UD02. Functional parts of a computer. Part 1

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

Desarrollo de Aplicaciones Web
1er Curso
Curso 2020-2021
Autor: Vicent Bosch
vicent.bosch@ceedcv.es

Recordatorio

ceedcv
CENTRE ESPECÍFIC D´EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

Esta presentación no sustituye los apuntes disponibles en el aula virtual.

Las apuntes oficiales son los que tenéis en el aula virtual

Contents

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

1. Historical Evolution
2. Computer system: definition
3. Computer architectures
4. Functional elements of a computer
5. Instructions
6. Questions

1. Historical evolution

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

1. Non-digital computers
2. Digital computers
   - ✓ General purpose
   - ✓ Transistors

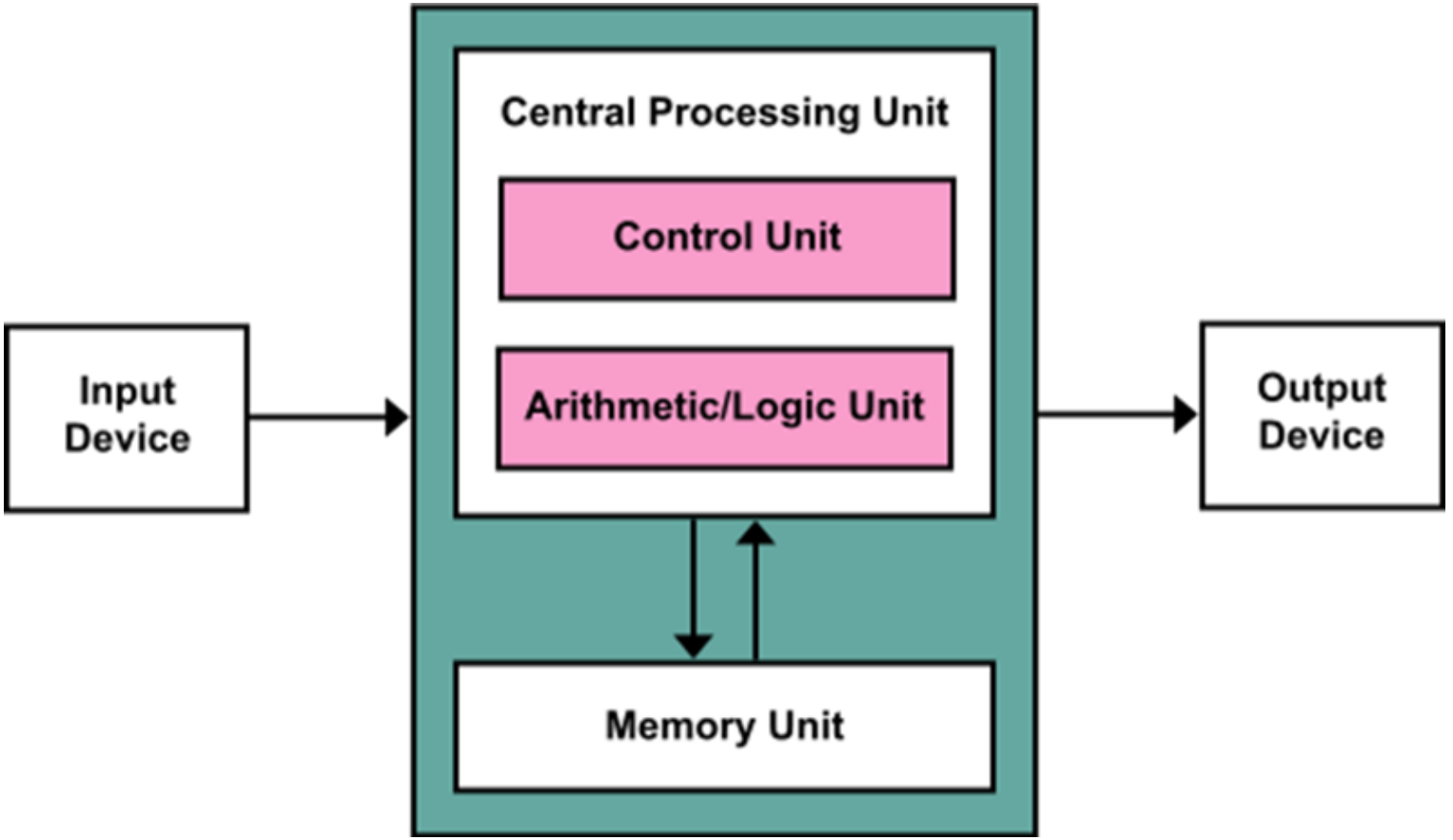
2. What’s a computer system?

