

# Architecture des ordinateurs

## Le matériel

1. La carte mère
2. Le processeur
3. La mémoire
4. Le disque dur
5. Quelques composants
6. Cas pratique

# La carte mère

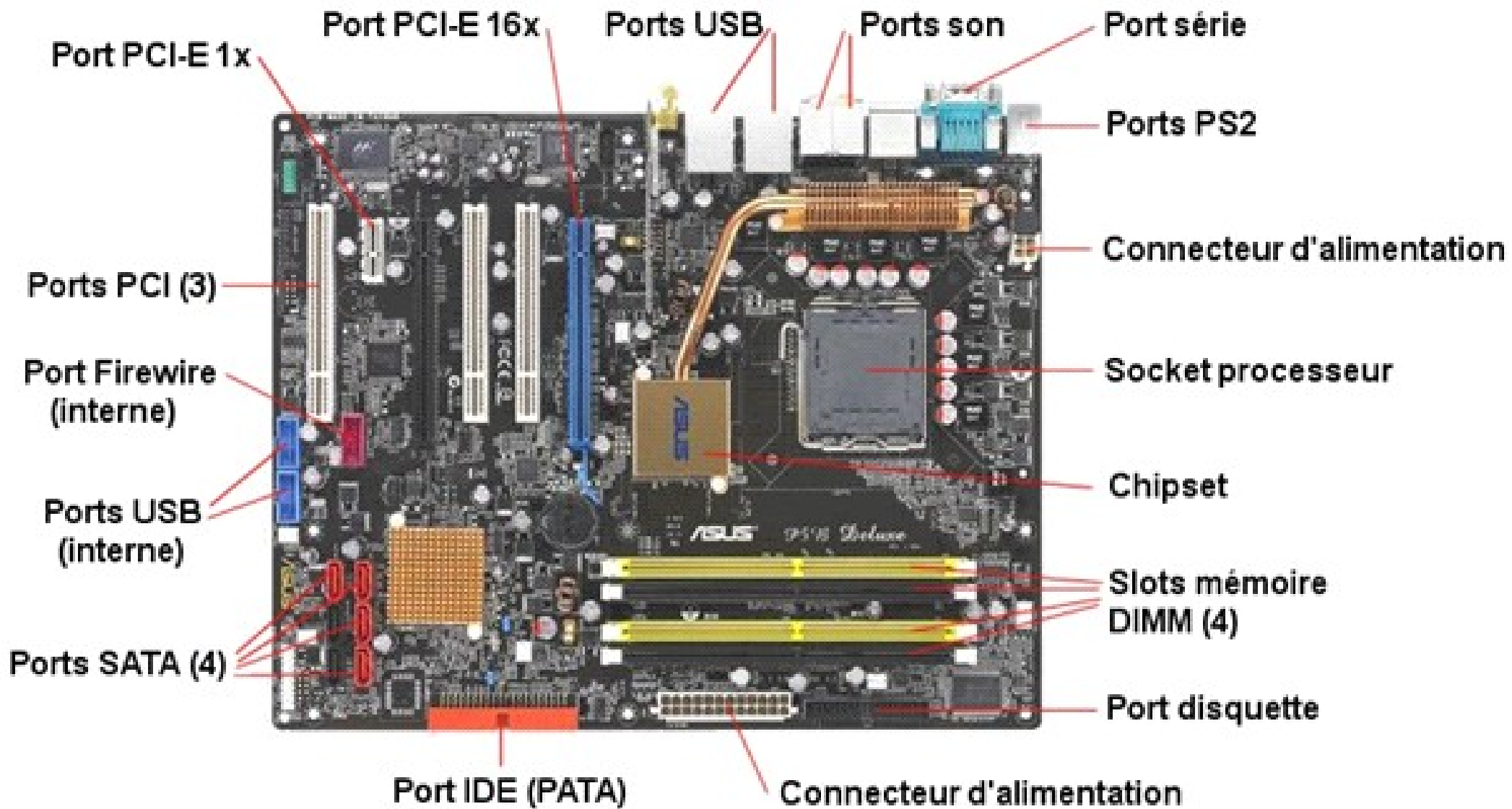
Élément principal de l'ordinateur

Elle définit le choix des autres composants

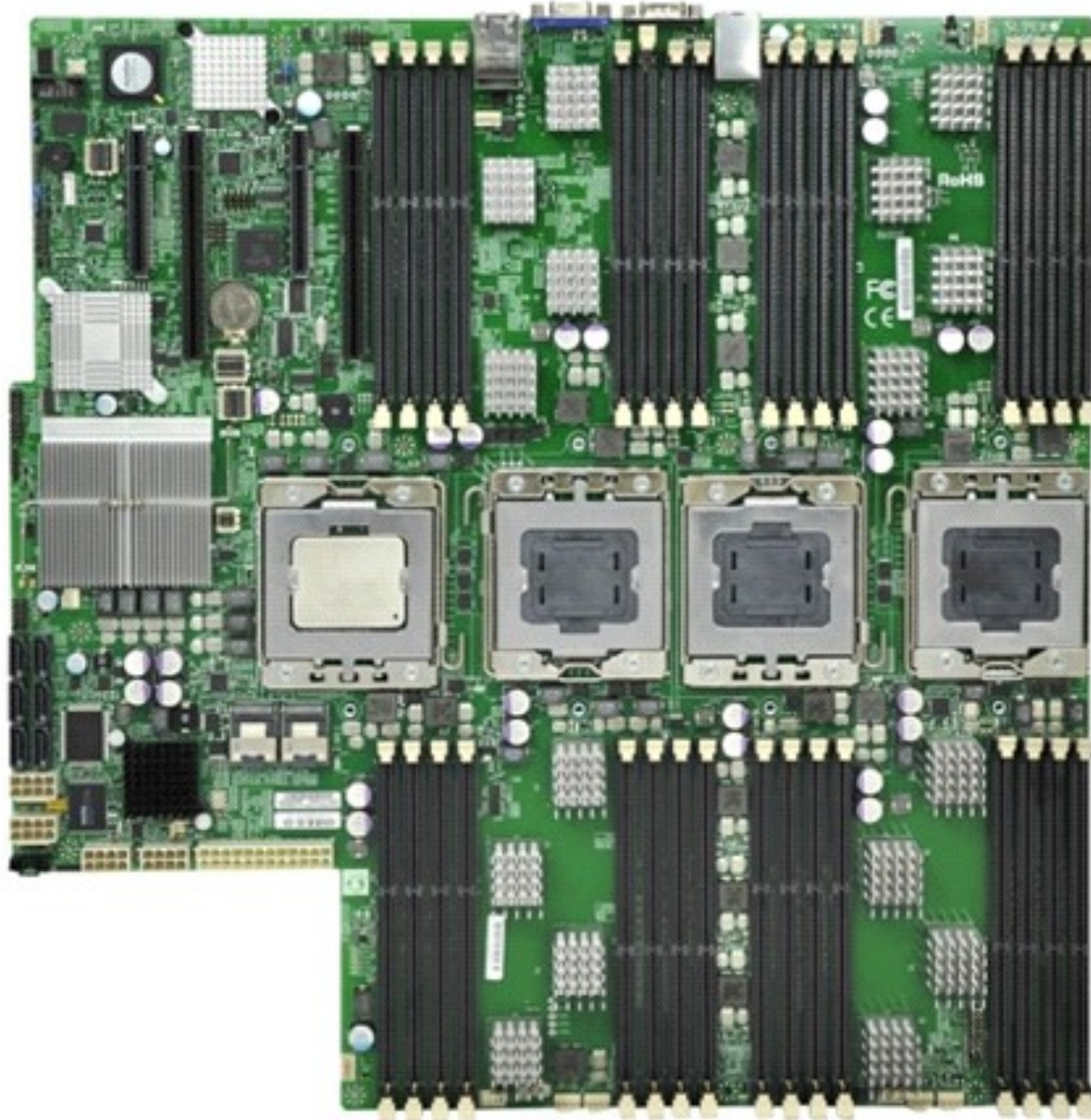
Elle prend la forme d'un grand circuit imprimé

Tous les autres composants se raccordent /  
branchent dessus

- Voici une carte mère de PC



- Voici une carte mère de serveur



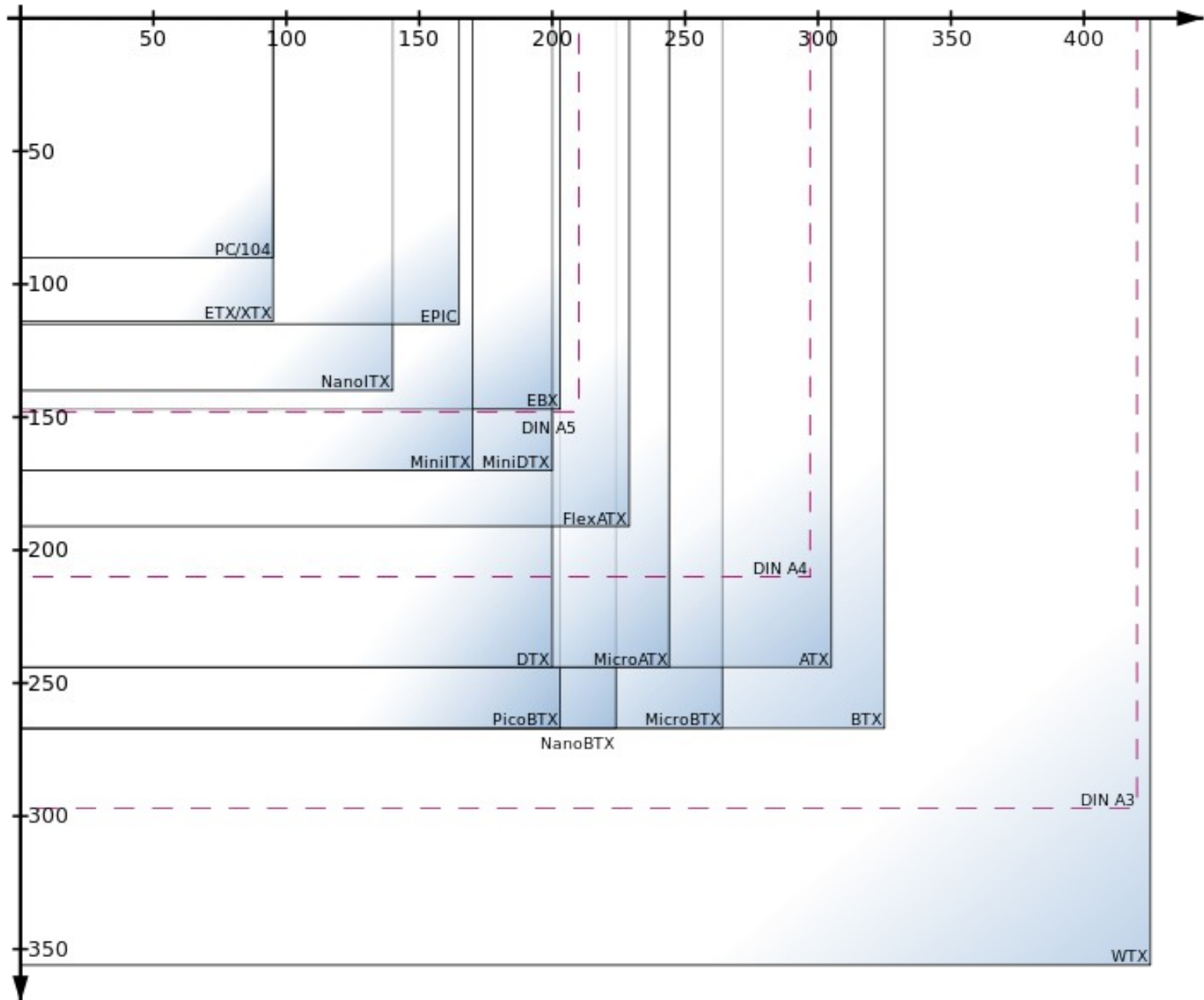
Voici les principales caractéristiques d'une carte mère :

- Le facteur d'encombrement
- Le chipset
- Le support processeur
- La mémoire vive
- Une horloge et une mémoire CMOS
- Un micrologiciel de configuration
- Des connecteurs d'extensions
- Les connecteurs d'entrée/sortie

Les facteurs d'encombrement les plus connus sont :

- L'ancien ATX devenu BTX avec toutes ces déclinaisons (micro, mini, etc...)
- ITX pour les cartes mères les plus petites (mini, nano)
- WTX pour les cartes mères de serveurs



