

Architecture des SoC

Stéphane Mancini



ENSIMAG, 3A SLE

Architecture des SoC

Introduction

S. Mancini

Plan Détailé

✖ Architecture des SoC

- ✖ Evolution de la complexité des systèmes numériques

- ❑ Constituants des SoC

- ✖ Qu'est ce qu'un SoC ?

- ✖ Constituants des SoC

- ✖ Interaction IP/I/O et CPU

- ❑ Exemple de plateforme : mobile

- ✖ Architecture générale ML2000 (Mobilink)

- ✖ Détails

- ✖ Blocs fonctionnels

- ✖ Modules "système"

- ❑ Exemple de plateforme : multimedia

- ✖ Viper Set-top-box (PNX8500)

- ✖ Cell

- ❑ Critères d'optimisations

✖ Exemple : optimisation de la consommation

- ✖ Expression de la consommation
- ✖ Réduction de la consommation
- ✖ L'espace des critères de conception

✖ Architectures du futur

- ✖ Les principaux enjeux
- ✖ Puissance de calcul
- ✖ Infrastructure de communication
- ✖ Memory Wall
- ✖ IBM Power 4 dual-core : 115 W

✖ Méthodologie

- ✖ Quelques termes
- ✖ A quoi sert un modèle ?
- ✖ Spécification exécutive
- ✖ Modèles de spécification

- ☆ Top/down ou bottom/up ?
 - ☆ La méthodologie top/down
 - ☆ La boucle Adéquation Algorithme Architecture
 - ☆ Problématique
 - ☆ Flot de conception numérique
 - ☆ Le codesign (ideal !)
-
- ☆ La synthèse d'architecture
 - ☆ La synthèse d'architecture (bis)
 - ✖ En résumé
 - ☆ Concevoir un SoC ?
 - ☆ Le cours "Architecture des SoC"

Architecture des SoC

Infrastructure de communication

S. Mancini

Plan Détailé

- ✖ **Bus système**
 - ★ Objectifs
 - ★ Problématique
 - ★ Physiologie
 - ★ Exemple : communication CPU/IP
 - ★ Exemple : communication CPU/IP
- ✖ **Eléments d'un bus système**
 - Connectique
 - ★ Bus Trois Etats
 - ★ Bus "Cross bar"
 - Contrôleur de bus
 - ★ Contrôleur de bus
 - ★ Arbitrage
 - ★ Gestion des interblocages
 - ★ Exercice : arbitre de bus
- ✖ **Infrastructure de communication**
 - Topologie
 - ★ Espace d'adressage
 - ★ Hiérarchie de bus
 - DMA
 - ★ DMA : "Direct Memory Access"
 - ★ Caractéristiques d'un DMA
 - Schémas communs
 - ★ De CPU à CPU
 - ★ Contrôle d'un chemin de données
 - ★ Terminal
 - ★ Coopération
 - ★ Accès à une ressource commune

✖ Exemple de bus

- Coreconnect
- ★ Architecture
- ★ CoreConnect - Exemple
- CoreConnect - Contrôle
- PLB Signaux
- PLB Master
- PLB Slave
- PLB & pipeline de transferts
- Exercice : interface à une BRAM
- PLB read
- PLB Read
- PLB Write
- PLB Write
- PLB burst read
- PLB burst read
- PLB burst write
- PLB burst write
- AMBA
- ★ AMBA - Architecture
- ★ Signaux
- ★ Connectique AHB
- ★ Contrôle
- ★ Exemple : AMBA AHB master
- ★ Exemple : AMBA AHB slave
- ★ Exemple : AMBA AHB protection

✖ Conclusion

- ★ Conclusion

Architecture des SoC

Architecture mémoire

S. Mancini

Plan Détailé

- ✖ **Introduction**
 - ★ Impact de l'architecture mémoire
 - Architecture des mémoires
 - ★ Architecture des mémoires
 - ★ Organisation en colonnes
 - ★ RAM multi-port
- ✖ **Technologie des points mémoire**
 - SRAM
 - ★ Technologie des SRAM
 - ★ Densité des SRAM
 - ★ Performances des SRAM
 - Famille (S)DRAM
 - ★ Technologie des DRAM et SDRAM
 - ★ Architecture des SDRAM
 - ★ Accès aux SDRAM
- ✖ **Hiérarchie mémoire**
 - Objectifs
 - ★ Objectifs d'une hiérarchie mémoire
 - ★ Principe d'une hiérarchie mémoire
- ★ Rafraîchissement des DRAM
- ★ Densités des SDRAM
- ★ Performance des SDRAM
- ★ En résumé
 - Autres points mémoire
 - ★ ROM
 - ★ FLASH
 - ★ Registre
 - Conclusion
 - ★ Types de mémoire
 - ★ Contrôleur mémoire

- ★ Types de mémoires
- Scratch-Pad RAM et BRAM
- ★ Scratch-Pad RAM
- ★ BRAM

★ Caches standards

- Objectifs
- ★ Stratégies d'allocation des blocs
- ★ Mémoire associative
- ★ Temps d'accès aux caches
- ★ Performance d'un cache
- ★ Politique de remplacement
- ★ Amélioration du taux de défaut de cache
- ★ Principaux paramètres d'efficacité
- ★ Exemple : les coeurs Intel
- En résumé

★ Méthodologie

- ★ Spécificité des mémoires dans les systèmes intégrés
- ★ Utilisation des mémoires

★ TD : estimation de mouvement

- ★ Principe de l'estimation de mouvement
- ★ Blanc !
- ★ Questions
- ★ Solution

Architecture des SoC

Etudes de cas- Multimédia

S. Mancini

Plan Détailé

- ✖ **Contexte**
 - ❑ Présentation
 - ★ Applications
 - ★ Particularités
 - ❑ Standards vidéo
 - ★ Format vidéo
 - ★ MPEG2
 - ❑ 3D
 - ★ Principe de la 3D Z-buffer
 - ★ Pipeline 3D
 - ★ Performances des accélérateurs 3D
- ✖ **Set top Box PNX8500**
 - ❑ Environnement
 - ★ Environnement
 - ★ Classe d'application
 - ❑ Architecture logicielle
 - ★ Systèmes d'exploitations
 - ★ Streaming Software
 - ❑ Architecture générale
 - ★ Les processeurs
 - ★ MIPS
 - ★ Trimédia- parties opératives
 - ★ Trimédia-Architecture mémoire
 - ★ Trimédia-Mécanismes systèmes
 - ❑ Bus, mémoire et 3D
 - ❑ Les bus
 - ★ Architecture des Bus
 - ★ Architecture Mémoire
 - ★ Architecture “Système”
 - ❑ Partitionnement de la 3D
 - ★ Vidéo

