

[19]中华人民共和国国家知识产权局

[51]Int. Cl⁶

G06F 11/00

G06F 12/02

H04L 12/56

[12] 发明专利说明书

[21] ZL 专利号 94194287.2

[45]授权公告日 1999 年 10 月 13 日

[11]授权公告号 CN 1045675C

[22]申请日 94.11.23 [24]颁证日 99.7.16

[21]申请号 94194287.2

[30]优先权

[32]93.11.26 [33]SE [31]9303932

[86]国际申请 PCT/SE94/01119 94.11.23

[87]国际公布 WO95/14970 英 95.6.1

[85]进入国家阶段日期 96.5.24

[73]专利权人 艾利森电话股份有限公司

地址 瑞典斯德哥尔摩

[72]发明人 T·L·郝林

[56]参考文献

EP0323310 1989. 7. 5 H04L011/20

US4049956 1977. 9.20 G11C29/00

审查员 夏国红

[74]专利代理机构 中国专利代理(香港)有限公司

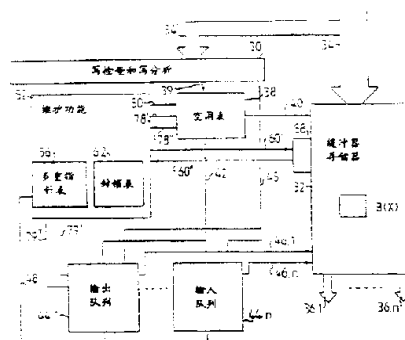
代理人 王 勇 王忠忠

权利要求书 7 页 说明书 17 页 附图页数 5 页

[54]发明名称 容错队列系统

[57]摘要

本发明涉及在带有一个或多个交换端口的公共缓冲器存储器(32)的包交换中缓冲数据的一容错队列系统,所述系统使用一些标识缓冲器存储器中的存储位置的指针,同一指针的多重拷贝通过使用一多重指针表(56)以受控方式管理,在多重指针表中记录有包含在系统中各自指针的拷贝的数目。维护功能(52)检查一指针的拷贝数是否与在多重指针表(56)中为各自指针记录的数值一致。维护功能(52)周期地检查所有指针和附属的缓冲器位置,每次一个。此前要执行初始化过程,其间通过从指针流中滤除指针而将其从运行中取出,直到该队列系统清空所谈论的指针的每一拷贝,指针流中的指针从为在队列中的排队指针到存储器的输出端口的输出队列表(44)返回到空闲表(38),然后比较所谈论的指针的返回拷贝数与在多重指针表(56)中的数值。维护功能然后可以按照一个或多个运行方式执行循环控制。



专利文献出版社出版

ISSN 1008-4274

权利要求书

1. 为执行从一联机维护的空闲地址表 (6) 往存储器中自由存储位置执行写操作 (4) 而在存储器 (2) 中选择存储位置, 和收集 (8) 早先存储在存储器中的数据而计算地址的一队列系统, 其特征在于, 存储位置暴露于一测试过程 (16), 其在系统运行期间在空闲周期中周期地对每一存储位置执行, 和其结果用于维护空闲表 (6), 从系统控制的系统使用中释放为测试而选择的地址。

2. 按照权利要求 1 的系统, 其特征在于, 存储位置 (34) 的测试在等待其由系统使用释放之后执行, 和仅在存储位置通过该测试才执行到系统使用的返回。

3. 按照权利要求 1 的系统, 其特征在于, 指向存储位置的地址由把要测试的存储位置的内容拷贝到一自由而无错误的存储位置而释放, 于是地址计算重新导向使用新的存储地址, 最终只将经受住测试的受测试位置返回给系统使用。

4. 按照权利要求 1 的系统, 其特征在于, 未通过测试的存储位置在当包含所谈论的存储位置的整个存储器块从系统使用释放时重新测试。

5. 按照权利要求 1 的系统, 其特征在于, 使为执行一确定的存储位置的测试的存储器测试逻辑运行

在第一方式, 其中所有的地址以由测试逻辑控制的脱机测试方式测试, 在此期间, 带可改正错误的存储位置的数目被计数和对其中存在的不可改正的错误加以标识, 和

在第二联机方式，在此期间存储器测试逻辑仅检查正被测试的存储器位置的数据，而其它存储位置被送去以在正常运行时的数据运行，其地址由系统控制。

6. 为在带有一个或多个交换输出端口的公共缓冲器存储器 (32) 的包交换中缓存数据的队列系统，所述系统使用一些标识缓冲器存储器中存储位置的指针，和在其中这些指针在不同的逻辑表 (38、44、56、62) 之间移动来指明在所指的缓冲器位置的数据要执行的操作，其特征在于一维护功能 (52)，其连续地保证该系统对每一在缓冲器存储器 (32) 中有效的包位置包含一个而且只包含一个指针。

7. 为在带有一个或多个交换输出端口的公共缓冲器存储器 (32) 的包交换中缓存数据的队列系统，所述系统使用一些标识缓冲器存储器中存储位置的指针，和在其中这些指针在不同的逻辑表 (38、44、56、62) 之间移动来指明在所指的缓冲器位置的数据要执行的操作，其特征在于，同一指针的拷贝以一受控方式通过使用一多重指针表 (56) 而管理，在多重指针表中记录有包含在系统中的各自指针的拷贝数目，和在其中一维护功能 (52) 检查一指针的拷贝数是否与为该各自的指针记录在多重指针表 (56) 中的数值相符。

8. 按照权利要求 7 的队列系统，其特征在于，除多重指针表 (56) 之外，还使用下面的各自指针类别的逻辑表：空闲指针的空闲表 (38)、在输出端口队列中的排队指针的输出队列表 (44)、和封锁指针的封锁表 (62)。

9. 按照权利要求 8 的队列系统, 其特征在于, 维护功能 (5 2) 周期地检查所有指针和附属的缓冲器位置, 每次一个, 以一初始化过程先导, 其中通过从指针流中滤除该指针将其从运行中取出, 这些指针从输出队列表 (4 4) 返回到空闲表 (3 8), 直到该对列系统清空所谈论的指针的每一拷贝, 然后比较正被讨论的指针的拷贝的返回数与在多重指针表 (5 6) 中的值。

1 0. 按照权利要求 9 的队列系统, 其特征在于, 该维护功能具有第一运行方式, 其在初始化过程后, 并或者由下述事实激发, 即作为所述初始化过程的结果而发现大于一确定限制数目的指针数, 它们各自的拷贝数与在多重指针表中为所谈论的指针的数值不符, 或者由一外部信号激发, 而把该指针返回到空闲表 (3 8)、把在多重指针表 (5 6) 中的数值复位到无多重拷贝、和在封锁表 (6 2) 中表明无指针被封锁。

1 1. 按照权利要求 6 或 9 的队列系统, 其特征在于, 该维护功能具有第一运行方式, 其在初始化过程后, 并或者由下述事实激发, 即作为所述初始化过程的结果而发现大于一确定限制数目的丢失或多重指针, 或者由一外部信号激发, 而把每一指针的一个拷贝返回到空闲表 (3 8), 和在封锁表 (6 2) 中指明无指针被封锁。

1 2. 按照权利要求 1 0 的队列系统, 其特征在于维护功能 (5 2) 仅有一个另外的运行方式。

1 3. 按照权利要求 1 1 的队列系统, 其特征在于维护功能 (5 2) 仅有一个另外的运行方式。

1 4. 按照权利要求 1 2 或 1 3 的队列系统, 特征在于, 维护功能有一第二运行方式, 其由第一维护方式运行遍历所有指针而激发, 然后初始化过程再次对每一指针执行, 每次一个, 而缓冲器存储器测试以低的优先级对由所谈论的指针标识的非封锁存储器位置排队, 而所有

