

# Introduction au développement logiciel

## Notes de cours: hardware

Georges-André Silber

Octobre 2025

### Table des matières

<b>Transistors et circuits intégrés</b>	<b>2</b>
<b>MOSFET</b>	<b>2</b>
<b>Portes logiques et mémoires</b>	<b>2</b>
<b>Loi de Moore</b>	<b>4</b>
<b>Machine de Von Neumann</b>	<b>4</b>
<b>Hiérarchie mémoire, caches et pages</b>	<b>4</b>
<b>Coût des opérations sur un processeur</b>	<b>7</b>
<b>Pipelines</b>	<b>7</b>
<b>Architectures parallèles et types de parallélisme</b>	<b>8</b>
<b>Top 500 : Fugaku</b>	<b>8</b>

1. Le cœur d'un ordinateur est son processeur, qui agit sur une mémoire principale organisée en cases de tailles fixes, accessibles par des adresses, en utilisant des cases mémoire de travail appelées registres.
2. Fondamentalement, un processeur est un automate exécutant, lorsqu'il est sous tension, une suite d'instructions se trouvant en mémoire, en commençant à une adresse fixe stockée dans un registre spécial appelé *Program Counter* (PC).
3. Les ordinateurs modernes ont une architecture de type Von Neumann.
4. Un processeur ne peut se programmer que grâce aux instructions de son architecture externe (ISA, *instruction set architecture*).
5. Les instructions du processeur peuvent être regroupées dans trois classes principales : affectations, tests et branchements.
6. L'affectation consiste à altérer le contenu d'une case mémoire, en copiant une valeur d'une case mémoire vers une autre, en altérant éventuellement le contenu par une opération arithmétique.
7. Le test consiste à comparer deux valeurs, constantes ou se trouvant en mémoire. Le résultat de ce test peut ensuite être utilisé pour exécuter ou non une instruction.
8. Après l'exécution d'une instruction, le processeur exécute normalement l'instruction se trouvant juste après en mémoire, dont l'adresse est stockée dans le registre PC (le *compteur ordinal* ou *program counter*). Le branchement permet de faire exécuter au processeur une instruction se trouvant à une autre adresse, en altérant PC.
9. Le langage de programmation constitué par l'ISA d'un processeur est *Turing complete* car il permet, suivant le théorème de Böhm–Jacopini, d'implémenter les trois structures de contrôles que sont la séquence, la sélection via un booléen (le *if*) et l'itération (le *while*).

10. Andrew Tanenbaum, distingue trois niveaux matériels dans un ordinateur : niveau 2, architecture externe ou de jeu d'instructions (ISA) ; niveau 1, microarchitecture ; niveau 0, réalisation en logique numérique. Exemple (fictif) : niveau architecture externe, il existe huit registres d'usage général. L'instruction ADD effectue une addition entre deux quelconques de ces registres et met le résultat dans un quelconque troisième ; niveau microarchitecture (architecture interne) : le traitement des instructions est pipeliné et deux unités arithmétiques et logiques sont capables d'effectuer l'addition de l'instruction ADD ; niveau réalisation : le processeur est réalisé dans une technologie CMOS avec trois niveaux de métaux et une finesse de gravure de 130 nanomètres.

## Transistors et circuits intégrés

Alors que les premiers ordinateurs utilisaient des relais électriques ou des tubes à vide, l'invention du *transistor* en 1947 par William Shockley va permettre de construire des circuits électroniques de plus en plus complexes (figure 1). Le *circuit intégré*, technologie inventée simultanément par Jack Kilby de Texas Instruments en 1958 (prix nobel de physique en 2000) et Robert Noyce de Fairchild Semiconductors, utilisant du silicium semi-conducteur, va permettre aux ordinateurs modernes tels que nous les connaissons aujourd'hui de se développer. Robert Noyce fonde Intel avec Gordon Moore en 1968, société qui va inventer le *microprocesseur* en 1971, l'Intel 4004 composé de 2 300 transistors.

