



ÉLECTRONIQUE ET COMPOSANTS DES SYSTÈMES

CHAPITRE 2: ÉLÉMENTS D'UN ORDINATEUR

Dr. Boudebza Souaad

souad.boudebza@univ-jijel.dz

Site du cours: <https://sites.google.com/esi.dz/ecs-site/accueil>

- ▶ Principaux éléments d'un ordinateur
- ▶ Fonctionnement de l'ordinateur
- ▶ Unité centrale de traitement (CPU)
- ▶ Mémoire principale
- ▶ Unités d'entrée/sortie
- ▶ Bus

PRINCIPAUX ÉLÉMENTS D'UN ORDINATEUR

COMPOSANTS DE BASE

Unité centrale



Appareil photo

PÉRIPHÉRIQUES
D'ENTRÉE



Clé USB



Webcam

Ecran



Clavier

Souris



Carte mémoire



Lecteur de cartes



Imprimante



Modem



Disque dur externe

PÉRIPHÉRIQUES DE
SORTIES

PÉRIPHÉRIQUES D'ENTRÉE/
SORTIES

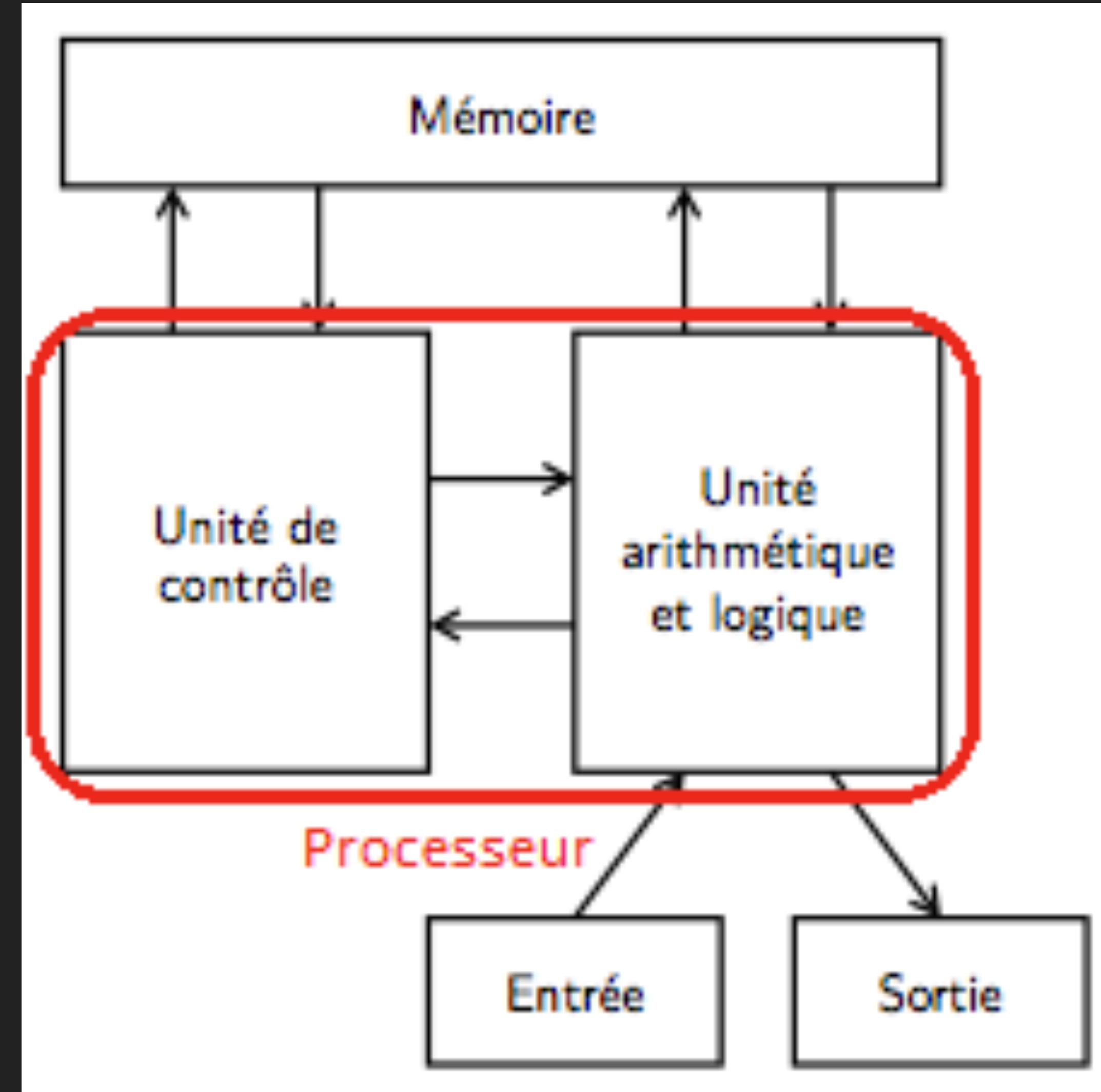
PRINCIPAUX ÉLÉMENTS D'UN ORDINATEUR

L'ARCHITECTURE DE VON NEUMANN

- ▶ Toutes les réalisations actuelles d'ordinateurs reposent sur les concepts développés par John von Neumann (1946), ou ce qu'on appelle *l'architecture de John von Neumann*.
- ▶ Cette architecture est fondée sur quatre composants:
 1. Une *mémoire centrale*, qui contient les programmes et les données,
 2. Une *unité centrale de traitement* qui exécute un programme stocké en mémoire centrale. Elle est composée de:
 - ▶ *Unité de contrôle* ou unité commande
 - ▶ *Unité arithmétique et logique* ou unité de calcul
 3. Des *unités d'entrée/sortie* permettant l'échange d'informations avec les unités périphériques.
 4. Ces différents composants sont reliés par des voies de communication appelés *bus*.

PRINCIPAUX ÉLÉMENTS D'UN ORDINATEUR

L'ARCHITECTURE DE VON NEUMANN



FONCTIONNEMENT DE L'ORDINATEUR

- ▶ Le programme et les données sont chargés en mémoire centrale (d'où le nom machine à programme enregistré)
- ▶ L'exécution d'un programme se déroule comme suit:
 - ▶ Les instructions du programme sont amenés séquentiellement (une par une) à l'unité de contrôle qui les analyse et déclenche le traitement approprié en envoyant les signaux à l'unité arithmétique et logique. Le passage à l'instruction suivante est automatique.
 - ▶ Le traitement peut nécessiter de faire appel aux unités d'entrée/sorties ou à la mémoire principale.
 - ▶ Les informations sont manipulées sous forme numérique (en binnaire).

UNITÉ CENTRALE DE TRAITEMENT (CPU)

STRUCTURE

- ▶ CPU(Central processing Unit) ou processeur.
- ▶ Le cerveau de l'ordinateur.
- ▶ Physiquement, est un *circuit intégré* qui regroupe plusieurs millions voir des milliards de *transistors* élémentaires interconnectés.

UNITÉ CENTRALE DE TRAITEMENT (CPU)

STRUCTURE

- ▶ CPU(Central processing Unit) ou processeur.
- ▶ Il est constitué:
 - ▶ Unité de contrôle : qui s'occupe de gérer l'exécution des instructions d'un programme.
 - ▶ Unité arithmétique et logique qui réalisent les calculs.

UNITÉ CENTRALE DE TRAITEMENT (CPU)

STRUCTURE

- ▶ Unité de contrôle contient:
- ▶ Registres: des zones de stockage temporaires d'accès très rapide pour stocker une donnée, instruction ou adresse.
- ▶ décodeur: un dispositif de décodage des instructions
- ▶ séquenceur de commande qui active les circuits nécessaires à l'exécution de l'instruction en cours.
- ▶ Cette unité a besoin des signaux d'une horloge pour enchaîner les commandes. L'horloge est généralement externe à l'unité.

