

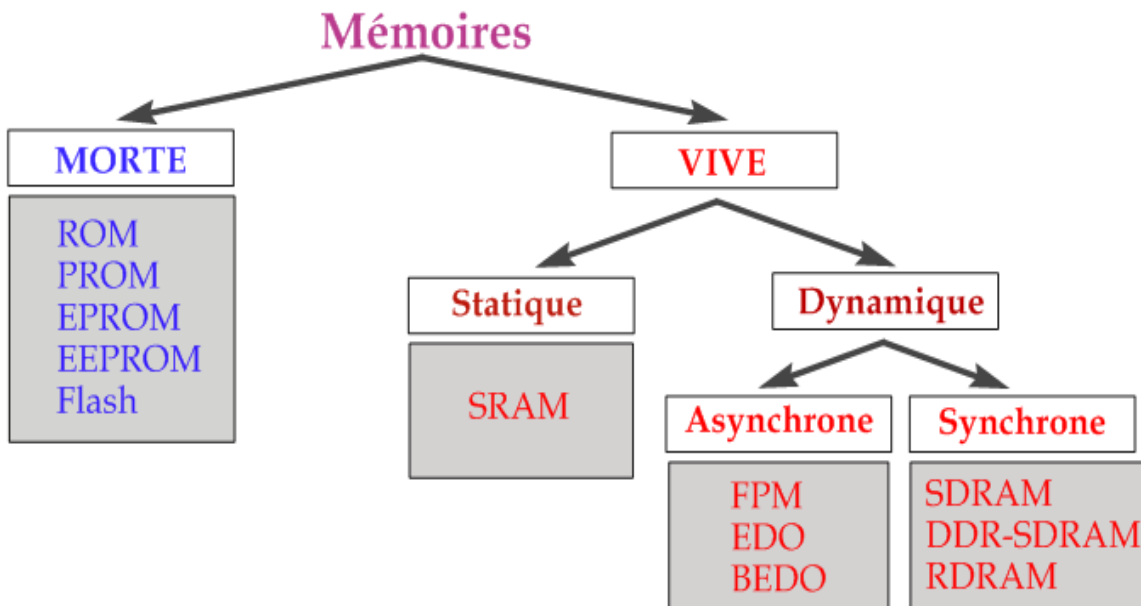
# Mémoires

## 1. Introduction

La mémoire est le deuxième composant en importance dans un ordinateur. Lorsque les programmes sont exécutés, ils sont d'abord chargés en mémoire afin d'être traités. De plus, lorsque vous éditez un texte, celui-ci est situé dans la mémoire au moment de l'édition. Chaque tâche effectuée sur un ordinateur doit temporairement enregistrer des informations en mémoire.

## 2. Classification

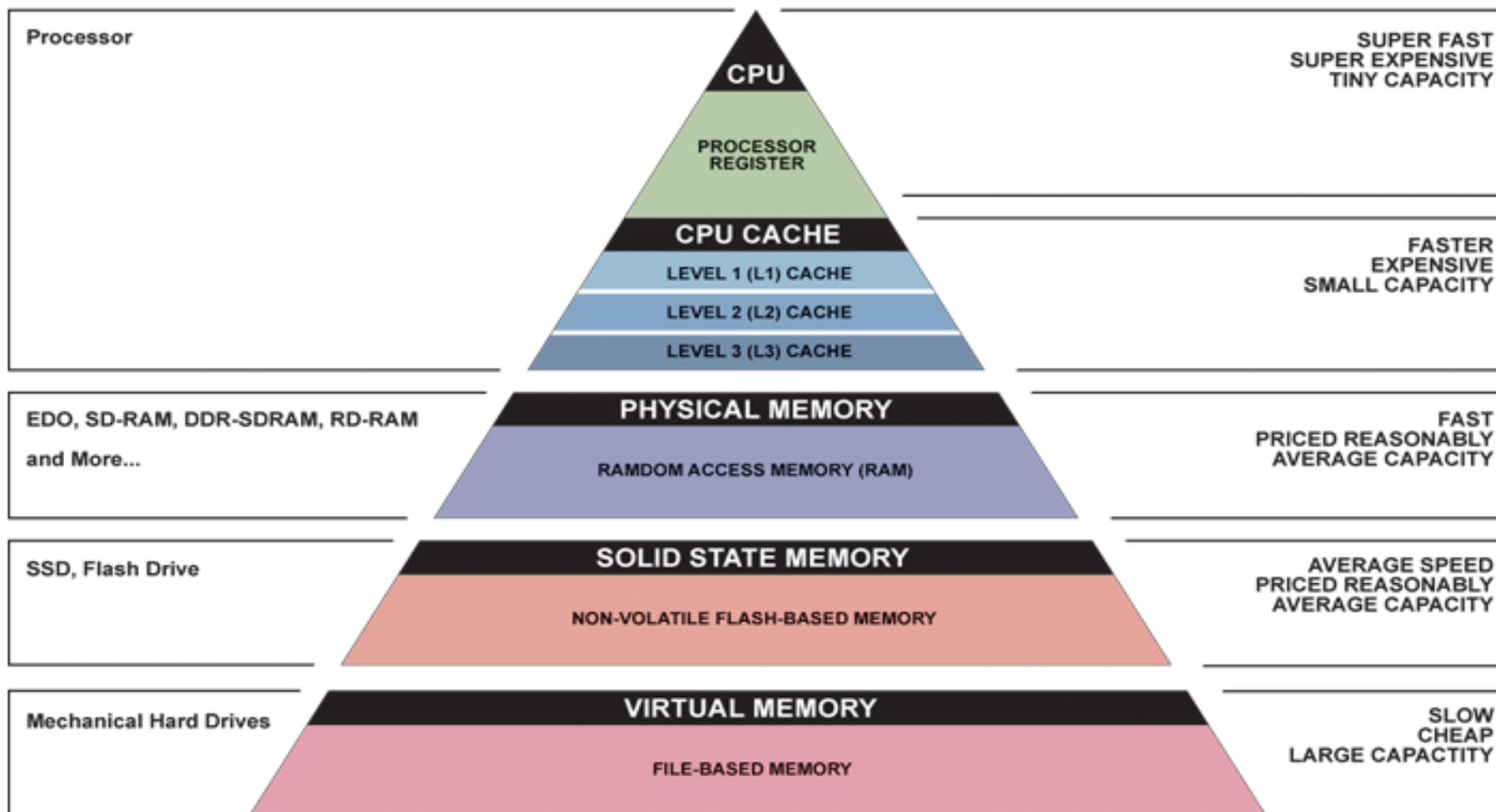
Les différentes mémoires peuvent être classées comme indiqué sur le schéma ci-dessous :



On distingue la mémoire vive (ou RAM) de la mémoire morte (ou ROM).

On peut également donner une autre classification qui tient compte de l'éloignement par rapport au processeur :

- les **registres** sont les éléments de mémoire les plus rapides. Ils sont situés au niveau du processeur et servent au stockage des opérandes et résultats intermédiaires.
- la **mémoire cache** est une mémoire rapide de faible capacité destinée à accélérer l'accès à la mémoire centrale en stockant les données les plus utilisées.
- la **mémoire centrale** contient les programmes (code + données) et est plus lente que les deux mémoires précédentes.
- La **mémoire d'appui** est l'équivalent de la mémoire cache pour la mémoire de masse.
- la **mémoire de masse** est un support de stockage généralement de grande capacité et sert au stockage et à l'archivage des informations.



▲ Simplified Computer Memory Hierarchy  
Illustration: Ryan J. Leng

### 3. La mémoire vive : RAM

La mémoire vive ou RAM (**Random Access Memory**) pour *mémoire à accès aléatoire*, est une mémoire volatile, cela signifie que si l'on coupe l'alimentation, les données qu'elle contient sont perdues. Ses caractéristiques sont les suivantes :

- elle sert à stocker les programmes exécutés par le processeur
- elle est accessible en lecture et en écriture
- elle est organisée sous forme matricielle

Il existe deux grandes familles de mémoires vives :

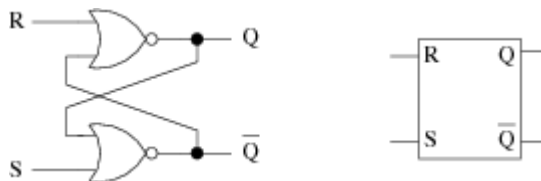
1. les RAM **Statiques** : SRAM
2. les RAM **Dynamiques** : DRAM

Caractéristiques des mémoires

Type de mémoire	Temps d'accès
SRAM	6 à 25 ns
DRAM	60 à 120 ns
Disque Dur	100 à 200 ms

#### 3.1. La mémoire vive statique : SRAM

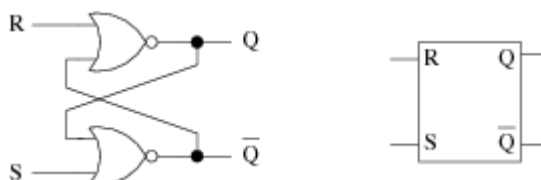
- Chaque bit d'une **SRAM** est formé par une **bascule (latch)** constituée par 4 à 6 transistors. L'information stockée peut être maintenue sans dégradation pendant une centaine d'heures. L'intérêt de ce type de mémoire est sa vitesse (quelques ns) mais son coût est prohibitif.
- une bascule est composée de deux **portes NOR** constituée chacune de deux transistors (soit un total de 4 transistors pour un bit d'information). En conséquence on utilisera la SRAM lorsque le facteur vitesse est critique et notamment pour des mémoires de petite taille comme la mémoire cache.



Bascule RS constituée de 2 portes NOR

La bascule RS comporte :

- deux entrées notées **R** (Reset) et **S** (Set)
- deux sorties désignées par **Q** et  $\neg Q$  qui ont normalement des valeurs inverses.



En fonctionnement normal les entrées **R** et **S** sont en position **basse** (ou à 0). On doit alors mettre une des entrées en position **haute** (1) pour changer l'état de la bascule. L'étude de la bascule RS montre que :

- la mise à 1 momentanée de **S** force **Q** à 1 et  $\neg Q$  à 0
- la mise à 1 momentanée de **R** force **Q** à 0 et  $\neg Q$  à 1
- la mise à 1 momentanée de **R** et **S** provoque une situation indésirable. Cela provoque la mise à 0 de **Q** et  $\neg Q$ . Lorsque **S** et **R** reviennent en position basse l'état de la sortie est déterminé par la porte qui revient à 0 en premier.

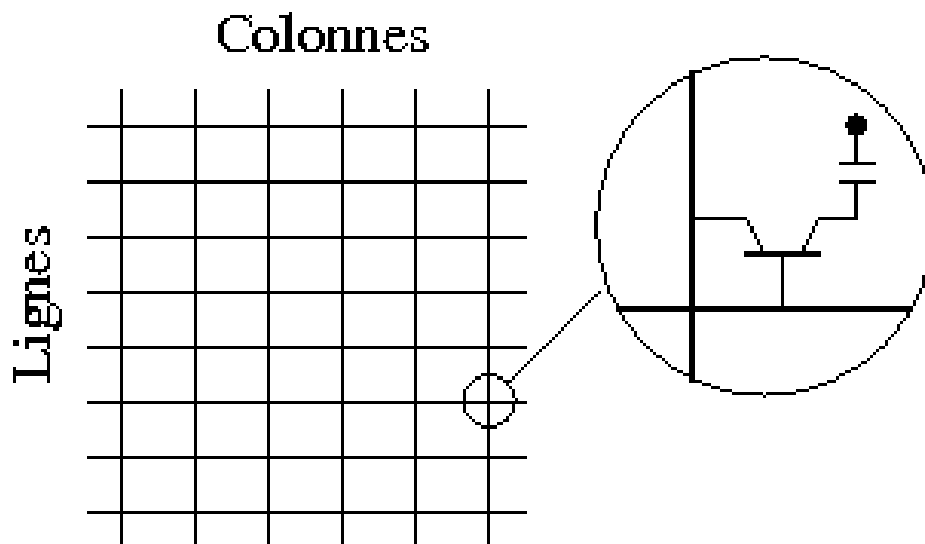
On supposera que le cas **R = S = 1**, qui consiste à vouloir mettre la bascule à 0 et à 1 ne se produira jamais.

On peut résumer le fonctionnement de la bascule RS par le tableau suivant :

<b>S</b>	<b>R</b>	Sortie
0	0	inchangée
0	1	<b>Q = 0</b>
1	0	<b>Q = 1</b>
1	1	erreur

### 3.2. La mémoire vive dynamique : DRAM

- Chaque bit d'une DRAM est constitué par un **transistor** et un **condensateur** :
  - le condensateur détermine la valeur du bit : le bit vaut 1 si le condensateur est chargé, il vaut 0 dans le cas contraire.
  - le transistor gère l'accès au condensateur



- L'inconvénient des DRAM est que le condensateur possède une tendance naturelle à se décharger. Pour que l'information reste cohérente, on va devoir réaliser un **rafraîchissement** de la mémoire toutes les quelques millisecondes. Ce rafraîchissement consiste à lire et à réécrire la donnée.
- Les avantages de la DRAM sont :
  - sa grande densité d'intégration (car un bit est représenté par un transistor),
  - son faible coût de fabrication,
  - sa faible consommation électrique (entre un sixième à la moitié de celle des SRAM).

