4/5/23

Q2.

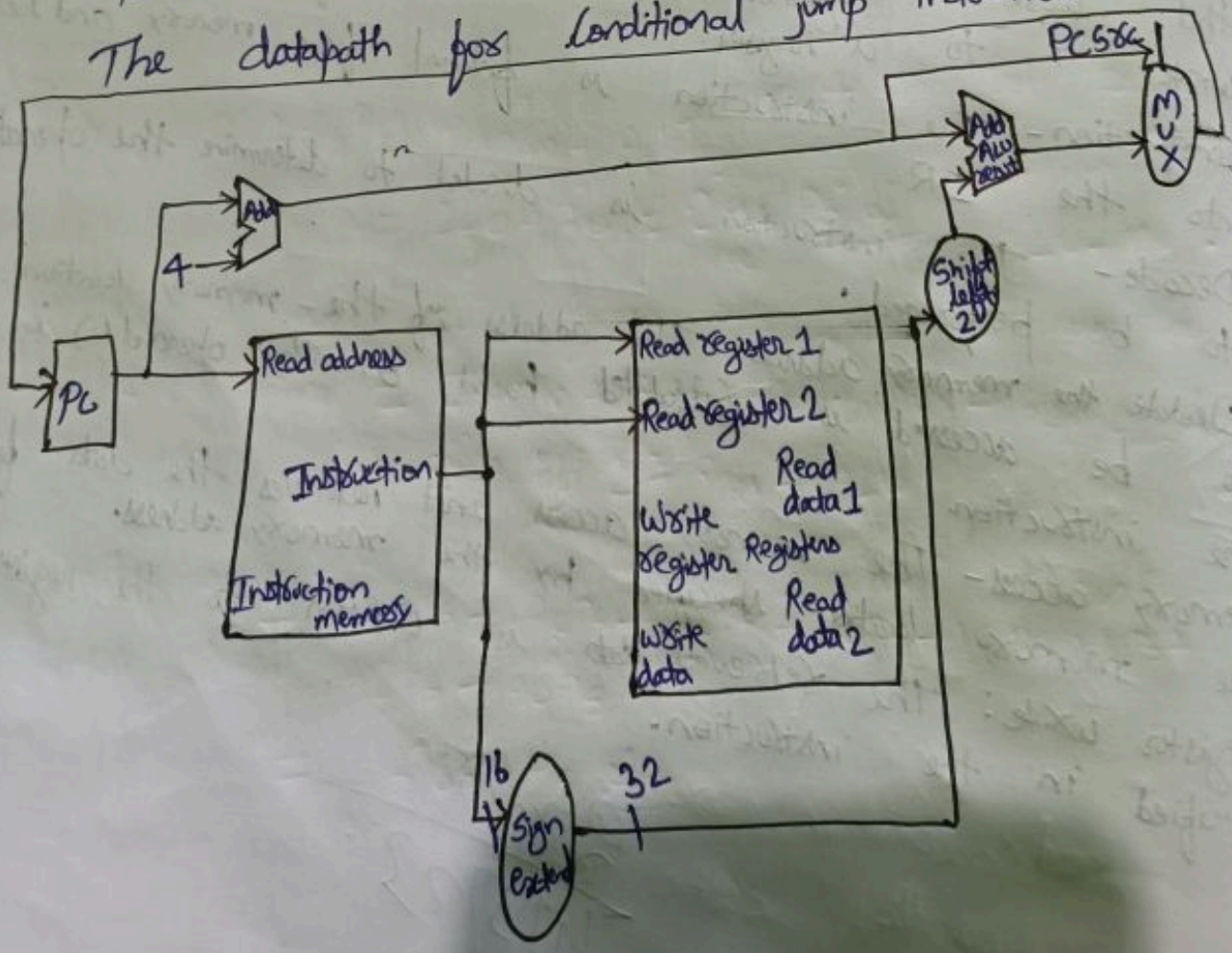
Soln

Assignment - 8

Q1) Design and explain datapath for Conditional jump instruction.

