

Architecture des ordinateurs VI

Frédéric Boulanger

CentraleSupélec



Mémoires



Mémoires

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)

Mémoires

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)
- ▶ dans ou hors du processeur : mémoire statique (≈ 10 ns)

Mémoires

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)
- ▶ dans ou hors du processeur : mémoire statique (≈ 10 ns)
- ▶ hors du processeur : mémoire dynamique (≈ 100 ns)

Mémoires

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)
- ▶ dans ou hors du processeur : mémoire statique (≈ 10 ns)
- ▶ hors du processeur : mémoire dynamique (≈ 100 ns)
- ▶ hors du processeur : mémoire flash ($\approx 10 \mu\text{s}/50$ ns)

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)
- ▶ dans ou hors du processeur : mémoire statique (≈ 10 ns)
- ▶ hors du processeur : mémoire dynamique (≈ 100 ns)
- ▶ hors du processeur : mémoire flash ($\approx 10 \mu\text{s}/50$ ns)
- ▶ hors du processeur : disque SSD ($\approx 100 \mu\text{s}$)

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)
- ▶ dans ou hors du processeur : mémoire statique (≈ 10 ns)
- ▶ hors du processeur : mémoire dynamique (≈ 100 ns)
- ▶ hors du processeur : mémoire flash ($\approx 10 \mu\text{s}/50$ ns)
- ▶ hors du processeur : disque SSD ($\approx 100 \mu\text{s}$)
- ▶ hors du processeur : disque magnétique (≈ 10 ms)

Mémoires

La mémoire permet de stocker des informations :

- ▶ dans le processeur : bascules, registres (< 1 ns)
- ▶ dans ou hors du processeur : mémoire statique (≈ 10 ns)
- ▶ hors du processeur : mémoire dynamique (≈ 100 ns)
- ▶ hors du processeur : mémoire flash ($\approx 10 \mu\text{s}/50$ ns)
- ▶ hors du processeur : disque SSD ($\approx 100 \mu\text{s}$)
- ▶ hors du processeur : disque magnétique (≈ 10 ms)
- ▶ hors du processeur : disque optique (≈ 150 ms)

Mémoires

Non volatile

Le contenu est préservé en l'absence d'alimentation :

- ▶ ROM (Read Only Memory), mémoire morte
Contenu figé à la conception

Mémoires

Non volatile

Le contenu est préservé en l'absence d'alimentation :

- ▶ ROM (Read Only Memory), mémoire morte
Contenu figé à la conception
- ▶ PROM (Programmable ROM)
Contenu programmable une seule fois (fusibles)

Non volatile

Le contenu est préservé en l'absence d'alimentation :

- ▶ ROM (Read Only Memory), mémoire morte
Contenu figé à la conception
- ▶ PROM (Programmable ROM)
Contenu programmable une seule fois (fusibles)
- ▶ EPROM (Erasable Programmable ROM)
Contenu programmable et effaçable (UV)

Non volatile

Le contenu est préservé en l'absence d'alimentation :

- ▶ ROM (Read Only Memory), mémoire morte
Contenu figé à la conception
- ▶ PROM (Programmable ROM)
Contenu programmable une seule fois (fusibles)
- ▶ EPROM (Erasable Programmable ROM)
Contenu programmable et effaçable (UV)
- ▶ EEPROM (Electrically Erasable PROM)
Contenu programmable et effaçable électriquement

Non volatile

Le contenu est préservé en l'absence d'alimentation :

- ▶ ROM (Read Only Memory), mémoire morte
Contenu figé à la conception
- ▶ PROM (Programmable ROM)
Contenu programmable une seule fois (fusibles)
- ▶ EPROM (Erasable Programmable ROM)
Contenu programmable et effaçable (UV)
- ▶ EEPROM (Electrically Erasable PROM)
Contenu programmable et effaçable électriquement
- ▶ mémoire flash
Effaçable électriquement par page (grille flottante)

