



[12] 发明专利说明书

专利号 ZL 200510073073.4

[45] 授权公告日 2007 年 12 月 19 日

[11] 授权公告号 CN 100356335C

[22] 申请日 2005.5.27

[21] 申请号 200510073073.4

[30] 优先权

[32] 2004.5.28 [33] US [31] 10/857,459

[73] 专利权人 国际商业机器公司

地址 美国纽约

[72] 发明人 A·基塔莫恩 A·库尔卡尼

G·D·麦金托什 G·L·鲁泽克

[56] 参考文献

US2003/0131039A1 2003.7.10

US6711700B2 2004.3.23

US6493837B1 2002.12.10

US2003/0056155A1 2003.3.20

审查员 李俊

[74] 专利代理机构 北京市中咨律师事务所

代理人 于静 李峥

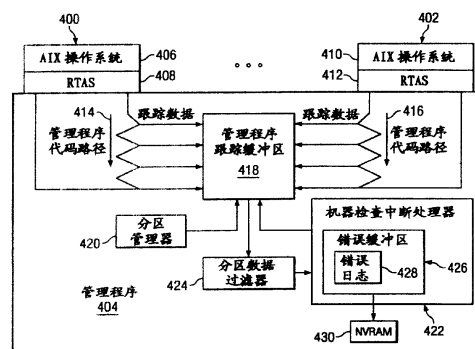
权利要求书 3 页 说明书 12 页 附图 5 页

[54] 发明名称

保存跟踪数据的方法和装置

[57] 摘要

一种用于在逻辑分区数据处理系统中处理跟踪数据的方法、装置和计算机指令。引起异常的分区会被标识以响应检测到所述异常。所述分区是所述逻辑分区数据处理系统内一组分区中的一个分区。所述被标识的分区的所述跟踪数据被存储在错误日志或其他数据结构中，以供机器检查中断处理器使用。



1. 一种在逻辑分区数据处理系统中处理跟踪数据的方法，所述方法包括：

响应于检测到异常，标识所述逻辑分区数据处理系统内的一组分区中与所述异常关联的分区以形成被标识的分区；以及

将所述被标识的分区的所述跟踪数据存储在错误日志中以供机器检查中断处理器使用，其中所述错误日志包括在机器检查中断处理器中。

2. 根据权利要求1的方法，进一步包括：

将所述错误日志存储在非易失性存储装置中。

3. 根据权利要求2的方法，其中所述非易失性存储装置是非易失性随机存取存储器。

4. 根据权利要求1的方法，其中所述标识所述分区的步骤包括：

标识最新跟踪数据表项中的分区标识符。

5. 根据权利要求1的方法，其中所述标识步骤和所述存储步骤由平台固件来执行。

6. 根据权利要求1的方法，其中所述存储步骤由所述机器检查中断处理器来执行。

7. 根据权利要求1的方法，其中所述标识步骤由分区数据过滤器来执行。

8. 根据权利要求1的方法，其中只有所述被标识的分区的所述跟踪数据被存储在所述错误日志中。

9. 根据权利要求1的方法，其中所述跟踪数据初始位于平台固件跟踪缓冲区中。

10. 一种用于处理跟踪数据的逻辑分区数据处理系统，所述逻辑分区数据处理系统包括：

标识装置，所述标识装置用于响应于检测到异常，标识所述逻辑分区数据处理系统内的一组分区中与所述异常关联的分区以形成被标识的分区；以及

存储装置，所述存储装置用于将所述被标识的分区的所述跟踪数据存储在错误日志中以供机器检查中断处理器使用，其中所述错误日志包括在

机器检查中断处理器中。

11. 根据权利要求 10 的数据处理系统，其中所述存储装置是第一存储装置并且进一步包括：

用于将所述错误日志存储在非易失性存储装置中的第二存储装置。

12. 根据权利要求 11 的数据处理系统，其中所述非易失性存储装置是非易失性随机存取存储器。

13. 根据权利要求 10 的数据处理系统，其中所述标识装置包括：

用于标识最新跟踪数据表项中的分区标识符的标识装置。

14. 根据权利要求 10 的数据处理系统，其中所述标识装置和所述存储装置位于平台固件中。

15. 根据权利要求 10 的数据处理系统，其中所述存储装置位于所述机器检查中断处理器中。

16. 根据权利要求 10 的数据处理系统，其中所述标识装置位于分区数据过滤器中。

17. 根据权利要求 10 的数据处理系统，其中只有所述被标识的分区的所述跟踪数据被存储在所述错误日志中。

18. 根据权利要求 10 的数据处理系统，其中所述跟踪数据初始位于平台固件跟踪缓冲区中。

19. 一种用于处理跟踪数据的设备，所述设备包括：

第一装置，所述第一装置用于响应于检测到异常，标识逻辑分区数据处理系统内的一组分区中与所述异常关联的分区以形成被标识的分区；以及

第二装置，所述第二装置用于将所述被标识的分区的所述跟踪数据存储在错误日志中以供机器检查中断处理器使用，其中所述错误日志包括在机器检查中断处理器中。

20. 根据权利要求 19 的设备，进一步包括：

用于将所述错误日志存储在非易失性存储装置中的第三装置。

21. 根据权利要求 20 的设备，其中所述非易失性存储装置是非易失性随机存取存储器。

22. 根据权利要求 19 的设备，其中所述第一装置包括：

用于标识最新跟踪数据表项中的分区标识符的子装置。

23. 根据权利要求 19 的设备，其中所述第一装置和所述第二装置由平台固件来操作。

24. 根据权利要求 19 的设备，其中所述第二装置由所述机器检查中断处理器来操作。

25. 根据权利要求 19 的设备，其中所述第一装置由分区数据过滤器来操作。

26. 根据权利要求 19 的设备，其中只有所述被标识的分区的所述跟踪数据被存储在所述错误日志中。

27. 根据权利要求 19 的设备，其中所述跟踪数据初始位于平台固件跟踪缓冲区中。

28. 一种逻辑分区数据处理系统，所述系统包括：

总线系统；

存储器，所述存储器与所述总线系统相连，其中所述存储器包括一组指令；以及

处理单元，所述处理单元与所述总线系统相连，其中所述处理单元执行所述指令组，以便标识所述逻辑分区数据处理系统内的一组分区中与异常关联的分区以形成被标识的分区，以响应检测到所述异常；以及将所述被标识的分区的跟踪数据存储在错误日志中以供机器检查中断处理器使用，其中所述错误日志包括在机器检查中断处理器中。

