

Architecture élémentaire

Samy Meftali

Samy.meftali@univ-lille.fr

Bureau 224. Bâtiment M3 extension

Évaluation

➤ Contrôle continu

- test 1 - un lundi en cours (début novembre) la 6^{ieme} séance? (4 points) (T1)
- test 2 - durant la session d'examen (T2)
- Test3 - Note de TD/TP sur 16 (T3)
 - 12 séances TP sont pointés pour 1 point
 - 4 points pour l'enseignant de TD/TP
- $T_{cc} = T1 + T3$
- note finale = $\max ((T2 + T_{cc})/2, T2)$

12 séances

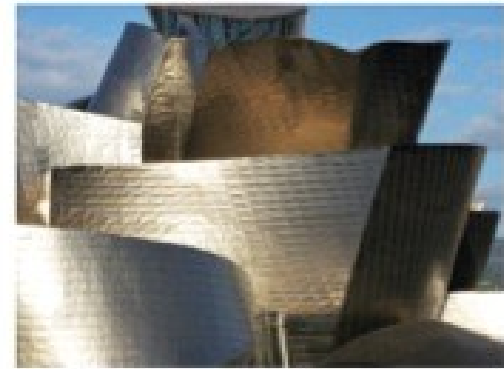
1. C'est quoi l'architecture?
2. De l'électronique à l'informatique
3. Du binaire à l'information - Quelques circuits élémentaires
4. Circuits séquentiels
5. Synthèse d'un circuit séquentiel
6. Contrôle continu ?
7. Un modèle d'exécution : Von Neumann
8. Instructions machine
9. De l'instruction à la micro programmation
10. La fonction mémoire
11. Le bus système
12. Les entrées/sorties

Un cours d'architecture pour des informaticiens

- **Sans architecture pas d'informatique!**
- **Comprendre comment ça marche permet de mieux utiliser un ordinateur**
- **Demain le hardware deviendra aussi programmable**
 - **Il faudra être capable d'en tirer un bénéfice**

Le bouquin à lire!!!

- **Computer Organization and Architecture 9th Edition**
- <http://http://williamstallings.com/ComputerOrganization/>
- <http://williamstallings.com/ComputerOrganization/COA9e-student/>



