# PREFACE

This book is intended as an introductory logic design book for students in computer science, computer engineering, and electrical engineering. It has no prerequisites, although the maturity attained through an introduction to engineering course or a first programming course would be helpful.

The book stresses fundamentals. It teaches through a large number of examples. The philosophy of the author is that the only way to learn logic design is to do a large number of design problems. Thus, in addition to the numerous examples in the body of the text, each chapter has a set of Solved Problems, that is, problems and their solutions, as well as a large set of Exercises. In addition, there are a set of laboratory experiments that tie the theory to the real world. The Appendix provides the background to do these experiments with a standard hardware laboratory (chips, switches, lights, and wires), a breadboard simulator (for the PC or Macintosh), and two schematic capture tools. The course can be taught without the laboratory, but the student will benefit significantly from the addition of 8–10 selected experiments.

Although computer-aided tools are widely used for the design of large systems, the student must first understand the basics. The basics provide more than enough material for a first course. The schematic capture laboratory exercises and a section on Hardware Design Languages in Chapter 6 provide some material for a transition to a second course based on one of the computer-aided tool sets.

Chapter 1 gives a brief overview of number systems as it applies to the material of this book. (Those students who have studied this in an earlier course can skip to Section 1.2.) It then discusses the steps in the design process for combinational systems and the development of truth tables.

Chapter 2 introduces switching algebra and the implementation of switching functions using common gates—AND, OR, NOT, NAND, NOR, and Exclusive-OR. We are only concerned with the logic behavior of the gates, not the electronic implementation.

Chapter 3 deals with two simplification techniques, the Karnaugh map and Iterated Consensus. It provides methods for solving problems (up to six variables with the map) with both single and multiple outputs.

Chapter 4 is concerned with the design of larger combinational systems. It introduces a number of commercially available larger devices, including adders, decoders, encoders and priority encoders, and multiplexers. That is followed by a discussion of the use of logic arrays—ROMs, PLAs, and PALs for the implementation of medium-scale combinational systems. Finally, two larger systems are designed.

Chapter 5 introduces sequential systems. It starts by examining the behavior of latches and flip flops. The design process for sequential systems is then presented. Before designing sequential systems, several systems are analyzed. The special case of counters is studied next. Finally, the solution of word problems, developing the state table or state diagram from a verbal description of the problem is presented in detail.

Chapter 6 looks at larger sequential systems. It starts by examining the design of shift registers and counters. Then, PLDs are presented. Two techniques that are useful in the design of more complex systems, ASM diagrams and HDLs, are discussed next. Finally, two examples of larger systems are presented.

Chapter 7 deals with state reduction and state assignment issues. First, a tabular approach for state reduction is presented. Then partitions are utilized both for state reduction and for achieving a state assignment that will utilize less combinational logic.

A feature of this text is the Solved Problems. Each chapter has a large number of problems, illustrating the techniques developed in the body of the text, followed by a detailed solution of each problem. Students are urged to solve each problem (without looking at the answer) and then compare their solution with the one shown.

Each chapter concludes with a large set of exercises. Solution to these will be made available through the Web.

Another unique feature of the book is the laboratory exercises, included in the Appendix. Four platforms are presented—a hardware based Logic Lab (using chips, wires, etc.); a hardware lab simulator that allows the student to "connect" wires on the computer screen; and two circuit capture programs, LogicWorks IV and Altera Max+plus II. Enough information is provided about each to allow the student to perform a variety of experiments. A set of 25 laboratory exercises are presented. Several of these have options, to allow the instructor to change the details from one term to the next.

We teach this material as a 4-credit course that includes an average of 3 1/2 hours per week of lecture, plus, typically, eight laboratory exercises. (The lab is unscheduled; it is manned by Graduate Assistants 40 hours per week; they grade the labs.) In that course we cover

Chapter 1: all of it

Chapter 2: all but 2.11

Chapter 3: all of 3.1

Chapter 4: all but 4.8. However, there is a graded design problem based on that material (10 percent of the grade; students usually working in groups of 2 or 3).

Chapter 5: all, though sometimes we skip 5.6

Chapter 6: 6.1, 6.2, and 6.3. We sometimes have a second project based on 6.6.

Chapter 7 and Section 3.2: We often have some time to look at one of these. We have never been able to cover both.

With less time, the coverage of Section 2.10 could be minimized. Section 3.1.5 is not needed for continuity; Section 3.1.6 is used somewhat in the discussion of PLAs in Section 4.7.2. Chapter 4 is not needed for anything else in the text, although many of the topics are useful to students elsewhere. Sections 5.5 and 5.6 could be eliminated without loss of continuity. As is the case for Chapter 4, the instructor can pick and choose among the topics of Chapter 6. With a limited amount of time, Section 7.1 could be covered. With more time, it could be skipped and state reduction taught using partitions (7.2 and 7.3).

## ACKNOWLEDGMENTS

I want to thank my wife, Allyn, for her encouragement and for enduring endless hours when I was closeted in my office working on the manuscript. Several of my colleagues at Florida Atlantic University have read parts of the manuscript and have taught from earlier drafts. I wish to acknowledge especially Mohammad Ilyas, Imad Mahgoub, Oge Marques, and Abhi Pandya for their help. In addition, I wish to express my appreciation to my two chairs, Mohammad Ilyas and Roy Levow, who made assignments that allowed me to work on the book. Even more importantly, I want to thank my students who provided me with the impetus to write a more suitable text, who suffered through earlier drafts of the book, and who made many suggestions and corrections. I want to thank Visram Rathnam for his contributions to the section on Altera tools. The reviewers

Nader I. Rafla, *Boise State University;*

Nicholas C. K. Phillips, *Southern Illinois University;*

Walter B. Ligon, III, *Clemson University;*

Jonathan Hill, *Worcester Polytechnic Institute;*

Anura Jayasumana, *Colorado State University;*

James K. Archibald, *Brigham Young University;*

Charles B. Silio, Jr., *University of Maryland;*

Gary J. Minden, *University of Kansas;*

and Michael McCool, *University of Waterloo,*

provided many useful comments and suggestions. The book is much better because of their efforts. Finally, the staff at McGraw-Hill, particularly Melinda Dougharty, Catherine Fields Shultz, Kay Brimeyer, Betsy Jones, Michelle Meerdink, Wayne Harms, Phil Meek, Sandy Ludovissy, Audrey Reiter, and John Wannemacher, have been indispensable in producing the final product, as have Brittney Corrigan-McElroy and her staff at Interactive Composition Corporation.