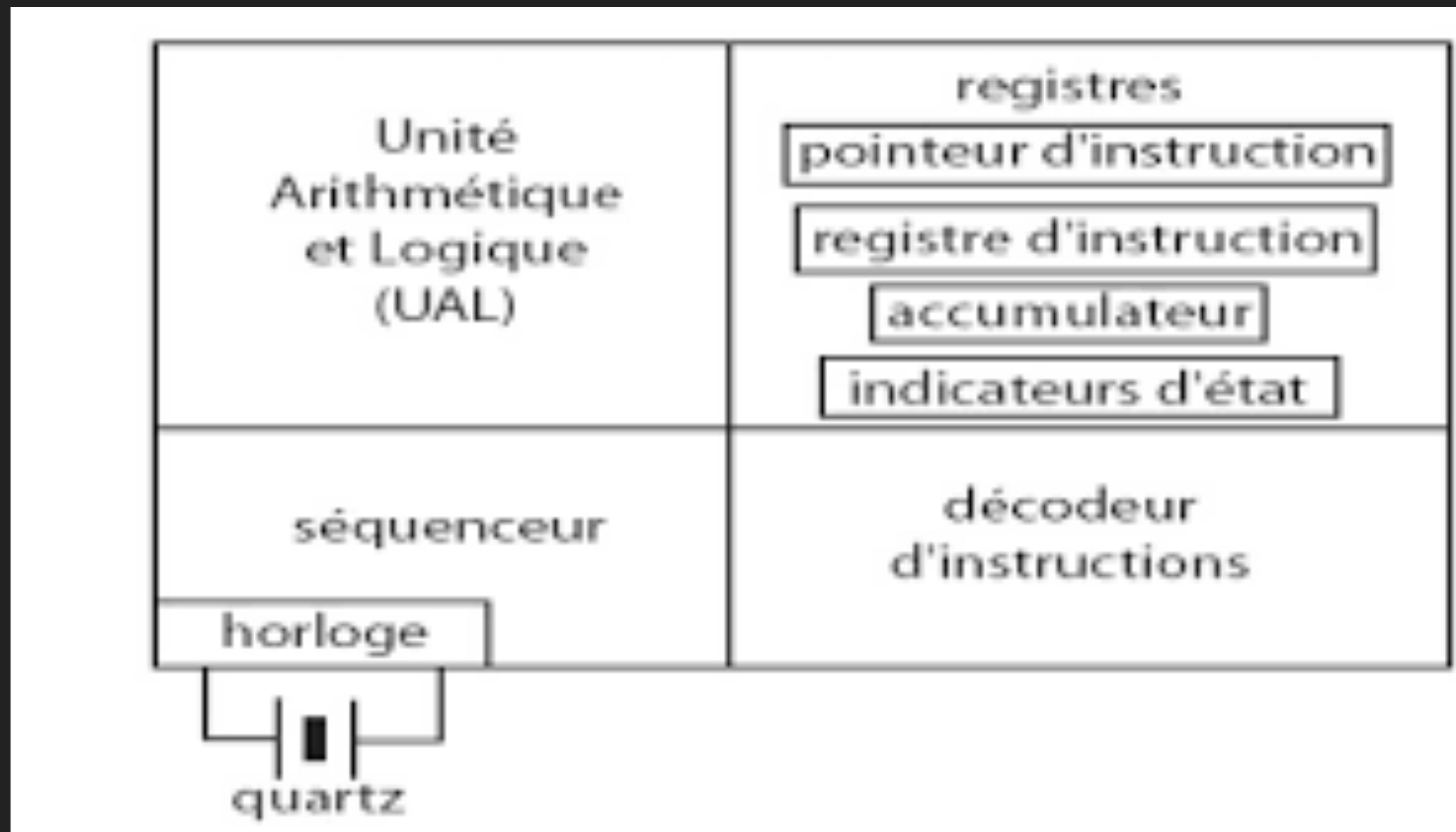
UNITÉ CENTRALE DE TRAITEMENT (CPU)

STRUCTURE

- ▶ Unité arithmétique et logique ou unité de calcul contient des circuits électroniques qui réalisent effectivement les opérations désirées.
- ▶ Ces opérations sont principalement l'addition, sa soustraction, la multiplication, la division, la negation (inversion des bits), les opérations logiques (ET, OU, OU exclusif).
- ▶ Les opérandes nécessaires pour ces opérations se trouvent dans des registres de cette unité. Ces registres sont accessibles au programmeur.

UNITÉ CENTRALE DE TRAITEMENT (CPU)

STRUCTURE



UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge: en Hz (MHz ou GHz)
- ▶ CIP (Cycle Par Instruction)
- ▶ MIPS (Millions Instructions Par Seconde)
- ▶ Format de données

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ Les processeurs sont cadencés par ce que l'on appelle une horloge.
- ▶ L'horloge est réalisée au moyen d'un dispositif à quartz. Elle peut générer des impulsions, qui sont utilisées pour à synchroniser les circuits présents dans notre processeur.
- ▶ Un « cycle » est la durée entre deux impulsions.
- ▶ Au cours de chaque cycle, le processeur peut effectuer une opération de base, telle que récupérer une instruction, accéder à la mémoire ou écrire des données. Les opérations complexes nécessitent plusieurs cycles d'horloge.

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ Le nombre des cycles de l'horloge qui ont lieu durant une seconde s'appelle la fréquence du processeur.
- ▶ Cette fréquence se mesure en hertz (Hz), généralement en GHz (gigahertz).

