



(12) 发明专利

(10) 授权公告号 CN 101714112 B

(45) 授权公告日 2013. 05. 01

(21) 申请号 200910170643. X

第 20 行至第 5 栏第 40 行、说明书附图 1-4.

(22) 申请日 2009. 09. 01

CN 1035018 A, 1989. 08. 23, 全文.

US 5539878 A, 1996. 07. 23, 全文.

(30) 优先权数据

2008-258974 2008. 10. 03 JP

审查员 胡雅娟

(73) 专利权人 富士通株式会社

地址 日本神奈川县川崎市

(72) 发明人 菱沼武夫 目崎义宪 石桥修

(74) 专利代理机构 隆天国际知识产权代理有限公司 72003

代理人 姜燕 陈晨

(51) Int. Cl.

G06F 11/22 (2006. 01)

G06F 9/445 (2006. 01)

(56) 对比文件

EP 0442651 B1, 1995. 08. 02, 说明书第 2 栏

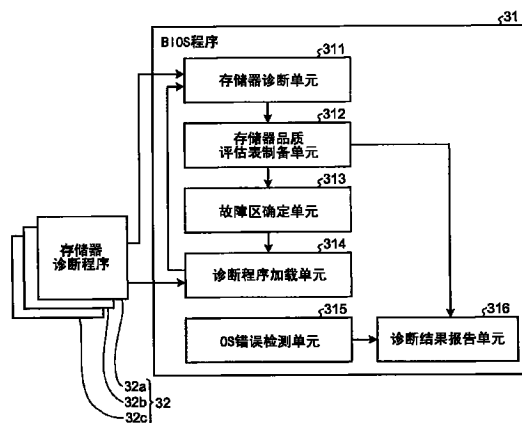
权利要求书2页 说明书10页 附图6页

(54) 发明名称

诊断计算机设备中的主存储器的系统及方法

(57) 摘要

本发明涉及一种包括主存储器的计算机设备,该计算机设备包括:第一存储器诊断单元,用于通过执行第一存储器诊断程序来确定主存储器中的故障区;以及存储单元,用于存储可重定位的第二存储器诊断程序。而且,该计算机设备包括第二存储器诊断单元,用于将第二存储器诊断程序加载到主存储器的除了由第一存储器诊断单元确定的故障区之外的其它区域中。本发明在未确定主存储器是否包含故障区的情况下,不必将存储器诊断程序加载到主存储器中。这可避免不能正常执行存储器诊断程序的情况。因此,简化了无故障区和故障区的确定,实现了高效的存储器诊断处理。



1. 一种诊断计算机设备中的主存储器的系统,该系统包括:

第一存储器诊断单元,用于通过执行第一存储器诊断程序来确定所述主存储器中的故障区;以及

第二存储器诊断单元,用于将多个存储器诊断程序从较小程序规模的那一个存储器诊断程序依序加载到所述主存储器中除了由所述第一存储器诊断单元确定的所述故障区之外的区域中,而且是被加载到所述主存储器中除了由所述第二存储器诊断单元利用所述多个存储器诊断程序中已经加载的存储器诊断程序重新计算和确定的故障区之外的区域,其中所述多个存储器诊断程序的每一个的程序规模大于该第一存储器诊断程序。

2. 根据权利要求1所述的系统,其中,所述第一存储器诊断单元执行所述第一存储器诊断程序,所述第一存储器诊断程序不被加载到所述主存储器中。

3. 根据权利要求1所述的系统,还包括存储器品质评估表制备单元,用于制备存储器品质评估表,其中由所述第一存储器诊断单元进行的所述主存储器的诊断的结果与所述主存储器的地址信息相关联。

4. 根据权利要求3所述的系统,其中,所述第二存储器诊断单元参照所述存储器品质评估表提取所述主存储器上被所述第一存储器诊断单元诊断为错误的地址信息,并且将与所提取的地址信息对应的区域作为故障区从用于加载所述多个存储器诊断程序的区域中排除。

5. 根据权利要求3所述的系统,还包括:

操作系统错误检测单元,用于检测操作系统启动时的错误;以及

诊断结果报告单元,用于在所述操作系统错误检测单元已经检测到启动错误时,报告由所述第二存储器诊断单元进行的所述主存储器的诊断的结果。

6. 根据权利要求5所述的系统,其中,被并入到基本输入输出程序中的所述第一存储器诊断单元、第二存储器诊断单元、存储器品质评估表制备单元、操作系统错误检测单元和诊断结果报告单元的每一个均在通电时启动,并且在所述基本输入输出程序已经启动时被执行,其中所述基本输入输出程序是用来控制将要输入到所述主存储器和从所述主存储器输出的数据的控制程序。

7. 一种诊断计算机设备中的主存储器的方法,包括以下步骤:

通过执行第一存储器诊断程序来确定所述主存储器中的故障区;以及

将多个存储器诊断程序从较小程序规模的那一个存储器诊断程序依序加载到所述主存储器中除了通过确定故障区的步骤而确定的故障区之外的其它区域中,而且是被加载到所述主存储器中除了利用所述多个存储器诊断程序中已经在加载时加载的存储器诊断程序重新计算和确定的故障区之外的区域,其中所述多个存储器诊断程序的每一个的程序规模大于该第一存储器诊断程序。

8. 根据权利要求7所述的方法,其中,所述确定故障区的步骤执行所述第一存储器诊断程序,而所述第一存储器诊断程序不被加载到所述主存储器中。

9. 根据权利要求7所述的方法,还包括以下步骤:制备存储器品质评估表,其中通过所述确定故障区的步骤进行的所述主存储器的诊断的结果与所述主存储器的地址信息相关联。

10. 根据权利要求9所述的方法,其中,参照所述存储器品质评估表,加载所述多个存

存储器诊断程序的步骤提取所述主存储器上通过所述确定故障区的步骤而被诊断为错误的地址信息,并且将与所提取的地址信息对应的区域作为故障区从用于加载所述多个存储器诊断程序的区域中排除。

11. 根据权利要求 9 所述的方法,还包括以下步骤:

检测操作系统启动时的错误;以及

当检测错误的步骤已经检测到启动错误时,报告通过加载所述多个存储器诊断程序的步骤进行的所述主存储器的诊断的结果。

12. 根据权利要求 11 所述的方法,其中,被并入到基本输入输出程序中的所述确定故障区的步骤、加载所述多个存储器诊断程序的步骤、制备存储器品质评估表的步骤、检测启动错误的步骤以及报告诊断结果的步骤的每一个均在通电时启动,并且在所述基本输入输出程序已经启动时执行,其中所述基本输入输出程序是用来控制将要输入到所述主存储器和从所述主存储器输出的数据的控制程序。

## 诊断计算机设备中的主存储器的系统及方法

[0001] 相关申请交叉引用

[0002] 本申请基于 2008 年 10 月 3 日提交的在先日本专利申请号 2008-258974, 并要求享有其优先权, 该专利申请的全部内容通过参考被并入本文。