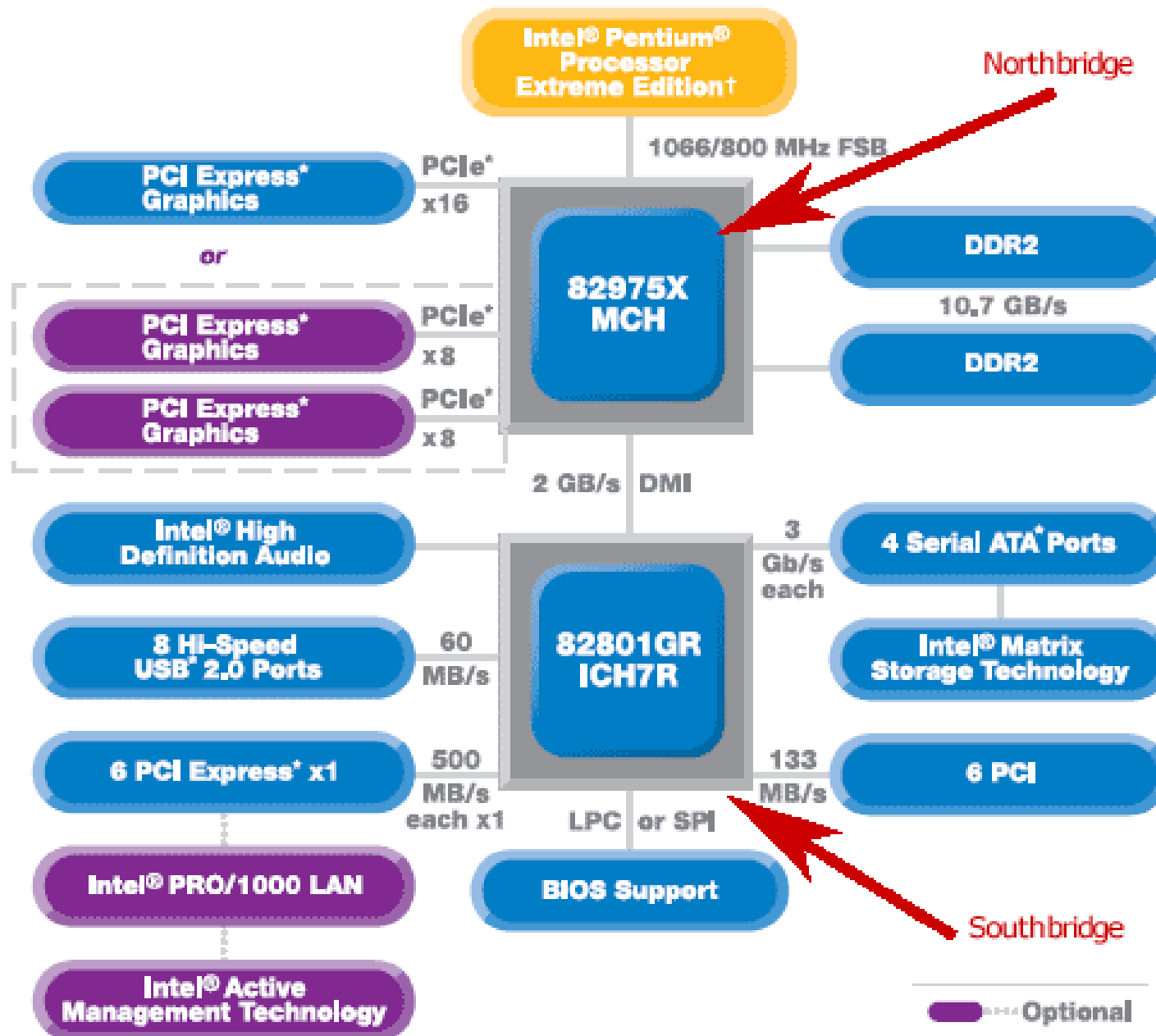


## Le chipset

- C'est un jeu de composants / circuits / puces
- Il est chargé de coordonner les échanges entre les différents composants de l'ordinateur
- Il est intégré à la carte mère (ou au processeur)
  - pas d'évolution possible
- Il intègre plusieurs fonctionnalités (puce graphique, carte son, etc.)
  - pas besoin de faire des dépenses supplémentaires

Le chipset est composé de deux entités :

- Le Northbridge (pont nord) : en charge des composants dits rapides (processeur, mémoire, carte graphique)
- Le Southbridge (pont sud) : en charge des périphériques lents (disque dur, ports USB, ...)



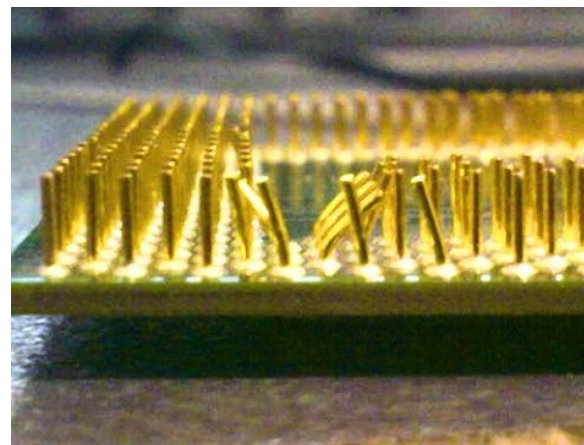
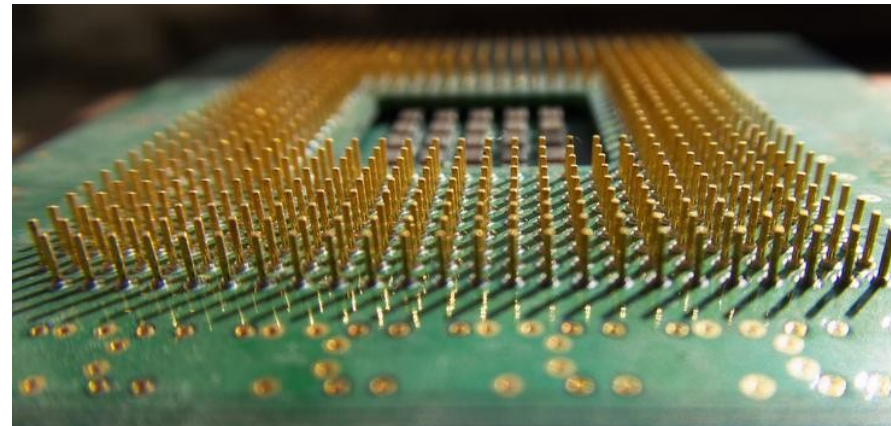
Le support processeur a connu plusieurs évolutions :

- Le slot (fente) : connecteur rectangulaire dans lequel on enfiche le processeur



Le support processeur a connu plusieurs évolutions :

- Le socket (embase) : connecteur carré possédant un grand nombre de trous dans lequel on enfiche le processeur



Le socket ZIF (Zero Insertion Force) : connecteur carré possédant un grand nombre de broches dans lequel on « pose délicatement » le processeur

Ce socket possède une petite manette :

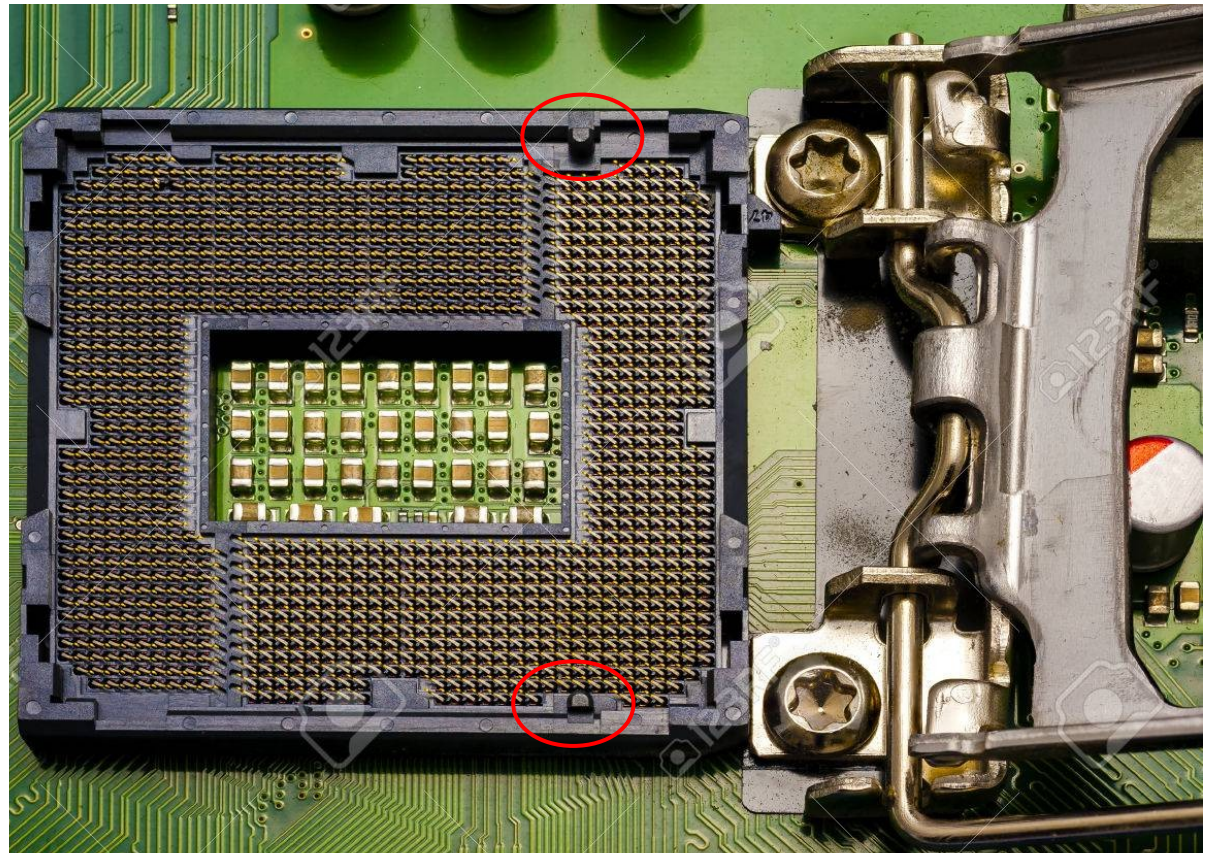
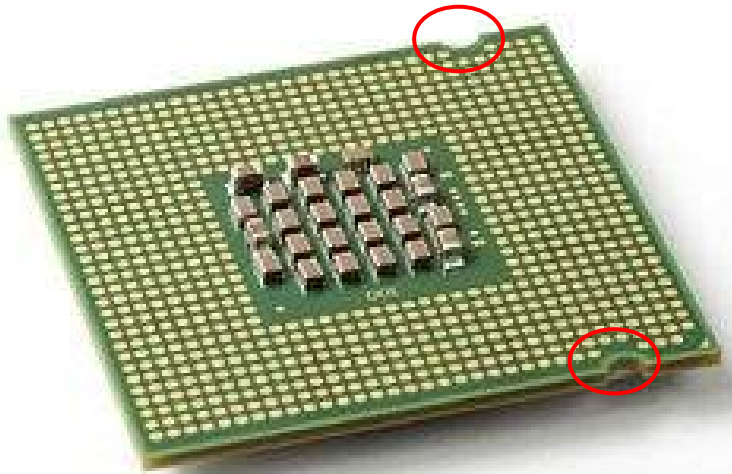
- Levée, elle permet l'insertion du processeur sans aucune pression
- Rabaisée, elle maintient le processeur sur son support.

Le processeur possède un détrompeur :

- un coin tronqué
- une marque de couleur

devant être aligné avec la marque correspondante sur le support.







Le processeur nécessite généralement un refroidissement adapté :

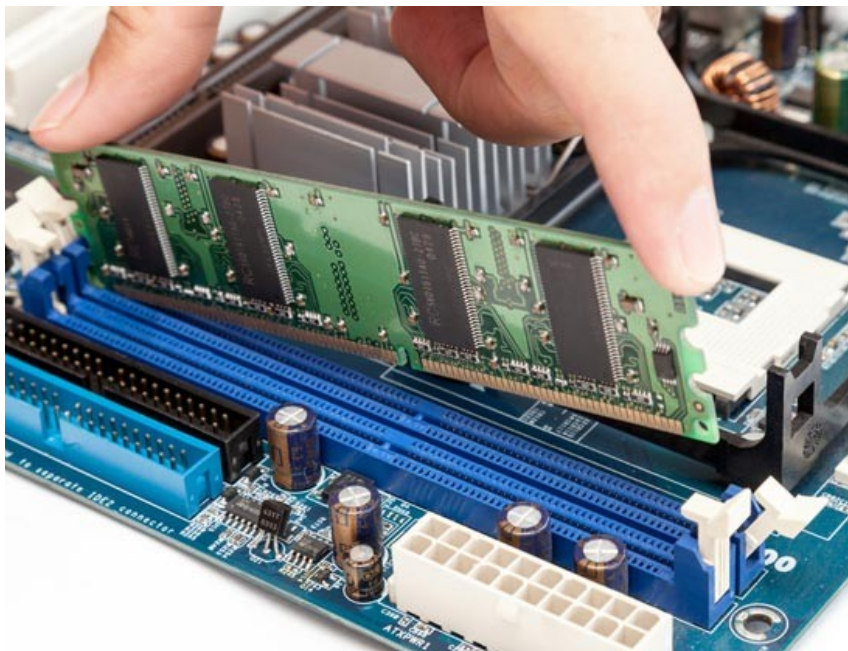
- Soit par air



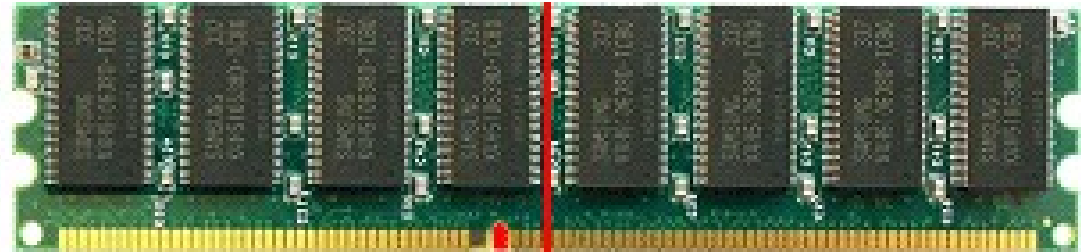
- Soit par « eau »



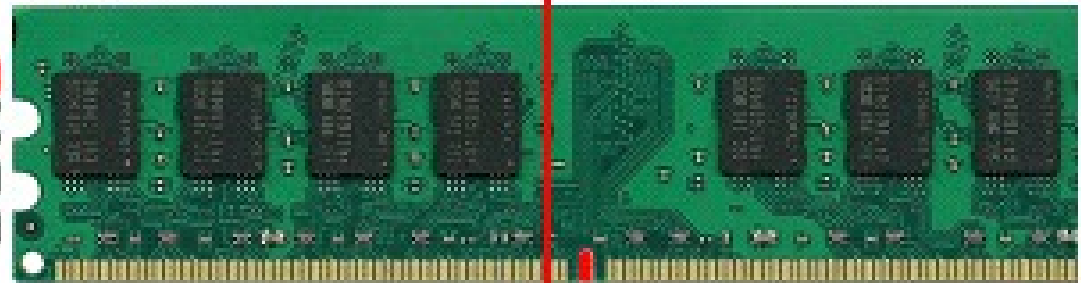
La mémoire vive se présente sous la forme de barrettes qui se branchent sur les connecteurs de la carte mère.



