cours 4

la stratégie mémoire

cours 4

la stratégie mémoire

à ce jour, il n'existe pas de technologie optimale pour satisfaire le besoin en mémoire d'un ordinateur

plan

– caractéristiques

plan

- caractéristiques
- modes d'accès

plan

- caractéristiques
- modes d'accès
- mémoire principale

plan

- caractéristiques
- modes d'accès
- mémoire principale
- mémoire cache

plan

- caractéristiques
- modes d'accès
- mémoire principale
- mémoire cache
- pentium II et SDRAM

caractéristiques

- types physiques de mémoires
- durée de mémorisation
- emplacement
- capacité
- performance
- hiérarchie mémoire

définitions

définitions

mémoire : dispositif capable de *conserver* et *restituer* une information

définitions

mémoire : dispositif capable de *conserver* et *restituer* une information

mot mémoire : ensemble de bits pouvant être lus ou écrits *simultanément*

les différents types physiques de mémoires

– semi-conducteur (registre, mémoire principale, ...)

les différents types physiques de mémoires

– semi-conducteur (registre, mémoire principale, ...)
– magnétique (disquette, disque dur, ...)

les différents types physiques de mémoires

– semi-conducteur (registre, mémoire principale, ...)
– magnétique (disquette, disque dur, ...)
– optique (cd-rom, dvd-rom, ...)

durée de mémorisation

durée de mémorisation

– fonction du temps

durée de mémorisation

- fonction du temps
  - quasi-permanente : disque, ROM

durée de mémorisation

- fonction du temps
  - quasi-permanente : disque, ROM
  - temporaire : RAM

durée de mémorisation

- fonction du temps
  - quasi-permanente : disque, ROM
  - temporaire : RAM
- fonction de la présence d'alimentation électrique

durée de mémorisation

- fonction du temps
  - quasi-permanente : disque, ROM
  - temporaire : RAM
- fonction de la présence d'alimentation électrique
  - sensible : RAM

durée de mémorisation

- fonction du temps
  - quasi-permanente : disque, ROM
  - temporaire : RAM
- fonction de la présence d'alimentation électrique
  - sensible : RAM
  - insensible : disque

emplacement

emplacement

– interne au processeur : registre

emplacement

– interne au processeur : registre
– interne à la carte mère : mémoire principale

emplacement

- interne au processeur : registre
- interne à la carte mère : mémoire principale
- externe à la carte mère (mémoire secondaire) : disque

emplacement

- interne au processeur : registre
- interne à la carte mère : mémoire principale
- externe à la carte mère (mémoire secondaire) : disque
- externe à l'unité centrale (mémoire tertiaire) : bande magnétique

capacité

nombre d'informations stockables

exprimée en octet (byte) ou en multiple d'octet

- kilo 1K $= 2^{10} = 1024$
- méga 1M $= 2^{20} = 1\ 048\ 576$
- giga 1G $= 2^{30} = 1\ 073\ 741\ 824$
- téra 1T $= 2^{40} = 1\ 099\ 511\ 627\ 776$
- péta 1P $= 2^{50} = 1\ 125\ 899\ 906\ 842\ 620$

capacité

capacité

1 caractère ('a', '?', '2', ...) = 1 octet

capacité

1 caractère ('a', '?', '2', ...) = 1 octet

le petit robert (2600 pages) = environ 180 Mbits

performance

performance

temps d'accès: temps nécessaire à une opération de lecture/écriture

performance

temps d'accès : temps nécessaire à une opération de lecture/écriture

débit : quantité d'informations lues/écrites par unités de temps (exemple : Mo/s)

performance (ordre de grandeur)

ordre	registre	cache	mémoire principale	disque
capacité	1 Ko	0,1 - 1 Mo	0,1 - 1 Go	10 - 100 Go
temps d'accès		1 ns	1 - 10 ns	1 - 10 ms
débit	100 Go/s	10 Go/s	1-10 Go/s	100 Mo/s

4 Go/s : 32 bits toutes les nanosecondes !

hiérarchie

l'idéal ?

hiérarchie

l'idéal ? posséder une mémoire illimitée et très rapide !

hiérarchie

l'idéal ? posséder une mémoire illimitée et très rapide !

mais

hiérarchie

l'idéal ? posséder une mémoire illimitée et très rapide !

mais

1. on ne sait pas fabriquer une mémoire illimitée

hiérarchie

l'idéal ? posséder une mémoire illimitée et très rapide !

mais

1. on ne sait pas fabriquer une mémoire illimitée
2. le temps d'accès augmente avec la capacité

hiérarchie

l'idéal ? posséder une mémoire illimitée et très rapide !

mais

1. on ne sait pas fabriquer une mémoire illimitée
2. le temps d'accès augmente avec la capacité

l'idée :

hiérarchie

l'idéal ? posséder une mémoire illimitée et très rapide !

mais

1. on ne sait pas fabriquer une mémoire illimitée
2. le temps d'accès augmente avec la capacité

l'idée : seules les données les plus utilisées nécessitent un temps d'accès très petit

hiérarchie

la mémoire est organisée en une hiérarchie

hiérarchie

la mémoire est organisée en une hiérarchie

– de la mémoire la plus rapide à la moins rapide

hiérarchie

la mémoire est organisée en une hiérarchie

– de la mémoire la plus rapide à la moins rapide
– de la capacité la plus faible à la capacité la plus grande

hiérarchie

la mémoire est organisée en une hiérarchie

– de la mémoire la plus rapide à la moins rapide
– de la capacité la plus faible à la capacité la plus grande
– du composant le plus couteux au composant le moins couteux