Grille flottante



Jonction PN

Grille flottante



Jonction PN

Grille flottante

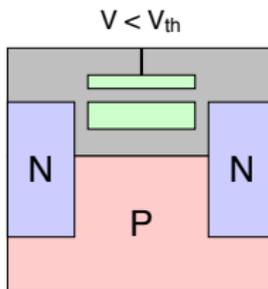


Jonction PN

Grille flottante



Jonction PN

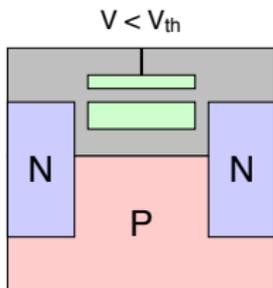


Transistor (1) bloqué

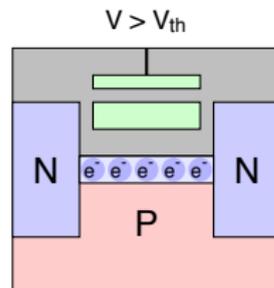
Grille flottante



Jonction PN



Transistor (1) bloqué



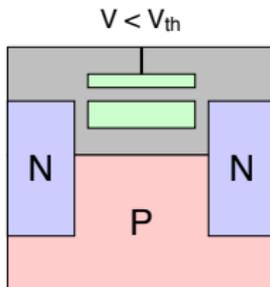
Transistor (1) passant



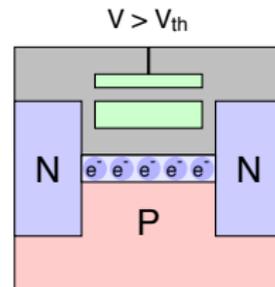
Grille flottante



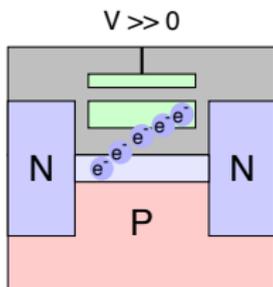
Jonction PN



Transistor (1) bloqué



Transistor (1) passant

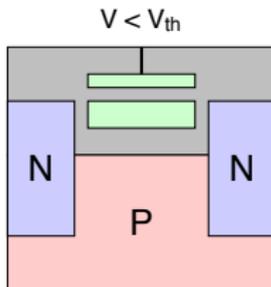


Injection d' e^-

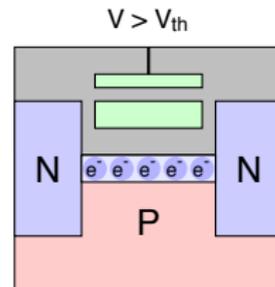
Grille flottante



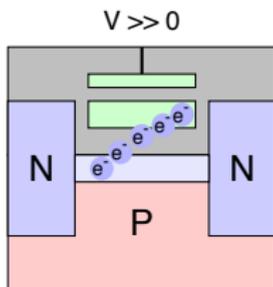
Jonction PN



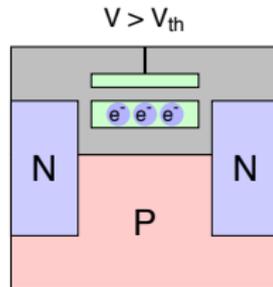
