



(12)发明专利

(10)授权公告号 CN 102959519 B

(45)授权公告日 2016.10.19

(21)申请号 201180032537.0

(22)申请日 2011.03.15

(65)同一申请的已公布的文献号

申请公布号 CN 102959519 A

(43)申请公布日 2013.03.06

(30)优先权数据

10-2010-0061455 2010.06.28 KR

PCT/2010/006068 2010.09.07 KR

(85)PCT国际申请进入国家阶段日

2012.12.28

(86)PCT国际申请的申请数据

PCT/KR2011/001803 2011.03.15

(87)PCT国际申请的公布数据

W02012/002635 K0 2012.01.05

(73)专利权人 现代自动车株式会社

地址 韩国,首尔

专利权人 起亚自动车株式会社

梨花女子大学校产学协力团

(72)发明人 崔炳珠 徐珠瑛 梁承完 林真用

金泳秀 吴定锡 权海永 张丞延

(74)专利代理机构 北京尚诚知识产权代理有限公司

公司 11322

代理人 龙淳

(51)Int.Cl.

G06F 11/22(2006.01)

(56)对比文件

US 2009/089622 A1, 2009.04.02,

US 5383192 A, 1995.01.17,

CN 1149735 A, 1997.05.14,

W0 2007/039486 A2, 2007.04.12,

US 20080209443 A1, 2008.08.28,

审查员 郭从征

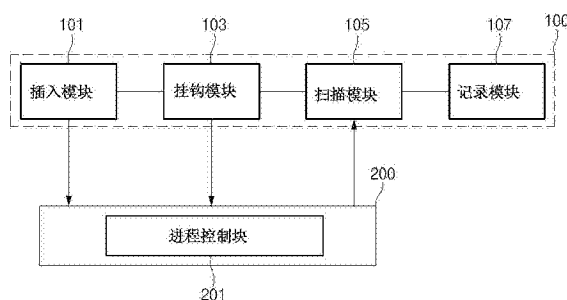
权利要求书2页 说明书7页 附图3页

(54)发明名称

系统测试设备

(57)摘要

本发明涉及一种系统测试设备,所述系统测试设备包括:插入模块,其配置为将测试代理插入进程控制块;挂钩模块,其配置为在发生涉及测试目标的事件时使用测试代理将测试目标挂钩到测试代码;扫描模块,其配置为在测试目标挂钩时,收集与发生涉及测试目标的事件的进程有关的测试信息;以及记录模块,其配置为存储由扫描模块收集的测试信息。



1. 一种系统测试设备,所述系统测试设备用于包括多个进程和含有关于所述多个进程的运行时间执行信息的进程控制块的系统,所述系统测试设备包括:

插入模块,其配置为将测试代理插入所述进程控制块;

挂钩模块,其配置为在发生涉及测试目标的事件时使用所述测试代理将所述测试目标挂钩到测试代码;

扫描模块,其配置为在所述测试目标挂钩到所述测试代码时,收集与发生涉及所述测试目标的事件的进程有关的测试信息;以及

记录模块,其配置为存储由所述扫描模块收集的测试信息。

2. 根据权利要求1所述的系统测试设备,其中所述进程控制块包括所述多个进程的资源使用现状、进程状态、优先级顺序和函数表中的至少一个。

3. 根据权利要求2所述的系统测试设备,其中所述测试代码根据缺陷类型被定义。

4. 根据权利要求3所述的系统测试设备,其中所述挂钩模块基于根据检测类型的测试位置来鉴别存储器区域,并将已鉴别的存储器区域挂钩到所述测试代码。

5. 根据权利要求3所述的系统测试设备,其中所述测试信息包括:程序执行信息、进程背景信息和系统状态信息中的至少一个。

6. 根据权利要求2所述的系统测试设备,其中所述扫描模块测量有关所述进程控制块的性能的信息。

7. 根据权利要求6所述的系统测试设备,其中有关所述进程控制块的性能的信息包括:处理器使用率、存储器使用率、页面错误、进程背景信息、进程调用栈信息、进程堆使用率、进程状态,以及进程优先级中的至少一个。

8. 根据权利要求6所述的系统测试设备,其中所述扫描模块以预定的时间间隔测量关于所述进程控制块的性能的信息。

9. 根据权利要求6所述的系统测试设备,其中所述扫描模块测量有关所述进程控制块的进程和线程的信息。

10. 根据权利要求9所述的系统测试设备,其中所述进程控制块的进程的性能因数包括ID、进程状态、进程的优先级、堆使用率、进程的运行时间、用户时间,以及内核时间中的一个或更多。

11. 根据权利要求9所述的系统测试设备,其中所述进程控制块的线程的性能因数包括ID、运行状态、基本优先级、当前优先级、用户时间、内核时间,以及调用栈中的一个或更多。

12. 根据权利要求2所述的系统测试设备,其中涉及所述测试目标的事件包括所述多个进程之间的消息传递、共享数据的转移、共享存储器的使用、消息队列的使用,以及同步对象的使用中的至少一者。

