

# AES. Architectures et systèmes

## I/ Histoire de l'architecture matérielle

### 1/ Introduction

Le mot *informatique* a été créé par Philippe Dreyfus en 1962 avec les mots *information* et *automatique*.

**L'informatique** recouvre tout ce qui est traité de manière automatique par une machine comme un ordinateur, un smartphone, un objet connecté. Elle **recueille** des **données**, les **organise**, les **stocke**, les **traite** pour obtenir de l'information, elle les **transmet**.

Dans un système informatique, on trouve des composants physiques, c'est la partie matérielle, le hardware, et des programmes ou logiciels, le software, qui s'exécutent sur cette partie matérielle.

L'efficacité des programmes dépend de la qualité de leur conception mais également des caractéristiques du matériel. Il est donc important de connaître le fonctionnement interne d'une machine afin d'utiliser les algorithmes les plus adaptés.

### 2/ Histoire

La composition d'un ordinateur a beaucoup évolué avec le temps. On peut distinguer plusieurs générations d'ordinateurs. Chacune d'entre elle peut être caractérisé par des critères de volume, de masse, de coût, de langage tous liés aux progrès de la technologie.

Dans les années 1940, les premiers ordinateurs sont lourds, volumineux et consomment énormément d'énergie (rejetée sous forme de chaleur). Seuls les militaires ont les moyens d'en construire et de les utiliser, on peut citer par exemple l'ENIAC aux Etats-Unis, le Colossus en Angleterre et la série Z1, Z2 et Z3 en Allemagne.

- Le Z3 allemand (de Konrad Zuse, son inventeur) est achevé en 1941, il est programmable, entièrement automatique et fonctionne avec des relais électromécaniques : il pouvait effectuer trois additions en moins d'une seconde et une multiplication en trois secondes en utilisant les nombres flottants. Il semblerait qu'il soit le premier ordinateur au monde.  
Heureusement que le gouvernement allemand de l'époque n'a pas réellement soutenu les projets de Zuse, les qualifiant « d'intéressant mais non essentiels pour l'effort de guerre ».
- Le Colossus anglais est opérationnel en 1943 sous l'impulsion d'Alan Turing, c'est grâce à lui que les messages -réputés inviolables- des machines Enigma nazies seront déchiffrés. Tout comme l'ENIAC américain, il utilise des tubes à vide au lieu de relais électromécaniques.



Le Z3 allemand



L'ENIAC

Quelques chiffres pour l'ENIAC : 30 tonnes, 30 mètres cubes, 18000 tubes et environ 100 fois plus rapide que le Z3.  
On n'est pas prêt d'en avoir un dans sa chambre 😊.

Le premier **compilateur** est conçu en 1951 par Grace Hopper, informaticienne dans l'armée américaine. Elle sera aussi à l'origine du langage Fortran en 1954 et du COBOL en 1959.

**Compilateur** : application traduisant un programme écrit dans un langage que l'humain peut comprendre en langage machine.

C'est aussi dans les années 50 que le **transistor** se développe : il est beaucoup plus petit, très bon marché, fiable et plus rapide que les tubes électroniques. Mécaniquement, la puissance de calcul augmente, le coût et la taille des ordinateurs diminuent, les grandes universités (américaines surtout) commencent à s'en équiper.



Un transistor



Un tube électronique

**Transistor** : interrupteur électronique capable également d'amplifier des signaux électriques.

Une petite vidéo ici : <https://www.youtube.com/watch?v=zjY17c7WNNw>

L'invention du **circuit intégré** est un nouveau tournant dans les années 60. Ces circuits, beaucoup plus fins, contiennent des millions de transistors gravés dans du silicium. Désormais, des ordinateurs peuvent être embarqués dans les missions Apollo de la NASA. L'AGC, qui pèse une trentaine de kilos, avec clavier, écran et interface utilisateur gère la navigation et le pilotage de la fusée et du module lunaire. Sa puissance est l'équivalent de celle d'une calculatrice actuelle de collège.



