

基于 MIPS 内核的 SoC 软硬件协同仿真

王江, 刘佩林, 陈颖琪

(上海交通大学图像通信与信息处理研究所, 上海 200030)

摘要: 针对基于 MIPS 系列处理器内核的高清电视解码 SoC, 构建了一个软硬件协同仿真环境。连接 MIPS 处理器内核的 VMC 模型和 SoC 的 RTL 模型, 利用 VMC 模型支持 MIPS 指令集的特性运行测试汇编程序, 实现了 SoC 软硬件的同步调试, 有效地提高了系统验证的效率。

关键词: VMC; 片上系统; 软硬件协同仿真

Software and Hardware Co-simulation of SoC Based on MIPS Core

WANG Jiang, LIU Peilin, CHEN Yingqi

(Inst. of Image Communication & Information Processing, Shanghai Jiaotong Univ., Shanghai 200030)

【Abstract】 A software and hardware co-simulation environment has been built for the HDTV decoder SoC based on MIPS processor core. Using the VMC's support of MIPS instruction set, and by running the assemble program for test in this environment which connects the VMC model of the MIPS processor core and the RTL model of the HDTV SoC, the software and hardware of SoC synchronously can be debugged, which can lead to a high-efficiency of SoC verification.

【Key words】 VMC; SoC; Software and hardware co-simulation

SoC是当前集成电路发展的主流方向,在SoC整个设计流程中RTL仿真具有重要的意义。传统的嵌入式系统设计,硬件和软件的开发相对独立,只有等硬件完成以后才能够软硬件结合起来进行系统验证,整个系统的开发周期被延长。软硬件协同仿真工具提供了一个集成的软件环境,工程师可以把硬件的RTL级模块和嵌入式软件在同一个验证环境下进行同步调试。这样,软件工程师可以较早地在硬件设计上直接调试软件,硬件工程师可以较早地得到更加真实的输入激励,从而有效地缩短了系统的设计时间^[1]。

在 HDTV 解码 SoC 验证仿真中我们使用了 MIPS 内核的 VMC(Verilog Module Compiler)模型和 NClaunch 仿真工具构建了软硬件协同仿真环境。VMC 模型可以模拟真实 CPU 的全部功能并能够运行 MIPS 的指令集。利用这种功能,根据 SoC 系统中各个功能模块以及整体的测试要求编写汇编程序,运行程序后观察模块中相应寄存器或 SoC 引脚的响应验证其逻辑功能的正确性。这样的软硬件协同仿真可以在流片之前及时发现硬件设计中的问题,同时优化软件的设计。HDTV 解码 SoC 的 RTL 设计使用了 Verilog 和 VHDL 两种不同的语言,因此选用可以进行两种语言混合仿真的 NClaunch 仿真工具。

1 SoC 和 VMC

从结构上来说,SoC是在一块芯片上集成了控制部件(微处理器)、执行部件(I/O接口、微型开关、微机械)和存储部件(存储器或者存储器接口),能够自成体系、独立工作的芯片^[2]。SoC能最大程度满足嵌入式系统的要求,可以减少功耗,减少印制板上部件数和管脚数,有利于板卡的性能改善,减少系统开发商成本,尤其适合数字化产品开

发,如手持设备、信息家电等^[2]。

HDTV解码SoC是高清晰度数字电视音频解码在系统级芯片上的应用。SoC平台具体到HDTV解码系统,就是把 MPEG2 系统层解码(DeMux)、音频解码(AD)、OSD以及接口I/O控制等功能模块都集成在一块系统级芯片上^[3]。由于 SoC 平台内一些主要的 IP,如 TSD、OSD 等功能模块的设计仿真已经完成,因此工作的重点就在于 SoC 总线架构的设计,VMC 仿真的重点也就是 SoC 总线框架部分的验证仿真。

