



(12)发明专利申请

(10)申请公布号 CN 109902001 A

(43)申请公布日 2019.06.18

(21)申请号 201910111480.1

(22)申请日 2019.02.12

(71)申请人 科华恒盛股份有限公司

地址 361101 福建省厦门市火炬高新区火炬园马垄路457号

申请人 漳州科华技术有限责任公司

(72)发明人 张蓬勃 胡欣 洪开慧

(74)专利代理机构 石家庄国为知识产权事务所 13120

代理人 田甜

(51)Int.Cl.

G06F 11/36(2006.01)

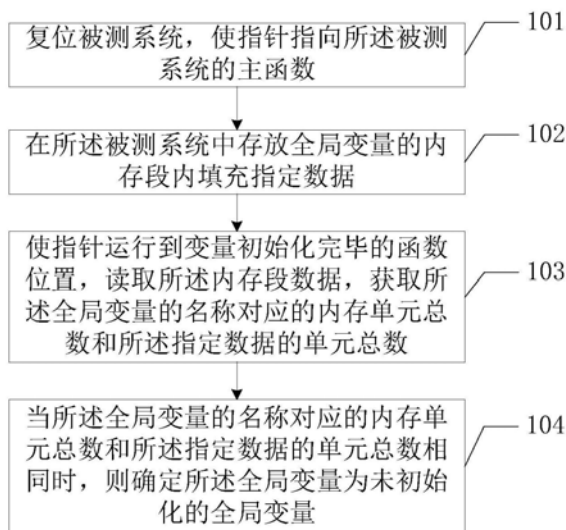
权利要求书2页 说明书9页 附图3页

(54)发明名称

对未初始化变量的检测方法及终端设备

(57)摘要

本发明适用于嵌入式软件检测技术领域,提供了一种对未初始化变量的检测方法及终端设备,该方法包括:复位被测系统,使指针指向所述被测系统主函数;在所述被测系统中存放全局变量的内存段内填充指定数据;使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数;当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量。本方法可以通过脚本自动化检测的方法,实现快速检测嵌入式软件中未初始化的全局变量。



1. 一种对未初始化变量的检测方法,其特征在于,包括:
  - 复位被测系统,使指针指向所述被测系统的主函数;
  - 在所述被测系统中存放全局变量的内存段内填充指定数据;
  - 使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数;
  - 当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量。
2. 如权利要求1所述的未初始化变量的检测方法,其特征在于,在所述复位被测系统,使指针指向所述被测系统的主函数之前,还包括:
  - 嵌入式处理器的集成开发环境IDE具备调试服务器脚本DSS工具包;
  - 调用执行所述DSS工具包;
  - 所述复位被测系统,使指针指向所述被测系统的主函数,包括:
    - 调用脚本调试复位指令,复位被测系统,使指针指向所述被测系统的主函数。
3. 如权利要求2所述的未初始化变量的检测方法,其特征在于,在所述被测系统中存放全局变量的内存段内填充指定数据,包括:
  - 调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器RAM中所有内存单元填充指定数据。
4. 如权利要求3所述的未初始化变量的检测方法,其特征在于,在所述调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器RAM中所有内存单元填充指定数据之后,还包括:
  - 调用脚本调试断点指令,在所述被测软件系统中的初始化函数后一个语句设置断点;
  - 调用脚本调试全速运行指令,全速运行所述被测软件系统到断点语句。
5. 如权利要求3或4所述的未初始化变量的检测方法,其特征在于,所述读取所述内存段,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数,包括:
  - 调用脚本调试内存读取指令,读取所述.ebss段RAM内存单元的所有指定数据;
  - 调用脚本调试变量读取指令,读取所述.ebss段地址对应的所述全局变量的名称;
  - 当获取到所述全局变量的名称时,获取第一计数器累计的所述全局变量的名称对应的内存单元总数,并获取第二计数器累计的所述指定数据的单元总数。
6. 如权利要求5所述的未初始化变量的检测方法,其特征在于,所述当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量,包括:
  - 检测是否读取到下一个全局变量的名称;
  - 当读取到下一个全局变量的名称时,则检测所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数是否相同;
  - 当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,确定当前检测的内存区对应的所述全局变量为未初始化的全局变量。
7. 如权利要求6所述的未初始化变量的检测方法,其特征在于,所述方法还包括:
  - 当确定当前检测的内存区对应的所述全局变量为未初始化的全局变量时,将所述未初始化的全局变量发送给控制窗口对应的控制单元,并将所述第一计数器和所述第二计数器

