(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101243407 B
(45)授权公告日 2012.05.16

(21)申请号 200680029573.0

(22)申请日 2006.07.26

(30)优先权数据
102005037246.5 2005.08.08 DE

(85)PCT申请进入国家阶段日
2008.02.13

(86)PCT申请的申请数据
PCT/EP2006/064690 2006.07.26

(87)PCT申请的公布数据
WO2007/017386 DE 2007.02.15

(73)专利权人 罗伯特·博世有限公司
地址 德国斯图加特

(72)发明人 R·韦伯尔 B·米勒 R·格梅利克

(74)专利代理机构 中国专利代理(香港)有限公司 72001
代理人 卢江 刘宗杰

(51)Int.Cl.
G06F 11/16(2006.01)

(56)对比文件
US 2004/0123201 A1,2004.06.24,说明书第20,26-27,55-58段、权利要求2,4,24-25.
CN 1336587 A,2002.02.20,全文.
CN 1434941 A,2003.08.06,说明书第10页12-20行.
US 6615366 B1,2003.09.02,全文.

审查员 李芳

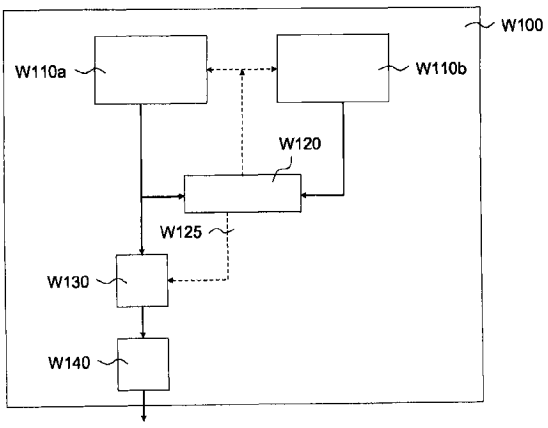
权利要求书 1 页 说明书 6 页 附图 5 页

(54)发明名称

用于控制计算器系统的方法和装置

(57)摘要

一种用于控制具有至少两个执行单元和一个比较单元的计算器系统的方法,该计算器系统工作在时钟同步的模式中,并且将至少两个执行单元的结果相比较,其特征在于,在通过比较单元识别出错误之时或之后在至少一个执行单元上运行针对该执行单元的错误识别机制。



CN 101243407 B

1. 一种用于控制具有至少两个执行单元 (W110a，W110b) 和一个比较单元 (W120-123) 的计算器系统的方法，该计算器系统工作在时钟同步的模式中，并且将至少两个执行单元 (W110a，W110b) 的结果相比较，其特征在于，在通过比较单元 (W120-123) 识别出错误之时或之后中断在恰好一个执行单元 (W110a，W110b) 上的当前的指令序列，在该一个执行单元 (W110a，W110b) 上运行错误识别机制，在运行错误识别机制的持续时间期间断开所述至少两个执行单元 (W110a，W110b) 的比较单元 (W120-123)，并且在至少一个另外的执行单元 (W110a，W110b) 上继续处理按照规定的程序序列。

2. 根据权利要求 1 所述的方法，其特征在于，如果错误识别机制没有识别出错误，则在运行完错误识别机制之后继续执行按照规定的程序流程。

3. 根据权利要求 2 所述的方法，其特征在于，在一个执行单元 (W110a，W110b) 上找到错误之时或之后断开该出现错误的执行单元 (W110a，W110b)。

4. 根据权利要求 3 所述的方法，其特征在于，停用所述比较单元 (W120-123)。

5. 根据权利要求 3 或 4 所述的方法，其特征在于，在停用至少一个部件时产生错误信号，该错误信号被提供给应用。

6. 根据权利要求 1 所述的方法，其特征在于，在出现错误之后仅用一个执行单元 (W110a，W110b) 进行的运行在时间上是有限的，而且最迟在事先确定的时间结束之后就断开所述计算器系统。

7. 一种用于控制具有至少两个执行单元 (W110a，W110b) 和一个比较单元 (W120-123) 的计算器系统的装置，该计算器系统工作在时钟同步的模式中，并且将至少两个执行单元 (W110a，W110b) 的结果相比较，其特征在于，具有被构造用于进行如下操作的装置：在通过比较单元 (W120-123) 识别出错误之时或之后中断在恰好一个执行单元 (W110a，W110b) 上的当前的指令序列，在该一个执行单元 (W110a，W110b) 上运行错误识别机制，在运行错误识别机制的持续时间期间断开所述至少两个执行单元 (W110a，W110b) 的比较单元 (W120-123)，并且在至少一个另外的执行单元 (W110a，W110b) 上继续处理按照规定的程序序列。

用于控制计算器系统的方法和装置

技术领域

[0001] 本发明涉及按照独立权利要求的前序部分的用于在具有两个核的处理器系统中出现错误时保持系统功能的装置和方法，以及对应的处理器系统。

背景技术

[0002] 从现有技术公知为了识别错误采用例如微控制器（μC）的冗余，还有一个 μC 如 CPU（中央处理单元）的部件的冗余。在此冗余计算的数据和冗余产生的信号要由比较单元就一致性进行比较。

