

La technologie mémoire ○●○○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○○○ Les problèmes d'écriture ○○○○ Technologie mémoireMémoire cache — 1 / 44

# Technologie mémoire

## Mémoire cache

### Chapitre 6

J.-C. Dubacq

IUT de Villetaneuse  
Université Paris 13

S1D 2009

La technologie mémoire ●○○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○○○ Les problèmes d'écriture ○○○○ Technologie mémoireMémoire cache — 4 / 44

## Qualifier une mémoire

- Méthode d'accès :
  - Séquentiel : accès successif à tous les emplacements, ordre fixe ;
  - Direct : accès en temps constant à n'importe quel élément ;
  - Mixte : voisinage de la donnée, puis accès séquentiel ;
  - Associatif : recherche par clé en temps constant ;
- Type de support physique :
  - Semi-conducteur ;
  - Magnétique ;
  - Optique.
- Résistance à la coupure d'énergie : volatile, non volatile ;
- Effaçable, inscriptible une fois, non effaçable.

La technologie mémoire ○●○○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○○○ Les problèmes d'écriture ○○○○ Technologie mémoireMémoire cache — 5 / 44

## Quantifier une mémoire

- Capacité : nombre de mots, taille du mot (en général 1 octet) ;
- Taille physique et densité de mémoire ;
- Unité de transfert : mot, bloc, page, fichier ;
- Performances :
  - Temps d'accès : obtenir l'information une fois demandée ;
  - Temps de cycle : temps entre deux accès consécutifs ;
  - Latence : Temps d'accès - Temps de cycle ;
  - Débit de transfert :  $\frac{\text{Quantité de données par accès}}{\text{Temps de cycle}}$ .

Rappel : 1Mo = 10<sup>6</sup> o, 1Mio = 2<sup>20</sup> o, 1Ko = 10<sup>3</sup> o, 1Kio = 2<sup>10</sup> o, 1 octet = 8 bits.

La technologie mémoire ○●○○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○○○ Les problèmes d'écriture ○○○○ Technologie mémoireMémoire cache — 6 / 44

## La quantité de données

### Définition (Équation de la quantité de données)

$T = n \times D$  $n = 2^b$	<ul style="list-style-type: none"> <li>● <math>T</math> : capacité totale du support ;</li> <li>● <math>D</math> : taille d'un mot ;</li> <li>● <math>n</math> : nombre de mots adressables ;</li> <li>● <math>b</math> : nombre de bits pour coder une adresse.</li> </ul>
-----------------------------------	---

- 10<sup>3</sup> bits : carte à bande magnétique
- 10<sup>6</sup> bits : un fax d'une page
- 10<sup>9</sup> bits : Capacité d'un CD ou du génome humain
- 10<sup>12</sup> bits : Un disque dur moyen en 2008
- 10<sup>15</sup> bits : 1/10<sup>e</sup>taille des serveurs de Google
- 10<sup>18</sup> bits : Tout ce qui est imprimé dans le monde.

La technologie mémoire ○○○●○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○ Les problèmes d'écriture ○○○○

La mémoire vive Technologie mémoire Mémoire cache — 8 / 44

## Situation physique de la mémoire vive (rappel)

- Le CPU est connecté à un contrôleur mémoire (ou graphique) : *northbridge*.
- northbridge* connecté au *southbridge*, connecté aux périphériques plus lents (joue le rôle de tampon).
- Fréquence du FSB=fréquence bus mémoire (de 100 à 1250 MHz).
- Cycle mémoire : inverse, souvent de 1 à 5 ns.
- Mémoire cache L1 : dans le processeur, avec les registres.
- Mémoire cache L2 : BSB, connexion plus rapide que FSB.

La technologie mémoire ○○○●○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○ Les problèmes d'écriture ○○○○

La mémoire vive Technologie mémoire Mémoire cache — 9 / 44

## La mémoire RAM (ou mémoire vive)

- Lecture/Écriture commandée par signaux électriques : CS (lecture d'adresse), OE (lecture de données), WE (écriture de données) ;

- Volatile ;
- Accès direct : *Random Access Memory*.