Un circuit intégré

En 1971, un ensemble de circuits intégrés constituent le **processeur**, le cœur de l'ordinateur. La démocratisation de l'informatique va commencer, notamment avec l'apparition du langage C en 1972. Vers la fin des années 70, les premiers ordinateurs personnels pour le grand public vont faire leur apparition.



Apple II, 1977



TRS-80, 1977

Plus de photos sur la lignée des TRS-80 ici : <https://vistapointe.net/radio-shack-trs-80.html>

L'histoire d'Apple et sa lutte contre IBM ici : <https://www.youtube.com/watch?v=b1J8cMUtGuc>

Auteur : rétro Québec, durée : 13 min 11 sec

Les années 2000 voient l'arrivée de smartphones fonctionnant avec un système d'exploitation (comme Android) et des applications en plus de leur fonction de téléphonie. Les premières photos numériques de qualité datent de 2010.

En savoir plus sur Grace Hopper : <https://www.linternaute.fr/science/biographies/1778104-grace-hopper-biographie-courte-dates-citations/>

En savoir plus sur l'histoire de l'ordinateur : <https://jeromeabel.net/files/pdf/OrdinateurHistorique.pdf>

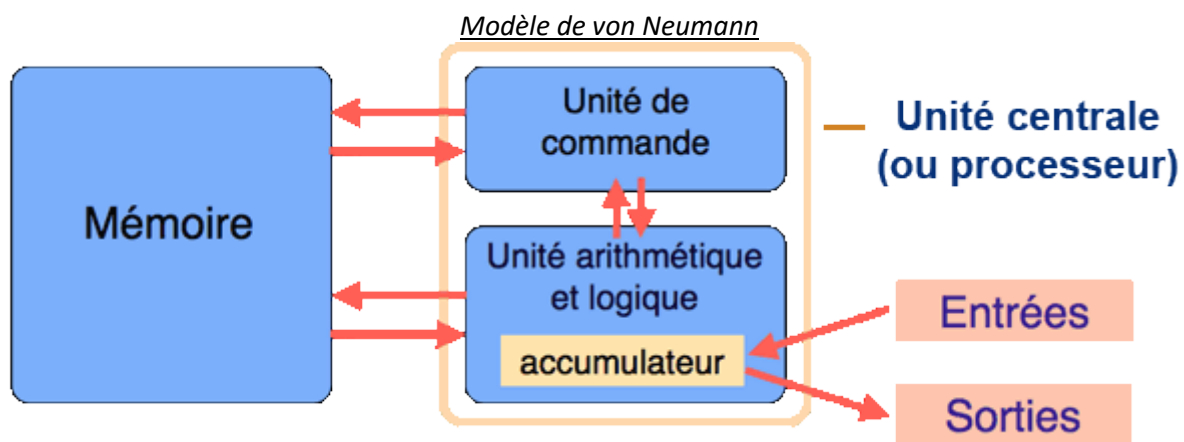
## II/ Architecture de von Neumann

### 1/ Descriptif

L'architecture des ordinateurs actuels repose sur le modèle de John von Neumann qui travaille comme consultant dans le projet ENIAC. L'architecture a gardé son nom et repose sur l'idée de programme enregistré. La mémoire de l'ordinateur, dans laquelle était stocké des données, devaient aussi stocker des programmes.

Un ordinateur est constitué de quatre parties distinctes :

- La **mémoire** dont on distingue deux types : celle de type **RAM** (en anglais *Random Access Memory*) et celle de type **ROM** (en anglais *Read Only Memory*) accessible seulement en lecture.
- Les **bus** qui sont des fils électriques reliant les différents composants d'un ordinateur.
- Les **entrées-sorties** (E/E ou en anglais I/O pour *Input/Output*) pour échanger avec l'extérieur.
- Le **processeur** ou **CPU** (en anglais *Central Processing Unit*) constitué de l'**unité de commande** (gère l'exécution des instructions machines) et de l'**unité arithmétique et logique** appelé UAL (l'accumulateur est une mémoire extrêmement rapide utilisée pour stocker des calculs intermédiaires).



