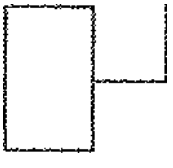
dandelon.com
© 2008 AGI-Information Management Consultants
May be used for personal purposes only or by
libraries associated to dandelon.com network.

Digital Design Essentials

Richard S. Sandige
California Polytechnic State University at San Luis Obispo, CA



Prentice Hall

PRENTICE HALL, Upper Saddle River, New Jersey 07458

Contents

PART I COMBINATIONAL LOGIC DESIGN

7 *Number Systems, Number Representations, and Codes* 7

1.1 Why Study Digital Design? 1

1.1.1 Learning to Think in Logical Terms 1

- Learning to Think in Logical Terms 1
- Learning to Organize Your Thought Processes 1
- Learning Digital Foundation Material 2

1.2 Digital Systems Organization 2

1.2.1 Computer Circuit Components and Interconnections 2

- Typical Combinational Logic Circuit Components 2
- Typical Sequential Logic Circuit Components 3
- Typical Circuit Diagrams or Schematics 4

1.2.2 Hierarchical Design Methodologies 7

- Top-Down Design Methodology for Systems 7
- Bottom-Up Design Methodology for Systems 8
- Middle-Out Design Methodology for Systems 8

1.3 Back to the Basics 8

1.3.1 Digital Circuits vs. Analog Circuits 8

- Digitized Signal for the Human Heart 8
- Discrete Signals vs. Continuous Signals 9

1.4 Binary Number System 10

1.4.1 Decimal, Binary, Octal, and Hexadecimal Numbers 10

- Explicit Notation for Numbers 10
- Digit Notation for Weighted Positional Number Systems 11

1.4.2 Conversion Techniques 12

- Subtract the Weights Method for Decimal to Binary Conversions 12
- Groups of 3 Method for Conversions from Binary to Octal and Back 12
- Groups of 4 Method for Conversions from Binary to Hex and Back 13

- Polynomial Function Method for Conversions from Any Base to Decimal 14
- Repeated Radix Division Method for Conversions from Decimal Integers to Any Base 16
- Repeated Radix Multiplication Method for Conversions from Decimal Fractions to Any Base 18

EXERCISES TO TEST YOUR KNOWLEDGE 20

1.5 Signed Number Representations 20

1.5.1 Sign Magnitude (SM) Representation 21

1.5.2 Diminished Radix Complement (DRC) Representation 22

1.5.3 Radix Complement (RC) Representation 23

EXERCISES TO TEST YOUR KNOWLEDGE 25

1.6 Binary Arithmetic 25

1.6.1 Addition of Signed and Unsigned Binary Numbers 25

1.6.2 Overflow, Its Meaning, and Its Detection 26

1.6.3 Indirect Subtraction by Addition Method 28

EXERCISES TO TEST YOUR KNOWLEDGE 29

1.7 Binary Codes 30

1.7.1 Minimum Number of Bits for Keypads and Keyboards 30

1.7.2 Commonly Used Codes: BCD, XS3, ASCII, etc. 30

1.7.3 Modulo 2 Addition, Conversions from Unsigned Binary (UB) to Reflective Gray Code (RGC) and Back 33

1.7.4 Seven-Segment Code 36

1.7.5 A Small Keypad Switch Matrix 37

EXERCISES TO TEST YOUR KNOWLEDGE 38

1.8 Worked Exercises 39

Problems 50

Boolean Algebra, Functions, and Minimization 59

2.1 Boolean Algebra Landmark Publications and Prominent People 59

2.1.1 Boole's Laws of Thought 59

2.1.2 Huntington's First Set of Postulates 59

2.1.3 Shannon's Symbolic Analysis of Switching Circuits 59

2.1.4 Other Prominent People 59

2.2 Huntington's Postulates, Truth Tables, and Graphic Symbols 60

2.2.1 Huntington's First Set of Postulates 60

2.2.2 Definition of: an Expression, a Literal, and a Hierarchy of Binary Operators 61
2.2.3 Principle of Duality 61
2.2.4 Truth Tables and Graphic Symbols for the Binary Operators + (OR), • (AND), and ~ (Complement) 63
2.2.5 Other Important Truth Tables and Graphic Symbols (NOR, NAND, XOR, and XNOR) 64
2.2.6 Writing Truth Table for Functions 67