保存跟踪数据的方法和装置

技术领域

本发明一般涉及改进的数据处理系统，具体地说，涉及用于处理数据的方法和装置。更具体地说，本发明涉及用于在逻辑分区数据处理系统中管理跟踪数据的方法、装置和计算机指令。

背景技术

诸如国际商业机器公司的 IBM eServer P690、惠普公司的 DHP9000 Superdome Enterprise Server 以及 Sun 微系统公司的 Sunfire 15K Server 之类的日益增加的大型对称多处理器数据处理系统并非被用作单个大型数据处理系统。相反，这些类型的数据处理系统被分区并且被用作更小的系统。这些系统也被称为逻辑分区 (LPAR) 数据处理系统。数据处理系统中的逻辑分区功能允许单个操作系统的多个拷贝或多个异构操作系统在单个数据处理系统平台上同时运行。其中运行有操作系统映像的分区被分配平台资源的一个非重叠子集。这些平台可分配的资源包括一个或多个在体系结构上截然不同的处理器及其中断管理区域、系统存储器区域以及输入/输出 (I/O) 适配器总线槽。分区的资源通过平台的固件被呈现给操作系统映像。

在某一平台内运行的每个不同的操作系统或操作系统映像被相互保护，以使一个逻辑分区上的软件错误不会影响任何其他分区的正确运行。通过分配将由每个操作系统映像直接管理的平台资源的不相交集合，并通过提供用于确保各个映像均无法控制尚未分配给该映像的任何资源的机制，可以提供这种保护。此外，可以阻止在控制操作系统的已分配资源时出现的软件错误影响任何其他映像的资源。

因此，每个操作系统映像或每个不同的操作系统都直接控制平台内可分配资源的不同集合。就逻辑分区数据处理系统中的硬件资源而言，这些资源被在各个分区之间分隔地共享。这些资源可以包括例如输入/输出

(I/O) 适配器、存储器 DIMM、非易失性随机存取存储器 (NVRAM) 以及硬盘驱动器。LPAR 数据处理系统中的每个分区都可以在不必开关整个数据处理系统电源的情况下被反复引导和关闭。

当逻辑分区数据处理系统遇到故障时，需要与过程和系统状态相关的数据来帮助识别和分析故障。在当前的逻辑分区数据处理系统中，由于当前系统设计的原因，某些诊断故障所需的数据并不可用。例如，平台固件包括一个跟踪工具，其允许跟踪固件中的代码路径。逻辑分区数据处理系统中使用的平台固件的一个实例是管理程序，其可从国际商业机器公司获得。

借助当前使用的跟踪工具，当每个分区作出平台固件调用时，显示平台固件中所采取的代码路径的跟踪信息以及关键数据值被写入一个跟踪缓冲区中。当分区遇到错误并且错误路径同关键数据值一起被跟踪时，此跟踪信息尤为重要。

当前，所有逻辑分区模式数据处理系统平台都支持管理程序跟踪工具，该工具用于在管理程序执行期间，将管理程序代码执行跟踪点数据写入位于管理程序空间中的跟踪缓冲区。在系统出现故障的情况下，该管理程序跟踪数据对于现场的有效故障分析非常关键。

这种情况产生了一个问题，在大型配置中，处理器被专用于多个分区，其中这些分区写入同一缓冲区。这些缓冲区通常以循环的方式来组织。因此，如果出现分区崩溃，跟踪数据会很快地被逻辑分区数据处理系统中的其他分区所改写。结果，帮助诊断问题所需的关键数据可能会丢失。

一种解决方案是创建更大的缓冲区。进而，随着分区数量的增加，该跟踪缓冲区的大小也需要增大以容纳额外的分区。缓冲区结构必须被预先分配考虑到的最大配置，因为每个逻辑分区数据处理系统都被单独配置并允许进行动态配置。结果，对于较小的配置，存在被浪费的存储器空间。进而，在系统存储器占较高价格的系统中，浪费的空间增加了逻辑分区数据处理系统的成本。

因此，具有一种用于保存跟踪数据的改进的方法、装置和计算机指令是有利的。

发明内容

本发明提供了一种用于处理逻辑分区数据处理系统中的跟踪数据的方法、装置和计算机指令。引起异常的分区被标识以响应检测到所述异常。所述分区是所述逻辑分区数据处理系统中的一组分区中的一个分区。用于所述被标识的分区的跟踪数据被存储在用于机器检查中断处理器的错误日志或其他数据结构中。

附图说明

在所附权利要求中说明了被认为是本发明特性的新颖特征。但是，当结合附图阅读时，通过参考以下对示例性实施例的详细说明，可以最佳地理解发明本身及其优选使用方式、进一步的目的和优点，这些附图是：

图 1 是其中可以实现本发明的数据处理系统的方块图；

图 2 是其中可以实现本发明的示例性逻辑分区平台的方块图；

图 3 是用于处理跟踪数据的现有逻辑分区数据处理系统中的组件的示意图；

图 4 是根据本发明的一个优选实施例的用于管理跟踪数据以消除安全弱点的配置的示意图；

图 5 是根据本发明的一个优选实施例的用于过滤跟踪数据的过程的流程图；以及

图 6 是根据本发明的一个优选实施例的用于在错误日志中保存跟踪数据的过程的流程图。

具体实施方式

现在参考附图，具体地说，参考图 1，图 1 示出了其中可以实现本发明的数据处理系统的方块图。数据处理系统 100 可以是一个对称多处理器 (SMP) 系统，所述系统包括多个连接到系统总线 106 的处理器 101、102、103 和 104。例如，数据处理系统 100 可以是网络中作为服务器实现的 IBM eServer，其是位于纽约阿蒙克的国际商业机器公司的产品。可替代地，可以采用单处理器系统。同时连接到系统总线 106 的是存储器控制器/高速缓冲存储器 108，它提供与多个本地存储器 160-163 的接口。I/O 总线桥 110

与系统总线 106 相连并提供与 I/O 总线 112 的接口。如图所示，存储器控制器/高速缓冲存储器 108 与 I/O 总线桥 110 可以是一体的。

数据处理系统 100 是一个逻辑分区 (LPAR) 数据处理系统。因此，数据处理系统 100 可以使多个异构操作系统(或单个操作系统的多个实例)同时运行。这些多个操作系统中的每个操作系统都可以在其中执行任意数量的软件程序。数据处理系统 100 被逻辑地分区，以使不同的 PCI I/O 适配器 120-121、128-129 和 136、图形适配器 148 以及硬盘适配器 149 可以被分配给不同的逻辑分区。在此情况下，图形适配器 148 为显示装置(未示出)提供了连接，而硬盘适配器 149 提供了连接以控制硬盘 150。

