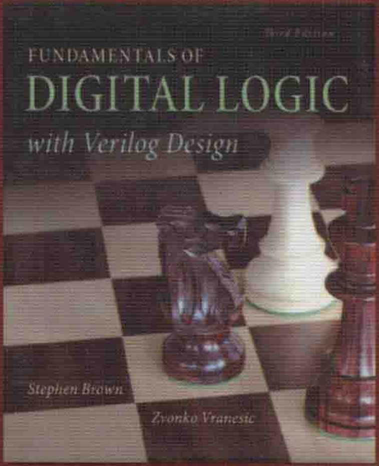
清华版双语教学用书



数字逻辑与Verilog设计

（第3版）

Fundamentals of Digital Logic with Verilog Design

Third Edition

Stephen Brown
Zvonko Vranesic 著

罗 嵘 选译

清华大学出版社

清华版双语教学用书

数字逻辑与Verilog设计

（第3版）

Fundamentals of Digital Logic with Verilog Design

Third Edition

Stephen Brown
Zvonko Vranesic 著

窦 嵘 选译

清华大学出版社
北 京

Stephen Brown, Zvonko Vranesic
Fundamentals of Digital Logic with Verilog Design, Third Edition
ISBN: 9780073380544

图书在版编目（CIP）数据

数字逻辑与 Verilog 设计：第3版：英、汉/（美）布朗（Brown，S.），（美）弗兰民斯克（Vranesic，Z.）著；罗嵘选译. --北京：清华大学出版社，2014
书名原文：Fundamentals of Digital Logic with Verilog Design, Third Edition
清华版双语教学用书
ISBN 978-7-302-36685-0

Ⅰ. ①数… Ⅱ. ①布… ②弗… ③罗… Ⅲ. ①数字逻辑－双语教学－教材－英、汉 ②硬件描述语言－程序设计－双语教学－教材－英、汉 Ⅳ. ①TP302.2 ②TP312

中国版本图书馆 CIP 数据核字（2014）第 117283 号

责任编辑：盛东亮
封面设计：常雪影
责任校对：时翠兰
责任印制：沈　露

出版发行：清华大学出版社
　网　　址：http://www.tup.com.cn，http://www.wqbook.com
　地　　址：北京清华大学学研大厦A座　　**邮　　编**：100084
　社 总 机：010-62770175　　**邮　　购**：010-62786544
　投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
　质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：清华大学印刷厂
装 订 者：三河市溧源装订厂
经　　销：全国新华书店
开　　本：185mm×230mm　　**印　　张**：54.75　　**字　　数**：1206千字
版　　次：2014年8月第1版　　**印　　次**：2014年8月第1次印刷
印　　数：1～2000
定　　价：99.00元

产品编号：055133-01

To Susan and Anne

序

本书英文原著的作者是多伦多大学的 Stephen Brown 和 Zvonko Vranesic 教授。他们长期从事相关教学和研究工作，具有丰富的经验。Stephen Brown 教授还是 Altera 公司全球大学计划的总负责人。

译者曾多次和 Stephen Brown 教授交流，并在 2010 年通过 Altera 公司大学培训计划，由 Stephen Brown 教授授予该公司的工程师培训资格。

此次有幸接受清华大学出版社盛东亮编辑邀请选译本书，给出书中每节的导读和相关术语，又重温了一遍这本在国内外大学本科教育以及工程师培训中广泛使用的经典教科书。

本书共包括 11 章正文和两篇附录。为了让读者了解传统的人工进行数字设计的基本理论，前六章主要还是介绍数字逻辑的基础，且此部分内容可以作为一个学期的数字逻辑设计导论课程。由于数字系统的规模越来越大，采用计算机辅助设计工具来完成数字电路的设计势在必行，因此本书从第 2 章开始就将相关知识融入到基础理论中，让读者能在阅读和学习过程中潜移默化地掌握 Verilog 代码编写风格，达到事半功倍的效果。

此外，本书基本上每章正文最后都有问题求解案例，让读者了解如何求解典型的习题。因此，该书作为高等院校的本科数字逻辑设计课程的参考教材或者从事数字设计的工程师的参考书籍，是非常有效的。

该书第 2 版由夏宇闻等进行了全文翻译（2008 年），本版书中的术语和定义的翻译参考了第 2 版。

罗　嵘

2014 年 4 月于清华大学

About the Authors

Stephen Brown received his B.A.Sc. degree in Electrical Engineering from the University of New Brunswick, Canada, and the M.A.Sc. and Ph.D. degrees in Electrical Engineering from the University of Toronto. He joined the University of Toronto faculty in 1992, where he is now a Professor in the Department of Electrical & Computer Engineering. He is also the Director of the worldwide University Program at Altera Corporation.

