



(12)发明专利

(10)授权公告号 CN 103154837 B

(45)授权公告日 2017.06.16

(21)申请号 201180048858.X

(72)发明人 E.埃哈特 W.格里斯鲍姆

(22)申请日 2011.08.10

(74)专利代理机构 北京市柳沈律师事务所
11105

(65)同一申请的已公布的文献号

申请公布号 CN 103154837 A

代理人 谢强

(43)申请公布日 2013.06.12

(51)Int.Cl.

G05B 19/042(2006.01)

(30)优先权数据

G05B 9/03(2006.01)

102010039607.9 2010.08.20 DE

(85)PCT国际申请进入国家阶段日

(56)对比文件

US 7043728 B1, 2006.05.09,

2013.04.09

US 20060247796 A, 2006.11.02,

(86)PCT国际申请的申请数据

CN 1879068 A, 2006.12.13,

PCT/EP2011/063753 2011.08.10

CN 1228173 A, 1999.09.08,

(87)PCT国际申请的公布数据

审查员 王波

W02012/022661 DE 2012.02.23

(73)专利权人 西门子公司

权利要求书1页 说明书3页 附图1页

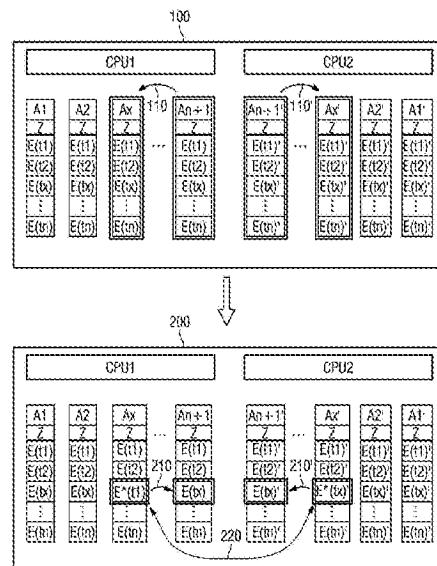
地址 德国慕尼黑

(54)发明名称

用于自动化系统的过程冗余控制的方法

(57)摘要

本发明涉及一种用于具有至少两个控制器(CPU1,CPU2)的自动化系统的过程冗余控制的方法,其中,每个控制器(CPU1,CPU2)依次执行n个任务区块(t₁,t₂,t_x,...,t_n),其中用于任务区块的执行的可传输的输出数据(E(t₁),E(t₂),E(t_x),...,E(t_n))储存在比任务区块的个数超过一个的工作区域(A₁,A₂,A_x,...,A_{n+1},A_{1'},A_{2'},A_{x'},...,A_{n+1'})中,所述工作区域分别容纳了每个任务区块的输出数据,其中,额外多出的一个工作区域(A_{n+1},A_{n+1'})作为系统工作区域容纳当前可传输的输出数据。一个非常简单而且可靠的用于同步数据管理和冗余控制器控制的方法可以如下实现,在一个冗余控制器内的任务区块开始的时候,分别把之前同步的内容从系统工作区域传输到工作区域,然后在任务区块被执行的时候更新所述内容,如果被更新的内容在冗余控制器内一致,则在下一个任务区块开始之前再次接受所述被更新内容到系统工作区域。



1. 一种用于具有至少两个控制器的自动化系统的过程冗余控制的方法,其中,每个控制器依次执行数个(n)任务区块,其中,用于任务区块的执行的可传输的输出数据储存在比任务区块的个数超过一个的工作区域,所述工作区域分别容纳了每个任务区块的输出数据,并且其中,额外多出的一个工作区域作为系统工作区域容纳当前可传输的输出数据,并且为了执行任务区块在每个控制器中如下使用:

-在待执行的任务区块开始时(100),将系统工作区域的当前的内容传输(110,110')到工作区域,

-在被执行的任务区块结束时(200),将至少两个控制器的工作区域的随着被执行的任务区块的结果更新的输出数据相互比较(220),其中,在传输(110,110')之前和/或之后,对所述控制器的当前的系统工作区域的连续的传输计数器(Z,Z')进行比较,如果一个控制器中的系统工作区域和用于任务区块的工作区域之间或另一控制器中的系统工作区域和用于任务区块的工作区域之间出现传输计数器的偏差,或者两个CPU的工作区域之间出现传输计数器的偏差,则重复该进程,并且

