[19] 中华人民共和国国家知识产权局

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

[12] 发明专利说明书

专利号 ZL 200510049298.6

[45] 授权公告日 2008 年 1 月 30 日

[11] 授权公告号 CN 100365590C

[22] 申请日 2005.1.31
[21] 申请号 200510049298.6
[73] 专利权人 浙江大学
地址 310027 浙江省杭州市西湖区浙大路 38 号
[72] 发明人 卜佳俊 陈 纯 沈格俊 赵 军 柯化成
[56] 参考文献
CN1487415 A 2004.4.7
CN1359496 A 2002.7.17
JP2001318805 A 2001.11.16
CN1286431 A 2001.3.7
CN1287309 A 2001.3.14
审查员 吉张媛

[74] 专利代理机构 杭州求是专利事务所有限公司
代理人 林怀禹

权利要求书 1 页 说明书 4 页

[54] 发明名称

在嵌入式系统模拟器上调试应用程序的方法

[57] 摘要

本发明公开了在嵌入式系统模拟器上调试应用程序的方法。 本发明的方法通过读取操作系统编译时生成的内核符号文件，得到位于模拟器内存中的应用程序进程控制块信息，从而获得对应用程序的控制能力，实现对运行于模拟器中操作系统之上的应用程序的调试。 其优势在于不需要操作系统的网络支持，不需要运行额外的调试器程序，从而避免了嵌入式操作系统上缺乏调试器的问题；同时，本方法不在虚拟器的操作系统上执行额外的程序，提高了模拟器的性能，改善响应速度。

1. 一种在嵌入式系统模拟器上调试应用程序的方法，其特征在于该方法的步骤如下：

1）调试器使用者指定待调试应用程序在模拟器上操作系统中的进程标识号；

2）在符号分析加载模块中，对操作系统编译时生成的内核符号表进行扫描分析，得到进程控制块结构实例的内存地址，并遍历所有进程控制块结构实例，根据步骤 1）中指定的进程标识号取得待调试应用程序的进程控制块的内存地址；

3）如果操作系统支持虚地址机制，则虚拟内存访问模块实现与模拟器上运行的操作系统同样的地址转换算法，根据 2）中获得的内存地址计算出对应的物理地址，如若遇见所对应的内存页不在物理内存中，则需进一步解析进程控制块结构中的内存数据结构来获取其在模拟文件系统或交换文件中所在的块号并将页加载至内存，若上层运行的操作系统不支持虚地址机制，则本步骤可以省去；

4）调试接口模块接受来自程序开发人员的各种调试指令，送至调试代理模块，并将调试代理模块接受指令后产生的调试信息接受并显示；

5）调试代理模块收到 4）中调试接口模块传来的调试指令，若调试指令为插入断点，则将目的断点地址对应的指令修改为自行定义格式的断点指令，被覆盖的原指令连同断点位置被保存下来，当虚拟机遇到该断点指令时，则停机等待从调试器发过来的调试指令；若调试指令为现场查询指令或现场修改指令，则根据 2）中获得的进程控制块中存储的内存地址对属于进程的内存地址或者由模拟器模拟的寄存器进行相应的操作；若调试指令为继续执行，则将保存的原指令恢复，让虚拟机恢复运行状态；若为显示进程列表指令，则调用符号分析加载模块根据 2）所述获取进程列表数据结构在内核中的内存地址，并遍历该结构；若为指定调试进程指令，则在模拟器内部维护一个当前调试进程信息的数据结构。

在嵌入式系统模拟器上调试应用程序的方法

技术领域

本发明涉及计算机程序调试技术，特别是在嵌入式系统模拟器上调试应用程序的方法。

背景技术

随着嵌入式设备普及，例如个人数字助手（PDA），手机等，基于便携移动设备的应用大量涌现。而嵌入式设备上的应用程序一般都在与目标平台异构的机器环境下开发，不能在开发环境下调试，且嵌入式设备的开发评估的硬件仿真设备一般都比较昂贵，因此调试成为制约着嵌入式软件开发的重要因素。

计算机系统模拟器技术是在某种架构的计算机中通过模拟目标架构计算机的硬件特征和其指令执行的过程，从而实现在一台计算机上虚拟出多台计算机的技术。流行的计算机系统模拟器例如：Bochs，可以在多种体系结构的机器上模拟多台 Intel 80x86 体系的计算机，并运行 Linux、freebsd 等操作系统的 Intel 80x86 版本；armulator，可以在 Intel 80x86 体系结构的机器上模拟 ARM 体系的计算机，并运行 ucLinux 的 ARM 移植版本等。

利用计算机系统模拟器的特点，在模拟器上进行嵌入式软件的调试成为解决嵌入式系统调试问题的一种解决方案，但是现有的基于模拟器的调试技术通常只能调试运行于嵌入式系统模拟器上的操作系统，而对操作系统之上的应用程序无能为力，这限制了基于模拟器调试技术发挥作用的空间；或者通过在操作系统上额外运行调试器，例如 gdb，但如果在该操作系统上无可用的调试器（如 ucLinux），则无法使用；或者在操作系统上运行调试器代理，例如 gdb server，但要求模拟器支持虚拟的网络连接同时操作系统必须实现 TCP/IP 等网络协议栈，而这些要求许多嵌入式操作系统不能满足，即时满足，也增加了模拟器执行的程序，影响模拟器性能。

进程是计算机程序执行的基本单位，也是调试的基本单位。关于进程的基本信息都储存在操作系统的进程控制块结构中，包括待调试进程的唯一标识和进程的上下文信息（与 CPU 相关的状态，包括寄存器，页表基址，段基址等）。因而透过模拟器上的操作系统调试应用软件的关键就在于获得操作系统内存空间中的进程控制块信息。

