



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

June 21, 2021

Prof. Marco Santambrogio

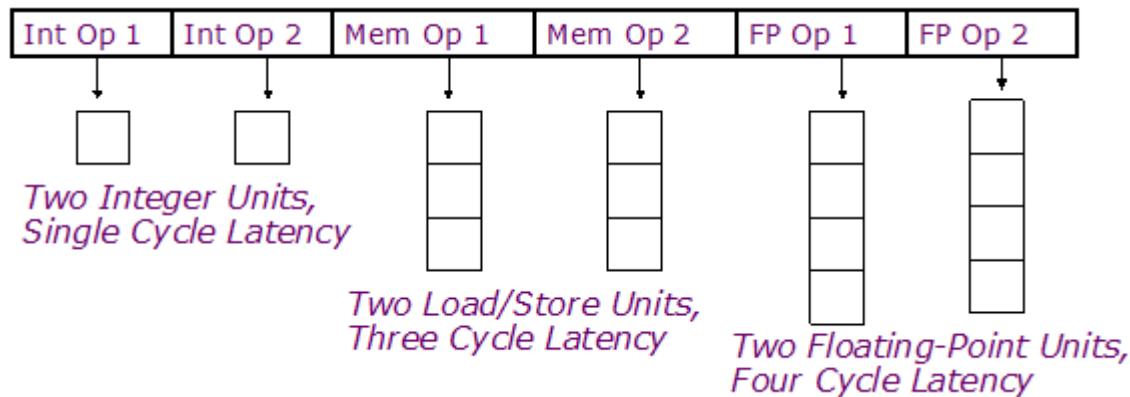
Name
Last Name
Code Name (Codice Persona)

WRITE YOUR NAME and **CODE NAME**.
Pick 2 out of th 3 problems, **DO NOT DO ALL THE 3!**

Problem 1 (40%)	
Problem 2 (40%)	
Problem 3 (40%)	
Total (100%)	

Problem 1

Consider the following VLIW “architecture” with multi-cycled unpipelined functional units.



