“X”的单元可以表示立即的触发结果。此外,仅填充“R”的任何单元可以表示如果接收到原因,则结果将被触发以重置。以字母“T”开头的CEM 1800的单元可以指示原因直接触发结果,但是具有时间延迟。时间延迟可以以预定的增量进行设置。例如,“T1”可对应于10秒的时间延迟,“T2”可对应于20秒的时间延迟等。

[0137] CEM 1800还包含仅包含数字的单元,由此这些单元可以对应于“使能器”。特别地,单元中的数字标识使能器所属于的群组,其中对于每个群组可以有一个或多个使能器。CEM 1800中以数字开头并具有其它字符的单元可以表示只有在如果使能器也被触发的情况下才触发结果的关系。在一些实施例中,与对应使能器(或多个使能器)相关的每个单元必须“开启”以便结果被触发。在其它实施例中,特定使能器群组中的原因的任何组合可以进行组合以触发结果。类似地,可能需要使能器的任何组合来触发结果。

[0138] 例如,在如图18所示的CEM 1800中,原因1801是群组1的使能器。因此,如果原因1802被触动,则对应的结果可能不被触发,除非原因1801也被触动。该因果关系被认为是许可关系,因为只有当使能器“开启”时才触发结果。继续该示例,单元1803指示:因果关系属于使能器群组1,并且当原因信号1802变高并且使能器原因信号1801也为高或开启时,将以对应于长度T1的时间延迟来触发结果1805。单元1804指示:因果关系是许可的并属于使能器群组1,并且如果原因信号1802被使能(即,原因信号1801开启)和触动,则将立即触发结果1806。

[0139] 可以在上述所有先前的方法700、1000、1200、1400、1500和1700中实现CEM 1800。此外,本文描述的用于实现CEM或CEM的逻辑的监控块和结果块可以用于实现例如CEM 1800的复杂逻辑功能和相关逻辑功能,或者其它CEM中的任何其它逻辑功能。尽管增加了CEM1800的复杂度,管理CEM的益处仍然适用。此外,示例性CEM并非旨在是限制性的,并且可以结合用于实现过程工厂中的安全逻辑的因果矩阵的任何未来实施例来实现上述方法700、1000、1200、1400、1500和1700中的任何方法。

[0140] 图7、图10、图12、图14、图15和图17中的方法700、1000、1200、1400、1500和1700中的每一个可以通过软件、固件或硬件或者软件、固件、软件和/或硬件的某些组合来实现。另外,虽然图7、图10、图12、图14、图15和图17的流程图被描述为例程,但是这些流程图可以通过软件、硬件、固件或者软件、固件和/或硬件的组合来实现。

[0141] 用户界面(诸如上述用户界面)的实施例可以整体地或部分地由例如根据软件程序配置的处理器来实现。例如,工作站18a或20a或某种其它计算机可以整体地或部分地实现上述用户界面。用于实现用户界面的实施例的软件程序可以体现在储存在诸如硬盘、RAM、以电池为后备电源的RAM、ROM、CD-ROM、PROM、EPROM、EEPROM、DVD、闪存等的有形介质上的软件中、或在与处理器相关联的存储器(诸如RAM)中,但是本领域普通技术人员将容易地理解,整个程序或其部分可以替代地由不同于处理器的设备执行,和/或以公知的方式体现在固件和/或专用硬件中。

[0142] 尽管本发明容许各种修改和替代构造,但是其某些说明性实施例已经在附图中示出并且在本文中进行了详细描述。然而,应该理解的是,没有意图将本公开内容限制为所公开的具体形式,而是相反,意图是覆盖落入如由所附权利要求限定的本公开内容的精神和

范围内的所有修改、替代构造和等同物。

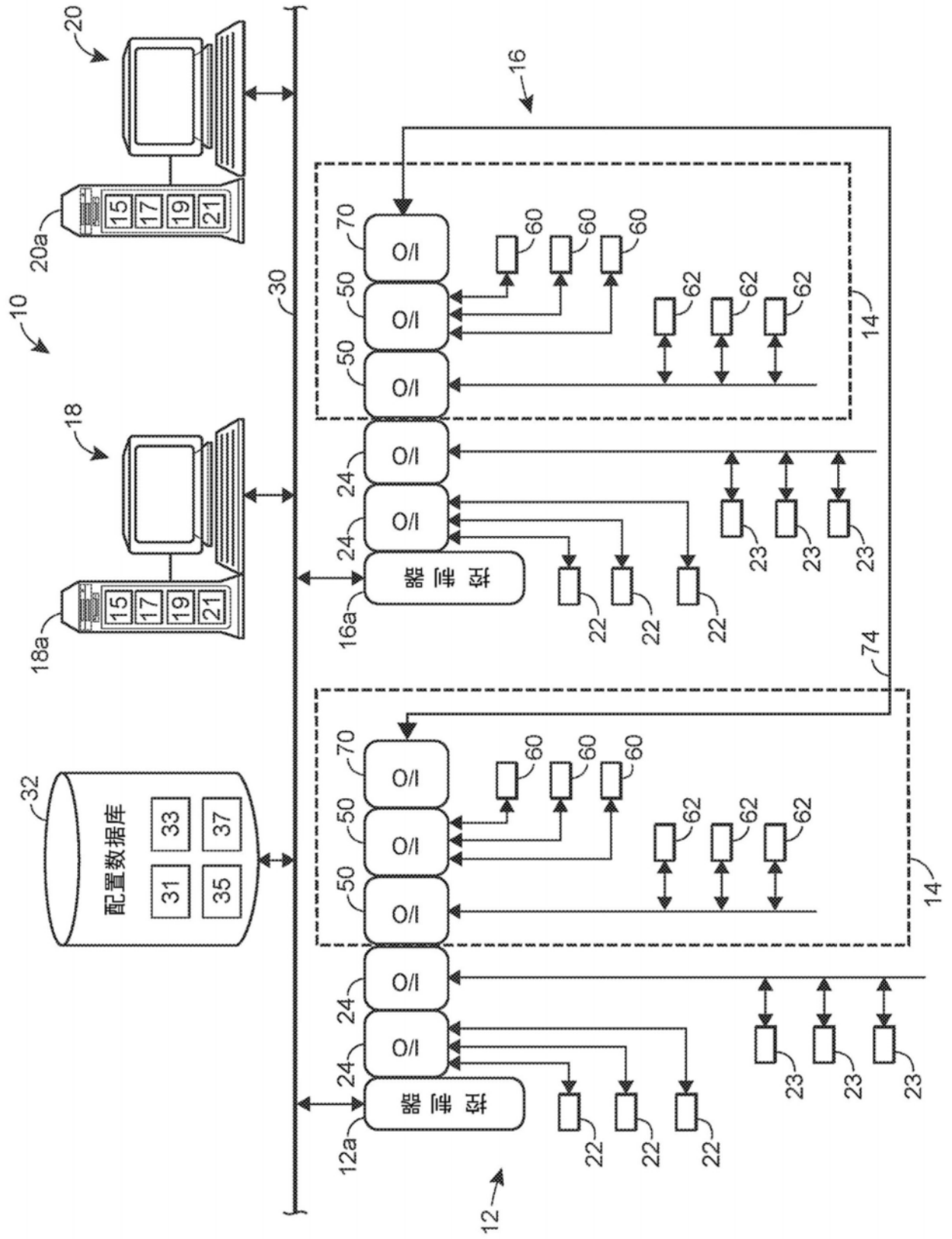

20
20a
15
17
19
21
10
30
16
18
18a
32
配置数据库
33
37
31
35
70
50
I/O
60
62
14
24
23
22
控制器
16a
74
12a
12