Computer
Organization
and Architecture
Designing for Performance
Ninth Edition

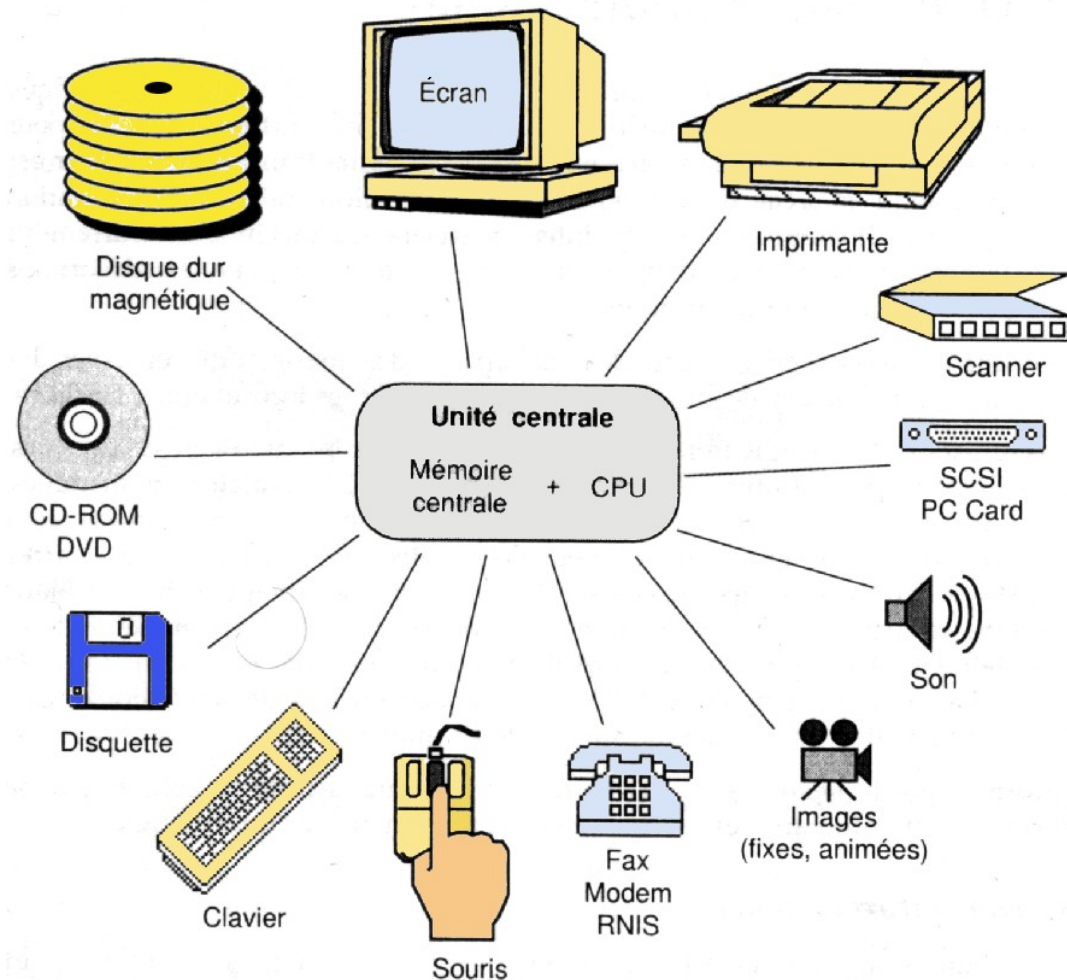
William Stallings

À la fin de cette unité

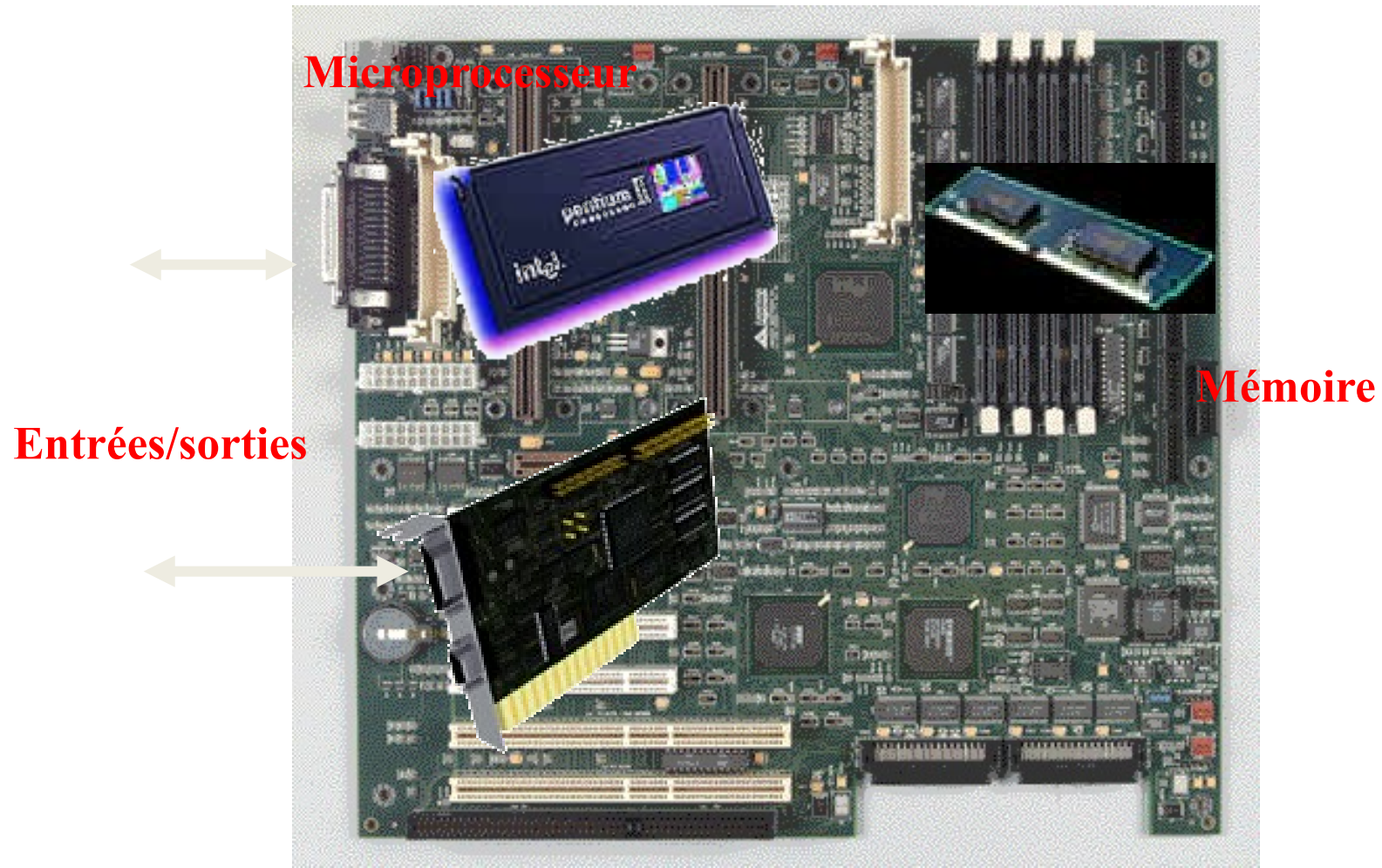
- Vous aurez pris connaissance de l'évolution phénoménale de l'ordinateur depuis ses débuts.
- Vous comprendrez pourquoi on utilise le binaire en informatique;
- Vous serez capable de construire un circuit élémentaire sur une carte électronique
- Vous connaîtrez les composantes essentielles de la *Machine de von Neumann*.
- Vous construirez un processeur simple sur une carte électronique

Principaux éléments d'un ordinateur

- **Unité centrale**
- **Cédérom ou DVD**
- **Disque dur**
- **Disquette**
- **Clavier**
- **Souris**
- **Écran ou projecteur**
- **Modem**
- **Scanner**
- **Carte de son**
- **Images et vidéo**
- **Fax modem**
- **Ports SCSI, USB, etc.**



Vue coté architecture...



Parlons chiffre!