### 技术领域

[0003] 本申请讨论的实施例涉及一种计算机设备以及一种诊断存储器的方法。

### 背景技术

[0004] 计算机设备使用许多种类的诊断测试来诊断每个内置装置的缺陷部分。作为计算机设备的一个例子, 公开了一种诊断工具, 其运行存储器诊断测试程序来诊断内置存储器中的存储功能是否正常工作, 例如是否可以正常地从存储器读出预定数据、向存储器正常写入预定数据。例如参见日本特开专利公开号 H6-4413。

[0005] 在最简单的诊断测试中, 存储器诊断程序被预先存储在计算机设备的基本输入/输出系统(BIOS)的一部分中。当计算机设备启动时, 存储器诊断程序与 BIOS 一起运行。

[0006] 然而, BIOS 通常具有非常有限的存储能力, 其中存储了如初始程序装入程序(initial program loader IPL)之类的另一程序。因此, BIOS 仅能包含简单的存储器诊断程序。因此, 在传统的计算机设备中, 在运行 BIOS 且启动操作系统之后, 用于执行复杂存储器检查的存储器诊断程序通常被操作系统(OS)加载到如随机存取存储器(RAM)类的主存储器中并被执行。

[0007] 顺便提一下, 只有在主存储器的所有地址中没有发现故障时, 传统的计算机才能工作正常。但是即使通过预先检查主存储器发现了主存储器中有一些故障区, 目前还是期望有一种不使用故障区而能正常运行的新型计算机设备。

[0008] 为了当在主存储器中存在故障区的情况下仍正常运行计算机设备, 就需要在操作系统启动之前对故障区进行检测。

[0009] 但是, BIOS 中存储的存储器诊断程序不能执行使 OS 能够正常工作的存储器诊断。

[0010] 在这种情况下可以考虑如下的技术。传统上由 OS 处理的复杂存储器诊断程序被存储在除 BIOS 之外的存储器中, 如存储在只读存储器(ROM)中。同时, 在主存储器的预定地址加载复杂存储器诊断程序的加载程序被存储在 BIOS 中。这使计算机设备在引导(boot up)时能够启动复杂的存储器诊断程序。

[0011] 然而, 该结构会导致出现这种情形, 即在存储器诊断之前, 存储器诊断程序被加载到主存储器中。也就是说, 存储器诊断程序被加载到主存储器中的时候, 并不知道加载了存储器诊断程序的主存储器的区域是否是有故障的。因此, 存在存储器诊断程序本身无法被执行的可能性。

### 发明内容

[0012] 根据本发明的一个方面, 一种具有主存储器的计算机设备包括: 第一存储器诊断

单元,用于通过执行第一存储器诊断程序来确定所述主存储器中的故障区;存储单元,用于存储可重定位的第二存储器诊断程序;以及第二存储器诊断单元,用于将所述第二存储器诊断程序加载到所述主存储器中除了由所述第一存储器诊断单元确定的所述故障区之外的区域中。

[0013] 根据本发明的另一个方面,一种诊断计算机设备中的主存储器的方法,包括以下步骤:通过执行第一存储器诊断程序来确定所述主存储器中的故障区;以及将第二存储器诊断程序加载到所述主存储器中除了通过确定故障区的步骤而确定的故障区之外的其它区域中。

[0014] 根据本发明的再一个方面,一种计算机可读存储介质,其中存储有用来诊断计算机设备中的主存储器的计算机程序,所述计算机程序使得所述计算机设备执行以下步骤:通过执行第一存储器诊断程序来确定所述主存储器中的故障区;以及将第二存储器诊断程序加载到所述主存储器中除了通过确定故障区的步骤确定的故障区之外的区域中。

[0015] 本发明在未确定主存储器是否包含故障区的情况下,不必将存储器诊断程序加载到主存储器中。这样可避免不能正常执行存储器诊断程序的情况。因此,简化了无故障区和故障区的确定,实现了高效的存储器诊断处理。

[0016] 借助权利要求中具体指出的元件及组合可实现和达到本发明的目的和优点。

[0017] 应该理解,前述概括性描述和下文的详细描述都是示例性及解释性的,不是对本发明保护范围加以限制。

## 附图说明

[0018] 图 1 是示出根据本发明实施例的计算机设备的结构的方框图;

[0019] 图 2 是示出图 1 的动态随机存取存储器(DRAM)的细节;

[0020] 图 3 是示出图 1 的 BIOS 程序的细节的功能框图;

[0021] 图 4A 是在已经执行存储器诊断程序 32a 之后的存储器品质评估表(memory-quality evaluation table);

[0022] 图 4B 是示出当将存储器诊断程序 32b 加载 DRAM 中时 DRAM 上的分配区;

[0023] 图 4C 是在已经执行存储器诊断程序 32b 之后的存储器品质评估表;

[0024] 图 4D 是示出当将存储器诊断程序 32c 加载 DRAM 中时 DRAM 上的分配区;

[0025] 图 4E 是在已经执行存储器诊断程序 32c 之后的存储器品质评估表;

[0026] 图 4F 示出通过对 DRAM 进行诊断得出的结果;以及

[0027] 图 5 是示出根据本发明实施例的计算机设备执行的存储器诊断处理的步骤流程图。

## 具体实施方式

[0028] 将参考附图对本发明的优选实施例进行解释。

[0029] 下面描述根据一实施例的计算机设备的特征。计算机设备包括主存储器;第一存储器诊断单元,其通过执行第一存储器诊断程序来识别出主存储器的故障区或缺陷区;存储单元,其存储可重定位的第二存储器诊断程序;以及第二存储器诊断单元,其将存储在存储单元中的第二存储器诊断程序加载到除由第一主存储器诊断单元识别出的主存储器缺

陷区之外的其它区域(称为“非故障”或“非缺陷”区)中,并执行该第二存储器诊断程序。

[0030] 也就是说,根据该实施例的计算机设备通过执行多个存储器诊断程序来对计算机设备中诸如动态随机存取存储器(DRAM)等主存储器进行诊断。

[0031] 具体而言,计算机设备中诸如只读存储器(ROM)等存储单元预先存储有多个不同的存储器诊断程序。第一存储器诊断单元通过运行多个存储器诊断程序中的第一存储器诊断程序来对主存储器进行诊断。

[0032] 多个存储器诊断程序各自诊断不同的存储功能,并根据存储功能的诊断级别具有不同的程序规模。也就是说,用于诊断较高级存储功能的存储器诊断程序具有较大的程序规模。第一存储器诊断程序属于存储功能中的最低诊断级别,具有最小的程序规模。

[0033] 根据该实施例的第一存储器诊断单元执行第一存储器诊断程序,而不将第一存储器诊断程序加载到主存储器中。因此,首先由第一存储器诊断单元不使用主存储器而执行对存储功能的简单诊断。然后,由第二存储器诊断单元执行对存储功能的更高级别的诊断。因此,在不清楚主存储器中是否有缺陷区的阶段,存储器诊断程序不必被加载到主存储器中。这避免了存储器诊断程序变成不可执行的情况。