modes d'accès

- accès aléatoire
- accès par le contenu
- accès séquentiel
- accès direct

accès aléatoire

le mode d'accès le plus employé

– mémoire principale
– mémoires caches

accès aléatoire

chaque mot mémoire est associé à une adresse unique

accès aléatoire

chaque mot mémoire est associé à une adresse unique

fonctionnement : cf. chapitre introduction

accès aléatoire

chaque mot mémoire est associé à une adresse unique

fonctionnement : cf. chapitre introduction

la taille d'une adresse dépend de la capacité

taille de l'adresse

capacité : 4 Go (i.e., $4 \times 2^{30} \times 8$ bits)

taille de l'adresse

capacité : 4 Go (i.e., $4 \times 2^{30} \times 8$ bits)

taille du mot adressable : 1 octet

taille de l'adresse

capacité : 4 Go (i.e., $4 \times 2^{30} \times 8$ bits)

taille du mot adressable : 1 octet

taille de l'adresse : 32 bits ($log_2(4 \times 2^{30})$)

accès aléatoire

opérations associées à ce mode d'accès

– lecture(adr)
– écriture(adr,donnée)

temps d'accès indépendant des accès précédents

accès LIFO

accès à des données résidant dans une pile

utilisation

- appel et de retour de sous programme
- sauvegarde de contexte
  - suspension de programme
  - interruption

accès LIFO

opérations associées à ce mode d'accès

– écriture(donnée)
– lecture
– sommet
– vide

exemple

opération	donnée lue
écriture(abc)	
écriture(def)	
écriture(ghi)	
lecture()	ghi
écriture(jkl)	
lecture()	jkl
lecture()	def
lecture()	abc

accès par le contenu

employé principalement par les mémoire caches

accès par le contenu

employé principalement par les mémoire caches

pas de notion d'adresse !

accès par le contenu

employé principalement par les mémoire caches

pas de notion d'adresse !

un mot est retrouvé par une partie de son contenu

accès par le contenu

divisée en 2 parties :

1. une partie contenant un descripteur unique (clé)
2. une partie contenant le mot associé à la clé

accès par le contenu

divisée en 2 parties :

1. une partie contenant un descripteur unique (clé)
2. une partie contenant le mot associé à la clé

lors d'une opération (lecture/écriture) une clé peut être comparée en parallèle avec toutes les clés stockées

accès par le contenu

opérations associées à ce mode d'accès

- écriture(clé,donnée)
- lecture(clé)
- existe(clé)
- retirer(clé)

temps d'accès constant

exemple

clé	donnée
c_3	données$_3$
c_2	données$_2$
c_1	données$_1$
c_0	données$_0$

exemple

lecture(c_x)

exemple

lecture(c_x)

– c_x est comparé simultanément avec tous les c_i

exemple

lecture(c_x)

– c_x est comparé simultanément avec tous les c_i

– si $c_i = c_x$ alors renvoyer donnees$_x$

accès séquentiel

archivage d'importants volumes de données

accès séquentiel

archivage d'importants volumes de données

employé par les bandes magnétiques

accès séquentiel

archivage d'importants volumes de données

employé par les bandes magnétiques

informations écrites les une derrière les autres

accès séquentiel

archivage d'importants volumes de données

employé par les bandes magnétiques

informations écrites les une derrière les autres

pour accéder à une donnée, il faut avoir lu les précédentes

accès séquentiel

opérations associées à ce mode d'accès

– début : se positionner sur la première donnée
– lecture : lire une donnée
– écriture(données) : écrire donnée
– fin : se positionner après la dernière donnée

temps d'accès variable

accès direct

employé par les disques, les CD

accès direct

employé par les disques, les CD

données regroupées en blocs

accès direct

employé par les disques, les CD

données regroupées en blocs

chaque bloc est associé à une adresse unique

accès direct

employé par les disques, les CD

données regroupées en blocs

chaque bloc est associé à une adresse unique

accéder à une donnée

accès direct

employé par les disques, les CD

données regroupées en blocs

chaque bloc est associé à une adresse unique

accéder à une donnée

– accéder au bloc qui la contient

accès direct

employé par les disques, les CD

données regroupées en blocs

chaque bloc est associé à une adresse unique

accéder à une donnée

– accéder au bloc qui la contient
– se déplacer séquentiellement jusqu'à sa position

accès direct

opérations associées à ce mode d'accès

– lecture(bloc,déplacement)
– écriture(bloc,déplacement,donnée)

