Brock J. LaMeres

Quick Start Guide to Verilog

Springer

Quick Start Guide to Verilog

Quick Start Guide to Verilog

1st Edition

Brock J. LaMeres

Springer

Brock J. LaMeres
Department of Electrical & Computer Engineering
Montana State University
Bozeman, MT, USA

ISBN 978-3-030-10551-8 ISBN 978-3-030-10552-5 (eBook)
https://doi.org/10.1007/978-3-030-10552-5

Library of Congress Control Number: 2018968403

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The classical digital design approach (i.e., manual synthesis and minimization of logic) quickly becomes impractical as systems become more complex. This is the motivation for the modern digital design flow, which uses hardware description languages (HDL) and computer-aided synthesis/minimization to create the final circuitry. The purpose of this book is to provide a quick start guide to the Verilog language, which is one of the two most common languages used to describe logic in the modern digital design flow. This book is intended for anyone that has already learned the classical digital design approach and is ready to begin learning HDL-based design. This book is also suitable for practicing engineers that already know Verilog and need quick reference for syntax and examples of common circuits. This book assumes that the reader already understands digital logic (i.e., binary numbers, combinational and sequential logic design, finite state machines, memory, and binary arithmetic basics).

Since this book is designed to accommodate a designer that is new to Verilog, the language is presented in a manner that builds foundational knowledge first before moving into more complex topics. As such, Chaps. 1–6 provide a comprehensive explanation of the basic functionality in Verilog to model combinational and sequential logic. Chapters 7–11 focus on examples of common digital systems such as finite state machines, memory, arithmetic, and computers. For a reader that is using the book as a reference guide, it may be more practical to pull examples from Chaps. 7–11 as they use the full functionality of the language as it is assumed the reader has gained an understanding of it in Chaps. 1–6. For a Verilog novice, understanding the history and fundamentals of the language will help form a comprehensive understanding of the language; thus it is recommended that the early chapters are covered in the sequence they are written.

Bozeman, MT, USA — Brock J. LaMeres

Acknowledgments

For Kylie. Your humor brings me laughter and happiness every day. Thank you.

Contents