EXERCISES TO TEST YOUR KNOWLEDGE 69

2.3 Boolean Algebra Theorems 69
2.3.1 Literal Count Reduction 69
2.3.2 Proving Theorems 71
• Proving Theorems via Boolean Algebra 71
• Proving Theorems via Truth Tables (Principle of Perfect Induction) 72

EXERCISES TO TEST YOUR KNOWLEDGE 74

2.4 Canonical or Standard Forms of Functions 75
2.4.1 Obtain ing Standard SOP Forms of Functions 75
• Compact Minterm Forms 76
2.4.2 Obtaining Standard POS Forms for Functions 78
• Compact Maxterm Forms 80
2.4.3 Relationship Between Minterms and Maxterms 82
2.4.4 Converting Between Compact Forms of Functions 84

EXERCISES TO TEST YOUR KNOWLEDGE 85

2.5 Function Minimization 86
2.5.1 A Systematic Algebraic Reduction (SAR) Technique 86
2.5.2 K-map Reduction 90
• K-map Organizations 90
• Filling in or Plotting Functions in K-maps 91
• Reading Reduced Expressions from K-maps 96
• Don't Care Outputs in K-maps 101

EXERCISES TO TEST YOUR KNOWLEDGE 102

2.6 Worked Exercises 103

Problems 113

3 Analyzing and Syntbesizing Combinational Logic Circuits 12 7

3.1 Standard Graphic Symbols and Two-Valued Switching Circuits 121

3.1.1 Switching Circuits for the OR, AND, and Complement Operations 121
3.1.2 Contrasting Pass Logic Switching Circuits with Regenerative Logic Switching Circuits 124
3.1.3 Integrated-Circuit Devices 125
3.1.4 Switching Circuits for the NOR, NAND, Buffer, XOR, and XNOR Operations 126
3.1.5 All Possible Single-and Two-Variable Functions 129

EXERCISES TO TEST YOUR KNOWLEDGE 131

3.2 Designing Logic Circuits 132
3.2.1 Analyzing and Designing Relay Logic Circuits 132
3.2.2 Analyzing IC Logic Circuits 133
3.2.3 Designing IC Logic Circuits 135
3.2.4 Generating Detailed Schematic Diagrams 138
3.2.5 Designing Circuits Using DeMorgan Equivalent Symbols and Equivalent Signal Lines 139
- Equivalent Graphic Symbols for NAND, NOR, AND, and OR Gates 140
- Equivalent Signal Lines 141
- Designing Circuits with NAND Gates and with NOR Gates 142

EXERCISES TO TEST YOUR KNOWLEDGE 144

3.3 Compressing Truth Tables and K-Maps 145
3.3.1 Technique for Compressing Truth Tables and K-Maps 145
3.3.2 Plotting, Filling, and Reducing Compressed K-Maps 149

EXERCISES TO TEST YOUR KNOWLEDGE 151

3.4 Glitches and Their Causes 152
3.4.1 Static Function Hazards and Static Logic Hazards 152
- Logic Hazard-Free Functions 153

3.4.2 Dynamic Hazards 155

EXERCISES TO TEST YOUR KNOWLEDGE 156

3.5 Types of Functions and Delays 156
3.5.1 Trivial Functions and Delays 156
3.5.2 Simple Functions and Delays 157
- Decoders 158
- Designing Circuits with Decoders 159