13. 根据权利要求12所述的系统测试设备,其中所述测试代码是用于检查有关所述多个进程之间的消息传递、消息队列、共享存储器以及同步对象中的至少一者的缺陷类型代码。

14. 根据权利要求12所述的系统测试设备,其中所述测试信息包括有关所述多个进程的调用栈信息、在所述多个进程之间传输和接收的通信类型、通信ID、通信数据以及在所述多个进程之间的通信中发生的延迟时间中的至少一个。

15. 根据权利要求12所述的系统测试设备,其中所述测试信息包括有关系统的动态状

态的信息、有关进程状态的信息以及有关所述多个进程之间的通信的信息。

16. 一种系统测试系统, 所述系统测试系统用于包括多个进程和含有关于所述多个进程的运行时间执行信息的进程控制块的系统, 所述系统测试系统包括:

输入设备, 其配置为接收用于驱动所述系统测试系统的信号;

系统测试设备, 其配置为当所述系统测试系统响应于通过所述输入设备接收的所述信号而运转时, 将测试代理插入所述进程控制块, 使用所述测试代理将测试目标挂钩到测试代码, 并从所述多个进程中收集与发生涉及测试目标的事件的进程有关的测试信息, 以及

分析设备, 其配置为分析由所述系统测试设备收集的测试信息。

17. 根据权利要求16所述的系统测试系统, 其中所述进程控制块包括所述多个进程的资源使用现状、进程状态、优先级顺序和函数表中的至少一个。

18. 根据权利要求16所述的系统测试系统, 其中所述测试代码挂钩到的测试位置可根据缺陷类型而改变。

19. 根据权利要求16所述的系统测试系统, 其中所述测试信息包括程序执行信息、进程背景信息、系统状态信息、处理器使用率、存储器使用率、页面错误、关于所述多个进程的调用栈信息、在所述多个进程之间传输和接收的通信类型、通信ID及通信数据, 以及在所述多个进程之间的通信中发生的延迟时间中的至少一个。

20. 根据权利要求16所述的系统测试系统, 还包括:

输出设备, 其被配置为以图像或语音的形式输出由所述分析设备分析的测试的结果。

系统测试设备

技术领域

[0001] 本发明涉及系统测试设备,并更特别涉及用于使用包括关于系统的执行信息的进程控制块,实时收集关于系统测试的信息的系统测试设备。

背景技术

[0002] 在包括各种软件和硬件部分的嵌入式系统中,在系统运行的环境中互操作性很重要。因此,不是包括软件的用于测试的虚拟数据而是使用在系统实际运行的实践环境中获得的实际数据的操作也很重要。如果想要测试系统,则需要这样一种方案,其通过使用实际运行数据执行系统测试,从而真正有效地检测潜在缺陷或系统实际运行时发生的运行缺陷。

[0003] 此外,为了独立地测试形成系统的软件部分,需要许多测试资源和成本,这是因为必须使用用于替代对应部件的另外设备。因此,需要这样一种测试设备,其可以实时测试包括在系统中的全部进程,并使存储器空间开销或执行对应代码所致的性能开销最小化。

发明内容

[0004] [技术问题]

[0005] 本发明的实施例旨在提供一种系统测试设备,该系统测试设备通过使用挂钩(hooking)技术对系统执行存储器错误测试、性能测试和用于进程间通信的测试,能够测试实际运行的系统,即使在系统的实时状况中也能执行动态测试,并且,能够检查在系统运行时发生的缺陷。

[0006] [技术解决方案]

[0007] 一方面,本发明涉及一种系统测试设备。该系统测试设备包括:插入模块,其配置为使测试代理(test agent)插入至进程控制块;挂钩(hooking)模块;其配置为当发生涉及测试目标的事件(event)时,使用测试代理使测试目标挂钩到测试代码;扫描模块,其配置为当测试目标挂钩到测试代码时,收集与已发生其中涉及测试目标的事件的进程有关的测试信息;以及记录模块(logging module),其配置为存储由扫描模块收集的测试信息。

[0008] 另一方面,本发明涉及一种系统测试系统。该系统测试系统包括输入设备,其配置为接收用于驱动系统的信号,系统测试设备,其配置为在系统响应于通过输入设备接收的信号运行时,将测试代理插入进程控制块,使用测试代理将测试目标挂钩到测试代码,并从多个进程中收集与已发生涉及测试目标的事件的进程有关的测试信息,以及分析设备,其配置为分析由系统测试设备收集的测试信息。

[0009] [发明效果]

[0010] 根据本发明,因为将不是源代码的实际运行代码作为目标,所以可以在系统实际运行时执行用于系统的测试。因此,可以在系统实际运行时测试存储器错误、系统性能和进程间通信。

[0011] 此外,因为在没有将个别代码插入每个进程的情况下使用利用包括在进程控制块

中的信息的挂钩方法,所以能够使系统开销最小化,并且能够在不中断系统的情况下执行用于系统的测试。

[0012] 此外,尽管进程之间的共享资源受限制,但能够实时收集关于测试的信息,并且能够基于该信息检查测试的结果。

附图说明

[0013] 图1是示出使用根据本发明的实施例的系统测试设备的整个系统的视图;