temps d'accès variable

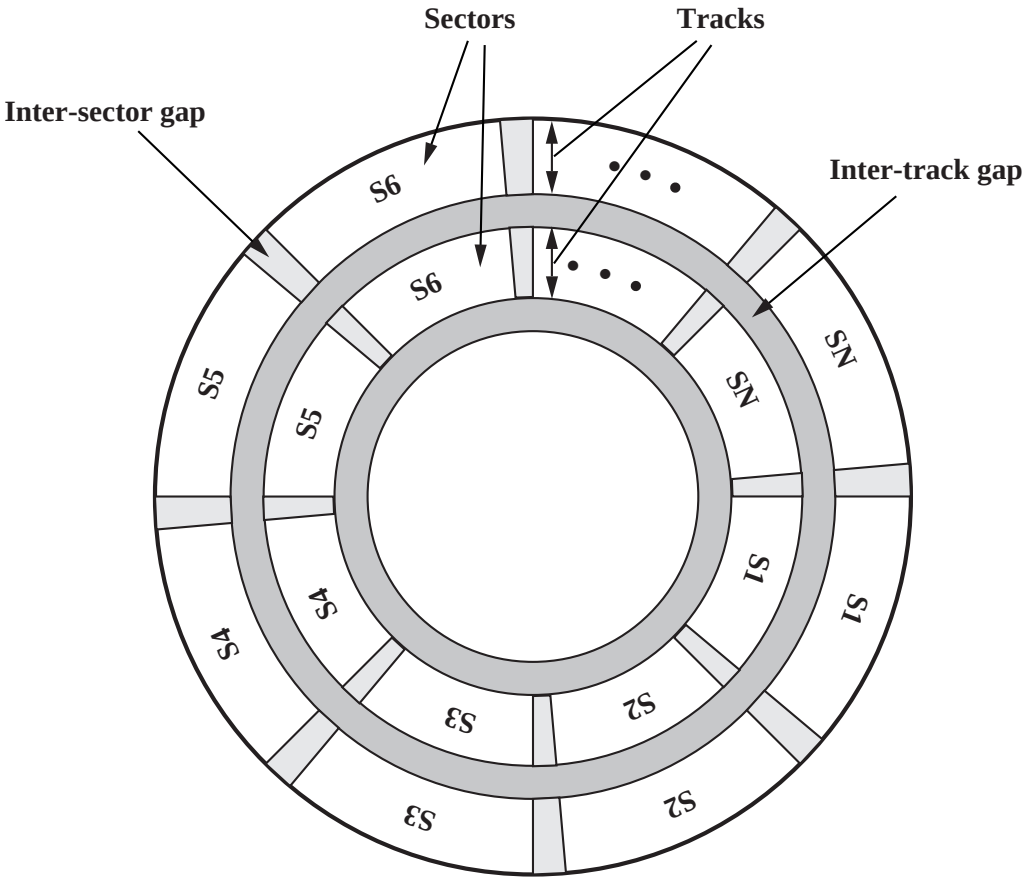


Figure 5.1 Disk Data Layout

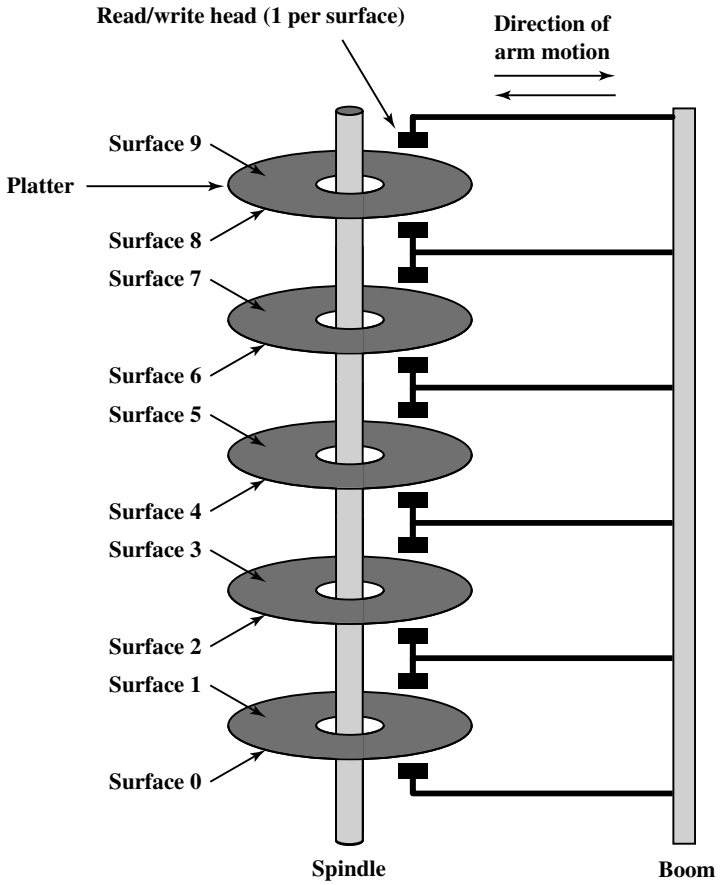


Figure 6.5 Components of a Disk Drive

Table 6.2 Typical Hard Disk Drive Parameters

Characteristics	Seagate Barracuda 180	Seagate Cheetah X15-36LP	Seagate Barracuda 36ES	Toshiba HDD1242	IBM Microdrive
Application	High-capacity server	High-performance server	Entry-level desktop	Portable	Handheld devices
Capacity	181.6 GB	36.7 GB	18.4 GB	5 GB	1 GB
Minimum track-to-track seek time	0.8 ms	0.3 ms	1.0 ms	—	1.0 ms
Average seek time	7.4 ms	3.6 ms	9.5 ms	15 ms	12 ms
Spindle speed	7200 rpm	15K rpm	7200	4200 rpm	3600 rpm
Average rotational delay	4.17 ms	2 ms	4.17 ms	7.14 ms	8.33 ms
Maximum transfer rate	160 MB/s	522 to 709 MB/s	25 MB/s	66 MB/s	13.3 MB/s
Bytes per sector	512	512	512	512	512
Sectors per track	793	485	600	63	—
Tracks per cylinder (number of platter surfaces)	24	8	2	2	2
Cylinders (number of tracks on one side of platter)	24,247	18,479	29,851	10,350	—

mémoire principale

mémoire principale

l'accès à la mémoire principale est le chemin le plus important dans l'ordinateur

mémoire principale

- types
- organisation
- mémoire dynamique

définition

mémoire

– interne
– à base de semi-conducteurs
– mode d'accès aléatoire

types

– mémoires volatiles : RAM (Random Access Memory)
– mémoire non volatiles : ROM (Read Only Memory)

mémoire RAM

stocke des données temporaires

principalement 2 types

- RAM dynamique (DRAM)
  - utilisée pour la mémoire principale
- RAM statique (SRAM)
  - principalement utilisée pour les caches

mémoire RAM

– DRAM

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation
  - nécessitent un rafraîchissement périodique

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation
  - nécessitent un rafraîchissement périodique
  - simples, denses, peu couteuses

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation
  - nécessitent un rafraîchissement périodique
  - simples, denses, peu couteuses
- SRAM