3.5.3 *Complex Functions and Delays 160*

- Multiplexers (Data Selectors) 161
- Designing Circuits with MUXs 163
- Designing Gate-Level Circuits with Multiple Outputs 165

EXERCISES TO TEST YOUR KNOWLEDGE 167

3.6 Programmable Logic Devices (PLDs) 168

3.6.1 PRO Ms 170
3.6.2 PLAs 171
3.6.3 PALsorGALs 172
3.6.4 Designing with PROMs 174
3.6.5 Designing with PLAs 175
3.6.6 Designing with PALs or GALs 176

EXERCISES TO TEST YOUR KNOWLEDGE 177

3.7 Positive Logic Convention and Direct Polarity Indication 178

3.7.1 Signal Names Revisited 178
3.7.2 Converting Circuits and Signal Matching 179

EXERCISES TO TEST YOUR KNOWLEDGE 182

3.8 Case Study Number 1 (BCD to 7-segment Display System) 183

3.8.1 The Gate-Level Design Approach 185
3.8.2 The MUX Design Approach 186
3.8.3 The PLD Design Approach 187
3.8.4 The Off-the-Shelf Design Approach 188

3.9 Worked Exercises 189

Problems 215

4 Additional Circuit Synthesis Techniques 233

4.1 Distributed Connections 233

4.1.1 Open-Collector and Open-Drain Outputs 233
4.1.2 Dot-AND Circuit Connection 234
4.1.3 Analyzing Circuits That Use a Dot-AND Connection 236
4.1.4 Designing Circuits That Use a Dot-AND Connection 237

EXERCISES TO TEST YOUR KNOWLEDGE 238

4.2 Three-State Outputs and the Disconnect State 239

4.2.1 Data Bus Sharing 241

EXERCISES TO TEST YOUR KNOWLEDGE 243

4.3 More about MUXs and Decoders/DMUXs 243
4.3.1 Designing Digital MUX Trees 244
- A Simple MUX Tree Design 245
- Designing Larger MUX Trees 245

4.3.2 Designing Digital DMUX Trees 248
- A Simple DMUX Tree Design 250
- Designing Larger DMUX Trees 251

EXERCISES TO TEST YOUR KNOWLEDGE 252

4.4 More about XOR and XNOR Symbols and Functions 253
4.4.1 Odd and Even Functions 254
4.4.2 Single-Bit Error Detection Scheme 255
4.4.3 Comparators 256

EXERCISES TO TEST YOUR KNOWLEDGE 257

4.5 Adder and Subtractor Design 258
4.5.1 Obtaining Half Adder and Half Subtractor Modules 258
4.5.2 Obtaining the Full Adder Module 260
4.5.3 Ripple-Carry Concept 262
4.5.4 Carry-Anticipation or Carry Look-Ahead Adders 264
- Ripple-Carry Adders vs. Carry Look-Ahead Adders 264
- Design of Carry Look-Ahead Adders 266

4.5.5 Carry-Save Adders 268

EXERCISES TO TEST YOUR KNOWLEDGE 270

4.6 Multiplier Design 271
4.6.1 Modular Design Process Using Two-Operand Adders 2 71
4.6.2 Modular Design Process Using a Multiple-Operand Adder 273

EXERCISES TO TEST YOUR KNOWLEDGE 274

4.7 ALU Design via a Data-Flow Modular Design Process 274

EXERCISES TO TEST YOUR KNOWLEDGE 278

4.8 Case Study Number 2 (Keypad Display System) 279
4.8.1 Obtaining Encoder Equations by Inspection 282
4.8.2 PLD Design Approach via MACHXL 282
4.8.3 PLD Design Approach via StateCAD 284

4.9 Worked Exercises 287

Problems 304

PART II SEQUENTIAL LOGIC DESIGN

5 *Feedback Circuits 311*

5.1 Latch Circuits 311

5.1.1 Analyzing S-R and~S—R Basic Latch Circuits 311

- The Simple ON/OFF Light Switch 311
- Circuit Delay Models 312
- Characteristic Equations 316
- PS/NS Tables 316
- K-Maps for Sequential Logic Circuits 317
- State Diagrams and ASM Charts 319
- Timing Diagrams 322