[0003] 具有冗余 CPU 的微控制器也称为双核微控制器（双核 μC）。在双核 μC 中，两个 CPU 可以节拍同步、即并行地（按照时钟同步 (Lockstep) 的模式）工作，也可以错开几个节拍地工作。两个 CPU 都接收相同的输入数据并处理相同的程序或相同的指令。如果在冗余实施的核之一中出现影响该核的至少一个输出信号的错误，则会导致待比较的数据出现差异，这种差异将由比较单元识别出。在此输出信号除了“数据输出”之外还可以包括指令地址和控制信号。比较单元在识别出待比较的信号之间存在差异时产生状态或错误信号，利用该信号可以将该比较结果向外发布。但是，如果不对冗余实施的单元采取额外的错误识别机制，就无法找到出现错误的部件，也不能确定错误原因的类型。

[0004] 如果在与安全有关的控制和调节系统中使用上述冗余，则通常在识别出冗余获得的信号之间存在差异时将整个系统切换到“安全状态”，即使这种差异的原因是只具有很短的作用时间的瞬时错误。在诸如 ESP 系统的汽车系统中，“安全状态”大都意味着切断系统的连接。

[0005] 由于半导体结构变得越来越小，预计瞬时处理器错误会越来越多，这些处理器错误例如由于宇宙射线引起。为了能够克服瞬时错误而不切端系统的连接，并且在运行期间能够容忍或甚至“治愈”该错误，在现有技术中已经存在几种解决方案：利用大多很费事的方法通过特定于应用的、通常由模型支持的似然性来检测错误，必要时触发计算器系统的复位。计算器系统重新初始化，并在初始化时间和可选的“补偿检查 (Wiedergutpruefung)”之后（在例如几百 ms 之后）重新进入运行就绪状态（所谓的“前向恢复”）。在非实时应用中（例如金融市场上的交易）特定于应用地在交易之前建立一种状态，将该状态存储起来，并且当确认该交易成功结束时才将该状态作为无效的丢弃。如果在该交易期间出现错误，则返回到所存储的起始时刻（“后向恢复”）。在实时系统中这样的解决方案非常费事，而且大多要在对处理器系统复位或补偿检查的持续时间内中断该功能。

[0006] 随着汽车中的电子调节系统的功能范围增大，诸如具有转向干预的 ESP 等系统的断开不是在每个运行状态下都意味着过渡到安全的系统状态。

发明内容

[0007] 本发明要解决的技术问题是一种用于运行双核处理器（或双核处理器系统）的方法，其目标是提高抵抗错误的鲁棒性和增大在处理器系统中出现瞬时和永久错误时系统功

能的（部分）可用性。在优选的实施例中，这可以在对各个程序部分保持原始的执行时间的情况下实现。

[0008] 在根据现有技术的、运行在时钟同步模式下的双核计算器中，一个 CPU 作为主机工作，第二 CPU 作为从机工作。从机 CPU 的结果只能用于比较主机 CPU 的结果。只有主机 CPU 可以将结果写到数据 / 地址总线上或写入 CPU 寄存器中。

[0009] 本发明的优点在于可以将主机功能交替地分配到至少两个执行单元上，并由此交替使用在时钟同步模式下运行的双核或多核计算器的核结果。由此在考察特定的边界条件下即使在识别到冗余算出的结果之间存在差异之后也能保持处理器系统的有限运行。这尤其在实时应用中是非常有利的，在实时应用中不是在每个运行状态下都期望基于处理器错误而断开系统。

[0010] 在优选的实施例中给出另一个优点，即可以在处理器系统的执行单元中查找到错误，停用该出现错误的执行单元，并且继续运行具有未出现错误的执行单元的系统，直到达到对断开不重要的系统状态，或超过事先确定的、在该模式下的最大运行持续时间。