因此，例如假设数据处理系统 100 被分成三个逻辑分区 P1、P2 和 P3。各个 PCI I/O 适配器 120-121、128-129、136、图形适配器 148、硬盘适配器 149、各个主机处理器 101-104 以及来自本地存储器 160-163 的存储器，都被分配给这三个分区中的每个分区。在这些实例中，存储器 160-163 可以采取双列直插存储器模块 (DIMM) 的形式。通常，并不在每一 DIMM 的基础上将 DIMM 分配给各分区。相反，某一分区将得到由平台所见的总体存储器的一部分。例如，处理器 101、来自本地存储器 160-163 的存储器的某些部分以及 I/O 适配器 120、128 和 129 可以被分配给逻辑分区 P1；处理器 102-103、来自本地存储器 160-163 的存储器的某些部分以及 PCI I/O 适配器 121 和 136 可以被分配给逻辑分区 P2；并且处理器 104、来自本地存储器 160-163 的存储器的某些部分、图形适配器 148 和硬盘适配器 149 可以被分配给逻辑分区 P3。

在数据处理系统 100 中执行的每个操作系统都被分配给不同的逻辑分区。因此，在数据处理系统 100 中执行的每个操作系统只访问在其逻辑分区之内的那些 I/O 单元。因此，例如，高级交互执行 (AIX) 操作系统的实例可以在分区 P1 中执行，所述 AIX 操作系统的第二个实例(映像)可以在分区 P2 中执行，并且 Linux 或 OS/400 操作系统可以在逻辑分区 P3 中运行。

连接到 I/O 总线 112 的外围组件互连 (PCI) 主机桥 114 提供与 PCI 局部总线 115 的接口。多个 PCI 输入/输出适配器 120-121 可以通过 PCI 至 PCI 桥 116、PCI 总线 118、PCI 总线 119、I/O 槽 170 和 I/O 槽 171 与

PCI 总线 115 相连。PCI 至 PCI 桥 116 提供与 PCI 总线 118 和 PCI 总线 119 的接口。PCI I/O 适配器 120 和 121 被分别放置在 I/O 槽 170 和 171 中。典型的 PCI 总线实现将支持四至八个 I/O 适配器（即，用于附加连接器的扩展槽）。每个 PCI I/O 适配器 120-121 都提供了数据处理系统 100 与诸如例如其他网络计算机之类的输入/输出设备（其是数据处理系统 100 的客户）之间的接口。

附加的 PCI 主机桥 122 提供用于附加的 PCI 总线 123 的接口。PCI 总线 123 与多个 PCI I/O 适配器 128-129 相连。PCI I/O 适配器 128-129 可以通过 PCI 至 PCI 桥 124、PCI 总线 126、PCI 总线 127、I/O 槽 172 和 I/O 槽 173 与 PCI 总线 123 相连。PCI 至 PCI 桥 124 提供与 PCI 总线 126 和 PCI 总线 127 的接口。PCI I/O 适配器 128 和 129 被分别放置在 I/O 槽 172 和 173 中。以这种方式，通过各个 PCI I/O 适配器 128-129，可以支持诸如例如调制解调器或网络适配器之类的附加 I/O 设备。通过这种方式，数据处理系统 100 允许到多个网络计算机的连接。

插入到 I/O 槽 174 中的存储器映射的图形适配器 148 可以通过 PCI 总线 144、PCI 至 PCI 桥 142、PCI 总线 141 和 PCI 主机桥 140 与 I/O 总线 112 相连。硬盘适配器 149 可以被放置在 I/O 槽 175 中，该 I/O 槽 175 与 PCI 总线 145 相连。依次地，该总线与 PCI 至 PCI 桥 142 相连，桥 142 又通过 PCI 总线 141 与 PCI 主机桥 140 相连。

PCI 主机桥 130 为 PCI 总线 131 提供了接口以便连接到 I/O 总线 112。PCI I/O 适配器 136 与 I/O 槽 176 相连，槽 176 通过 PCI 总线 133 与 PCI 至 PCI 桥 132 相连。PCI 至 PCI 桥 132 与 PCI 总线 131 相连。该 PCI 总线还将 PCI 主机桥 130 连接到服务处理器邮箱接口和 ISA 总线访问中转逻辑 194 以及 PCI 至 PCI 桥 132。服务处理器邮箱接口和 ISA 总线访问中转逻辑 194 转发去往 PCI/ISA 桥 193 的 PCI 访问。NVRAM 存储 192 与 ISA 总线 196 相连。服务处理器 135 通过其局部 PCI 总线 195 与服务处理器邮箱接口和 ISA 总线访问中转逻辑 194 相连。服务处理器 135 还通过多个 JTAG/I²C 总线 134 与处理器 101-104 相连。JTAG/I²C 总线 134 是 JTAG/扫描总线（参见 IEEE 1149.1）和 Phillips I²C 总线的组合。但是，可替代地，JTAG/I²C 总线 134 可以由单独的 Phillips I²C 总线或单独的 JTAG/

扫描总线来代替。主机处理器 101、102、103 和 104 的所有 SP-ATTN 信号都被一起连接到服务处理器的中断输入信号。服务处理器 135 具有其自己的本地存储器 191，并可访问硬件操作面板 190。

当数据处理系统 100 最初接通电源时，服务处理器 135 使用 JTAG/T²C 总线 134 来询问系统（主机）处理器 101-104、存储器控制器/高速缓冲存储器 108 以及 I/O 桥 110。完成该步骤之后，服务处理器 135 已了解数据处理系统 100 的资源（inventory）和拓扑。服务处理器 135 还在通过询问主机处理器 101-104、存储器控制器/高速缓冲存储器 108 以及 I/O 桥 110 而发现的所有部件上执行内置自测试（BIST）、基本正确性测试（BAT）和存储器测试。BIST、BAT 和存储器测试期间检测到的故障的所有错误信息都由服务处理器 135 来收集和报告。

如果在取出 BIST、BAT 和存储器测试期间被发现具有故障的部件之后，有意义/有效的系统资源配置仍然是可能的，则数据处理系统 100 被允许继续将可执行代码载入本地（主机）存储器 160-163。然后，服务处理器 135 释放主机处理器 101-104 以便执行被载入本地存储器 160-163 的代码。当主机处理器 101-104 从数据处理系统 100 内的各自的操作系统来执行代码时，服务处理器 135 进入监视和报告错误的模式。由服务处理器 135 监视的项目类型包括例如冷却风扇转速和运行、热传感器、电源调节器以及由处理器 101-104、本地存储器 160-163 和 I/O 桥 110 所报告的可恢复或不可恢复的错误。