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

3. Computer architectures

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

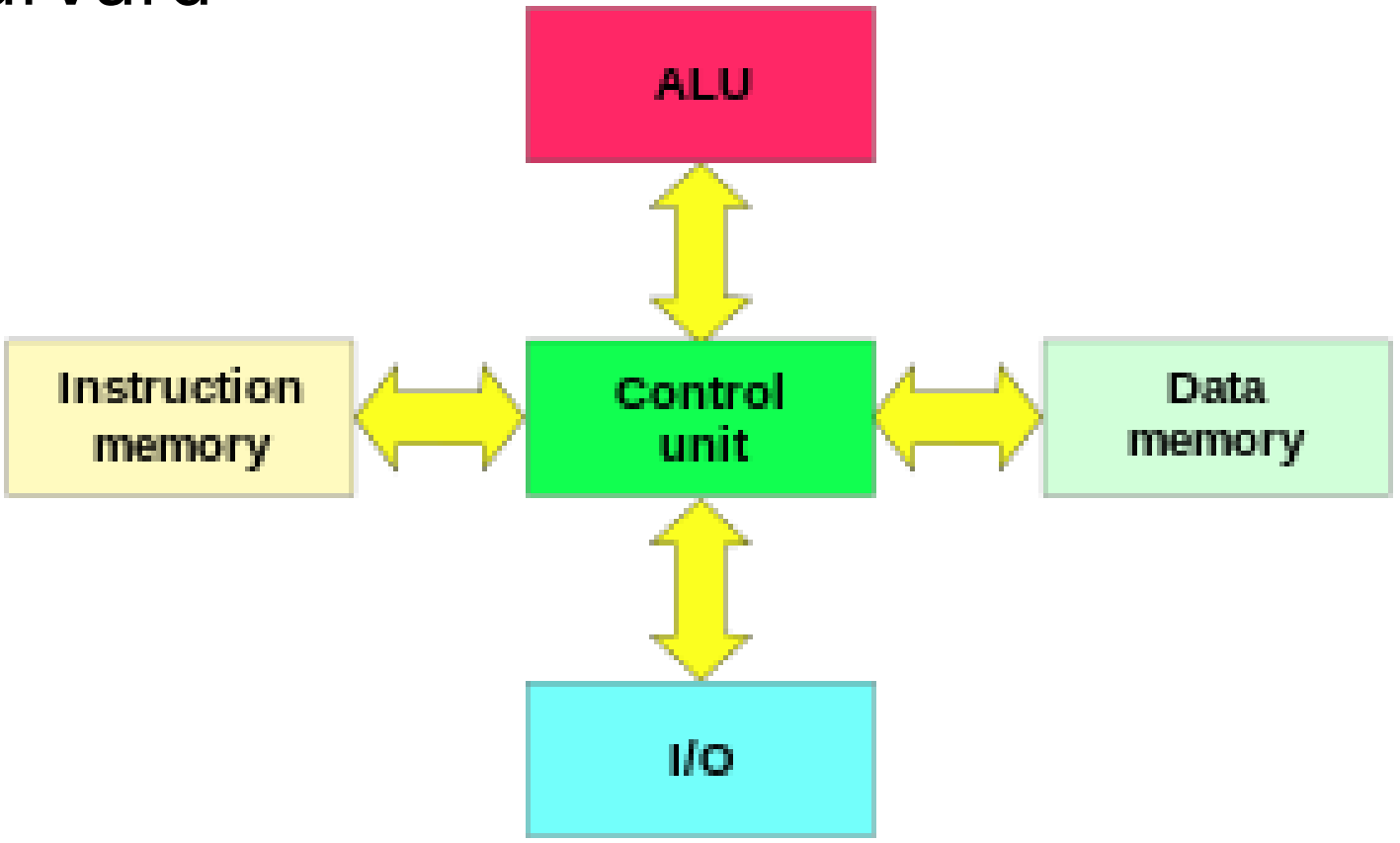
1.Von Neumann



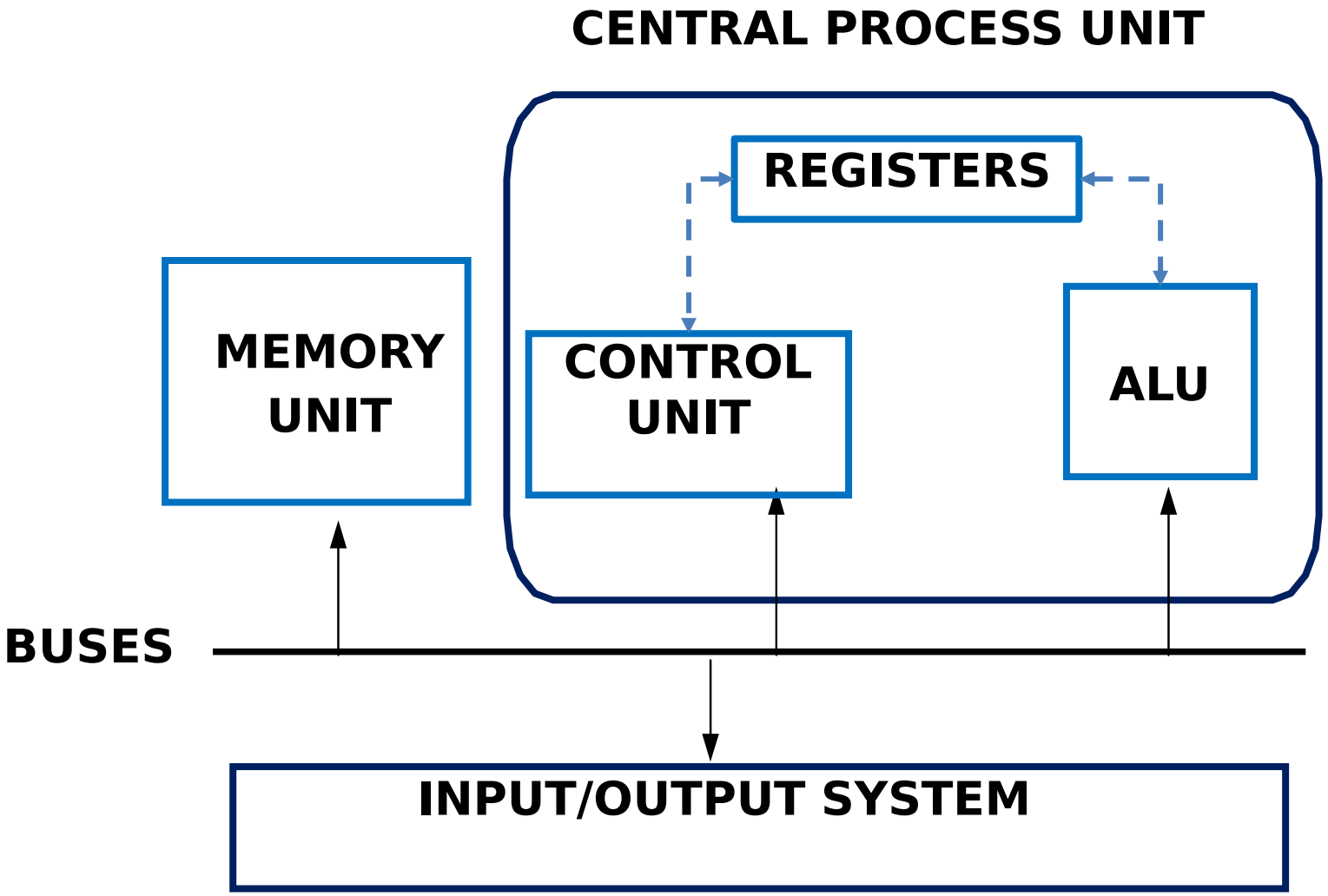
3. Computer architectures

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

1.Harvard



4. Functional elements of a computer



4. Functional elements of a computer: *memory unit*

ceedcv
CENTRE ESPECÍFIC D´EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

- ✓ Word size (8,16,32,64,…bits)
- ✓ Addresses
- ✓ Content

Address	Content (from bit to bit)
0	0..31
1	32..63
2	64..95
3	96..127
4	128..159
5	160..191

5. Instructions (1)

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

- ✓ An application is developed using a programming language.
- ✓ Each line of code of an application is “converted” to set of binary instructions (compilation process).
- ✓ The CPU must have a instruction set to “understand” what to do with each binary instruction.
- ✓ The CU decodes the instruction using its associated format (fields).

5. Instructions (2)

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

- ✓ The CPU has an ***instruction set*** which contains all the instructions that can execute.
- ✓ Each instruction is based on an ***instruction format***. There can be different instruction formats, depending on different facts such as the ***addressing modes***.
- ✓ An addressing mode is the way a data is referenced in the instruction.

OP_CODE (2 bits)	ADDRESS OPERAND_1 (3 bits)	ADDRESS OPERAND_2 (3 bits)
1 0	0 1 0	1 0 1

OP_CODE (2 bits)	ADDRESS OPERAND_1 (3 bits)	DATA (3 bits)
1 1	0 1 0	1 0 1

5. Instructions (3). Addressing mode

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

OP_CODE (2 bits)	ADDRESS OPERAND_1 (3 bits)	ADDRESS OPERAND_2 (3 bits)
1 0	0 1 0	1 0 1

