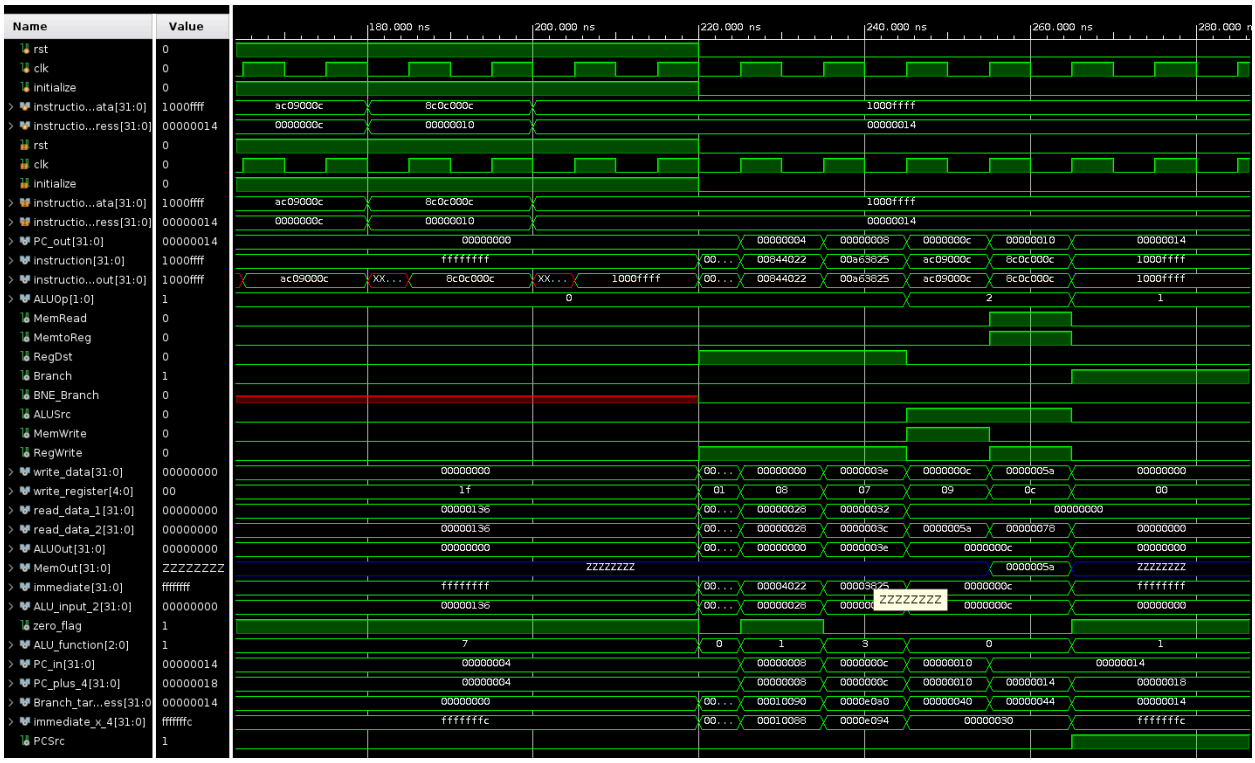
Leonardo Mattos Martins
U25267206
Lab 6 Writeup

Lab Tasks

1. Simulate the Basic CPU




The CPU has reset portion for 220 ns before executing the instructions provided in the tb_cpu.v testbench. This CPU is implemented in a single-cycled way, where no pipeline registers delay the execution of an instruction for longer than one clock cycle.

