1

Verilog —

A Tutorial Introduction

Getting Started
A Structural Description 2
Simulating the binaryToESeg Driver 4
Creating Ports For the Module 7
Creating a Testbench For a Module 8
Behavioral Modeling of Combinational Circuits 11
Procedural Models 12
Rules for Synthesizing Combinational Circuits 13
Procedural Modeling of Clocked Sequential Circuits 14
Modeling Finite State Machines 15
Rules for Synthesizing Sequential Systems 18
Non-Blocking Assignment ("<=") 19
Module Hierarchy
The Counter 21
A Clock for the System 21
Tying the Whole Circuit Together 22
Tying Behavioral and Structural Models Together 25
Summary 27
Exercises 28

2

Logic Synthesis 35

Overview of Synthesis 35
Register-Transfer Level Systems 35
Disclaimer 36
Combinational Logic Using Gates and Continuous Assign 37
Procedural Statements to Specify Combinational Logic 40
The Basics 40
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complications — Inferred Latches</td>
<td>42</td>
</tr>
<tr>
<td>Using Case Statements</td>
<td>43</td>
</tr>
<tr>
<td>Specifying Don’t Care Situations</td>
<td>44</td>
</tr>
<tr>
<td>Procedural Loop Constructs</td>
<td>46</td>
</tr>
<tr>
<td><strong>Inferring Sequential Elements</strong></td>
<td>48</td>
</tr>
<tr>
<td>Latch Inferences</td>
<td>48</td>
</tr>
<tr>
<td>Flip Flop Inferences</td>
<td>50</td>
</tr>
<tr>
<td>Summary</td>
<td>52</td>
</tr>
<tr>
<td><strong>Inferring Tri-State Devices</strong></td>
<td>52</td>
</tr>
<tr>
<td><strong>Describing Finite State Machines</strong></td>
<td>53</td>
</tr>
<tr>
<td>An Example of a Finite State Machine</td>
<td>53</td>
</tr>
<tr>
<td>An Alternate Approach to FSM Specification</td>
<td>56</td>
</tr>
<tr>
<td><strong>Finite State Machine and Datapath</strong></td>
<td>58</td>
</tr>
<tr>
<td>A Simple Computation</td>
<td>58</td>
</tr>
<tr>
<td>A Datapath For Our System</td>
<td>58</td>
</tr>
<tr>
<td>Details of the Functional Datapath Modules</td>
<td>60</td>
</tr>
<tr>
<td>Wiring the Datapath Together</td>
<td>61</td>
</tr>
<tr>
<td>Specifying the FSM</td>
<td>63</td>
</tr>
<tr>
<td><strong>Summary on Logic Synthesis</strong></td>
<td>66</td>
</tr>
<tr>
<td><strong>Exercises</strong></td>
<td>68</td>
</tr>
</tbody>
</table>

## 3 Behavioral Modeling

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Process Model</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>If-Then-Else</strong></td>
<td>75</td>
</tr>
<tr>
<td>Where Does The ELSE Belong?</td>
<td>80</td>
</tr>
<tr>
<td>The Conditional Operator</td>
<td>81</td>
</tr>
<tr>
<td><strong>Loops</strong></td>
<td>82</td>
</tr>
<tr>
<td>Four Basic Loop Statements</td>
<td>82</td>
</tr>
<tr>
<td>Exiting Loops on Exceptional Conditions</td>
<td>85</td>
</tr>
<tr>
<td><strong>Multi-way Branching</strong></td>
<td>86</td>
</tr>
<tr>
<td>If-Else-If</td>
<td>86</td>
</tr>
<tr>
<td>Case</td>
<td>86</td>
</tr>
<tr>
<td>Comparison of Case and If-Else-If</td>
<td>89</td>
</tr>
<tr>
<td>Casez and Casex</td>
<td>90</td>
</tr>
<tr>
<td><strong>Functions and Tasks</strong></td>
<td>91</td>
</tr>
<tr>
<td>Tasks</td>
<td>93</td>
</tr>
<tr>
<td>Functions</td>
<td>97</td>
</tr>
<tr>
<td>A Structural View</td>
<td>100</td>
</tr>
<tr>
<td><strong>Rules of Scope and Hierarchical Names</strong></td>
<td>102</td>
</tr>
<tr>
<td>Rules of Scope</td>
<td>102</td>
</tr>
<tr>
<td>Hierarchical Names</td>
<td>105</td>
</tr>
</tbody>
</table>
Summary 106
Exercises 106

