

# Architecture des Ordinateurs (AO)

Akli ABBAS

Université Akli Mohand Oulhadj - Bouira  
Département d'Informatique  
2<sup>ème</sup> année - Licence informatique  
2019 - 2020

Email: abbasakli@gmail.com

**Disponible sur:**

<https://sites.google.com/a/esi.dz/a-abbas/>

- ❶ **Chapitre 1 : Organisation générale de l'unité centrale d'un Ordinateur**
  - Généralités sur l'Ordinateur
  - Architecture de Base :  
Le modèle de Harvard et de Von Neumann, Processeur, Mémoire et Bus
- ❷ **Chapitre 2 : Architecture Interne des Processeurs**
  - Introduction
  - Les Registres
  - Unité Arithmétique et Logique (UAL)
  - Unité de Commande (U.C)
  - Jeu d'instruction
  - Mode d'adressage
  - Étapes d'exécution d'une instruction
- ❸ **Chapitre 3 : Étude des cas : Processeur 80x86**
- ❹ **Chapitre 4 : Architectures des processeurs récents**

# Objectifs

- Comprendre l'**architecture** d'une machine von newman.
- Comprendre **les étapes** de déroulement de l'exécution d'une **instruction**.
- Comprendre le principe des différents **modes d'adressage**.

## ① Représentation des nombres

- Système de numérique (base 10, base 2, base 8, Base 16)
- Le système binaire
- Organisation des donnée (Bit, octets, Mot)
- Système hexadécimal
- Opération arithmétique sur le bit
- Opération logique sur les bit

## ② Algèbre de Boole

- Fonction booléenne et table de vérité
- Manipulation algébrique et expression booléenne
- Formes canonique
- Simplification des fonctions booléenne
- ...

# Organisation générale de l'unité centrale d'un Ordinateur

- **Généralités sur l'Ordinateur**
- **Architecture de Base :**
  - Le modèle de Harvard et de Von Neumann, Processeur, Mémoire et Bus

## Qu'est ce qu'un programme ?

- Un programme est un ensemble d'instructions exécutées dans un ordre bien déterminé.
- Un programme est exécuté par un processeur (machine).
- Un programme est généralement écrit dans un langage évolué (Pascal, C, VB, Java, etc.).
- Les instructions qui constituent un programme peuvent être classifiées en 4 catégories :
  - **Les Instructions d'affectations** : permet de faire le transfert des données
  - **Les instructions arithmétiques et logiques.**
  - **Les Instructions de branchement** (conditionnelle et inconditionnelle)
  - **Les Instructions d'entrées sorties.**

- Un **programme** écrit dans un **langage de haut niveau** (Pascal, C...) doit être traduit dans le **langage de la machine** pour d'être exécuté par le processeur.

- **Langage machine :**

Est la suite de bits qui est interprétée (compréhensible) par le processeur.

C'est le seul langage que le processeur puisse traiter. Il est composé d'instructions et de données **codées en binaire**. (Ex. 10110000 01100001)

**Problème :** le langage machine est difficile à comprendre par l'humain.

**Idée** : trouver un langage compréhensible par l'homme qui sera ensuite converti en langage machine  $\mapsto$  **Langage d'assemblage**

**langage assembleur** : exprimer les instructions élémentaires (binaire) de façon symbolique. Il représente le langage machine sous une forme lisible par un humain. (Ex. mov 61)

**Une instruction assembleur = une instruction machine**

**Exemple** : 10110000 01100001 (instruction machine) = mov 61 (instruction assembleur)



## Comment s'exécute un programme dans la machine ?

- Pour comprendre le mécanisme d'exécution d'un programme  $\Rightarrow$  il faut comprendre le mécanisme de l'exécution d'une instruction .
- Pour comprendre le mécanisme de l'exécution d'une instruction  $\Rightarrow$  il faut connaître l'architecture de la machine (processeur) sur la quelle va s'exécuter cette instruction.

## **que veut-on dire par architecture ?**

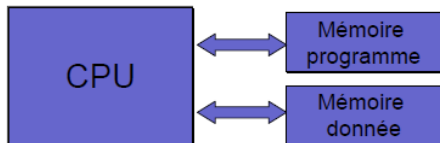
L'architecture d'un système représente l'organisation de ses différentes composantes et de leurs interconnexions.

Deux types d'architecture d'ordinateur :

- ① Le Modèle de Harvard
- ② Le Modèle de Von Neumann

## Le Modèle de Harvard (1944)

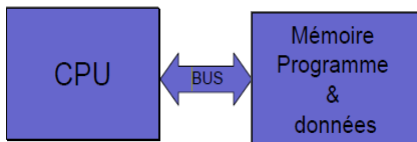
L'architecture de type **Harvard** est une conception qui sépare physiquement la **mémoire de données** et la **mémoire programme**. L'accès à chacune des deux mémoires s'effectue via deux bus distincts.



