Development of Web-Based Educational Modules for Testing VHDL Models of Digital Systems

Sucharita Gopalakrishnan

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

Dr. F.G. Gray, Chairman
Dr. J.R. Armstrong
Dr. D.S. Ha

August 6th, 1997
Blacksburg, Virginia

Keywords: VHDL, Testing, World Wide Web, Internet, Teaching

Development of Web-Based Educational Modules for Testing VHDL Models of Digital Systems

Sucharita Gopalakrishnan

(ABSTRACT)

The exponential growth of the World Wide Web (WWW) on the Internet and accompanying WWW browsers has promoted opportunities for new methods of teaching and learning. Teaching does not simply involve presenting textual information over the Internet along with a few hyperlinks, but requires effective user engagement with the teaching module. This is the main challenge in website design.

The objective of this thesis is the development of an effective training module made available over the Internet so as to train acquisition and maintenance personnel on how they can use VHDL to design and maintain digital systems. The educational modules provide extensive information on VHDL modeling and testing styles and standards at various abstraction levels. The Sobel edge detector model was chosen as an example to explain the various concepts of modeling and testing. This model was chosen since it was thought to be simple enough for any student to understand, yet complex enough to explain most of the VHDL concepts of modeling and testing. The course material on test bench development at various levels of abstraction, reuse of test bench models, use of configurations for simulation of mixed abstraction and mixed data type models, testing techniques and WAVES was developed as a part of the current thesis.

Finally a complete section on website design has been included which explains the design strategy adopted for developing the website and the various key issues involved in presenting teaching modules over the Internet.

Acknowledgments

I would like to express my sincere thanks to my advisor Dr. F.G. Gray for providing constant support and guidance throughout my research work. It has been an honor and a pleasure working under his guidance. I would also like to thank my co-advisor Dr. J.R. Armstrong, for his valuable ideas and suggestions and Dr. D.S. Ha for serving on my graduate committee.

Special thanks goes out to Dr. Frank of Research Triangle Institute, North Carolina for his numerous ideas and suggestions.

I would like to dedicate this work to my parents who have always encouraged me at all times. Their love and support are the main reasons for all my achievements. I owe a lot to my husband, Sriram for always listening to me and instilling confidence within me. Without his constant encouragement and inspiration, I could not have successfully completed this present research work.

Finally thanks are due to my project partner Weihong Song for her wholehearted assistance in research and other matters and also I would like to specially thank my friend, Mohammed Osman for his valuable and timely suggestions throughout this past year.

Table of Contents

Chapter 1. INTRODUCTION.....1

1.1 Motivation.....1
1.2 Task Description.....2
1.3 Research Contributions.....2
1.4 Thesis Organization.....5

Chapter 2. BACKGROUND.....6

2.1Test benches.....6
2.1.1 Introduction.....6
2.1.2 Types of Test benches.....7
2.1.3 Test Vector Generation.....7
2.2 Testing Strategies.....8
2.3 Overview of the Sobel Edge detection system.....9
2.4 WAVES.....13
2.5 Computer based Education.....14

Chapter 3. TEST BENCH DEVELOPMENT.....16
3.1 Introduction.....16
3.2 Test Bench development for a behavioral model..... 18
3.2.1 Methodology.....18
3.2.2 Implementation.....18
3.2.2.1 Test Setup.....19
3.2.2.2 Clock Generator component.....20
3.2.2.3 Sobel Edge Detector component.....21

3.2.2.4 Test bench module (TBM).....21
3.2.2.5 Comparators.....27
3.2.2.6 Configuration declarations.....29
3.2.3 Example Test results.....30
3.3 Test Bench development for a structural model.....34
3.3.1 Methodology.....34
3.3.2 Testing procedure.....34
3.3.3 Configuration Example.....35
3.4 Test Bench development of Register Transfer Level model.....39
3.5 Test Bench development for gate level models.....40
3.5.1 Methodology.....40
3.5.2 Implementation.....40

Chapter 4. TESTING TECHNIQUES.....45
4.1 Introduction.....45
4.2 Memory Testing.....46
4.2.1 Introduction.....46
4.2.2 Methodology.....48
4.2.3 Testing Procedure.....48
4.3 Regression Testing.....54
4.3.1 Introduction.....54
4.3.2 Methodology.....54
4.3.3 Testing Procedure.....55
4.4 Diagnostic Testing.....56
4.4.1 Introduction..... 56
4.4.2 Methodology for Diagnostic testing.....56
4.4.3 Testing Procedure.....58

Chapter 5. WAVES.....62
5.1 Introduction.....62
5.2 Example WAVES Test benches.....65

Chapter 6. TEACHING OVER THE INTERNET.....75
6.1 Introduction.....75
6.2 Goals of the Computer-Based Educational module.....76
6.3 Methodology for Teaching.....77
6.4 Implementation.....79
6.4.1 Implementation of Test Plan.....79
6.4.2 Implementation of Test Results.....81
6.4.3 Implementation of Memory Testing.....82
6.4.4 Implementation of Configurations.....84