其它非封锁的存储器位置在测试时以系统数据进行，和其中存储器测试结果用于返回该指针到空闲表（38），如果该缓冲器存储器测试给出一正确结果的话；而如果该缓冲器存储器测试指出由该指针定义的区域故障的话，封锁该指针使其不在系统中使用，

被封锁的指针暴露给和第一缓冲器存储器测试同样类型的第二缓冲器存储器测试，其由在队列中维护的每一测试数据操作规定，只要系统数据存储在包含由被封锁的指针标识的位置的缓冲器存储器段的某处，

第二测试的结果用于返回该指针到空闲表（38），如果该测试给出一正确结果的话；而被封锁的指针在超过一第二限制数的数目的连续测试中认可后首先返回到空闲表。

15. 按照权利要求8或9的队列系统，其特征在于，该维护功能（52）有一些不同的运行方式。

16. 按照权利要求11的队列系统，其特征在于，该维护功能有一由第一方式运行遍历所有指针而激发的第二运行方式，然后该初始化过程对这些指针再次执行，

一简单的写和读测试排队，等待对由其拷贝数与在多重指针表中记录的数值不符的指针所标识的缓冲器存储器位置的执行，该写和读测试的结果用于将所谈论的指针返回到空闲表，如果该测试给出一正确结果的话；但如果该测试指出由所谈论的指针标识的区域故障的话，封锁该指针使其不在系统中使用，

17. 按照权利要求16的队列系统，其特征在于，该维护功能有一第三运行方式，其由第二方式运行遍历所有指针而激发，于是该初始化过程对非封锁指针再次执行，

为检测在第二运行方式中由简单的写和读测试未检测到的故障类型的缓冲器存储器测试排队，等待在由初始化过的指针所标识的缓冲器存储器位置执行，所述存储器位置以测试数据进行，而所有其它非封锁缓冲器存储器位置在对每一封锁的指针测试期间以系统数据进行，

该测试的结果用于返回指针到空闲表，如果该缓冲器存储器测试给出一正确结果；但如果该缓冲器存储器测试指出在由该指针标识的区域内故障的话，封锁该指针使其不在系统中使用，

18. 按照权利要求17的队列系统，其特征在于，该维护功能有一第四运行方式，其由第三方式运行遍历所有指针而激发，于是早先封锁的指针通过对所有指针周期地执行所述初始化过程，一次一个，也被接受测试，

和在第三方式中同样类型的缓冲器存储器测试应用于非封锁指针，而早先封锁的指针暴露给同一类型的由在队列中维护的每一测试数据操作规定的缓冲器存储器测试，只要系统数据存储包含有由所谈论的早先封锁的指针所标识的位置的缓冲器存储器段的某处，

该测试的结果以和第三方式中同样的方式用于非封锁指针和早先封锁的指针的未验收测试，而早先封锁的指针在经过由一限制数定义的一些连续测试合格后首先返回空闲表。

19. 按照权利要求8的队列系统，其特征在于，多重指针和封锁指针所用的表合而为一。

20. 在为执行从一联机维护的空闲地址表往存储器中自由存储位置执行写操作而在存储器中选择存储位置，和为收集早先存储在存

存储器中的数据而计算地址的一队列系统中测试所述存储位置的一种方法，包括

从系统控制的系统使用中释放为测试选择的地址，

把选择的存储位置暴露给在系统运行时的空闲周期周期地对每一存储位置测试的过程，和

为维护所述空闲表使用所述测试过程的结果。

2 1. 按照权利要求 2 0 的方法，包括

在等待存储位置由系统使用释放以后执行对其的测试，和
仅当该存储位置通过测试后返回该存储位置到系统使用。

2 2. 按照权利要求 2 0 的方法，包括

通过把要测试的存储位置的内容拷贝到一自由而无错误的存储位置而释放指向该存储位置的地址，

重新引导由新存储位置使用的地址计算，和

仅当被测试的存储位置通过测试后返回其到系统使用。

2 3. 按照权利要求 2 0 的方法，包括

当包含所谈论的存储位置的整个存储器块从系统使用中释放时重新测试未通过测试的存储位置。

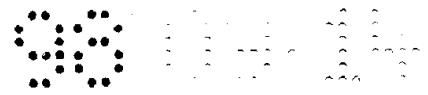
2 4. 按照权利要求 2 0 的方法，包括

a) 执行一确定存储位置的测试，这是通过
在第一运行方式中以由测试逻辑控制的脱机测试方式测试所有地址，

在所述测试期间计数有可修复错误的存储位置的数目，和

标明不可修复错误，而进行的；

b) 在第二联机方式中只检查正被测试的存储器位置的数据，和



强迫其它存储位置以正常运行时的数据运行，其地址由系统控制。

说明书

容错队列系统

发明的技术领域

本发明按照第一方面涉及一个为执行从一联机维护的空闲地址表往存储器中自由存储位置写操作而在存储器中选择位置的队列系统。该队列系统也为收集早先存储在存储器中的数据计算地址。

按照第二和第三方面，本发明涉及一个在带有用于一个或多个交换端口的公共缓冲器存储器的包交换机中缓冲存储数据的队列系统。该系统使用一些表示缓冲器存储器中存储位置的指针。这些指针在不同的逻辑表间移动来指明对在所指的缓冲器位置的数据要执行的操作。

本发明进一步还涉及分别测试与上面指出的种类的队列系统有关的存储位置和指针的方法。

相关技术的叙述

当在计算机和其它设备中使用存储器时，地址计算确定为存储一定数据要使用的物理位置。最简单的直接寻址、确定和计算要使用哪一地址是在程序编译时执行的，或与具有固定程序的设备的硬件设计有关。间接寻址和链接表是更为灵活的寻址方法的例子。这里，地址信息存储在其它地址位置或计数器等等中。因此地址计算并不固定，而是在执行时变化，以适应存储的数据的数量和格式。

带直接寻址的简单的先进先出（F I F O）存储器最常以一种增量写和读地址的计数器实现，或在它们最简单的形式中以在移位寄存

器中的固定地址实现。关于动态间接寻址，使用也存储在存储器中的地址指针。在系统起动时，这些指针被初始化以便得到一组系统以之工作的有效的指针。因而系统的功能依赖于该指针的初始化在运行时不被破坏。指针值中的位错误导致系统中持续的故障。对可靠性要求很高的连续操作的系统这是一个问题。

大半导体存储器传统地装备有冗余以改善制造产量。此间的冗余指的是行和列的备用电路，其可以连接到电路上以取代正常的电路。这可以通过熔断熔丝或在工厂中永久性地对电路重新布线的其它方法获得。

进行测试以决定需要修复的电路。如果出现制造错误而且它们能够通过把备用电路连接到电路上而修复，则确定制造错误的类型并进行修复。

对不带冗余的半导体存储器，地址和存储器中的物理位置的映射是在设计该存储器时确定的。对带冗余的半导体存储器，这种映射部分地在设计时设定，而当有错误的存储器位置要进行修复时在制造时完成。

在盘式存储器的场合，在把该盘投入运行前为确定在盘上可以使用的区段的格式化操作时要使用测试过程。其目标是能够使用带有坏区段的盘。所用的错误假设是盘上一定的位置可能由于制造不完美而损坏。区分好的和坏的扇区可以在格式化过程中进行且可认为不需改变。在格式化过程中盘地址错误的风险不需考虑。

在一定的系统中使用错误校正码（ECC）来获得对运行时的位错误的容错性能。这些在运行时出现的错误可以是软的或硬的，亦即暂时干扰或固定错误。在每一字中能够被改正的位的数目是确定的。

