CHAPTER 1 : Introduction

1.1 Motivation

VHDL is becoming increasingly popular as a way to capture complex digital electronic circuits for both simulation and synthesis. Digital circuits captured using VHDL can be easily simulated, are more likely to be synthesizable into multiple target technologies and can be archived for later modification and reuse.

An understanding of VHDL enhances an engineer’s ability to communicate ideas and designs. Since the main purpose of VHDL is to describe basic digital design concepts and practices, it is advantageous to align learning VHDL with learning digital design.

One of the most compelling reasons for anyone to become experienced with and knowledgeable in VHDL is its acceptance as a standard in the electronic design community. Using a standard language such as VHDL will virtually guarantee that one does not have to throw away and recapture design concepts simply because the design entry method that one has chosen is not supported in a newer generation of design tools. Using a standard language also means that one is able to take advantage of the most up-to-date design tools.

One of the primary advantages of VHDL lies in its ability to capture the operation of a system in a number of abstraction levels, using a coherent syntax and semantics across the levels, and to simulate that system by combining models at different levels of abstraction. It is hence possible to simulate designs in which we have high- level descriptions for some sub-modules with low-level implementations of other sub-modules. During maintenance of the system, redesigns or alterations of a subsystem can be easily verified by just replacing the VHDL model for that subsystem and re-simulating with the original test set.

A major focus of the ***VITAMINS*** (VHDL Interactive Training for Maintenance and Acquisition Specialists) project here at Virginia Tech, is the development of educational modules suitable for use on the Internet. As VHDL and related standards evolve, the Air force needs innovative, low cost ways of providing education for their personnel on the use of VHDL for acquisition and maintenance. The VITAMINS project fulfills this educational need by providing Computer Based Educational (CBE) modules that illustrate how VHDL models can be used to satisfy the multiple needs of acquisition and maintenance personnel within the DOD (Department of Defense) community.

1.2 Task Description

The main aim of this project is to educate Air Force acquisition and maintenance specialists in the role of VHDL applications so that they can apply the knowledge acquired in their work. A training module was developed to help the acquisition personnel better their skills at negotiating and supervising development of VHDL models and increase the capability of maintenance personnel to use VHDL models for development of effective test benches and verification strategies. The educational modules, apart from presenting information on VHDL modeling styles and standards, also provide hands-on experience on VHDL analysis, simulation and use of synthesis tools. The course material on test bench development, various testing strategies, reuse of the models at various levels of abstraction and overview of an existing test bench generator was developed as a part of the current thesis.

Apart from presenting the course information, links have been provided to Department of Defense Handbook (MIL-HDBK-62) which is a rich source of material on VHDL related standards, coding standards and naming conventions. Also care has been taken to link the various commercial tools, VHDL models available over the Internet, RASSP (Rapid Prototyping of Application Specific Signal Processors) information on remote sites and various other on-line data with the educational module. The module has been so designed that the user can easily navigate through the course to find the specific information needed for a specific project.

1.3 Research Contributions

The following research contributions were made :

1. An effective testing methodology was developed to test a model. The test bench model developed to test the MUT (Model under test) at the behavioral level of abstraction can be reused to test the MUT at the lower levels as well. The following features ensured reusability of the test bench model.

- Use of configuration declarations: Configurations were used to bind the architecture of the MUT at different abstraction levels to its instance in the test bench. In this way, when the MUT was tested at the lower levels of abstraction, the architecture and library name of the MUT corresponding to the abstraction level were replaced and the additional sub-components required at that particular level were added and the MUT was then simulated with the same test bench and test patterns as used with the MUT at the behavioral level.

- Use of Generics : Any important device and system parameters which required to be changed at different abstraction levels were declared as generics, and the values for these were provided only in the configuration file. In this way, while simulating the MUT at different abstraction levels, the values of generics which had to be modified were done so only in the configuration file without having to change the test bench code.

- Also emphasis has been placed on good documentation and coding style which improves readability of the VHDL code and facilitates reuse.

