



(12) 发明专利

(10) 授权公告号 CN 110688313 B

(45) 授权公告日 2022. 11. 18

(21) 申请号 201910915607.5

(22) 申请日 2019.09.26

(65) 同一申请的已公布的文献号
申请公布号 CN 110688313 A

(43) 申请公布日 2020.01.14

(73) 专利权人 天津津航计算技术研究所
地址 300308 天津市东丽区空港经济区保
税路357号

(72) 发明人 赵昶宇 辛文增

(74) 专利代理机构 中国兵器工业集团公司专利
中心 11011
专利代理师 王雪芬

(51) Int. Cl.
G06F 11/36 (2006.01)

(56) 对比文件

CN 101937232 A, 2011.01.05

CN 106292336 A, 2017.01.04

CN 106295809 A, 2017.01.04

CN 104468267 A, 2015.03.25

CN 109117371 A, 2019.01.01

审查员 王斌

权利要求书1页 说明书4页 附图2页

(54) 发明名称

一种VxWorks操作系统下软件测试的故障注入方法

(57) 摘要

本发明涉及一种VxWorks操作系统下软件测试的故障注入方法,涉及嵌入式软件测试和VxWorks操作系统技术领域。本发明通过在软件源代码中增加测试代码的方式,能够在特定的位置和特定的时刻对软件源代码中的全局变量、局部变量、寄存器、堆栈以及软件接口等的故障注入和测试,提高软件的测试覆盖率和可靠性。该方法不需要修改原有的软件源代码,也不需要额外的监控器设备,只是在软件运行时,根据故障注入的时间和位置增加部分测试软件源代码,通过增加的这部分代码,可以实现软件运行过程中动态地改变代码中关键的全局变量、局部变量、寄存器、硬件地址和软件接口等信息内容,完成软件故障注入测试。



1. 一种VxWorks操作系统下软件测试的故障注入方法,其特征在于,包括以下步骤:

S1、建立软件故障测试用例库;

S2、通过telnet工具或Shell工具向目标系统注入故障;

步骤S1具体为:对嵌入式软件进行功能故障树分析,通过系统级和详细级失效模式及影响分析,最终从软件源代码中识别出特定关键的全局变量、局部变量、寄存器、硬件地址和软件接口这些信息;然后基于识别出的这些信息进行软件测试用例设计,形成软件故障测试用例库;

步骤S2中,在telnet工具、Shell工具或串口工具中键入一些特定的指令完成向目标系统的故障注入,这些指令是根据故障测试用例而自定义的不同函数,在每个函数内完成对全局变量的赋值,用这种方法实现故障的动态注入;

步骤S2具体为:在目标系统软件源代码中增加消息队列B和任务T1,该任务用于接收消息队列B中的故障测试用例,并根据每个故障测试用例的功能和性能要求,增加对应的函数,每个函数的功能是向消息队列B发送对应的故障测试用例,故障测试用例是按照约定的网络通讯协议制定的指令报文;

步骤S2具体包括:在目标系统源代码初始化时,启动任务T1,在telnet工具或Shell工具或串口工具中键入一些特定的指令,每一个指令均由一个具体的函数实现,该函数完成向消息队列B发送指令报文,用于设置XXX_Debug的值为TRUE,并通知目标系统给相关的全局变量赋值;目标系统在任务T1中收到消息队列B中的数据后,通过解析该报文内容,将报文中涉及到的每个XXX_Debug的值设置为TRUE,并直接修改目标系统软件源代码中相关全局变量的值,然后目标系统按照该故障测试用例设置的故障时序或故障测试路径完成软件测试工作,执行完一条故障测试用例后,在telnet工具或Shell工具或串口工具中键入特定的函数名,该函数用于通知目标系统将每个XXX_Debug的值设置为FALSE,将全局变量的值设置为初始值。

