EC413 Discussion 8
Lab 6

Xiteng Yao

Lab 6

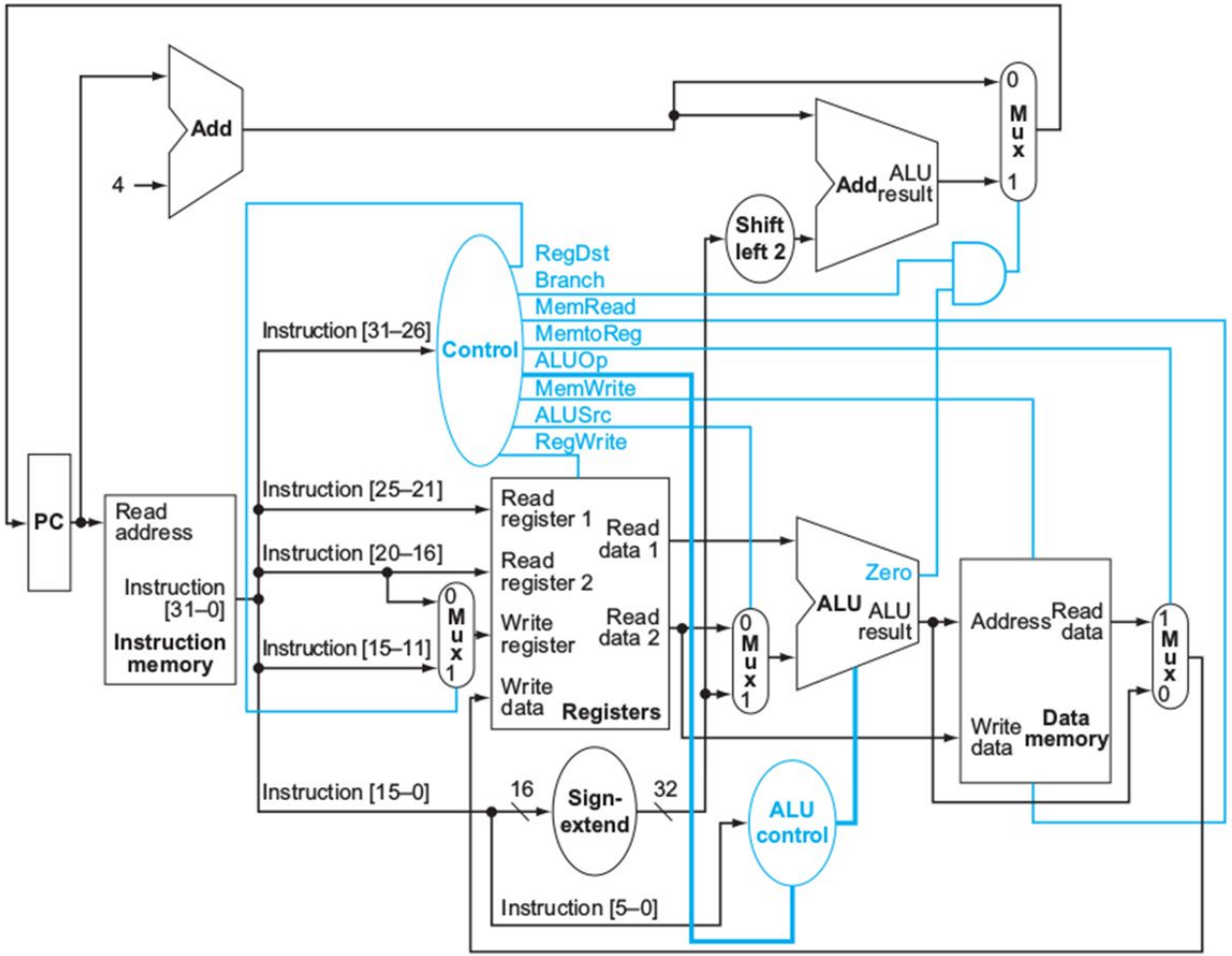
Tasks

- Simulate CPU operations
- Add SLT
- Add ADDI
- Add J
- Add BNE
- Add LUI

CPU Given

Just run the testbench.

Notice that the instructions do not start executing until reset is set to 0. Then the PC starts which triggers the CPU.



