Digital Design

Preview Edition

FRANK VAHID
University of California, Riverside

WILEY
John Wiley & Sons, Inc.

To my family, Amy, Eric, Kelsi, and Maya;
and to those engineers who apply their skills
to build things that improve the human condition.

SENIOR ACQUISITION EDITOR AND PRODUCT MANAGER	CATHERINE FIELDS SHULTZ
ASSOCIATE EDITOR	KELLY BOYLE
SENIOR PRODUCTION EDITOR	VALERIE A. VARGAS
MARKETING MANAGER	FRANK LYMAN
COVER ILLUSTRATION	MICHAEL JUNG
COVER DESIGNER	MADELYN LESURE
PRODUCTION SERVICES	INGRAO ASSOCIATES

This book was set in 10/12 Times Roman by Wiley/Australia and printed and bound by Banta/Harrisonburg. The cover was printed by Lehigh Press.

This book is printed on acid free paper.

To order books or for customer service please call 1-800-CALL WILEY (225-5945).

ISBN-13 978-0-471-46784-7
ISBN-10 0-471-46784-7

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

Preface

TO STUDENTS ABOUT TO STUDY DIGITAL DESIGN

Digital circuits, which form the basis of general-purpose computers as well as special-purpose devices like cell phones or video game consoles, are dramatically changing the world. Studying digital design not only gives you the confidence that comes with fundamentally understanding how digital circuits work, but also introduces you to an exciting and useful possible career direction. This statement applies regardless of whether your major is Electrical Engineering, Computer Engineering, or even Computer Science (in fact, the need for digital designers with strong computer science skills continues to increase). I hope you find this subject to be as interesting, exciting and useful as I do.

Throughout this book, I have tried not only to introduce concepts in the most intuitive and easy to learn manner, but I have also tried to show how those concepts can be applied to real-world systems, such as pacemakers, ultrasound machines, printers, automobiles, and cell phones. Young and capable engineering students (including computer science) sometimes leave their major, claiming they want a job that is more "people oriented." Yet we need those people-oriented students more than ever, as engineering jobs are increasingly people-oriented, in several ways. First, engineers usually work in *tightly-integrated groups* involving numerous other engineers (rather than "sitting alone in front of a computer all day" as many students believe). Second, engineers often work *directly with customers* (such as business people, doctors, lawyers, government officials, etc.), and must therefore be able to connect with those non-engineer customers. Third, and in my opinion most importantly, *engineers build things that dramatically impact people's lives*. Needed are engineers who combine their enthusiasm, creativity, and innovation with their solid engineering skills to conceive and build new products that improve people's quality of life.

I've included "Designer Profiles" at the end of most chapters. The designers, whose experience levels vary from just a year to several decades, and whose companies range from small to huge, share with you their experiences, insights and advice. You'll notice how commonly they discuss the people aspects of their jobs. You may also notice their enthusiasm and passion for their jobs.

TO INSTRUCTORS OF DIGITAL DESIGN

This book breaks from the 1970s/1980s digital design view emphasizing size-limited design, instead emphasizing the 2000s situation of ***register-transfer-level (RTL)*** design. By cleanly distinguishing the topic of basic design from optimization, two topics previously inseparably intertwined, the book allows a first course on digital design to reach and even emphasize the topic of RTL design. A student exposed to RTL design in a first course will have a more relevant view of the modern digital design field, leading not only to a better appreciation of modern computers and other digital devices, but a more accurate understanding of careers involving digital design. Such an accurate understanding is critical to attract computing majors to careers involving some amount of digital design, and to create a cadre of engineers with the comfort in both “software” and “hardware” necessary in modern embedded computing system design.

The distinguishment of basic design from optimization should not be interpreted as avoiding a bottom-up approach or glossing over important steps — the book takes a concrete bottom-up approach, starting from transistors, and building incrementally up through gates, flip-flops, registers, controllers, datapath components, etc. Rather, the distinguishment enables the student to initially develop a solid understanding of basic design, before considering the more advanced topic of optimization, akin to how a physics book introduces Newton’s laws of motion initially assuming frictionless surfaces and no wind resistance. Furthermore, optimization today involves more than just size minimization, instead requiring a broader understanding of tradeoffs among size, performance, and power, and even of tradoffs among custom digital circuits and microprocessor software. Again, coverage is kept concrete and appropriate to an introductory digital design course.

Nevertheless, the book distinguishes basic design from optimization in a way that cleanly provides an instructor maximum flexibility to introduce optimization at the times and to the extent desired by the instructor. In particular, the optimization chapter’s subsections (Chapter 6) each correspond directly to one earlier chapter, such that Section 6.2 can directly follow Chapter 2, Section 6.3 can follow Chapter 3, 6.4 can follow 4, and 6.5 can follow 5.

Several additional features of the book include:

- *Extensive use of applied examples*. After describing a new concept and providing basic examples, the book provides examples that apply the concept to applications recognizable to a student, like a seat belt unfastened warning system, a computerized checkerboard game, a color printer, or a digital video camera. Furthermore, the end of most chapters include a product profile, intended to give students an even broader view of the applicability of the concepts, and introducing clever application-specific concepts the students may find very interesting — like the idea of beamforming in an ultrasound machine, or of filtering in a cellular phone.
- *Learning through discovery.* The book emphasizes understanding the need for new concepts, which not only helps students learn and remember the concepts, but develops reasoning skills that can apply the concepts to other domains. For example, rather than just defining a carry-lookahead adder, the book shows intuitive but inefficient approaches to building a faster adder, eventually solving the inefficiencies and leading to (“discovering”) the carry-lookahead design.
- *Introduction to FPGAs*. The book includes a fully bottom-up introduction to FPGAs, showing students concretely how a circuit can be converted into a bitstream that programs the individual lookup tables, switch matrices, and other programmable components in an FPGA. This concrete introduction eliminates the mystery of the increasingly-common FPGA devices.