最常使用的码可以改正单位错误和检测双位错误。最近表明，错误改正码和备用行的结合提供一容错系统，其具有很高程度的半导体存储器制造错误的修复性，参见 I E E E 计算机学报，第 4 1 卷 9 号，1992 年 9 月，1078 - 1087 页。

设备在离开制造之前对它的修复在工厂的生产线上需要测试装置和熔丝编程（为影响其功能而对电路连接的持久改变 - 也叫熔丝或抗熔丝）及包含熔丝在内的制造工艺。这意味着由于延长了的制造周期而增加了成本。对于修复操作的成功程度由下述事实限制，即它们是基于使用备件行和列，和相对于处理零星单个的错误的灵活性。

使用纠错码处理分离的单个错误。然而为充分的效率起见，需要与带冗余行的一些种类的系统结合。按照现有技术状态这意味着熔丝编程和在电路离开制造厂前测试。

美国专利 4 0 4 9 9 5 6 提供现有技术状态的一个例子，它涉及运行时错误检测和叙述工作在时分多路复用方式下的存储器的管理。到来的数据字暂时存储在一多级 (step) 存储器的各自的级中并为以后的修正而读出。进行测试时不需把存储器从正常运行中取出。该存储器由若干 n 级组成，它们周期地在 n 个时隙的一个重复扫描周期被寻址，在该时隙内二进制字写入各自的级中或从其中读出。写入存储器级的字在紧随的下一周期被读出。通过奇偶校验检测故障。

美国专利 3 8 6 3 2 2 7 叙述每一单个的电子控制元件的选择和测试，这些控制元件用于当存储器联机时存取在一核心存储器中不同的核心元件。因而存在一个存储器系统联机测试的问题，而不是存储包含在存储器中的元素的存储器的测试。也未叙述使用奇偶校验。

美国专利 5 0 1 6 2 4 8 叙述一用于包传输的排队系统，而未提及初始化或维护。

概述

本发明的第一目的是通过持续地维护在带有动态间接寻址的队列系统中的指针处理而提供容错性能，使得该系统总有能力复位一组正确的指针而独立于系统结束时的状态。

第二目的是获得对在系统运行时可能出现的存储器错误的容错性能。

本发明的第三目的是通过以一新的方式组织和使用冗余电路而改善大半导体存储器的制造产量，通过使用容错技术，带有小的制造错误的存储器可以以正确方式运行而不必拒绝。通过在测试过程适当地设定拒绝水平以便维持备件容量，存储器管理系统也可能从在运行时出现的错误中恢复。

按照本发明，上面提到的目的以下面的方式全部或部分地实现。

在按照以引言方式指出的第一方面的队列系统中，存储位置暴露于一测试过程，其在系统操作时的空闲周期对每一存储位置周期地执行。该过程的结果用于维护空闲表。为测试选择的地址从由系统控制的系统使用中释放。

在一个实施例中存储位置的测试是在等待其从系统的使用中释放后执行的，并仅在该存储位置通过测试后才执行返回到系统使用。

在另一实施例中，相对于存储位置的地址通过把要测试的存储位置的内容拷贝到一自由和无错误的位置而释放。于是地址计算重新导向使用该新的存储位置，最后仅在其通过测试后被测试的存储位置才返回到系统使用。

在再一个实施例中，未通过测试的存储位置在当包含所谈论的存储位置的整个存储器块从系统使用中释放时重新测试。

在再一个实施例中，为执行一确定的存储位置的测试的存储器测试逻辑以第一方式运行，其中所有的地址在由测试逻辑控制的脱机测试中测试，其间执行对带可修正错误的存储位置的计数和指明在其中存在的不可修正的错误，以及在第二方式的联机方式运行，其间存储器测试逻辑只检查被测试的存储器位置的数据，而其它存储位置被送去以在正常运行时的数据运行，其地址由系统控制。

在按照以引言方式指出的第二方面的队列系统中，维护功能连续地保证该系统对在缓冲器存储器中的每一有效包位置包含一个并且只包含一个指针。

在按照以引言方式指出的第三方面的队列系统中，同一指针的拷贝通过使用一多重指针表而以一受控方式处理，在该多重指针表中记录有包含在各自指针的系统中的拷贝数，和在其中维护功能检查一指针的拷贝的数目是否与在各自指针的多重指针表中记录的数值相符。

在第一实施例中，除多重指针表之外还使用下面各自的指针类别的逻辑表：空闲指针的空闲表、对输出端口队列中的排队指针的输出队列表、和被封锁指针的封锁表。维护功能周期地检查所有的指针和附属缓冲器位置，每次一个，由一初始化过程先导。在这一过程中，该指针通过从指针流中被滤除而从操作中取出，其从输出队列表返回到空闲表，直到该队列系统清空所谈论的指针的每一拷贝，于是比较返回的所谈论的指针的拷贝的数目与在多重指针表中的数值。

在这一实施例的第一进一步的发展中维护功能具有第一运行方式，其在初始化过程之后，并要么由下述事实激发，即作为所述初始

化过程的结果找到多于一确定限制数目的指针，其各自的拷贝数与所提到的指针在多重指针表中记录的数值不符；或者由一外部信号激发，而把这些指针返回到空闲表，把在多重指针表中的数值复位为无多重拷贝，和在封锁表中指明无指针被封锁。

在第二进一步的发展中该维护功能具有第一运行方式，其在初始化过程之后，并要么由下述事实激发，即多于一确定限制数目的丢失指针或多重指针作为所述初始化过程的结果而发现；或者由一外部信号激发，而把每一指针的拷贝返回到空闲表，和在封锁表中指明无指针被封锁。

该维护功能可能仅有一另外的运行方式，亦即第二方式。在该场合下，这一第二运行方式由第一维护方式遍历所有指针而激发。于是，对每一指针再一次执行初始化过程，每次一个，同时，缓冲器存储器测试以低的优先级在由所谈论的指针标识的非封锁存储器位置上排队。所有其它非封锁的存储器位置在测试时以系统数据进行。该存储器测试的结果用于给空闲表返回该指针，如果缓冲器存储器测试给出一正确的结果的话；但是如果缓冲器存储器测试指出由该指针定义的区域故障的话该指针在系统中被封锁。被封锁的指针暴露给第二缓冲器存储器测试，在队列中维护由每一测试数据操作调整的和第一个同样类型的该测试，只要系统数据存储包含由被封锁的指针标识的位置的缓冲器存储器段中的某处。第二测试的结果用于返回指针到空闲表，如果该测试给出正确的结果的话；而被封锁的指针在超过一第二限制数的数目的连续测试中被认可后首次返回到空闲表。

维护功能有一些不同的运行方式的情况中，它可以有由第一方式运行遍历所有指针而激发的第二运行方式，之后初始化过程再次对指

针执行。一简单的写和读测试排队在其拷贝数与在多重指针表中记录的数值不符的指针所标识的缓冲器存储器位置执行。写和读测试的结果用于将所谈论的指针返回到空闲表，如果该测试给出正确结果的话；但如果该测试指出由所谈论的指针标识的区域之内有故障的话，封锁该指针使其不在系统中使用。

第三运行方式由第二方式运行遍历所有指针而激发，于是初始化过程再次在非封锁的指针上执行。为检测在第二运行方式中由简单的写和读测试未检测到的类型的故障的缓冲器存储器测试被排队，以在由初始化过的指针标识的缓冲器存储器位置上执行。所述存储器位置由测试数据进行，而所有其它的非封锁的缓冲器存储器位置在测试每一封锁的指针期间以系统数据进行。该测试的结果用于返回该指针到空闲表，如果该缓冲器存储器测试给出一正确结果的话；但是，如果该缓冲器存储器测试指出在由该指针标识的区域有故障的话，封锁该指针使其不在系统中使用。