图1

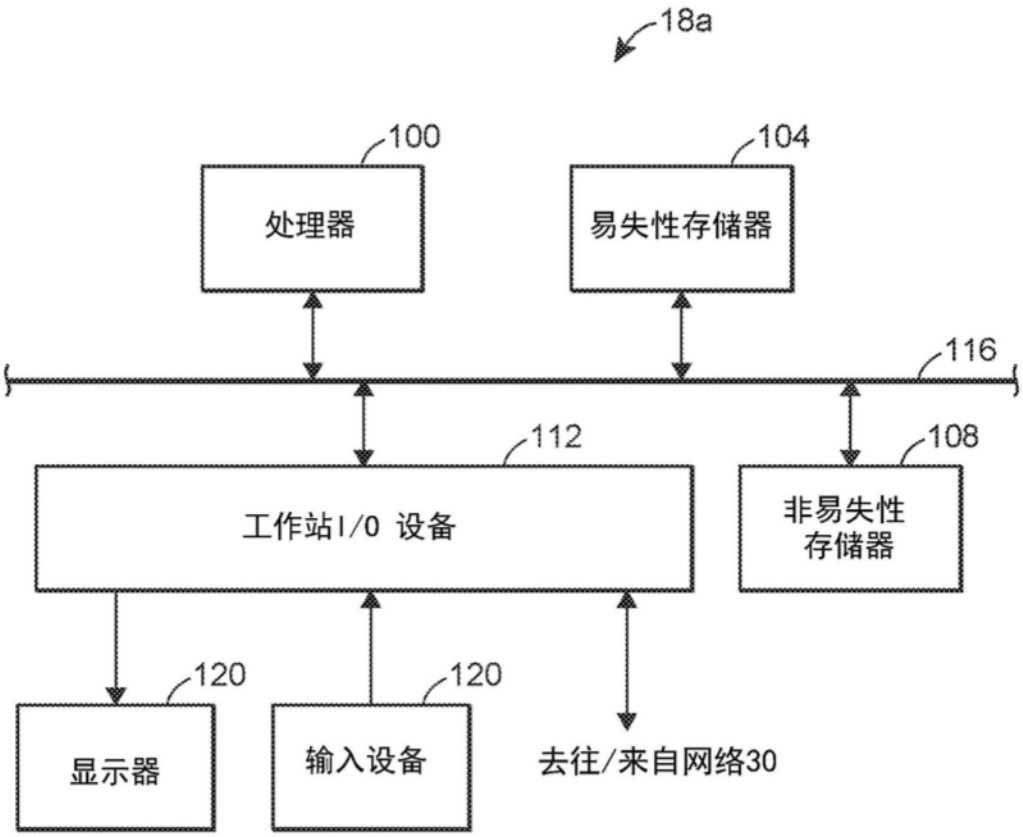

18a
100
处理器
104
易失性存储器
116
112
工作站I/O 设备
108
非易失性
存储器
120
显示器
120
输入设备
去往/来自网络30

图2

	结果1	结果2	结果3	结果4	结果5	结果6	结果7
原因1							
原因2			X	X	X		
原因3				X			
原因4				X	X	X	
原因5				X		X	
原因6			X				
原因7							