[0011] 优选地，描述了一种用于控制具有至少两个执行单元和一个比较单元的计算器系统的方法，该计算器系统工作在时钟同步的模式中，并且将至少两个执行单元的结果相比较，其特征在于，在通过比较单元识别出错误之时或之后在至少一个执行单元上运行针对该执行单元的错误识别机制。优选地，描述了一种方法，其特征在于，在通过比较单元识别出错误之时或之后中断在至少两个执行单元上的当前的指令序列，并在至少两个执行单元上运行错误识别机制。优选的描述了一种方法，其特征在于，在通过比较单元识别出错误之时或之后中断在恰好一个执行单元上的当前的指令序列，并在该执行单元上运行错误识别机制，在运行错误识别机制的持续时间期间断开至少两个执行单元的比较单元，并且在至少一个另外的执行单元上继续处理按照规定的程序序列。优选的描述了一种方法，其特征在于，如果错误识别机制没有识别出错误，则在运行完错误识别机制之后继续执行按照规定的程序流程。优选的描述了一种方法，其特征在于，在一个执行单元上找到错误之时或之后断开该出现错误的执行单元。优选的描述了一种方法，其特征在于，停用所述比较单元。优选的描述了一种方法，其特征在于，在停用至少一个部件时产生错误信号，该错误信号将提供给应用。优选的描述了一种方法，其特征在于，在出现错误之后仅用一个执行单元进行的运行在时间上有限，而且最迟在事先确定的时间结束之后就断开所述计算器系统。优选的描述了一种方法，其特征在于，所述断开在事先确定的时间结束之前就已经通过由应用产生的信号断开。优选地，描述了一种用于控制具有至少两个执行单元和一个比较单元的计算器系统的装置，该计算器系统工作在时钟同步的模式中，并且将至少两个执行单元的结果相比较，其特征在于，具有这样构成的装置，在通过比较单元识别出错误之时或之后在至少一个执行单元上运行针对该执行单元的错误识别机制。优选的描述了一种装置，其特征在于，具有用于取消至少两个执行单元的时钟同步的耦合并向任意一个执行单元分配主机功能的装置。优选的描述了一种装置，其特征在于，具有用于存储用于执行单元的错误识别机制的装置。优选的描述了一种装置，其特征在于，具有用于在需要时向至少一个执行单元输入针对错误识别机制的指令和 / 或程序的装置。优选的描述了一种装置，其特征在于，具有用于停用比较单元的装置。

附图说明

[0012] 其它优点和优选实施方式从权利要求的特征以及说明书中得出。

[0013] 图 1 示出具有主机 CPU 和从机 CPU 的双核处理器，

[0014] 图 2 示出具有两个系统接口的双核处理器，

[0015] 图 3 示出双核处理器以及比较单元的附加输入信号，

[0016] 图 4 示出双核处理器以及比较单元的附加错误信号，

[0017] 图 5 借助流程图示出用于在处理器系统中处理错误的第一方法，

[0018] 图 6 借助流程图示出用于在处理器系统中处理错误的第二方法。

具体实施方式

[0019] 图 1 示出处理器系统 W100，其具有多个执行单元 W110a、W110b 如双核计算器以及优选用 HW 实现的比较单元 W120。该处理器系统运行在时钟同步模式。在该运行方式中，优选在每个时钟周期之后比较执行单元的结果。执行单元在此可以实现为处理器 / 核 /CPU，也可以实现为 FPU（浮点单元）、DSP（数字信号处理器），协处理器或 ALU（算术逻辑单元），并分别具有任意数量的所属的寄存器组。在此恰好有一个执行单元通过中断或释放单元 W130 与系统接口 W140 连接，或者直接与处理器系统的数据 / 地址总线连接。该执行单元产生唯一的要在处理器系统中继续处理的结果。因此，与系统接口 W130 或者处理器系统的数据 / 地址总线连接的执行单元称为主机。至少一个另外的执行单元的输出信号只输入比较单元 W120，并在该比较单元中用于对主机的输出信号进行似然性检查。比较单元 W120 通过信号 W125 控制中断或释放单元 W130，该信号 W125 是代表该比较的信息。这种恰好具有两个执行单元并且执行单元作为 CPU 实现的系统在现有技术中作为双核微控制器公知。

[0020] 与公知的、在时钟同步模式下运行的双核微控制器不同，在本发明的第一实施例中，在保持特定的边界条件下即使冗余的执行单元的输出信号之间存在差异也将一个值写入寄存器或存储器中或输出到数据 / 地址总线上。但在此主机功能不是固定地分配给一个执行单元，而是可以分配给不同的执行单元。该分配根据静态确定的机制进行或者动态确定。

[0021] 在图 2 所示的第二实施例中，处理器系统 W101 包含相对于图 1 的处理器系统 W100 扩展了的比较单元 W121、两个中断或释放单元 W130a、W130b，执行单元 W110a、W110b 通过该中断或释放单元 W130a、W130b 与系统接口 W140a、W140b 或者与数据 / 地址总线连接，而且该中断或释放单元 W130a、W130b 由比较单元通过信号 W126a、W126b 控制。在此在整个处理器系统中总是只向一个执行单元分配主机功能，即总是最多只有一个执行单元与系统接口或数据 / 地址总线连接。主机功能的分配或主机功能的切换在此通过控制中断或释放单元 W130a、W130b 进行。这些中断或释放单元由比较单元 W121 根据至少两个执行单元的输出信号的比较结果来控制。

[0022] 在图 3 所示的实施例中，主机功能的切换通过比较单元 W122 进行，该比较单元根据至少一个输入信号 W160 或者该输入信号的标志通过借助信号 W126a 和 W126b 来控制中断或释放单元 W130a、W130b，在至少两个执行单元 W110a、W110b 之间切换主机功能或者断开所述系统。

[0023] 输入信号 W160 或者该输入信号的标志可以根据时间或指令计数器（例如每 10 个

时钟周期或每 10 个指令）来产生，优选通过专用硬件产生，或者可以通过操作系统例如根据对运行时对象的调度来产生（例如可以在每次调用运行时对象时切换或者在每个操作系统周期时切换），或者可以根据程序代码中的标志，或者可以通过中断或中断请求单元的信号产生，或者可以根据对程序和 / 或数据存储器中的特定存储区的访问而产生。