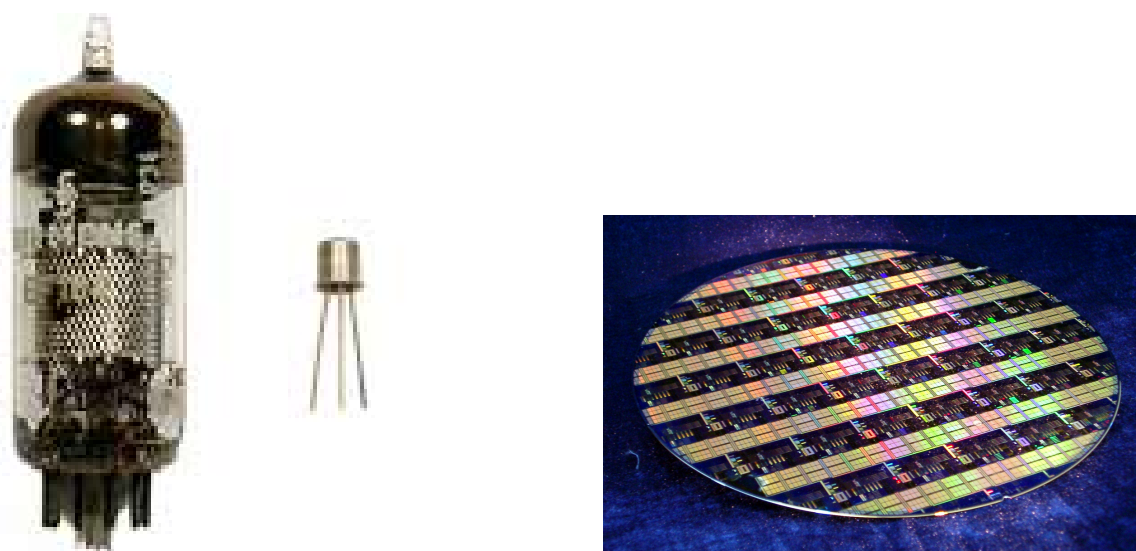


FIGURE 1 – Tube à vide, transistor et *Wafer* de circuits intégrés

## MOSFET

La technologie centrale de l'informatique moderne est le transistor à effet de champ à grille isolée (*Metal-oxide-semiconductor field-effect transistor*), inventé de façon théorique en 1920, utilisé pour la première fois dans les circuits intégrés en 1965 (figure 2)<sup>1</sup>. Premier transistor compact qui a pu être miniaturisé et produit en masse. Deux types, *p* et *n*, utilisés conjointement en technologie CMOS (*Complementary MOS*).

La finesse de gravure des transistors en technologie MOSFET ne cesse de décroître depuis les années 70. Cette technologie est toujours utilisée aujourd'hui (table 1).

## Portes logiques et mémoires

La combinaison de transistors, condensateurs, résistances et diodes dans les circuits intégrés permet de réaliser des circuits électroniques complexes, pour calculer (figure 3) et stocker de l'information.

1. [https://en.wikipedia.org/wiki/MOSFET#/media/File:Threshold\\_formation\\_nowatermark.gif](https://en.wikipedia.org/wiki/MOSFET#/media/File:Threshold_formation_nowatermark.gif)

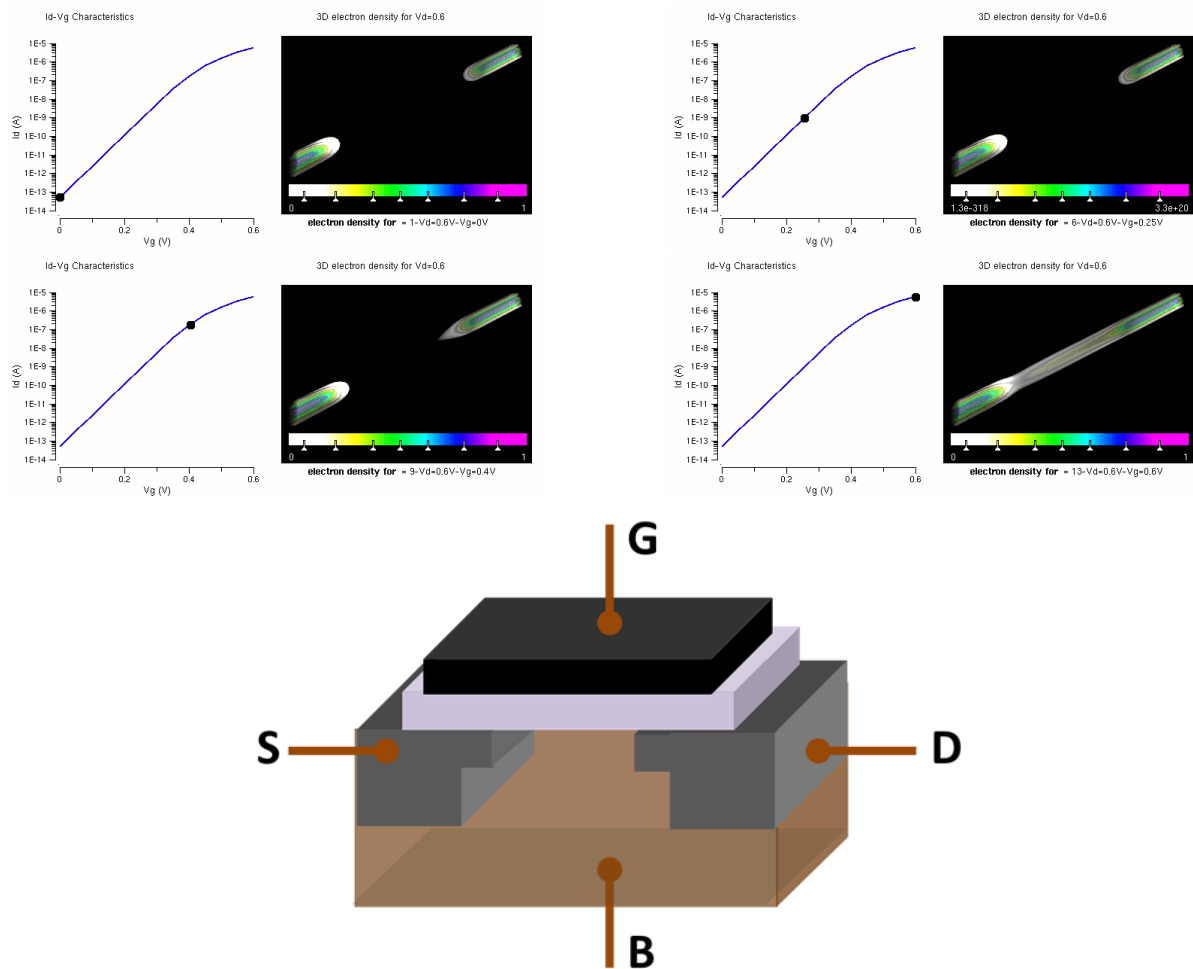


FIGURE 2 – Simulation d'un effet de seuil et schéma d'un MOSFET, avec sa grille G, sa source S, son drain D et son substrat B.

