Computer System Design Lab # 6
ISA Write

Amy Guo
Lab Partner: Charlie Coleman

March 12th, 2018

Pre-Lab: Not Applicable

Objective: The objective of this lab is to generate a clock/counter at a specific frequency using digital logic, VHDL, and C.

Circuit Diagram: Not Applicable

Outcome Predictions: We expect to be able to modify a clock rate using a C program on a Spartan 3 connected to a programmed Mesa 4i38.

Equipment:

- Xilinx Spartan 3 connected to a Mesa 4i38
- PC
- Oscilloscope

Procedure:

1. Using VHDL, design an ISA device that is capable of generating a variable frequency, 50% duty cycle clock/8 bit counter.
2. In the VHDL design, allow for the clock divisor to be set by writing data into the IO space of the PC.
3. Map the pins of the VHDL design to the Spartan 3/Mesa 4i38
4. Place the clock pins on IO lines so that they can be examined using an oscilloscope.
5. Write a C program in DOS that takes a desired frequency, calculates the divisor, and writes to divisor to IO space.

Recalculations and Predictions: N.A

Data and Observations: No data was collected for this lab, but the clock waveforms were observed using an oscilloscope.

Analysis & Discussion: We used the oscilloscope to display/measure the frequencies of the clock. We struggled a bit because we were off by a factor of 2, but after diagnosing the issue, we fixed it. The C program eventually made the right clock times.

Lab Questions:

Q: Discuss the difficulties of choosing a valid IO port on a PC compatible.

A: Many times during the laboratory exercise, the clock signal would be at 50MHz and never change. We found out after much trial and error that the ports we were using to write and read the divisor from were in use by a different device, so the data was not transferring properly. After testing with a couple of other ports, we were able to find some IO space that was not in use, and therefore could accurately test our design.

Q: How did you handle allowing the master clock to be seen on the output, that is, how did you handle dividing the master clock by 1?

A: In order to divide the clock by 1, I simply allowed the master clock to be the LSB of the counter and incremented the other 7 bits on the rising edge of the master clock.

Results: The experiment worked just as was expected. The clock output was very close to the input frequency, and the only noticeable errors were when the clock rate did not divide evenly into the master clock rate, causing some rounding errors. Initially, problems were faced when the divisor/frequencies were placed into an integer in the C program, as this caused overflow and the output divisor was incorrect. Once the data types were switched to long, this worked much better.

Conclusions: This lab exercise provided us with more practice with VHDL and some experience creating programs for DOS in C. VHDL is a vital skill for anyone who wants to work in digital design fields, as it can be used to implement a vast array of components and is used for many large scale/advanced systems.

Creating DOS programs may seem unimportant, as DOS is a fairly dated operating system, but this is also important for working with companies that use legacy systems and for designs that don't have large amounts of processing power. DOS provides a relatively basic and lightweight operating system for simple designs. Overall this lab was a success.