[0024] 主机功能的分配或切换可以根据上述条件之一，根据比较单元 W122 的比较结果，或者多个这些条件的组合来进行。

[0025] 在执行单元的输出信号之间存在差异时，由比较单元产生内部错误信号。根据通过信号 W160 通知给比较单元的系统状态，可以不断开该系统而将主机功能从一个执行单元切换到另一个执行单元上。对于输出信号的每一个其它差异重复该过程，即主机功能分别划分给其它执行单元。在此要注意，主机通过各自的系统接口 W140 与比较结果无关地传递其结果。比较单元只检测一个差异，但是不禁止各主机的写入。在比较单元 W122 中可以包含其它根据错误计数器在超过预定个数的错误之后断开所述系统的装置，所述错误计数器对已识别出的差异计数。该系统还可以如图 4 所示通过比较单元 W123 产生外部错误信号 W170。该错误信号可以在外部单元、操作系统或应用中进行分析，并通过信号 W160 通知比较单元 W123 应当断开系统。这些实施方式的共同点是 :在出现错误时处理器系统不是立即断开而是继续运行。通过交换主机功能，在一个执行单元中出现永久错误时也可以使至少每个第二结果是正确的。根据不同的应用功能，这足以使系统在一定的时间期间以足够的功能品质继续运行。

[0026] 很多在汽车中用于制备信号和调节机械系统的功能都具有鲁棒的设计，即短暂的干扰（例如由于 EMV 射线或由于干扰参数影响调节回路）在这样的系统中不会具有对安全很关键的影响，并由此可以容忍。但是保持时间较长的干扰不能被这样的“鲁棒”系统所容忍。在这样的鲁棒功能中，处理器系统不必在出现错误之后、即在通过比较单元识别出差异之后立即断开。在作用持续时间较短的瞬时错误原因中，该错误在下次调用时大多不再出现。通过交替使用执行单元的输出信号或在具有多个执行单元的处理器系统中交替分配主机功能，在一个执行单元中的永久错误也不会长时间影响应用，而是断断续续地影响。由此在错误情况下，直到一个错误被清楚地识别为永久错误，或者应用系统达到适于断开的系统状态时，才断开处理器系统。

[0027] 在另一个实施例中，在识别出至少两个执行单元的输出信号之间存在差异时中断在所有执行单元上对当前指令序列（程序块，任务）的处理。在所有执行单元中不是中断指令序列，而是处理错误识别例程如 BIST(Built-In-Self_Test，内建自测试）或者基于软件的自测试。通过将错误识别例程的结果与存储的参考值进行比较，可以识别并找到错误。在识别和找到错误时，断开出现错误的执行单元。没有出现错误的单元继续运行，直到达到确定要断开的系统状态为止。出现错误的执行单元的断开是这样来进行的，停用比较单元并且分配给该执行单元的中断或释放单元 W130a 或 W130b 不允许该执行单元连接到系统接口或地址 / 数据总线，或这样来进行，不向该执行单元输入指令、数据和 / 或时钟信号。

[0028] 为了停用比较单元存在各种可能。一方面可以向比较单元发送信号，利用该信号启动或停用比较逻辑或比较功能。为此在该比较单元中添加附加的逻辑电路，该逻辑电路可以根据这样的信号启动或停用比较功能。另一个可能在于，不向比较单元输入要比较的数据。第三种可能在于，在系统层面上忽略按照图 4 的比较单元 W123 的错误信号 W170，中

断错误信号 W170 本身，或者在这种情况下不将比较结果用于产生控制信号，如图 2 和图 3 所示的信号 W126a 和 W126b。所有这些可能的共同点在于：它们在系统中产生一个状态，在该状态下执行单元的输出信号何时不同是没有什么意义的。如果通过某种措施在比较单元或其输入或输出信号中达到该状态，则将该比较单元称为消极或停用的。

[0029] 如果在运行错误识别机制时没有在执行单元中找到错误，则在时钟同步中启动下个任务。如果又识别出输出信号之间存在差异，则再次进行上述过程，但是必须限制重复的次数 n。该限制可以根据应用的错误容许时间来进行。如果在重复 n 次之后又识别出错误，则立即断开该系统。

