Book Reviews

K. Coffman

Real World FPGA Design with Verilog

Prentice Hall, Upper Saddle River, New Jersey, USA, 2000, pp. xv, 291, ISBN 0–13–099851–6

The title of this book describes well its contents, purpose and, in some sense, the methods. It covers the complete design cycle, from the specification in Verilog up to the implementation in FPGA devices, and, at the same time, it constantly warns about real world problems that should be taken care of. Though FGPA is in the central topic of this book, one chapter — the last one — deals with other types of ASICs, too. With the book comes a CD containing the demo version of the Silos III simulation tool, the editor for Verilog, the evaluation version of an electronics formula tool and listings from the book — totaling some 20 Mbytes of data.

The book consists of nine chapters, a glossary and a list of acronyms, as well as the bibliography and the index, totaling 291 pages, with 121 figures, 14 tables and 128 listings. It should be noted that some figures, though only a minority of them, are actually tables.

The first chapter is the introductory one. First there is a short description of the design process in an imaginary company. The example is designing a trivial overheat detector and it describes the steps from the request to FPGA fitting. Afterwards a review of Verilog HDL is given. This review takes most of the chapter and mentions synthesizable Verilog elements, Verilog hierarchy, built-in logic primitives, latches and flip-flops, blocking and nonblocking assignments and miscellaneous Verilog syntax items.

The second chapter is an overview of the design process. It starts with the review of the lowest elements used for the physical realization of a design, namely analog elements that build up digital ones. After that, each step of the design process is described in detail, from the specification in Verilog to the technology. For each step possible problems are outlined and some recommendations are given.

The third chapter entitled *A Digital Circuit Toolbox* is a toolbox of different digital elements, their Verilog models and FPGA realizations. The author describes three state signals and busses, bidirectional busses, priority encoders, state machines, adders, subtractors and multipliers. Part of the chapter considers some issues with synthesis, namely area/speed optimization, trade-offs between speed and latency and delays in FPGA logic elements. The fourth chapter continues these issues with some additional circuitry like different types of counters, shift registers, RAMs, FIFOs, and the like.

In the fifth chapter Verilog's simulation and test capabilities are described. In particular it explains compiler directives (words starting with $) and possibilities for automated testing. This chapter is motivated by the fact that sooner or later one's design will end up in a physical device like FPGA. To demonstrate real world design problems the author uses LeonardoSpectrum and Xilinx's FPGA devices and accompanying tools in a design process of an 8-bit equality comparator. In this respect the chapter is quite detailed.

The seventh chapter reviews possible architectures from two vendors, that a designer can use to implement her/his designs. It starts with a FPGA technology selection check- list and continues with the overview of some of Xilinx FPGA and Altera CPLD architectures.

Reusability is the topic of the eigth chapter. Here the author discusses pros of reusability, different ways of achieving it, and also gives some guidance for the designers in order for their product to be as reusable as possible.

Finally, the ninth chapter deviates a bit from the rest of the book since it is not concerned with designing for FPGAs, but with the way to realize the designs in ASICs. It starts with a list of differences between FPGAs and ASICs, continues with a discussion on the possibility to transfer FPGA designs into ASIC ones, and finishes with ASICs specifics. This last part takes about half of the chapter.

At the end of the book there is an afterword of about a page, that takes a look into the future with million-gates FPGA devices. Then comes a resource list with links to Internet sites of different companies and some public discussion groups. Next is the glossary with nearly 150 terms and acronyms and the bibliography.

This book is quick (and "dirty", as the author puts it) guide to FPGA design process in the real world. The author assumes neither previous experience with FPGA or ASIC design process nor any knowledge in that respect, although some basic knowledge from digital design is required. The book is written quite well, is easy to understand and interesting to read. Finally, it is of great value for new FPGA designers.

Stjepan Groš
Faculty of Electrical Engineering and Computing
University of Zagreb
Zagreb
Croatia

B. Zeidman

Verilog Designer's Library

Prentice Hall, Upper Saddle River, New Jersey, USA, 1999, pp. xv, 411, ISBN 0–13–081154–8

The rationale of this book is to be read by digital hardware designers. It is analogous to the one with algorithms for programmers, and it fulfills that purpose quite successfully. There are many models of different functional blocks in it, both behavioral and RTL, that can be used and modified by the reader in her/his own designs. It is targeted to both experienced and novice designers — primarily for those using, or intending to use, Verilog as their HDL of choice. While in the case of experienced designers its purpose will primarily be some sort of a reference and/or occasional help, for novice designers it has the additional purpose as an introduction into coding techniques and modeling with the Verilog HDL.

This book consists of 412 pages of text matter, which comprehends 29 chapters subdivided into 7 parts, the appendix, glossary, index and a short author biography. Apart from numerous listings, mainly in Verilog but some in C as well, there are also 67 figures, and 5 tables. The first part is introductory and therefore quite short, while in all other parts chapters have identical structure. First, there is a short introduction into the module's functionality, it is followed by the behavioral code of the functional unit, the RTL code, and, eventually, the simulation code. All the listings are also given on the accompanying CD-ROM, which has about 500K of filled space.

The first part, consisting of five comparatively short chapters, is spread out on 45 pages, and introduces the reader into general coding techniques. It begins commenting various styles and explains how to write a good-quality code, that is easy to understand and fast enough to simulate and synthesize. There are examples of what to do and what to try to avoid. The recommendations given in this chapter are used in all other parts of the book.

The second part, entitled *Basic Building Blocks*, has 4 chapters, which discuss the modeling of a simple J-K flip-flop, an 8-bit barrel shift register, a 8-bit counter and a 32-bit synchronous adder. The third part, *State Machines*, consists of three chapters, two of which deal with Moore and Mealy state machines, while the third one is dedicated to the design of state machines for FPGAs specifically. The fourth part — entitled *Miscellaneous Complex Functions* — gives the behavioral and RTL code for a number of elements with diverse functionality including a linear feedback shift register, an encrypter and a decrypter, a phase locked loop, and unsigned and signed integer multipliers. As its name suggests, the fifth part *Error Detection and Correction* deals with functional blocks for performing error detection and correction. There are 4 chapters, which describe Verilog modeling of the following error detection and correction

algorithms: parity generator and checker, Hamming code, simple checksum and cyclic redundancy check. The sixth part — *Memories*, totaling four chapters, contains models of memories, in particular RAM models, like dual port RAM, synchronous FIFO and synchronizing FIFO. Finally, the seventh part, again consisting of four chapters, deals with models of memory controllers; the chapters are dedicated to describing an SRAM/ROM controller, a synchronous SRAM controller, a DRAM controller and a fast page mode DRAM controller.

Stjepan Groš
Faculty of Electrical Engineering and Computing
University of Zagreb
Zagreb
Croatia