第四运行方式由第三方式运行遍历所有的指针而激发，于是同样早先封锁的指针通过对所有指针周期地、一次一个执行所述初始化过程被接受测试。和在第三方式中同一类型的存储器测试被应用于非封锁指针，而早先封锁的指针暴露给同一类型的缓冲器存储器测试，在队列中维护由每一测试数据操作调整的该缓冲器存储器测试，只要系统数据存储包含由所谈论的早先封锁的指针所标识的位置的缓冲器存储器段的某处。测试结果以和在第三方式下同样的方式用于为非封锁指针和早先封锁的指针的非接受测试，而早先封锁的指针在经过由一限制数定义的数目的连续测试合格后首先返回到空闲表。

按照本发明的的存储器管理系统在系统运行时提供动态的对制造错误的容错性能，不需在工厂中通过重新配置来测试和修复。在制造完成后进行测试以确定该存储器是否为打算的应用有足够的存储容量。

附图的简要说明

下面参考示于附图中的一实施例更严密地叙述本发明，附图中：

图 1 非常原理地示出一按照本发明的一个方面的队列系统；

图 2 部分地以框图形式示出一包交换的实施例，其中使用了本发明的不同方面；

图 3 表示包含在维护功能中的主运行方式的一流程图，其用于测试在按照图 2 的包交换中的地址指针和缓冲器存储器位置；和

图 4 - 7 表示图 3 的各自的方式的流程图。

实施例的详细说明

图 1 非常原理地示出为从一联机维护的地址空闲表 6 向存储器 2 中的自由存储位置上执行写操作（箭头 4）而在数据存储器 2 中选择存储位置、和为收集（箭头 8）早先存储在存储器 2 中的数据而进行地址计算的一个队列系统。最后提到的地址被传输（箭头 10）到存储器 2 中使用过的存储空间的表 12。

按照本发明的第一方面，存储器的存储器位置借助于测试逻辑 14 经历一测试过程，其周期地（箭头 16）在系统运行时的空闲周期中对每一存储位置执行。要被测试的位置借助于在表 12 中包含的地址选择（箭头 18），被选择的位置从被系统控制的系统使用中释放。测试结果用于维护空闲表 6（箭头 20）。

存储位置的测试可以在等待其从系统使用释放后执行。可选的另外方案为指向要被测试的存储位置的地址可以由拷贝这一要被测试的存储位置的内容到一自由而无错误的新存储位置而释放。地址计算随即被重新导向为使用这一新的存储位置。

在这两种场合下，仅当其通过测试后被测试的存储位置最终返回系统使用，亦即返回到空闲表 6。未经过测试的存储位置在当包含所谈论的存储器位置的所有存储器块从系统使用释放时重新测试。

为执行对给定存储器位置测试的存储器测试逻辑 1 4 可优选操作于两种方式。

在第一方式下所有的地址在一种由该逻辑控制的脱机测试中测试，在此期间带有可修正错误的存储位置的数目被计数，而其中存在的不可修正的错误被标识。

在该部件测试期间，一完整的存储器测试以例如所谓的“垮步方式”（march pattern）执行，参见例如“使用跨步测试来测试 S R A M S”（ADJvandeGoor, IEEE “计算机设计和测试”，9 3 年 3 月，第 1 0 卷，第 1 号）和“为可修复嵌入 R A M 用的内部自诊断”（S, Robert Trenter, 等, IEEE “计算机设计和测试”，9 3 年 6 月，第 1 0 卷，第 2 号）。这为消除不可操作电路而施行。由于按照本发明的队列系统提供一定量的容错性能，所以带有错误存储位置的电路也可以归为正确一类。因此错误数目在测试时被记录。如果错误数目小于设定的拒绝门限，则该部件被接受。

在第二方式中，作用于联机方式，逻辑 2 0 仅监视正被测试的到存储器位置的数据，而其它存储位置被强迫以在正常运行时的数据运行，其地址由系统控制。

现在更详细地叙述本发明的一个实施例。在一类包交换中，参照图 2，到达的包在分析功能 30 分析包的头部中的内容后被写入到对到达的包公共的缓冲器存储器 32 中的存储位置；缓冲器存储器对到达的包是共同的。缓冲器存储器 32 以到达包流 34 和一些离开该交换的包流的输出信道 36. 1、...、36. n 示意性地表示。到启动该功能的分析功能 30 的数据流由 34' 表示。

写操作由包含指向相应于在缓冲器存储器 32 中的空闲存储位置的地址指针的一地址表 38 指导执行。该地址表在下面也简短地称为“空闲表”。更具体地，分析功能 30 以一信号，按照箭头 39，指示空闲表 38 一方面为所谈论的包提供指针，和另一方面，在按照从空闲表 38 到缓冲器存储器 32 的箭头 40 指示的，把该包写入由该指针指向的缓冲器存储器后，把该指针传输到包含一些 F I F O 指针队列 44 的指针队列系统，每一输出 36 有一指针队列。

更具体地，该指针于是按照箭头 42 从空闲表 38 移动到一 F I F O 指针队列 44. n 用于交换输出 36. n，其为由分析该包的头部内容而指向的包而选择。指针队列的选择由从分析功能 30 到所谈论的指针队列的控制信号 45 执行。

到达的包的头部可能包含几个目的地址，亦即该包的内容可能送往几个交换输出。在这种场合，该指针被拷贝到在缓冲器存储器 32 中选择的存储位置以便获得和所谈论的交换输出的数目同样多的指针。下面也为在这样一组指向缓冲器存储器 32 中的同一存储位置的指针中的一个指针使用“多重指针”这一名称。这些多重指针被指向所谈论的输出的每一个指针队列。

当一个指针，亦即一单指针或一多重指针，作为在其FIFO队列44. n中第一个出现时，由该指针指向的数据包，按照箭头44. n和由该指针的指向，从在缓冲器存储器32中它的存储位置读取到所谈论的输出36. n中。

其后该指针按照箭头48和50通过维护功能52重新导向空闲表38。维护功能52在下面将更为周密地叙述，它监视要返回到空闲表38中的指针。更具体地，如果指针未被维护功能52选择测试的话，它将移动到空闲表中。

在维护功能52中同一指针的拷贝以一受控方式管理。更具体地，这通过使用包含在维护功能52中的一多重指针表56而执行，在该表中记录有包含在系统中的各自的指针的拷贝数。维护功能52也检查一指针的拷贝数与在多重指针表56中为各自指针记录的数值是否相符。

在下面对本发明的实施例更为详细的叙述中使用名称X表示正被测试的指针，这里X属于所有指针的集合。相应地，由指针X指向的在缓冲器存储器32中的缓冲器存储器位置用B(X)表示。

维护功能52以在下面更仔细地叙述的方式执行缓冲器存储器位置B(X)的测试。为此目的，指针X按照箭头60'传输到缓冲器存储器32的测试逻辑58。对缓冲器存储器位置B(X)的测试操作的结果按照箭头60"从测试逻辑58给维护功能52发信号。

从下面的叙述中同样更为仔细地看出，如果这一测试指出一存储器位置故障的话，则相应的指针将被封锁以在系统中使用，并提供给同样包含在维护功能52中的一封锁表62。多重和被封锁的指针的表也可能结合为一张表。

更具体地，该维护功能 5 2 周期地检查所有的指针和附属的缓冲器位置，每次一个，由一初始化过程先导。这里用“所有的”指针一词指的是将存在于系统中，亦即包含在空闲表 3 8 中、输出队列 4 4 中、多重指针表 5 6 中和封锁表 6 2 中的所有指针。在初始化过程中，每一指针值，每次一个，从运行中取出一方面检查队列系统 38、4 4 的指针处理是否正确，另一方面检查由所谈论的指针标识的、在缓冲器存储器 3 2 中的数据存储区 B (X)。