[0034] 如果第一存储器诊断程序被加载到主存储器中,以占据其上相对小的区域,则可以由第一存储器诊断单元将第一存储器诊断程序加载到主存储器中。

[0035] 第二存储器诊断单元将存储在诸如 ROM 等存储单元中的第二存储器诊断程序加载到主存储器中除由第一存储器诊断单元识别出的缺陷区之外的其它区域中,并执行该第二存储器诊断程序。也就是说,第二存储器诊断单元将由第一存储器诊断单元识别出的缺陷区从主存储器的用来加载第二存储器诊断程序的区域中排除。在主存储器的所有区域中,缺陷区被定义为诸如读出和写入功能等存储功能不能正常工作的区域。第二存储器诊断程序执行较高级别的存储功能诊断,并且规模比第一存储器诊断程序规模更大。

[0036] 这样,根据该实施例的计算机设备可以避免这样的情况:第二诊断程序被错误地加载缺陷区,从而不能正常执行第二诊断程序。

[0037] 然后,第二存储器诊断单元通过执行已经被加载到非缺陷区的部分区域中的第二存储器诊断程序对主存储器再次进行诊断。接着,第二存储器诊断单元基于由第二存储器诊断程序诊断的主存储器的结果在除之前识别为缺陷区之外的区域中识别新的缺陷区。而且,第二存储器诊断单元将最新和之前发现的缺陷区作为缺陷区而从待加载下一存储器诊断程序(即,第三存储器诊断程序和之后的程序)的区域中排除。重复这些处理,直到存储在诸如 ROM 等存储单元中的所有存储器诊断程序已经结束运行。因此,第二存储器诊断程序之后的存储器诊断程序总是被加载到主存储器的非故障区中。因此,这些存储器诊断程序可以被正常执行。

[0038] 以这种方式,根据该实施例,首先执行不使用主存储器或者仅占据主存储器的小部分的第一存储器诊断程序,以便对主存储器进行简单诊断。然后,利用诊断结果,对需要被加载到主存储器中且可以提供对存储器的详细诊断的第二存储器诊断程序的设置位置进行识别。因此,这就降低了第二存储器诊断程序之后的存储器诊断程序因被加载到主存储器中存储功能不能正常工作的缺陷区的可能性,并且避免了第二存储器诊断程序之后的存储器诊断程序变成不可执行的情况。

[0039] 参照图 1,对根据本发明的实施例的计算机设备的结构进行解释。图 1 是示出根据

一实施例的计算机设备的结构的方框图。图 2 是示出图 1 中示出的 DRAM 的细节。

[0040] 如图 1 中所示, 计算机设备 1 包括中央处理单元(CPU) 10、DRAM 20、ROM 30、显示器控制器 40 和连接上述元件的总线 60。

[0041] CPU 10 控制计算机设备 1 的整体运行。具体地, 当计算机设备 1 启动时, CPU 10 从 ROM 30 中读取基本输入输出系统(BIOS) 程序 31 (将在后文描述), 并执行 BIOS 程序 31 以检查诸如 DRAM 20 等装置中是否有错误。CPU 10 还通过执行 BIOS 程序 31 来执行操作系统(OS) 的启动以及对各装置的控制。

[0042] DRAM 20 是计算机设备 1 的主存储器(主存储单元)。具体而言, DRAM20 是其中加载了待由 CPU 10 执行的程序(包括 BIOS 程序 31)的装置, 并且被用作工作存储器以暂时存储与程序相关的各种数据。而且, 如图 2 中所示, DRAM 20 被划分成多个区(1)至(20), 以便加载待由 CPU 10 执行的程序。

[0043] 用作存储单元的 ROM 30 存储 CPU 10 执行的数据和各种程序。ROM 30 存储的程序包括 BIOS 程序 31 和多个存储器诊断程序 32。

[0044] BIOS 程序 31 是基本输入输出程序。具体而言, 在计算机设备 1 被打开时, BIOS 程序 31 开始运行, 所述 BIOS 程序 31 是控制 OS、应用程序、DRAM 20 和外围装置(显示器、键盘等)之间的输入和输出数据的控制程序。而且, BIOS 程序 31 执行存储器诊断处理, 以确认可以从 DRAM 20 中正常读取数据和向 DRAM 20 正常写入数据, 并确认与存储器保护功能和错误校正功能相关的功能能正常工作。下文详细描述 BIOS 程序 31 执行存储器诊断处理的具体结构。

[0045] 当 BIOS 程序 31 对 DRAM 20 执行存储器诊断处理时执行多个存储器诊断程序 32, 多个存储器诊断程序 32 是检查 DRAM 20 的各个存储功能是否正常工作(例如, 预定数据是否能正常写入存储器中及从存储器中正常读出)的程序集合。在此实施例中, 存储器诊断程序 32 包括三个存储器诊断程序 32a, 32b 和 32c。存储器诊断程序 32a, 32b 和 32c 分别是为诊断截然不同的存储功能而设计的, 根据存储功能的诊断级别具有不同的程序规模。例如, 在此实施例中, 32a 具有最低的存储功能诊断级别, 然后依次是 32b 及 32c, 它们的程序规模也以相同次序增加。在三个存储器诊断程序中, 存储器诊断程序 32a 具有最低的存储功能诊断级别以及最小的程序规模。

[0046] 显示器控制器 40 连接到显示器 50, 向其提供诸如 DRAM 20 的存储器诊断结果等输出。显示器 50 是图像显示装置, 能可视地显示来自显示器控制器 40 的输出。

[0047] 参照图 3, 详细描述使 BIOS 程序 31 能够在 DRAM 20 上执行存储器诊断处理的具体结构。图 3 的方框图示出了图 1 的 BIOS 程序 31 的细节, 并示出了由 CPU 10 执行存储在 ROM 30 中的 BIOS 程序 31 时可以实现的功能。

[0048] BIOS 程序 31 包括存储器诊断单元 311、存储器品质评估表制备单元 312、故障区确定单元 313、诊断程序加载单元 314、OS 错误检测单元 315 和诊断结果报告单元 316。在 BIOS 程序 31 启动(被执行)时, 执行这些功能单元 311-316。

[0049] 如上所述, 当计算机设备 1 启动时, 首先执行 BIOS 程序 31。因此, 在 OS 引导之前执行 DRAM 20 的存储器诊断处理, 使得即使在 OS 没有启动的情况下存储器诊断程序也能正常运行。