10 $\mu$ m	1971	800 nm	1987	130 nm	2001	22 nm	2012	3 nm	2022
6 $\mu$ m	1974	600 nm	1990	90 nm	2003	14 nm	2014	2 nm	2023
3 $\mu$ m	1977	350 nm	1993	65 nm	2005	10 nm	2016	1.4 nm	2029
1.5 $\mu$ m	1981	250 nm	1996	45 nm	2007	7 nm	2018		
1 $\mu$ m	1984	180 nm	1999	32 nm	2009	5 nm	2020		

TABLE 1 – Finesse de gravure des transistors MOSFET depuis 1971.

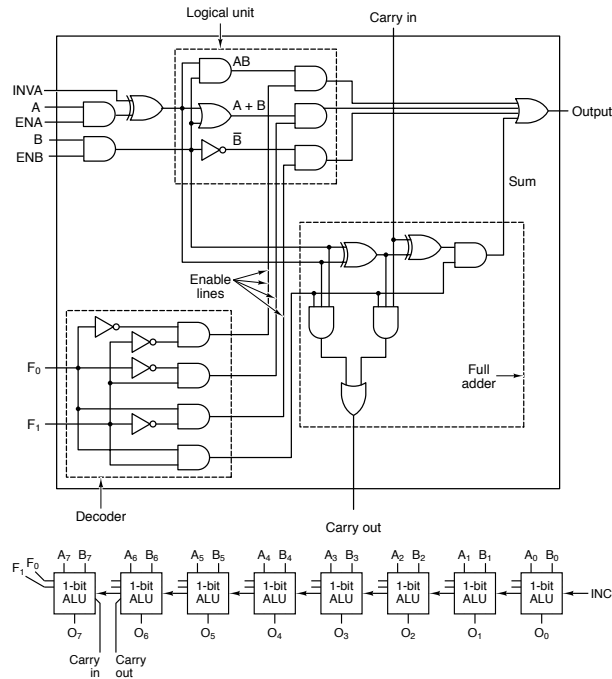


FIGURE 3 – Additionneur 1 bit et 8 bits

## Loi de Moore

Gordon Moore, cofondateur d'Intel, a prédit en 1965 un accroissement exponentiel du nombre de transistors dans un circuit intégré par unité de surface : doublement tous les ans en 1965, ajusté à tous les deux ans à partir de 1975 (figure 4). Observation empirique, par certains côtés une prophétie autoréalisatrice.

## Machine de Von Neumann

L'architecture d'un ordinateur (figure 5) d'aujourd'hui est toujours similaire à celle proposée par le mathématicien Jon Von Neumann en 1945 dans le cadre du projet EDVAC (l'un des tout premiers ordinateurs électroniques en mode binaire). L'architecture de von Neumann décompose l'ordinateur en 4 parties :

- l'unité arithmétique et logique (UAL) : opérations de base;
- l'unité de contrôle, chargée du séquençage des opérations;
- la mémoire qui contient à la fois les données et le programme qui indiquera à l'unité de contrôle les calculs à faire sur ces données;
- les dispositifs d'entrée-sortie, qui permettent de communiquer avec le monde extérieur.

## Hierarchie mémoire, caches et pages

*There are 2 hard problems in computer science : cache invalidation, naming things, and off-by-1 errors.*

