

# Calcul scientifique et programmation: systèmes, outils, architecture

## Introduction à l'architecture des ordinateurs

Alexei Stoukov

*Alexei.Stoukov@enseeiht.fr ou Alexei.Stoukov@imft.fr*

**Christophe Airiau<sup>1</sup>**

Septembre 2017

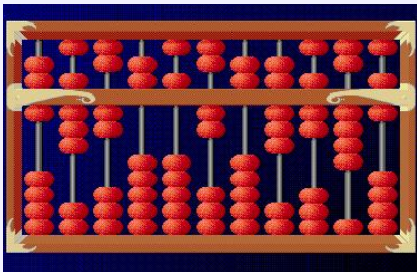


# Sommaire

- 1 Histoire
- 2 Lois
- 3 Calcul scientifique
- 4 Éléments de base
- 5 Architectures

## Constat simple

- L'homme n'est pas une « bête de calcul »
- Trop lent
- Pas assez fiable
- Il va chercher à améliorer sa façon de calculer

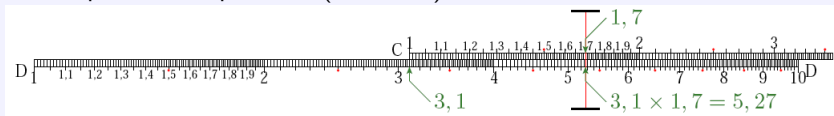


- 1300 av. J.C. :  
Invention de l'**abaque** ou  
**boulard** compteur

- 1620 : Invention de la **règle à calculer** (algorithmes de John Napier)



- Principe :  $\log(a) + \log(b) = \log(a \times b)$
- Exemple de multiplication ( $3.1 \times 1.7$ ) :



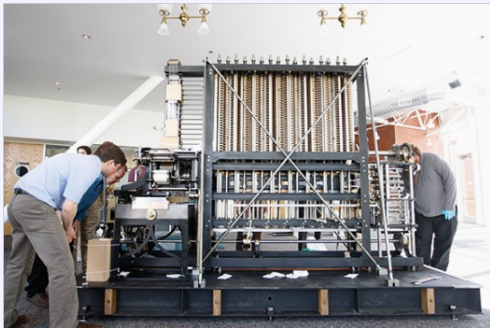
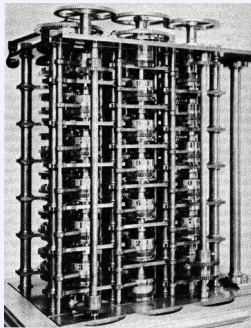
Plus de détails : [http://tnerual.eriogerg.free.fr/utilisation\\_regle\\_calcul.pdf](http://tnerual.eriogerg.free.fr/utilisation_regle_calcul.pdf)

- 1642 : Invention de la **machine à calculer** par Blaise Pascal



- "Pascaline"
  - Addition
  - Soustraction
- En 1673 Gottfried Wilhelm von Leibniz
  - Multiplication
  - Addition

- 1820 : Apparition des **calculateurs mécaniques à 4 fonctions** (+, −, ×, /)
- 1834 : Invention de la **difference engine** par Charles Babbage
  - évaluation de fonctions au moyen de différences finies
  - cette machine n'a jamais été achevée de son vivant



Plus de détails : [http://en.wikipedia.org/wiki/Difference\\_engine](http://en.wikipedia.org/wiki/Difference_engine)

- Babbage définit les principaux concepts des machines informatiques :
  - un dispositif d'entrée et de sortie
  - un organe de commande gérant le transfert des nombres et leur mise en ordre pour le traitement
  - un magasin permettant de stocker les résultats intermédiaires ou finaux
  - un moulin chargé d'exécuter les opérations sur les nombres ;
  - un dispositif d'impression.
- Les analogues en termes contemporains :
  - un clavier et un moniteur
  - une unité de commande (composant d'un microprocesseur, mais au départ organe séparé)
  - un outil de stockage (mémoire vive, disque dur, supports amovibles)
  - une unité de calcul (intégrée aujourd'hui dans les microprocesseurs, mais au départ distincte de l'unité de commande)
  - une imprimante

- 1885 : Invention du **Comptometer**, calculateur mécanique à 4 fonctions avec clavier et imprimante intégrés
- 1886 : 1er succès commercial d'une machine à additionner
- 189x : Apparition des **calculatrices électromécaniques** où un moteur électrique remplace la manivelle



