Preface

In 1988 I began to teach full-custom VLSI design. In 1990 I started teaching ASIC design instead, because my students found it easier to get jobs in this field. I wrote a proposal to The National Science Foundation (NSF) to use electronic distribution of teaching material. Dick Lyon helped me with preparing the first few CD-ROMs at Apple, but Chuck Seitz, Lynn Conway, and others explained to me that I was facing a problem that Carver Mead and Lynn had experienced in trying to get the concept of multichip wafers adopted. It was not until the publication of the Mead-Conway text that people accepted this new idea. It was suggested that I must generate interest using a conventional format before people would use my material in a new one (CD-ROM or the Internet). In 1992 I stopped writing papers and began writing this book - a result of my experiments in computer-based education. I have nearly finished this book twice. The first time was a copy of my notes. The second time was just before the second edition of Weste and Eshragian was published - a hard act to follow. In order to finish in 1997 I had to stop updating and including new ideas and material and now this book consists of three parts: Chapters 1-8 are an introduction to ASICs, 9-14 cover ASIC logical design, and 15-17 cover the physical design of ASICs.

The book is intended for a wide audience. It may be used in an undergraduate or graduate course. It is also intended for those in industry who are involved with ASICs. Another function of this book is an "ASIC Encyclopedia," and therefore I have kept the background material needed to a minimum. The book makes extensive use of industrial tools and examples. The examples in Chapters 2 and 3 use tools and libraries from MicroSim (PSpice), Meta Software (HSPICE), Compass Design Automation (standard-cell and gate-array libraries), and Tanner Research (L-Edit). The programmable ASIC design examples in Chapter 4-8 use tools from Compass, Synopsys, Actel, Altera, and Xilinx. The examples in Chapter 9 (covering low-level design entry) used tools from Exemplar, MINC, AMD, UC Berkeley, Compass, Capilano, Mentor Graphics Corporation, and Cadence Design Automation. The VHDL examples in Chapter 10 were checked using QuickVHDL from Mentor, V-System Plus from Model Technology, and Scout from Compass. The Verilog examples in Chapter 11 were checked using Verilog-XL from Cadence, V-System Plus, and VeriWell from Wellspring Solutions. The logic synthesis examples in Chapter 12 were checked with the ASIC Synthesizer product family from Compass and tools from Mentor, Synopsys, and UC Berkeley. The simulation examples in Chapter 13 were checked with QuickVHDL, V-System/Plus, PSpice, Verilog-XL, DesignWorks from Capilano Computing, CompassSim, QSim, MixSim, and HSPICE. The test examples in Chapter 14 were checked using test software from Compass, Cadence, Mentor, Synopsys and Capilano's DesignWorks. The physical design examples in Chapters 15-17 were generated and tested using Preview, Gate Ensemble, and Cell Ensemble (Cadence) as well as ChipPlanner, ChipCompiler, and PathFinder (Compass). All these tools are installed at the University of Hawaii.

I wrote the text using FrameMaker. This allows me to project the text and figures using an LCD screen and an overhead projector. I used a succession of Apple Macintosh computers: a PowerBook 145, a 520, and lastly a 3400 with 144 MB of RAM, which made it possible for me to create updates to the index in just under one minute. Equations are "live" in FrameMaker. Thus, can be updated in a lecture and the new result displayed. The circuit layouts are color EPS files with enhanced B&W PICT previews created using L-Edit from Tanner Research. All of the Verilog and VHDL code examples, compiler and simulation input/output, and the layout CIF that were used in the final version are included as conditional (hidden) text in the FrameMaker document, which is approximately 200 MB and just over 6,000 pages (my original source material spans fourteen 560 MB optical disks). Software can operate on the hidden text, allowing,

for example, a choice of simulators to run the HDL code live in class. I converted draft versions of the VHDL and Verilog LRMs and related standards to FrameMaker and built hypertext links to my text, but copyright problems will have to be solved before this type of material may be published. I drew all the figures using FreeHand. They are "layered" allowing complex drawings to be built-up slowly or animated by turning layers on or off. This is difficult to utilize in book form, but can be done live in the classroom.

A course based on FPGAs can use Chapter 1 and Chapters 4-8. A course using commercial semicustom ASIC design tools may use Chapters 1-2 or Chapters 1-3 and then skip to Chapter 9 if you use schematic entry, Chapter 10 (if you use VHDL), or Chapter 11 (if you use Verilog) together with Chapter 12. All classes can use Chapters 13 and 14. FPGA-based classes may skim Chapters 15-17, but classes in semicustom design should cover these chapters. The chapter dependencies - Y (X) means Chapter Y depends on X - are approximately: 1, 2(1), 3(2), 4(2), 5(4), 6(5), 7(6), 8(7), 9(2), 10(2), 11(2), 12(10 or 11), 13(2), 14(13), 15(2), 16(15), 17(16).

I used the following references to help me with the orthography of complex terms, style, and punctuation while writing: Merriam-Webster's Collegiate Dictionary, 10th edition, 1996, Springfield, MA: Merriam-Webster, ISBN 0-87779-709-9, PE1628.M36; The Chicago Manual of Style, 14th edition, Chicago: University of Chicago Press, 1993, ISBN 0-226-10389-7, Z253.U69; and Merriam-Webster's Standard American Style Manual, 1985, Springfield, MA: Merriam-Webster, ISBN 0-87779-133-3, PN147.W36. A particularly helpful book on technical writing is BUGS in Writing by Lyn DuprE, 1995, Reading, MA: Addison-Wesley, ISBN 0-201-60019-6, PE1408.D85 (this book grew from Lyn DuprE's unpublished work, Style SomeX, which I used).