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
```

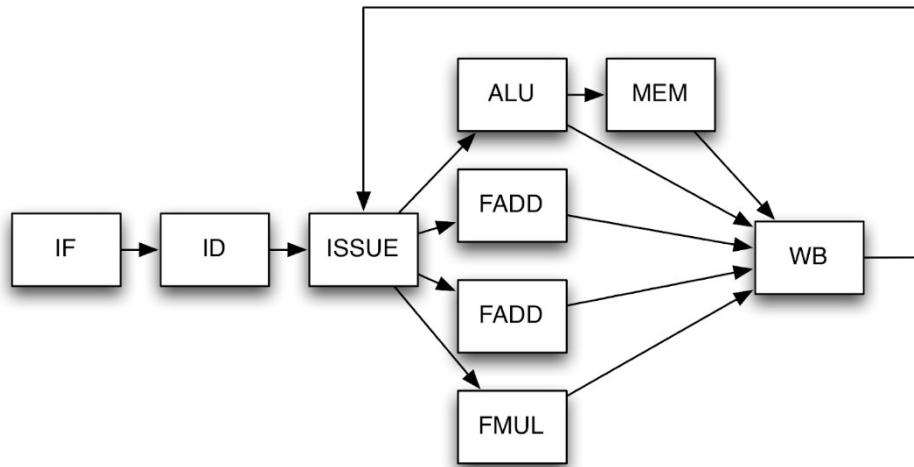
1.A schedule the following code for the VLIW with an IN-ORDER ISSUE.
Branch completed with 1 cycle delay slot (branch solved in ID stage).

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

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

Problem 2

In this problem we will examine the execution of a code segment on the following single-issue out-of-order processor:



You can assume that:

- All functional units are pipelined
- ALU operations take 1 cycle
- MEM operations take 3 cycles (includes time in ALU)
- FP add instructions take 3 cycles
- FP multiply instructions take 5 cycles
- The WB unit has a single write port.
- RF can write first half of the CC and Read in the second half.
- There is no register renaming
- Instructions are fetched, decoded and issued in order
- The issue stage is a buffer of unlimited length that holds instructions waiting to start execution
- An instruction will only enter the issue stage if it does not cause a WAR or WAW hazard.
- Only one instruction can be issued at a time, and in the case multiple instructions are ready, the oldest one will go first

LD	F1, 0(R1)
LD	F2, 0(R1)
ADDD	F1, F1, F2
ADDI	R1, R1, 8
MULD	F1, F1, F3
SD	F1, 16(R1)

1.A List all the possible dependencies/conflicts in the code

1.B Solve considering: WAW and WAR are solved in the FIRST HALF of the WB.

1.C Consider now that the WARs are solved at the END of the ISSUE.

Problem 3

Assume that the following code is executed on a CPU with Tomasulo

Instruction	ISSUE	START EXE	WB
I1: LD F6 32+ R2	1	2	5
I2: LD F2 45+ R3	2	6	9
I3: MULTD F0 F4 F3	3	4	15
I4: ADD F8 F2 F6	4	10	16
I5: DIVD F12 F8 F0	5	17	28
I6: SUBD F8 F6 F2	17	18	23

- A. List all the possible dependencies/conflicts in the code.
- B. Is there a “configuration” that can respect the shown execution?
How many units? Which kind? What latency? How many Reservation Stations?
- C. If the previous table was not correct, please, write the right one.

Answer 3.A

RAW F6 I1-I4
RAW F6 I1-I6

RAW F2 I2-I4
RAW F2 I2-I6

RAW F0 I3-I5
RAW F8 I4-I5

WAR F8 I5-I6
WAW F8 I4-I6

1

A

	INT1	INT2	MU1	MU2	FPU1	FPU2
C1	beg \$t6,\$t7,END	1	lw \$t2, vecB(\$t6)	lw \$t3, vecC(\$t6)		
C2		2			2	
C3		3			3	
C4	add \$t3,\$t3,4	1	lw \$t2, vecA(\$t6)	lw \$t0, vecB(\$t6)		
C5		2				
C6		3				
C7	add \$t6,\$t6,4	1		lw \$t3, vecC(\$t6)		
C8	DELAY SLOT					
C9	bit \$t6,\$t7,LOOP	MINIMIZED ACTIVE CCB				
C10	NEXT INSTRUCTION					
C11						
C12						
C13						
C14						
C15						

1

B

	INT1	INT2	MU1	MU2	FPU1	FPU2
C1	beq \$t6,\$t7,END	1	lw \$t2, vecB(\$t6)	lw \$t3, vecC(\$t6)		
C2		2			2	
C3		3			3	
C4	add \$t3,\$t3,4	5	lw \$t2, vecA(\$t6)	lw \$t0, vecB(\$t6)		
C5	add \$t6,\$t6,4	1	lw \$t3, vecC(\$t6)			
C6	DELAY SLOT		PIPERINED FU			
C7	bit \$t6,\$t7,LOOP					
C8	NEXT INSTRUCTION					
C9						
C10						
C11						
C12						
C13						
C14						
C15						

2 ALU 1cc MEM 3cc FP+ 3cc FP. 5cc
IS UNLIMITED

LD	$F_1, 0(R_1)$	RAW	F1	H-13
LD	$F_2, 0(R_1)$	WAW	F1	I-13
ADDD	F_1, F_1, F_2			
ADDI	$R_1, R_1, 8$			
MULD	F_1, F_1, F_3	WAW	F1	I-15
SD	$F_1, 16(R_1)$			

RAW F1 15-16

RAKU F2 12-13

WAR RY 11-14

WAR P1 12-14

RAK R1 14-16

WAR F1 13-15

2B

2c

NOTHING CHANGES, AS STILL WAW ARE MANAGED IN THE
SAME WAY AS BEFORE

3

- I1: LD F6 32+ R2
- I2: LD F2 45+ R3
- I3: MULTD F0 F4 F3
- I4: ADD F8 F2 F6
- I5: DIVD F12 F8 F0
- I6: SUBD F8 F6 F2

✓ RAW F6 11-14 ✓ RAW F6 11-16
 ✓ RAW F2 12-14 ✓ RAW F2 12-16
 ✓ RAW F0 13-15 ✓ RAW F8 14-15
 WAR F8 15-16 WAW F8 14-16

B

Instruction	ISSUE	START EXE	WB
I1: LD F6 32+ R2	1	2	5
I2: LD F2 45+ R3	2	6	9
I3: MULTD F0 F4 F3	3	4	15
I4: ADD F8 F2 F6	4	10 ✓✓	16
I5: DIVD F12 F8 F0	5	17 ✓✓	28
I6: SUBD F8 F6 F2	17	18 ✓✓	23

1 LDUNIT

1 FPf
UNIT

! TOMASULO DO NOT WORRIES ABOUT WAR AND WAW

1 LDU RS1, RS2 3cc

1 MUL RS3, RS4 11cc

1 ADD RS5 5cc

EXTRA THEORY) IS GPU AN EXAMPLE OF A MULTICORE ARCHITECTURE?

No. A multicore architecture integrates multiple cores in the same CPU.

It can be classified as a MIMD architecture. Instead, GPU exploits

multiple processor to execute a certain instruction to different data.

They are closer to a SIMD architecture.