



(12)发明专利申请

(10)申请公布号 CN 108351815 A

(43)申请公布日 2018.07.31

(21)申请号 201680066047.5

(22)申请日 2016.10.21

(30)优先权数据

102015222321.3 2015.11.12 DE

(85)PCT国际申请进入国家阶段日

2018.05.11

(86)PCT国际申请的申请数据

PCT/EP2016/075381 2016.10.21

(87)PCT国际申请的公布数据

W02017/080793 DE 2017.05.18

(71)申请人 西门子股份公司

地址 德国慕尼黑

(72)发明人 M.安布鲁斯特 M.比肖夫

C.布克尔 L.菲格 A.齐尔克勒

(74)专利代理机构 中国专利代理(香港)有限公司 72001

代理人 杜荔南 刘春元

(51)Int.Cl.

G06F 11/07(2006.01)

B60W 50/00(2006.01)

G06F 11/20(2006.01)

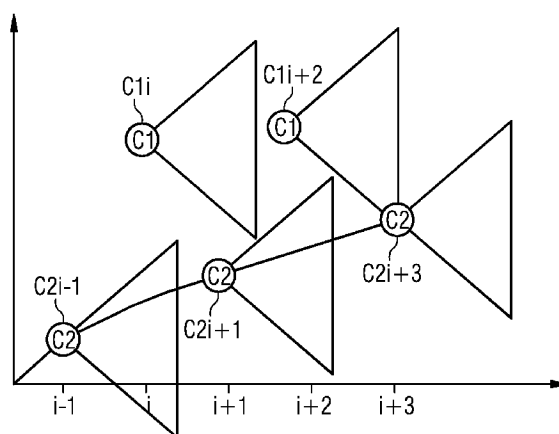
权利要求书2页 说明书7页 附图3页

(54)发明名称

用于运行多核处理器的方法

(57)摘要

根据本发明,多核处理器的至少两个处理器核用于双通道地计算安全关键的应用程序。在此,运算操作并不在每个计算循环中在两个处理器核上冗余地计算,而是两个处理器核在不同的工作循环中承担不同的应用程序的负荷。因此,以有利的方式避免所需计算容量的翻倍。为了实现处理器核的相互监控,运算操作交替地在两个处理器核上计算。通过所描述的错误识别机制可以识别随机的错误。尽管根据本发明的错误识别的质量略低于现有技术中已知的具有并行冗余多通道计算的“双通道运行”中的质量。但是,相对于对计算能力的较低开销的要求,错误识别的质量可以排在后面,尤其是在对于控制系统需要经济地实现的情况下。本发明因此将对足够安全的错误识别的要求与计算能力的经济设计统一起来。



1. 用于运行多核处理器的方法, 在所述多核处理器上执行应用程序, 所述应用程序包括多个循环的运算操作, 其中为了计算相应的运算操作规定时间上计量的工作循环,

所述方法的特征在于,

- 在多核处理器的根据分配模式所指派的处理器核上执行运算操作的计算;
- 在获得当前运算操作的结果之后, 在当前的工作循环内和根据比较模式产生在当前结果与至少一个工作循环之前的运算操作的至少一个结果之间的至少一个间距;
- 在至少一个间距在期望值之外的情况下输出出错指示;
- 在多核处理器的根据分配模式所指派的处理器核上执行后续的运算操作的计算。