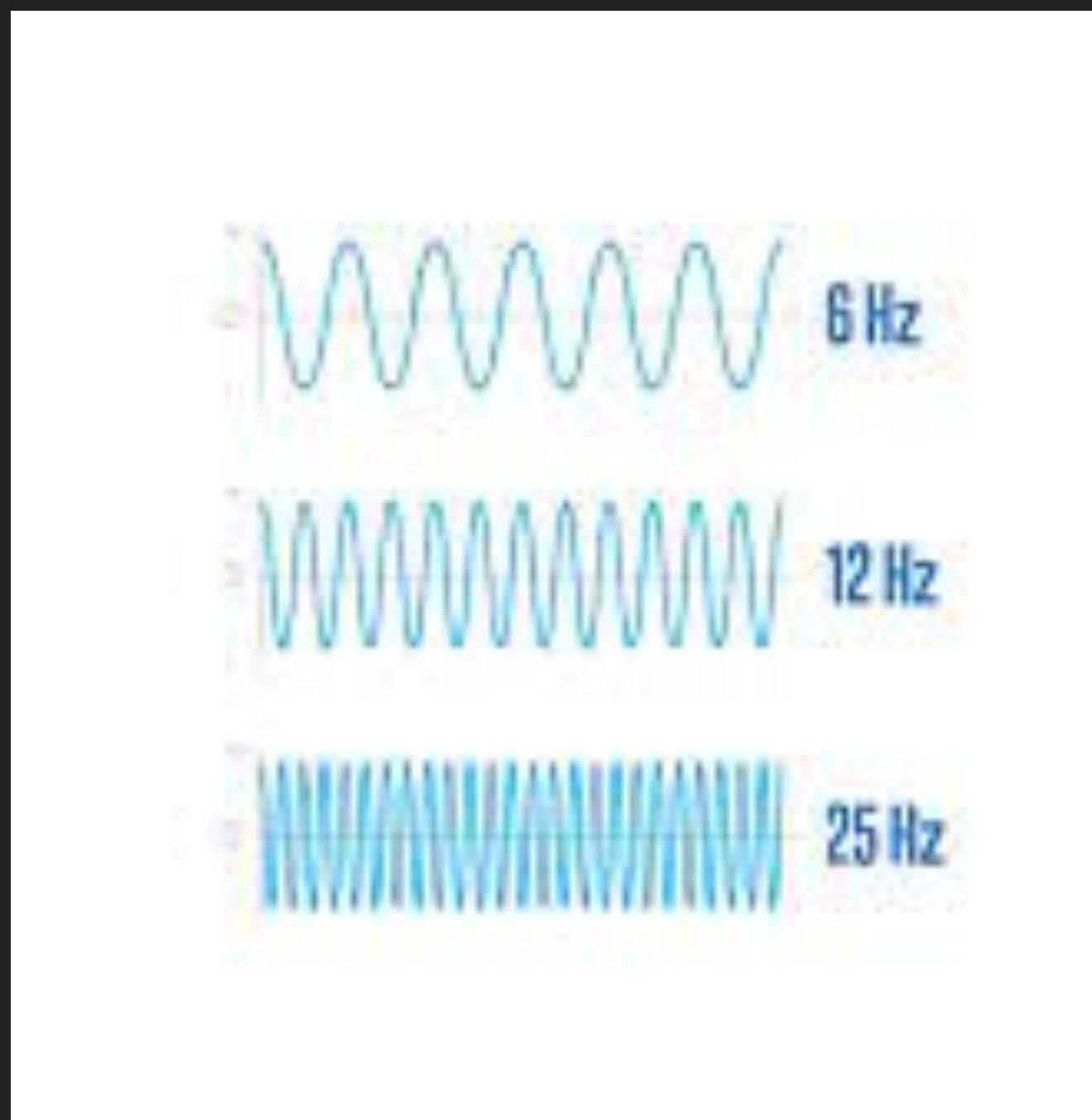
Ordres de grandeur: Hertz

K	Kilo	10^3
M	Méga	10^6
G	Giga	10^9
T	Tera	10^{12}

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ Exemple:



UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ Le temps de cycle horloge de processeur est l'inverse de la fréquence:

$$T_{cycle_h} = \frac{1}{F} \text{ (avec F en Hz)}$$

- ▶ Il est souvent mesuré en nanoseconde (ns)

Ordres de grandeur: seconde

m	milli	10^{-3}
μ	micro	10^{-6}
n	nano	10^{-9}
p	pico	10^{-12}

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ **Exemple:** Un processeur dont la vitesse d'horloge est de 3,2 GHz

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ **Exemple:** Un processeur dont la vitesse d'horloge est de 3,2 GHz exécute 3,2 milliards de cycles par seconde.
- ▶ Le temps de cycle d'horloge de ce processeur est:

$$T_{cycleH} = \frac{1}{F} = \frac{1}{3,2 \times 10^9} = 0,315ns$$

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ Fréquence d'horloge:
- ▶ La vitesse d'horloge est importante pour déterminer les performances globales du processeur, mais ce n'est pas le seul facteur.
- ▶ Il existe d'autres indicateurs de performance, tel que le CPI (cycles par instruction) ou le MIPS (Millions Instructions Par Seconde).

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ CPI (Cycle Par Instruction)
- ▶ A chaque cycle le processeur exécute une action, correspondant à une instruction ou une partie d'instruction. Donc chaque instruction nécessite un certain nombre de cycles d'horloge pour s'exécuter, c'est ce qu'on appelle CPI.
- ▶ Le CPI est une mesure qui consiste à évaluer le nombre de cycles d'horloge requis pour exécuter chaque instruction par un processeur.
- ▶ Le CPI d'un programme donné est calculé en divisant le nombre de cycles d'horloge requis pour exécuter ce programme par le nombre d'instructions exécutées pour le faire tourner.

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ CPI (Cycle Par Instruction)
- ▶ **Exemple 1:**
- ▶ Un programme donné est constitué de 4200 instructions. S'il faut 16000 cycles pour exécuter le programme sur un processeur cadencé à 2,1 MHz , quelle est la valeur de CPI?

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ CPI (Cycle Par Instruction)
- ▶ Exemple 1:
- ▶ Un programme donné est constitué de 4200 instructions. S'il faut 16000 cycles pour exécuter le programme sur un processeur cadencé à 2,1 MHz , quelle est la valeur de CPI?

$$CPI = \frac{Nbr_cycles_prg}{Nbr_ins_prg} = \frac{16000}{4200} = 3,8 = 4$$

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ CPI (Cycle Par Instruction)
- ▶ **Exemple 2:**
- ▶ Un programme exécuté sur une machine donnée requiert 1000000 de cycles. Si le système réalise un CPI de 40, combien d'instructions ont été exécutées pour faire tourner le programme?

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ CPI (Cycle Par Instruction)
- ▶ **Exemple 2:**
- ▶ Un programme exécuté sur une machine donnée requiert 1000000 de cycles. Si le système réalise un CPI de 40, combien d'instructions ont été exécutées pour faire tourner le programme?

$$Nbr_{ins_prg} = \frac{Nbr_{cycles_prg}}{CPI} = \frac{1000000}{40} = 25000$$

CARACTÉRISTIQUES

- ▶ *CPI (Cycle Par Instruction)*
- ▶ **Exemple 3:**
- ▶ Soit un processeur ayant une fréquence de 3,2 GHz et un CPI de 40. Un programme est exécuté sur ce processeur en 6,552 ms. Combien d'instructions ont été exécutées pour faire tourner le programme?

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ CPI (Cycle Par Instruction)
- ▶ **Exemple 3:**
- ▶ Soit un processeur ayant une fréquence de 3,2 GHz et un CPI de 40. Un programme est exécuté sur ce processeur en 6,552 ms. Combien d'instructions ont été exécutées pour faire tourner le programme?

$$Nbr_{ins_prg} = \frac{Nbr_{cycles_prg}}{CPI}$$

$$Nbr_{cycles_prg} = \frac{T_{exe_prg}}{T_{cycle_h}}$$

$$Nbr_{ins_prg} = \frac{T_{exe_prg}}{CPI \times T_{cycle_h}}$$

$$T_{cycle_h} = \frac{1}{F}$$

$$Nbr_{ins_prg} = \frac{T_{exe_prg} \times F}{CPI} = \frac{6,552 \times 10^{-3} \times 3,2 \times 10^9}{40} = 524160$$

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ MIPS (Millions Instructions Par Seconde)
- ▶ Indique la vitesse à laquelle la machine pouvait exécuter des instructions.
- ▶ On calcul cette vitesse en divisant le nombre d'instructions traitées dans un programme par le temps requis pour faire tourner ce programme. Le plus souvent, la valeur obtenue est mesurée en millions d'instructions par seconde (MIPS).

$$MIPS = \frac{Nbr_{ins_prg}}{T_{exe_prg}} \times 10^{-6} \text{ avec } T_{exe_prg} \text{ en seconde}$$

- ▶ On peut la calculer en utilisant le CPI:

$$MIPS = \frac{F}{CPI} \quad \text{avec } F \text{ en MHz}$$

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ *MIPS (Millions Instructions Par Seconde)*
- ▶ **Exemple 1:** Un programme donné est constitué de 4200 instructions. S'il faut 16000 cycles pour exécuter le programme sur un processeur cadencé à 2,1 GHz , quelle est la valeur de MIPS?

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ MIPS (Millions Instructions Par Seconde)
- ▶ **Exemple 1:** Un programme donné est constitué de 4200 instructions. S'il faut 16000 cycles pour exécuter le programme sur un processeur cadencé à 2,1 GHz , quelle est la valeur de MIPS?

$$T_{cycle_h} = \frac{1}{F} = \frac{1}{2,1 \times 10^9} = 0,476 \times 10^{-9} s$$

$$MIPS = \frac{4200}{16000 \times 0,476 \times 10^{-9}} \times 10^{-6} = 551,47$$

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ *MIPS (Millions Instructions Par Seconde)*
- ▶ **Exemple 2:** Soit un processeur cadencé à 2,4 GHz. Le nombre de cycles d'horloge nécessaires pour exécuter une instruction sur ce processeur est 8 cycles. Combien de MIPS pouvant être exécutées par ce processeur?

