

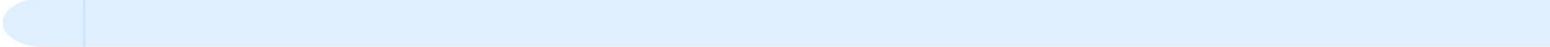


Ecole Doctorale Sciences et Techniques de l'Ingénieur
Département Génie Electrique

SoC (System On Chip) Conception Haut niveau

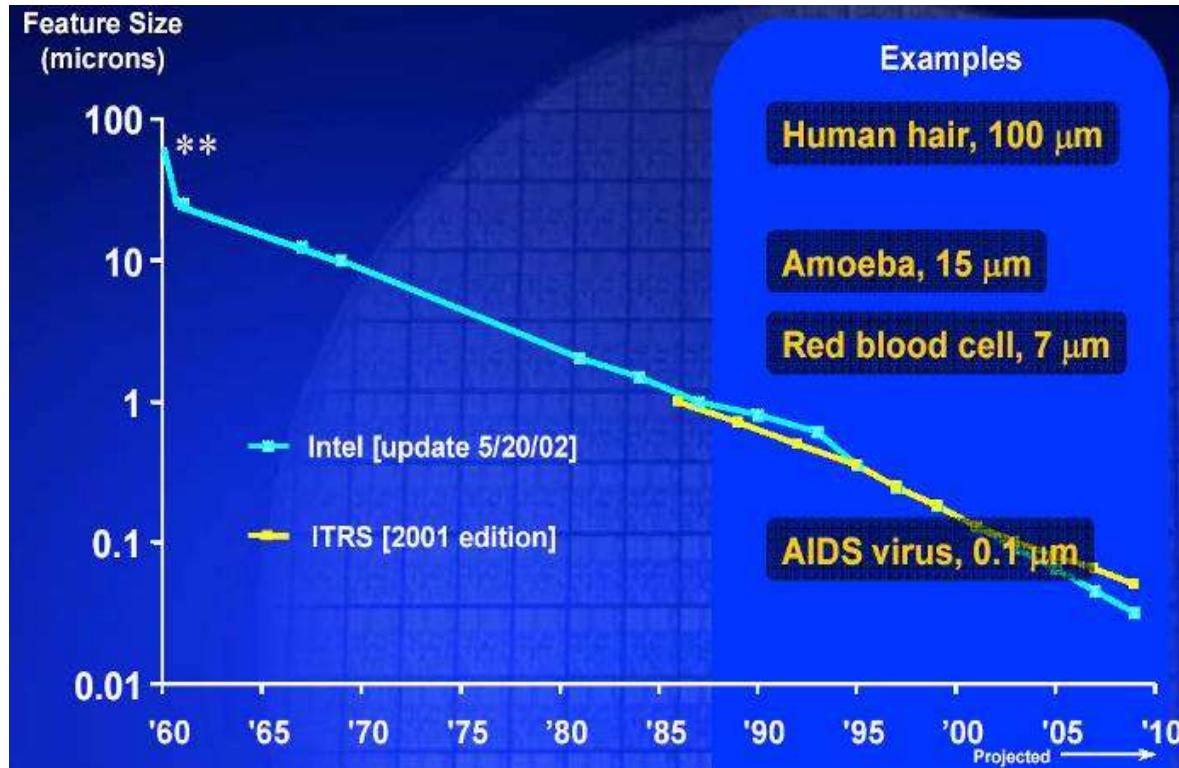
*Ingénieur 3ème année GE option: SMART
Mastères de recherche : Signaux, Systèmes et Données (SSD)
Information System techniques (IST)*

Dr. Faten BEN ABDALLAH
Faten.benabdallah@enit.utm.tn

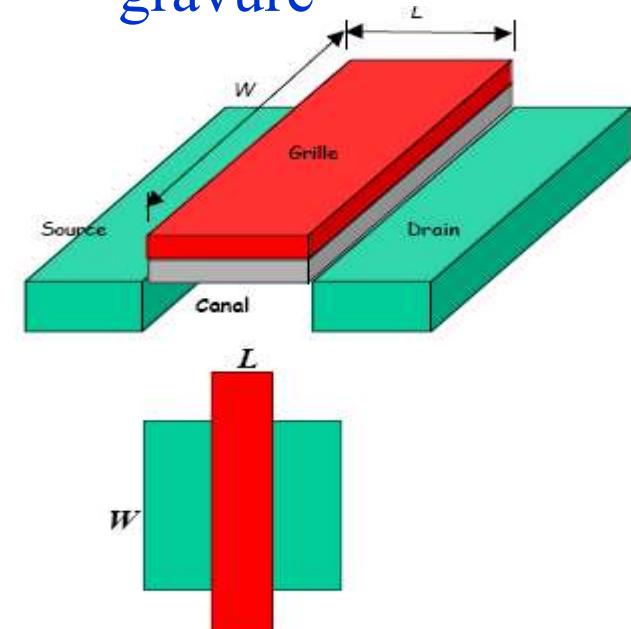


I. Introduction: Contexte & défis

Evolution des Transistors: Taille



Technologie de gravure



Nano-technologie: $L < 100\text{nm}$

- 10 et 7 nm en production, 5 et 3 nm dans la phase préparation
 - Nouvelle génération de technologie de gravure tous les 2-3 ans
- Un nombre de plus en plus grand sur la même tranche de silicium

Evolution des Transistors: Nombre

microprocesseurs Intel :

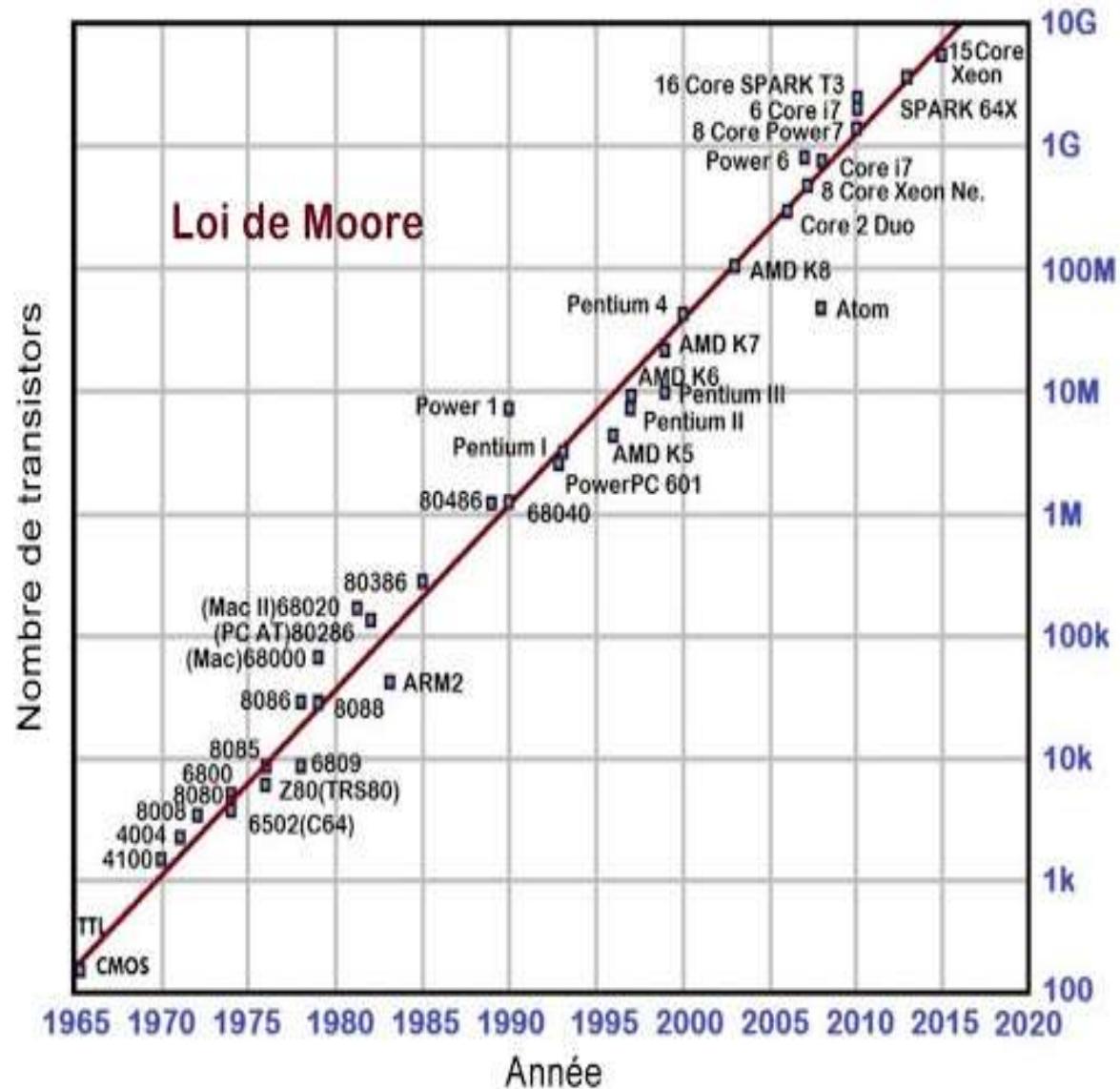
- 1971 : 4004 : 2 300 transistors
- 1978 : 8086 : 29 000 transistors
- 1982 : 80286 275 000 transistors
- 1989 : 80486 : 1,16 millions de transistors
- 1993 : Pentium : 3,1 millions de transistors
- 1995 : Pentium Pro : 5,5 millions de transistors
- 1997 : Pentium II : 27 Millions de transistors
- 2001 : Pentium 4 : 42 millions de transistors
- 2004 : Pentium Extreme Edition : 169 millions de transistors
- ...
- 2023: intel core i9: 50 milliards de transistors
- ...
- 2030: intel vise le trillion (un milliard de milliard) de transistors

