



国外电子信息经典教材

A Practical Guide for
SystemVerilog Assertions

SystemVerilog Assertions 应用指南

(美) Srikanth Vijayaraghavan
Meyyappan Ramanathan
陈俊杰

编著
等译



清华大学出版社

EISBN: 0-387-26049-8

A Practical Guide for SystemVerilog Assertions

Srikanth Vijayaraghavan, Meyyappan Ramanathan

Copyright©2005 by Springer Press Ltd.

Authorized translation from the English language edition published by Springer Ltd.

All Rights Reserved. For sale in the People's Republic of China only.

Chinese simplified language edition published by Tsinghua University Press.

本书中文简体字版由施普林格出版公司授权清华大学出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字：01-2006-4712

版权所有，翻印必究。举报电话：010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

本书防伪标签采用特殊防伪技术，用户可通过在图案表面涂抹清水，图案消失，水干后图案复现，或将表面膜揭下，放在白纸上用彩笔涂抹，图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

SystemVerilog Assertions 应用指南/(美)维加亚拉哈文(Vijayaraghavan,S.),(美)拉门那斯(Ramanathan, M.)著;陈俊杰 等译.

—北京:清华大学出版社,2006.10

书名原文:A Practical Guide for SystemVerilog Assertions

ISBN 7-302-13441-3

I. S… II. ①维…②拉…③陈… III. 电子电路—电路设计:计算机辅助设计 IV. TN702

中国版本图书馆 CIP 数据核字(2006)第 081310 号

出版者:清华大学出版社

<http://www.tup.com.cn>

社总机:010-62770175

组稿编辑:王军

封面设计:孔祥丰

印刷者:北京牛山世兴印刷厂

装订者:北京市密云县京文制本装订厂

发行者:新华书店总店北京发行所

开本:153×230 印张:20.5 字数:294千字

版次:2006年10月第1版 2006年10月第1次印刷

书号:ISBN 7-302-13441-3/TP·8442

印数:1~3000

定 价:39.80元(含光盘)

地 址:北京清华大学学研大厦

邮 编:100084

客户服务:010-62776969

文稿编辑:徐燕萍

版式设计:孔祥丰

教师信息反馈表

为了更好地为您提供相关服务,使您的教学工作更加轻松高效,也使您的教学成果能够得以推广,请按下表填写您的基本信息、意见和要求。然后剪下寄到:北京清华大学出版社第五事业部(邮编 100084);您也可以把意见反馈到 fwkbook@tup.tsinghua.edu.cn。邮购咨询电话:010-62770177/75 转 3505。您的支持将对我们的工作提供有益的帮助,谢谢合作!

一、基本信息:

姓名: _____ 所在院校: _____

职称/职务: _____ 系别专业: _____

联系电话: _____ E-mail: _____

通信地址: _____ 邮编: _____

二、您所授课程的学术领域(或课程名):

电子信息与电气工程 计算机软件及理论 计算机网络 操作系统

计算机科学理论 数据库 信息系统 信息安全

计算机系统结构 计算机语言与程序设计

计算机基础理论 图形学和 CAD

多媒体技术 计算机通信技术 人工智能

其他: _____

三、请选择您所需新书信息的学科,我们将根据您选择的内容定期给您发送新书信息。

电子信息 计算机语言与程序设计 计算机网络

计算机原理与硬件技术 计算机操作系统

图形图像、多媒体技术 计算机辅助设计 办公自动化 软件工程

软件理论与系统 数据库 信息安全

人工智能 信息系统与工程 计算机在其他领域中的应用

其他: _____

四、您希望我们就本书提供哪些服务?

五、近期您有何教材或专著出版计划?

六、您对清华大学出版社教材营销工作有何评价及建议?

我们对您的反馈表示真诚的谢意!

