

交换控制电路功能验证平台设计

张健, 黄蓓, 王玉艳

(华东计算技术研究所, 上海 200233)

摘要: 针对超大规模集成电路设计验证的复杂性, 以多口以太网交换控制电路为验证对象, 分析了高级验证语言的特点, 研究了验证策略, 设计了验证平台, 实现了复杂电路的功能验证。验证结果表明, 采用高级验证语言构建验证平台可有效地缩短验证周期、降低验证成本和提高验证质量。

关键词: 功能验证; 交换控制电路; 验证语言; 验证对象

Function Verification Platform Design for Switch Controller

ZHANG Jian, HUANG Bei, WANG Yuyan

(East-China Institute of Computer Technology, Shanghai 200233)

【Abstract】 Regarding the high verification's complexity of VLSI. This paper refers to a multi-port switch controller as a design under verification, analyzes the feature of high verification language, researches the strategy of verification, designs a verification testbench and accomplishes functional verification of the switch controller. The benefit of using the high verification language to build a verification testbench is of shorter verification cycle, less cost of verification, higher quality of verification.

【Keyword】 Functional verification; Switch controller; Verification language; DUV

随着微电子技术的进步, 单个芯片集成的元件数可达数千万甚至上亿, 制造能力的提高允许设计工程师在单芯片上实现复杂系统, 但集成电路设计规模增加使验证工作量急剧上升。2003年国际半导体技术路线图(International Technology Roadmap for Semiconductor, ITRS)指出, 验证已经成为集成电路设计流程中开销最大的工作。目前, 有些集成电路设计项目的验证工程师人数超过了设计工程师。

验证工作量增加的原因如下: (1)设计规模正如摩尔定律所指出的呈指数性增长。如果用设计中的状态数目来衡量功能的复杂程度, 则设计的功能复杂度随设计规模呈指数性增长。如此惊人的增长速度, 对验证提出了极大挑战; (2)历史上对设计流程中的其它环节(如逻辑综合、布局布线和测试产生等)关注颇多, 而对验证重视不够, 使得验证成为目前集成电路设计的瓶颈^[1]。

为了缓解集成电路设计中日益突出的矛盾, 推出了OpenVera和E语言等多种高级验证语言, 目的是简化构建验证平台、提高验证效率和缩短验证周期。

1 验证语言介绍

验证语言是验证工程师开发验证平台和生成验证用例的工具, 目前使用的验证语言有VHDL、Verilog HDL、OpenVera和E语言。

VHDL和Verilog HDL是广泛使用的硬件描述语言, 具有语法简单、可读性好和描述能力强的特点, 适合对系统进行行为级和寄存器传输级(Register Transfer Level, RTL)建模。将它们作为描述验证平台的语言, 可以与设计对象实现无缝连接, 使用同一个编译器和仿真器。但是VHDL和Verilog HDL没有描述复杂行为的高级数据结构, 也不支持面向对象编程, 不适合描述过程和算法。

OpenVera语言是Synopsys公司推出的一种高级验证语言, 提供了与C/C++相似的语法, 验证工程师容易掌握。它具有功能如下: (1)支持验证用例自动生成, 减轻了验证工程师编写验证用例的工作量; (2)支持结果自动检查功能, 省略了验证工程师对大量输出波形的分析工作。

OpenVera语言具有如下特点^[2]:

- (1)提供高级数据结构;
- (2)提供与RTL代码及硬件的接口;
- (3)支持面向对象编程;
- (4)激励自动随机产生;
- (5)结果自动比对;
- (6)支持硬件并行性。

E语言是Versity公司推出的高级验证语言, 具有较强的验证用例自动生成和验证结果分析能力, 便于提高验证覆盖率。但它的语法与C/C++有较大的区别, 验证工程师需要花费更多的时间去熟悉和掌握。

2 验证对象描述

本文的验证对象是多口以太网交换控制电路。本电路内置了8个符合IEEE 802.3协议的10/100M自适应物理层收发器(PHY), 9个支持全双工、符合802.3协议的介质访问控制器(MAC), 通过3块芯片级联可扩展成24+3个端口。