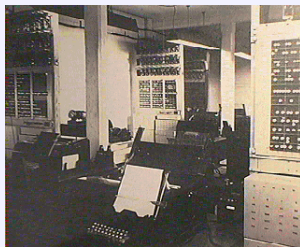


- 1890 : Invention d'une **trieuse électromécanique à cartes perforées** (pour palier au recensement désastreux de 1880) par Hollerith qui va ainsi créer une compagnie qui va devenir IBM (Etats-Unis)



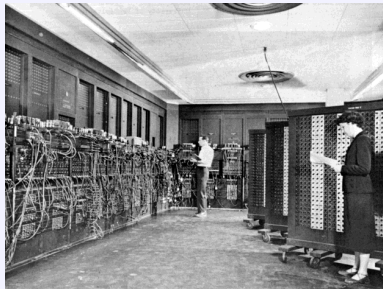
La machine de Hollerith (Archives d'IBM)

- 1938 : Construction d'un **ordinateur programmable à relais électromécaniques** par Konrad Zuse utilisant le **système binaire** et non le décimal (Allemagne)
- 1938 : Mise au point du **Colossus**, ordinateur de décryptage des messages secrets allemands (Angleterre)



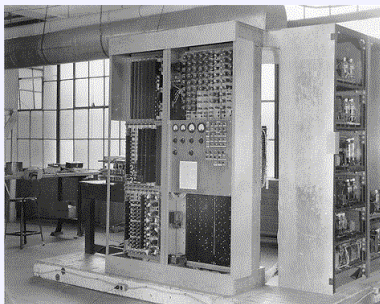
Colossus

- 1942 : Réalisation de l'**Eniac**, 1<sup>er</sup> ordinateur non mécanique (Etats-Unis)



- 30 tonnes, 1500  $m^2$ , 18000 tubes à vides, consommation 160 KW
- 20 nombres à dix chiffres signés permettant chacun de réaliser 5000 additions simples chaque seconde (pour un total de 100000 additions par seconde)
- calculs en décimal, programmation manuelle par câbles, résultats sur cartes perforées
- tables balistiques, bombe H
- fonctionne jusqu'au 2 octobre 1955

- 1947 : Invention du **transistor** par le laboratoire Bell Labs
- 1952 : **EDVAC** (Electronic Discrete Variable Automatic Computer) 1<sup>er</sup> ordinateur dont le programme est stocké en mémoire et non câblé (John von Neumann)
  - mémoire centrale de 1024 mots
  - 1<sup>er</sup> bogue : une mite (bug) logée dans le culot d'une lampe provoque un faux contact



- 1953 : IBM commercialise son premier ordinateur électronique à **programme stocké en mémoire** : le **701**
- 1958 : Texas Instruments invente le **circuit imprimé**

- 1960 : IBM sort le **modèle 7000**, premier ordinateur complètement **transistorisé**
- 1964 : Sortie du **modèle 360 d'IBM**, premier **mini ordinateur**
- Invention de la structure de **bus** avec le **DEC PDP-8**
- 1964 : Commercialisation de la première **calculatrice de poche** : **TI-2500 DataMath**



DEC PDP 8

- 1971 : Naissance du premier micro-ordinateur **Kenback 1**
  - mémoire 256 octets et 3 registres de programmation
  - 62 modèles vendus à 750\$



- 1972 : Réalisation du 1er **micro-processeur Intel 4004**

- 1975 : Sortie de l'**Altair 8800** de MITS, premier **micro-ordinateur personnel**
  - Micro-processeur Intel 8080 à 2 Mhz
  - 2000 ventes la 1ère année à 600\$
  - 1ère machine utilisée par Bill Gates ...



- 1976 : Commercialisation du **Apple I**, premier micro-ordinateur avec un clavier et un moniteur (Steve Wozniak et Steve Jobs)
  - processeur MOS Technology 6502 8 bits 1 MHz
  - bus système 1 MHz
  - mémoire vive 8 Ko, extensible à 64 Ko
  - mémoire morte 256 octets
  - Affichage 40x24 caractères
- 1981 : Naissance du premier **PC d'IBM**
  - processeur Intel 8088 à 4.77 MHz
  - mémoire vive 16 Ko extensible à 256 Ko
  - jusqu'au deux lecteurs de disquettes de 160 ko
  - interprète du langage BASIC Microsoft en mémoire morte



- 1984 : Sortie du **Macintosh 128**, première utilisation commerciale d'une **interface graphique** et d'une **souris**
  - micro-processeur Motorola MC68000 à 8 Mhz
  - mémoire vive 128 Ko sur la carte mère, extensible à 512 Ko
  - mémoire morte 64 Ko
  - lecteur de disquette 400 Ko 3,5"
  - sortie son mono
  - écran intégré 9" monochrome (512 x 342 pixels)



