**EC413 Project Report**

Wasim Khan & Gavin Hiroe

All instructions go through states 0 and 1. State 0 fetches the next instruction by incrementing the PC. State 1 decodes the instruction and calculates the next instruction in the PC. This also prepares the CPU to branch if needed.

**NOP**

State 2 will set all control lines to 0, thereby creating a bubble.

**R-Type**

In State 3, the two Reg values will be read into the ALU from the A and B registers, and the opcode will provide the operation done in the ALU.

In State 14, the ALU output will be written back into the register file.

**JMP**

In State 4, the next instruction address will be read into the PC from the 26-bit immediate given in the instruction. This immediate is zero extended to accommodate 32 bits and is then given to the PC.

**Branches**

All branches go to state 19, where the register file is read to obtain the operands that will let the ALU calculate the branch target. After state 19, all branches will have the ALU subtract the two operands, and PCWriteCond will be turned on so that the proper next instruction is obtained from the PC. BranchCond will be set based on the type of branch and output a 1 or 0 to either let the PC go to the branch target or the subsequent instruction instead.

**I-Type (SE)**

In state 9, the ALU operands will be the A register and the sign extended immediate.

In State 14, the ALU output will be written back into the register file.

**I-Type (ZE)**

In state 9, the ALU operands will be the A register and the zero extended immediate.

In State 14, the ALU output will be written back into the register file.

**LI**

State 11 simply writes the immediate data from DMEM to the register file

**SWI**

State 12 enables the register file to be read.

State 17 writes the zero extended immediate to DMEM.

**LW & SW**

In state 13, the ALU will take register A and the sign extended immediate as operands and add them to obtain the correct location in DMEM.

Then, both instructions will take the ALUOut result and either read the data at that address for LW or write the result to DMEM for SW. For the LW instruction, there is an extra state that writes the read data to the register file.

**LWI**

State 18 will simply write the appropriate data from DMEM to the register file.