Chapter 7. CONCLUSIONS AND FUTURE WORK.....86

REFERENCES.....88

APPENDIX A Test Bench and Configuration models.....91

APPENDIX B Files related to Memory and Diagnostic testing.....109

APPENDIX C WAVES files.....119

VITA.....132

List of Illustrations

Figure 2.1 Basic Test bench block diagram.....6
Figure 2.2 A real world image.....8
Figure 2.3 An Image Processing system.....10
Figure 2.4 Typical arrangement of a window of pixels for an image.....10
Figure 2.5 Representation for Quantized Phase.....11
Figure 2.6 Flowchart of the Sobel Algorithm.....13
Figure 3.1 Test bench module for Sobel Edge detector.....19
Figure 3.2 Block diagram and input and output waveforms of the Clock Generator component.20
Figure 3.3 Entity and architecture of the Clock generator component.....21
Figure 3.4 Block diagram of the Sobel Edge Detector.....21
Figure 3.5 Entity and Architecture declarations of the test bench.....23
Figure 3.6 Section of the code of the Memory Process where the input file is read.....25
Figure 3.7 Format of input image file.....25
Figure 3.8 Portion of the image data required by the MUT to start the edge detection process....26
Figure 3.9 Section of the code for the memory process where the magnitude and direction outputs are written into files.....27
Figure 3.10 Block diagram of the Comparator component.....27
Figure 3.11 Portion of the source code for the Comparator component.....28
Figure 3.12 Configuration declaration for Sobel Edge detector.....30
Figure 3.13(a) Input image.....32
Figure 3.13(b) Expected response.....32
Figure 3.13(c) Edge detected output image.....32
Figure 3.14(a) Input image of M1A1 tank.....33
Figure 3.14(b) Edge detected output image of the M1A1 tank.....33
Figure 3.15 Hierarchy of a Structural model of the Sobel edge detector.....35
Figure 3.16 Structural model for Window processor.....36
Figure 3.17 Diagram of the Window Processor with empty sockets.....37

Figure 3.18 Portion of the source code for the configuration for structural model of the MUT...38
Figure 3.19 Diagram for Window Processor with chips inserted in the sockets.....38
Figure 3.20 Portion of the source code of the test bench model to compare the outputs of the Horizontal filter at the behavioral and gate level.....42
Figure 3.21 Configuration for the Horizontal filter.....42
Figure 3.22 Simulation Output with generic constant *wait_time* = 0 ns for the behavioral model of the Horizontal filter.....43
Figure 3.23 Report generated for the Simulation shown in figure 3.22.....43
Figure 3.24 Simulation Output with generic constant *wait_time* = 100 ns for the behavioral model of the Horizontal filter.....44
Figure 3.25 Report generated for the Simulation shown in figure 3.24.....44
Figure 4.1 Test setup for Memory testing.....48
Figure 4.2 Memory module.....49
Figure 4.3 Entity declaration of memory module 50
Figure 4.4 Behavioral architecture of the faulty memory model.....51
Figure 4.5 Error report generated with a test pattern of all 0's in the input image.....52
Figure 4.6 Input Test Pattern - Combination of 1's and 0's.....52
Figure 4.7 Error report generated with a test pattern shown in Figure 4.6.....52
Figure 4.8 Configuration declaration for the multiple fault memory model.....53
Figure 4.9 Test results for Memory Testing.....54
Figure 4.10 Configuration declaration for Regression testing.....55
Figure 4.11 Simulation results with the custom memory module.....56
Figure 4.12 Flowchart for diagnostic testing.....57
Figure 4.13 Test Setup for Diagnostic Testing.....58
Figure 4.14 Input Image Pattern.....58
Figure 4.15 Expected Response for threshold value set between "000" and "3F8".....59
Figure 4.16 Expected Response for threshold value set between "3F9" and "FFF".....59
Figure 4.17 Output Response of the MUT for threshold value set at "002".....59
Figure 4.18 Output Response of the MUT for the threshold value set at "00F".....60
Figure 4.19 Flowchart for diagnostic testing of Sobel Edge detector.....61
Figure 5.1 Sample Waveform.....63
Figure 5.2 Two representations of the above waveform.....63
Figure 5.3 Portion of the original MUT file specifying the entity declaration for the MUT.....65
Figure 5.4 Portion of the modified MUT file specifying the entity declaration for the MUT.....66
Figure 5.5 External file developed for testing the MUT.....66
Figure 5.6 Waves Header file created for Sobel Edge detector.....67
Figure 5.7 UUT_Test_pins file for Sobel edge detector.....68
Figure 5.8 Portion of the source code of the WGP package where the external file is specified..68
Figure 5.9 Portion of the source code of the WGP package specifying timing values.....69
Figure 5.10 Waveforms generated for the Clock and Outputs of the MUT according to the timing specifications in figure 5.9 in the WGP package.....69
Figure 5.11 Package for the Waveform Generation Procedure.....70
Figure 5.12 Portion of the original source code for entity declaration of the test bench model....70

Figure 5.13 Portion of the modified source code of Figure 5.12 where generic constants have been added.....70
Figure 5.14 Portion of the original source code of the architecture body of the test bench.....71
Figure 5.15 Portion of the modified source code of the architecture body of the test bench.....72
Figure 5.16 Portion of the original source code for comparison of outputs.....72
Figure 5.17 Section of the source code where the outputs of the MUT are compared with the expected response.....74
Figure 6.1 Top portion of the Test plan table.....80
Figure 6.2 Bottom portion of the Test plan table.....80
Figure 6.3 Portion of the Test results section.....82
Figure 6.4 Portion of Memory testing - Testing procedure and test results.....83
Figure 6.5 Portion of the Configurations chapter.....85

List of Tables

3.1. Test plan table..31