

一种基于 PLI 和 Simics 的微处理器协同验证平台

周文涛,樊晓桢,黄小平
(西北工业大学 航空微电子中心,陕西 西安 710072)

摘 要: 面向微处理器验证,构建了一种基于 Verilog PLI 和 Simics 模拟器的微处理器验证平台.该平台通过 Verilog PLI,利用设计的控制模块将待测微处理器设计模型与 Simics 模拟器相连,协同自动化比较运行结果.该平台应用在龙腾 R 处理器的验证中.结果显示该平台有效增加了验证人员对验证过程的可观测性和可控性,具有灵活性高,仿真速度快等特点.

关键词: 微处理器验证; Verilog PLI; Simics

中图分类号: TP368.1

文献标识码: A

文章编号: 1000-7180(2010)11-0162-04

A Microprocessor Co-verification Platform Based on PLI and Simics

ZHOU Wen-tao, FAN Xiao-ya, HUANG Xiao-ping

(Aviation Microelectronics Center, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: We setup a microprocessor co-verification platform based on Verilog PLI and Simics for microprocessor verification. By the Verilog PLI, the platform connects the design microprocessor module with Simics, and compares the results automatically. The platform was used in "Longtium R" microprocessor verification. The results show that this platform enhances the observability and control of the verification process, and is characterized by lightness, quickness of simulation.

Key words: microprocessor verification; Verilog PLI; Simics

1 引言

随着芯片设计复杂度的增大,验证的复杂度也在不断增大.验证已经成为芯片设计的瓶颈.因此,寻找更加丰富的验证手段或者更有效的验证方法十分重要^[1].

基于模拟器的验证的方法已经在越来越多的微处理器设计验证中得到应用^[2].文中提出一种基于 PLI 和 Simics 模拟器的微处理器的验证平台.该平台继承了 Verilog PLI 和 Simics 的特点,灵活性高,仿真速度快,错误定位能力强,有很好的移植性,有效增加了验证人员对验证过程的可观测性和可控性.该平台^[2]已在龙腾 R 验证中应用.

2 相关工作

2.1 Verilog PLI 介绍

Verilog PLI^[3]提供了一种标准的机制来访问和控制 Verilog 代码,它主要有 2 个版本:PLI1.0 和 PLI2.0. PLI 主要提供以下 3 种功能:

(1) PLI 允许用户编写自定义的系统函数.用户编写相应的 PLI 程序并连接到仿真器后,可以在 Verilog 代码中使用这些系统函数.一旦这些自定义的函数在仿真过程中被调用,仿真器就会找到对应的用户编写的 PLI 程序来执行,从而实现了 Verilog 的扩展.

(2) PLI 允许用户在自己的 PLI 程序中与 Ver-

ilog 硬件进行交互,比如读写 reg 变量、存储器访问等.这样,用户就可以在自定义的函数中对硬件执行某些用 Verilog 语言难以完成的操作.

(3) PLI 提供了一种函数回调的机制.用户可以将某个 wire/reg 等信号挂上一个 PLI 程序中的 C 函数,以后每当该信号变化时,对应的 C 函数都会被调用,从而方便地实现信号监测,同时 PLI 还能让用户控制仿真的过程,比如暂停、退出和往日志文件中写信息等.

2.2 Simics 介绍

Simics^[4]是由 Virtutech 公司开发的全系统仿真平台.该平台不仅包括 CPU 还包括可以被操作系统和相关设备驱动访问的硬件设备,可以运行真实的固件(firmware)和完全未经过修改的内核和驱动代码,它在指令级进行抽象,因而具有很好的性能,并且达到可以运行商用软件的功能精度.该平台提供了包括 X86、MIPS、PowerPC、Sparc 等主流指令集系统的配置,包括处理器和相应外围设备模型,以及已经安装好 linux 操作系统的虚拟磁盘镜像.使用这些配置,使用者可以在宿主机上对相应的虚拟系统仿真^[5].

Simics 提供了良好的控制台操作界面,控制台上不仅提供了单步、多步、断点状态保存等控制命令,还能查看寄存器值、执行程序的汇编代码和物理地址等.对 Simics 模拟器的控制有两种方式.一种是通过控制台的 CLI(command line interface)命令行接口,直接手工加载或配置模块;另一种是加载由配置好的模块组成的脚本. Simics 中的模块可以通过设置已有模块的可变参数来实现,也可以是利用应用编程接口(API)来设计自定义功能的模块.这种自定义功能的模块在 Simics 中称为设备(device),用户有三种语言去编写设备,分别是 DML(device modeling language)、Python、C/C++.其中,用 DML 是最方便、高效的,考虑到有些 API 函数如 Micro-Architectural Interface 函数仅支持 C/C++,所以对性能要求严格的模块常用 C/C++来编写. Simics 中提供了丰富的 API 函数,在其参考手册中所有的 API 函数的说明^[6].