- ★ Données standardisées gérées par TSSA
- ★ Exemple TSSA simple
- ★ Description de l'application
- Exemple d'application
 - ★ Objectifs
 - ★ Organisation de l'application
 - ★ Organisation du contrôle
 - ★ Déroulement de l'application
 - Le circuit
 - ★ Le circuit
- ✖ Set top box IBM STB03xxxx
 - ★ Set top box IBM STB03xxx
 - ★ Set top box IBM STB04xxx
 - ★ Architecture mémoire
- ✖ Plateforme multimédia “portable”
 - ★ Nomadik vidéo portable
- ✖ Playstation 2 & 3
 - PS2 : contexte
 - ★ Architecture générale
 - ★ Performances
 - EE : architecture matérielle
 - ★ Vue d'ensemble
 - ★ Processeur principal
 - ★ Architecture de VU1 (et VU0)
 - ★ Architecture mémoire
 - ★ Partitionnement de la 3D
 - EE : architecture logicielle
 - ★ Modèle de programmation
 - ★ Flot de données
 - EE : le circuit
 - ★ Le circuit
 - PS3 & Cell
 - ★ Architecture de la PS3
 - ★ Cell : schéma bloc
 - ★ Cell : EIB
 - ★ Cell : performance DMA
 - ★ Cell : le circuit
 - ✖ GPU
 - ★ Objectifs des (GP)-GPU
 - ★ GeForce 8, 9 et GT200 (Tesla) : Architecture globale
 - ★ GeForce Tesla : Architecture TPC
 - ★ GeForce Tesla : Architecture SP
 - ★ GeForce Tesla : Circuit
 - ★ Fermi : Architecture globale
 - ★ Fermi : Architecture SM
 - ★ Fermi : Circuit

Architecture des SoC

Etudes de cas- Téléphonie mobile

S. Mancini

Plan détaillé

- ✖ **Contexte**
 - ✖ Evolution de la téléphonie mobile
 - ✖ Performances
 - ✖ Puissances de calcul
- ✖ **GSM**
 - ❑ Présentation
 - ✖ Principe général
 - ✖ Partitionnement type
 - ❑ ML 2000
 - ✖ Architecture générale ML2000 (Mobilink)
 - ✖ Détails
 - ✖ Blocs fonctionnels
 - ✖ Modules “système”
 - ✖ Version “Multimédia” et GPRS de ML2000
 - ❑ Plateformes mobiles multimédia
 - ✖ Exemple : nomadik
 - ✖ Plateforme GPRS multimédia : détail des bus
 - ❑ Conclusion
- ✖ **Conclusion**
- ✖ **WCDMA**
 - ❑ Complexité
 - ✖ Objectifs
 - ✖ Modulation CDMA
 - ✖ Fonctionnalités d'un terminal 3G
 - ✖ Partitionnement Système
 - ❑ Architecture du récepteur
 - ✖ Récepteur mono-utilisateur
 - ✖ Complexité des récepteurs mobiles
 - ✖ Partitionnement logiciel/matériel
 - ✖ Partitionnement Matériel/Logiciel de M-gold
 - ✖ Plateforme UMTS M-gold
 - ✖ Le circuit
- ✖ **Station de base**
 - ✖ Objectifs
 - ✖ Exemple d'architecture
 - ✖ Perspectives : SDR (Software Defined Radio)

Architecture des SoC

Etudes de cas- Network Processor

S. Mancini

Plan Détailé

✖ Contexte

- ✖ Architecture des réseaux
- ✖ Couches OSI
- ✖ Performances types
- ✖ Objectifs des network-processor
- ✖ Classification des Network Processor

✖ IXP1200

- Introduction
 - ✖ Environnement
 - ✖ Vue d'ensemble
- Modules
 - ✖ StrongArm
 - ✖ Micro Engine
 - ✖ Micro Engine registres et accès mémoires
 - ✖ Architecture des Micro Engines
- Architecture logicielle
 - ✖ Architecture logicielle
 - ✖ Architecture logicielle
 - ✖ Architecture logicielle de l'ARM
- IXP2800

✖ Module FBI

- ✖ Le contrôleur de SRAM
- ✖ Gestion des listes chaînées
- ✖ Le contrôleur SDRAM
- Architecture mémoire et bus
 - ✖ Architecture mémoire et bus
 - ✖ Carte mémoire (extrait)
- Communication inter-modules
 - ✖ Micro-Engines - Micro-Engines
 - ✖ Micro-Engine - ARM
 - ✖ Micro-Engines - Accélérateur
 - ✖ ARM - Accélérateur

- ★ Architecture générale
- ★ Micro Engine
- Conclusion
- ★ Conclusion
- ✖ **C-Port**
 - Présentation
 - ★ Vue d'ensemble
 - ★ Exemple de flux de données
 - Processseurs et accélérateurs
 - ★ Processeur de canal
 - ★ Executive Processor
 - ★ Table Lookup Unit
 - ★ Queue Management Unit
 - ★ Buffer Management Unit
 - ★ Fabric Processor
 - Les Bus
- ★ Topologie
- ★ Noeuds du Ring Bus
- ★ Topologie du Ring Bus
- Logiciel
 - ★ Architecture Logicielle
- Résumé
 - ★ Conclusion
 - ★ Le circuit
- ✖ **IBM NP4GS3**
 - ★ IBM NP4GS3
 - ★ Flot de données
 - ★ Architecture d'un DPPU
- ✖ **Conclusion**
 - ★ Conclusion

Architecture des SoC

Introduction

S. Mancini



Qui sommes-nous ?

