Technology 实用技术

SystemVerilog 验证

测试平台编写指南

（原书第二版）

〔美〕克里斯·斯皮尔 著
张春 麦宋平 赵益新 译

科学出版社
www.sciencep.com

SystemVerilog 验证

测试平台编写指南

（原书第二版）

〔美〕克里斯·斯皮尔 著
张 春 麦宋平 赵益新 译

科学出版社
北京

图字：01-2009-4623 号

内 容 简 介

本书讲解了 SystemVerilog 语言的工作原理，介绍了类、随机化和功能覆盖率等测试手段和概念，并且在创建测试平台方面提供了很多引导性的建议。本书借助大量的实例说明 SystemVerilog 的各种验证方法，以及如何根据实际的应用情况选择最优的方法达到尽可能高的覆盖率。而且，重点演示了如何使用面向对象编程(OOP)的方法建立由覆盖率驱动并且受约束的基本的随机分层测试平台，此外，还论述了 SystemVerilog 与 C 语言的接口技术。

本书可供具有一定 Verilog 编程基础的电路工程技术人员使用，也可作为高等院校电子类、自动化类、计算机类的学生参考书。

图书在版编目(CIP)数据

SystemVerilog 验证/(美)克里斯·斯皮尔著；张春等译. —北京：科学出版社，2009

ISBN 978-7-03-025306-4

Ⅰ. S… Ⅱ. ①克…②张… Ⅲ. 硬件描述语言-程序设计 Ⅳ. TP312

中国版本图书馆 CIP 数据核字(2009)第 147212 号

责任编辑：赵方青 杨 凯 / 责任制作：董立颖 魏 谨

责任印制：赵德静 / 封面设计：郝晓燕

北京东方科龙图文有限公司 制作

http://www.okbook.com.cn

科学出版社 出版

北京东黄城根北街 16 号

邮政编码：100717

http://www.sciencep.com

北京天时彩色印刷有限公司 印刷

科学出版社发行 各地新华书店经销

*

2009 年 9 月第 一 版 开本：B5(720×1000)

2009 年 9 月第一次印刷 印张：24 1/2

印数：1—4 000 字数：541 000

定 价：55.00 元

(如有印装质量问题，我社负责调换)

***** 译者序

SystemVerilog 语言的出现只有短短几年的时间，目前市面上关于 SystemVerilog 语言的中文书籍并不多见，而且大多都是介绍 SystemVerilog 语言的设计特性。实际上，SystemVerilog 语言除了具有设计特性外，还具有验证及其他诸多方面的特性。“验证”经常被认为是简单的仿真，这当然是一种误解，本书将告诉你其中缘由。

本书主要介绍 SystemVerilog 语言的验证技术，尤其侧重阐述如何使用受约束的随机测试来达到令人满意的覆盖率。原著作者克里斯·斯皮尔(Chris Spear)是一名资深的数字电路工程师，在软件编程方面有很丰富的经验，书中的很多观点和例子就来自于作者平时工作的积累。本书没有深奥的理论，叙述上深入浅出。而且由于作者同时也精通 C++、Verilog 和 Vera 等编程语言，所以书中对于 SystemVerilog 与这些语言之间的差别以及易混淆的地方交代得十分清楚，特别适合 SystemVerilog 的初学者阅读。

本书的翻译过程颇为波折，前后总共持续了一年多的时间。当我们在 2008 年初开始着手翻译工作时，使用的还是本书的第一版。但翻译工作进行到将近一半时，获悉本书的第二版即将发行，于是转为等待翻译第二版。第二版除了章节内容上有所增补以外，原有章节的很多字句也有所改动，只得重新翻译、校对。

本书的翻译具体分工如下：第 6、11 章的翻译由张春负责；前言和第 1、2、3、7、9 章的翻译由麦宋平负责；第 4、5、8、10、12 章的翻译由赵益新负责；全书的审校和最终定稿由张春负责。