[0050] 存储器诊断单元 311 通过运行存储器诊断程序 32a 而开始对 DRAM 20 进行诊断,

存储器诊断程序 32a 是存储器诊断程序 32 的第一存储器诊断程序。而且,存储器诊断单元 311 不用将存储器诊断程序 32a 加载到 DRAM 20 中,就能运行存储器诊断程序 32a。如上所述,在存储器诊断程序 32 中,存储器诊断程序 32a 包含最低的存储功能诊断级别,具有最小的程序规模。存储器诊断程序 32a 的程序规模被设置的小,就消除了将存储器诊断程序 32a 加载到 DRAM 20 中的必要。这可避免在不清楚 DRAM 20 是否有缺陷区时将存储器诊断程序加载到 DRAM 20 中。

[0051] 存储器品质评估表制备单元 312 生成存储器品质评估表,该表将存储器诊断单元 311 的诊断结果和 DRAM 20 上的地址信息相关联。在下文解释存储器诊断处理的步骤时,再详细解释存储器品质评估表。

[0052] 故障区确定单元 313 基于存储器诊断单元 311 的诊断结果确定 DRAM 20 中的故障区或缺陷区。具体而言,故障区确定单元 313 参照由存储器品质评估表制备单元 312 生成的存储器品质评估表,提取 DRAM 20 上被存储器诊断单元 311 诊断为错误的地址信息,从而基于所提取的地址信息确定 DRAM 20 中的故障区。在这种情况下,存储器诊断单元 311 和故障区确定单元 313 作为第一存储器诊断单元,通过执行第一存储器诊断程序 32a 来确定 DRAM 20 (主存储器)中的故障区。

[0053] 诊断程序加载单元 314 将存储器诊断程序 32b 加载到 DRAM 20 的所有区域(1)至(20)中除由故障区确定单元 313 确定的故障区之外的区域中,其中该存储器诊断程序 32b 是存储在 ROM 30 中的第二存储器诊断程序。更具体讲,诊断程序加载单元 314 参照由存储器品质评估表制备单元 312 制备的存储器品质评估表,提取被存储器诊断单元 311 诊断为错误的 DRAM 20 的地址信息。然后,诊断程序加载单元 314 将与所提取的作为故障区的地址信息对应的区域从用于加载存储器诊断程序 32b 的区域中排除。换言之,诊断程序加载单元 314 将除故障区之外的一部分区域设置为用于加载第二存储器诊断程序 32b 的区域。

[0054] 然后,存储器诊断单元 311 通过执行由诊断程序加载单元 314 加载的存储器诊断程序 32b,重新对 DRAM 20 的所有区域(1)至(20)中除由故障区确定单元 313 确定的故障区之外的区域进行诊断。在这种情况下,诊断程序加载单元 314 和存储器诊断单元 311 用作第二存储器诊断单元,它们将第二存储器诊断程序 32b 加载到除由故障区确定单元 313 确定的故障区之外的其它区域中,并执行第二存储器诊断程序 32b。

[0055] 接着,基于存储器诊断单元 311 的新诊断结果,故障区确定单元 313 确定 DRAM 20 的所有区域(1)至(20)中除之前被故障区确定单元 313 确定为故障区之外的新的故障区。诊断程序加载单元 314 从用于加载由存储器诊断单元 311 接着执行的存储器诊断程序 32c 的区域中将由故障区确定单元 313 先前确定和新确定的故障区作为故障区而排除。

[0056] 重复由功能单元 311-314 执行的处理,直到存储器诊断单元 311 执行完所有的存储器诊断程序 32(32a-32c)。由于此原因,存储器诊断程序 32 总是被加载到 DRAM 20 中的故障区之外的非故障区。因此,就可以正常执行存储器诊断程序。

[0057] OS 错误检测单元 315 检测在 OS 启动时的错误。这类 OS 启动错误包括计算机设备 1 停止工作,和不接收键盘或鼠标输入的状态(称作挂起, hung-up),以及 OS 本身不能连续处理的状态(称作蓝屏和黑屏)。当事先测量计算机设备 1 在正常操作中的启动时间时,确定超时持续时间(timeout duration)并将其存储在 ROM 30 的预定区域中,即使在过了超时持续时间后,也可发现挂起、蓝屏或黑屏,OS 错误检测单元 315 被设计为对 OS 启动错

误进行检测。

[0058] 当 OS 错误检测单元 315 检测到 OS 启动错误时,诊断结果报告单元 316 以可视方式向显示器控制器 40 输出由存储器诊断单元 311 诊断的 DRAM 20 的存储器诊断结果。诊断结果报告单元 316 作为报告由存储器诊断单元 311 诊断的 DRAM 20 的存储器诊断结果的装置。该诊断结果报告单元 316 还被配置为输出由存储器品质评估表制备单元 312 制备的存储器品质评估表,作为由存储器诊断单元 311 对 DRAM 20 执行存储器诊断的结果。

[0059] 参照图 4A 到图 4F,具体描述了 BIOS 程序 31 如何对 DRAM 20 执行存储器诊断处理的步骤。图 4A 到图 4F 示出了 BIOS 程序 31 在 DRAM 20 上执行存储器诊断处理的步骤。图 4A 和图 4C 分别示出在执行存储器诊断程序 32a 和 32b 之后由存储器品质评估表制备单元 312 制备的存储器品质评估表。图 4B 和图 4D 分别示出 DRAM 20 中将要加载由诊断程序加载单元 314 设置的后继的存储器诊断程序 32b 和 32c 的区域。而且,图 4E 示出在执行存储器诊断程序 32c 之后由存储器品质评估表制备单元 312 制备的存储器品质评估表,而图 4F 示出 DRAM 20 的存储器诊断的最终结果。

[0060] 如图 4A 所示,当存储器诊断单元 311 执行存储器诊断程序 32a 时,存储器品质评估表制备单元 312 产生存储器品质评估表,该表将存储器诊断单元 311 的诊断结果和 DRAM 20 的地址信息相关联。在此情况下,存储器诊断程序 32a 不被加载到 DRAM 20 中就被存储器诊断单元 311 执行。在由存储器品质评估表制备单元 312 制备的存储器品质评估表中,数值“1”作为 DRAM 20 的存储器诊断结果被分配给存储器诊断单元 311 诊断为非故障的 DRAM 20 的地址,而数值“0”作为 DRAM 20 的存储器诊断结果被分配给由存储器诊断单元 311 诊断为错误的 DRAM 20 的地址。在图 4B 中,区域(3)和(19)被分配了“0”,它们被诊断为错误。

[0061] 接着,如图 4B 所示,故障区确定单元 313 参照图 4A 中示出的存储器品质评估表确定 DRAM 20 中的故障区。在图 4A 中,在 DRAM 20 的所有区域(1)至(20)中,被诊断为错误的区域(3)和(19)被确定为有故障。

[0062] 如图 4B 中所示,诊断程序加载单元 314 将 DRAM 20 的所有区域中除去由故障区确定单元 313 确定的故障区之外的一部分区域选择为用于加载存储器诊断单元 311 接下来要执行的存储器诊断程序 32b 的区域。在图 4A 和图 4B 中,在 DRAM 20 的所有区域(1)至(20)中,除故障区(3)和(19)之外的一部分区域被选择为用于加载接下来待执行的存储器诊断程序 32b 的区域。