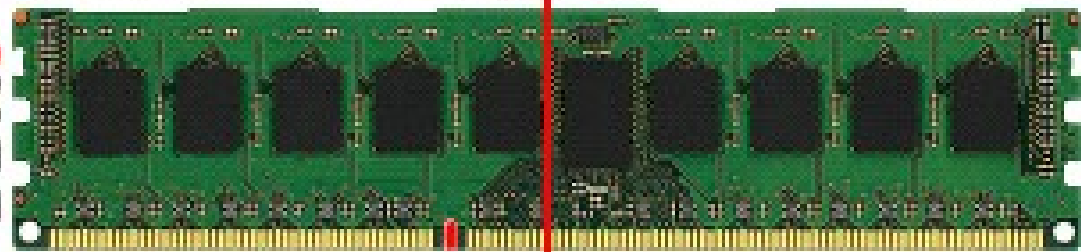
**DDR**



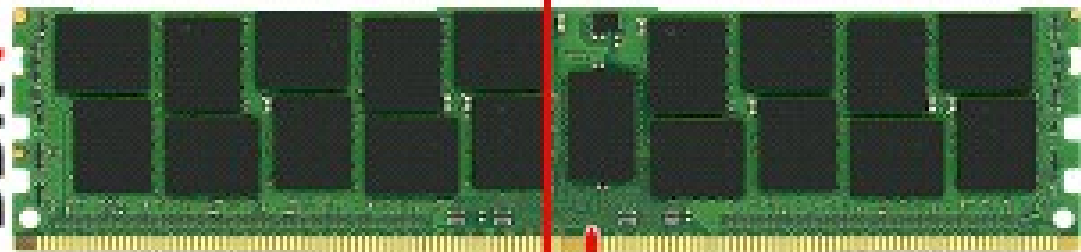
**DDR2**



**DDR3**

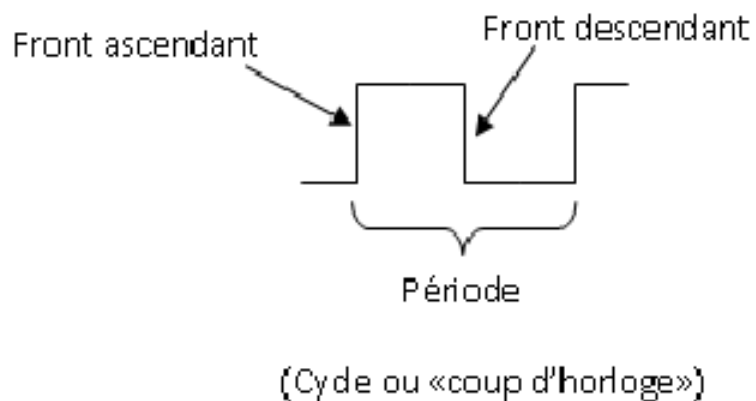


**DDR4**



Les différents composants peuvent discuter entre eux grâce à une horloge :

- On l'appelle RTC (**R**eal **T**ime **C**lock)
- C'est un cristal ou résonateur qui vibre à une certaine fréquence
  - les « tops » d'horloge
- Ces « tops » permettent de cadencer les composants



Éteindre l'ordinateur revient à couper l'alimentation de la carte mère

Lorsque l'ordinateur est remis sous tension, il est toujours à l'heure

Un circuit électronique appelé CMOS (**C**omplementary **M**etal-**O**xide **S**emiconductor) conserve les informations comme :

- l'heure et la date
- les informations du système
- ...



La carte mère est équipée d'un micrologiciel de configuration que l'on appelle :

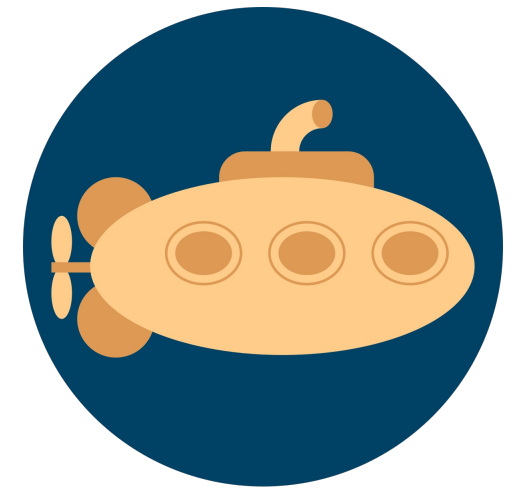
- BIOS (**B**asic **I**nput/**O**utput **S**ystem)
- EFI / UEFI (**U**nified **E**xtensible **F**irmware **I**nterface)



coreboot



PHOENIX  
TECHNOLOGIES



U-Boot



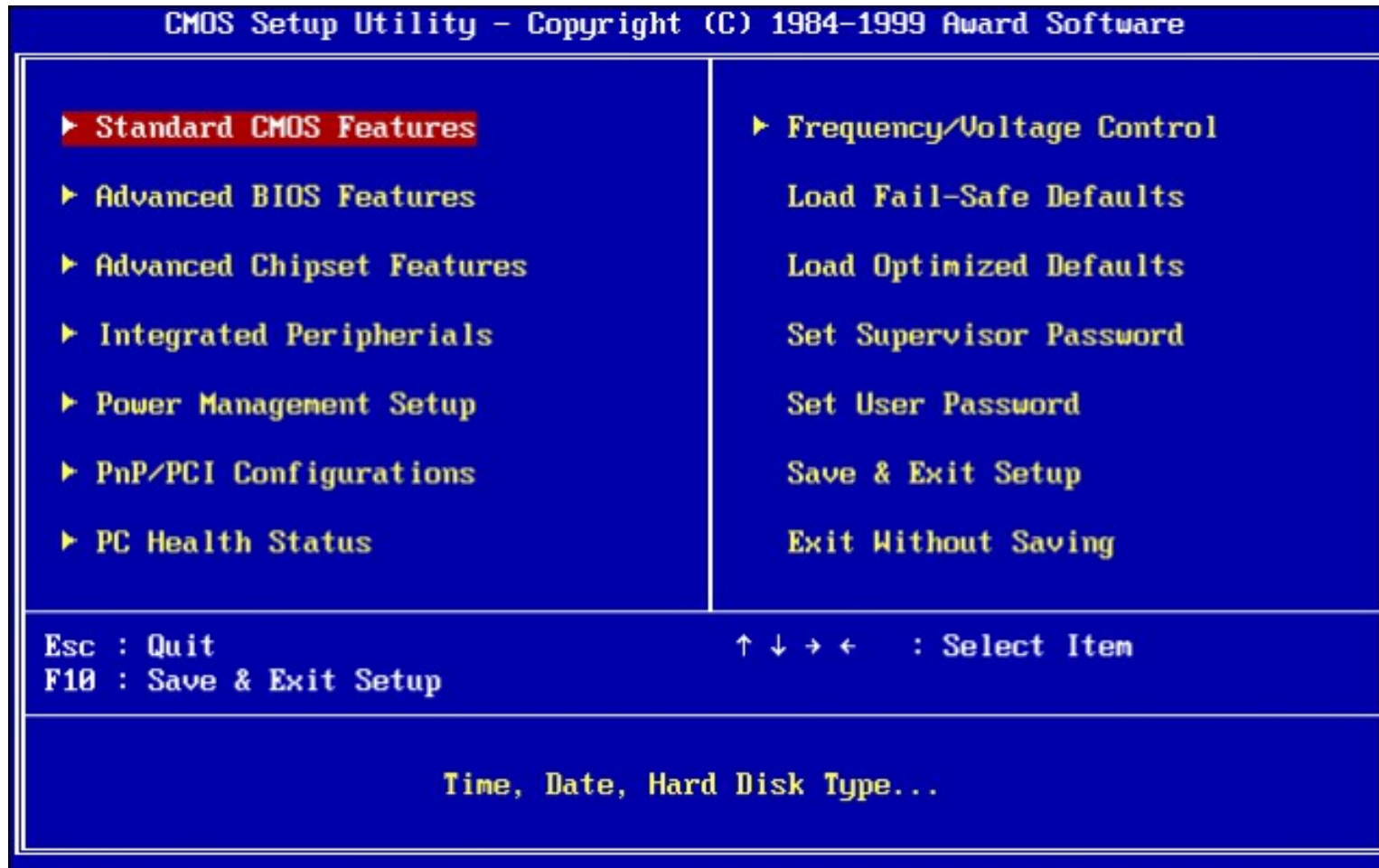
AWARD  
SOFTWARE  
INTERNATIONAL®  
INC.



Ce micrologiciel :

- est stocké sur une mémoire EEPROM (**E**lectrically-**E**rasable **P**rogrammable **R**ead-**O**nly **M**emory) ou flash
- permet de vérifier que tous les composants nécessaires sont présents → phase appelée **Power-On Self-Tests (POST)**
- utilise les données contenues dans le CMOS pour connaître la configuration matérielle
- Est configurable grâce à une interface accessible au démarrage du système (touche **Suppr**, **F2**, **F10** ou **F12**)

Voici un exemple de BIOS



Voici un exemple d'UEFI



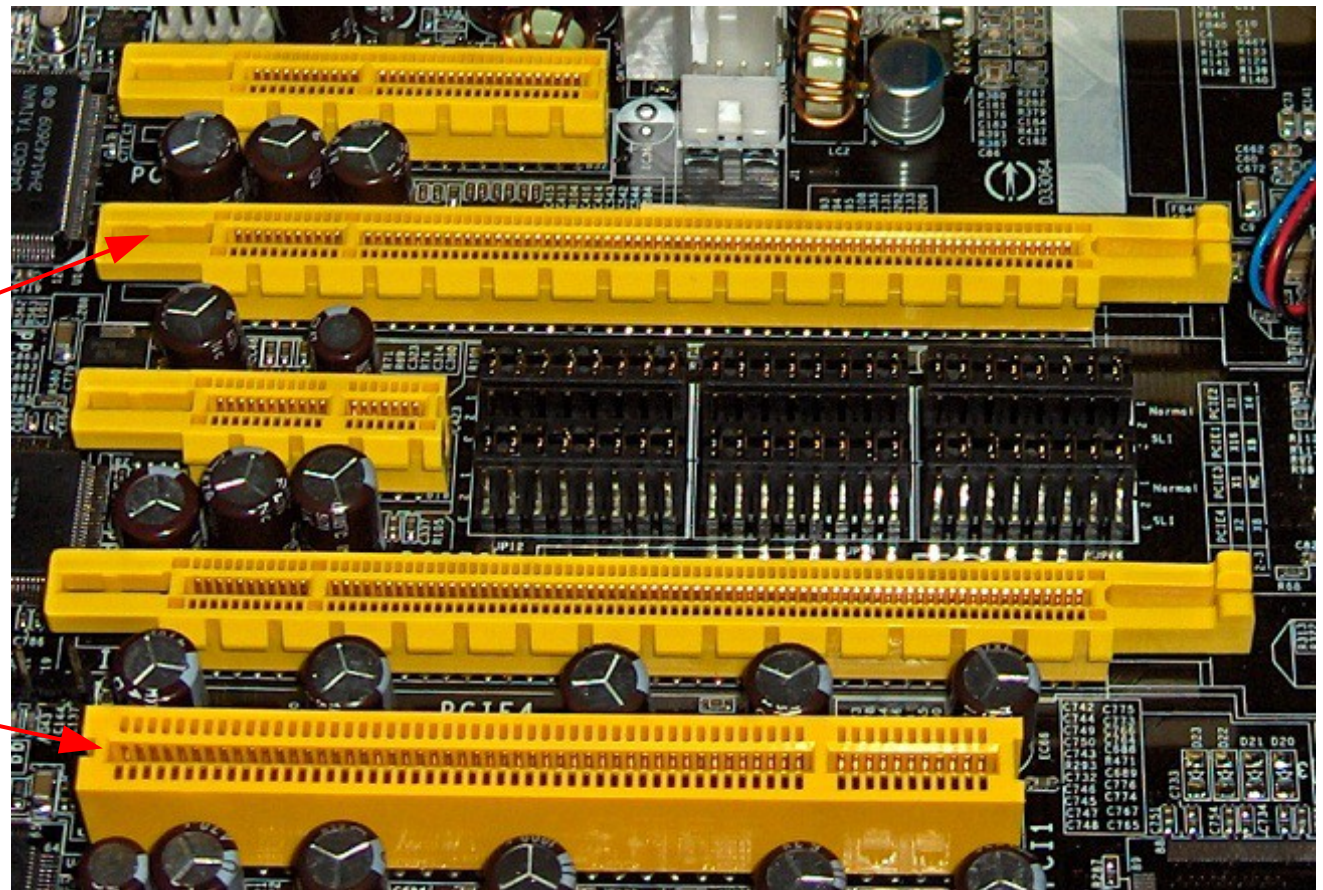


Les connecteurs d'extensions permettent de connecter des cartes additionnelles à la carte mère

Les plus utilisés sont les connecteurs PCI et PCI-Express

PCI-Express

PCI



Les cartes filles peuvent remplir différentes fonctions :

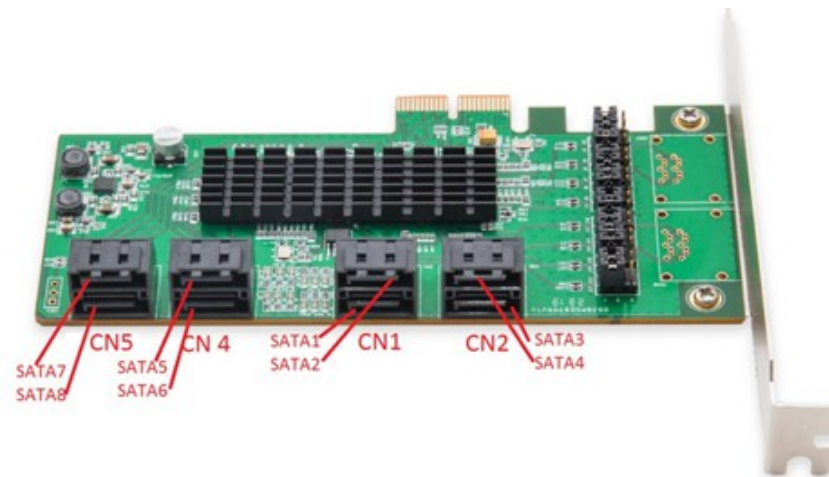
- Carte son



- Carte réseau



- Carte SATA (disques durs)



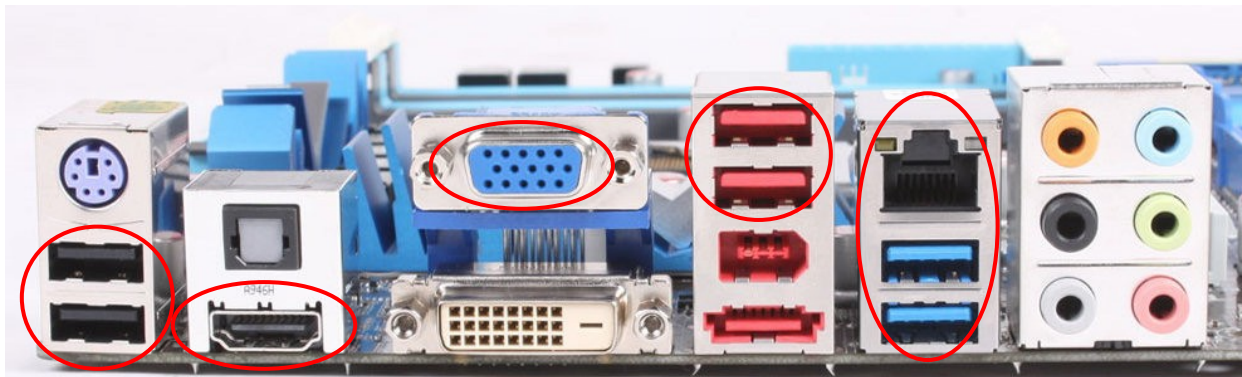
Les cartes qui profitent le plus de la bande passante du PCI-E restent les cartes graphiques...





