

# Computer Architecture - Unit 2

Final Exam, 05.Jul.2021, 10:30AM

Name: \_\_\_\_\_ Last name: \_\_\_\_\_

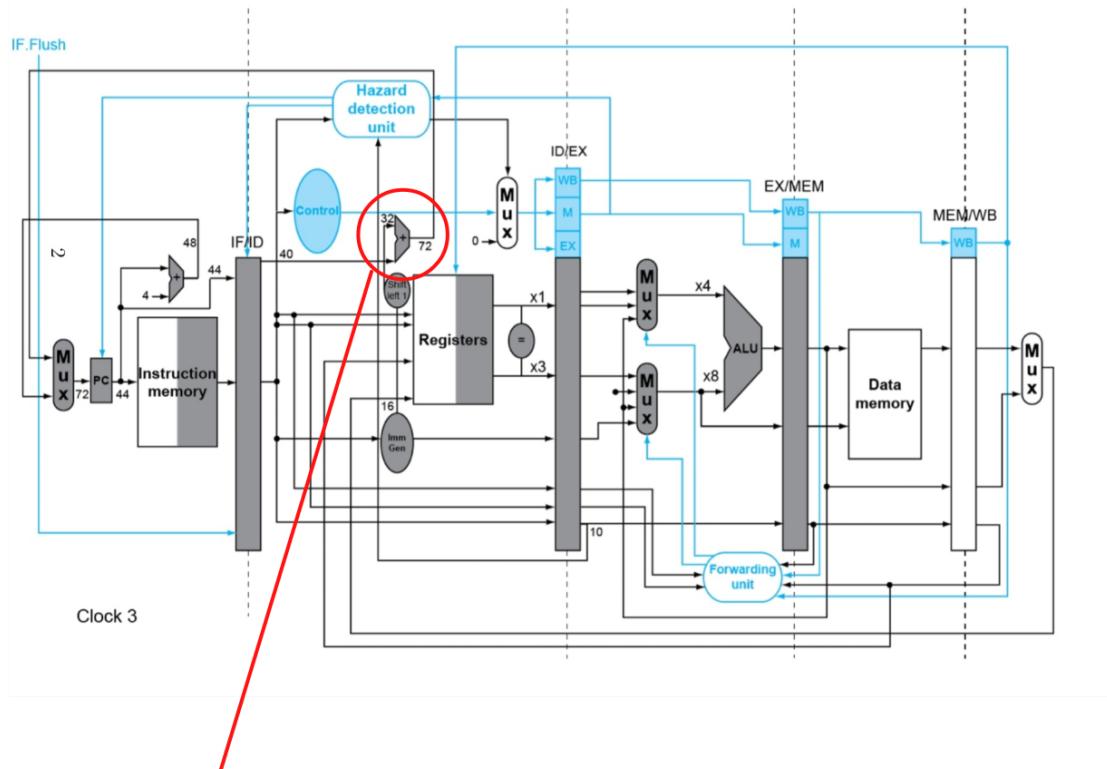
Infostud ID: \_\_\_\_\_

**Exercise 1.** How many clock cycles are needed to complete the following code on the Risc-V pipelined architecture? (Figure in the next page.) Remember that:

- Execution completes when the last instruction exits the pipeline.
- If you read and write the same register of the register file in the same clock cycle, the value that is read is the value that is being written.