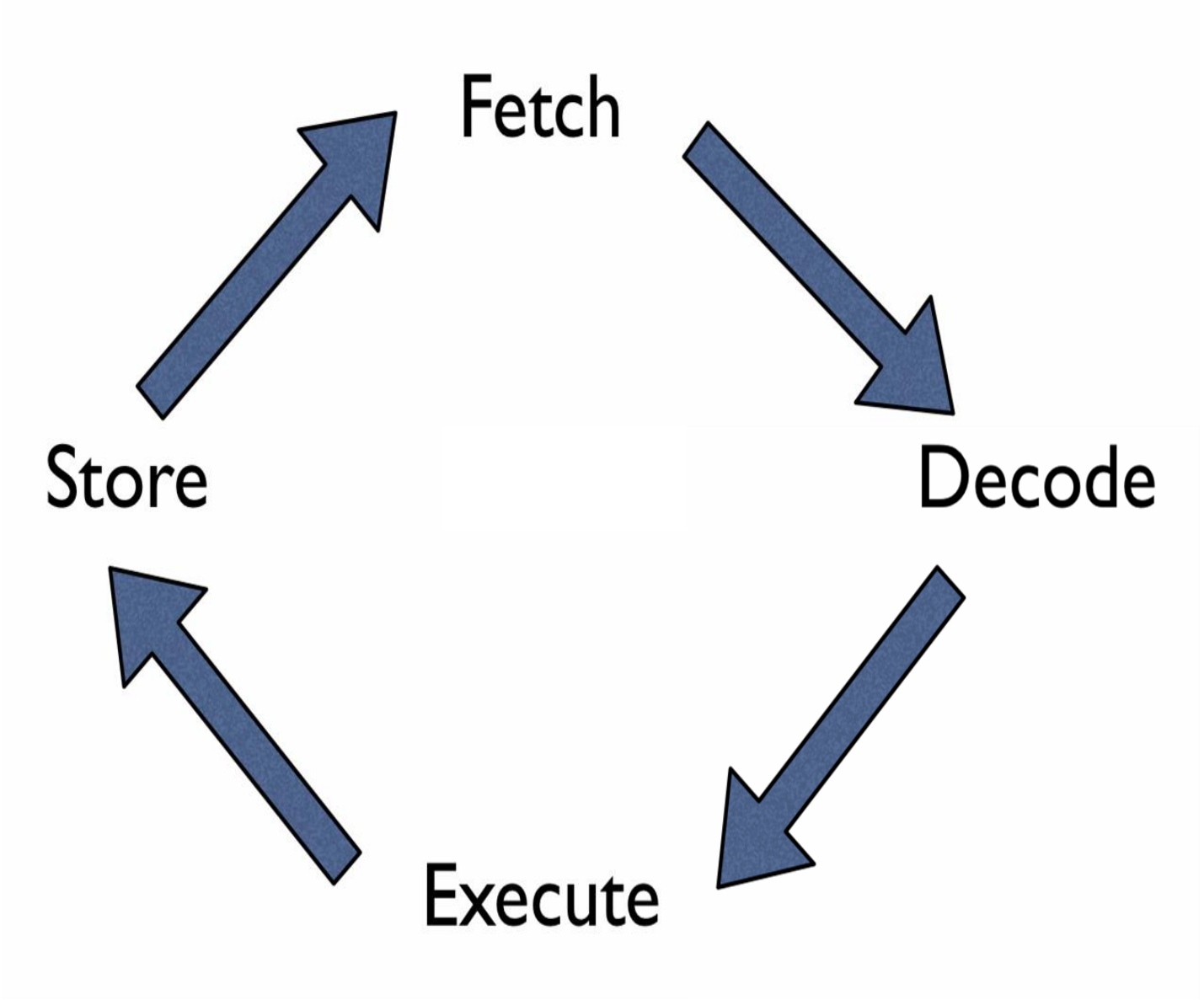
Address	Content (Word size = 16 bits)
...	
0 1 0	$A893_{16}$
0 1 1	$239C_{16}$
1 0 0	$F057_{16}$
1 0 1	0013_{16}
...	

5. Instructions: example

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

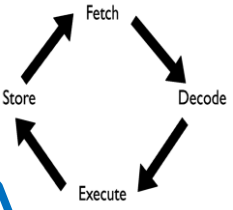
- ✓ 8 bits used to codify an instruction.
- ✓ Instruction format (from left to right):
  - ✓ 2 bits to identify the instruction. For example, the *add* instruction:
    - the op_code is 00
    - add Operand1, Operand2, Destination →Destination=Operand1+Operand2
  - ✓ 3 groups of 2 bits to identify an internal register (R0, R1, R2, R3)
- ✓ If the CPU has to execute **00 00 11 01 →add R0 R3 R1**
- ✓ Which addressing mode is using?

5. Instruction cycle (0)



5. Instruction cycle (1)

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA



✓ **Fetch the instruction**

CENTRAL PROCESS UNIT

MEMORY UNIT (hex)

Address	Content
4	8
5	5
6	A109
7	C0A7
8	FA73
...	

REGISTERS

PC 0x6 **PC 0x7**

RI **RI A109**

ALU

CONTROL UNIT

MAR → 0x6

MDR → A109

AC

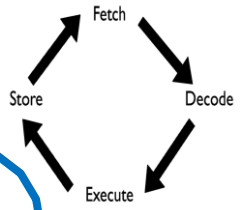
BUSES

General Purpose Register for the ALU. Stores the result of an ALU operation

PC 0x7 **at the end of the cycle**

5. Instruction cycle (2)

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA



✓ **Decode the instruction**

CENTRAL PROCESS UNIT

MEMORY UNIT (hex)

Address	Content
4	8
5	
6	A109
7	C0A7
8	FA73
...	

