CSE 141 – HOMEWORK 4 (10 PTS): MULTI-CYCLE CPU & PIPELINING

1. **(1 pt)** Consider a processor with 5-stage pipeline (1 cycle / stage). Suppose we have a program with 10 instructions. How many cycles are required to finish the program:

   (a) No pipelining

   50

   (b) Pipelining

   14

2. **(3 pt)** The 5 stages of the processor have the following latencies:

	Fetch	Decode	Execute	Memory	Writeback
A.	300ps	400ps	350ps	500ps	100ps
B.	200ps	150ps	120ps	190ps	140ps

   Assume that when pipelining, each pipeline stage costs 20ps extra for the registers between pipeline stages.

   (a) Non-pipelined processor: what is the cycle time? What is the latency of an instruction? What is the throughput?

   A. 1650ps

   B. 800ps

   THROUGHPUT: 1/(cycle time) instructions per second

   (b) Pipelined processor: What is the cycle time? What is the latency of an instruction? What is the throughput?

   A. 500 to ~530ps

   B. 200 to ~230ps

   THROUGHPUT: 1 instruction per cycle

   (c) If you could split one of the pipeline stages into 2 equal halves, which one would you choose? What is the new cycle time? What is the new latency? What is the new throughput?

   A. 420ps

   B. 210ps

   LATENCY: cycle time * 6, as there is now a new pipeline stage

   THROUGHPUT: 1 instruction per cycle

3. **(3 pt)** Given these timings for individual stages of the datapath:

   IF: 100ps
   ID: 60ps
   EX: 170ps

MEM: 250ps
WB: 140ps

Assume the processor is designed to execute all the usual instructions unless stated otherwise.

(a) What is the clock cycle time of a single cycle processor?

720ps

(b) How long does it take for an add instruction to execute on a single cycle processor?

720ps

(c) What is the clock cycle time of the multi-cycle processor?

250ps

(d) How long does it take for an add instruction to execute on the multi cycle processor?

250ps

(e) How long does it take for a ld instruction to execute on the single cycle processor, if the MEM stage now takes 50ps less and WB stage takes 70ps more? How long does it take for the case of a multi cycle processor?