Simics 具有快速仿真的能力,可以达到甚至超过真实硬件的速度,它不仅仅可以在指令级仿真目标机的应用程序,而且同时能收集程序运行中的处理器状态,如寄存器的值、pc 值等.

3 微处理器协同验证平台的设计

设计微处理器验证平台的目的是为了待测的

微处理器的验证提供一种更有效的验证手段.一方面,PLI 扩展了 Verilog,并能灵活的与 Verilog 硬件模型交互;另一方面,Simics 模拟器能提供正确的指令级运行的参照模型,同时 Simics 中的 API 提供了丰富的控制函数.结合这两方面设计验证平台,也就能继承 PLI 和 Simics 的特性.

图 1 是设计的该微处理器协同验证平台的总体结构,该平台由控制模块、待测模块、Simics 模块三部分组成.控制模块采用基于共享内存的方式分别和待测模块、Simics 模块通信.该平台是多进程验证平台.

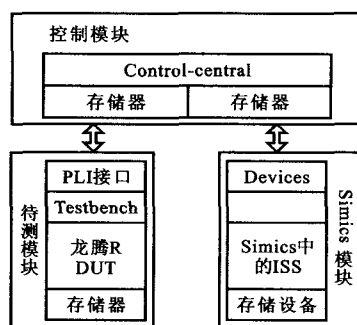


图 1 微处理器协同验证平台结构

3.1 控制模块

控制模块由存储器和 Control-central 控制中心组成,Control-central 的主要功能有:(1)初始化存储器,设置公用变量的值如程序断点的 pc 值、同步的步长等;(2)与另两个模块建立连接,接受数据,分别存储两个模块的处理器状态如寄存器的值、pc 值等;(3)负责设置验证调试模式,可以直接将两种处理器状态自动比较并以日志的方式输出,也可以在每个同步点用 Simics 模块的处理器状态修改待测模块的处理器状态等.

3.2 待测模块

待测模块由 PLI 接口函数、Testbench、“龙腾 R”DUT、存储器组成.主要功能有:(1)在硬件仿真器中仿真整个设计;(2)读取共享内存区的同步步长等公用参数;(3)通过 PLI 接口函数发送接收处理器状态.

3.3 Simics 模块

Simics 模块由 Devices、ISS、存储设备三部分组成.Devices 为加载到 Simics 模拟器中的自定义功能设备,利用 Simics 中的 API 收集处理器状态,如利用 Sim_instruction_read_input_reg 返回输入寄存器的值. ISS 可以直接通过脚本设置为 PowePC750 的指令集.该模块的主要功能通过 De-

vices 发送处理器状态,同时控制 Simics 模拟器的仿真进度.

4 运行与控制方式

在该平台中,待测模块和 Simics 模块是作为两个独立的进程运行在操作系统上的.该平台的工作流程如下:先在控制模块中设置验证调试模式、同步步长等;然后启动 Simics,加载配置好的脚本,同时在硬件仿真器中仿真待测设计;Simics 模块在同步点通过 Devices 向控制模块发送处理器状态,同时待测模块也通过 PLI 接口函数向控制模块发送处理器状态;控制模块对两个模块发送过来的处理器状态进行一致性检测,若出现不一致,则可能出现错误.根据验证调试模式,可以将错误点记录并继续执行,也可以用正确的处理器状态修改待测模型中的错误,然后继续执行.下面就该平台的一些关键点加以阐述.

4.1 平台同步机制

该平台在同步点对两个模型的处理器状态进行一致性检查,因此同步点的选取和同步方式对验证的效果很重要.常用的同步方式有两种:基于时钟的同步和基于步长的同步.由于 Simics 模拟器的处理器模型是以指令集仿真器 ISS 的形式实现的,因此测试程序的执行是以指令粒度来执行的,任何指令的执行只需要一拍;而龙腾 R 待测模型中指令的执行是基于系统时钟的,需要多拍完成.因此采用基于步长的同步较为合理.龙腾 R 采用流水线结构,当一条指令的最后一条微指令流出流水线,则标志着该条指令执行完成.因此,待测模型中的同步点的设置需要获取指令完成的标志.

本平台采用基于同步步长的方式,即两模块同时执行相同数目的指令.当同步步长设置为 1 的时候,该平台中的两个模型的同步点最多,比较频度最高,也最能发现所有的不一致,但耗时较多.相反,当同步步长设置较大时,能起到加快速度的效果.因此,在验证中可先设置较长的步长,出现错误时,再减少步长,这样就起到了在同步机制下更准确地定位错误位置的效果.

4.2 验证调试方法

通常情况下,Simics 模块和待测模块在同步点分别向控制模块发送处理器状态,然后由控制模块来比较、修改并输出结果,错误点可做特别处理.同时 Simics 模拟器提供了设置断点的功能,PLI 能与 Verilog 硬件进行交互.利用这些特点,可以更加丰

富验证调试的方法,先在 Simics 中设置断点,Simics 运行到断点自动停止,读取此刻的处理器状态,并通过 PLI 接口函数将处理器状态复制到待测模型中,这样就加快了验证的速度.也可以将某个寄存器的信号挂上一个 PLI 中的 c 函数,在每个同步点监视该信号,这样可以使调试更有方向,增加了验证过程的观测性和可控性.

