Verilog® HDL

A Guide to Digital Design and Synthesis

Samir Palnitkar

SunSoft Press
A Prentice Hall Title

Table of Contents

About the Author v
Foreword xxxi
Preface xxxiii
Acknowledgments xxxvii

Part 1: Basic Verilog Topics 1

1. Overview of Digital Design with Verilog HDL 3
1.1 Evolution of Computer Aided Digital Design 3
1.2 Emergence of HDLs 4
1.3 Typical Design Flow 5
1.4 Importance of HDLs 6
1.5 Popularity of Verilog HDL 7
1.6 Trends in HDLs 8

2. Hierarchical Modeling Concepts 11
2.1 Design Methodologies 11
2.2 4-bit Ripple Carry Counter 13
2.3 Modules 15
2.4 Instances 17
2.5 Components of a Simulation 18
2.6 Example 20
2.6.1 Design Block 20
2.6.2 Stimulus Block 22
2.7 Summary 24
2.8 Exercises 25

3. Basic Concepts 27
3.1 Lexical Conventions 27
3.1.1 Whitespace 27
3.1.2 Comments 28
3.1.3 Operators 28

3.1.4 Number Specification.......28
Sized numbers.......28
Unsized numbers.......29
X or Z values.......29
Negative numbers.......30
Underscore characters and question marks.......30
3.1.5 Strings.......30
3.1.6 Identifiers and Keywords.......30
3.1.7 Escaped Identifiers.......31
3.2 Data Types.......31
3.2.1 Value Set.......31
3.2.2 Nets.......32
3.2.3 Registers.......33
3.2.4 Vectors.......34
3.2.5 Integer , Real, and Time Register Data Types.......34
Integer.......34
Real.......35
Time.......35
3.2.6 Arrays.......36
3.2.7 Memories.......36
3.2.8 Parameters.......37
3.2.9 Strings.......37
3.3 System Tasks and Compiler Directives.......38
3.3.1 System Tasks.......38
Displaying information.......38
Monitoring information.......40
Stopping and finishing in a simulation.......41
3.3.2 Compiler Directives.......42
`define.......42
`include.......43
3.4 Summary.......43
3.5 Exercises.......44
4. Modules and Ports.......47
4.1 Modules.......47
4.2 Ports.......51
4.2.1 List of Ports.......51
4.2.2 Port Declaration.......52
4.2.3 Port Connection Rules.......53

Inputs . . . 53
Outputs . . . 54
Inouts . . . 54
Width matching . . . 54
Unconnected ports . . . 54
Example of illegal port connection . . . 54
4.2.4 Connecting Ports to External Signals . . . 55
Connecting by ordered list . . . 55
Connecting ports by name . . . 56
4.3 Hierarchical Names . . . 57
4.4 Summary . . . 58
4.5 Exercises . . . 59
5. Gate-Level Modeling . . . 61
5.1 Gate Types . . . 62
5.1.1 And/Or Gates . . . 62
5.1.2 Buf/Not Gates . . . 64
Bufif/notif . . . 66
5.1.3 Examples . . . 68
Gate-level multiplexer . . . 68
4-bit full adder . . . 72
5.2 Gate Delays . . . 76
5.2.1 Rise, Fall, and Turn-off Delays . . . 76
Rise delay . . . 76
Fall delay . . . 76
Turn-off delay . . . 76
5.2.2 Min/Typ/Max Values . . . 77
Min value . . . 77
Typ val . . . 77
Max value . . . 78
5.2.3 Delay Example . . . 79
5.3 Summary . . . 81
5.4 Exercises . . . 82
6. Dataflow Modeling . . . 85
6.1 Continuous Assignments . . . 86
6.1.1 Implicit Continuous Assignment . . . 87
6.2 Delays . . . 88
6.2.1 Regular Assignment Delay . . . 88

