Ira A. Fulton Schools of Engineering
School of Computing, Informatics and Decision Systems Engineering

SER 450 **COMPUTER ARCHITECTURE**

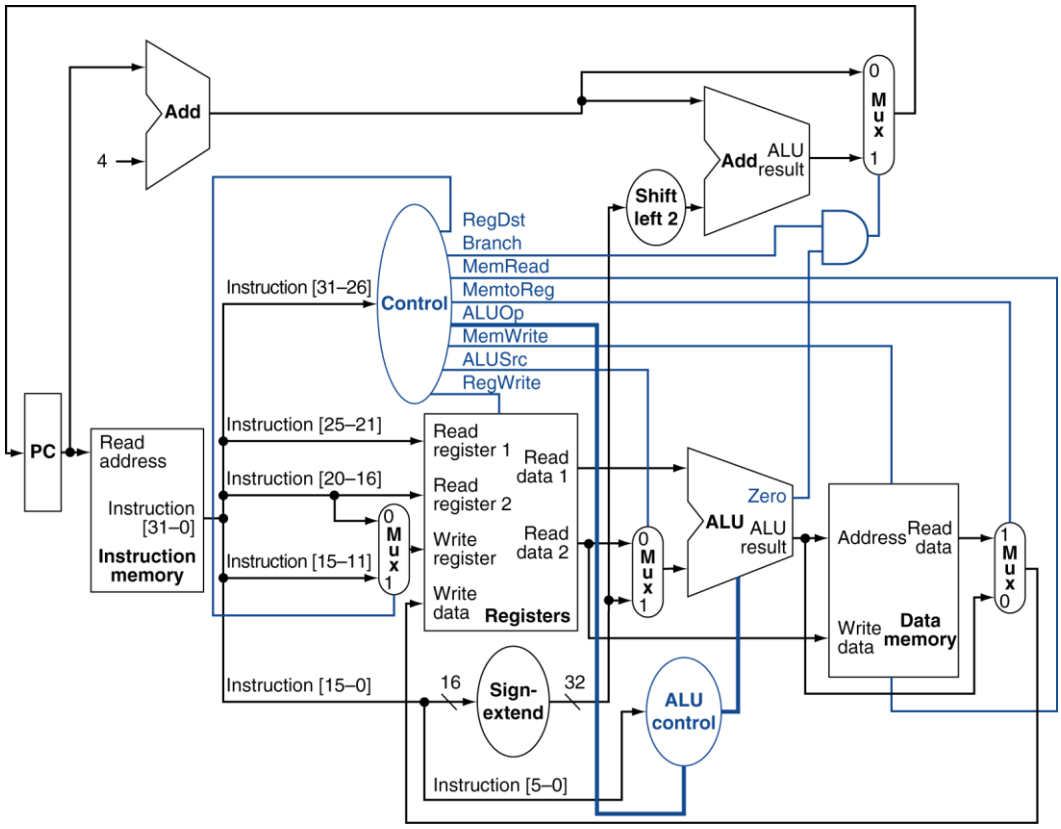
Your Name: Jacob Hreshchyshyn

When completing this worksheet, please use a different color text or typeface so that we can easily identify your work. Remember to show your work / give justification for your answers.

Portions of this material are derived from "Computer Organization and Design", Patterson & Hennessy.

PRACTICE PROBLEMS 4A

For problems in this practice set, it may be helpful to refer to the following diagram of the single-cycle processor and control.



SER 450 **COMPUTER ARCHITECTURE**

Problem 1:

Consider the following instruction:

```
AND Rd, Rs, Rt

Interpretation:  Reg[Rd] = Reg[Rs] AND Reg[Rt]
```

What are the values of the control signals generated by this control function?

RegDst	1_2	ALUOp	10_2
Branch	0_2	MemWrite	0_2
MemRead	0_2	ALUSrc	0_2
MemtoReg	0_2	RegWrite	1_2

The above values are justified by Figure 4.18 in page 266 of the textbook.

Which resources/blocks perform useful function for this instruction?

Figure 4.19 in page 266 of the textbook helps illustrate the resources performing useful function for the instruction. Essentially, every resource except the Sign Extension resource, the Data Memory resource, the Shift left 2 resource, the Add ALU resource, and the Branch And Gate resource perform useful function for the instruction. All exceptions are grayed out, indicating no use when executing the instruction.

