

基于 FPGA 的软硬件协同仿真加速技术

江霞林 周剑扬 杨银涛 林晓立
(厦门大学 电子工程系 厦门 361005)

摘要 在系统设计中,硬件复杂电路设计的调试与仿真工作对于设计者来说十分困难。为了降低仿真复杂度,加快仿真速度,本文提出利用 FPGA 加速的思想,实现软硬件协同加速仿真。经过实验,相对于纯软件仿真,利用软硬件协同加速仿真技术,仿真速度提高近 30 倍,大大缩短了仿真时间。

关键词 现场可编程门阵列 软硬件协同仿真 仿真加速

FPGA Based Accelerator for Hardware/Software Co-Simulation

JIANG Xia-lin, ZHOU Jian-yang, YANG Yin-tao, LIN Xiao-li
(Department of Electronic Engineering, Xiamen University, Xiamen 361005)

Abstract In system design, debugging for the design becomes increasingly difficult and designers want more efficient and high-performance verification and debugging solutions. As the design becomes larger and more complex, the pure software simulation suffers from the speed problem. In this paper, we present a new debugging methodology: FPGA based accelerator for hardware/software co-simulation. Experimental results show that the performance gain is up to 30 times over the pure software simulation.

Keywords FPGA; Hardware/Software Co-Simulation; Simulation Speed-up

1 前言

在数字集成电路的设计中,当设计工程师在用硬件描述语言(HDL:Hardware Description Language)完成设计之后,需要通过仿真来检验设计是否满足预期的功能。在仿真中,设计工作者需要为设计项目建立一个测试平台,这个测试平台为设计项目提供尽可能完备的测试激励,并提供可供观测

的输出响应。根据这些输出响应信息,设计工程师便可以判断设计项目是否满足预期的功能。在进行仿真工程时,设计工程师一般先对各个功能模块进行仿真验证,全部通过后再对整个系统设计进行仿真。当设计工程师在仿真中发现错误,就需要进行仔细调试,找出错误发生的原因并加以修改。

随着系统设计的复杂性不断增加,当设计集成度超过百万门后,设计正确性的验证比设计本身还

要费劲，系统仿真的实时性很难满足要求。在针对复杂电路进行软件仿真时，系统的仿真时间往往需要占据大部分的设计时间。我们常常会为了仿真电路的某些功能，而不得等上几个小时甚至几天。如何提高仿真效率，减少仿真复杂度，缩短仿真时间，将成为系统设计中的关键一环。文献[1]、[2]提出利用基于 C 语言的设计和验证方法来代替传统的基于 HDL 语言设计的仿真，从而加快仿真速度，但是这种方法只适用设计的早期阶段。为了方便而快速的实现仿真验证，及时得到测试数据，本文提出运用硬件加速的思想^{[3][4][5]}，采用硬件仿真平台和软件仿真平台相互通信，即通过主机上运行的仿真软件与硬件平台相结合，实现软硬件协同加速仿真，仿真速度可以提高 30 倍。

2 软硬件协同加速仿真

在传统的设计与验证过程中，设计工程师首先将复杂的系统逐模块的用硬件描述语言表述，待所有模块在仿真器上单独验证通过后，通过模块间整合进行局部和整个设计的仿真，如图 1 所示。

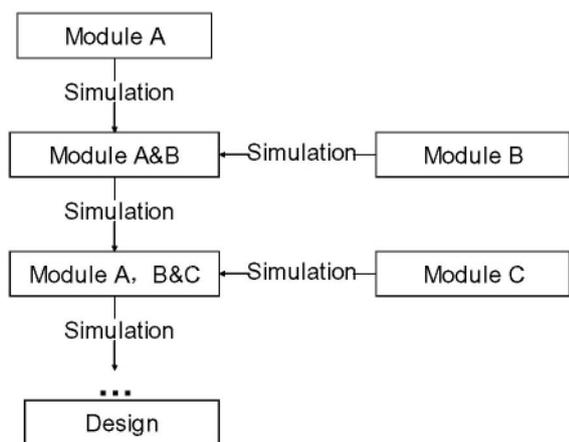


图 1 设计验证进程

假设模块 Master 和模块 Slave 是整个复杂设计中的一部分。模块 Master 负责把输入数据进行数据处理，随后把处理后数据发送到下一个模块 Slave，Slave 模块完成一个功能复杂的算法运算，运算结束后把结果返回到模块 Master 中，进行下一步操作，