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation
  - nécessitent un rafraîchissement périodique
  - simples, denses, peu couteuses
- SRAM
  - bascules comme unités de mémorisation

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation
  - nécessitent un rafraîchissement périodique
  - simples, denses, peu couteuses
- SRAM
  - bascules comme unités de mémorisation
  - rafraîchissement inutiles

mémoire RAM

- DRAM
  - condensateurs comme unités de mémorisation
  - nécessitent un rafraîchissement périodique
  - simples, denses, peu couteuses
- SRAM
  - bascules comme unités de mémorisation
  - rafraîchissement inutiles
  - rapides, couteuse

mémoire ROM

Read-Only, Read-Mostly Memory

stocke des informations permanentes

– programmes systèmes
– microprogrammes
– . . .

mémoire ROM

plusieurs types

mémoire ROM

plusieurs types

– ROM : écriture unique lors de la fabrication

mémoire ROM

plusieurs types

– ROM : écriture unique lors de la fabrication
– PROM : écriture unique après fabrication

mémoire ROM

plusieurs types

- ROM : écriture unique lors de la fabrication
- PROM : écriture unique après fabrication
- EPROM
  - admet un nombre d'écriture limité
  - effaçable par ultra-violet

mémoire ROM

plusieurs types

- ROM : écriture unique lors de la fabrication
- PROM : écriture unique après fabrication
- EPROM
  - admet un nombre d'écriture limité
  - effaçable par ultra-violet
- mémoire flash : effaçable électriquement

Table 5.1 Semiconductor Memory Types

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level		
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		

organisation

élément de base : la cellule mémoire

3 connexions

– une entrée de sélection
– une entrée de contrôle
  (Output Enable ou Write Enable)
– une ligne bidirectionnelle de donnée