5.1.2 Design of Basic Asynchronous Sequential Logic Circuits 323

- Basic Latch Circuit Design 323
- Clock (Oscillator) Circuit Design 324

EXERCISES TO TEST YOUR KNOWLEDGE 325

5.2 Design of Gated Sequential Logic Circuits 326

5.2.1 Gated S-R Latch Circuit Design 327

5.2.2 Gated D Latch Circuit Design 328

EXERCISES TO TEST YOUR KNOWLEDGE 331

5.3 Flip-Flop Circuits 331

5.3.1 The Basic Idea of Edge- Triggered Flip-Flops 332

5.3.2 The Basic Idea of Pulse-Triggered Flip-Flops 335

5.3.3 Other Flip-Flop Types 337

5.3.4 Designing Simple Synchronous Counter Circuits with T Flip-Flops 340

EXERCISES TO TEST YOUR KNOWLEDGE 342

5.4 Analyzing Circuits That Use Flip-Flops and Latches 345

5.4.1 Moore-, Mealy-, and Mixed-Type Synchronous State Machines 345

5.4.2 Analyzing Simple Counter Circuits 346

5.4.3 Analyzing Synchronous Systems in General 350

5.4.4 Analyzing IC Counter Circuits via Dependency Notation 356

5.4.5 Designing Shortened Length Counters 360

5.4.6 Reducing Metastability in Synchronous Systems 361

5.4.7 Debouncing Switches with Latches 363

EXERCISES TO TEST YOUR KNOWLEDGE 366

5.5 Case Studies 368
5.5.1 Case Study Number 3 (KeypadDisplay System with Storage) 368
5.5.2 Case Study Number 4 (MultiplexedDisplay System) 372
5.5.3 Case Study Number 5 (Frequency Division for Digital Watch Circuit) 377
5.6 Worked Exercises 381
Problems 404

6 Introduction to Sequential Circuit Design 413

6.1 Synchronous Finite State Machine Design 413
6.1.1 Classical Design with D Flip-Flops 414
• Obtaining D Excitation Input Equations from K-Maps 415
• Determining the Maximum Clock Frequency of a Synchronous State Machine 424
• Functional Simulation of D Flip-Flop Circuits 427
6.1.2 Classical Design with T Flip-Flops 431
• Obtaining *T* Excitation Input Equations from K-maps 432
• Functional Simulation of T Flip-Flop Circuits 434
6.1.3 Classical Design with J-K Flip-Flops 435
• Obtaining *J-K* Excitation Input Equations from K-maps 436
• Functional Simulation of J-K Flip-Flop Circuits 437

EXERCISES TO TEST YOUR KNOWLEDGE 438

6.2 State Assignment Encoding, Shift Register Counters, and Adding an Enable Input 441
6.2.1 Contrasting Full-Encoded and One-Hot Encoded State Assignments 441
• Power-On Reset Circuits 441
• Circuit Design for Full Encoding vs. One-Hot Encoding 443
6.2.2 Additional Types of Counters (Shift Register Counters) 446
• Ring Counters 447
• Twisted Ring Counters 449
• Linear Feedback Shift Register Counters 451
6.2.3 Adding an Enable Input 453
• Full-Encoded Stoppable Counters 453
• One-Hot Encoded Stoppable Counters 455
• Flip-Flop Connections for One-Hot Encoding 458
• Designing with Flip-Flops That Contain a Synchronous Enable Input 459

EXERCISES TO TEST YOUR KNOWLEDGE 46 1

6.3 Newer Design Methods for Finite State Machines 463

6.3.1 Set OR Hold 1 Method—for Obtaining D Excitation Equations 463

- Obtaining *D* Excitation Equations Directly from State Diagrams, ASM Charts, or Timing Diagrams 464