出版说明

电子信息产业是一项新兴的高科技产业，有“朝阳产业”之称，有着巨大的潜力和广阔的发展前景。近年来，我国电子信息产业的飞速发展，大大推动了对电子信息类人才的需求，迫切需要我国的高等院校能够培养出大批符合企业要求的电子信息类人才。

教育与教材的关系始终是密不可分的，教材的合适与否会直接影响到培养人才的质量好坏。虽然目前我国高校中现行的电子信息类教材曾经对我国电子信息类人才的培养做出了非常重要的贡献，但是确实普遍存着一些问题，如“课程系统老化”、“内容落伍”、“惯性大，更新速度慢”、“针对性差”、“缺乏原创精品”等等，教学内容和课程体系的改革已经成为目前教学改革过程中的当务之急。

基于这种背景，我们决定在国内引进并推出一套“国外电子信息经典教材”，通过系统地研究和借鉴国外一流大学的相关教材，为我国高校的课程改革和国际化教学进程提供参考和推动作用。

为了组织该套教材的出版，我们在国内聘请了一批资深的专家和教授，共同成立了教材编审委员会。由编委会结合目前国内高校电子信息类专业的课程体系和教学内容，从 McGraw-Hill Education、Thomson Learning、John Wiley & Sons 和 Springer 等一批国际著名的教育出版集团，精选出一套“国外电子信息经典教材”。列选的每本教材都经过了国内相应领域的资深专家推荐和审读，对于一些基础类的专业课程，我们列选了多种不同体系、不同风格和不同层次的教材，以供不同要求和不同学时的同类课程使用。为了确保该套教材的质量，我们聘请了高校相应专业的资深教师和相应领域的专家担纲译者，加强了该套教材各个出版环节的编审力量和质量控制。另外，为了丰富国内的教学资源，

我们在引进教材的同时也积极引进了教材配套的教学资源。

该套教材的读者对象为电子信息与电气工程类专业的本科生，同时兼顾相关工程学科各专业的本科生或研究生。该套教材既可作为相应课程的教材或教学参考书，也适于相应技术领域的工程师和技术人员参考或自学。

尽管我们作了种种努力，但该套教材书目选择的恰当性，内容的合理性，都还有待于通过教学实践来检验。首先感谢选用该套教材的广大教生对我们的支持，同时期待广大读者积极为该套教材提出意见或建议。

清华大学出版社

序

在 20 世纪 80 年代中期, Gateway 设计自动化公司(Gateway Design Automation Inc.)发明了 Verilog, 那时集成电路的设计过程与现在有很大的不同。Verilog 的作用和功能从一开始就不断演变, 直至发展成为今天的 SystemVerilog。

ASIC(专用集成电路)的功能验证逐渐成为一项困难的事情, 至于到底有多难一直处在推测和争论中。2001 年, EE Times 引用了 Cisco 系统工程副总裁 Andreas Bechtolsheim 对验证的困难程度所作的一个较高的估计:

“设计验证所费的时间仍然占整个芯片开发时间的 80%。”

然而在 2004 年, EE Times 对参加设计自动化会议(DAC)的 662 名专业人士做了一个民意测验, 得出功能验证占整个集成设计过程的 22%这样一个结论。

22%和 80%的巨大差距显示了人们对验证和集成电路设计与开发的其他阶段的认识有多么模糊。许多验证工作是设计工程师自己完成的, 但仍然是验证过程的一部分, 而这部分的验证工作同样可以受益于辅助专业验证工程师的验证工具。

不管实际的比例占多少(假设能够精确测量出来), 功能验证依然是集成电路设计的一个重要组成部分。验证也是硅片一次成功的关键一步。即使当掩膜的成本超过了 100 万美元, 但相对于由于返工而耽搁几周导致的错失良机来说, 这些钱是微不足道的。任何工具, 只要能减少验证费用和提高尽早流片的可能性, 都应该积极采纳。

断言作为软件开发的一部分已经很多年了, 而基于断言的验证(Assersion-Based Verification, ABV)才刚刚流行起来。奇怪的是, 硬件描述过程与软件设计过程在某种程度上越来越相似。然而, 在硬件设计中我们想要断言和检验的属性, 与软件世界里的属性有着根本上的不同。

硬件和软件编程模式的差异是时序。硬件语言如: Verilog,

有表示时序的机制，而过程式编程语言(C、C++、Java 等)却没有。所以用不着奇怪在断言的软件方法中没有定义处理时序的方式。

SystemVerilog 是 Gateway 公司的 Verilog 的最新后继者，包括了 SystemVerilog 断言(SystemVerilog Assersions, SVA)——允许工程师把 ABV 应用到他们设计中的一系列工具。SVA 有很丰富的语法，可以在序列、属性和(完全的)断言中支持时间概念。