### Comparaison DRAM/SRAM

	Statique (SRAM)	Dynamique (DRAM)
Cellule de base	bascule	condensateur
Temps d'accès	10 ns	70 ns
Prix	élevé	faible
Consommation	faible	élevée

La technologie mémoire ○○○●○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○ Les problèmes d'écriture ○○○○

La mémoire vive Technologie mémoire Mémoire cache — 10 / 44

## Le bistable (SRAM)

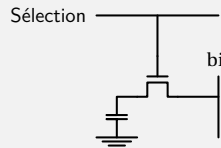
- Cellule complexe : au moins six transistors ;
- Écriture :
  - Appliquer valeur  $X$  à mémoriser sur bit et  $\bar{X}$  sur  $\bar{\text{bit}}$  ;
  - Tension élevée sur ligne sélection ;
- Lecture par comparaison :
  - Tension élevée sur ligne sélection ;
  - bit et  $\bar{\text{bit}}$  sont fournis par la cellule.

La technologie mémoire ○○○●○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ○○○○○○○○○○ Les problèmes d'écriture ○○○○

La mémoire vive Technologie mémoire Mémoire cache — 11 / 44

## Mise en grille de la SRAM (16 mots, 4 bits)

## La DRAM : les gros volumes



- Cellule : 1 condensateur + 1 transistor ;
- Organisation en grilles (lignes/colonnes) similaire à la SRAM : les colonnes sont les adresses successives, les lignes sont les paquets d'adresses.
- Chaque bit (dans un même octet) est dans une grille différente (on lit plusieurs grilles à la fois pour avoir un octet complet).
- La ligne **bit** passe par toutes les cellules d'une même colonne. Au bout, un système (RAS) la connecte à un amplificateur opérationnel qui peut *lire* les variations de tension et en *imposer* une autre : en dessous de  $V_{dd}/2$ , c'est de plus en plus petit (jusqu'à  $V_{ss}$ ) qui est imposé, au-dessus, c'est de plus en plus grand (jusqu'à  $V_{dd}$ ).

## Le mode page

- Registre à l'entrée des décodeurs de lignes et de colonnes, synchronisés sur RAS (lignes) et CAS (colonnes) ;
- Une fois mémorisée, une (demie-)adresse sert jusqu'à être remplacée ;
- Méthode de lecture normale : mémoriser adresse de rang, mémoriser adresse de colonne : lecture du bit voulu ;
- Méthode de lecture en mode page : compteur interne pour lire toutes les colonnes (pas de CAS) ou boucle, mémoriser adresse de rang, mettre 0 dans adresse colonne (lire), puis mettre 1 dans adresse colonne (lire), puis recommencer jusqu'à dernière colonne ;
- Efficacité accrue : 1 cycle pour mémoriser la ligne, puis 1 cycle par colonne.
- En plus, pas de temps de préchargement si on lit tout une page.

## La DRAM : lecture, écriture, rafraîchissement

### Lecture d'un bit

- 1 RAS est fermé, et les lignes **bit** sont préchargées à  $V_{dd}/2$ , puis sélection à 1 pour toute la ligne.
  - 2 Si cond. chargé, dérive de tension vers  $V_{dd}$ , sinon vers  $V_{ss}$  ;
  - 3 Détection, puis changement de **bit** en valeur lue par l'AO : *réécriture*
  - 4 Seul le bit demandé est sorti de la mémoire. Tant que la « page » (ligne) est ouverte, on peut lire très vite plusieurs colonnes (pas de préchargement à refaire).
- **Rafraîchissement** : une ligne doit être lue régulièrement (64ms) pour conserver la tension des condensateurs. Un compteur interne trace la ligne courante de rafraîchissement.
  - **Écriture** : identique à lecture, sauf que le bit concerné est forcé au lieu d'être lu par l'AO.