设计框图如图 2 所示。

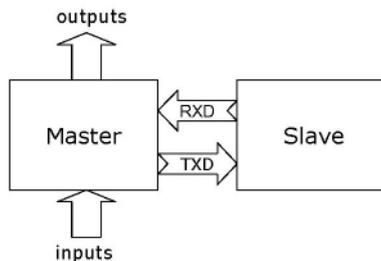


图 2 设计例子框图

设计工程师在完成模块 Master 和模块 Slave 的 HDL 设计后，用 HDL 仿真器软件分别对两个模块进行仿真验证，模块 Master 的仿真时间花费了五分钟，模块 Slave 花费了十五分钟，两个模块进行联合仿真花费了二十分钟。如果设计不正确，则要对设计进行重新修改和仿真直到验证通过为止，重复的仿真工作将要花费几天甚至几星期。为了缩短仿真时间，本文提出利用硬件加速的思想，对设计进行软硬件协同加速仿真。模块 Master 和模块 Slave 的功能首先分别在软件上仿真验证通过，待模块 Slave 经综合实现后，把模块 Slave 下载到硬件中，模块 Master 仍然运行在软件上，通过 HDL 仿真工具提供的外部接口实现软硬件间的数据交互，进行模块 Slave 和模块 Master 的联合仿真验证，一旦仿真通过，把模块 Master 和模块 Slave 都放入硬件中进行加速仿真验证，这时两个模块的联合仿真时间将大大缩短。

本文描述的加速仿真技术实现框图如图 3 所示。DUT (Design Under Test) 由可综合的 Verilog HDL 语言设计完成。DUT 综合实现后，下载到现场可编程门阵列 (FPGA Field Programmable Gate Array) 中进行加速仿真验证。运行在 HDL 仿真器上的测试文件 TestBench 给 DUT 发送测试激励并响应输出信息，FPGA 与 HDL 仿真器间的信息交换由仿真器提供的 Verilog 编程语言接口 (PLI: Programming Language Interface)^[6] 来实现。Verilog PLI 为 Verilog 代码调用 C 语言编写的函数提供了一种机制，它提供了 C 语言动态链接程序与仿真器的接口，可以实现 C 语言和 Verilog 语言的协同仿