有了 SVA，设计和验证工程师可以把对硬件设计的期望行为进行编码，可以创建对总线协议的详细检验。这些(相对)简洁的描述能用于模拟、形式验证和作为设计的附加文档。

SVA 将会在集成电路的设计和验证的方法上有很重大的影响，这一点是很明了的。学习了 SVA 的语法及如何把它应用于您的设计中，您将受益匪浅。本书将会帮助您学习和应用 SVA。书中举了一些例子，包括 PCI(外设部件互连)总线协议，主要用来说明如何编写 SVA 及其模拟结果。

本书中 SVA 语言的详细例子对理解基于时序的断言概念和语法很有帮助。这些例子使这本书更加名副其实。它是所有 SystemVerilog 设计和验证工程师的必备用书。

最后，我的女儿 Stevie，声称没有人会读一本书的序。如果读完了本书的序，请您发个简短的电子邮件告诉她。她的 e-mail: steviechayut@gmail.com。

谢谢！

Ira Chayut
验证架构师，Nvidia 公司

中文版序

超大规模集成电路随着系统芯片(SoC)时代的到来其功能日益复杂,规模也日益庞大,无论是芯片设计工程师还是芯片验证工程师都面临着前所未有的挑战。设计工程师和验证工程师需要借助更为有效的高级设计语言和验证技术来实现 SoC 的正确设计。

SystemVerilog 作为一种硬件描述和验证语言 HDVL(Hardware Description And Verification Language), 是随着技术发展和市场需求应运而生的。SystemVerilog 为 Verilog-2001 标准提供了一系列的扩展,这些扩展有效地提升了硬件设计、模拟验证和测试平台的整体效率,显著地降低了芯片设计的风险,有助于缩短产品的上市时间。SystemVerilog 于 2005 年 11 月成为电子设计的一种新标准语言(SystemVerilog, IEEE Std 1800-2005)。

在当今流行的验证技术中,基于断言的验证(Assertion-Based Verification, ABV)是一种很有价值的主流验证技术。作为 SystemVerilog 的重要组成部分, SystemVerilog Assertion 提供了丰富的断言指令,使得设计工程师和验证工程师可以快速地对复杂设计验证行为进行定义。

近年来,我国大陆地区的半导体行业发展迅速,从芯片的设计到制造、测试、封装,国内已基本形成了完整的 IC 产业链。但是,我们同时也应看到,与其他发达国家和地区相比,无论是自主创新、技术开发还是人才培养方面都存在着较大的差距。我们应该群策群力,积极支持通过多种方式引进国外先进的技术出版物和书籍。《SystemVerilog Assertions 应用指南》的翻译出版就是引进集成电路设计优秀技术书籍的一个范例。本书可作为电子通讯专业高年级本科生和研究生的教学用书,也可作为设计工程师和验证工程师的应用参考书。

国家 863 集成电路设计专项总体组组长
国际集成电路人才基地专家委员会主任

严晓浪 教授

2006 年 8 月

译者的话

由于研发项目的缘故，有机会比较早地接触 Open Vera Assertion(OVA)以及后来的 SystemVerilog Assertions(SVA)这项新技术。然而几个月以后，研发项目做完，对这项技术还只是“不求甚解”。

2005 年在公司总部 MountainView 参加年度 Offsite 会议，资深市场总监 Steve Smith 把这本由 Springer 出版社刚刚出版的新书介绍给我时，顿时有一种如获珍宝、相见恨晚的感觉！

本书的英文作者 Srikanth Vijayaraghavan 和 Meyyappan Ramanathan 是两位资深应用顾问。他们对于现代数字系统设计流程、先进的验证环境以及不同的 IC 设计特点有着深刻独到的见解。本书的特点是不仅系统地介绍了 SVA 这种硬件验证语言(HVL)的基本语法，而且针对不同类型的 IC 设计深入浅出地介绍了 SVA 的应用。本书分 8 章，其中第 0, 1, 2 章介绍了 ABV(基于断言的验证)方法学、SVA 的语法及用一个实例介绍了 SVA 的应用。第 3, 4, 5, 6, 7 章分别讨论了 SVA 在各种典型设计中的应用。这些典型设计模型包括了：有限状态机(FSM)，数据通道，存储控制器，基于 PCI 局部总线系统和测试平台(testbench)。无论是对刚刚接触断言的新手还是资深设计验证工程师，本书都是案前必备的一本参考书！