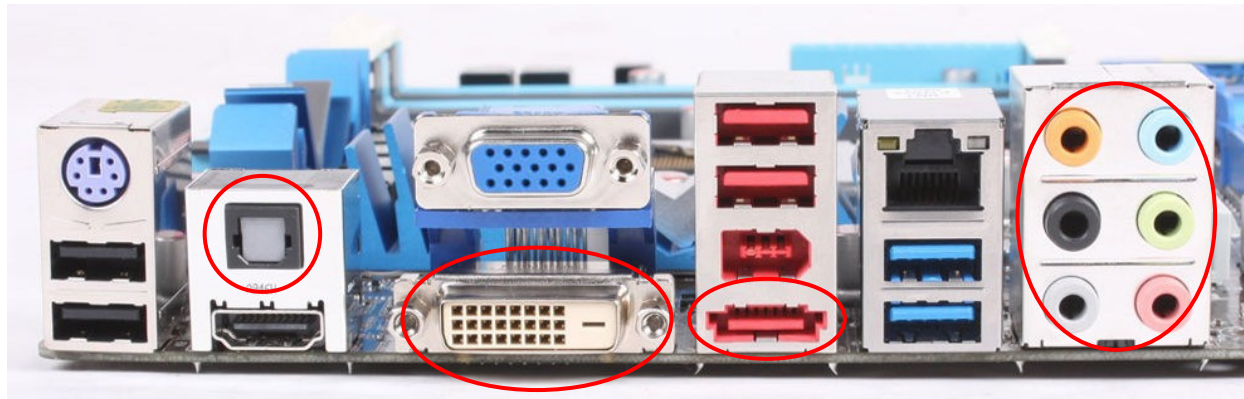
La carte mère possède un certain nombre de connecteurs d'entrées-sorties regroupés sur le « panneau arrière » :

- Ports USB (1.1, 2.0 ou 3.0), permettant de connecter des périphériques plus récents
- Connecteur RJ45 (appelés LAN ou port Ethernet) permettant de connecter l'ordinateur à un réseau
- Connecteur VGA (appelé SUB-D15), permettant de connecter un écran. Ce connecteur correspond à la carte graphique intégrée
- Connecteur HDMI (High Definition Multimedia Interface), permettant de connecter un écran ou téléviseur



La carte mère possède un certain nombre de connecteurs d'entrées-sorties regroupés sur le « panneau arrière » :

- Connecteur DVI (Digital Visual Interface), permettant de connecter un écran (ici DVI-D)
- Connecteur E-SATA (External-Serial Advanced Technology Attachment), permettant de relier un disque dur externe
- Connecteur S/PDIF (Sony/Phillips Digital Interface), permettant de brancher un kit d'enceintes ou un amplificateur de salon
- Prises audio, permettant de connecter des enceintes acoustiques ou une chaîne hi fi, ainsi qu'un microphone. Ce connecteur correspond à la carte son intégrée.

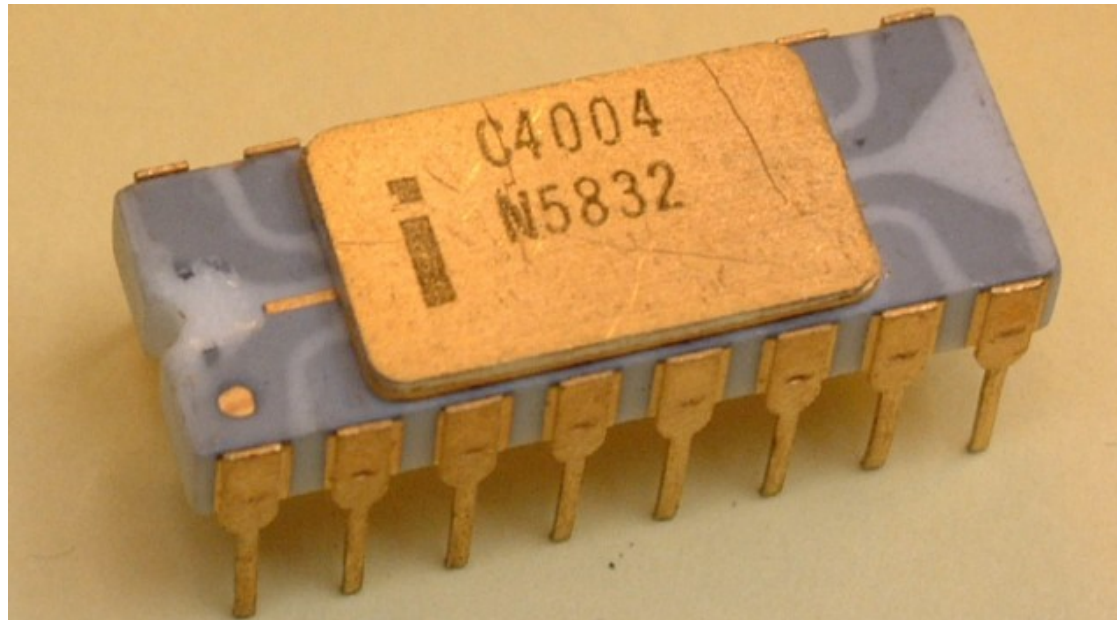


# **Le processeur**

Le processeur (noté CPU, pour Central Processing Unit) est un circuit électronique cadencé au rythme d'une horloge interne

Le premier microprocesseur (Intel 4004) a été inventé en 1971

Il s'agissait d'une unité de calcul de 4 bits, cadencé à 108 kHz



Le CPU manipule des informations codées sous forme binaire, et exécute des instructions stockées en mémoire

A chaque top d'horloge, le processeur exécute une action, correspondant à une instruction ou une partie d'instruction

Voici quelques unités de mesure :

- La fréquence d'horloge s'exprime en Hertz (Hz)
- Les CPI (**C**ycles **P**ar **I**nstruction) représentent le nombre moyen de cycles d'horloge nécessaires à l'exécution d'une instruction
- Les MIPS (**M**illions d'**I**nstructions **P**ar **S**econde) caractérisent la puissance du processeur →  $\text{MIPS} = \text{fréquence} / \text{CPI}$
- Les FLOPS (**F**loating point **O**perations **P**er **S**econd) permettent d'évaluer le nombre d'opérations en virgule flottante du FPU (Floating Point Unit)

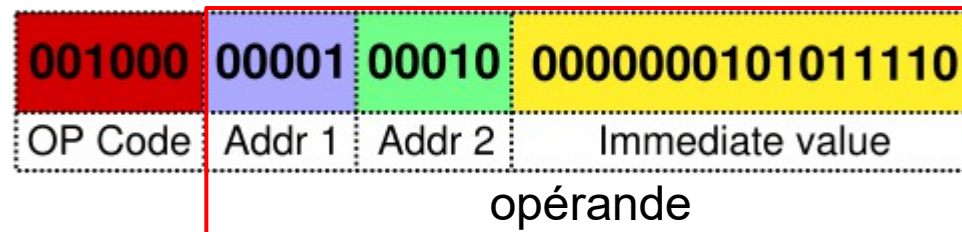


Une instruction est une opération élémentaire effectuée par le processeur.

Les instructions sont stockées dans la mémoire principale avant d'être traitées par le processeur.

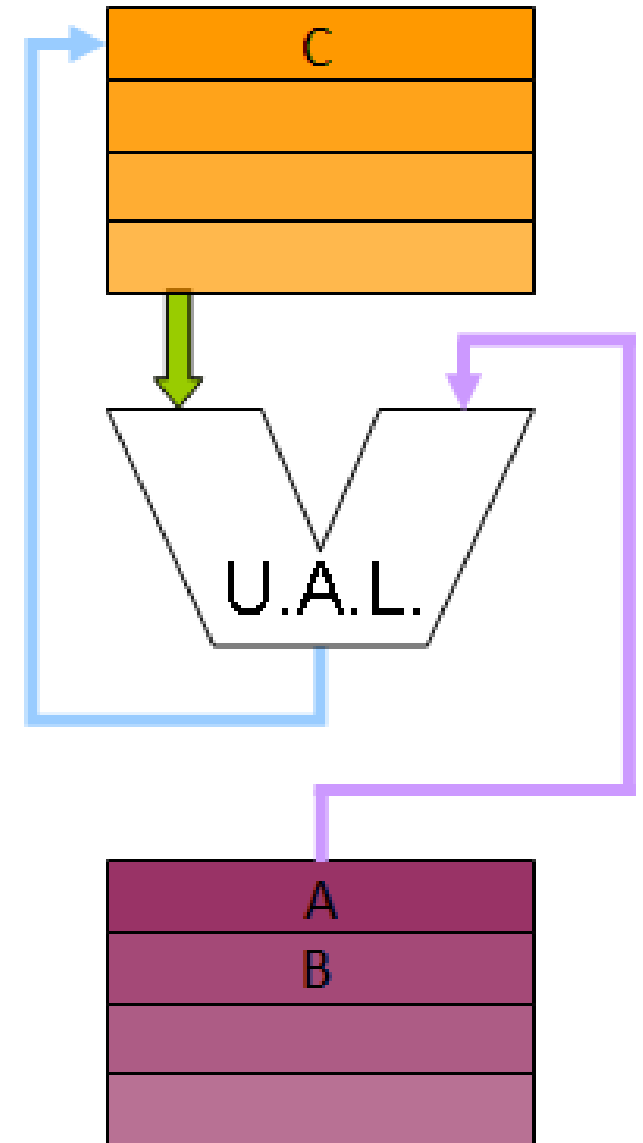
Une instruction est composée de deux parties :

- Opcode (*operation code*) : désigne l'opération à effectuer
  - Arithmétique (addition, soustraction, etc.)
  - Logique (AND, OR, XOR, ...)
  - Copie de données (push, pop, ...)
  - ...
- L'opérande : adresses mémoires sur lesquelles effectuer l'opération
  - Optionnel, dépend du type de processeur



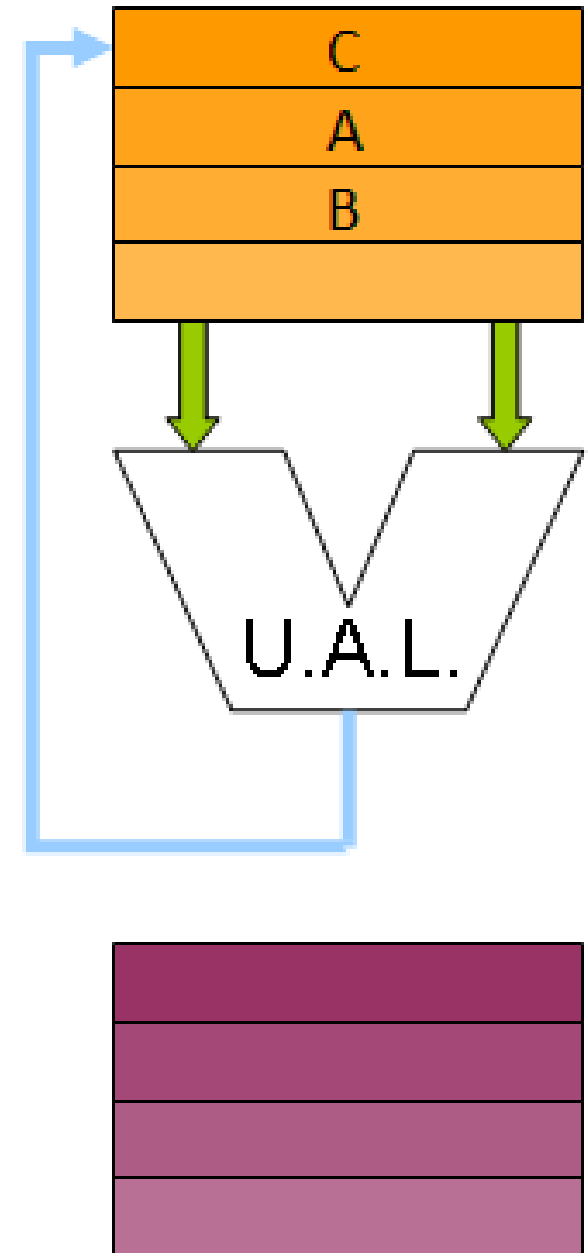
Prenons l'opération suivante :  $C = A + B$

- Avec un CPU à 2 opérandes, l'opération demande 2 instructions :
  - move A to C  
déplace la valeur de A dans l'emplacement mémoire C
  - add B to C  
additionne la valeur de B dans l'emplacement mémoire A



Prenons l'opération suivante :  $C = A + B$

- Avec un CPU à 3 opérandes, l'opération demande 1 instruction :  
→ add a,b,c

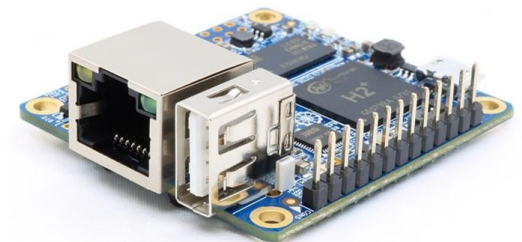


Le nombre d'opérandes dépend de l'architecture et du nombre de bits du CPU. Voici des exemples :

2 opérandes → les architectures AVR en 16 bits



3 opérandes → les architectures ARM en 32 bits



L'unité de contrôle décode les informations arrivant et les envoie à l'unité de d'exécution.