Transistor (1) bloqué



Transistor (1) passant



Injection d'e⁻

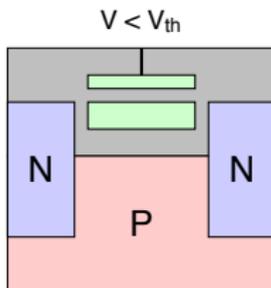


Transistor (0) bloqué

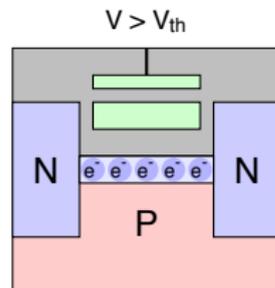
Grille flottante



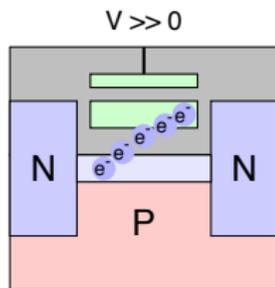
Jonction PN



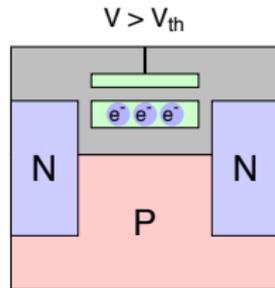
Transistor (1) bloqué



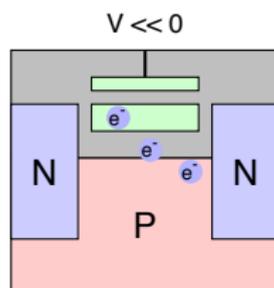
Transistor (1) passant



Injection d'e-



Transistor (0) bloqué



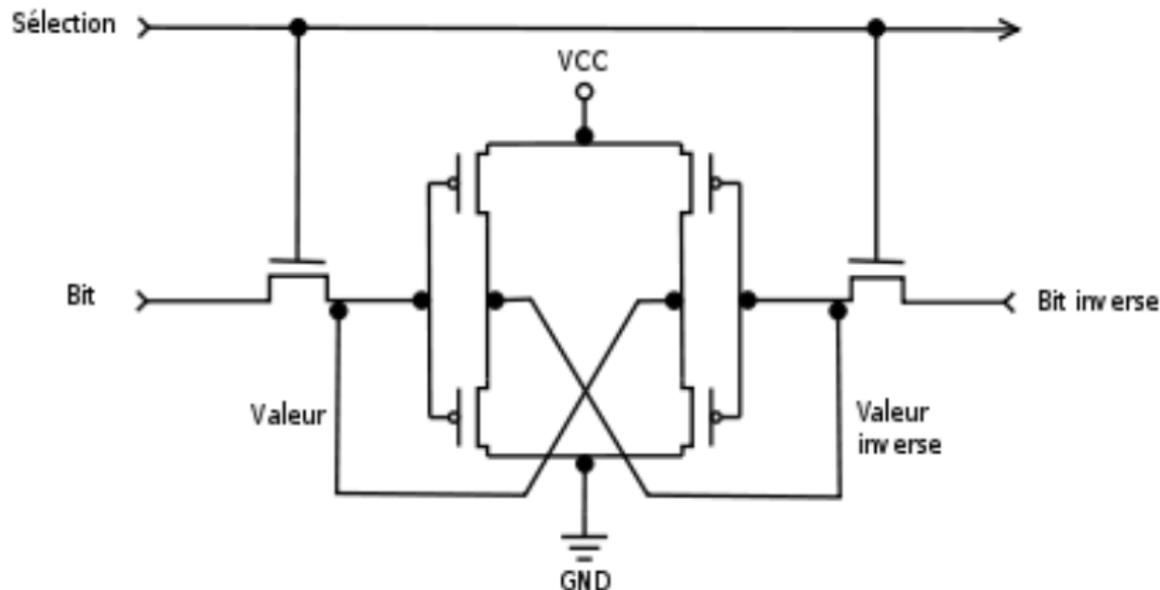
Extraction d'e-

Mémoires

Volatile

Le contenu est perdu en l'absence d'alimentation :

- ▶ SRAM (Static Random Access Memory)
Contenu stable (bascules RS, 6 transistors), rapide

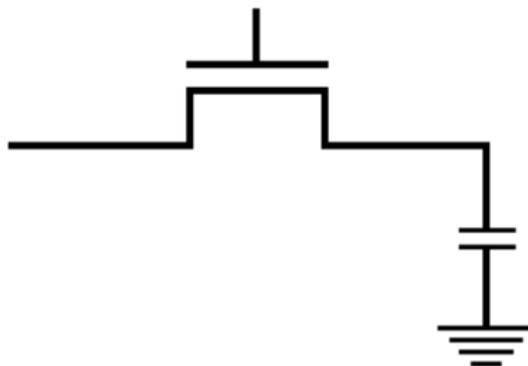


Mémoires

Volatile

Le contenu est perdu en l'absence d'alimentation :