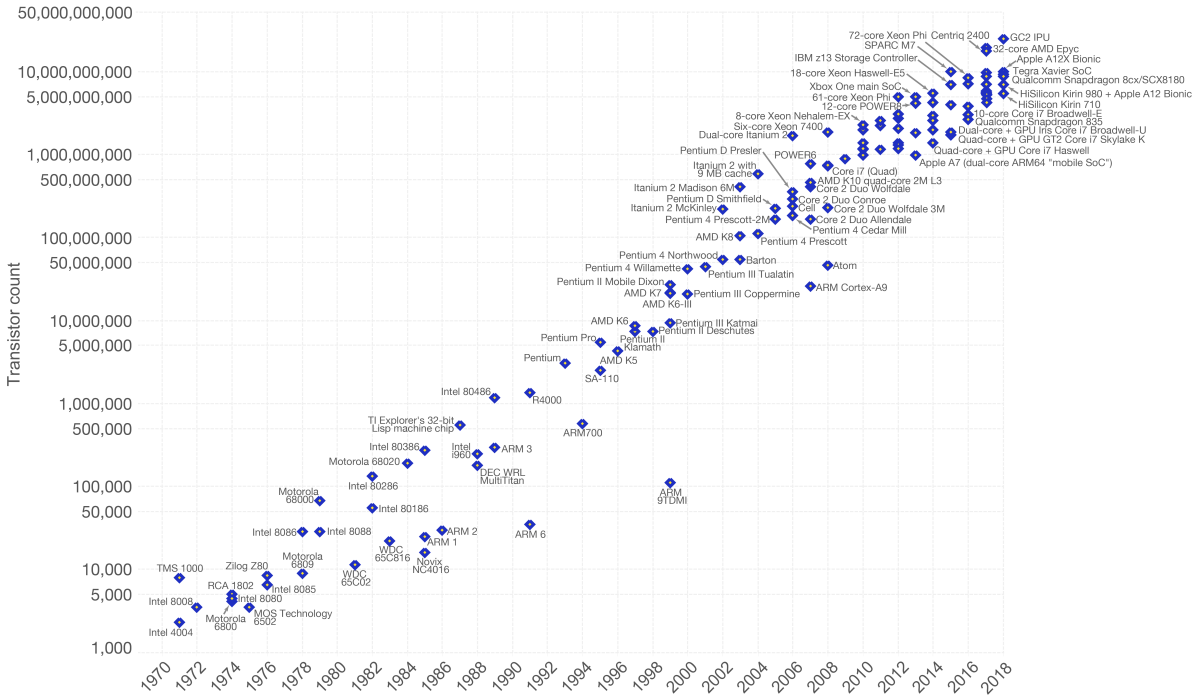
– Leon Bambrick<sup>2</sup>

La mémoire d'un ordinateur se compose de différentes technologies, que l'on a coutume de représenter de manière hiérarchique dans une pyramide (figure 6), le haut de la pyramide est le plus rapide et a le moins de capacité, le bas de la pyramide est le plus lent et a le plus de capacité.

2. <https://twitter.com/secretGeek/status/7269997868?s=20>

# Moore's Law – The number of transistors on integrated circuit chips (1971-2018)

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are linked to Moore's law.



Data source: Wikipedia ([https://en.wikipedia.org/wiki/Transistor\\_count](https://en.wikipedia.org/wiki/Transistor_count))  
The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.  
Licensed under CC-BY-SA by the author Max Roser.

FIGURE 4 – Illustration de la loi de Moore : nombre de transistors par puce de circuit intégré, 1971–2018.

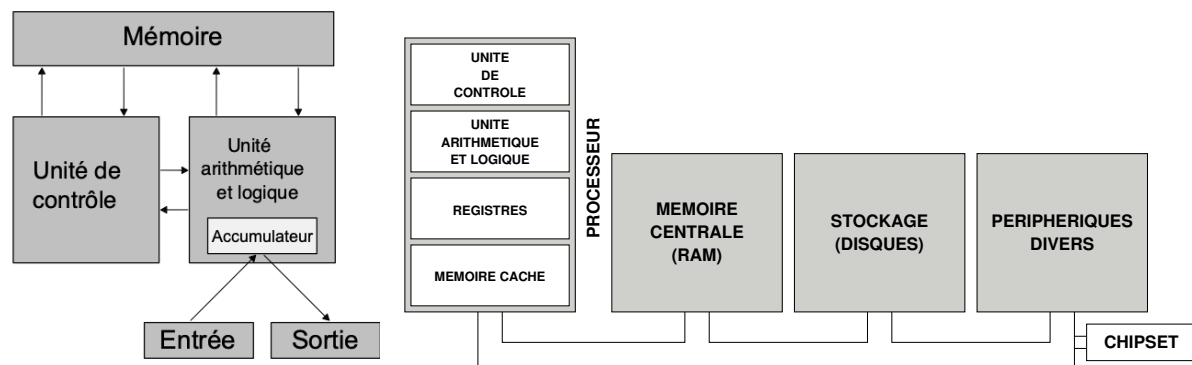


FIGURE 5 – Architecture de von Neumann et d'un ordinateur moderne.

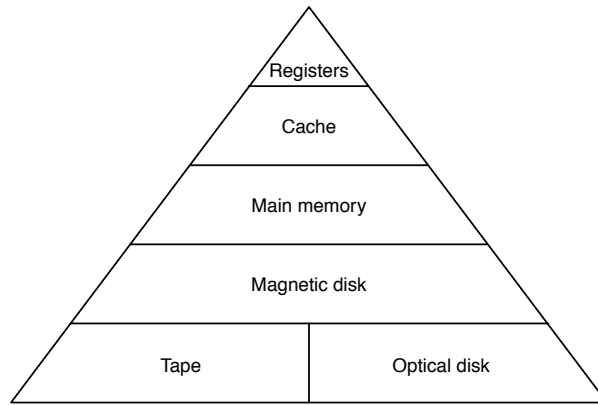


FIGURE 6 – Hiérarchie mémoire dans un ordinateur

Afin de tirer parti des propriétés de localité spatiale et de localité temporelle dans les programmes, des *caches* mémoires ont été introduits (figure 7) pour la première fois en 1968 dans l'IBM 360/85. L'unité de base au sein d'un cache est la *ligne de cache*, composée d'une suite consécutive de quelques dizaines d'octets contigus en mémoire centrale – typiquement 32 ou 64. Il y a habituellement entre 1 et 3 caches consécutifs entre la RAM et le processeur, de tailles croissantes : de plusieurs Ko pour le cache le plus proche du processeur à plusieurs Mo pour le cache le plus éloigné. Ces caches contiennent des données, des instructions et des correspondances entre des adresses physiques et des adresses logiques (TLB).