Source : <https://interstices.info/>

En savoir plus sur John von Neumann ici : [https://www.youtube.com/watch?v=c9pL\\_3tTW2c](https://www.youtube.com/watch?v=c9pL_3tTW2c)  
Documentaire très complet d'Arte, durée : 56 minutes.

### 2/ Mémoires

On distingue plusieurs types de mémoires : la **ROM**, la **RAM**, les **mémoires de stockage** et les **registres**.

#### Mémoire de type ROM

Ce type de mémoire permet de stocker des données en l'absence de courant électrique, il s'agit de la **ROM** (*Read Only Memory*, dont la traduction littérale est *mémoire en lecture seule*) appelée **mémoire morte**, parfois *mémoire non volatile* car elle ne s'efface pas lors de la mise hors tension du système.

Elle permet notamment de **conserver les données nécessaires au démarrage de l'ordinateur**. En effet, ces informations ne peuvent être stockées sur le disque dur étant donné que les paramètres du disque (essentiels à son initialisation) font partie de ces données vitales à l'amorçage.

**A savoir** : la mémoire ROM est une mémoire non volatile et accessible en lecture seule qui comporte le nécessaire pour faire démarrer le système.

## Aller plus loin :

Différentes mémoires de type *ROM* contiennent des données indispensables au démarrage, c'est-à-dire :

- Le **BIOS** est un programme permettant de piloter les interfaces d'entrée-sortie principales du système, d'où le nom de *BIOS ROM* donné parfois à la puce de mémoire morte de la carte-mère qui l'héberge.
- Le **chargeur d'amorce**: un programme permettant de charger le système d'exploitation en mémoire (vive) et de le lancer. Celui-ci cherche généralement le système d'exploitation sur le disque dur.
- Le **Setup CMOS**, c'est l'écran disponible à l'allumage de l'ordinateur permettant de modifier les paramètres du système (souvent appelé *BIOS* à tort...).
- Le **Power-On Self Test (POST)**, programme exécuté automatiquement à l'amorçage du système permettant de faire un test du système (c'est pour cela par exemple que l'on voit le système "compter" la RAM au démarrage).

Etant donné que les ROM sont beaucoup plus lentes que les mémoires de types RAM (une ROM a un temps d'accès de l'ordre de 150 ns tandis qu'une mémoire de type SDRAM a un temps d'accès d'environ 10 ns), les instructions contenues dans la ROM sont parfois copiées en RAM au démarrage, on parle alors de *shadowing* (en français cela pourrait se traduire par *ombrage*, mais on parle généralement de *mémoire fantôme*).

Source : <https://www.commentcamarche.net/contents/765-la-memoire-morte-rom#la-memoire-morte-rom>

## Mémoire de « masse »

Elle sert à stocker à **long terme** des **grandes quantités d'informations**. Les technologies les plus courantes de mémoires de masse sont *électromécaniques* (disques durs – HDD) ou à *semi-conducteurs* (SSD, clefs USB, ...), elles visent à obtenir une capacité de stockage élevée à faible coût et ont généralement une **vitesse inférieure** aux autres mémoires. Ce type de mémoire est bien sûr **non volatile** et donc n'a pas besoin d'alimentation électrique pour contenir des informations.

L'accès en lecture et écriture n'est pas très rapide.

Ordres de grandeur :

- *capacité* : jusqu'à 10 To (HDD), jusqu'à 1 Go (SSD)
- *vitesse* : jusqu'à 500 Mo/s (SSD)



Mémoire de type SSD (en anglais, Solid State Drive)



Mémoire de type HDD (en anglais Hard Disk Drive)

**A savoir :** les mémoires de masse ont une très grande capacité et servent à stocker des données. Accessibles en lecture et écriture, elles ont un temps d'accès long et ne peuvent pas être utilisées directement par le processeur.