2. SLT

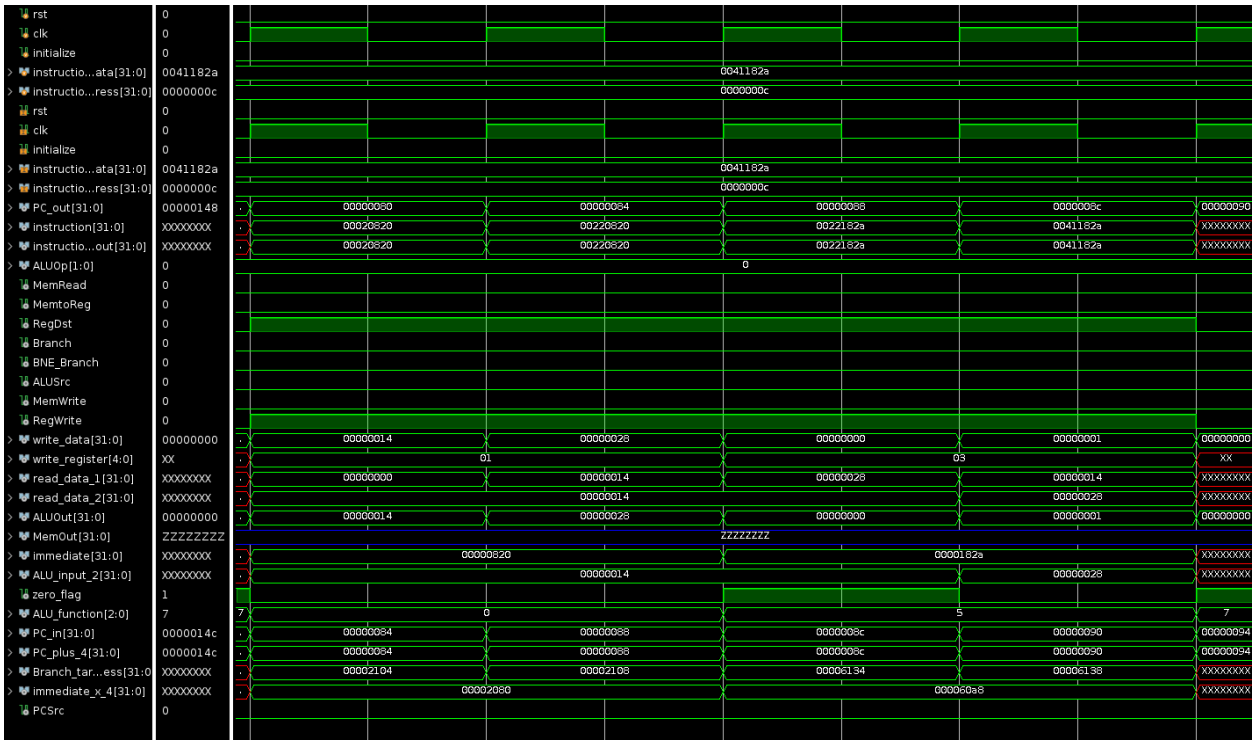
Instructions:

ADD R1,R0,R2

ADD R1,R0,R2

SLT R3,R1,R2

SLT R3,R2,R1




Modifications:

ALU_Control.v - added the slt instruction (func = 3’d5)