# Et aujourd'hui (TOP500 du mois de juin 2016)

## Sunway TaihuLight (National Supercomputing Center in Wuxi, Chine)

- 40960 super-noeuds des calcul, 32 Gio RAM par noeud
- 1 super-noeud = 4 processeurs SW26010 65 cores à 1.45GHz = 260 cores
- 40 armoires
- 10 649 600 cores au totale
- Mémoire RAM 1.4 Pio ( $\approx 1.4 \times 10^{15}$  octets  $\approx 1\,000\,000$  Gio)
- 93 PFlops :  $93 \times 10^{15}$  opérations par seconde avec des nombres à virgule flottante)

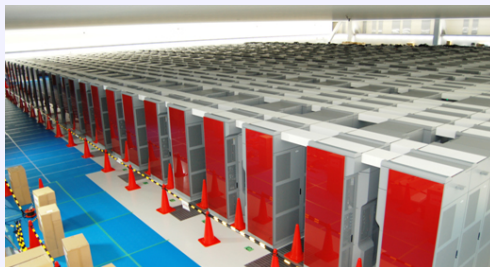




# TOP500 du mois de juin 2011



- RIKEN et Fujitsu  
**K computer**  
(Rikagaku Kenkyusho - Institut de recherche scientifique, Japon)
- **K** comme mot japonais "Kei" -  $10^{16}$
- 864 armoires (racks)
- 88 128 2.0GHz 8-core SPARC64 VIIIfx processeurs
- 706 624 coeurs au totale



# K computer

- 1 armoire contient 96 noeuds de calcul plus 6 noeuds d'entrée/sortie
- 1 noeud = 1 processeur SPARC64 VIIIfx 8-core cadencé à 2.0GHz, 16GB de mémoire vive
- 1 PB ( $\approx 10^{15}$  Octets) de mémoire vive au total
- $10.5 \times 10^{15}$  opérations par seconde avec des nombres à virgule flottante
- consommation électrique 12.6 MW (processeurs, refroidissement à eau)
- Coût de maintenance environ \$10 millions/an.



# K computer

## How fast is the K computer?

If the world's **7 billion people** could perform **one computation per second**, it would take about **17 days** to complete **10,000,000,000,000,000** (one Kei) computations.



The K computer can complete **one Kei** of computations in **just one second**.

Source Fujitsu

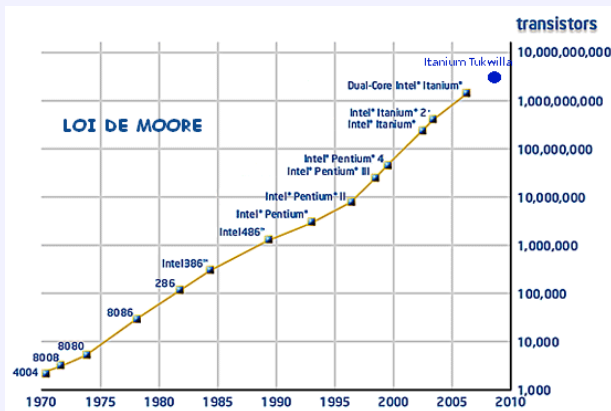
# Bilan

L'ordinateur est toujours :

- plus fiable
- plus petit
- plus rapide
- plus efficace
- moins cher

# Loi de Moore

- Le nombre de transistors que l'on peut placer sur une puce est multiplié par 4 tous les 3 ans



# Autres lois associées

## Loi relative aux vitesses de transmission

- Le débit maximal d'information disponible sur Internet suit une courbe exponentielle dont la pente est encore plus forte que celle de la loi de Moore

## Loi de Parkinson

- La quantité de données, comme les gaz parfaits, tend à remplir la capacité disponible de mémoire
- Avoir plus de mémoire, encourage les applications plus gourmandes en mémoire
- Le besoin en mémoire des systèmes sur le marché double tous les dix-huit mois et ceci depuis 1985
- Heureusement, la loi de Moore est là pour suivre avec la puissance de calcul

# Autres lois associées

## Loi de Machrone

- Quelle que soit la manière de s'y prendre, la machine que vous voulez coûte toujours 5000 \$

## Loi de Rock

- L'investissement nécessaire pour fabriquer des semi-conducteurs double tous les quatre ans

## Loi de Gates

- La vitesse d'exécution des logiciels est divisée par deux tous les dix-huit mois
- Heureusement, la loi de Moore compense
- Loi baptisée en référence au ex-patron de Microsoft (Bill Gates) et en clin d'oeil aux performances des logiciels édités par cette société