L'unité d'exécution effectue les calculs grâce à :

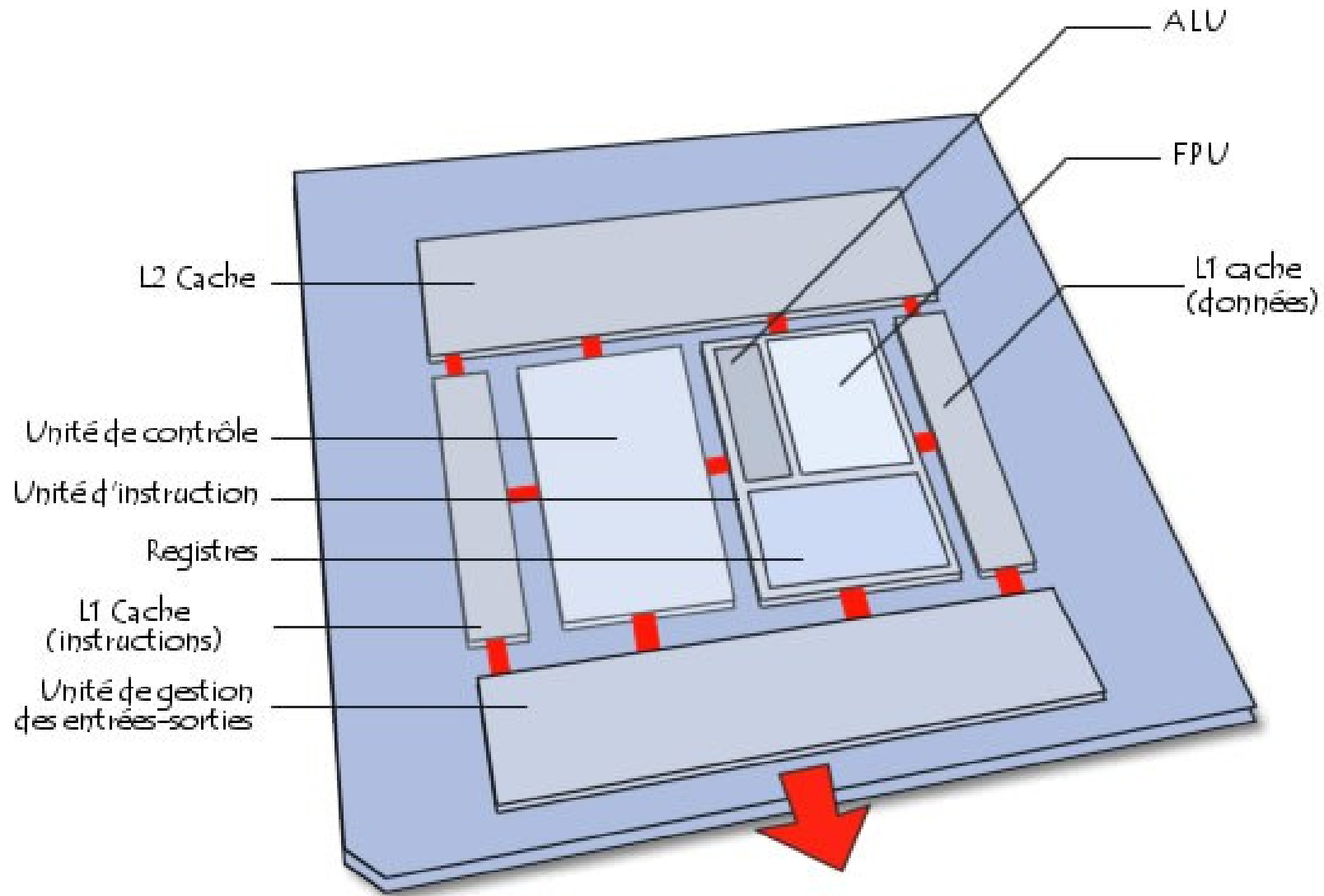
- l'Unité Arithmétique et Logique (UAL ou ALU) pour les calculs basiques arithmétiques et les opérations logiques
- la FPU pour les calculs complexes non entiers

Les données sont stockées dans deux types de mémoires :

- Les registres
  - très petites (8, 16, 32 ou 64 bits)
  - sont très rapides et très chères
- Les mémoires caches
  - plus grosses que le registre = plus de stockage
  - moins rapides mais moins chères que les registres

L'unité de gestion des entrées-sorties gère le flux d'information de la mémoire centrale

On peut schématiser le CPU de la manière suivante :



Les traitements peuvent être accélérés par des jeux d'instructions.

Le jeu d'instruction d'un processeur détermine son architecture.

Le processeur travaille grâce à un nombre limité de fonctions câblées sur circuits électroniques.

La plupart des opérations peuvent être réalisées à l'aide de fonctions basiques.

Certaines architectures incluent des fonctions évoluées dans le processeur tel que **MMX** (**M**ulti**M**edia e**X**tension), **3DNow !** / **SSE** (**S**treaming **S**IMD e**X**tension), etc.

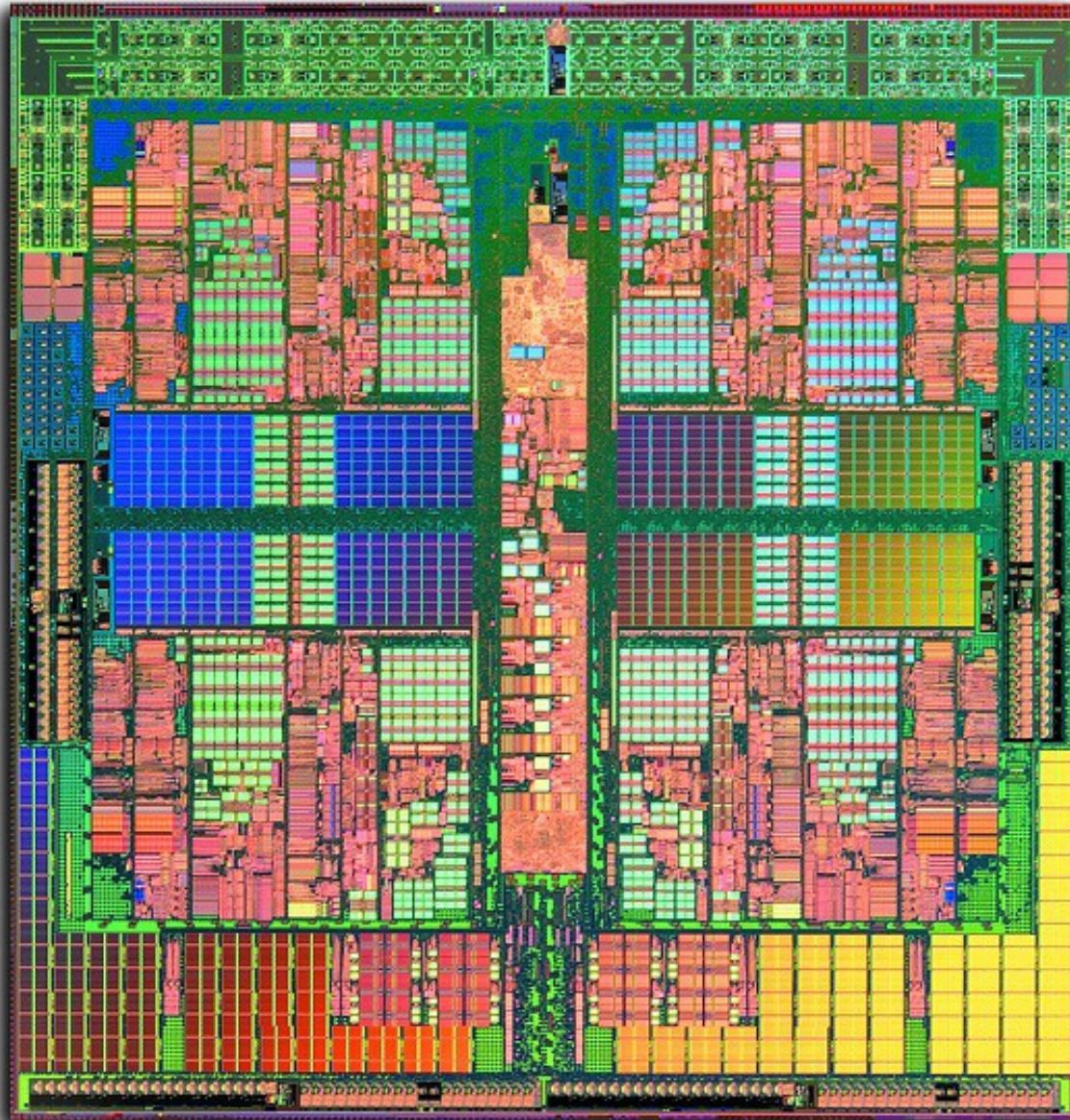
On peut classer les CPU en fonction de la complexité de leurs jeux d'instructions :

- **CISC (Complex Instruction Set Computer)**
  - Consiste à câbler des instructions complexes
  - Coût élevé car fonctions évoluées (gravure complexe)
  - Instructions de longueurs variables = plusieurs cycles sont nécessaires pour le traitement
- **RISC (Reduced Instruction Set Computer)**
  - Aucune fonction évoluée câblée
  - Les programmes doivent être traduits en instructions simples
  - Une instruction = un cycle d'horloge
  - Coût de fabrication réduit



Les processeurs sont capables de traiter plusieurs instructions en même temps : c'est le parallélisme de l'exécution :

- HyperThreading
  - Définit deux processeurs logiques pour un processeur logique
  - Deux threads simultanés = SMT (**S**imultaneous **M**ulti **T**hreading)
  - C'est une « supercherie » pour envoyer les données en masse au CPU
- Multi-coeurs
  - Processeur composé de plusieurs unités de calcul
  - Il faut que les logiciels utilisent plusieurs cœurs pour que le gain de performance soit visible



Die d'un processeur quad-core AMD  
Opteron

# La mémoire

Il existe deux types de mémoires :

- La mémoire centrale :
  - Également appelée mémoire vive
  - Elle permet de stocker les données d'exécution des programmes
  - Se vide quand on éteint l'ordinateur
  
- La mémoire de masse :
  - Également appelée mémoire physique ou externe
  - Elle permet de stocker les informations à long terme
  - Elle correspond à un disque dur, CD, clé USB, ...

Une mémoire possède les caractéristiques suivantes :

- La capacité : volume global d'informations (en bits) que la mémoire peut stocker
- Le temps d'accès : intervalle de temps entre la demande de lecture/écriture et la disponibilité de la donnée
- Le temps de cycle : intervalle de temps minimum entre deux accès successifs
- Le débit : volume d'information échangé par unité de temps
- La volatilité : aptitude d'une mémoire à conserver les données lorsqu'elle n'est plus alimentée électriquement.

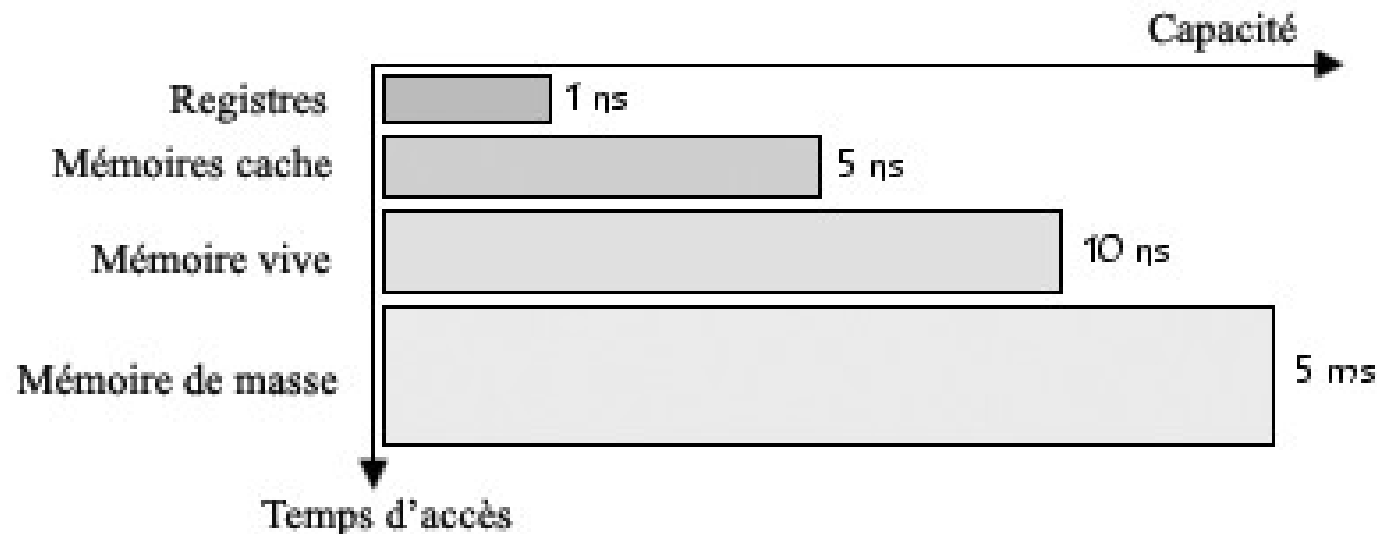
Les mémoires rapides sont les plus onéreuses.

C'est pourquoi des mémoires différentes sont utilisées dans un ordinateur.

Elles sont interfacées les unes avec les autres et organisées de façon hiérarchique.

Les plus rapides se trouvent à proximité du processeur.

Les moins rapides servent à stocker les informations de manière permanente.