En savoir plus sur le fonctionnement d'un disque dur ici : <https://www.youtube.com/watch?v=MC6CZmHgc8I>

Auteur : Imdinfo, Durée : 4 min 36 sec

## Mémoire centrale (type RAM)

La **RAM** (en anglais *Random Access Memory*) est une mémoire volatile. Elle est accessible en lecture et écriture. Elle stocke des données et les programmes exécutés par le processeur.

Elle est organisée en cellules qui contiennent chacune une donnée ou une instruction repérée par un nombre entier écrit en base hexadécimale : une **adresse mémoire**.

Le temps d'accès (assez rapide, autour de 10 nanosecondes) à chaque cellule est le même, on parle donc de mémoire à accès aléatoire ou à accès direct.



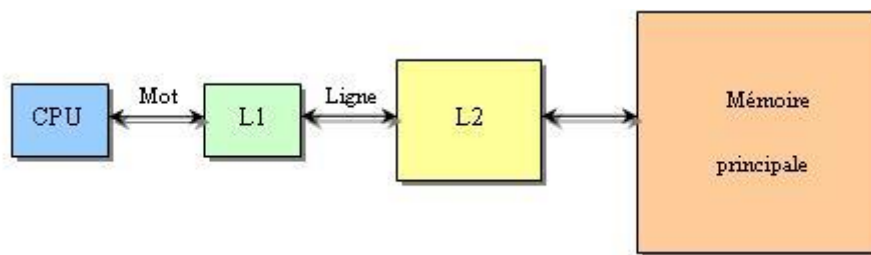
Une barrette de SDRAM

**A savoir :** la mémoire centrale est une mémoire volatile, accessible en lecture et écriture. Elle est directement reliée au processeur.

## Mémoire cache

Pour pouvoir adapter la très grande vitesse du processeur (plusieurs milliards d'opérations par seconde) à celle plus faible de la mémoire centrale (temps d'accès de 10 nanosecondes), on place entre eux une mémoire rapide, la **mémoire cache**, qui contient les instructions et données en cours d'utilisation car, la plupart du temps, les données qui viennent d'être utilisées ont une plus grande probabilité d'être réutilisées que d'autres.

Elle est actuellement organisée en deux niveaux (représentés ici par L1 et L2) :



La capacité de L1 est de quelques Ko et celle de L2 de quelques Mo.

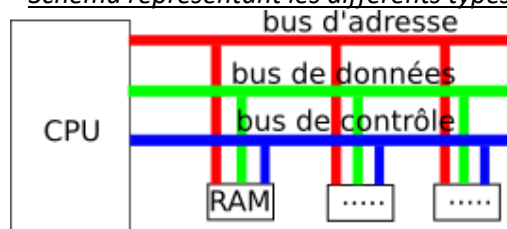
## 3/ Carte mère et bus

Les données doivent circuler entre les différentes parties d'un ordinateur, notamment entre la mémoire vive et le CPU. Le système permettant cette circulation est appelé **bus** et est assuré par la **carte mère**.

Il existe, trois grands types de bus :

- Le **bus d'adresse** permet de faire circuler des adresses (par exemple l'adresse d'une donnée à aller chercher en mémoire).
- Le **bus de données** permet de faire circuler des données.
- Le **bus de contrôle** permet de spécifier le type d'action (exemples : écriture d'une donnée en mémoire, lecture d'une donnée en mémoire).

Schéma représentant les différents types de bus





## 4/ Périphériques d'entrée et sortie

Les périphériques permettent de communiquer avec l'ordinateur.

On distingue les périphériques **d'entrée** (qui amènent des informations dans l'unité centrale) et les périphériques de **sortie** (qui proviennent de l'unité centrale sous forme d'affichage le plus souvent). Un périphérique comme une clé USB peut être à la fois en entrée et sortie.



**A savoir :** un périphérique d'entrée permet de transmettre des données à l'unité centrale. Il peut s'agir d'action d'un utilisateur (clavier, souris etc.) mais aussi de capteurs (thermomètre, balance etc.).

**A savoir :** un périphérique de sortie permet de transmettre la réponse de l'unité centrale à l'utilisateur (affichage à l'écran, impression, son etc.)