SLT - Set Less Than

Just modify the ALU and ALU control unit, the function code for SLT is 42 or 0x2a in the ALU Control

```
else if (func == 3'd5) // added for SLT
out = (a<b) ? 1'b1: 1'b0;
```

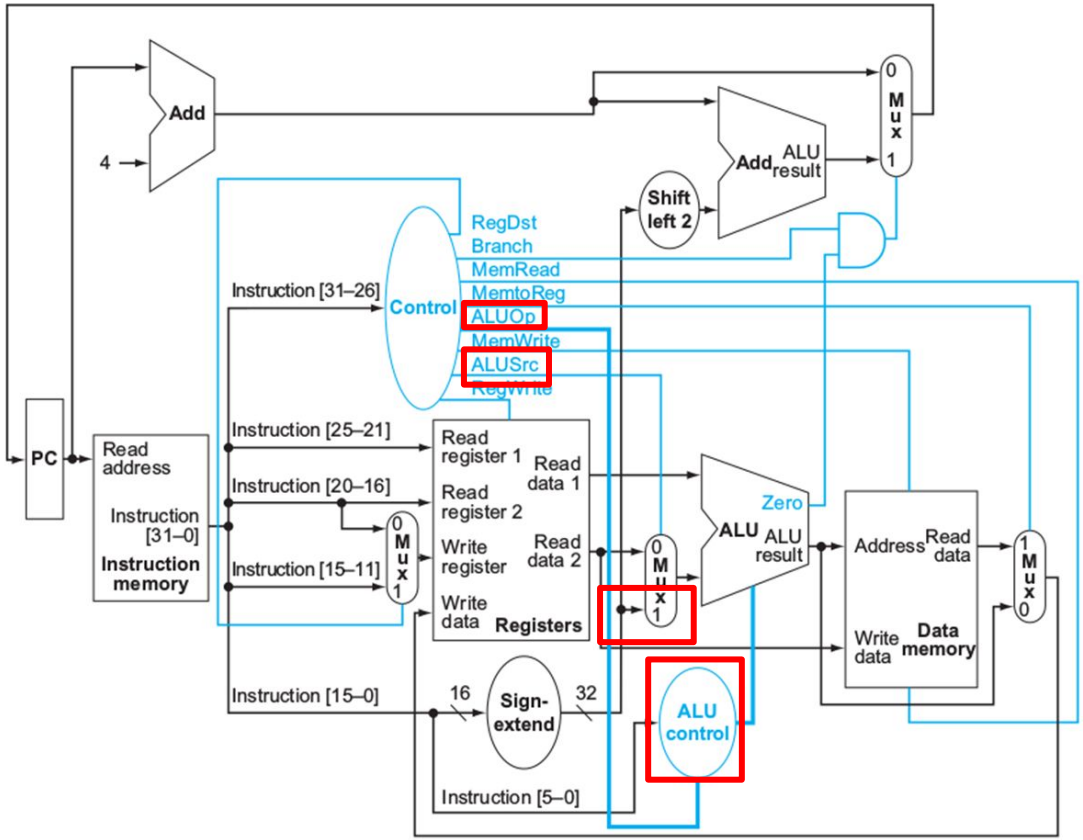
```
else if (instruction == 6'h2a) // added for SLT
func = 3'd5;
else
func = 3'd7;
```

ADDI - Add Immediate

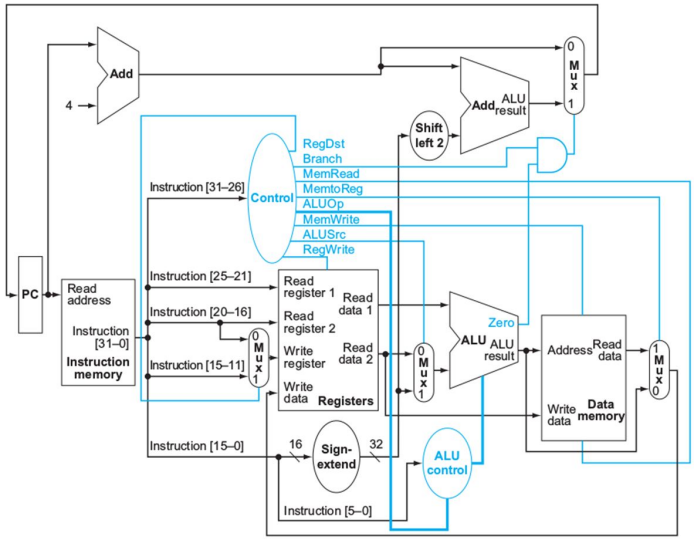
Add a separate case in the control module that is similar to the ADD case