Soln The datapath for Conditional jump ; in Conditional jump two registers are read depending on their condition the program counter may take new address if condition is true. To calculate the jump address, if the condition is satisfied then jump address taken and if the condition is not satisfied then sequential instruction taken. Hence a multiplexer is there to select the conditional jump address.

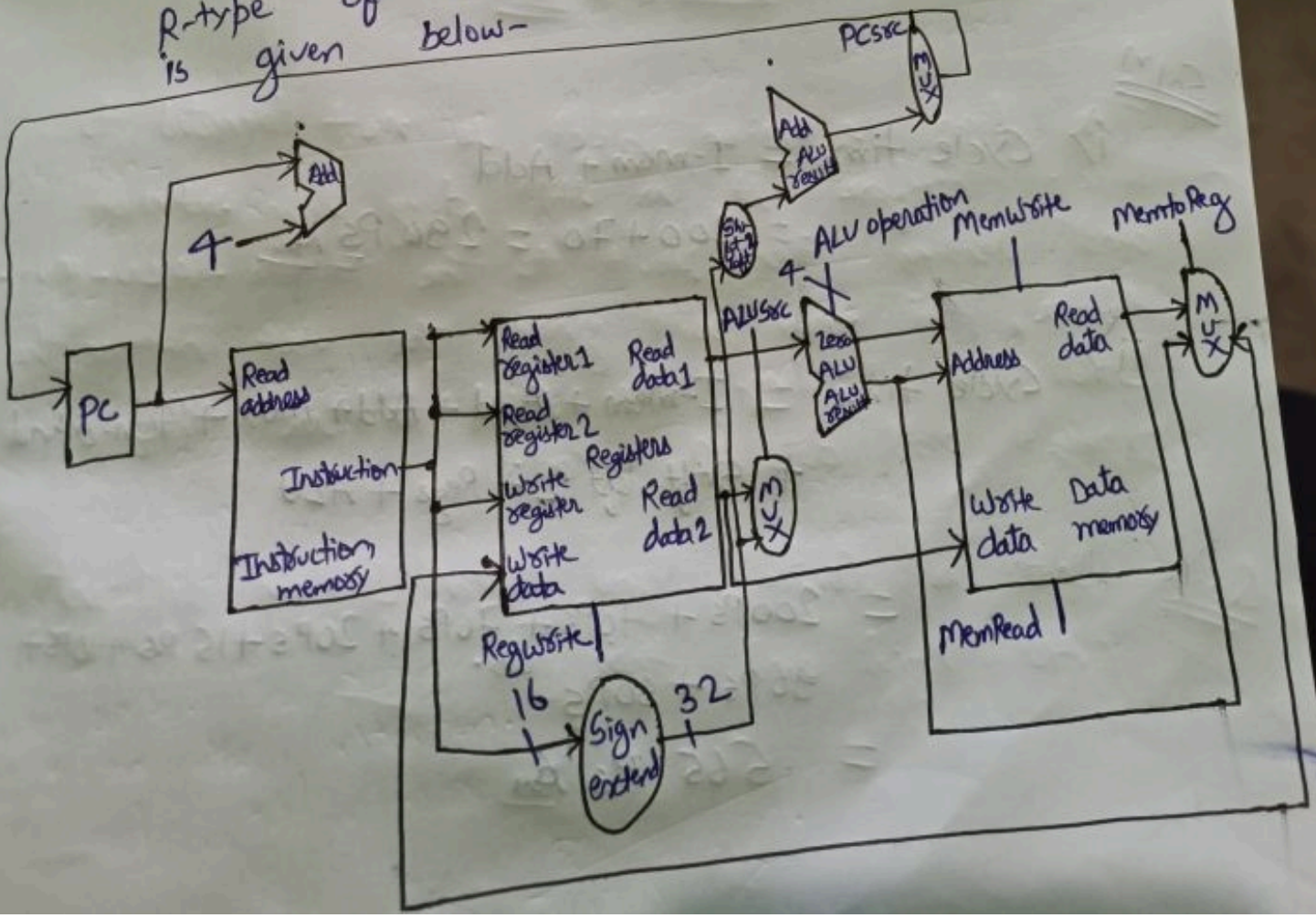
The datapath for conditional jump instruction is—



Q2. Design and explain the datapath for R-type of instruction and I-type instruction together.