- ❶ Séparation des mémoires programme et données
  - *Un bus de données programme,*
  - *Un bus de données pour les données,*
  - *Un bus d'adresse programme,*
  - *Un bus d'adresse pour les données.*
- ❷ Meilleure utilisation du CPU :
- ❸ Chargement du programme et des données en parallèle

## Le Modèle de Von Neumann (1946)

L'architecture **Von Neumann** est la base des architectures des ordinateurs et elle est composée : d'un **processeur (CPU)**, d'une **mémoire principale** et d'un **dispositifs d'entrées sorties** pour communiquer avec l'extérieur.



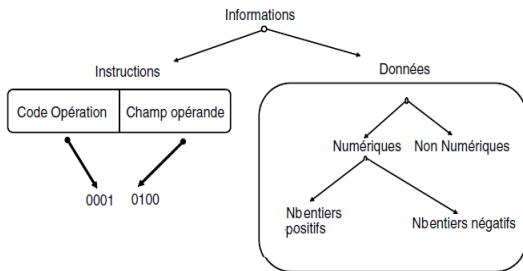
- ① Un seul chemin (bus) d'accès à la mémoire ;
  - Un bus de données (programme et données),
  - Un bus d'adresse (programme et données)
- ② Limite l'accès à la mémoire.
- ③ Architecture des processeurs d'usage général ;

## Le processeur

- Le processeur, parfois appelé **CPU** (Central Processing Unit) est un circuit électronique qui est divisé en deux parties :
  - ① **Unité de commande (U.C)** est responsable de la lecture en mémoire et du décodage des instructions machine ;
  - ② **Unité de traitement**, appelée aussi **Unité Arithmétique et Logique (U.A.L.)**, exécute les instructions machine qui manipulent les données.
- Pour chaque instruction machine, le processeur effectue schématiquement les opérations suivantes :
  - ① lire en mémoire l'instruction à exécuter ;
  - ② effectuer le traitement correspondant ;
  - ③ passer à l'instruction suivante.

## La mémoire principale

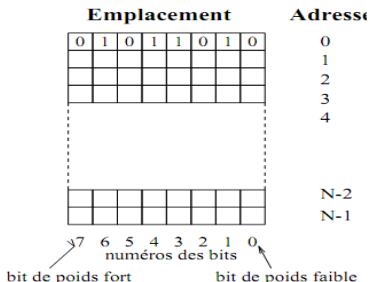
- La mémoire principale (MP) permet de stocker les informations (instructions et données).
- Les différentes informations que l'on trouve dans la mémoire principale :



- Toute information manipulée par le processeur est sous forme **binaire**.

## Structure de la mémoire principale

- La mémoire est divisée en emplacements de taille fixe (en général 1 octet = 8 bits) utilisés pour stocker instructions et données.



- Dans une mémoire de taille  $N$ , on a  $N$  emplacements mémoires, numérotés de 0 à  $N-1$ . Chaque emplacement est repéré par son numéro, appelé **adresse**.

## Structure de la mémoire principale (2)

- La capacité (taille) de la mémoire est le nombre d'emplacements, exprimé en général en kilo-octets ou en méga-octets, voire plus.

---

1 K (Kilo)	$2^{10}$	= 1024
1 M (Méga)	$2^{20}$	= 1 048 576
1 G (Giga)	$2^{30}$	= 1 073 741 824
1 T (Téra)	$2^{40}$	= 1 099 511 627 776

---

- Deux opérations sont possibles sur la mémoire :
  - Écriture** : le processeur fournit une valeur et une adresse, et la mémoire range la valeur à l'emplacement indiqué par l'adresse ;
  - Lecture** : le processeur demande à la mémoire la valeur contenue à l'emplacement dont il indique l'adresse. Le contenu de l'emplacement lu reste inchangé.



## Unité de transfere entre la mémoire et le processeur

- Les opérations de lecture et d'écriture portent en général sur plusieurs octets contigus en mémoire : un **mot** mémoire.
- La taille d'un mot mémoire dépend du type de processeur :
  - 1 octet dans les processeurs 8 bits (ex. Motorola 6502) ;
  - 2 octets dans les processeurs 16 bits (ex. Intel 8086) ;
  - 4 octets dans les processeurs 32 bits (ex. Intel 80486 ou Motorola 68030).
  - 8 octets dans les processeurs 64 bits (ex. Les Itanium d'Intel, AMD64 de AMD ou Intel 64 d'Intel)

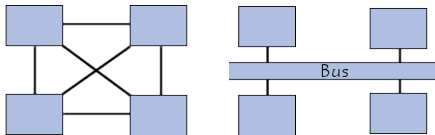
## Les différents type de mémoire

RAM = mémoire vive = mémoire système = mémoire volatile

- **rapidité d'accès** : essentielle pour fournir rapidement les données au processeur.
- **volatilité** : les données sont perdues dès que l'ordinateur cesse d'être alimenté en électricité.
- **SRAM (Static RAM)** :
  - ① utilise des bascules pour mémoriser l'info ;
  - ② très rapide mais coûteuse en composants ;
  - ③ temps d'accès : de l'ordre de 1 ns ;
  - ④ utilisée pour le cache, par exemple.
- **DRAM (Dynamic RAM)** :
  - ① utilise des charges de condensateurs (plus économique) ;
  - ② moins rapide que la SRAM, nécessite des rafraîchissements ;
  - ③ temps d'accès : de l'ordre de 10 ns ;
  - ④ utilisée pour la mémoire principale.