- En revanche le temps d'accès de la DRAM est plus important que celui de la SRAM (environ 50-70 ns).
- Il existe deux types de mémoires dynamiques :

### **3.2.1. Mémoires synchrones et asynchrones**

---

- 3.2.1.1. **Mémoire asynchrone** : pour ce type de mémoire, l'intervalle de temps entre deux accès mémoire consécutif n'est pas régulier. Le processeur ne sait donc pas quand l'information qu'il attend est disponible et doit attendre (wait-state) que la mémoire lui transmette les données.
- 3.2.1.2. **Mémoire synchrone** : la cadence de sortie des informations est régulière, on évite ainsi les états d'attente (wait state) du processeur.

### **3.3. Comparaison SRAM / DRAM**

---

Étant donné les caractéristiques des SRAM et DRAM, on peut en déduire les propriétés suivantes :

- les DRAM sont plus lentes que les SRAM car durant le rafraîchissement on ne peut accéder aux données
- cependant, les DRAM ont une densité d'intégration plus grande que les SRAM : en effet le couple transistor + condensateur occupe moins de place que les 4 à 6 transistors des SRAM

En résumé :

- la SRAM est rapide mais chère
- la DRAM est lente mais bon marché

## **4. La mémoire morte : ROM**

- Les mémoires mortes ou **ROM (Read Only Memory)** sont des mémoires non volatiles qui sont vouées à être accédées en lecture en fonctionnement normal.
- Elles contiennent du code et des données qui ne sont pas amenés à changer souvent.
- Les ROM contiennent généralement les routines d'accès de base aux périphériques.

### **4.1. ROM classiques**

- L'information contenue dans ces mémoires est enregistrée de manière irréversible lors de la fabrication du circuit.
- Le principe de réalisation de ces mémoires est le réseau de diodes.

### **4.2. PROM**

- La mémoire **PROM (Programmable ROM)** est une ROM à enregistrement irréversible qui peut être programmée par un utilisateur grâce à un dispositif appelé programmeur de PROM.

### **4.3. EPROM et EEPROM**

- Les **EPROM (Erasable Programmable ROM)** présentent l'avantage de pouvoir être effacées et réécrites.
  - Pour effacer une EPROM il faut la soumettre à un rayonnement ultraviolet pendant 30 minutes.
  - Elles sont facilement reconnaissables en raison de la présence sur leur face supérieure d'une fenêtre de quartz obturée par un adhésif afin de les soustraire aux rayonnements ultraviolets naturels (soleil, néon).
- Les **EEPROM (Electrically EPROM)** quant à elles utilisent un procédé électrique pour l'effacement des données. Elles sont donc bien plus pratiques que les EPROM.

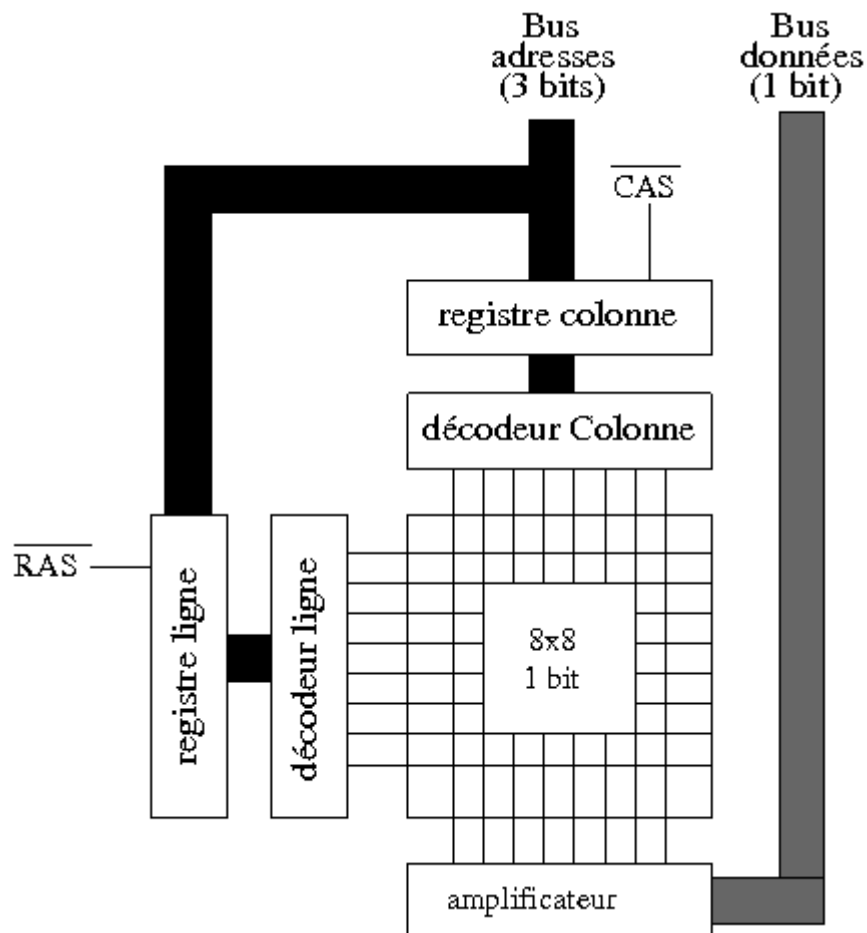
#### 4.4. Flash

- Les mémoires Flash réalisent la synthèse des technologies **EEPROM** et **SRAM**.
- Elles sont rapides, ne nécessitent pas de rafraîchissement.
- Elles sont effaçables et reprogrammables rapidement par blocs de 64 Ko.
- son prix de revient est assez élevé et on ne peut la reprogrammer qu'un nombre limité de fois (environ 100.000).

### 5. Organisation de la mémoire

#### 5.1. Composant mémoire

- Un composant mémoire peut être considéré comme un tableau à une dimension de cellules mémoires (1 bit).
- Cependant, il est plus intéressant d'implanter la mémoire sous forme d'une matrice de cellules (on dit également *mots*).
- Un composant mémoire est donc formé de :
  - une matrice composée de lignes **L** et de colonnes **C**,
  - à chaque intersection (ligne, colonne) on trouve des bascules (**SRAM**), ou des condensateurs-transistors (**DRAM**).

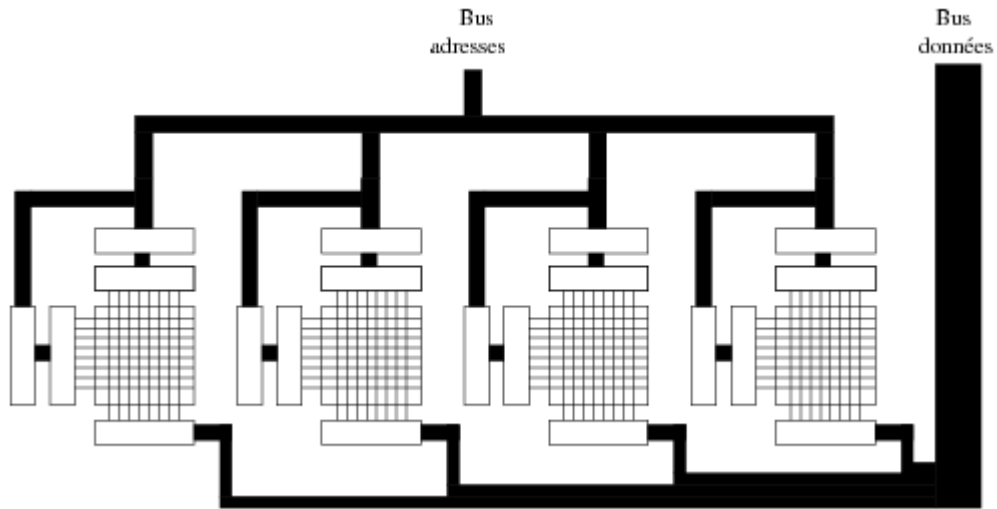


Organisation d'un composant mémoire

**Remarque :** un composant de  $128 \times 128$  cellules de 1 bit comporte 16384 bits, on note également :  $16\text{ K} \times 1$ .

## 5.2. Module mémoire

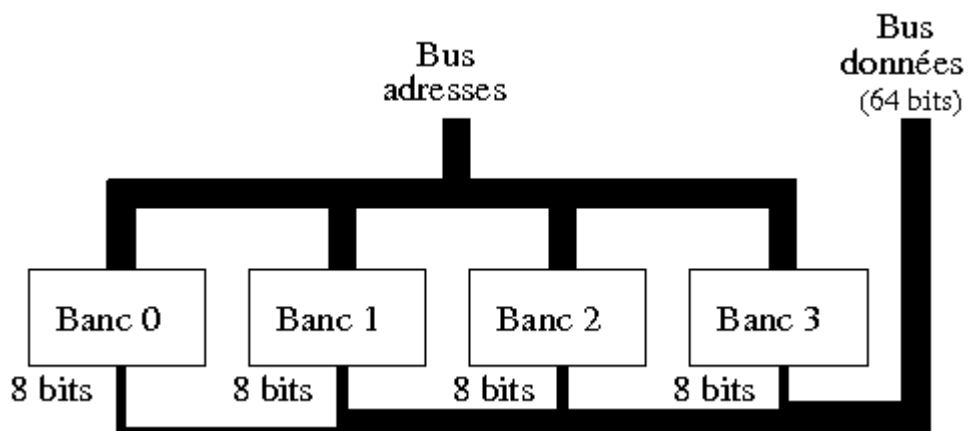
- L'unité de mémoire en informatique est l'octet (**8 bits**).
- Pour pouvoir constituer la mémoire en octets on met dans un même boîtier 8 composants 1 bit.
- Le schéma de la figure ci-dessous donne un exemple de boîtier 4 bits.



Organisation d'un module mémoire 4 bits

## 5.3. Barrette mémoire

- Les bus d'adresse des processeurs actuels sont de **64 bits** pour les Pentium (32 bits pour les 80486).
- Pour *remplir* le bus de données on utilise le même principe que pour les modules mémoires en utilisant des bancs mémoire.
- Pour une barrette composée de 4 bancs (ou modules) les octets sont stockés modulo 4 :
  - le banc 0 contient les octets 0, 4, 8, ...,
  - le banc 1 les octets 1, 5, 9, ...,
  - etc...



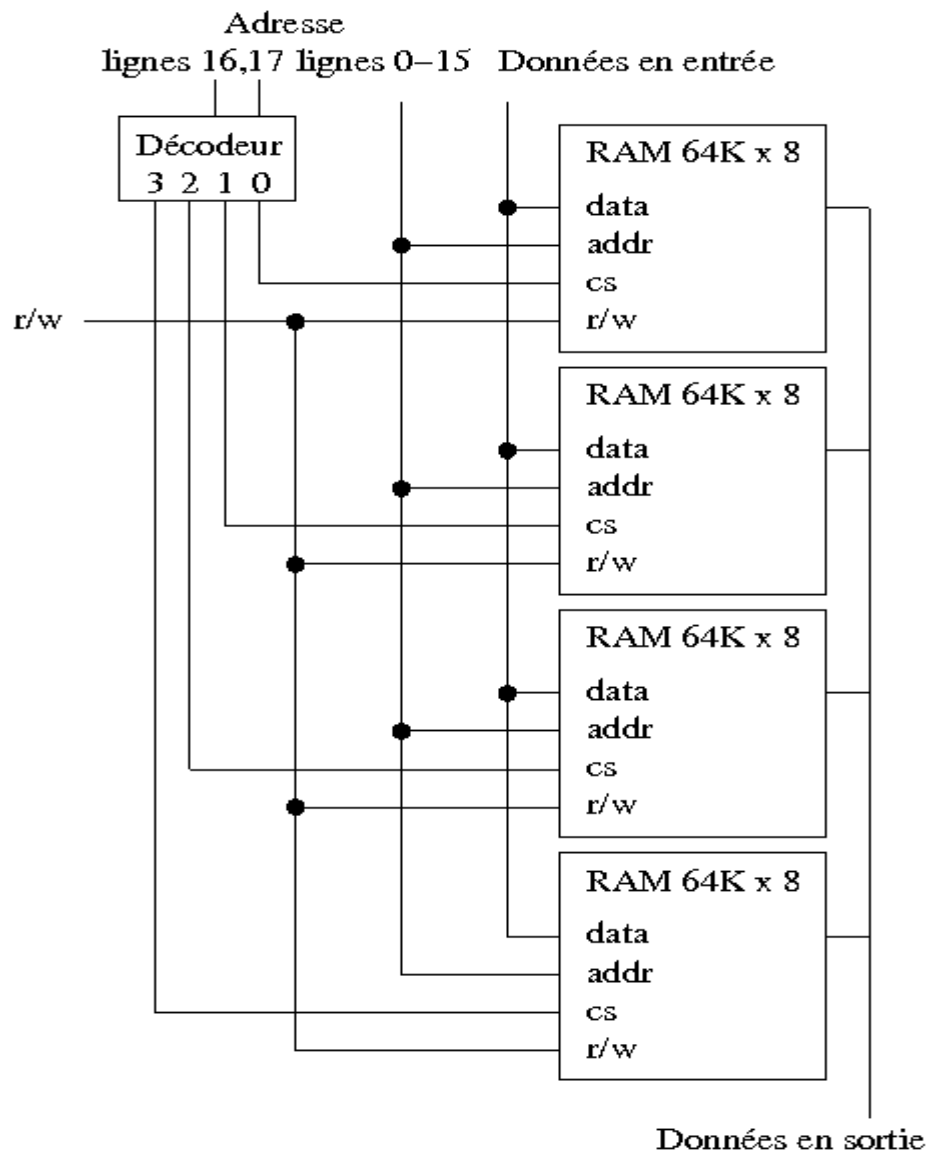
Barrette mémoire organisée en bancs mémoire

#### 5.4. Extension de la mémoire

- Il existe plusieurs manières d'augmenter la capacité d'une mémoire.
- La méthode la plus simple consiste à créer une matrice plus grande.
  - pour doubler la capacité d'un circuit mémoire comportant  $2^m$  cellules, il suffit de rajouter une ligne d'adresse.
  - On obtient alors une mémoire de  $2^{(m+1)}$  mots.
  - L'inconvénient tient au fait que le décodeur doit être augmenté en conséquence et double alors de taille restreignant l'espace attribué à la matrice de bits.
- En fait on préfère utiliser des procédés qui consistent à élaborer une mémoire de plus grande capacité à partir de boîtiers déjà existants.