His research interests include field-programmable VLSI technology and computer architecture. He won the Canadian Natural Sciences and Engineering Research Council's 1992 Doctoral Prize for the best Ph.D. thesis in Canada and has published more than 100 scientific research papers.

He has won five awards for excellence in teaching electrical engineering, computer engineering, and computer science courses. He is a coauthor of two other books: *Fundamentals of Digital Logic with VHDL Design*, 3rd ed. and *Field-Programmable Gate Arrays.*

Zvonko Vranesic received his B.A.Sc., M.A.Sc., and Ph.D. degrees, all in Electrical Engineering, from the University of Toronto. From 1963–1965 he worked as a design engineer with the Northern Electric Co. Ltd. in Bramalea, Ontario. In 1968 he joined the University of Toronto, where he is now a Professor Emeritus in the Departments of Electrical & Computer Engineering and Computer Science. During the 1978–1979 academic year, he was a Senior Visitor at the University of Cambridge, England, and during 1984–1985 he was at the University of Paris, 6. From 1995 to 2000 he served as Chair of the Division of Engineering Science at the University of Toronto. He is also involved in research and development at the Altera Toronto Technology Center.

His current research interests include computer architecture and field-programmable VLSI technology.

He is a coauthor of four other books: *Computer Organization and Embedded Systems*, 6th ed.; *Fundamentals of Digital Logic with VHDL Design*, 3rd ed.; *Microcomputer Structures*; and *Field-Programmable Gate Arrays.* In 1990, he received the Wighton Fellowship for "innovative and distinctive contributions to undergraduate laboratory instruction." In 2004, he received the Faculty Teaching Award from the Faculty of Applied Science and Engineering at the University of Toronto.

He has represented Canada in numerous chess competitions. He holds the title of International Master.

PREFACE

This book is intended for an introductory course in digital logic design, which is a basic course in most electrical and computer engineering programs. A successful designer of digital logic circuits needs a good understanding of basic concepts and a firm grasp of the modern design approach that relies on computer-aided design (CAD) tools.

The main goals of the book are (1) to teach students the fundamental concepts in classical manual digital design and (2) illustrate clearly the way in which digital circuits are designed today, using CAD tools. Even though modern designers no longer use manual techniques, except in rare circumstances, our motivation for teaching such techniques is to give students an intuitive feeling for how digital circuits operate. Also, the manual techniques provide an illustration of the types of manipulations performed by CAD tools, giving students an appreciation of the benefits provided by design automation. Throughout the book, basic concepts are introduced by way of examples that involve simple circuit designs, which we perform using both manual techniques and modern CAD-tool-based methods. Having established the basic concepts, more complex examples are then provided, using the CAD tools. Thus our emphasis is on modern design methodology to illustrate how digital design is carried out in practice today.

TECHNOLOGY

The book discusses modern digital circuit implementation technologies. The emphasis is on programmable logic devices (PLDs), which is the most appropriate technology for use in a textbook for two reasons. First, PLDs are widely used in practice and are suitable for almost all types of digital circuit designs. In fact, students are more likely to be involved in PLD-based designs at some point in their careers than in any other technology. Second, circuits are implemented in PLDs by end-user programming. Therefore, students can be provided with an opportunity, in a laboratory setting, to implement the book's design examples in actual chips. Students can also simulate the behavior of their designed circuits on their own computers. We use the two most popular types of PLDs for targeting of designs: complex programmable logic devices (CPLDs) and field-programmable gate arrays (FPGAs).

We emphasize the use of a hardware description language in specifying the logic circuits, because the HDL-based approach is the most efficient design method to use in practice. We describe in detail the IEEE Standard Verilog HDL language and use it extensively in examples.