赋值为0;

当未读取到下一个全局变量的名称时,或者当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数不相同,或者将所述第一计数器和所述第二计数器赋值为0之后,检测所述.ebss段的地址是否为最后一个单元;

当所述.ebss段的地址为最后一个单元时,调用脚本调试复位指令,对所述被测系统进行复位。

8. 一种对未初始化变量的检测装置,其特征在于,包括:

复位模块,用于复位被测系统,使指针指向所述被测系统的主函数;

填充模块,用于在所述被测系统中存放全局变量的内存段内填充指定数据;

读取模块,用于使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数;

确定模块,用于当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量。

9. 一种终端设备,包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1至7任一项所述方法的步骤。

10. 一种计算机可读存储介质,所述计算机可读存储介质存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至7任一项所述方法的步骤。

## 对未初始化变量的检测方法及终端设备

### 技术领域

[0001] 本发明属于嵌入式单元测试技术领域,尤其涉及一种对未初始化变量的检测方法及终端设备。

### 背景技术

[0002] 嵌入式软件系统运行前需要对引用的全局变量进行声明和初始化。如果变量引用前没有对全局变量进行初始化,则其初始值是不固定的。对全局变量进行引用时,其运行结果不可预期,并且每次运行现象不同,这样不仅可能带来经济损失,而且给软件开发人员对问题定位带来困难,导致软件漏洞无法全部查找,增加软件开发人员调试程序的时间投入。

### 发明内容

[0003] 有鉴于此,本发明实施例提供了一种对未初始化变量的检测方法及终端设备,可以通过脚本自动化检测的方法,实现快速检测嵌入式软件中未初始化的全局变量。

[0004] 本发明实施例的第一方面提供了一种对未初始化变量的检测方法,包括:

[0005] 复位被测系统,使指针指向所述被测系统的主函数;

[0006] 在所述被测系统中存放全局变量的内存段内填充指定数据;

[0007] 使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数;

[0008] 当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量。

[0009] 在一实施例中,在所述复位被测系统,使指针指向所述被测系统的主函数之前,还包括:

[0010] 嵌入式处理器的集成开发环境IDE具备调试服务器脚本DSS工具包;

[0011] 调用执行所述DSS工具包;

[0012] 所述复位被测系统,使指针指向所述被测系统的主函数,包括:

[0013] 调用脚本调试复位指令,复位被测系统,使指针指向所述被测系统的主函数。

[0014] 在一实施例中,在所述被测系统中存放全局变量的内存段内填充指定数据,包括:

[0015] 调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器RAM中所有内存单元填充指定数据。

[0016] 在一实施例中,在所述调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器RAM中所有内存单元填充指定数据之后,还包括:

[0017] 调用脚本调试断点指令,在所述被测软件系统中的初始化函数后一个语句设置断点;

[0018] 调用脚本调试全速运行指令,全速运行所述被测软件系统到断点语句。

[0019] 在一实施例中,所述读取所述内存段,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数,包括:

- [0020] 调用脚本调试内存读取指令,读取所述.ebss段RAM内存单元的所有指定数据;
- [0021] 调用脚本调试变量读取指令,读取所述.ebss段地址对应的所述全局变量的名称;
- [0022] 当获取到所述全局变量的名称时,获取第一计数器累计的所述全局变量的名称对应的内存单元总数,并获取第二计数器累计的所述指定数据的单元总数。
- [0023] 在一实施例中,所述当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量,包括:
- [0024] 检测是否读取到下一个全局变量的名称;
- [0025] 当读取到下一个全局变量的名称时,则检测所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数是否相同;
- [0026] 当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,确定当前检测的内存区对应的所述全局变量为未初始化的全局变量。
- [0027] 在一实施例中,所述方法还包括:
- [0028] 当确定当前检测的内存区对应的所述全局变量为未初始化的全局变量时,将所述未初始化的全局变量发送给控制窗口对应的控制单元,并将所述第一计数器和所述第二计数器赋值为0;
- [0029] 当未读取到下一个全局变量的名称时,或者当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数不相同,或者将所述第一计数器和所述第二计数器赋值为0之后,检测所述.ebss段的地址是否为最后一个单元;
- [0030] 当所述.ebss段的地址为最后一个单元时,调用脚本调试复位指令,对所述被测系统进行复位。
- [0031] 本发明实施例的第二方面提供了一种对未初始化变量的检测装置,包括:
- [0032] 复位模块,用于复位被测系统,使指针指向所述被测系统的主函数;
- [0033] 填充模块,用于在所述被测系统中存放全局变量的内存段内填充指定数据;
- [0034] 读取模块,用于使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数;
- [0035] 确定模块,用于当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量。
- [0036] 本发明实施例的第三方面提供了一种终端设备,包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现上述实施例中提供的对未初始化变量的检测方法的步骤。
- [0037] 本发明实施例的第四方面提供了一种计算机可读存储介质,所述计算机可读存储介质存储有计算机程序,所述计算机程序被处理器执行时实现上述实施例中提供的对未初始化变量的检测方法的步骤。
- [0038] 本发明实施例与现有技术相比存在的有益效果是:本发明实施例通过复位被测系统,使指针指向所述被测系统的主函数;在所述被测系统中存放全局变量的内存段内填充指定数据;使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数;当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量,可以实现快速检测嵌入式软件中未初始化的全局变量。并且整个检测过程,不需额

