Dipartimento di Elettronica e Informazione

Politecnico di Milano

20133 Milano (Italia)
Piazza Leonardo da Vinci, 32
Tel. (+39) 02-2399.3400
Fax (+39) 02-2399.3411

Advanced Computer Architecture

May 10, 2023

Prof. D. Conficconi, and M. D. Santambrogio

Name
Last Name
Professor:

The exam will last 75’

Problem 1 (10%)	
Problem 2 (30%)	
Problem 3 (20%)	
Problem 4 (30%)	
Problem 5 (10%)	
Total (100%)	

Problem 1

A VLIW Architecture has to have multiple Program Counters to load the necessary Multiple Data.
Given the previous statement, confirm if it is TRUE or FALSE and **effectively support** your answer.

Circle the **right** answer: True False

Question 1.5

Increasing the number of stages in a pipeline, it is always improving the performance.
Given the previous statement, confirm if it is TRUE or FALSE and **effectively support** your answer.

Circle the **right** answer: True False

Problem 2

Assume that the following code has been executed on a CPU with SCOREBOARD.

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	**LD F5, 16 (R1)**	1	2	6	7
I2	**ADDD F12, F5, F2**	2	8	14	15
I3	**MULTD F2, F4, F3**	3	4	15	16
I4	**DIVD F1, F12, F5**	4	16	27	28
I5	**SD F1, 4 (R1)**	5	29	33	34
I6	**SUBD F2, F12, F4**	17	18	24	25

A. List all the possible conflicts in the code.

B. Is there a "configuration" that can respect the shown execution?
How many units? Which kind? What latency?

C. If the previous table was not correct, please, write the right one and specify the number, kind and latency for each unit.

Answer 2.B

A possible configuration consists of:

- 2 MUL/DIV units and 11 CC of latency
- 1 ADDD/SUBD unit and 6 CC of latency
- 2 Memory Unit and 4 CC of latency
- 1 write port is enough

Other solutions are possible.

Answer 2.C

Considering the previous answers, no answer is needed here.

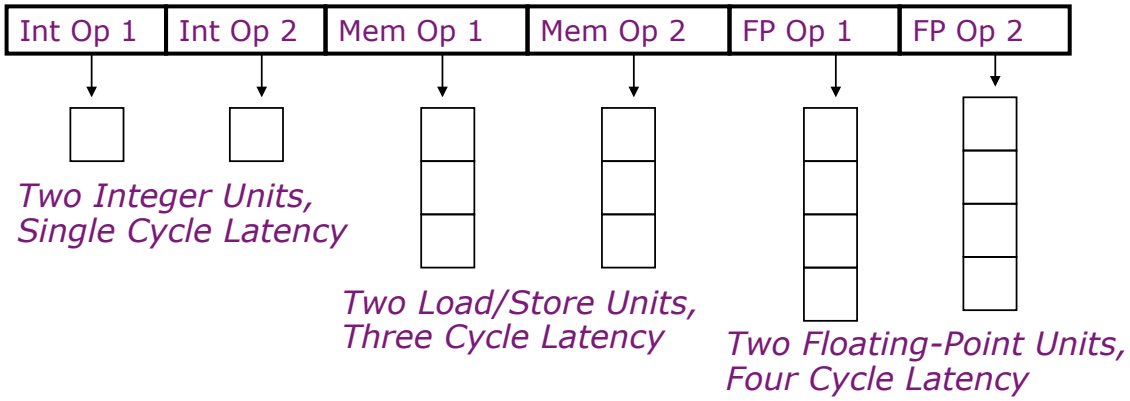
	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD F5, 16 (R1)				
I2	ADDD F12, F5, F2				
I3	MULTD F2, F4, F3				
I4	DIVD F1, F12, F5				
I5	SD F1, 4 (R1)				
I6	SUBD F2, F12, F4				

Problem 3

Explain the key idea of Dynamic Scheduling. Describe two dynamic-scheduling-based architectures and explain their main differences.

Problem 4

Considering the following VLIW "architecture":



Considering the following portion of assembly, describe the corresponding VLIW code:

Considering the following portion of assembly.

```
LOOP:   beq $t6,$t7, END
        lw $t2,VECTB($t6)
        lw $t3,VECTC($t6)
        sw $t2,VECTA($t6)
        addi $t3,$t3,4
        sw $0,VECTD($t6)
        sw $t3,VECTC($t6)
        addi $t6,$t6,4
        blt $t6,$t7, LOOP
```

4.A schedule the following code for the VLIW with an IN-ORDER ISSUE. Branch completed with 1 cycle delay slot (branch solved in ID stage).