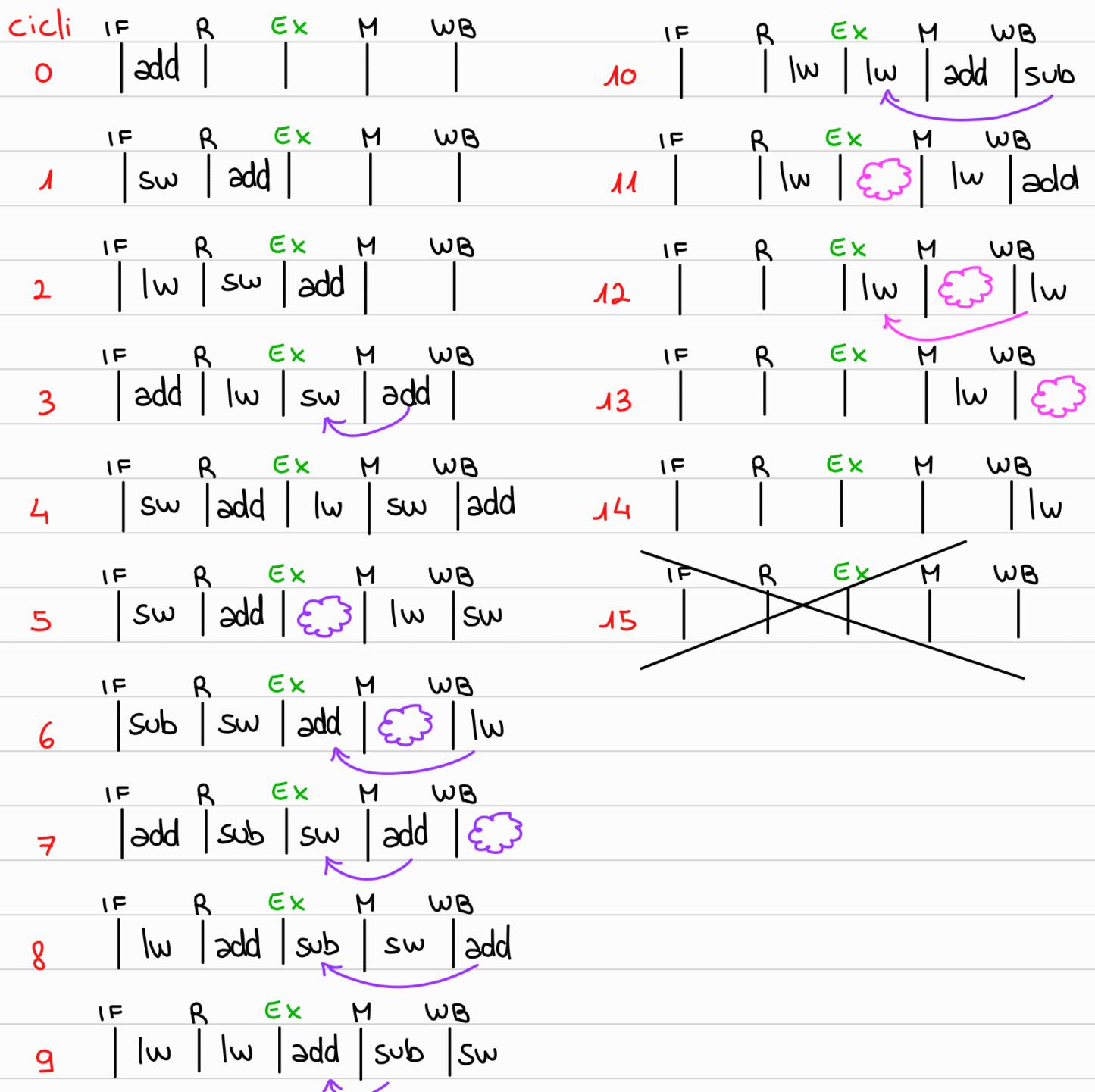
Show a figure of the pipeline with bubbles and forwardings.

```
add s1, s0, s0
sw s0, 4(s1)
lw s1, 8(s0)
add s1, s1, s0
sw s1, 12(s2)
sub s3, s1, s0
add s1, s3, s3
lw s7, 0(s3)
lw s5,4(s7)
```



This is the MUX that does the branch.