外占用芯片资源,检测结果准确且速度快的特点。

### 附图说明

[0039] 为了更清楚地说明本发明实施例中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0040] 图1是本发明实施例提供的一种对未初始化变量的检测方法的实现流程示意图;

[0041] 图2是本发明实施例提供的另一种对未初始化变量的检测方法的实现流程示意图;

[0042] 图3是本发明实施例提供的对未初始化变量的检测装置的示例图;

[0043] 图4是本发明实施例提供的终端设备的示意图。

### 具体实施方式

[0044] 以下描述中,为了说明而不是为了限定,提出了诸如特定系统结构、技术之类的具体细节,以便透彻理解本发明实施例。然而,本领域的技术人员应当清楚,在没有这些具体细节的其它实施例中也可以实现本发明。在其它情况中,省略对众所周知的系统、装置、电路以及方法的详细说明,以免不必要的细节妨碍本发明的描述。

[0045] 为了说明本发明所述的技术方案,下面通过具体实施例来进行说明。

[0046] 本发明实施例提供一种对未初始化变量的检测方法,如图1所示,该方法包括以下步骤:

[0047] 步骤101,复位被测系统,使指针指向所述被测系统的主函数。

[0048] 可选的,在复位被测系统之前,导入调试服务器脚本(Debug Server Scripting, DSS)工具包,创建脚本环境对象,获取调试服务器并开始调试服务器对话。自动调用脚本调试复位指令,复位被测系统,使指针指向被测系统main主函数。

[0049] 本方法的执行主体可以为电脑终端,可以为计算机可读存储介质,本方法并不限定为上述执行主体,以上仅为示例说明。

[0050] 可选的,该方法要求的测试载体为嵌入式芯片最小系统板和嵌入式处理器的集成开发环境(Integrated development environment, IDE),两者使用仿真器(Joint Test Action Group, JTAG)连接并进行在线仿真。软件开发环境具有DSS功能接口,脚本语言一般支持Java或Javascript、Perl、Python、TCL等高级语言。DSS工具包在软件开发环境中,可以利用脚本指令取代手动操作,如软件复位、设置断点、全速运行、单步运行、内存填充、内存读取及内存地址对应变量的读取等功能。因此,通过一定逻辑顺序编写脚本程序,可以在最小系统板仿真环境下实现自动化测试。例如,本步骤中自动调用脚本调试复位指令,复位被测系统,使指针指向被测系统main主函数。

[0051] 步骤102,在所述被测系统中存放全局变量的内存段内填充指定数据。

[0052] 自动调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器(random access memory, RAM)中所有内存单元填充指定数据。

[0053] 可选的,指定数据可以为0x55AA等数据。嵌入式软件中的.ebss段用于全局变量的

存放。

[0054] 可选的,在本步骤之后还包括自动调用脚本调试断点指令,在所述被测软件系统中的初始化函数后一个语句设置断点;

[0055] 自动调用脚本调试全速运行指令,全速运行所述被测软件系统到断点语句。

[0056] 步骤103,使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数。

[0057] 可选的,本步骤包括:自动调用脚本调试内存读取指令,读取所述.ebss段RAM所有内存单元对应的所有指定数据;自动调用脚本调试变量读取指令,读取所述.ebss段地址对应的所述全局变量的名称;