2. 根据权利要求1所述的方法,

其特征在于,

根据分配模式规定将运算操作分别交替地指派给多核处理器的处理器核中的一个处理器核。

3. 根据上述权利要求中任一项所述的方法,

其特征在于,

在所述指派交替到多核处理器的第二处理器之前, 根据分配模式针对可预给定的多个工作周期规定到第一处理器核的指派。

4. 根据权利要求3所述的方法,

其特征在于,

在输出出错指示时, 提高指派给第一处理器核的多个工作循环。

5. 根据上述权利要求2至4中任一项所述的方法,

其特征在于,

根据分配模式规定将运算操作分别交替地、尤其旋转地指派给多核处理器的至少三个处理器核中的一个处理器核。

6. 根据上述权利要求中任一项所述的方法,

其特征在于,

根据比较模式, 从一个工作循环之前的运算操作的结果与当前工作循环的结果中确定第一间距。

7. 根据权利要求6所述的方法,

其特征在于,

在第一间距在对于第一间距预期的值之外的情况下输出出错指示。

8. 根据权利要求6所述的方法,

其特征在于,

- 从两个工作循环之前的运算操作的结果和当前工作循环的结果中确定第二间距;
和/或

- 从两个工作循环之前的运算操作的结果和一个工作循环之前的运算操作的结果中确定第三间距; 和/或

- 从一个工作循环之前的运算操作的结果与当前工作循环的结果的差中确定第一差;
和/或

- 从两个工作循环之前的运算操作的结果与一个工作循环之前的运算操作的结果的

差中确定第二差。

9. 根据权利要求8所述的方法，

其特征在于，

在如下情况下输出出错指示，

- 第二间距小于第一间距；以及
- 第二间距小于第三间距；以及
- 第一差具有与第二差不同的符号。

10. 根据权利要求8所述的方法，

其特征在于，

- 从三个工作循环之前的运算操作的结果和一个工作循环之前的运算操作的结果中确定第四间距；

- 从三个工作循环之前的运算操作的结果和两个工作循环之前的运算操作的结果中确定第五间距；

- 从三个工作循环之前的运算操作的结果与两个工作循环之前的运算操作的结果的差中确定第三差。

11. 根据权利要求10所述的方法，

其特征在于，

在如下情况下输出出错指示，

- 第二间距小于第一间距；以及
- 第二间距小于第三间距；以及
- 第四间距小于第三间距；以及
- 第四间距小于第五间距；以及
- 第一差具有与第三差不同的符号；以及
- 第三差具有与第二差不同的符号。

12. 根据上述权利要求中任一项所述的方法，

其特征在于，

根据比较模式针对每个工作循环规定至少一个间距的确定。

13. 根据上述权利要求1至12中任一项所述的方法，

其特征在于，

根据比较模式针对每第n个工作循环规定至少一个间距的确定，其中n是自然数。

14. 计算机程序产品，其具有用于当计算机程序产品在控制系统处由至少一个多核处理器实施时执行上述权利要求中任一项所述的方法的装置。

用于运行多核处理器的方法

技术领域

[0001] 本发明涉及一种根据权利要求1的前序部分所述的用于运行多核处理器的方法。

背景技术

[0002] 现代和未来的车辆配备有许多电子控制的功能,这些功能对车辆的控制系统提出了在其安全性和可用性方面提高的要求。

[0003] 目前,使用基于双控制计算机(DCC)的高度安全的且高度可用的控制系统,利用所述双控制计算机可以确保防故障地执行软件功能。为此,需要在两个独立的微处理器上执行相同的软件。微处理器的外围功能——即非易失性和易失性的存储器单元、网络连接单元、资源管理等——在两种分离的处理路径上进行,这两种分离的处理路径也被称为双倍“双通道”处理方式的“通道”。在确定的时间点,两个微处理器的结果相互交换并且在两个微处理器中彼此比较。

[0004] 若在所谓的通道中的一个通道中或在其通信连接中出现错误,利用该比较在所述通道中的至少一个通道中识别出不同的结果。因此,双控制计算机被视为有故障的并且关断。由此确保:没有有误的控制信号被双控制计算机发出并且因此实现“故障沉默”特性。通过提供在第一双控制计算机有故障时承担其处理的另一双控制计算机甚至可以实现“故障后仍能操作(Fail-Operational)”特性。这种在使用两个独立地工作的处理器情况下通过双通道运行来保证的错误识别概率通过高硬件开销实现。

发明内容

[0005] 本发明所基于的任务是提出用于实现具有高可用性和完整性的控制系统的设备和方法,该控制系统需要对硬件的更低开销并且同时能够实现硬件资源的最佳利用。

[0006] 该任务根据本发明通过具有权利要求1的特征的方法来解决。

[0007] 在根据本发明的方法中规定多核处理器的运行,在所述多核处理器上优选执行安全关键的应用程序,所述应用程序包括多个循环的运算操作(Rechenoperation)。术语“循环的运算操作”尤其包括调节量的多级的计算,其中在离散时间点为控制系统输送数字化的操纵量(Stellgröße),在那里时间同步地计算所述操纵量并且作为数字输出信号输出。为了计算相应的运算操作,规定时间上计量的工作循环,该作业循环优选显著小于所基于的控制回路的最小时间常数。

[0008] 根据本发明规定,规定分配模式,根据该分配模式将运算操作的计算输送给多核处理器的一个核。在获得当前运算操作的结果之后,在当前工作循环内和根据比较模式执行在当前结果与至少一个工作循环之前的运算操作的至少一个结果之间的至少一个间距。若至少一个间距在期望值之外,则输出出错指示。接着,在多核处理器的根据分配模式所指派的另一核上进行后续的运算操作的计算。

[0009] 根据本发明,多核处理器的处理器核(核)用于多通道地计算安全关键的应用程序,其中处理器核在每个工作循环中交替。

[0010] 通过根据本发明借助比较模式来比较在当前结果与至少一个工作循环之前的运算操作的至少一个结果之间的至少一个间距,可以识别随机错误。尽管根据本发明的错误识别的质量略低于在现有技术中已知的具有并行冗余多通道计算的“双通道运行”中的质量。但是,相对于对计算能力的较低开销的要求,错误识别的质量可以排在后面,尤其是在对于控制系统需要经济地实现的情况下。本发明因此将对足够安全的错误识别的要求与计算能力的经济设计统一起来。