## 5/ Microprocesseur (CPU)

Le microprocesseur est le "cœur" d'un ordinateur : les instructions sont exécutées au niveau du CPU.

Il est schématiquement constitué de 3 parties :

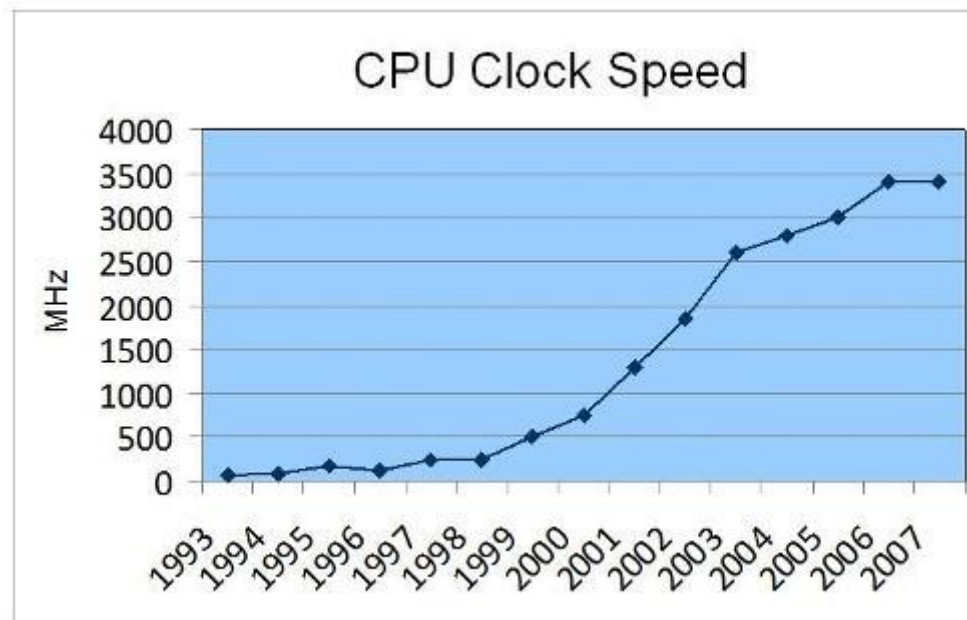
- les **registres** permettent de mémoriser de l'information (donnée ou instruction) au sein même du CPU. Leur nombre et leur taille sont variables en fonction du type de microprocesseur. Dans la suite on nommera ces registres R1, R2, R3...
- L'**unité arithmétique et logique** (UAL ou ALU en anglais) est chargée de l'exécution de tous les calculs que peut réaliser le microprocesseur.
- L'**unité de commande** permet d'exécuter les instructions (les programmes).



Un CPU

Le CPU dispose d'une horloge interne qui cadence l'accomplissement des instructions. L'unité est appelé **cycle**. Si elle est cadencée à 3 GHz, cela signifie qu'il y a trois milliards cycles par seconde. Jusqu'en 2004, la fréquence des processeurs a augmenté linéairement (puissance doublée tous les 18/24 mois selon la loi de Moore) mais stagne car la chaleur dégagée ralentit les processus et la miniaturisation a atteint ses limites.

Evolution de la cadence de l'horloge d'un CPU en fonction des années



Un article sur l'évolution de la performance des processeurs ici :

<https://sites.google.com/site/tpesurlesprocesseurs/l-historique-des-processeurs>

Un article illustrant la fin de la loi de Moore : <https://www.zdnet.fr/actualites/la-loi-de-moore-est-morte-encore-une-fois-et-pour-de-bon-39879081.htm>

Pour augmenter la puissance des ordinateurs, on augmente le nombre de processeurs (on peut en avoir plusieurs dizaines). Cela étant, il faut que les programmes prennent en compte ce fait (multithreading) et il ne faut pas oublier qu'ils doivent se partager la même mémoire centrale et les mêmes bus, le gain n'est donc pas proportionnel. Il faudra attendre une nouvelle technologie, l'**ordinateur quantique** est en test actuellement et semble prometteur.