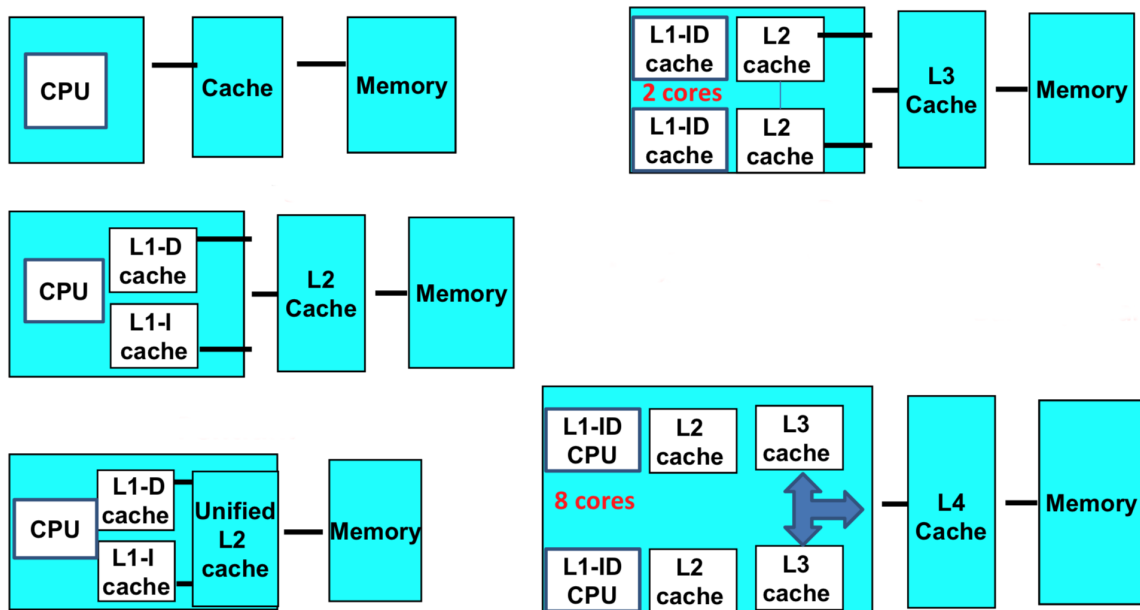


FIGURE 7 – Différentes architectures de cache

La plupart des systèmes d'exploitations utilisent un système de pagination transformant la RAM de la machine en un cache de pages de plusieurs Ko (par exemple 4 Ko), une partie des disques étant utilisée pour stocker des pages non utilisées à un instant  $t$ . La quantité de mémoire vive disponible sur le système est alors la somme de la RAM et de cet espace de stockage.

## Coût des opérations sur un processeur

Il y a un écart important en terme de nombre de cycles entre les différentes opérations<sup>3</sup> effectuées dans un microprocesseur (figure 8).

