###### *CSE 260 – Introduction to Digital Logic and Computer Design Jon Turner*

Lab 3 Report

##### *Your name 3/4/2013*

***Part A***. (20 points) Paste the VHDL for the stack component.

paste your code here

***Part B.*** (15 points) Write a testbench to verify the operation of your stack module and paste the source for your testbench below. Your testbench should add a series of unique values that fill the stack and should then pop items off the stack. You can save yourself some time and effort by writing some simple utility procedures to control the stack input signals when performing the *push*, *pop* and *swap* operations. Include assertions in your testbench to check that the value on the top of the stack is correct at all times, and that the full and empty signals are correct. Be sure to check that a push on a full stack has no effect, and that a pop on an empty stack has no effect. Also be sure to check that the swap operation works correctly. Include comments in your testbench to explain what you’re doing.

paste your code here

***Part C.*** (15 points) Paste one or more screenshots from your *stack* simulation below. Be sure to include the stack contents (expand the stack to show each stack value as a separate waveform) and stack pointer values in your simulation output. Make sure all text is easily legible. Your screenshots should contain information so that the TAs can verify your answers to the questions below.

*paste your screenshot(s) here*

At what time in the simulation does the stack first become full?

At what time is the first push on a full stack?

At what time does the stack first become empty after being non-empty?

When is the first successful swap operation?

When is the first pop on an empty stack?

***Part D.*** (20 points). Paste your source for *stackCalc* below

paste your code here

***Part E*** (10 points). Draw a block diagram of your *stackCalc* circuit below. You can show most of the control logic as a large block labeled *control*. However, the *stack* component and any additional flip flops or registers used by the *stackCalc* component should be shown as separate blocks. Also, you should show how different data values can be input to the stack and you should label signals appropriately to make your diagram as clear and unambiguous as you can.

*paste your diagram here*

***Part F***. (15 points) Simulate your *stackCalc* circuit using the provided *testStackCalc* testbench. Verify that there are no error reports on the simulation console from the assert statements in the testbench. Paste a screenshot of the first 3 microseconds of the simulation below. Include all internal state from your *stackCalc* component and all the stack signals in your waveform display. Show the stack in expanded form. Use the unsigned integer radix for all the numeric values.

*paste your screenshot here*

What values are on the stack at time 850 nanoseconds?

What values are on the stack at 1.7 nanoseconds?

Paste a copy of the portion of the simulation from 3 microseconds to the end below.

*paste your screenshot here*

What calculator operation is being performed at time 3090 ns? How does this operation change the values on the stack? Is the resulting stack state correct?

What calculator operation is being performed at time 3200 ns? How does this operation change the values on the stack? Is the resulting stack state correct?

***Part G***. (15 points) Simulate your completed circuit using the provided *testTop* testbench. Include the inputs and outputs of the *stackCalc* component in your waveform window, plus the stack and stack pointer signals from the *stack* component (in this case, you need not expand the stack). In addition, include the *selekt* and *nuChar* signals from *outMod* component, plus the *cb* signal from the *lcdDisplay* component. Show the *nuChar* and *cb* signals in ASCII format. Paste a screenshot showing the portion of the simulation from 0 microseconds to 6 microseconds below.

*paste your screenshot here*

What calculator operations are performed during this period?

How many items are on the stack at the end of this period?

Paste a screenshot showing the portion of the simulation from 29 to 31 microseconds.

*paste your screenshot here*

What operation is performed at 29.2 microseconds?

What operation is performed at 30.8 microseconds?

Paste a screenshot showing the period from 15 to 20 milliseconds (not microseconds) below.

*paste your screenshot here*

What would you expect to see on the LCD display at this point? Why?

***Part H***. (10 points) Proceed to this part only after you have completed the simulation in Part *G* and have convinced yourself that the complete circuit will work correctly when transferred to the prototype board. Prototype your circuit using one of the prototype boards available in Bryan 316. Once you have your circuit loaded onto the board and you have convinced yourself that it works correctly, fill in your name below on the printed copy and have one of the TAs check it and sign their name below, after assigning the appropriate number of demo points..

Student name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ has successfully demonstrated the *stakCalc* circuit on the prototype board.

TA name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

TA signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Demo points (out of 10):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Comments (if the circuit does not work 100% correctly, make a note of all issues below):