- ▶ SRAM (Static Random Access Memory)
Contenu stable (bascules RS, 6 transistors), rapide
- ▶ DRAM (Dynamic RAM)
Contenu évanescent (capacité), lecture destructrice



Mémoires

Volatile

Le contenu est perdu en l'absence d'alimentation :

- ▶ SRAM (Static Random Access Memory)
Contenu stable (bascules RS, 6 transistors), rapide
- ▶ DRAM (Dynamic RAM)
Contenu évanescent (capacité), lecture destructrice
- ▶ SDRAM (Synchronous DRAM)
Mémoire dynamique fonctionnant de manière synchrone

Mémoires

Volatile

Le contenu est perdu en l'absence d'alimentation :

- ▶ SRAM (Static Random Access Memory)
Contenu stable (bascules RS, 6 transistors), rapide
- ▶ DRAM (Dynamic RAM)
Contenu évanescent (capacité), lecture destructrice
- ▶ SDRAM (Synchronous DRAM)
Mémoire dynamique fonctionnant de manière synchrone
- ▶ DDR SDRAM (Double Data Rate SDRAM)
SDRAM permettant deux accès par cycle d'horloge (un par front)

Mémoires

Structure d'un boîtier

Les cellules de mémoire sont organisées en matrice

- ▶ une partie de l'adresse sélectionne une ligne, l'autre une colonne

Mémoires

Structure d'un boîtier

Les cellules de mémoire sont organisées en matrice

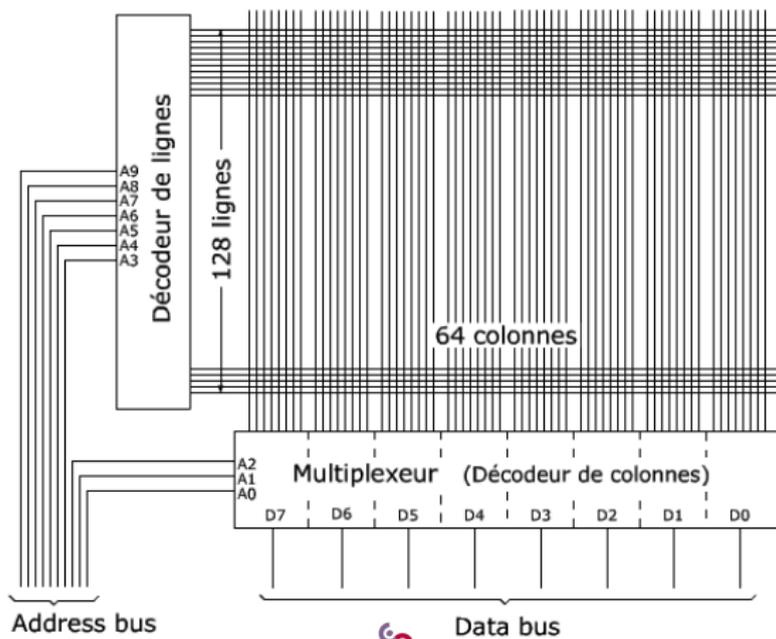
- ▶ une partie de l'adresse sélectionne une ligne, l'autre une colonne
- ▶ pour la mémoire dynamique, on rafraîchit colonne par colonne

Mémoires

Structure d'un boîtier

Les cellules de mémoire sont organisées en matrice

- ▶ une partie de l'adresse sélectionne une ligne, l'autre une colonne
- ▶ pour la mémoire dynamique, on rafraîchit colonne par colonne



Mémoires

Structure d'un boîtier

Les cellules de mémoire sont organisées en matrice

- ▶ une partie de l'adresse sélectionne une ligne, l'autre une colonne
- ▶ pour la mémoire dynamique, on rafraîchit colonne par colonne