# Le calcul scientifique et les ordinateurs

- 1950 : Calcul de trajectoires d'engins et de satellites
- 1960 : Calcul d'explosions et calculs de structures
- 1970 : Calcul d'écoulements des fluides incompressibles visqueux et compressibles non visqueux
- 1980 : Calcul de vibrations sismiques, météo, fluides 3D industriels (stationnaires, turbulents)
- 1990 : Calcul de systèmes couplés (fluides-structures, océan-atmosphère, chimie complexe, ..)
- 2000 : Calcul des pollutions automobiles et industrielles, modélisation et simulation du climat, repliement des protéines, simulation des catastrophes naturelles (éruption volcanique, tremblement de terre)
- 2010 : Traitement des données délivrées par le Large Hadron Collider (Worldwide LHC Computing Grid), estimation de l'ampleur des répliques d'un séisme juste après la première secousse, études du virus de la grippe A (H1N1), Computational Neuroscience ( $1\text{mm}^3$  actuellement).



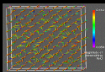
# Technical and Scientific Applications

## Sustained Performance Requirements

### Materials Science

#### ■ Magnetic materials:

- ▶ Current: ~2000 atoms  
2.64 TF/s, 512 GB
- ▶ Future: hard drive simulation: **30 TF/s, 2 TB**



#### ■ Electronic structures:

- ▶ Current: ~300 atom: 0.5 TF/s, 100 GB
- ▶ Future: ~3000 atom: **50 TF/s, 2 TB**

#### ■ Molecular dynamics:

- ▶ Current:  $10^9$  atoms, nsec scale: 1 TF/s, 50GB
- ▶ Future: alloys, usec scale: **20 TF/s, 4 TB**

### Fusion Modeling

#### ■ All-Orders Spectral Algorithm – RF effects in plasmas.

- ▶ Current: 300,000 x 300,000 system
- ▶ Total Computation Requirement: 28 PFLOPS, 230 GB
- ▶ Sustained Performance (8 hours): **1 TF/s; 230 GB**
- ▶ Future: 6,000,000 x 6,000,000 system
- ▶ Total Computational Requirement:  $5.7 \times 10^{10}$  FLOPS (160 hours and 1 PF/s)

Source : D. Bailey  
NERSC

### Climate Modeling

#### ■ Current state-of-the-art:

- ▶ Atmosphere: 1 x 1.25 deg spacing, 29 layers.
- ▶ Ocean: 0.25 x 0.25 deg spacing, 60 layers.
- ▶ 52 seconds CPU time per simulated day.

#### ■ Future: Ocean mesoscale eddy simulation

- ▶ Atmosphere: 0.5 x 0.5 deg spacing.
- ▶ Ocean: 0.125 x 0.125 deg spacing.
- ▶ Sustained Performance: **17 TF/s.**



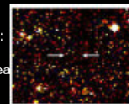
#### ■ Future: Tropical cumulus cloud simulation

- ▶ 2 to 3 orders of magnitude more than above

### Astrophysics

#### ■ Type Ia Supernova simulation:

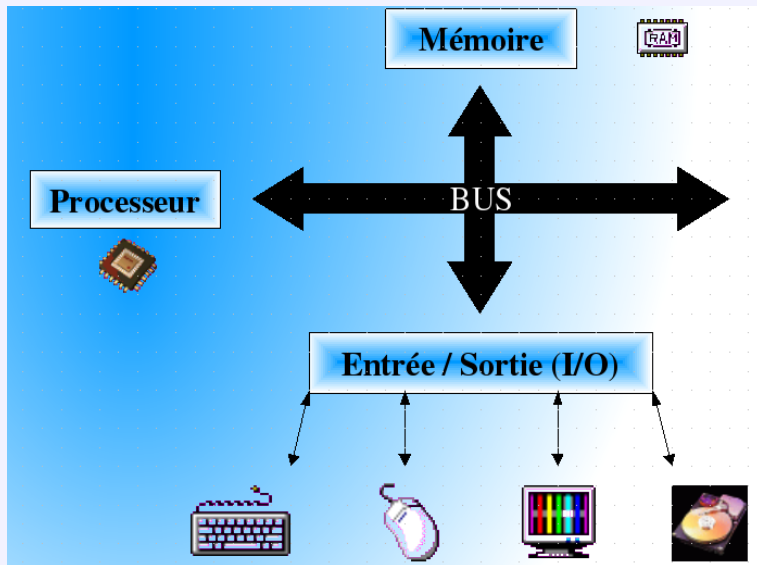
- ▶ 3-D models will require 2,000,000 CPU-hours per year and greater than 256 GB



#### ■ Analysis of cosmic microwave background data:

- ▶ Future BOOMERANG:
  - Total Computational Requirement:  $1.0 \times 10^{10}$  FLOPS
- ▶ Future PLANCK data :
  - $1.0 \times 10^{10}$  flops