发明内容

本发明的目的在于提供一种在嵌入式系统模拟器上调试应用程序的方法。

本发明采用的技术方案是：

1）调试器使用者指定待调试应用程序在模拟器上操作系统中的进程标识号；

2）在符号分析加载模块中，对操作系统编译时生成的内核符号表进行扫描分析，得到进程控制块结构实例的内存地址，并遍历所有进程控制块结构实例，根据步骤 1）中指定的进程标识号取得待调试应用程序的进程控制块的内存地址；

3）如果操作系统支持虚地址机制，则虚拟内存访问模块实现与模拟器上运行的操作系统同样的地址转换算法，根据 2）中获得的内存地址计算出对应的物理地址，如若遇见所对应的内存页不在物理内存中，则需进一步解析进程控制块结构中的内存数据结构来获取其在模拟文件系统或交换文件中所在的块号并将页加载至内存，若上层运行的操作系统不支持虚地址机制，则本步骤可以省去；

4）调试接口模块接受来自程序开发人员的各种调试指令，送至调试代理模块，并将调试代理模块接受指令后产生的调试信息接受并显示；

5）调试代理模块收到 4）中调试接口模块传来的调试指令，若调试指令为插入断点，则将目的断点地址对应的指令修改为自行定义格式的断点指令，被覆盖的原指令连同断点位置被保存下来，当虚拟机遇到该断点指令时，则停机等待从调试器发过来的调试指令；若调试指令为现场查询指令或现场修改指令，则根据 2）中获得的进程控制块中存储的内存地址对属于进程的内存地址或者由模拟器模拟的寄存器进行相应的操作；若调试指令为继续执行，则将保存的原指令恢复，让虚拟机恢复运行状态；若为显示进程列表指令，则调用符号分析加载模块根据 2）所述获取进程列表数据结构在内核中的内存地址，并遍历该结构；若为指定调试进程指令，则在模拟器内部维护一个当前调试进程信息的数据结构。

本发明和技术相比具有的有益的效果是：其优势在于不需要操作系统的网络支持，不需要运行额外的调试器程序，从而避免了嵌入式操作系统上缺乏调试器的问题；同时，本方法不在虚拟器的操作系统上执行额外的程序，提高了模拟器的性能，改善响应速度。

具体实施方式

某 ARM 模拟器中，实现了对 ARMv4 版本指令的解释执行，模拟了 Atmel 公

司 AT91EV40 开发板，在其上能够运行 ucLinux 版本 2.0.0。该版本 ucLinux 编译时生成内核符号文件为/boot/system.map。待调试的程序名假设为 dcbuggedproc,调试器为 debugger,模拟器名为 simon, 在模拟器中实现了符号分析加载模块、虚拟内存访问模块、调试代理模块。同时在另外独立运行的调试器客户端中包含了调试接口模块，该模块和模拟器的调试代理模块通过套接字进行通讯，通讯格式遵循 GDB 远程调试协议。

1）调试器 debugger 通过符号分析加载模块获取的进程列表来确定待调试程序 debuggedproc 在目标操作系统上的进程 ID，假设为 1000。此时，调试器可以通过 attach 1000 指令根据技术方案中 1）所述指定待调试的进程 ID。

2）符号分析加载模块，对操作系统编译时生成的内核符号表 system.map 进行扫描分析，得到进程控制块结构数组 task 对应的内存地址，task 数组中的项为进程控制块结构，通过遍历所有进程控制块结构实例，根据技术方案中 1）中指定的进程标识号取得待调试应用程序的进程控制块的内存地址。根据操作系统支持虚地址与否，该内存地址可能是虚拟地址，也可能是物理地址；

3）虚拟内存访问模块，如果操作系统支持虚地址机制，则该模块实现与模拟器上运行的操作系统同样的地址转换算法，根据技术方案中 1）中获得的虚地址计算出对应的物理地址，如若遇见所对应的内存页不在物理内存中，则需进一步解析进程控制块结构中的内存数据结构来获取其在模拟文件系统或交换文件中所在的块号并将页加载至内存；若上层运行的操作系统不支持虚地址机制，则本步骤可以省去；

4）调试接口模块，该模块接受来自程序开发人员的各种调试指令（如显示进程列表、指定调试进程、查看内存、修改内存、插入断点、查看断点、删除断点、查看寄存器、修改寄存器、单步执行等），依据 gdb 远程调试协议格式的要求解析出调试指令及其调试参数并送至调试代理模块，并将调试代理模块接受指令后产生的反馈调试信息接受并显示；

5）调试代理模块，该模块收到调试接口传来的调试指令及其调试参数，若调试指令为插入断点，则将目的断点地址对应的指令修改为自行定义格式的断点指令，被覆盖的原指令连同断点位置被保存下来，当虚拟机遇到该指令时，则停机等待其它调试指令；若调试指令为现场查询指令或现场修改类指令，则根据技术方案中 2）中获得的进程控制块中存储的内存地址对属于进程的内存地址或者由模拟器模拟的寄存器进行相应的操作；若调试指令为继续执行，则将保存的原指令恢复，让虚拟机恢复运行状态。若为显示进程列表指令，则调用

符号分析加载模块根据技术方案中2)所述获取进程列表数据结构在内核中的内存地址，并遍历该结构。若为指定调试进程指令，则在模拟器内部维护一个当前调试进程信息的数据结构。

使用本方法不需要在ucLinux下运行额外的调试程序（例如GDB），也不要求该ucLinux实现TCP/IP协议，极大的减少了模拟器在解析这些程序或协议的指令时所造成的性能损耗。