几个月艰苦的翻译工作是一种磨练，也是一种提升。国内可以参阅的关于 SVA 的资料非常有限。许多技术词汇的翻译在业内还没有达成共识，有些词汇几乎是第一次翻译。这对于翻译团队来说是一个巨大的挑战。我们常常对如何恰当地翻译一个词汇进行反复的争论和推敲。在翻译审校过程中，发现了几处英文原书的错误，经原作者同意，在中文版中得以纠正。

本书的原序，前言，第 0 章，第 2 章和第 3 章由陈俊杰和马青娇翻译。第 1 章由黄劲楠翻译。第 4 章和第 5 章由叶擎双翻译。第 6 章和第 7 章由杨天翔翻译。陈俊杰负责全书的整理和审校。

本书虽然经过仔细校对，但由于译者水平有限，难免存在错误和疏漏之处，希望读者批评指正。欢迎大家通过 SystemVerilogAssertion@googlegroups.com 和译者联系。

衷心地感谢下列业内资深人士以及国内外教授学者审阅本书并提出了许多宝贵的意见：王欣，Steve Ye，刘敬军，陈海慧，郑昊，唐叔平，谢原等。

衷心地感谢严晓浪教授百忙之中为本书写了中文序。

衷心地感谢公司同仁特别是 Steve Smith, Sadeghian, Chris K, Alessandro Fasan, 潘建岳, 冯成, 鲍志伟, 赵懿等的鼓励和支持。

衷心地感谢 Synopsys 公司和清华大学出版社的支持，本书的中文版才得以如期和读者见面。

最后，衷心感谢所有家人的支持。感谢女儿陈怡冰，你的诞生是一个最美丽的奇迹！

陈俊杰

2006年8月

前 言

2002 年中期，我们的经理给我们发了一个电子邮件，问道：“谁愿意去支持 OVA？”我们从脑子中迸出的第一个想法就是“究竟什么是 OVA？”和其他几个工程师交谈后，我们知道了它是 OPEN VERA 语言的一个子集。OVA 是指“OPEN VERA 断言(Open VERA Assersions, 简写 OVA)”，它是一种描述性的语言，能描述时序上的条件。就如同过去一样，为了满足对技术的渴求，我们同意做 OVA 的支持。在两个月内，我们学习了这种语言，并开始培训客户，在六个月内培训了 200 个左右的客户。客户洪水般涌进教室，给我们留下深刻的印象。我们确信这是验证领域下一件最好的事情。当客户们匆忙接受完培训，他们并没有开发任何 OVA 的代码。这是因为验证技巧和这种语言都是新的。一些工具刚开始支持这些语言结构。没有多少 IP(Intellectual Property)可以使用。很自然，客户并没有我们想像的那么满意。

同时，Synopsys 公司把 OPEN VERA 语言捐献给 Accellera 委员会，使其成为 SystemVerilog 语言的一部分，其他几个公司为 SystemVerilog 语言的形成作了一些贡献。在 DAC 2004，Accellera 委员会把 SystemVerilog 3.1 定为一个标准。断言语言被纳入 SystemVerilog 语言并成为了标准的一部分。这就是通常所说的“SystemVerilog Assertion(SVA)”。我们继续培训客户基于断言的验证，不过现在仅仅教 SVA。我们能清楚地看到客户更习惯于使用预开发的断言库，而不乐意编写定制的断言代码。是什么阻碍了他们？是工具吗？不，工具是现成的。是语言吗？或许，但它如今已是一个标准，所以不应该是它。

经过一番深入的讨论，我们认识到，缺乏例子来演示 SVA 的结构可能是阻碍客户使用这项新技术的原因。比较典型的是缺乏专家指导导致了如此低的采纳率。这时我想到出版一本关于 SVA 的“烹饪书”可能有用——即一本充满例子的书，这本书可以作为指导书，用来教授这种语言。这个项目就是这样启动的。我们