-如果工作区域的用于控制器中的任务区块的内容相互一致,则将所述工作区域的被更新的内容接受(210,210')到系统工作区域以及开始下一个任务区块。

2. 根据权利要求1所述的方法,其特征在于,在被执行的任务区块结束(200)时,比较系统工作区域的一个连续的传输计数器(Z,Z')并且将其递增。

3. 根据权利要求1所述的方法,其特征在于,在一个中断封锁期间将被更新的工作区域的内容接受到系统工作区域。

4. 根据权利要求1所述的方法,其特征在于,如果至少两个控制器的随着被执行的任务区块的结果更新的输出数据相同,则所述内容被判断为一致。

5. 根据上述权利要求4所述的方法,其特征在于,如果所述输出数据的数字和相互一致,则所述输出数据相等。

6. 根据权利要求1至5中任一项所述的方法,其特征在于,作为被更新的工作区域的内容仅仅将所述被更新的输出数据接受到系统工作区域。

7. 根据权利要求1至5中任一项所述的方法,其特征在于,控制器的所述系统工作区域是中央存储的系统工作区域的拷贝,并且该中央存储的系统工作区域在下一个待执行的任务区块开始之前被控制器的系统工作区域的当前内容所替代。

用于自动化系统的过程冗余控制的方法

[0001] 本发明涉及一种用于自动化系统的过程冗余控制的方法。

[0002] 为了设备或者过程的可靠运行的冗余自动化系统是多方为人所公知的。在这种系统中控制器是被划分到两个或多个子系统中,这些子系统独立地并且同时执行单独的控制或者调节任务。在此,每个子系统都具有一个自己的控制器,即所谓的CPU,其作为计算单元负责对之前所规划的自动化功能的执行。这些功能以机器指令的形式对于CPU被划分为一系列任务区块—所谓的任务(Tasks),后者被所述控制器依次处理。

[0003] 如果出于可靠性原因特定的任务应该冗余地被多个子系统或CPU来执行,则所述任务必须被同步执行。否则可能在所述子系统中读取有分歧的(divergierende)数据并且由此在各个单独的任务区块完成或执行之后得出不同的结果。所述待控制的设备或者待控制的过程的可靠的运行由此也不能被确保。

[0004] 因此,本发明要解决的技术问题是,提供一种用于可靠的冗余自动化系统的方法。

[0005] 该技术问题是通过根据本发明的方法来解决的,也就是一种用于具有至少两个控制器的自动化系统的过程冗余控制的方法,其中,每个控制器依次执行数个任务区块,其中用于任务区块的执行的可传输的输出数据储存在比任务区块的个数超过一个的工作区域,所述工作区域分别包含了每个任务区块的输出数据,而那个额外的工作区域作为系统工作区域容纳当前可传输的输出数据,并且为了执行任务区块在每个控制器中如下使用:在待执行的任务区块开始时,将系统工作区域的当前的内容传输到工作区域,在被执行的任务区块结束时,将随着被执行的任务区块的结果更新的至少两个控制器的工作区域的输出数据相互比较,并且将所述工作区域的被更新的内容接受到系统工作区域,以及,如果工作区域的用于控制器中的任务区块的内容相互一致,则开始下一个任务区块。

[0006] 在冗余的控制器的任务区块开始时分别把之前同步的内容从系统工作区域传输到工作区域,然后在任务区块被执行的时候更新上述内容并且紧接着将此被更新的内容(如果其与在冗余控制器中一致)在下一个任务区块开始之前再次传输到系统工作区域,由此,得出一种非常简单而可靠的方法,用于在自动化系统中同步和一致地并因此无矛盾地数据维护和冗余控制。由此排除了异常的结果的传递以及从而基于异常的输出数据的控制的延续。比任务区块的个数超过一个的工作区域的数量和因此的一个额外的用于输出数据的传输和接受的系统工作区域的引入,实现了一种带有高度可支配的冗余性的自动化系统,其同时也是能够防止错误因而非常可靠的。根据本发明的方法最后还实现了,自动化功能性独立于系统功能性。用于自动化功能的任务可以基于当前的和一致的数据(该数据在系统中还永久可用),在任何时间与系统无关地开始。用于数据的一致性检测的额外测试程序不再被要求,而是已经在流程中没有时间延迟地绑定。因而它是一种非常简单的用于冗余控制的方法,由此减少了开发成本、测试成本和维护成本。

[0007] 尤其具有优势的是,根据本发明的方法用于多核系统(即,带有多个处理器的CPU)的应用。通过所述方法的应用,在一个核中的这些处理器上的并行和冗余的任务流程,实现了尤其高的处理速度和计算性能,因为否则常见的高的管理和协调开销被取消了。

[0008] 优选地,工作区域的被更新的内容会在每一个被执行的任务的结束时的中断封锁

(Interruptsperrre) 期间被接受到该系统工作区域。这意味着，每个任务执行仅仅还需要一个中断封锁，从而流程速度可以被最大化。

[0009] 在冗余控制器中的一个任务的执行之后根据各自内容的数字和 (Quersumme)，有利地进行各自工作区域的被更新的内容的比较。在此，该数字和比较可以例如按照公知的校验和比较的方法来实行。其可以不需要更大计算消耗地实行，从而带来了流程速度的最大化。