首先介绍 HDTV 解码 SoC 的结构,如图 1 所示。

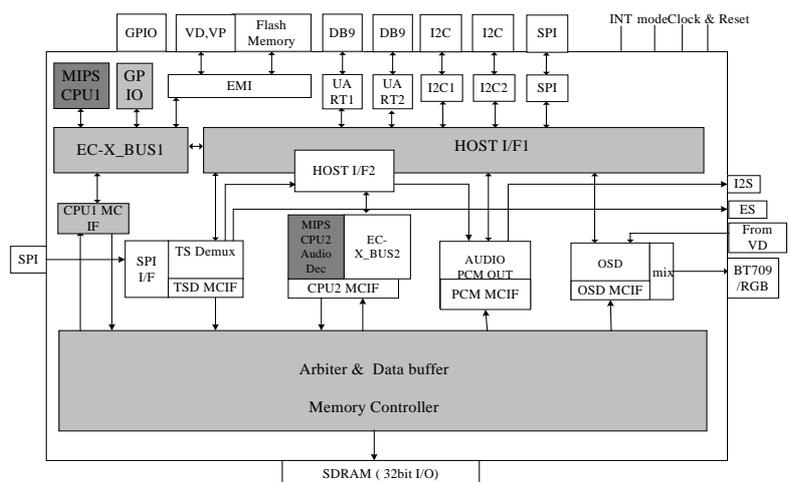


图 1 HDTV 解码 SoC 结构

基金项目: 国家“863”计划基金资助项目(2003AA1Z1070)

作者简介: 王江(1981—),男,硕士生,主研方向:芯片设计验证,数字图像处理;刘佩林,教授、博导;陈颖琪,讲师、博士

收稿日期: 2005-08-30 **E-mail:** trurwong@sjtu.edu.cn

成一个简单的系统进行测试，验证GPIO模块的逻辑功能。GPIO验证结束后，按照图4中模块标号依次加载到整个系统中，每个模块验证结束后加载下一个模块，直到整个系统组建完成，验证无误后整个SoC系统就可以下载到FPGA进行PCB板级FPGA原型仿真验证了。

3 仿真环境和结果

本次仿真是在Solaris操作系统下进行的，使用的仿真工具是Cadence公司的NClaunch(v04.00)。NClaunch是Cadence公司开发的一个逻辑验证的工具，实际上是编译器、仿真器和其它一些工具组成的工具集，兼容VHDL和Verilog或者二者的混合仿真。下面以CPU1和CPU2的联合仿真为例说明验证仿真的过程。

首先根据测试要求使用MIPS的汇编语言写出测试程序：

CPU1的程序：(1) 初始化MC；(2) 在SDRAM中CPU2的启动地址写入CPU2运行的程序；(3) 设置CPU2相关的寄存器并启动CPU2，CPU2启动并运行程序；(4) 等待CPU2运行结束后，读取CPU2写在SDRAM中地址为0xA1A01000的数值，从GPIO输出口输出该数。

CPU2的程序：在SDRAM中地址0xA1A01000的单元写入数值0xA0000000。

使用MIPS的SDE编译器生成机器码，用Xilinx公司的Core Generator工具生成boot_ram，该模块能够存储CPU1的程序机器码。在NClaunch中建立工程，编译链接后运行，在波形窗口中观察对应信号值，如图5所示。

观察图5中GPIO的输出端口uc_out[31:0]，能观察到程序中输出的数值0xA0000000，说明两个CPU的程序均运行完成。在CPU1的EC总线接口（以EB_开头）中EB_RData[63:0]接口上观察到依次执行的每条指令，在EB_WData[63:0]接口上能观察到CPU1写出的数值，根据程序运行的情况同时也可以观察CPU2的EC总线接口，根据其程序运行情况进行调试。

(上接第212页)