Loi de Moore

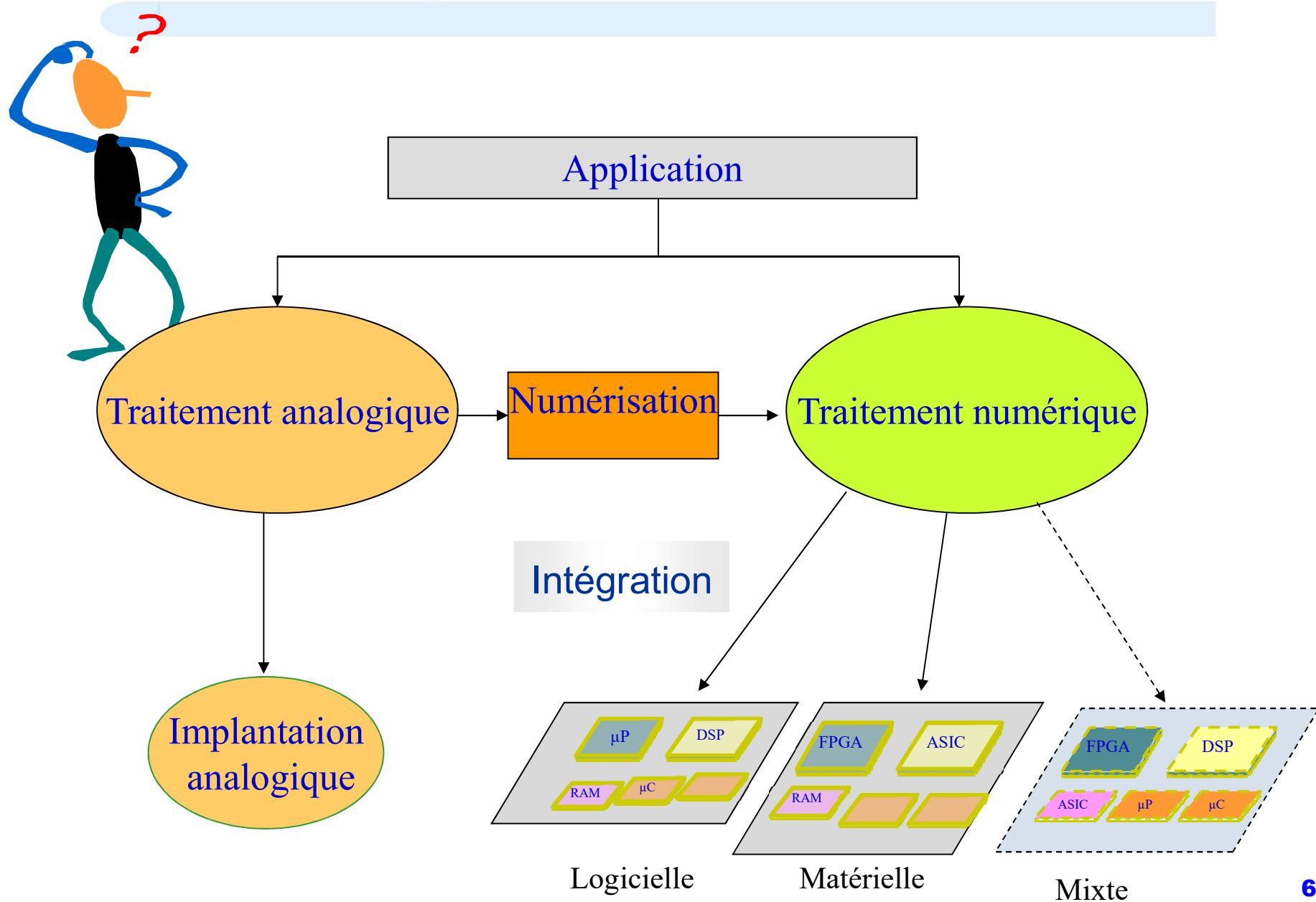
Seuls Samsung et TSMC, le fournisseur taïwanais d'Apple sont capables de franchir l'obstacle suivant des 5 nm. IBM, Toshiba, Sony.. tous les autres géants de l'électronique ont déclaré forfait.

Et si la prochaine étape, fixée à 3 nm, est peut-être atteignable, personne, n'ira au-delà.

