HONGFU Campus of BUPT
Changping, Beijing 102200
(+86) 82319092

June 7, 2015

Mr. Andy Watson
Queen Mary University of London,
Mile End Road,
City of London, E1 4NS

LABORATROY REPORT
VHDL SIMULATION LANGUAGE
OF NAND AND ADDERS

INTRODUCTORY SUMMATY

We recently completed the lab session to learn how to use VHDL and ModelSim to design and simulate some basic circuits, namely: a NAND gate and three types of Adders (Half, Full and Parallel). The results of simulation are put down as it is, and we are to present what we have obtained to you in the informal report. A longer, formal report will follow at the end of the semester.

Based on the experiment, we can safely promise you that the capacity of using VHDL for more complex circuits have been available. And we will present all the simulation done by computers to give you a general understand of our progress.

LAB MATERIALS

The first stage in the design of our circuits is to create one of the most basic components; this is the NAND gates. In order to do this, our lab analysis relies on two tools:[1]

1. A VHDL tool called “Xilinx Project Navigator”.
2. A simulation environment for verifying the source code and timing models of our designs.

LAB PROCEDURE

In order to obtain the final PARALLEL ADDER, we have to compose the circuit step by step; design a HALFADDER and a FULL ADDER before we reach the final stage.

[1] This report assumes the following tool versions are used: *Xilinx ISE WebPack 7.1i* and *ModelSim XE III Starter*.

PART 1 - NAND Gate Implementation

At this stage, we follow these steps to get started.

a. Open the Xilinx Project Navigator tool.
b. Create a directory where we wish to save our project as indicated by the example given by instruction.
c. After filling in the appropriate fields, click the NEXT> button.
d. Create the entity and architecture.
e. Input the prepared codes.

PART 2 – Design of a HALF ADDER

The next stage of the design consists of using our NAND gate design to make a HALF ADDER. In order to do this, we must create a HALF ADDER entity and architecture pair that calls the NAND design.

a. Add a new VHDL source to your project and create the entity as described; call it "HALFADDER".
b. Input the prepared codes.
c. Used inside of an architecture is known as a component, declare our own component.
d. Add signals act like wires in a circuit and use them to connect entities together.

PART 3 – Design of a PARALLEL ADDER

In order to accomplish this, you will need two FULL ADDERs connected. A PARALLEL ADDER can add two bits numbers and produce a 2-bit output and a 1-bit carry. We first work out the logic by ourselves.

a. Draw a diagram of our proposed circuit.
b. Input the prepared codes.
c. Then proceed as a designer would and simulate our design, check the waveform diagram for errors, and then work backwards in order to detect any errors.
d. Write below the VHDL code both for the PARALLEL ADDER' entity and its architecture, and then copy that code to the appropriate window in the VHDL tool.

PROBLEMS ENCOUNTERED

Since the versions of simulation software is relatively low, it can be difficult to find the right buttons to carry on the experiments. Besides, adding

test work bench is easy while doing the simulation is hard, since we are not familiar about the interface of *Xilinx*.

CONCLUSION

We have finished the hardware labs prior to this one (i.e., Lab 1 and Lab 2) and understand the logic behind the circuits since the same designs are used in this lab. And we smoothly went through the whole process to add NANDs, HALFADDERs, FULLADDERs and PARELLELADDER to the project. Some problems aroused though, we get over them and finally step to the last stage. Therefore, we present all the record of the experiment to express the willingness to help to any your experiments related to VHDL.

I will sent you an e-mail to discuss our study and any further experiments you may wish us to do in the next period.

Sincerely,

Zengmingyu XIA