6.2.2 Implicit Continuous Assignment Delay 89
6.2.3 Net Declaration Delay 89
6.3 Expressions, Operators, and Operands 90
6.3.1 Expressions 90
6.3.2 Operands 91
6.3.3 Operators 91
6.4 Operator Types 92
6.4.1 Arithmetic Operators 93
Binary operators 93
Unary operators 94
6.4.2 Logical Operators 94
6.4.3 Relational Operators 95
6.4.4 Equality Operators 96
6.4.5 Bitwise Operators 97
6.4.6 Reduction Operators 98
6.4.7 Shift Operators 99
6.4.8 Concatenation Operator 99
6.4.9 Replication Operator 100
6.4.10 Conditional Operator 100
6.4.11 Operator Precedence 101
6.5 Examples 102
6.5.1 4-to-1 Multiplexer 102
Method 1: logic equation 102
Method 2: conditional operator 103
6.5.2 4-bit Full Adder 104
Method 1: dataflow operators 104
Method 2: full adder with carry lookahead 105
6.5.3 Ripple Counter 106
6.6 Summary 112
6.7 Exercises 112

7. Behavioral Modeling 115
7.1 Structured Procedures 116
7.1.1 initial Statement 116
7.1.2 always Statement 118
7.2 Procedural Assignments 119
7.2.1 Blocking assignments 119
7.2.2 Nonblocking Assignments 120
Application of nonblocking assignments 122
7.3 Timing Controls 124

7.3.1 Delay-Based Timing Control . . . 124
Regular delay control . . . 125
Intra-assignment delay control . . . 126
Zero delay control . . . 127
7.3.2 Event-Based Timing Control . . . 127
Regular event control . . . 128
Named event control . . . 128
Event OR control . . . 129
7.3.3 Level-Sensitive Timing Control . . . 129
7.4 Conditional Statements . . . 130
7.5 Multiway Branching . . . 131
7.5.1 case Statement . . . 132
7.5.2 casex, casez Keywords . . . 134
7.6 Loops . . . 135
7.6.1 While Loop . . . 135
7.6.2 For Loop . . . 137
7.6.3 Repeat Loop . . . 138
7.6.4 Forever loop . . . 139
7.7 Sequential and Parallel Blocks . . . 140
7.7.1 Block Types . . . 140
Sequential blocks . . . 140
Parallel blocks . . . 141
7.7.2 Special Features of Blocks . . . 143
Nested blocks . . . 143
Named blocks . . . 143
Disabling named blocks . . . 144
7.8 Examples . . . 145
7.8.1 4-to-1 Multiplexer . . . 145
7.8.2 4-bit Counter . . . 146
7.8.3 Traffic Signal Controller . . . 147
Specification . . . 147
Verilog description . . . 149
Stimulus . . . 151
7.9 Summary . . . 153
7.10 Exercises . . . 154

8. Tasks and Functions . . . 157
8.1 Differences Between Tasks and Functions . . . 157
8.2 Tasks . . . 158

8.2.1 Task Declaration and Invocation 159
8.2.2 Task Examples 160
Use of input and output arguments 160
Asymmetric Sequence Generator 161
8.3 Functions 162
8.3.1 Function Declaration and Invocation 162
8.3.2 Function Examples 163
Parity calculation 164
Left/right shifter 165
8.4 Summary 166
8.5 Exercises 166

9. Useful Modeling Techniques 169
9.1 Procedural Continuous Assignments 169
9.1.1 assign and deassign 169
9.1.2 force and release 171
force and release on registers 171
force and release on nets 172
9.2 Overriding Parameters 172
9.2.1 defparam Statement 172
9.2.2 Module_Instance Parameter Values 173
9.3 Conditional Compilation and Execution 175
9.3.1 Conditional Compilation 175
9.3.2 Conditional Execution 176
9.4 Time Scales 177
9.5 Useful System Tasks 179
9.5.1 File Output 179
Opening a file 179
Writing to files 180
Closing files 180
9.5.2 Displaying Hierarchy 181
9.5.3 Strobing 182
9.5.4 Random Number Generation 182
9.5.5 Initializing Memory from File 183
9.5.6 Value Change Dump File 185
9.6 Summary 186
9.7 Exercises 188