[0058] 当获取到所述全局变量的名称时,获取第一计数器累计的所述全局变量的名称对应的内存单元总数,并且获取第二计数器累计的所述指定数据的单元总数,可选的,可用Var\_Cnt表示全局变量的名称对应的内存单元总数,可用Fill\_Cnt表示填充的指定数据的单元总数;

[0059] 步骤104,当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,则确定所述全局变量为未初始化的全局变量。

[0060] 可选的,当Fill\_Cnt=Var\_Cnt时,则确定当前检测的内存区对应的所述全局变量为未初始化的全局变量,之后将所述未初始化的全局变量发送给控制窗口对应的控制单元,并将所述第一计数器和所述第二计数器赋值为预设值,可选的,预设值可以为0。这样就可以在控制窗口获得所有未初始化的全局变量,并且可以对全局变量进行准确定位。

[0061] 本发明实施例提供一种对未初始化变量的检测方法,通过在被检测软件系统中的.ebss段RAM所有内存单元进行指定数据(0x55AA)的填充,获取全局变量的名称对应的内存单元总数和填充的指定数据的单元总数,当两者总数相同时,则可确定当前检测的全局变量为未初始化的全局变量。整个检测过程,具有操作简单,不需额外暂用芯片资源,检测结果准确且速度快等特点。

[0062] 本发明实施例提供另一种对未初始化变量的检测方法,如图2所示,该方法包括:

[0063] 步骤201,嵌入式处理器的集成开发环境IDE具备调试服务器脚本DSS工具包。

[0064] 可选的,该方法要求的测试载体为嵌入式芯片最小系统板和嵌入式处理器的集成开发环境IDE,两者使用仿真器JTAG连接并进行在线仿真。嵌入式处理器的集成开发环境IDE具有DSS功能接口,脚本语言一般支持Java或Javascript、Perl、Python、TCL等高级语言。DSS工具包在软件开发环境中,可以利用脚本指令取代手动操作,如软件复位、设置断点、全速运行、单步运行、内存填充、内存读取及内存地址对应变量的读取等功能。因此,通过一定逻辑顺序编写脚本程序,可以在最小系统板仿真环境下实现自动化测试。例如,本步骤中自动调用脚本调试复位指令,复位被测系统,使指针指向被测系统main主函数。

[0065] 可选的,导入DSS工具包,创建脚本环境对象。

[0066] 步骤202,调用执行所述DSS工具包。

[0067] 可选的,获取调试服务器脚本,开始调试服务器对话。

[0068] 步骤203,调用脚本调试复位指令,复位被测系统,使指针指向所述被测系统的主函数。

[0069] 本步骤中运行DSS工具包时,可以自动调用脚本调试复位指令。

[0070] 步骤204,调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器RAM中所有内存单元填充指定数据。

[0071] 可选的,自动调用脚本调试内存填充指令,对被测系统中.ebss段的RAM所有内存单元进行指定数据的填充。可选的,指定数据可以为0x55AA。对于嵌入式软件.ebss段用于全局变量的存放。

[0072] 步骤205,调用脚本调试断点指令,在所述被测软件系统中的初始化函数后一个语句设置断点。

[0073] 设置断点的作用为终止测试程序。

[0074] 步骤206,调用脚本调试全速运行指令,全速运行所述被测软件系统到断点语句。

[0075] 步骤207,调用脚本调试内存读取指令,读取所述.ebss段RAM内存单元的所有指定数据。

[0076] 本步骤为读取填充在.ebss段RAM中的所有指定数据。

[0077] 步骤208,调用脚本调试变量读取指令,读取所述.ebss段地址对应的所述全局变量的名称。

[0078] 步骤209,当获取到所述全局变量的名称时,获取第一计数器累计的所述全局变量的名称对应的内存单元总数,并获取第二计数器累计的所述指定数据的单元总数。

[0079] 需要说明的是,这里第一计数器和第二计数器仅是为了区分计数器进行的命名,并不是对计数器进行功能排位。

[0080] 步骤210,检测是否读取到下一个全局变量的名称。

[0081] 步骤211,当读取到下一个全局变量的名称时,则检测所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数是否相同。

[0082] 可选的,判断Fill\_Cnt与Var\_Cnt是否相等。

[0083] 步骤212,当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,确定当前检测的内存区对应的所述全局变量为未初始化的全局变量。