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ *MIPS (Millions Instructions Par Seconde)*
- ▶ **Exemple 2:** Soit un processeur cadencé à 2,4 GHz. Le nombre de cycles d'horloge nécessaires pour exécuter une instruction sur ce processeur est 8 cycles. Combien de MIPS pouvant être exécutées par ce processeur?

$$MIPS = \frac{F}{CPI} = \frac{2,4 \times 10^3}{8} = 300$$

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ MIPS (Millions Instructions Par Seconde)
- ▶ **Exemple 3:** Lors de l'exécution d'un programme donné, l'ordinateur A exécute 100 MIPS contre 75 MIPS pour l'ordinateur B. En revanche, l'ordinateur A prend 60 secondes pour exécuter le programme, tandis que l'ordinateur B n'en requiert que 45. Comment cela est-il possible?

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ MIPS (Millions Instructions Par Seconde)
- ▶ **Exemple 3:** Lors de l'exécution d'un programme donné, l'ordinateur A exécute 100 MIPS contre 75 MIPS pour l'ordinateur B. En revanche, l'ordinateur A prend 60 secondes pour exécuter le programme, tandis que l'ordinateur B n'en requiert que 45. Comment cela est-il possible?
- ▶ **Réponse:**
 - ▶ L'ordinateur A prends plus de temps pour exécuter une instruction, même s'il parvient à exécuter plus d'instructions par secondes.
 - ▶ Le temps d'exécution d'une instruction dépend du CPI et de temps de cycle d'horloge du processeur (ou la fréquence).

UNITÉ CENTRALE DE TRAITEMENT (CPU)

CARACTÉRISTIQUES

- ▶ *Format de données*
- ▶ Indique la taille des données que peut traiter le processeur.
- ▶ Il est exprimé en bits.

ORGANISATION

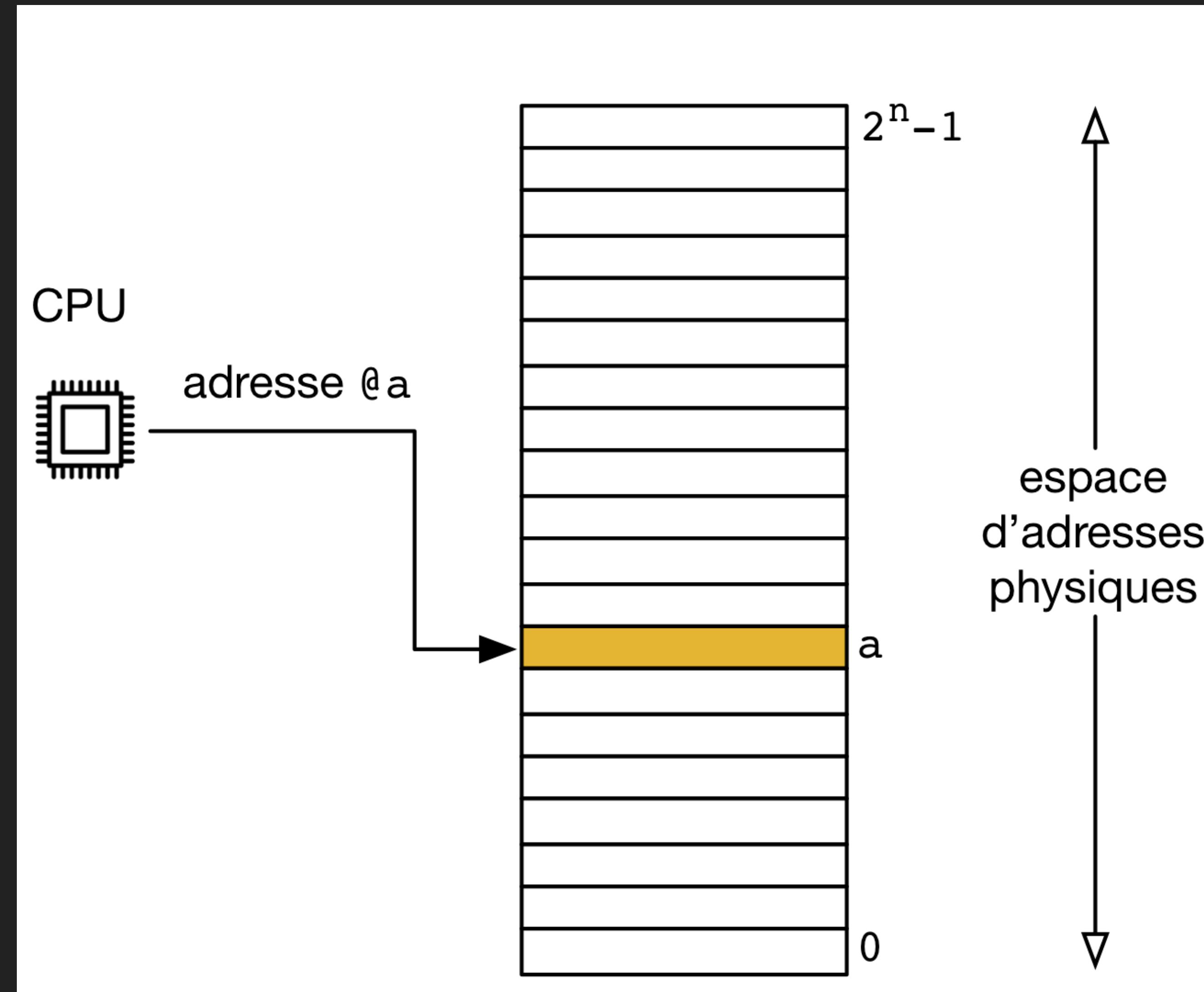
- ▶ On appelle *mémoire* tout dispositif capable *d'enregistrer, de conserver et de restituer* des informations.
- ▶ La *mémoire centrale* est une mémoire interne, réalisée à base de semi-conducteurs.
- ▶ C'est l'organe principale de rangement des informations. Elle contient principalement deux types d'informations: les *instructions* des différents programmes et les *données* traitées lors de leurs l'exécution.
- ▶ Les informations sont stockées en binaire: au niveau physique, la mémoire ne contient que des *bits* qui constituent l'unité élémentaire d'information.
- ▶ Les bits sont regroupés en 8, ou ce qu'on appelle *octet* (ou *byte* en anglais).
- ▶ Avec 8 bits, il est possible de coder 2^8 informations différentes.

ORGANISATION

- ▶ La mémoire centrale est divisée physiquement en cellules.
- ▶ Chaque cellule correspond à un mot mémoire et possède une adresse qui lui est propre.
- ▶ La longueur du mot mémoire a beaucoup varié d'une machine à une autre, par exemple, 1, 4, 8, 16, 32, 64 bits. À présent les valeurs 32, et 64 se sont généralisés dans la plupart des ordinateurs.
- ▶ Le mot mémoire est l'unité d'information adressable, c'est à dire que toute opération de lecture ou d'écriture porte sur un mot mémoire.
- ▶ À chaque mot mémoire est associé une adresse indiquant sa position en mémoire et un contenu (instruction ou donnée)

MÉMOIRE PRINCIPALE

ORGANISATION



ORGANISATION

- ▶ Il y a deux types de registres dans la mémoire centrale:
 - ▶ Le registre d'adresse qui contient l'adresse d'un mot mémoire
 - ▶ un registre d'adresse de n bits permet d'adresser 2^n mots différents.
 - ▶ Le registre de données qui contenait un mot mémoire.
- ▶ Exemple:
 - ▶ Si la mémoire comporte 256 mots, le registre d'adresse doit avoir une taille de : $\log_2(256)=\log_2 (2^8)=8$ bits.
 - ▶ Un registre de 32 bits permet d'adresser $2^{32}=4294667296$ mots mémoire.

