CSCE 611
Lab 4
MIPS R-Type Arithmetic Instructions
Due Date: 11/2 (by midnight)

Objective

In this lab you will design and test the fetch, execute, and write-back stages of the MIPS pipeline and implement the following instructions:

- Arithmetic: **`add, sub, addu, subu, mult, multu`**
- Logical: **`and, or, nor, xor`**
- Shift: **`sll, srl, sra`**
- Comparison: **`slt, sltu`**
- Miscellaneous: **`mfhi, mflo, nop`**

Testing Procedure

Instead of designing a full test bench, you will use two programs to test your CPU. **You will still need a minimal test bench for driving the clock and reset inputs.**

Your first program will consist of the program on Lecture 7, slide 21, which tests each instruction once. You will verify the correct operation of this program by using Modelsim to verify the contents of the register file after all the instructions are executed once.

Your second program, which you will write yourself, will continually convert the binary value represented on the DE2 switches as an unsigned integer from 0 – 2^{18}-1 (262143) and display the value on the 7-segment displays. Make sure your design includes the 7-segment decoders from lab 1. For testing this code, use MARS and Modelsim.

I recommend that you use the algorithm in lecture 7 to write this program but this is not required, as long as your program fulfills the requirements. Note that this algorithm will require that you initialize at least two registers (10 and 0.1) using the initial block in the register file Verilog code.

Submit your projects through the course Moodle site (http://dropbox.cse.sc.edu). Upload all Verilog source, assembly code, hexfiles, and testvector files. Also, schedule your demo with the TA.