## Ordres de grandeur de la RAM

- Performances de la SRAM : latence de 7 ns pour les plus performantes, 70 ns pour les anciennes.
- Performances de la SRAM en mode page (on commence à demander une réponse alors que la précédente n'est pas encore arrivée) : un peu moins de la moitié.
- Taille d'une DRAM : chaque ligne comporte jusqu'à des centaines de milliers de cellules.
- Performances de la DRAM en accès aléatoire : 25 à 40 ns pour DDR-1 (et environ 20 ns pour précharger).
- On agrandit la largeur du bus mémoire pour augmenter le débit (lecture par mots de plus en plus gros) et on accélère l'horloge des DRAM (granularité beaucoup plus fine, moins de temps perdu).

La technologie mémoire 0000000000●00 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 La mémoire vive Technologie mémoireMémoire cache — 16 / 44

## Beaucoup de types de RAM

- DRAM, SRAM, déjà vues ;
- NV-RAM : non-volatile RAM (plusieurs types) ;
- Fast Page Mode DRAM : améliore la vitesse en mode page ;
- Extended Data Out DRAM (1993) : permet de précharger un rang sans perdre les données lues ;
- Synchronous DRAM (1997) : travail en mode synchrone, permet d'enchaîner les opérations sans attendre ;
- DDR SDRAM (2000) : travaille sur des cycles deux fois plus courts ;
- DRDRAM, VRAM, SGRAM existent aussi ;
- Pseudo-Static RAM : DRAM utilisée comme de la SRAM.

La technologie mémoire 0000000000●00 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 La mémoire non-volatile Technologie mémoireMémoire cache — 19 / 44

## Les mémoires non-volatiles

- Inventée dans les années 50 ;
- Maintenant quasi-exclusivement de la mémoire Flash ;
- PROM : possibilité d'enregistrer le contenu en ROM (signaux électriques grillent des contacts) ;
- EPROM : PROM effaçable par rayonnement UV ;
- EEPROM : PROM effaçable par signaux électriques ;
- Mémoire Flash : une forme d'EEPROM ;
- Mémoire Flash : taille de mot plutôt 512 octets.

La technologie mémoire 0000000000●00 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 La mémoire non-volatile Technologie mémoireMémoire cache — 18 / 44

## ROM : Read-Only Memory

- Cellule de base : 1 fil (contact ouvert ou fermé) ;
- Information figée à la fabrication ;
- Coût élevé (conception) ;
- Délai important de fabrication ;
- Non-volatile, non-effaçable ;
- Utile pour microprogramme, bibliothèques, programmes systèmes ;
- Légèrement plus lent que RAM : utilisation de *shadow ram* (recopie en RAM de la ROM, et désactivation de la ROM).

La technologie mémoire 0000000000●00 La hiérarchie mémoire ●000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 Problématique Technologie mémoireMémoire cache — 21 / 44

## Vitesse CPU et mémoire

- SRAM rapide, mais chère ;
- DRAM plus lente, moins chère ;
- Chaque variante a un rapport vitesse-prix-taille différent ;
- Problème : accès à la mémoire nécessaire à chaque cycle processeur ;
- Fréquence processeur : Entre 1 et 3 GHz ;
- Fréquence mémoire : autour de 200 MHz ;
- 1 cycle mémoire = 5 à 15 cycles processeur !
- Comment donner l'illusion d'une mémoire rapide, de grande capacité, et pas chère ?

La technologie mémoire 000000000000 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 Problématique Technologie mémoireMémoire cache — 22 / 44

## Hiérarchie mémoire

Registres	} semi-conducteur	{	Souvent volatile
SRAM			Faible capacité
DRAM			Chère
Disque magnétique	} mémoire secondaire	{	Accès séquentiel ou mixte
Disque optique			Lente
Bande			Volumineuse

On bâtit une hiérarchie qui va du processeur vers les supports les plus gros, plus lents et moins chers.  
 Quand on a plus assez dans un niveau, on prend dans le suivant.