# L'ordinateur : Vue globale



# Principaux composants

## Processeur (microprocesseur)

Manipule les données selon les instructions préprogrammées

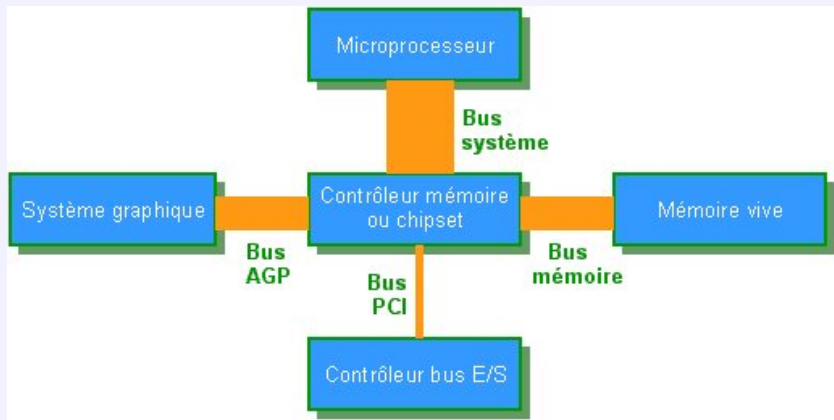
## Mémoire vive

Stocke des données relatives à l'exécution des processus (programmes) et au contrôle des instructions

## Bus : Ensemble de connexions

- Le bus d'adresse permet au processeur de communiquer avec le périphérique via son adresse
- Le bus de données véhicule les données et les instructions en provenance ou à destination du processeur
- Le bus de contrôle (bus de commandes) les transferts de données entre le processeur et les périphériques (mémoire, entrées / sorties)

# Principaux composants



Source : [http ://informatech.online.fr/articles/bus/index2.php](http://informatech.online.fr/articles/bus/index2.php)

# Unités et mesures

- Un **bit** (**binary unit**) : la plus petite quantité d'information représentable dans un ordinateur
  - 0 ou 1 dans un contexte numérique
  - *faux* ou *vrai* dans un contexte logique
- Un octet (byte) = 8 bits. Les multiples :
  - 1 kilo-octet = 1024 octets =  $2^{10}$  octets
  - 1 méga-octet = 1024 kilo-octets =  $2^{20}$  octets
- Attention, la "nouvelle norme" de la Commission électrotechnique internationale (l'Amendement 2 de la norme CEI 60027-2 de 1999) définit :
  - 1 kibi-octet (Kio) = 1024 octets =  $2^{10}$  octets
  - 1 mébi-octet = 1024 kibi-octets =  $2^{20}$  octets
  - par contre, 1 kilo-octet = 1000 octets etc...

# Unités et mesures

- Débit binaire :
  - le nombre de bits transmis en 1 seconde, en *bit/s*
  - 1 *mégabit/s* = 1000000 *bits* = 125000 *octets*  $\approx 122.07$  *kilooctets* (125000/1024)
- Vitesse de transmission :
  - le nombre d'octets (ou de kilo-octets) transmis en une seconde, en prenant en compte tous les facteurs (codage, réseau partagé etc...)

# Processeur (microprocesseur, CPU)

La performance d'un CPU peut être évaluée par le nombre d'instructions qu'il exécute à chaque seconde,  $i/s$ . Cette donnée peut se décomposer comme suit :

$$i/s = i/c \times c/s$$

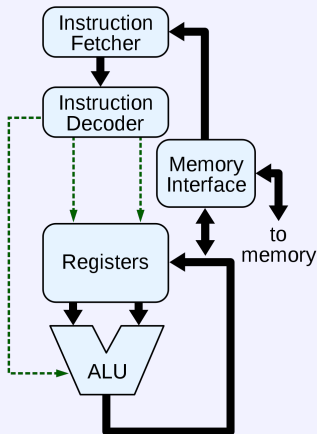
où  $c$  correspond au nombre de cycles processeur,  $i/c$  correspond au nombre moyen d'instructions exécutées à chaque cycle (IPC) et  $c/s$  est le nombre de cycles par seconde, soit la fréquence d'horloge  $F$ . Ainsi :

$$i/s = IPC \times F$$

Cette formule simple nous montre que l'IPC et la fréquence sont les deux principaux facteurs de performance d'un CPU