Stéphane MANCINI
Maître de Conférence
Enseignement : ENSIMAG
Recherche : TIMA
stephane.mancini@inpg.fr

Frédéric PETROT
Professeur
Enseignement : ENSIMAG
Recherche : TIMA
frederic.petrot@inpg.fr
www.tima.inpg.fr

1- Architecture des SoC- Introduction

S. Mancini 

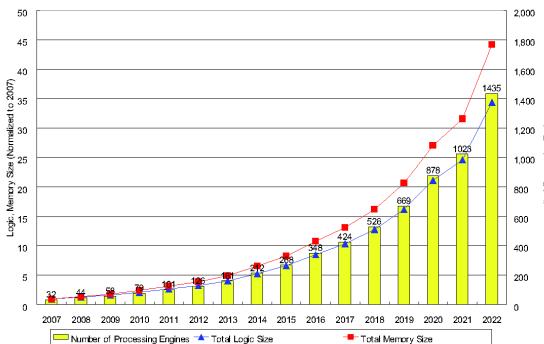
Plan

Architecture des SoC

- Constituants des SoC
- Exemple de plateforme : mobile
- Exemple de plateforme : multimedia
- Critères d'optimisations
- Architectures du futur
- Méthodologie
- En résumé

2- Architecture des SoC- Introduction

Evolution de la complexité des systèmes numériques



3- Architecture des SoC- Introduction
Architecture des SoC

S. Mancini 

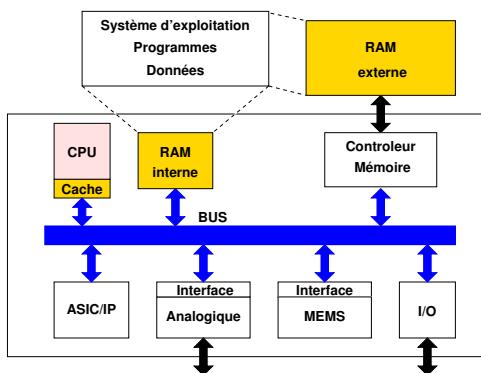
Plan

Architecture des SoC

- Constituants des SoC
- Exemple de plateforme : mobile
- Exemple de plateforme : multimedia
- Critères d'optimisations
- Architectures du futur
- Méthodologie
- En résumé

4- Architecture des SoC- Introduction

Qu'est ce qu'un SoC ?



5- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini 

Constituants des SoC



Les **bus** **système** permettent aux différents blocs de s'échanger des données.

Un bus système est caractérisé par :

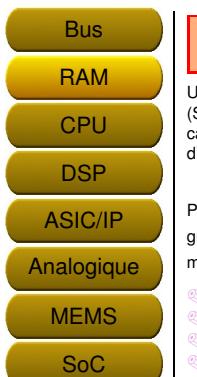
- Son débit
- Son protocole

Il existe plusieurs compromis entre la complexité du bus et ses performances.

6- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini 

Constituants des SoC



Les mémoires servent à stocker des programmes/données.

Un SoC est constitué de différents types de mémoire (SRAM, DRAM, FFD). Chacune de ces technologies a des caractéristiques (surface par bit, consommation, méthode d'accès et débit) spécifiques.

Pour optimiser le système, les programmes/données sont répartis et éventuellement dupliqués dans différents types de mémoire.

Une **hiérarchie mémoire** est un ensemble de mémoires de différents types et de mécanismes de gestion plus ou moins automatiques.



Adapté aux calculs irréguliers
Souplesse de programmation

Puissance de calcul faible

Utilisation {
 ★ Gestion du système
 ★ Interface utilisateur
 ★ Gestion temps réel
 }

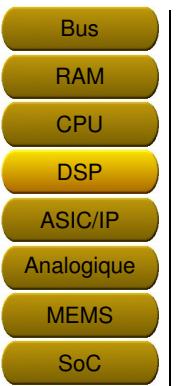
7- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini 

8- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini 

Constituants des SoC



Puissance de calcul moyenne
 Instructions et parties opératives spécifiques
 Programmation complexe

- Utilisation
- Calculs réguliers
 - Contrôle temps réel
 - Traitement de signal (parole, images simples, ...)

9- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini

Constituants des SoC



Les ASICs sont utilisés pour leur grande puissance de calcul.

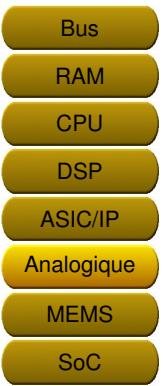
Grande puissance de calcul
 Traitements réguliers
 Consommation réduite
 Rigidité
 Temps de développement
 Peu adapté à un contrôle complexe

La souplesse des ASICs est obtenue en prévoyant de pouvoir les paramétriser.

10- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini

Constituants des SoC



Les composants analogiques sont principalement utilisés pour :

- La conversion analogique/numérique
- Le filtrage (RF)
- La mise en forme de signaux en provenance des capteurs

Très grande intégration
 Faible consommation
 Bruit
 Conception complexe
 Grande dépendance à la technologie

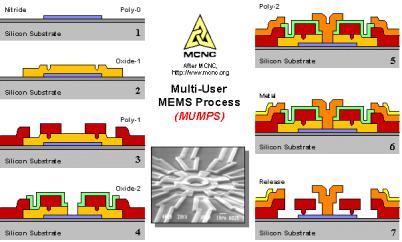
11- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini

Constituants des SoC



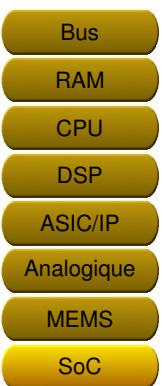
Les MEMS (**Micro-Electro-Mechanical Systems**) et MOEMS (**Micro-Opto-Electro-Mechanical Systems**) sont des capteurs et actuateurs intégrés.



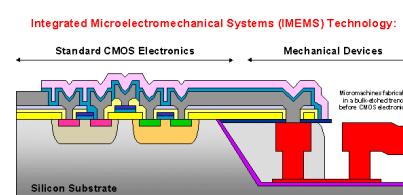
12- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini

Constituants des SoC



Et si on mettait tout ensemble ?



