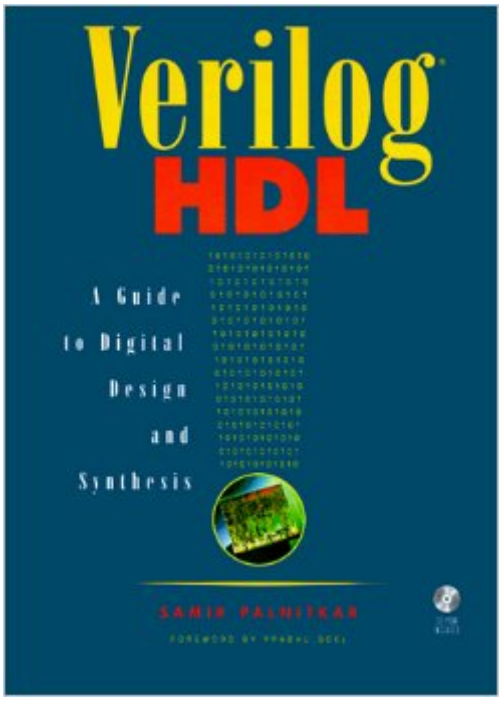
Verilog HDL



Synopsis

Stresses the practical design perspective of Verilog rather than emphasizing only the language aspects. The information presented is fully compliant with the upcoming IEEE 1364 Verilog HDL standard. CD ROM included.

Book Information

Hardcover: 396 pages

Publisher: Prentice Hall PTR; Bk&CD-Rom edition (January 15, 1996)

Language: English

ISBN-10: 0134516753

ISBN-13: 978-0134516752

Product Dimensions: 9.5 x 7.1 x 1 inches

Shipping Weight: 1.9 pounds

Average Customer Review: 3.8 out of 5 starsÂ Â See all reviewsÂ (40 customer reviews)

Best Sellers Rank: #562,104 in Books (See Top 100 in Books) #64 inÂ Books > Computers & Technology > Hardware & DIY > Microprocessors & System Design > Computer Design #116 inÂ Books > Textbooks > Engineering > Electrical & Electronic Engineering #1128 inÂ Books > Engineering & Transportation > Engineering > Electrical & Electronics > Electronics

Customer Reviews

This book is suitable for its time, but the contents are more intended for Verilog models. It talks a lot about the Verilog syntax; however, the title mislead me - VERY LITTLE discussion about Verilog for Synthesis. There is only one chapter devoted to systhesis, and this chapter happens to be the LAST chapter of the book.Don’t get me wrong. This is a good book for beginners in Verilog HDL. It is a "MIX-BAG" of syntax only for understanding (NOT synthesis). After reading this book, one should be familiar with the application of Verilog for Simulation and Modeling, but one may still have to get a synthesis book or training to further utilize the power of Verilog for FPGA or ASIC design (I am an FPGA designer). I gave it 3 stars due a misleading title and the fact that only a single chapter was devoted to synthesis topic.One topic bothers me. Why would someone use Verilog HDL to model the Transisor Level (he calls it the switch level)? Isn’t this the reason SPICE programs were designed to do? Besides that, Verilog HDL is still YET to add ANALOG capabilities.

As a first book on Verilog this is not a good book. I could not find any thing I was looking for in the index. There is only about a quarter of a page on memories and only a four line example. Most of

the examples were poor and incomplete. This book is not for beginners. I found about four other books that were good references, such as A Verilog HDL Primer, Third Edition by J. Bhasker and Verilog HDL: Digital Design and Modeling by Joseph Cavanagh. Both books have good examples and explanations. If your just starting out buy these two books and not Verilog HDL by Samir Palnitkar.

This book starts from very basic knowledge of Verilog. It assumes no prior knowledge of the language. It starts from explaining the different data types of Verilog with very clear examples. Then it shows you how to create a module and why is it important to divide your design into several modules. Then it teaches you how to create a circuit at gate level modeling. The schematics of the the code is given in the book, so you can clearly see the relationship between the schematics and its corresponding Verilog code. Then it moves to data flow modeling, again with very clear examples. After that, it moves to Behavioral Modeling. This the only book that gives me a very good understanding of always block, initial block, reg data type, and most importantly blocking and non blocking assignment. It has excellent example of how to create a state machine. The rest of the book teach you some tips that will make it easier for you to write your code. It also has some very good examples of how to create a testbench. Almost every example has its own testbench. So, you will learn how to create a testbench as you read the book from the very beginning. I only have 2 complains of the book. One is it doesn't tell you which part of the language is synthesizable until near the end of the book. Second is it doesn't tell you which part of the examples is the new features in Verilog 2001. This could be problematic if your simulator and synthesis tool does not fully support Verilog 2001. But other than that, I am fully satisfied. When I first learned Verilog, I browsed many other books. And this one is absolutely the best one. Without this book, I won't be able to finish my projects ahead of time with an A+. Good job Mr. Palnitkar!

The early chapters are an excellent guide to the syntax and wiring connections, they clearly outline potential pitfalls. Chapters 5 & 6 cover designs from too low a level. I'm sure other real-world designers will agree that modeling a flip-flop with gates just flops. I'm on Ch. 7 now and except for a reference on (min:typ:max) back to (yech!) chapter 5 the author finally gets back to real world design and synthesis (though he doesn't differentiate). Design for synthesis is a different animal than test design and where they diverge should be hi-lighted. But the explanations are clear and eventually the important levels of verilog design are covered. So if read thoughtfully it is a very good intro to verilog, but beware the useless filler.

I used this book to learn Verilog and if you read it from beginning to end, you might learn the gist of the language...but that’s it. The book is virtually useless as any kind of reference source. The index is almost unusable (if you want to learn about the keywords "fork" or "join", for example, good luck. They aren’t even listed in the index, along with just about everything else). Descriptions of how the language works are cryptic and overly brief, though the examples are sometimes helpful.I seldom write reviews of books, but this one has annoyed me so much that I felt compelled to do so.All in all, it’s better than no book at all, but not much better.

Like the other Verilog books I’ve read, this one seems to be a manual for those who wish to model rather than those who wish to synthesize actual circuits. The book concentrates too much on behavioral code and gate level design, neither of which are as important or as difficult as synthesizable code.

The 2nd Edition of this book should be renamed, "Verilog 2001 HDL". Palnitkar added Verilog 2001 to this edition but neglected to mention the difference between Verilog and Verilog 2001 in his text. For example in his 2nd edition he says that arrays can be declared with any number of dimensions. But he fails to add that is only true for Verilog 2001. Many simulators and synthesis tools don’t support all of verilog 2001 yet so you may have trouble getting the example code working in a real design.

Library Introduction to Verilog Digital VLSI Design with Verilog: A Textbook from Silicon Valley Technical Institute VLSI Chip Design with the Hardware Description Language VERILOG: An Introduction Based on a Large RISC Processor Design