服务处理器 135 负责保存和报告与数据处理系统 100 中所有被监视项目相关的错误信息。服务处理器 135 还可根据错误类型和已定义的阈值来采取行动。例如，服务处理器 135 可能注意到某一处理器的高速缓冲存储器中出现过多的可恢复错误并判定这是硬故障的预兆。基于此判定，服务处理器 135 可以标记该资源，以便在当前运行会话期和将来初始程序装入（IPL）期间取消配置。IPL 有时也被称为“引导”或“引导程序”。

可以使用各种商用计算机系统来实现数据处理系统 100。例如，可以使用国际商业机器公司的 IBM eServer iSeries Model 840 系统来实现数据处理系统 100。此类系统可以支持使用 OS/400 操作系统（其也可从国际商业机器公司获得）进行逻辑分区。

本领域的技术人员将理解，图 1 中所示的硬件可以有所变化。例如，除所示硬件之外或替代所示硬件，还可以使用诸如光盘驱动器之类的其他外围设备。所示实例并非旨在暗示本发明的体系结构限制。

现在参考图 2，图 2 示出了其中可以实现本发明的示例性逻辑分区平台的方块图。逻辑分区平台 200 中的硬件可以被实现为例如图 1 中的数据
处理系统 100。逻辑分区平台 200 包括分区硬件 230、操作系统 202、204、206、208 以及分区管理固件 210。操作系统 202、204、206 和 208 可以是在逻辑分区平台 200 上同时运行的单个操作系统的多个副本或多个异构的操作系统。可以使用 OS/400 来实现这些操作系统，它们被设计成与诸如管理程序之类的分区管理固件相互作用。OS/400 在这些示例性实施例中仅用作一个实例。当然，取决于特定的实施方式，可以使用诸如 AIX 和 Linux 之类的其他类型的操作系统。操作系统 202、204、206 和 208 位于分区 203、205、207 和 209 中。管理程序软件是可以用来实现分区管理固件 210 的软件的一个实例，并可从国际商业机器公司获得。固件是存储在没有电力的情况下也可保持其内容的存储器芯片中的“软件”，所述存储器芯片诸如，例如，只读存储器（ROM）、可编程 ROM（PROM）、可擦写可编程 ROM（EPROM）、电可擦写可编程 ROM（EEPROM）以及非易失性随机存取存储器（非易失性 RAM）。

此外，这些分区还包括分区固件 211、213、215 和 217。分区固件 211、213、215 和 217 可以使用初始引导程序代码、IEEE-1275 标准开放固件以及运行时抽象软件（RTAS，其可从国际商业机器公司获得）来实现。当分区 203、205、207 和 209 被实例化时，由平台固件 210 将引导程序代码的副本加载到分区 203、205、207 和 209。此后，控制被传递给引导程序代码，然后引导程序代码加载开放固件和 RTAS。与分区关联或分配给分区的处理器然后被指派给分区的存储器以执行分区固件。

分区硬件 230 包括多个处理器 232-238、多个系统存储器单元 240-246、多个输入/输出（I/O）适配器 248-262 以及存储单元 270。各个处理器 232-238、存储器单元 240-246、NVRAM 存储装置 298 以及 I/O 适配器 248-262 可以被分配给逻辑分区平台 200 内的多个分区中的一个分区，每个分区与操作系统 202、204、206 和 208 中的一个操作系统相对应。

分区管理固件 210 为分区 203、205、207 和 209 执行多种功能和服务，以创建和实施对逻辑分区平台 200 的分区。分区管理固件 210 是一种与底层硬件相同的固件实现的虚拟机。因此，分区管理固件 210 允许通过虚拟化逻辑分区平台 200 的所有硬件资源来同时执行独立的操作系统映像 202、204、206 和 208。

服务处理器 290 可以用于提供诸如处理分区中的平台错误之类的各种服务。这些服务还可用作服务代理以将错误报告给供应商（例如国际商业机器公司）。不同分区的运行可以通过诸如硬件管理控制台 280 之类的硬件管理控制台来进行控制。硬件管理控制台 280 是一个单独的数据处理系统，系统管理员通过它可以执行各种功能，包括将资源重新分配给不同的分区。

现在参考图 3，图 3 示出了用于处理跟踪数据的现有逻辑分区数据处理系统中的组件的示意图。在此示例性的实例中，分区 300 包含操作系统 302 和 RTAS 304。分区 306 包含操作系统 308 和 RTAS 310。在这些示出的实例中，最多可以存在 255 个分区。所有这些分区都通过诸如管理程序 312 之类的平台固件来进行管理。

在分区 300 调用管理程序 312 时，将生成管理程序代码路径 318。从这些调用生成的跟踪数据被存储在管理程序跟踪缓冲区 316 中。此跟踪数据被存储在图 4 的管理程序跟踪缓冲区 418 的表项中。跟踪数据包含这些实例中显示所采取的代码路径的跟踪信息和关键数据值。除了跟踪数据之外，每个表项还包括分区标识符以标识作出调用（其产生该跟踪数据）的分区。当出现代码路径时，分区管理器 314 还将跟踪信息写入管理程序跟踪缓冲区 316。当该分区管理器 314 调用管理程序 312 时，此跟踪信息被写入。分区管理器 314 是管理程序 312 中的一个组件。该组件用于管理分区并包括诸如开始和终止分区之类的功能。由分区管理器 314 作出的调用被存储在管理程序跟踪缓冲区 316 中。

以类似的方式，当由分区 306 作出对管理程序 312 的调用时，将形成管理程序代码路径 320，同时用于这些调用的跟踪数据被存储在管理程序跟踪缓冲区 316 中。以类似的方式，当管理程序 312 收到分区 306 的调用时，将形成管理程序代码路径 320。用于此路径的跟踪数据也被存储在管

理程序跟踪缓冲区 316 中。

当每个分区被激活时，操作系统被加载并开始执行。当诸如操作系统 302 之类的操作系统需要平台资源时，操作系统作出对 RTAS 304 的 RTAS 调用，这依次作出对管理程序 312 的管理程序调用。当管理程序调用执行时，将执行特殊的“跟踪点”，其中跟踪数据被写入管理程序跟踪缓冲区 316 而不隔离分区数据。换言之，用于由所有分区作出的所有调用的所有跟踪数据都被置于此缓冲区中。

不仅是代表分区执行管理程序调用的管理程序数据被写入单个缓冲区，而且分区管理器 314（其是管理程序工具）也将其跟踪数据写到同一跟踪缓冲区。此外，机器检查中断处理器 322 使用相同的跟踪工具并将信息存储在管理程序跟踪缓冲区 316 中。