根据交换控制电路的功能和结构, 需要验证的内容有接口时序、单播转发、广播转发、CPU端口收发包、基于端口的虚拟网、生成树协议、端口优先级、MAC地址学习和老化、MAC地址表深度、流量控制、QoS(Quality of Service)、

作者简介: 张健(1977—), 男, 硕士生, 主研方向: 计算机系统结构, 数字系统设计; 黄蓓, 硕士生; 王玉艳, 高工

收稿日期: 2005-09-01 **E-mail:** jzhangsun@hotmail.com

MIB(Management Information Base)计数器、链路聚合、端口镜像、非交叉线速流量转发、全交叉线速流量转发和数据包容错等。

交换控制电路的结构如图 1 所示。

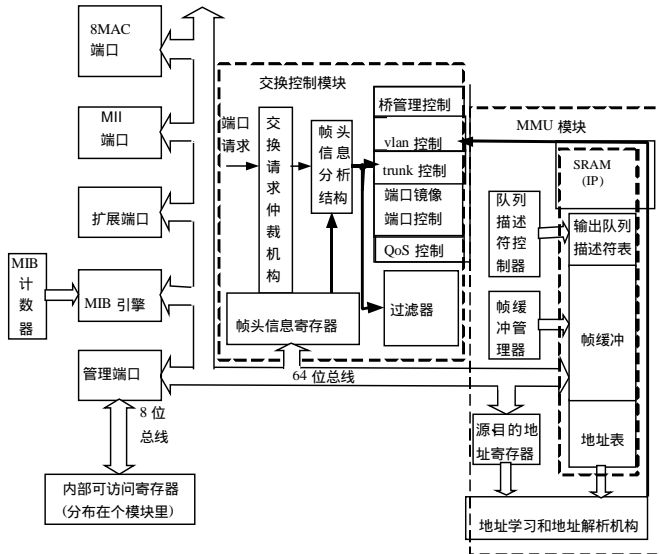


图 1 交换控制电路结构

3 验证平台设计

集成电路的简易验证流程^[2]如图 2 所示。整个验证流程分为验证平台设计、平台实现和调试和提高验证覆盖率 3 个阶段。验证平台设计是验证流程中最重要的部分，直接影响平台的实现和维护。验证平台设计包括确定验证策略和平台结构设计。

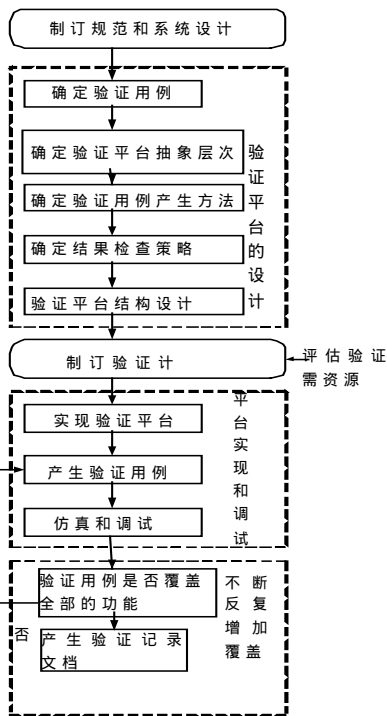


图 2 验证流程

3.1 验证策略研究

为使验证用例覆盖电路的全部功能和提高验证效率，需要进行验证策略的研究。根据交换控制电路的结构和预期的功能，研究内容涉及以下 3 方面^[2]：

(1) 确定验证用例

1) 确定验证用例实际是划分验证空间。有以下 3 种划分方法：

根据功能特性划分，交换控制电路的验证用例有单播转发、CPU 端口收发包、基于端口的 VLAN、支持生成树协议、优先级功能、地址学习、解析和老化、MAC 地址学习能力、流量控制、端口统计计数器、链路聚合和端口镜像等；

