



Verilog® Introduction

Student Workbook

© 2011 Mentor Graphics Corporation
All rights reserved.

This document contains information that is proprietary to Mentor Graphics Corporation. The original recipient of this document may duplicate this document in whole or in part for internal business purposes only, provided that this entire notice appears in all copies. In duplicating any part of this document, the recipient agrees to make every reasonable effort to prevent the unauthorized use and distribution of the proprietary information.

This document is for information and instruction purposes. Mentor Graphics reserves the right to make changes in specifications and other information contained in this publication without prior notice, and the reader should, in all cases, consult Mentor Graphics to determine whether any changes have been made.

The terms and conditions governing the sale and licensing of Mentor Graphics products are set forth in written agreements between Mentor Graphics and its customers. No representation or other affirmation of fact contained in this publication shall be deemed to be a warranty or give rise to any liability of Mentor Graphics whatsoever.

MENTOR GRAPHICS MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

MENTOR GRAPHICS SHALL NOT BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS PUBLICATION OR THE INFORMATION CONTAINED IN IT, EVEN IF MENTOR GRAPHICS CORPORATION HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

RESTRICTED RIGHTS LEGEND 03/97

U.S. Government Restricted Rights. The SOFTWARE and documentation have been developed entirely at private expense and are commercial computer software provided with restricted rights. Use, duplication or disclosure by the U.S. Government or a U.S. Government subcontractor is subject to the restrictions set forth in the license agreement provided with the software pursuant to DFARS 227.7202-3(a) or as set forth in subparagraph (c)(1) and (2) of the Commercial Computer Software - Restricted Rights clause at FAR 52.227-19, as applicable.

Contractor/manufacturer is:

Mentor Graphics Corporation

8005 S.W. Boeckman Road, Wilsonville, Oregon 97070-7777.

Telephone: 503.685.7000

Toll-Free Telephone: 800.592.2210

Website: www.mentor.com

SupportNet: supportnet.mentor.com/

Send Feedback on Documentation: supportnet.mentor.com/doc_feedback_form

TRADEMARKS: The trademarks, logos and service marks ("Marks") used herein are the property of Mentor Graphics Corporation or other third parties. No one is permitted to use these Marks without the prior written consent of Mentor Graphics or the respective third-party owner. The use herein of a third-party Mark is not an attempt to indicate Mentor Graphics as a source of a product, but is intended to indicate a product from, or associated with, a particular third party. A current list of Mentor Graphics' trademarks may be viewed at: www.mentor.com/trademarks.

End-User License Agreement: You can print a copy of the End-User License Agreement from: www.mentor.com/eula.

Table of Contents

Module 1	
Introduction	11
Objectives	11
Course Outline	12
Recommended Verilog Texts	13
(S)Verilog Design Flow	14
Behavioral Model	15
RTL Model	16
Synthesize or Simulate?	17
Gate-Level Model	18
What is SystemVerilog?	19
System-Level Simulation	20
The Verilog Language	21
Verilog History	22
Verilog Extensions	23
What is a Hardware Description Language?	24
Simulation Environment	25
What is Synthesis?	26
Basic Modeling Structure	27
An Example	28
2:1 MUX – Verilog Example	29
Module 2	
Lexical Conventions	31
Objectives	31
Lexical Conventions	32
Verilog Keywords	33
White Space	34
Comments	35
Strings	36
Identifiers	37
Escaped Identifiers	38
Case Sensitivity	39
Logic Values	40
Integers and Real Numbers	41
Integers and Real Numbers Examples	43
System Tasks and Functions	44
Text Formatting Codes	45
Compiler Directives	46
‘timescale Compiler Directive	48
ModelSim Introduction Lab	49

Module 3	
Data Types	51
Objectives	51
Data Types	52
Nets	53
Variables	54
Conflict Resolution	55
Arrays	56
Array Indices	57
Memories	58
Example — Multi-Dimension Array	59
Accessing Memories	60
Indexed Vector Part-Selects	61
Parameters	62
Module 4	
Basic Code Structure	63
Objectives	63
Basic Code Structure	64
Verilog Module	65
Ports	66
Module Declarations	67
Module Instances	68
Example Instantiation	69
Module Hierarchy	70
Quiz	71
Parameterized Module (Verilog 2001)	72
Parameterized Module (Verilog 2001) Example	73
Parameterized Module (Verilog 95)	74
Parameterized Module (Verilog 95) Example	75
Using the Ports of a Module	76
Port Connection Rules	77
Quiz	78
Precision Synthesis Tutorial	79
Module 5	
Procedural Assignments	81
Objectives	81
Procedural Assignments	82
Procedural Blocks	83
Procedural Statement Groups	84
Timing Controls (Procedural Delays)	85
Verilog Scheduler — Events and Processes	86
Time and Event Queues	87
Zero Time Loops	88
”Deadlocks”	89
Verilog Scheduler — Event Queue Segments	90
Procedural Assignments	91

