Lab 7: RPN Calculator

Lab Report Guidelines

Due: Tuesday, April 24, 2012 at 11:59PM (ALL SECTIONS)
(Optional no-penalty extension to Monday, April 30, 2012 at 12 NOON)

For this lab, you (as a group) will write a report explaining what you built in this lab and how they work. In addition, you will submit your screenshots and code. You should put these together in one document and submit a PDF on Blackboard. There will be a 10%/day penalty for late submissions. Each team will turn in one document, and all team members will receive the same grade. Only one member of each group has to submit their report on Blackboard. Blackboard will register the submission under all team members.

You should first read the general lab report guidelines on the ESE171 website. There, you will find the various sections you must include in your report, as well as an outline of what to include in each section.

For Lab 7, you must include the following screenshots/code:

1. Module schematics or code (THEORY SECTION)
   a. One-pulse module
   b. Control unit
   c. Stack (including schematics of all sub-modules)
   d. ALU (the main schematic for the ALU itself, not the top-level from Lab 5; you don’t have to include all the small sub-modules)
   e. Top-level

2. Test Bench Waveforms or VHDL Test Bench Code (RESULTS SECTION)
   a. One-pulse module
   b. Control unit
   c. Stack (including simulations for all sub-modules)
   d. ALU (from Lab 5)
You also have to answer a series of discussion questions in the discussion section of the report. Answer these in paragraph form, with each question on a new paragraph.

1. You were told the main board buttons are not debounced and thus to not use them. Why is a lack of debouncing an issue in Lab 7? How is the debouncing issue different from the issue solved with the one-pulse module?

2. What are some advantages of using RPN to implement a calculator (such as the HP-42S) versus the traditional calculator implementation (where values are entered in mathematical expression order)? What are some disadvantages?

3. Your friend tells you that he would like to be able to negate the number at the top of the stack, and says that adding that functionality shouldn’t be difficult. He’s right: you simply flip all the bits and add 1. You’ve been doing that since your first days in ESE171. However, your friend graciously tells you that one change must be made to the stack logic. What is it and why do you have to make it? *Hint: Let’s say this negate operation is denoted by **NEG**. Then, will “5 4 NEG +” in RPN result in “5 + (-4)”?

4. How would you extend the stack to store 16 elements instead of 4?