add S<sub>1</sub>, S<sub>0</sub>, S<sub>0</sub>  
 sw S<sub>0</sub>, 4 (S<sub>1</sub>) no hazard  
 lw S<sub>1</sub>, 8 (S<sub>0</sub>) read-read  
 add S<sub>1</sub>, S<sub>1</sub>, S<sub>0</sub> b+fw  
 sw S<sub>1</sub>, 12 (S<sub>2</sub>) fw  
 sub S<sub>3</sub>, S<sub>1</sub>, S<sub>0</sub> fw  
 add S<sub>1</sub>, S<sub>3</sub>, S<sub>3</sub>  
 lw S<sub>7</sub>, 0 (S<sub>3</sub>) b+fw  
 lw S<sub>5</sub>, 4 (S<sub>7</sub>)



**Exercise 2.** Consider a two-way associative cache with four sets and block of 8 words. The cache is initially empty and replacement of blocks is done by using LRU. Show which of the following memory accesses are hit and which are misses: 128, 16, 4, 184, 68, 204, 104, 28, 252, 240, 244, 40, 136, 164, 56, 256, 152, 180, 156, 64, 48.

2 - ways associative  
4 sets  
32 bytes block size

4	∅ 8	0
5	1	1
2	6	2
3	7	3

$$M \quad 128 : 32 = 4$$

$$4 \text{ m} 4 = 0$$

$$H \quad 16 : 32 = 0$$

$$0 \text{ m} 4 = 0$$

$$H \quad 4 : 32 = 0$$

$$0 \text{ m} 4 = 0$$

$$M \quad 184 : 32 = 5$$

$$5 \text{ m} 4 = 1$$

$$H \quad 68 : 32 = 2$$

$$2 \text{ m} 4 = 2$$

$$M \quad 204 : 32 = 6$$

$$6 \text{ m} 4 = 2$$

$$H \quad 104 : 32 = 3$$

$$3 \text{ m} 4 = 3$$

$$H \quad 28 : 32 = 0$$

$$0 \text{ m} 4 = 0$$

$$M \quad 252 : 32 = 7$$

$$7 \text{ m} 4 = 3$$

$$H \quad 240 : 32 = 7$$

$$7 \text{ m} 4 = 3$$

$$H \quad 244 : 32 = 7$$

$$7 \text{ m} 4 = 3$$

$$H \quad 40 : 32 = 1$$

$$1 \text{ m} 4 = 1$$

$$H \quad 136 : 32 = 4$$

$$4 \text{ m} 4 = 0$$

$$H \quad 56 : 32 = 1$$

$$1 \text{ m} 4 = 1$$

$$H \quad 164 : 32 = 5$$

$$5 \text{ m} 4 = 1$$

$$M \quad 256 : 32 = 8$$

$$8 \text{ m} 4 = 0$$

$$H \quad 152 : 32 = 4$$

$$4 \text{ m} 4 = 0$$

$$H \quad 180 : 32 = 5$$

$$5 \text{ m} 4 = 1$$

Ex.3 consider the following code  
(single clock architecture)

.data

v: .word 0,1,7,4,2 ... # array of 1.024 integers  
w: .word 0,0,0,0,0... # array of 1.024 integers  
n: .word 1.024 # length of the arrays

.text

lui \$0, 0x10012

lw \$2, 0(\$0) # length of the arrays

lui \$0, 0x10010 # index of the 1st array

lui \$1, 0x10011 # index of the 2nd array

loop: {  
    lw \$0, 0(\$0) # loading in to the integers of the 1st array  
    sw \$0, 0(\$1) # storing the integers into the other array  
    addi \$2,\$2,-1 # decrementing the length  
    addi \$0,\$0,4 # incrementing the index of the 1st array  
    addi \$1,\$1,4 # incrementing the index of the 2nd array  
    bne \$2,zero,loop  
    ecall  
    li \$7, 10  
    ecall

8 x 1.024 = 8.192  
MEM. ACCESSES

This program copy one array into another.

