Project Recommendations

You may receive a bonus of up to 5% on your final grade for completing a project that uses your lab computer beyond lab 8. The number of points that you receive will depend on the complexity of the project that you complete.

You have the option of designing your own project, for up to 5% of your grade. Have your project approved by your TA or by Prof. Leeser.

You can also choose one of the projects suggested below.

1 Hardware projects

1. (2 %) Modify the shift instruction to support different amounts of shift. The instruction format remains the same as before, but the shifter now accepts the amount to shift in the immediate field.

   ```
   sra
   Type: I-type
   Opcode: 0x9
   Syntax: sra $destination, $source, SHAMT
   Description:
   Shifts $source SHAMT bits to the right using an arithmetic shift and stores
   the result in $destination.

   sll
   Type: I-type
   Opcode: 0xA
   Syntax: sll $destination, $source, SHAMT
   Description:
   Shifts $source SHAMT bits to the left and stores
   the result in $destination.
   ```

2. (4 or 5 %) Add your own instruction to your processor. Opcodes 1110 and 1111 are currently unused. Add the hardware required to the datapath to support your instruction. Modify zassemble.c to generate assembly to support your new instruction. Write code that uses your instruction in a program. The number of points depends on the complexity of the instruction that you add and/or your code.

3. (3 %) Add a switch to your design that chooses between single stepping the clock for your datapath and using the free running clock. Connect it to one of the switches on the TUL PYNQ board or on the add-on board. Bonus point if you use the add-on board.

2 Software Projects

1. (5 %) Modify the assembler so that it accepts a data segment as well as text. The assembler will now have to generate two files, one for instruction memory and one for data memory. Write a program to run on your computer that makes use of both instruction and data memory.

2. (3 %) Calculate the average of a list of values. Then display the result on the seven segment display (SSD) on the add-on board. Note that you have to initialize the data memory with the values in a file. Name your file with the extension *.coe. To calculate the average, sum the values and then divide by the total. You can restrict the length of the list to be a power of two. All the steps should be coded in zassembly and run on your lab computer.

3. (5 %) Find the minimum distance from the mean. For a list of values, calculate the mean (item 1) and then find the value with the minimum distance to the mean. To find the distance, subtract the number from the mean. The distance is the absolute value of the result. Display the value of the minimum distance on the SSD on the add-on board. All the steps should be coded in zassembly and run on your lab computer.