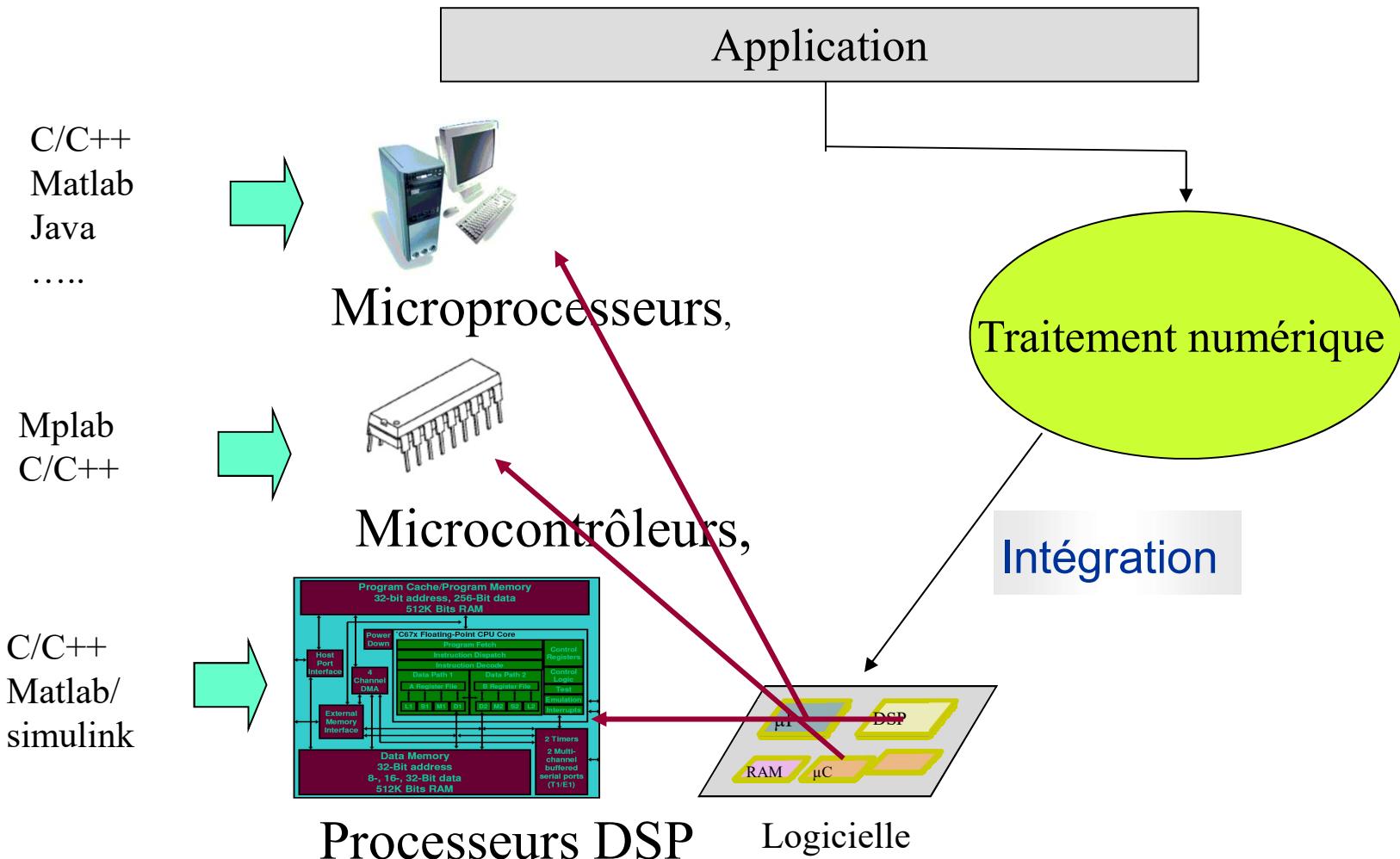
En 2023 au plus tard, il en sera fini de la loi de Moore.



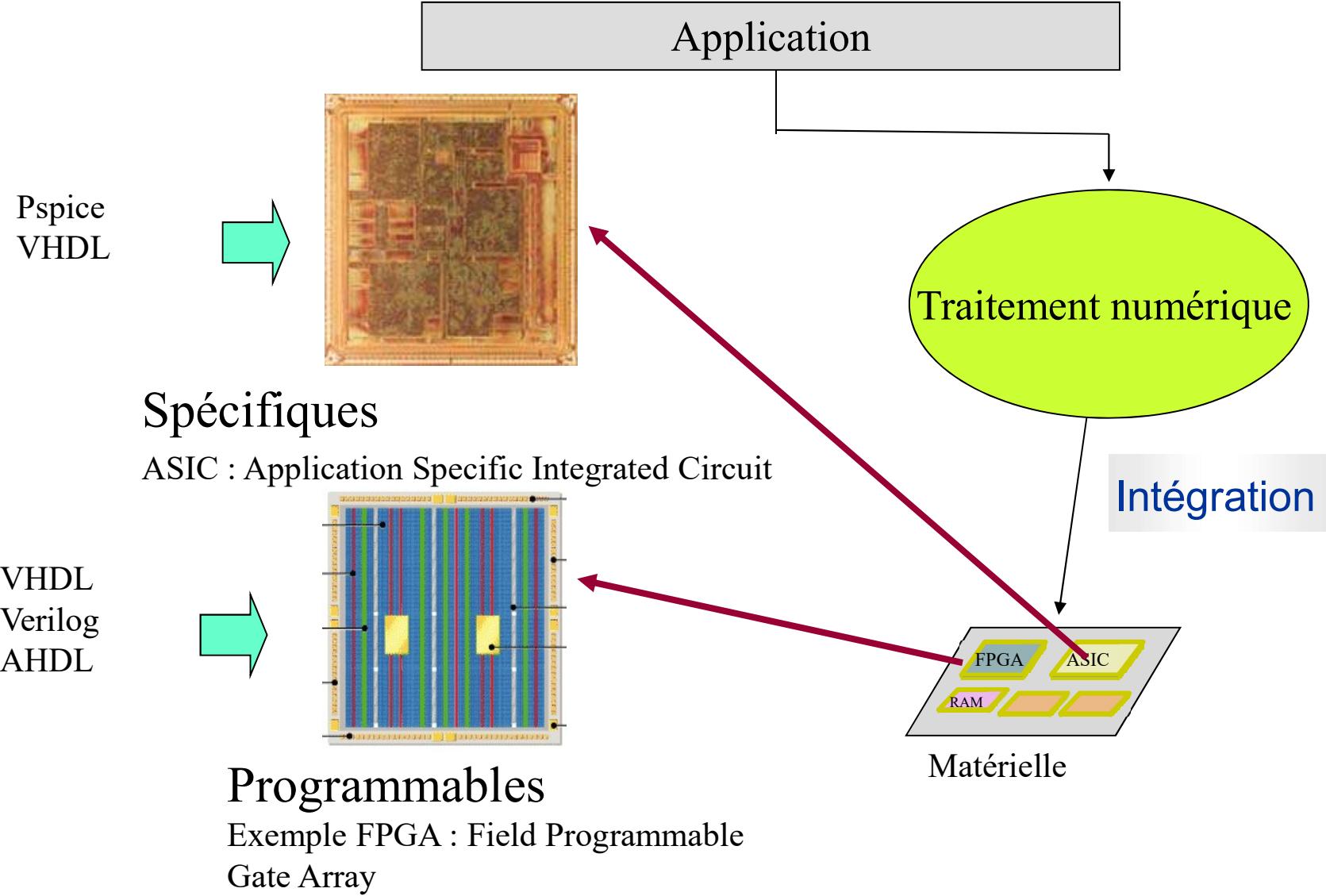
Cible d'implantation



Architecture logicielle



Architecture matérielle



Modèles de traitement

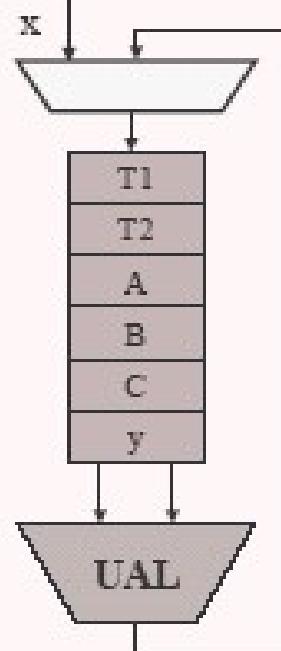
μP
μC
DSP

$$y = Ax^2 + Bx + C$$

FPGA
ASIC

Solution logicielle

Implémentation Temporelle

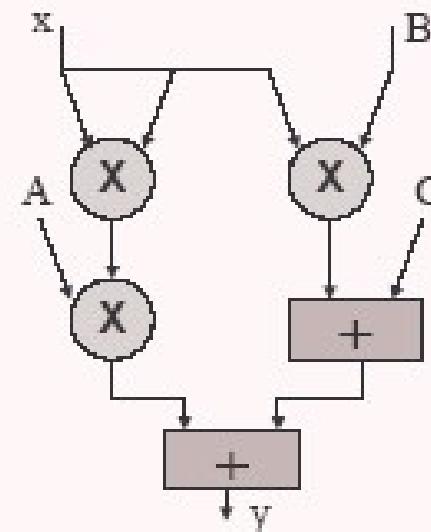


```
T1 <- x  
T2 <- A * t1  
T2 <- t2 + B  
T2 <- t2 * t1  
Y   <- t2 + C
```