#### a) Augmentation de l'espace d'adressage

- Comme le montre la figure ci-dessous, afin d'obtenir une mémoire de 256 Ko, on peut utiliser 4 boîtiers de 64 Ko.
- Une adresse mémoire est alors codée sur 18 bits.
- Les 16 bits de poids faible donnent l'adresse à l'intérieur d'un boîtier et les 2 bits de poids fort donnent le numéro du boîtier à sélectionner.

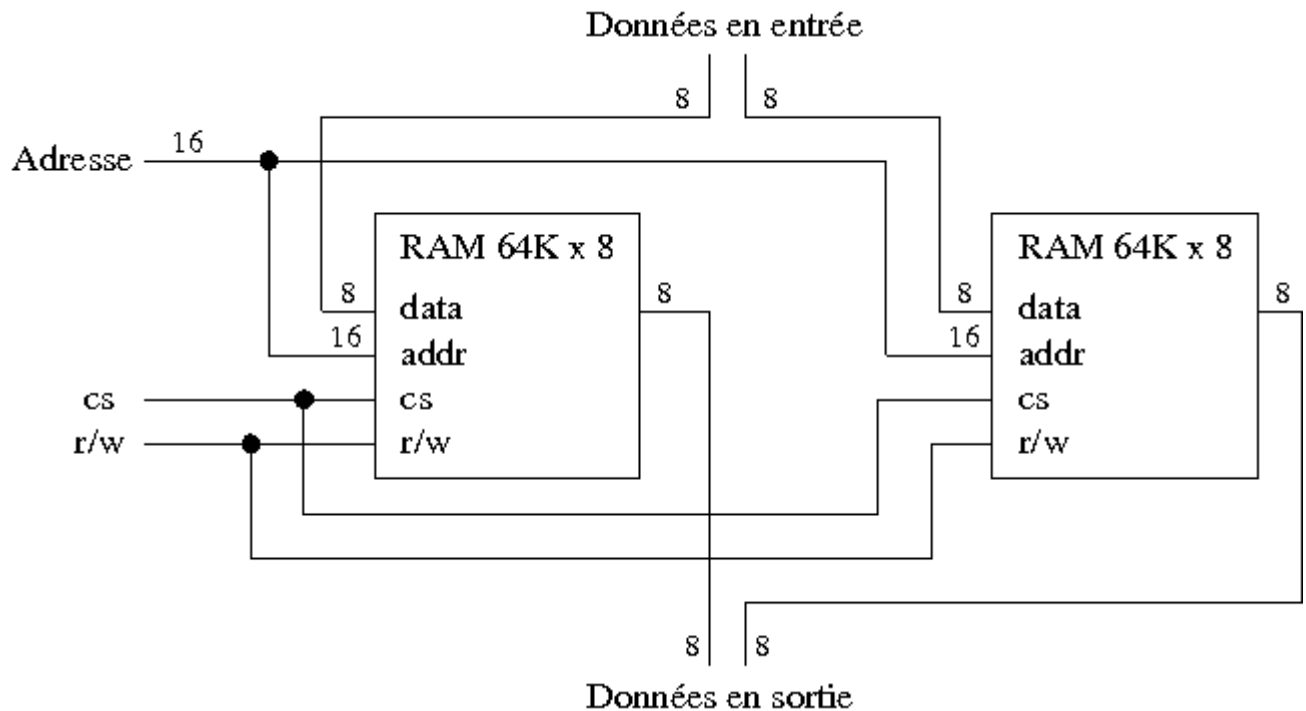


Organisation d'une mémoire de 256 Ko à partir de blocs de 64 Ko



## b) Augmentation de la longueur d'un mot

- Une autre méthode d'extension de la mémoire consiste à augmenter la longueur d'un mot mémoire.
- La figure suivante montre comment passer d'une mémoire de 64 K mots de 8 bits à une mémoire de 64 K de 16 bits.
- En fait on a une mémoire de  $2 \times 64$  Ko soit 128 Ko.



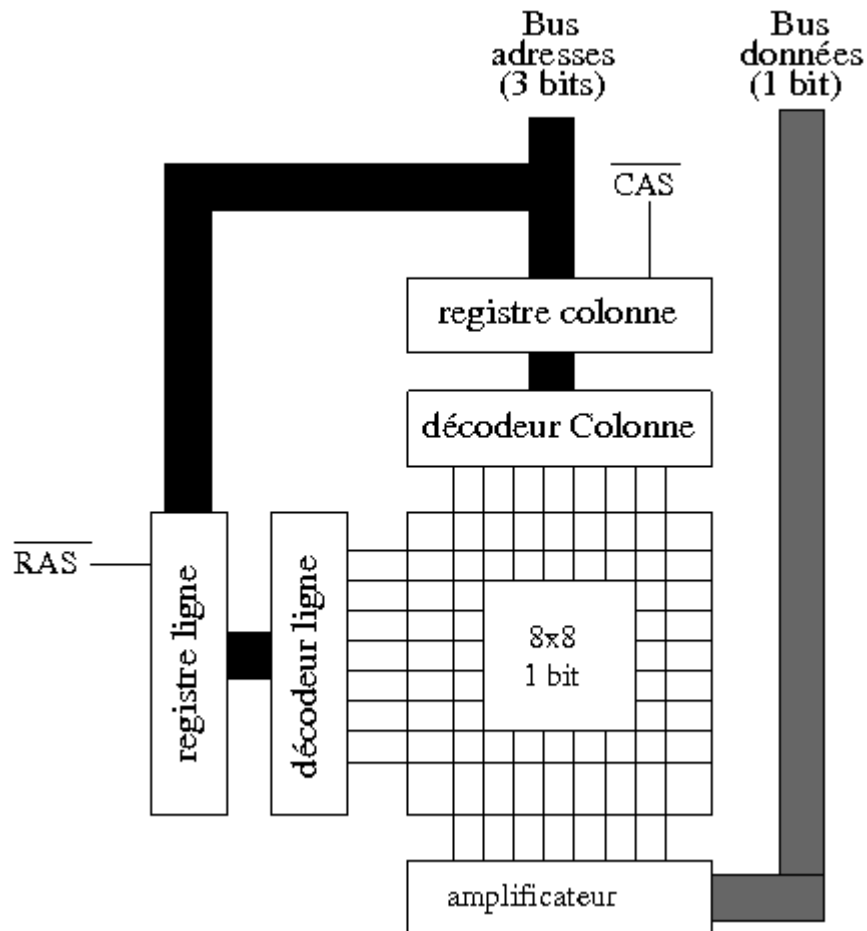
Organisation d'une mémoire de 64 K mots de 16 bits

## 5.5. Alignement

- Le microprocesseur Intel 8086 possède :
  - un bus d'adresse de 20 bits
  - et un bus de données de 16 bits
- Son organisation mémoire ressemble à celle de la section précédente.
- Quand le CPU accède à un mot de 16 bits situé à une adresse paire, on dit qu'il accède à un mot **aligné**, c'est à dire que les deux octets situés à la même adresse peuvent être lus ou écrits en un seul cycle horloge.
- Par contre s'il accède à un mot situé à une adresse impaire, c'est à dire un mot non aligné, il lui faut deux cycles horloge.
- Le 8086 est conçu pour accéder à des mots alignés ou non. Le fait que le mot soit aligné ou non joue alors sur la vitesse d'exécution de certaines opérations.

## 5.6. Accès mémoire

Considérons le composant mémoire de la figure suivante :



### Organisation d'un composant mémoire

- On utilise ici une technique appelée *multiplexage d'adresse* fondée sur le fait que la mémoire n'est pas conçue sous forme d'un tableau mais d'une matrice de cellules mémoire.
- Cette technique possède plusieurs avantages que nous allons détailler :
  - Si on considère qu'une mémoire de **16384 bits** ( $= 2^{14}$ ) est représentée sous forme d'un tableau on doit utiliser **14 lignes** sur le **bus d'adresse**.
  - Par contre si on modélise la mémoire sous la forme d'une matrice de **128 × 128 bits** on peut alors utiliser deux décodeurs **7 vers 128** (un pour les lignes et un pour les colonnes) mais qui sont beaucoup plus simples (en nombre de lignes de courant et de portes logiques) qu'un seul décodeur **14 vers 16384** (On rappelle qu'un décodeur composé de **n** entrées comporte **2<sup>n</sup>** portes **AND** et **n** entrées sur chaque porte).
  - D'autre part, plutôt que d'utiliser un bus d'adresses de **14** lignes on utilise un bus de **7** lignes sur lequel on envoie le numéro de ligne suivi du numéro de colonne auxquelles on désire accéder. L'intérêt de cette technique consiste à accélérer l'accès aux cellules mémoire. En effet une fois le numéro de ligne stocké dans le registre de ligne, le décodeur de ligne peut commencer à travailler. On envoie ensuite le numéro de colonne sur le bus d'adresses.

## 6. Les différentes DRAM

### Évolution des DRAM

Année	Désignation	Type	Largeur Bus
1987	Fast Page Mode (FPM)	Asynchronous	4 ou 8 octets
1990	Extended Data Out (EDO)	Asynchronous	4 ou 8 octets
1997	SDRAM PC 66	Synchronous	8 octets
1998	SDRAM PC 100	Synchronous	8 octets
1999	SDRAM PC 133	Synchronous	8 octets
1999	Rambus PC600/700/800	Synchronous	2 octets
2000	DDR-SDRAM PC 1600	Synchronous	8 octets
2001	DDR-SDRAM PC 2100	Synchronous	8 octets
2002	DDR-SDRAM PC 2700/3200	Synchronous	8 octets
2002	Rambus PC 1066	Synchronous	2 octets
2003	DDR-SDRAM PC 3500/4000/4400	Synchronous	8 octets
2003	DDR2-SDRAM PC 4200	Synchronous	8 octets
2007	DDR3	Synchronous	8 octets

#### 6.1. RAM FPM (asynchrone)

- La mémoire FPM (*Fast Page Mode*) fut utilisée sur les processeurs 80386 et 80486 d'Intel. Elle permet d'améliorer la vitesse de transfert des données en accélérant le transfert de données contiguës.

#### 6.2. RAM EDO (asynchrone)

- La mémoire EDO (*Extended Data Out*) est technologiquement identique à la FPM mais consiste en une amélioration logicielle qui permet l'envoi de données pendant le décodage de la prochaine adresse mémoire à accéder.

#### 6.3. Mémoire SDRAM (synchrone)

- La mémoire SDRAM (*Synchronous DRAM*) est capable d'échanger des données en se synchronisant avec le signal d'horloge pour des fréquences de 66, 100 ou 133 Mhz.
- La mémoire est découpée en deux bancs ce qui permet d'accéder à l'un pendant la configuration du second.
- Cette mémoire est utilisable en mode *rafale* 2, 4,8 ou 512 octets.

