



ModelSim® Advanced Topics

Student Workbook

© 1991-2011 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: Functional Verification Overview	11
Objectives	11
Module Overview	12
Digital Simulation Design Flow	13
Functional Verification	14
Digital Functional Verification	15
Functional Verification Techniques	16
Hardware Assisted Functional Verification	17
Digital Simulation Using ModelSim	18
Summary	20

Module 2

Tcl/Tk Overview	21
Objectives	21
Module Overview	22
Why Tcl/Tk?	23
Some Examples of Using Tcl/Tk in Debugging	24
Tcl Overview	25
Tk Overview	26
Tcl Commands	27
Basic Tcl Syntax	28
Tcl Variables	29
Command Substitution	30
Quotes	31
Curly Braces	32
Control Structures	33
Simulation Commands	34
Using ModelSim Tcl Commands to Add Signals to the Wave Window	35
Using ModelSim Tcl Commands to Organize Signals in the Wave Window	36
Grouping Signals in the Wave Window	37
Tcl Script Example	38
Creating a Simulation Script	39
Simulation Script Example	40
Tk Widget Overview	41
Tk Commands	42
Simple Tk Example	43
Adding a Customized Button to the Wave Window	44
Additional Tcl/Tk Resources	46
Summary	47
Lab 1: Tcl/Tk Application	48

Module 3	
Code Coverage	49
Objectives	49
Code Coverage	50
Purpose of Code Coverage	51
ModelSim Code Coverage	52
Types of Code Coverage	53
Unified Coverage Database (UCDB)	55
Coverage Modes	56
Statement Coverage	57
Branch Coverage	58
Expression Coverage	60
Condition Coverage	61
Toggle Coverage	63
Extended Toggle Coverage	64
Finite State Machine (FSM) Coverage	65
Invoking Code Coverage: Two-Step Flow	66
Invoking Code Coverage: Alternative Two-Step Flows	67
Invoking Code Coverage: Three-Step Flow	68
Union of Coverage Types	70
Enabling Code Coverage in the GUI, 2-Step Flow	71
Enabling Code Coverage in the GUI, 3-Step Flow	72
Code Coverage Windows in ModelSim	73
Code Coverage in the Structure Window	74
Code Coverage Analysis Window	75
Viewing Toggle Coverage	79
Toggle Coverage Data	80
Instance Coverage Window	82
Code Coverage in the Source Window	83
Excluding Lines and Files From Coverage Statistics	86
Saving and Recalling Exclusions	87
Creating Coverage Reports	88
Saving Coverage Data	89
Reloading Coverage Data in the GUI	91
Merging Coverage Data in the GUI	93
Invoking Coverage View Mode in the GUI	94
Coverage View Mode with UCDB Loaded	95
Generating HTML Reports	96
Managing Coverage Data	97
Managing Coverage Data Off-Line	99
Understanding Optimization Effects	100
Customizing the Optimization Level for Coverage	101
Summary	103
Lab 2: Code Coverage	104
Module 4	
vopt for Performance	105
Objectives	105

Table of Contents

Module Overview	106
Optimization (vopt)	107
vopt — The Big Picture	108
vopt Mode versus Full Debug Mode	109
Default vopt Flow	111
Preserving Visibility and Enabling Access	112
Design Optimization and Visibility	114
Adjusting Design Visibility Through the GUI	115
Design Visibility versus Performance	118
3-Step vopt Flow	119
2-Step vopt Flow	120
1-Step vopt Flow	122
vopt: Use for Different Flows	123
Enabling vopt in Alternate Flow (VoptFlow = 0)	124
Summary	125
Lab 3: vopt for Performance	126
Module 5	
Optimization Flows and Methods	127
Objectives	127
Module Overview	128
ModelSim Structure and Flow	129
Optimization Flow	130
RTL/Behavioral Designs	131
vopt Commands for RTL Simulation	132
Verilog Delay Modes	133
Verilog Delay Mode for Gate-Level Simulation	134
Verilog Gate-Level Without SDF	137
Verilog Gate-Level With SDF	138
vopt Switches for Gate-Level Simulation	139
Increasing Verilog Gate-Level Optimizations	141
Analyzing Cell Optimization Results	143
Compiled SDF	145
Verilog Coding Style for Maximum Performance	147
Direct Programming Interface	149
vcom Performance Commands	150
VHDL Coding for Performance	151
Measuring Time and Memory Usage	154
Alternate Elaboration Flow	155
When to Use an Elaboration File	157
Tips for Performance	158
General Performance Issues	160
Summary	161
Module 6	
Statistical and Memory Profiler	163
Objectives	163
Module Overview	164

Challenges	165
Statistical and Memory Profilers	166
Statistical Sampling Profiler	167
Profile On	169
Graphical Views	170
Understanding In and Under	171
Profile Details Window	175
Integration With the Source Window	176
Saving a Profile Report File	177
Memory Allocation Profiler	178
Capacity Analysis	179
Displaying Capacity Objects in the Wave Window	181
Summary	182
Lab 4: Analyzing Performance	183
Module 7	
Virtual Objects and Signal Spy	185
Objectives	185
Module Overview	186
Virtual Objects	187
Virtual Signals	188
Create Meaningful Definitions With Virtual Signals	190
Virtual Regions	191
Virtual Functions	192
Virtual Types	195
Other Virtual Commands	196
User-Defined Radices	198
Managing Radices	200
Global Signal Radix	201
Combining Signals	202
Virtual Signal Builder	203
Signal Spy	204
init_signal_spy Procedure	206
enable_signal_spy Procedure	207
disable_signal_spy Procedure	208
init_signal_spy VHDL Utility	209
Probing Verilog Signals From the VHDL Testbench	211
\$init_signal_spy Verilog Task	212
VHDL Extended and Verilog Escaped Identifiers	217
Summary	219
Lab 5: Signal Spy	220
Module 8	
Waveform Compare	221
Objectives	221
Module Overview	222
Saving Waveform Datasets	223
Opening Datasets	224