Part 2: Advanced Verilog Topics 191

10. Timing and Delays 193
10.1 Types of Delay Models 194
10.1.1 Distributed Delay 194
10.1.2 Lumped Delay 195
10.1.3 Pin-to-Pin Delays 196
10.2 Path Delay Modeling 197
10.2.1 Specify Blocks 198
10.2.2 Inside Specify Blocks 199
Parallel connection 199
Full connection 200
specparam statements 201
Conditional path delays 202
Rise, fall, and turn-off delays 203
Min, max, and typical delays 204
Handling x transitions 205
10.3 Timing Checks 205
10.3.1 $setup and $hold checks 206
$setup task 206
$hold task 207
10.3.2 $width Check 207
10.4 Delay Back-Annotation 208
10.5 Summary 210
10.6 Exercises 211

11. Switch-Level Modeling 213
11.1 Switch-Modeling Elements 213
11.1.1 MOS Switches 214
11.1.2 CMOS Switches 215
11.1.3 Bidirectional Switches 216
11.1.4 Power and Ground 217
11.1.5 Resistive Switches 218
11.1.6 Delay Specification on Switches 219
MOS and CMOS switches 219
Bidirectional pass switches 220
Specify blocks 220
11.2 Examples 220
11.2.1 CMOS Nor Gate 220

11.2.2 2-to-1 Multiplexer 223
11.2.3 Simple CMOS Flip-Flop 224
11.3 Summary 226
11.4 Exercises 227

12. User-Defined Primitives 229
12.1 UDP basics 229
12.1.1 Parts of UDP Definition 230
12.1.2 UDP Rules 231
12.2 Combinational UDPs 231
12.2.1 Combinational UDP Definition 231
12.2.2 State Table Entries 232
12.2.3 Shorthand Notation for Don't Cares 234
12.2.4 Instantiating UDP Primitives 234
12.2.5 Example of a Combinational UDP 235
12.3 Sequential UDPs 238
12.3.1 Level-Sensitive Sequential UDPs 239
12.3.2 Edge-Sensitive Sequential UDPs 240
12.3.3 Example of a Sequential UDP 242
12.4 UDP Table Shorthand Symbols 244
12.5 Guidelines for UDP Design 245
12.6 Summary 246
12.7 Exercises 247

13. Programming Language Interface 249
13.1 Uses of PLI 251
13.2 Linking and Invocation of PLI Tasks 252
13.2.1 Linking PLI Tasks 252
Linking PLI in Verilog-XL 253
Linking in VCS 254
13.2.2 Invoking PLI Tasks 254
13.2.3 General Flow of PLI Task Addition and Invocation 255
13.3 Internal Data Representation 255
13.4 PLI Library Routines 259
13.4.1 Access Routines 259
Mechanics of access routines 260
Types of access routines 261
Examples of access routines 261
13.4.2 Utility Routines 268
Mechanics of utility routines 268

Types of utility routines 268
Example of utility routines........ 269
13.5 Summary........ 272
13.6 Exercises 274

14. Logic Synthesis with Verilog HDL 275
14.1 What Is Logic Synthesis?........ 275
14.2 Impact of Logic Synthesis 278
14.3 Verilog HDL Synthesis 280
14.3.1 Verilog Constructs 280
14.3.2 Verilog Operators 281
14.3.3 Interpretation of a Few Verilog Constructs........ 283
The assign statement........ 283
The if-else statement........ 284
The case statement 285
for loops 285
The function statement........ 286
14.4 Synthesis Design Flow........ 287
14.4.1 RTL to Gates 287
RTL description........ 288
Translation 288
Unoptimized intermediate representation........ 288
Logic optimization 288
Technology mapping and optimization........ 288
Technology library 289
Design constraints........ 289
Optimized gate-level description 290
14.4.2 An Example of RTL-to-Gates........ 291
Design specification 291
RTL description........ 291
Technology library 292
Design constraints........ 293
Logic synthesis 293
Final, Optimized, Gate-Level Description........ 293
IC Fabrication 296
14.5 Verification of Gate-Level Netlist........ 296
14.5.1 Functional Verification........ 296
Timing verification 299