衷心感谢清华大学微电子学研究所的王志华教授，他在本书翻译之初就提出了很多具有指导性的意见，并且为翻译工作提供了很多支持。

衷心感谢科学出版社的支持，正是出版社各位编辑的鼓励和督促，以及他们勤勤恳恳的工作，才使得本书的中译本得以如期与读者见面。

由于本书的翻译稿出现第一版和第二版交叉，新词汇又比较多，囿于译者的经验和水平，虽然经过多次仔细的斟酌和校对，仍难免存在不准确和纰漏的地方，请读者不吝批评指正！

译　者

2009 年 8 月

前　言

本书的内容

本书可以作为学习 SystemVerilog 验证语言的初级阶段读物。书中描述了语言的工作原理并且包含了很多例子，这些例子演示了如何使用面向对象编程(OOP)的方法建立一个基本的、由覆盖率驱动并且受约束的随机分层测试平台。本书在创建测试平台方面有很多引导性的建议，能够帮你弄清楚为什么要使用类、随机化和功能覆盖率的概念。一旦你掌握了这门语言，就可以通过参考文献中所列举的方法学方面的书籍来学习关于建立测试平台的更多信息。

本书的目标读者

如果你要创建测试平台，那么本书可以提供必要的内容。如果你只用过 Verilog 或 VHDL 编写测试而现在想学 SystemVerilog 语言，那么本书可以教会你使用新的语言特性。Vera 和 Specman 的用户可以学到如何把一种语言同时用在设计和验证上。如果你读过 SystemVerilog 的语言参考手册(LRM)，就会发现它里面塞满了各种语法，但却没有任何关于结构选择方面的引导。

我像很多客户一样，在以前的职业生涯中，一直都使用 C 和 Verilog 这样的语言来编写测试，所以当面向对象编程(OOP)语言出现以后，就必须一切都从头学起。几乎所有典型的错误我都碰到过，所以我把它们都写下来，这样你就不会再犯同样的错误了。

在读本书之前，你应该能熟练地使用 Verilog-1995。而有关 Verilog-2001、SystemVerilog 设计结构和 SystemVerilog 断言的知识则不是必须的。

第二版新增的内容

相比于 2006 年出版的第一版，《SystemVerilog 验证》的这个新版本有很多改进。

- SystemVerilog 语言参考手册(LRM)2008 年新版本中有很多大大小小的变化。本书尽量把相关的最新信息囊括进来。
- 很多读者向我询问过关于 SystemVerilog 概念上进一步的细节。我把几乎所有的这些交流都集成到本书的扩展解释或代码样例中。从第 2 章开始，几乎每一张图片和每一个例子都是重新编写、修改或者调整过的。在最初的 10 章里，增加了 50 多页的新内容和 70 多个新例子。总共算起来，新版本比旧版本大概增加了 1/3 的篇幅。
- 有读者希望看到更多的样例，尤其是大型的样例。为了顺应这种要求，新版本在第 4 章的末尾增加了一个定向测试平台，在第 11 章中增加了一个完整的受约束的随机测试平台。

• 并不是所有的测试平台都用 SystemVerilog 编写，所以新版本增加了第 12 章，用于说明如何通过直接编程接口把 C 或 C++代码与 SystemVerilog 连接起来。

• 很多工程师喜欢从索引开始阅读，所以本书把索引条目加倍。大家也很喜欢交叉引用，所以本书在这方面也有增加，这样读者就可以根据需要以跳跃的方式阅读。

• 最后，特别感谢那些向我提建议的读者，他们指出了第一版中存在的错误，从低级的语法错误到代码错误，那些问题代码显然是在我刚刚结束从亚洲到波士顿长达 18 小时的飞行之后的那个早上写下来的。新版本经过了很多遍校验和复查，但我必须再次声明，所有错误都应归咎于我。

创建 SystemVerilog 的原因

在 1990 年末，Verilog 硬件描述语言（HDL）成为描述硬件仿真和综合方面应用最广泛的语言。但是，被 IEEE 定为标准的最初两个版本（1364-1995 和 1364-2001）中只有一些简单的结构可以用于创建测试。它们的验证能力无法满足设计规模的增长，所以后来出现了商用的硬件验证语言（HVL），例如 OpenVera 和 e 语言。那些不愿意购买商用验证工具的公司只能花费大量人力去创建自己的定制工具。

