Lab 2

Jason Reynolds
Vinay Vazir
CPEG 324 Lab 2

Abstract:

The goal for this project was to introduce core VHDL skills such as how to write VHDL, how to create a testbench for a component and how to simulate and view the results of the testbench.

Division of Labor:

The lab was broken down into three components. The first component was the mux, the second component was the shift register and the third component was the adder/subtractor. Both team members completed the mux individually, as we decided it would be good practice to learn VHDL and how to write a testbench, as well as a good way to make sure all of our tools were working correctly. Since it was a basic component, we could easily check each other's work and correct any mistakes.

The second and third part was the shift register and adder/subtractor. Vinay worked on the shift register, creating both the 4-bit shift register and the 8-bit shift register as well as associated testbenches. Jason worked on the adder subtractor, as well as associated testbenches.

Strategy:

For the mux, we both decided to work separately on our own version of the component so we would both have a better understanding of how it worked. After we completed the mux, we compared how we did it with each other, and created several test benches to test the functionality of both of them. We also used the mux to test the functionality of ghdl and gtkwave, making sure they worked correctly.

For the shift register, Vinay first started by working out how the shift register worked using various online resources, before laying out a framework in vhdl. Then he started working out the load and hold functionality, before going on to implement the shift left operation. Using the shift left operation as a base, he created the shift right operation, and then set up the circuit to work with a demux that would select the correct bit to replace with the I_SHIFT_IN. After that he created the schematic of the 4-bit-shift register.

After that, Vinay created the 8-bit-shift register by taking two 4-bit-shift registers and combining their inputs together, and creating two demux circuits that would determine what bit to shift across the registers when shifting right and left. He then created testbench cases and a schematic for the 8-bit-shift register to ensure it functioned correctly.

For the 4 bit adder/subtractor, Jason decided to use a carry look ahead (CLA) adder. This is composed of four, 1 bit, full adder “slices” and 1 “block”. The block calculates the carry while each slice calculates one of the result bits. Jason also implemented a separate component to check for the underflow by examining the sign bit of the inputs and results. To test this component, both positive and negative numbers were added and the result, the carryout/overflow, and the underflow were all examined. While not every possible combination was checked, every input, from 0000 to 1111, was used for the test bench on both inputs.

Results:

We were successfully able to produce a 4-bit and 8-bit shift register, as well as an adder/subtractor with overflow/underflow. Both of us created a working 4-to-1 mux.

Conclusion:

This was a good project to get more comfortable with VHDL as well as implementing components. In the future, the adder/subtractor could probably be improved by creating another component to replace the full adder slice that calculates the most significant bit. While it certainly works as is, it may be more efficient to tailor that slice more specifically to the needs of this adder/subtractor since the adder is signed. The only trouble experienced while working on the adder/subtractor was getting the logic worked out for the underflow.

Appendix I:

Name	Hours spent
Jason Reynolds	10
Vinay Vazir	11

Appendix II:

All code located here:
https://github.com/Syncla/CPEG324/tree/master/lab2

Mux:
https://github.com/Syncla/CPEG324/blob/master/lab2/mux.vhdl

4 Bit Shift Register:
https://github.com/Syncla/CPEG324/blob/master/lab2/4BitSchematic.pdf
https://github.com/Syncla/CPEG324/blob/master/lab2/shift_reg.vhdl

8 Bit Shift Register:
https://github.com/Syncla/CPEG324/blob/master/lab2/8BitSchematic.pdf
https://github.com/Syncla/CPEG324/blob/master/lab2/shift_reg_8.vhdl

Adder/Subtractor
https://github.com/Syncla/CPEG324/blob/master/lab2/Problem3Diagram.png
https://github.com/Syncla/CPEG324/blob/master/lab2/addsub.vhdl

Appendix III:
All test files can be found here:
https://github.com/Syncla/CPEG324/tree/master/lab2

Mux:
https://github.com/Syncla/CPEG324/blob/master/lab2/mux_tb.vhdl
https://github.com/Syncla/CPEG324/blob/master/lab2/mux_tb.vcd

4 Bit Shift Register:
https://github.com/Syncla/CPEG324/blob/master/lab2/shift_reg_tb.vhdl
https://github.com/Syncla/CPEG324/blob/master/lab2/shift_reg_tb.vcd

8 Bit Shift Register:
https://github.com/Syncla/CPEG324/blob/master/lab2/shift_reg_8_tb.vhdl
https://github.com/Syncla/CPEG324/blob/master/lab2/shift_reg_8.vcd

Adder/Subtractor
https://github.com/Syncla/CPEG324/blob/master/lab2/addsub_tb.vhdl
https://github.com/Syncla/CPEG324/blob/master/lab2/addsub.vcd