ORGANISATION

- ▶ La mémoire centrale se compose de deux parties :
 - ▶ la RAM (Random Access Memory) pour mémoire à accès aléatoire: une mémoire vive chargé de stocker les programmes et les données en cours du traitement. On peut écrire et lire dedans. C'est une mémoire volatile qui ne conserve pas les données en absence du courant.
 - ▶ la ROM (Read Only Memory) pour mémoire à lecture seule: une mémoire morte qui contient les premières instructions nécessaires à la machine pour démarrer. C'est une mémoire permanente (non-volatile), son contenu est maintenu lorsque le courant est coupé.

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ *La volatilité:*
- ▶ Elle caractérise la permanence des informations dans une mémoire.
- ▶ Si une mémoire perd son contenu lorsque la source de l'alimentation est coupée alors la mémoire est dite volatile (exemple de RAM).
- ▶ Si la mémoire ,e perd pas son contenu lorsque la source de l'alimentation est coupée la mémoire est dite non volatile ou permanente ou encore stable (exemple de ROM).

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ Mode d'accès à l'information (lecture/écriture):
- ▶ Sur une mémoire, on peut effectuer deux opérations:
 - ▶ Lecture: consiste à récupérer ou à restituer une information à partir de la mémoire. Le processeur demande à la mémoire la valeur contenue à l'emplacement dont il indique dans l'adresse. Le contenu de l'emplacement reste inchangé.
 - ▶ Ecriture: consiste à enregistrer une nouvelle information ou à modifier une information déjà existante dans la mémoire. Le processeur donne une valeur et une adresse et la mémoire range la valeur à l'emplacement indiqué par l'adresse.
- ▶ Il existe des mémoires qui offrent les deux modes lecture/écriture: mémoires vives et des mémoires qui offrent seulement la possibilité de la lecture (mémoire mortes)
- ▶ Les opérations de lecture/écriture portent sur des mots mémoires.

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ *La capacité:*
- ▶ La *capacité* d'une mémoire est le nombre (quantité) d'information qu'on peut enregistrer dans cette mémoire.
- ▶ Elle est calculée en fonction du *nombre de mots mémoire* ainsi que du *nombre de bits par mot:*

$$Capacite = Nbr_MM \times Taille_MM = Nbr_Cellules \times Taille_MM$$

- ▶ Elle s'exprime en bits, octets ou multiples d'octets.

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ *La capacité:*
- ▶ Exemple: Soit une machine a 262144 mots mémoire de 64 bits. Quelle est sa capacité en o, Ko et Mo.

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ La capacité:
- ▶ Exemple: Soit une machine a 262144 mots mémoire de 64 bits. Quelle est sa capacité en o, Ko et Mo.

Ordres de grandeur: octet

K	Kilo	2^{10}	1 024
M	Méga	2^{20}	1 048 576
G	Giga	2^{30}	1 073 741 824
T	Tera	2^{40}	1 099 511 627 776

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ La capacité:
- ▶ Exemple: Soit une machine a 262144 mots mémoire de 64 bits.
Quelle est sa capacité en o, Ko et Mo.
- ▶ $Capacite = 262144 \times 8 = 2097152 \text{ o}$
- ▶ $Capacite = \frac{2097152}{2^{10}} = 2048 \text{ Ko}$
- ▶ $Capacite = \frac{2048}{2^{10}} = 1,99 \text{ Mo}$

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ Taille du mot mémoire:
- ▶ Elle dépend du type du processeur:
 - ▶ 1 octet dans les processeur 8 bits (Motorola 6502)
 - ▶ 2 octets dans les processeurs 16 bits (Intel 8086)
 - ▶ 4 octets dans les processeurs 32 bits (Intel 80486 ou Motorola 68030)
 - ▶ 8 octets dans les processeurs 64 bits (les Itanium d'Intel, AMD64 de AMD ou Intel 64 d'Intel)

CARACTÉRISTIQUES

- ▶ Le temps d'accès:
- ▶ C'est le temps qui s'écoule entre le lancement d'une opération d'accès (lecture ou écriture) et son accomplissement.
- ▶ Le temps d'accès est un critère important pour déterminer les performances d'une mémoire ainsi que les performances d'une machine.

CARACTÉRISTIQUES

- ▶ Le temps de cycle:
- ▶ Représente l'interval de temps minimal qui doit séparer deux demandes successives de lecture ou d'écriture.
- ▶ Le temps de cycle est toujours supérieur au temps d'accès. Car la mémoire a besoin d'opérations supplémentaire entre deux accès (stabilisation de signaux, synchronisation, rafraîchissement, etc).

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ Le débit:
- ▶ C'est le nombre d'informations lues ou écrites par seconde.
- ▶ Il est exprimé en bits (octet, ou multiples d'octets)/seconde
- ▶ Il est calculé en divisant la taille du mot mémoire par temps de cycle mémoire.

$$debit = \frac{\text{TailleMM}}{T_{cycle}}$$

MÉMOIRE PRINCIPALE

CARACTÉRISTIQUES

- ▶ Le débit:
- ▶ Exemple: Quel est le débit d'une mémoire ayant une format de données de 2 octets et un temps de cycle mémoire de 23 ns?

CARACTÉRISTIQUES

- ▶ Le débit:
- ▶ Exemple: Quel est le débit d'une mémoire ayant une format de données de 2 octets et un temps de cycle mémoire de 23 ns?

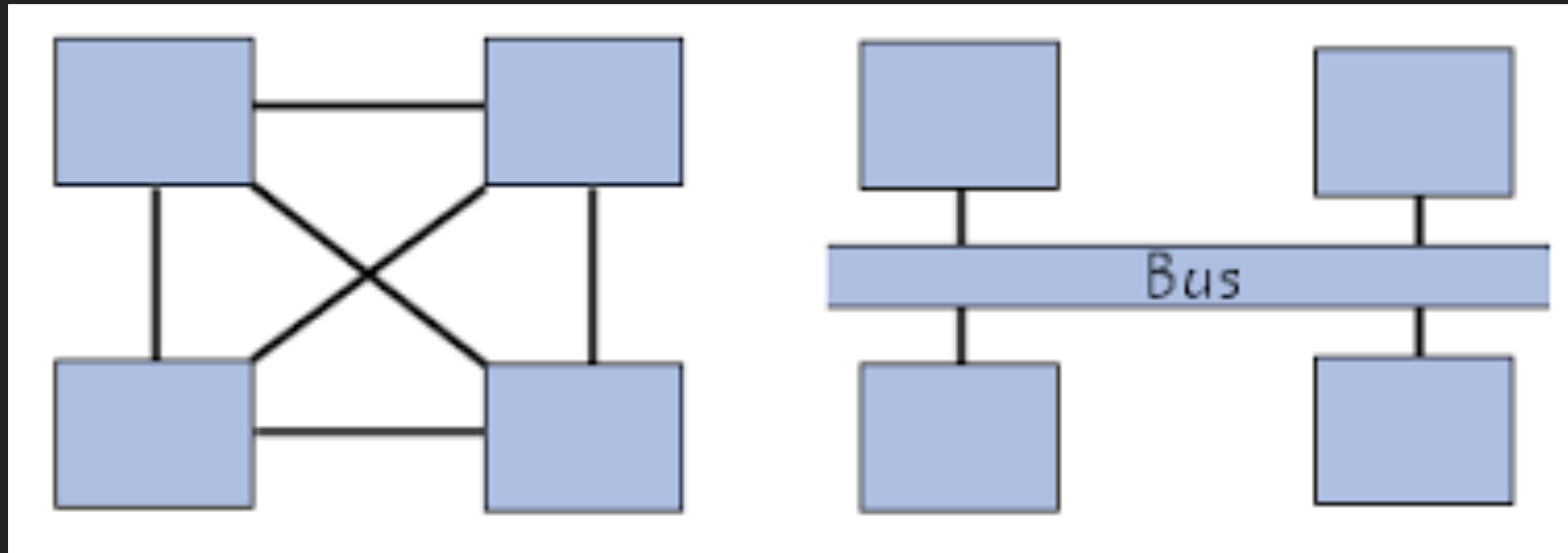
$$debit = \frac{\text{TailleMM}}{T_{cycle}} = \frac{16}{23 \times 10^{-9}} = 695652174 \text{ bits/s} = 86956522 \text{ o/s} = 82,92 \text{ Mo/s}$$