[0030] 根据图 4 的另一个优选实施例基于具有双核体系和优选用 HW 实现的比较单元的处理器系统，该处理器系统除了时钟同步运行模式之外还可以在至少一个第二运行模式中运行，在该第二运行模式中两个执行单元 W110a、W110b 同时处理不同的程序、程序段或指令。如果该处理器系统工作在时钟同步运行模式下，并且比较单元确定结果之间存在差异，则在例如 W110b 的、恰好没有与系统接口或数据 / 地址总线连接的执行单元中中断当前程序段或运行时对象（下面称为“任务”）的执行，并启动错误识别例程（例如 BIST）。例如 W110a 的另一个执行单元继续处理当前任务，但是其具有 50％的统计学错误概率。如果在 W110b 上的错误识别例程直到在 W110a 上运行的任务结束时才在 W110b 中识别出错误（例如通过与存储的参考值相比较），则 W110b 断开，而 W110a 继续工作在单模式（没有比较，或具有被停用的比较单元）下，直到整个系统达到对于断开不重要的状态为止。然后断开微处理器系统。如果 W110b 直到 W110a 的任务结束时都没有找到错误，则下个任务又在时钟同步模式下开始，但这次 W110b 与系统接口或数据 / 地址总线连接。如果不再出现差异，则在前面的任务中出现的差异很有可能是瞬时错误的影响。如果又出现差异，则这一次在执行单元 W110a 中中断当前任务并启动错误识别例程（例如 BIST）。该过程一直重复到下个（或者可配置个数的）“调度回合”（操作系统周期）开始时。如果然后还存在结果之间的差异，尽管没有找到错误，仍然可以推断出是没有通过错误识别机制找到的永久错误，并完全断开微处理器系统。

[0031] 在图 5 中示出用于在执行单元的输出信号之间出现差异之后控制处理器系统的第一方法。

[0032] 在步骤 510，在至少两个执行单元中处理相同的指令或程序段。

[0033] 在步骤 520 中，对该至少两个执行单元的输出信号就一致性进行比较。如果输出信号相同或者在限定的容差带中，则重新开始步骤 510，这一次用新的程序段或指令和 / 或数据开始。如果在步骤 520 识别出输出信号之间存在差异，则下一步执行步骤 530。

[0034] 在步骤 530 中，中断当前的程序处理，并在所有执行单元上执行错误识别例程。在此必须中断该执行单元与系统接口或数据 / 地址总线的连接。

[0035] 在步骤 540 中，错误识别例程的结果分别与参考值相比较，该参考值与错误识别例程的程序代码一起存储。如果在该比较中出现差异，则将具有导致差异的比较结果的执行单元标识为有错，并在下一步执行步骤 550。如果没有出现差异，则重新开始步骤 510，这一次用新的程序段或指令和 / 或数据开始。

[0036] 在步骤 550 中，停用标识为有错的执行单元以及比较单元。执行单元的断开例如可以这样进行，不向该执行单元输入指令、数据和 / 或时钟信号，或者中断该执行单元与比

较单元以及与系统接口或数据 / 地址总线的连接。

[0037] 在步骤 560，处理器系统继续用剩下的、没有错误的执行单元工作。在具有两个执行单元的处理器系统中，这意味着单核运行。该单核运行在与安全有关的系统中是时间有限的。

[0038] 在步骤 570 中，在达到断开条件之后断开处理器系统或者切换到限定的安全状态，该断开条件例如是超过针对单核运行的时间限制。

[0039] 在图 6 中示出用于在执行单元的输出信号之间出现差异之后控制处理器系统的另一种方法。

[0040] 在步骤 605 中，主机功能从第一执行单元转换到第二执行单元上。

[0041] 在步骤 610 中，在至少两个执行单元中处理相同的指令或程序段。

[0042] 在步骤 620 中，对该至少两个执行单元的输出信号就一致性进行比较。如果输出信号相同或者在限定的容差带中，则重新开始步骤 610，这一次用新的程序段或指令和 / 或数据开始。如果在步骤 620 识别出输出信号之间存在差异，则下一步执行步骤 630。

[0043] 在步骤 630 中，在至少一个执行单元，而且至少在与系统接口或数据 / 地址总线连接的执行单元上继续执行当前的程序序列。在至少一个另外的执行单元上执行错误识别例程。在此必须停用比较单元。

[0044] 在步骤 640 中，错误识别例程的结果分别与参考值相比较，该参考值与错误识别例程的程序代码一起存储。如果在该比较中出现差异，则将具有导致差异的比较结果的执行单元标识为有错，并在下一步执行步骤 650。如果没有出现差异，则重新开始步骤 605，这一次用新的程序段或指令和 / 或数据开始。

[0045] 在步骤 650 中，断开标识为有错的执行单元。这例如可以这样进行，不向该执行单元输入指令、数据和 / 或时钟信号，或者中断该执行单元与比较单元以及与系统接口或数据 / 地址总线的连接。

[0046] 在步骤 660，处理器系统继续用剩下的、没有错误的执行单元工作。在具有两个执行单元的处理器系统中，这意味着单核运行。该单核运行在与安全有关的系统中是时间有限的。