Traitement séquentiel
=> 5 cycles d'horloge

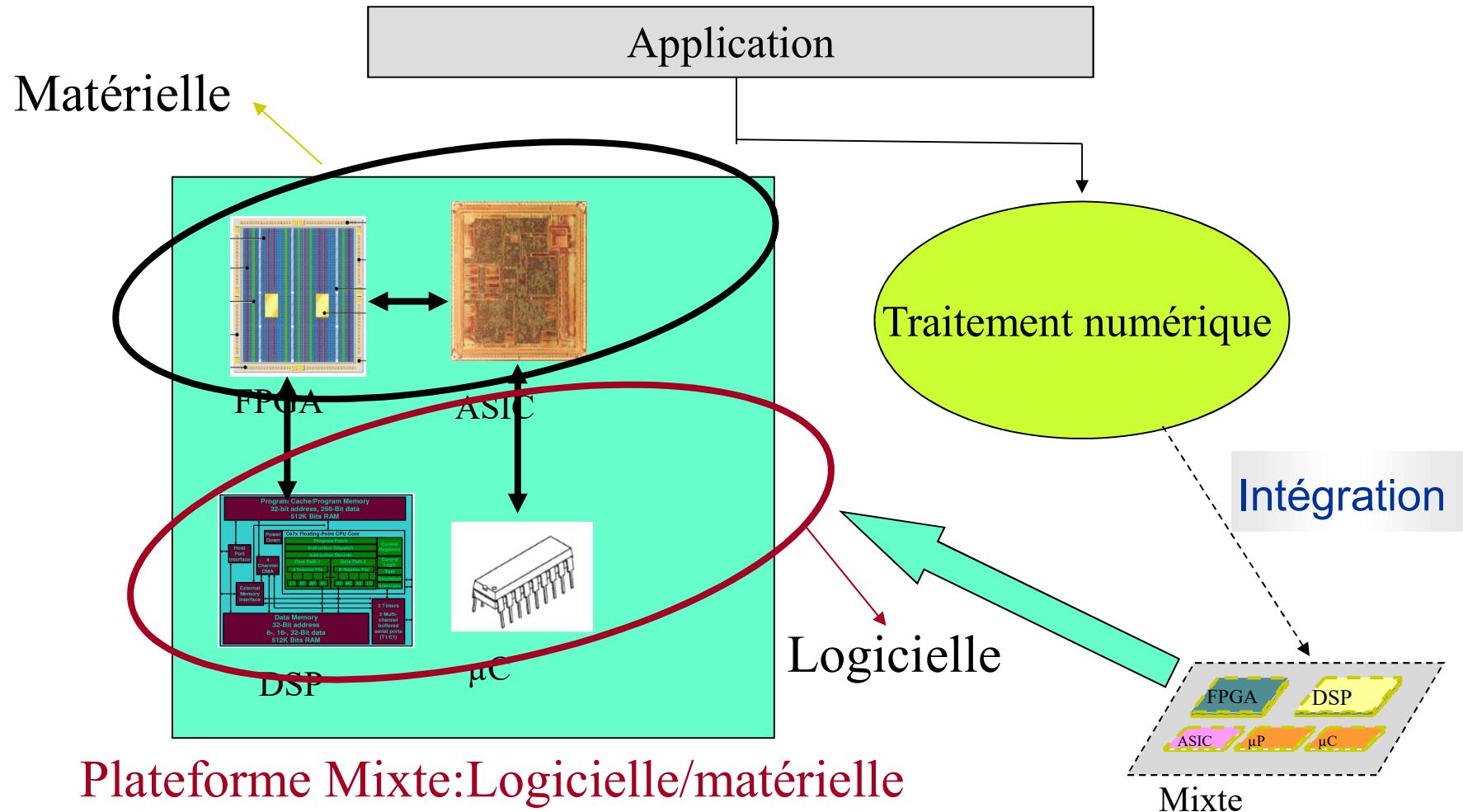
Solution matérielle

Implémentation Spatiale



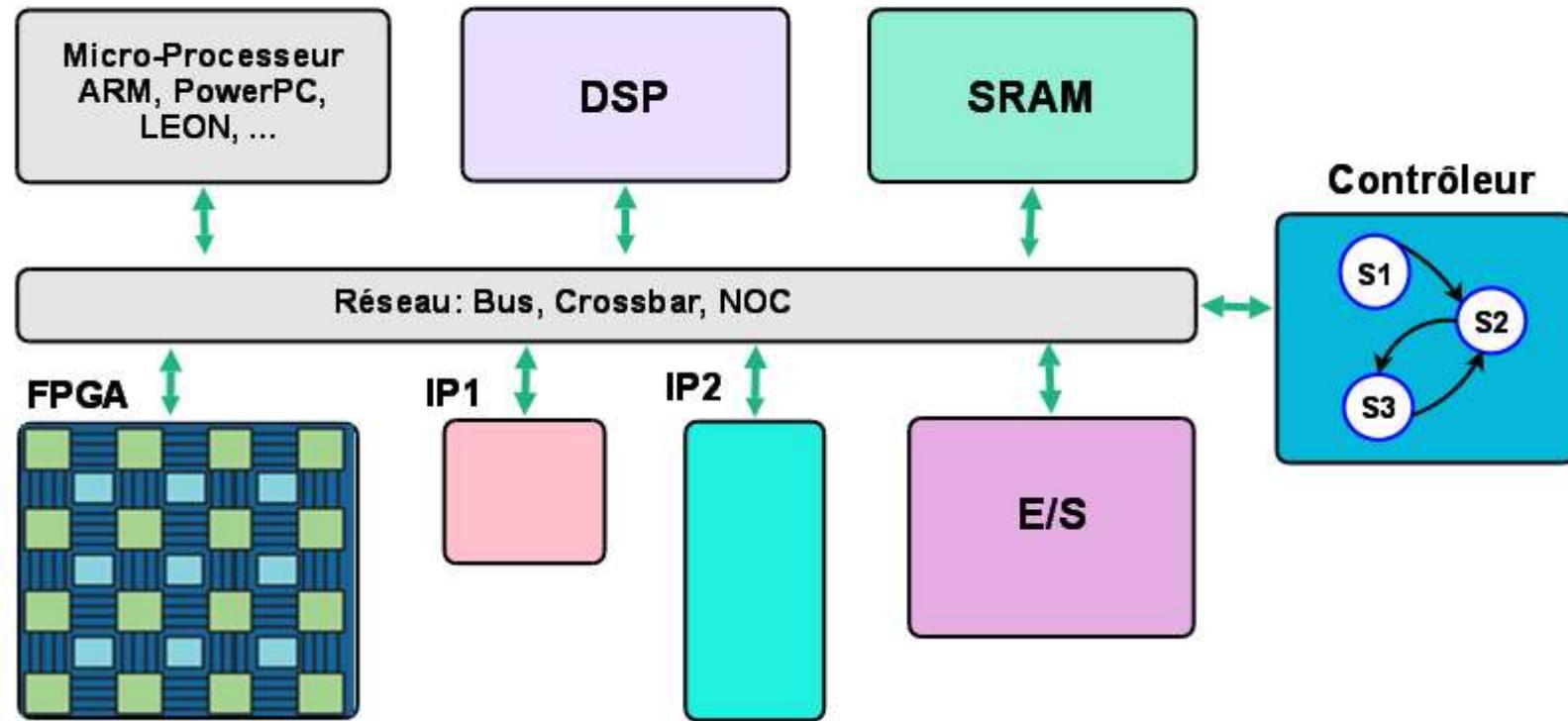
Traitement parallèle
=> 3 cycles d'horloge

Architecture mixte: Système sur puce (SoC)



Exemple d'un SOC hétérogène

Système sur Puce (SOC)



SOC: Puce regroupant tous les éléments électroniques (μ -processeur, composants spécifiques, mémoires...) nécessaires à la réalisation d'un Système (produit) complet.