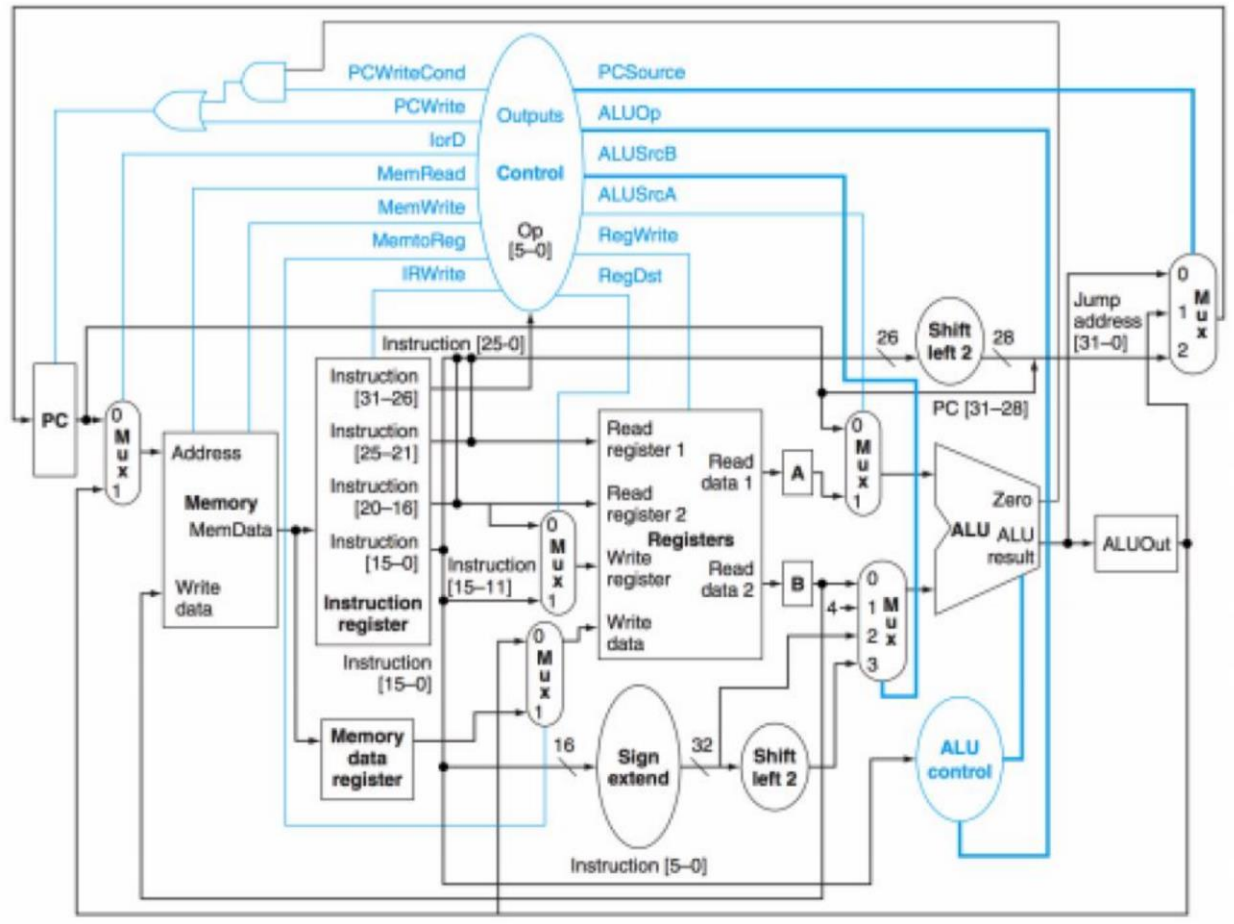
New MEM: 200. New WB: 210

- **Single-cycle processor = 740ps**
- **Multi-cycle processor = 210ps**

(f) Suppose we remove support for all loads and stores from the design with the original latencies. What would the new clock cycle time be for the single-cycle processor? What would it be for the multi-cycle processor?

- **Single-cycle processor = 520ps**
- **Multi-cycle processor = 170ps**

4. **(5 pt)** Consider the multi-cycle processor shown in the following figure. For each instruction and the stage of execution provided, show which control signals are used and what are their corresponding values. [Note: For the control signals you do not list for each answer, we will assume them to be 0 if they are critical signals for the given stage; and the critical ones will be taken as don't-cares.]