Modules

Les boîtiers mémoire sont organisés en modules

Mémoires

Structure d'un boîtier

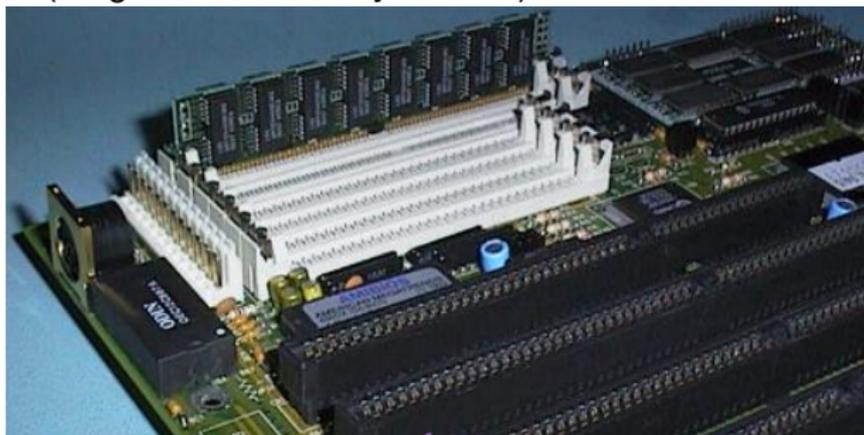
Les cellules de mémoire sont organisées en matrice

- ▶ une partie de l'adresse sélectionne une ligne, l'autre une colonne
- ▶ pour la mémoire dynamique, on rafraîchit colonne par colonne

Modules

Les boîtiers mémoire sont organisés en modules

- ▶ SIMM (Single Inline Memory Module)



Mémoires

Structure d'un boîtier

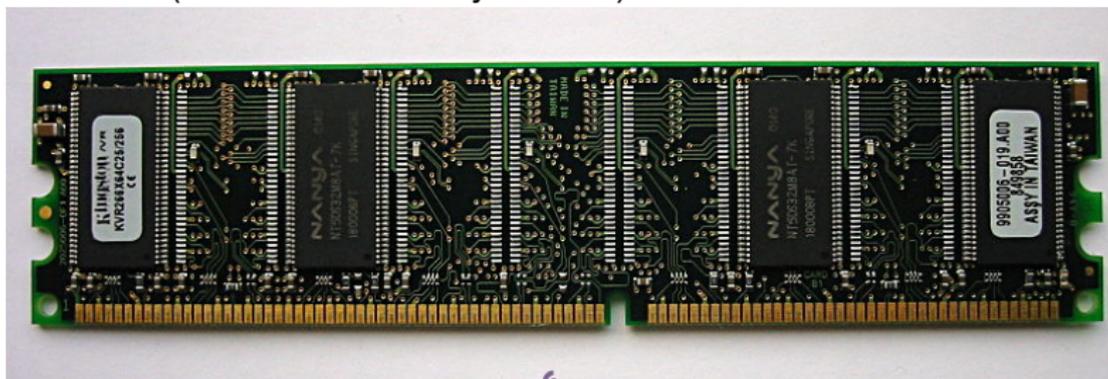
Les cellules de mémoire sont organisées en matrice

- ▶ une partie de l'adresse sélectionne une ligne, l'autre une colonne
- ▶ pour la mémoire dynamique, on rafraîchit colonne par colonne

Modules

Les boîtiers mémoire sont organisés en modules

- ▶ SIMM (Single Inline Memory Module)
- ▶ DIMM (Dual Inline Memory Module)



Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go



Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD
- ▶ plusieurs centaines de Go à quelques To



Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD
- ▶ plusieurs centaines de Go à quelques To

Archivage

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD
- ▶ plusieurs centaines de Go à quelques To

Archivage

- ▶ stockage de très longue durée

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD
- ▶ plusieurs centaines de Go à quelques To

Archivage

- ▶ stockage de très longue durée
- ▶ bandes magnétiques

Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD
- ▶ plusieurs centaines de Go à quelques To

Archivage

- ▶ stockage de très longue durée
- ▶ bandes magnétiques
- ▶ plusieurs centaines de Go à quelques To