Composants des SoCs:

1) µprocesseurs

- CPU (Central Processing Unit) : processeur généraux
 - Fonctions : contrôle, interface utilisateur, traitements légers
 - Exemples : ARM9, ST20, SH4, Leon mais pas Intel Pentium 4 !...
- DSP (Digital Signal Processor)
 - Fonctions : Traitement du signal, calculs complexes
 - Exemples : Ti TMS320C55x, etc.
- Processeurs VLIW (Very Long Instruction Word)
 - Fonctions : traitement multimédia
 - Exemples : ST210...
- Supports de la partie logicielle du système

Composants des SoCs:

2) Composants dédiés

- Accélération d'une fonction particulière
 - Serait trop lent en logiciel !
 - Compromis
 - Complexité
 - Accélération
 - Consommation
 - Flexibilité
- Plusieurs possibilités de réalisation :
 - 100% matérielle: ASIC, FPGA;
ex : Décodeurs Vidéos « câblés »
 - Semi-matérielle: Utilisation d'un cœur de processeur pour simplifier le matériel;
ex : Décodeurs audios, traitements plus difficiles à câbler

IP (Intellectual Property)

Composant virtuel

- Hard IP (IP matériel):
 - Netlist entière,
 - routage et optimisation pour une librairie technologique spécifique et layout personnalisé,
 - Non flexible ni portable
 - Prédictif (puissance, taille et performances connus optimisés pour une technologie donnée).
- Soft IP (IP logiciel):
 - Composant fourni sous forme HDL synthétisable ou SystemC.
 - Flexible (paramètres génériques pouvant être changés)
=> réutilisable dans de nombreuses applications,
 - Non prédictif (puissance, surface et temps)
- Firm IP (IP flexible):
 - Délivré sous forme de Netlist,
 - Possibilité d'optimiser les performances à travers le placement, routage.

Composants des SoCs:

3) Mémoires

- ROM, RAM, Flash...
 - Souvent hors du SoC
- Dans la puce :
 - Contrôleur(s) mémoires
 - Petites mémoires internes
 - Mémoires caches, « fifo » (tampons/files d'attente)
- Fonctions
 - Stockage temporaire (RAM)
 - Programme interne (ROM), possibilité de mise à jour (Flash)

Composants des SoCs:

4) Composants utilitaires

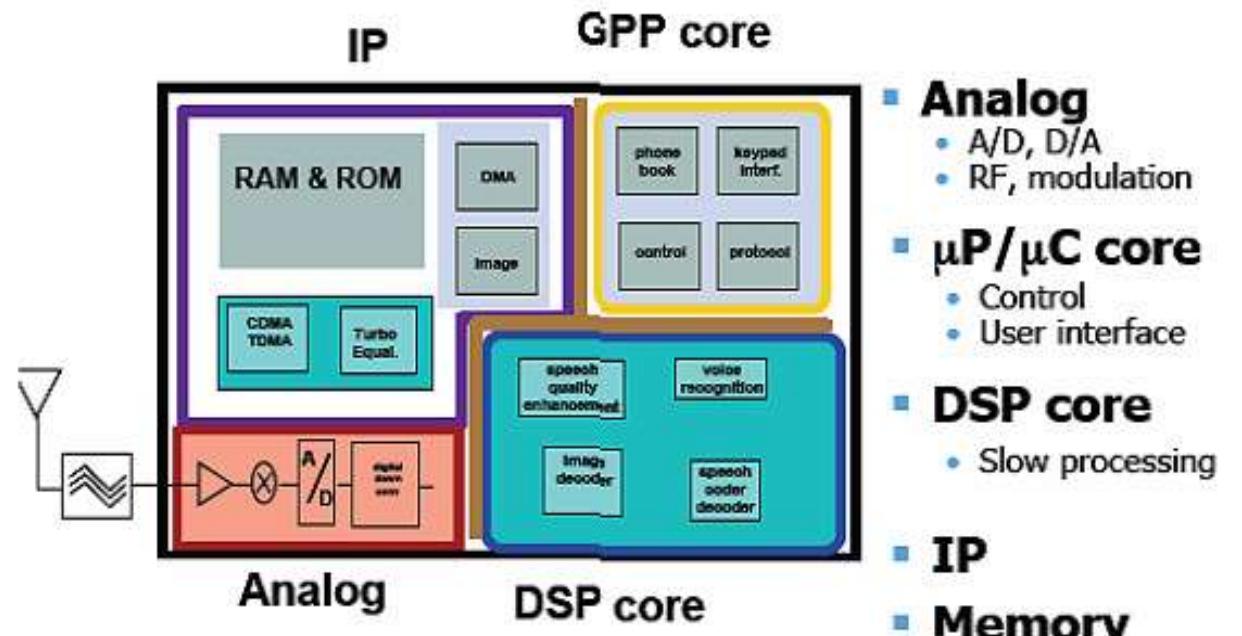
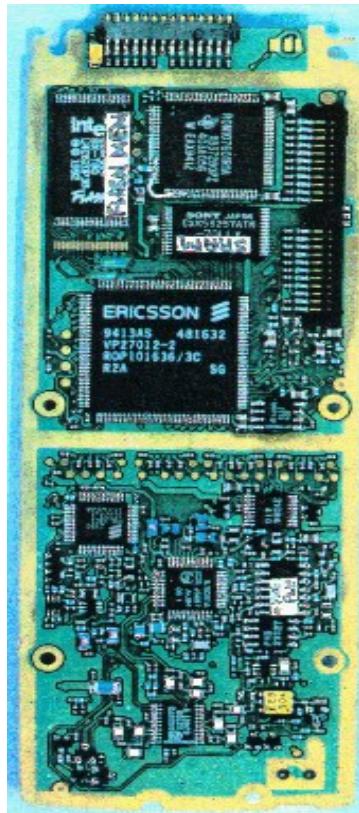
- DMAC (Direct Memory Access Controller)
 - Transferts mémoires/mémoires, mémoires/périphériques
 - Décharge le CPU (pas d'attente liée aux transferts)
- Timer, RTC (Real-Time Clock)
 - Mesure de l'écoulement du temps
 - Utilisations :
 - Contrôle du nb d'images par secondes
 - Programmation de délais d'expiration
 - Utilisation par OS Temps Réel
- Contrôleur d'interruptions (ITC)
 - Centralisation de tous les signaux d'interruptions
 - Informations sur l'émetteur de l'interruption

Composants des SoCs:

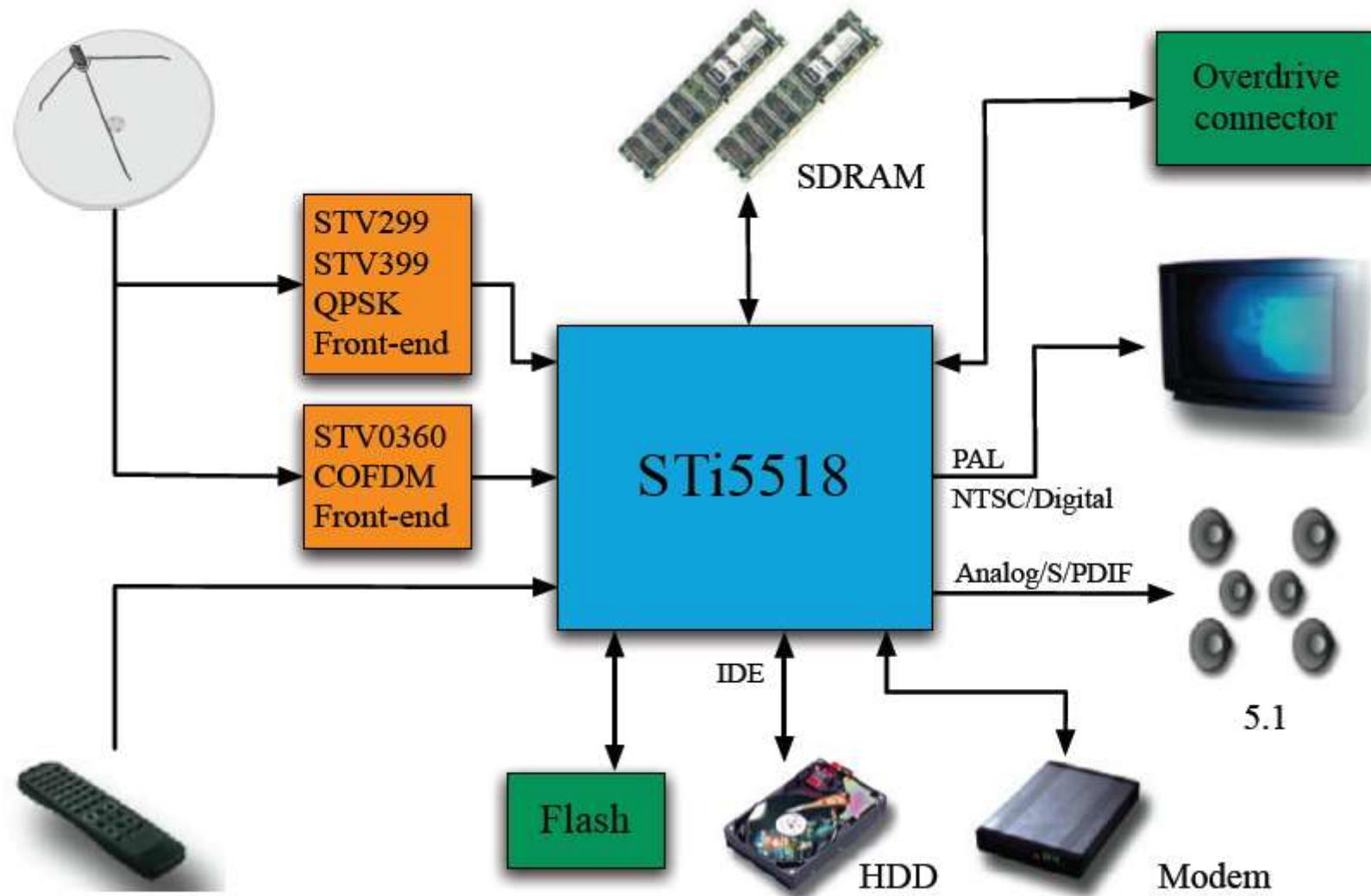
4) Entrées/sorties

- GPIO (General Purpose Inputs/Outputs)
 - programmation/lectures de broches du circuit
 - Utilisation : lectures de boutons, clavier simple
- Composants ports série
 - UART (Universal Asynchronous Receiver/Transmitter) : port RS232
 - SSP (Synchronous Serial Port) : port série haute vitesse
 - Utilisation : branchement de composants externes, debuggage
- Pilotage de bus de périphériques
 - IDE(Integrated Drive Electronics) /ATA... : Disque dur interne, DVD
 - USB
 - IEEE 1394 aka FireWire (camescopes numériques, Liaisons haute vitesse...)
- Composants RF: Bluetooth, GPS,GSM...