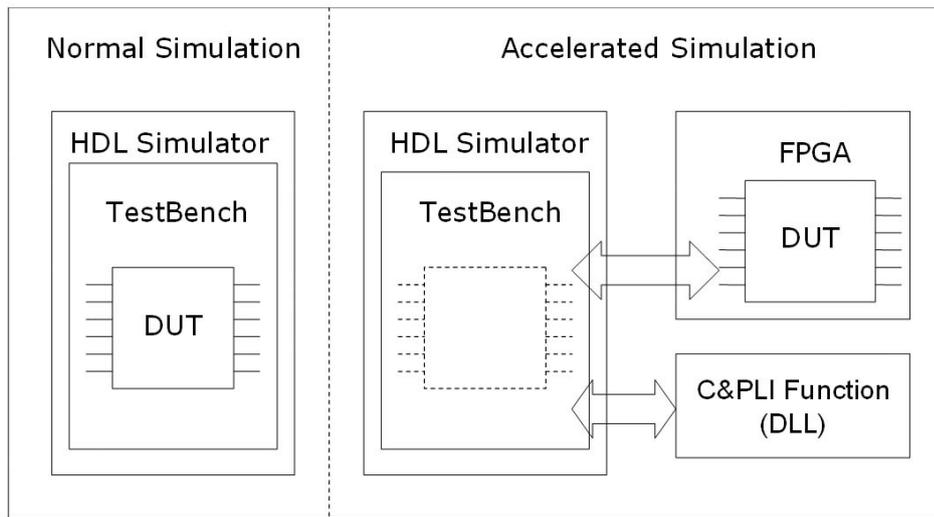


图3 加速仿真

起实现对 HDL 设计文件实现协同仿真。同时,相对于大多数的 HDL 仿真软件来说,ModelSim 在仿真速度上也有明显优势,并且它支持众多的 FPGA 厂家库,是设计工程师做 FPGA 设计的 RTL 级和门级电路仿真的首选。

硬件平台主要以 FPGA 为核心,采用了一片 Xilinx 公司推出的 Virtex-5 XC5VSX95T -

真。由于 C 语言在过程控制方面比 Verilog 语言有优势,可以用 C 程序来产生测试激励和读取信号的值。以 Windows 平台为例,用户通过运用 C 语言和 Verilog PLI 编写接口函数,编译代码并生成动态链接库(DLL:Dynamic Link Library),然后在由 Verilog 语言编写的 TestBench 中调用这些函数。在执行 TestBench 文件进行仿真时,TestBench 中的 C 函数一旦链接成功,C 函数将详细信息传递给 HDL 仿真器,执行 C 函数就可以像仿真 Verilog 代码一样进行仿真。这样,设计工程师利用 Verilog PLI 接口创建自己的系统调用任务和系统函数,就可以通过 C 语言编程对 DUT 进行辅助仿真,达到 Verilog 语法所不能实现的功能。

3 仿真实例

软硬件协同加速仿真平台框图如图 4 所示。我们使用的主机配置为 2.66GHz Intel Core2 处理器和 2GB 内存。软件平台 HDL 仿真器运行于主机上,完成对仿真过程的控制和检测。HDL 仿真器采用 Mentor Graphics 子公司 Model Tech 公司出品的 ModelSim 软件,它全面支持 VHDL 和 Verilog 语言的 IEEE 标准,可以实现 VHDL、Verilog 以及 VHDL-Verilog 混合设计的仿真,还能够与 C 语言一

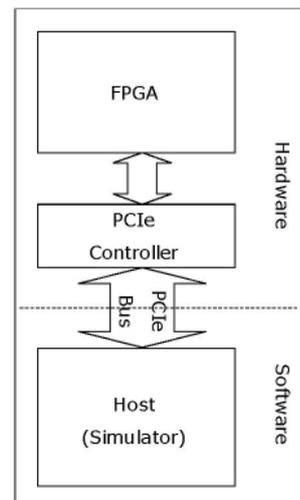


图4 软硬件协同加速仿真平台框图

起实现 HDL 设计文件实现协同仿真。同时,相对于大多数的 HDL 仿真软件来说,ModelSim 在仿真速度上也有明显优势,并且它支持众多的 FPGA 厂家库,是设计工程师做 FPGA 设计的 RTL 级和门级电路仿真的首选。

FF1136 芯片,它内部有丰富的逻辑资源,包括 14720 片 Slices,8784Kb 的 BlockRam,640 片 DSP 48E Slices,16 个 GTP 收发器,640 个可配置 I/O 管脚。此外,该芯片内嵌了一个 PCI-express Endpoint Block 硬核。Xilinx 公司提供的 IP 核 endpoint Block Plus for PCIe[7]解决方案适用于 Virtex-5 SXT FPGA 架构,该 IP 核例化了 Virtex-5 SXT 器件中内嵌的 Virtex-5 Integrated Block for PCI Express,为实现单片可配置 PCI-express 总线解决方案提供了可能。PCI-express 总线作为下一代高性能 I/O 互联技术和标准的局域 I/O 总线,将广泛应用于未来各种计算