Il existe plusieurs types de mémoire :

- La mémoire vive ou RAM (**R**andom **A**ccess **M**emory)
  - Stocke de manière temporaire les données lors de l'exécution d'un programme
  - Elle est volatile
- La mémoire morte (ROM ou EEPROM)
  - Elle est non volatile mais très lente
  - En lecture seule mais possible de "réécrire" dessus (EEPROM)
- La mémoire FLASH
  - Compromis entre RAM et ROM
  - Utilisée dans les clés USB

# Le disque dur



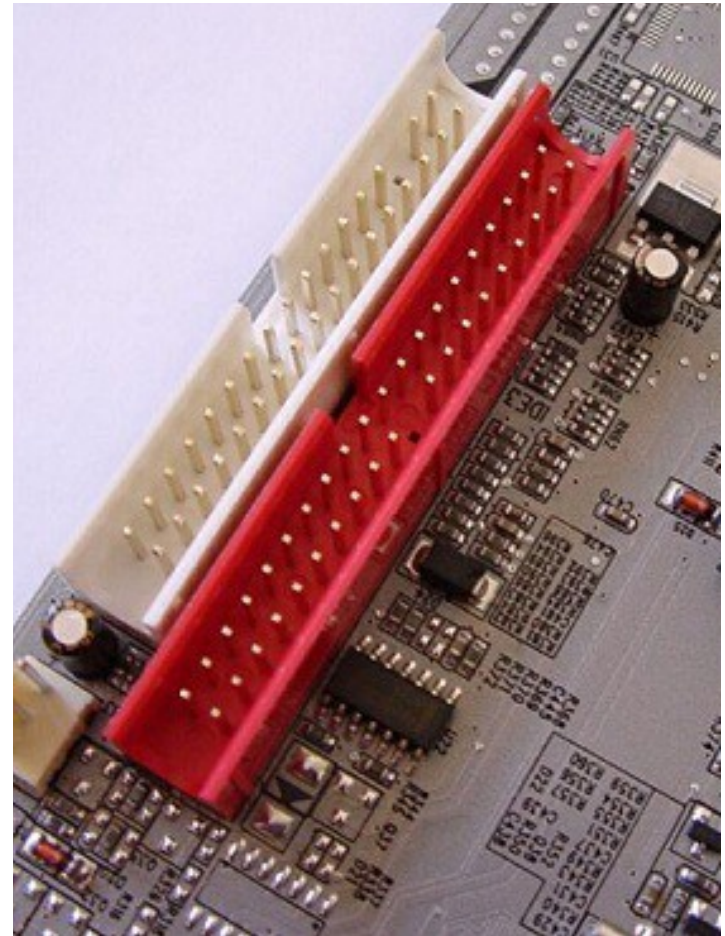
- Conserver les données de manière permanente contrairement à la mémoire vive
- Relié à la carte-mère par l'intermédiaire d'un contrôleur de disque dur faisant l'interface entre le processeur et le disque dur
- Le contrôleur :
  - gère les disques qui lui sont reliés
  - interprète les commandes envoyées par le processeur
  - achemine les données au disque concerné

On distingue plusieurs technologies :

- PATA (**P**arallel **A**dvanced **T**echnology **A**ttachment)

ou

- IDE (**I**ntegrated **D**rive **E**lectronics)



On distingue plusieurs technologies :

- SCSI (**S**mall **C**omputer **S**ystem **I**nterface)



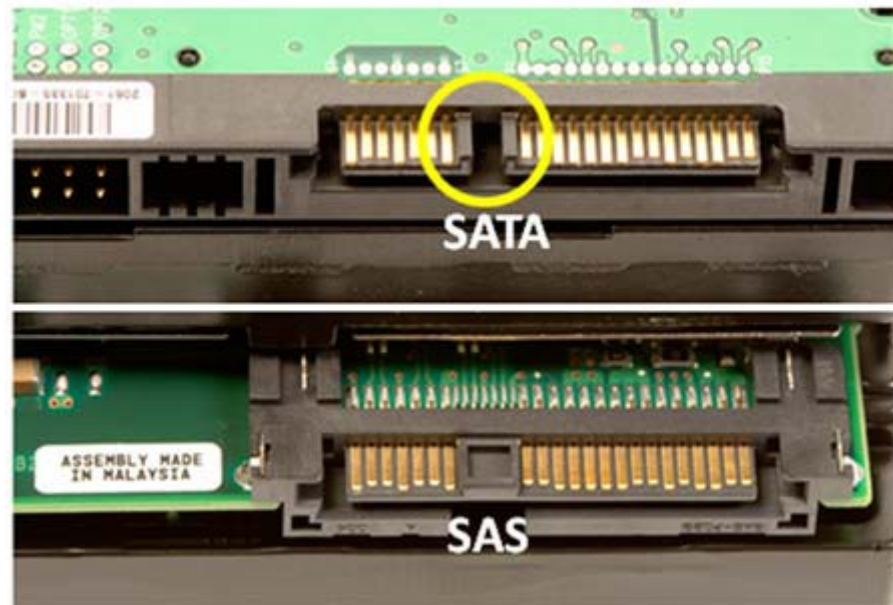
On distingue plusieurs technologies :

- SATA (**S**erial **A**dvanced **T**echnology **A**ttachment)



On distingue plusieurs technologies :

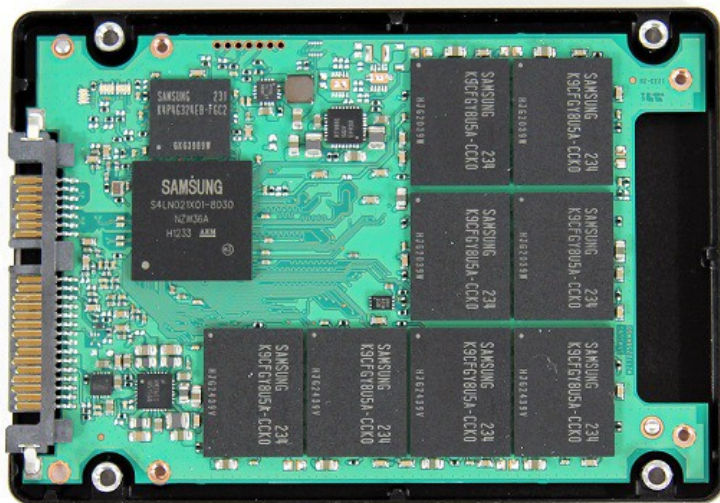
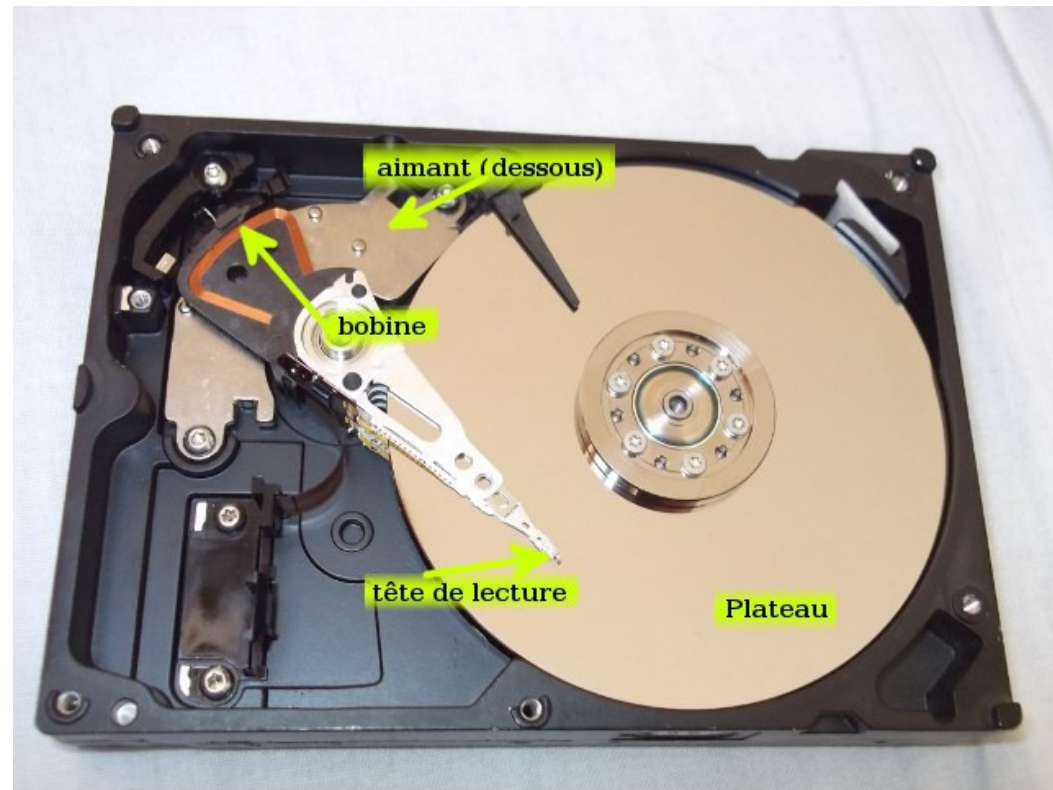
- SAS (**S**erial **A**ttached **S**CSI)





On distingue deux types de disques durs :

- Les HDD (**H**ard **D**isk **D**rive)
- Les SSD (**S**olid **S**tate **D**rive)





Voici le fonctionnement d'un HDD :

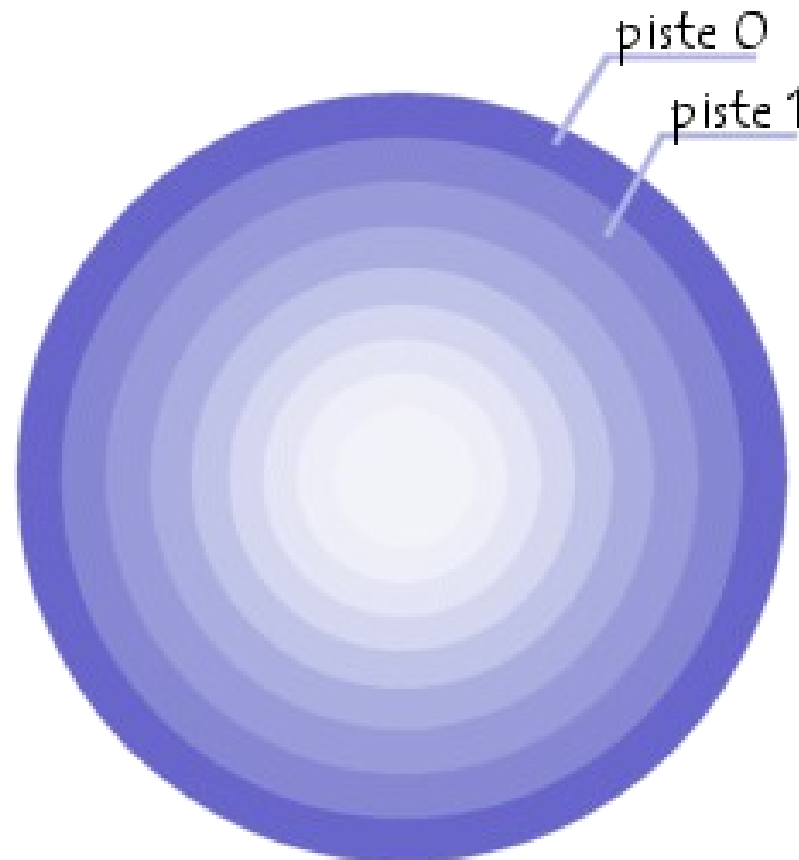
- Plusieurs disques rigides empilés à très faible distance les uns des autres appelés plateaux
- Ils tournent très rapidement autour d'un axe à plusieurs milliers de tours par minute
- Les bits sont stockés très proches les uns des autres sur une fine couche magnétique de quelques microns d'épaisseur
- La lecture et l'écriture se fait grâce à des têtes de lecture
- La rotation des plateaux génère un vent de 250 km/h qui maintient les têtes à quelques microns du plateau

- Si une ou plusieurs têtes entrent en contact avec la surface des plateaux, cela s'appelle un atterrissage
- Cela provoque la destruction des informations situées à cet endroit
- Une imperfection sur la surface telle qu'une poussière aura le même effet



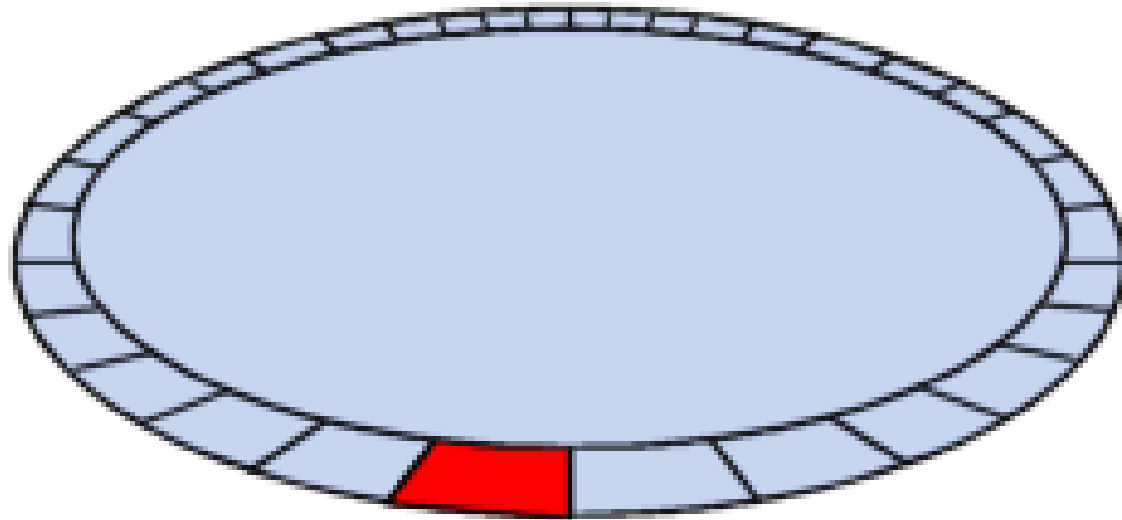
Un disque dur s'organise de la manière suivante :

- Les données sont organisées en cercles concentriques appelés « pistes », créées par le formatage de bas niveau.