CONTROL UNIT

MAR → 0x4

MDR → 0x8

REGISTERS

PC 0x7

RI A109

R5 → 5

ALU

A109→ 1 0 1 0 0 0 0 1 0 0 0 0 0 1 0 1

OP_CODE (4 bits)	ADDRESS OPERAND_1 (6 bits)	INTERNAL REGISTER (6 bits)
1 0 1 0	0 0 0 1 0 0	0 0 0 1 0 1

Search for the instruction in the instruction set

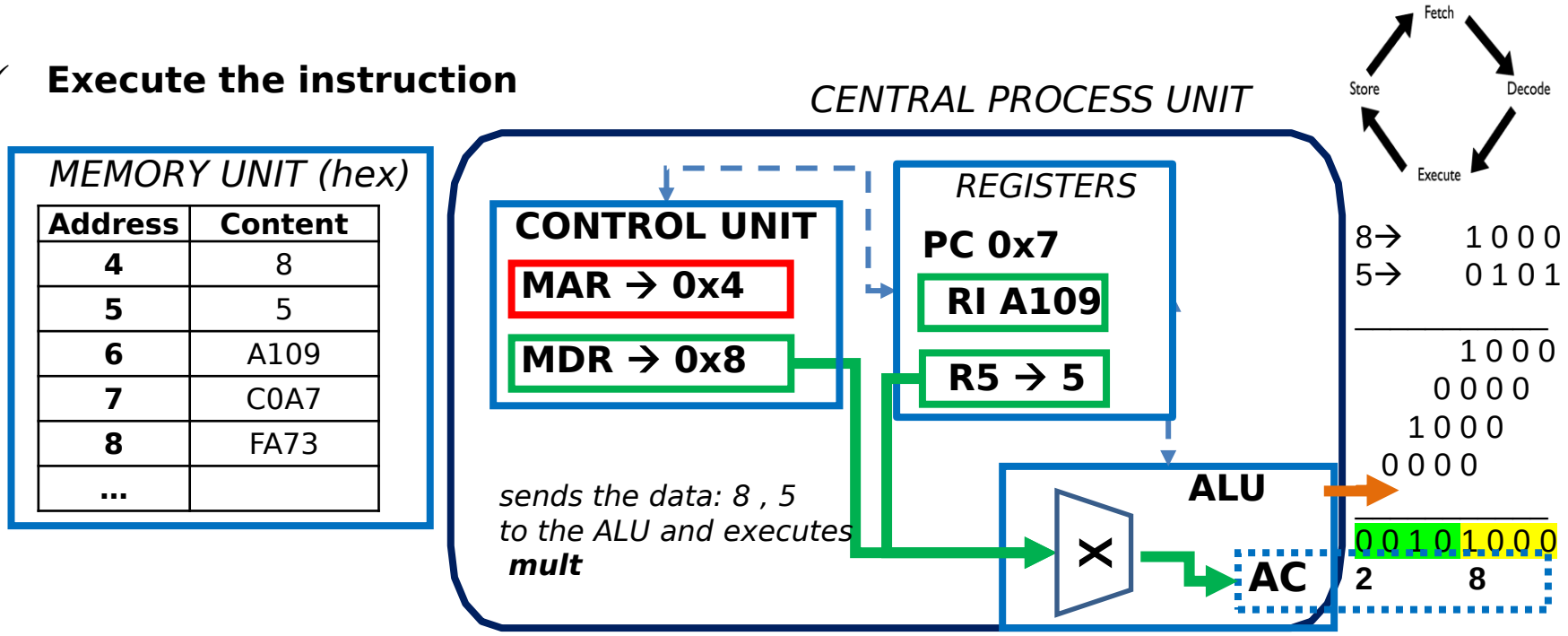
Search for the address in the memory unit where the content is stored