LES UNITÉS D'ENTRÉES/SORTIES

- ▶ Les unités d'Entrées/Sorties (E/S, Input/Output ou I/O en anglais) sont des éléments qui permettent de transférer les informations entre l'unité de centrale et les unités périphériques.
- ▶ Parmi les unités d'E/S les plus courantes le bus DMA (Direct Memory Access), l'accès direct à la mémoire qui s'occupe de gérer toutes les opérations d'E/S.
- ▶ Les unités périphériques se répartissent en deux classes:
 - ▶ Les unités qui permettent à l'ordinateur d'échanger les données avec l'extérieur (écran, clavier, imprimante, modem...)
 - ▶ Les mémoires auxiliaires (ou mémoire de masse) (disques, bandes, mémoires magnétiques...) qui permettent de stocker de façon permanente beaucoup d'informations à moindre coût.

- ▶ Les informations échangées entre composants de l'ordinateur circulent sur des *bus*.
- ▶ Un bus est un ensemble *de fils (conducteurs) parallèles* qui assurent la transmission (transport) du même type d'information.
- ▶ Chaque *fils* transporte un *bit* (signal).

- ▶ Objectif: réduire le nombre de voies nécessaires à la communication des différents composants.



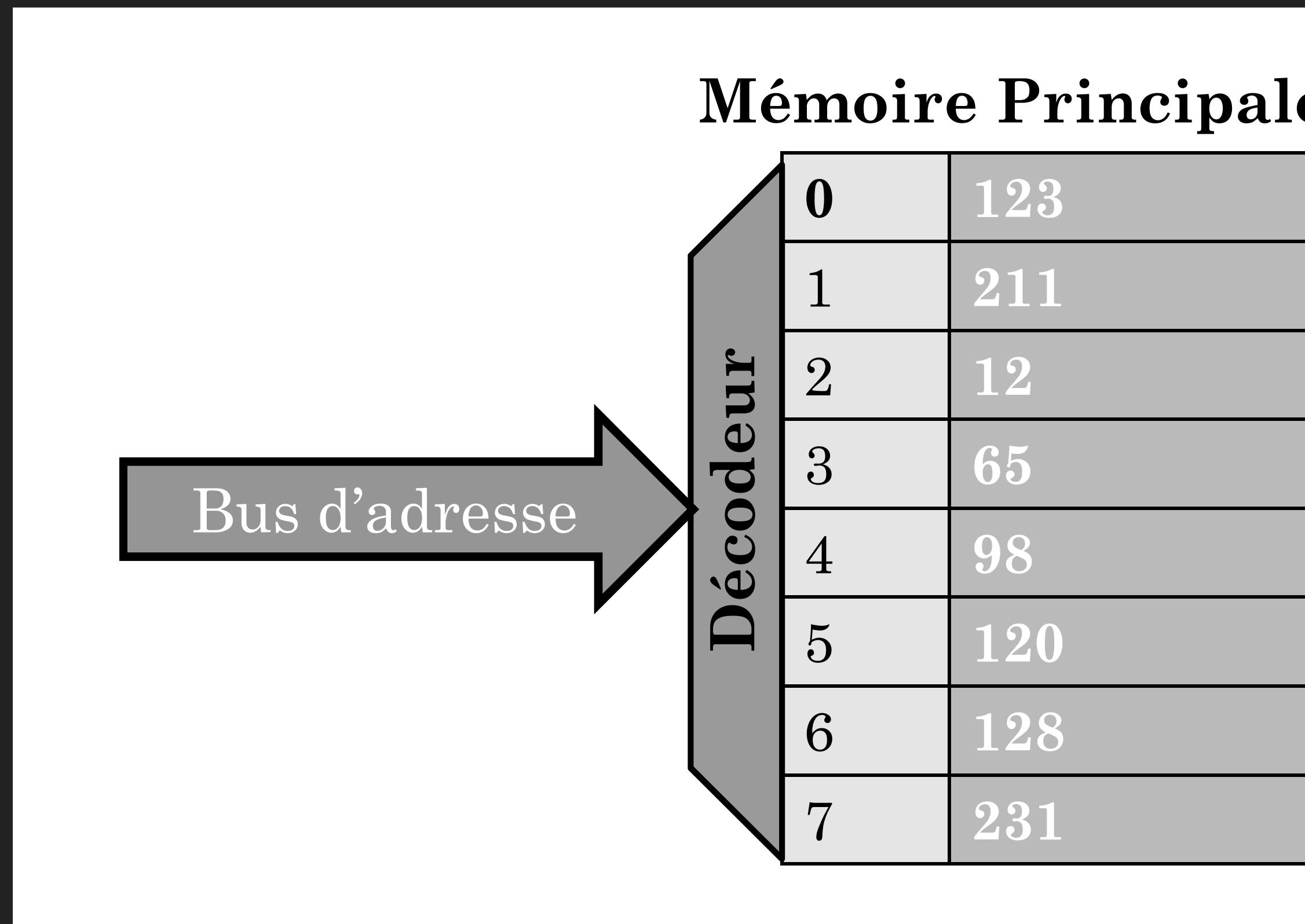
- ▶ Un bus est caractérisé par sa largeur (taille), c'est-à-dire le nombre de lignes physiques sur lesquelles les données sont envoyées = nombre de bits transmis en parallèle.
- ▶ Dans un ordinateur, on trouve trois types de bus véhiculant en parallèle trois types d'informations: données, adresses et commandes:
 - ▶ Bus de données
 - ▶ Bus d'adresse
 - ▶ Bus de commande

- ▶ **Bus de données:**
- ▶ C'est un bus bidirectionnel, permettant de transférer l'information entre les différents composants. Le processeur l'utilise pour lire ou écrire des données.
- ▶ La largeur du bus de données détermine la taille des mots mémoire auxquels on peut accéder.
- ▶ Exemple 1: Si le bus de données est de largeur de 16 fils (bits, conducteur), la taille des mots mémoire sont de 16 bits.
- ▶ Exemple 2: Si le bus de données est de largeur de 8 fils, et que chaque instruction a 16 bits, le processeur doit accéder deux fois à la mémoire par cycle d'instruction.

