CS204 - Computer Architecture
Course Project Phase 2 Details
Pipelined Implementation of 32bit RISC-V instruction execution

Deadline: 3rd Apr 2023. 11.55PM.

General guidelines:
1. You are expected to build on your Phase 1 code, and make it pipelined.
2. Introduce pipeline registers/inter-stage buffers between each of the execution steps.
3. Change the data path and control path to support pipelining.
4. You need to introduce logic of data forwarding, stalling, flushing the pipeline.
5. You'll need give two memories: one for text segment and another for data segment.
6. For control hazards, include one-bit branch prediction logic.

Input knobs that need to be included:

1. Enable/disable pipelining.
   If Knob1 is set, it would work as pipelined execution. If not set, its similar to Phase 1.

2. Enable/disable data forwarding.
   If this knob is not set, pipeline is expected to work with stalling.

3. Enable/disable printing values in the register file at the end of each cycle.

4. Enable/disable printing information in the pipeline registers at the end of each cycle (similar to tracing), along with cycle number.

5. This knob is like enabling Knob4 for a specific instruction. With this feature we will be able to see the pipeline registers information for a particular instruction of our interest. Here, the instruction can be specified as number (example, if the instruction we are interestd in is the 10th instruction in the input program, 10 will be taken as input).

If you notice, many of the knobs help you in debugging, during code development time. So do not keep them for the end.

Stats to be printed in an output file at the end of the simulation.

- Stat1: Total number of cycles
- Stat2: Total instructions executed
- Stat3: CPI
- Stat4: Number of Data-transfer (load and store) instructions executed
- Stat5: Number of ALU instructions executed
- Stat6: Number of Control instructions executed

- Stat7: Number of stalls/bubbles in the pipeline
- Stat8: Number of data hazards
- Stat9: Number of control hazards
- Stat10: Number of branch mispredictions
- Stat11: Number of stalls due to data hazards
- Stat12: Number of stalls due to control hazards

Test cases for Phase 2 are similar to Phase 1.

GUI requirements for bonus points (1%):
1. A block diagram of the five pipeline stages.
2. In each cycle, give details of the instruction being executed in each stage during that cycle.
3. Highlight the data hazard and control hazard cases.
4. Highlight the forwarding path taken to handle the data hazard.
5. Show the output of the 1-bit branch predictor whenever used.

Revert if you have any questions.