[0084] 步骤213,将所述未初始化的全局变量发送给控制窗口对应的控制单元。

[0085] 可选的,本步骤为将确定的未初始化的全局变量在控制窗口显示,以便对各个全局变量进行定位,然后将两个计数器清零。

[0086] 步骤214,将所述第一计数器和所述第二计数器赋值为0。

[0087] 需要说明的是,执行步骤211后,当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数不相同,执行步骤214。

[0088] 步骤215,读取所述.ebss段的下一个地址。

[0089] 可选的,执行步骤210后,当未读取到下一个全局变量的名称时,执行步骤215,即读取所述.ebss段的下一个地址,进而继续执行步骤216。

[0090] 步骤216,检测所述.ebss段的地址是否为最后一个单元。

[0091] 步骤217,当所述.ebss段的地址为最后一个单元时,调用脚本调试复位指令,对所述被测系统进行复位。

[0092] 当.ebss段的地址不是最后一个单元时,执行步骤208。

[0093] 本发明实施例提供了一种对未初始化变量的检测方法,通过采用DSS工具包进行自动化检测,实现快速定位嵌入式软件中未初始化的全局变量位置,从而可以减少软件漏

洞,并且节约软件开发人员的调试时间。

[0094] 本发明实施例提供一种对未初始化变量的检测装置,如图3所示,该装置包括:复位模块301,填充模块302,读取模块303,确定模块304。

[0095] 复位模块301,用于复位被测系统,使指针指向所述被测系统的主函数。

[0096] 可选的,所述装置还包括处理模块,用于接收并安装调试服务器脚本DSS工具包,以及执行所述DSS工具包。

[0097] 所述复位模块301,用于调用脚本调试复位指令,复位被测系统,使指针指向系统主函数。

[0098] 填充模块302,用于在所述被测系统中存放全局变量的内存段内填充指定数据。

[0099] 可选的,所述填充模块302,用于调用脚本调试内存填充指令,在所述被测系统中.ebss段的随机存取存储器RAM中所有内存单元填充指定数据。

[0100] 在填充模块302和读取模块303之间,还可以包括调用模块,用于调用脚本调试断点指令,在所述被测软件系统中的初始化函数后一个语句设置断点;调用脚本调试全速运行指令,全速运行所述被测软件系统到断点语句。

[0101] 读取模块303,用于使指针运行到变量初始化完毕的函数位置,读取所述内存段数据,获取所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数。

[0102] 可选的,当指针运行到变量初始化完毕的函数位置,所述读取模块303,用于调用脚本调试内存读取指令,读取所述.ebss段RAM内存单元的所有指定数据;调用脚本调试变量读取指令,读取所述.ebss段地址对应的所述全局变量的名称;当获取到所述全局变量的名称时,获取第一计数器累计的所述全局变量的名称对应的内存单元总数,并获取第二计数器累计的所述指定数据的单元总数。

[0103] 当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,确定模块304用于则确定所述全局变量为未初始化的全局变量。

[0104] 可选的,所述确定模块304,用于检测是否读取到下一个全局变量的名称;

[0105] 当读取到下一个全局变量的名称时,则检测所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数是否相同;

[0106] 当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数相同时,确定当前检测的内存区对应的所述全局变量为未初始化的全局变量。

[0107] 进一步的,所述装置还用于:当确定当前检测的内存区对应的所述全局变量为未初始化的全局变量时,将所述未初始化的全局变量发送给控制窗口对应的控制单元,并将所述第一计数器和所述第二计数器赋值为0;

[0108] 当未读取到下一个全局变量的名称时,或者当所述全局变量的名称对应的内存单元总数和所述指定数据的单元总数不相同,或者将所述第一计数器和所述第二计数器赋值为0之后,检测所述.ebss段的地址是否为最后一个单元,当所述.ebss段的地址为最后一个单元时,调用脚本调试复位指令,对所述被测系统进行复位。

[0109] 本发明实施例提供的一种对未初始化变量的检测装置,通过填充模块在被检测软件系统中的.ebss段RAM所有内存单元进行指定数据(0x55AA)的填充,读取模块获取全局变量的名称对应的内存单元总数和填充的指定数据的单元总数,当两者总数相同时,则确定模块可确定当前检测的全局变量为未初始化的全局变量。整个检测过程,具有操作简单,不