#### 6.4. Mémoire DDR-SDRAM

- La *Double Data Rate* SDRAM est un standard mis au point par Nec, Samsung et Toshiba.
- Elle double le taux de transfert des actuelles DRAM en permettant de lire les données sur le front montant et sur le front descendant du signal d'horloge.
- La DDR SDRAM fonctionne à 100, 133, 150, 166, 200 ou 233 Mhz et est capable d'atteindre des vitesses de transfert de l'ordre de 1.6 à 4,2 Go/s :

### Caractéristiques des DRAM

Norme	Norme JEDEC	Fréquence réelle	Bande Passante
DDR-SDRAM PC1600	DDR 200	100 Mhz × 2	1,6 Go/s
DDR-SDRAM PC2100	DDR 266	133 Mhz × 2	2,1 Go/s
DDR-SDRAM PC2700	DDR 333	166 Mhz × 2	2,7 Go/s
DDR-SDRAM PC3000	DDR 370	185 Mhz × 2	2,7 Go/s
DDR-SDRAM PC3200	DDR 400	200 Mhz × 2	3,2 Go/s
DDR-SDRAM PC3500	DDR 433	217 Mhz × 2	3,5 Go/s
DDR-SDRAM PC3700	DDR 466	233 Mhz × 2	3,7 Go/s
DDR-SDRAM PC4200	DDR 533	266 Mhz × 2	4,2 Go/s
DDR-SDRAM PC4400	DDR 550	275 Mhz × 2	4,4 Go/s
DDR2-SDRAM PC3200	DDR2 400	200 Mhz × 2	3,2 Go/s
DDR2-SDRAM PC4200	DDR2 533	267 Mhz × 2 × 2	4,2 Go/s
DDR2-SDRAM PC5300	DDR2 667	333 Mhz × 2 × 2	5,3 Go/s
DDR2-SDRAM PC6400	DDR2 800	400 Mhz × 2 × 2	6,4 Go/s
DDR2-SDRAM PC8000	DDR2 1066	533 Mhz × 2 × 2	8 Go/s
DDR3-SDRAM PC3-6400	DDR3 800	400 Mhz × 2 × 2	6,4 Go/s
DDR3-SDRAM PC3-8500	DDR3 1066	533 Mhz × 2 × 2	8,5 Go/s
DDR3-SDRAM PC3-10667	DDR3 10333	667 Mhz × 2 × 2	10,6 Go/s
DDR3-SDRAM PC3-12800	DDR3 1600	800 Mhz × 2 × 2	12,8 Go/s
DDR3-SDRAM PC3-14900	DDR3 1866	933 Mhz × 2 × 2	14,9 Go/s
RDRAM PC 800	PC 800	100 Mhz × 4 × 4 bancs	3,2 Go/s
RDRAM PC 1066	PC 1066	130 Mhz × 4 × 4 bancs	4,2 Go/s
RDRAM PC 1200	PC 1200	150 Mhz × 4 × 4 bancs	4,8 Go/s

La DDR 550 est apparue en décembre 2003 et elle se révèle donc plus performante que le DDR2 533. L'intérêt de la DDR2 est qu'elle permet une montée en puissance alors que la DDR1 est sensée ne pas pouvoir dépasser un certain seuil de fréquence (bien que sans cesse repoussé).

#### 6.5. Mémoire RDRAM ou DRDRAM

- La **Direct Rambus DRAM** développée par la société **Rambus** grâce au soutien d'Intel représente une amélioration de la DRAM basée sur une architecture à bus très rapide permettant de transférer 16 bits à 800 Mhz.
- Ce qui lui permet d'atteindre un taux de transfert de 1,6 Go/s si on utilise la technologie Dual-channel.
- La dernière évolution de la RDRAM, la PC 1200 permet quant à elle d'atteindre une bande passante théorique de 4,8 Go/s.

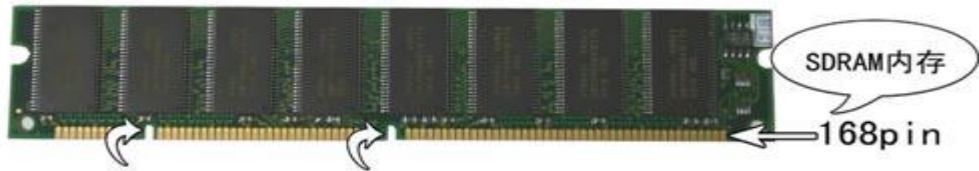
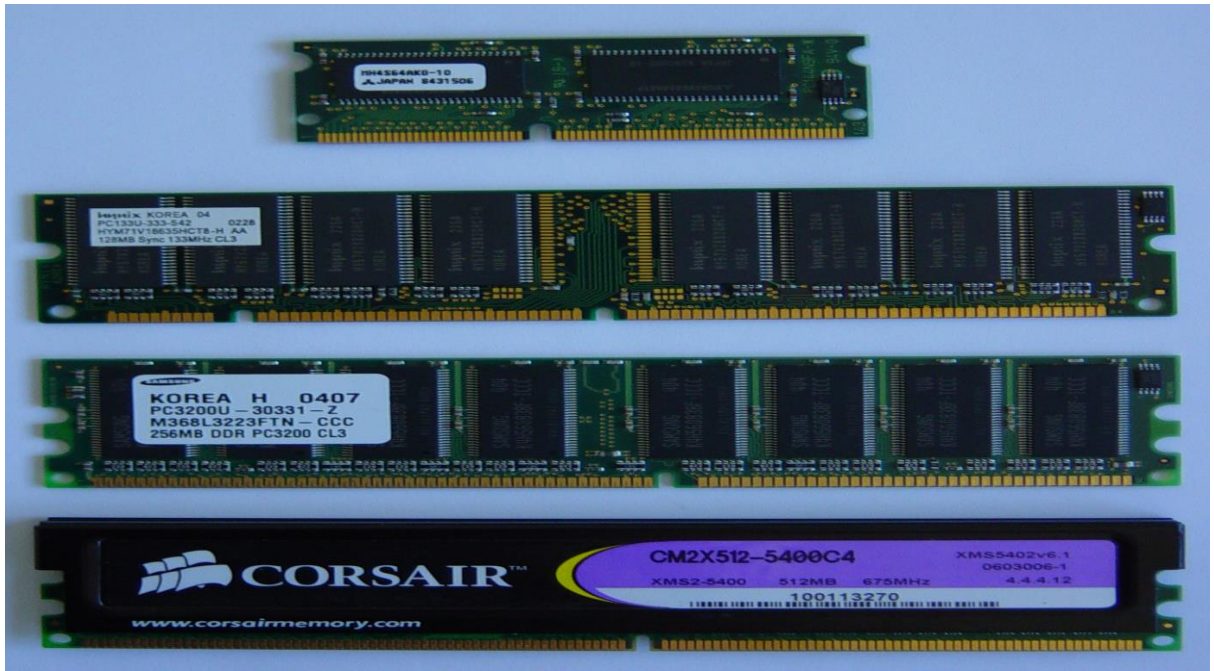
- C'est malheureusement une technologie propriétaire qui oblige à verser des royalties à RAMBUS et Intel.
- D'autre part sa conception est nettement plus coûteuse que celle de la DDR-SDRAM.

### 6.6. Mémoire DDR2-SDRAM

- La mémoire DDR2 est une évolution de la DDR.
- Ces deux types de mémoires possèdent des voltages ainsi que des connecteurs différents ce qui les rend incompatibles entre elles.
- La mémoire DDR vieillissante est peu à peu remplacée par la DDR2 qui offre une bande passante supérieure.

Memory Type	Clock Speed	Name	Bandwidth Single-Channel	Bandwidth Dual-Channel
DDR266	133 MHz DDR	PC2100	2,100 MB/s	4,200 MB/s
DDR333	166 MHz DDR	PC2700	2,700 MB/s	5,400 MB/s
DDR400	200 MHz DDR	PC3200	3,200 MB/s	6,400 MB/s
DDR2 400	200 MHz DDR	PC2 3200*	3,200 MB/s	6,400 MB/s
DDR2 533	266 MHz DDR	PC2 4300*	4,266 MB/s	8,533 MB/s
DDR2 667	333 MHz DDR	PC2 5300*	5,333 MB/s	10,666 MB/s
DDR2 800	400 MHz DDR	PC2 6400*	6,400 MB/s	12,800 MB/s
DDR2 1111	1111 MHz DDR	PC2 8888	8,888 MB/s	17,776 MB/s





## 6.7. Le Dual-Channel

- La technologie **Dual-Channel** double théoriquement la bande passante :
- on utilise **2** barrettes mémoires de même capacité et on accède à ces barrettes en **2 × 64** bits (soit **128** bits). En pratique, le gain du Dual Channel n'est que de **5%**.
- 

**Résultats sans et avec dual-channel pour N=64 sur une architecture 32 bits, les temps sont donnés en secondes**

<b>K</b>	<b>no DC</b>	<b>theoretical</b>	<b>with DC</b>	<b>gain</b>
4096	0.24	0.24	0.23	-
8192	0.48	0.48	0.48	-
16384	0.96	0.96	0.95	-
<b>32768</b>	<b>2.33</b>	<b>1.92</b>	<b>1.92</b>	<b>-17%</b>
65536	9.32	3.84	5.42	-41%
131072	27.63	7.68	17.56	-36%
262144	60.00	15.36	37.50	-38%

## 6.8. Fréquence et débit mémoire

- Le débit théorique d'une mémoire d'un ordinateur dépend de trois facteurs :
  - la **fréquence** du bus **FSB** (*Front Side Bus*) reliant le CPU à la mémoire,
  - la **largeur** du bus en nombre de bits,
  - un **coefficient** multiplicateur qui dépend de la technologie utilisée (normal = 1, DDR = 2, QDR = 4).
- Le débit mémoire est donc donné par la formule suivante :

$$\text{Débit} = \text{Fréquence} \times \text{Largeur} \times \text{Coef}$$

### Exemple 1:

- Un ordinateur doté d'un FSB à **133** Mhz de largeur **64** bits utilisant la technologie DDR possède un débit mémoire théorique de : **133 × 64 × 2 bits/s soit 2,128 Go/s.**

### Exemple 2:

- Un Pentium 4 doté d'un FSB à **100** Mhz *quad pumped* de largeur **64** bits utilisant de la **DRDRAM** possède un débit mémoire théorique de : **100 × 64 × 4 bits/s soit 3,2 Go/s.**

## 7. Détection et correction d'erreur

- Tout composant électronique est soumis à des erreurs provoquées par des variations de courant électrique, des champs magnétiques ou la chaleur.
- Les composants actuels sont assez fiables pour ne pas nécessiter de mécanismes de correction d'erreur.

- Cependant sous certaines conditions il est préférable d'avoir recours à de la mémoire capable de détecter des erreurs (milieu industriel, militaire, spatial).
- Les mémoires dotées de système de détection ou de correction d'erreur ont des performances similaires aux mémoires classiques (à 2 ou 3 % près).

#### a) Bit de parité (Parity Checking)

- Pour chaque groupe de 8 bits (octet) on ajoute un bit supplémentaire dit **bit de parité** qui prend la valeur **1** quand il y a un nombre pair de 1 dans le groupe de 8 bits.
- À chaque rafraichissement de la mémoire (dans le cas des DRAM) on effectue un contrôle de parité.
  - Si une erreur est détectée on envoie un signal **NMI** (*Non Masquable Interrupt*) au CPU qui décide soit d'arrêter la machine, soit de fermer l'application exploitant la zone de donnée erronée.
- Cette technique bien qu'intéressante possède cependant deux désavantages :
  - les mémoires sont plus difficiles à gérer et donc plus coûteuses,
  - on utilise 12,5 % d'espace en plus pour stocker le bit de parité,
  - on ne peut pas corriger l'erreur et la détection d'erreur n'est pas totale car si deux bits changent de valeur l'erreur ne sera pas détectée.

#### b) Correction de code (ECC)

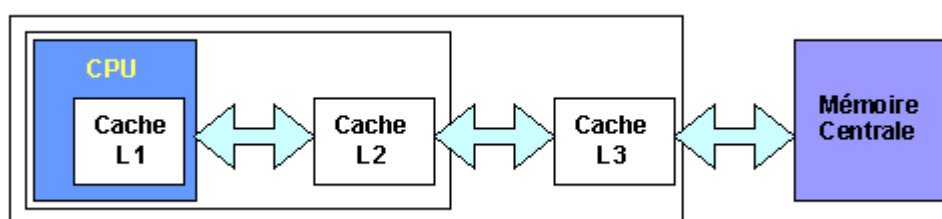
- Le principe de l'ECC (*Error Code Correction*) consiste à utiliser un algorithme pour corriger les erreurs quand elles sont détectées.
- Par exemple pour détecter des erreurs sur 64 bits de données on utilise 7 bits ECC qui codent la parité de sous-ensembles de bits de données ce qui permet de détecter une erreur sur un bit mais également de la corriger.
- En fait on s'arrange pour pouvoir détecter une erreur concernant jusqu'à 2 bits mais on n'est capable de corriger qu'un seul bit. On qualifie parfois la mémoire ECC de SECDED (*Single Error Correction with Double Error Detection*).