[0063] 接着,当存储器诊断单元 311 执行存储器诊断程序 32b 时,在 DRAM 20 的所有区域(1)至(20)中除故障区(3)和(19)之外的区域被重新诊断。

[0064] 如图 4C 中所示,存储器品质评估表制备单元 312 使用由存储器诊断单元 311 得到的新的诊断结果来更新存储器品质评估表。即,在存储器品质评估表中,在 DRAM 20 的所有区域(1)至(20)中,数值“2”作为存储器诊断结果被重新分配给存储器诊断单元 311 新诊断为正常的区域的地址。相反,对于存储器诊断单元 311 已经诊断为错误的区域地址,保持数值“1”作为存储器诊断结果。在图 4C 和图 4D 中,分别被分配了作为存储器诊断结果的数值“1”的区域(5)和(20)被重新诊断为错误。

[0065] 如图 4D 中所示,参照图 4C 中的存储器品质评估表,故障区确定单元 313 确定在 DRAM 20 的所有区域(1)至(20)中除旧的故障区(3)和(19)之外的新故障区。在图 4C 和

图 4D 中,在 DRAM 20 的所有区域(1)至(20)中,除旧的故障区(3)和(19)之外,被重新诊断为错误的区域(5)和(20)被重新确定为故障区。

[0066] 如图 4D 中所示,在 DRAM 20 的所有区域中除被故障区确定单元 313 确定为故障区之外的一部分区域被诊断程序加载单元 314 重新选择为用于加载存储器诊断单元 311 接下来要执行的存储器诊断程序 32c 的区域。在图 4C 和图 4D 中,在 DRAM 20 的所有区域(1)至(20)中除故障区(3)、(5)、(19)和(20)之外的一部分区域被重新选择为用于加载存储器诊断程序 32c 的区域。

[0067] 当存储器诊断单元 311 执行存储器诊断程序 32c 时,在 DRAM 20 的所有区域(1)至(20)中除故障区(3)、(5)、(19)和(20)之外的其它区域被重新诊断。

[0068] 如图 4E 中所示,存储器品质评估表制备单元 312 使用存储器诊断单元 311 得到的新的诊断结果更新存储器品质评估表。即,在存储器品质评估表中,在 DRAM 20 的所有区域中,数值“3”作为存储器诊断结果被重新分配给存储器诊断单元 311 重新诊断为正常的区域地址。相反,对于存储器诊断单元 311 已经诊断为错误的区域的地址,保持数值“2”作为存储器诊断结果。在图 4E 和图 4F 中,分配了作为存储器诊断结果的数值“2”的区域(4)被重新诊断为错误。

[0069] 通过执行上述处理,如图 4F 中所示,所获得的存储器诊断的最终结果与由 BIOS 程序 31 执行的存储器诊断处理一致。

[0070] 参照图 5 描述图 1 中计算机设备 1 的存储器诊断处理。图 5 是示出图 1 中计算机设备 1 的存储器诊断处理的流程图。图 5 的存储器诊断处理是在计算机设备 1 被打开后,通过运行 CPU 10 从 ROM 30 中读出的 BIOS 程序 31 而执行的处理。

[0071] 如图 5 中所示, BIOS 程序 31 的存储器诊断单元 311 执行多个存储器诊断程序 32 中的存储器诊断程序 32a,如步骤 S10,并逐个诊断 DRAM 20 的所有区域,如步骤 S11。存储器诊断单元 311 在没有将存储器诊断程序 32a 加载 DRAM 20 的情况下运行存储器诊断程序 32a。然后,存储器诊断单元 311 将存储器诊断结果和 DRAM 20 的地址信息存储到 DRAM 20 的每一区域中,且存储器诊断结果和 DRAM 20 的地址信息彼此关联,如步骤 S12。

[0072] 接着,存储器诊断单元 311 确定在 DRAM 20 的所有区域中是否存在未诊断的区域,如步骤 S13。所述确定是基于存储器诊断结果是否已经与 DRAM 20 的每个区域的地址信息相关联而做出的。如果确定在 DRAM 20 的所有区域中在某处存在有未诊断的区域,即步骤 S13 中为是,则存储器诊断单元 311 重复步骤 S11 到步骤 S13 的处理,直到在 DRAM 20 的所有区域中没有未诊断的区域。

[0073] 如果确定在 DRAM 20 的所有区域中没有未诊断的区域,即步骤 S13 中为否,则存储器诊断单元 311 请求存储器品质评估表制备单元 312 来制作存储器品质评估表。一旦接收到该请求,存储器品质评估表制备单元 312 制备存储器品质评估表,其中存储器诊断单元 311 的存储器诊断结果与 DRAM 20 的地址信息相关联,如步骤 S14,并且存储器品质评估表制备单元 312 将制备好的存储器品质评估表发送给故障区确定单元 313。

[0074] 然后,参照所接收的存储器品质评估表,故障区确定单元 313 提取 DRAM 20 上被存储器诊断单元 311 确定为错误的地址信息,以根据所提取的地址信息确定 DRAM 20 中的故障区,如步骤 S15。

[0075] 诊断程序加载单元 314 将 DRAM 20 的所有区域中除在步骤 S15 由故障区确定单元

313 确定为故障区之外的区域的一部分或多个部分确定为一个或多个加载区,如步骤 S16,并且将下一存储器诊断程序 32b 加载到所述加载区中,如步骤 S17。

[0076] 接着,存储器诊断单元 311 通过执行所加载的存储器诊断程序 32b,逐个诊断 DRAM 20 中除故障区之外的所有区域,如步骤 S18。然后,存储器诊断单元 311 将存储器诊断结果和 DRAM 20 上的地址信息存储在 DRAM 20 的每个区域中,所述存储器诊断结果和 DRAM 20 上的地址信息彼此相关联,如步骤 S19。

[0077] 然后,存储器诊断单元 311 确定在 DRAM 20 的所有区域中除故障区之外的区域中是否存在未诊断的区域,如步骤 S20。如果存储器诊断单元 311 确定存在未诊断区域,即步骤 S20 中为是,则存储器诊断单元 311 重复步骤 S18 到 S20 的处理,直到在 DRAM 20 的所有区域中除故障区之外的区域中不再发现未诊断区域。

[0078] 相反,如果存储器诊断单元 311 确定在 DRAM 20 的所有区域中除故障区之外的区域中不存在未诊断区域,即步骤 S20 中为否,则存储器诊断单元 311 请求存储器品质评估表制备单元 312 更新存储器品质评估表。一旦接收到该请求,存储器品质评估表制备单元 312 就更新存储器品质评估表,如步骤 S21。

[0079] 之后,存储器诊断单元 311 确定是否已经执行了所有存储器诊断程序 32a、32b 和 32c,如步骤 S22。如果存储器诊断单元 311 确定还未执行完所有存储器诊断程序,即步骤 S22 中为否,则处理返回步骤 S15。然后,重复步骤 S15 到 S22 的过程,直到执行完所有的存储器诊断程序 32a、32b 和 32c。

