JOINT INSTITUTE
交大密西根学院

Ve370 Introduction to Computer Organization

Homework 4

1. Exercise 4.12.1
2. Exercise 4.12.2
3. Exercise 4.12.3

Exercise 4.12

In this exercise, we examine how pipelining affects the clock cycle time of the processor. Problems in this exercise assume that individual stages of the datapath have the following latencies:

	IF	ID	EX	MEM	WB
a.	250ps	350ps	150ps	300ps	200ps
b.	200ps	170ps	220ps	210ps	150ps

4.12.1 [5] <4.5> What is the clock cycle time in a pipelined and non-pipelined processor?

4.12.2 [10] <4.5> What is the total latency of an `LW` instruction in a pipelined and non-pipelined processor?

4.12.3 [10] <4.5> If we can split one stage of the pipelined datapath into two new stages, each with half the latency of the original stage, which stage would you split and what is the new clock cycle time of the processor?

4. Exercise 4.13.1
5. Exercise 4.13.2
6. Exercise 4.13.3
7. Exercise 4.13.4
8. Exercise 4.13.5
9. Exercise 4.13.6

Exercise 4.13

In this exercise, we examine how data dependences affect execution in the basic 5-stage pipeline described in Section 4.5. Problems in this exercise refer to the following sequence of instructions:

	Instruction Sequence

a.	`SW R16,-100(R6)`
	`LW R4,8(R16)`
	`ADD R5,R4,R4`

b.	`OR R1,R2,R3`
	`OR R2,R1,R4`
	`OR R1,R1,R2`

4.13.1 [10] <4.5> Indicate dependences and their type.

4.13.2 [10] <4.5> Assume there is no forwarding in this pipelined processor. Indicate hazards and add NOP instructions to eliminate them.

4.13.3 [10] <4.5> Assume there is full forwarding. Indicate hazards and add NOP instructions to eliminate them.

	Without Forwarding	With Full Forwarding	With ALU-ALU Forwarding Only
a.	250ps	300ps	290ps
b.	180ps	240ps	210ps

4.13.4 [10] <4.5> What is the total execution time of this instruction sequence without forwarding and with full forwarding? What is the speedup achieved by adding full forwarding to a pipeline that had no forwarding?

4.13.5 [10] <4.5> Add NOP instructions to this code to eliminate hazards if there is ALU-ALU forwarding only (no forwarding from the MEM to the EX stage).

4.13.6 [10] <4.5> What is the total execution time of this instruction sequence with only ALU-ALU forwarding? What is the speedup over a no-forwarding pipeline?

10. Exercise 4.15.1
11. Exercise 4.15.2
12. Exercise 4.15.3

Exercise 4.15

In this exercise, we examine how the ISA affects pipeline design. Problems in this exercise refer to the following new instruction:

a.	ADDM Rd,Rt+Offs(Rs)	Rd=Rt+Mem[Offs+Rs]
b.	BEQM Rd,Rt,Offs(Rs)	If Rt=Mem[Offs+Rs] then PC = Rd

4.15.1 [20] <4.5> What must be changed in the pipelined datapath to add this instruction to the MIPS ISA?

4.15.2 [10] <4.5> Which new control signals must be added to your pipeline from 4.15.1?

4.15.3 [20] <4.5, 4.13> Does support for this instruction introduce any new hazards? Are stalls due to existing hazards made worse?

13. Exercise 4.21.1
14. Exercise 4.21.2
15. Exercise 4.21.3
16. Exercise 4.21.4
17. Exercise 4.21.5
18. Exercise 4.21.6

Exercise 4.21

This exercise is intended to help you understand the relationship between forwarding, hazard detection, and ISA design. Problems in this exercise refer to the following sequences of instructions, and assume that it is executed on a 5-stage pipelined datapath:

	Instruction sequence

a.	ADD R5,R2,R1
	LW R3,4(R5)
	LW R2,0(R2)
	OR R3,R5,R3
	SW R3,0(R5)

b.	LW R2,0(R1)
	AND R1,R2,R1
	LW R3,0(R2)
	LW R1,0(R1)
	SW R1,0(R2)

4.21.1 [5] <4.7> If there is no forwarding or hazard detection, insert NOPs to ensure correct execution.