[0011] 有利地,在目前未被加载安全关键的应用程序的处理的其余处理器核上可以执行用于其他安全关键的或非安全关键的应用程序的运算操作,使得尽管进行了多通道计算,但总体上没有花费对计算能力的显著的额外需求。尤其是,避免了所需计算能力的翻倍,所述计算能力在现有技术中在分别一个处理器核上具有冗余计算的双通道运行的情况下是已知的。

[0012] 本发明的其他有利构型是从属权利要求的主题。

[0013] 根据本发明的一个构型,规定将运算操作分别交替地指派给多核处理器的核中的一个核。本发明的构型尤其在两个或更多核处理器以双通道计算安全关键的应用程序来运行时是选择的手段,其中处理器核在每一个工作循环中都交替。

[0014] 在下文中所阐述的本发明的构型基于多级比较模式,所述多级比较模式基于如下的考虑:若在第一处理器核中或在与第一处理器核关联的存储器中在示例性的工作循环*i*中出现错误,则通过该第一处理器核所计算出的结果歪曲。在后续的工作循环*i+1*中,现在第二处理器核计算被歪曲的结果。在工作循环*i+2*中,又在第一处理器核中计算歪曲的结果。在每一个工作循环中,在两个处理器核的每个处理器核上比较在当前结果与至少一个过去的结果之间的间距。在此,最早在工作循环*i*中并且最迟在工作循环*i+2*中可以确定错误。

[0015] 根据本发明的一个构型,规定比较模式,根据该比较模式从一个工作循环之前的运算操作的结果和当前工作循环的结果中确定第一间距。如果该第一间距超过最大值,或换言之如果该第一间距在对于第一间距预期的值之外,则根据本发明输出出错指示。根据该构型,在分别交替的处理器核上双通道地计算安全关键的应用程序时,处理器核的错误计算在工作循环*i*中被识别出,在该工作循环中由该处理器核计算出的结果与在工作循环*i-1*中由另一处理器核计算出的结果偏差为使得当前结果距另一结果具有一间距,该间距在可预给定的最大值或最大间距之外。

[0016] 通过本发明规定的基于比较模式的一致性检验并不如例如在现有技术中已知的双通道运行中那样根据位标识(Bit-Identität)进行,。对此的原因在于,也将来自相继的工作循环中的输入数据用于相继的工作循环的运算操作。因为来自相继的工作循环的输入数据通常是不同的,因此结果或输出数据也可以围绕允许的间距不同。对于该允许的间距可预给定允许的最大间距,或者可替换地或附加地,从来自过去的工作循环的结果的间距中计算允许的间距。最后所提及的从来自过去的工作循环的结果的间距中计算允许的间距在后续的构型中予以阐述。

[0017] 通过根据本发明的措施可以识别随机的错误。在将相同的软件在不同的处理器核上执行时,一般不能识别出系统性的错误。此外,这也适于现有技术中已知的双通道运行。

[0018] 如果也应当识别系统错误,则在现有技术中已知的是,在双通道计算机的两个处理器上执行不同的软件,所述软件由此非常可能不包含相同的错误。在此,第二软件可以要

么具有与第一软件相同的功能范围,要么执行简化的计算。后者也称作包络函数。在两个情况中,即便在双通道计算机的情况下也不能执行位标识的比较,而是只能执行结果的比较。有利地,所述的两个方法可以与当前根据本发明的方法组合。为此,例如在芯C1上在循环i中将会实施应用程序A1,而在芯C2上在循环i+1中将会实施应用程序A2。在没有错误的情况下,所描述的错误识别将会不改变地工作,必要时以增大的允许的增量(Delta)工作。尤其是,当其中一个应用程序是包络函数时,则如现有技术中常见的那样规定较大的增量。由于功能的多样性,在该实施方式中描述的错误识别机制也将能够识别应用程序A1和A2中的错误或差异。

[0019] 根据本发明的一个构型,在工作循环i+1中确定过去的运算操作的结果之间的其他间距以及差,其中:

- 从两个工作循环之前的运算操作的结果和当前工作循环的结果中确定第二间距;和/或
- 从两个工作循环之前的运算操作的结果和一个工作循环之前的运算操作的结果中确定第三间距;和/或
- 从一个工作循环之前的运算操作的结果与当前工作循环的结果的差中确定第一差;和/或
- 从两个工作循环之前的运算操作的结果与一个工作循环之前的运算操作的结果的差中确定第二差。

[0020] 根据本发明的一个构型,在如下情况下输出出错指示:

- 第二间距小于第一间距;以及
- 第二间距小于第三间距;以及
- 第一差具有与第二差不同的符号。

[0021] 根据该构型,在分别交替的处理器核上双通道地计算安全关键的应用程序时在工作循环i+1中识别出处理器核的错误计算,其中由一个处理器核计算出的结果与由另一处理器核计算出的结果在系统上偏差。在此,在工作循环i-1和i+1中计算的结果的第二间距小于工作循环i和i+1中的结果的第一间距以及小于工作循环i-1和i中的结果的第三间距。此外,工作循环i-1和i中的结果的第一差具有与工作循环i和i+1中的结果的第二差不同的符号。

[0022] 根据本发明的一个构型,在工作循环i+2中确定过去的运算操作的结果之间的其他间距以及差,其中:

- 从三个工作循环之前的运算操作的结果和一个工作循环之前的运算操作的结果中确定第四间距;和/或
- 从三个工作循环之前的运算操作的结果和两个工作循环之前的运算操作的结果中确定第五间距;和/或
- 从三个工作循环之前的运算操作的结果与两个工作循环之前的运算操作的结果的差中确定第三差。

[0023] 根据本发明的一个构型,在如下情况下输出出错指示:

- 第二间距小于第一间距;以及
- 第二间距小于第三间距;以及

- 第四间距小于第三间距;以及
- 第四间距小于第五间距;以及
- 第一差具有与第三差不同的符号;以及
- 第三差具有与第二差不同的符号。

[0024] 根据该构型,在分别交替的处理器核上双通道地计算安全关键的应用程序时在工作循环 $i+2$ 中识别出处理器核的错误计算,其中由一个处理器核计算出的结果与由另一处理器核计算出的结果在系统上偏差。在此,在工作循环 i 和 $i+2$ 中计算出的结果的第二间距小于工作循环 $i+1$ 和 $i+2$ 中的结果的第一间距以及小于工作循环 i 和 $i+1$ 中的结果的第三间距。

[0025] 此外,在工作循环 $i-1$ 和 $i+1$ 中计算出的结果的第四间距小于工作循环 i 和 $i+1$ 中的结果的第三间距以及小于工作循环 $i-1$ 和 i 中的结果的第五间距。

[0026] 此外,工作循环 i 和 $i+1$ 中的结果的第一差具有与工作循环 $i-1$ 和 i 中的结果的第三差不同的符号,其中第三差又具有与工作循环 i 和 $i+1$ 中的结果的第二差不同的符号。

[0027] 根据本发明的一个构型,规定比较模式,该比较模式在每个工作循环中规定至少一个间距的确定以及与前面的间距和/或差的比较。可替换地,仅在保留的工作循环中例如在每第四或第 n 个工作循环中确定间距或差和/或它们的比较。此外,当分别针对每个处理器核所确定的结果远离彼此地漂移和/或朝向极限值运动时,根据比较模式也可以规定增加比较循环。

附图说明

[0028] 在下文中借助附图进一步阐述本发明的其他实施例和优点。在附图中示出:

图1:在离散工作循环时两个分别交替被计算的运算操作的结果的示意性视图,其中绘制了在当前结果与后续结果之间的间距的相应预期的值域;

图2:在离散工作循环时两个分别交替被计算的运算操作的结果的示意性视图,其中绘制了在当前结果与后续结果之间的相应间距;

图3:两个运算操作关于时间的结果的示意性视图,其中所基于的运算操作包含积分调节元件;

图4:在离散工作循环时两个以第一采样率分别交替被计算的运算操作的结果的示意性视图,其中所基于的运算操作包含积分调节元件;以及

图5:在离散工作循环时两个以第二采样率分别交替被计算的运算操作的结果的示意性视图,其中所基于的运算操作包含积分调节元件。

具体实施方式

[0029] 在图1和图2中示出了时间图表,在该时间图表的纵坐标上绘制了在离散时间点时两个分别交替地由两个处理器核中的一个处理器核所计算的运算操作的结果 $C2i-1$ 、 $C1i$ 、 $C2i+1$ 、 $C1i+2$ 、 $C2i+3$ (其中相应对应的附图标记前缀 $C1$ 代表第一处理器核,并且 $C2$ 代表第二处理器核)。在横坐标上绘制的离散时间点对应于工作循环 $i-1$ 、 i 、 $i+1$ 、 $i+2$ 、 $i+3$ 。

[0030] 在图1中绘制了在当前结果与后续结果之间的间距的相应预期的值域,参见从相应的点状的结果值 $C2i-1$ 、 $C1i$ 、 $C2i+1$ 、 $C1i+2$ 、 $C2i+3$ 出发的三角形区域。

[0031] 根据本发明,循环地处理两个基本上相同的运算操作的处理器核在每一个工作循环 $i-1$ 、 i 、 $i+1$ 、 $i+2$ 、 $i+3$ 中交替。由此,分别未参与运算操作的该处理的处理器核可以处理其他任务,使得不浪费冗余的计算能力。同时,在运算操作的处理中的错误也影响在其他处理器核上分别其他的运算操作。