4.B How the does the scheduling change if we consider an IN-ORDER ISSUE and pipelined FU?

Answer 4.A

	INT1	INT2	MU1	MU2	FPU1	FPU2
C1						
C2						
C3						
C4						
C5						
C6						
C7						
C8						
C9						
C10						
C11						
C12						
C13						
C14						
C15						

Answer 4.B

	FU1	FU2	FU3	FU4	FU5	Notes
C1						
C2						
C3						
C4						
C5						
C6						
C7						
C8						
C9						
C10						
C11						
C12						
C13						
C14						
C15						

Problem 1

A VLIW Architecture has to have multiple Program Counters to load the necessary Multiple Data.
Given the previous statement, confirm if it is TRUE or FALSE and **effectively support** your answer.

Circle the **right** answer: True ✓ False

VLIW → SINGLE INSTRUCTIONS
⇒ ONLY ONE P.C.

Question 1.5

Increasing the number of stages in a pipeline, it is always improving the performance.
Given the previous statement, confirm if it is TRUE or FALSE and **effectively support** your answer.

Circle the **right** answer: True ✓ False

POTENTIALLY HIGHER OVERLAPPINGS
AND STAGES WHERE TO STALL

2

LD F5, 16 (R1)
ADDD F12, F5, F2
MULTD F2, F4, F3
DIVD F1, F12, F5
SD F1, 4 (R1)
SUBD F2, F12, F4

RAW F5 I1-I2 WAR F2 I2-I3

RAW F5 I1-I4 WAR F2 I2-I6

RAW F12 I2-I4

RAW F12 I2-I6 WAW F2 I3-I6

RAW F1 I4-I5

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD F5, 16 (R1)	1	2	6	7
I2	ADDD F12, F5, F2	2	8	14	15
I3	MULTD F2, F4, F3	3	4	15	16
I4	DIVD F1, F12, F5	4	16	27	28
I5	SD F1, 4 (R1)	5	29	33	34
I6	SUBD F2, F12, F4	17	18	24	25

✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓

2 MEM 4 cc

1 ADDD/SUBD 6 cc

1 MULD 11 cc

1 DIVD 11cc

3

DYNAMIC SCHEDULING TRIES TO SIMPLIFY INSTRUCTION EXECUTIONS BY DYNAMICALLY ADAPTING TO THE GIVEN CONTEXT. INTRODUCES OUT-OF-ORDER EXECUTIONS

SCOREBOARD

4-STAGE PIPELINE ARCHITECTURE

- DISTINGUISH ISSUE STAGE AND READ OPERAND STAGE
- RELIES ON A CENTRALIZED CONTROLLER

TOMASULO

3-STAGE PIPELINE ARCHITECTURE

- RELIES ON A DISTRIBUTED CONTROLLER THAT ALLOWS TO "AVOID" WAR BUT DEPENDS ON THE AVAILABLE RESERVATION STATIONS RS, THAT EXPLOITS AN "IMPLICIT REGISTER RENAMING"
- CENTRALIZED COMMIT PHASE HANDLED BY A COMMON DATA BUS CDB

4

A

	INT1	INT2	MU1	MU2	FPU1	FPU2
C1	beq $t6, $t7, END	1	lw $t2, VECTB($t6)	lw $t3, VECTC($t6)		
C2		2				
C3		3				
C4	addi $t3, $t3, 4	1	sw $t2, VECTA($t6)	sw $0, VECTD($t6)		
C5		2				
C6		3				
C7	addi $t6, $t6, 4 1		sw $t3, VECTC($t6)			
C8	1 DELAY SLOT					
C9	blt $t6, $t7, LOOP					
C10						
C11						
C12						
C13						
C14						
C15						

B

PIPELINED FU => NO NEEDS TO WAIT THE PRECEDING INSTRUCTION TO END UNLESS THERE ARE DEPENDENCIES

	FU1	FU2	FU3	FU4	FU5	Notes
C1	beq $t6, $t7, END	1	lw $t2, VECTB($t6)	lw $t3, VECTC($t6)		
C2		2				
C3		3				
C4	addi $t3, $t3, 4	1	sw $t2, VECTA($t6)	sw $0, VECTD($t6)		
C5	addi $t6, $t6, 4	1	sw $t3, VECTC($t6)			
C6	DELAY					
C7	bit $t6, $t7, LOOP					
C8						
C9						
C10						
C11						
C12						
C13						
C14						
C15						