CHAPTER 1

INTRODUCTION

1.1 Motivation

The computer-aided engineering revolution of the last 20 years has made tremendous strides in reducing the time to develop electronic systems and components and in increasing the complexity of their functions while at the same time reducing their size, power, and weight. Hardware description Language (HDLs) such as VHDL have made it possible for circuit and board designs to be done without resorting to paper, allowing computers to manage the design database and automate the translation between various representations of the system.

VHDL was developed to provide a standardized and technology independent way to describe formally the behavior and structure of digital electronic systems. It offers the technical means to provide functional, timing, and other specifications for digital electronic systems in a form that will be useful long after the original system is delivered. Technology independence permits the separation of the behavior function (plus timing) from its implementation, which makes incorporating new technology easier.

VHDL descriptions specify exactly what functions a new device would have to perform and the timing information associated with it. Through simulation of these descriptions, the design of a new device can be accurately modeled before being physically verified. Also, it allow the detailed structure of a design to be synthesized from a more abstract specification, allowing designers to concentrate on more strategic design decisions and reducing time to market.

Although VHDL modeling can provide a bonanza of benefits, VHDL models must be used effectively to reduce overall development costs. VHDL developers need to understand all of VHDL modeling theory including syntax
and language structures to develop an electronic component. A tool to translate designed component diagram to synthesizable VHDL code is needed to solve this problem.

It is expected that by using the tool provided, developers can more quickly generate electronic component without knowing any VHDL Code.

1.2 Task Description

The objective for this thesis is to develop a user friendly, low cost and effective VHDL code generator based on a given component diagram.

This thesis discusses the development of a web based GUI application for generating VHDL code based on a given component diagram. This application presents the multi user design process with drag and drop features for easier VHDL modeling.

1.3 Contributions

The development of the VHDL Code Generator (DiaHDL) is the main topic of this thesis, including the following specific items.

**Development of a web based GUI application for VHDL Modeling (DiaHDL)**

We present such a tool which is used to generate VHDL code based on a given component diagram. This code can be directly analyzed and synthesized. This tool provides a graphical user interface that allows the user to select a digital component, customize their parameter and generate the VHDL code.

**Development of a converter application for use in the library creation process**

VHDL code cannot be used directly for DiaHDL, a converter is needed to convert VHDL code to JHO file structures.
Development of JHO libraries to the DiaHDL tool

Several libraries already been develop to show the functionality of the application that has been developed (DiaHDL). The following libraries are included:

1. Ripple Carry Adder / Subtractor (with or without pipelining)
2. Carry Look Ahead Adder / Subtractor (with or without pipelining)
3. Signed Array Multiplier
4. Signed Carry Save Array Multiplier
5. FIR Filter

All of these libraries are synthesizable and have already been tested on Xilinx ISE Design Suite 10.1.

1.4 Thesis Organization

Chapter 2, “Background” gives an introduction to the concept of VHDL modeling, Java, JavaScript and PHP programming language.

Chapter 3, “Design System” describes the planning stages of this application through the creation of flow diagram with its explanations.

Chapter 4, “Implementation System” explains the implementation steps based on previous design plan.

Chapter 5, “Test and Analysis” discusses the testing of this application to generate VHDL based on a component diagram.

Chapter 6, “Discussion and Future Work” is the concluding chapter of this thesis and it highlights the contribution of this thesis to a VHDL Code Generator. Also, suggestions are made for further refinements and improvement.