

## Architecture des processeurs

### *Architecture de von Neumann*

Le CPU (Central Processing Unit), plus couramment appelé **processeur**, est au cœur de l'ordinateur. C'est lui qui exécute les instructions des programmes. L'architecture globale des processeurs est appelée **Architecture de von Neumann**, du nom du mathématicien, et physicien, américano-hongrois John von Neumann (1903–1957). Ce n'est pas la seule architecture possible, mais c'est celle qui est encore utilisée par tous les ordinateurs.

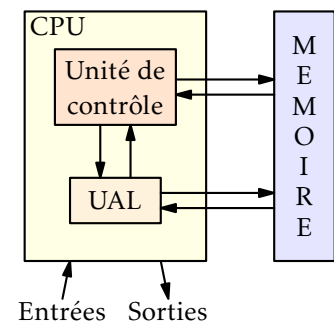
Un processeur s'articule autour de trois parties :

- **L'UAL** qui réalise les calculs ou les opérations logiques ;
- **l'unité de contrôle** qui exécute les instructions des programmes ;
- une petite quantité de mémoire, appelée **registres**, qui permet de stocker les résultats intermédiaires et les valeurs nécessaires pour les opérations en cours.

Le processeur interagit avec la mémoire vive afin de lire et d'écrire les valeurs des variables, leur nombre excédant rapidement le nombre de registres disponibles. Afin de communiquer, les différents composants utilisent des **bus**.

Une particularité de l'architecture de von Neumann, c'est que les instructions des programmes sont également stockées en mémoire avec les données. Un des registres du processeur contient l'adresse de la prochaine instruction à exécuter. Il est donc tout à fait possible de modifier les instructions en cours d'exécution. C'était nécessaire, initialement, pour réaliser des boucles. De nos jours il y a d'autres méthodes plus simples pour faire des boucles, mais il est toujours possible de modifier les instructions d'un programme en cours d'exécution. C'est notamment ce que font, ou cherchent à faire, les virus informatiques.

Un processeur peut avoir plusieurs **cœurs**, qui sont autant de "petits" processeurs permettant d'exécuter en parallèle plusieurs programmes ou de se partager certains calculs pour obtenir un résultat plus rapidement. Il n'est malheureusement pas possible de le faire à chaque fois et la **programmation parallèle** reste très difficile.



### *Jeu d'instructions*

Chaque processeur est capable d'exécuter un certain nombre d'instructions qui dépendent des circuits qu'il contient. L'ensemble de ces instructions s'appelle le **jeu d'instructions**.

Chaque instruction est composée d'une suite de bits. On appelle cela le **langage machine**. Typiquement, un processeur 32 bits acceptera des instructions de 32 bits. Ces instructions sont de plusieurs natures, mais on peut les regrouper en 3 grands types :

- copier la valeur d'un registre en mémoire, à l'adresse donnée ;
- copier le contenu de la mémoire à l'adresse donnée dans un registre ;
- effectuer une opération arithmétique ou logique et mettre le résultat dans un registre.

Afin de déterminer l'action à effectuer, l'instruction est décodée, en général en 2 parties :

- le **code opération** (ou opcode) qui indique le type de traitement à réaliser ;
- les **opérandes** qui indiquent les données à manipuler.

Il y a 3 types d'opérandes possibles :

- une **valeur immédiate** qui sera utilisée dans l'opération, comme un entier par exemple ;
- un registre du processeur, dans lequel la valeur sera lue ou écrite ;
- une adresse en mémoire vive, indiquant le registre de la RAM dans lequel écrire ou lire la valeur.

Le type du processeur définit la taille des valeurs que peuvent prendre les valeurs dans les registres et la taille des adresses. Ainsi, pour un processeur 32 bits, les valeurs manipulées sont représentées sur 32 bits. Pour des entiers naturels, cela fait donc des valeurs allant de 0 à 4 294 967 295. Ce qui explique aussi pourquoi un ordinateur 32 bits ne peut pas avoir plus de 4Go de mémoire, puisqu'il n'a pas assez d'adresses pour avoir plus.

Les registres sont généralement notés R0, R1, et ainsi de suite. Il y a néanmoins 2 registres particuliers. Le premier, nommé IP (instruction pointer), ou PC (program counter), contient l'adresse, dans la mémoire vive, de la prochaine instruction à exécuter. Le second, IR (instruction register) contient l'instruction qui va être exécutée immédiatement par le processeur. À chaque nouveau cycle, le processeur effectue les étapes suivantes :

- 1) il copie dans le registre IR le contenu de la mémoire vive à l'adresse pointée par IP ;
- 2) il décode l'instruction contenue dans IR, ce qui active le circuit électrique qui réalise l'opération visée ;
- 3) il exécute l'instruction décodée et met à jour la valeur de IP pour continuer le programme.

Puisqu'une instruction peut modifier directement la valeur de IP, il est possible de réaliser des boucles ou des tests.

---

### *Le meilleur processeur*

---

Fondamentalement, tous les processeurs sont capables de faire la même chose. Certains seront juste plus rapides ou il sera plus simple de programmer certaines fonctionnalités. S'il est possible d'écrire un programme permettant de réaliser une tâche donnée pour un processeur, il est possible d'écrire un programme faisant la même tâche pour un autre processeur, même si le jeu d'instructions est différent. Par exemple certains processeurs sont capables de réaliser directement des multiplications et des divisions alors que pour des modèles plus anciens, il fallait écrire un bout de programme pour effectuer ces opérations.

Il y a également des processeurs ultra spécialisés pour certaines opérations, comme les processeurs sonores ou ceux des cartes graphiques.

---

### *Les programmes informatiques*

---

L'utilisation conjointe d'un processeur et de la RAM permet de faire des ordinateurs programmables qui ne sont pas conçus pour exécuter un unique programme mais au contraire qui peuvent changer de programmes. Les premiers programmes ont été écrits avant l'invention du premier ordinateur. La mathématicienne britannique Ada Lovelace (1815–1852) a publié ce qui est considéré comme le premier programme informatique, appelé *Note G*, et qui a pour but de calculer les nombres de Bernoulli. Ce programme était destiné à la *Machine Analytique* de Charles Babbage, qui n'a jamais été construite.

Lorsque les premiers ordinateurs programmables sont arrivés, au milieu du 20<sup>e</sup> siècle, c'est le langage machine qui a été utilisé. Rapidement, Grace Hopper (1906–1992), une des premières informaticienne américaine prône pour l'utilisation de langages plus proches des langues naturelles. Elle créera le premier compilateur en 1951 et le langage Cobol en 1959. Un **compilateur** permet de prendre un programme écrit dans un langage de haut niveau comme C ou Java et de le transformer en programme en langage machine. On peut voir cela comme la traduction d'un livre. La traduction est faite une bonne fois pour toute. Elle dépend du type de processeur de la machine et du système d'exploitation.

En parallèle, des langages interprétés, comme Lisp, Basic ou Python, ont été développés. Un **interpréteur** est un programme qui prend un autre programme et le traduit instruction par instruction en langage machine au fur et à mesure de l'exécution. C'est comme une traduction simultanée lors de l'interview d'une personne étrangère. La traduction est faite à chaque fois que le programme est exécuté.