2. 如权利要求1所述的方法,其特征在于,所述指令报文用于实现单点故障测试,或故障序列测试。

3. 如权利要求1或2所述的方法,其特征在于,在步骤S2之后还包括步骤S3、目标系统记录测试结果并进行数据分析。

4. 如权利要求3所述的方法,其特征在于,步骤S3中,目标系统收集故障注入后带故障运行时的系统状态信息,以系统日志的方式进行记录。

一种VxWorks操作系统下软件测试的故障注入方法

技术领域

[0001] 本发明涉及嵌入式软件测试和VxWorks操作系统技术领域,具体涉及一种VxWorks操作系统下软件测试的故障注入方法。

背景技术

[0002] 将故障注入技术作为软件测试技术,一般来讲是制定特定的故障模型,采用人为的、有意识的方式产生故障,并施加于待测系统中,用来加速该系统的错误和失效的产生,同时观测并反馈对所注入故障的响应信息,通过分析对系统进行验证和评价的过程。基于软件的故障注入通常是根据故障模型,通过修改目标系统的内存映像来模拟系统硬件或软件故障的发生,按故障注入的时刻可分为运行前注入和运行时注入。

[0003] 目前基于嵌入式系统软件故障注入的测试方法有以下几种:

[0004] (1) 直接修改源程序的静态故障注入方法。该方法直接修改被测系统源代码模拟故障产生,不需要额外接口资源和监控软件,属于运行前注入故障的方式。该方法容易模拟永久故障,模拟时序相关的故障逻辑时,需要编写复杂的故障模拟代码,费时费力,且易引入新的故障,不满足灵活性和重用性要求;

[0005] (2) 软件开发环境支持的故障注入方法。该方法适用于软件开发人员在编码、调试阶段通过设置断点来修改内存或寄存器内容产生所需的故障,具有简单灵活的特点。但是由于设置断点会导致测试不能连续运行,对于涉及多任务和中断处理的软件,很难测试任务之间或中断处理与主流程之间的相互影响;

[0006] (3) 设计运行时的软件故障注入器,利用故障脚本描述语言、故障库和故障注入器算法获取故障现场的信息。该方法能够有效的将故障注入被测软件中,便于捕捉并重现故障信息。但是上述方法的实现比较复杂,通常需要设计主控制器、故障注入器、脚本解析器、监视器、数据采集器和数据收集器等,技术门槛较高,还需要针对不同类型的故障注入模型设计相应的故障注入器算法,灵活性较差。

发明内容

[0007] (一) 要解决的技术问题

[0008] 本发明要解决的技术问题是:针对目前基于嵌入式系统软件故障注入的测试方法中存在的不足,提出一种在VxWorks操作系统下软件测试的故障注入方法,以在软件测试过程中有效获得故障现场信息。

[0009] (二) 技术方案

[0010] 为了解决上述技术问题,本发明提供了一种VxWorks操作系统下软件测试的故障注入方法,包括以下步骤:

[0011] S1、建立软件故障测试用例库;

[0012] S2、通过telnet工具或Shell工具向目标系统注入故障。

[0013] 优选地,步骤S1具体为:对嵌入式软件进行功能故障树分析,通过系统级和详细级

失效模式及影响分析,最终从软件源代码中识别出特定关键的全局变量、局部变量、寄存器、硬件地址和软件接口这些信息;然后基于识别出的这些信息进行软件测试用例设计,形成软件故障测试用例库。

[0014] 优选地,步骤S2中,在telnet工具、Shell工具或串口工具中键入一些特定的指令完成向目标系统的故障注入,这些指令是根据故障测试用例而自定义的不同函数,在每个函数内完成对全局变量的赋值,用这种方法实现故障的动态注入。

[0015] 优选地,步骤S2具体为:在目标系统软件源代码中增加消息队列B和任务T1,该任务用于接收消息队列B中的故障测试用例,并根据每个故障测试用例的功能和性能要求,增加对应的函数,每个函数的功能是向消息队列B发送对应的故障测试用例,故障测试用例是按照约定的网络通讯协议制定的指令报文。