在初始化过程期间，队列系统 3 8、4 4 清空在测试下的指针和其所有拷贝，如果有的话，然后检查发现的拷贝数是否等于在多重指针表 5 6 中为在测试下的指针记录的数值。

维护功能 5 2 可以在其后按照一种或几种运行方式运行。不管选择何种方式，初始化过程是每一种运行方式的第一步。

参考图 3 和在有几种运行方式的场合的一个实施例中，维护功能可能有四级 7 0、7 2、7 4 和 7 6。最低级 7 0 称为“指针调整”和只导向指针的循环。其它三级扩展为对缓冲器存储器 3 2 的不同的读 / 写测试。

如同已经提到的，队列系统 3 8、4 4 在初始化过程清空指针 X 的所有的拷贝，如果有的话。在非实时系统中，这可以以几种不同的方式获得。在实时系统中，通常缩减为一种，仅处理正被返回到空闲表 3 8 的指针。

为清空上面叙述的类型的实时队列系统的指针 X，空闲表 3 8 必须首先清空其中所有的拷贝，如果有的话。这是通过从返回指针流 4 8 中滤除指针 X 而执行的，在图 4 - 7 中的步骤 7 8，以便阻止其被写入空闲表 3 8 中。更具体地，该滤除是这样启动的，维护功能 5 2

按照箭头 7 8' 给空闲表 3 8 发出一信号，为在当时空闲表中最后一个指针置一指示符。借助于信号 7 8"，维护功能 5 2 随后获得该指示符到达空闲表 3 8 中最前方位置的信息。当在滤除步骤的开始时在空闲表 3 8 中存在的所有指针被供给输出队列 4 4 时，空闲表 3 8 肯定清空了所有的指针 X。

之后，输出队列 4 4 必须清空指针 X 的拷贝，如果有的话。这种清空是，更具体地，通过维护功能 5 2 按照箭头 7 9' 发出一信号到输出队列 4 4 而启动的，为在当时在各自的输出队列中的最后一个指针置一指示符。维护功能 5 2 以按照箭头 7 9" 的信号随后从输出队列获得信息：如此标识的指针在各自的队列中首次终止。于是维护功能 5 2 一方面继续从返回指针流 4 8 滤除指针 X，同时等待在空闲表 3 8 的清空结束时位于输出队列 4 4 中最后的指针在队列中首次终止。当最慢的队列符合这一判据时，系统已清空了指针 X。

之后，在图 5 - 7 中的步骤 8 0，检查返回指针数是否与在多重指针表 5 6 中指针 X 的值相符。如果不是这样的话，产生一指针错误信号。这个错误信号以某种不同的措施终止，这取决于被执行的维护方式。在除“指针调整”方式之外的所有方式中，指针值被封锁而错误计数器（未示出）增量，图 5 - 7 中的步骤 8 2。如果多于 α 个错误被发现，在图 5 - 7 中的步骤 8 4，当前维护方式终止，错误计数器置为零并跳转到方式“指针调整”，在图 5 - 7 中的箭头 8 6。 α 是对在多重指针表 5 6 中的错误希望的容忍程度。在“指针调整”方式中指针错误信号不以任何措施终止，参见图 4。

在系统起动时，必须使一正确的指针组尽快地对队列系统可用。因此，“指针调整”过程被起动。这可以既由按照上面的指针错误出

现的结果而执行，也可由外部信号的激发执行。这里以及以后所说的指针错误意思是，如同上面叙述的，在初始化过程期间返回指针的拷贝数与在多重指针表中为所谈论的指针所陈述的数目不符。指针一经检查，其即被引入空闲表 3 8 和可以被用于包的排队。

当调整了所有的指针后对所有的指针执行维护级 7 2，下面也称为“快速测试”。其后维护功能 5 2 进入维护级 7 4，下面也称为“全测试”，然后开始执行维护级 7 6 “带重新测试的全测试”，其为当前可运行的状态，然而它可以在指针错误出现时被中断。

如果多于 α 的几个错误的指针（多重或丢失指针）被检测到，维护功能 5 2 总是以“指针调整”重新起动，正如上面类似地叙述的那样，而不依赖于正在执行的其它维护功能，这在图 3 中，以及图 5 - 7 中由箭头 8 6 指出。

参考图 4，方式 7 0 “指针调整”在步骤 8 8 把每一指针 X 返回到空闲表 3 8，把多重指针表 5 6 中的值复位为“无多重拷贝”，和在封锁表 6 2 中复位到“无封锁”。当最后一个指针被处理，参见步骤 9 0，指针错误计数器置为零，步骤 9 2。随后方式 7 0 结束在方式 7 2 中，箭头 9 4。

方式 7 2 “快测试”于是由第一方式 7 0 运行遍历所有的指针而被起动，参考图 5。然后对所有指针执行初始化过程和对由指针 X 标识的缓冲器存储器位置 B (X) 执行一简单的写和读联机测试，步骤 9 6。如果发现需要的话，这一测试可以以低的或最低的优先级在可以对缓冲器存储器执行的操作之中排队后执行。

写和读测试的结果用于返回当今指针到空闲表 3 8，如果该测试给出一正确结果的话：但是封锁其在系统中使用并将其引入封锁表 6

2, 如果测试指出由所谈论的指针标识的区域内故障的话, 在最后的指针后, 测试进入方式 7 4, 参见步骤 9 8 和箭头 1 0 0。

和在早先上面叙述的脱机测试相比, 其中图 1 中的测试逻辑 1 4 控制和检查存储器的所有数据和所有地址, 在图 2 中联机测试的测试逻辑 5 8 仅具有对那些地址位置的写和读的权利, 这些地址位置由初始化过程从系统使用释放, 其中系统清空一指针值。对于今后每一件叙述的, 仅当无更高优先级的对存储器的系统操作来到时, 该测试逻辑才被强迫操作。

在按照图 5 的方式 7 2 中的简单的写和读测试中一第一位模式被写入由被释放的指针标识的地址位置中, 一旦这点执行以后, 这些位置被读取以测试该存储单元能够存储这一第一位模式。于是该过程重复第二次, 这一次以第二位模式为第一模式的逆。如果读取的数据与写的数数据不相对应, 则以一错误信号的形式产生图 2 中的信号 6 0”。

参考图 6, 第三运行方式 7 4 “全测试”由第二方式 7 2 运行遍历所有指针运行以后被激发, 并以对未由方式 7 2 封锁的指针执行初始化过程重新开始。于是一缓冲器存储器测试以最低优先级排队对未被封锁的缓冲器存储器位置 B (X) 执行, 以检测未在第二运行方式中由简单写和读测试检测到的故障, 参见步骤 1 0 2 和 1 0 4。对这些存储器位置以测试数据写和读进行测试, 而对所有其它未被封锁的缓冲器存储器位置在为每一被封锁的指针测试期间以系统数据写和读进行测试。

该测试的结果用于返回该指针到空闲表 3 8, 如果该缓冲器存储器测试给出一正确结果的话; 但是如果该缓冲器存储器测试指出由该

指针标识的区域内故障的话，封锁该指针使其不在系统中使用并将其引入封锁表 6 2。在最后的指针之后，测试进入方式 7 6，参见步骤 1 0 6 和箭头 1 0 8。