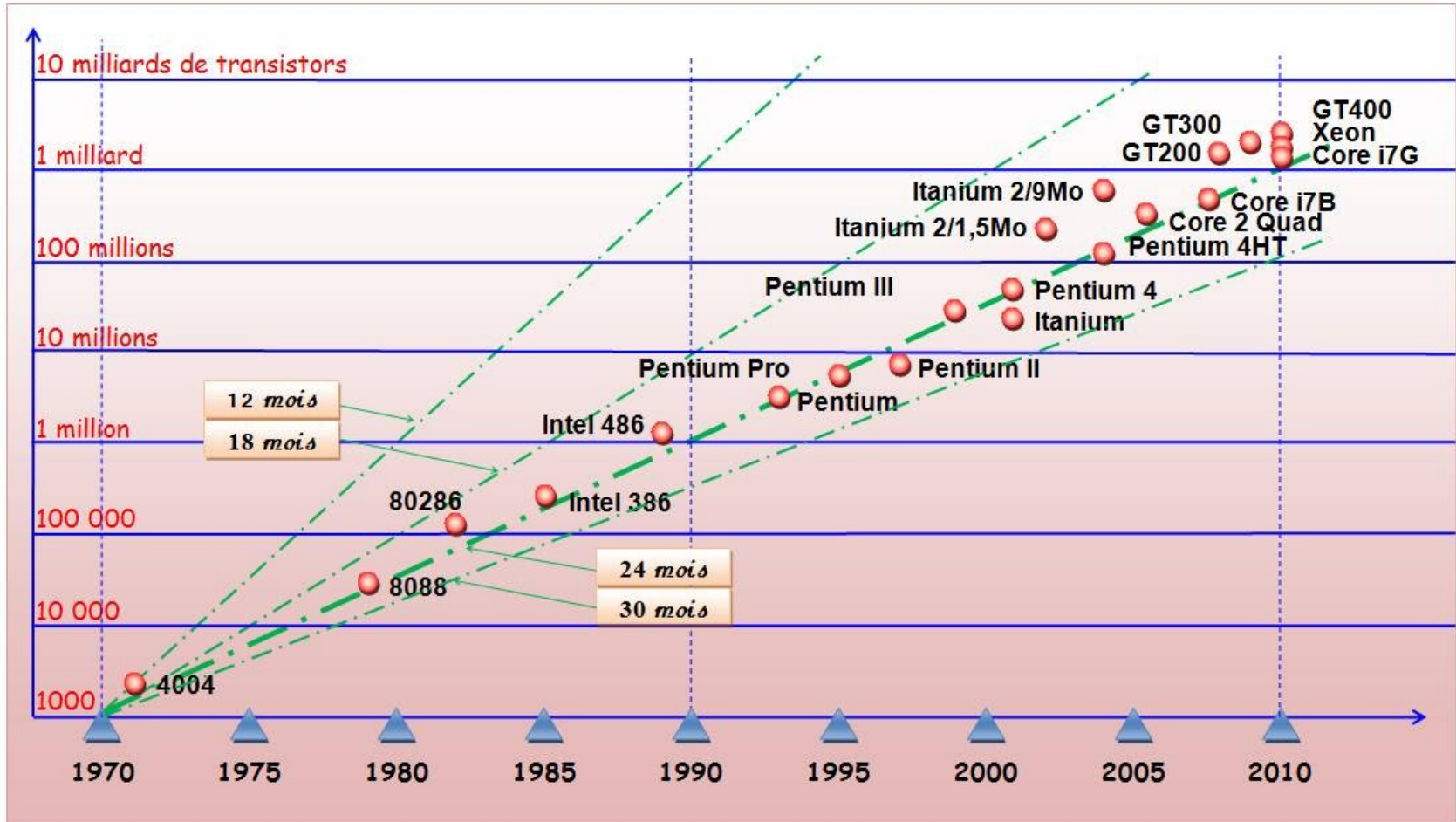
➤ Unités de mesure de capacité

Nom	Symbole	Puissances binaires et valeurs en décimal	Nombre	Hexa	Ordre de grandeur SI décimal
unité	o/B	$2^0 = 1$	un(e)	1	$10^0 = 1$
kilo	ko/Ko kB/KB	$2^{10} = 1\ 024$	mille	400	$10^3 = 1\ 000$
méga	Mo/MB	$2^{20} = 1\ 048\ 576$	million	100000	$10^6 = 1\ 000\ 000$
giga	Go/GB	$2^{30} = 1\ 073\ 741\ 824$	milliard	40000000	$10^9 = 1\ 000\ 000\ 000$
téra	To/TB	$2^{40} = 1\ 099\ 511\ 627\ 776$	billion	10000000000	$10^{12} = 1\ 000\ 000\ 000\ 000$
péta	Po/PB	$2^{50} = 1\ 125\ 899\ 906\ 842\ 624$	billiard	400000000000	$10^{15} =$ 1 000 000 000 000 000
exa	Eo/EB	$2^{60} = 1\ 152\ 921\ 504\ 606\ 846\ 976$	trillion	1000000000000000	$10^{18} =$ 1 000 000 000 000 000 000

➤ Unités de mesure de temps

- ms = milliseconde = 10^{-3} s = 0,001 s
- μ s = microseconde = 10^{-6} s = 0,000 0001 s
- ns = nanoseconde = 10^{-9} s = 0,000 000 001 s
- ps = picoseconde = 10^{-12} s = 0,000 000 000 001 s

Toujours plus...



Pourquoi l'évolution de l'architecture?

- Loi de Moore: 2X plus de transistors / 18 mois
 - En 20 ans: plus 1000X (de 4 à 0.13 microns)
 - Le milliard de transistors est dépassé, plusieurs GHz
- Différence croissante entre le délai intra-chip et le délai extra-chip
- Importance croissante des interconnexions entre les transistors

Défis

- Taille et puissance
 - Comment gérer l'augmentation de puissance consommée?
- Performance du système
 - Qu'est-ce qui définit la performance?
 - Où doit-on miser pour augmenter la performance?
- Complexité
 - Comment gérer la conception de systèmes de plus en plus gros?

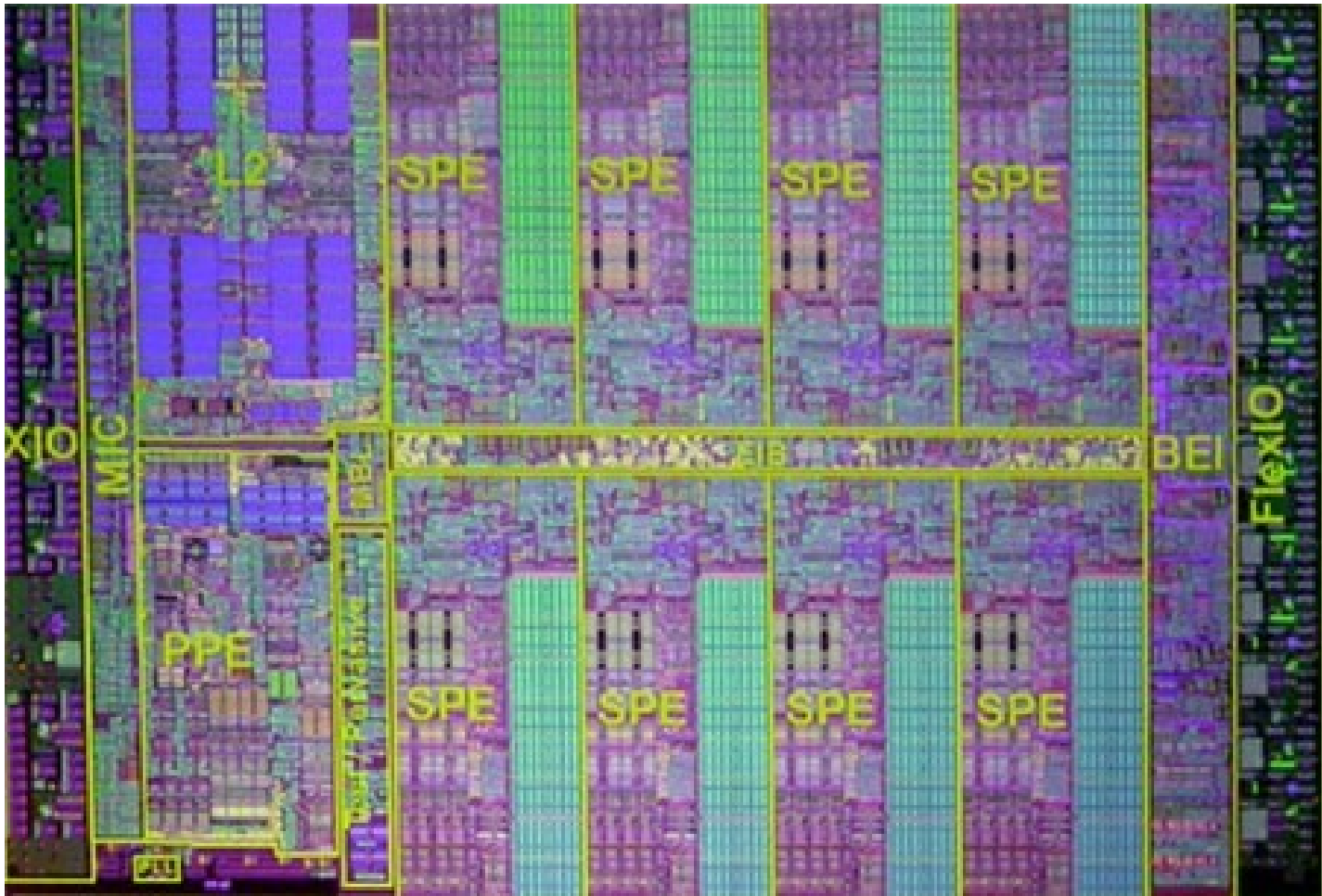
Force motrice derrière l'architecture des ordinateurs