[0014] 图2示出根据本发明的实施例的系统测试设备的构造;

[0015] 图3是示出根据本发明的实施例的系统测试方法的流程图;并且

[0016] 图4a至图4c是示出根据本发明的实施例的存储器测试方法、性能测试方法和测试进程间通信的方法的流程图。

具体实施方式

[0017] 本发明涉及在系统实际运行的环境中的测试并可以在从系统的起动到系统的结束的动态状况下测试系统的设备和方法。本发明可以使用挂钩方案,对在系统内运行的全部进程执行测试。

[0018] 图1是示出使用根据本发明的实施例的系统测试设备的整个系统的视图。

[0019] 如图1所示,整个系统包括系统测试设备100、系统200、输入设备300、分析设备400和输出设备500。

[0020] 系统测试设备100配置为测试系统200的存储器缺陷、系统200的性能,或包括在系统200中的两个或更多进程之间的通信。系统测试设备100,如图1所示,可以与系统200独立地进行设置,或可以以通信测试设备100包括在系统200中的形式实施。

[0021] 系统200是系统测试设备100的测试目标。系统200是其中集成各种软件和硬件部分的嵌入式系统。该系统,即系统测试设备100的测试目标,指代包括多个进程的一个或更多系统。因此,多个系统中的存储器缺陷或系统性能可以成为系统测试设备100的测试目标,并且不仅是包括在一个系统中的多个进程之间的通信,而且包括在不同系统中的进程之间的通信也可以成为系统测试设备100的测试目标。

[0022] 系统200可以包括多个进程相互执行通信的全部的嵌入式系统和计算机系统,其可以是例如信息娱乐(infotainment)系统、多媒体系统或有线/无线网络系统。例如,如果系统200是车辆信息娱乐系统,则系统200以音频视频导航(AVN:Audio Video Navigation)功能为基础,可以包括地面数字多媒体广播(TDMB:Terrestrial Digital Multimedia Broadcasting)、传输协议专家组(TPEG:Transport Protocol Expert Group)、IPOD、USB、蓝牙(Bluetooth)、语音识别(VR:Voice Recognition)、停车诱导系统(PGS:Parking Guidance System)、无线电数据系统(RDS:Radio Data System)、车辆CDC(VCDC:Vehicle CDC)、驾驶员信息系统(DIS:Driver Information System)、后座娱乐系统(RSE:Rear Seat Entertainment)、混合动力汽车(HEV:Hybrid Electric Vehicle)显示器(display),以及空调单元(AHU:Air Handling Unit)显示器(display)。

[0023] 输入设备300是用于驱动系统测试设备100和系统200的用户输入设备。输入设备300可以具体化为键盘或触摸屏。响应于通过输入设备300接收的信号,驱动系统200。此外,

系统测试设备100响应于通过输入设备300接收的信号,执行用于系统测试的操作。

[0024] 分析设备400分析由系统测试设备100收集的测试信息,并检查系统200的测试结果。系统测试设备100可以与分析设备400一起集成为一个设备。在此情况下,系统测试设备100可以通过收集测试信息并同时分析已收集测试信息,来检查测试的结果。

[0025] 输出设备500以图像或语音的形式输出由系统测试设备100收集的测试信息或由分析设备400分析的测试结果。使用输出设备500的用户可以检查系统200的测试结果。

[0026] 图2示出根据本发明的实施例的系统测试设备100的构造。

[0027] 如图2所示,系统测试设备100包括插入模块101、挂钩模块(hooking module)103、扫描模块105和记录模块107。在图2中,假设连接到系统测试设备100的系统200,即测试目标,是包括两个或更多进程的单个系统。

[0028] 系统200包括进程控制块201。进程控制块201包括关于包括在系统200中的全部进程的运行时间(run-time)执行信息。运行时间执行信息是指当系统200运转时全部进程之间的执行信息。因此,进程控制块201可以包括对包括在系统200中的全部进程的资源使用的现状、进程状态、优先级顺序,以及函数表。更具体地,进程控制块201可以包括进程ID、进程名称、当前进程、当前线程、正在在系统200中执行的进程的列表、模块的列表、系统页大小(system page size)、可用物理存储器大小、CPU使用率、正在使用的资源的列表、正在请求的I/O设备状态、关于打开文件(opened file)的信息,以及关于寄存器的信息。因此,在全部系统中包括上述的信息的数据结构或设备对应于本发明的进程控制块。

[0029] 插入模块101将测试代理插入进程控制块201。测试代理可以基于包括在进程控制块201中的信息将测试目标挂钩到测试代码,并收集测试信息。测试代码挂钩到的测试位置可以根据缺陷类型来定义。在本发明中,挂钩到测试代码的测试目标可以是存储器区域即测试目标、包括关于系统性能的信息的代码或生成通信相关事件的代码。在本发明中,系统200在测试代理被插入至进程控制块201之前正常运转(operate)。系统200在不受系统测试设备100影响的情况下正常运转。