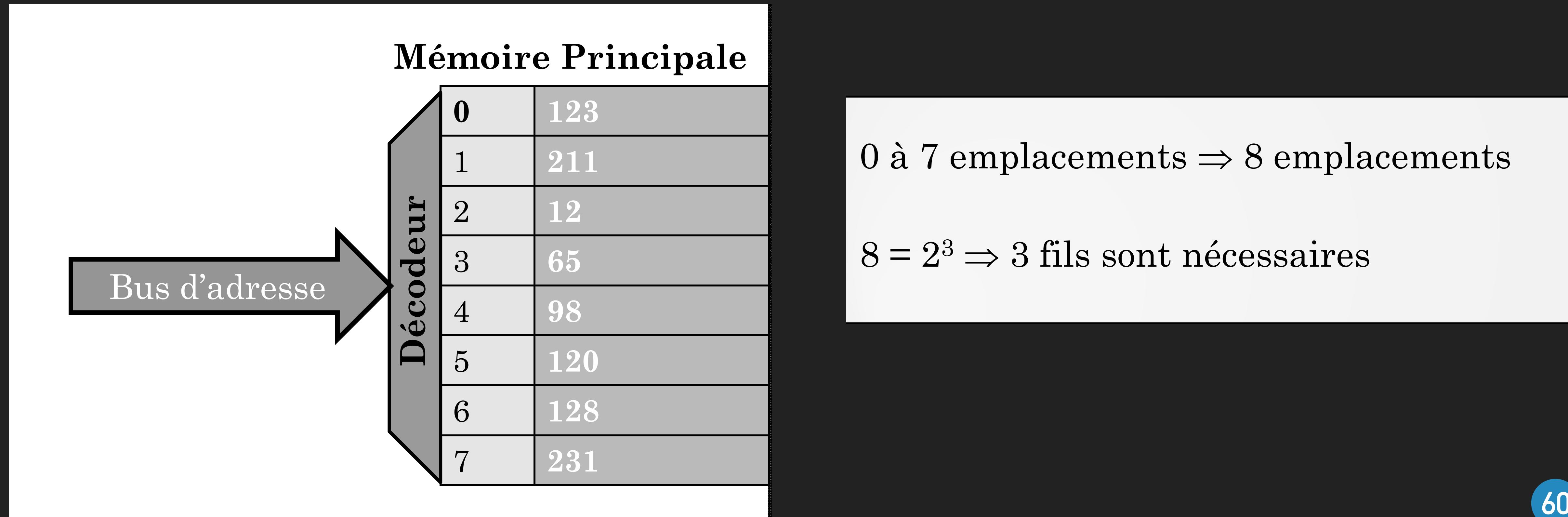
- ▶ **Bus d'adresse:**
- ▶ C'est un bus unidirectionnel, seul le processeur envoie des adresses des cellules mémoire et des entrées/sorties aux quelles il veut accéder. Il sert pour désigner la source ou la destination des données du bus de données.
- ▶ La largeur du bus d'adresse détermine la taille de la mémoire (le nombre de mots mémoires).
- ▶ Si il contient de N fils, on utilise des adresses de N bits et donc la mémoire peut posséder au maximum 2^N adresses (emplacements) différents de 0 à 2^N-1

- ▶ **Bus d'adresse:**
- ▶ La présence d'un "décodeur d'adresses" est nécessaire du fait de la multiplication des périphériques autour du processeur.
- ▶ Le processeur peut communiquer sur le bus de données avec les différentes mémoires et les différents périphériques. Pour éviter les conflits on attribue à chaque périphérique une zone d'adresse et une fonction "décodage d'adresse" chargée de "mettre" le processeur en communication avec le bon périphérique.

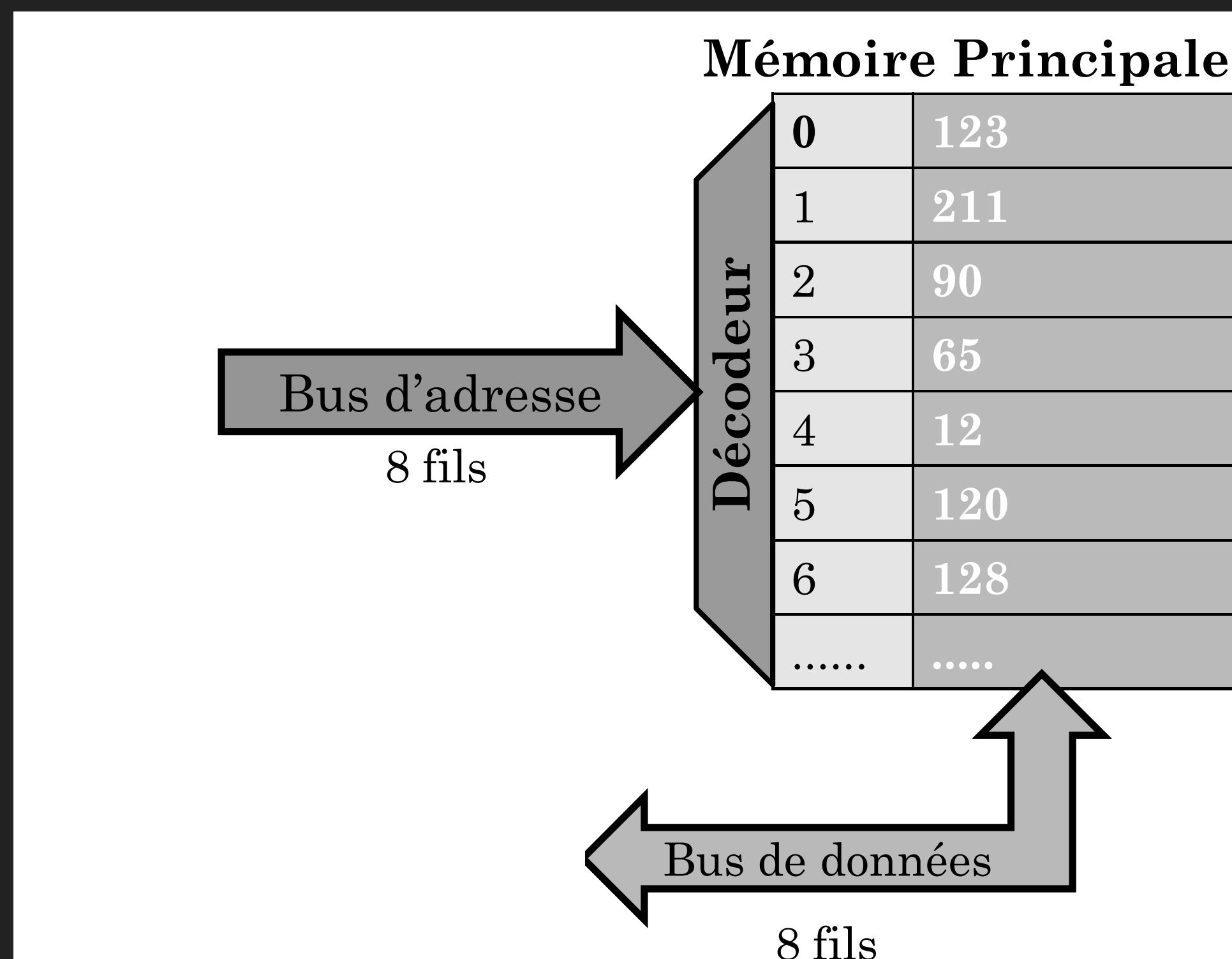
- ▶ **Bus d'adresse:**
- ▶ Exemple: Quel est le nombre minimal des fils nécessaire pour adresser cette mémoire?