## 8. La mémoire cache

### 8.1. Principe

- Il est un fait avéré que les microprocesseurs ont toujours été plus rapides que les mémoires.
- L'accroissement du temps d'accès des mémoires va de pair avec un accroissement de la vitesse des CPUs et l'écart reste donc constant.
- L'accès mémoire représente donc un goulet d'étranglement puisque le CPU passe du temps à attendre les informations provenant de la mémoire
- On considère en effet que les performances des processeurs doublent tous les ans et demi (loi de Moore) alors que celles des mémoires doublent tous les 10 ans.
- La résolution du problème n'est cependant pas technologique mais plutôt d'ordre économique. Il est techniquement possible de réaliser des mémoires presque aussi rapides que des CPU mais leur coût de production est relativement onéreux. On préfère donc utiliser des mémoires centrales plus lentes mais de coût raisonnable.
- Pour pallier la lenteur relative des mémoires centrales on a recours à une solution matérielle et logicielle nommée **mémoire cache** ou antémémoire.



Mémoire cache primaire et secondaire

On distingue en fonction de leur taille et de leur localité :

- la mémoire cache de **premier niveau (L1)** située à l'intérieur du processeur et dont la taille est de deux fois 8 Ko (Pentium) ou deux fois 16 Ko (Pentium Pro)
- la mémoire cache de **second niveau (L2)** située dans le boîtier du processeur et dont la taille varie de 256 à 512 Ko.
- la mémoire cache de **troisième niveau (L3)** située à présent au niveau du processeur et dont la taille varie de 6 à 24 Mo.

#### Taille des caches L3 des Itanium (Intel)

Version	Taille L3
Dual Core Itanium 2 9050	24 Mo
Dual Core Itanium 2 9040	18 Mo
Dual Core Itanium 2 9030	8 Mo
Dual Core Itanium 2 9020	12 Mo
Itanium 2 9010	6 Mo

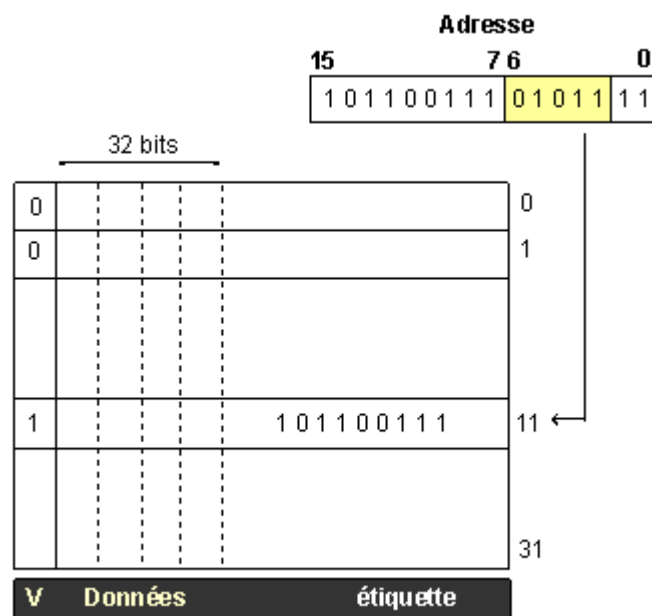
- Le principe de la mémoire cache est simple : il consiste à stocker dans une mémoire rapide et de faible capacité les données les plus souvent utilisées.
- La mémoire cache repose sur le principe de :
  - **localité** : on a une forte probabilité d'accéder à la donnée à l'adresse P+1 si on a accédé à la donnée à l'adresse P
  - **temporalité** : on a une forte probabilité d'accéder à la donnée à l'adresse P au temps T+1 si on a accédé à la donnée à l'adresse P au temps T

On distingue 3 architectures de caches :

- les caches à **accès direct** (*direct mapping*) pour lesquels une adresse sera toujours stockée au même emplacement dans la table. La mémoire centrale étant plus grande que la mémoire cache cela implique que des adresses appartenant à la même classe d'équivalence seront toujours stockées dans la même entrée du cache.
- les caches à **accès direct à n groupes** (*n-way direct mapping* ou *set associative*) qui pallient le défaut du cache direct en permettant de stocker n adresses de la même classe d'équivalence pour la même ligne de cache
- les caches **associatifs** (*fully associative*) pour lesquels une adresse peut être stockée n'importe où dans la table. Dans ce cas le champ étiquette n'est pas suffisant pour identifier l'adresse, il faut stocker l'adresse en totalité.

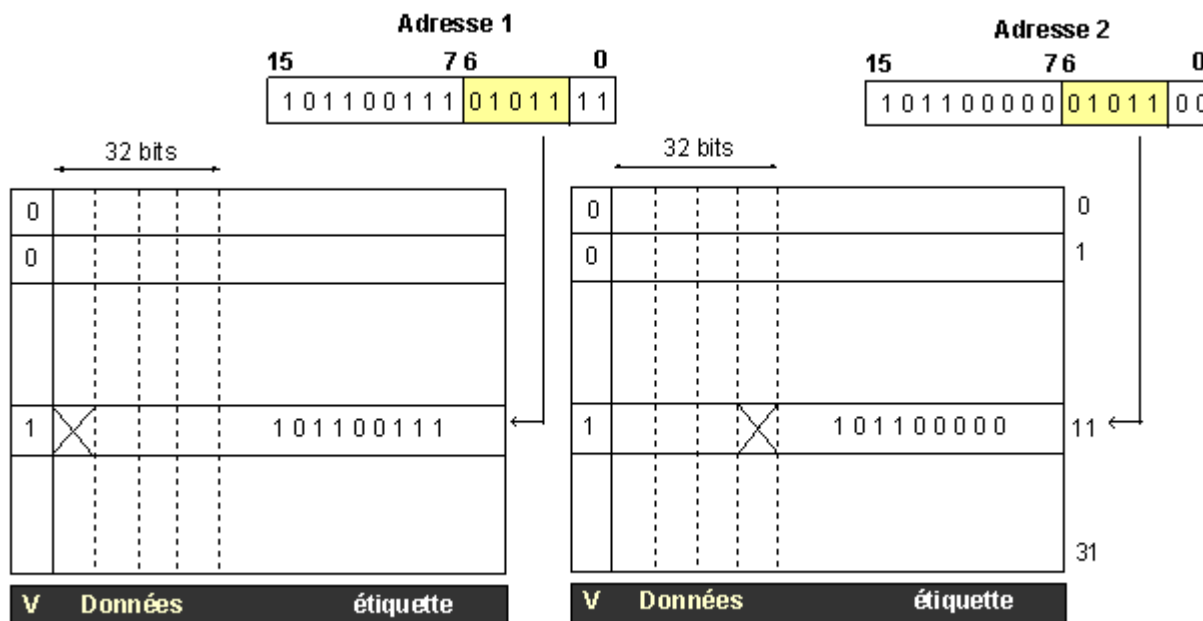
Une mémoire cache est en fait un dispositif de stockage des données et de repérage de leur emplacement en mémoire. On peut représenter ce dispositif sous forme d'une table comportant au minimum trois champs :

- un bit indiquant si la ligne de la table est valide ou non
- la donnée à stocker
- un champ permettant d'identifier l'adresse qui est soit l'adresse entière soit une étiquette qui identifie l'adresse réelle à laquelle la donnée se trouve.



**Cache à accès direct de 32 entrées pour des adresses sur 16 bits et des données sur 32 bits**

Un mécanisme simple de décomposition d'une adresse mémoire en deux parties appelées *index* et *étiquette* (ou numéro de bloc) est utilisé afin d'identifier l'emplacement référencé par une entrée du cache.



Cache à accès direct à deux groupes

## 8.2. Cohérence des données

Un problème important lié à la mémoire cache concerne la cohérence des données. Si une entrée du cache est utilisée pour stocker une nouvelle donnée, la donnée précédemment stockée doit être mise à jour en mémoire centrale. On peut envisager deux stratégies de mise à jour :

- l'écriture immédiate (**write through**) consiste à mettre immédiatement à jour la donnée en mémoire centrale dès lors que sa valeur dans le cache est modifiée.
- l'écriture différée (**write back**) consiste à mettre à jour la mémoire centrale lors de la modification de l'entrée du cache correspondante.

Un compromis entre coût de production, complexité et performances doit être trouvé lors de la conception d'un cache. Par exemple la stratégie d'écriture immédiate augmente le trafic vers la mémoire centrale. Si la stratégie d'écriture différée pallie ce problème elle en crée un nouveau : en effet, certains circuits d'entrées sorties de type **DMA** (*Direct Memory Access*) sont capables de lire ou écrire des données en mémoire sans passer par le processeur et risquent par exemple de lire des données qui ne sont pas à jour. On peut contourner ce problème mais cela augmente la complexité du système.

## 8.3. Cache inclusif ou exclusif

Les caches L1 et L2 fonctionnent en mode :

- **inclusif** : une donnée peut être stockée à la fois dans L1 et L2. L'espace cache total est alors :

$$\text{Taille (L1+L2)} - \text{Taille(L1)} = \text{Taille(L2)}$$

Cette méthode n'a d'intérêt que si le cache L2 est plus grand que le cache L1. On considère que  $\text{Taille(L2)} \geq 4 \times \text{Taille(L1)}$  pour que L2 soit efficace.

- **exclusif** : une donnée est stockée soit dans L1, soit dans L2. L'espace cache total est alors :

$$\text{Taille(L2)} + \text{Taille(L1)}$$

L'espace cache est ici plus grand que dans la méthode précédente mais il faut faire en sorte que les données ne soient pas stockées dans les 2 caches à la fois.

*AMD utilise un cache L2 exclusif alors qu'Intel utilise un cache inclusif depuis le Pentium Pro.*

## 8.4. Performances

La performance d'un cache est donnée par le ratio suivant :

$$\text{Performance} = \text{Nombre d'accès fructueux} / \text{Nombre total d'accès}$$

Type de cache	Performance
Accès direct	60 à 80 %
à n groupes	80 à 90%
Associatif	90 à 95%

L'intérêt des caches à accès direct repose sur la vérification rapide de la présence ou non dans le cache d'une adresse. Pour un cache associatif la recherche est beaucoup plus complexe : le but du cache étant l'accès rapide aux données on doit éviter une recherche linéaire sur chaque entrée du cache afin de vérifier la présence d'une adresse. On utilise donc des circuits spécialisés qui comparent simultanément toutes les entrées du cache. Le cache associatif est donc complexe à mettre en œuvre et en conséquence beaucoup plus coûteux que le cache à accès direct. Cependant le cache associatif présente un taux de succès plus élevé que le cache à accès direct pour lequel un nombre limité d'adresses appartenant à la même classe d'équivalence se trouve dans le cache.

	Pentium 4 (Prescott)	Pentium D (Presler)	Core Duo (Yonah)	Core 2 Duo (Conroe)	Ahtlon 64 X2 (Windsor)	AMD FX- 8530 (Vishera)
<b>L1 Instructions</b>	12k micro- op 8-way	12k micro- op 8-way	32 ko 8-way	32 ko 8-way	64 ko 2-way	64 ko 2-way
<b>L1 Données</b>	16 ko 8-way	16 ko 8-way	32 ko 8-way	32 ko 8-way	64 ko 2-way	16 ko 4-way
<b>Latence L1 (cycles)</b>	3	4	3	3	3	4

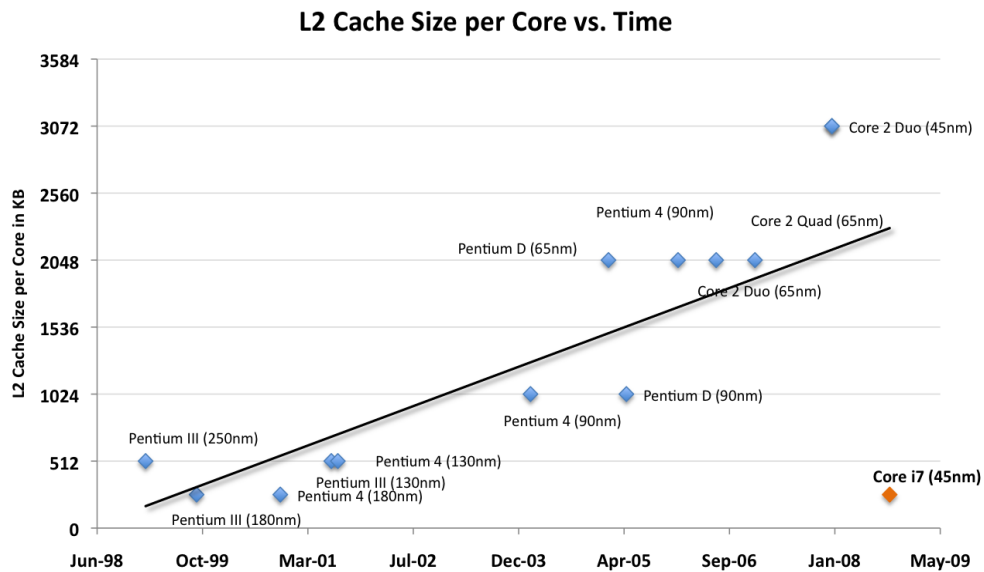
<b>L2</b>	1 Mo 8-way	2 x 2Mo 8-way	2 Mo 8-way partagés	4 Mo 16-way partagés	2 x 512 ko 16-way	2048 ko 16-way
<b>Latence L2 (cycles)</b>	31	31	14	14	20	20