[0030] 此外,插入模块101可以检查正在系统200中运行(operate)的进程的列表,并基于检查的列表将测试代理插入进程控制块201。

[0031] 挂钩模块103使用由插入模块101插入的测试代理将测试目标挂钩到测试代码。测试代码根据有待测试的目标和缺陷类型而改变。即,测试代码根据存储器缺陷测试、性能测试和通信测试而不同。即使在通信测试中,测试代码也可以根据有待测试的缺陷类型而不同。此外,挂钩模块103可以改变测试代码被挂钩到的测试位置。

[0032] 术语“挂钩到测试代码”是指在涉及测试目标的事件在正在运行(operate)的系统200内发生时,被插入的测试代理将OS的进程控制块挂钩,从而执行包括原始代码操作的测试代码。挂钩模块103可以基于包括在进程控制块中的信息检查涉及测试目标的事件是否发生。

[0033] 扫描模块105在测试代码被挂钩时,收集与已发生涉及测试目标的事件的进程有关的测试信息。这里,测试信息可以不仅包括关于已产生事件的信息,而且包括与根据测试代码的挂钩新近发生的事件有关的信息。已收集测试信息根据测试步骤、测试目标和有待测试的缺陷类型而不同。

[0034] 因此,扫描模块105收集关于分析系统测试结果所需的系统动态状态的信息,例如

函数参数和返回值、性能测量值,以及共享资源使用列表。

[0035] 记录模块107在存储空间中存储由扫描模块105收集的测试信息。存储空间可以包括在记录模块107中,或可以体现为不同于记录模块107的装置。

[0036] 图1的分析设备400可以分析由扫描模块105收集的测试信息,或由记录模块107存储的测试信息。

[0037] 图3是示出根据本发明的实施例的系统测试方法的流程图。

[0038] 如图3所示,在步骤S110,测试代理被插入至作为测试目标的系统200的进程控制块201。系统200在测试代理被插入至进程控制块201之前正常运转。

[0039] 接下来,在涉及测试目标的事件发生时,在步骤S120,被插入的测试代理将测试目标挂钩到测试代码。在测试目标挂钩到测试代码时,测试代码可以代替已发生事件的进程,被执行。

[0040] 在测试目标挂钩到测试代码时,在步骤S130,收集与发生涉及测试目标的事件的进程有关的测试信息。在步骤S140,存储已收集的测试信息。这里,测试信息可以仅在测试目标挂钩到测试代码时收集,或测试信息也可以在测试目标挂钩到测试代码时和在测试目标挂钩到测试代码之后,系统运转时收集。

[0041] 图4a到图4c是示出根据本发明的实施例的存储器测试方法、性能测试方法和测试进程间通信的方法的流程图。

[0042] 图4a示出存储器缺陷测试的系统测试方法。首先,在步骤S150定义缺陷模型。缺陷模型是指根据缺陷类型有待检测的存储器的位置和用于发现和修复缺陷的测试信息。在系统200正常运行时,可以根据缺陷模型生成能收集测试信息的测试代码。

[0043] 接下来,在步骤S151测试代理被插入至系统200的进程控制块201。测试代理可以被插入至在系统200中运行的进程的列表中的至少一个。

[0044] 接下来,在步骤S152,挂钩模块103基于缺陷模型和包括在进程控制块201中的信息鉴别有待挂钩的存储器区域,并将测试代码挂钩到已鉴别的存储器区域。如果鉴别存储器区域的结果,测试位置是系统(SYSTEM)应用程序进程接口(API:Application Process Interface)函数,则挂钩模块103将进程控制块201的API集(API set)鉴别为挂钩区域。如果鉴别存储器区域的结果,测试位置是动态链接库(DLL:Dynamic Linking Library)输出(export)函数,则挂钩模块103将进程控制块201的输入地址表(IAT:Import Address Table)鉴别为挂钩区域。在其它情况下,挂钩模块103将进程控制块201的代码空间鉴别为挂钩区域。

[0045] 在图4a中,术语“挂钩到测试代码”是指当在系统200内,在存储器即测试目标的位置发生对存储器的事件例如写入、读取或访问时,已插入的测试代理将OS的进程控制块挂钩,以执行包括原始代码操作的测试代码。

[0046] 接下来,在步骤S153,收集关于作为测试目标的存储器的状态的信息。例如,可以收集鉴别缺陷的原因所需的程序执行信息、鉴别缺陷的位置所需的进程内容信息,以及分析存储器性能所需的系统状态信息。

[0047] 在步骤S153收集的测试信息在步骤S154存储。

[0048] 图4b示出用于执行性能测试的系统测试方法。在根据本发明的系统测试方法中,为了分析性能瓶颈及其原因,可以挂钩源自进程控制块201的数据中的系统执行信息例如

页面错误率和处理器使用率,并且可以基于挂钩的结果分析系统性能。

[0049] 这里,术语“性能”是指系统或部件在给定系统的状况下执行功能的程度。性能测试是关于具体性能需求是否满足系统的评估。可以通过性能测试来分析性能瓶颈及其原因,并且可以通过解决性能瓶颈及其原因来改善系统性能。