[0016] 优选地,所述指令报文用于实现单点故障测试,或故障序列测试。

[0017] 优选地,所述指令报文用于实现单点故障测试,或故障序列测试。

[0018] 优选地,步骤S2具体包括:在目标系统源代码初始化时,启动任务T1,在telnet工具或Shell工具或串口工具中键入一些特定的指令,每一个指令均由一个具体的函数实现,该函数完成向消息队列B发送指令报文,用于设置XXX_Debug的值为TRUE,并通知目标系统给相关的全局变量赋值;目标系统在任务T1中收到消息队列B中的数据后,通过解析该报文内容,将报文中涉及到的每个XXX_Debug的值设置为TRUE,并直接修改目标系统软件源代码中相关全局变量的值,然后目标系统按照该故障测试用例设置的故障时序或故障测试路径完成软件测试工作,执行完一条故障测试用例后,在telnet工具或Shell工具或串口工具中键入特定的函数名,该函数用于通知目标系统将每个XXX_Debug的值设置为FALSE,将全局变量的值设置为初始值。

[0019] 优选地,在步骤S2之后还包括步骤S3、目标系统记录测试结果并进行数据分析。

[0020] 优选地,步骤S3中,目标系统收集故障注入后带故障运行时的系统状态信息,以系统日志的方式进行记录。

[0021] (三)有益效果

[0022] 本发明通过在软件源代码中增加测试代码的方式,能够实现在特定的位置和特定的时刻对软件源代码中的全局变量、局部变量、寄存器、堆栈以及软件接口等的故障注入和测试,提高软件的测试覆盖率和可靠性。该方法不需要修改原有的软件源代码,也不需要额外的监控器设备,只是在软件运行时,根据故障注入的时间和位置增加部分测试软件源代码,通过增加的这部分代码,可以实现软件运行过程中动态地改变代码中关键的全局变量、局部变量、寄存器、硬件地址和软件接口等信息内容,完成软件故障注入测试。该方法实现简单,对被测试目标系统软件影响较小,极大地提高了嵌入式软件的容错性、可靠性和安全性。

附图说明

[0023] 图1为本发明的软件测试故障注入流程图;

[0024] 图2为本发明的软件测试故障注入的原理图。

具体实施方式

[0025] 为使本发明的目的、内容和优点更加清楚，下面结合附图和实施例，对本发明的具体实施方式作进一步详细描述。

[0026] 针对目前软件测试过程中难以有效获得故障现场信息的问题，本发明提出了一种VxWorks操作系统下软件测试的故障注入方法。该方法首先通过对嵌入式软件建立详细级失效模式及影响分析，建立软件故障测试用例库；然后利用telnet工具或Shell工具或串口工具向目标系统键入一些特定的指令，完成向目标系统的动态故障注入；最后对故障运行前后的测试结果进行分析和评估，能够实现在特定的位置和特定的时刻对软件源代码中的全局变量、局部变量、寄存器、堆栈以及软件接口等的故障注入和测试。如图1、图2所示，该方法具体包括以下步骤：

[0027] S1、建立软件故障测试用例库

[0028] 对嵌入式软件进行功能故障树分析，通过系统级和详细级失效模式及影响分析，最终从软件源代码中识别出关键的全局变量、局部变量、寄存器、硬件地址和软件接口等信息；然后基于识别出的这些信息进行软件测试用例设计，形成软件故障测试用例库；

[0029] S2、通过telnet工具或Shell工具向目标系统注入故障，本步骤中，在telnet工具、Shell工具或串口工具中键入一些特定的指令完成向目标系统的故障注入，这些指令是由测试人员根据故障测试用例而自定义的不同函数，在每个函数内完成对全局变量的赋值，用这种方法实现故障的动态注入。具体包括步骤：

[0030] S21、如果软件源代码从寄存器、硬件地址或是软件接口中读取的信息保存在局部变量A中，则在软件源代码中局部变量A获得当前寄存器、硬件地址或是软件接口中的信息后增加如下代码：