**Cache/Memory Latency Comparison**  
Source Anandtech.com

Processeur	L1	L2	L3	Main Memory
AMD FX-3850 (4 Ghz) 2012	4	20	76	?
AMD FX-8150 (3.6GHz)	4	21	65	195
AMD Phenom II X4 975 BE (3.6GHz)	3	15	59	182
AMD Phenom II X6 1100T (3.3GHz)	3	14	55	157
Intel Core i5 2500K (3.3GHz)	4	11	25	148
Intel Core i7 3960X (3.3GHz)	4	11	30	167

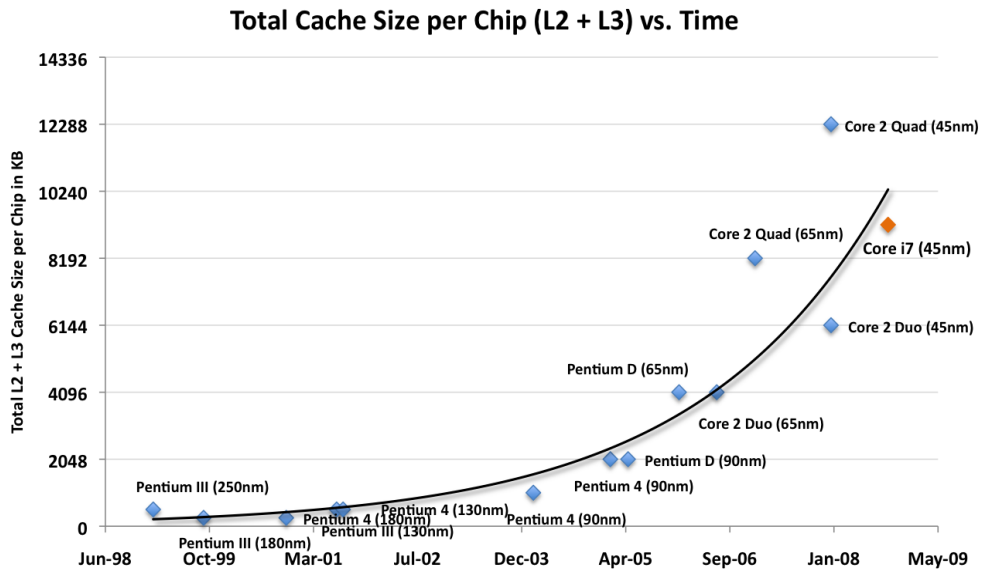
### 8.5. Évolution de la taille du cache

Évolution de la taille du cache L2 au cours du temps pour les processeurs Intel



(Source [AnandTech.com](http://AnandTech.com))

Évolution de la taille du cache L2 au cours du temps pour les processeurs Intel



(Source [AnandTech.com](http://AnandTech.com))

L'arrivée des processeurs multi-cœurs a modifié les besoins en cache L2/L3 :

- avec un cœur on tente de posséder des caches L1, L2 de plus en plus grands et performants
- avec plusieurs cœurs, il faut alimenter de manière équivalente tous les cœurs, d'où un cache L3 de plus en plus grand, alors que le cache L2 peut diminuer (cas du Gulftown : 256 ko L2, 12 Mo L3)

## 8.6. Cache et programmation

### a) Un exemple concret : le produit de matrices

L'utilisation d'un cache n'est pas la panacée. En effet si le fait de disposer d'une mémoire cache laisse à penser au programmeur qu'il peut se décharger de la gestion de la mémoire et en attribuer la gestion au compilateur, certaines désillusions sont à prévoir. Soit par exemple, le morceau de code suivant qui réalise le produit de deux matrices carrées de *double* de dimension N :

```

int A[N][N], B[N][N], C[N][N];

for (int i=0; i < N; ++i) {
  for (int j=0; j < N; ++j) {
    for (int k=0; k < N; ++k) {
      C[i][j] += A[i][k] * B[k][j];
    }
  }
}

```



L'exécution de ce code sur un Pentium III (Tualatin 1,4 Ghz) pour des valeurs croissantes de N donne :

- N = 800 : 34s
- N = 840 : 22s

### Exemple produit de matrices

Si le calcul pour N=800 est de 34s, il devrait être au moins égal à 34s pour N=840. Pour expliquer ce phénomène il suffit d'étudier les caractéristiques des caches L1 de données et L2 de ce processeur

Cache L1	
taille du cache L1	16 ko
associativité L1-D	4-way
taille d'une ligne L1	32 octets
Cache L2	
taille du cache L2	256 ko
associativité L1	8-way
taille d'une ligne L2	32 octets

#### b) Cache L1-D

$$\begin{aligned}
 16 \text{ ko} &= 2^4 \times 2^{10} = 2^{14} \text{ octets} \\
 2^{14} / 4 \text{ (way)} &= 2^{14} / 2^2 = 2^{12} \\
 2^{12} / 32 \text{ (octets par ligne)} &= 2^{12} / 2^5 = 2^7 = 127
 \end{aligned}$$

il existe donc 127 lignes par cache associatif, une adresse sera donc décomposée sous la forme suivante :

...	11	10	9	8	7	6	5	4	3	2	1	0
	7 bits							5 bits				
	ligne							déplacement				

Une matrice 800 × 800 de double occupe 5.120.000 octets soit 0x4E2000 en hexadécimal. Les trois matrices A, B et C sont stockées aux adresses suivantes :

- A : 0x093D4C00
- B : 0x08EF2C00
- C : 0x08A10C00

On s'aperçoit que les dimensions de matrices qui posent problème correspondent à des défauts de cache concernant les caches L1 et L2.

#### c) Cache L2

$$\begin{aligned}
 256 \text{ ko} &= 2^8 \times 2^{10} = 2^{18} \text{ octets} \\
 2^{18} / 8 \text{ (way)} &= 2^{18} / 2^3 = 2^{15} \\
 2^{15} / 32 \text{ (octets par ligne)} &= 2^{15} / 2^5 = 2^{10} = 1024
 \end{aligned}$$

il existe donc 1024 lignes par cache associatif, une adresse sera donc décomposée sous la forme suivante :

...	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	10 bits										5 bits				
	ligne										déplacement				

Ici, ce sont bien sûr les défauts de cache (cache miss) du cache L2 qui sont pénalisants.  
Autres dimensions de matrice posant problème :

- (double) 416 :  $416 \times 416 \times 8 = 1.384.448 = 0x152000$
- (double) 864 :  $864 \times 864 \times 8 = 5.971.968 = 0x5B2000$
- (intégrer) 1280 :  $1280 \times 1280 \times 4 = 6.553.600 = 0x640000$

#### d) localité des données

Afin d'améliorer les performances du cache, on est parfois amené à décomposer les données :

```

struct { /* 1600 bytes */
    int a, c, e;
    char b, d;
} array_of_struct [100];

struct { /* 1400 bytes */
    int a[100], c[100], e[100];
    char b[100], d[100];
} struct_of_array;

struct { /* 1200 bytes */
    int a, c, e;
} hybrid_struct_of_array_ace[100];

struct { /* 200 bytes */
    char b, d;
} hybrid_struct_of_array_bd[100];

```

Il est également préférable d'utiliser des structures de données alignées sur des bloc de 4, 8, 16 ou 32 octets :



```
// structure sur 7 octets
```

```
typedef struct { short x,y,z; char a; }
```

```
// ajouter un champ pour aligner sur 8 octets
```

```
typedef struct { short x,y,z; char a; char dummy; }
```

## 9. La mémoire de masse

### 9.1. Le disque dur (hard drive)

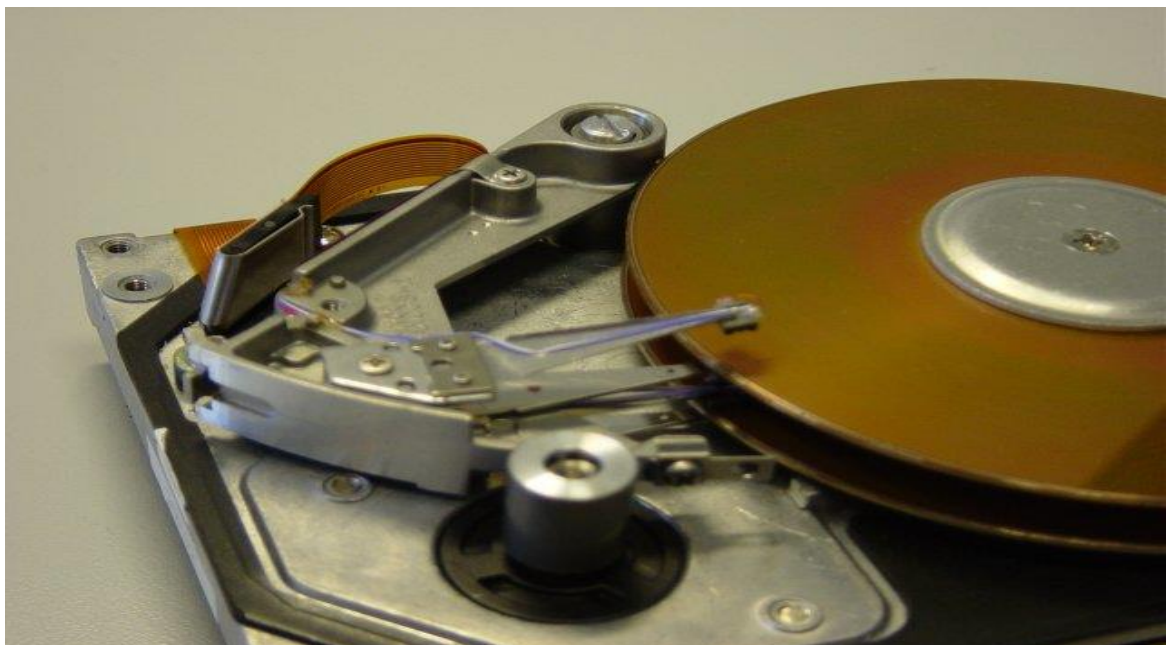
- Le disque dur est l'un des composants (ou périphérique) de l'ordinateur servant à conserver les données de manière **persistante**, contrairement à la mémoire vive, qui est volatile. On qualifie également les média de stockage de grande capacité de **mémoire de masse**.



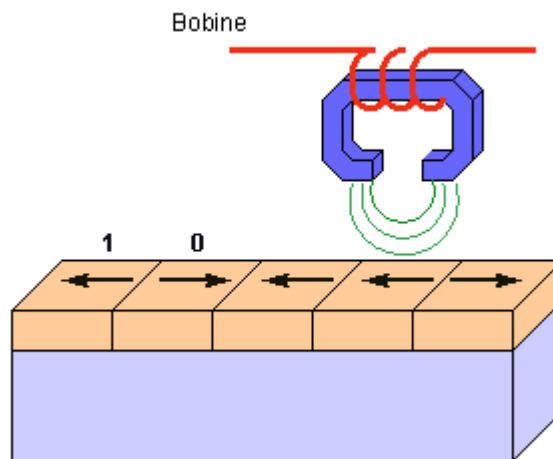
### 9.2. Organisation interne

- Un disque dur est constitué d'un ou plusieurs **plateaux** d'une taille de 3"1/2 ou 2"1/2 entre lesquels passent des têtes de lecture/écriture solidaires d'un bras qui autorise le déplacement des têtes sur toute la surface du disque.
- Les plateaux tournent à une vitesse de 4200, 5400, 7200, 10000, 12000 ou 15000 tr/min.