305

310

图3

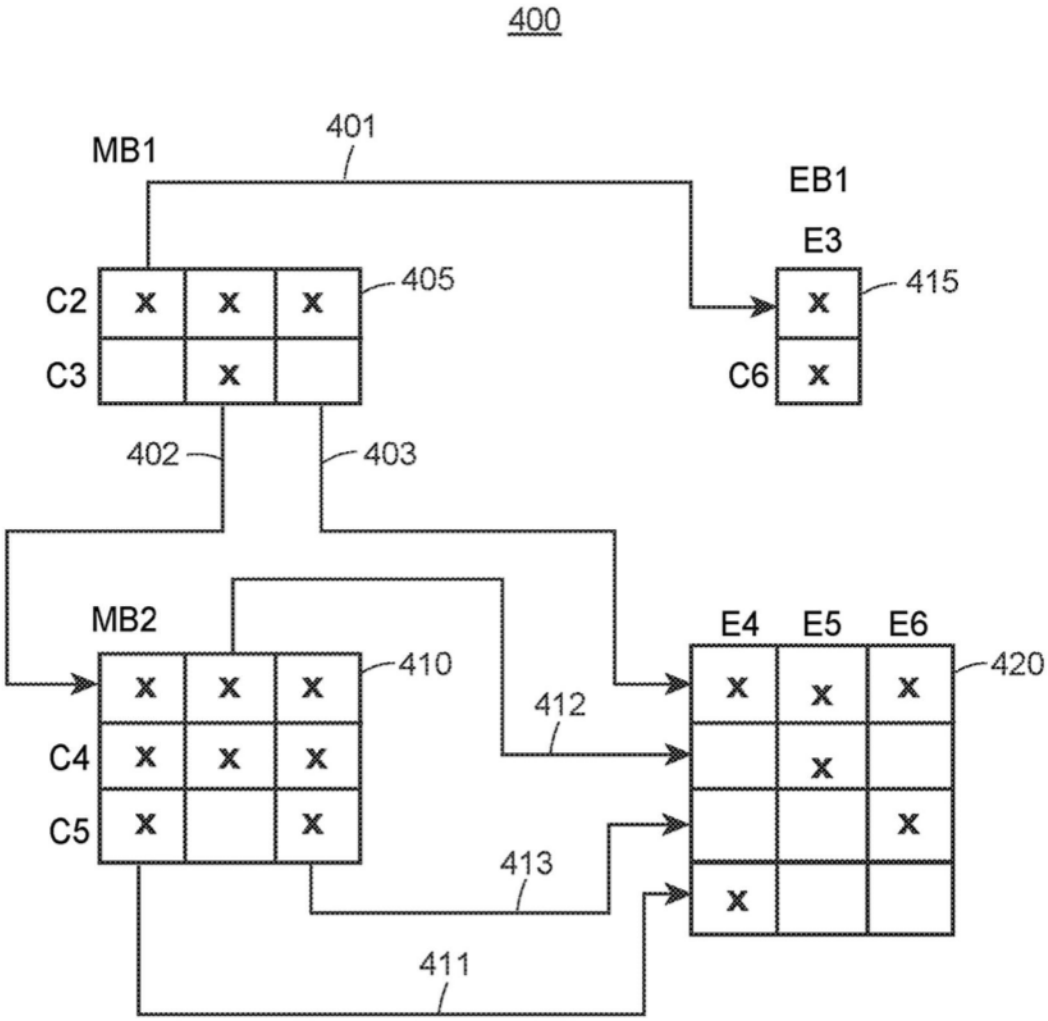

400
MB1
401
EB1
E3
C2
405
415
C3
C6
402
403
MB2
E4
E5
E6
410
420
412
C4
C5
413
411

图4

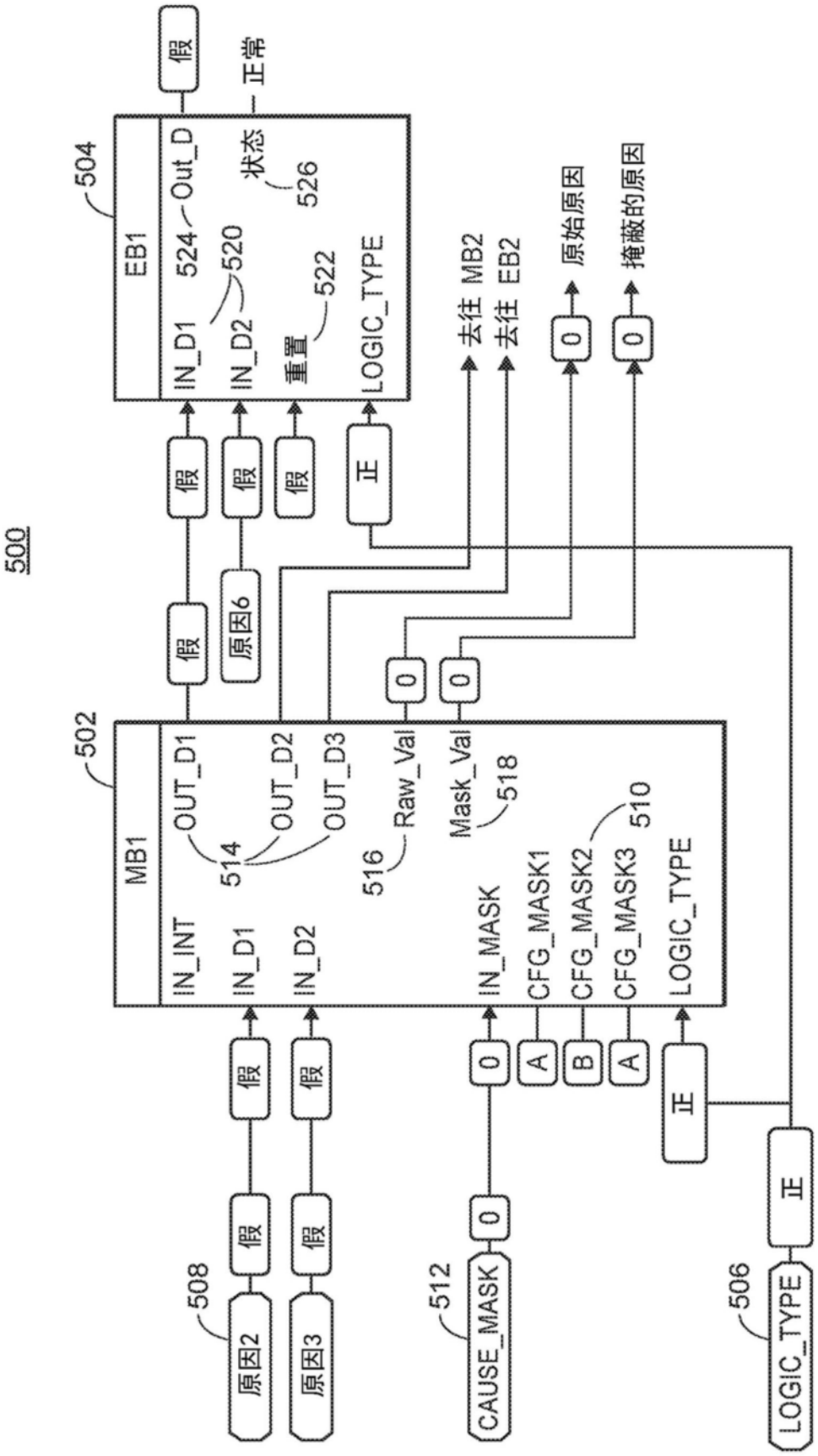

500
502
MB1
IN_INT
IN_D1
IN_D2
OUT_D1
OUT_D2
OUT_D3
Raw_Val
Mask_Val
IN_MASK
CFG_MASK1
CFG_MASK2
CFG_MASK3
LOGIC_TYPE
514
516
518
510
504
EB1
IN_D1
IN_D2
重置
LOGIC_TYPE
Out_D
状态
524
520
522
526
正常
假
正
0
A
B
原因6
原因2
原因3
CAUSE_MASK
LOGIC_TYPE
508
512
506
去往 MB2
去往 EB2
原始原因
掩蔽的原因