6.3.2 Set OR Clear Method—for Obtaining T Excitation Equations 468

- Obtaining *T* Excitation Equations Directly from State Diagrams, ASM Charts, or Timing Diagrams 469

6.3.3 Set-Clear Method—for Obtaining J-K Excitation Equations 471

- Obtaining *J-K* Excitation Equations Directly from State Diagrams, ASM Charts, or Timing Diagrams 472

EXERCISES TO TEST YOUR KNOWLEDGE 474

6.4 Case Studies 481

6.4.1 Case Study 6 (Synchronous Vending Machine Controllers) 481

Design Overview 481
Full-Encoded 1-Coin Machine (FE1CM) 482
One-Hot Encoded 1 -Coin Machine (OHE1 CM) 484
Full-Encoded 2-Coin Machine (FE2CM) 485
One-Hot Encoded 2-Coin Machine (OHE2CM) 486
Full-Encoded 3-Coin Machine (FE3CM) 489
Graphical Entry of State Diagrams Using StateCAD® and StateBench™ 491

6.4.2 Case Study 7 (Fundamental-Mode (Clockless) Vending Machine Controllers) 494

- Synchronous and Asynchronous Design Differences 494
- Fundamental-Mode (Clockless) 1-Coin Machine (FM1CM) 495
- Fundamental-Mode (Clockless) 2-Coin Machine (FM2CM) 496
- Essential Hazards 500

6.4.3 Case Study 8 (Pulse-Mode (Clockless) Vending Machine Controllers) 501

- Input Signal Restrictions for Pulse-Mode Circuits 501
- Pulse-Mode (Clockless) 1-Coin Machine (PM1CM) 502
- Pulse-Mode (Clockless) 2-Coin Machine (PM2CM) 503
- Pulse-Mode (Clockless) 3-Coin Machine (PM3CM) 505

6.5 Worked Exercises 507

Problems 557

PART III HDL CHIP DESIGN

7 VHDL Synthesis by Example 571

7.1 Writing VHDL Source Code for Combinational Logic Circuits 571

7.1.1 Different VHDL Architectures for the XOR function 573

- Behavioral Architecture 573
- Data-Flow Architecture 574
- Structural Architecture 575

7.1.2 VHDL Source Code for Additional Combinational Logic Circuits 578

- Majority Function 578
- Two-out-of-Three Event Detector 579
- Two-to-Four Decoder 579
- Four-to-One MUX 580
- Full Adder 581
- Four-Bit Adder with a Carry In and a Carry Out 581
- Simple Arithmetic Logic Unit 582

7.1.3 Hierarchical Design Using a Top-Level Schematic 583

- **Hierarchical Design for the XOR Function 583**

EXERCISES TO TEST YOUR KNOWLEDGE 585

7.2 Writing VHDL Source Code for Synchronous Logic Circuits 587

7.2.1 Templates of VHDL Architectures 587

- Transparent D Latch vs. D Flip-Flop 587
- Stoppable 1-Bit Binary Up Counter with Moore and Mealy Outputs 589

7.2.2 VHDL Source Code for Additional Synchronous Logic Circuits 592

- Two-Bit Binary Up Counter (Full Encoded) 592
- Two-Bit Binary Up Counter (One-Hot Encoded) 594
- Synchronous Mealy-Type State Machine, M2 596
- Three-Bit Ring Counter 599
- Full-Encoded 2-Coin Machine (FE2CM) 601

EXERCISES TO TEST YOUR KNOWLEDGE 603

7.3 Summary 605

7.4 Worked Exercises 605

Problems 624

APPENDIICES

A. Digital Circuit Types 631

B. Selected Digital Logic ICs with Pinout Diagrams 634

C. Selected Programmable Logic Devices 638

D. Bibliography 657

Index 661