[0032] 若在一个处理器核 $C1$ 中或在与该处理器核 $C1$ 关联的存储器中例如在工作循环 i 中出现错误,则通过该处理器核 $C1$ 所计算的结果 $C1i$ 被歪曲。在接下来的工作循环 $i+1$ 中,现在另一处理器核 $C2$ 计算这次被歪曲的结果 $C2i+1$ 。

[0033] 在接下来的工作循环 $i+2$ 中,又在处理器核 $C1$ 中计算被歪曲的结果 $C1i+2$ 。根据按照根据本发明的比较模式的构型,在两个处理器核 $C1$ 、 $C2$ 中规定以下的监控机制,所述监控机制可以最早在工作循环 i 中并且最晚在工作循环 $i+2$ 中确定存在错误。

[0034] a. 在工作循环 i 中:根据图1在工作循环 i 和 $i+1$ 中计算的结果 $C1i$ 和 $C2i+1$ 的第一间距超过了最大值。换言之,在工作循环 $i+1$ 中所计算的结果 $C2i+1$ 在对于最大间距所预期的值的三角形区域之外。

[0035] b. 在工作循环 $i+1$ 中:在工作循环 $i-1$ 和 $i+1$ 中计算的结果 $C2i-1$ 和 $C2i+1$ 的第二间距小于在工作循环 i 和 $i+1$ 中所计算的结果 $C1i$ 和 $C2i+1$ 的第一间距。此外,在工作循环 $i-1$ 和 $i+1$ 中计算的结果 $C2i-1$ 和 $C2i+1$ 的第二间距小于在工作循环 $i-1$ 和 i 中所计算的结果 $C2i-1$ 和 $C1i$ 的第三间距。此外,来自工作循环 $i-1$ 和 i 的结果的第一差(即 $(C2i-1)-(C1i)$)具有与来自工作循环 i 和 $i+1$ 的结果的第二差(即 $(C1i-C2i+1)$)不同的符号。

[0036] c. 在工作循环 $i+2$ 中:在工作循环 i 和 $i+2$ 中计算的结果 $C1i$ 和 $C1i+2$ 的第二间距小于在工作循环 $i+1$ 和 $i+2$ 中所计算的结果 $C2i+1$ 和 $C1i+2$ 的第一间距以及小于在工作循环 i 和 $i+1$ 中计算的结果 $C1i$ 和 $C2i+1$ 的第三间距。此外,结果 $C2i-1$ 和 $C2i+1$ 的在工作循环 $i-1$ 和 $i+1$ 中所计算的第四间距小于结果 $C1i$ 和 $C2i+1$ 的在工作循环 i 和 $i+1$ 中所计算的第三间距以及小于结果 $C1i$ 和 $C2i-1$ 和 $C1i$ 的在工作循环 $i-1$ 和 i 中所计算的第五间距。此外,在工作循环 i 和 $i+1$ 中计算出的结果 $(C1i)-(C2i+1)$ 的第一差具有与在工作循环 $i-1$ 和 i 中计算出的结果 $(C2i-1)-(C1i)$ 的第三差不同的符号,其中第三差又具有与在工作循环 i 和 $i+1$ 中所计算出的结果 $(C1i)-(C2i+1)$ 的第二差不同的符号。

[0037] 图2示出了第一间距 $A1$ 、第二间距 $A2$ 、第三间距 $A3$ 、第四间距 $A4$ 和第五间距 $A5$ 。

[0038] 若对最大梯度没有规定,因此可以首先在工作循环 $i+2$ 中可靠地识别出存在错误。该测试可以根据需要连续地在每一个工作循环中执行,或例如在每第 n 个循环中执行。

[0039] 附加地,也可以限定更窄的最大间距,在检测出错误之前该更窄的最大间距允许被超过一次或多次。一致性检验不能如在“真实的”双通道运行中那样根据位标识来进行,因为这两个处理器核使用来自相继循环的输入数据用于相继的工作循环中的计算。

[0040] 因为输入数据在相继的工作循环中将是不同的,可能围绕通过预定义的最大间距而受到限制,所以输出数据也可以围绕允许的增量不同。对于该增量,允许的值可以是已知的或由输入数据的间距来计算。

[0041] 根据本发明的优点是将所需的计算性能减半,而不相对于双通道计算提高利用应用程序实现的数字调节器的循环时间。尽管这导致一致性检验——增量一致性而不是位标识——的质量下降并且错误响应减慢最多两个工作循环,但相对于双通道计算,无错误情况下的调节器的循环时间并未增加。

[0042] 根据本发明的其他实施形式,若应用程序至少部分地实现如下数字调节器,则采取附加措施,所述数字调节器至少部分地包含积分调节元件、即具有I分量的调节器,或者另外地将过去的系统状态一起包括到计算中。