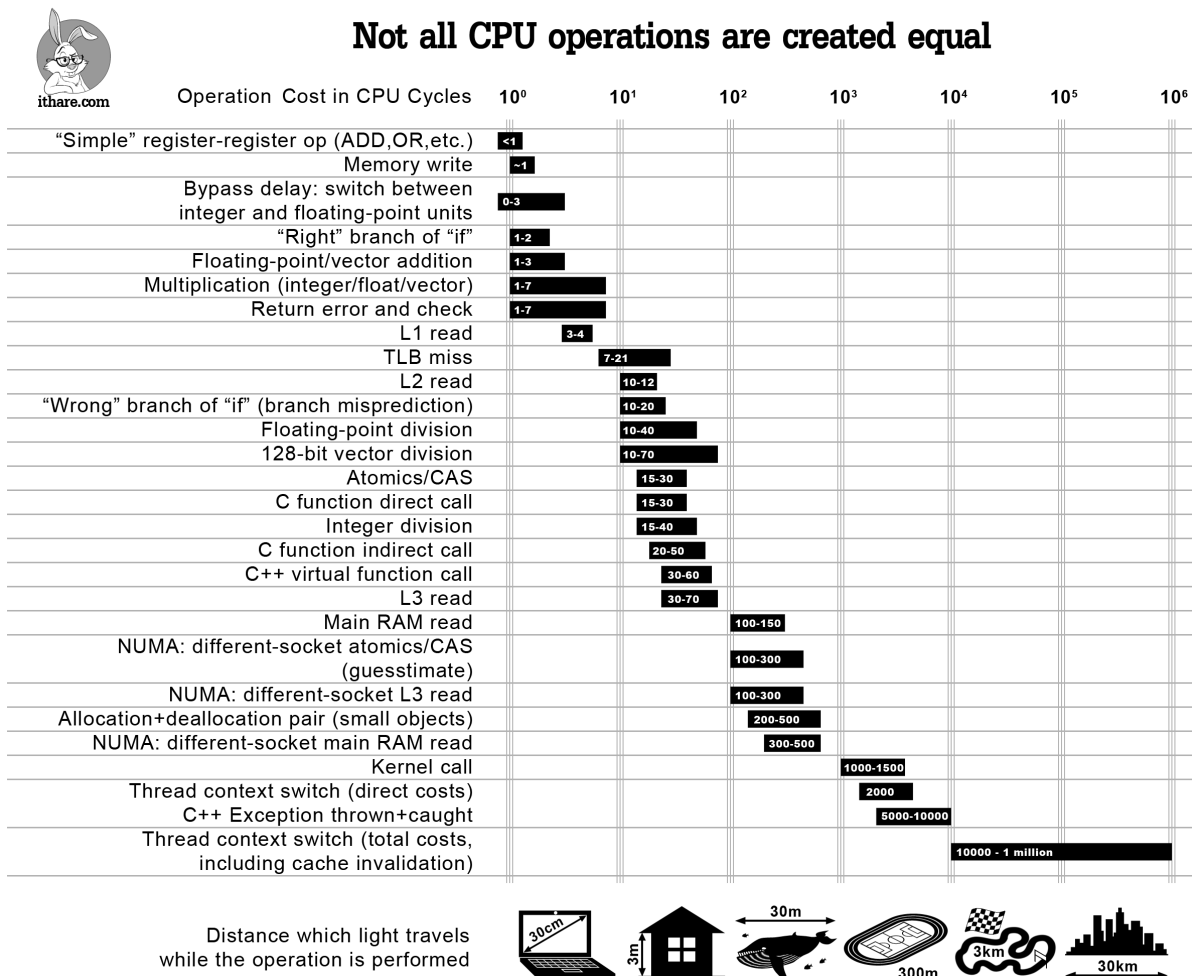


FIGURE 8 – Coûts de diverses opérations CPU en nombre de cycles processeur

## Pipelines

L'exécution d'une instruction dans un processeur se fait en plusieurs étapes. Pour une introduction au concept, considérons les 5 étapes suivantes :

- IF (Instruction Fetch) charge l'instruction à exécuter dans le pipeline
- ID (Instruction Decode) décode l'instruction et adresse les registres
- EX (Execute) exécute l'instruction sur la ou les unités arithmétiques et logiques
- MEM (Memory), dénote un transfert depuis un registre vers la mémoire dans le cas d'une instruction du type STORE (accès en écriture) et de la mémoire vers un registre dans le cas d'un LOAD (accès en lecture).
- WB (Write Back) stocke le résultat dans un registre. La source de ce résultat peut être la mémoire (suite à une instruction LOAD), l'unité de calcul (suite à un calcul à l'étape EX) ou bien un registre (cas d'une instruction MOV).

3. <http://ithare.com/infographics-operation-costs-in-cpu-clock-cycles/>

Chaque instruction s'exécute donc en 5 cycles dans le meilleur des cas (latence) – certaines étapes prennent un temps variable, notamment en cas d'accès mémoire. Sans pipeline (figure 9) l'exécution de trois instructions prend donc au minimum 15 cycles. Sur un processeur tournant par exemple à 2 GHz –  $2 \cdot 10^9$  cycles par seconde, cela fait donc 400 millions d'instructions se finissant par seconde dans le meilleur des cas. Avec un pipeline (figure 9), on atteint dans le meilleur des cas 2 milliards d'instructions se finissant par seconde.

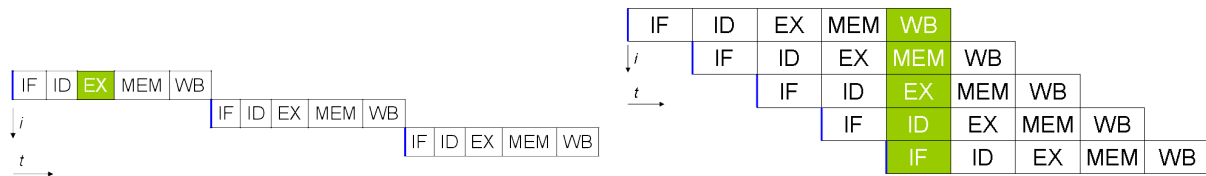


FIGURE 9 – Exécution du cycle *decode-fetch-execute* sans et avec un pipeline.

La plupart des processeurs modernes intègrent plusieurs pipelines (figure 10), certains de type VLIW (figure 10) utilisent des instructions *longues* permettant d'effectuer plusieurs opérations dans l'UAL.

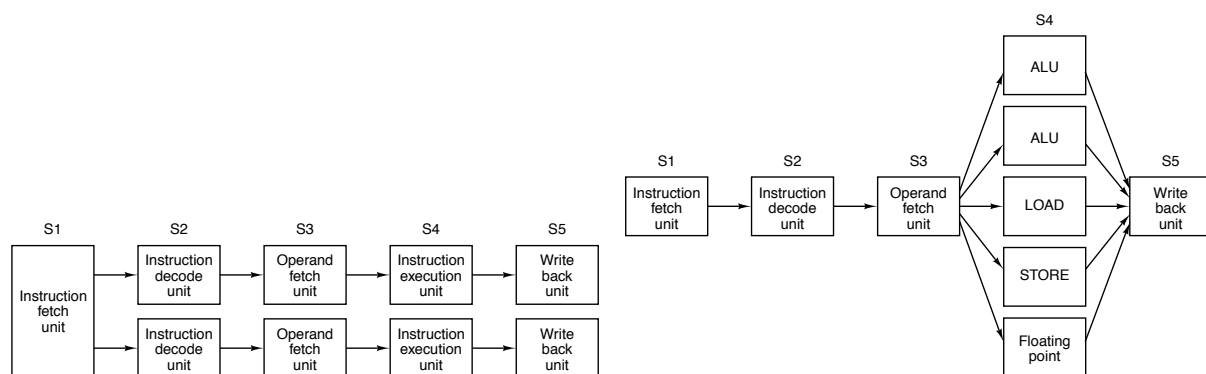


FIGURE 10 – Pipeline double

## Architectures parallèles et types de parallélisme

Il existe deux grandes familles de machines multiprocesseurs, mémoire partagée ou à mémoire distribuée (figure 11), chaque processeur pouvant lui-même être subdivisé en plusieurs cœurs.