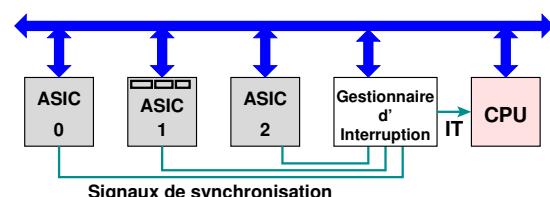
13- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

S. Mancini

Interaction IP/IO et CPU

L'insertion d'un ASIC dans un système nécessite :

- Son adaptation au bus système
- Accès aux registres de configuration
- Gestion des transferts de données
- La mise en oeuvre de mécanismes de synchronisation



14- Architecture des SoC- Introduction
Architecture des SoC- Constituants des SoC

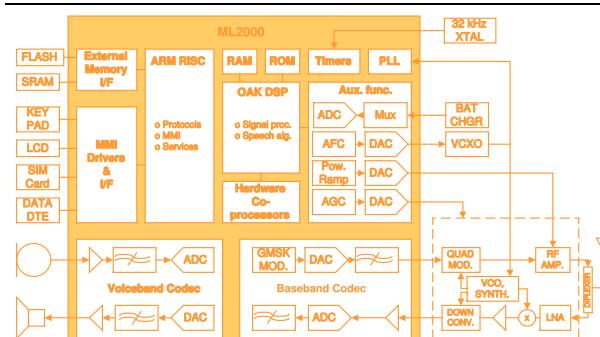
S. Mancini

Plan

- Architecture des SoC**
 - Constituants des SoC
 - Exemple de plateforme : mobile
 - Exemple de plateforme : multimedia
 - Critères d'optimisations
- Architectures du futur**
- Méthodologie**
- En résumé**

15- Architecture des SoC- Introduction

Architecture générale ML2000 (Mobilink)

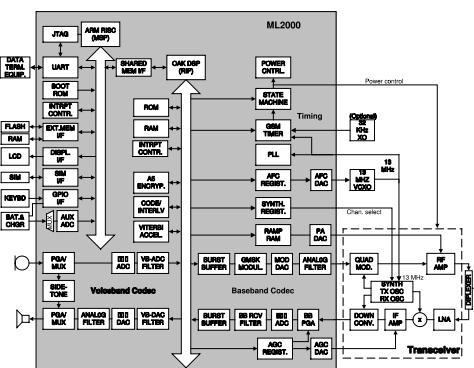


Processeurs à 52 MHz

16- Architecture des SoC- Introduction
Architecture des SoC- Exemple de plateforme : mobile

S. Mancini

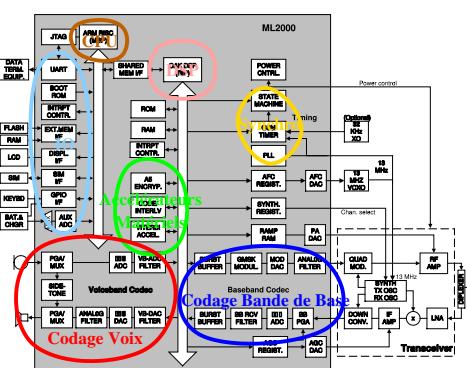
Détails



17- Architecture des SoC- Introduction
Architecture des SoC- Exemple de plateforme : mobile

S. Mancini

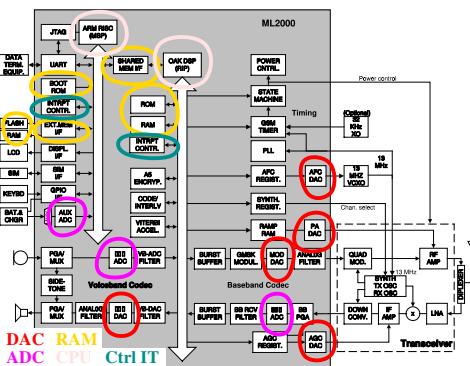
Blocs fonctionnels



18- Architecture des SoC- Introduction
Architecture des SoC- Exemple de plateforme : mobile

S. Mancini

Modules "système"



19- Architecture des SoC- Introduction
Architecture des SoC- Exemple de plateforme : mobile

S. Mancini

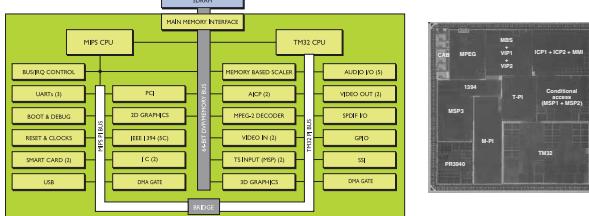
Plan

Architecture des SoC

- Constituants des SoC
- Exemple de plateforme : mobile
- Exemple de plateforme : multimedia
- Critères d'optimisations
- Architectures du futur
- Méthodologie
- En résumé

Viper Set-top-box (PNX8500)

Le SoC PNX8500 (Philips) est dédié aux applications multimédias telles que les récepteurs de télévision numérique (Set-Top Box).

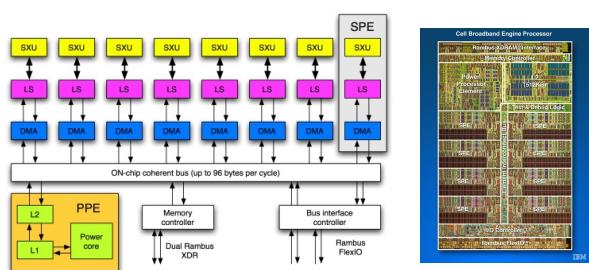


21- Architecture des SoC- Introduction
Architecture des SoC- Exemple de plateforme : multimedia

S. Mancini

Cell

Le Cell est le processeur au cœur de la playstation III



22- Architecture des SoC- Introduction
Architecture des SoC- Exemple de plateforme : multimedia

S. Mancini

Plan

- Architecture des SoC
 - Constituants des SoC
 - Exemple de plateforme : mobile
 - Exemple de plateforme : multimedia
 - Critères d'optimisations
- Architectures du futur
- Méthodologie
- En résumé

Exemple : optimisation de la consommation

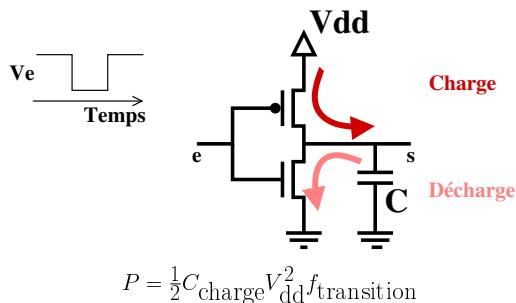
Comment réduire la consommation d'un SoC ??
Autrement dit, quelles sont les méthodes dont dispose un concepteur ?
Quels sont les niveaux d'intervention et libertés autorisées ?