[0047] 在步骤 670 中，在达到断开条件之后断开处理器系统或者切换到限定的安全状态，该断开条件例如是超过针对单核运行的时间限制。

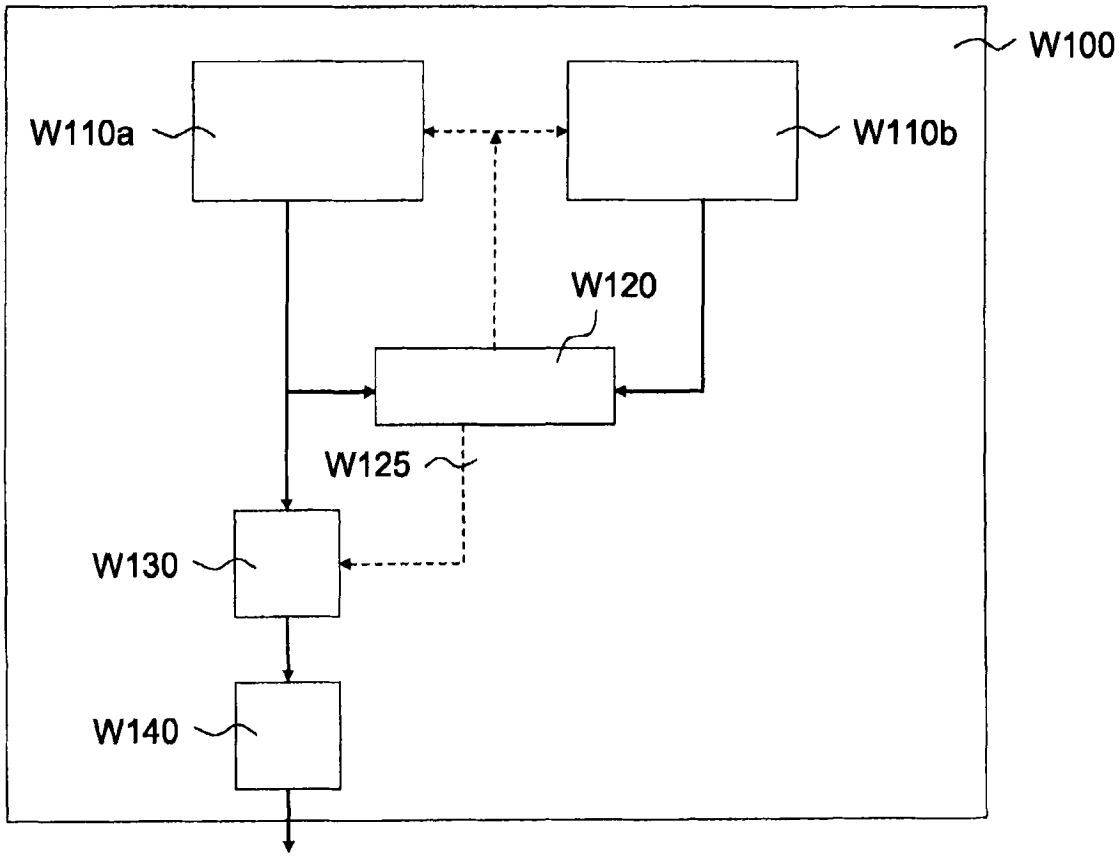


图 1

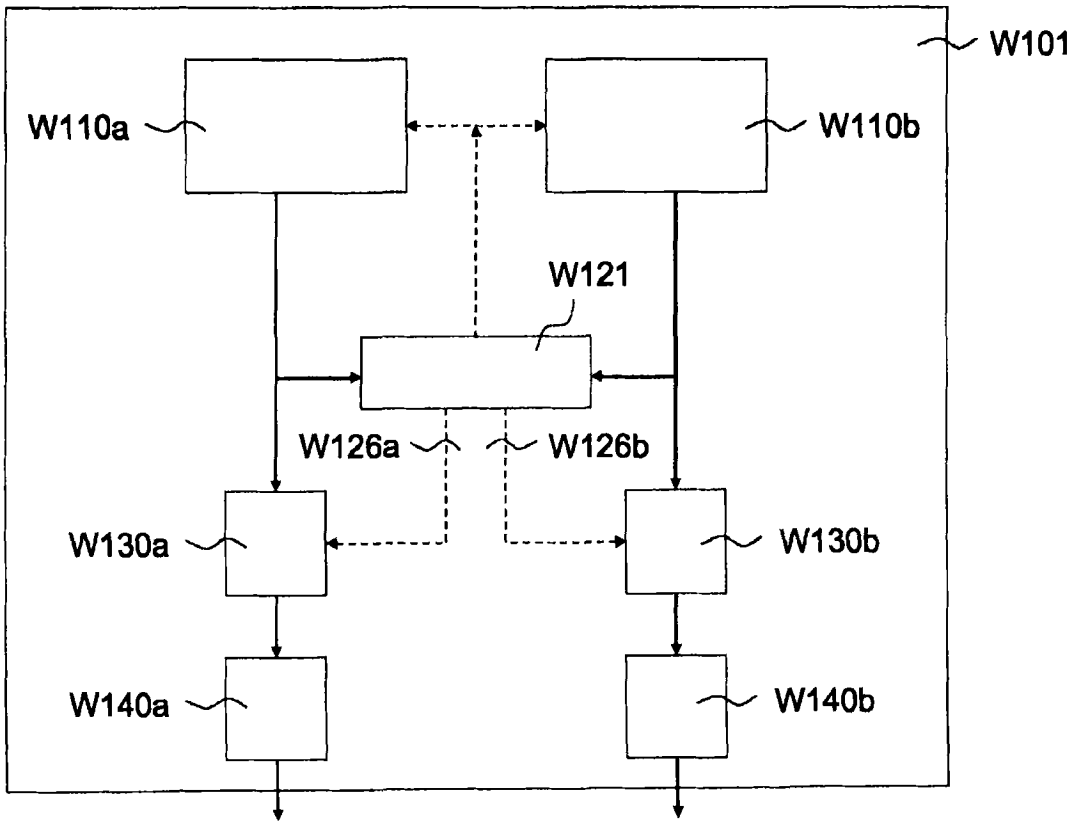