努力把过去两年中在教授这门科目时所学的东西写出来。但是在这个领域还有很多东西需要去学，这本书只是把我们所学到的跟大家分享。

如何阅读这本书

这本书的写作方式可以使工程师快速掌握 SystemVerilog 断言。

第 0、1 和 2 章，可以使您充分了解基础语法和一些通用的模拟技巧。阅读完这三章，读者应该能在他们的设计 / 验证环境中写断言。

第 3、4、5 和 6 章是不同类型的设计的“烹饪书”。读者如果在他们自己的环境里遇到类似的设计可以参考这些章节，以这些章节作为起点开始写断言。这些章节也可以作为指导。

如果您是基于一断言验证的新手，则需要阅读完第 0 章~第 2 章，才能开始其他章节。如果您熟悉 SVA 语言，就可以根据需要进行参考这些章节。

第 0 章——这是关于基于断言的验证(ABV)方法论的白皮书。这一章介绍了 ABV 的方法学和功能覆盖的重要性。

第 1 章——用简单的例子讨论了 SVA 的语法和详细分析了在动态模拟中执行 SVA 结构的过程。包括了模拟波形和事件表以供读者参考。要了解每个 SVA 结构的细节，用户可以参考 SystemVerilog 3.1 a 语言参考手册(LRM)的第 17 章。

第 2 章——用一个实例系统说明 SVA 模拟的方法。主题囊括了协议解析、模拟控制和功能覆盖。

第 3 章——用两个不同的有限状态机(FSM)模型作为例子，举例说明如何用 SVA 验证 FSM。

第 4 章——举例说明用 SVA 验证一个数据通道。用 JPEG 设计的一部分来演示如何用 SVA 验证控制信号和数据。

第 5 章——举例用 SVA 验证一个存储控制器。这个控制器支持不同类型的存储如：SDRAM、SRAM、FLASH 等。

第 6 章——举例用 SVA 验证一个基于 PCI 局部总线的系统。使用了一个 PCI 系统配置的例子，用 SVA 验证不同的 PCI 协议。

第 7 章——用一个测试平台(testbench)的例子验证断言,也讨论了在验证断言的精度背后的理论。

随书附一张光盘。本书中的所有例子都可以用 VCS 2005.06 发行版运行,也包括运行这些例子的脚本范例。VCS 是 Synopsys 公司的注册商标。

致谢

下面的人由于他们对完成本书所做出的巨大贡献,在这里作者对他们表示真诚的谢意:

Anupama Srinivasa, DSP 解决方案架构师, AccelChip 公司;

Jim Kjellsen, Staff 应用顾问, Synopsys 公司;

Juliet Runhaar, 资深应用顾问, Synopsys 公司;

我们同样感谢下面的人,他们参与审阅本书并提供了很多建设性的建议:

Ira Chayut, Bohran Roohipour, Irwan Sie, Ravindra Viswanath, Parag Bhatt, Derrick Lin, Anders Berglund, Steve Smith, Martin Michael, Jayne Scheckla, Rakesh Cheerla, Satish Iyengar

有用的链接

www.systemVerilogforall.com——我们维护的网页,提供关于 SystemVerilog 语言的技巧、例子和讨论。

www.accellera.org——Accellera 委员会的官方网站。可以从这里下载 SystemVerilog LRM。这里还有一些有用的论文和有关最新标准的介绍。

目 录

第 0 章 基于断言的验证	1
第 1 章 SVA 介绍	5
1.1 什么是断言	5
1.2 为什么使用 SystemVerilog 断言(SVA)	6
1.3 SystemVerilog 的调度	8
1.4 SVA 术语	9
1.4.1 并发断言	9
1.4.2 即时断言	10
1.5 建立 SVA 块	11
1.6 一个简单的序列	12
1.7 边沿定义的序列	13
1.8 逻辑关系的序列	15
1.9 序列表达式	15
1.10 时序关系的序列	16
1.11 SVA 中的时钟定义	18
1.12 禁止属性	19
1.13 一个简单的执行块	21
1.14 蕴含操作符	21
1.14.1 交叠蕴含	22
1.14.2 非交叠蕴含	23
1.14.3 后续算子带固定延迟的蕴含	24
1.14.4 使用序列作为先行算子的蕴含	25
1.15 SVA 检验器的时序窗口	27
1.15.1 重叠的时序窗口	29
1.15.2 无限的时序窗口	30