- Fin des années 80, début 90:
 - Performance des processeurs
- Fin des années 90, début 2000:
 - Puissance
 - Accès à la mémoire lent
 - Fiabilité
 - Coût de développement, maintenance de logiciel
 - Performance des processeurs

Comment améliorer la performance?

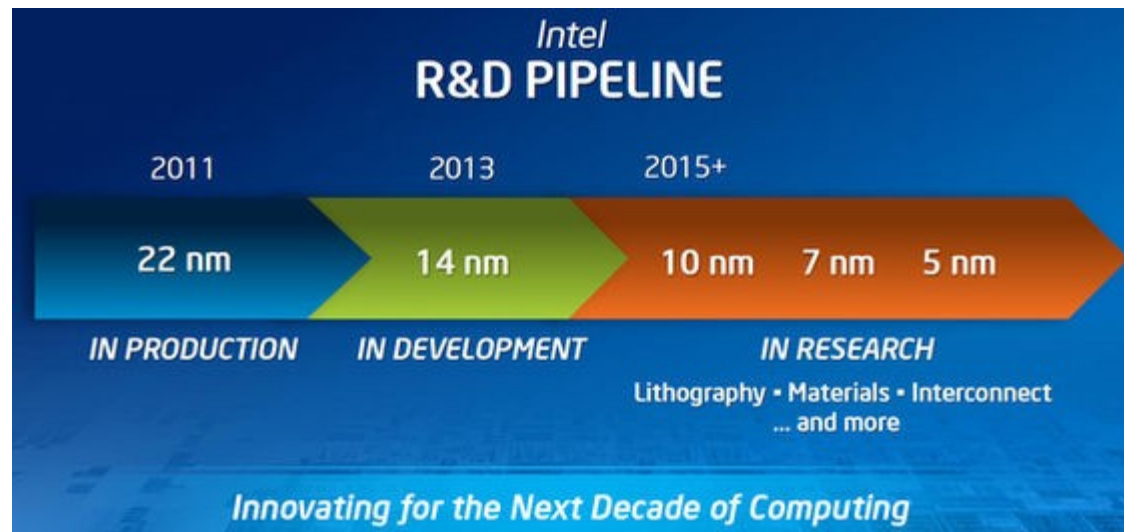
- Matériel
 - Dispositifs
 - Ensemble d'instructions (RISC vs CISC)
- Conception
 - Techniques de synthèse
 - Techniques de test
- Parallélisme
 - MLP (Machine Level Parallelism)
 - ILP (Instruction Level Parallelism)

Le Cell de votre PS3



Dynamique de l'industrie du Semiconducteur

- De la gravure horizontale et verticale dite en 3D
 - Relance la loi de Moore
- 2009: Intel 32nm, 2010: 28nm , ça continue 22 en 2012. 14 en 2013