[0010] 特别有利的是，作为工作区域的被更新的内容仅仅将被更新的输出数据接受到系统工作区域，因为在此仅仅该任务的结果被接受而工作区域的所有其它内容保持不变。

[0011] 下面，要根据附图示例性地解释所述发明。图示的是在n个任务t₁, t₂, t_x到t_n中的一个单独的任务执行t_x的非常示意性的流程。在此每个任务代表一个带有用于待控制的自动化功能的控制指令或机器指令的任务区块。

[0012] 在所示的流程开始的时间点100，例如自动化系统的启动之后或前一个任务执行结束之后，开始任务t_x，紧接着执行并且在时间点200结束，然后必要时开始下一个任务。任务t_x的执行在此仅仅通过一个100和200之间的箭头表示。在该任务的执行期间，自动化系统的控制指令或机器指令以公知的方式被转换和执行，使得在此的任务区块的执行不必被更具体的展示和描述。对于此发明更为本质的是，创立一个额外的工作区域(即，所谓的系统工作区域)并将此应用于两个时间点100和200，也就是每个任务区块的执行的开始和结束，以便实现自动化功能的一种冗余和无错误并且由此是可靠的控制。

[0013] 在目前的实施例中提供有用于冗余自动化系统的过程控制的两个控制器CPU1和CPU2，其依次分别执行之前规划的n个任务区块t₁, t₂, t_x到t_n。输出数据E(t₁) 到E(t_n) 和E(t₁)' 到E(t_n)' 被指定到这n个任务区块，所述输出数据对于每个CPU被存储在A₁到A_n和A₁' 到A_n' 的n个工作区域。在这n个工作区域以外，在两个控制器CPU1和CPU2中分别设置有第(n+1)个工作区域作为所谓的系统工作区域A_{n+1}或A_{n+1}'，其包含有当前的可传输的输出数据并且被用于任务区块的执行，如同下面要结合任务区块t_x描述的那样。所述任务t_x在所有连接到冗余系统的CPU(在此为CPU1和CPU2)中同时开始。在此，在每个任务开始的时候，对于当前的任务t_x系统工作区域A_{n+1}或A_{n+1}'的全部内容会被拷贝到相应的工作区域A_x或A_x'，如同在附图中以附图标记110对于CPU1和附图标记110'对于CPU2所表明的那样。在此，通过系统工作区域的传输计数器Z的比较来确保拷贝时的数据一致性。该写入计数器比较在此在传输之前和/或之后进行。如果CPU1中的系统工作区域A_{n+1}和用于任务区块t_x的工作区域A_x之间或CPU2中的系统工作区域A_{n+1}'和用于任务区块t_x的工作区域A_x'之间出现传输计数器的偏差，或者两个CPU的工作区域之间出现传输计数器的偏差，则重复所述进程。当传输计数器相一致时，在工作区域A_x或A_x'中准备好当前的内容之后实行实际的任务流程，而独立于伙伴 (Partner) CPU，也即是说在执行期间不进行CPU的同步化和不带有唤醒报警信号闭塞地给出指令之间的 (Befehlsgranularer) 可中断性。尽管有多任务功能性和指令之间的可中断性，对于整个流程这导致了一个单任务系统(没有任务协调和没有同步化来用于以此为基础的功能性)。在任务结束时，通过整个任务的结果可以构成一个数字和以及准备好一个用于比较的伙伴分量，其在图示中以附图标记220表示。

[0014] 如果数字和关于内容相等，那么在一个中断封锁之内将任务结果E*(t_x) 或E*(t_x)' 拷贝(附图标记210和210')到CPU1和CPU2各自的系统工作区域A_{n+1}或A_{n+1}' 并且系

统工作区域An+1或An+1'中的连续计数器Z递增。紧接着可以开始下一个执行区块。由此,每个单独的任务可以以当前的和一致的数据在任意时间点同步开始于冗余控制器中。

[0015] 如果数字和不相同,那么可以考虑以下的做法:

[0016] a.)暂存所确定的数字和,重新开始执行任务tx并比较;

[0017] b.)如同在防止错误的系统中常见的那样,取消自动化过程并且使自动化系统处于安全状态;