4.3 通信方式

该平台在运行时涉及到控制模块与待测模块以及控制模块与 Simics 模块不同进程之间的通信.进程间的通信有多种方式,设计中采用了基于共享内存的方式,在共享内存区为每个寄存器建立一个链表,待测模块向控制模块发送数据时,直接写入共享内存区;Simics 模块也一样将数据发送到另一个共享内存区.控制模块直接从两个共享内存区读取数据并比较,或者用一个共享内存区的数据去修改另一共享内存区的数据.

4.4 测试激励

一个好的测试激励对验证的效果起到很重要的作用.文中主要由 vxworks 应用操作系统程序作为激励.对于微处理器系统级验证,使用操作系统或者大型应用程序作为激励更能发现错误.本平台中涉及到两个模型之间的协同比较,因此输入的测试激励必须一样.首先将测试程序如操作系统编译成二进制码;在 Simics 模块中,通过 load-binary 命令读入内部虚拟存储器;在待测模块中将相同的二进制文件存入 RAM 中,然后两模块各自取指令并执行.

5 应用结果及分析

龙腾 R 是西北工业大学航空微电子中心设计的兼容 PowerPC750 指令系统的 32 位嵌入式微处理器.利用 Simics-3.0.12、ncsim、PLI2.0 构建了协同验证平台,并在龙腾 R 验证中应用.表 1 给出了在几种测试向量下的应用效果.

表 1 测试向量应用效果

仿真激励	指令条数 (万条)	仿真时间 (小时)	发现错误 (个)
Vxworks	320	3.3	13
随机生成	598	5.1	7
手工编写	40	0.8	9

由结果可以看出,基于该平台的仿真能达到 269 条/s,速度较快;同时利用操作系统作为激励可以发现更多的错误.

6 结束语

随着微处理器验证复杂度的加大,采用一种合适的验证方法,可加快验证的进度;同时多一种验证手段也提高了验证结果的可信度.文中提出的基于 PLI 和 Simics 的微处理器协同验证平台为微处理器验证提供了一种新的手段,该平台继承了 Verilog PLI 和 Simics 的特点,具有仿真速度快,过程控制能力强,有很好的移植性,增加了验证过程的可观测性和可控性等特点.

参考文献:

- [1] Janick Bergeron. Writing testbenches functional verification of HDL models[M]. 北京:电子工业出版社,2006:6.
- [2] Chang Yousung, Seungjong Lee. Verification of microprocessor using real world applications[C]// Proceedings of the 36th ACM/IEEE conference on Design Automation.

(上接第 161 页)

参考文献:

- [1] Maranas C, Floudas C. Finding all solutions of nonlinearly constrained systems of equations[J]. Journal of Global Optimization, 1995,7(2):143-182.
- [2] Robillson S. Extension of newton's method to nonlinear functions with values in a cone[J]. Numerical Mathematics, 1972(19):341-347.
- [3] Hentenryck P, Miehél L, Deville Y. Nomerica: a modeling language for global optimization [M]. Cambridge: The MIT Press, 1997:1-16.
- [4] 邹秀芬,刘敏忠,吴志健,等.解约束多目标优化问题的一种鲁棒的进化算法[J].计算机研究与发展,2004,41

USA: New Orleans, 181-184.

- [3] IEEE Std 1364-2001 IEEE standard Verilog hardware description language[S]. 2001.
- [4] Peter S Magnusson, Magnus Christensson, Jesper Eskilsson, et al. Simics: a full system simulation platform[J]. IEEE Computer, 2002,35(2):50-58.
- [5] 张山岗,高德远.基于虚拟机的兼容微处理器验证平台[J].微电子学与计算机,2005,22(1):141.
- [6] Virtutech AB: Simics reference manual [S]. Sweden, Stockholm, Simics-3.0, 2006.

作者简介:

周文涛 男,(1984-),硕士研究生.研究方向为专用集成电路设计、微处理器体系结构.

樊晓娅 男,(1962-),教授,博士生导师.研究方向为计算机体系结构、VLSI 系统设计.

黄小平 男,(1980-),讲师.研究方向为专用集成电路设计、VLSI 系统设计.

(6):985-990.

- [5] 卢侃等.混沌动力学[M].上海:上海远东出版社,1990.
- [6] 褚美茹,崔明勇.混沌遗传算法在配电网无功优化中的应用[J].信息技术,2004,28(10):48-50.
- [7] Ge J X, Chou S C, Gao X S. Geometrie constraint solving with optimization methods[J]. CAD, 2002,31(14):867-879.

作者简介:

赵空 男,(1973-),硕士,高级讲师.研究方向为参数化建模、智能优化算法.