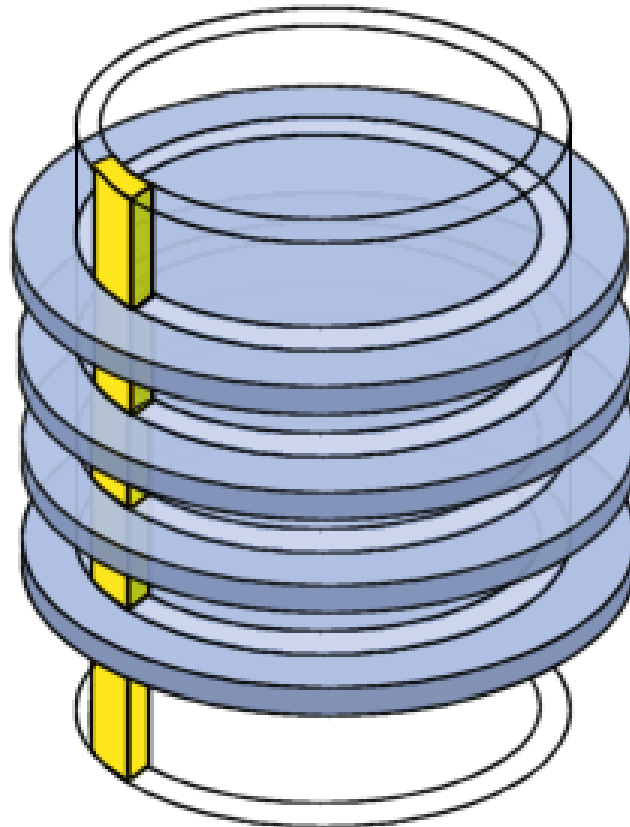
Un disque dur s'organise de la manière suivante :

- Les pistes sont séparées en quartiers que l'on appelle secteurs, contenant les données



Un disque dur s'organise de la manière suivante :

- On appelle cylindre l'ensemble des données situées sur une même piste sur des plateaux différents



Un disque dur s'organise de la manière suivante :

- Le système d'exploitation exploite des blocs formés de plusieurs secteurs appelés clusters
- Un cluster (unité d'allocation) est la zone minimale que peut occuper un fichier sur le disque
- Un fichier devra donc occuper plusieurs secteurs (un cluster).



Le SSD, quant à lui, stocke les données sur de la mémoire flash.

Cette mémoire est répartie sur plusieurs modules.

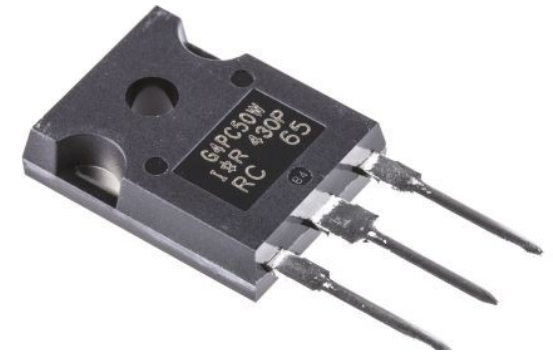
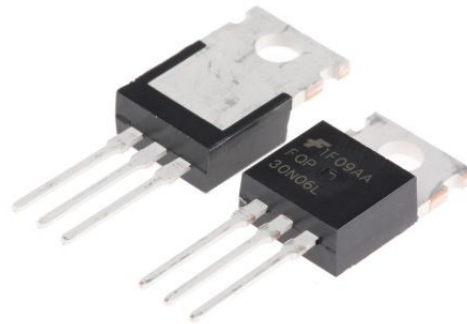
Ces modules sont pilotés par un contrôleur qui répartit les données de manière uniforme.

Les données échangées entre le système d'exploitation et la mémoire transitent par un buffer.

# Quelques composants

Notre ordinateur fonctionne seulement grâce à quelques composants :

- Le plus connu est le transistor (BJT, MOSFET, IGBT)

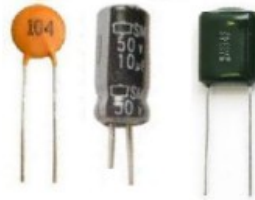


- Et bien sûr les composants basiques :

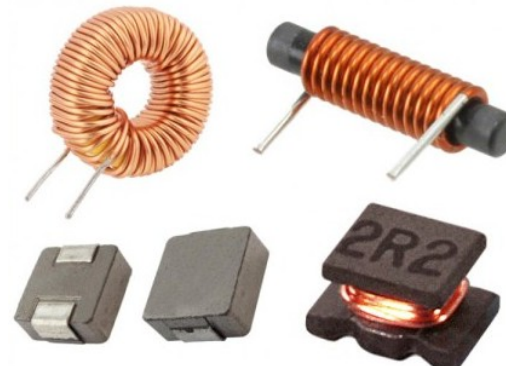
→ résistance



→ condensateur



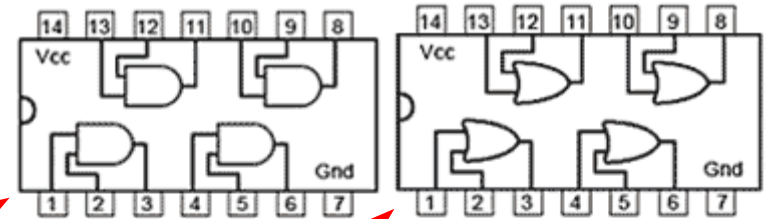
→ inductance (bobine)



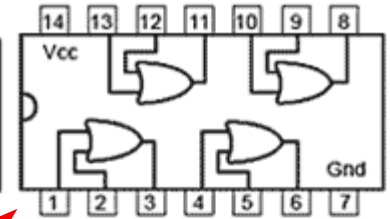
Certain de ces composants sont agencés de façon à réaliser des opérations simples. On appelle cela un circuit intégré

On peut les regrouper en différentes catégories :

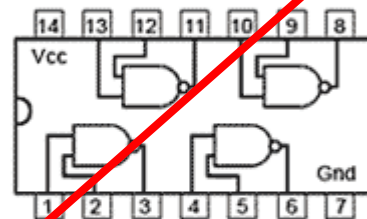
- Circuit de logique
- Puce mémoire
- Circuit de gestion de l'énergie
- ...



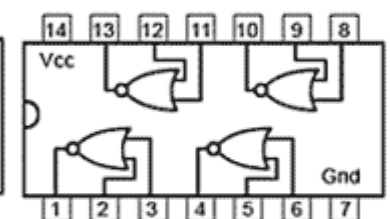
7408 Quad 2 input  
AND Gates



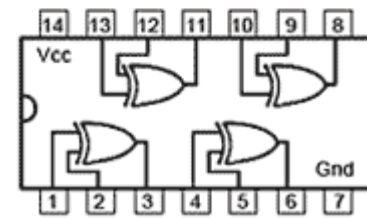
7432 Quad 2 input  
OR Gates



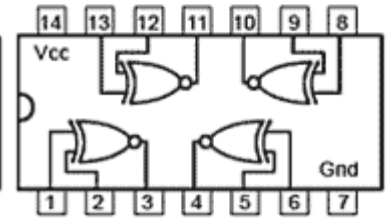
7400 Quad 2 input  
NAND Gates



7402 Quad 2 input  
NOR Gates



7486 Quad 2 input  
XOR Gates



747266 Quad 2 input  
XNOR Gates

Certain de ces circuits intégrés basiques sont agglomérés pour former un circuit intégré dédié à un traitement spécifique.

C'est ce que l'on appelle un ASIC (**A**pplication **S**pecific **I**ntegrated **C**ircuit)



ASIC pour faire de l'Ethernet



ASIC pour miner du bitcoin



ASIC pour décoder des MP3



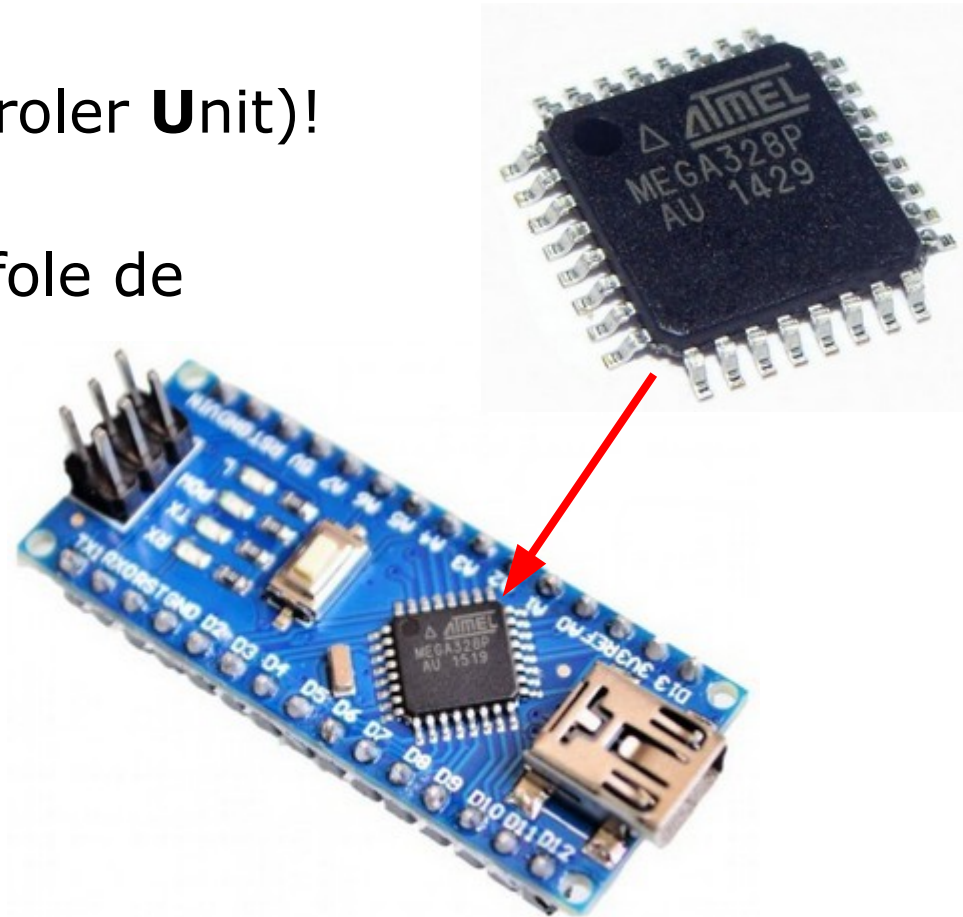
Quand le circuit intégré embarque :

- Une unité de calcul (CPU)
- De la mémoire RAM (pour l'exécution du programme)
- De la flash (pour stocker le programme)

on appelle cela un MCU (**M**icro**c**ontroler **U**nit)!

Le monde des objets connectés raffole de ce type de circuit imprimé.

Le plus connu est certainement l'Arduino...



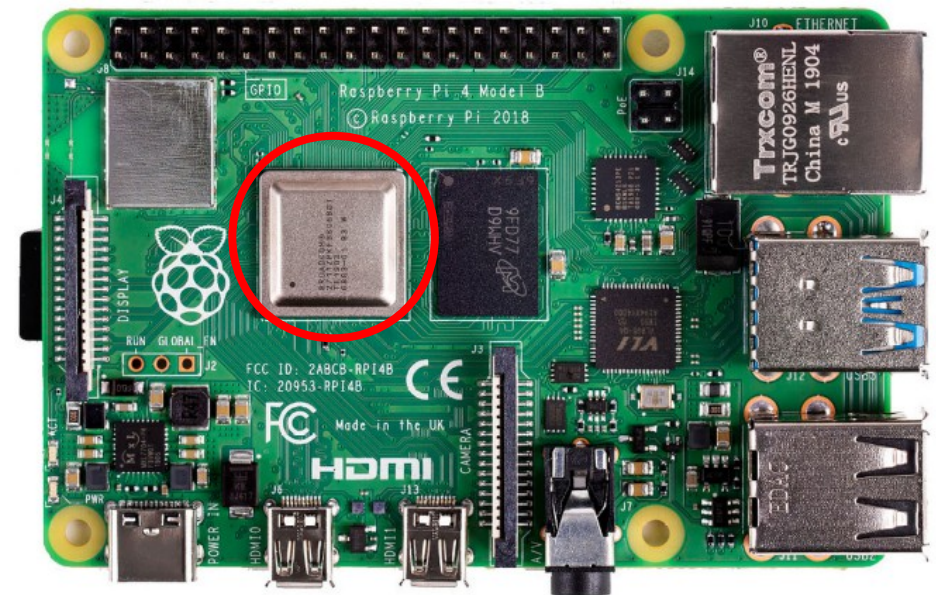


Quand le circuit intégré embarque tous les composants d'un ordinateur :

- Un CPU
- Un contrôleur de mémoire
- Un contrôleur de stockage
- Un contrôleur d'entrée / sortie (USB, Ethernet, HDMI, ...)

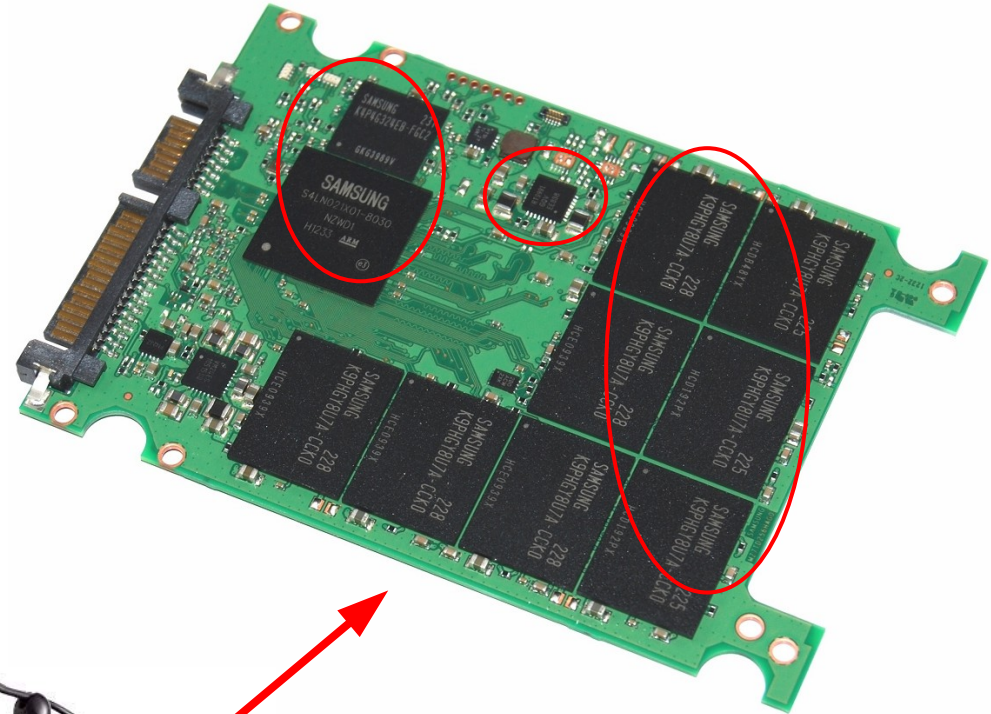
On appelle cela un SoC (**S**ystem **O**n a **C**hip) !

