ECE 486/586

Term Project

Portland State University

Basics

- **Objective**
  - Develop an execution-driven MIPS-lite pipeline simulator
- **Programming Language**
  - Any high-level programming language (C, C++, JAVA etc.)
- **Simulator Inputs**
  - Memory image for the simulated program
  - Provided by the instructor
- **Simulator Output**
  - Program output (register values, memory contents)
  - Instruction type frequency statistics
  - Execution time in cycles

Simulator Components

- **Trace Reader**
  - Reads the memory image and passes the next instruction to the instruction decoder
- **Instruction Decoder**
  - Interprets instruction type , determines the source and destination registers
- **Functional Simulator**
  - Simulates instruction behavior, keeps track of register and memory state changes
- **Pipeline Simulator**
  - Keeps track of current clock cycle
  - Maintains track of instruction in each pipeline stage in each cycle
  - Identify different sources of stalls and hazards
  - Propagates instructions from one pipeline stage to next

Simulator Details

- You will write a simulator which models both the functional and timing behavior of a 5-stage MIPS-like pipelined processor
  - Pipeline details covered in class (Lectures slides 5a, 5b, 5c, 5d)
- You will need to do two things:
  - Simulate the computation performed by an instruction and record its impact on the machine state
  - Quantify the impact of instruction execution on the program execution time
    - You will simulate if this instruction needs to be stalled and what is the stall penalty
    - You will have to **visualize** the 5-stage pipeline and the instruction in every stage, and then program your simulator with that in mind

Logistics and Timeline

- You should form groups of up to 4 students
- Your simulator implementation should follow the detailed project specification document posted on Canvas
- Each group will be provided with trace(s) that will be used to test the simulator
- At the completion of the project, you will need to turn in a project report and all your source code. The project report should include all the simulation results and also put your code in the Appendix of the report, in addition to turn in the source code files. Zip all files into one .zip file to turn in on course Canvas.
- Important Dates
  - Self assign Project Team on course Canvas by: Friday, April 11
  - Project specs uploaded on course website: Tuesday of Week 1
  - Final project report due: Friday of Week 10, June 6