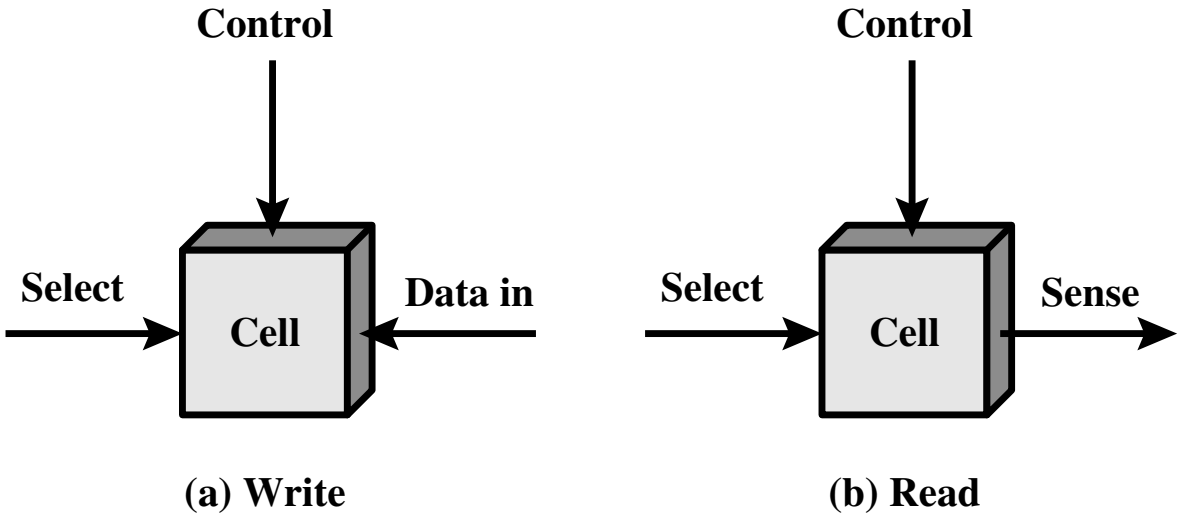


Figure 5.1 Memory Cell Operation

organisation

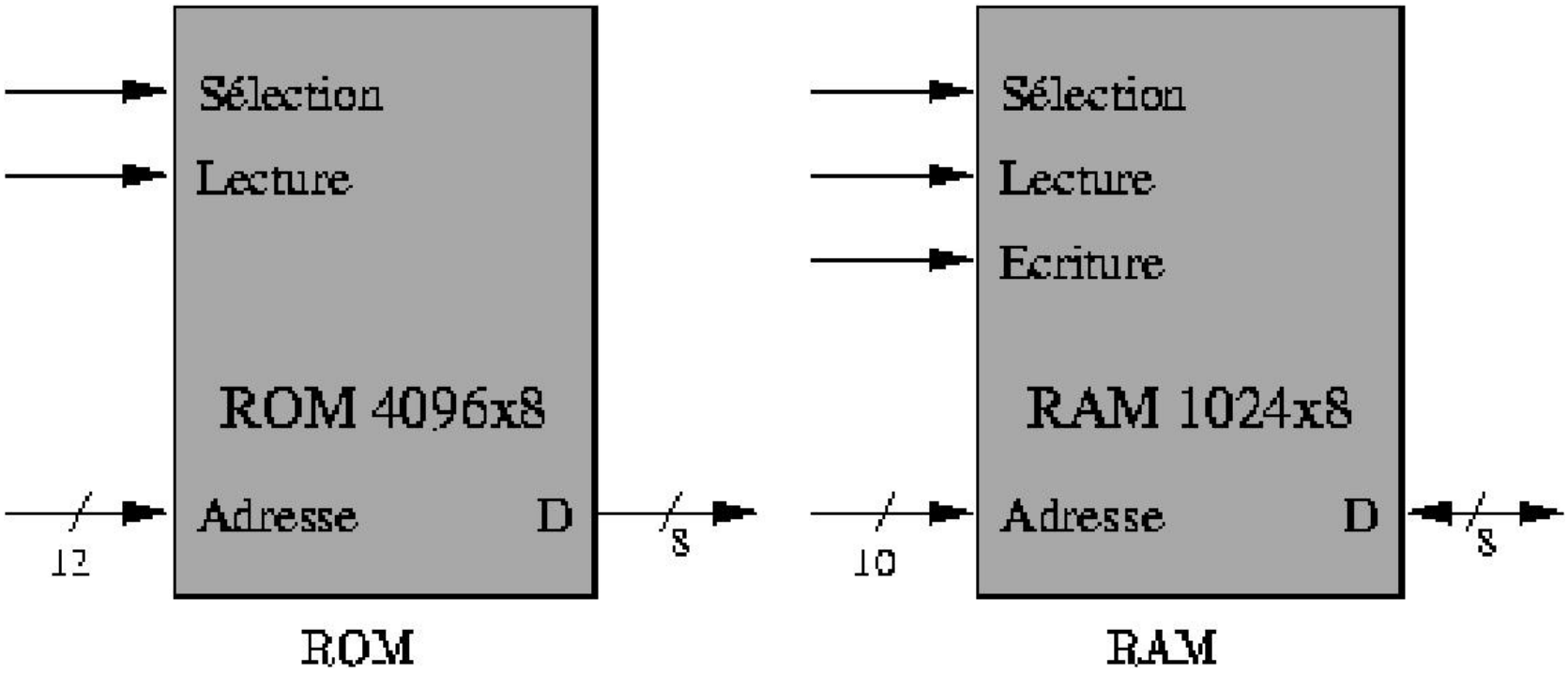
circuit mémoire RAM de M mots de B bits

organisée comme une matrice de M lignes et B colonnes

$log_2(M)$ lignes d'adresse

B lignes de données

exemple


Sélection
Lecture
ROM 4096x8
Adresse
D
12
8
ROM
Sélection
Lecture
Ecriture
RAM 1024x8
Adresse
D
10
8
RAM

exemple

un espace adressable de 2^{16} mots de 32 bits

4096 mots mémoires en RAM

4096 mots mémoire en ROM

utilisation de

– circuits RAM de 1024×8 bits
– circuits ROM de 4096×8 bits

exemple

un mot = 4 circuits de 8 bits en parallèle

carte d'adresse mémoire

- 4096 mots mémoires RAM aux adresses les plus basses (de 0 à 4096),
- 4096 mots de mémoires ROM aux adresses les plus élevées (de 61440 à 65535)
- autres adresses inoccupées

exemple

- 4 étages de RAM (adresses de 0 à 4096)
  - adresses de 0 à 1023
  - adresses de 1024 à 2047
  - adresses de 2048 à 3071
  - adresses de 3072 à 4095
- 1 étage de ROM (adresses de 61440 à 65535)

exemple

étage	bits d'adresse
RAM 0	0000 00xx xxxx xxxx
RAM 1	0000 01xx xxxx xxxx
RAM 2	0000 10xx xxxx xxxx
RAM 3	0000 11xx xxxx xxxx
inoccupé	
ROM 4	1111 xxxx xxxx xxxx

exemple

trouver un mot dans cette mémoire

exemple

trouver un mot dans cette mémoire

– trouver l'étage

exemple

trouver un mot dans cette mémoire

– trouver l'étage
– trouver l'adresse dans l'étage

convention

adresse de l'octet de poids faible dans le mot

convention

adresse de l'octet de poids faible dans le mot

– convention petit bout (little endian)

convention

adresse de l'octet de poids faible dans le mot

– convention petit bout (little endian)
  – adresse la plus faible du mot
  – employé par la famille i386

convention

adresse de l'octet de poids faible dans le mot

- convention petit bout (little endian)
  - adresse la plus faible du mot
  - employé par la famille i386
- convention gros bout (big endian)

convention

adresse de l'octet de poids faible dans le mot

- convention petit bout (little endian)
  - adresse la plus faible du mot
  - employé par la famille i386
- convention gros bout (big endian)
  - adresse la plus élevée du mot
  - employé par la famille 68000

exemple

petit bout	gros bout
03 02 01 00	00 01 02 03
07 06 05 04	04 05 06 07
11 10 09 08	08 09 10 11
15 14 13 12	12 13 14 15
...	...

mémoire dynamique DRAM

– nécessite un rafraîchissement
– grande capacité
– organisée en matrice
– multiplexage du bus adresse

mémoire dynamique DRAM

– nécessite un rafraîchissement
– grande capacité
– organisée en matrice
– multiplexage du bus adresse

base de la mémoire principale depuis plus de 20 ans

DRAM

l'adresse de n bits d'un mot est envoyé en 2 fois $n/2$ bits

utilisation des signaux

– RAS (Row Adress Select) : bits de poids faible
– CAS (Column Adress Select) : bits de poids fort

exemple

mémoire de 16 Mbit

4 méga mots de 4 bits

adresse sur 22 bits

4 matrices de 2048×2048 éléments

mot formé par un élément dans chaque matrice

refresh counter mémorise le numéro de ligne devant subir un rafraîchissement

mémoire cache

mémoire cache

augmentation des performances

mémoire cache

augmentation des performances

– microprocesseurs : environ 55% par an

mémoire cache

augmentation des performances

– microprocesseurs : environ 55% par an
– mémoire : environ 7% par an

mémoire cache

augmentation des performances

– microprocesseurs : environ 55% par an
– mémoire : environ 7% par an

comment compenser cette différence ?

définition

niveau de mémorisation

définition

niveau de mémorisation

– intermédiaire

définition

niveau de mémorisation

– intermédiaire
– rapide

définition

niveau de mémorisation

– intermédiaire
– rapide
– de petite capacité

définition

niveau de mémorisation

– intermédiaire
– rapide
– de petite capacité
– stockant les données les plus récemment accédées

définition

niveau de mémorisation

– intermédiaire
– rapide
– de petite capacité
– stockant les données les plus récemment accédées

accès moins couteux que l'accès à la mémoire principale

définition

typiquement situé entre

– le processeur et la mémoire principale
– le processeur et un autre cache
– le processeur et un disque
– etc...