图5

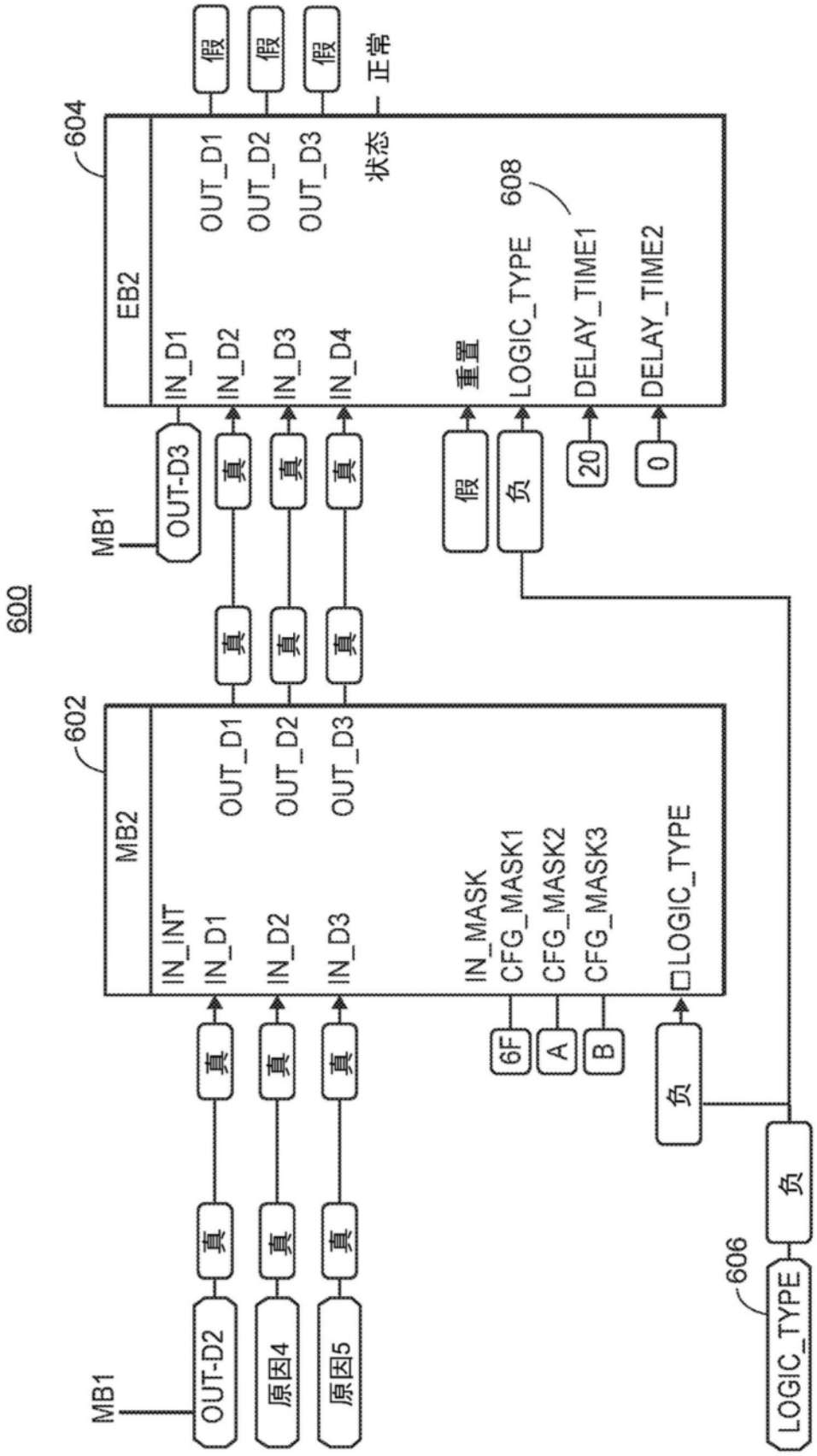

600
602
MB2
IN_INT
IN_D1
IN_D2
IN_D3
OUT_D1
OUT_D2
OUT_D3
IN_MASK
CFG_MASK1
CFG_MASK2
CFG_MASK3
LOGIC_TYPE
604
EB2
IN_D1
IN_D2
IN_D3
IN_D4
OUT_D1
OUT_D2
OUT_D3
状态
重置
LOGIC_TYPE
608
DELAY_TIME1
DELAY_TIME2
MB1
OUT-D2
原因4
原因5
OUT-D3
606
LOGIC_TYPE
真
假
负
6F
A
B
20
0
正常

图6

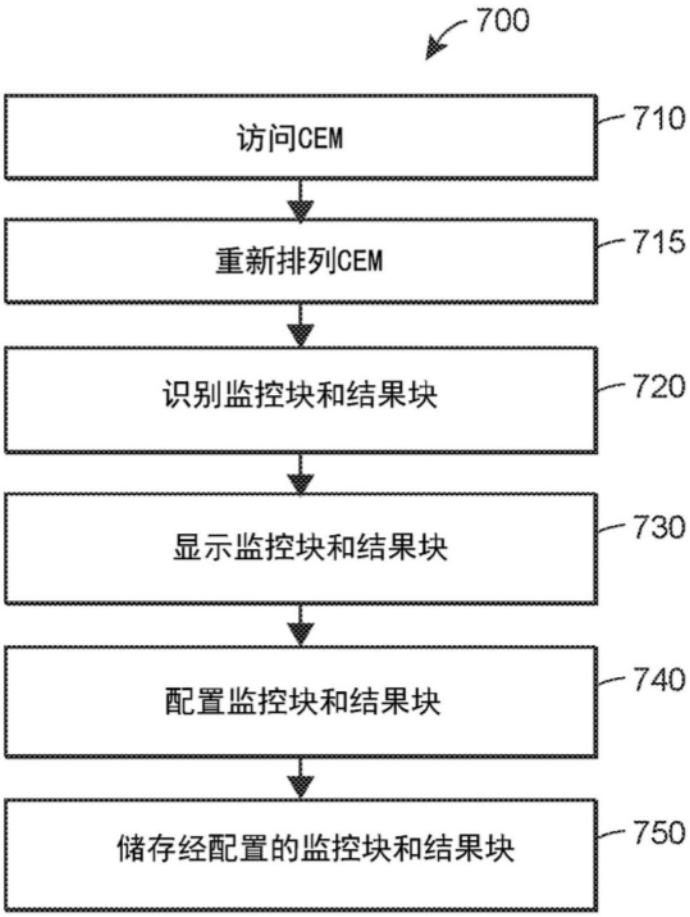

700
访问CEM
710
重新排列CEM
715
识别监控块和结果块
720
显示监控块和结果块
730
配置监控块和结果块
740
储存经配置的监控块和结果块
750

图7

800

	结果1	结果2	结果3	结果4	结果5	结果6	结果7	结果8	结果9	结果10	结果11	结果12	结果13	结果14	结果15	结果16	结果17
原因1	X	X											X	X			
原因2	X	X											X	X			
原因3	X	X											X	X			
原因4			X	X	X								X	X			
原因5			X	X	X								X	X			
原因6			X	X	X												
原因7			X	X	X												
原因8			X	X	X	X	X	X	X								X
原因9			X	X	X	X	X	X	X								X
原因10			X	X	X	X	X	X	X								X
原因11	X	X															
原因12																X	X
原因13																	
原因14													X	X			
原因15													X	X			
原因16						X	X	X	X				X	X			

