

Architecture des ordinateurs, modèle de von Neumann

1ere NSI

L'unité de base dans les ordinateurs est le transistor, il s'agit en quelque sorte d'un interrupteur qui laisse ou ne laisse pas passer le courant. On parle d'un état bas (0) et d'un état haut (1).

De nos jours les transistors sont regroupés dans des circuits intégrés. C'est grâce à ses transistors que l'on réalise les opérations booléennes.



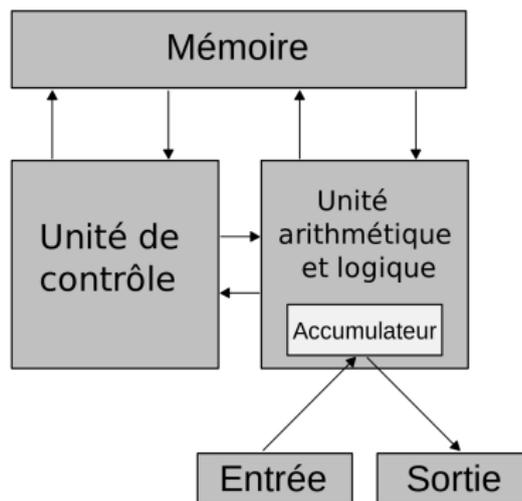
Un transistor



Un circuit intégré

L'architecture de von Neumann

Le nom de cette architecture est appelée ainsi en référence au mathématicien John von Neumann. Il a mis au point la première description d'un ordinateur dont le programme serait stocké dans la mémoire.



L'architecture von neumann

L'unité de contrôle est la partie qui gère:

- L'ordre des instructions du programme.
- Quelles opérations sont à effectuer en fonction des instructions.

L'unité arithmétique et logique est la partie qui effectue les opérations logique (ET, OU, ...) ainsi que les opérations mathématiques (addition, ...).

La mémoire est la partie qui stocke les données en cours d'utilisation, dans le processeur il s'agit du registre. Il peut stocké des données ou des instructions.

La RAM ou "Random access memory".

On peut représenter la mémoire comme une série de cases qui contiennent une information (0 ou 1). Dans la RAM la mémoire est gérée 8 bit par 8 bit (8 valeurs 0 ou 1 à la suite).

Chaque bit d'information dans la ram est composé d'un transistor (pour gérer si la valeur du bit est à 0 ou à 1) et d'un condensateur qui peut tenir une charge électrique un court moment.

Il y a une discussion constante entre le processeur et la RAM.
En effet le registre ayant une capacité en mémoire très limitée, la plupart des données des programmes sont stockées dans la RAM.
Elles ne seront récupérées par le processeur que lorsqu'on voudra effectuer des opérations à l'aide de celles-ci.

Nous allons parler de quelques instructions du langage machine (le langage compris par l'ordinateur):

- STR R3,125 " stocker a l'adresse 123 le contenu du registre 3" LDR stocke dans le registre.
- ADD R1,R0,128 " ajoute R0 et la valeur 123 dans R1".
- MOV R0, 4 " mets le contenu de l'adresse mémoire 4 dans le registre 0".
- CMP permet de comparer des valeurs et de verifier l'égalité.
- BNE permet d'aller vers l'instruction suivant si CMP donne une différence dans la comparaison.

Le but n'est pas de connaître le langage machine par coeur mais de pouvoir l'interpréter a l'aide des commandes.

Nous allons étudier le langage machine a l'aide de l'équivalence entre le langage machine et les instructions python.

langage machine

```
MOV R0, #4
STR R0,30
MOV R0, #8
STR R0,75
LDR R0,30
CMP R0, #10
BNE else
MOV R0, #9
STR R0,75
B endif
else:
LDR R0,30
ADD R0, R0, #1
STR R0,30
endif:
MOV R0, #6
STR R0,23
HALT
```

```
x = 4
y = 8
if x == 10:
    y = 9
else :
    x=x+1
z=6
```

<http://www.peterhigginson.co.uk/AQA/>

Nous allons observer le fonctionnement du code vu précédemment sur ce site.