1.16	“ended” 结构	32
1.17	使用参数的 SVA 检验器	35
1.18	使用选择运算符的 SVA 检验器	36
1.19	使用 true 表达式的 SVA 检验器	38
1.20	“\$past” 构造	39
1.21	重复运算符	42
1.21.1	连续重复运算符[*]	43
1.21.2	用于序列的连续重复运算符[*]	44
1.21.3	用于带延迟窗口的序列的连续重复运算符[*]	46
1.21.4	连续运算符[*]和可能性运算符	47
1.21.5	跟随重复运算符[->]	48
1.21.6	非连续重复运算符[=]	50
1.22	“and” 构造	51
1.23	“intersect” 构造	54
1.24	“or” 构造	56
1.25	“first_match” 构造	58
1.26	“throughout” 构造	60
1.27	“within” 构造	61
1.28	内建的系统函数	63
1.29	“disable iff” 构造	65
1.30	使用“intersect” 控制序列的长度	66
1.31	在属性中使用形参	68
1.32	嵌套的蕴含	70
1.33	在蕴含中使用 if/else	71
1.34	SVA 中的多时钟定义	73
1.35	“matched” 构造	75
1.36	“expect” 构造	76
1.37	使用局部变量的 SVA	77
1.38	在序列匹配时调用子程序	79
1.39	将 SVA 与设计连接	81
1.40	SVA 与功能覆盖	83

第 2 章 SVA 模拟方法论	85
2.1 一个被验证的实例系统	85
2.1.1 主控设备	86
2.1.2 中间设备	88
2.1.3 目标设备	90
2.2 块级验证	91
2.2.1 SVA 在设计块中的应用	92
2.2.2 仲裁器的验证	92
2.2.3 模拟中针对仲裁器的 SVA 检验	94
2.2.4 主控设备的验证	96
2.2.5 模拟中针对主控设备的 SVA 检验	98
2.2.6 胶合(Glue)的验证	100
2.2.7 模拟中针对胶合逻辑(glue logic)的 SVA 检验	102
2.2.8 目标设备的验证	104
2.2.9 模拟中针对目标设备的 SVA 检验	106
2.3 系统级验证	108
2.4 功能覆盖	114
2.4.1 实例系统的覆盖率计划	115
2.4.2 功能覆盖小结	124
2.5 用于创建事务日志的 SVA	124
2.6 用于 FPGA 原型测试的 SVA	127
2.7 SVA 模拟方法的小结	131
第 3 章 SVA 在有限状态机中的应用	133
3.1 设计例子——FSM1	134
3.1.1 FSM1 的功能描述	134
3.1.2 FSM1 的 SVA 检验器	139
3.2 设计实例——FSM2	143
3.2.1 FSM2 的功能描述	144
3.2.2 FSM2 的 SVA 检验器	148
3.2.3 有时序窗口协议的 FSM2	155
3.3 在 FSM 中应用 SVA 的小结	159

第 4 章 SVA 用于数据集约型(DATA INTENSIVE)的设计	161
4.1 简单乘法器的检验	161
4.2 设计实例——算术单元	163
4.2.1 WHT 算术	163
4.2.2 WHT 硬件的实现	164
4.2.3 WHT 模块的 SVA 检验器	165
4.3 设计实例——JPEG 的数据通路设计	168
4.3.1 三模块的深入探讨	169
4.3.2 用于 JPEG 设计的 SVA 检验器	172
4.3.3 JPEG 模型的数据检验	176
4.4 数据集约型设计的小结	182
第 5 章 SVA 储存器	183
5.1 存储控制系统实例	183
5.1.1 CPU-AHB 接口操作	183
5.1.2 存储控制器的操作	186
5.2 SDRAM 的验证	189
5.3 SRAM/FLASH 的验证	208
5.4 DDR-SDRAM 的验证	215
5.5 存储器 SVA 的小结	217
第 6 章 SVA 协议接口	219
6.1 PCI 简介	220
6.1.1 一个 PCI 读出事务的实例	222
6.1.2 PCI 写入事务实例	223
6.2 PCI 系统实例	224
6.3 情形 1——主控 DUT 设备	225
6.4 情形 2——目标 DUT 设备	243
6.5 情形 3——系统级断言	259
6.6 用于标准协议的 SVA 小结	263
第 7 章 对检验器的检验	265
7.1 断言验证	266