Le calcul sur GPU en tant que coprocesseur de calcul permet de calculer efficacement de manière parallèle sur certains types de problème (figure 12).

## Top 500 : Fugaku

Depuis 1993, le site [top500.org](https://www.top500.org)<sup>4</sup> établit la liste des 500 ordinateurs les plus puissants de la planète. En novembre 2022, l'ordinateur le plus puissant était l'ordinateur japonais Fugaku<sup>5</sup> construit par Fujitsu, composé de 7 630 848 cœurs (158 976 nœuds) et 4,85 PiB (pébibits :  $4,85 \times 2^{50} = 5 460 614 548 186 726$  octets). Sa puissance maximale mesurée par Linpack est de 442 PFlop/s.

Chaque nœud est un processeur à architecture Arm comprenant 52 cœurs, dont 4 cœurs dédiés au système d'exploitation et 4 dédiés aux échanges mémoire NUMA (figure 13). Ce processeur a des instructions vectorielles de 512 bits.

4. <https://www.top500.org>

5. <https://www.top500.org/system/179807/>, <https://www.r-ccs.riken.jp/en/fugaku/project/outline>



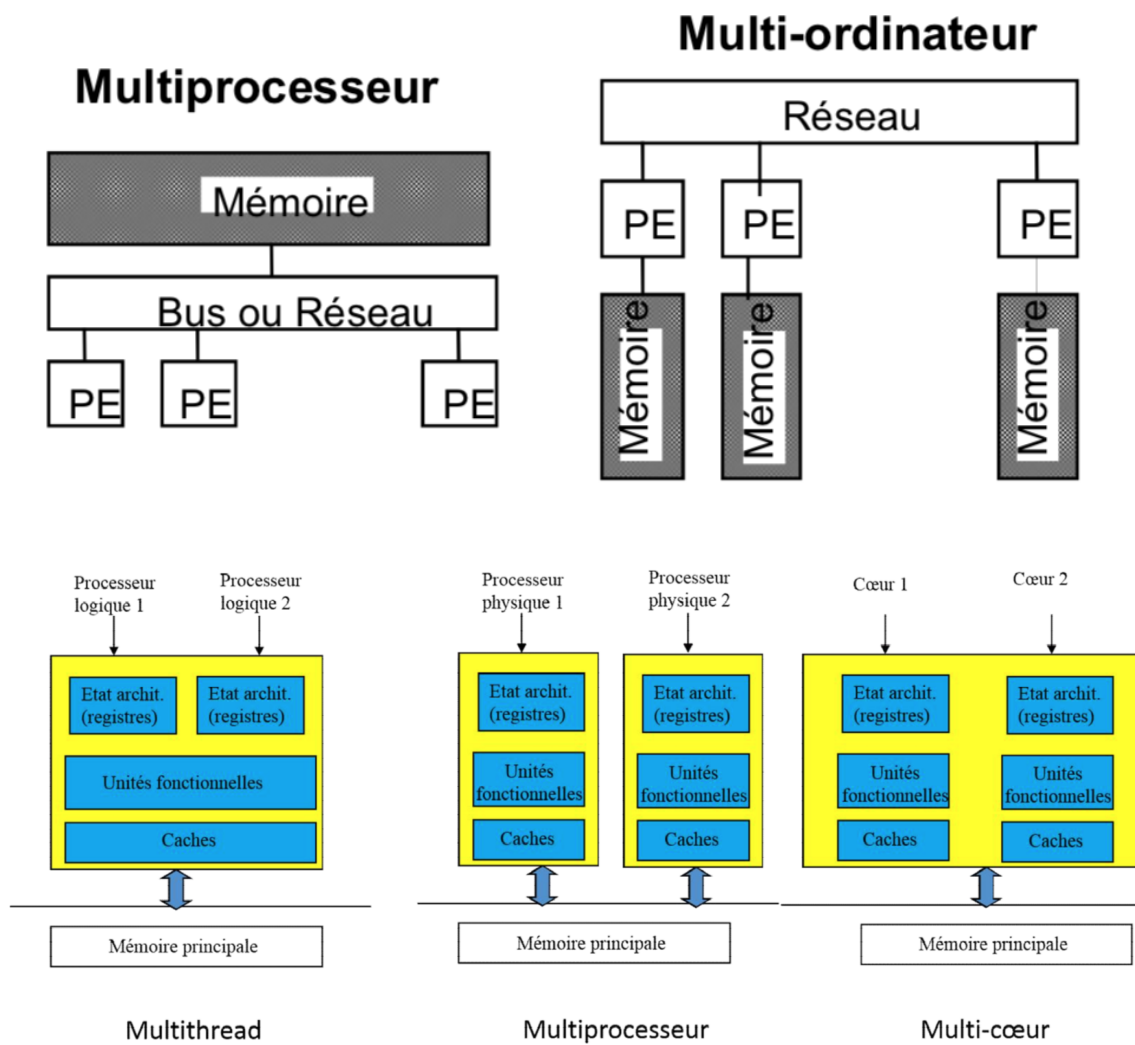


FIGURE 11 – Architectures multiprocesseurs

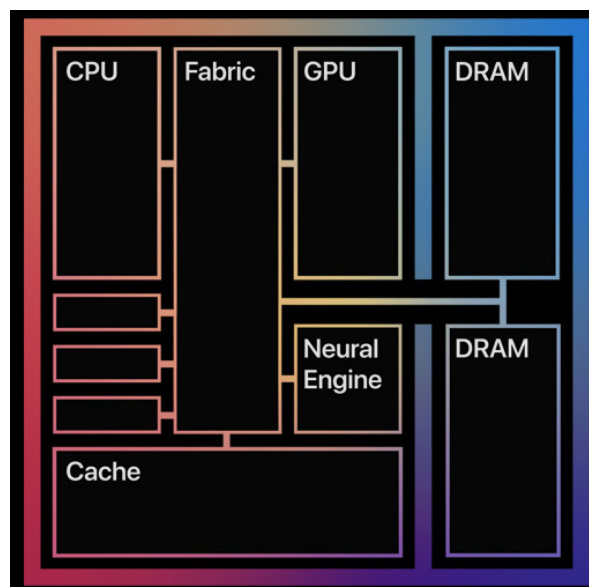
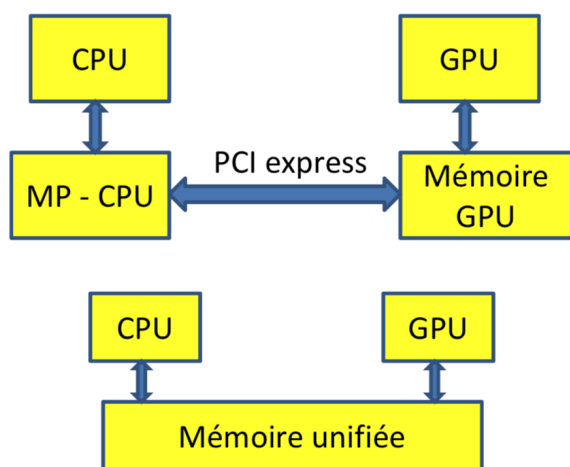


FIGURE 12 – Relation entre CPU et GPU, partage de la mémoire, architecture du processeur M1 d'Apple (à droite)

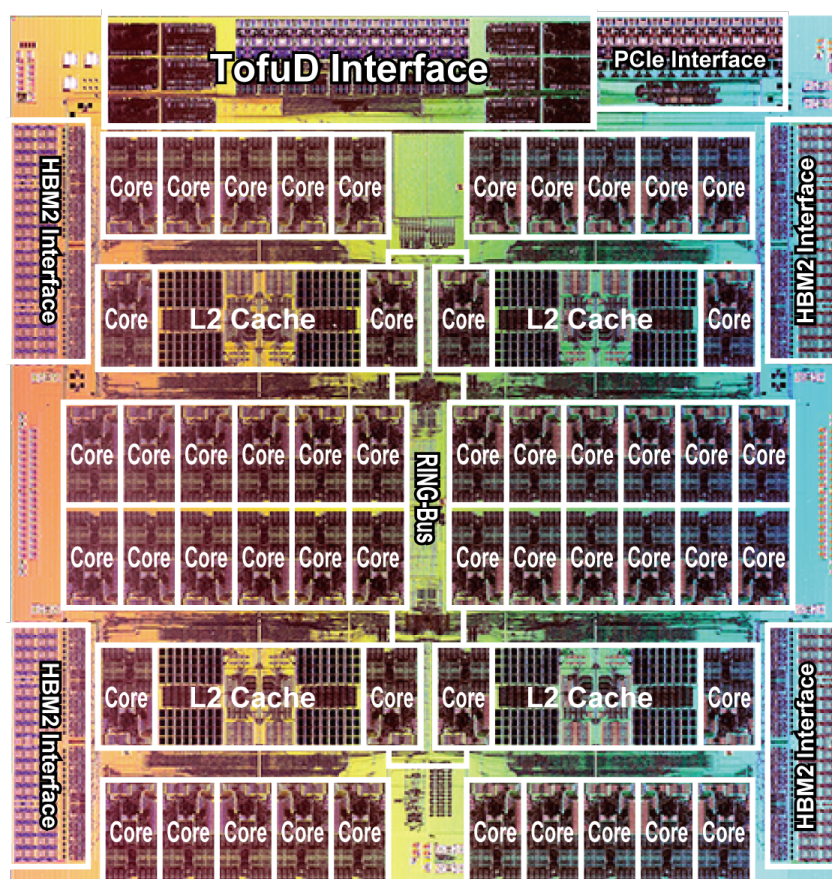


FIGURE 13 – Puce Fujitsu A64FX à architecture Armv8.2-A SVE 512bit.