SCOPE OF THE BOOK

This edition of the book has been extensively restructured. All of the material that should be covered in a one-semester course is now included in Chapters 1 to 6. More advanced material is presented in Chapters 7 to 11.

Chapter 1 provides a general introduction to the process of designing digital systems. It discusses the key steps in the design process and explains how CAD tools can be used to automate many of the required tasks. It also introduces the representation of digital information.

Chapter 2 introduces the logic circuits. It shows how Boolean algebra is used to represent such circuits. It introduces the concepts of logic circuit synthesis and optimization, and shows how logic gates are used to implement simple circuits. It also gives the reader a first glimpse at Verilog, as an example of a hardware description language that may be used to specify the logic circuits.

Chapter 3 concentrates on circuits that perform arithmetic operations. It discusses numbers and shows how they can be manipulated using logic circuits. This chapter illustrates how Verilog can be used to specify the desired functionality and how CAD tools provide a mechanism for developing the required circuits.

Chapter 4 presents combinational circuits that are used as building blocks. It includes the encoder, decoder, and multiplexer circuits. These circuits are very convenient for illustrating the application of many Verilog constructs, giving the reader an opportunity to discover more advanced features of Verilog.

Storage elements are introduced in Chapter 5. The use of flip-flops to realize regular structures, such as shift registers and counters, is discussed. Verilog-specified designs of these structures are included.

Chapter 6 gives a detailed presentation of synchronous sequential circuits (finite state machines). It explains the behavior of these circuits and develops practical design techniques for both manual and automated design.

Chapter 7 is a discussion of a number of practical issues that arise in the design of real systems. It highlights problems often encountered in practice and indicates how they can be overcome. Examples of larger circuits illustrate a hierarchical approach in designing digital systems. Complete Verilog code for these circuits is presented.

Chapter 8 deals with more advanced techniques for optimized implementation of logic functions. It presents algorithmic techniques for optimization. It also explains how logic functions can be specified using a cubical representation as well as using binary decision diagrams.

Asynchronous sequential circuits are discussed in Chapter 9. While this treatment is not exhaustive, it provides a good indication of the main characteristics of such circuits. Even though the asynchronous circuits are not used extensively in practice, they provide an excellent vehicle for gaining a deeper understanding of the operation of digital circuits in general. They illustrate the consequences of propagation delays and race conditions that may be inherent in the structure of a circuit.

Chapter 10 presents a complete CAD flow that the designer experiences when designing, implementing, and testing a digital circuit.

Chapter 11 introduces the topic of testing. A designer of logic circuits has to be aware of the need to test circuits and should be conversant with at least the most basic aspects of testing.

Appendix A provides a complete summary of Verilog features. Although use of Verilog is integrated throughout the book, this appendix provides a convenient reference that the reader can consult from time to time when writing Verilog code.

The electronic aspects of digital circuits are presented in Appendix B. This appendix shows how the basic gates are built using transistors and presents various factors that affect circuit performance. The emphasis is on the latest technologies, with particular focus on CMOS technology and programmable logic devices.

What Can Be Covered in a Course

Much of the material in the book can be covered in 2 one-quarter courses. A good coverage of the most important material can be achieved in a single one-semester, or even a one-quarter course. This is possible only if the instructor does not spend too much time teaching the intricacies of Verilog and CAD tools. To make this approach possible, we organized the Verilog material in a modular style that is conductive to self-study. Our experience in teaching different classes of students at the University of Toronto shows that the instructor may spend only three to four lecture hours on Verilog, describing how the code should be structured, including the use of design hierarchy, using scalar and vector variables, and on the style of code needed to specify sequential circuits. The Verilog examples given in the book are largely self-explanatory, and students can understand them easily.

The book is also suitable for a course in logic design that does not include exposure to Verilog. However, some knowledge of Verilog, even at a rudimentary level, is beneficial to the students, and it is a great preparation for a job as a design engineer.

One-Semester Course

The following material should be covered in lectures:

- Chapter 1—all sections.
- Chapter 2—all sections.
- Chapter 3—Sections 3.1 to 3.5.
- Chapter 4—all sections.
- Chapter 5—all sections.
- Chapter 6—all sections.

One-Quarter Course

In a one-quarter course the following material can be covered:

- Chapter 1—all sections.
- Chapter 2—all sections.