[0080] 如果存储器诊断单元 311 确定已经执行完所有存储器诊断程序,即步骤 S22 中为是,则存储器诊断单元 311 进到步骤 S23。

[0081] 在步骤 S23 中,OS 错误检测单元 315 检测启动时是否有 OS 错误,或者是否有 OS 启动错误。如果其结果是检测到 OS 启动错误,即步骤 S23 中为是,则 OS 错误检测单元 315 向诊断结果报告单元 316 发出检测信号。一旦接收到来自 OS 错误检测单元 315 的检测信号,诊断结果报告单元 316 就向显示器控制器 40 输出存储器品质评估表作为 DRAM 20 的存储器诊断结果,其中该存储器品质评估表由存储器品质评估表制备单元 312 在步骤 S21 中更新。随后,在显示器 50 上显示存储器品质评估表,如步骤 S24,以此结束存储器诊断处理。相反,如果未检测到 OS 启动错误,即步骤 S23 中为否,则 OS 错误检测单元 315 完成存储器诊断处理。

[0082] 如上所述,在此实施例中,存储器诊断单元 311 执行存储器诊断程序 32a (即第一存储器诊断程序),以对 DRAM 20 进行诊断。故障区确定单元 313 基于存储器诊断单元 311 的诊断结果确定 DRAM 20 中的故障区。存储器诊断单元 311 和故障区确定单元 313 用作第一存储器诊断单元。此外,诊断程序加载单元 314 将存储器诊断单元 311 接着将要执行的存储器诊断程序 32b 加载到 DRAM 20 中除了故障区之外的部分区域中。存储器诊断单元 311 通过执行由诊断程序加载单元 314 加载的存储器诊断程序 32b,重新诊断 DRAM 20 中除了由故障区确定单元 313 确定的故障区之外的区域。诊断程序加载单元 314 和存储器诊断单元 311 用作第二存储器诊断单元。因此,该实施例可以避免将存储器诊断程序加载主存储器的故障区中(在故障区中存储功能不能正常工作),以此防止出现不能正常执行存储器诊断程序的情形。

[0083] 而且,用作第一存储器诊断单元的存储器诊断单元 311 在没有将存储器诊断程序

32a 加载到 DRAM 20 的情况下运行存储器诊断程序 32a。因此,在未确定主存储器是否包含故障区的情况下,不必将存储器诊断程序加载到主存储器中。这可避免不能正常执行存储器诊断程序的情况。

[0084] 作为结果,这样就简化了对无故障区(healthy area)和故障区的确定,实现了高效的存储器诊断处理。由于存储器品质评估表制备单元 312 制备的存储器品质评估表将由存储器诊断单元 311 进行的存储器诊断的结果与 DRAM 20 的地址信息相关联,所以可容易地区分故障区和除故障区之外的区域。这样就能够实现更高效的存储器诊断处理。

[0085] 故障区确定单元 313 在用作第二存储器诊断单元时,提取 DRAM 20 上被存储器诊断单元 311 诊断为错误的区域的地址信息。从 DRAM 20 的用来加载第二存储器诊断程序 32b 的区域中排除与所提取的地址信息对应的有故障的区域。结果,就能很容易地区分故障区和除故障区之外的区域,从而很容易地确定故障区,以此实现更高效的存储器诊断处理。

[0086] OS 错误检测单元 315 检测 OS 启动错误。当 OS 错误检测单元 315 检测 OS 启动错误时,诊断结果报告单元 316 报告存储器诊断单元 311 的诊断结果。结果,由于用户可以确认存储器诊断单元 311 的诊断结果,所以用户可以立即找到 DRAM 20 中的故障区并采取适当措施,如将故障区进行调换。

[0087] 每个功能单元 311-316 都在通电时启动,并被并入到 BIOS 程序 31 中, BIOS 程序 31 控制将被输入到 DRAM 20 和从 DRAM 20 输出的数据。而且,在 BIOS 程序 31 启动时,这些功能单元都被执行。因此,在 OS 启动之前,可以执行 DRAM 20 的存储器诊断处理。即使在 OS 不启动时,也可以正常执行存储器诊断程序。

[0088] 到此描述了根据本发明的多个实施例。在权利要求所公开的概念范围内,本发明还可以涵盖与上述描述有所区别的各种实施例。

[0089] 例如,在实施例中,第一存储器诊断程序在没有被第一存储器诊断单元加载到主存储器的情况下即可被执行。但这并不是唯一的单个实施例。本发明还涵盖下面的实施例。若第一存储器诊断程序当被加载到主存储器中时占据的是较小的区域,则由第一存储器诊断单元加载到主存储器中。这样,可以首先执行对主存储器的简单诊断。在这种情况下,使用第一存储器诊断单元的诊断结果,分别由第二存储器诊断单元和第三存储器诊断单元将第二和第三存储器诊断程序连续加载到主存储器中,其中所述第二和第三存储器诊断程序当被加载到主存储器中时会占据较大的区域。这样就能够实现高级存储器诊断功能。

[0090] 在实施例中,描述的是对 DRAM、主存储器进行诊断的存储器诊断处理。但这并不是唯一的单个实施例。本发明还涵盖下面的实施例。还可以应用用来诊断设置在 CPU 内部并能够高速读出、写入的高速缓冲存储器(cash memory)的存储器诊断处理。

[0091] 实施例中的 BIOS 程序 31 和存储器诊断程序 32 不一定必须存储在 ROM30 中。例如,这些程序可以存储在诸如闪存等可重写、非易失性存储器中,并且可以被 CPU 10 读出和执行。BIOS 程序 31 和存储器诊断程序 32 还可以从经互联网连接的服务器下载到闪存中。而且,这些程序可以存储在诸如 CD-ROM 等存储介质中,并且可以通过其驱动器从存储介质读入到闪存中。

[0092] 根据本发明的所述实施例,执行第一存储器诊断程序以便执行对主存储器的简单诊断,其中第一存储器诊断程序不使用主存储器或者只占据主存储器的一小部分。利用诊断结果可以识别出第二诊断程序的设置位置,其中该第二诊断程序需要被加载到主存储器

中并且能够执行详细诊断。结果,所述实施例可以降低第二存储器诊断程序被加载到主存储器的故障区中的可能性,并避免无法执行第二存储器诊断程序的情况。

[0093] 本文记载的所有例子和条件性语言都旨在用于教示目的,以帮助读者理解发明者为改进现有技术而提出的发明和概念,应将所有例子和条件性语言解读为不局限于这些具体记载的例子和条件,说明书中对这些例子的组织也不涉及对本发明的优点和缺点的示意。尽管已经详细描述了本发明的实施例,但应理解在不偏离本发明的精神和范围的情况下,可对其进行各种变化、替代和更改。

