###### *CSE 260 – Digital Computers: Organization and Logical Design Jon Turner*

Lab 1 Solution

##### *1/28/2013*

***Part A*** (20 points). Paste the VHDL for your modified calculator below. Highlight your modifications to the code by making them **bold**. Note that the next paragraph is formatted using the ”code style” which uses a fixed-width font and has appropriately spaced tabs. Please always use this paragraph style for your VHDL code. Also, be sure to format your code so that lines do not wrap-around in the lab report. Points will be deducted for code that is badly formatted or difficult to read.

paste your code here

***Part B.*** (10 points) Draw a block diagram of the calculator including your modifications to it. Include a block labeled *ErrorDetector* that represents the logic that determines if an error has occurred. Your diagram need not show how this block is implemented, but it should show all the signals that connect to it. Your diagram should also include a flip flop top hold the error bit.

It’s ok to submit a hand-drawn figure on the printed copy only, but please, make it neat, well-organized and legible.

*paste (or draw) your diagram here*

***Part C.*** (20 points). Modify the *testCalculator* testbench to include additional tests to verify the error detection feature added to the calculator. You must verify that it detects errors for both values of the *mode* input, that error bit is cleared when a *clear* operation is performed and that other operations are blocked when the error bit is set. Highlight the tests you added by making them **bold**.

paste your code here

Run the simulator using the *testCalculator* testbench and paste a screenshot of the waveform window below. Set the radix to hexadecimal for the *dIn* and *result* signals. Note that by default, the simulator displays signals in white on a black background. This works poorly when printed. You can change this using the Preferences item in the Edit menu of the waveform window. Please make the waveform area white and make the signal colors and all text black. You should only have to do this one time. After that, your preferences will be remembered. If your screenshot extends over a longer time period than 400 ns, split it into two screenshots and paste them one after the other. That is, first paste a screenshot that covers the first 400 ns of your simulation run, then paste a second screenshot for the next 400 ns. Make sure that all text is clearly legible on the printed copy. If not reduce the time period covered by each screenshot until it is legible.

*paste your screenshot here*

Find the place in your simulation where *mode*=0 and *error* goes high for the first time. What are the values of the two operands at this moment? Explain why these input values should trigger an arithmetic error.

Find the place in your simulation where *mode*=1 and *error* goes high for the first time. What are the values of the two operands at this moment? Explain why these input values should trigger an arithmetic error.

***Part D.*** (10 points) Paste a copy of the modified version of top below.

paste your code here

***Part E***. (10 points) Draw a block diagram of your top level circuit below. It should include the input module, output module and calculator and all signals connecting them. Label all signals with the names used by the top module.

*paste (or draw) your diagram here*

***Part F***. (15 points) Run the simulator using the provided *testTop* testbench. Do not modify the testbench. When you run the simulation, you will notice that all the interesting stuff happens near the very end of the simulation. Focus on this part when you are checking to make sure your circuit works correctly. Add signals to the waveform display to help you verify that the circuit works correctly. Specifically, be sure to include all of the calculator’s internal signals and the *nuChar* signal in the output module. Organize the signals so that related signals are grouped together and use dividers to label different groups of signals. Paste a screenshot of your simulation output showing the time period from 11,001 microseconds to 11,003 microseconds. Make sure that all text is clearly legible on the printed output.

*paste your screenshot here*

What are the operand values for the first addition operation in this time period? What is the result of this addition? Is it correct? Is an error detected at this point?

What is the value of *nuChar* at this point? Is it what you expect? Are the LEDs correct?

Paste another screenshot below, this time covering the time period from 11,012 microseconds to 11,015 microseconds.

*paste your screenshot here*

What are the operands for the first addition operation in this time range? What result is produced? Is an error detected at this point?