La technologie mémoire 000000000000 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 Problématique Technologie mémoireMémoire cache — 24 / 44

## Pourquoi ça marche ?

- Principe de localité : à tout instant, un programme accède à une partie petite de son espace d'adressage ;
- Deux types de localité :
  - Temporelle Si un mot a été utilisé récemment, il a plus de chances d'être réutilisé (exemple : segment de données) ;
  - Spatiale Si un mot a été utilisé récemment, les mots avec adresses voisines ont plus de chances d'être utilisés (exemple : segment de texte, tableaux) ;
- On fait donc migrer plus près du processeur les données les plus récemment accédées ;
- On fait aussi migrer les données ayant des adresses voisines, donc on déplace toujours les données en blocs.

La technologie mémoire 000000000000 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 Problématique Technologie mémoireMémoire cache — 23 / 44

## Principe de la hiérarchie mémoire

- À tout moment, on ne copie des informations que d'un niveau vers le niveau immédiatement adjacent ;
- Le niveau supérieur est plus près du processeur ;
- Il est plus rapide, plus cher, plus petit ;
- Le niveau inférieur est plus loin du processeur ;
- Il est plus gros, plus lent, moins cher ;
- Unité de transfert : bloc (souvent).

La technologie mémoire 000000000000 La hiérarchie mémoire 000000 La mémoire cache 0000000000 Les problèmes d'écriture 0000  
 Problématique Technologie mémoireMémoire cache — 25 / 44

## Terminologie

- Quand on essaye d'accéder à une information à un niveau, on peut la trouver, ou ne pas la trouver. Quand on la trouve, on a un succès, et sinon un échec ;
- Le taux de succès  $\tau_S$  est la proportion des accès réussis ;
- Le taux d'échec  $\tau_E = 1 - \tau_S$  est le contraire ;
- Le temps de succès  $T_S$  est le temps d'accès dans le cas d'un succès ;
- La pénalité d'échec  $T_P$  est le temps de déplacement d'un bloc de la mémoire de niveau inférieur vers le niveau supérieur. En cas d'échec, le temps d'accès est  $T_E = T_S + T_P$  ;
- On a bien sûr  $T_P \gg T_S$ .

La technologie mémoire ○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○● La mémoire cache ○○○○○○○○○○ Les problèmes d'écriture ○○○○

Problématique Technologie mémoireMémoire cache — 26 / 44

## La hiérarchie mémoire d'un ordinateur

- Registres (1ns, 100 octets);
- Mémoire cache niveau 1 (1ns, 10-100 kio);
- Mémoire cache niveau 2 (externe) (SRAM) (10ns, 100 à 1000 kio);
- Mémoire cache niveau 3 (parfois) (DRAM) (1 à 100 Mio);
- Mémoire principale (DRAM) (100 à 10000 Mio);
- Disques durs (10 à 1000 Go);
- Bandes (quelques centaines de Go);
- Réseau...

La technologie mémoire ○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ●○○○○○○○○○ Les problèmes d'écriture ○○○○

Organisation de la mémoire cache Technologie mémoireMémoire cache — 29 / 44

## Méthodologie de la mémoire cache

- On a besoin d'informations : on va les chercher uniquement à côté;
- Défaut de cache ramène l'information;
- Analogie : épicerie locale;
- Quand le produit demandé n'est pas là, l'épicerie la demande au grossiste en grande quantité et donne le produit demandé;
- Le grossiste est plus loin, mais a plus de choix.

La technologie mémoire ○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ●○○○○○○○○○ Les problèmes d'écriture ○○○○

Organisation de la mémoire cache Technologie mémoireMémoire cache — 28 / 44

## Performance d'une mémoire cache

- On veut un temps d'accès moyen réduit;
- Temps d'accès moyen :  $T_m = T_s + \tau_E \times T_P$

**Exemple**

- Accès SRAM : 10ns;
- Accès DRAM : 70ns;
- Accès DRAM en mode page : 20ns;
- Taux succès : 90% en mode page, 30%;

Temps moyen normal :  $10 + 0,7 \times 70 = 59 \text{ ns}$   
 Temps moyen mode page (8 mots) :  $10 + 0,1 \times (70 + 7 \times 20) = 31 \text{ ns}$

La technologie mémoire ○○○○○○○○○○○○ La hiérarchie mémoire ○○○○○○ La mémoire cache ●○○○○○○○○○ Les problèmes d'écriture ○○○○

Organisation de la mémoire cache Technologie mémoireMémoire cache — 30 / 44

## Découpage d'une adresse pour identifier

- Classifier l'information : connue uniquement par son adresse;
- Répartition en blocs :  $x$  bits de poids faibles utilisés comme colonne, reste utilisé comme ligne;
- Un bloc est donc identifié par son numéro (les bits de poids fort);
- Découpage en champs!