图8

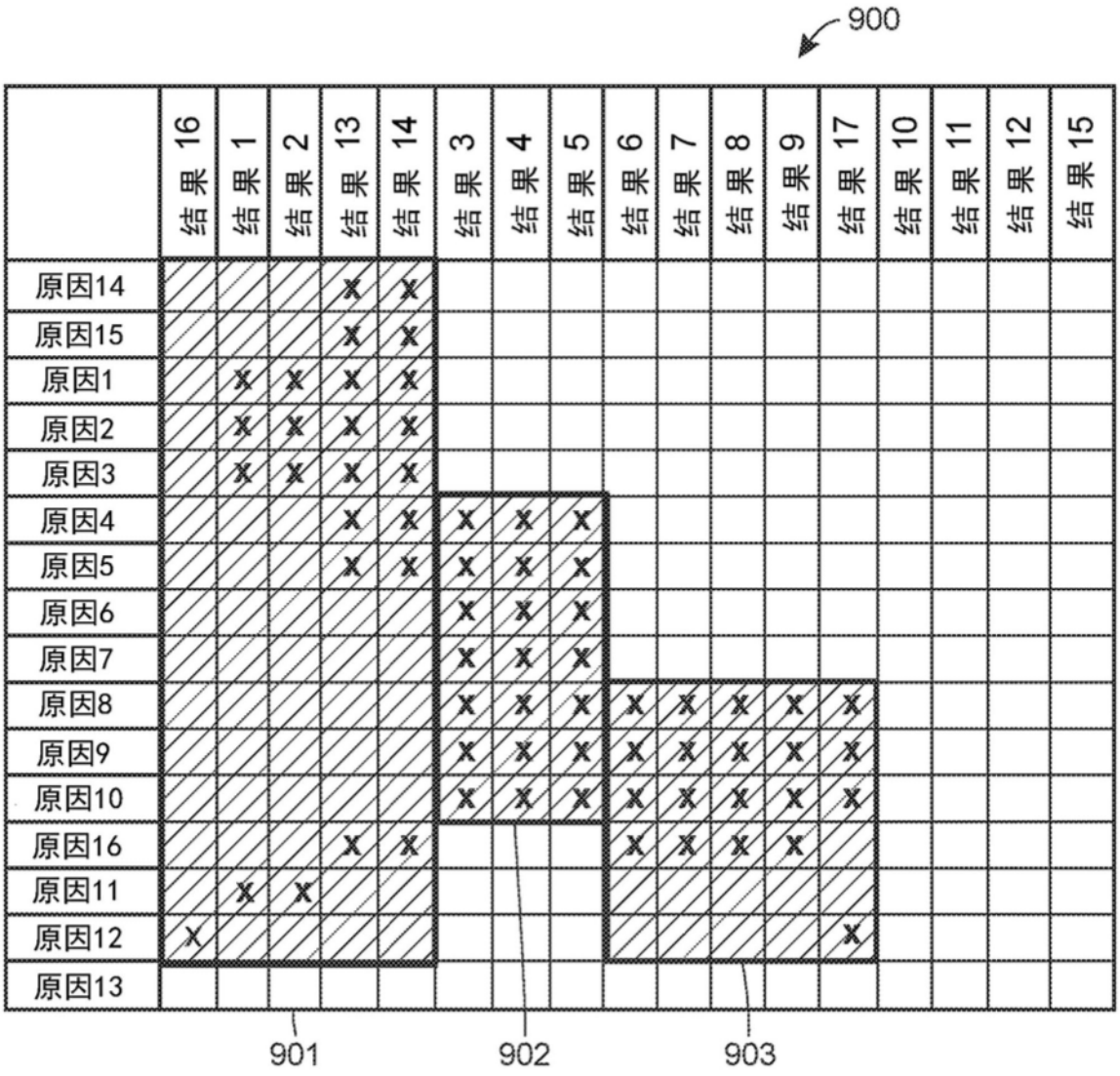

900
结果16
结果1
结果2
结果13
结果14
结果3
结果4
结果5
结果6
结果7
结果8
结果9
结果17
结果10
结果11
结果12
结果15
原因14
原因15
原因1
原因2
原因3
原因4
原因5
原因6
原因7
原因8
原因9
原因10
原因16
原因11
原因12
原因13
901
902
903

图9

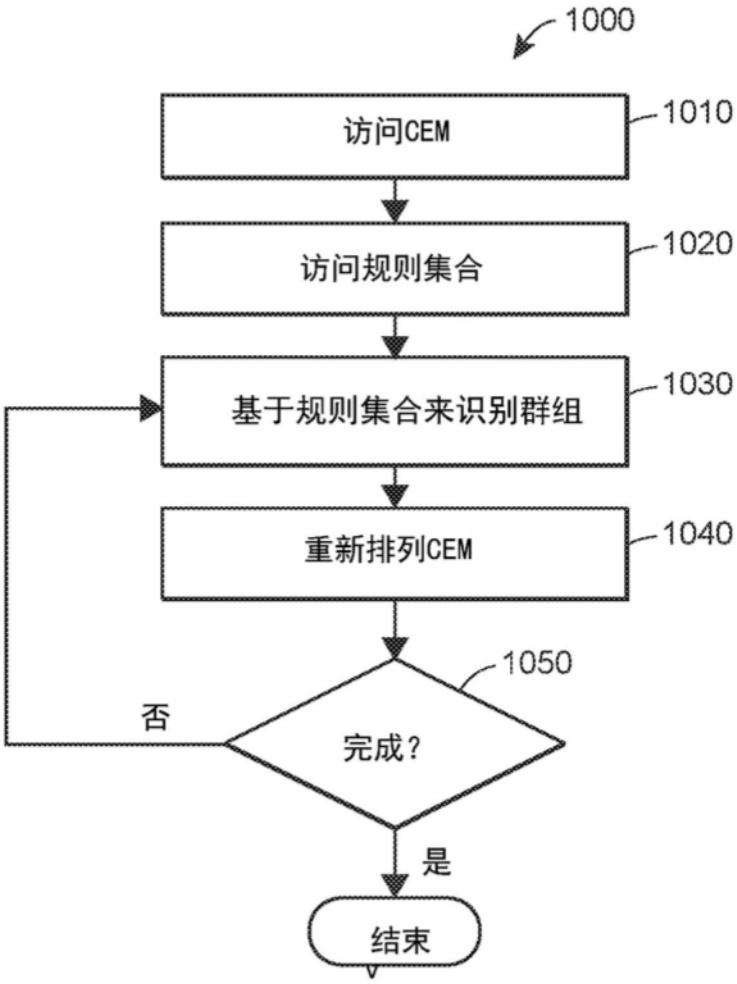
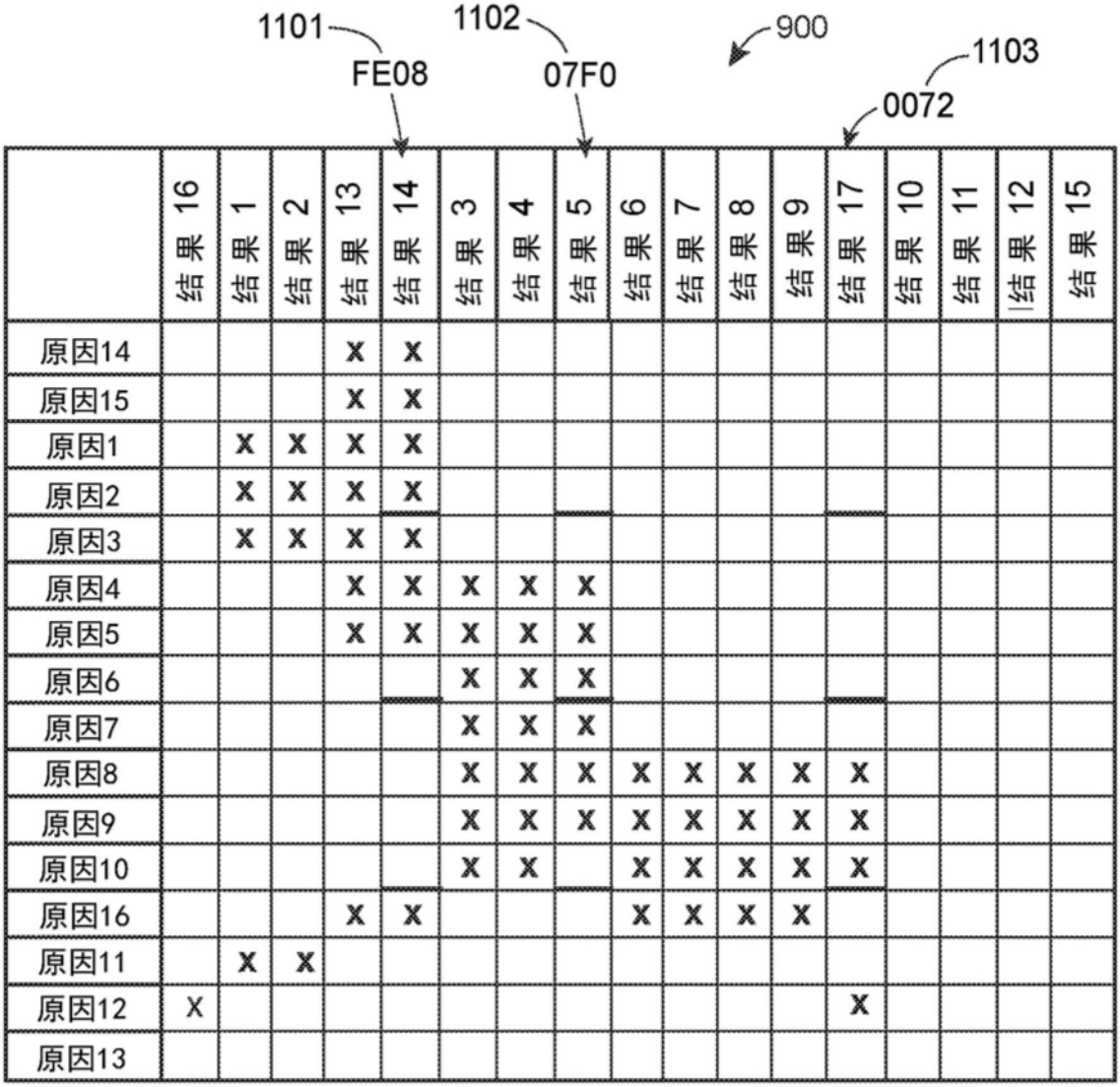