在按照第三运行方式 7 4 的完全联机测试的场合下，寻址逻辑的错误也将被检测。一般这需要将不同于在存储器位置 B (X) 中的数据的数据写到所有其它的地址中。按照本发明，系统数据写入其它非封锁位置和从其中读出。这实现了在早先叙述的脱机测试中和受控数据同样的功能。由系统恰好把和测试逻辑使用来测试的数据同样的数据写入在测试下的具有地址解码错误重迭存储器位置的地址而造成的寻址错误被漏于发现的风险由连续重复所有的测试和由足以封锁该指针的一个错误检测而最小化。因此该测试逻辑等待系统数据在读检查执行之前被写入所有其它使用的位置。

第四运行方式 7 6 “带重新测试的全测试”于是由第三方式 7 4 运行遍历所有指针这一事实而激发，参考图 7。先前封锁的指针也因此被接受由所述对所有指针周期地，每次一个，执行的初始化过程测试。于是对非封锁指针施行和第三方式中同样类型的缓冲器存储器测试，步骤 1 1 0 和 1 1 2。早先封锁的指针经受同样类型的由在队列中保持的每一测试数据操作而规定的缓冲器存储器测试，只要有系统数据存储包含在包含存储区 B (X) 的缓冲器存储器段的某一处，其由早先封锁的指针 X 标识，步骤 1 1 4。该测试结果以和在第三方式中同样的方式用于非封锁指针和早先封锁的指针的不允许测试，而早先封锁的指针直到在由限制值 β 定义的一定数目的连续测试中被接受后才返回到空闲表 3 8。

第四运行方式 7.6 是基于对下述事实的理解，在连续的操作中，也必须考虑在测试下的存储器位置的寻址错误可能破坏系统数据的风险。因此，在寻址错误可能发生的整个段清空系统数据以前不允许从测试逻辑往早先封锁的位置写数据。其后等待在所有其它位置写入系统数据，就像在第三运行方式中一样，因为所有的读操作可能破坏出错的存储器中的数据，所以在读检验对由早先封锁的、而现在处于更新的测试下的指针标识的缓冲器位置执行之前再次等待从存储器段清空系统数据。

在单运行方式的场合的一个实施例中，基本上有合并按照图 6 和 7 的最后两种运行方式的问题。

更具体地，缓冲器存储器 3.2 的测试在初始化过程后以最低优先级为在由非封锁指针标识的缓冲器存储器位置上执行而排队，参见图 6 中步骤 1.0.2 和 1.0.4。这些存储器位置以测试数据进行，所有其它非封锁缓冲器存储器位置在测试过程中以系统数据进行。其后测试结果以下述方式使用：即该指针返回到空闲表 3.8，如果缓冲器存储器测试给出正确结果的话；但如果缓冲器存储器测试指出在由该指针定义的区域故障的话，封锁该指针不在系统中使用，

被封锁的指针供给封锁表 6.2 和经受由在队列中维护的每一测试数据操作规定的同一类型的缓冲器存储器测试，只要系统数据存储在包含由被封锁的指针标识的位置的缓冲器存储器段的某一处，参见图 7 中步骤 1.1.4。其后，使用测试结果以便在缓冲器存储器测试给出一正确结果时把该指针返回空闲表 3.8。被封锁的指针在大于限制值 β 数目的连续测试中被接受以后首次返回到空闲表 3.8 中。