Le plus connu est certainement le Raspberry Pi...



# Cas pratique

Si on analyse l'architecture de notre SSD, on peut « comprendre » son fonctionnement :



- Ces puces servent au stockage des données (K9PHGY8U7A-CCK0) :
  - Gravure 20nm
  - 64 Go de densité
  - Flash de type NAND
- Cette puce sert de tampon (K4P4G324EB-FGC2) :
  - Flash de type DDR2 (533MHz)
  - 1 Go de densité
- Voici le processeur du SSD (S4LN021X01-8030) :
  - Architecture ARM Cortex R4
  - Cadencé à 300Mhz
  - 3 cœurs (lecture / écriture / optimisation)





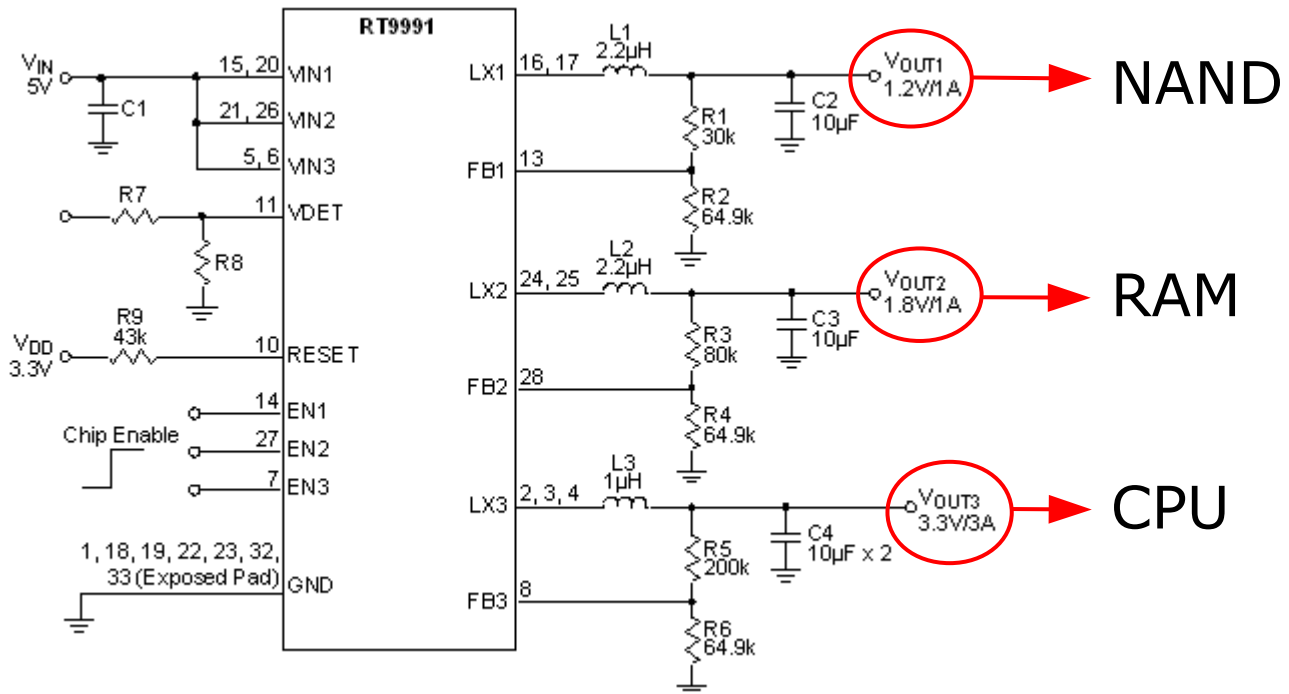
- Ce circuit imprimé sert à la gestion de l'énergie pour les SSD

(RT9991) :

- 3 étages d'alimentation indépendants
- Consommation minimale ( $70\mu\text{A}$ )
- Découpe jusqu'à 2MHz

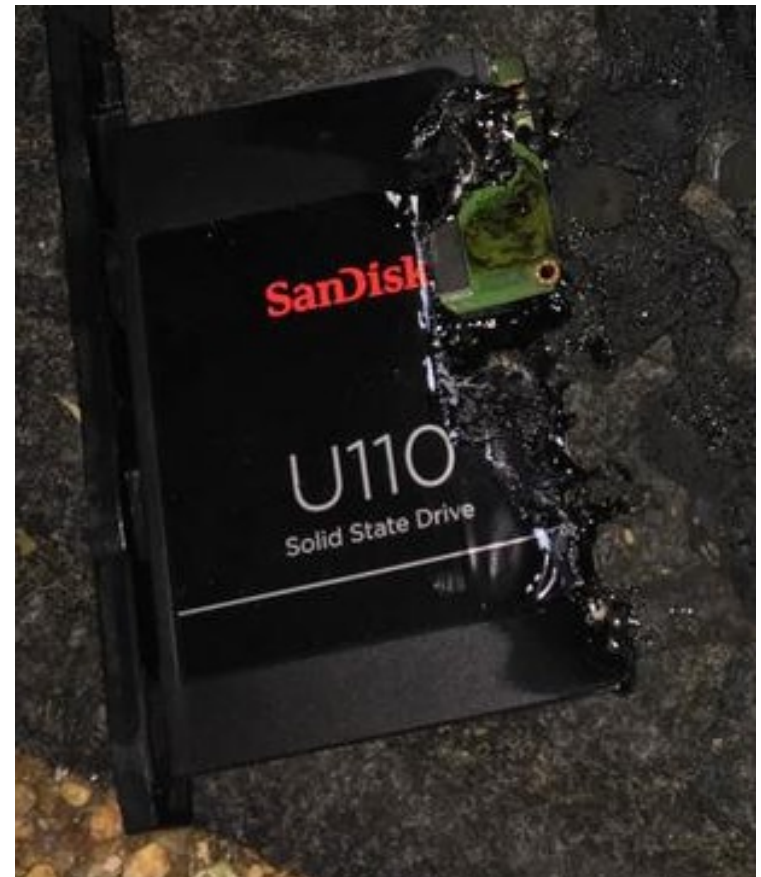


On peut retrouver son diagramme sur Internet



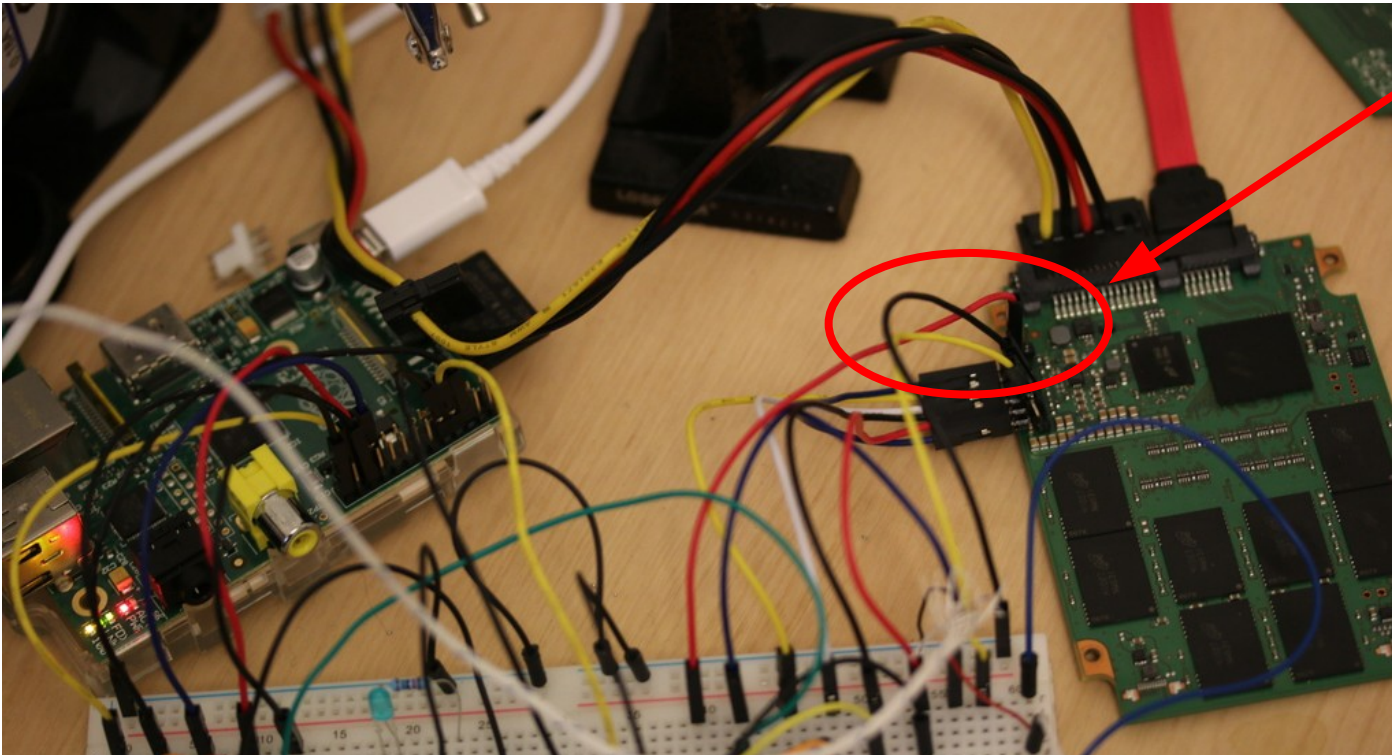
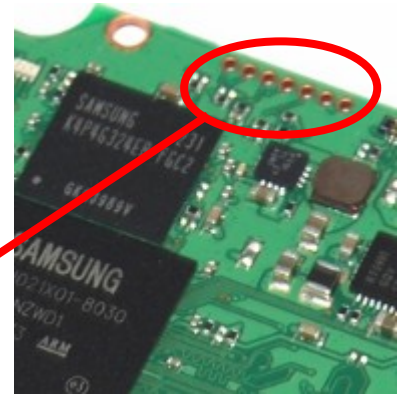
Les SSD ont une durée de vie limitée par rapport aux HDD.

Que faire pour ne pas perdre ces données ?!?



Tous les microprocesseurs ont un port JTAG (**J**oint **T**est **A**ction **G**roup) pour pouvoir envoyer des commandes directement au processeur.

On pourrait utiliser un debugger JTAG mais... comme cela coûte cher, on va plutôt utiliser un Raspberry Pi avec un "petit morceau de code".





Il ne reste plus qu'à envoyer des instructions au processeur

```
> cpu0 arm disassemble 0x8009abbc 16
0x8009abbc 0xe7d01004 LDRB r1, [r0, r4]
0x8009abc0 0xe7d62004 LDRB r2, [r6, r4]
0x8009abc4 0xe1510002 CMP r1, r2
0x8009abc8 0xe2841001 ADD r1, r4, #0x1
0x8009abcc 0x13a05000 MOVNE r5, #0x0
0x8009abd0 0xe20140ff AND r4, r1, #0xff
0x8009abd4 0xe354001f CMP r4, #0x1f
0x8009abd8 0x3affffff7 BCC 0x8009abbc
0x8009abdc 0xe3550001 CMP r5, #0x1
0x8009abe0 0x159f000c LDRNE r0, [r15, #0xc]
0x8009abe4 0x1bfe990f PLNE 0x80041028
0x8009abe8 0xe1a00005 MOV r0, r5
0x8009abec 0xe28dd044 ADD r13, r13, #0x44
0x8009abf0 0xe8bd80f0 LDM r13!, {r4, r5, r6, r7, r15}
0x8009abf4 0x801d4038 ANDHIS r4, r13, r8, LSR r0
0x8009abf8 0xe92d43fe STMDB r13!, {r1, r2, r3, r4, r5, r6, r7, r8, r9, r14}
> cpu0 mmw 0x8009abe4 0xe3a05001
> cpu0 arm disassemble 0x8009abbc 16
0x8009abbc 0xe7d01004 LDRB r1, [r0, r4]
0x8009abc0 0xe7d62004 LDRB r2, [r6, r4]
0x8009abc4 0xe1510002 CMP r1, r2
0x8009abc8 0xe2841001 ADD r1, r4, #0x1
0x8009abcc 0x13a05000 MOVNE r5, #0x0
0x8009abd0 0xe20140ff AND r4, r1, #0xff
0x8009abd4 0xe354001f CMP r4, #0x1f
0x8009abd8 0x3affffff7 BCC 0x8009abbc
0x8009abdc 0xe3550001 CMP r5, #0x1
0x8009abe0 0x159f000c LDRNE r0, [r15, #0xc]
0x8009abe4 0xe3a05001 MOV r5, #0x1
0x8009abe8 0xe1a00005 MOV r0, r5
0x8009abec 0xe28dd044 ADD r13, r13, #0x44
0x8009abf0 0xe8bd80f0 LDM r13!, {r4, r5, r6, r7, r15}
0x8009abf4 0x801d4038 ANDHIS r4, r13, r8, LSR r0
0x8009abf8 0xe92d43fe STMDB r13!, {r1, r2, r3, r4, r5, r6, r7, r8, r9, r14}
> resume
```

On en reconnaît certaines :

AND → ET logique

MOV → déplace

ADD → addition