根据接口划分，交换控制电路的验证用例有 10/100M 自适应 PHY 接口、MII 接口、SPI 接口和扩展接口等；

根据边际状况划分，涉及的边际状况包括规范边际状况和设计边际状况。规范边际状况是规范规定的边际行为，例如，连续的 1 536B 帧(最长帧)或连续的 64B 帧(最短帧)；设计边际状况是与设计方法相关的边际行为，例如，先进先出(First-In First-Out, FIFO)缓存器深度引起的上溢和下溢。边际状况是一般验证用例难以覆盖到的功能，需要设计专门的验证用例进行验证。

2) 根据功能特性、接口和边际状况，可以找到所需的全部验证用例。

(2) 确定基本数据单位

确定基本数据单位就是明确验证平台处理的数据单位是比特、字节还是数据帧。交换控制电路处理的基本数据单位是以太网帧，由目的地址、源地址、帧类型/长度、数据部分和循环冗余校验码组成，长度为 64B~1 536B^[3]。交换控制电路验证平台处理的基本数据单位是数据帧(以太网帧)。

(3) 确定激励产生和结果比对策略

激励产生方法通常有随机、受控(为验证某一特定功能而特定产生)和受控随机(把随机约束到某一范围内)3 种方法。一个复杂系统的验证需要同时采用这 3 种方法。

例如，交换控制电路验证平台的大部分验证用例需要随机长度帧(64B ~1 536B)；针对 MAC 接口接收功能验证需要受控长度帧(最短 64B，最长 1 536B)；针对交换电路处理能力验证需要受控随机长度帧(64 B 占 10%，1 536 B 占有 90%)。

结果比对策略通常有如下 3 种：

1) 实时自核对技术。实时地把预期的响应与 DUV(design under verification)的响应比对；

2) 参考模型。一个用高级语言(如 C/C++)描述的 DUV 模型。实时参考模型比对是实时地把 DUV 响应与参考模型响应比对；

3) 事后比对。仿真结束后，再进行比对。输出结果形成文件，用文本处理语言(perl 或 C/C++)处理结果文件。

在交换控制电路验证平台中，采取的比对策略是实时自核对技术，比对结果正确的帧可以从缓存中删除，不占用缓存资源。错误的帧留在缓存中，仿真结束后，将错误帧的内容输出到文件，以备查错。

3.2 验证平台结构设计

验证平台结构如图 3 所示。

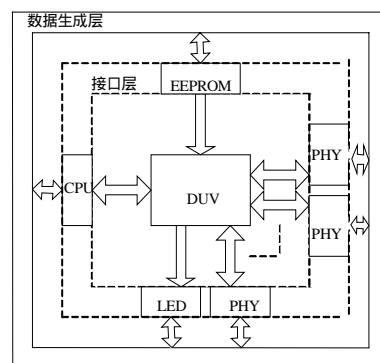


图 3 验证平台结构

交换控制电路验证平台处理的基本数据单位是以太网帧，整个验证平台功能如下：

(1) 数据生成层功能

1)产生验证用例。根据规范要求产生数据帧;把数据帧分发到指定的某个PHY类或CPU类,由指定端口的接口层驱动DUV。

2)产生比对模板数据。比对模板数据是接口层进行数据包比对的依据,根据电路系统设计方案中的功能详细说明,产生期望的数据包比对模板。

3)其它一些辅助功能,包括运行记录和结果分析。

(2)接口层功能

1)按规范要求驱动或接收(监视)接口信号。如果验证对象驱动输出的关系不符合规范要求,接口层就会认为验证对象的代码设计有错误。交换控制电路有10/100M自适应PHY接口、MII接口、SPI接口和扩展接口,由于扩展接口用于3块芯片级联,当作内部信号,不在接口层处理。如图3所示,用PHY类实现10/100M自适应PHY接口和MII接口,用CPU类实现SPI接口。PHY类和CPU类构成验证平台的接口层,实现验证用例与DUV的接口关系。

2)输出数据(结果)自动比对。数据比对的基本单位是帧,数据比对在接口层的PHY类和CPU类中实时自动完成,PHY类和CPU类把比对结果保存到文件,仿真结束用它进行结果检查。