然而，此现有系统将所有信息都存储在管理程序跟踪缓冲区 316 中。结果，由于这些类型的缓冲区中的有限空间，在现有系统中数据可能会被覆盖。在大型配置中，当所有处理器都专用于多个分区且写入以类似方式组织的同一缓冲区时，上述问题会增加。如果发生分区崩溃，管理程序代码路径 318 中的跟踪数据将很快由其他分区覆盖，造成可以帮助诊断问题的关键数据丢失。

现在参考图 4，图 4 示出了根据本发明的一个优选实施例的用于管理跟踪数据以消除安全弱点的配置的示意图。在此示例性的实例中，存在分区 400 和分区 402 并通过管理程序 404 来进行管理。分区 400 包含操作系统 406 和 RTAS 408，而分区 402 包含操作系统 410 和 RTAS 412。当分区 400 或分区 402 产生对管理程序 404 的调用时，将生成管理程序代码路径 414 和管理程序代码路径 416，同时跟踪数据被存储在管理程序跟踪缓冲区 418 中。

进而，分区管理器 420 可以产生调用，其中跟踪数据被存储在管理程序跟踪缓冲区 418 中。机器检查中断处理器 422 也将数据存储在管理程序跟踪缓冲区 418 中。

机器检查中断处理器 422 包括错误缓冲区 426。该缓冲区包括错误日志 428。该错误日志包括一个错误专用部分，该部分具有可以由选定或特定分析程序使用的格式。该日志还包括一个可以用于存储示例性实例中的

跟踪数据的用户区。

在示例性实例中，每次出现机器检查异常时，管理程序跟踪缓冲区 418 中的跟踪数据便被复制到错误缓冲区 426 内的错误日志 428 中。在这些示例性实例中，复制由机器检查中断处理器 422 来执行。异常是更改程序中正常控制流的错误情况。异常可以由硬件或软件产生或引起。硬件异常包括例如重置、中断或来自存储器管理单元的信号。异常可以由算术逻辑单元或浮点单元的诸如除零、溢出、下溢之类的数字错误或诸如特权指令、保留指令、陷阱指令或未定义指令之类的指令解码错误来产生。

很大比例的分区崩溃也会产生机器检查异常。本发明的机制使用这类事件来获得跟踪数据并以避免跟踪数据丢失的方式来存储跟踪数据。具体地说，在这些示例性实例中，并没有使用错误日志 428 的用户部分。该部分被用于存储跟踪数据。错误日志的任何未使用的部分或错误缓冲区 426 中的其他未使用的空间都可以被用于存储来自管理程序跟踪缓冲区 418 的跟踪数据。每次出现机器检查异常时，机器检查中断处理器 422 将跟踪缓冲区数据写到错误日志 428 的未使用部分。此特性允许在错误日志被传递给所有运行分区时可以容易地使用这些数据进行调试。

只需对机器检查中断处理器 422 进行少量简单编码更改即可包括此附加功能。在由异常启动的机器检查过程中，指向跟踪数据的当前指针被用于将引起机器检查的最后分区数据从管理程序跟踪缓冲区 418 复制到错误日志 428 中。可以对跟踪数据进行过滤以便只将其中发生异常的分区中的跟踪数据复制到错误日志 428 中。在这些示例性实例中，此过滤可以在管理程序 404 内的分区数据过滤器 424 中实现。分区数据过滤器 424 通过检查在管理程序跟踪缓冲区 418 的最后跟踪表项中发现的分区标识符来标识其中发生异常的分区。此 ID 被用于过滤由机器检查中断处理器 422 请求的跟踪数据。当然，可以在其他位置实现此过滤器，例如在机器检查中断处理器 422 内。

如果分区已终止，则当机器检查中断处理器 422 被调用时，分区数据仍将存在于管理程序跟踪缓冲区 418 中。机器检查中断处理器 422 将跟踪数据复制到错误日志 428 来保存跟踪数据以用于调试目的。根据错误日志 428 中可用空间的数量，并非所有跟踪数据都可以被从管理程序跟踪缓冲

区 418 复制到错误日志 428 中。机器检查中断处理器 422 将数据复制到错误日志 428 中，直到所有跟踪数据都已被复制或者直到错误日志 428 中不再有可用空间。

在机器检查中断处理器 422 将分区跟踪数据复制到错误日志 428 之后，该日志被写入诸如 NVRAM 430 之类的非易失性存储装置中。该日志由特殊的错误收集例程来复制。然后该例程将该日志复制到所有活动分区以进行分析。在这些实例中，所述例程是机器检查中断处理器的一部分。

现在参考图 5，图 5 示出了根据本发明的一个优选实施例的用于过滤跟踪数据的过程的流程图。图 5 中所示的过程可以在诸如图 4 的分区数据过滤器 424 之类的过滤过程中实现。当收到来自诸如图 4 的管理程序跟踪缓冲区 418 之类的跟踪缓冲区的数据请求时，该过程被启动。在这些示例性实例中，通过机器检查中断处理器来接收所述请求。

所述过程开始于接收对跟踪数据的请求（步骤 500）。接着，标识最后跟踪数据表项中的分区标识符（PID）（步骤 502）。然后，选择跟踪数据表项（步骤 504）。所选的表项是尚未被发送给请求程序的表项。在这些示例性实例中，从最新到最旧来选择各表项。换言之，首先选择最新的表项用于传输。

接着，从所选的跟踪数据表项获取数据（步骤 506）。然后，判定数据是否用于标识的 PID（步骤 508）。如果数据用于标识的 PID，则数据被返回机器检查中断处理器（步骤 510）。接着，判定是否存在对跟踪数据的更多请求（步骤 512）。如果不存在对跟踪数据的更多请求，则此后过程终止。

返回参考步骤 508，如果数据不可用于标识的 PID，则选择下一个跟踪数据表项以进行处理（步骤 514），然后过程返回如上所述的步骤 506。在步骤 512，如果对跟踪数据作出了更多请求，则过程返回如上所述的步骤 514。

现在参考图 6，图 6 示出了根据本发明的一个优选实施例的用于在错误日志中保存跟踪数据的过程的流程图。图 6 所示的过程可以在诸如图 4 的机器检查中断处理器 422 之类的机器检查中断处理器中实现。当出现异常需要机器检查中断处理器进行处理时，该过程被启动。