- *HDL coverage flexibility.* The book's organization cleanly allows instructors to cover HDLs intermixed with the introduction of design concepts, to cover HDLs later, or to not cover HDLs at all. The HDL chapter's subsections (Chapter 9) each correspond to an earlier chapter, such that Section 9.2 can directly follow Chapter 2, 9.3 can follow 3, 9.4 can follow 4, and 9.5 can follow 5. Furthermore, rather than the book choosing one of the popular languages — VHDL, Verilog, and the relatively new SystemC — the book provides equal coverage of all three of those HDLs. And, we use our extensive experience in synthesis with commercial tools to create HDL descriptions well-suited for synthesis, in addition to being suitable for simulation.

HOW TO USE THIS BOOK

This book was designed to allow flexibility to choose among the most common approaches to material coverage. We describe several approaches below.

RTL-focused approach

An RTL-focused approach would simply cover the first 6 chapters in order:

1. Introduction (Chapter 1)
2. Combinational logic design (Chapter 2)
3. Sequential logic design (Chapter 3)
4. Combinational and sequential component design (Chapter 4)
5. RTL design (Chapter 5)
6. Optimizations and Tradeoffs (Chapter 6), to the extent desired
7. Physical implementation (Chapter 7) and/or Processor design (Chapter 8) to the extent desired.

We think this is a great way to order the material, resulting in students doing interesting RTL designs in about 7 weeks. HDLs can be introduced at the end if time permits, or left for a second course on digitial design (as done at UCR), or covered immediately after each chapter — all three approaches appear to be quite common.

Traditional approach with some reordering

This book can be readily used in a traditional approach that introduces optimization along with basic design, with a slight difference from the traditional approach being the swapping of coverage of combinational components and sequential logic, as follows:

1. Introduction (Chapter 1)
2. Combinational logic design (Chapter 2) followed by combinational logic optimization (Section 6.2)
3. Sequential logic design (Chapter 3) followed by sequential logic optimization (Section 6.3)
4. Combinational and sequential component design (Chapter 4) followed by component tradeoffs (Section 6.4)
5. RTL design (Chapter 5) to the extent desired, followed by RTL optimization/tradeoffs (Section 6.5)

6. Physical implementation (Chapter 7) and/or Processor design (Chapter 8) to the extent desired.

This is a very reasonable and effective approach, completing all discussion of one topic (e.g., FSM design as well as optimization) before moving on to the next topic. The reordering from a traditional approach introduces basic sequential design (FSMs and controllers) before combinational components (e.g., adders, comparators, etc.). Such reordering may lead into RTL design more naturally than a traditional approach, following instead an approach of increasing abstraction rather than the traditional approach that separates combinational and sequential design. HDLs can again be introduced at the end, left for another course, or integrated after each chapter. This approach could also be used as an intermediary step when migrating from a fully-traditional approach to an RTL approach. Migrating might involve gradually postponing the Chapter 6 sections — for example, covering Chapters 2 and 3, and then Sections 6.2 and 6.3, before moving on to Chapter 4.

Completely traditional approach

The book could also be used in a completely traditional approach, as follows:

1. Introduction (Chapter 1)
2. Combinational logic design (Chapter 2) followed by combinational logic optimization (Section 6.2)
3. Combinational component design (Sections 4.1, 4.3, 4.4, 4.5, 4.7, 4.8, 4.9) followed by combinational component tradeoffs (Section 6.4 — Adders)
4. Sequential logic design (Chapter 3) followed by sequential logic optimization (Section 6.3)
5. Sequential component design (Chapter 4, Sections 4.2, 4.6, 4.10) followed by sequential component tradeoffs (Section 6.4 — Multipliers)
6. RTL design (Chapter 5) to the extent desired, followed by RTL optimization/tradeoffs (Section 6.5)
7. Physical implementation (Chapter 7) and/or Processor design (Chapter 8) to the extent desired.

This is the most widespread approach during the past two decades, with the addition of RTL towards the end. Although the emphasized distinction between combinational and sequential design may no longer be relevant in the era of RTL design (where both types of design are intermixed), some people believe that such distinction makes for an easier learning path, which may be true. HDLs can be included at the end, left for a later course, or integrated throughout.

ACKNOWLEDGEMENTS

Many people and organizations contributed to this edition of the book.

- Bill Zobrist, formerly at John Wiley and Sons, has been instrumental in making this project happen from the very beginning. He originally took a risk publishing my earlier book, Embedded System Design. That project, combined with Bill's encouragement, led me to realize the need for this new book on dital design. Kelly Boyle and Catherine Fields Shultz of Wiley also contributed significantly.

- Ryan Mannion contributed many items to the book, including the appendices, numerous examples and exercises, several subsections, the exercise solutions manual, fact-checking, extensive proofreading, and much assistance during production.
- Roman Lysecky developed numerous examples and exercises, and contributed most of the content of the HDL chapter. Roman and Susan Lysecky provided much proofreading assistance.
- Numerous reviewers provided outstanding feedback on various versions of the book.
- The importance of the support provided to my research and teaching career by the National Science Foundation cannot be overstated.

ABOUT THE COVER

The cover's image of shrinking squares graphically depicts the amazing real-life phenomena of digital circuits ('computer chips') shrinking in size by one half roughly every 18 months, for several decades now, a phenomena often referred to as Moore's Law. Such shrinking has enabled incredibly powerful computing circuits to fit in tiny devices, like modern cell phones, medical devices, and portable video games.

See the book's website at `http://www.cs.ucr.edu/~vahid/dd` for additional materials, or to submit comments, corrections, or suggestions.