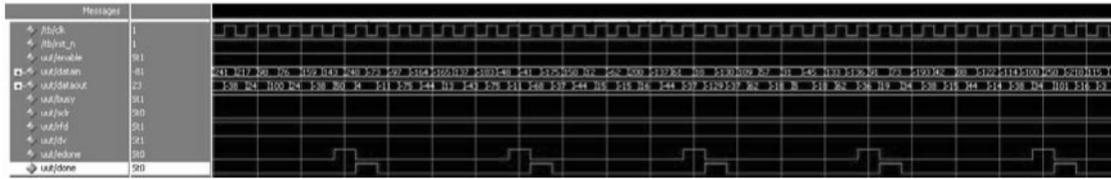


图 5 IFFT 模块仿真波形

机平台。

在实验中，我们选用 Xilinx 公司的 IP 核 FFT (Fast Fourier Transform) v6.0 [8] 模块作为 DUT, 该 FFT 核配置成 Pipelined Streaming I/O 的方式，它可以实现对任意间隔或者连续数据帧的处理。FFT 模块在 ModelSim 软件上运行的仿真结果波形如图 5 所示。

我们分别对 DUT 进行纯软件仿真和软硬件协同加速仿真测试，实验结果如图 6 所示。仿真结果与主机配置、设计的复杂度以及仿真时钟周期数有关。

4 结论

本文利用硬件加速的思想，提出了基于 FPGA 的软硬件协同加速仿真技术。用可综合的 Verilog 语言编写的设计测试文件(DUT)经综合实现后，下载到 FPGA 中，TestBench 仍然运行在主机的仿真器 ModelSim 软件上，通过 Verilog 编程语言接口(PLI)进行软硬件间的数据交互，从而实现了对 DUT 的软硬件协同加速仿真。实验结果表明，相对于纯软件仿真，运用软硬件协同加速仿真技术仿真速度提高了 30 倍，这大大缩短了仿真时间，从而达到缩短设计周期的目的。

参考文献

- [1] J. Yim, Y. Hwang, C. Park, H. Choi, W. Yang. "A C-Based RTL Design Verification Methodology for Complex Microprocessor". In Design Automation Conference, 1997.
- [2] S. Liao, S. Tjian, R. Gupta, "An Efficient

仿真时钟 周期数	仿真实际时间(s)		加速 倍数
	软件(后仿)	硬件加速	
512	23.33	0.766	30.46
1024	48	1.579	30.4
1536	71.61	2.421	29.59
2048	96.92	3.235	29.96

图 6 实验结果对比

Implementation of Reactivity for modeling hardware in the scenic Environment, In Design Automation Conference, 1997.

[3] Eric S. Chung, Eriko Nurvitadhi, James C. Hoe, Babak Falsafi, Ken Mai, "FPGA-accelerated Hybrid Functional Simulation", Computer Architecture Lab at Carnegie Mellon (CALCM) Technical Report, February 2007.

[4] Sangjun Yang, Heejun Shim, Wooseung Yang, Chong-Min Kyung, "A new RTL debugging methodology in FPGA-based verification platform", IEEE Asia-Pacific Conference on ASIC, 2004.

[5] Srihari Cadambi, Chandra S Mulpuri, Pranav N Ashar, "A Fast, Inexpensive and Scalable Hardware Acceleration Technique for Functional Simulation", Design Automation Conference, 2002.

[6] The Verilog PLI Handbook, Second Edition, Stuart Sutherland, Kluwer Academic Publishers, Boston, Massachusetts, copyright 2002.

[7] Xilinx, LogiCORE? IP Endpoint Block Plus v1.9 for PCI Express User Guide, September 2008.

[8] Xilinx, Fast Fourier Transform v6.0 Datasheet, September 2008.