***Plus la vitesse de rotation est importante, plus le disque est performant.***



### 9.3. Principe de lecture / écriture



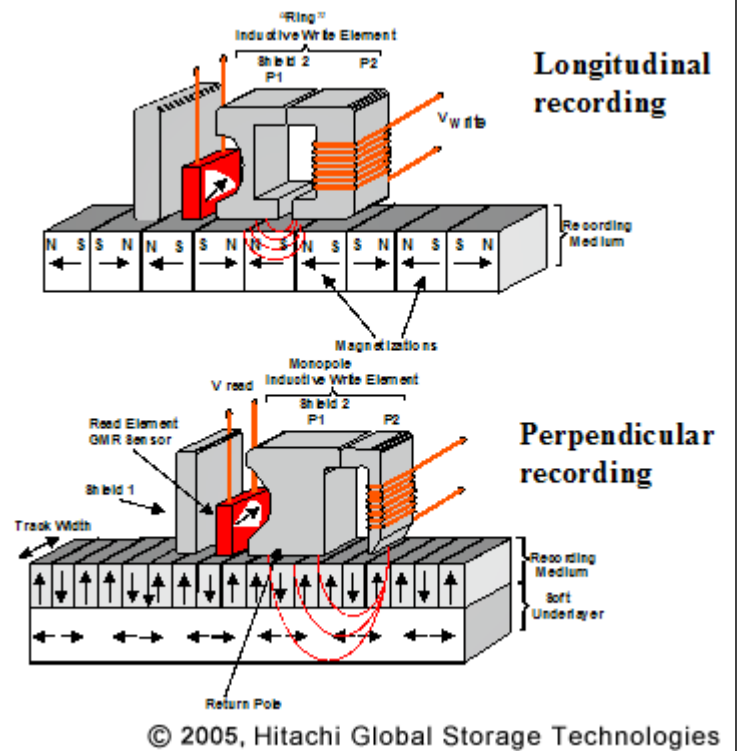
On utilise un matériau dur sur lequel on dispose une couche d'oxyde de fer. La tête de lecture/écriture est composée par un aimant autour duquel est enroulée une bobine.

- **Pour écrire :**  
on fait passer un courant électrique dans la bobine ce qui crée un champ magnétique. Les lignes de champ magnétique traversent la couche d'oxyde et orientent celui-ci en créant de petits aimants dont le sens est donné par le sens du courant dans la bobine.
- **Pour lire :**  
on fait passer la tête de lecture/écriture sur le support magnétisé qui crée un courant induit dans la bobine dont le sens indique s'il s'agit d'un 0 ou d'un 1.

Il faut faire attention à ce que la tête de lecture ne touche pas la couche d'oxyde sinon elle risquerait de l'endommager. En fait la vitesse de rotation du disque donne naissance à un phénomène de sustentation. Lors de l'arrêt de l'ordinateur on déplace les têtes sur une piste appelée *landing track* qui peut sans problème être endommagée puisqu'elle ne contient pas de données.

La miniaturisation des zones de stockages engendre des problèmes comme l'effet *super paramagnétique* pour lequel chaque élément possède une influence sur ses voisins et peut même dans certains cas conduire à modifier la valeur des voisins. Ce problème est dû à l'enregistrement **horizontal** des données.

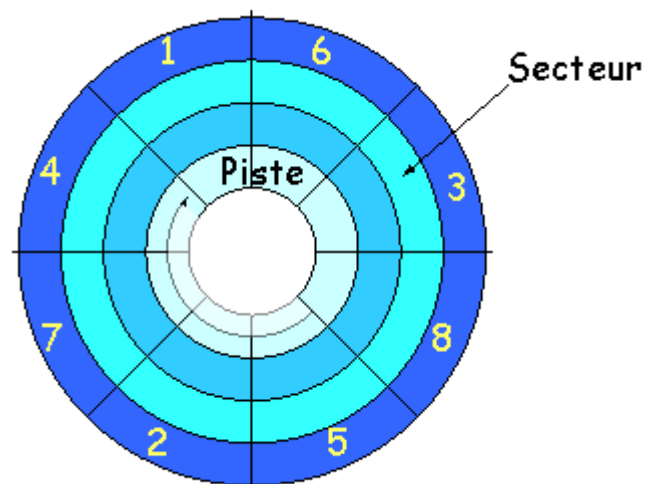
Pour remédier à ce problème on a introduit une technique qui permet un enregistrement **vertical** (encore appelé perpendiculaire) ce qui permet d'accroître la capacité des disques durs. (Hitachi prévoit qu'en 2007 on disposera de disques durs 3,5 pouces d'1 To)



## 9.4. Formatage

Il existe deux types de formatages :

- le formatage de **bas niveau** dit formatage physique
- le formatage de **haut niveau** dit formatage logique



### 1. Formatage de bas niveau

Le formatage de bas niveau a pour but d'organiser la surface du disque en éléments simples : pistes (cylindres), secteurs qui permettront de localiser l'information. Le formatage de bas niveau est effectué en usine lors de la fabrication du disque dur.

## 2. Formatage de haut niveau

Le formatage de haut niveau permet d'organiser les pistes (cylindres) et secteurs en un **système de fichier** qui sera géré par un système d'exploitation (Windows, Linux, Unix, OS2, ...).

Pour formater un disque il faut utiliser la commande `format` sous MSDOS/Windows :

`format c:`

### Les différents systèmes de fichiers

Système d'exploitation	Système de fichiers
DOS	FAT 16
Windows 95	FAT 32
Windows NT,XP	NTFS
Linux	Ext2, Ext3
OS/2	HPFS

#### a) Cas de la FAT 16

Durant le formatage on crée la FAT (File Allocation Table) qui contient la liste des clusters du disque ou de la partition. On répartit plusieurs secteurs en un **cluster** qui devient la plus petite unité d'allocation.

Avec une FAT 16, utilisée avec MSDOS, les adresses des clusters sont codés sur 16 bits, soit 65536 clusters différents (en fait 65526 car 10 clusters sont réservés).

- Pour un disque de 512 Mo, la taille d'un cluster est de  $2^{29} / 2^{16} = 8$  ko.
- Pour un disque de 4 Go, la taille d'un cluster serait de  $2^{34} / 2^{16} = 256$  ko

***La FAT 16 permet de gérer des disques d'une capacité n'excédant pas 2 Go.***

#### b) Cas de la FAT 32

La FAT 32 a vu le jour en 1997, elle code les adresses de clusters sur 28 bits (les 4 derniers bits étant réservés). Elle permet de gérer des disques de 260 Mo à 2 To.

#### Taille cluster en FAT 32

Taille de partition	Taille cluster
disquette	512 octets
260 Mo à 8 Go	4 ko
8 Go à 16 Go	8 ko
16 Go à 32 Go	16 ko
32 Go à 2 To	32 ko

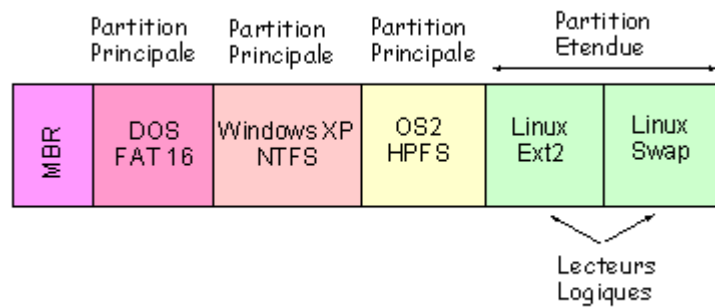
### 9.5. Partitionnement

- Le partitionnement est effectué avant le formatage de haut niveau et permet de placer plusieurs systèmes de fichiers différents (donc plusieurs systèmes d'exploitation) sur le même disque. Le partitionnement peut également servir à séparer programmes et données.

Il y a 3 sortes de partitions :

- les partitions principales
- la partition étendue
- les lecteurs logiques
- Un disque peut contenir au maximum **4 partitions principales** dont une seule est dite active (c'est elle qui permettra de démarrer la machine avec le système d'exploitation par défaut).

On peut également organiser le disque en 3 partitions principales et une partition étendue. Dans ce cas la partition étendue peut être subdivisée en lecteurs logiques



- Le secteur de démarrage (appelé **Master Boot Record** ou MBR en anglais) est le premier secteur d'un disque dur (cylindre 0, tête 0 et secteur 1), il contient la table de partition principale et le code qui, une fois chargé en mémoire, va permettre d'amorcer le système (booter).

Pour partitionner un disque on peut utiliser :

- la commande `fdisk` avant d'installer un système d'exploitation
- ou des utilitaires comme Partition Magik.

Le **MBR** contient toutes les informations relatives au disque dur (fabricant, numéro de série, nombre d'octets par secteur, nombre de secteurs par cluster, nombre de secteurs,...). Ce secteur est donc le secteur le plus important du disque dur, il sert au setup du BIOS à reconnaître le disque dur. Lorsqu'il est endommagé ou effacé par un virus votre disque dur est inutilisable.

## 9.6. Grandeurs caractéristiques et performances

### a) Capacité

Connaissant :

- le nombre de cylindres (C) (= nombre de pistes)
- le nombre de têtes de lecture/écriture (H)
- le nombre de secteurs par piste (S)

On peut calculer la capacité d'un disque dur :

$$\text{Capacité} = C \times H \times S \times 512 \text{ octets}$$

Les capacités des disques durs actuels varient de 40 Go à 1 To.

Remarque : les constructeurs donnent généralement la taille des disques durs en divisant la capacité par 1000 au lieu de 1024. Par **exemple** pour un disque de caractéristiques C=4092, H=16, S=63, la taille devrait être de 2014 Mo, or on indique 2111 Mo.



### b) Taux de transfert

Le taux de transfert indique le nombre de ko que l'on peut transférer en une seconde.

#### Quelques caractéristiques de disques dur Mesures réalisées par hdtune

Disque	Capacité	Vitesse de rotation (tr/min)	Tx de Transfert maximum (Mo/s)	Temps d'accès (ms)
Hitachi HTS548060M	60 Go	5400	31	18,4
Hitachi Travelstar	40	4200	10	18,8
Maxtor Diamond Max Plus	160	7200	56	15,5

### c) Temps de latence

Le temps de latence correspond à l'intervalle de temps entre le moment où la tête de lecture/écriture trouve les données étant placée sur la piste adéquate. Le temps de latence est donné par la formule suivante :

$$T_{\text{latence}} = (60 / \text{vitesse de rotation}) / 2$$

Pour un disque tournant à 10000 tr/min le temps de latence est de 3 ms.

### d) Temps d'accès

Temps nécessaire au passage sur la piste suivante

### e) Temps moyen d'accès

Intervalle de temps entre la réception d'une requête et la mise à disposition des données.

### f) Fabricants de disques dur

- Maxtor
- Hitachi
- Seagate
- Western Digital

### 9.7. Les Solid State Disks (SSD)

Fin 2008, les SSD commencent à devenir *abordables* mais restent encore chers.

Le principe de fonctionnement des SSD repose sur l'utilisation de mémoire Flash qui a tendance à s'user à force de lecture/écriture :

- **la mémoire Flash NAND** : l'accès est réalisé en série et ce type de mémoire se montre intéressant pour la lecture/écriture de fichiers de grande taille
  - SLC (Single-Level Cell) : stockage d'un bit par cellule, faible consommation, taux de transfert élevé : 100.000 cycles avant usure
  - MLC (Multi-Level Cell) : stockage de deux bits par cellule : 10.000 cycles avant usure
- **la mémoire Flash NOR** : adaptée à des accès aléatoires

On utilise généralement de la Flash NAND car sa densité de stockage est plus grande.

#### Comparatif disque dur magnétique (DDM)/ Solid State Disk (SSD)

Critère	DDM	SSD
Consommation	importante / moyenne	faible
nuissance sonore	importante / moyenne	nulle
lecture	moyenne	rapide
écriture	moyenne	lente
cycles	très importants	100.000 / 10.000
Prix	moyen	très élevé

Marque / Modèle	Capacité	Format	Ecriture	Lecture	Prix	Prix / Go
Patriot	200 Go	2,5"	275 Mo/s	285 Mo/s	774 &euro	3,87 &euro / Go
OCZ Colossus	250 Go	3,5"	230 Mo/s	260 Mo/s	749 &euro	3,00 &euro / Go
Samsung F2EG Eco Green	2 To	3.5"	5400 tr/min		112 &euro	0,056 &euro / Go



## 10. CDRom, DVDRom et Blu(e) Ray

CDRom, DVDRom et disque BluRay (ou BlueRay) sont trois déclinaisons de la même technologie qui repose sur la lecture et l'écriture grâce à un laser.

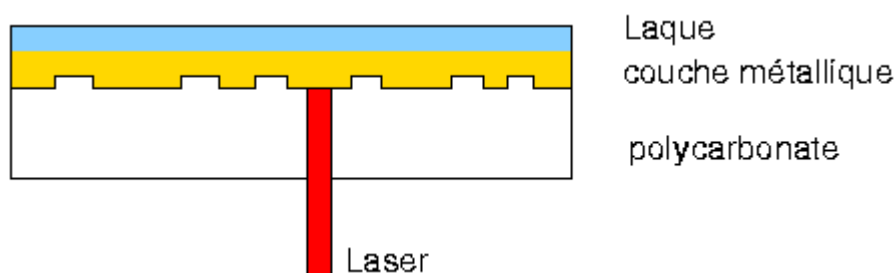
### 10.1. Caractéristiques

Le Compact Disc a été inventé par Sony et Philips en 1981 dans le but de fournir un support audio et vidéo de haute qualité.