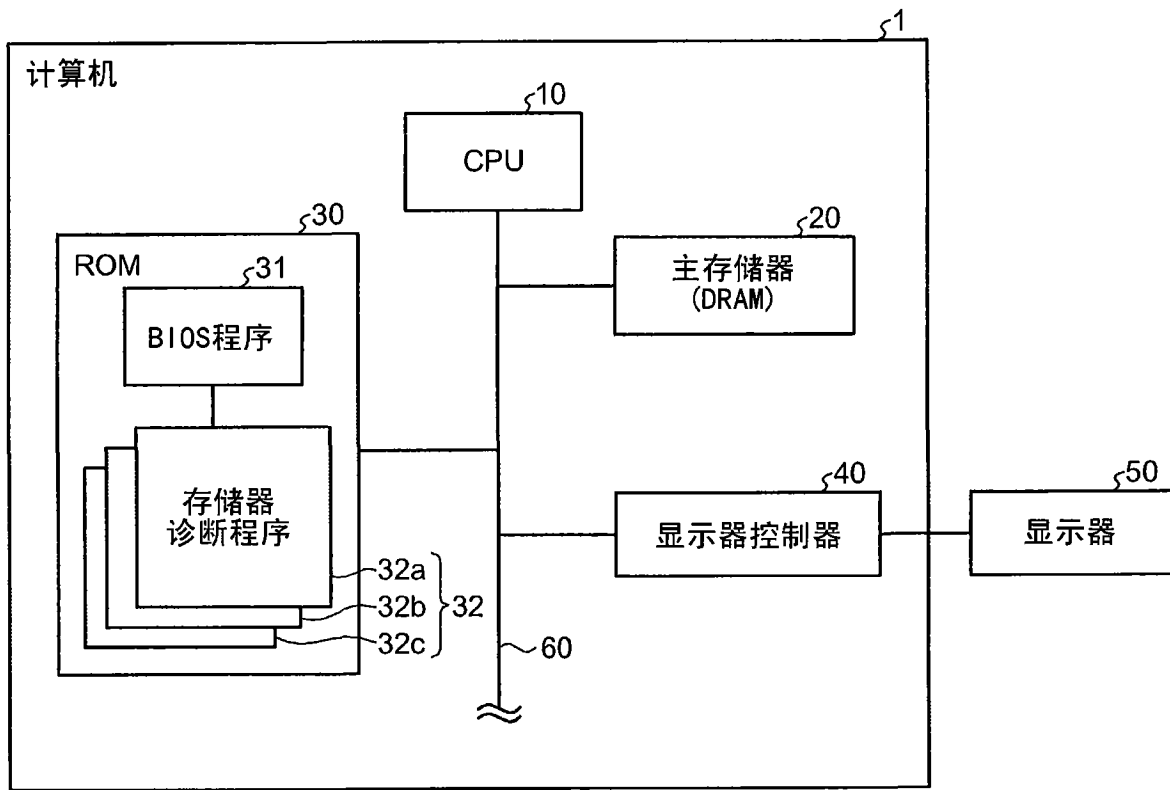


图 1

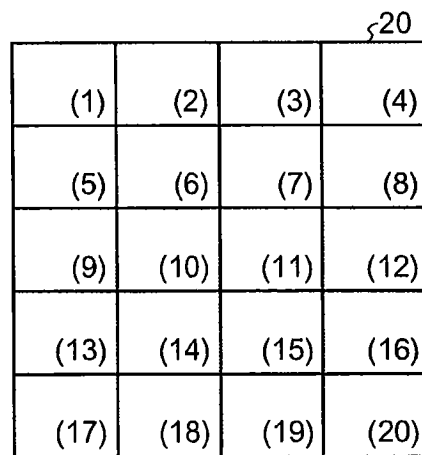


图 2

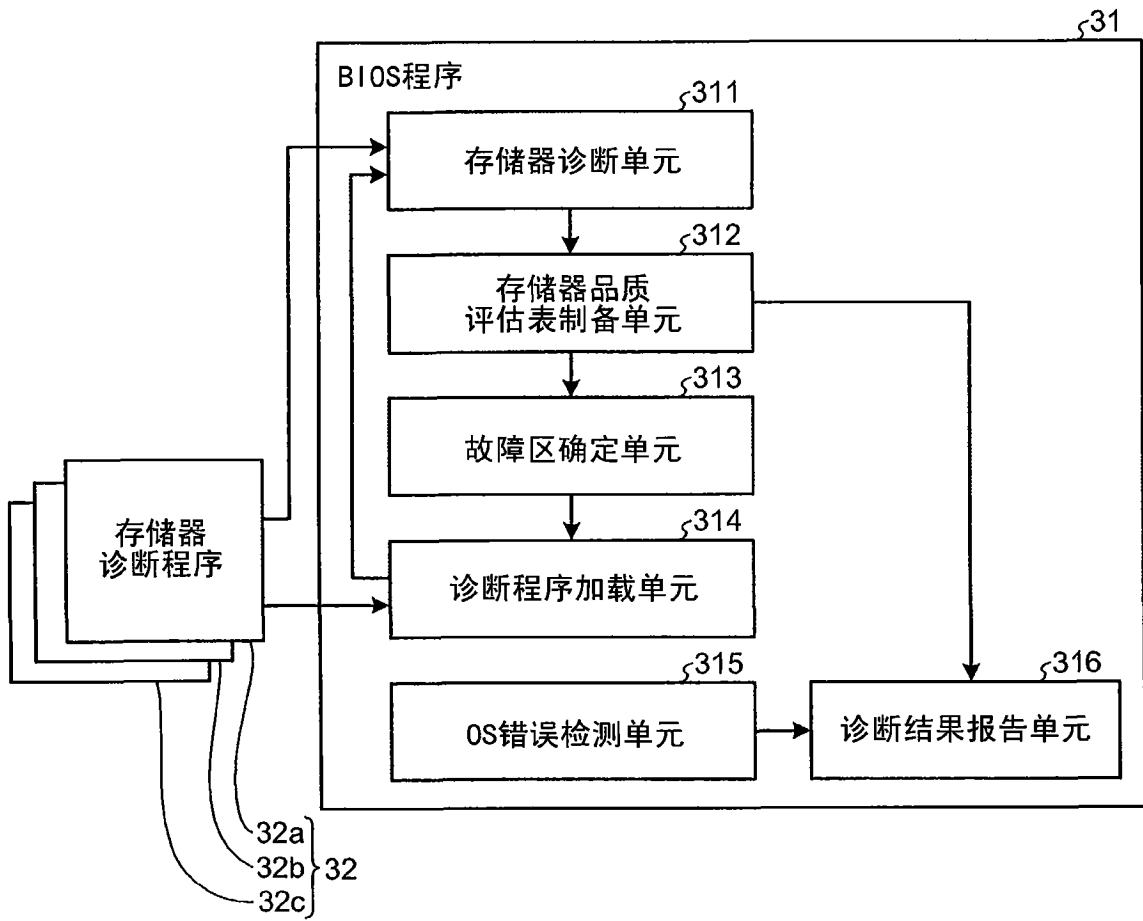


图 3

存储器地址	诊断结果
1	1
2	1
3	0
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	1
17	1
18	1
19	0
20	1