Mémoires

Mémoire centrale

- ▶ mémoire située près du processeur
- ▶ RAM, mémoire flash
- ▶ de quelques Mo à quelques Go

Mémoire de masse

- ▶ stockage de longue durée
- ▶ disque durs magnétiques, SSD
- ▶ plusieurs centaines de Go à quelques To

Archivage

- ▶ stockage de très longue durée
- ▶ bandes magnétiques
- ▶ plusieurs centaines de Go à quelques To
- ▶ lent, accès séquentiel, faible coût



Hierarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)

Hiéarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)
- ▶ grande capacité = lent (DRAM, disque)



Hiéarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)
- ▶ grande capacité = lent (DRAM, disque)

Solution



Hierarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)
- ▶ grande capacité = lent (DRAM, disque)

Solution

- ▶ combiner une petite mémoire rapide



Hiéarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)
- ▶ grande capacité = lent (DRAM, disque)

Solution

- ▶ combiner une petite mémoire rapide
- ▶ ... avec une grande mémoire lente



Hiéarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)
- ▶ grande capacité = lent (DRAM, disque)

Solution

- ▶ combiner une petite mémoire rapide
- ▶ ... avec une grande mémoire lente
- ▶ et faire en sorte que les données voulues soient souvent dans la mémoire rapide.



Hiéarchies de mémoires

Principes



Hierarchies de mémoires

Principes

- ▶ **localité temporelle**

quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt

Hierarchies de mémoires

Principes

- ▶ **localité temporelle**

quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt

- ▶ boucles

Hiérarchies de mémoires

Principes

- ▶ **localité temporelle**

quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt

- ▶ boucles
- ▶ usage répété de variables

Hierarchies de mémoires

Principes

- ▶ **localité temporelle**

quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt

- ▶ boucles
- ▶ usage répété de variables

- ▶ **localité spatiale**

quand on accède à une donnée,
il y a des chances qu'on accède ensuite à des données voisines



Hierarchies de mémoires

Principes

▶ localité temporelle

quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt

- ▶ boucles
- ▶ usage répété de variables

▶ localité spatiale

quand on accède à une donnée,
il y a des chances qu'on accède ensuite à des données voisines

- ▶ séquentialité du code, boucles



Hiéarchies de mémoires

Principes

▶ localité temporelle

quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt

- ▶ boucles
- ▶ usage répété de variables

▶ localité spatiale

quand on accède à une donnée,
il y a des chances qu'on accède ensuite à des données voisines

- ▶ séquentialité du code, boucles
- ▶ tableaux, codage contigu de l'information



Hierarchies de mémoires

Application

Hierarchies de mémoires

Application

- ▶ localité temporelle

Hierarchies de mémoires

Application

- ▶ **localité temporelle**
 - ▶ on conserve dans une mémoire rapide les données que l'on consulte



Hierarchies de mémoires

Application

- ▶ **localité temporelle**
 - ▶ on conserve dans une mémoire rapide les données que l'on consulte
 - ▶ quand la mémoire rapide est pleine, on élimine les données les plus anciennes



Hiéarchies de mémoires

Application

- ▶ **localité temporelle**
 - ▶ on conserve dans une mémoire rapide les données que l'on consulte
 - ▶ quand la mémoire rapide est pleine, on élimine les données les plus anciennes
- ▶ **localité spatiale**



Hiéarchies de mémoires

Application

- ▶ **localité temporelle**
 - ▶ on conserve dans une mémoire rapide les données que l'on consulte
 - ▶ quand la mémoire rapide est pleine, on élimine les données les plus anciennes
- ▶ **localité spatiale**
 - ▶ on charge dans la mémoire rapide les données voisines de celle que l'on consulte



Hierarchies de mémoires

Application

- ▶ **localité temporelle**
 - ▶ on conserve dans une mémoire rapide les données que l'on consulte
 - ▶ quand la mémoire rapide est pleine, on élimine les données les plus anciennes
- ▶ **localité spatiale**
 - ▶ on charge dans la mémoire rapide les données voisines de celle que l'on consulte
- ▶ on mémorise d'où viennent les données présentes dans la mémoire rapide (descripteur)