$n$	$x$	$0$
numéro de page	numéro de colonne	

## Le cache associatif

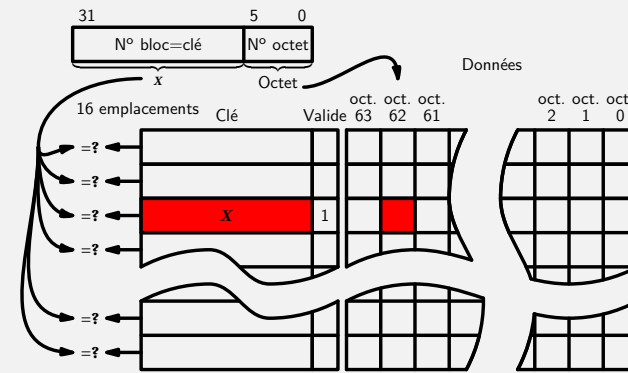
- On demande une adresse, dans un certain bloc  $X$  ;
- La mémoire cache contient un certain nombre de blocs, ainsi que l'indication de leurs numéros ;
- Les blocs sont de toute façon tous aussi dans la RAM ;
- Chacun de ces emplacements peut contenir le bloc demandé  $X$  ou un autre ;
- On parcourt chacun des emplacements de blocs, et on regarde si on trouve  $X$  ;
- Si on trouve  $X$ , succès (on cherche le bon octet, et on l'envoie) ;
- Sinon, échec (on trouve un bloc à éliminer, et on met  $X$  à la place).

## Politique de remplacement

- Bit « valide » décrit si le numéro du bloc correspond bien à une copie de la mémoire ;
- Quand il y a échec d'accès, il faut remplacer un bloc par le bloc voulu ;
- Remplacement aléatoire : un bloc au hasard est éliminé ;
- Remplacement LRU (*least recently used*) : le matériel garde la trace des accès mémoires les plus récents, on remplace le bloc le moins utilisé récemment ;
- Remplacement du plus vieux : le matériel garde la trace du bloc le plus ancien, on remplace ce bloc ;
- On met le nouveau bloc à la place.

## Représentation d'un cache associatif

On prend 32 bits d'adresse, 64 octets/bloc, cache de 1 kio

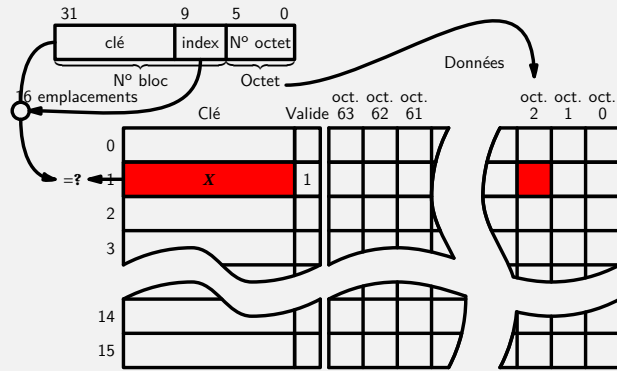


## Le cache direct

- Défaut du cache associatif : parcours de tous les emplacements pour trouver le bon bloc ;
- Cache direct : les bits de poids faible du numéro de bloc forment un *index cache* ;
- Les autres bits du numéro de bloc forment la clé ;
- Un bloc ne peut aller que dans un seul emplacement ;
- La clé est plus courte ;
- Deux blocs consécutifs ne sont pas dans le même emplacement (car  $index = \text{poids faible}$ ) ;
- Choix direct du bloc à remplacer en cas d'échec (un seul bloc possible).