ALU.v - added slt function (a<b)?1:0

3. ADDI

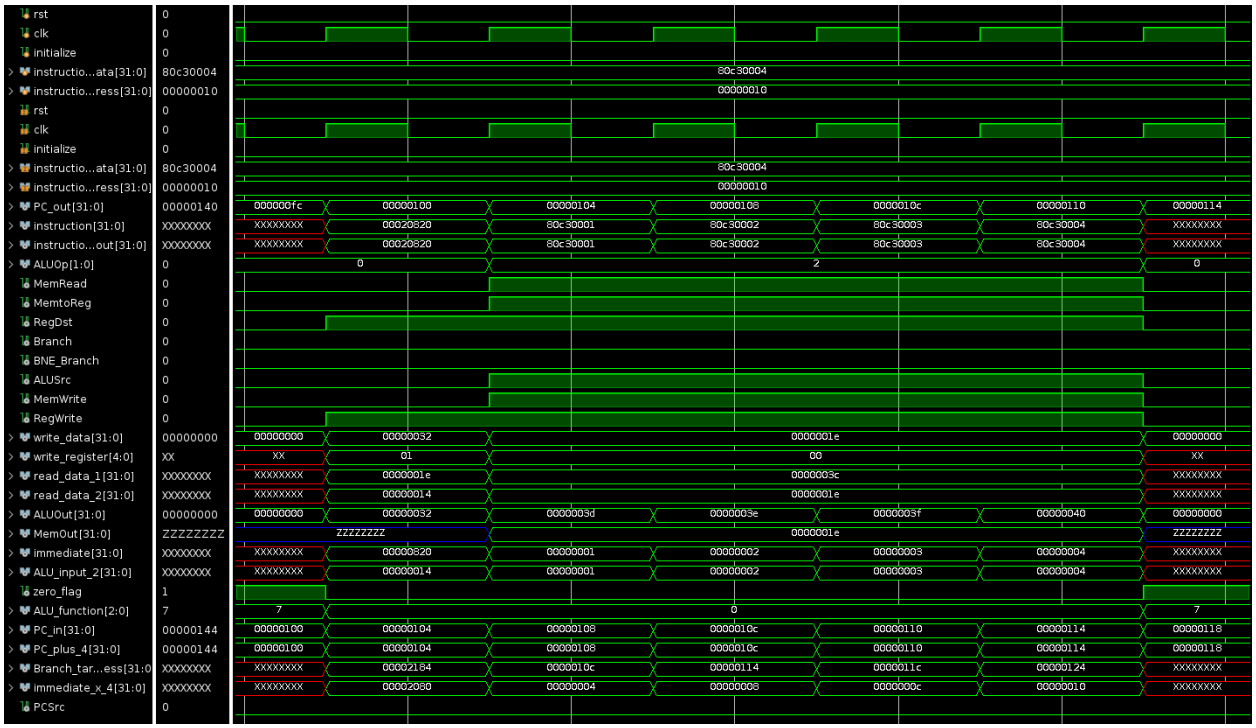
   ADD R1,R0,R2

   ADDI R3,R2,1

   ADDI R3,R2,2

   ADDI R3,R2,3

   ADDI R3,R2,4




Modifications:

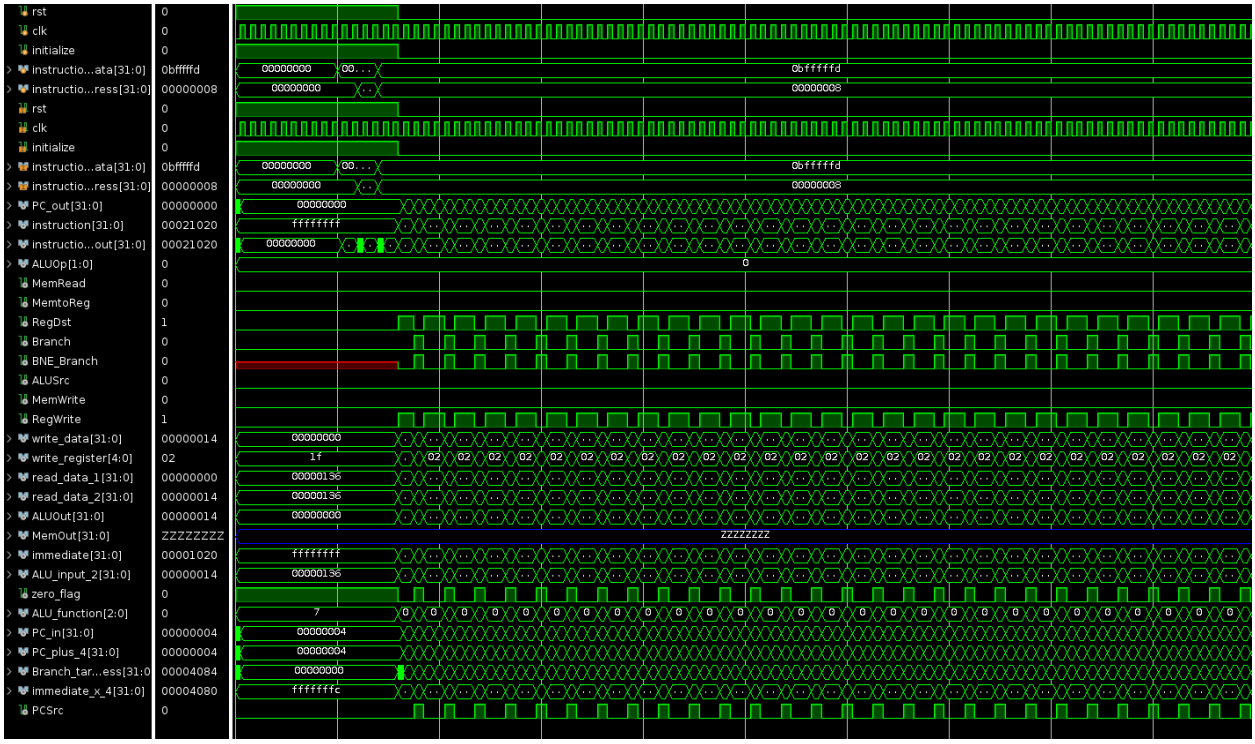
control.v - added ADDI instruction with required opcode and the following settings (ALUOp = 2'b10; MemRead = 1'b1; MemtoReg = 1'b1; RegDst = 1'b1; Branch = 1'b0; ALUSrc = 1'b1; MemWrite = 1'b1; RegWrite = 1'b1; BNE_Branch = 1'b0;)