Table of Contents

Simple buf Example.	92
Blocking Assignments	93
Simple buf: Blocking Assignment.	94
Delayed Blocking Assignments.	95
Simple buf: Delayed Blocking Assignment.	97
Blocking Intra-Procedural Delayed Assignment	98
Simple buf: Blocking Intra-Procedural Delay	99
Quiz — Question 1	101
Quiz — Question 2	102
Non-Blocking Assignments.	103
Race Solved!	104
Swap Accomplished!	105
Simple buf: Non-Blocking Assignment.	106
Do Not Mix Blocking and Non-Blocking	107
Delayed Non-Blocking Assignments.	108
Simple buf: Delayed Non-Blocking Assignment	109
Non-Blocking Intra-Procedural Delayed Assignments	110
Multiple Delayed NBAs	111
Simple buf: Non-Blocking Intra-Procedural Delay	112
Procedural Assignment Example.	113
pound.v	114
Procedural Blocks	115
Rules of Thumb -1-	116
Quiz	117
Rules of Thumb -2-	118
Lab 1: Pipelined Circuit.	119
Typical Test Fixture.	120
Test Fixture — test_pipe.v.	121
Correct Output — pipe.v	122
Lab (Part 2): Synthesize the Design	123
Blocking Assignments and Synthesis Quiz	124
Non-Blocking Assignments and Synthesis	125
Module 6	
Operators.	127
Objectives	127
Operators	128
Equality Operators	130
Operator Examples	131
Bit Lengths Resulting From Expressions	134
Signed Arithmetic	135
Integers or Variables?	136
Term Sharing	137
Resource Topology	138
Module 7	
Procedural Statements	139
Objectives	139

Procedural Statements	140
if/if-else	141
case	142
casez, casex	143
case Example	144
casex Example	145
casez Example	146
Xs in Synthesis	147
'case' and Synthesis	148
Which to Use: case or if-else?	149
Inferred Latches in Synthesis	150
Inferred Latch Warning	151
Avoiding Inferred Latches	152
Lab 2: Seven Segment Decoder	153
Test Fixture — test_seven_seg.v	154
Lab (Part 2): Synthesize the Design	155
forever	156
repeat	157
while	158
for	159
Ever See a Hardware for?	160
Naming Statement Groups	161
disable	162
Module 8	
Combinational Logic	163
Objectives	163
Combinational Logic	164
Sensitivity Lists	165
Sensitivity Lists and Verilog 2001	166
Lab 3: Sensitivity List Mux	167
test_mux_sens.v	168
Correct Output — Sensitivity List	169
Transport Delay! (aka Glitch Catcher)	170
Continuous Assignments	171
Implicit Continuous Assignments	172
Inertial Delay! (aka Glitch Swallower)	173
Lab 4: Continuous Assignment Mux	174
test_mux_cont.v	175
Correct Output — Continuous Assignment	176
Gate-Level Primitives	177
Module 9	
Procedure Calls	179
Objectives	179
Procedure Calls	180
User-Defined Functions	181
Function — Example	183

Table of Contents

Functions in Synthesis	184
Lab 5: Seven Segment Decoder Function	185
Test Fixture – test_7seg_func.v	186
Lab (Part 2): Synthesize the Design	187
Procedural Code	188
Alternative 7Seg Solutions	189
User-Defined Tasks	190
Example Task and Call	191
Example Test Fixture — test_pipe.v	192
Lab 6: 7-Segment Decoder Testbench Task	193
Testbench (test_seven_seg_func.v)	194
Example Output	195
Generate Blocks	196
for-generate Example	197
if-generate Example	198
Module 10	
Simulation Timing and Delay	199
Objectives	199
Timing Sources	200
Pre Synthesis	201
Post Synthesis — Specify Blocks	202
Standard Delay Format	204
SDF Files	205
ModelSim & SDF	206
Module 11	
Bidirectionals and Tristate	207
Objectives	207
z in Synthesis	208
Tri-State Buffer Inference — Sensitivity List	209
Tri-State Buffer Inference — Continuous Assignment	210
Tri-State Buffer on the Output of a Flip-Flop	211
Bi-Directional Ports and Tri-State Buses	213
Bi-Directional Ports and Tri-State Buses	214
Lab 7: Model a Behavioral SRAM	215
SRAM Timing	216
Test Fixture — test_beh_sram.v	217
Module 12	
Finite State Machines	219
Objectives	219
Finite State Machines	220
FSM — Implicit versus Explicit	221
Implicit FSM Style	222
Explicit FSM Style	223
State Machine Example — Key	224
Explicit FSM Style #1	225