Which resources/blocks produce outputs, but their outputs are not used for this instruction? Which produce no outputs?

Outputs that are not used include the Branch And Gate. Resources that produce no outputs include the Sign Extension resource and the Data Memory resource.

Problem 2:

The basic single-cycle MIPS implementation in the provided figure can only implement some instructions. New instructions can be added to an existing instruction set architecture (ISA), but the decision whether or not to do that depends, among other things, on the const and complexity the proposed addition introduces into the processor datapath and control. For this problem, refer to the following proposed new instruction:

```
LWI Rt,Rd(Rs)

Interpretation:  Reg[Rt] = Mem[Reg[Rd] + Reg[Rs]]
```

Which existing blocks, if any, can be reused for this instruction?

Figure 4.20 in page 267 of the textbook helps illustrate the resources used in a load instruction, which helps illustrate what resources can be reused for this new instruction. Because of the similar functionality between LW and the proposed LWI, all resources highlighted in the figure should be able to be reused for this instruction.

Which new functional blocks (if any) are needed for this instruction?

Since two registers are being read to determine what is to be stored in the destination register, the second Read Register port in the Registers resource will be needed. There also needs to be a way to calculate the addition of the two registers, which can be accomplished using another ALU.

What new signals do we need (if any) from the control unit to support this instruction?

The addition of a new ALU requires a new control signal to indicate the need for its use.

Problem 3:

When processor designers consider a possible improvement to the processor datapath, the decisions usually depends on the cost/performance trade-off. For this problem, assume that we are starting with the datapath shown in the provided figure, the different elements have latencies and costs as show in the following table

	I-Mem	Add	Mux	ALU	Regs	D-Mem	Control
Latency	400ps	100ps	30ps	120ps	200ps	350ps	100ps
Cost	1000	30	10	100	200	2000	500

Consider adding a multiplier to the ALU. This addition will add 300ps to the latency of the ALU and will add a cost of 600 to the ALU. The result will be 5% fewer instructions executed because we will no longer need to emulate MUL instructions.

What is the clock cycle time with and without this improvement?

From the slides, we know that we can determine the clock cycle time by considering the longest delay. This is represented by the critical path, which is the load instruction. The path is from I-Mem, to Regs, to ALU, to D-Mem, to Mux, which feeds the data back to Regs. Therefore the clock cycle time without improvement would be:

400 ps + 200 ps + 30 ps (because there is still a need to select the correct ALU input) + 120 ps + 350 ps + 30 ps = 1130 ps without improvement.

The improved ALU increases the cycle time of the ALU from 200 ps to 300 ps, an increase of 100 ps. Therefore, we can find the new cycle time by adding 100 ps to the cycle time without the improvement.

1130 ps + 100 ps = 1230 ps with improvement.

What is the speedup achieved by adding this improvement?

It helps to remember that CPU Time = IC * CPI * Clock Cycle Time. We can find the speedup by finding CPU Time Old/CPU Time New.

(IC * CPI * 1130 ps)/(0.95 IC * CPI * 1230 ps).

IC and CPI are the same between the two times. IC with the improvement should be reduced by 5%, hence the multiplication of 0.95 to IC in the denominator. With terms cancelled out, we get the following expression:

1130ps/(0.95 * 1230 ps) = 0.9670517758.

Thus, there is no improvement adding the multiplier to the ALU.

Compare the cost/performance ratio with and without this improvement

To find the total cost, we must refer to the diagram of our single-cycle processor control, associate costs with each component, and sum it all up. We know we have 1 I-Mem, 1 Regs, 1 Control, 1 main ALU, 1 D-Mem, 2 Adders, and 4 Mux's.

1000 + 200 + 500 + 100 + 2000 + 2 * 30 + 4 * 10 = $3900

Using the new ALU, the cost would be as follows:

1000 + 200 + 500 + 600 + 2000 + 2 * 30 + 4 * 10 = $4400

The relative cost is $4400/3900 = 1.128205128

The cost/performance can then be obtained using the speedup calculated earlier.

1.128205128/0.9670517758 = 1.16 approximately with the improvement.

Without the improvement, the cost/performance ratio would be 1:1 simply because we already have the cost with the unimproved ALU, so there is nothing to compare this cost to but itself.