14.6 Modeling Tips for Logic Synthesis 299
14.6.1 Verilog Coding Style 299
Use meaningful names for signals and variables 299
Avoid mixing positive and negative edge-triggered flip-flops 300
Use basic building blocks vs. Use continuous assign statements 300
Instantiate multiplexers vs. Use if-else or case statements 301
Use parentheses to optimize logic structure 301
Use arithmetic operators *, /, and % vs. Design building blocks 301
Be careful with multiple assignments to the same variable 302
Define if-else or case statements explicitly 302
14.6.2 Design Partitioning 303
Horizontal partitioning 303
Vertical Partitioning 304
Parallelizing design structure 305
14.6.3 Design Constraint Specification 306
14.7 Example of Sequential Circuit Synthesis 306
14.7.1 Design Specification 306
14.7.2 Circuit Requirements 306
14.7.3 Finite State Machine (FSM) 307
14.7.4 Verilog Description 307
14.7.5 Technology Library 310
14.7.6 Design Constraints 311
14.7.7 Logic Synthesis 311
14.7.8 Optimized Gate-Level Netlist 311
14.7.9 Verification 314
14.8 Summary 316
14.9 Exercises 317

Part 3: Appendices 319

A. Strength Modeling and Advanced Net Definitions 321
A.1 Strength Levels 321
A.2 Signal Contention 322
A.2.1 Multiple Signals with Same Value and Different Strength 322
A.2.2 Multiple Signals with Opposite Value and Same Strength 322
A.3 Advanced Net Types 322
A.3.1 tri 322
A.3.2 trireg 323

A.3.3 tri0 and tri1 . . . 324
A.3.4 supply0 and supply1 . . . 324
A.3.5 wor, wand, trior, and triand . . . 324

B. List of PLI Routines . . . 327
B.1 Conventions . . . 327
B.2 Access Routines . . . 327
B.2.1 Handle Routines . . . 327
B.2.2 Next Routines . . . 329
B.2.3 Value Change Link (VCL) Routines . . . 331
B.2.4 Fetch Routines . . . 331
B.2.5 Utility Access Routines . . . 334
B.2.6 Modify Routines . . . 335
B.3 Utility (tf_) Routines . . . 336
B.3.1 Get Calling Task/Function Information . . . 336
B.3.2 Get Argument List Information . . . 336
B.3.3 Get Parameter Values . . . 337
B.3.4 Put Parameter Value . . . 337
B.3.5 Monitor Parameter Value Changes . . . 338
B.3.6 Synchronize Tasks . . . 338
B.3.7 Long Arithmetic . . . 339
B.3.8 Display Messages . . . 340
B.3.9 Miscellaneous Utility Routines . . . 340
B.3.10 Housekeeping Tasks . . . 341

C. List of Keywords, System Tasks, and Compiler Directives . . . 343
C.1 Keywords . . . 343
C.2 System Tasks and Functions . . . 344
C.3 Compiler Directives . . . 344

D. Formal Syntax Definition . . . 345
D.1 Source Text . . . 346
D.2 Declarations . . . 349
D.3 Primitive Instances . . . 351
D.4 Module Instantiations . . . 352
D.5 Behavioral Statements . . . 353
D.6 Specify Section . . . 355
D.7 Expressions . . . 359

D.8 General 361

E. Verilog Tidbits 363

Origins of Verilog HDL 363

Open Verilog International (OVI) 363

Interpreted, Compiled, Native Compiled Simulators 363

Event-Driven Simulation, Oblivious Simulation 364

Cycle-Based Simulation 364

Fault Simulation 364

Verilog Newsgroup 364

Verilog FTP Site 365

Verilog Simulators 365

Verilog Related Mosaic Sites 365

F. Verilog Examples 367

F.1 Synthesizable FIFO Model 367

Input ports 368

Output ports 368

F.2 Behavioral DRAM Model 376

Input ports 376

Inout ports 376

Bibliography 381

Index 383