这场生产能力上的危机（连同设计上出现的类似问题）催生了 Accellera，它是一个由公司和用户共同组建的联盟，旨在创建下一代的 Verilog。来自 OpenVera 语言的捐赠构成了 SystemVerilog 作为 HVL 的基础。Accellera 的目标最终于 2005 年 11 月达成，IEEE 采纳了 SystemVerilog 作为标准，标准号为 P1800-2005。

统一语言的重要性

“验证”通常被认为是一种从根本上有别于设计的行为。这种区分导致了出现一些专注于验证语言开发的活动，同时也使得验证和设计工程师们在原则上产生了重大分歧，甚至使他们的沟通出现了障碍。SystemVerilog 较好地解决了这一问题，它的语言特性是两个阵营都能接受的。两边的工程师们都不用放弃自己赖以成功的优势，同时，设计和验证工具在语法和语义上的统一又便利了他们的沟通。例如，一个设计工程师即使无法编写面向对象的测试平台环境，那他也很容易读懂一个这样的测试并明白其内容，这使得设计和验证的工程师们可以在一起鉴别和解决问题。同样地，设计者明白自己模块的工作原理，他是为模块编写断言的最优人选，但验证工程师在创建模块间的断言方面可能会有更宽的视野。

把设计、测试平台和断言结构集中到一种语言里的另一个好处是：测试平台可以更容易地访问环境中的所有部分，而不需要采用专门的应用编程接口（API）。硬件验证语言（HVL）的价值在于它能够创建高层次、高灵活度的测试，而不在于它的循环结构或者声明风格。SystemVerilog 的基础正是那些为工程师们所长期使用的 Verilog 结构。

方法学的重要性

学习一门语言的语法与学习使用一种工具有一些不同。本书介绍的验证技术，专注于使用受约束的随机测试，它利用功能覆盖率来衡量进度并指导验证。随着章节的展开，语言和方法学的特性会同时呈现。有关方法学的更多内容，可参考 Bergeron et al.

(2006)。

SystemVerilog最有意义的优点在于,它允许用户在多个项目中使用连续一贯的语法来构造可靠并且可重复的验证环境。

关于SystemVerilog和SystemC在高层次设计上的比较

由于SystemVerilog集成了面向对象编程、动态线程和线程间通信等特性,所以它可以用于系统设计。当谈到SystemVerilog的应用时,IEEE的标准会首先提及结构建模,然后才是设计、断言和测试。SystemC也可以用于结构建模。

SystemC和SystemVerilog的主要不同之处在于:

- SystemVerilog提供了一种建模语言。你不需要学习C++和标准模板库(standard template library)就可以创建模型。
- SystemVerilog简化了自顶而下的设计。你可以在SystemVerilog中创建模型,然后在下一层再重新定义每个模块。初始的系统级模型可以被重新用作参考模型。
- 软件开发者希望有一个既快速又便宜甚至是免费的硬件仿真器。在SystemC和SystemVerilog中都可以创建高性能的事务级模型。SystemVerilog仿真器需要许可,软件开发者可能不想为此付费。SystemC可以是免费的,但前提是你所有的模型都能够在SystemC中获取到。

本书概要

SystemVerilog语言包含了设计、验证、断言和其他方面的很多特性。本书的内容主要集中在用于验证设计的结构上。使用SystemVerilog可以有很多种解决问题的途径。本书解释了各种解决方案之间的折中。