[0043] 图3示出了由第一处理器核C1和第二处理器核C2确定的运算操作关于时间的两个相应结果的示意表示,其中所基于的运算操作包含积分调节元件。由第二处理器核C2确定的结果曲线基本上跟随运算操作的理想值曲线ID,而由第一处理器核C1确定的结果曲线漂移。

[0044] 因为两个或者更多个处理器核获得略微不同的输入值,所以输出值同样可以略微不同。这在根据本发明的增量一致性检验的范围中是允许的。但若两个处理器核的调节目标略高于和略低于理想值,那么两个处理器核中的积分变量可以不断增大,因为每个处理器核在相同方向上总是看到略微偏差。

[0045] 这能导致所操控的组件(Aggregat)的抖动,因为这两个调节器沿着相反的方向越来越强地调节。一旦在所述调节器中的一个调节器中的积分器变量达到极限值,例如变量的值域边界,则达到临界状况。现在,所述调节器中的一个调节器不能再相应地反控制并且调节值漂移。尽管这在上文所描述的条件下会导致可靠的关断。但系统的可靠性不再有,因为在此情况下处理器核的交替产生错误。为了避免该问题,规定合适的漂移补偿。为此,例如积分器的值在两个处理路径中可以相互交换。为了避免可能的错误传播,建议限制积分器的交换值。在正常运行中允许的积分器值可以根据从调节对象的动态性和调节器的设计来确定。积分器值的限制并不关键,因为该限制在最坏的情况下可能导致调节器特性的减慢,但不会导致不稳定。

[0046] 如果调节器的输入信号以与工作循环频率——即工作循环的时间值的倒数——类似的频率振荡,则可能发生不稳定。这可能导致在调节器输入端处向一个处理器核总是传送过高的值,而向另一个处理器核总是传送过低的值,并且由此操纵量(Stellgröße)根据图4振荡。该特性根据上文的调节被识别为错误并且因此应予以避免。

[0047] 以下构型适合于:

- 避免欠采样。调节器原则上应设计为使得采样率显著高于调节量的频率。四个或更多的因数经运行验证,参见图5。在所有其动态性充分已知的调节对象中,可以并且应当应用该措施。

[0048] - 处理器核以《华尔兹节拍(Walzertakt)》的交替或类似的不连续的交替:若调节对象的动态性未知,则可以改变处理器核交替的节奏。例如,运算操作的计算可以总是输送给第一处理器核两次,接着输送给第二处理器核C2一次。由于在处理器核交替时不对称的周期持续时间,不能得出调节量的如下频率:该频率导致上文所述的与不期望的错误识别相关联的特性。

[0049] - 使用三个或更多核处理器。例如,运算操作的计算可以一次性地输送给第一处理器核C1,随后输送给第二处理器核C2,随后输送给第三处理器核C3。因此,其中所述处理器核中的一个处理器核中的错误可以与振荡输入数据区分开来,因为所述处理器核中的一个处理器核中的错误只会在每第三个循环中出现。

[0050] - 积分器值反馈连带限制被反馈的积分器值的有效值域如上地执行。

[0051] - 将系统状态与历史比较:若系统状态从多个过去的值中计算出,则不同的输入

数据也可以导致在计算所述系统状态时不同的结果。这里为了避免错误识别,要么允许在计算错误状态时的时间增量,要么在处理路径之间交换状态。

[0052] 替代于上文所描述的方法,根据本发明的一个可替代的构型,两个处理路径可以访问相同的存储器区域。由此,所有历史数据、积分器值等对于两个处理路径会是相同的并且由此不需要上述机制。这种简化的代价是,共同使用的存储器区域变成共同的错误原因区域。对于一些应用情况,如果在共同的错误原因区域中未发现的错误的概率通过合适的措施、例如ECC(Error-Correcting Code,纠错码)或存储器扰码是足够小的,则这是可接受的。

根据本发明,多核处理器的至少两个处理器核被用于双通道地计算安全关键的应用程序。在此,运算操作并不在每个计算循环中在两个处理器核上冗余地计算,而是两个处理器核在不同的工作循环中承担不同的应用程序的负荷。因此,以有利的方式避免所需计算容量的翻倍。为了实现处理器核的相互监控,运算操作交替地在两个处理器核上计算。通过所描述的错误识别机制可以识别随机的错误。

[0053] 尽管根据本发明的错误识别的质量略低于现有技术中已知的具有并行冗余多通道计算的“双通道运行”中的质量。但是,相对于对计算能力的较低开销的要求,错误识别的质量可以排在后面,尤其是在对于控制系统需要经济地实现的情况下。本发明因此将对足够安全的错误识别的要求与计算能力的经济设计统一起来。