图 4A

s20

1 <sub>(1)</sub>	1 <sub>(2)</sub>	<del>0<sub>(3)</sub></del>	1 <sub>(4)</sub>
1 <sub>(5)</sub>	1 <sub>(6)</sub>	1 <sub>(7)</sub>	1 <sub>(8)</sub>
1 <sub>(9)</sub>	1 <sub>(10)</sub>	1 <sub>(11)</sub>	1 <sub>(12)</sub>
1 <sub>(13)</sub>	1 <sub>(14)</sub>	1 <sub>(15)</sub>	1 <sub>(16)</sub>
1 <sub>(17)</sub>	1 <sub>(18)</sub>	<del>0<sub>(19)</sub></del>	1 <sub>(20)</sub>

图 4B

存储器地址	诊断结果
1	2
2	2
3	0
4	2
5	1
6	2
7	2
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2
16	2
17	2
18	2
19	0
20	1

图 4C

s<sup>20</sup>

<del>2<sub>(1)</sub></del>	2 <sub>(2)</sub>	<del>0<sub>(3)</sub></del>	2 <sub>(4)</sub>
<del>1<sub>(5)</sub></del>	2 <sub>(6)</sub>	2 <sub>(7)</sub>	2 <sub>(8)</sub>
2 <sub>(9)</sub>	2 <sub>(10)</sub>	2 <sub>(11)</sub>	2 <sub>(12)</sub>
2 <sub>(13)</sub>	2 <sub>(14)</sub>	2 <sub>(15)</sub>	2 <sub>(16)</sub>
2 <sub>(17)</sub>	2 <sub>(18)</sub>	<del>0<sub>(19)</sub></del>	<del>1<sub>(20)</sub></del>

图 4D

存储器地址	诊断结果
1	3
2	3
3	0
4	2
5	1
6	3
7	3
8	3
9	3
10	3
11	3
12	3
13	3
14	3
15	3
16	3
17	3
18	3
19	0
20	1

图 4E

3 <sub>(1)</sub>	3 <sub>(2)</sub>	<del>0<sub>(3)</sub></del>	<del>2<sub>(4)</sub></del>
<del>1<sub>(5)</sub></del>	3 <sub>(6)</sub>	3 <sub>(7)</sub>	3 <sub>(8)</sub>
3 <sub>(9)</sub>	3 <sub>(10)</sub>	3 <sub>(11)</sub>	3 <sub>(12)</sub>
3 <sub>(13)</sub>	3 <sub>(14)</sub>	3 <sub>(15)</sub>	3 <sub>(16)</sub>
3 <sub>(17)</sub>	3 <sub>(18)</sub>	<del>0<sub>(19)</sub></del>	<del>1<sub>(20)</sub></del>

图 4F

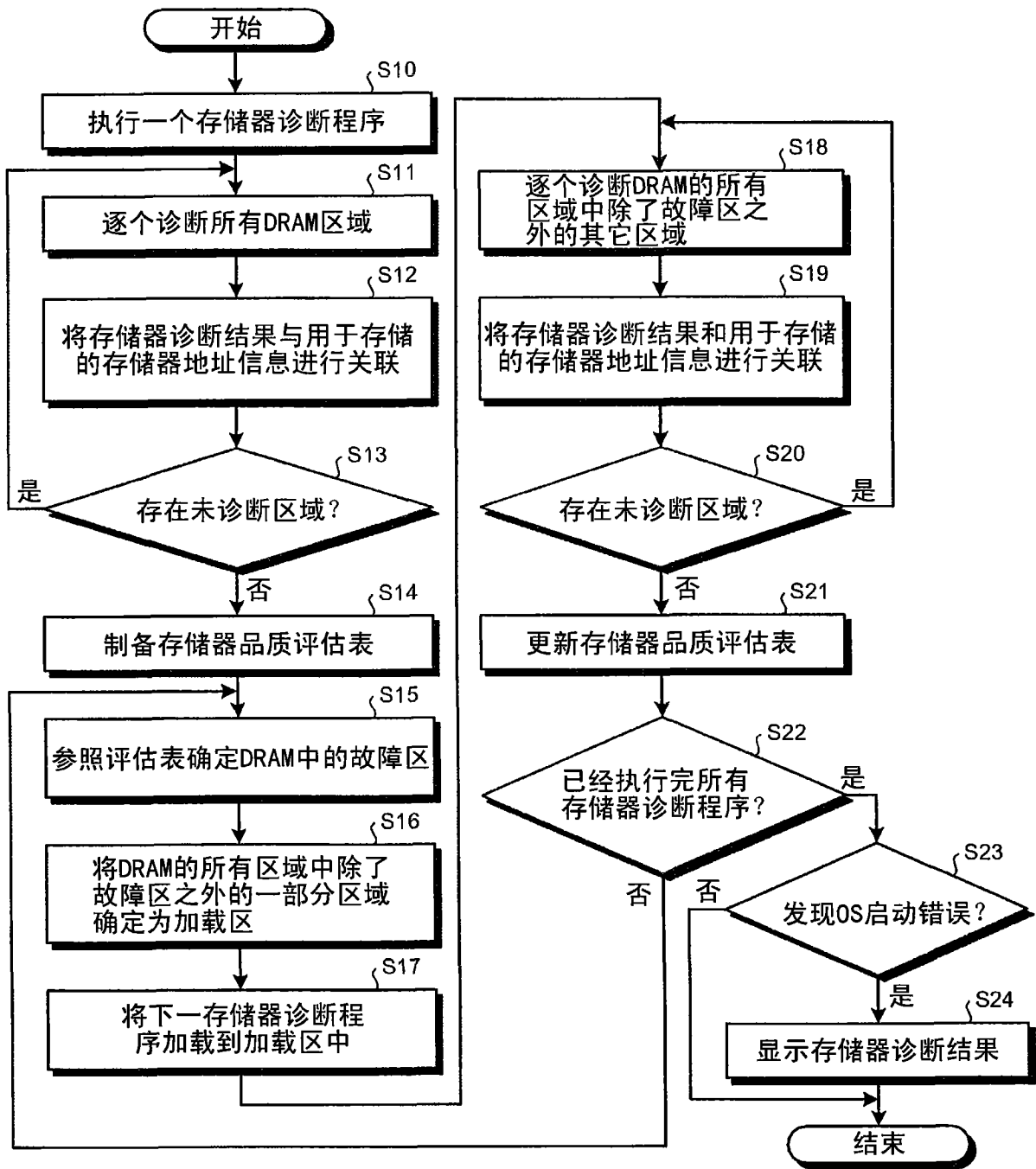


图 5