23- Architecture des SoC- Introduction

24- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Expression de la consommation



Autres sources de consommation :

- ★ Courant de court-circuit
- ★ Fuites de courant

25- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Réduction de la consommation



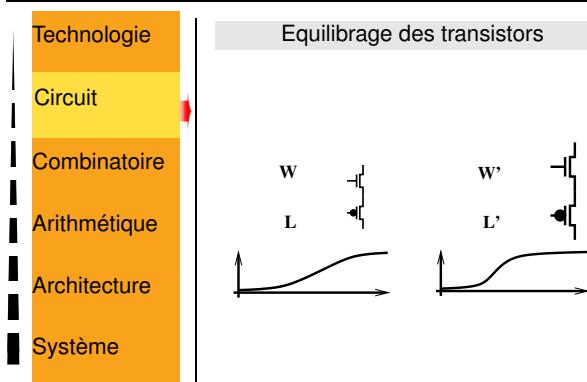
Réduction des capacités

- ★ Réduction de la taille des transistors
- ★ Amélioration de la technologie
- ★ Accroissement des investissements

26- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Réduction de la consommation



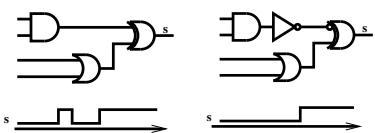
27- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Réduction de la consommation



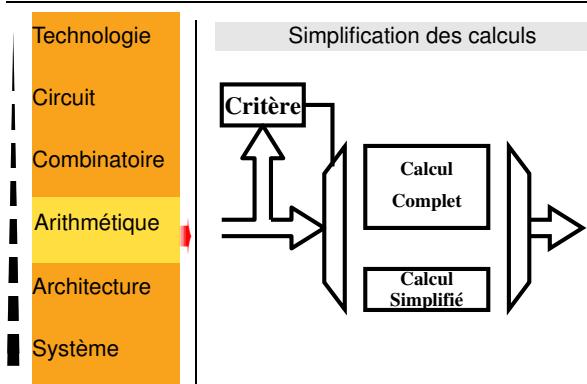
Equilibrage des temps de propagation



28- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Réduction de la consommation



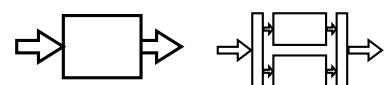
29- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Réduction de la consommation



Parallélisation

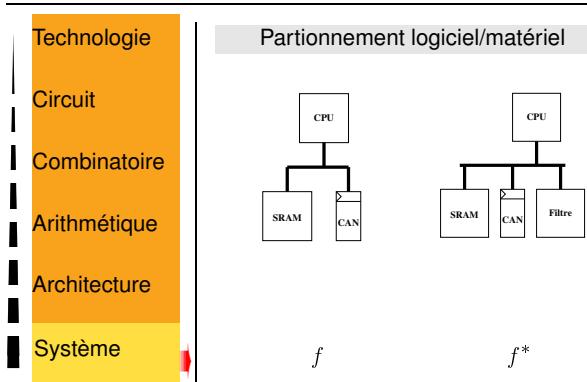


f, V_{dd}^*

$\frac{f}{2}, V_{dd}$

S. Mancini

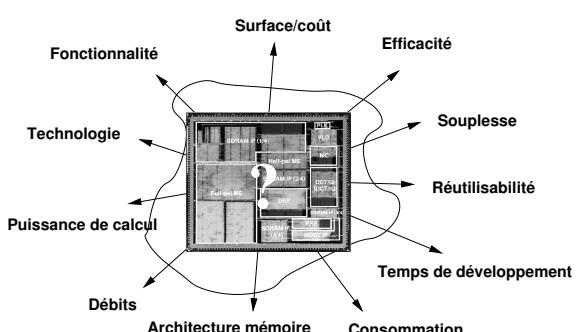
Réduction de la consommation



31- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

L'espace des critères de conception



32- Architecture des SoC- Introduction
Architecture des SoC- Critères d'optimisations

S. Mancini

Plan

- Architecture des SoC
- Architectures du futur
- Méthodologie
- En résumé

33- Architecture des SoC- Introduction

S. Mancini

Puissance de calcul

Densité d'intégration :

- ★ 90 nm mixte : 430 KPortes/mm²
- ★ 40 nm : 1 600 KPortes/mm²

Un CPU RISC 32 bits simple (MIPS) \approx 30 000 portes !

Loi de Moore : Le nombre de transistors disponibles double tous les 2 ans !

La surface des circuits reste globalement constante !



35- Architecture des SoC- Introduction
Architectures du futur

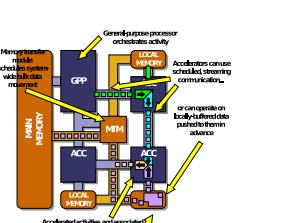
S. Mancini

Memory Wall

La puissance de calcul croît plus vite que les performances des mémoires externes !

- ★ Le débit s'accroît : DDR (2003)=0.4 Gbit/s, DDR3 (2008)=1.6 Gbit/s par bit de data
- ★ La latence reste constante : \approx 14 ns

Les flux de données doivent être finement gérés.
Des **hiérarchies mémoire** efficaces sont nécessaires !



37- Architecture des SoC- Introduction
Architectures du futur

S. Mancini

Plan

- Architecture des SoC
- Architectures du futur
- Méthodologie
- En résumé

39- Architecture des SoC- Introduction

S. Mancini

Les principaux enjeux

Accroître la puissance de calcul par le parallélisme !

La fréquence des coeurs n'augmentera quasiment plus : il faut paralléliser.

On prévoit plusieurs centaines de coeurs sur un circuit !

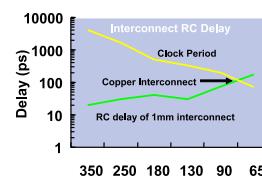
Il y a de nombreuses difficultés :

- Les applications doivent être parallélisables et réparties sur les coeurs
- L'infrastructure de communication doit être adaptée
- L'accès aux données devient un goulet d'étranglement (Memory Wall)
- Le système doit être tolérant aux fautes/pannes
- Il faut dissiper la chaleur !

34- Architecture des SoC- Introduction
Architectures du futur

S. Mancini

Infrastructure de communication



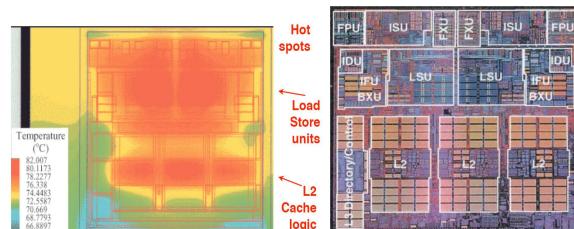
Les fils deviennent plus lents que les portes !

- ★ L'estimation des chaînes critiques devient complexes !
- Le placement routage est primordial
- ★ On ne peut plus faire de communications qui traversent un circuit !
- L'infrastructure de communication devient asynchrone !