1000
访问CEM
1010
访问规则集合
1020
基于规则集合来识别群组
1030
重新排列CEM
1040
1050
完成?
否
是
结束

图10



	结果16	结果1	结果2	结果13	结果14	结果3	结果4	结果5	结果6	结果7	结果8	结果9	结果17	结果10	结果11	结果12	结果15
原因14				X	X												
原因15				X	X												
原因1		X	X	X	X												
原因2		X	X	X	X												
原因3		X	X	X	X												
原因4				X	X	X	X	X									
原因5				X	X	X	X	X									
原因6						X	X	X									
原因7						X	X	X									
原因8						X	X	X	X	X	X	X	X				
原因9						X	X	X	X	X	X	X	X				
原因10						X	X		X	X	X	X	X				
原因16				X	X				X	X	X	X					
原因11		X	X														
原因12	X												X				
原因13																	

图11

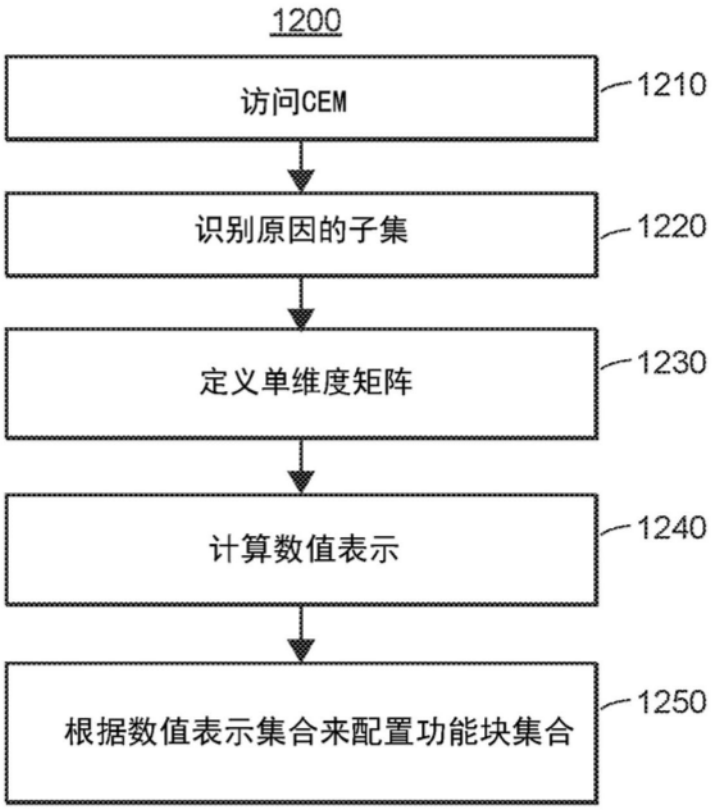

1200
访问CEM
1210
识别原因的子集
1220
定义单维度矩阵
1230
计算数值表示
1240
根据数值表示集合来配置功能块集合
1250

图12

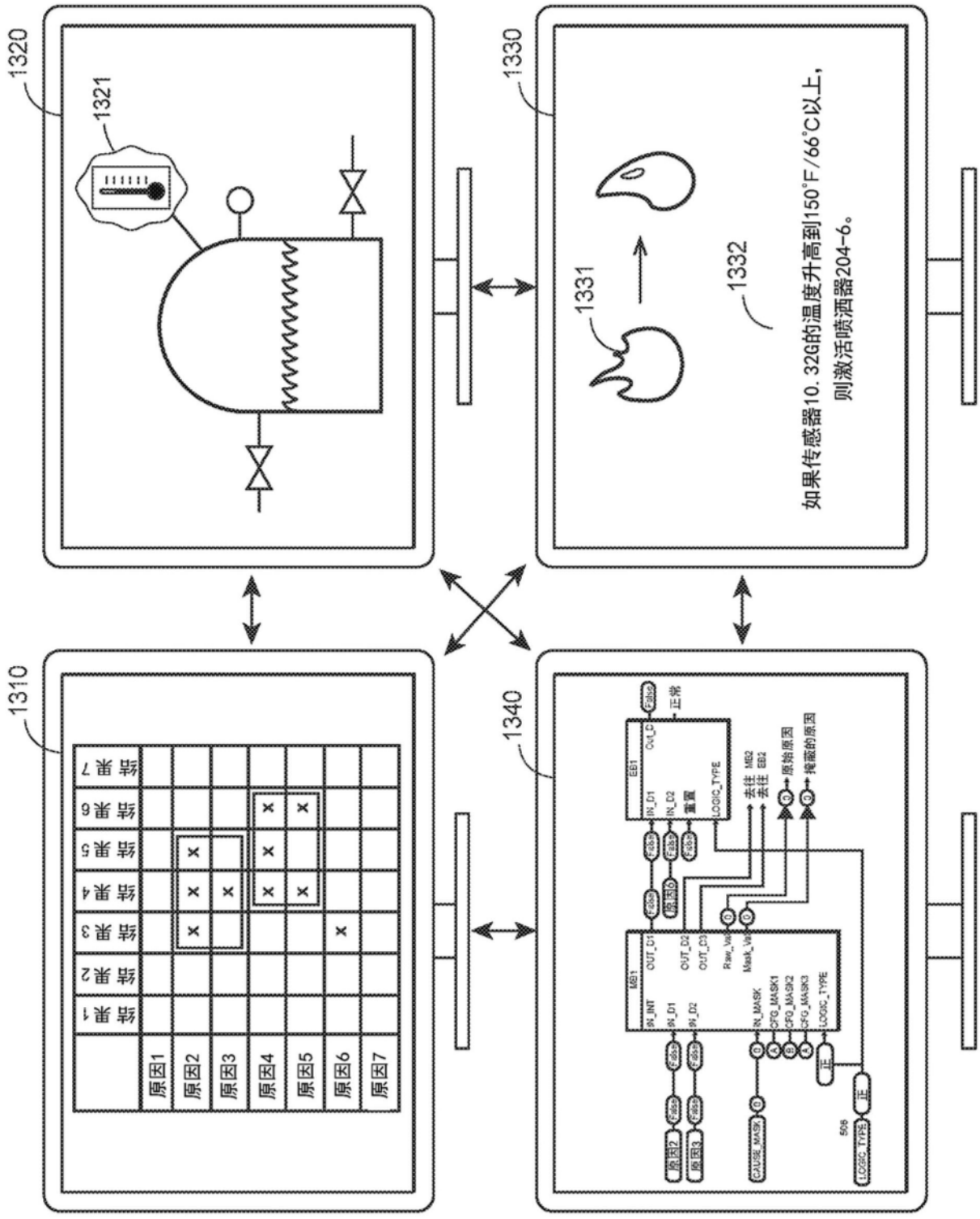

1320
1321
1330
1331
1332
如果传感器10.32G的温度升高到150°F/66°C以上，
则激活喷洒器204-6。
1310
结果1
结果2
结果3
结果4
结果5
结果6
结果7
原因1
原因2
原因3
原因4
原因5
原因6
原因7
1340
正常
去往 MB2
去往 EB2
原始原因
掩蔽的原因
原因2
原因3
508