Example A (Sequential Logic)	226
Example A (Combinational Logic)	227
Example B (Sequential Logic).....	228
Example B (Combinational Logic)	229
Explicit FSM Style #2	230
Example C (Sequential Logic).....	231
Example C (Combinational Logic)	232
One-Hot Encoding	233
Example D (Sequential Logic)	234
Example D (Combinational Logic)	235
Example E (Sequential Logic).....	236
Example E (Combinatorial Logic).....	237
State Encoding Pragmas	238
Lab 8: FSM	239
Hierarchy	240
Environment Hierarchy	241
Sample Instruction Flow	242
Suggested Flow Diagram.....	243
test_sm.v	244
Example Output.....	245
Lab (Part 2): Synthesize the Design	246
Lab (Part 3): Different Encodings (Optional)	247
Module 13	
Verification Cookbook	249
Objectives	249
Code Configuration	250
“define” and “ifdef” for Code Modification	251
Load Config Files Once.....	252
Source Code Libraries	253
‘uselib.....	254
Example ‘uselib Code	255
Library Mapping Files	256
Configuration Blocks.....	257
Configurations Example (Compiled Libraries)	258
Initializing Memories.....	259
\$readmem<h/b> Image Files	260
Hierarchical References.....	261
Where are Hierarchical References Used?	262
\$monitor and \$strobe	263
force and release	264
force/release.....	265
force Command in ModelSim	266
Using Specify Blocks in Behavioral Code	267
Bus Monitor Tasks.....	268
Monitor Example	269
Random Number Example.....	270
Writing to a File.....	271

Table of Contents

Write to a File Example	272
Verilog 2001 File IO Extensions	273
Testbench — Clock Generation	274
Testbench — Timeout Tests	275
Timing Offset Tests (Sweepers)	276
Timing Offset Example	277
Lab 9: Write a Testbench From Scratch	278
Lab: FIFO to Test	279
Lab 9: Suggested Strategy	280
Lab 9: Testbench Schematic	281
Appendix A	
ModelSim Tutorial	283
Objectives	283
Main Window	284
ModelSim Windows	285
Drag and Drop Feature	286
Context-Sensitive Menus	287
Customizing the GUI Layout	288
Customize the Main Window Layout	289
Customize Window Placement	290
Customize the Grouping of Windows	291
Starting ModelSim	292
Creating a Library	293
Compiling Files	294
Making Simple Source Edits	295
Compiling a File After Edits	296
Loading a Design for Simulation	297
Design Loaded for Simulation	298
Adding Signals to the Wave Window	299
Running the Simulation	300
Setting a Breakpoint in the Source	301
Stepping Through Source	302
Working With Cursors	303
Transcript Window and Script Files	305
Exiting ModelSim	306
Additional Information	307
Appendix B	
Precision Tutorial	311
Objectives	311
Precision Synthesis Tour	312
Starting Precision Synthesis	313
Precision Synthesis GUI	314
Basic Synthesis Flow Using Precision	315
Add Input Files in Precision GUI	316
Choosing a Technology for Your Design	317
Choosing a Specific Device for Your Design	318

Compiling and Synthesizing Your Design	320
Analyzing Synthesis Transcript	321
Analyzing Synthesis Messages	322
Analyzing Synthesis Results	323
Schematic Viewer	324
Schematic Navigation With Strokes	326
Technology Schematic	328
Critical Path	331
Critical Path Schematic	332
Critical Path Timing Report	333
Synthesis Reports	334
Synthesis Area Report	335
Synthesis Timing Report	336
Additional Synthesis Reports	337
Precision Synthesis Options	338
Place and Route Software	342
Exiting Precision Synthesis	344
Appendix C	
Generate Block Example	345
Objectives	345
Generate Example — 74ls170	346
Generate Example — 74ls170 detail	347
Generate Example — Verilog (1 of 2)	348
Generate Example — Verilog (2 of 2)	349
Appendix D	
Sample Solutions	351
Objectives	351
Lab 1: Sample Solution — pipe.v	352
Lab 2: Sample Solution — seven_seg.v	353
Lab 3: Sample Solution — mux_sens.v	354
Lab 4: Sample Solution — mux_cont.v	355
Lab 5: Sample Solution — seven_seg_func.v	356
Lab 6: Sample Solution — test_seven_seg_func.v	357
Lab 7: Sample Solution — beh_sram.v	358
Lab 8: Sample Solution — sm.v (1 of 2)	359
Lab 8: Sample Solution — sm.v (2 of 2)	360
Lab 8: Sample Solution — sm_seq.v	361
Lab 9: Sample Solution test_fifo.v (1 of 5)	362
Lab 9: Sample Solution test_fifo.v (2 of 5)	363
Lab 9: Sample Solution test_fifo.v (3 of 5)	364
Lab 9: Sample Solution test_fifo.v (4 of 5)	365
Lab 9: Sample Solution test_fifo.v (5 of 5)	366