

Lab Report 6

Computer Architecture Lab

201651015 Dipansh Khandelwal

201651008 Aman Singh

11th March, 2019

Aim

Record for each instruction in which clock-cycle it is in each of the traversed pipeline stages.

Questions and Answers

Problem - 1

ld F6, 34(R2)

ld F2, 45(R3)

multd F0, F2, F4

subd F8, F6, F2

divd F10, F0, F6

addd F6, F8, F2

Welcome to the Scoreboarding Demo - Mozilla Firefox

IIIT-V Login Page x Welcome to Scoreboarding x Demo x Welcome to the Scoreboard x +

dark.eit.lth.se/darklab/scoreboard/demo/index1.htm

SCOREBOARDING TECHNIQUE FOR DYNAMIC SCHEDULING

Number of Instructions: 6

Enter Instructions Below:

LD	F6	34	R2
LD	F2	45	R3
MULTD	F0	F2	F4
SUBD	F8	F6	F2
DIVD	F10	F0	F6
ADDD	F6	F8	F2
None			
None			
None			
None			

Enter the number of execution cycles taken by the functional units:

Integer: 1 FP Add: 2

Clock Cycle No. 62

Instruction Status

	Issue	Read Op	Exec	Write Result
LD F6 34 R2	1	2	3	4
LD F2 45 R3	5	6	7	8
MULTD F0 F2 F4	6	9	19	20
SUBD F8 F6 F2	7	9	11	12
DIVD F10 F0 F6	8	21	61	62
ADDD F6 F8 F2	13	14	16	22

Functional unit status

Func. Unit Name	Busy	Op	dest	S1	S2	FU	FU	Fj?	Fk?
			Fi	Fj	Fk	Qj	Qk	Rj	Rk
Integer	No								
mult1	No								
mult2	No								
add	No								
divide	No								

Destination result status



Answer the following questions:-

1. After how many clock cycles can this program branch back to the beginning?

62

2. Does re-ordering influence the execution time of this program and how?

Yes, we can save two clock cycles by swapping the first two instructions. Previously, the third instruction completed at 20th clock cycle but if we swap it, the same instruction would be completed on the 18th clock cycle.

3. Is there a Write-after-Read hazard present and how is it solved?

Write-after-Read hazard is present between the last two instructions. Write-after-read is a hazard but it is causing no stalls, so there is no need to solve it.

Problem - 2

ld F0, 0(R1)

addd F4, F0, F2

sd F4, 0(R1)

ld F0, -8(R1)

addd F4, F0, F2

sd F4, -8(R1)

Welcome to the Scoreboarding Demo - Mozilla Firefox

IIIT-V Login Page x Welcome to Scoreboarding x Demo x Welcome to the Scoreboard x

dark.eit.lth.se/darklab/scoreboard/demo/index1.htm

SCOREBOARDING TECHNIQUE FOR DYNAMIC SCHEDULING

LD F0 0 R1

ADDD F4 F0 F2

SD F4 0 R1

LD F0 -8 R1

ADDD F4 F0 F2

SD F4 -8 R1

None

None

None

None

Enter the number of execution cycles taken by the functional units:

Integer: 1 FP Add: 2

FP Multiply: 10 FP Divide: 40

Clock Cycle No. 24

Instruction Status

	Issue	Read Op	Exec	Write Result
LD F0 0 R1	1	2	3	4
ADDD F4 F0 F2	2	5	7	8
SD F4 0 R1	9	10	11	12
LD F0 -8 R1	13	14	15	16
ADDD F4 F0 F2	14	17	19	20
SD F4 -8 R1	21	22	23	24

Functional unit status

			dest	S1	S2	FU	FU	Fj?	Fk?
Func. Unit Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
integer	No								
mult1	No								
mult2	No								
add	No								
divide	No								

Register result status

Answer the following questions:-

1. After how many clock cycles can this program branch back to the beginning?

24

2. Does re-ordering influence the execution time of this program and how?

Yes, we can save two clock cycles by swapping the third and fourth instructions. Previously, the third instruction completed at 12th clock cycle but if we swap it, the same instruction would be completed on the 10th clock cycle.

3. Is there a Write-after-Read hazard present and how is it solved?

No, there isn't any Write-after-Read Hazard.