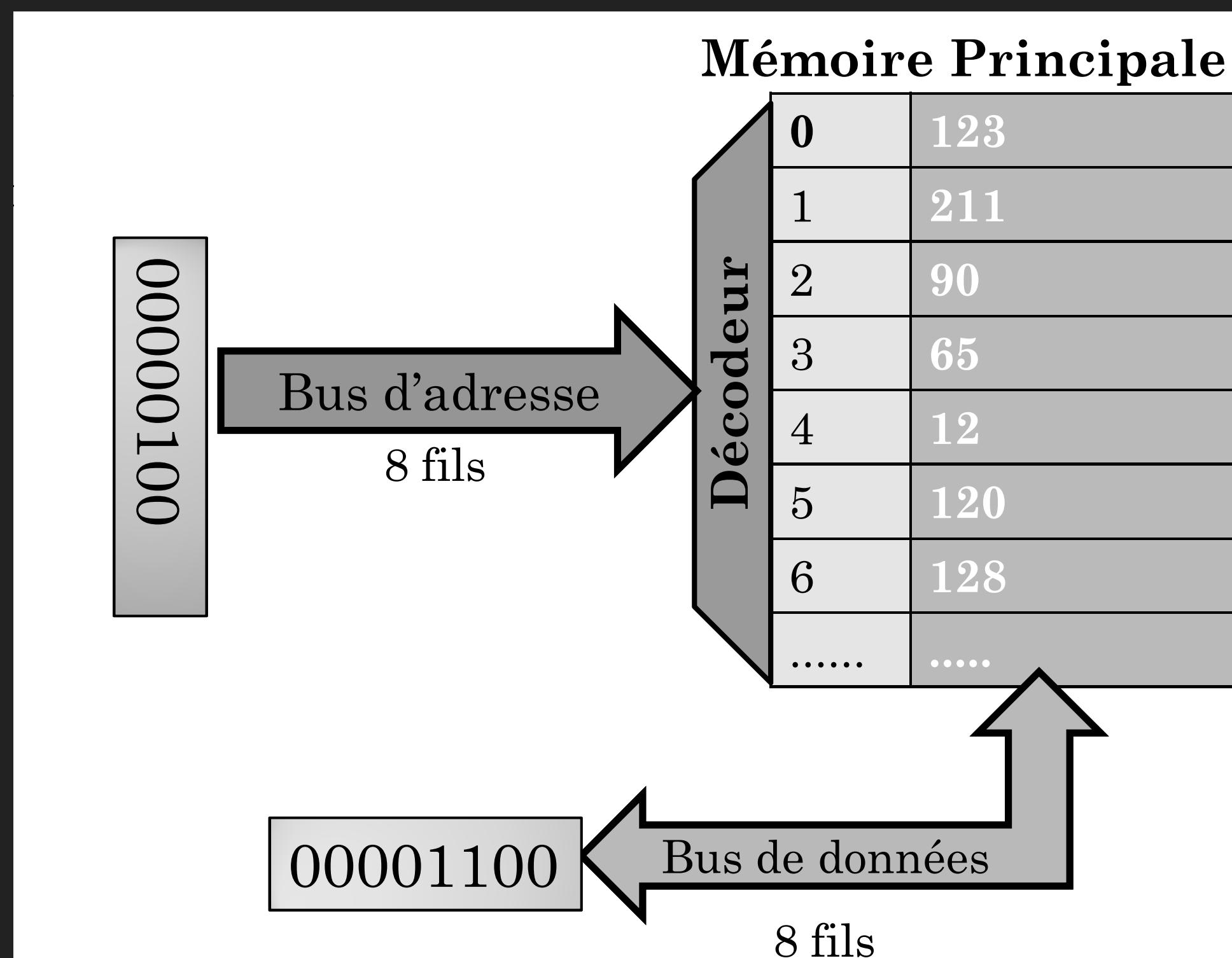
- ▶ **Bus d'adresse:**
- ▶ Exemple: Quel est le nombre minimal des fils nécessaire pour adresser cette mémoire?



- ▶ **Bus d'adresse:**
- ▶ Exemple: Supposons que le processeur veut lire le 5 eme emplacement, quel sera le contenu du bus d'adresse ainsi que du bus de données?

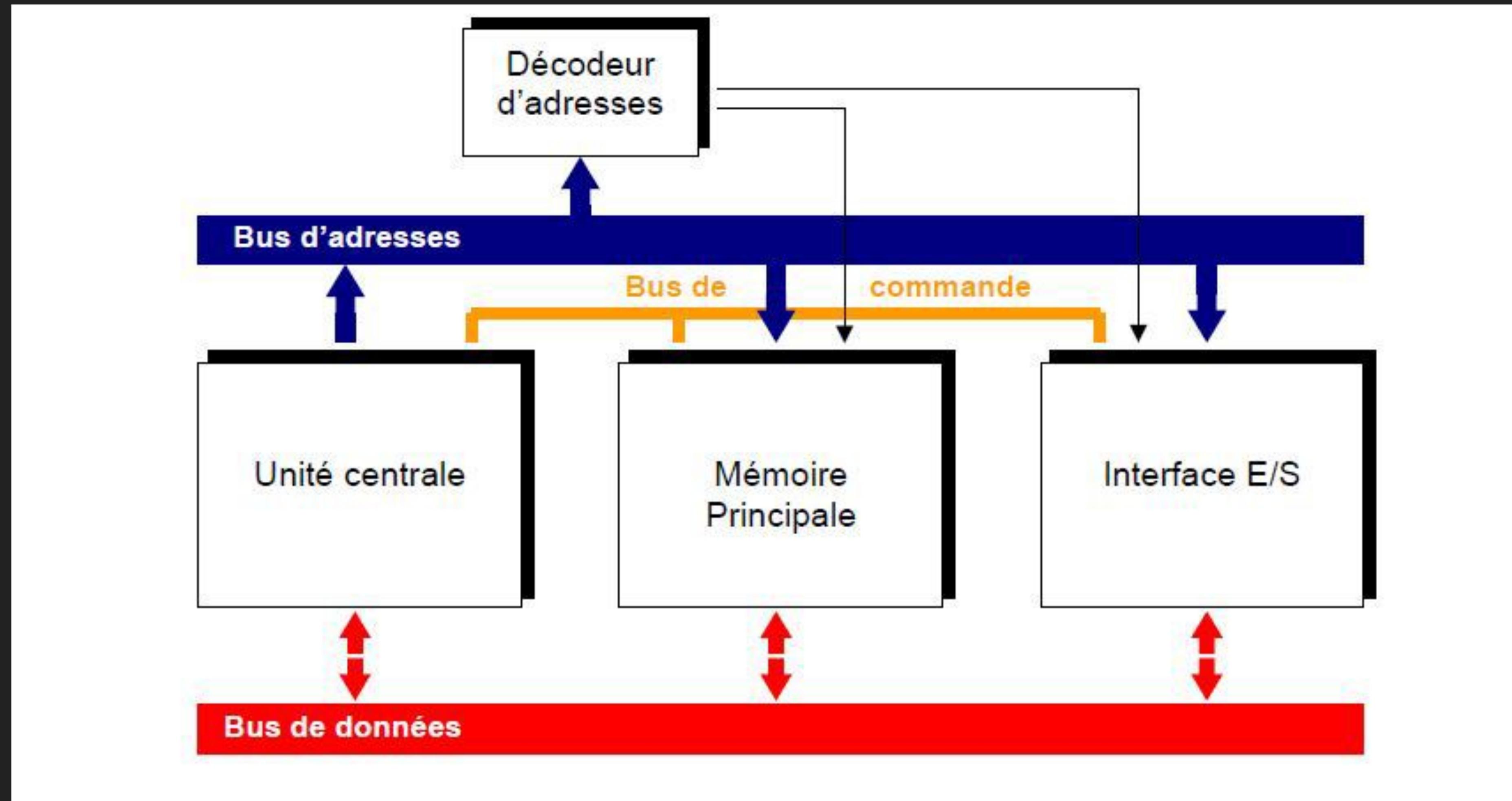


- ▶ **Bus d'adresse:**
- ▶ Exemple: Supposons que le processeur veut lire le 5 eme emplacement, quel sera le contenu du bus d'adresse ainsi que du bus de données?



- ▶ **Bus de commande:**
- ▶ Le bus de commande (ou de contrôle) sert à contrôler l'accès et l'utilisation des bus de données et d'adresses. Ces bus étant partagées par tous les composants, il doit exister un moyen pour contrôler leur utilisation.
- ▶ C'est par ce bus que le processeur indique la nature des opérations à effectuer.

► Schéma de l'architecture de Von Neumann



QUESTIONS?