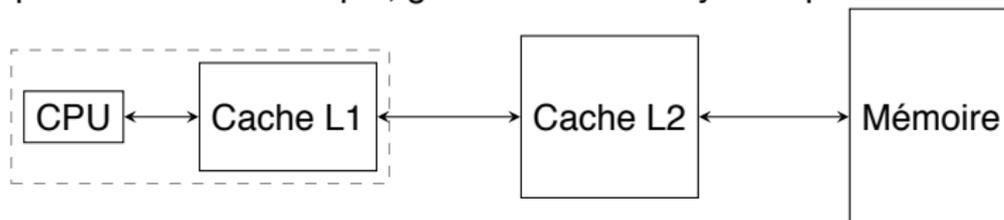
Hiérarchies de mémoires

Cache

Hiérarchies de mémoires

Cache

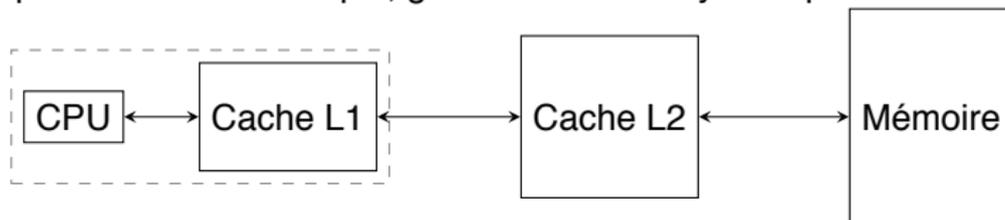
- ▶ petite mémoire statique, grande mémoire dynamique



Hiéarchies de mémoires

Cache

- ▶ petite mémoire statique, grande mémoire dynamique

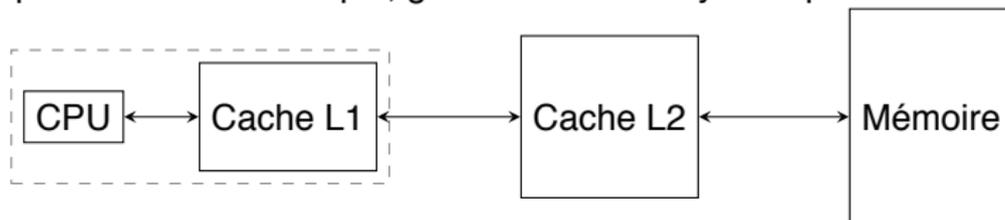


- ▶ caches L1 séparés pour instructions et données

Hiéarchies de mémoires

Cache

- ▶ petite mémoire statique, grande mémoire dynamique



- ▶ caches L1 séparés pour instructions et données

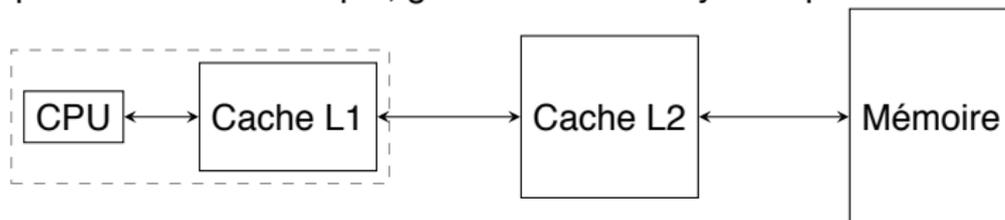
Cache de disque

mémoire dynamique, disque dur : but = accélérer les accès au disque

Hiéarchies de mémoires

Cache

- ▶ petite mémoire statique, grande mémoire dynamique



- ▶ caches L1 séparés pour instructions et données

Cache de disque

mémoire dynamique, disque dur : but = accélérer les accès au disque

Mémoire virtuelle

mémoire dynamique, disque dur : but = augmenter la capacité mémoire

Suite...

Entrées-sorties