2. Effective use of configuration declarations can provide significant assistance in the configuration management of models. In this thesis, advantages of using configurations especially for implementing mixed abstraction level and mixed data type models was thoroughly explored. Configurations are very useful for both these types of models. In the case of mixed abstraction level models, by changing the configuration file, components at different levels of abstraction can be chosen by just changing the library and architecture name without having to modify any other file. In the case of mixed data type models wherein one sub-component may be of a different data type than all the other components of the MUT, the data type of the signals being fed to this sub-component needs to be converted and again the output signal of the component needs to be re-converted to the original data type. Instead of redesigning the sub-component in order to take care of these requirements, a simple and efficient method was adopted, where configurations were used to achieve the same purpose. The data type conversions required were carried out in the configurations. In this manner, mixed data type models can be easily tested, without having to re-design components for each implementation and without having to change test benches or test data files.

3. Methodologies were developed for different testing techniques namely memory testing, regression testing and diagnostic testing.

- Memory Testing : In order to test the Memory Processor of the Sobel Edge detector, a technique was devised, where different faults were embedded into the memory module and then a test bench model was developed which compared the contents of each memory cell with the expected response and generated a report indicating the faulty cells in the module. Again configurations were use to select the different faulty architectures that were created for testing.

- Regression Testing : The main advantage of regression testing is that it facilitates model-year upgrades. Whenever any existing component in a system is modified or upgraded the functionality and performance of the system with the new component needs to be tested. A simple and novel way to carry out this testing procedure again uses configurations where the entity, architecture and library name in the configuration specification statement for the modified component is replaced and the whole system is simulated with the same set of test vectors and the same test bench module as used before without having to modify any of the other sub-components in the system. The output response of the simulation can then be checked against the expected response to determine how the new component effected the performance of the system.

- Diagnostic Testing : Whenever a system is tested, its resulting response is analyzed to ascertain whether it behaved correctly. If incorrect behavior is observed, then the second step is to ***diagnose*** or locate the cause of the misbehavior. The methodology developed for diagnostic testing can locate the faulty component of the MUT by simply observing the output response pattern. The faulty component is then replaced with an error-free one and the MUT is tested again, until the expected response is obtained.

4. Test Bench generation with WAVES : WAVES was used to successfully generate test bench models for the Sobel edge detector at the structural as well as the RTL level. However, in order to generate the test bench models, the file containing the Sobel edge detector model had to be modified in order to suit WAVES. The external file which provides the input test vectors also had to be developed in a particular fashion suited for WAVES. Finally, the test bench model that was created by WAVES also could not be used as generated by WAVES, but had to be modified to suit the design requirements of the Sobel Edge detector. This thesis carefully documents the restrictions and modifications to WAVES files needed to effectively test and diagnose digital systems.

5. Methodology for Development of Web based modules: Conveying information over the Web is unlike conventional classroom teaching wherein there is accessibility to the instructor. This was a major challenge in the current thesis. When you attempt to present information intended to educate and teach readers over the Internet, the fact that the information provided should be lucid yet objective, complete yet terse has to be an important consideration. A major contribution of this thesis is to develop and document effective techniques for teaching using web-based technology. While developing the web based module care was taken to present the material in a manner so as to involve the user in his/her learning process, while maintaining objectivity

1.4 Thesis Organization

Chapter 2, "***Background* ”** gives an overview of VHDL test benches and its potential capabilities. Background information on the Sobel algorithm, testing techniques and the WAVES tool have been detailed. Finally the development of the computer-based module over the Internet is discussed.

Chapter 3, ***“Methodology for Test Bench development”*** describes the development of test benches at different levels of abstraction and reusability of the models such that the model developed at a higher level can be used at a lower level. This chapter also discusses importance of configurations and their capabilities.

Chapter 4, ***“Testing Techniques”*** discusses various testing techniques for maintenance and diagnostic purposes. Information on memory testing, regression testing and diagnostic tests are presented with several simulation examples.

Chapter 5, ***”WAVES”*** gives an overview of WAVES. A tutorial on how to use this tool and its capabilities are discussed. Also several test bench models developed for the Sobel Edge detector at different levels of abstraction are shown and explained.

Chapter 6, ***”Teaching over the Internet”*** includes the strategies on how the entire course has been effectively presented over the Internet and the techniques on teaching over the Internet.

Chapter 7, ***”Conclusions and Future Work”*** is the concluding chapter of this thesis. It highlights the contribution of this thesis towards VHDL test bench development and validation techniques and makes suggestions for further improvement.