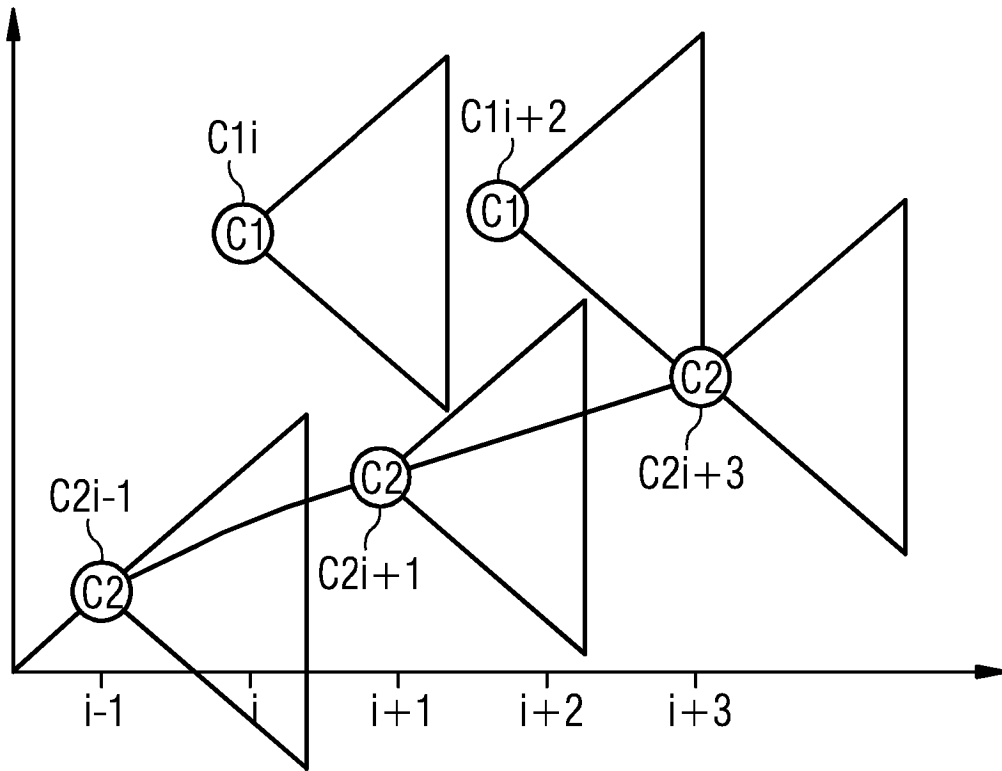


图 1

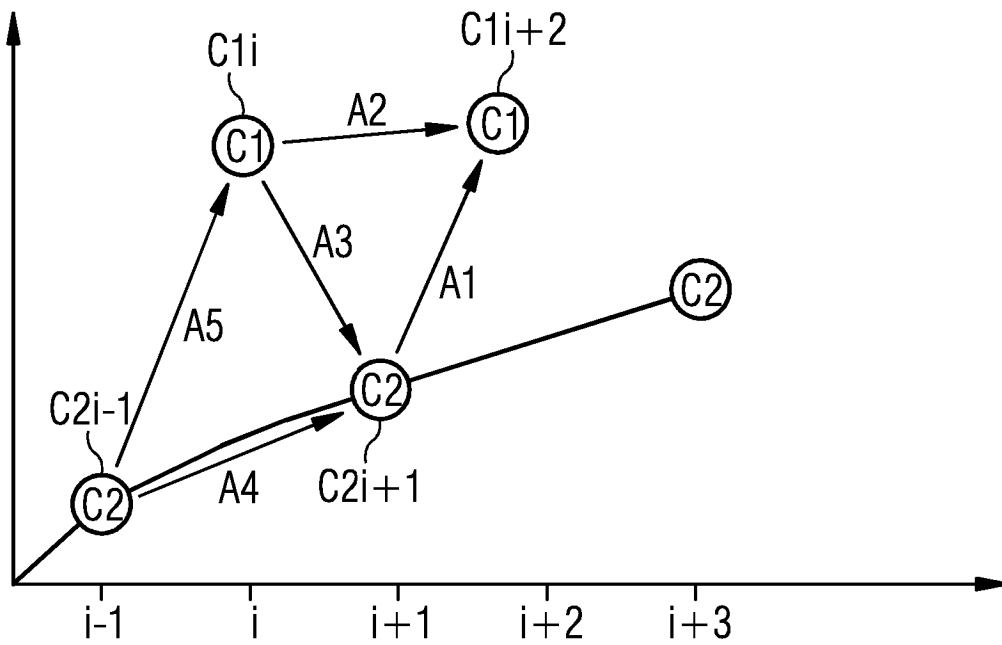


图 2

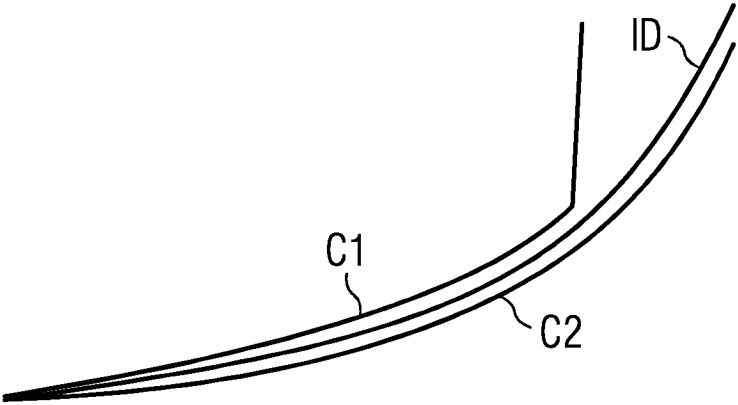