36- Architecture des SoC- Introduction
Architectures du futur

S. Mancini

IBM Power 4 dual-core : 115 W



38- Architecture des SoC- Introduction
Architectures du futur

S. Mancini

Quelques termes

Méthode

technique de résolution de problème caractérisée par un ensemble de règles bien définies qui conduisent à une solution correcte.

Méthodologie

un ensemble structuré et cohérent de modèles, méthodes, guides et outils permettant de déduire la manière de résoudre un problème.

Modèle

une représentation d'un aspect partiel et cohérent du «monde» réel précède toute décision ou formulation d'une opinion est élaboré pour répondre à la question qui conduit au développement d'un système

40- Architecture des SoC- Introduction
Méthodologie

S. Mancini

A quoi sert un modèle ?

❑ Abstraction

Éliminer des détails, focaliser sur un point de vue du système. Travailler à différentes échelles de complexité et de temps.

❑ Analyse

Etude des propriétés du modèle (vérification de propriétés). Extrapolation au système réel représenté.

❑ Communication

Discussion et échanges avec d'autres personnes. Echanges entre outils.

❑ Génération/Production

Produire une représentation d'un autre niveau (autre modèle). Produire le système réel.



Le modèle à retenir est fonction de l'objectif visé !

41- Architecture des SoC- Introduction
Méthodologie

S. Mancini

Modèles de spécification

Modèle	Fonctionnalité	Temps	Surface	Consommation
C, Matlab	Exécutable			
ESTEREL	id	Synchrone		
SystemC	id	Synchrone		
VHDL cptl	id	Synchrone		
VHDL RTL	Comb.	Séquentiel		Activité
Netlist	Comb.	Séquentiel	Porte	Porte
Netlist annotée	Comb.	Séquentiel	Porte	Vraie
Layout			Vraie	
HSPICE	Electrique	Continu		Vraie

43- Architecture des SoC- Introduction
Méthodologie

S. Mancini

Spécification exécutive

Ce sont toutes les spécifications que l'on peut exécuter ou simuler.

★ Spécification algorithmique

- ❑ Purement algorithmique
Matlab, C (virgule flottante, virgule fixe)

- ❑ Temporel et algorithmique
ESTEREL, LUSTRE, COSSAP, SPW, SimuLink, VHDL comportementale

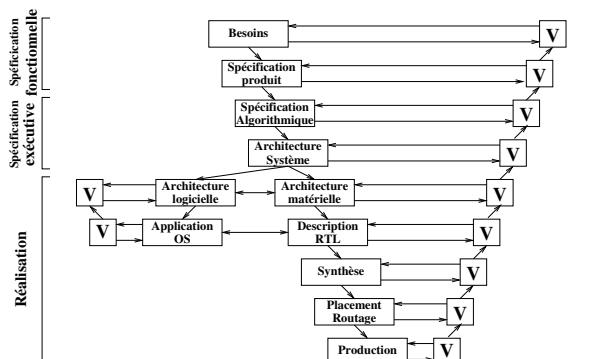
★ Spécification de l'architecture système

- ❑ Localisation des fonctions (processeur, opérateurs de calculs)
- ❑ Modèles de communication

42- Architecture des SoC- Introduction
Méthodologie

S. Mancini

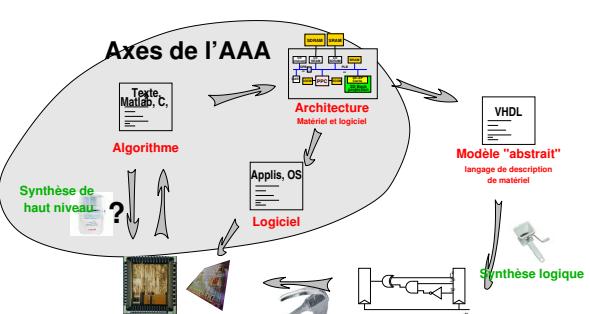
La méthodologie top/down



45- Architecture des SoC- Introduction
Méthodologie

S. Mancini

La boucle Adéquation Algorithme Architecture



46- Architecture des SoC- Introduction
Méthodologie

S. Mancini

Problématique

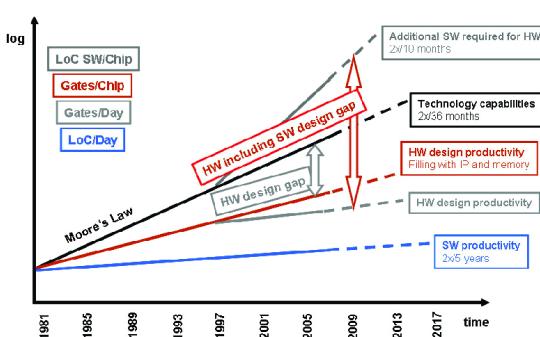
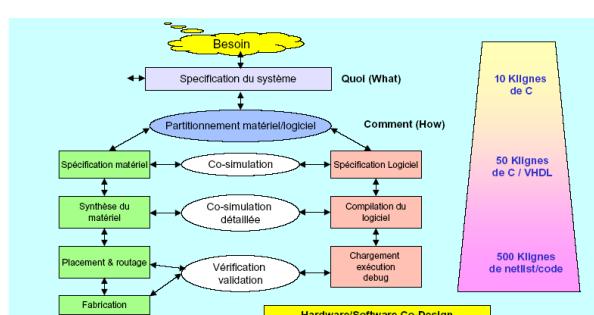


Figure DESN3 Hardware and Software Design Gaps Versus Time⁵

47- Architecture des SoC- Introduction
Méthodologie

S. Mancini

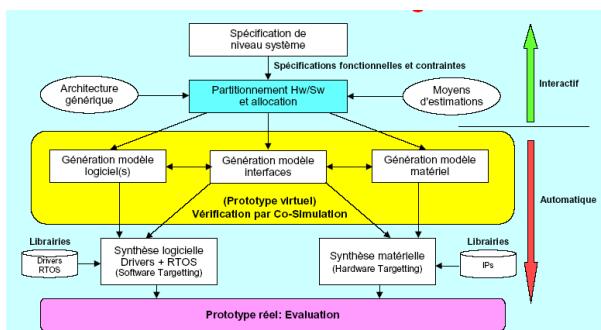
Flot de conception numérique



48- Architecture des SoC- Introduction
Méthodologie

S. Mancini

Le codesign (ideal !)