需额外暂用芯片资源,检测结果准确且速度快等特点。

[0110] 应理解,上述实施例中各步骤的序号的大小并不意味着执行顺序的先后,各过程的执行顺序应以其功能和内在逻辑确定,而不对本发明实施例的实施过程构成任何限定。

[0111] 图4是本发明一实施例提供的终端设备的示意图。如图4所示,该实施例的终端设备4包括:处理器401、存储器402以及存储在所述存储器402中并可在所述处理器401上运行的计算机程序403,例如对未初始化变量的检测程序。所述处理器401执行所述计算机程序403时实现上述各个对未初始化变量的检测方法实施例中的步骤,例如图1所示的步骤101至104,或者图2所示的步骤201至步骤217,所述处理器401执行所述计算机程序403时实现上述各装置实施例中各模块的功能,例如图3所示模块301至304的功能。

[0112] 示例性的,所述计算机程序403可以被分割成一个或多个模块,所述一个或者多个模块被存储在所述存储器402中,并由所述处理器401执行,以完成本发明。所述一个或多个模块可以是能够完成特定功能的一系列计算机程序指令段,该指令段用于描述所述计算机程序403在所述对未初始化变量的检测装置或者终端设备4中的执行过程。例如,所述计算机程序403可以被分割成复位模块301,填充模块302,读取模块303,确定模块304,各模块具体功能如图3所示,在此不再一一赘述。

[0113] 所述终端设备4可以是桌上型计算机、笔记本、掌上电脑及云端服务器等计算设备。所述终端设备可包括,但不仅限于,处理器401、存储器402。本领域技术人员可以理解,图4仅仅是终端设备4的示例,并不构成对终端设备4的限定,可以包括比图示更多或更少的部件,或者组合某些部件,或者不同的部件,例如所述终端设备还可以包括输入输出设备、网络接入设备、总线等。

[0114] 所称处理器401可以是中央处理单元(Central Processing Unit,CPU),还可以是其他通用处理器、数字信号处理器(Digital Signal Processor,DSP)、专用集成电路(Application Specific Integrated Circuit,ASIC)、现成可编程门阵列(Field-Programmable Gate Array,FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。

[0115] 所述存储器402可以是所述终端设备4的内部存储单元,例如终端设备4的硬盘或内存。所述存储器402也可以是所述终端设备4的外部存储设备,例如所述终端设备4上配备的插接式硬盘,智能存储卡(Smart Media Card,SMC),安全数字(Secure Digital,SD)卡,闪存卡(Flash Card)等。进一步地,所述存储器402还可以既包括所述终端设备4的内部存储单元也包括外部存储设备。所述存储器402用于存储所述计算机程序以及所述终端设备4所需的其他程序和数据。所述存储器402还可以用于暂时地存储已经输出或者将要输出的数据。

[0116] 所属领域的技术人员可以清楚地了解到,为了描述的方便和简洁,仅以上述各功能单元、模块的划分进行举例说明,实际应用中,可以根据需要而将上述功能分配由不同的功能单元、模块完成,即将所述装置的内部结构划分成不同的功能单元或模块,以完成以上描述的全部或者部分功能。实施例中的各功能单元、模块可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中,上述集成的

单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。另外,各功能单元、模块的具体名称也只是为了便于相互区分,并不用于限制本申请的保护范围。上述系统中单元、模块的具体工作过程,可以参考前述方法实施例中的对应过程,在此不再赘述。

[0117] 在上述实施例中,对各个实施例的描述都各有侧重,某个实施例中沒有详述或记载的部分,可以参见其它实施例的相关描述。

[0118] 本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本发明的范围。

[0119] 在本发明所提供的实施例中,应该理解到,所揭露的装置/终端设备和方法,可以通过其它的方式实现。例如,以上所描述的装置/终端设备实施例仅仅是示意性的,例如,所述模块或单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通讯连接可以通过一些接口,装置或单元的间接耦合或通讯连接,可以是电性,机械或其它的形式。

[0120] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0121] 另外,在本发明各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。