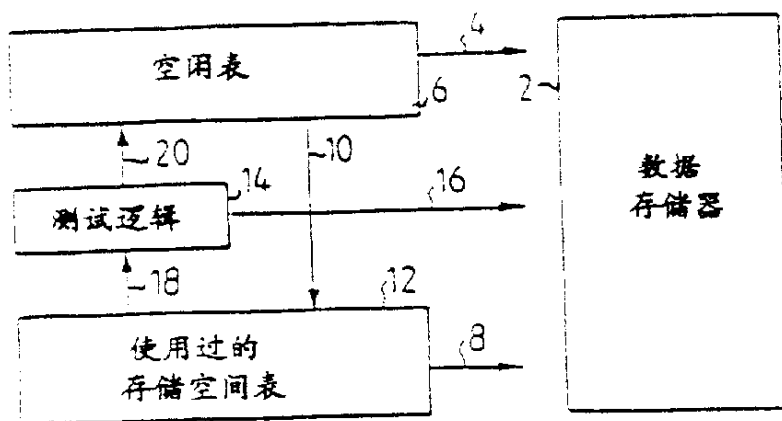


图1

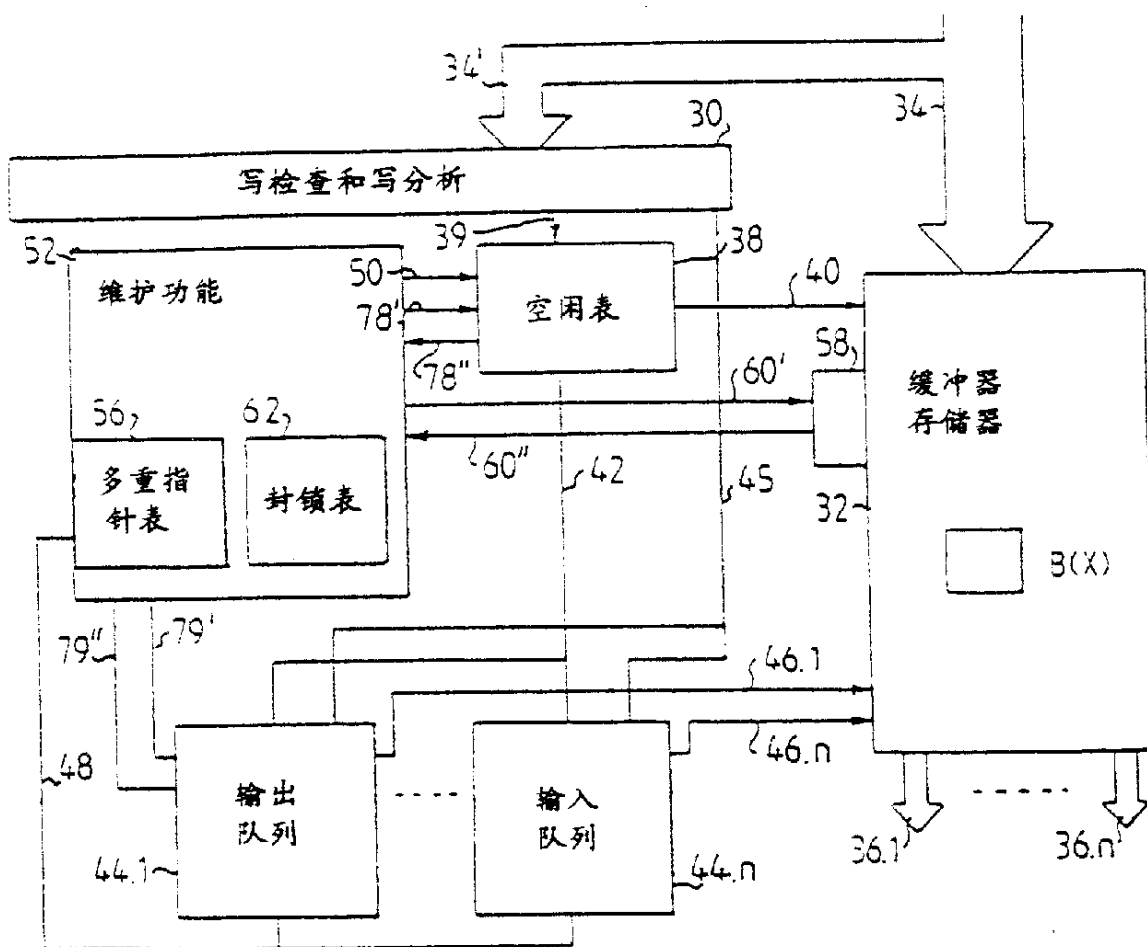


图2

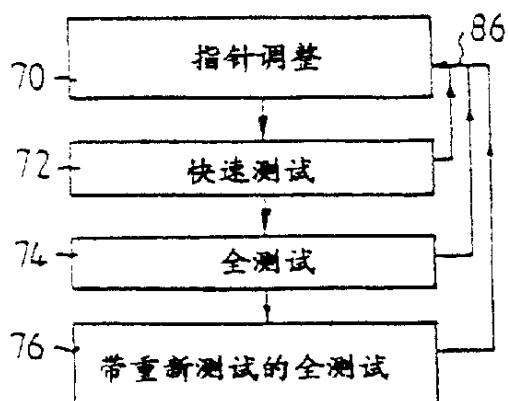


图3

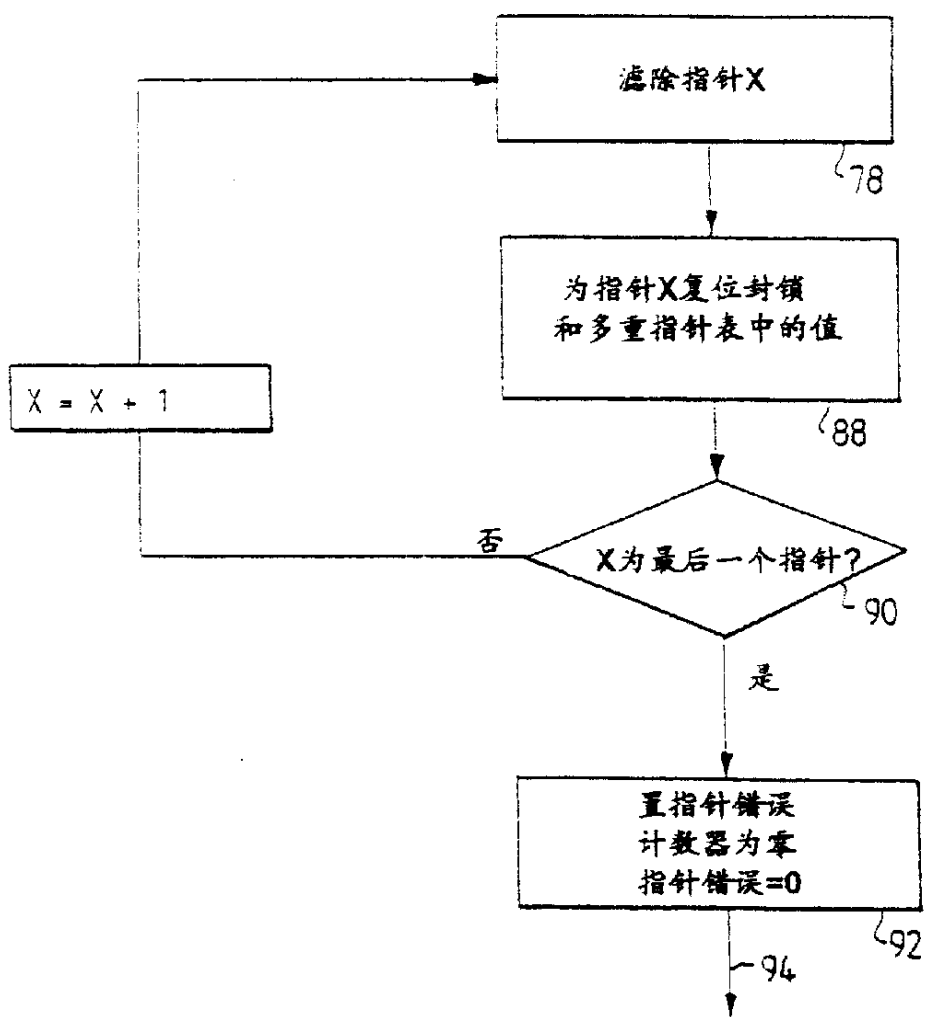


