

LAB 2 EXERCISE

a.) Show the timing of this instruction sequence for the RISC pipeline without any forwarding or bypassing hardware but assuming a register read and a write in the same clock cycle “forwards” through the register file. Assume that the branch is handled by flushing the pipeline. If all memory references take 1 cycle, how many cycles does this loop take to execute?

Assumptions:

- No forwarding/bypassing.
- Register read/write in same cycle forwards through register file.
- Branch resolved in ID, flush IF and ID on taken branch (2-cycle penalty).
- Stalls due to data hazards.

Pipeline Table (first iteration, subsequent similar):

Inst	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
LD	ID	ID	EX	ME	WB															
DADD I		IF	ID			ID	EX	ME	WB											
SD				IF			ID	EX	ME	WB										
DADDI(R2)					IF			ID	EX	ME	WB									
DSUB									ID	EX	ME	WB	WB							

BNEZ							IF			ID	E X	M E								
LD(next)													IF	I D	E X	M E	W E			

- Stalls:

- `DADDI` : 2 stalls (cycles 4–5) waiting for R1 from `LD` (WB cycle 5).
- `SD` : 1 stall (cycle 6) waiting for R1 from `DADDI` (WB cycle 8).
- `DADDI(R2)` : 1 stall (cycle 9) due to prior stalls.
- `DSUB` : 2 stalls (cycles 11–12) due to prior stalls.
- `BNEZ` : 1 stall (cycle 14) waiting for R4 from `DSUB` (WB cycle 15).
- `LD` (next): 3 stalls (cycles 14–16) waiting for R2 from `DADDI(R2)` (WB cycle 12).
- Branch taken: 2-cycle flush (cycles 16–17).
- Cycles per iteration: 17 cycles (6 instructions + 9 stalls + 2 flush).
- Total cycles: $(99 \times 17 + 15)$ (final iteration, no flush) = 1698 cycles.

b.) Forwarding, Predict Branch Not Taken

Assumptions:

- Normal forwarding/bypassing.
- Branch predicted not taken, flush IF/ID if taken (2-cycle penalty).

Pipeline Table:

Inst	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
LD	I F	I D	E X	M E	W B															
DADDI		I F	I D	E X	M E	W B														
SD			I F	I D	E X	M E	W B													
DADDI(R2)				I F	I D	E X	M E	W B												
DSUB					I F	I D	E X	M E	W B											
BNEZ						I F	I D	E X	M E	W B										
LD (next)								I F			I D	E X	M E	W B						

- Stalls:

- `LD` (next) stalls 2 cycles (cycles 9–10) waiting for R2 from `DADDI(R2)` (EX cycle 8).
- Branch taken: 2-cycle flush.
- Cycles per iteration: 10 cycles (6 instructions + 2 stalls + 2 flush).
- Total cycles: $(99 \times 10 + 8)$ (final iteration, no flush) = 998 cycles.

c.) Assume the RISC pipeline with a single-cycle delayed branch and normal forwarding and bypassing hardware. Schedule the instructions in the loop including the branch delay slot. You may reorder instructions and modify the individual instructions operands, but do not undertake other loop transformations that change the number or opcode of the instructions in the loops. Show a pipeline timing diagram and compute the number of cycles needed to execute the entire loop

Assumptions:

- Single-cycle delayed branch (execute delay slot).
- Normal forwarding.
- Reorder instructions, modify operands, same opcodes/number.

Reordered Loop:

loop:

LD R1, 0(R2)

DADDI R1, R1, 1

SD 4(R2), R1

DADDI R2, R2, 4

DSUB R4, R3, R2

BNEZ R4, loop

LD R1, 4(R2) ; delay slot

Pipeline table

Inst	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
LD	I F	I D	E X	M E	W B															
DADDI		I F	I D	E X	M E	W B														
SD			I F	I D	E X	M E	W B													
DADDI(R2)				I F	I D	E X	M E	W B												
DSUB					I F	I D	E X	M E	W B											
BNEZ						I F	I D	E X	M E	W B										
LD (delay)							I F	I D	E X	M E	W B									

- Stalls: None, as forwarding and scheduling resolve hazards.

- Cycles per iteration: 7 cycles (6 instructions + 1 delay slot).

- Total cycles: $(100 \times 7 = 700)$ cycles.