图 2

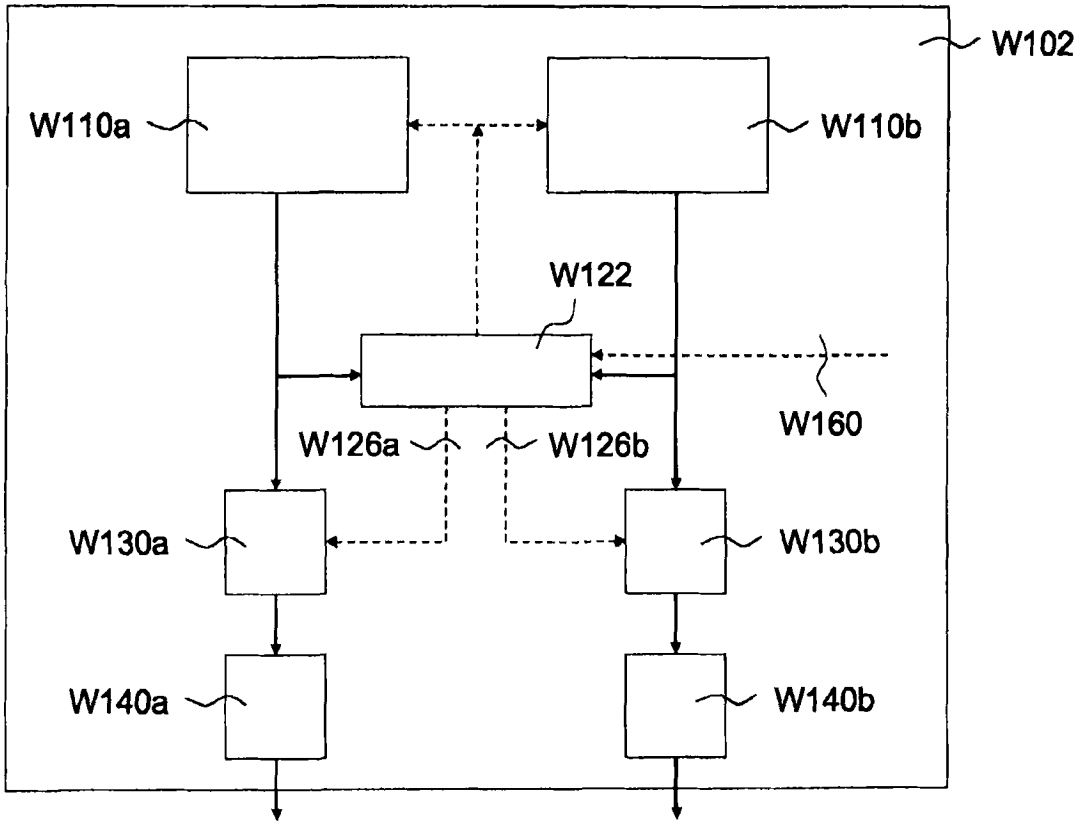

W102
W110a
W110b
W122
W160
W126a
W126b
W130a
W130b
W140a
W140b

图 3

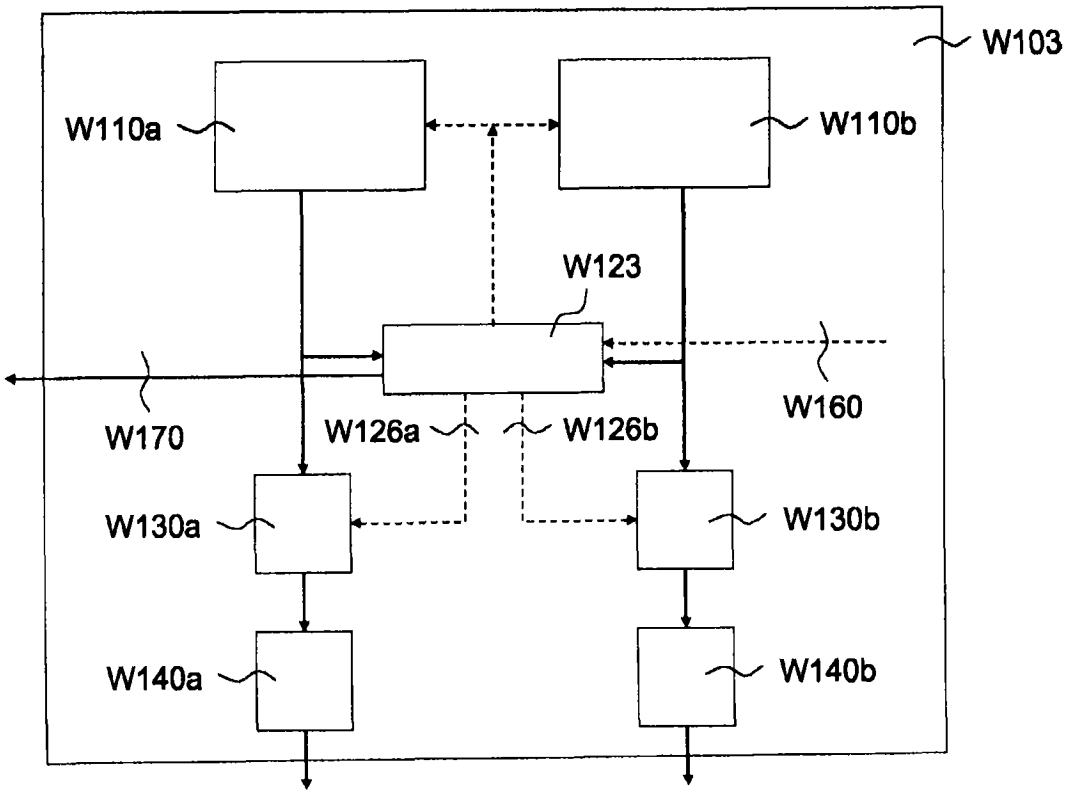

W103