Ce qu'il y a de mieux aujourd'hui



PRESENTED BY
UNIVERSITY OF
MANNHEIM

ICL
INNOVATIVE
COMPUTING LABORATORY
UNIVERSITY OF TENNESSEE

Lawrence Berkeley
National Laboratory

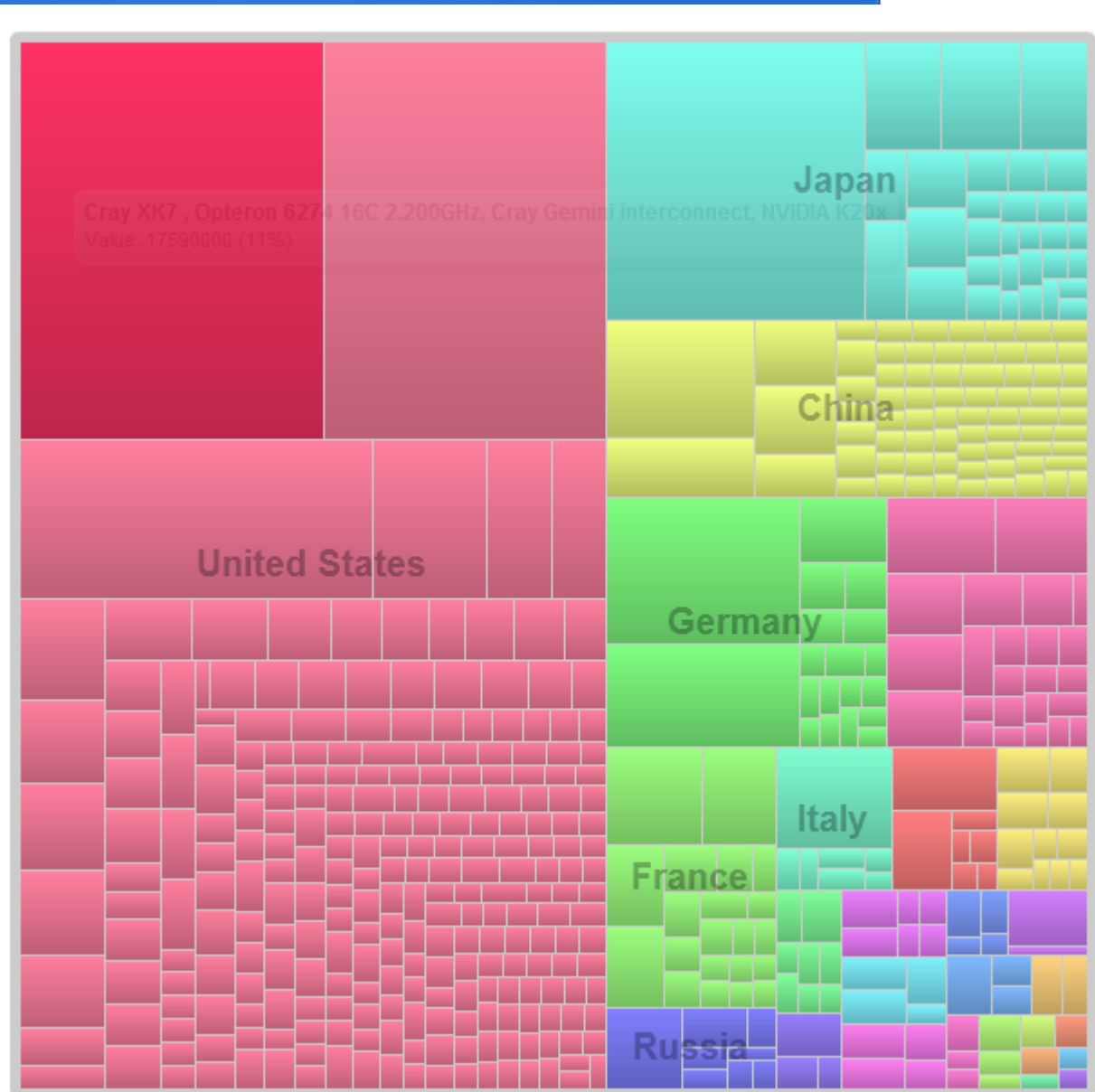
FIND OUT MORE AT
www.top500.org

	NAME	SPECS	SITE	COUNTRY	CORES	R _{MAX} P _{FLOP/S}	POWER MW
1	TITAN	Cray XK7, Operon 6274 16C 2.2 GHz + Nvidia Kepler GPU, Custom interconnect	DOE/OS/ORNL	USA	560,640	17.6	8.3
2	SEQUOIA	IBM BlueGene/Q, Power BQC 16C 1.60 GHz, Custom interconnect	DOE/NNSA/LLNL	USA	1,572,864	16.3	7.9
3	K COMPUTER	Fujitsu SPARC64 VIIIfx 2.0GHz, Custom interconnect	RIKEN AICS	Japan	705,024	10.5	12.7
4	MIRA	IBM BlueGene/Q, Power BQC 16C 1.60 GHz, Custom interconnect	DOE/OS/ANL	USA	786,432	8.16	3.95
5	JUQUEEN	IBM BlueGene/Q, Power BQC 16C 1.60 GHz, Custom interconnect	Forschungszentrum Jülich	Germany	393,216	4.14	1.97