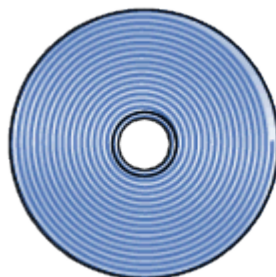
- En 1984, les spécifications techniques du CD ont été étendues afin de permettre le stockage de données
- En 1990 Kodak met au point le CD-R (pour Réinscriptible) ou inscriptible une fois.
- En 1993 est mise au point la norme Vidéo-CD qui permet de stocker 74 min de vidéo sur un CD.
- un CD est capable de stocker 650 ou 700 Mo de données et 74 ou 80 min de musique.
- Le taux de transfert d'un CD-ROM est de 150 ko/s, mais on peut atteindre jusqu'à 7200 ko/s (48X)

Un CD-ROM est un disque de 12 cm de diamètre composé de plusieurs couches :

- une couche de polycarbonate laissant passer la lumière
- une couche métallique réfléchissante (or ou argent) composée de plats et de creux
- une laque anti-UV



L'organisation du CD n'est pas la même que celle des disques durs. On a ici une seule piste principale gravée en spirale (d'une longueur de 5,77 km).



### 10.2. Principe de lecture écriture

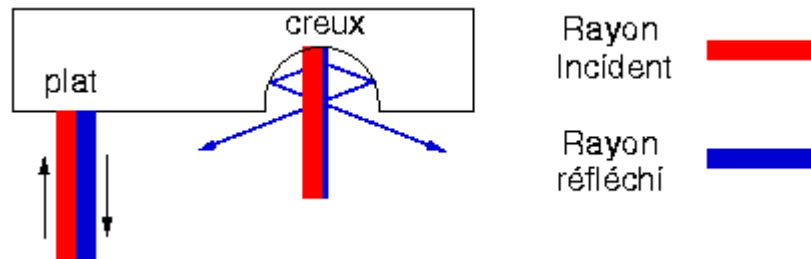
Plusieurs techniques existent en fonction du type de CD : CD-ROM, CD-R, CD-RW.

Le principe de lecture/écriture utilise un rayon laser (Light Amplification by Stimulated Emission of Radiation) **infrarouge** d'une longueur d'onde de 780 nm.

### a) Lecture

Elle consiste à mesurer le déphasage entre l'onde incidente et l'onde réfléchi. Lorsque le laser rencontre un creux il se crée un déphasage que l'on peut mesurer.

De manière un peu plus schématique (et un peu éloignée de la réalité) on peut concevoir que lorsque la lumière rencontre un plat, l'onde incidente est totalement réfléchi. Par contre, lorsque la lumière rencontre un creux, seule une partie est réfléchi. On peut mesurer l'intensité du rayon réfléchi afin de déterminer si on est dans un creux ou un plat.



### b) Écriture

Pour l'écriture les techniques sont assez similaires qu'il s'agisse d'un CD-R ou d'un CD-RW. Dans le cas d'un CR-R on ajoute une couche de colorant organique. Pour les CD-RW on utilise un alliage métallique. Tous deux ont la propriété de changer d'état devenant plus ou moins opaque (moins réfléchissant) si on les expose à de fortes températures (500 degrés).

Dans le cas des CD-RW le CD peut être effacé en utilisant un laser à 200 degrés.

## 10.3. Graveurs CD

Lors de l'achat d'un graveur de CD, trois coefficients sont fournis :

- la vitesse maximum de gravage des CD-R
- la vitesse maximum de gravage des CD-RW
- la vitesse maximum de lecture des CD

On trouve actuellement des graveurs 52x/24x/52x

## 10.4. Le DVD

Le DVD est apparu en 1997 et est principalement dédié à la vidéo. Son format est le même que celui du CD mais il utilise un laser rouge (650 et 635 nm) ce qui autorise une plus grande finesse lors de la gravure (et donc de la lecture).

Son taux de transfert est de l'ordre de 10 Mbits/s.

Il existe 3 types de DVD réinscriptibles et incompatibles :

- DVD-R (2,6 Go)
- DVD-RAM de Toshiba et Matsushita (3,95 à 4,7 Go)
- DRD-RW de Sony, Philips et HP (4,7 Go)
  
- les vitesses de gravure des DVD sont de l'ordre de 4x pour les DVD-R et 2,4x pour les CD-RW. (8x voire 16x prochainement)

### Graveurs Asus DRW-20B1S et DRW-20B1ST

Vitesses d'écriture	
DVD+R DL / DVD-R DL	12x / 12x
DVD+R / DVD-R	20x / 20x
DVD+RW / DVD-RW	8x / 6x
DVD-RAM	12x
CD-R	48x
CD-RW	32x
Vitesses de lecture	
DVD-ROM	16x
DVD-RAM	12x
CD-ROM	48x

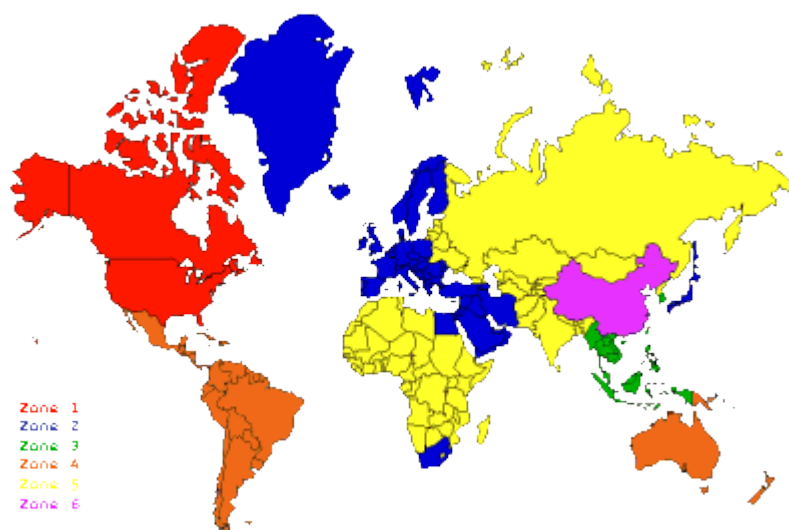
- La vitesse de lecture est de 8x, 12x ou 16x (22 Mo/s).

DVD	Faces	Couches	L/E	Capacité	Audio	Vidéo
DVD-5	1	1	lecture	4,7 Go	9h30m	2h
DVD-9	1	2	lecture	8,5 Go	17h30	4h
DVD-10	2	1	lecture	9,4 Go	19h	4h30
DVD-18	2	2	lecture	17 Go	35h	8h
DVD-R	1	1	une seule écriture	3,95 à 4,7 Go	9h30	2h
DVD-RAM	1	1	lecture/écriture	2,6 Go	?	1h
DVD-RW	1	1	lecture/écriture	4,7 Go	?	2h

Afin de mieux contrôler la distribution des DVD, le monde a été découpé en 6 zones. L'Amérique fait partie de la zone 1 et la France de la zone 2.

Les formats des différentes zones sont incompatibles. Cependant les lecteurs de DVD pour PC autorisent la lecture de DVD par changement de zone. un maximum de 5 changements est autorisé.

À noter qu'il existe des DVD de salon multizones.



## 10.5. Blu-Ray et HD-DVD

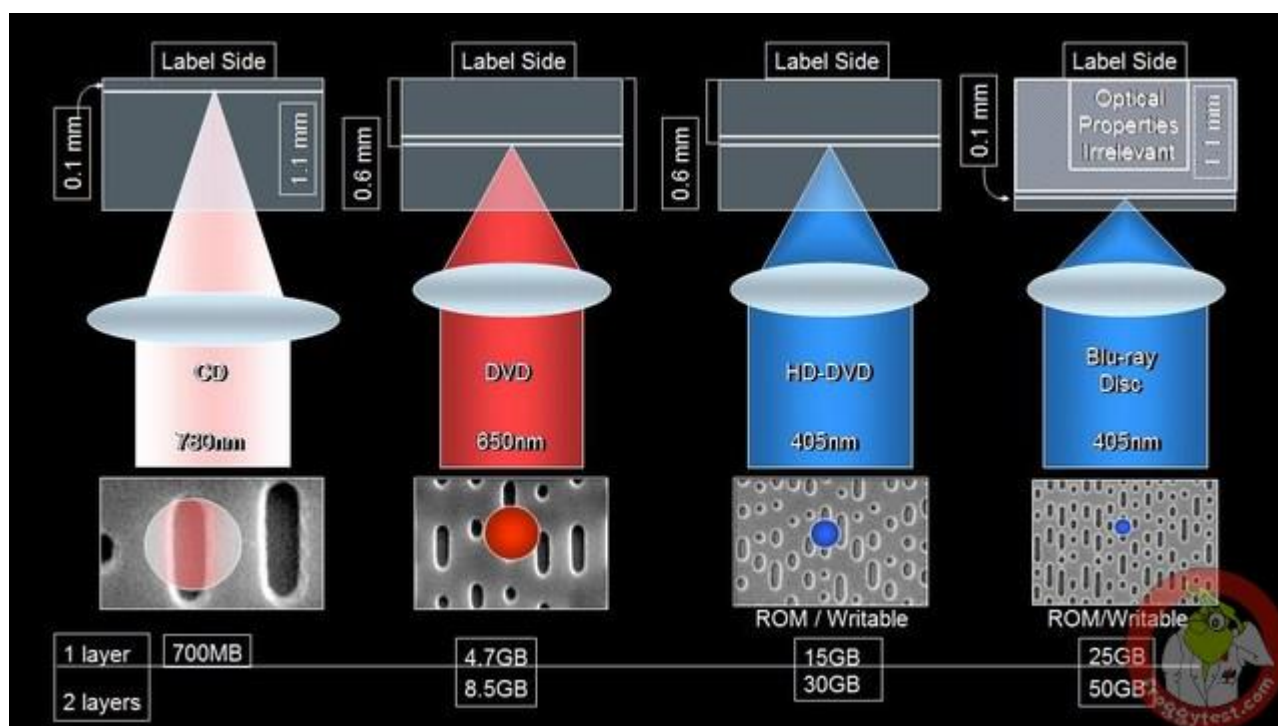
Le Blu-Ray et HD-DVD sont deux nouvelles technologies optiques qui sont les successeurs du DVD. Avec l'introduction de la TV haute définition (HDTV) la capacité de stockage des DVD devient insuffisante. En effet, le DVD supporte une résolution de **720 × 480** pixels, alors que la HDTV fonctionne en **1920 × 1080**. Dans ce cas, 2h de TV HD demandent 22 Go d'espace de stockage (bien supérieur au 17 Go des DVD double face, double couche).

Cette technologie a été développée à partir de 2002 par un consortium constitué d'Apple, Dell, Hitachi, HP, Sony, Pioneer, Panasonic, Sharp, Samsung, Philips, TDK, Thomson. Le HD-DVD, quant à lui, a été développé par Microsoft, HP et Intel.

Le média de stockage est toujours un disque optique de 12cm de diamètre, similaire aux CD/DVD, mais utilisant un laser bleu violet de 405 nm et permettant de stocker 27 Go.

Depuis février 2003, les industriels peuvent acheter la licence leur permettant de fabriquer et commercialiser des lecteurs/graveurs BRD.

Propriété	Blu-Ray	HD-DVD
Capacité	25 / 54 Go (simple/double couche)	15 / 30 Go (simple/double couche)
Codecs	MPEG-2, MPEG-4, AVC, VC-1	



Comparaison CD / DVD /HD-DVD /BRD (Source froggytest.com)

**Novembre 2007** Les spécifications du HD DVD 2.0 triple couche permettant de stocker jusqu'à 51 Go de données sur un média HD DVD en utilisant trois couches de stockage sont acceptées. Grâce à cette standardisation, certains fabricants vont commencer à concevoir des disques HD DVD de 51 Go en masse.

En ce qui concerne le gravage :

- pour un CD : 48X
- pour un DVD : 16X
- pour un BRD : 8X (45min pour graver en vitesse 2X)

## 10.6. Le disque holographique (HVD)

La technologie holographique permet de graver les données en 3 dimensions sous forme d'hologramme. L'intérêt est un plus grand gain de stockage. Un HVD possède une capacité maximale théorique de stockage de 3,9 Téraoctets soit 6000 fois la capacité d'un CD-ROM.

On utilise deux lasers (vert et rouge) qui une fois superposés permettent de lire les données par interférences.

InPhase vient de débiter la commercialisation de disques holographiques de 300 Go. Cette méthode de stockage holographique permet d'offrir de très grosses densités de stockage sur des galettes de la taille d'un CD classique, de 1,5 mm d'épaisseur.

La technologie est arrivée sur le marché en fin d'année dernière, sous la forme d'un premier lecteur Tapestry HDS-300R. Ce format est en revanche réservé aux professionnels pour l'instant, le lecteur coûte à lui seul 18 000 dollars, et il faudra déboursier 180 dollars par disque de 300 Go.

La technologie est coûteuse, mais très prometteuse. InPhase prévoit des disques de 1600 Go d'ici 2010, avec une durée de vie de 50 ans par disque. Parfait pour un archivage durable de masse, avec des débits de transfert annoncés à 20 Mo/s. Une version réenregistrable du disque holographique devrait aussi arriver dès 2008.