49- Architecture des SoC- Introduction
Méthodologie

S. Mancini

La synthèse d'architecture

La synthèse d'architecture RTL consiste à transformer un modèle séquentiel en un modèle concurrent, au niveau RTL.

La **synthèse de système** a pour objectif de passer d'un modèle logiciel à une architecture logicielle/matérielle parallèle.

Leur automatisation est difficile et n'est possible que dans certains contextes.

C'est l'avenir de la conception des systèmes embarqués !

50- Architecture des SoC- Introduction
Méthodologie

S. Mancini

La synthèse d'architecture (bis)

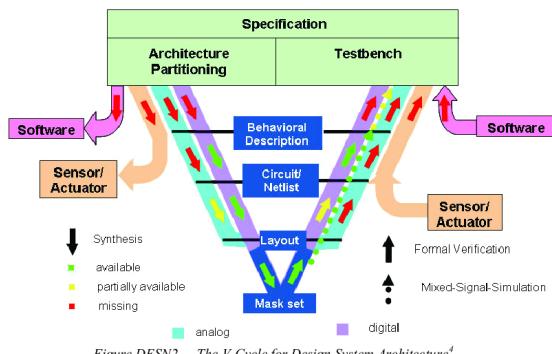


Figure DESN2 The V-Cycle for Design System Architecture⁴

51- Architecture des SoC- Introduction
Méthodologie

S. Mancini

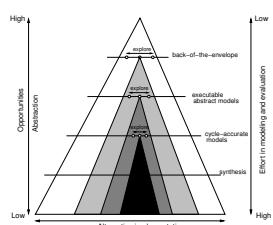
Plan

- Architecture des SoC
- Architectures du futur
- Méthodologie
- En résumé**

Concevoir un SoC ?

Il est nécessaire de prendre en compte les contraintes de l'électronique intégré dès le départ. Les contraintes matérielles influent directement sur les choix au plus haut niveau.

Le choix des algorithmes d'une application est déterminant !
Les optimisations ultérieures ne permettront pas de réduire la complexité intrinsèque.



53- Architecture des SoC- Introduction
En résumé

S. Mancini

Le cours "Architecture des SoC"

Stéphane Mancini

- Introduction
- Infrastructure de communication
- Hiérarchie mémoire
- Transformation de programme
- Etudes de cas et GPU

Frédéric Pétrot

- Architecture avancée des processeurs
- GALS et basse consommation
- Network On Chip
- Gestion des données dans les architectures parallèles
- OS SMP/AMP

... et un examen

54- Architecture des SoC- Introduction
En résumé

S. Mancini

Architecture des SoC

Introduction
S. Mancini

Plan Détailisé

- * Architecture des SoC**
 - Evolution de la complexité des systèmes numériques
 - Constituants des SoC
 - Qu'est ce qu'un SoC ?
 - Constituants des SoC
 - Interaction IP/I/O et CPU
 - Exemple de plateforme : mobile
 - Architecture générale ML2000 (Mobilink)
 - Détails
 - Blocs fonctionnels
 - Modules "système"
 - Exemple de plateforme : multimedia
 - Viper Set-top-box (PNX8500)
 - Cell
 - Critères d'optimisations
- * Exemples :**
 - Optimisation de la consommation
 - Expression de la consommation
 - Réduction de la consommation
 - L'espace des critères de conception
- * Architectures du futur**
 - Les principaux enjeux
 - Puissance de calcul
 - Infrastructure de communication
 - Memory Wall
 - IBM Power 4 dual-core : 115 W
- * Méthodologie**
 - Quelques termes
 - A quoi sert un modèle ?
 - Spécification exécutive
 - Modèles de spécification

- Top/down ou bottom/up ?
- La méthodologie top/down
- La boucle Adéquation Algorithme Architecture
- Problématique
- Flot de conception numérique

- En résumé
- Concevoir un SoC ?
- Le cours "Architecture des SoC"

Architecture des SoC

Infrastructure de communication

S. Mancini



Plan

- ✖ Bus système
- ❑ Eléments d'un bus système
- ❑ Infrastructure de communication
- ❑ Exemple de bus
- ❑ Conclusion

1- Architecture des SoC- Infrastructure de communication

Objectifs

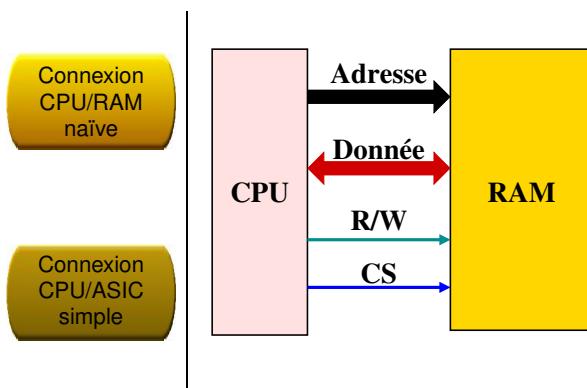
Les bus systèmes servent à assurer des transferts de données entre les modules

En général, les bus systèmes servent à établir des communications dont on **ne connaît pas à-priori** ni la topologie ni le séquencement exacts.

Ils ne garantissent qu'une **bande passante** maximum.

S. Mancini

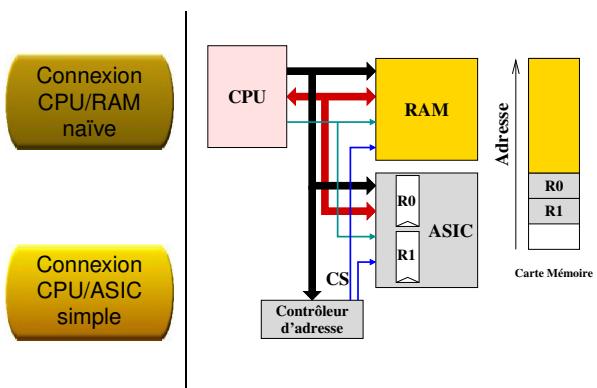
Problématique



3- Architecture des SoC- Infrastructure de communication
Bus système

S. Mancini

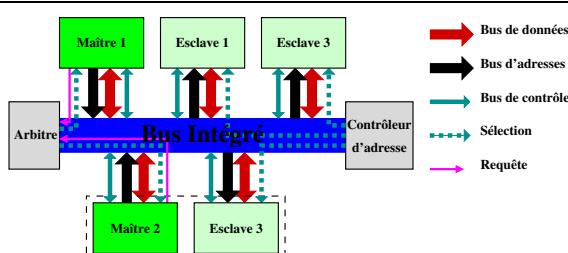
Problématique



4- Architecture des SoC- Infrastructure de communication
Bus système

S. Mancini

Physiologie



Caractéristiques d'un bus système :

- | | |
|---------------|------------------------------|
| ★ Connectique | ★ Caractéristiques physiques |
| ○ Données | ○ Protocole |
| ○ Adresses | |
| ○ Contrôle | |

5- Architecture des SoC- Infrastructure de communication
Bus système

S. Mancini

Exemple : communication CPU/IP

On souhaite faire communiquer un microprocesseur avec un module matériel (IP).