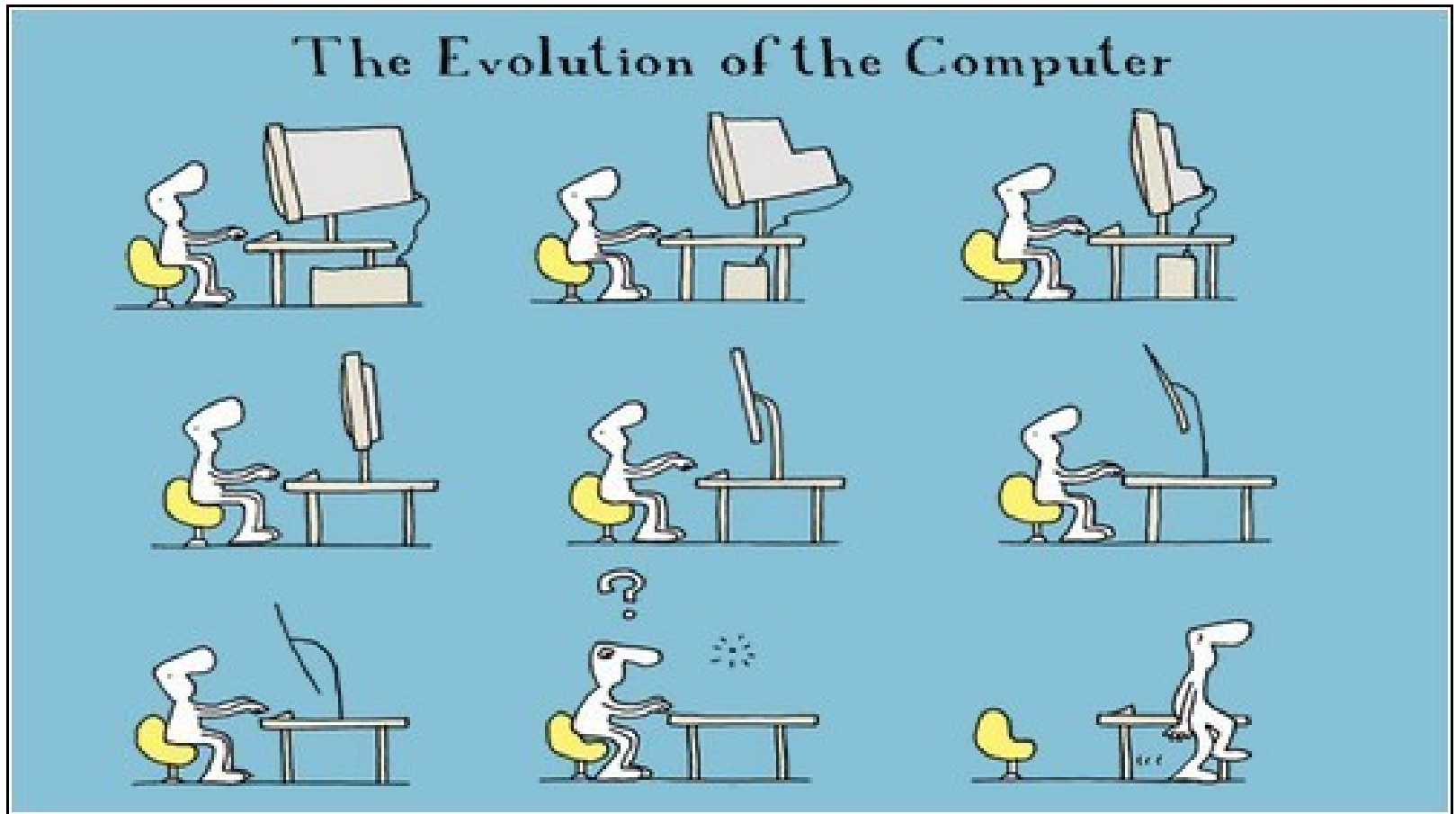
PERFORMANCE DEVELOPMENT



Un signe de puissance mondiale

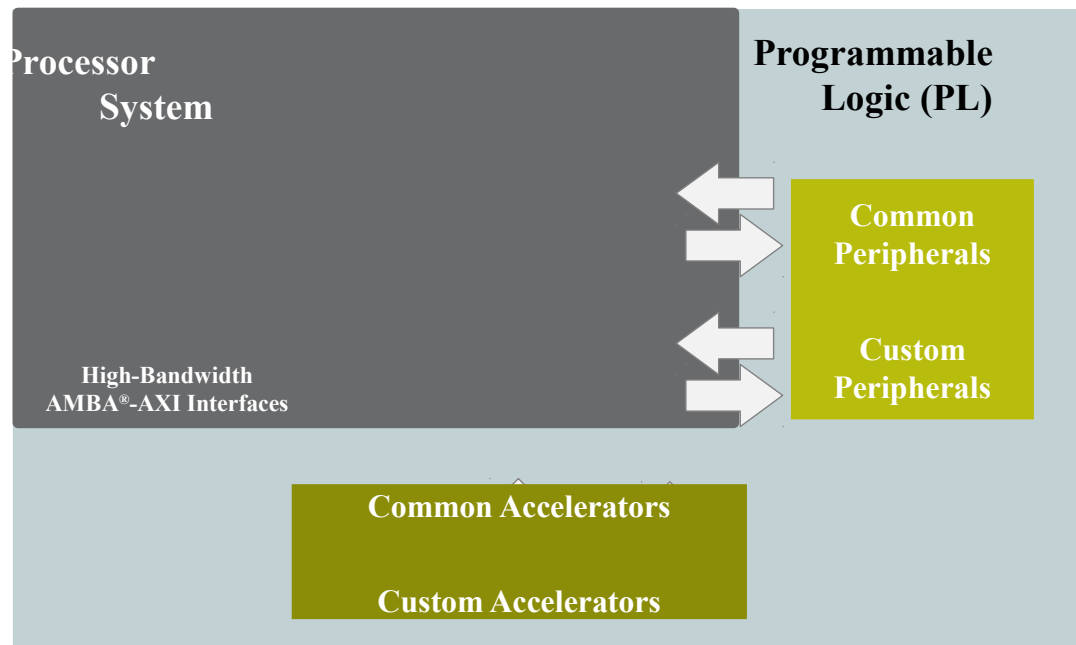


Et l'encombrement...



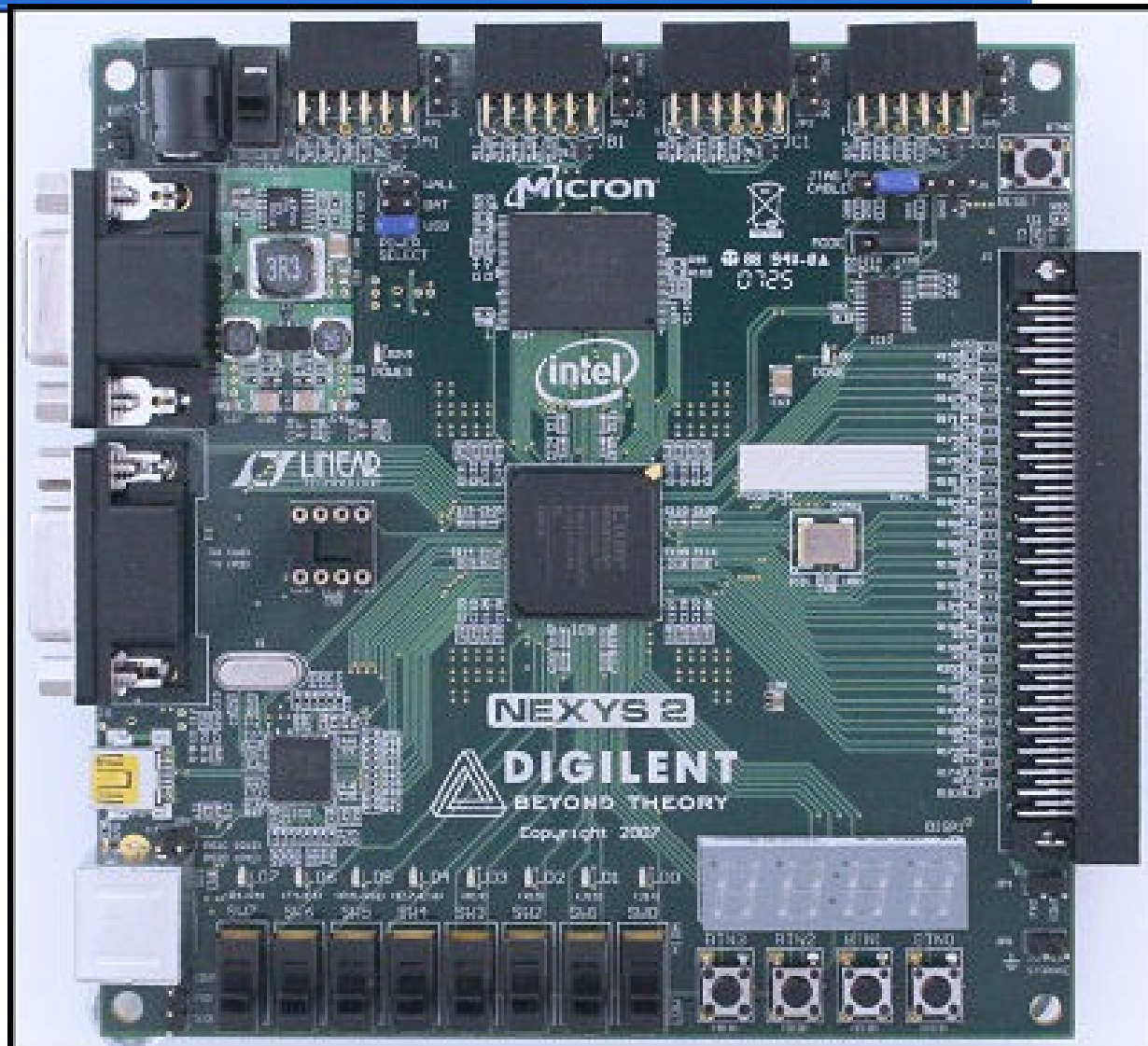
Les nouveau circuits programmable : FPGA