图 3

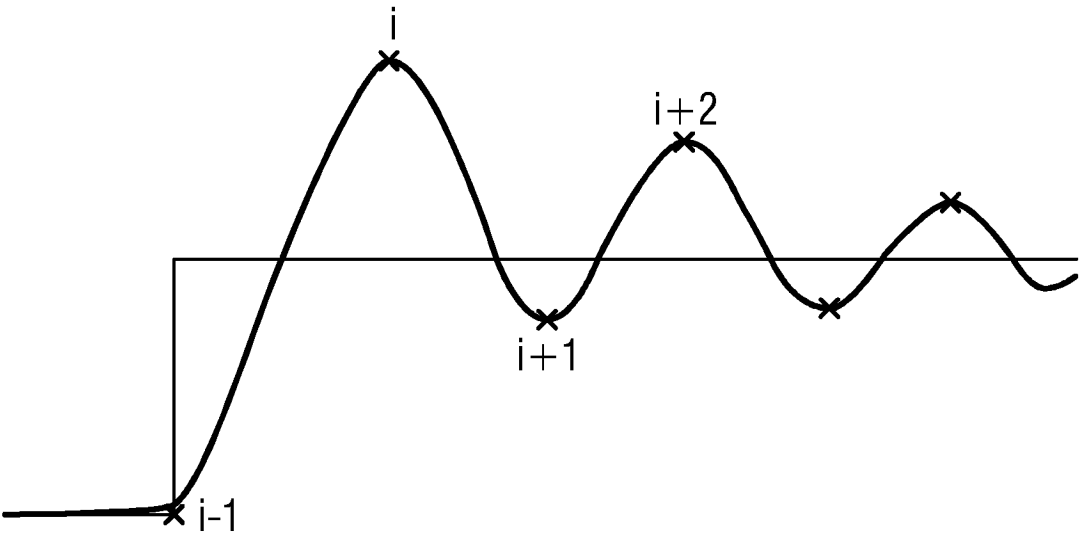


图 4

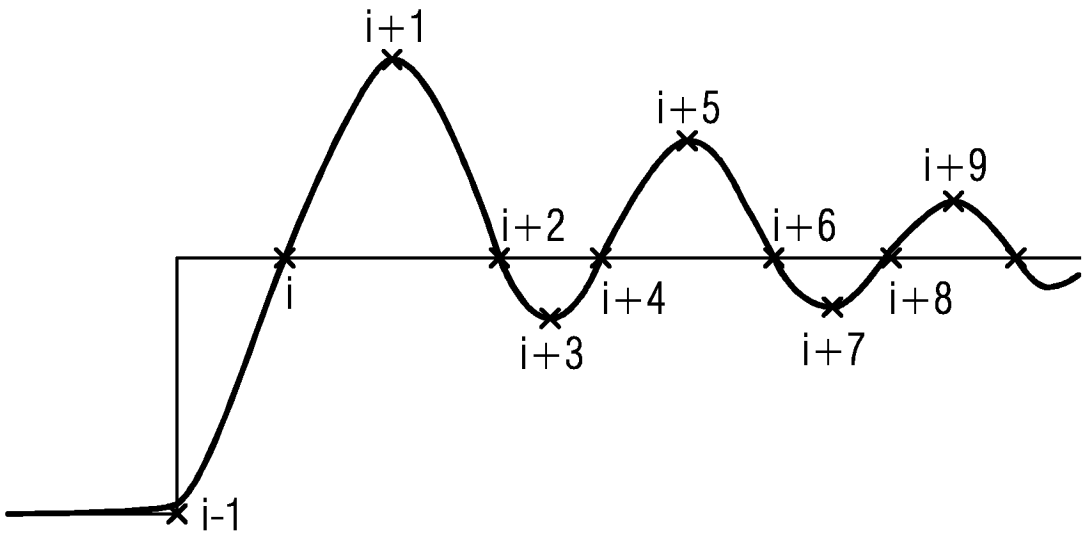


图 5