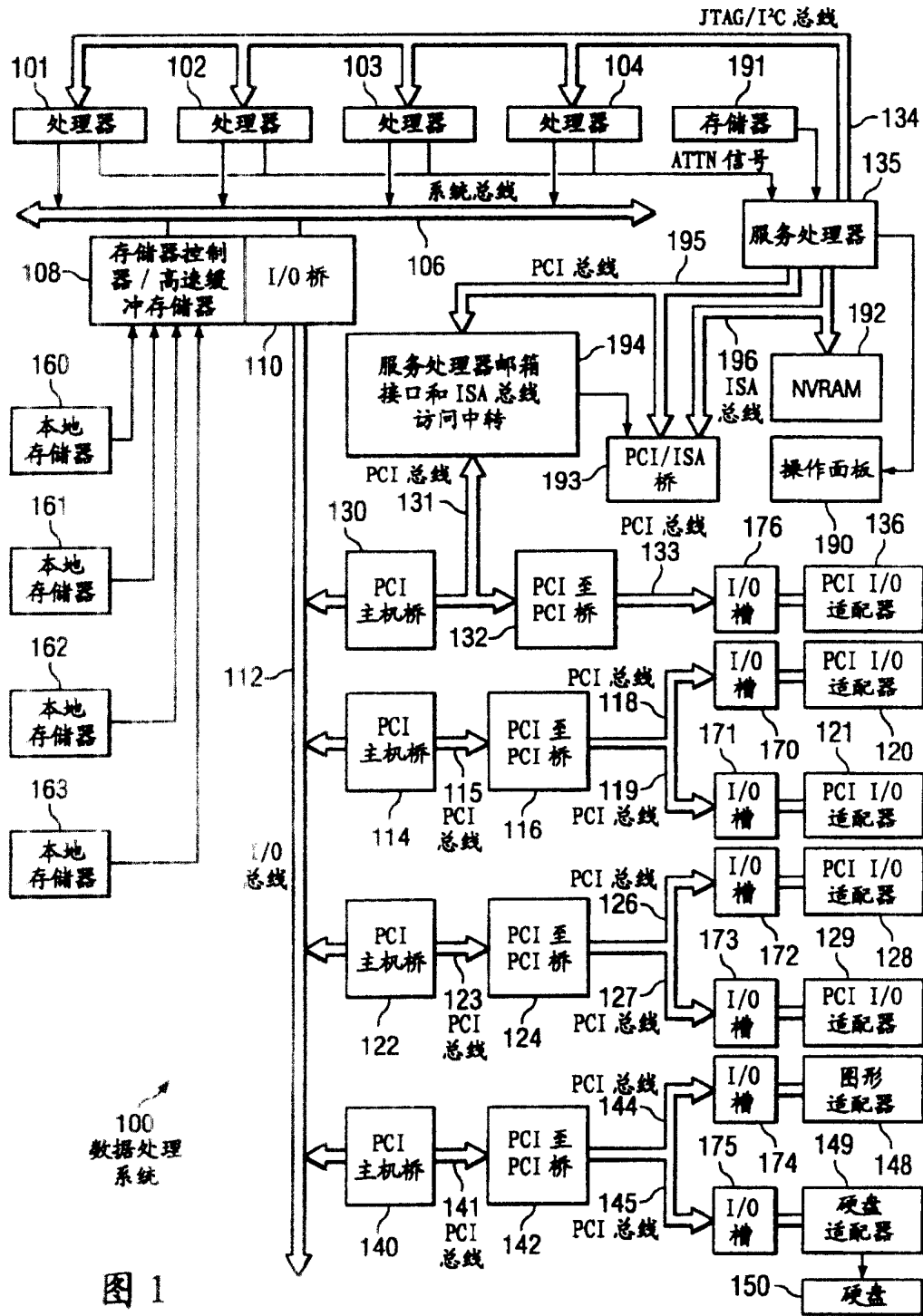
所述过程开始于标识指向平台固件跟踪数据的指针（步骤 600）。该指针指向跟踪数据缓冲区中的当前记录。接着，从跟踪缓冲区请求跟踪数据缓冲区中用于记录的跟踪数据（步骤 602）。该指针可以被递减以请求跟踪缓冲区中的不同记录以便进行检查。响应于该请求，判定是否接收到跟踪数据（步骤 604）。在这些实例中，将返回跟踪数据或说明没有更多数据存在的指示以响应该请求。然后，如果接收到跟踪数据，该数据被复制到错误日志（步骤 606）。接着，判定错误日志中是否有更多的可用空间（步骤 608）。如果错误日志中没有更多的可用空间，则过程终止。

返回参考步骤 608，如果错误日志中存在更多的空间，则过程返回如上所述的步骤 602。再次参考步骤 604，如果没有接收到跟踪数据，则过程终止。

因此，本发明提供了用于处理跟踪数据的改进的方法、装置和计算机指令。本发明的机制通过使用现有的错误日志空间存储跟踪数据来保存跟踪数据。在这些实例中，错误日志是与机器检查中断处理器关联的日志，或者是创建的以便由机器检查中断处理器使用的日志。每当发生将由机器检查中断处理器处理的异常时，数据被获取。此机制还过滤已获取的数据，以便只有来自引起异常的分区的数据被返回。

重要的是注意到，虽然在完整功能的数据处理系统的情况下说明了本发明，本领域的技术人员将理解，可以以指令的计算机可读介质的形式和各种形式来发布本发明的诸过程，并且本发明均可等同地适用而与实际用于执行发布的信号承载介质的特定类型无关。计算机可读介质的实例包括可记录型介质，诸如软盘、硬盘驱动器、RAM、CD-ROM、DVD-ROM 和传输型介质，诸如数字和模拟通信链路、使用诸如例如射频和光波传输之类的传输形式的有线或无线通信链路。计算机可读介质可以采取编码格式的形式，可以对其解码以便在特定数据处理系统内实际地使用。

出于示例和说明目的给出了对本发明的描述，并且所述描述并非旨在是穷举的或是将本发明限于所公开的形式。对于本领域的技术人员来说，许多修改和变化是显而易见的。例如，示例性实施例将跟踪数据存储于错误日志中。此数据可以存储在可由处理所述异常的过程访问的任何数据结构中。实施例的选择和描述是为了最佳地解释本发明的原理、实际应用，并且当适合于所构想的特定使用时，使得本领域的其他技术人员能够理解本发明的具有各种修改的各种实施例。