- Chapter 3—Sections 3.1 to 3.3 and Section 3.5.
- Chapter 4—all sections.
- Chapter 5—all sections.
- Chapter 6—Sections 6.1 to 6.4.

VERILOG

Verilog is a complex language, which some instructors feel is too hard for beginning students to grasp. We fully appreciate this issue and have attempted to solve it. It is not necessary to introduce the entire Verilog language. In the book we present the important Verilog constructs that are useful for the design and synthesis of logic circuits. Many other language constructs, such as those that have meaning only when using the language for simulation purposes, are omitted. The Verilog material is introduced gradually, with more advanced features being presented only at points where their use can be demonstrated in the design of relevant circuits.

The book includes more than 120 examples of Verilog code. These examples illustrate how Verilog is used to describe a wide range of logic circuits, from those that contain only a few gates to those that represent digital systems such as a simple processor.

All of the examples of Verilog code presented in the book are provided on the Authors' website at

www.eecg.toronto.edu/~brown/Verilog_3e

SOLVED PROBLEMS

The chapters include examples of solved problems. They show how typical homework problems may be solved.

HOMEWORK PROBLEMS

More than 400 homework problems are provided in the book. Answers to selected problems are given at the back of the book. Solutions to all problems are available to instructors in the *Solutions Manual* that accompanies the book.

POWERPOINT SLIDES AND SOLUTIONS MANUAL

PowerPoint slides that contain all of the figures in the book are available on the Authors' website. Instructors can request access to these slides, as well as access to the Solutions Manual for the book, at:

www.mhhe.com/brownvranesic

CAD Tools

Modern digital systems are quite large. They contain complex logic circuits that would be difficult to design without using good CAD tools. Our treatment of Verilog should enable the reader to develop Verilog code that specifies logic circuits of varying degrees of complexity. To gain proper appreciation of the design process, it is highly beneficial to implement the designs using commercially-available CAD tools. Some excellent CAD tools are available free of charge. For example, the Altera Corporation has its Quartus II CAD software, which is widely used for implementing designs in programmable logic devices such as FPGAs. The Web Edition of the Quartus II software can be downloaded from Altera's website and used free of charge, without the need to obtain a license. In previous editions of this book a set of tutorials for using the Quartus II software was provided in the appendices. Those tutorials can now be found on the Authors' website. Another set of useful tutorials about Quartus II can be found on Altera's University Program website, which is located at www.altera.com/education/univ.

Acknowledgments

We wish to express our thanks to the people who have helped during the preparation of the book. Dan Vranesic produced a substantial amount of artwork. He and Deshanand Singh also helped with the preparation of the solutions manual. Tom Czajkowski helped in checking the answers to some problems. The reviewers, William Barnes, New Jersey Institute of Technology; Thomas Bradicich, North Carolina State University; James Clark, McGill University; Stephen DeWeerth, Georgia Institute of Technology; Sander Eller, Cal Poly Pomona; Clay Gloster, Jr., North Carolina State University (Raleigh); Carl Hamacher, Queen's University; Vincent Heuring, University of Colorado; Yu Hen Hu, University of Wisconsin; Wei-Ming Lin, University of Texas (San Antonio); Wayne Loucks, University of Waterloo; Kartik Mohanram, Rice University; Jane Morehead, Mississippi State University; Chris Myers, University of Utah; Vojin Oklobdzija, University of California (Davis); James Palmer, Rochester Institute of Technology; Gandhi Puvvada, University of Southern California; Teodoro Robles, Milwaukee School of Engineering; Tatyana Roziner, Boston University; Rob Rutenbar, Carnegie Mellon University; Eric Schwartz, University of Florida; Wen-Tsong Shiue, Oregon State University; Peter Simko, Miami University; Scott Smith, University of Missouri (Rolla); Arun Somani, Iowa State University; Bernard Svihel, University of Texas (Arlington); and Zeljko Zilic, McGill University provided constructive criticism and made numerous suggestions for improvements.

The support of McGraw-Hill people has been exemplary. We truly appreciate the help of Raghu Srinivasan, Vincent Bradshaw, Darlene Schueller, Curt Reynolds, and Michael Lange. We are also grateful for the excellent support in the typesetting of the book that has been provided by Techsetters, Inc.

Stephen Brown and Zvonko Vranesic

CONTENTS