第1章验证导论,列出了各种验证技术,可作为学习和使用SystemVerilog语言的基础。这些引导性的建议强调了在分层测试平台环境下由覆盖率驱动的随机测试。

第2章数据类型,涵盖了新的SystemVerilog数据类型,如数组、结构、枚举类型和压缩变量。

第3章过程语句和子程序,展示了新的过程语句以及在任务和函数上的一些改进。

第4章连接设计和测试平台,展示了新的SystemVerilog验证结构,例如程序块、接口和时钟块,以及如何使用它们来建立测试平台并且把测试平台连接到待测设计上。

第5章面向对象编程基础,介绍了面向对象编程,解释了如何创建类、构造对象以及使用句柄。

第6章随机化,展示了如何使用SystemVerilog中受约束的随机激励产生机制,包括很多技术和样例。

第7章线程以及线程间的通信,展示了如何在测试平台中创建多线程,并且使用线程间的通信机制来实现线程间的数据交换以及它们的同步。

第8章面向对象编程的高级技巧指南,展示了如何使用面向对象编程来建立分层测试平台,以使得测试平台构件能被所有的测试所共享。

第9章功能覆盖率,解释了不同类型的覆盖率以及如何使用功能覆盖率来衡量验证计划的进展。

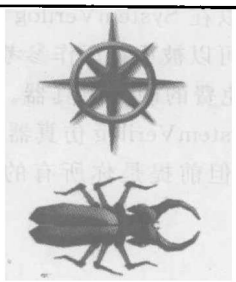
第10章高级接口，展示了如何使用虚接口来简化测试平台代码、连接多个设计配置，以及使用过程代码创建接口以使得你的测试平台和设计可以在一个更高的抽象层次上工作。

第11章完整的SystemVerilog测试平台，展示了在第8章的引导下创建的一个受约束的随机测试平台。用几个测试的例子来说明如何在不修改原来代码的情况下扩展测试平台的行为，当然这些做法都带有引入新漏洞的风险。

第12章SystemVerilog与C语言的接口，描述了如何使用直接编程接口把C或C++代码与SystemVerilog连接起来。

本书使用的图标

表1 本书图标

	表示验证方法学，用于引导你使用SystemVerilog测试平台的特性
	表示代码编写中常见的错误

最后的说明

如果你想了解有关SystemVerilog和验证方面的更多信息，可以在http://chris.spear.net/systemverilog上面找到很多资源。

这个网站中有本书大部分例子的源代码。所有例子都是在Synopsys的Chronologic VCS 2005.06，2006.06，和2008.03上面验证过。SystemVerilog语言参考手册涵盖了上千种新特性。本书主要集中在那些可用于验证的结构上，并且都在VCS中进行了实现。与其冒着代码出错的风险给出所有的语言特性，还不如使用验证过的例子。说到错误，如果你认为在本书中找到了错误，请登录到网站中的勘误页面进行核查。如果你是在某一章中第一个发现错误的人，那么你将得到一本我亲笔签名的免费赠书。

克里斯·斯皮尔
Synopsys公司
chris@spear.net

致　谢

很少有一个人单独创作一本书的情形。我想感谢所有帮助过我学习 SystemVerilog 以及审阅过本书的人，他们为此付出了大量的时间和精力。尤其要感谢所有在 Synopsys 公司的同事，包括所有耐心的经理，他们对我有莫大的帮助。

特别感谢 Shalom Bresticker、James Chang、David Lee、Ronald Mehler、Mike Mintz、Tim Pylant、Stuart Sutherland 和 Tuan Tran，他们审阅了非常粗糙的初稿并提出很多改进意见。当然，所有的错误归咎于我。

Janick Bergeron 激发了我的灵感，并提供了无数的验证技术的案例。没有他的指导，本书不可能成稿。

Alex Potapov、Horia Toma，以及 VCS 的研发团队对我的提问一直非常耐心，并提供了对于 SystemVerilog 特性非常有价值的观点。

Will Sherwood 激励我成为一名验证工程师，并且教给我很多突破问题的新思路。

下列人员指出了第一版本中的错误，并且就本书的改进提出了非常有价值的建议：Dan Abate、Steve Barrett、Mike Blake、Shalom Bresticker、John Brooks、Heath Chambers、Keith Chan、Luke Chang、Haihui Chen、Gunther Clasen、Hashem Heidaragha、Stefan Kruepe、Jimnan Kuo、Jim Lewis、Daguang Liu、Victor Lopez、Michael Macheski、Chris Macionski、Mike Mintz、Thinh Ngo、John Nolan、Ben Rahardja、Afroza Rahman、Chandrasekar Rajanayagam、Jonathan Schmidt、Chandru Sippy、Dave Snogles、Tuan Tran、Robin van Malenhorst、Hugh Walsh、Larry Widigen 和 Chunlin Zhang。

Jenny Bagdigian 始终督促我要一丝不苟。期望与他在狂欢节上见面！

联合航空一直提供安静的工作地点和足量的小吃。“要鸡肉还是意大利面？”尤响耳畔。

最后，还要特别感谢 Jay Mcinerney 教给我对于代词的直率用法。

所有商标和版权归属于它们各自的所有者。

目　录