Computer Architecture Project

Multi-Cycle Processor Design

All instructions ended up working fine to the point to which we could verify:

- Addition (all 3 addressing modes)
- Subtraction (all 3 addressing modes)
- Jump
- Logical OR (both addressing modes)
- Logical NAND (both addressing modes)
- Branch Equal
- Branch Not Equal
- Load Word
- Store Word
- Shift (all 3 variations)

These all instructions are giving the output as intended at the ALUOut and all of them are also being written back to the RegFile and Data memory. Instruction memory is also working as intended. On the other hand, PC updation is not working properly, so it works in some cases but fails in others. The code for the PC updating was checked multiple times and also implemented in a couple of different ways but we still could not get it to work for all the cases.

Of the 5 major components (PC, Data Memory, Instruction Memory, Reg File, ALU) only PC could not be integrated properly in the final datapath module. Others were integrated perfectly and all stages of the multicycle datapath (IF, ID, EX, MEM, WB) were checked and were being followed by the respective instructions. While testing, due to PC not being functional, we could not check multiple instructions at a time and hence had to check each individual instruction one after the other. We hardcoded the instruction memory, data memory and RegFile values and checked for each instruction separately. This can be found in the code as we have labelled this via the comment (/* HARDCODED*/).

All individual module test benches are appended at the end of the code and commented out for testing them out.

Submitted by:

Ram Goyal (2019A3PS0239P)

Tushar Sardana (2019A3PS0260P)

Aditya Agarwal (2019A3PS0202P)