découpée en *bloc* : ensemble de mots d'adresses contigües

principe

rechercher dans le cache $avant$ de rechercher dans la mémoire principale

principe

rechercher dans le cache $avant$ de rechercher dans la mémoire principale

– succès cache : la donnée est présente dans le cache
pas d'accès à la mémoire principale
– défaut de cache : la donnée est absente du cache
accès à la mémoire principale

principe

mémoire principale découpée en blocs de même taille

principe

mémoire principale découpée en blocs de même taille

accès à une adresse mémoire

principe

mémoire principale découpée en blocs de même taille

accès à une adresse mémoire

– si le bloc contenant l'adresse est dans le cache

principe

mémoire principale découpée en blocs de même taille

accès à une adresse mémoire

– si le bloc contenant l'adresse est dans le cache
  – la donnée à l'adresse est lue

principe

mémoire principale découpée en blocs de même taille

accès à une adresse mémoire

- si le bloc contenant l'adresse est dans le cache
  - la donnée à l'adresse est lue
- sinon

principe

mémoire principale découpée en blocs de même taille

accès à une adresse mémoire

- si le bloc contenant l'adresse est dans le cache
  - la donnée à l'adresse est lue
- sinon
  - le bloc contenant l'adresse est copié de la mémoire principale dans le cache

principe

mémoire principale découpée en blocs de même taille

accès à une adresse mémoire

- si le bloc contenant l'adresse est dans le cache
  - la donnée à l'adresse est lue
- sinon
  - le bloc contenant l'adresse est copié de la mémoire principale dans le cache
  - la donnée à l'adresse est lue

caractéristiques

- nombre de caches
- localisation
- contenu
- taille
- correspondance
- accès
- remplacement
- politique d'écriture

nombre de caches et localisation

utilisation de plusieurs caches

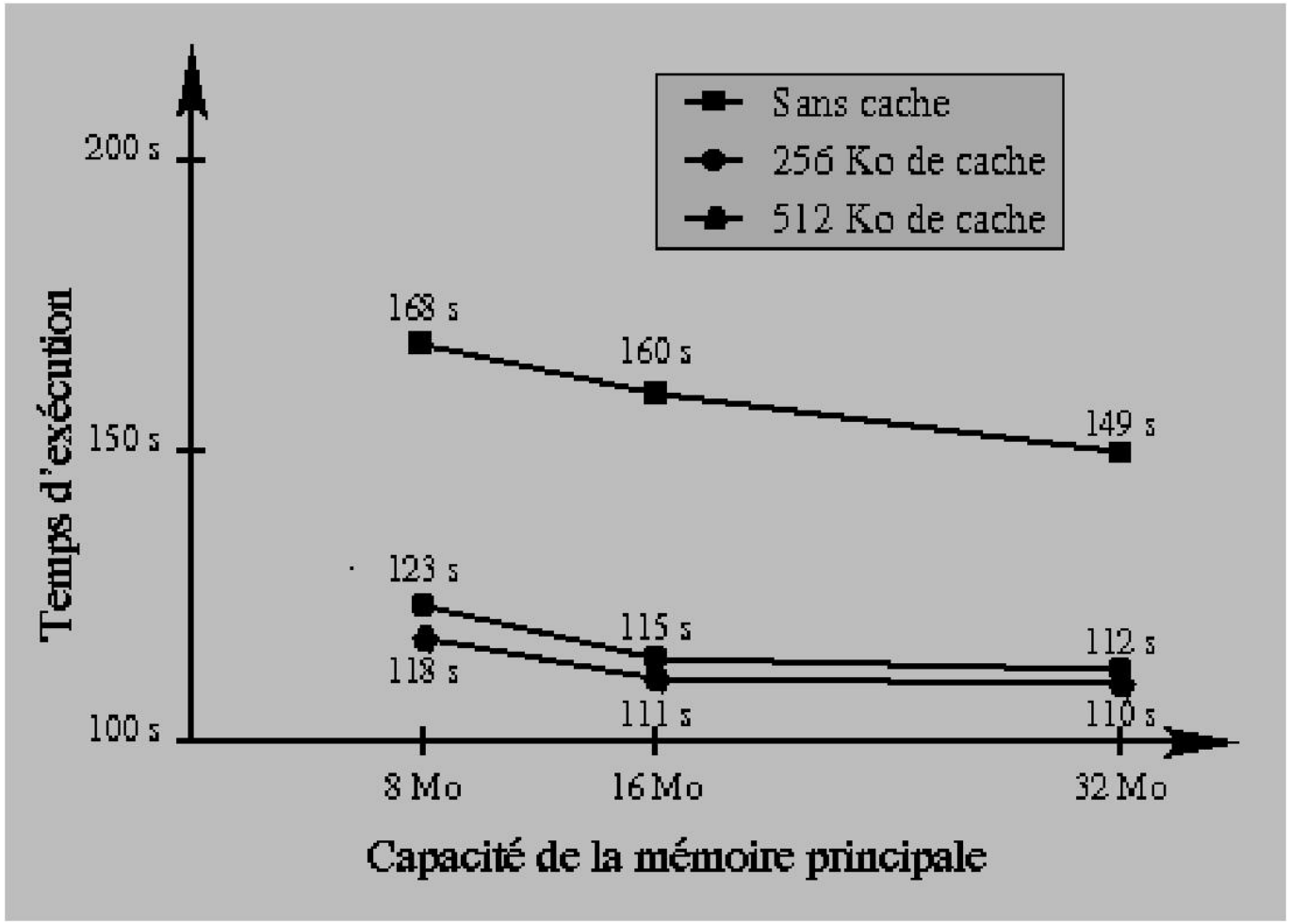
– situé dans le processeur
  (on-chip/on-die/internal)
– accessible via un bus externe au processeur
  (external)

organisés en niveaux (L1, L2, . . .)