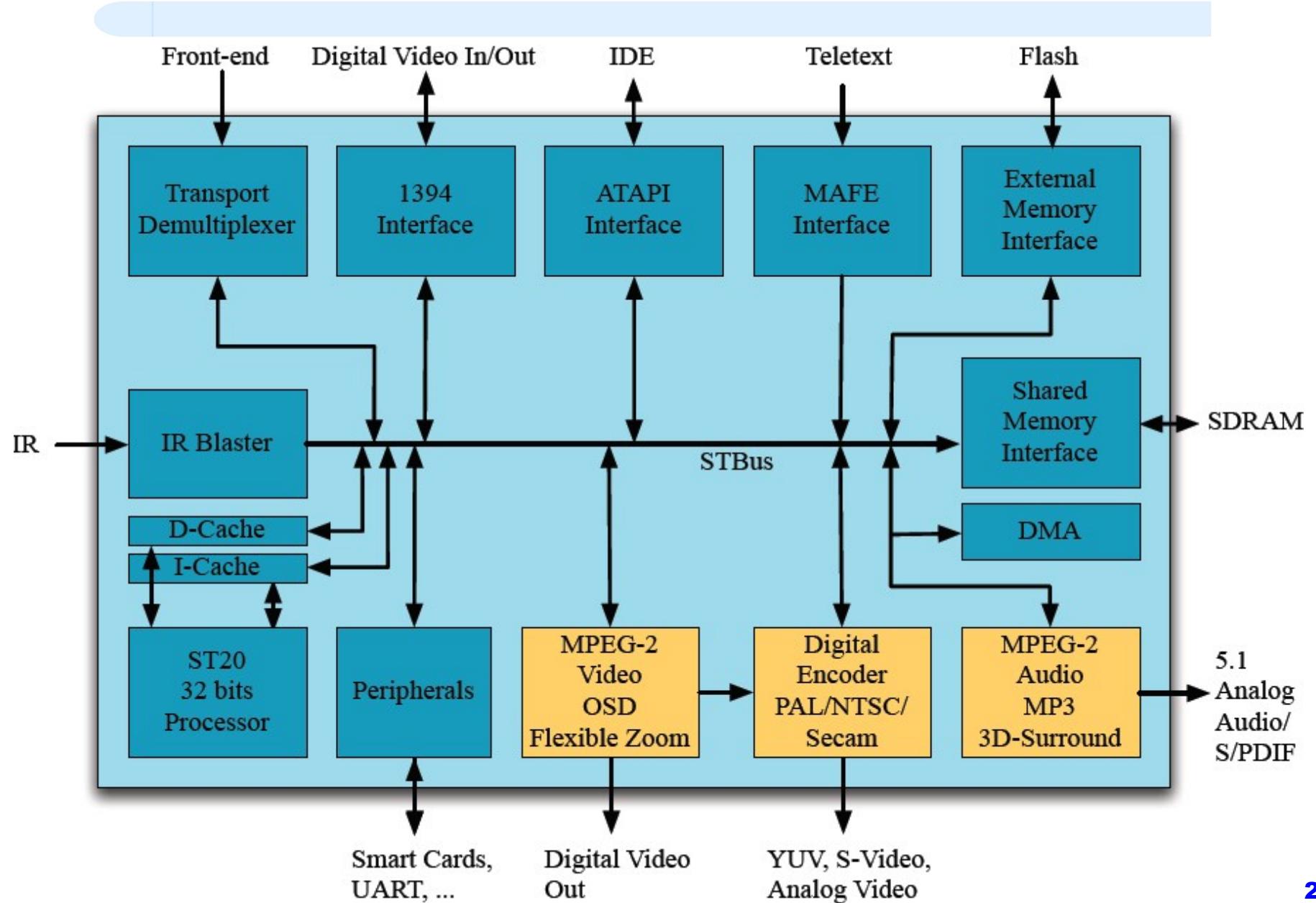
Exemple de SOC pour le Terminal 2G



Exemple de SOC: STi5518 pour Décodeur Satellite CDVB2300B

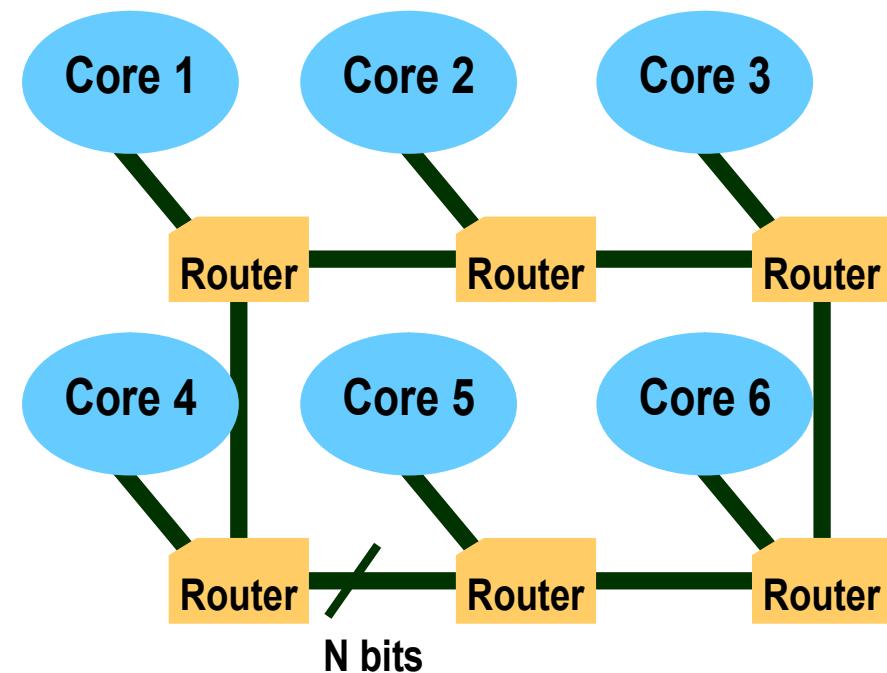
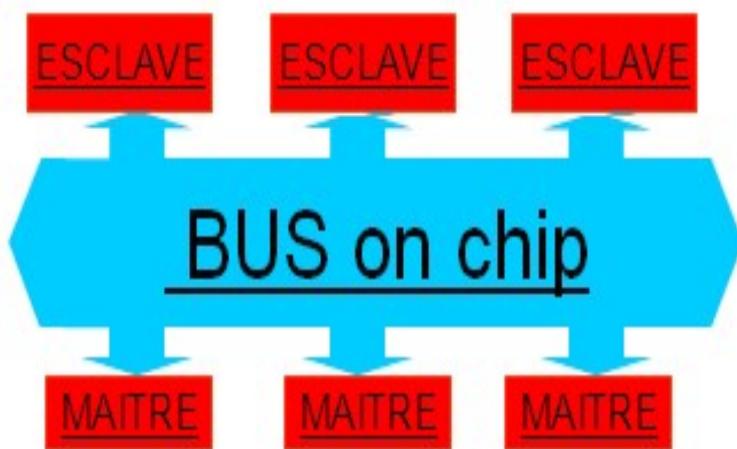


Exemple de SOC: STi5518



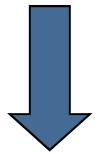
Interconnexions dans les SOCs

Network On Chip



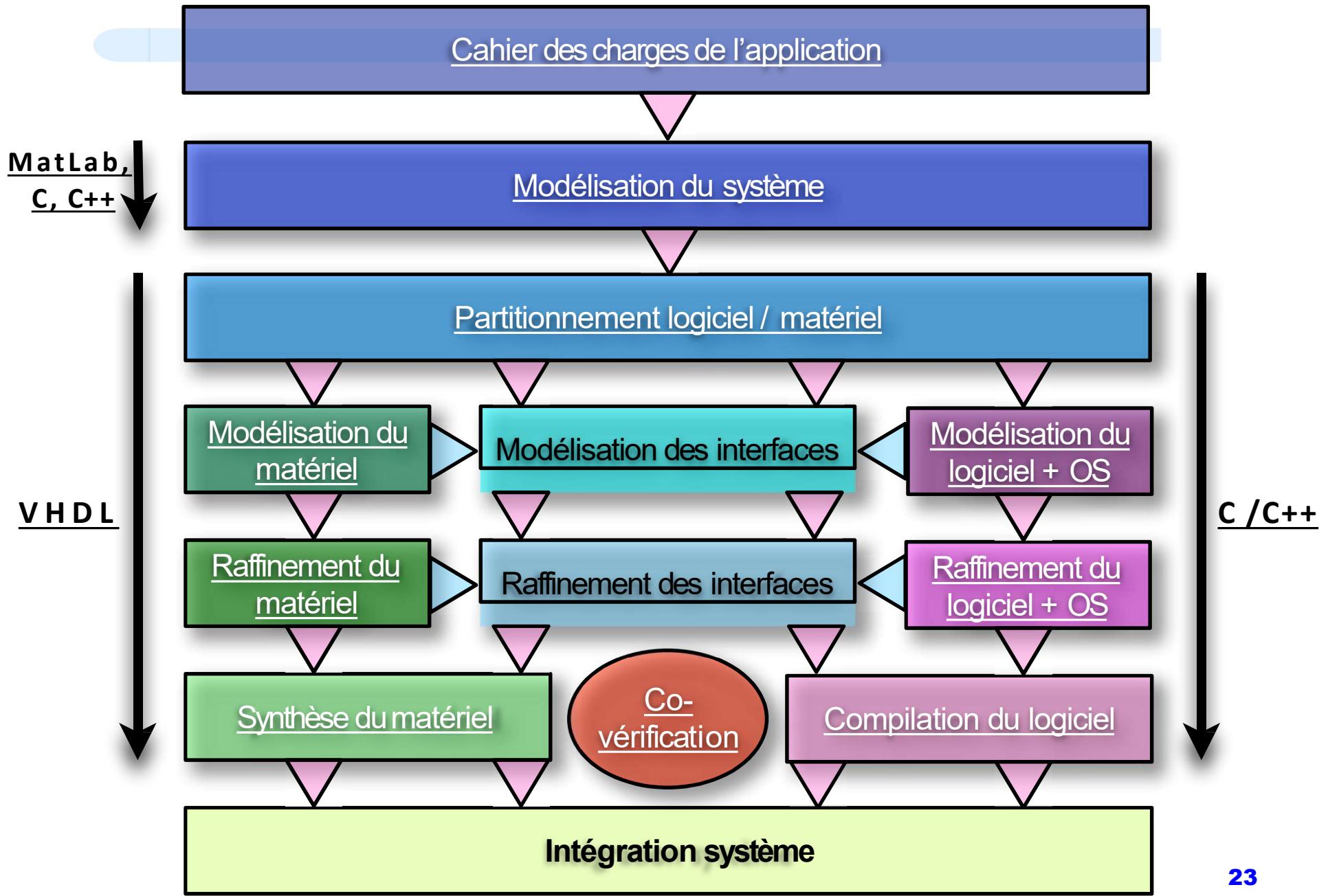
Besoins pour la conception des SoCs

- Besoin de modéliser le Hw à un haut niveau d'abstraction: plus le niveau de conception est haut \leftrightarrow le temps de conception est moindre
- Besoin de modéliser /exécuter et simuler Hw et Sw en même temps: Déconnection entre Hw/Sw engendre des erreurs et des coûts de développements très élevés
- Besoin d'utiliser les IP et le « Design Reuse »
 - Accélérer le développement des circuits
 - Réduire le Time To Market (TTM)
- Besoin de simplifier le test et la validation du système