- **field-programmable gate array**
- **Circuit intégré logique qui peut être reprogrammé après sa fabrication.**



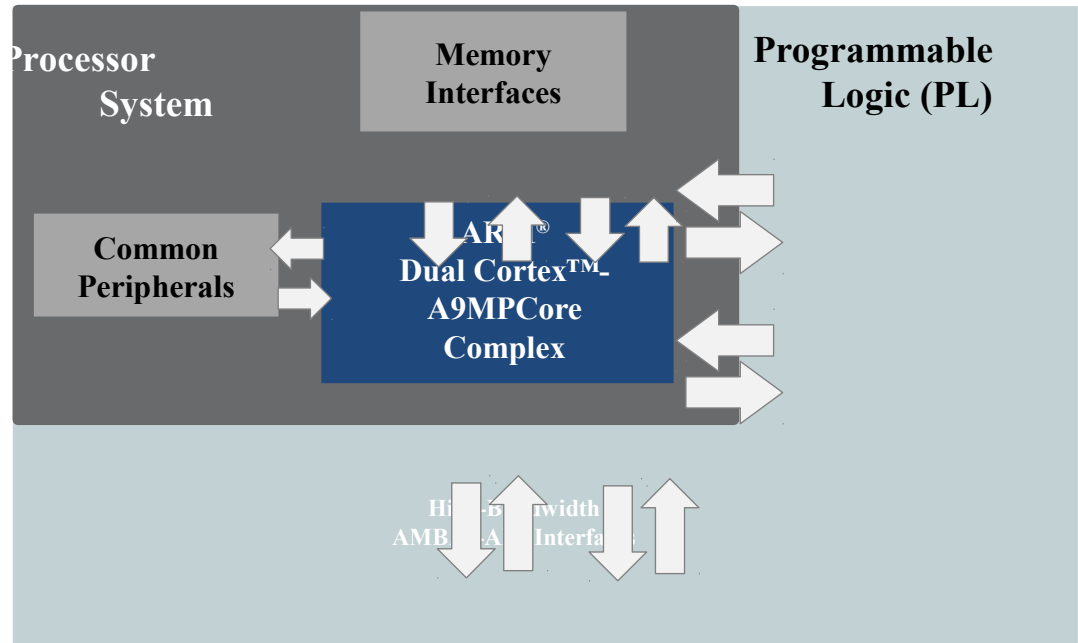
Change the program
Customize I/O interfaces
Add compute accelerators