3.3 验证平台实现

鉴于交换控制电路端口多、所支持的功能复杂以及现有验证语言的特点,选用OpenVera语言为交换控制电路验证平台的设计语言。根据验证策略和验证平台结构设计,验证平台OpenVera程序框架如下:

```
// 验证程序中所需的用户文件
// 主程序开始
program test{
// 定义和初始化9个phy(8个10/100M PHY + 1个MII)
// 定义和初始化cpu
// 复位 DUV
reset ();
// 调用 CPU 类中的函数配置与验证用例相关的寄存器
configure_register (cpuobj);
// 发送数据, send_frame()是数据生成层的主要函数,是完成
//验证用例的主体;
send_frame (phyobj, cpuobj);
{ // 生成数据帧
ethernet_frame ethernet_packet[i];
for(; ; // i 个帧 {
// 通过 phy 或 cpu 端口发送帧,src_portid 表示发帧端口号,参
//数 data, length 表示数据帧长度
phyobj[src_portid]. send_frame (ethernet_packet. data[i][],
ethernet_packet. length[].);
or
cpuobj. send_frame(ethernet_packet. data[i][], ethernet_packet.
length[].);
```

(上接第 236 页)

参考文献

- 1 李维是,郭强.液晶显示应用技术[M].北京:电子工业出版社,2000-03.
- 2 郭振东,常昌远,吴金等.STN-LCD 显示驱动技术[J].电子器件,2001,24(8):339-345.
- 3 Han R S. Process for Producing Shaded Color Images Using Dithering

```
// 在目的 phy 端口或 cpu 端口处设置比对模板,注意:此处只
//设置比对模板,具体的比对由 phy 或 cpu 自动完成
```

```
phyobj[dest_portid]. set_check_packet
(ethernet_packet. data[i][], ethernet_packet. length[].);
or
cpuobj.set_check_packet
(ethernet_packet. data[i][], ethernet_packet. length[].);
// 所有帧发送结束
// 调用 phy 或 cpu 中的比对结果进行分析
result(phyobj, cpuobj);}
```

4 验证效果评价

本文设计的验证平台已应用于多口以太网交换控制电路设计验证。根据验证流程,由平台自动生成验证用例,调用NC-Verilog 仿真和调试,记录和分析验证结果文件。采用本验证平台达到的效果如下:

(1)验证用例生成简单。只需对帧源地址、帧目的地址、帧长和帧错误类型(CRC 错、超长帧、超短帧和对齐错等)进行设置,用例由平台自动生成;

(2)验证用例重用性好。对不同功能点验证,只需重新配置寄存器和设置比对模板;

(3)结果检查简单。仿真结束后,查看结果文件即可知设计正确性,不需要在仿真过程中查看波形;

(4)验证质量高。相对于 Verilog HDL 验证平台,易于提高验证覆盖率,便于故障定位。

5 结束语

复杂电路设计验证必须采用高效的验证方法,以缩短验证周期、加快设计进度和提高设计质量。集成电路设计过程复杂、步骤多,为防止设计错误,需要在多个环节进行设计验证,不能仅靠单一的验证方法和验证工具。除本文介绍的适用于集成电路前端设计的功能验证方法,还有静态验证(静态时序分析、形式验证、断言验证和属性检验)和动态验证(后仿真和FPGA原型验证)等专用的工具和方法。随着集成电路设计复杂度的增加,验证工作量会越来越大,验证工作也会越来越重要。

参考文献

- 1 李晓维,吕涛,李光辉等.集成电路设计验证[EB/OL]. http://jct.ict.ac.cn/downloads/xsqy/qy1601.pdf, 2001.
- 2 Faisal I H, Knizar A. K. The Art of Verification with VERA [M]. California: Verification Central Fremont, 2001
- 3 IEEE Std. 802.3 Carrier Sense Multiple Access with Collision Detection Access Method and Physical Layer Specifications[S]. IEEE 802.3x, 2002.

Techniques[P]. USA: 5757347, 1998-05-26.

- 4 Lei Jianming, Zou Xuechen. Dynamic Dithering Algorithm and Frame Rate Control Technique for Liquid Crystal Display Controller[C]. Proceedings of the 5th International Conference on ASIC, 2003-10, 2: 740-743.
- 5 Synopsys Inc. Design Ware AHB Verification IP Databook[Z]. 2004-03.