[0050] 系统的性能瓶颈是由于受到限制的资源例如存储器、I/O设备和网络的冲突,系统性能恶化的现象。系统性能瓶颈可以是各种各样的,例如资源的缺乏、共享资源的冲突、资源的独占、资源的错误配置,以及资源的错误操作。

[0051] 首先,在步骤S160定义缺陷模型。缺陷模型是指根据缺陷类型有待测试的性能瓶颈的位置和表示确定性能瓶颈的系统状态的测试信息。在系统200正常运转时,可以根据已定义的缺陷模型生成能收集测试信息的测试代码。

[0052] 接下来,在步骤S161插入模块101将测试代理插入进程控制块201。插入模块101可以检查在系统200中运行的进程的列表,并基于已检测的列表将测试代理插入进程控制块201。即,涉及当前进程的进程控制块201的基址(base address)在具体存储器空间中管理,或根据环境作为先前固定的地址来管理。因此,可以预知关于涉及当前进程的进程控制块201的信息。

[0053] 在步骤S162挂钩模块103将测试目标挂钩到测试代码。

[0054] 在测试目标挂钩到测试代码时,在步骤S163扫描模块105收集关于系统性能的信息。例如,可以使用计时器中断,并且可以以预定时间间隔(例如1秒或100毫秒)测量性能数据。此外,为了不仅对每个系统而且对每个进程或线程分析系统性能,可以一边使进程控制块201的进程和线程列表循环,一边测量性能信息。

[0055] 为了精确跟踪系统性能瓶颈所发生的位置,扫描模块105可以对每个线程存储调用栈(Call-stack)信息。对每个监控被测量的信息如下。

[0056] 进程控制块201的性能信息可以包括处理器使用率、存储器使用率、页面错误、进程内容信息、进程调用栈信息、进程堆(process heap)使用率、进程状态以及进程优先级中的一个或更多。进程控制块201的进程的性能信息可以包括ID、进程状态、进程优先级、堆(heap)使用率、进程的运行时间、用户时间以及内核时间中的一个或更多。此外,进程控制块201的线程的性能信息可以包括ID、运行状态、基本优先级、当前优先级、用户时间、内核时间,以及调用栈信息中的一个或更多。

[0057] 接下来,在步骤S164,记录模块107在存储空间中存储已收集的测试信息。

[0058] 在用于执行性能测试的系统测试方法中,可以基于页面错误(pagefault)确定存储器瓶颈。例如,如果页面错误为高,则其可能是存储器瓶颈。页面错误是程序尝试访问在自己的地址空间中存在但现在不在系统存储器中的数据或代码时所发生的现象。在发生页面错误时,OS将对应数据提取到存储器,使程序继续运行,如同页面错误没有发生。由于OS对页面错误的异常处理,应用程序执行处理的时间被延迟,并且整体系统性能受影响。

[0059] 此外,在根据本发明的系统测试方法中,可以基于存储器使用率确定性能瓶颈。系统的存储器可以被分为物理存储器使用率、虚拟存储器使用率和对每个进程的堆存储器使用率。在根据本发明的系统测试方法中,基于存储器使用率的总和确定性能瓶颈。

[0060] 此外,在根据本发明的系统测试方法中,可以基于处理器使用率(或CPU使用率)确定性能瓶颈。例如,在根据本发明的系统测试方法中,如果处理器使用率保持为高并且具有

可用存储器,那么可以确定在CPU上具有瓶颈。相反,如果处理器使用率保持为高并且存储器已耗尽,则可以确定性能问题是存储器瓶颈而不是CPU。

[0061] 此外,在根据本发明的系统测试方法中,可以基于进程使用率确定性能瓶颈。进程使用率是指系统的执行时间,即已从总CPU使用率排除空闲时间(idle time)的时间。

[0062] 此外,在根据本发明的系统测试方法中,可以基于用户时间确定性能瓶颈。用户时间是指执行保持在用户空间中的时间,即执行应用程序花费的时间。

[0063] 此外,在根据本发明的系统测试方法中,可以基于内核时间确定性能瓶颈。内核时间意思是执行保持在内核空间中的时间,即内核处理服务花费的时间。

[0064] 图4c示出用于执行对复数进程之间的通信的测试的系统测试方法。执行图4c的系统测试方法的系统测试设备100可以收集与在系统200中的全部进程之间的共享资源有关,以及与通信是否精确执行有关的测试信息。

[0065] 首先,在步骤S170定义缺陷模型。这里,缺陷模型是指根据缺陷类型有待测试的进程间的通信位置和用于发现和修复缺陷的测试信息。在系统200正常运行时,可以根据已定义的缺陷模型生成能收集测试信息的测试代码。

[0066] 在步骤S171,测试代理被插入至系统200即测试目标的进程控制块201。插入模块101可以选择进程的列表中的两个或更多作为测试目标,并且将用于测试已选择的进程之间的通信的测试代理插入进程控制块201。