图4

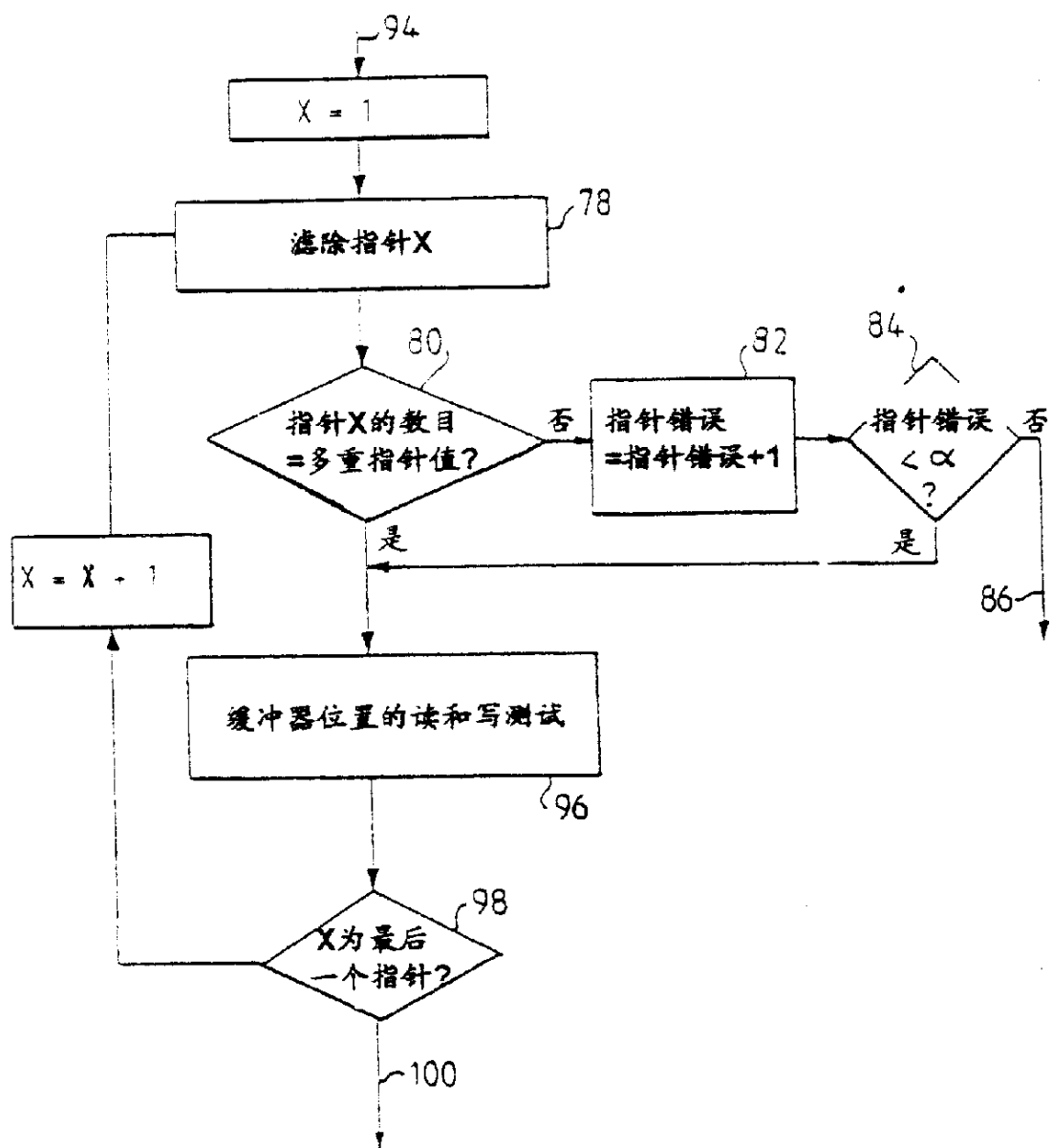


图5

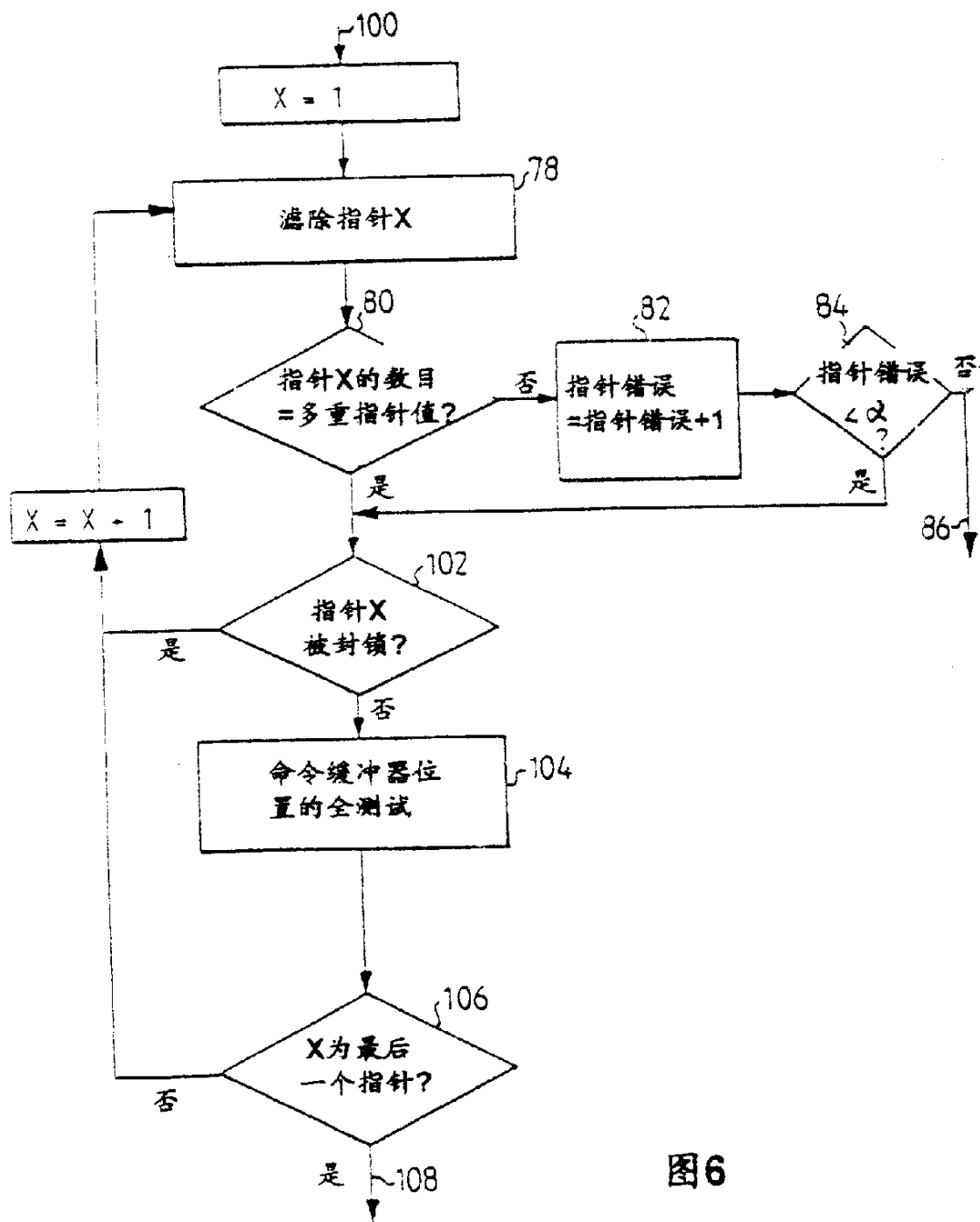


图6

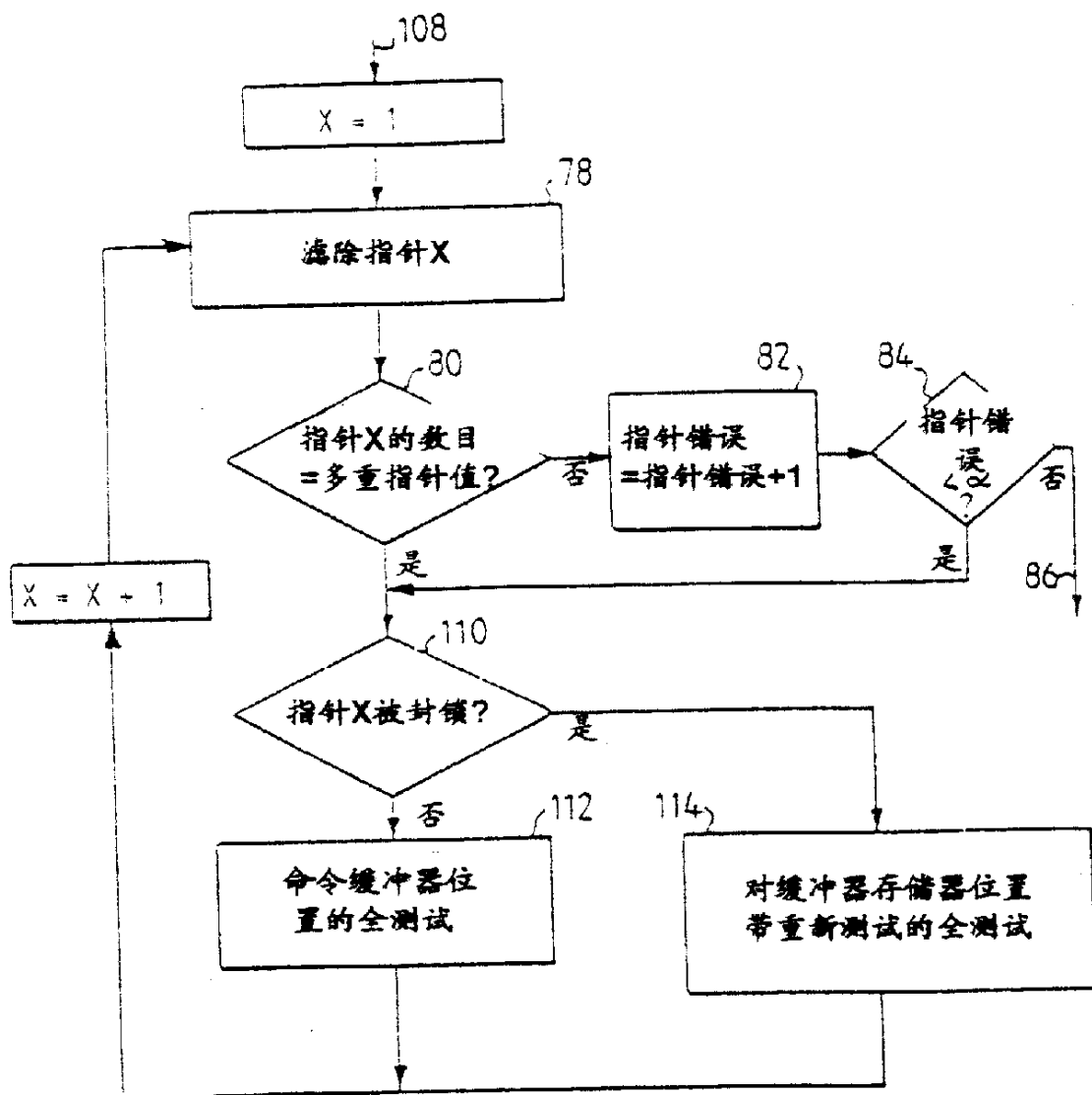


图7