## Bus

- Les informations échangées entre composants de l'ordinateur circulent sur des **bus**.
- **Bus** : est un ensemble de  $n$  fils parallèles servant à relier un ou plusieurs composants de l'ordinateur et à transporter  $n$  signaux binaires.
- **Objectif** : réduire le nombre de voies nécessaires à la communication des différents composants.



## Bus

### Caractéristiques :

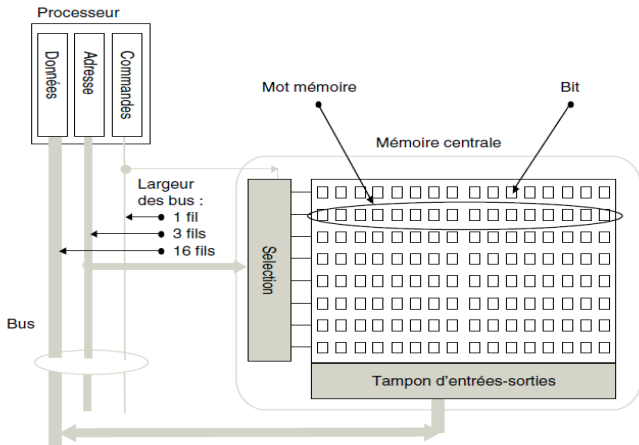
- **Largeur** : nombre de lignes physiques sur lesquelles les données sont envoyées = nombre de bits transmis en parallèle.
- **Fréquence (en Hz)** : nombre de paquets de données envoyés ou reçus par seconde.
- **Débit maximal** (ou taux de transfert maximal, bande passante) : quantité de données transportées par unité de temps :  
 $\text{Débit maximal (en Mo/s)} = \text{largeur du bus (en octets)} \times \text{fréquence (en Hz)}.$

### Exemple

Largeur 2 octets et fréquence 133 MHz  $\Rightarrow$  266 Mo/s

## Bus

- **Type** : Il existe trois types de bus : de **données**, d'**adresses** et de **commande** (ou de contrôle).



## Bus de données

- Le bus de données est un bus bidirectionnel( $\longleftrightarrow$ ) permettant de transférer l'information entre les différents composants (Ex. entre la mémoire et le CPU).
- L'information peut être :
  - Une instruction à exécuter par le processeur (mémoire  $\longrightarrow$  CPU)
  - Donnée stockée en mémoire (mémoire  $\longrightarrow$  CPU)
  - Donnée produite par un traitement du processeur (CPU  $\longrightarrow$  mémoire).

La largeur de bus de données détermine la taille des mots mémoires auxquels on peut accéder.

### Exemple

Le bus de données a une largeur de 16 fils  $\Rightarrow$  les mots mémoires accessibles (ou modifiables) ont 16 bits.

## Bus d'adresses

- Le bus d'adresse est un bus unidirectionnel ( $\rightarrow$ ) ; seul le processeur envoie des adresses.
- La largeur de bus d'adresses détermine la taille de la mémoire (le nombre d'emplacements). S'il est composé de  $n$  fils  $\Rightarrow$  On utilise donc des adresses de  $n$  bits  $\Rightarrow$  La mémoire peut posséder au maximum  $2^n$  emplacements (adresses 0 à  $2^n - 1$ ).

## Bus d'adresses

### Exercice

Quel est le nombre minimal des fils nécessaire pour adresser cette mémoire ?

Bus d'adresse

### Mémoire Principale

Décodeur	0	123
	1	211
	2	12
	3	65
	4	98
	5	120
	6	128
	7	231

### Solution

0 à 7 emplacements  $\Rightarrow$  8 emplacements

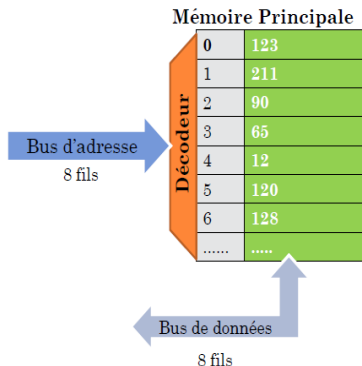
$8 = 2^3 \Rightarrow$  3 fils sont nécessaires



## Bus d'adresses

### Exercice

Supposant que le processeur veut lire le 5<sup>ème</sup> emplacement (12), Quel sera le contenu du bus d'adresse ainsi que le bus de données ?



### Solution

Bus d'adresses=00000100

Bus de données=00001100

## Bus de commande

- Le bus de commande (ou de contrôle) est un bus unidirectionnel ( $\rightarrow$ ). Seul le processeur envoie des commandes pour contrôler le fonctionnement des autres composants.
- C'est par ce bus que le microprocesseur indique la nature des opérations qu'il veut effectuer.
- Dans notre exemple il a une largeur d'un fil et donc le microprocesseur ne peut passer que deux commandes (la lecture et l'écriture)