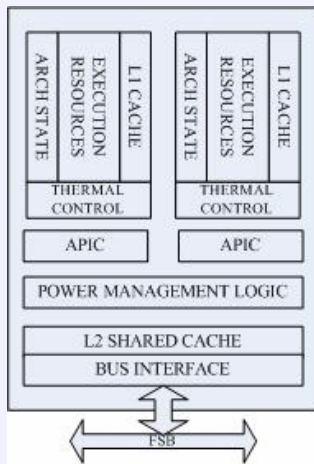
- La fréquence d'horloge ne représente pas les performances réelles d'un CPU

# Processeur (CPU)



*Schéma fonctionnel d'un processeur hypothétique.*

Source : Wikipedia

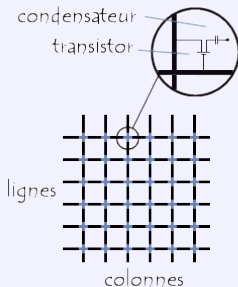


*Schéma fonctionnel d'un processeur de type Core2.*

Source : Intel



# Mémoire vive (RAM - Random Access Memory) : fonctionnement



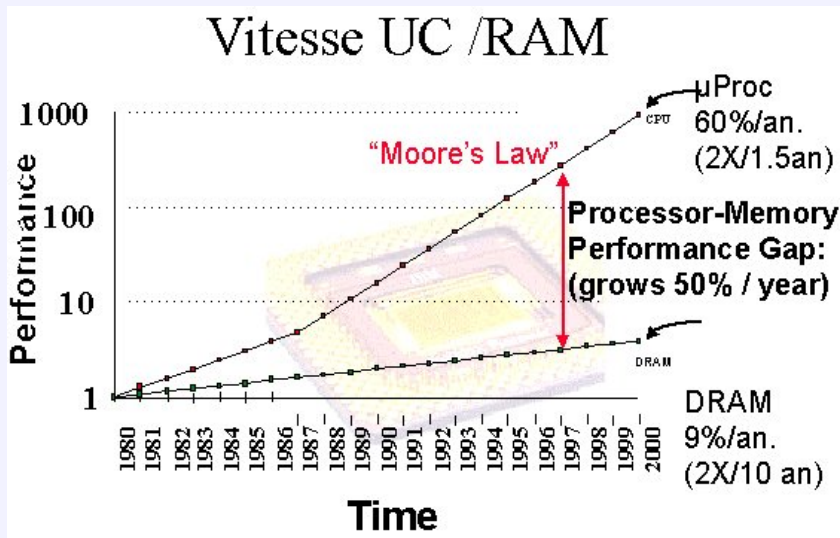
- Centaines de milliers de petits condensateurs emmagasinant des charges.
  - Condensateur chargé - état logique = 1, 0 sinon : 1 *bit* de la mémoire
  - Nécessité de rafraîchissement (cycle rafraîchissement = inverse de la fréquence de la mémoire)
- 
- Chaque condensateur est couplé à un transistor permettant de "récupérer" ou de modifier l'état du condensateur. Ces transistors sont rangés sous forme de tableau (matrice), c'est-à-dire que l'on accède à une case mémoire par une ligne et une colonne.

# Mémoire vive : caractéristiques

- Capacité : nombre d'octet qu'on peut stocker (Go)
- Type
  - Mémoire dynamique (DRAM, Dynamic Random Access Module), peu coûteuse. Utilisée pour la mémoire centrale de l'ordinateur
  - Mémoires statique (SRAM, Static Random Access Module), rapide et onéreuse. Utilisées pour les mémoires cache du processeur
- Temps d'accès à une donnée (case mémoire) dépend :
  - de temps du cycle de rafraîchissement (inverse de la fréquence de rafraîchissement)
  - du temps de latence
- Débit
  - P.ex. pour la mémoire de type DDR2-800 (Double Data Rate two Synchronous Dynamic Random Access Memory) cadencée à 400 MHz

$$400MHz \times 2 \text{ (double data)} \times 64bits/8 = 6400Mo/s$$

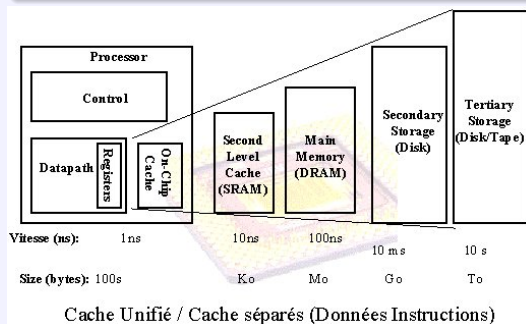
# Vitesse des processeurs et de la mémoire



# Hiérarchie de la mémoire

## Principe de localité

- Localité spatiale : les données ou instructions proches seront utilisées en même temps (tableau, vecteurs, ...)
- Localité temporelle : les données ou instructions récemment utilisées seront réutilisées dans un futur proche (boucles, ...)