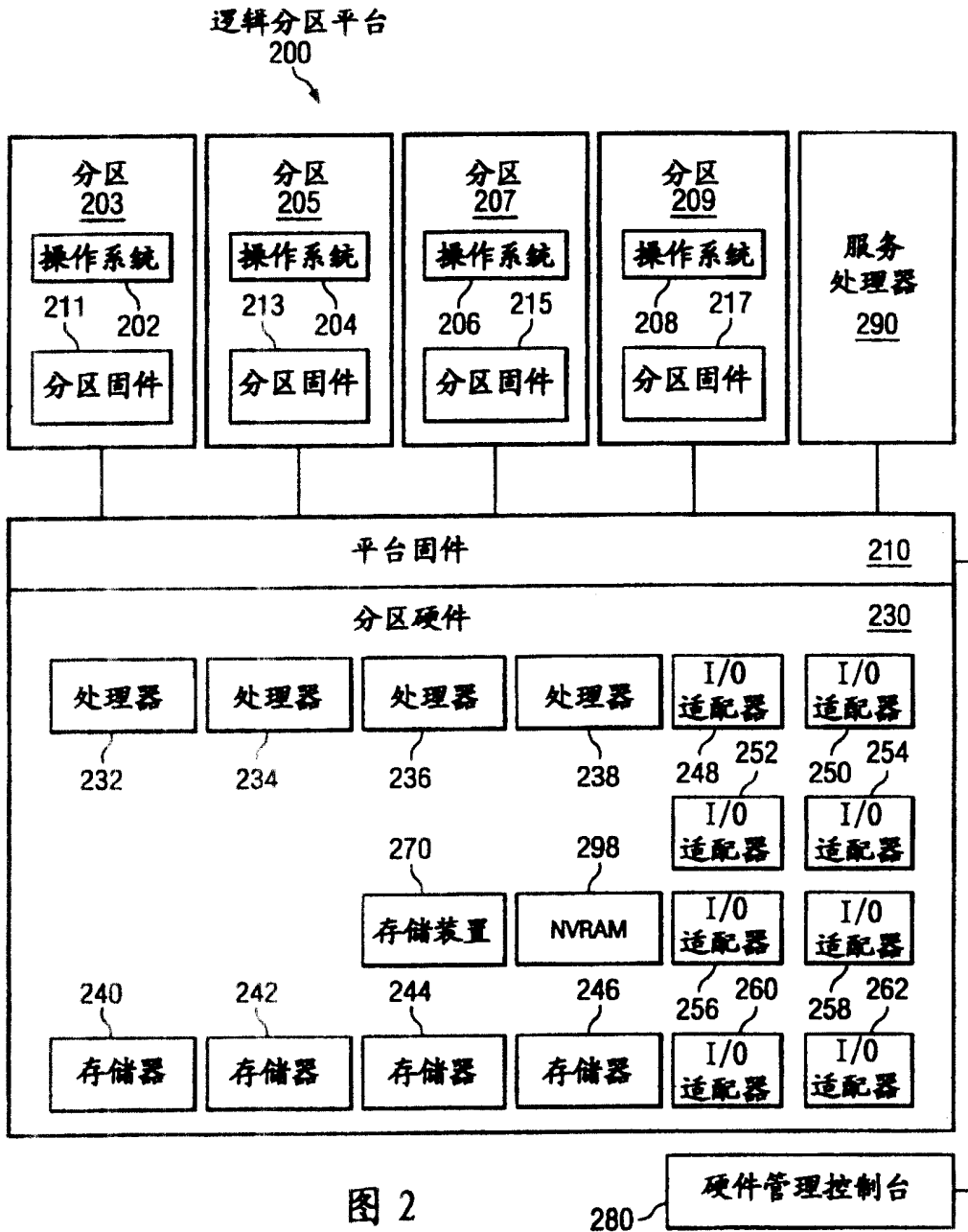


图 2

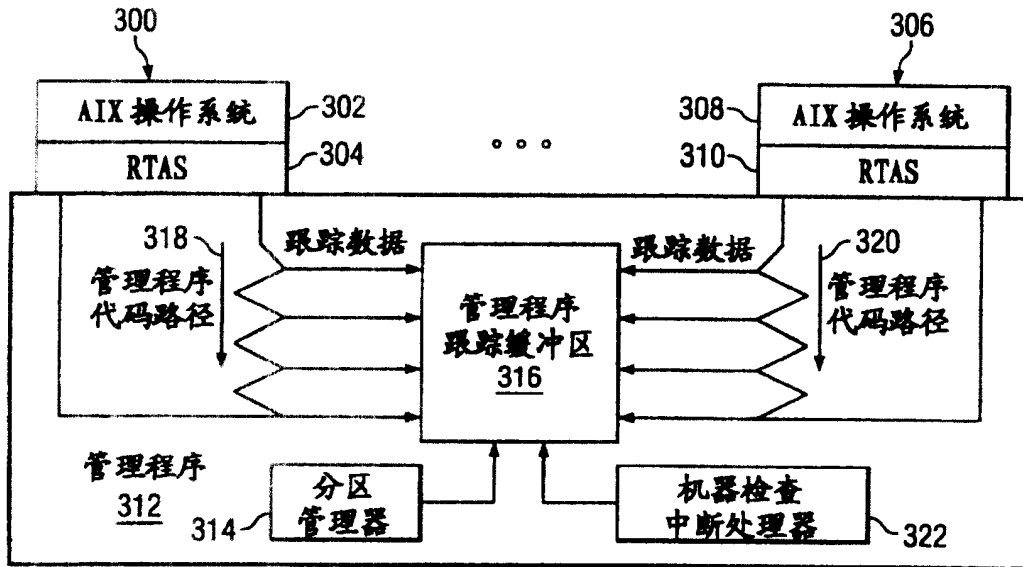


图 3

(现有技术)