4 结论

由图7~图10的合成结果以及表1和表2列出的实验数据可以看出，采用螺旋状匹配搜索的块拼贴合成算法大大加快了纹理合成速度，合成速度比文献[3]中Efros提出的算法平均提高了10倍以上，且样本纹理图越大，本算法合成速度越具有优势；与文献[4]中微软亚洲研究院提出的复杂的加速算法相比，本算法更加简单，易于实现；与基于点匹配的徐晓刚的多种子快速纹理合成算法和WL算法相比，避免了基于点匹配所产生的纹元错位、条带化等现象，并且本文的合成速度比徐晓刚的算法提高了20倍(在一个种子点的情况下)，比WL穷尽的搜索匹配点算法速度平均提高了400倍以上。此外，本算法纹理合成的质量从结构性纹理到一般性纹理都获得了很好的合成效果，但是本算法也继承了块匹配算法的缺点，对一些变化缓慢的纹理合成效果不是很理想。

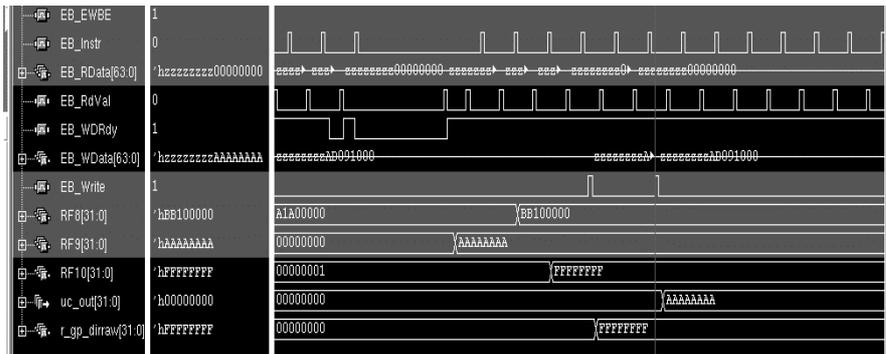


图5 仿真波形窗口

4 结束语

SoC软硬件协同仿真的实质是在一个计算机平台上运行嵌入式CPU的硬件模型。相比早期的纯硬件的逻辑仿真，它能够更早地发现在软件环境中出现的硬件设计问题，也能够根据硬件的响应改进软件的设计。这样可以减少在FPGA仿真中重复修改代码综合下载的时间，也减少了在流片后发现设计缺陷用软件修正的概率。VMC仿真之后的RTL代码在FPGA平台上的仿真验证也证明了VMC仿真的重要性，代码基本没有进行大的修改，有效地节约了时间。

参考文献

- 1 牛风举, 刘元成, 朱明程. 基于IP复用的数字IC设计技术[M]. 北京: 电子工业出版社, 2003-09.
- 2 张志敏. SoC-技术产业与并行发展[N]. 计算机世界报, 2003-12-22.
- 3 胡国荣. 高清晰度电视(HDTV)视频解码芯片设计[J]. 北京广播学院学报(自然科学版), 2000, (32)
- 4 Synopsys, Inc.. SmartModel Library User's Manual[Z]. 2002-07.
- 5 MIPS Technologies, Inc.. MIPS32™ 4Kc™ SMIC 0.18um 8K/8K Hard Core Design Kit Integrator's Guide (Revision 01.00)[Z]. 2004-06.
- 6 MIPS Technologies, Inc.. MIPS32 4K™ Processor Core Family Integrator's Manual (Revision 01.10)[Z]. 2002-09.

参考文献

- 1 Efros A A, Freeman W T. Image Quilting for Texture Synthesis and Transfer[C]. Proceedings of Computer Graphics ACM SIGGRAPH, Los Angeles, California, 2001: 341-346.
- 2 Liang L, Liu C, Xu Y, et al. Real-time Texture Synthesis by Patch-based Sampling[R]. Beijing: Microsoft Research Asia, 2001.
- 3 WEI L Y, Levoy M. Fast Texture Synthesis Using Tree-structured Vector Quantization[C]. Proceedings of SIGGRAPH'00, 2000: 479-488.
- 4 Ashikhmin M. Synthesizing Natural Textures[C]. pnoc. of Symposium on Interactive 3D Graphics, 2001: 217-226.
- 5 徐晓刚. 纹理合成技术研究[R]. 浙江大学博士后研究报告, 2001.