# Bus

- Largeur du bus
  - le nombre de lignes physiques sur lesquelles les données sont envoyées de manière simultanée
  - exprimé en bits (32 bits, 64 bits)
- Fréquence
  - le nombre de paquets de données envoyés ou reçus par seconde
  - exprimé en Hz (MHz)
- Débit maximal (taux de transfert maximal)
  - Débit = Fréquence  $\times$  Largeur  $\times$  Pumping
  - P.ex. pour la mémoire de type DDR2-800 (Double Data Rate two Synchronous Dynamic Random Access Memory) cadencée à 400 MHz avec le FSB à 200 MHz

$$\text{Memoire} : 400\text{MHz} \times 2 \text{ (double data)} \times 64\text{bits}/8 = 6400\text{Mo/s}$$

$$\text{FSB} : 200\text{MHz} \times 4 \text{ (quad pumped)} \times 64\text{bits}/8 = 6400\text{Mo/s}$$

# Architecture 32 bits et 64 bits

- Gestion des nombres entiers sur 32 ou 64 bits
- Par conséquent, la mémoire vive adressable est de :
  - $2^{32}$  octets soit 4 Go pour l'architecture 32 bits
  - $2^{64}$  octets soit 16 Exo pour l'architecture 64 bits
- Processeurs 32 bits :
  - Processeurs Intel jusqu'au Pentium IV
  - Processeurs AMD jusqu'au Athlon XP
  - Processeurs ARM (smartphones, Raspberry Pi)
- Processeurs Intel avec support Intel 64 (EM64T) : Core 2 (Duo, Quad,...), Intel Xeon Série 5xxx-9xx, Xeon LV, Intel Core Ix,
- Processeurs 64 bits :
  - Processeurs Intel Itanium et Itanium II
  - Processeurs AMD Athlon 64, Athlon FX, Athlon X2, Sempron 64, Turion et Opteron
  - Processeurs Sun Microsystems, SGI, CRAY, NEC, DEC, IBM,...

## Calcul scientifique : comment calculer plus vite ?

- Processeur(s) plus rapide(s) : plus d'opérations par seconde
- Bus plus efficace : augmentation de la bande passante
- Mémoire vive plus rapide

## Réponse des constructeurs : différentes architectures

- Super-scalaire
- Parallèle
- General-Purpose Processing on Graphics Processing Units (GPGPU)
- Intel Many Integrated Core Architecture (MIC)

## Architecture super-scalaire

- Optimisation d'un processeur (FPU) afin d'augmenter le nombre d'instruction par cycle
- Exemples :
  - Pipelining : plusieurs instructions se recouvrent dans leurs exécutions
  - Overlapping : parallélisation des opérations élémentaires
  - Chaining : enchaînement des instructions élémentaires



# Pipelining : la laverie

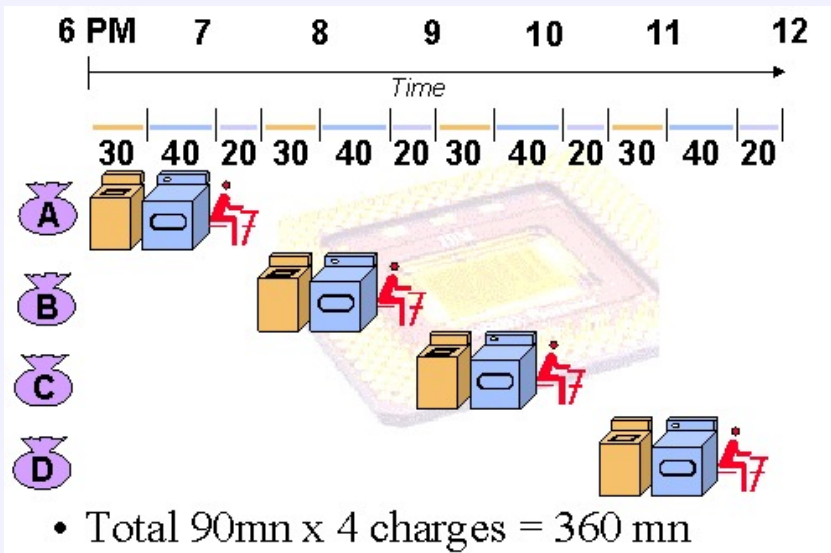
- Anne, Bernard, Claude, David ont chacun du linge à nettoyer, sécher, repasser