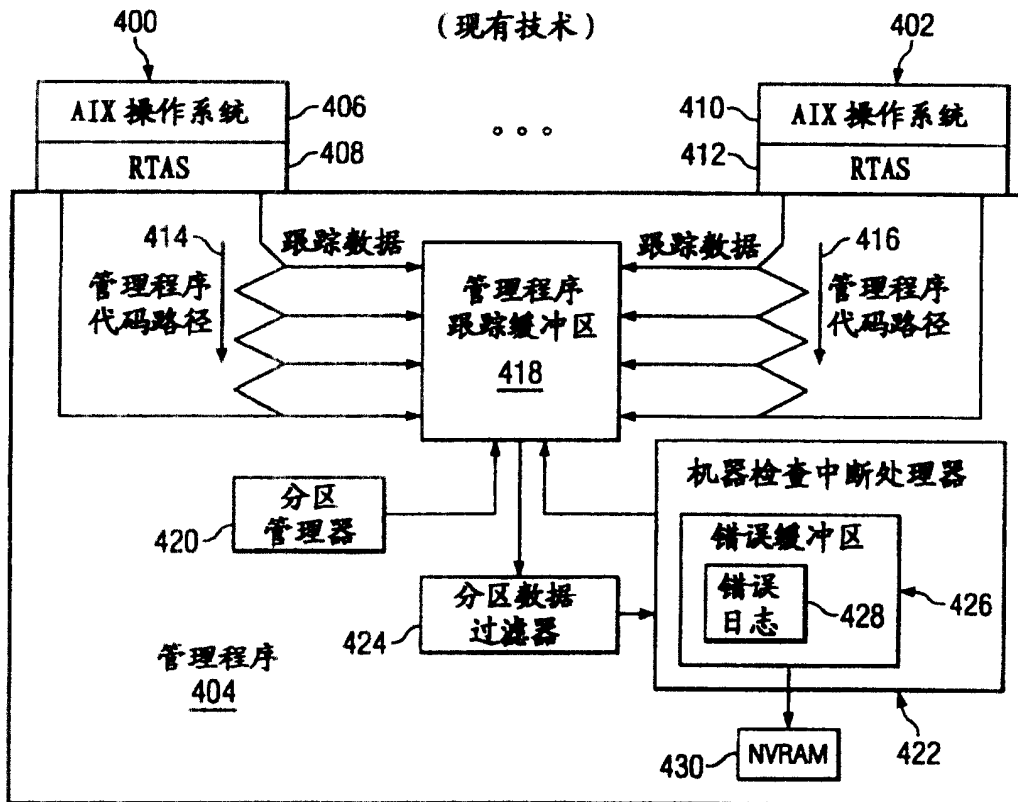


图 4

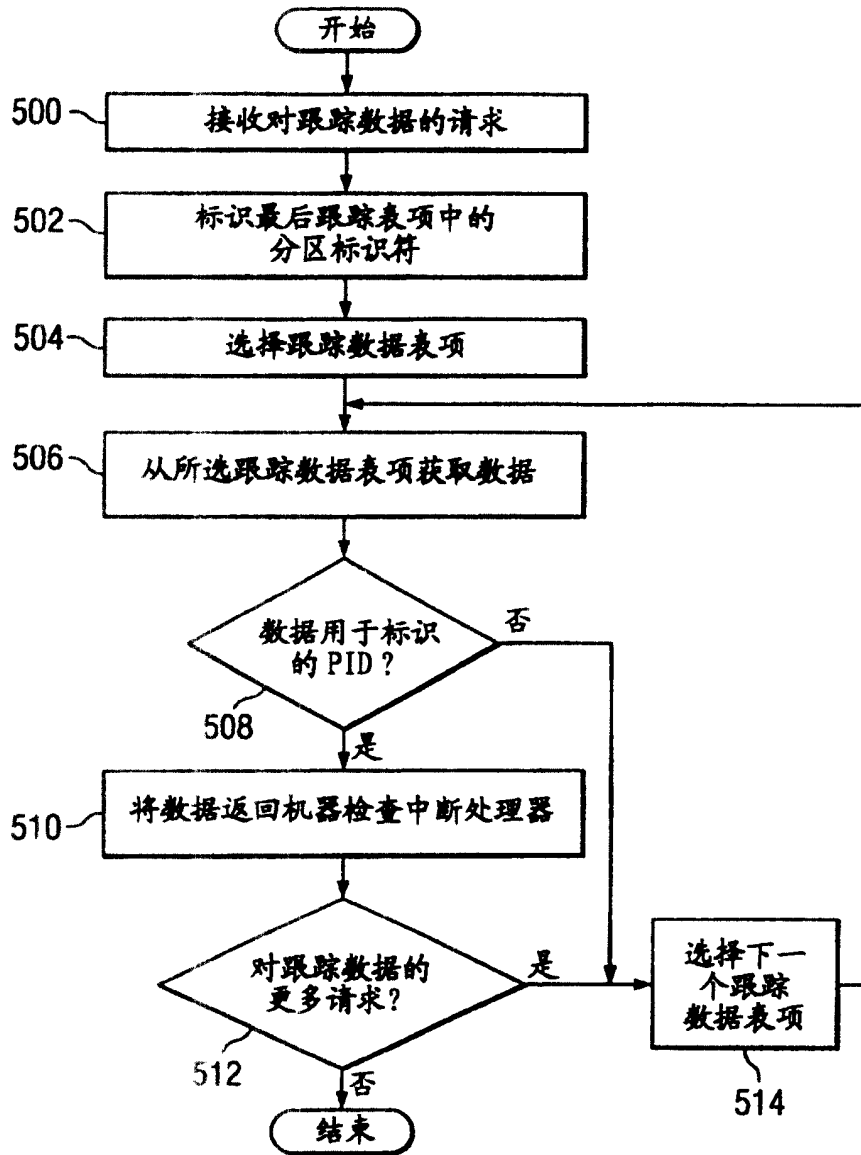


图 5

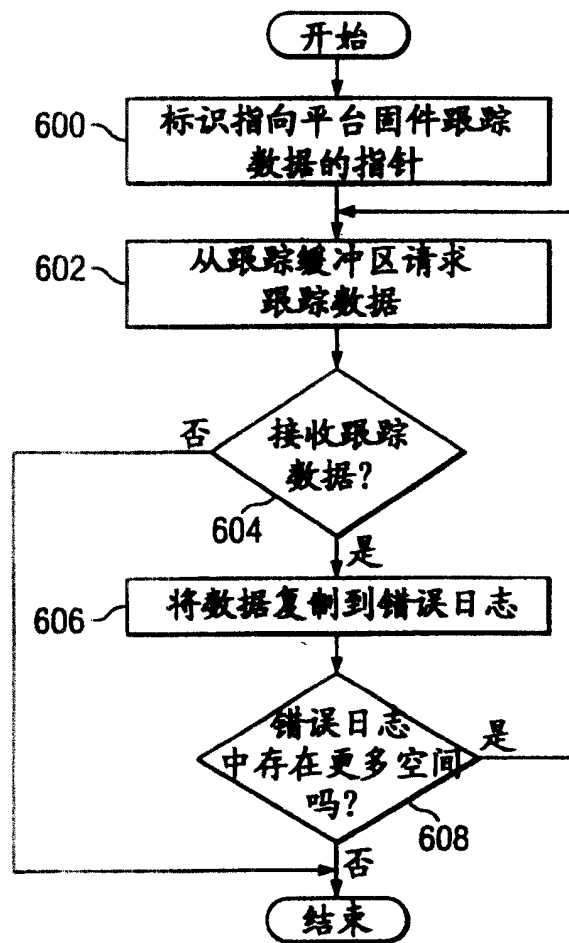


图 6