nombre de caches et localisation

l'utilisation d'un cache interne permet

– d'augmenter les performances
– de laisser le bus externe disponible


Sans cache
256 Ko de cache
512 Ko de cache
200 s
150 s
100 s
Temps d'exécution
168 s
160 s
149 s
123 s
115 s
112 s
118 s
111 s
110 s
8 Mo
16 Mo
32 Mo
Capacité de la mémoire principale

nombre de caches et localisation

organisation typique

– un ou plusieurs cache de niveau 1 (interne)
– un cache de niveau 2 (interne ou externe)
– parfois un cache de niveau 3 (externe)

nombre de caches et localisation

organisation typique

– un ou plusieurs cache de niveau 1 (interne)
– un cache de niveau 2 (interne ou externe)
– parfois un cache de niveau 3 (externe)

chaque niveau fonctionnant à une vitesse différente

on admet généralement que $|L2| \geq 4 \times |L1|$

contenu

cache interne

– un cache dédié au stockage des instructions
– un cache dédié au stockage des données

contenu

cache interne

– un cache dédié au stockage des instructions
– un cache dédié au stockage des données

le processeur pourra exécuter en parallèle

– la recherche des instructions
– l'exécution des instructions

taille du cache

taille du cache

– suffisament petit

taille du cache

- suffisament petit
  - coût peu élevé

taille du cache

- suffisament petit
  - coût peu élevé
  - temps d'accès le plus intéressant possible

taille du cache

- suffisament petit
  - coût peu élevé
  - temps d'accès le plus intéressant possible
- suffisament grand

taille du cache

- suffisament petit
  - coût peu élevé
  - temps d'accès le plus intéressant possible
- suffisament grand
  - éviter les défauts de cache

taille du cache

- suffisament petit
  - coût peu élevé
  - temps d'accès le plus intéressant possible
- suffisament grand
  - éviter les défauts de cache

les performances dépendent beaucoup de la nature des applications

correspondance

taille du cache $<<$ taille de la mémoire

3 stratégies de copie des blocs mémoire dans le cache

correspondance

taille du cache $<<$ taille de la mémoire

3 stratégies de copie des blocs mémoire dans le cache

– correspondance directe :
  1 emplacement unique dans le cache pour un bloc mémoire

correspondance

taille du cache $<<$ taille de la mémoire

3 stratégies de copie des blocs mémoire dans le cache

– correspondance directe :
  1 emplacement unique dans le cache pour un bloc mémoire
– correspondance totalement associative :
  1 bloc mémoire peut être placé n'importe où dans le cache

correspondance

– correspondance associative par ensemble : 1 bloc mémoire peut être placé dans n'importe quel bloc du cache parmi un ensemble de n blocs

correspondance

– correspondance associative par ensemble : 1 bloc mémoire peut être placé dans n'importe quel bloc du cache parmi un ensemble de n blocs

majorité des caches

– correspondance directe
– correspondance associative par ensemble de 2 ou 4 blocs

accès à un bloc du cache

construction des adresses mémoires en fonction de la correspondance

accès à un bloc du cache

construction des adresses mémoires en fonction de la correspondance

l'adresse mémoire d'un mot permet de trouver

accès à un bloc du cache

construction des adresses mémoires en fonction de la correspondance

l'adresse mémoire d'un mot permet de trouver

– le bloc auquel elle appartient

accès à un bloc du cache

construction des adresses mémoires en fonction de la correspondance

l'adresse mémoire d'un mot permet de trouver

– le bloc auquel elle appartient
– sa place dans le bloc

accès à un bloc du cache

construction des adresses mémoires en fonction de la correspondance

l'adresse mémoire d'un mot permet de trouver

– le bloc auquel elle appartient
– sa place dans le bloc
– la place de ce bloc dans le cache

adresse mémoire d'un mot

décomposée en deux partie

– un numéro de bloc

– un déplacement : l'adresse du mot dans le bloc

adresse mémoire d'un mot

décomposée en deux partie

– un numéro de bloc
  – un index : l'emplacement du bloc dans le cache

– un déplacement : l'adresse du mot dans le bloc

adresse mémoire d'un mot

décomposée en deux partie

- un numéro de bloc
  - un index : l'emplacement du bloc dans le cache
  - une étiquette : identifie le bloc parmi les blocs destinés au même emplacement
- un déplacement : l'adresse du mot dans le bloc

adresse mémoire

le cache maintient une table d'étiquettes

emplacement de bloc 1	étiquette de bloc mémoire 1
emplacement de bloc 2	étiquette de bloc mémoire 2
⋮	⋮
emplacement de bloc n	étiquette de bloc mémoire n

$n =$ taille du cache / taille de bloc

algorithme de remplacement

problème lors

– d'un défaut de cache
– d'une correspondace associative

quel bloc du cache va recevoir le nouveau bloc mémoire?

algorithmes de remplacement

diverses stratégies sont employées

algorithmes de remplacement

diverses stratégies sont employées

– choisir un bloc candidat de manière aléatoire

algorithmes de remplacement

diverses stratégies sont employées

– choisir un bloc candidat de manière aléatoire
– choisir le plus ancien bloc du cache
(FIFO, First In First Out)

algorithmes de remplacement

diverses stratégies sont employées

– choisir un bloc candidat de manière aléatoire
– choisir le plus ancien bloc du cache
  (FIFO, First In First Out)
– choisir le bloc le moins récemment utilisé
  (LRU Least Recently Used)

algorithmes de remplacement

diverses stratégies sont employées

– choisir un bloc candidat de manière aléatoire
– choisir le plus ancien bloc du cache
  (FIFO, First In First Out)
– choisir le bloc le moins récemment utilisé
  (LRU Least Recently Used)
– choisir le bloc le moins fréquemment utilisé
  (LFU Least Frequently Used)

algorithmes de remplacement

les plus efficaces

– LFU
– LRU
– aléatoire

les plus faciles à implanter

– aléatoire
– FIFO

politique d'écriture

problème lors

– d'une opération d'écriture en mémoire
– de la présence dans le cache du mot à écrire

où écrire ?