Soln In this instruction, the 16 bit data of the instruction is sign-extended and shift left by 2 and fed to a adder along with the result of address of sequential instruction, to calculate the jump address. If the condition is satisfied then jump address taken and if the condition is not satisfied then sequential instruction taken. Hence, a multiplexer is there to select the conditional jump address or sequential instruction address.

The design of the datapath for R-type of instruction and I-type instruction is given below-



Q3) Problems in this exercise assume that logic blocks needed to implement a processor's datapath have the following latencies:

I-mem	Add	Mux	ALU	Regs	D-Mem	Sign-Extend	Shift-left-2
200Ps	70Ps	20Ps	90Ps	90Ps	250Ps	15Ps	10Ps

i) If the only thing we need to do in a processor is fetch consecutive instructions, what would the cycle time be?

ii) What would the cycle time for conditional PC-relative branches?

iii) What would the cycle time for R-type instruction, lw instruction, sw instruction?

Soln

i) Cycle time = I-mem + Add
= 200 + 70 = 290 Ps Ans

ii) Cycle time = I-mem + Add + Add + Mux + Sign-Extend + Shift-left 2 + Regs + ALU

= 200Ps + 70Ps + 70Ps + 20Ps + 15 Ps + 10Ps + 90Ps + 90Ps

= 565 Ps Ans

iii) Cycle time for R-type instruction

= I-mem + Add + ALU + Regs +

= 200 Ps + 70Ps + 90Ps + 90Ps

= 450Ps

Cycle time for lw instruction

= I-mem + Add + ALU + Regs + Sign-Extend

= 200Ps + 70Ps + 90Ps + 90Ps + 15 Ps

= 465 Ps

Cycle time for sw instruction

= I-mem + Add + ALU + Regs + Sign-Extend

= 465 PS (Same as lw instruction)

Q4.) For the problems in this exercise, assume that there are no pipeline stalls and that the breakdown of executed instructions is as follows:

add	addi	not	beq	lw	sw
20%	20%	0%	25%	25%	10%

i) In what fraction all cycles is the data memory used?

ii) In what fraction of all cycles is the input of the sign-extend circuit needed? what is this circuit doing in cycles in which its input is not needed?

Soln

i) Data memory is used to lw and sw
So fraction to which data memory used

$= lw + sw$
$= 25 + 10$
$= 35\%$
$= 0.35$ Ans

ii) Sign extension is need for addi, beq, lw, sw

$= addi + beq + lw + sw$
$= 20\% + 25\% + 25\% + 10\%$
$= 80\%$
$= \cancel{0.08}\ 0.8$ Ans

The Sign extension block is not connected to logic. Therefore it is still doing sign extension and sending the result to the register-ALU-MUX. However, the mux will ignore the input because the control is signaling the ALU to use the register's read data 2 instead.

Q5) What is control signal design. Explain different control signal and their job.

Soln Once the datapath is ready, the flow of data and the operation and the data is controlled by a set of signals and these signal creation must be design and design of the signal to control the datapath is called control signal design. The opcode field of instruction and the function field and shift amount from R-type of instruction are used to design the control signal.

The different control signal used in the datapath are given below in the table -

Signal name	Effect when deasserted	Effect when asserted
i) RegDst	→ The register destination no. for the write register comes from the rt field.	→ The register destination no. for the write register comes from the rd field.
ii) Reg write	→ None	→ The register on the write register i/p is written with the value on the write data i/p.
iii) ALUSrc	→ The second ALU operand comes from the second register file output.	→ The second ALU operand is the sign-extended, lower 16 bits of the instruction.
iv) PCSrc	→ The PC is replaced by the output of the adder that computes the value of PC+4.	→ The PC is replaced by the output of the adder that computes the branch target.

Signal name	Effect when deasserted	Effect when asserted
v) MemRead	None	Data memory contents designated by the address i/p are put on the read data o/p.
vi) MemWrite	None	Data memory contents designated by the address i/p are replaced by the value on the write data input (i/p).
vii) MemtoReg	The value fed to the register write data i/p comes from the ALU.	The value fed to the register write data i/p comes from the data memory.