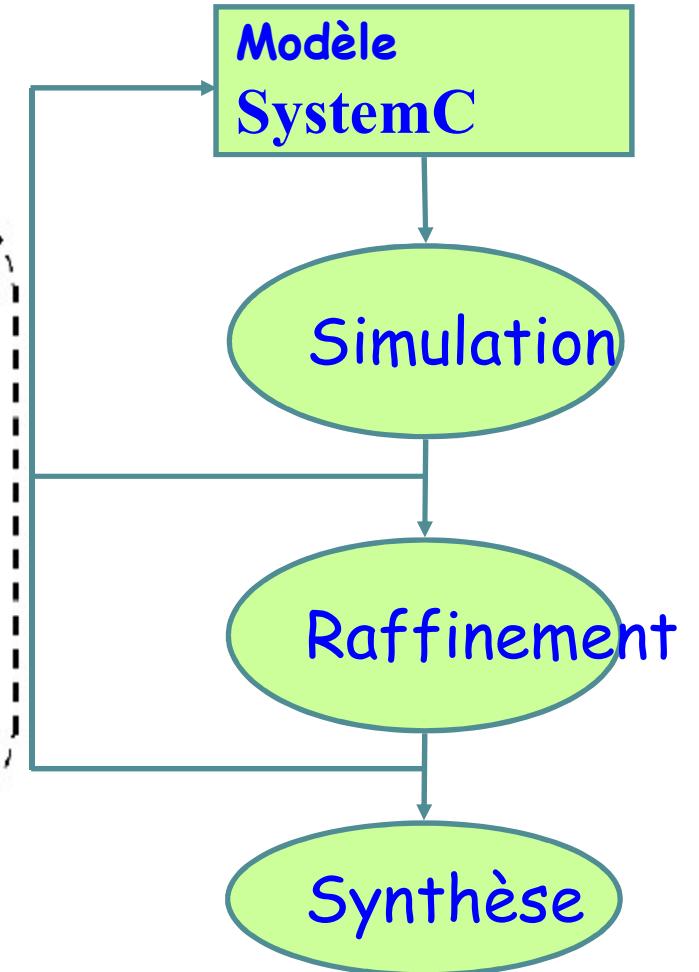
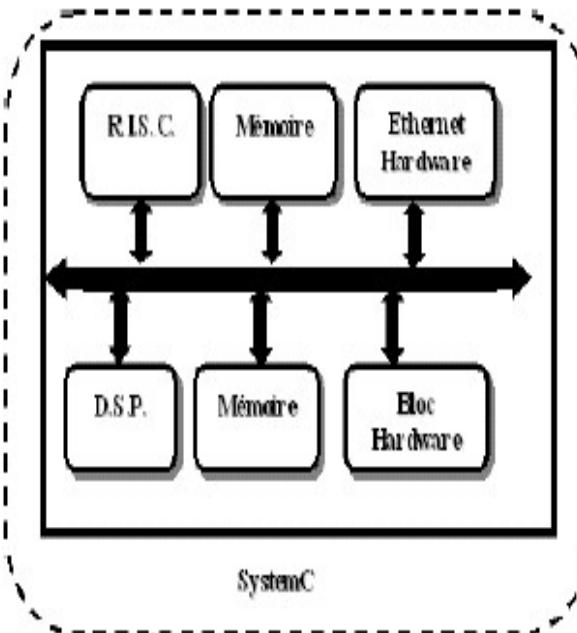
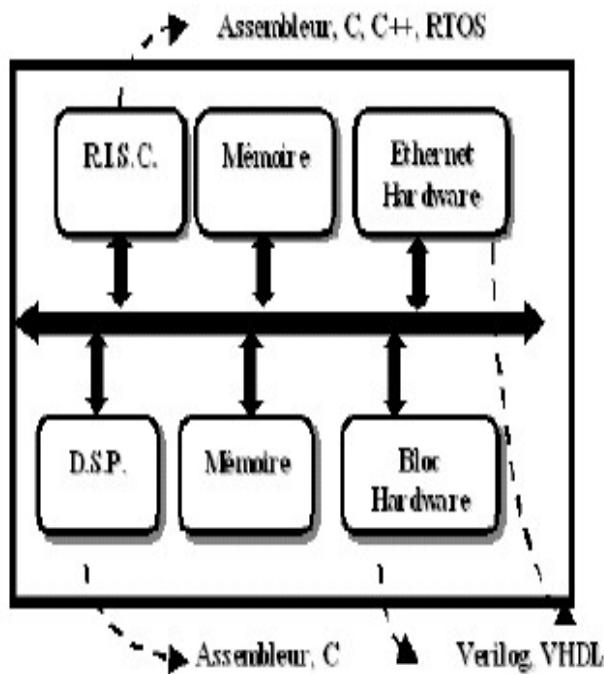


**Co-conception Hw/Sw
Co-design**

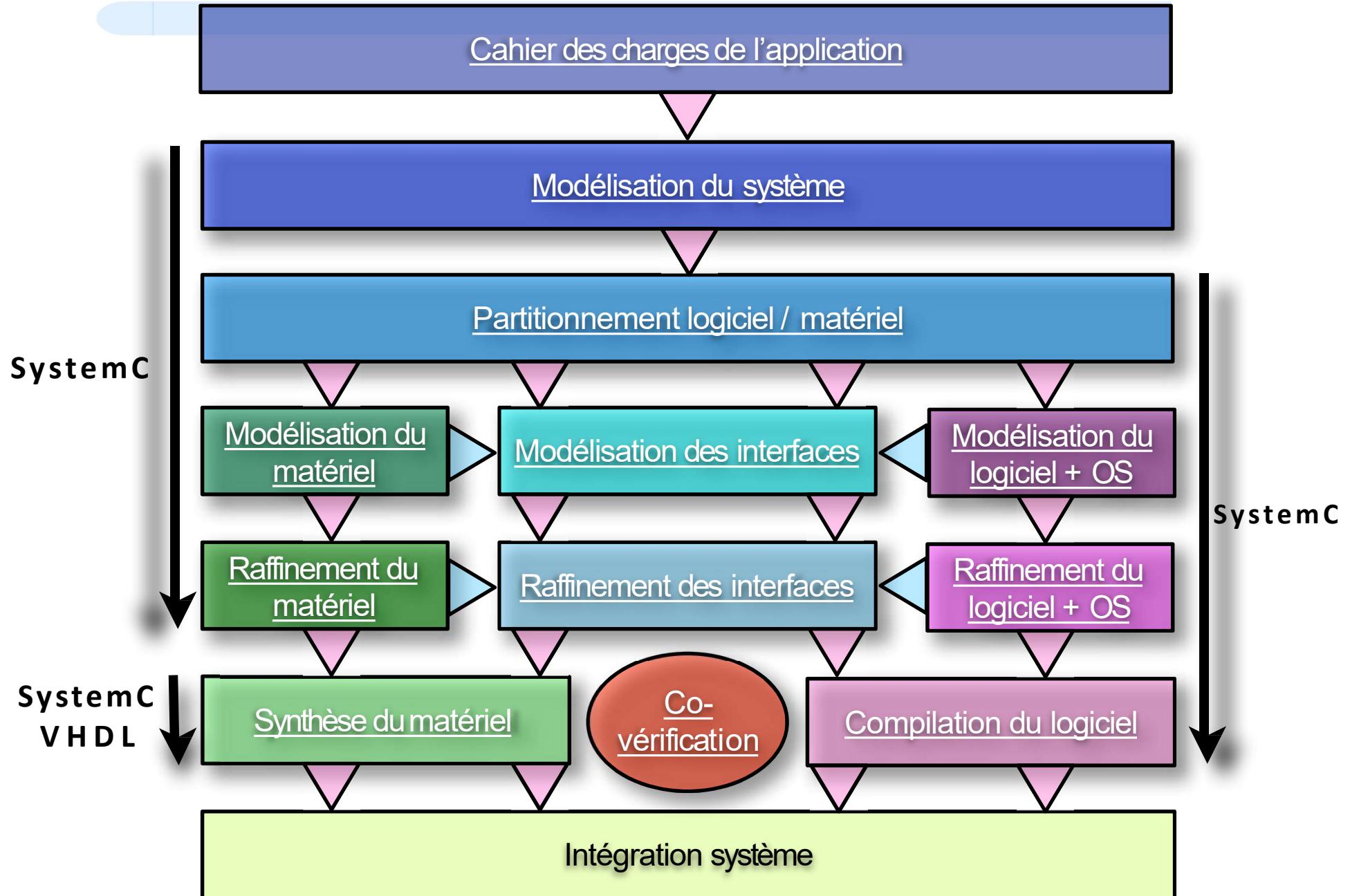
Flot de conception système usuel



SystemC: la solution aux Problèmes du codesign



Flot de conception avec SystemC



Plan du cours

- I. Introduction – Contexte & Défis
- II. SystemC: Modélisation conjointe Hw/Sw des SOCs
 - Introduction à SystemC
 - Elements du language
 - Notion de Concurrence
 - Flot SLD & niveaux d'abstraction
- III. Communication entre IPs dans les SOCs
 - Bus sur puce (BoC)
 - Réseaux sur puce (NoC)
 - Protocole OCP
- IV. Modélisation au niveau transaction en SystemC
 - Canaux prédéfinis
 - Canaux personnalisés
 - Modélisation TLM