La carte Nexys2

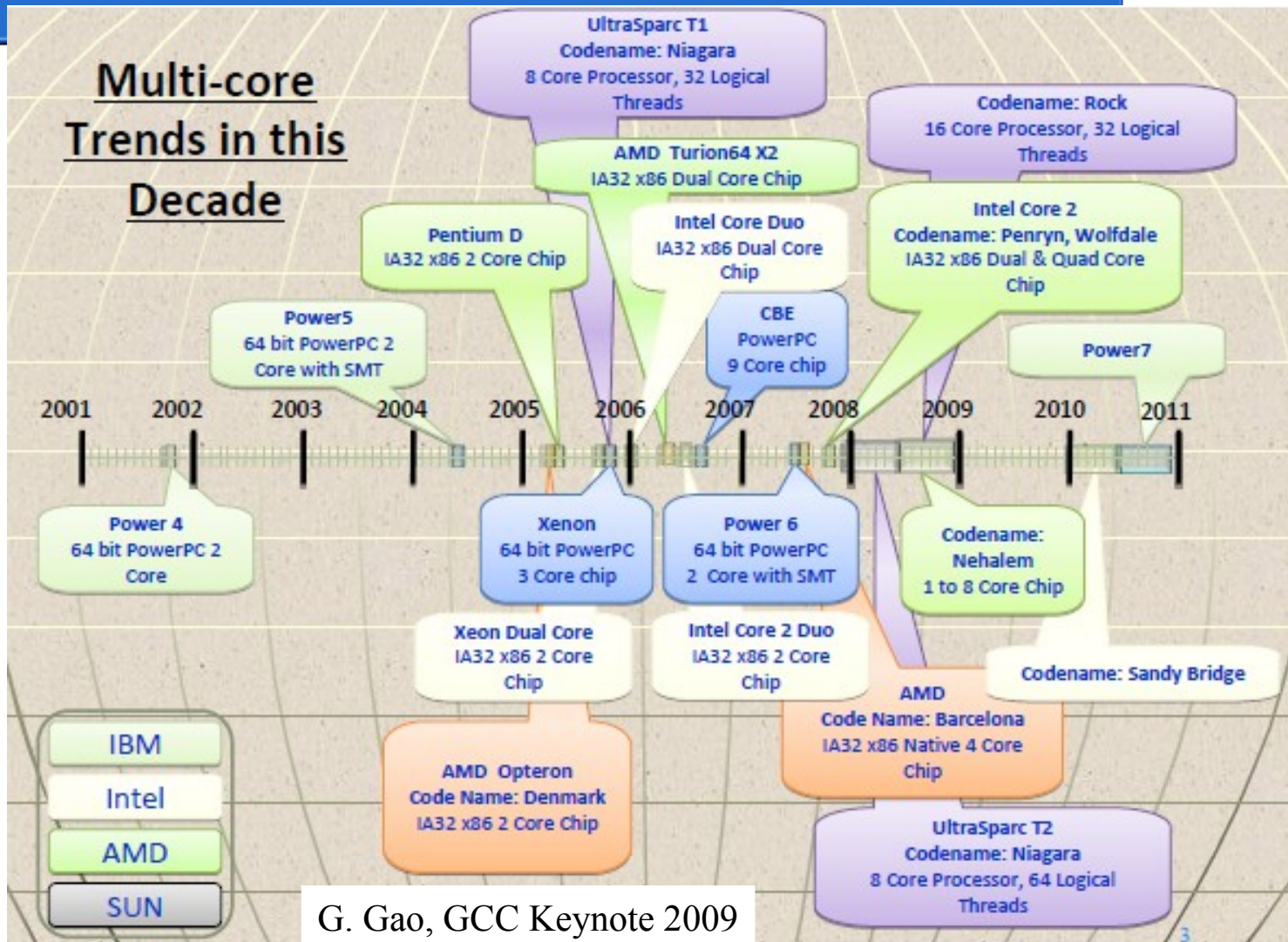


Vers les systèmes sur puce

Boots like a Processor
Acts like a Processor
Really ... is a Processor
... and more



La tendance Multi- et Many-Core



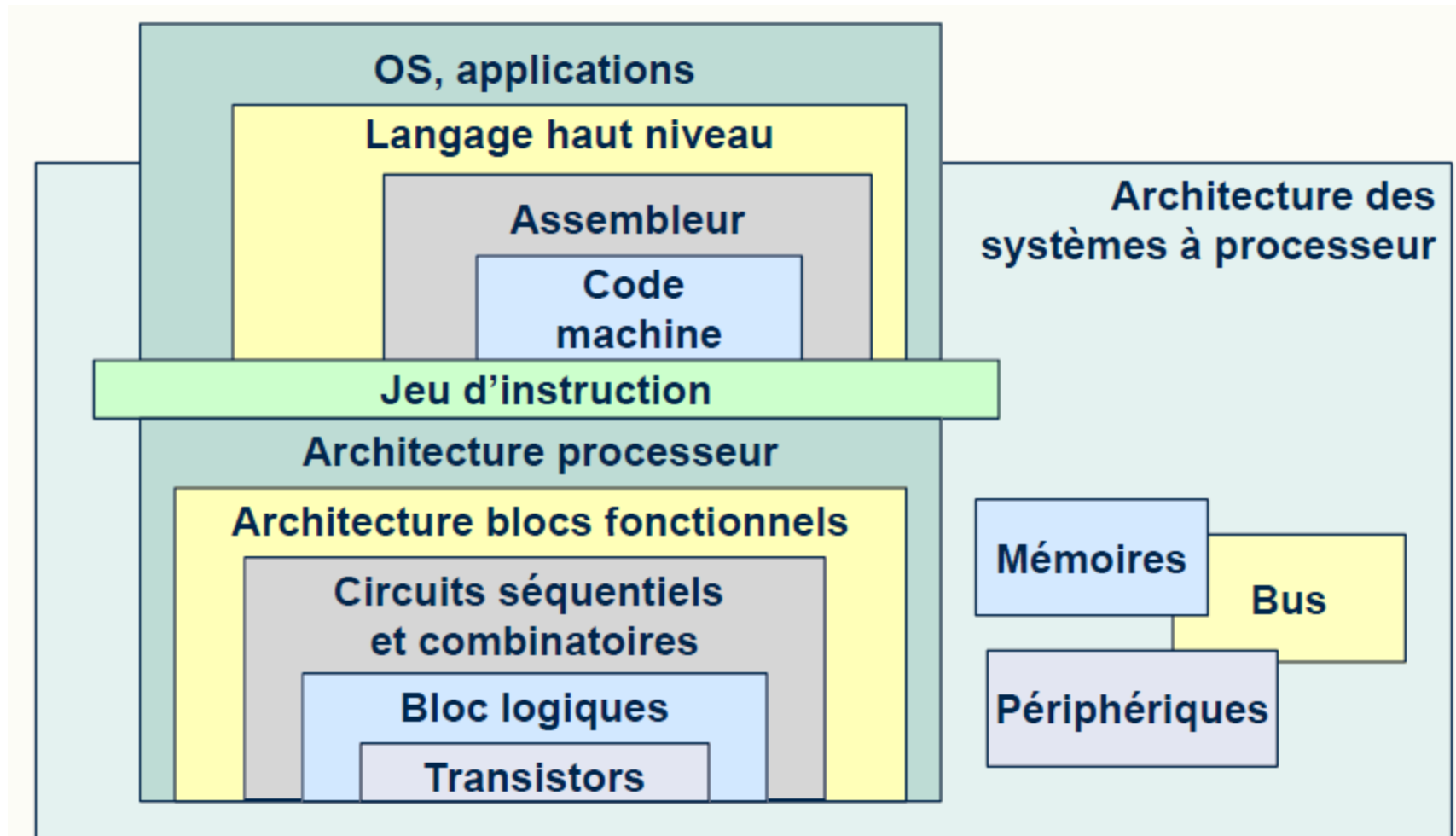
G. Gao, GCC Keynote 2009

Vue informaticienne de l'architecture

- Point de vue concepteur d'architecture
 - Définir des modèles d'exécution
 - Les valider : test preuve simulation prototypage
 - Les programmer : langages compilateurs

- Point de vue du programmeur
 - Comprendre le fonctionnement
 - Optimiser ses algorithmes
 - Critère de coût/performance
 - Réutiliser ses codes

Niveaux d'abstraction



A lire

- <http://www.commentcamarche.net/contents/histoire/ordinateur.php3>
- http://fr.wikipedia.org/wiki/Histoire_de_l%27informatique
- <http://fr.wikipedia.org/wiki/VHDL>
- http://fr.wikipedia.org/wiki/Loi_de_moore

Un livre en ligne

- <http://www.scribd.com/doc/47783341/Architecture-de-l'ordinateur>