Search for the internal register in the CPU where the content is stored

5. Instruction cycle (3)

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

- ✓ **Execute the instruction**



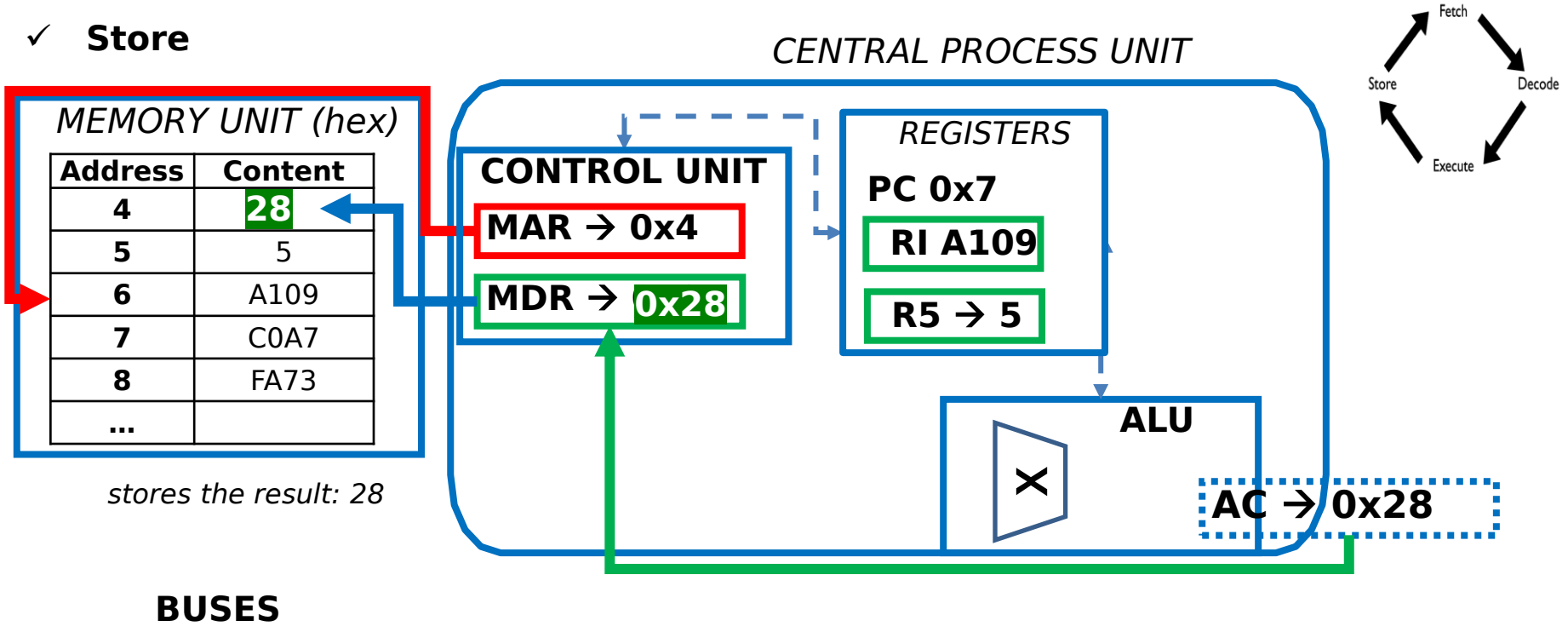
BUSES

Depends on the type of instruction: add, mult, load, store, etc...In the example let's suppose that the instruction send both contents (operand 1 and operand 2) to the ALU to multiply them and the result is stored in the address of operand 1

5. Instruction cycle (4)

ceedcv
CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

✓ **Store**



stores the result: 28

BUSES

TC 26th to 30th October

ceedcv
CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

- ✓ Review exercises about the instruction cycle.
- ✓ Configure Visual Studio Code and Python.
- ✓ Introduction to Python: basics.