1. **Question A:** what is the approximate total miss rate if you have an instruction cache and a data cache, both one-way associative with 8 blocks of 64 bytes?
2. **Question B:** What is the speed-up (how faster is it) if we use the Risc-V multiple issue architecture with loop unrolling of 4 loops instead of the standard pipelined architecture? Show the code. technique used to reduce the penalty given by the branches.

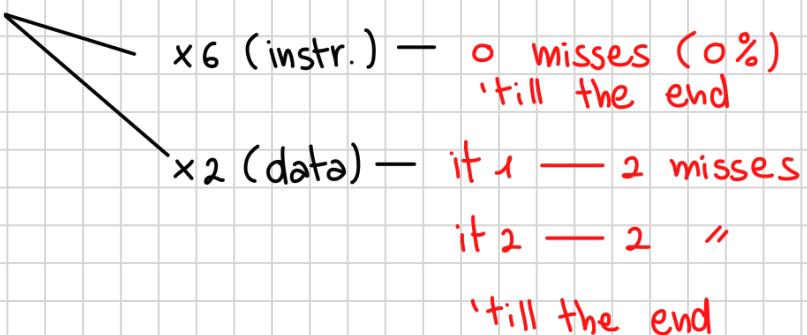
The block size is 64 bytes, so, considering that an instruction fits in 4 bytes (1 word), then all the code fits in 1 block.

There's 1 miss at the start of the execution.

There're 1.024 iterations (loops), and in TOT.  $8 \cdot 1.024 = 8.192$  MEM. accesses.

If we concentrate on the loop:

- x8 MEM. accesses into each loop



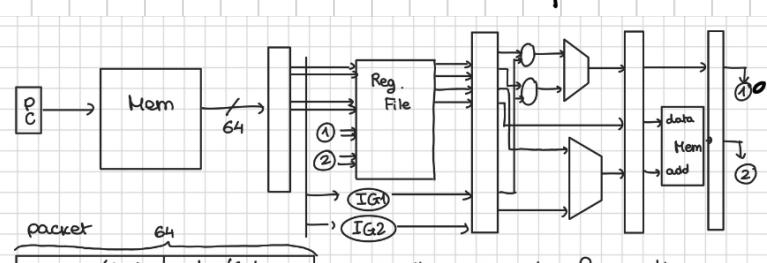
Pay attention to the DATA cache, because it's one-way assoc. and the block containing the data of the 1<sup>st</sup> array, and the block containing the data of the 2<sup>nd</sup> array go both to the set #0 of the cache.

### Question A

$$\text{TOT. Miss RATE} \approx \frac{\frac{n \cdot \text{of misses}}{\text{iterat.}}}{\frac{1.024 \cdot 8}{\text{mem. accesses}}} = \frac{\frac{1}{2}}{\frac{8}{4}} = \frac{1}{4} = 25\% \quad (0.25)$$

### Question B

for this question you need to know the multiple issue arch. and loop unrolling.



load two instr. instead of one  
(they are connected, packets)

→ it's divided because the architecture will be easier

\* There is no hazard unit

addi \$2,\$2,-4 | lw t<sub>0</sub>,0(\$0) 0

addi \$1,\$1,16 | lw t<sub>1</sub>,4(\$0) 1

| lw t<sub>2</sub>,8(\$0) 2

| lw t<sub>3</sub>,12(\$0) 3

✓

addi \$0,\$0,16 | sw t<sub>0</sub>,-16(\$1) 4

| sw t<sub>1</sub>,-12(\$1) 5

| sw t<sub>2</sub>,-8(\$1) 6

bne \$2,zero,loop | sw t<sub>3</sub>,-4(\$1) 7

$$\frac{6 \cdot 1024}{8 \cdot \frac{1024}{4}} = 3 \times \text{faster}$$

VM page Faults → 4

× 3 4KB pages + 1 xtra page

$$4\text{KB} \cdot 3.072 = 12(\overbrace{4+4+4})$$

For the code