# ICS 152, Problem Set 3

- Please show your work.
- Bottom line answers without proper explanation are worth **zero** points.

---

**1.** We wish to add an instruction that is a variant of *lw* (load word), which sums two registers to obtain the address of the data to be loaded and uses the R-format.
Add any necessary datapaths and control signals to the single cycle datapath described in *Figure 1*. Update the table shown in *Figure 2* to include the new instruction, the new control signals and any other changes you make.

**2.** Explain why it is impossible to modify the single cycle implementation described in *Figure 1* (without changing the register file) to support a "load with increment" instruction that performs the following operations?

*lw $ra, offset($rb)*
*addi $rb, $rb, 4*

**3.** Text Problem 5.14

**4.** Text Problem 5.28

**5.** Text Problem 5.36

**6.** Text Problem 5.37

# ICS 152, Problem Set 3

- Please show your work.
- Bottom line answers without proper explanation are worth **zero** points.

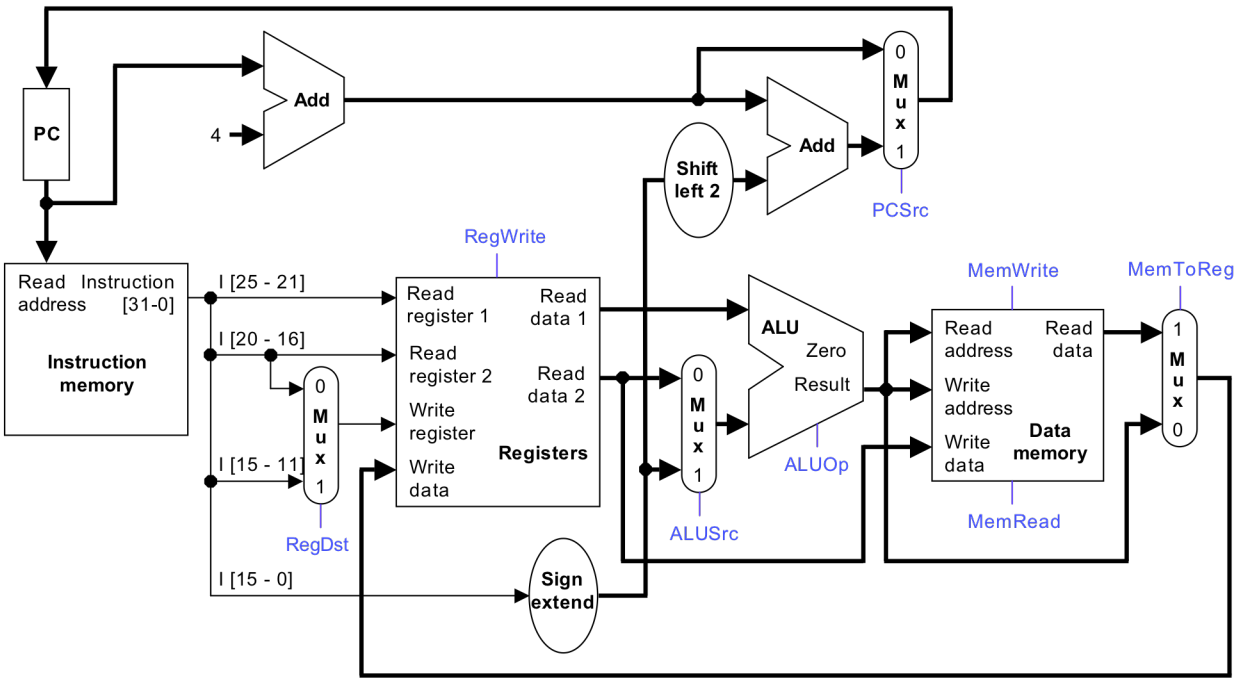


**Figure 1.** The basic single cycle MIPS datapath

| Instruction | RegDst | ALUSrc | MemToReg | RegWrite | MemRead | MemWrite | Branch | ALUOp1 | ALUOp0 |
|---|---|---|---|---|---|---|---|---|---|
| R-format | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| Lw | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Sw | x | 1 | x | 0 | 0 | 1 | 0 | 0 | 0 |
| beq | x | 0 | x | 0 | 0 | 0 | 1 | 0 | 1 |

**Figure 2.** Control signal values for the various instruction formats