[0018] c.)检查所规划好的任务,将期待相等的数字和与得出的不同的数字和相比较。

[0019] 本发明并不限于之前所描述的实施方式。而是也可以考虑对单个特征的组合、修改或补充,其可以带来其它有创造力的想法的可能的实施方式。例如,控制器CPU1和CPU2的系统工作区域An+1和An+1'可以表示一个中央存储的系统工作区域的拷贝,其中该中央存储的系统工作区域在下一个待执行的任务区块开始前被两个控制器的系统工作区域的当前内容所替代。

[0020] 对于根据本发明的方法的所有执行方式都重要的仅仅是,冗余控制器(假如如前所描述的是两个或者更多)之间的误差在每个执行区块的结束的时候的定位,以及,这样的错误立刻被识别从而不会带来错误结果在过程中的传递和对错误值的继续处理。由此,也可以识别RAM错误,其在传统的自动化系统的很长时间的连续运行中会个别地出现。此外,根据本发明的方法可以非常简单地实现对于冗余控制器的无冲击地接通,因为额外的控制器可以特别地按照任务被接通。

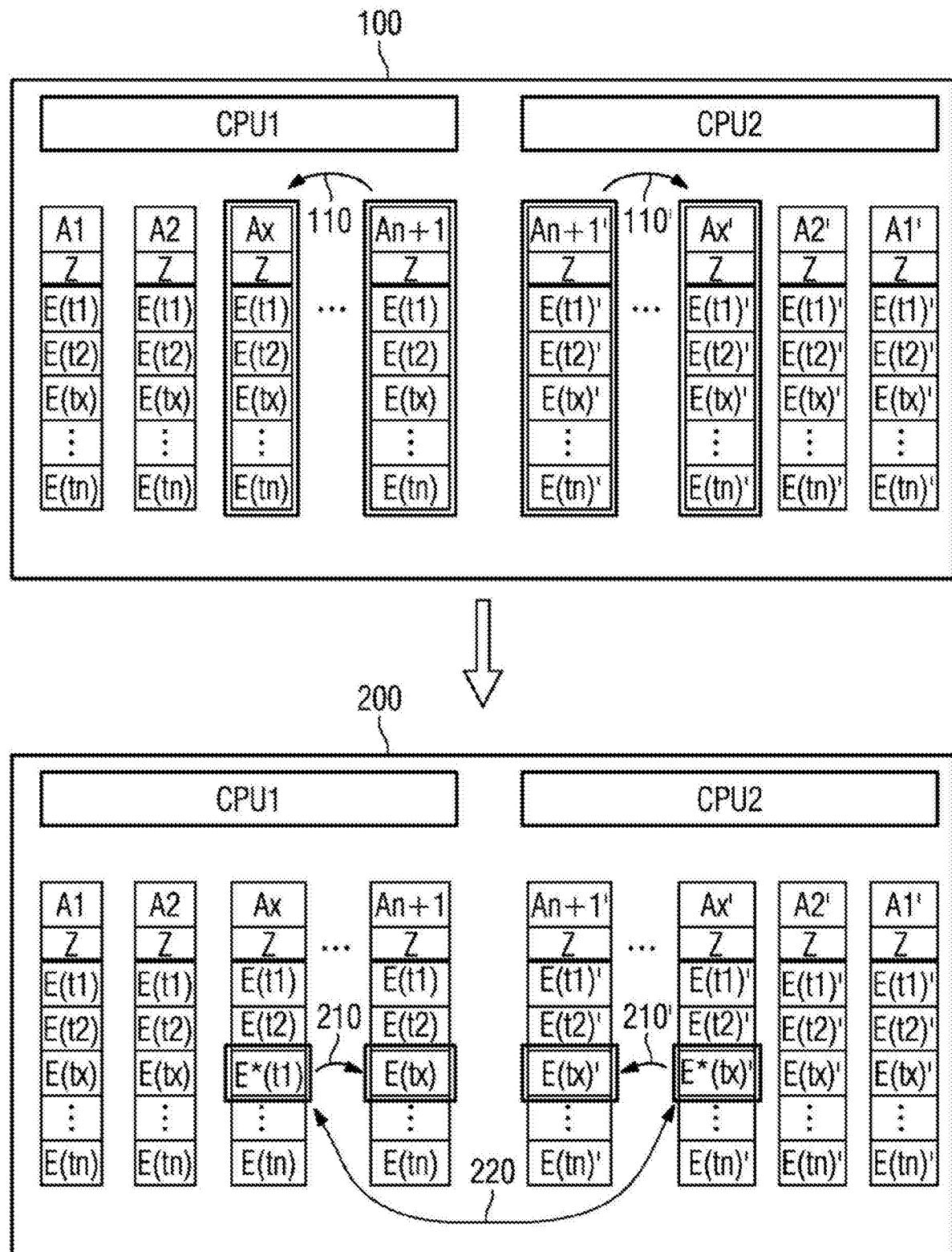


图1