politique d'écriture

plusieurs méthodes

politique d'écriture

plusieurs méthodes

- écriture simultanée (write through)
  - écrire dans le bloc du cache
  - écrire dans le bloc de la mémoire

politique d'écriture

plusieurs méthodes

- écriture simultanée (write through)
  - écrire dans le bloc du cache
  - écrire dans le bloc de la mémoire
- réécriture (write back)
  - écrire uniquement dans le bloc du cache
  - attendre que l'emplacement soit réquisitionné
  - écrire ce bloc en mémoire

politique d'écriture

et si le bloc n'est pas dans le cache ?

politique d'écriture

et si le bloc n'est pas dans le cache ?

- écriture allouée
  - charger le bloc de la mémoire dans le cache
  - effectuer l'opération d'écriture

politique d'écriture

et si le bloc n'est pas dans le cache ?

- écriture allouée
  - charger le bloc de la mémoire dans le cache
  - effectuer l'opération d'écriture
- écriture non allouée
  - effectuer l'écriture directement dans la mémoire

performance

temps d'accès mémoire moyen =

temps d'accès succès + taux d'échec × pénalité d'échec

performance

temps d'accès mémoire moyen =

temps d'accès succès + taux d'échec × pénalité d'échec

temps d'accès succès =

temps d'accès à une donnée du cache

performance

temps d'accès mémoire moyen =

temps d'accès succès + taux d'échec × pénalité d'échec

temps d'accès succès =

temps d'accès à une donnée du cache

taux d'échec =

nombre de défaut de cache / nombre d'accès cache

exemple

exécution d'une instruction

– décodage de l'instruction
– recherche des données en mémoire
– déclenchement des opérations sur ces données

exemple

durée d'un cycle horloge	: τ
pénalité d'échec	: 10 cycles
durée d'une instruction (sans référence mémoire)	: 2 cycles
nombre de références mémoire par instruction	: 1,33
taux d'échec	: 2%
temps d'accès succés	: négligeable

exemple

temps d'éxécution moyen d'une instruction $=$

$$(2 + 1{,}33 \times 2\% \times 10) \times \tau = 2{,}27\tau$$

exemple

temps d'éxécution moyen d'une instruction $=$

$(2 + 1{,}33 \times 2\% \times 10) \times \tau = 2{,}27\tau$

dans le cas où il n'y a pas de cache?

exemple

temps d'éxécution moyen d'une instruction $=$

$(2 + 1{,}33 \times 2\% \times 10) \times \tau = 2{,}27\tau$

dans le cas où il n'y a pas de cache?

temps d'exécution moyen d'une instruction $=$

$(2 + 1{,}33 \times 10) = 15{,}3\tau$

exemple : le pentium II

- 2 niveaux de cache
  - L1 : cache interne
    - cache de données
    - cache d'instructions
  - L2 : cache externe
- mémoire principale dynamique et synchrone (SDRAM)

caches

cache L1 (données)

– correspondance associative par ensemble
– blocs de 256 bits
– un algorithme LRU pour le remplacement
– une politique d'écriture write back

caches

cache L2 (données et instructions)

– correspondance associative par ensemble
– externe à 50% de la vitesse du processeur
– bloc de 256 bits

SDRAM

évolution synchrone de la DRAM

la DRAM standard est asynchrone

– demande de lecture/écriture
– attente
  temps d'accès du circuit DRAM
– confirmation du succès de l'opération

SDRAM

échange des données avec le processeur en se basant sur un signal d'horloge externe

- informations à traiter stockées dans des registres internes
- répond après un nombre de cycles d'horloge fixé

SDRAM

mode d'accès "en rafale" (burst mode)

– lecture/écriture de données d'adresses contigües
– seule l'adresse de début d'une séquence de mots est donnée
– pas de décodage des adresses suivantes

DDR-SDRAM

évolution de la SDRAM

2 mots de 64 bits par cycle d'horloge

utilisation du front montant et du front déscendant

évolution cache de la famille i386

processeur	cache
80386	0
80486	1 cache interne 8Ko
pentium	1 cache interne donnée 8 Ko
	1 cache interne instructions 8 Ko
pentium 2	1 cache interne donnée 16 Ko
	1 cache interne instructions 16 Ko
	1 cache externe 256 Ko

évolution cache de la famille i386

processeur	cache
pentium 3	1 cache interne donnée 16 Ko
(coppermine)	1 cache interne instructions 16 Ko
	1 cache interne l2 256 Ko
pentium 4	1 cache interne donnée 8 Ko
	1 cache interne de 12 000 micro-opérations
	1 cache interne l2 256 Ko (willamette)
	1 cache interne l2 512 Ko (northwood)

INTEL céléron 2	1 cache interne donnée 16 Ko
	1 cache interne instructions 16 Ko
	1 cache externe 128 Ko
AMD athlon	1 cache interne données 64 Ko
	1 cache interne instructions 64 Ko
	1 cache externe 256 Ko (thoroughbred)
	1 cache externe 512 Ko (barton)
Motorola G4	1 cache interne données 32 Ko
	1 cache interne instructions 32 Ko
	1 cache interne l2 256 Ko
	1 cache externe l3 1 Mo