Table of Contents

Managing Datasets.	225
Dataset and Compare Windows.	226
Compare Datasets Using Waveform Compare	227
Comparison Wizard	229
Waveform Compare Menus.	231
Compare by Signal Dialog Box.	232
Compare by Region Dialog Box	233
Add Clocks Dialog Box.	234
Differences.	235
Clocked versus Continuous Differences	236
Compare Objects in the List Window	237
Write Report	238
Save Differences	239
Save Rules	240
Comparing Hierarchical and Flattened Designs	241
Using Tcl Commands to Define a Comparison.	242
Tcl Compare Command.	243
Tcl Compare Example	244
Tcl Compare Example Using “when”	246
Summary	248
Lab 6: Waveform Compare	249
Module 9	
HDL Support and Gate-Level Simulation	251
Objectives	251
Module Overview	252
VHDL Support in ModelSim.	253
VHDL-2008 Functionality in ModelSim.	254
VHDL VITAL SDF.	255
SDF Annotation	257
Verilog Support in ModelSim	258
SystemVerilog	259
Why SystemVerilog?.	260
SystemVerilog Support in ModelSim	261
SystemVerilog Compilation Units.	262
SystemVerilog Functionality in ModelSim	263
Summary	264
Lab 7: Gate-Level Simulation and Simulating With SystemVerilog DPI.	265
Module 10	
FSM Viewer.	267
Objectives	267
Module Overview	268
FSM Recognition.	269
FSM Viewer Modes.	270
FSM Recognition in the Transcript Window.	271
Viewing FSM Information in the GUI	272
FSM List Window Menu	273

FSM Viewer and Wave Windows	274
FSM Properties	276
Integration With the Source Window	277
FSM Viewer Display Controls	278
FSM Viewer – Transitions to Reset	281
Export the FSM Viewer Window as an Image	283
Summary	284
Module 11	
Event Tracing	285
Objectives	285
Module Overview	286
Causality Traceback	287
Recommended Usage Flow	288
Alternative Usage Flow	289
Post-Simulation Causality Traceback	290
Terminology	291
Causality Traceback from the GUI	292
Tracing First Sequential From the Wave Window	293
Tracing From the Source Window	296
Tracing From the Objects Window	298
Tracing From the Schematic Window	299
Tracing Immediate Driver From the Wave Window	301
Tracing to the Root Cause	303
Tracing to the Root Cause of an ‘X’	304
Finding All Possible Drivers	305
Tracing from a Specific Time	306
Viewing Causality Path Details	307
Causality Traceback Preferences	308
Summary	309
Lab 8: Causality Traceback	310
Module 12	
Selected Advanced Debugging Topics	311
Objectives	311
Module Overview	312
Debugging Issues	313
Debugging With Breakpoints	314
Checkpoint and Restore	315
Contention and Float Checking	316
Toggle and Stability Checking	317
Contention, Float, and Stability Checking Script	318
Example: Contention Checking	319
Example: Float Checking	320
Example: Stability Checking	321
Iteration Violations	322
-nodebug Command	324
Source Window: Hyperlinked Navigation	326

Table of Contents

Text-Based Dataflow: Tracing Readers	327
Text-Based Dataflow: Tracing Drivers	328
Dataflow Window: Point-to-Point Tracing	329
List Window: Signal Search	330
Wave Window: Signal Search	331
Drag Source to Wave Window	332
Wave Window: Signal Grouping	333
Saving Waveforms Between Two Cursors	334
Wave Window: Expanded Time	335
Wave Window: Expanded Time at Active Cursor	336
Wave Window: Expanded All Time	337
Wave Window: Expanded Time Deltas in the List Window	338
Wave Window: Events Mode	339
Wave Window: Deltas Mode versus Events Mode	340
Post-Simulation Results Analysis	341
Post-Simulation Coverage Analysis	342
Related Functional Verification Training Courses	343
Summary	344
Lab 9: Debugging a Design	345

Appendix A

Transactions	347
Objectives	347
What Is a Transaction	348
Transaction Streams	350
Verilog Transactions in the Wave Window	351
Transactions in the Wave, List, and Objects Windows	352
Transaction Recording Flow	353
Recording Transactions in VHDL and Verilog	354
Verilog Recording Procedure	355
Verilog API Prototypes	361
Verilog API Example	362
Verilog Phase versus Parallel Transaction Example	363
VHDL API Prototypes	364
VHDL API Example	365
Transaction Life-Cycle	366
Recording Attributes and Relations	367
Viewing Transactions in the Wave Window	368
Viewing Related Transactions	369
Customizing Transaction Appearance	373
Debugging With Tcl	374
Transaction Debugging Commands	375
Summary	376
Lab 11: Transactions	377