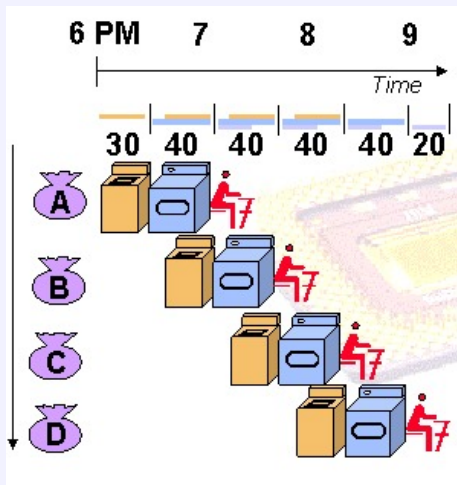
- Laver : 30minutes
- Sécher : 40 minutes
- Repasser 20 minutes
- Total : 90 minutes



# Pipelining : la laverie séquentielle



# Principe du Pipelining



- Temps :  $30 + 40 * 4 + 20 = 210$  min
- La latence (temps de chargement) ne change pas
- La vitesse est limitée par le traitement le plus long
- Plusieurs tâches en même temps
- Augmentation potentielle de la vitesse :  $\times 3$
- Avec les temps d'exécution différents, gain en vitesse de 1.5
- Les temps de chargement réduisent la vitesse

# Principe du Pipelining

Instr No.	Pipeline Stage						
<b>1</b>	IF	ID	EX	MEM	WB		
<b>2</b>		IF	ID	EX	MEM	WB	
<b>3</b>			IF	ID	EX	MEM	WB
<b>4</b>				IF	ID	EX	MEM
<b>5</b>					IF	ID	EX
<b>Clock Cycle</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>

- IF : Instruction Fetch
- ID : Instruction Decode
- EX : Execute
- MEM : Memory access
- WB : Register write back

Source : [http://en.wikipedia.org/wiki/Instruction\\_pipeline](http://en.wikipedia.org/wiki/Instruction_pipeline)

Au quatrième cycle d'horloge (la colonne verte), la première instruction est en phase de MEM, et la dernière instruction n'est pas encore entrée dans le pipeline

# Types d'Architectures parallèles

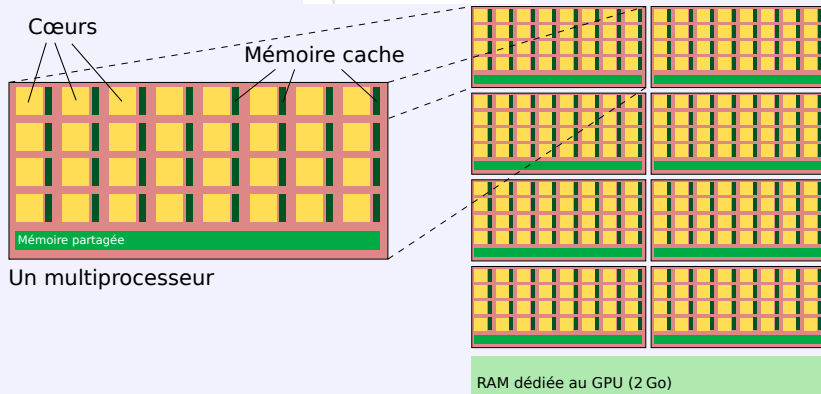
## Mémoire et communications

- Mémoire distribuée
- Mémoire partagée

## Niveau de parallélisation supporté

- Calcul multi-coeurs
- Symmetric multiprocessing (SMP)
- Calcul distribué (distributed memory multiprocessor)
- Cluster
- Calcul Massivement Parallèle
- Grid Computing

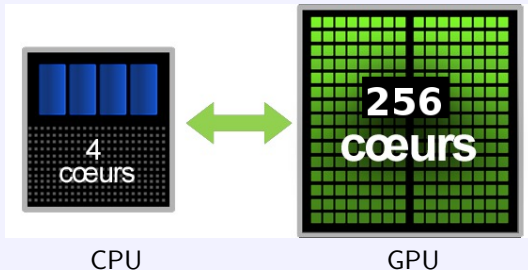
# General-Purpose Processing on Graphics Processing Units (GPGPU)



Un multiprocesseur

*Schéma de l'architecture d'une carte Nvidia Quadro 4000, avec 8 multiprocesseurs de 32 coeurs chacun (env.700 Euros au 09/2013)*

# General-Purpose Processing on Graphics Processing Units (GPGPU)



## Calcul en Mécanique des Fluides sur le(s) GPU (calcul hétérogène)

- Le CPU est utilisé pour les opérations sérielles (séquentielles), avec beaucoup de branchements et d'accès aléatoires à la mémoire
- Le GPU est utilisé pour des calculs massivement parallèles avec des flottants (multiplication matricielle p.ex.).