(a) beq $r1, $r3, loop - IF stage

MemRead

ALUOp, 0

ALUSrcB, 1

PCWrite

IRWrite

PCSource

(b) sw $t1, 4($t0) - ID stage

MemRead

ALUOp, 0

ALUSrcB, 1

PCWrite

IRWrite

PCSource

(c) j target - Completion stage

MemRead

ALUOp, 0

ALUSrcB, 1

PCWrite

IRWrite

PCSource

(d) lw $t0, 4096($r3) - MEM stage

MemRead

ALUOp, 0

ALUSrcB, 1

PCWrite

IRWrite

PCSource

(e) slt $r1, $r2, $r3 - WB stage

MemRead

ALUOp, 0

ALUSrcB, 1

PCWrite

IRWrite

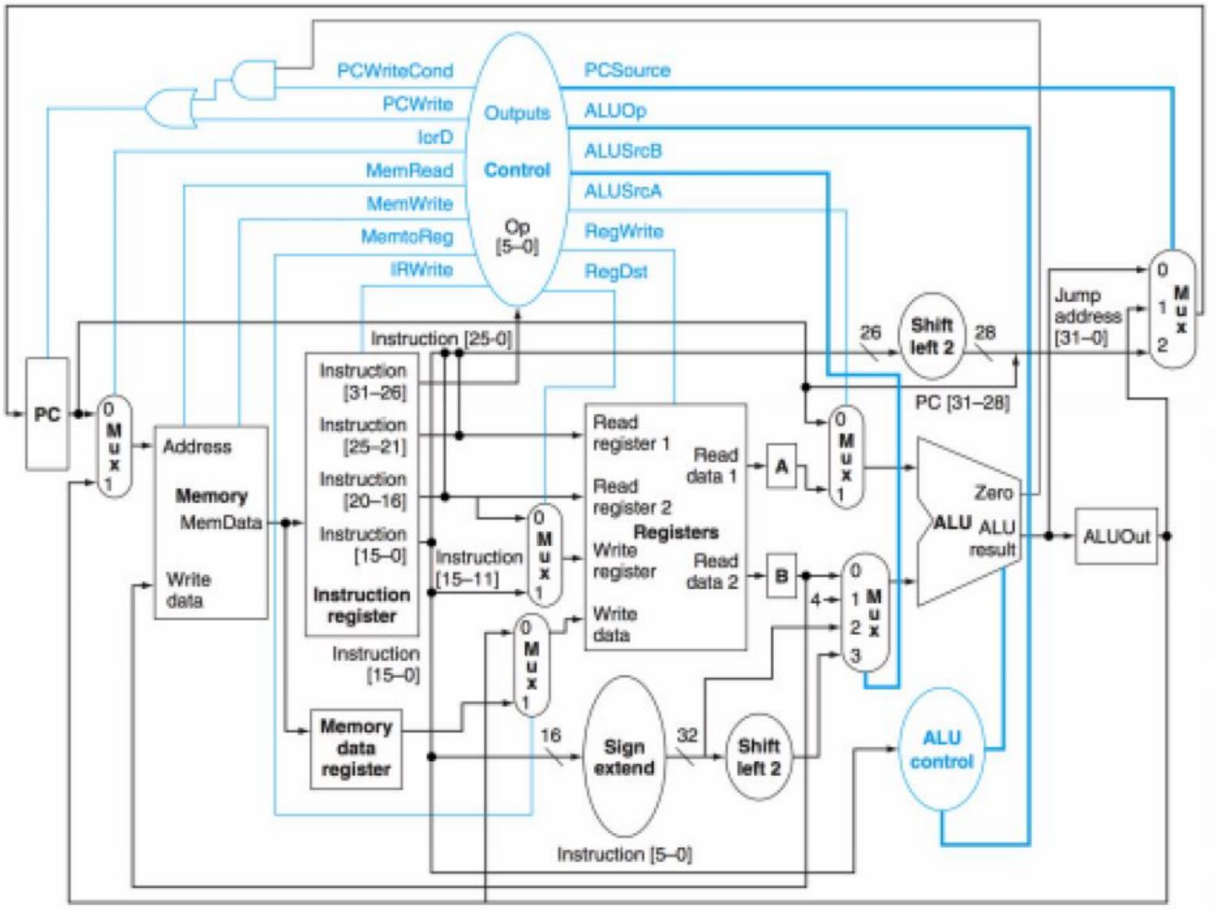
PCSource

5. **(5 pt)** Suppose we have a new multi-cycle processor. A new instruction in this processor accesses the stages of the multi-cycle processor in the following order [IF ID EX MEM EX MEM]. Describe one possible new instruction that would execute the stages in this order. You may assume these stages are different than those in our MIPS multi-cycle processor. Provide the Register Transfer Language (RTL) for the new instruction.

   MOD, e.g. mod r1 <- r2, r3. We perform modulus operation r2 with r3 and store the result in r1.

6. **(2 pt)** Suppose we wish to add support for a new instruction beqm, beqm rt, rs, target; does If R[$rt] == Mem[R[$rs]], then PC = BTA (standard BTA) What additional blocks do we need to add to our existing design to support the above instruction?

   Please list the changes made to the data-path.



7. **(Bonus - 2 pt)** Assume the classic 5-stage pipeline, with registers being written in the first half of WB and read in the last half of ID. There are two memory ports, one for instructions and one for data. Consider the following code:

```
LW    R9, 0(R5)
ADD   R1, R9, R8
SUB   R2, R1, R9
ADD   R3, R1, R2
```

   (a) How many cycles does the code take, from first fetch to last writeback, assuming no forwarding?
   14

   (b) How many cycles would the code take, from first fetch to last writeback, if we allow forwarding?
   9

8. **(Bonus - 2 pt)** Use the following code fragment:

Loop: LW R1, 0(R2)

ADD R1, R1, R7

SW 0(R2), R1

ADD R2, R2, R7

SUB R4, R3, R2

BNEZ R4, Loop

Show the timing of this instruction sequence for a 5-stage pipeline along with the number of cycles required to execute one iteration of the loop with no forwarding.

Number of cycles = …

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
LW																	
ADD																	
SW																	
ADD																	
SUB																	
BNEZ																	

9. **(Bonus - 2 pt)** Use the following code fragment:

Loop: LD R1, 0(R2)

ADD R1, R1, R7

SW 0(R2), R1

ADD R2, R2, R7

SUB R4, R3, R2

BNEZ R4, Loop

Show the timing of this instruction sequence for a 5-stage pipeline along with the number of cycles required to execute one iteration of the loop with forwarding. Assume registers can be written and read in the same cycle, during writeback. (The number of cycles for the execution of one iteration of the loop ends after the EX stage of BNEZ instruction)

Number of cycles = …

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
LW																	
ADD																	
SW																	
ADD																	
SUB																	
BNEZ																	