[0067] 当涉及通信的事件在多个进程之间发生时,在步骤S172已插入的测试代理将测试目标挂钩到测试代码。

[0068] 术语“挂钩到测试代码”是指当在正在运行的系统200内的多个进程之间发生涉及通信的事件时,已插入的测试代理将OS的进程控制块挂钩,以执行包括原始代码操作的测试代码。挂钩模块103可以基于包括在进程控制块201中的信息检查涉及通信的事件是否发生。这里,涉及通信的事件是指多个进程之间的消息传递、共享数据的转移、共享存储器的使用、消息队列(message queue)的使用以及同步对象的使用。

[0069] 测试代码用来检查缺陷类型,该缺陷类型涉及多个进程之间的消息传递、消息队列、共享存储器或同步对象。

[0070] 例如,测试代码可以包括第一函数例如SendMessage()或PostMessage(),和根据有待测试的缺陷的类型所需的第二函数。包括在系统中的进程分别包括第一函数。因此,当涉及通信的事件在第一进程和第二进程之间发生时,挂钩模块103以如下方式将测试目标挂钩到测试代码,即在第一进程和第二进程中执行第一函数的步骤中,已插入的测试代理执行测试代码,而不执行第一进程和第二进程的第一函数。基于包括在进程控制块201中的信息,检查第一函数在第一进程和第二进程中执行的步骤。

[0071] 其间,由于测试目标是多个进程之间的通信,于是测试代码可以根据有待测试的缺陷类型仅在发射器和接收器的一个中挂钩。

[0072] 在测试目标已挂钩时,在步骤S173,扫描模块105收集与已发生涉及通信的事件的多个进程之间的通信有关的测试信息。这里,多个进程之间的通信不仅包括涉及通信的已产生的事件,而且包括涉及根据测试代码的挂钩发生的通信的事件。已收集的测试信息可根据测试步骤、测试目的以及有待测试的缺陷类型而不同。

[0073] 因此,扫描模块105收集确定与多个进程之间的通信有关的缺陷所需的系统动态

状态的信息,例如函数参数和返回值、性能测量值,以及共享资源使用列表。

[0074] 测试信息可以包括关于多个进程的调用栈信息、在多个进程之间传输和接收的通信类型、通信ID及通信数据,以及在多个进程之间的通信中发生的延迟时间。

[0075] 在步骤S174,记录模块107在存储空间中存储由扫描模块105收集的测试信息。存储空间可以包括在记录模块107中,或可以体现为不同于记录模块107的装置。图1的分析设备400可以分析由扫描模块105收集的测试信息,或存储在记录模块107中的测试信息。

[0076] 用于对多个进程之间的通信执行测试的测试代码可以是用于检查涉及进程之间的消息传递、消息队列、共享存储器或同步对象的代码。

[0077] 涉及消息传递(message passing)的缺陷类型可以被分为在预定时间内的消息传递失败(failure,失败)(同步)、消息死锁(同步)、消息处理延迟(同步)、消息丢失(同步)、消息改变(同步)、插入消息队列失败(failure)(异步)、消息处理延迟(异步)、消息丢失(异步),以及消息改变(异步)。

[0078] 涉及消息队列的缺陷类型可以被分为消息队列生成失败、消息队列关闭失败、源自消息队列的读取失败,以及在消息队列中的写入失败。

[0079] 涉及共享存储器的缺陷类型可以是共享存储器生成失败。

[0080] 涉及同步对象的缺陷类型可以被分为使用没有生成的同步对象、未使用已生成的同步对象,以及未释放(non-release)已使用的同步对象。

[0081] 因此,根据检测类型,测试代码随之改变。

[0082] 如上所述,根据本发明的系统测试设备100在不影响正在运行的系统200的情况下可以插入测试代理、将测试代码挂钩,并且收集测试信息。系统测试设备100基于包括在系统200的进程控制块201中的信息执行系统测试。

[0083] 本发明可以在记录介质中以处理器可读的代码的形式实施。处理器可读的记录介质包括存储处理器可读数据的全部类型的记录装置。记录介质可以包括例如ROM、RAM、CD-ROM、磁带、软盘,以及光数据存储。此外,处理器可读的记录介质可以分布在连接网络的计算机系统中,并且可以以分布形式存储并执行处理器可读的代码。

[0084] 虽然联系实际的示范性实施例说明了本发明,但应理解的是,本发明不限于已公开的实施例,相反,旨在覆盖包括在所附权利要求的主旨和范围内的各种修改和等效安排。

[0085] [工业应用性]