The bibliography at the end of each chapter provides alternative sources if you cannot find what you are looking for. I have included the International Standard Book Number (ISBN) and Library of Congress (LOC) Call Number for books, and the International Standard Serial Number (ISSN) for journals (see the LOC information system, LOCIS, at http://www.loc.gov). I did not include references to material that I could not find myself (except where I have noted in the case of new or as yet unpublished books). The electronic references given in this text have (a last) access date of 4/19/97 and omit enclosing <> if the reference does not include spaces.

I receive a tremendous level of support and cooperation from industry in my work. I thank the following for help with this project: Cynthia Benn and Lyn DuprE for editing; Helen Goldstein, Peter Gordon, Susan London-Payne, Tracy Russ, and Juliet Silveri, all at Addison-Wesley; Matt Bowditch and Kim Arney at Argosy; Richard Lyon, Don North, William Rivard, Glen Stone, the managers of the Newton group, and many others at Apple Computer who provided financial support; Apple for providing support in the form of software and computers; Bill Becker, Fern Forcier, Donna Isidro, Mike Kliment, Paul McLellan, Tom Schaefer, Al Stein, Rich Talburt, Bill Walker, and others at Compass Design Automation and VLSI Technology for providing the opportunity for me to work on this book over many years and allowing me to test material inside these companies and on lecture tours they sponsored; Chuck Seitz at Caltech; Joseph Cavallaro, Bernie Chern, Jerry Dillion, Mike Foster, and Paul Hulina at the NSF; the NSF for financial support with a Presidential Young Investigator Award; Jim Rowson and Doug Fairbairn; Constantine Anagnostopolous, Pin Tschang and members of the ASIC design groups at Kodak for financial support; the disk-drive design group at Digital Equipment Corp. (Massachusetts), Hewlett-Packard, and Sun Microsystems for financial support; Ms. MOSIS and all of the staff at MOSIS who each have helped me at one point or another by providing silicon, technical support, and documentation; Bob Brodersen, Roger Howe, Randy Katz, and Ed Lee of UC Berkeley for

help while I was visiting UCB; James Plummer of Stanford, for providing me with access to the Terman Engineering Library as a visiting scholar, as well as Abbas El Gamal and Paul Losleben, also at Stanford, for help on several occasions; Don Bouldin at University of Tennessee; Krzysztof Kozminski at MCNC for providing Uncle layout software; Gershom Kedem at Duke University for the public domain tools his group has written; Sue Drouin, JosE De Castro, and others at Mentor Graphics Corporation in Oregon for providing documentation and tools; Vahan Kasardjhan, Gail Grego, Michele Warthen, Steve Gardner, and others at the University Program at Cadence Design Automation in San Jose who helped with tools, documentation, and support; Karen Dorrington and the Cadence group in Massachusetts; Andy Haines, Tom Koppin, Sherri Mieth, Velma Miller, Robert Nalesnik, Mike Sarpa, Telle Whitney, and others at Actel for software, hardware, parts, and documentation; Peter Alfke, Leslie Baxter, Brad Fawcett, Chris Kingsley, Karlton Lau, Rick Mitchell, Scott Nance, and Richard Ravel at Xilinx for support, parts, software, and documentation; Greg Hedmann at NorComp for data on FPGAs; Anna Acevedo, Suzanne Bailey, Antje MacNaughton, Richard Terrell, and Altera for providing software, hardware programmers, parts, and documentation; the documentation group and executive management at LSI Logic for tools, libraries, and documentation; Toshiba, NEC, AT&T/NCR, Lucent, and Hitachi (for documentation); NEC for their visiting scholar program at UH; Fred Furtek, Oscar Naval, and Claire Pinkham at Concurrent Logic, Randy Fish at Crosspoint, and Gary Banta at Plus Logic-all for documentation; Paul Titchener and others at Comdisco (now part of Cadence Design Automation) for providing design tools; John Tanner and his staff at Tanner Research for providing their tools and documentation; Mahendra Jain and Nanci Magoun, who let me debug early prototypes at the IDEA conference organized by ASIC Technology and News; Exemplar for providing documentation on its tools; MINC for providing a copy of its FPGA software and documentation; Claudia Traver and Synopsys for tools and documentation; Mentor Graphics Corporation for providing its complete range of software; Alain Hanover and others at ViewLogic for providing tools; Mary Shepherd and Jerry Walker at IEEE for help with permissions; Meta Software for providing HSPICE; Chris Dewhurst and colleagues at Capilano Computing for its design tools; Greg Seltzer (Model Technology) and Charley Rowley for providing V-System Plus with online documentation prototypes; Farallon and Telebit for the software and hardware I used for early experiments with telelectures. Many research students at the University of Hawaii helped me throughout this project including: Chin Huang, Clem Portmann, Christeen Gray, Karlton Lau, Jon Otaguro, Moe Lwin, Troy Stockstad, Ron Jorgenson, Derwin Mattos, William Rivard, Wendy Ching, Anil Aggrawal, Sudhakar Jilla, Linda Xu, Angshuman Saha, Harish Pareek, Claude van Ham, Wen Huang, Kumar Vadhri, Yan Zhong, Yatin Acharya, and Barana Ranaweera. Each of the classes that used early versions of this text at the University of Hawaii at Manoa have also contributed by finding errors. The remaining errors are mine.

Michael John Sebastian Smith
Palo Alto and Honolulu, 1997

0201500221P04062001