4 Concurrent Processes 109

Concurrent Processes 109
Events 111
Event Control Statement 112
Named Events 113
The Wait Statement 116
A Complete Producer-Consumer Handshake 117
Comparison of the Wait and While Statements 120
Comparison of Wait and Event Control Statements 121
A Concurrent Process Example 122
A Simple Pipelined Processor 128
The Basic Processor 128
Synchronization Between Pipestages 130
Disabling Named Blocks 132
Intra-Assignment Control and Timing Events 134
Procedural Continuous Assignment 136
Sequential and Parallel Blocks 138
Exercises 140

5 Module Hierarchy 143

Module Instantiation and Port Specifications 143
Parameters 146
Arrays of Instances 150
Generate Blocks 151
Exercises 154
6

Logic Level Modeling 157

Introduction 157
Logic Gates and Nets 158
Modeling Using Primitive Logic Gates 159
Four-Level Logic Values 162
Nets 163
A Logic Level Example 166
Continuous Assignment 171
Behavioral Modeling of Combinational Circuits 172
Net and Continuous Assign Declarations 174
A Mixed Behavioral/Structural Example 176
Logic Delay Modeling 180
A Gate Level Modeling Example 181
Gate and Net Delays 182
Specifying Time Units 185
Minimum, Typical, and Maximum Delays 186
Delay Paths Across a Module 187
Summary of Assignment Statements 189
Summary 190
Exercises 191

7

Cycle-Accurate Specification 195

Cycle-Accurate Behavioral Descriptions 195
Specification Approach 195
A Few Notes 197
Cycle-Accurate Specification 198
Inputs and Outputs of an Always Block 198
Input/Output Relationships of an Always Block 199
Specifying the Reset Function 202
Mealy/Moore Machine Specifications 203
A Complex Control Specification 204
Data and Control Path Trade-offs 204
Introduction to Behavioral Synthesis 209
Summary 210
8

Advanced Timing

Verilog Timing Models 211
Basic Model of a Simulator 214
Gate Level Simulation 215
Towards a More General Model 215
Scheduling Behavioral Models 218
Non-Deterministic Behavior of the Simulation Algorithm 220
Near a Black Hole 221
It's a Concurrent Language 223
Non-Blocking Procedural Assignments 226
Contrasting Blocking and Non-Blocking Assignments 226
Prevalent Usage of the Non-Blocking Assignment 227
Extending the Event-Driven Scheduling Algorithm 228
Illustrating Non-Blocking Assignments 231
Summary 233
Exercises 234

9

User-Defined Primitives 239

Combinational Primitives 240
Basic Features of User-Defined Primitives 240
Describing Combinational Logic Circuits 242
Sequential Primitives 243
Level-Sensitive Primitives 244
Edge-Sensitive Primitives 244
Shorthand Notation 246
Mixed Level- and Edge-Sensitive Primitives 246
Summary 249
Exercises 249
10 Switch Level Modeling 251

A Dynamic MOS Shift Register Example 251
Switch Level Modeling 256
  Strength Modeling 256
  Strength Definitions 259
  An Example Using Strengths 260
  Resistive MOS Gates 262
Ambiguous Strengths 263
  Illustrations of Ambiguous Strengths 264
  The Underlying Calculations 265
The miniSim Example 270
  Overview 270
  The miniSim Source 271
  Simulation Results 280
Summary 281
Exercises 281

11 Projects 283

Modeling Power Dissipation 283
  Modeling Power Dissipation 284
  What to do 284
  Steps 285
A Floppy Disk Controller 286
  Introduction 286
  Disk Format 287
  Function Descriptions 288
  Reality Sets In... 291
  Everything You Always Wanted to Know about CRC's 291
  Supporting Verilog Modules 292

Appendix A: Tutorial Questions and Discussion 293
  Structural Descriptions 293
  Testbench Modules 303
  Combinational Circuits Using always 303
<table>
<thead>
<tr>
<th>Source text</th>
<th>343</th>
</tr>
</thead>
<tbody>
<tr>
<td>Declarations</td>
<td>346</td>
</tr>
<tr>
<td>Primitive instances</td>
<td>351</td>
</tr>
<tr>
<td>Module and generated instantiation</td>
<td>353</td>
</tr>
<tr>
<td>UDP declaration and instantiation</td>
<td>355</td>
</tr>
<tr>
<td>Behavioral statements</td>
<td>355</td>
</tr>
<tr>
<td>Specify section</td>
<td>359</td>
</tr>
<tr>
<td>Expressions</td>
<td>365</td>
</tr>
<tr>
<td>General</td>
<td>370</td>
</tr>
<tr>
<td>Index</td>
<td>373</td>
</tr>
</tbody>
</table>