W110a
W110b
W123
W160
W170
W126a
W126b
W130a
W130b
W140a
W140b

图 4

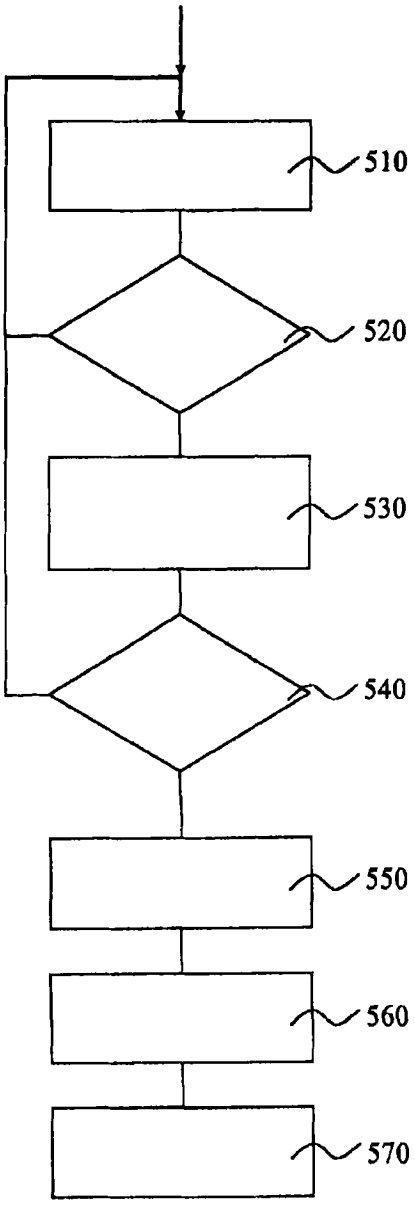


图 5

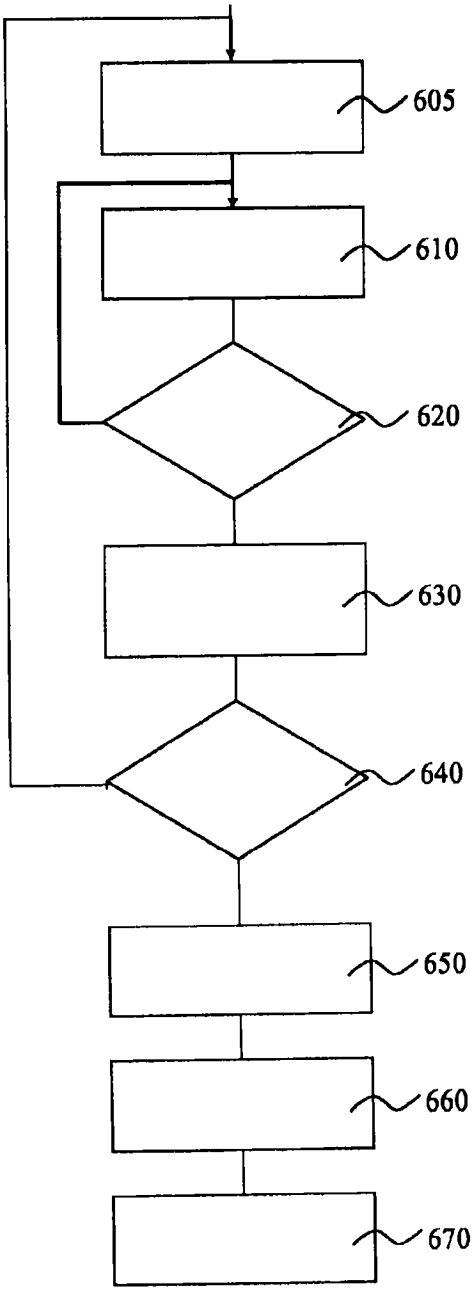


图 6