[0031] if (TRUE==XXX_Debug)

[0032] A=Global_XXX;

[0033] 上述Global_XXX为全局变量名称，XXX_Debug初始情况下的值为FALSE（即不改变当前获得的A的值），只有在接收到来自telnet工具、Shell工具或串口工具的指令后才将XXX_Debug的值设置为TRUE；

[0034] S22、在目标系统软件源代码中增加消息队列B和任务T1，并根据每个故障测试用例的功能和性能要求，增加对应的函数，每个函数的功能是向消息队列B发送对应的故障测试用例，故障测试用例是按照约定的网络通讯协议制定的报文，该报文可以实现单点故障（仅有一个变量、寄存器或硬件地址内容异常）测试，也可以完成故障序列测试；

[0035] 在目标系统源代码初始化时，启动任务T1，该任务用于接收消息队列B中的故障测试用例，软件测试人员在telnet工具或Shell工具或串口工具中键入一些特定的指令，每一个指令均由一个具体的函数实现，该函数完成向消息队列B发送指令报文，用于设置XXX_Debug的值为TRUE，并通知目标系统给相关的全局变量赋值；目标系统软件在任务T1中收到消息队列B中的数据后，通过解析该报文内容，将报文中涉及到的每个XXX_Debug的值设置为TRUE，并直接修改目标系统软件源代码中相关全局变量的值。上述操作完成后，目标系统按照该故障测试用例设置的故障时序或故障测试路径完成软件测试工作。执行完一条故障测试用例后，在telnet工具或Shell工具或串口工具中键入特定的函数名，该函数用于通知目标系统将每个XXX_Debug的值设置为FALSE，将全局变量的值设置为初始值；

[0036] S3、记录测试结果并进行数据分析

[0037] 目标系统既要记录无故障运行时的系统状态信息,也要收集故障注入后带故障运行时的系统状态信息,以系统日志的方式进行记录,便于软件测试人员事后对数据进行分析。

[0038] 在对目标系统进行故障注入前,目标系统本身应带有数据记录功能,在系统中的关键的状态和参数发生变化时,记录相关的参数数据。在进行故障注入后,由于测试人员人为的改变了系统状态(比如硬件地址中的内容、全局变量的值、软件接口数据等),目标系统应记录故障注入后的系统状态信息。在执行完故障注入后,对故障注入前后的记录数据进行分析 and 评估,用于辅助决策目标系统软件是否执行正常。

[0039] 以上所述仅是本发明的优选实施方式,应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明技术原理的前提下,还可以做出若干改进和变形,这些改进和变形也应视为本发明的保护范围。

