EEEE-220-Lab10

1) From mycourses download on your desktop this pdf file. This lab consists of one part.
2) **Objective**: The objective of this lab exercise is to capture and verify the functionality of your fmlRISC control unit design. You can use either VHDL or Verilog.
3) **Part 1: fmlRISC – CU – HDL capture and verification.**
4) Make sure you have all DP and CU related lecture handouts and notes.
5) It is important that your control signal table is complete at this time.
6) Based on the value of Bit2 of your personal design number, you'll have to implement your CU either hardwired or micro-programmed.
7) **For the HW-CU:**
    a. Create the necessary instruction word operation code and machine cycle decoders based on the examples provided in the handout dxp_chapter4b.pdf.
    b. Modify the 8-bit counter to the size required for your machine cycle counter.
    c. Create the control signals combinational logic synthesis block in a separate block/component. I recommend you implement this by converting your SOP functions into HDL dataflow code.
    d. Once you have all components, create the structural description of your CU in the top-level entity. Don't instantiate here IR and C-DFF. These are already instantiated in the DP, and therefore their outputs are input signals to the CU.
    e. For the testbench, you'll have to provide a clock signal, a reset signal, a carry_in signal, and a 4-bit value for the instruction word operation code. These are all your input signals.
    f. Using your DP testbench as a template, assert or verify all control signal values. Use the same sequence of machine cycles you used in the DP testbench.
8) **For the uP-CU:**
    a. Create the uMux, uSeq, and uPC components. Make sure that the reset initialization value for the latter is set to point to the location of the IF0 machine cycle uInstruction.
    b. Create a ROM block with the address and data bus widths necessary in your design.
    c. Create a fml_uP_ROM.mif. I recommend you use for data the binary radix, as shown in the figure below. This will allow you to "copy/paste" the content of your control signal table. I further suggest you create a comments header, in which vertically you capture the name of each control signal. This will be very helpful during debugging.

```
-- uProgram Memory Initialization File
WIDTH = 24;
DEPTH = 32;
ADDRESS_RADIX = DEC;
DATA_RADIX = BIN;
--          N....M.MIPPMMSIDF...M.RM      % Control Signal %
--          A    S CLELHLWOAS   D WW      % Names – arbitrary %
CONTENT BEGIN
0     :     000000001100000000000000;     % IF  0X006000%
1     :     000000110000000000000000;     % EX0 0X018000%
2     :     000000000100100000000100;     % JMPU1 %
3     :     000000100110000000000000;     % JMPU2 %
```

d. Once you have all components, create the structural description of your CU in the top-level entity. Don't instantiate here IR and C-DFF. These are already instantiated in the DP, and therefore their outputs are input signals to the CU.
e. For the testbench, you'll have to provide a clock signal, a reset signal, a carry_in signal, and a 4-bit value for the instruction word operation code. These are all your input signals.
f. Using your DP testbench as a template, assert or verify all control signal values. Use the same sequence of machine cycles you used in the DP testbench.

9) **Grading**:
   a. 2 points for the IF or IF0/IF1.
   b. 20 points for the machine cycles of the following ten instructions: ADD, SUB, INC, DEC, NOT, AND, OR, SHR, SWAP, CPY; 2 points per instruction cycle or 1 point per machine cycle, not counting the IF or IF0/IF1.
   c. 18 points for the remaining six instructions. 3 points per instruction cycle or 1 point per machine cycle, not counting the IF or IF0/IF1.
10) Show your complete design to the TA, preferably during her/his OH, i.e. before the due date and time. Within the first hour of next week's lab she/he won't have time to check all 16-22 designs.
11) Write your report and upload it along with your archived data path project in the dropbox on mycourses, as described in the lab policy.
12) Even if you don't complete your entire design before the due date and time, you should still upload whatever you have up to that point on mycourses!
13) This concludes this week's lab.