[0122] 所述集成的模块/单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时,可以存储在一个计算机可读取存储介质中。基于这样的理解,本发明实现上述实施例方法中的全部或部分流程,也可以通过计算机程序来指令相关的硬件来完成,所述的计算机程序可存储于一计算机可读存储介质中,该计算机程序在被处理器执行时,可实现上述各个方法实施例的步骤。。其中,所述计算机程序包括计算机程序代码,所述计算机程序代码可以为源代码形式、对象代码形式、可执行文件或某些中间形式等。所述计算机可读介质可以包括:能够携带所述计算机程序代码的任何实体或装置、记录介质、U盘、移动硬盘、磁碟、光盘、计算机存储器、只读存储器(ROM,Read-Only Memory)、随机存取存储器(RAM,Random Access Memory)、电载波信号、电信信号以及软件分发介质等。需要说明的是,所述计算机可读介质包含的内容可以根据司法管辖区内立法和专利实践的要求进行适当的增减,例如在某些司法管辖区,根据立法和专利实践,计算机可读介质不包括电载波信号和电信信号。

[0123] 以上所述实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围,均应

包含在本发明的保护范围之内。

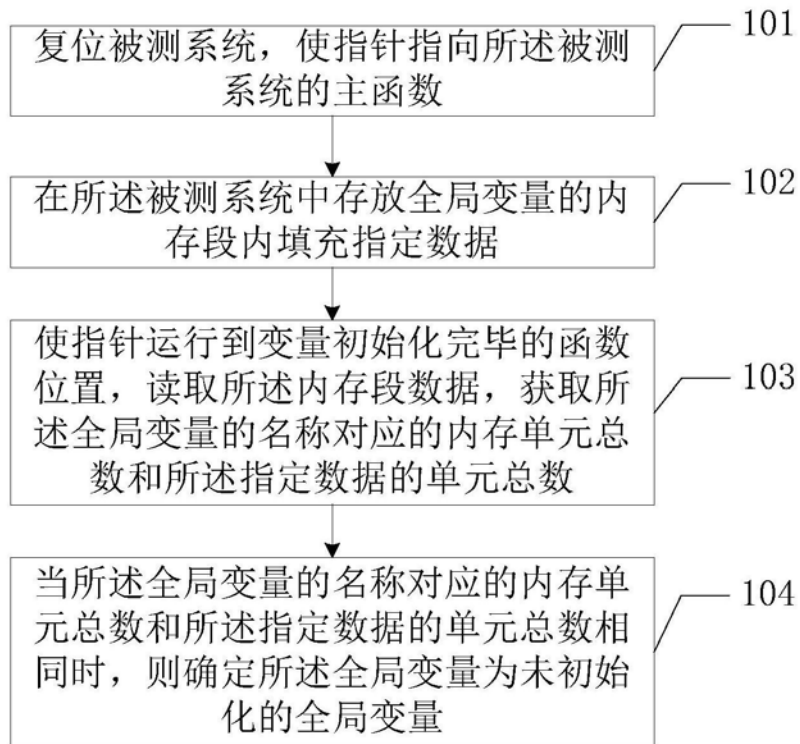


图1

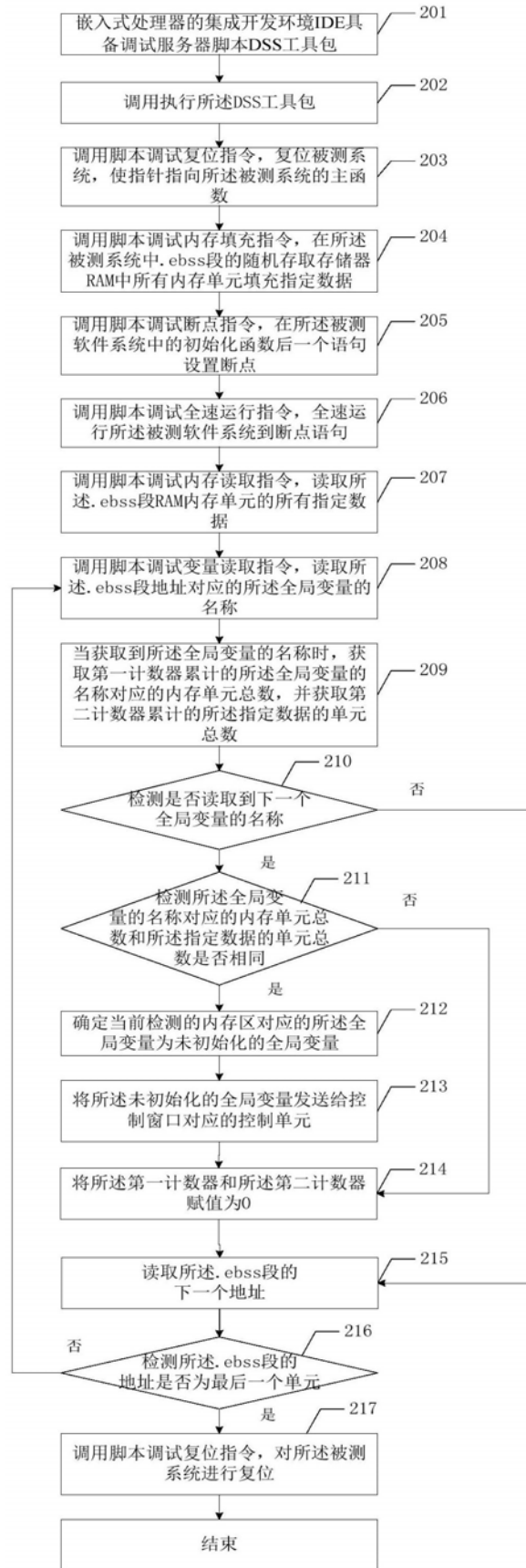


图2

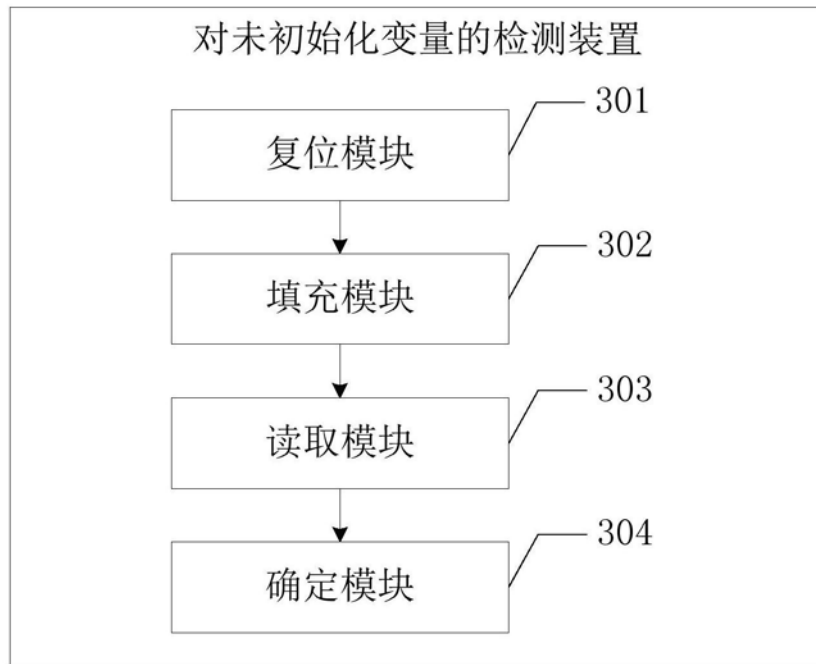


图3

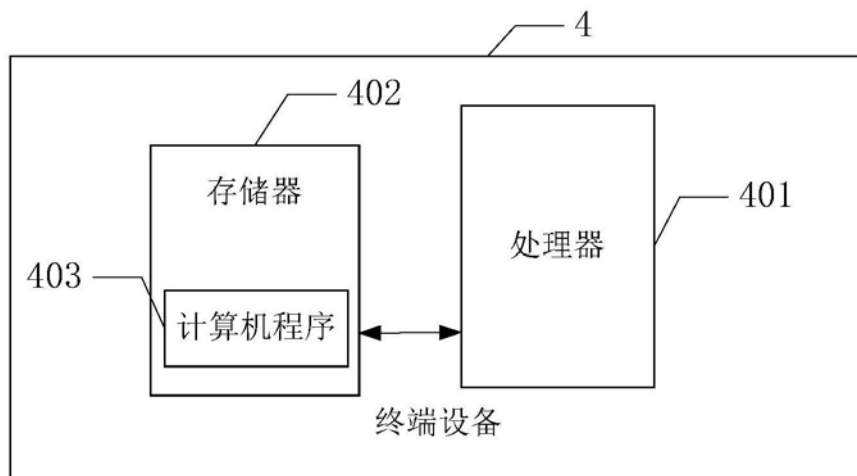


图4