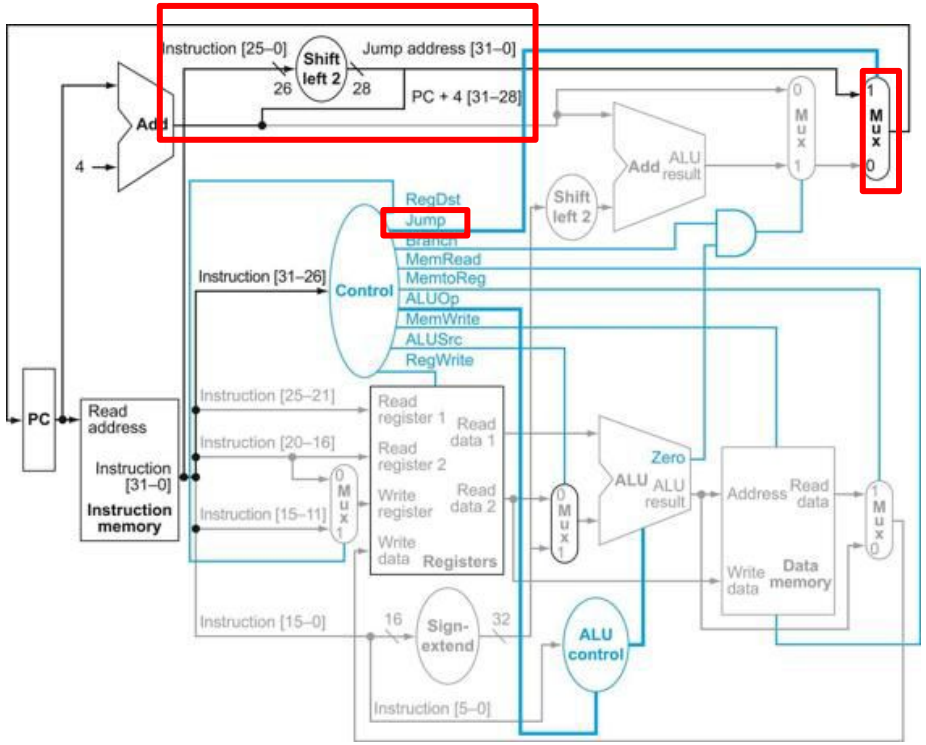
Set the ALU data source to immediate for using immediate as ADDI input (ALUSrc = 1'b1)

Also modify the ALUOp so that the ALU know this is an I type Add instruction (2'b10)



J - Jump





To do:

Add a Shift left by 2 module, a new mux, and corresponding control signal

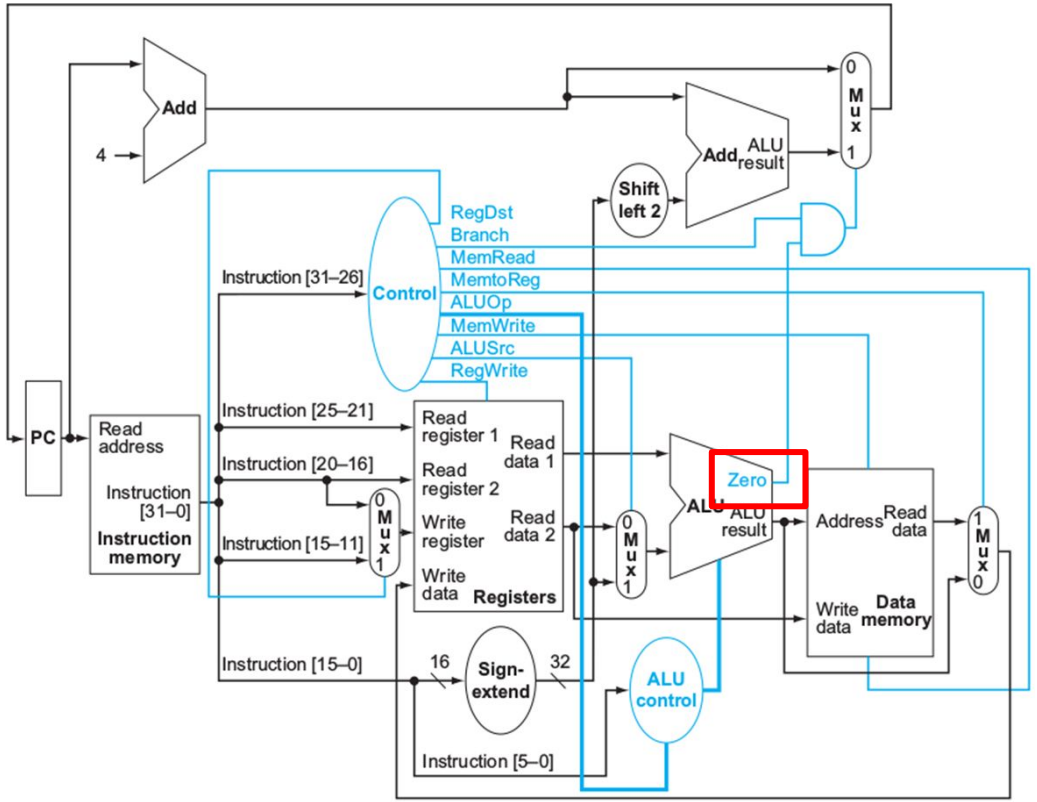
H&P Figure 4.24

BNE - Branch On Not Equal

BEQ is already implemented, so we only need to slightly modify it

BEQ uses the zero flag in the ALU, which goes to 1 when both inputs are equal

We need to add a new wire to negate the zero flag, and add a new signal from the control module to select from the zero flags

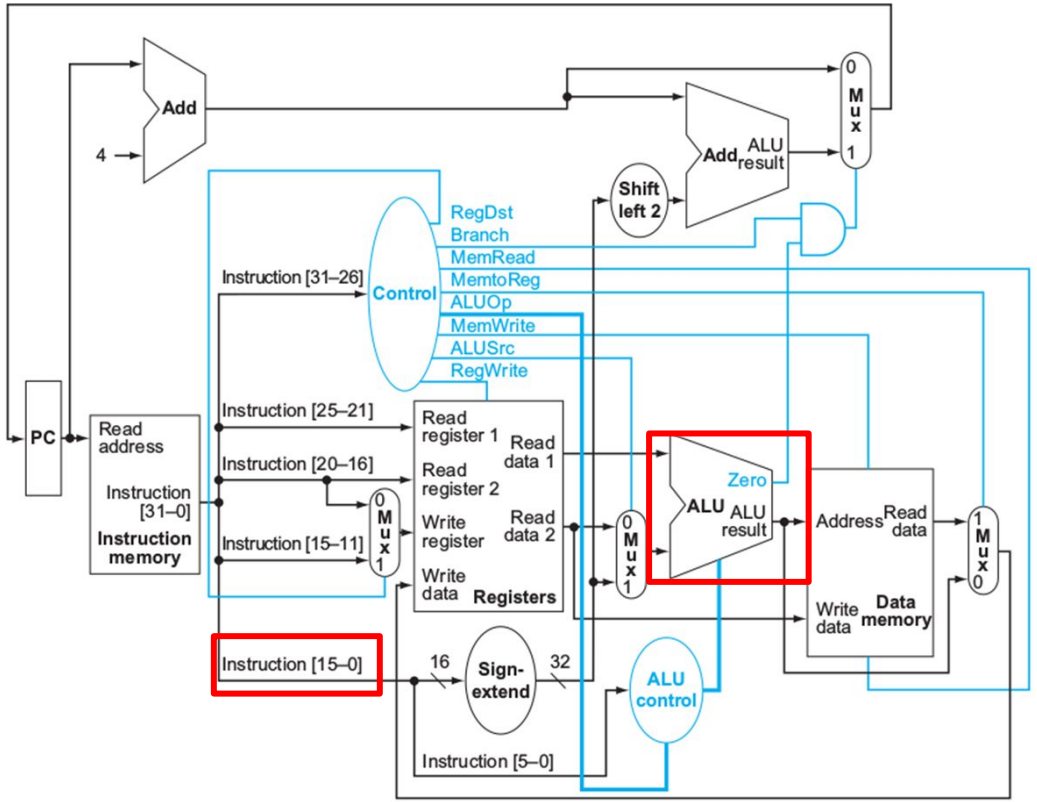


LUI - Load Upper Immediate

The lower 16 bits of an instruction is immediate

We can implement a shift left module and use a signal from the control module to control a mux to select from the two signals

Implement a pass through in the ALU module



Extra Credit and Notes

EC for this lab is more complicated but have more credits.

1(001)	add immediate	addiu	set less than imm.	set less than imm. unsigned	andi	ori	xori	load upper immediate

In the testbench, there is a BEQ instruction that will change the program counter, it may skip later instructions. For your convenience, you may comment out that when testing your instructions.