图13

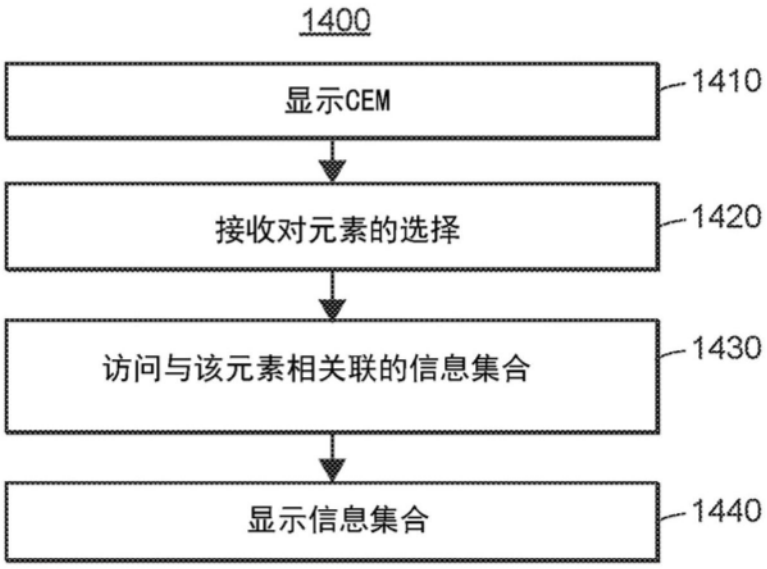

1400
显示CEM
1410
接收对元素的选择
1420
访问与该元素相关联的信息集合
1430
显示信息集合
1440

图14

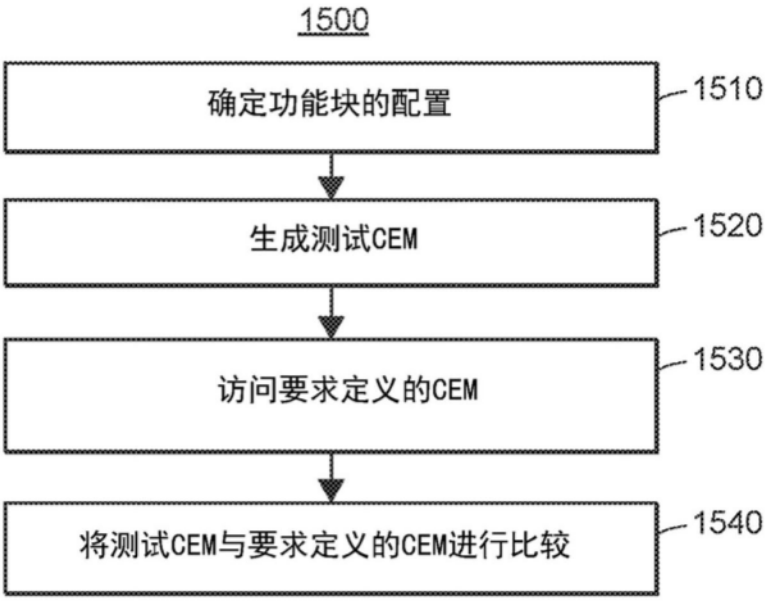

1500
确定功能块的配置
1510
生成测试CEM
1520
访问要求定义的CEM
1530
将测试CEM与要求定义的CEM进行比较
1540

图15

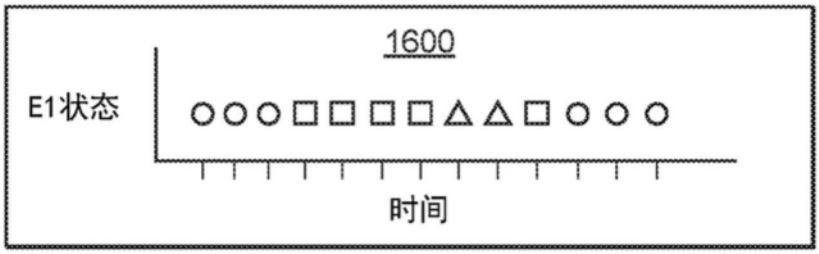

1600
E1状态
时间

图16A

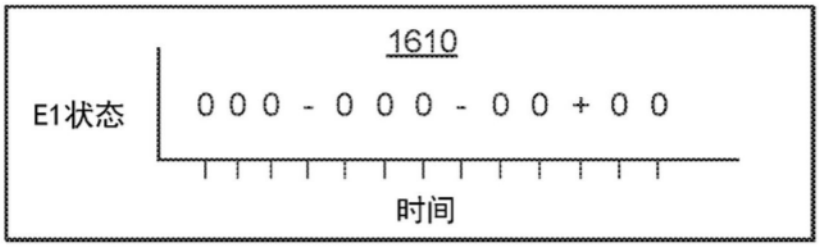


图16B

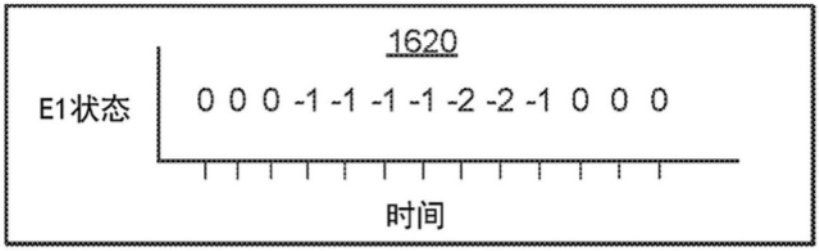


图16C

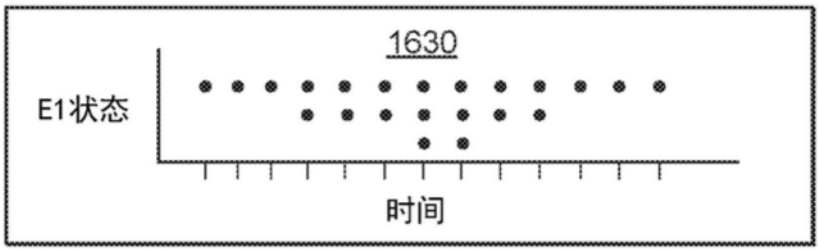


图16D

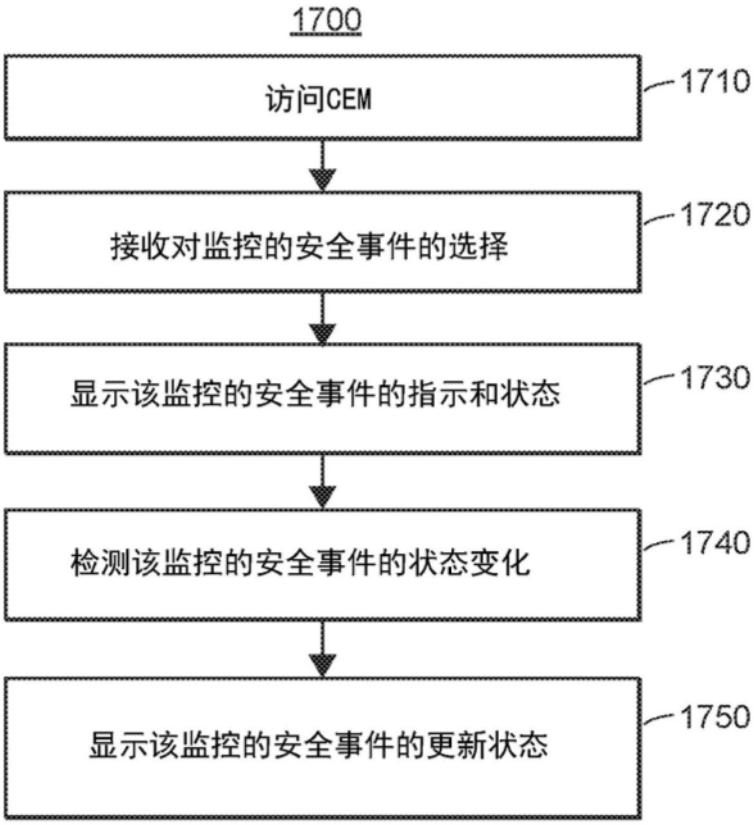

1700
访问CEM
1710
接收对监控的安全事件的选择
1720
显示该监控的安全事件的指示和状态
