Lesson Learned from ECE3300L

Choi Tim Antony Yung
Department of Electrical and Computer Engineering
California State Polytechnic University, Pomona
Pomona, United States
choiyung@cpp.edu

***Abstract*—From this class I learned about Verilog Hardware Description Language, using Verilog with Xilinx Vivado, the foundation of digital circuit, its design, its application, digital signal processing, simulation of analog signal with pulse width module and the development flow and troubleshooting process of development of digital circuit with Verilog.**

***Keywords*—Digital Circuit Design, Verilog, Vivado, FPGA, Digital Signal Processing, Pulse Width Modulation**

I. Introduction (*Heading 1*)

In Fall 2020, I have taken the ECE 3300 Digital Circuit Design Using Verilog class with Professor Yin of Cal Poly Pomona. This document details the valuable lessons I have learned from this class.

II. Verilog and Xilinx Vivado

First and foremost, I learned about Verilog for the first time from this class. The Verilog hardware description language assist on the design of physical circuit and synthesis of field programmable gate array by allowing the synthesizer, in this case Xilinx Vivado synthesizer to construct a schematic from behavioral description of circuit. Since I learned about Verilog and Vivado from this class I have used it, specifically the RTL Analysis feature, multiple times to generate a schematic describing a circuit I have in mind and described via Verilog HDL. Before I learned about this possible way to generate schematic I was using either the circuitikz package of LaTeX to construct a schematic or a capture program such as OrCAD Capture. However, it can take a long time to generate a schematic using either of them as the circuit get complicated. Verilog and Vivado solved this problem since multiple instances of same module can be created which greatly reduce the time used to put duplicate part of circuit in the schematic. Besides, using Verilog means I would not need to worry about the layout of the components on the circuit, which makes it so that I can generate schematic faster on the occasion of which the physical layout is not of concern.

III. Digital Logic Design

The other value of this class is the revisiting of digital logic circuit components, their principal of operation and their usage as a component of a digital circuit. Although these topics are covered in the prerequisite of this class, it is certainly advantageous to my career that I have a solid foundational understanding of what a digital circuit is consist of. In addition to the foundational understanding of the components of digital circuit, this class also covers a lot more applications of these components and the approaches we can take to design these circuits, which will certainly be useful for me as a future engineer.

IV. Usefulness of Field Programmable Gate Array

Before this class, the (somewhat incorrect) understanding I have of FPGA is that it is something that can be configured to simulate a circuit consist of logic gates. It is not until after I enrolled in this class that I realize that FPGA can be used to basically construct an entire digital circuit. Combined with analog or digital circuit elements built into the Digilent Nexys A7 development board, it made developing a digital circuit a lot more seamless and error-free. Although it is not likely that we will be using the same exact development board in the future when we are tasked with designing a digital circuit using FPGA, it is certainly beneficial to be familiarized with the development of digital circuit using FPGAs and the advantage it offers over other methods.

V. Digital Signal Processing and Simulation of Analog Signal with Pulse Width Modulation

Another important concept I learned from this class is pulse width modulation and processing of digital signal. The audio generation experiment has offered a chance for me to experiment with simulation of analog signal with PWM from digital signal. In the audio processing I was offered a chance to experiment on generate analog audio signal from digital signal and processing of digital signal. Since multiple pulse wave of different frequency need to be overlapped in order to generate an audio signal resembling a chord, rotation of inputs from generating module of pulse wave of different frequency was fed into a multiplexer with a counter as select, which allow the resulting signal to resemble overlapping pulse wave after an analog circuit smooth out the digital output signal.

VI. Development Flow and Debugging Verilog Source Code

In addition of learning about the syntax, the usage and the structure of Verilog source code, the hands-on experience has also offered me a chance to construct a circuit using Verilog and make mistakes from doing so. The experience of construction of a Verilog source code, the mistakes I made that resulted in errors during either synthesis, implementation or bit stream generation stage offered me a chance to learn from the mistakes I made in the process of constructing a digital circuit in Verilog such as syntax error, error in constraint file and some mistakes in constructing the logic of the digital circuit.

VII. Conclusion

This class offered a chance for me to learn, for more about the Verilog hardware description language, the Xilinx Vivado development environment, the principal of designing digital circuit logic, the principal of operation of FPGA and the advantage of using FPGA to construct a digital circuit, digital signal processing, simulation of analog signal and the

development flow and troubleshooting of Verilog source code.

REFERENCES

[1] Brown, A., 2020. *Nexys A7 Reference Manual [Digilent Documentation]*. [online] Reference.digilentinc.com. Available at: <https://reference.digilentinc.com/reference/programmable-logic/nexys-a7/reference-manual> [Accessed 20 October 2020].

[2] Brown, S. and Vranesic, Z., 2014. *Fundamentals Of Digital Logic With Verilog Design*. 3rd ed. New York: McGraw-Hill.