4. J

   ADD R2,R0,R2

   ADD R2,R0,R2

   J -3




   Modifications:

   cpu.v - Added a BNE_Branch control signal for both BNE and J instructions. For a J instruction, both BEQ and BNE_Branch signals are enabled such that it will jump regardless of BEQ or BNE.

   control.v - Added the BNE_Branch control signal to each of the other options in control, and added J instruction with the following settings (ALUOp = 2'b00; MemRead = 1'b0; MemtoReg = 1'b0; RegDst = 1'b0; Branch = 1'b1; ALUSrc = 1'b0; MemWrite = 1'b0; RegWrite = 1'b0;)

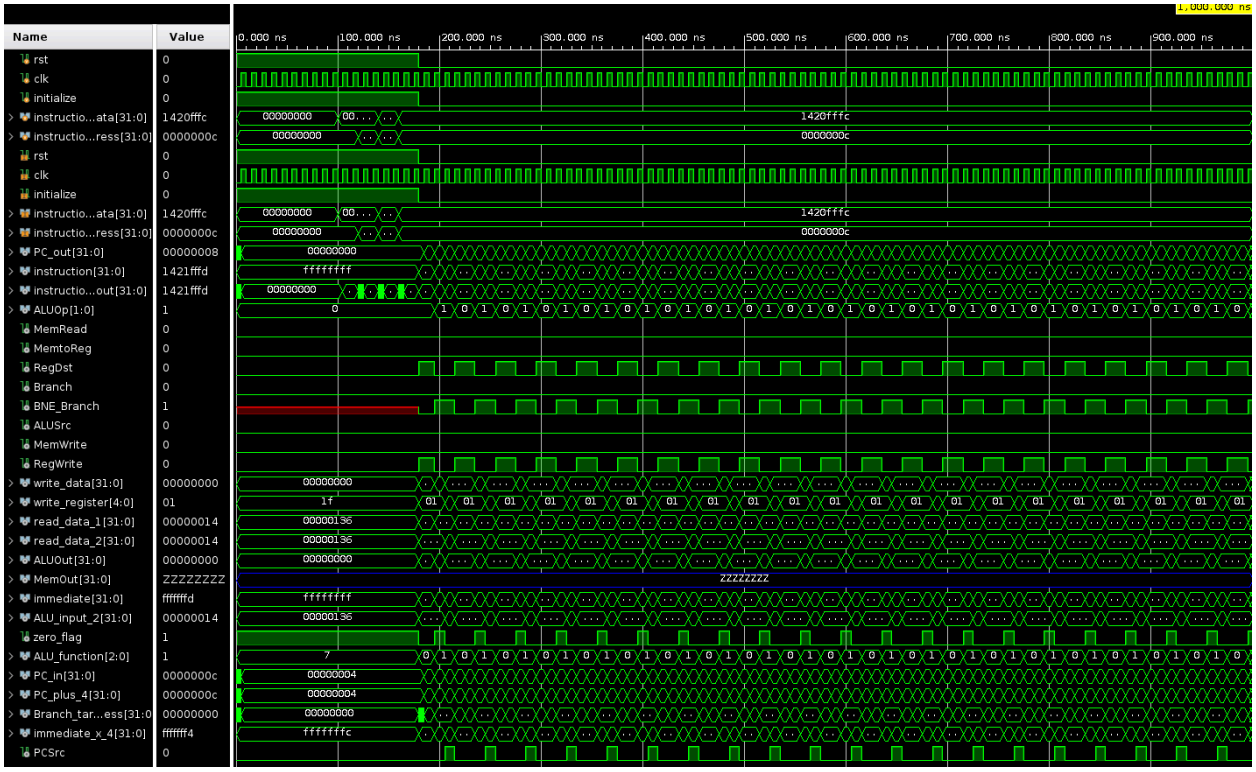
5. BNE

   ADD R1,R0,R2

   ADD R1,R0,R2

   BNE R1,R1,-3

   BNE R1,R2,-4




Modifications:

cpu.v - Added a BNE_Branch control signal and additional logic to branch if Branch and the zero_flag is on or if BNE_Branch and not zero_flag is on.