[0086] 本发明可以在系统实际运行时测试存储器错误、系统性能和进程间通信。

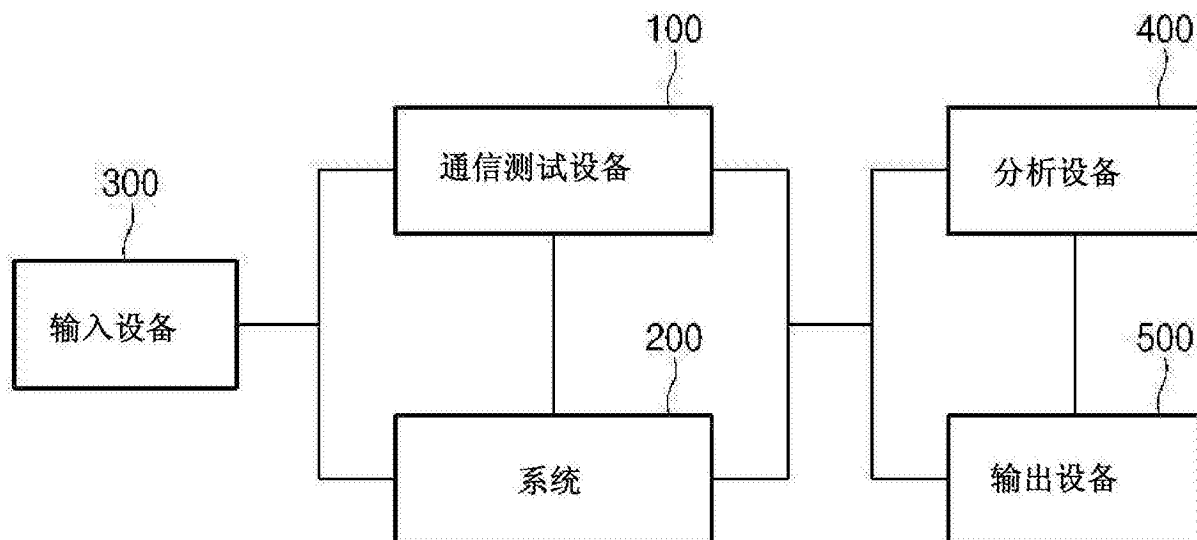


图1

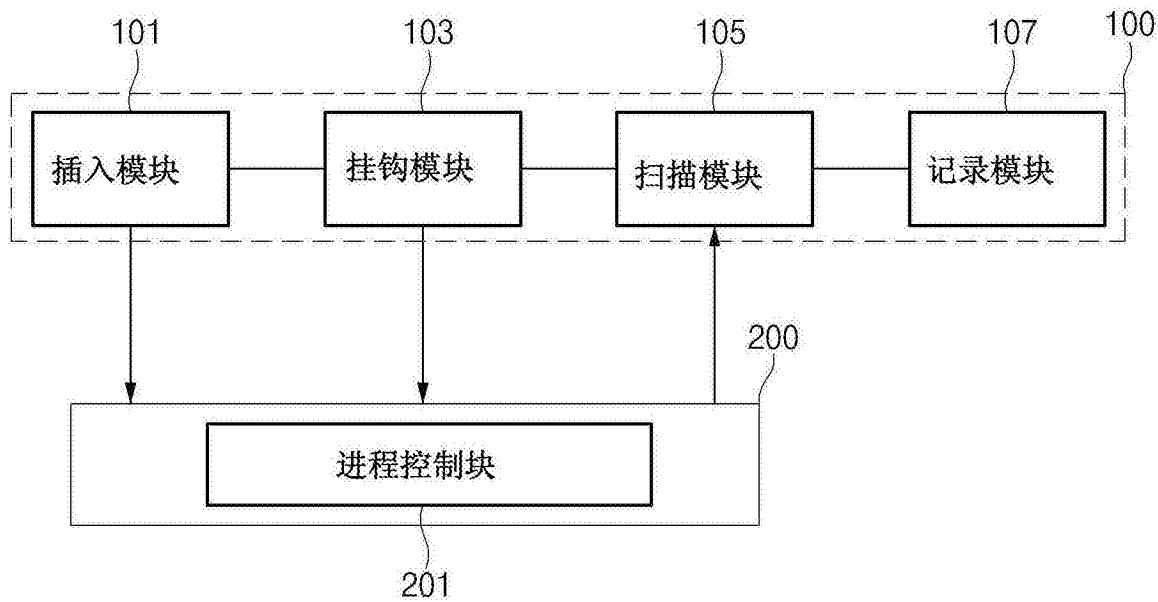


图2