En savoir plus sur l'ordinateur quantique ici : <https://www.youtube.com/watch?v=2aCS5mEeiwg>

Auteur : L'esprit Sorcier Officiel, Durée : 25 min 54 sec

## II/ Circuit et fonctions booléennes

Une machine n'est pas intelligente. Elle exécute ce qu'on lui demande de faire (même lancer des jeux idiots 😊). Son langage est un langage binaire et c'est un langage de programmation, plus proche du langage humain et traduit par un compilateur (pour le C par exemple) ou par un interpréteur (pour Python par exemple) qui permet de communiquer avec elle et de lui donner des instructions.

Son fonctionnement repose sur des **circuits électroniques**. Il s'agit ici de faire le **lien** entre ces circuits, le **calcul logique** et le **calcul binaire**.

Un circuit muni d'interrupteurs ou de transistor peut **laisser passer un courant** ou **pas**. On peut donc faire correspondre cet état avec des **1** et **0** ou deux valeurs booléennes **vrai** et **faux**. Avec plusieurs circuits en parallèle on peut ainsi créer des nombres en binaire.

Des petits circuits permettent de traduire des opérations simples :

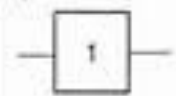

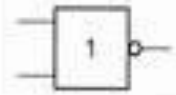

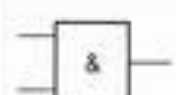



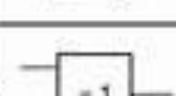


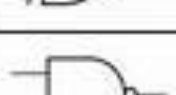
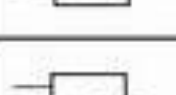
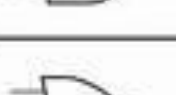
- L'**inversion d'un bit** avec une entrée et une sortie.
- Le **ET logique** et le **OU logique** avec deux entrées et une sortie.
- La **somme deux bits** avec deux entrées et deux sorties (une sortie contient la retenue).

A l'aide de ces petits circuits, on peut additionner deux nombres et trouver l'opposé d'un nombre et on en déduit les **autres opérations mathématiques**.

Des fonctions logiques de base, appelées portes logiques, permettent de réaliser fonction logique plus complexe.

**Rappel** : les concepts de **conjonction**, **disjonction** et **négation** sont les trois opérations fondamentales de l'algèbre booléenne.

Voici la représentation des portes logiques principales :

Porte OUI (YES)			entrée	sortie
			0	0
			1	1
Porte NON (NO)			entrée	sortie
			0	1
			1	0
Porte ET (AND)			entrées	sortie
			0 0	0
			0 1	0
			1 0	0
			1 1	1
Porte OU (OR)			entrées	sortie
			0 0	0
			0 1	1
			1 0	1
			1 1	1
Porte OU exclusif (XOR)			entrées	sortie
			0 0	0
			0 1	1
			1 0	1
			1 1	0
Porte NON-ET (NAND)			entrées	sortie
			0 0	1
			0 1	1
			1 0	1
			1 1	0
Porte NON-OU (NOR)			entrées	sortie
			0 0	0
			0 1	1
			1 0	1
			1 1	0

XOR, dans les années 1980, c'était ça 😊 : <https://www.youtube.com/watch?v=orwCZm9ugJ0>

Bon, ce n'est pas trop le sujet !



Exemple : un demi-additionneur sur 1 bit

Un demi-additionneur (HA) est un circuit qui génère la somme, S, et la retenue, C, résultant de l'addition de deux nombres de 1 bit, A et B.

$$\begin{array}{r} A \\ + B \\ \hline C \ S \end{array}$$

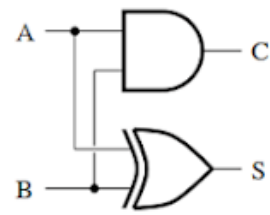
S : somme des unités

C : retenue

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S = A \oplus B \text{ (A XOR B)}$$

$$C = A.B \text{ (A AND B)}$$



Circuit logique correspondant

Une modélisation à l'aide du logiciel Logisim