## Cache direct

32 bits d'adresse, 64 octets/bloc, cache de 1 kio

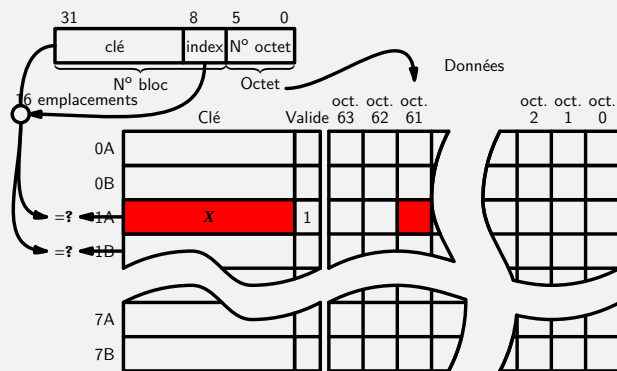


## Cache associatif par ensembles à n voies

- Cache associatif : beaucoup d'emplacements équivalents, stratégie de remplacement et identification complexe ;
- Cache direct : un seul emplacement pour un bloc, stratégie et identification simple ;
- Mais effet ping-pong entre blocs possible !
- Cache associatif par ensembles à  $n$  voies : comme cache direct, mais avec le choix entre plusieurs emplacements ;
- 2 voies donnent 2 emplacements possibles pour un bloc, 4 voies donnent 4 emplacements...
- taille de l'index cache diminue si quantité totale de mémoire constante.

## Cache associatif à 2 voies

2 voies, 32 bits d'adresse, 64 octets/bloc, cache de 1 kio



## Hiérarchie de cache

- Politique de remplacement pour cache associatif par ensembles : comme cache associatif, mais l'index est fixe (on doit jeter un bloc qui a le bon index) ;
- Évite l'effet ping-pong ;
- Cache direct équivalent à cache associatif par ensembles à 1 voie, et cache associatif équivalent à un cache associatif par ensembles à  $k$  voies ( $k=nb$  blocs).
- Ces méthodes de cache s'appliquent entre le cache niveau 1 (sur processeur) et cache niveau 2 (externe) ;
- Mais aussi entre cache niveau 2 et RAM/cache niveau 3 ;
- Souvent, le cache niveau 1 est divisé en cache d'instructions et cache de données ;

## Cohérence de cache

- La mémoire peut parfois être modifiée indépendamment du processeur (ex : contrôleur DMA) ;
- Le cache n'est alors plus une copie fidèle de la mémoire ;
- Dans ce cas, on invalide le bloc du cache (bit « valide » à 0) ;
- Problème similaire dans le cas de l'écriture ;
- On modifie la valeur  *dans le cache* ;
- Que doit-on faire pour la mémoire ?

## Écriture en cas d'échec cache

- Pour écrire, pas besoin de lire ;
- Méthode 1 : écrire uniquement dans mémoire ;
- Pas de déplacement du bloc vers le cache ;
- taux d'échec reste élevé ;
- En cas d'écriture au même endroit, très lent (cache inutile) ;
- marche bien avec écriture immédiate ;
- Méthode 2 : écriture avec allocation ;
- taux d'échec réduit ;
- complexe et trafic intense ;
- marche bien avec écriture différée.

## Écriture en cas de succès cache

- Méthode 1 : écriture immédiate ;
- écrire en même temps dans le cache et dans la mémoire ;
- méthode lente ;
- Méthode 2 : écriture différée ;
- on écrit seulement dans le cache ;
- on retient un bit par bloc de cache pour savoir s'il a été modifié ;
- quand un bloc est jeté, il doit d'abord être copié du cache vers la mémoire ;
- réduit le trafic, mais complexe.

## Conclusion sur le cache

- Paramètres nombreux :
  - temps moyen ;
  - taux succès ;
  - taille cache ;
  - taille bloc ;
  - associativité (nombre de voies) ;
  - stratégie de remplacement ;
  - écriture immédiate ou différée ;
  - allocation ou non sur échec en écriture ;
- Intégration dans la hiérarchie mémoire ;
- Utilisation en programmation.