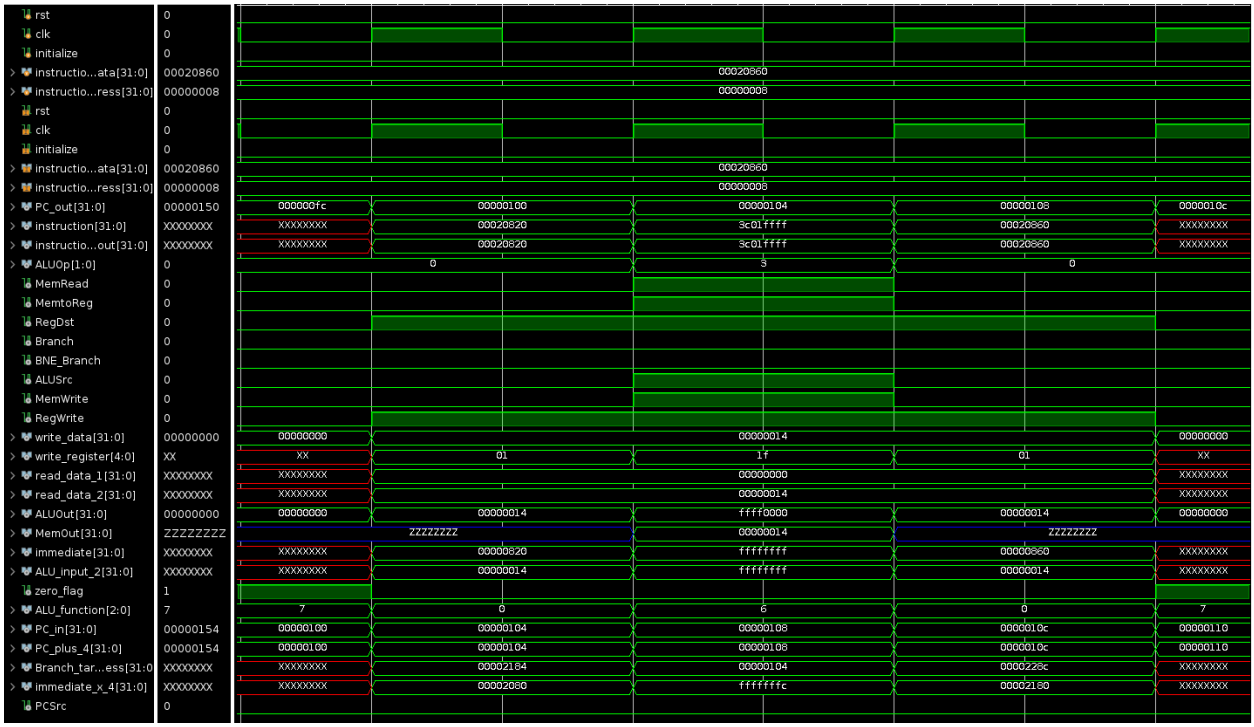
control.v - Added BNE instruction with the following settings (ALUOp = 2'b01; MemRead = 1'b0; MemtoReg = 1'b0; RegDst = 1'b0; Branch = 1'b0; ALUSrc = 1'b0; MemWrite = 1'b0; RegWrite = 1'b0; BNE_Branch = 1'b1)

6. LUI

   ADD R1,R0,R2

   LUI 0xFFFF

   ADD R1,R1,R2




control.v - added LUI instruction with the following settings (ALUOp = 2'b11; MemRead = 1'b1; MemtoReg = 1'b1; RegDst = 1'b1; Branch = 1'b0; ALUSrc = 1'b1; MemWrite = 1'b1; RegWrite = 1'b1; BNE_Branch = 1'b0)

ALU_control.v - added new function code (3’d6) for a <<16 instruction corresponding to ALUOp = 2’b11, which will shift the immediate to the upper 16 bits that can then be saved into the given register.

ALU.v - added <<16 function corresponding to function code 3’d6.