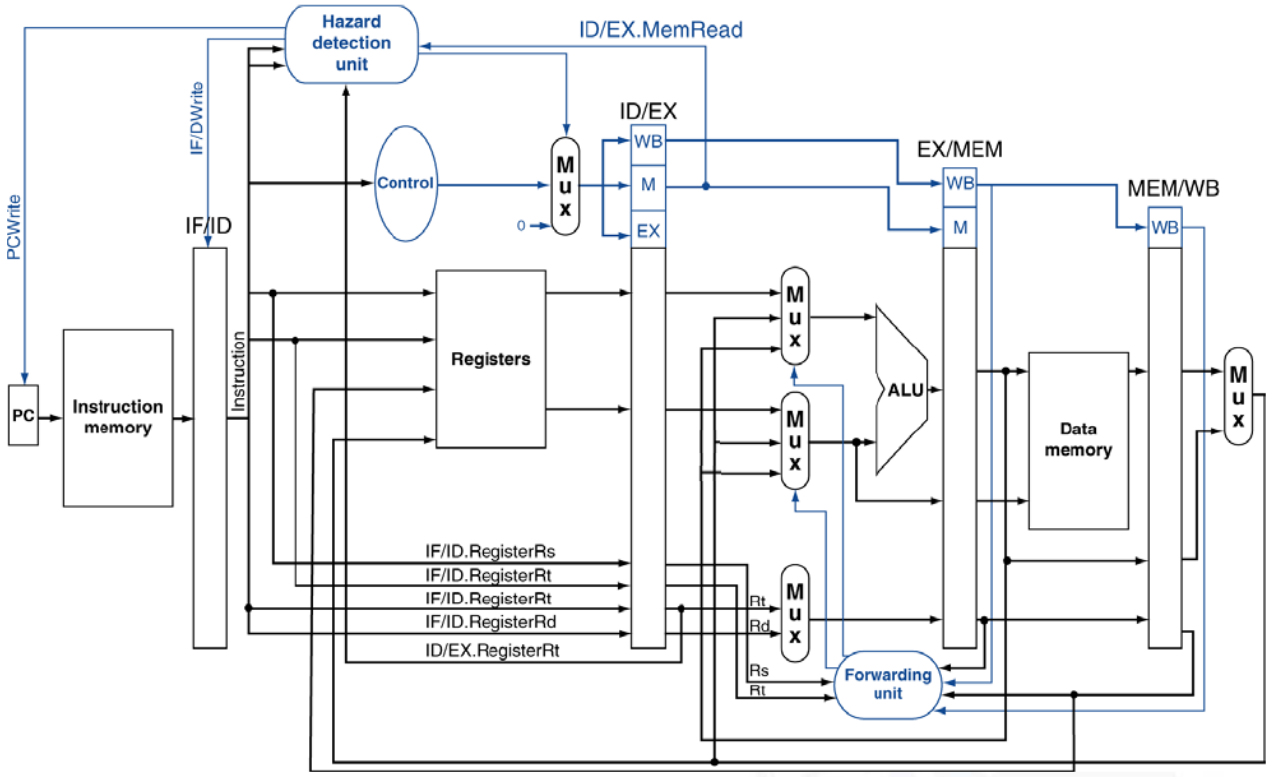
4.21.2 [10] <4.7> Repeat 4.21.1 but now use NOPs only when a hazard cannot be avoided by changing or rearranging these instructions. You can assume register R7 can be used to hold temporary values in your modified code.

4.21.3 [10] <4.7> If the processor has forwarding, but we forgot to implement the hazard detection unit, what happens when this code executes?

4.21.4 [20] <4.7> If there is forwarding, for the first five cycles during the execution of this code, specify which signals are asserted in each cycle by hazard detection and forwarding units in Figure 4.60.

4.21.5 [10] <4.7> If there is no forwarding, what new inputs and output signals do we need for the hazard detection unit in Figure 4.60? Using this instruction sequence as an example, explain why each signal is needed.

4.21.6 [20] <4.7> For the new hazard detection unit from 4.21.5, specify which output signals it asserts in each of the first five cycles during the execution of this code.



(questions above refer to the following diagram)