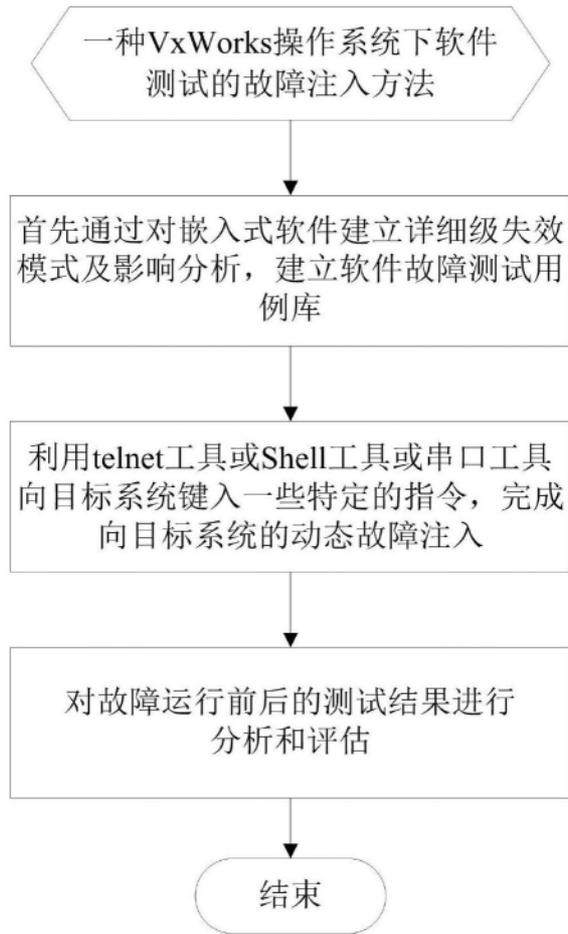


图1

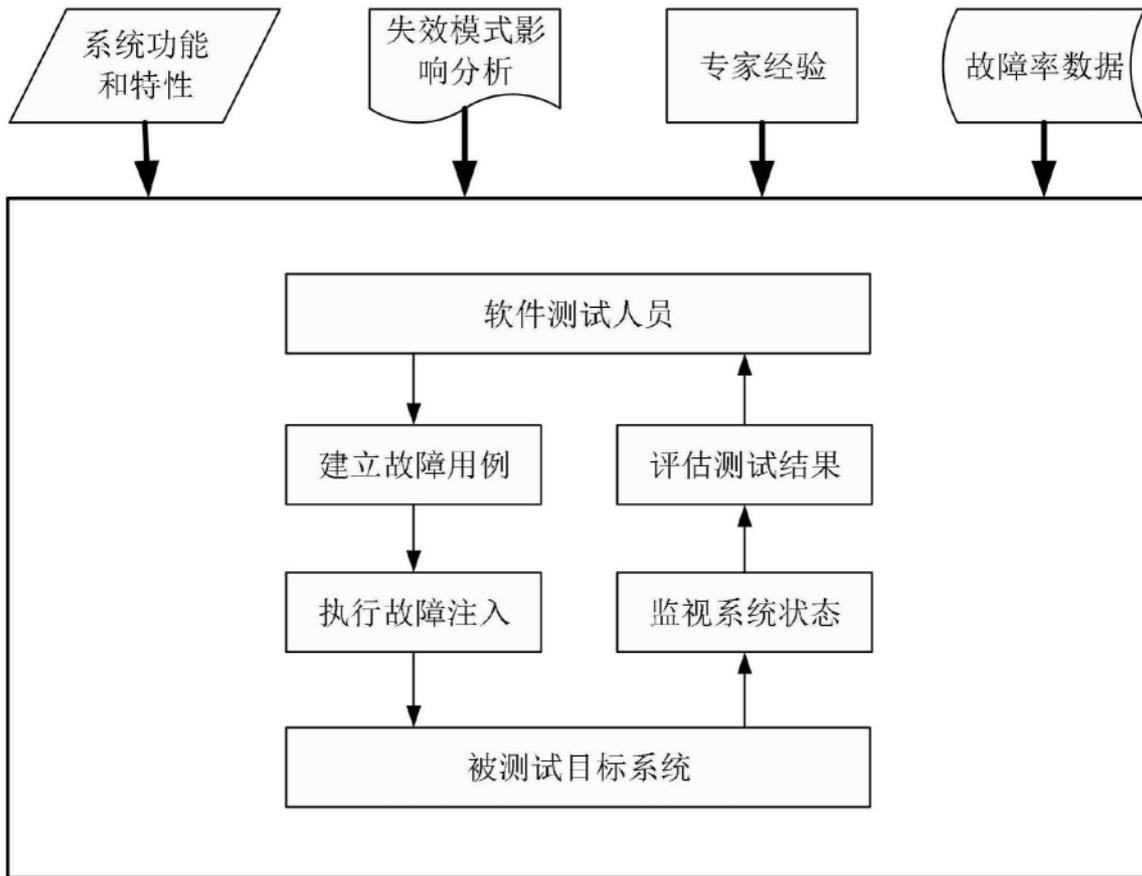


图2