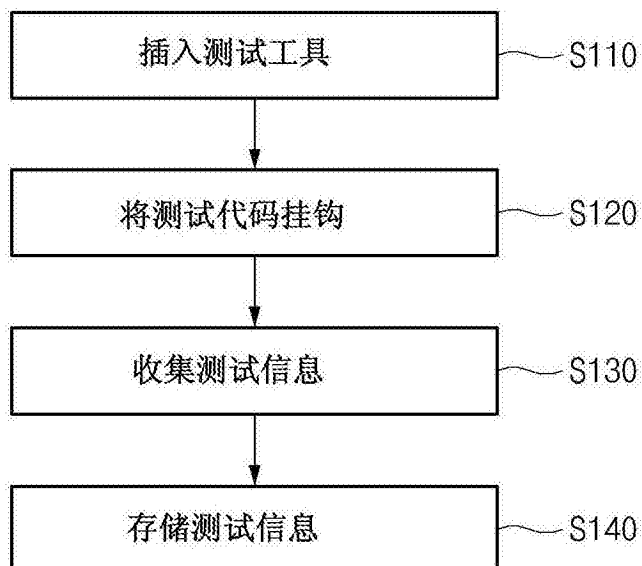


图3

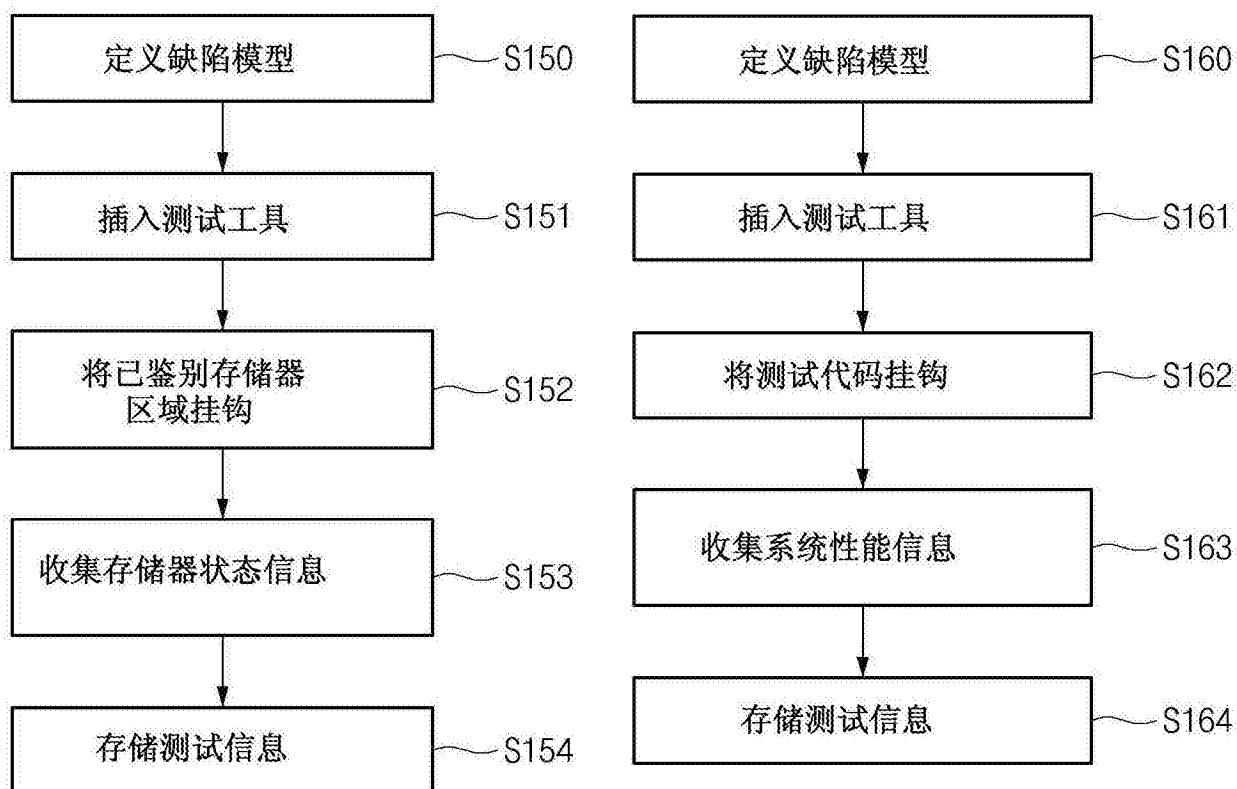


图4a

图4b

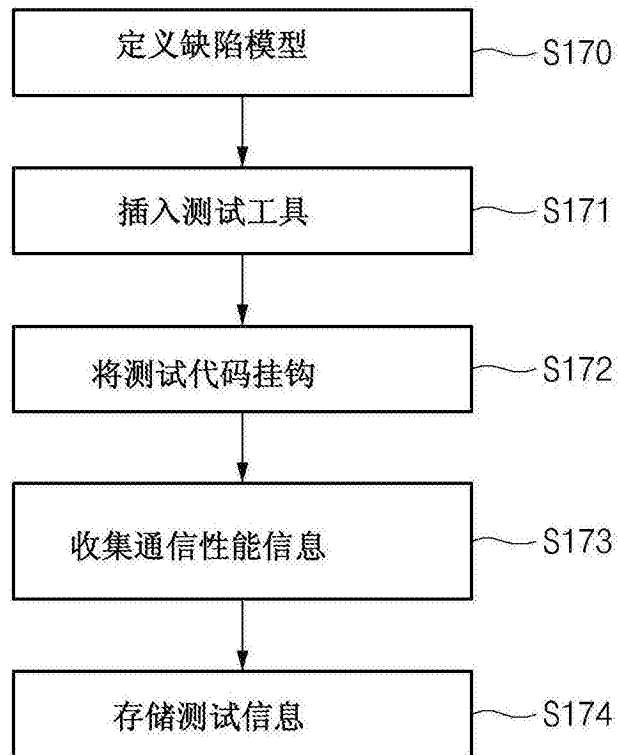


图4c