1730
检测该监控的安全事件的状态变化
1740
显示该监控的安全事件的更新状态
1750

图17

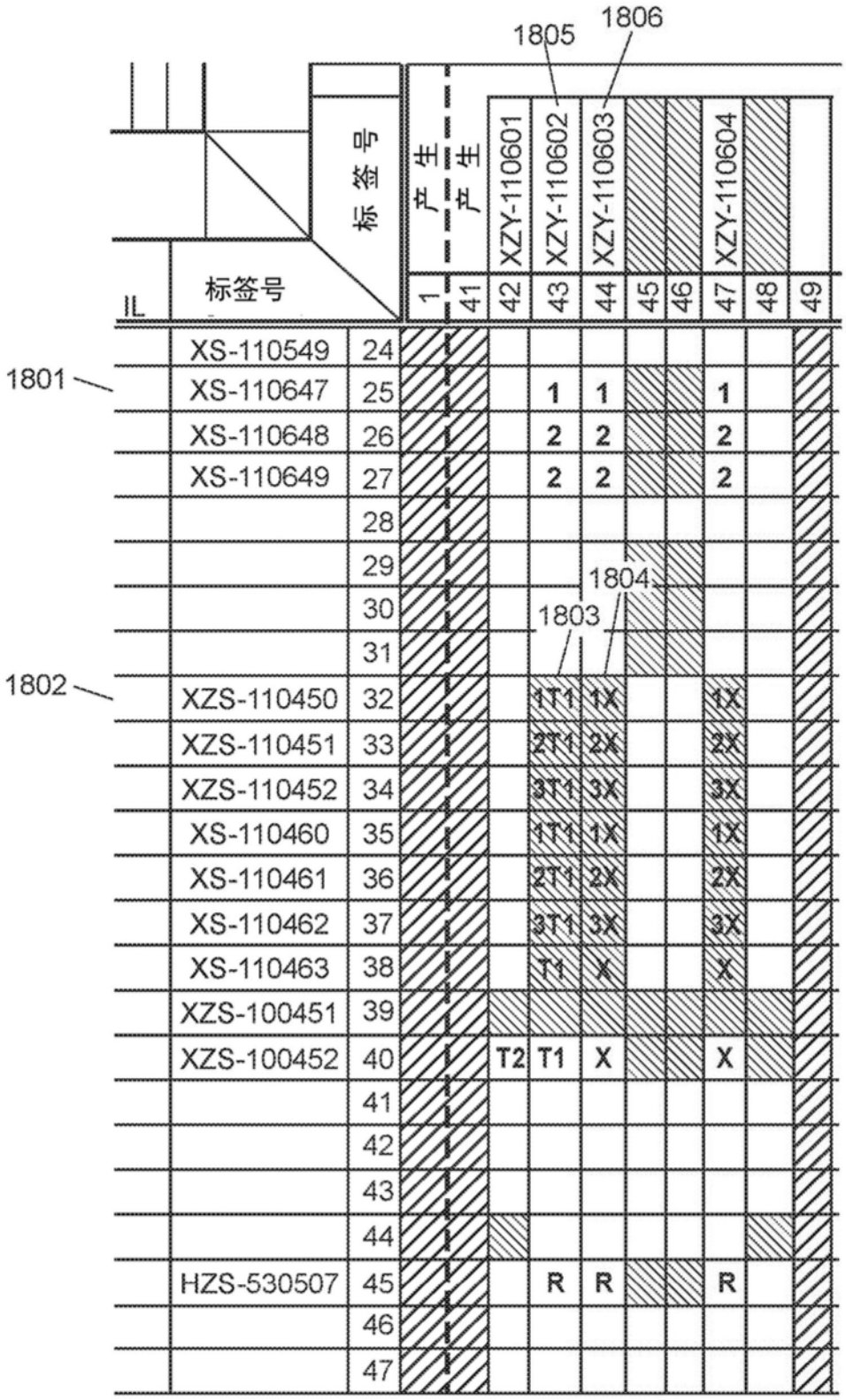

1805
1806
标签号
产生
产生
XZY-110601
XZY-110602
XZY-110603
XZY-110604
IL
标签号
1
41
42
43
44
45
46
47
48
49
1801
1802
1803
1804
XS-110549 24
XS-110647 25 1 1 1
XS-110648 26 2 2 2
XS-110649 27 2 2 2
28
29
30
31
XZS-110450 32 1T1 1X 1X
XZS-110451 33 2T1 2X 2X
XZS-110452 34 3T1 3X 3X
XS-110460 35 1T1 1X 1X
XS-110461 36 2T1 2X 2X
XS-110462 37 3T1 3X 3X
XS-110463 38 T1 X X
XZS-100451 39
XZS-100452 40 T2 T1 X X
41
42
43
44
HZS-530507 45 R R R
46
47

图18