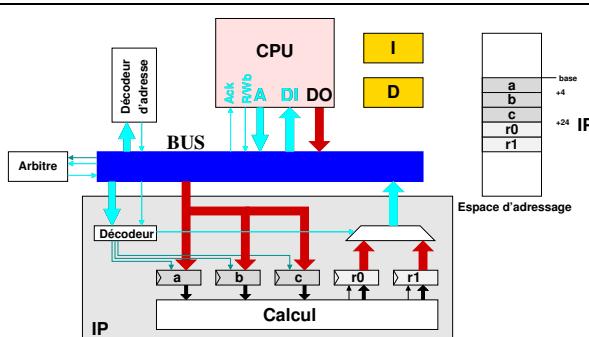
l'IP

- ❑ a besoin de différents paramètres pour réaliser le calcul
- ❑ envoie une interruption au CPU la fin des calculs
- ❑ stocke les résultats dans des registres

6- Architecture des SoC- Infrastructure de communication
Bus système

S. Mancini

Exemple : communication CPU/IP



7- Architecture des SoC- Infrastructure de communication
Bus système

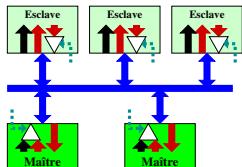
S. Mancini

Plan

- ❑ Bus système
- ✖ Eléments d'un bus système
 - ❑ Connectique
 - ❑ Contrôleur de bus
- ❑ Infrastructure de communication
- ❑ Exemple de bus
- ❑ Conclusion

8- Architecture des SoC- Infrastructure de communication

Bus Trois Etats



- Fréquence de fonctionnement faible
- Test complexe (buffer)
- Grande longueur des connexions
- Layout délicat (découplage)

9- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Connectique

S. Mancini

Contrôleur de bus

Le contrôleur de bus répond aux requêtes des maîtres. Il donne accès au bus à un des demandeurs selon :

- Une politique de priorité
- Une gestion temporelle du bus

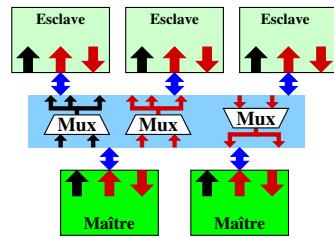
Les principaux critères d'efficacité sont :

- La garantie de bande passante
- Le temps d'attente d'accès maximum
- La préemptabilité

11- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Contrôleur de bus

S. Mancini

Bus "Cross bar"



- Fréquence élevée
- Accès simultanés
- Conçu avec des cellules standards
- Nombreux fils
- Layout complexe pour de nombreux maîtres
- Peu modulaire
- Difficile à observer

10- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Connectique

S. Mancini

Arbitrage

Les stratégies d'attribution du bus à un maître forment deux groupes :

- Les politiques statiques
 - Séquencement à-priori
 - Découpage temporel fixe
- Les politiques dynamiques
 - Priorités (statiques/dynamiques)
 - Tourniquet (Round-Robin)
 - Mixte

12- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Contrôleur de bus

S. Mancini

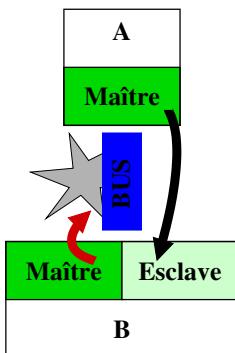
Gestion des interblocages

Il y a interblocage lorsqu'il y a :

- Demandes simultanées
- Attente circulaire
- Absence de préemption

Les solutions à l'interblocage sont :

- La prévention
 - Algorithme
 - Dimensionnement
 - Topologie
- La guérison
 - Détection
 - Préemption



13- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Contrôleur de bus

S. Mancini

Exercice : arbitre de bus

On s'intéresse à un système de m maîtres, numérotés de 0 à $m - 1$. Chacun des maîtres émet un signal de requête $r[0..m - 1]$ et reçoit un signal d'acquittement $a[0..m - 1]$. A tout instant, un seul des bits de a est valide. C'est celui dont le maître reçoit l'accès au bus.

Dans un premier temps, on s'intéresse à la conception du bloc `fix_prio`, qui implémente une priorité fixe, le maître n° 0 étant le plus prioritaire. `fix_prio` a en entrée un vecteur de bit $e[0..m - 1]$ et en sortie $s[0..m - 1]$. Seul le bit $s[j]$ vaut 1 si $e[j]$ est l'entrée valide la plus prioritaire.

- Dessinez le schéma porte d'un arbitre simple
- Comment le temps de propagation évolue-t-il ?
- Comment écrire ceci en VHDL le plus synthétiquement possible ?

14- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Contrôleur de bus

S. Mancini

Exercice : arbitre de bus

Afin d'éviter le phénomène de famine, les priorités sont maintenant tournantes. Pour rester simple, à chaque fois qu'une des requêtes est valide, les priorités sont toutes décalées d'un cran vers la gauche : $e[0]$ commence par la priorité maximum, puis prend la priorité de $e[m - 1]$, puis de $e[m - 2]$, etc ...

- En réutilisant `fix_prio`, donner un schéma du système
- Quelle est la nouvelle chaîne critique ?

Exercice : arbitre de bus

Pour ajouter de la souplesse dans la gestion des priorités, un maître peut maintenant demander une priorité complémentaire codée sur 3 bits, $p_j[2..0]$. La valeur de p_j est prioritaire sur la priorité usuelle (j étant le n° de maître). Si deux maîtres i et j ont des valeurs p_i et p_j identiques la résolution se fait d'après la priorité usuelle entre i et j .

- En réutilisant `fix_prio`, donner un schéma du système
- Quelle est la nouvelle chaîne critique ?
- Donnez un schéma du système avec priorité dynamique et tournante.

15- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Contrôleur de bus

S. Mancini

16- Architecture des SoC- Infrastructure de communication
Eléments d'un bus système- Contrôleur de bus

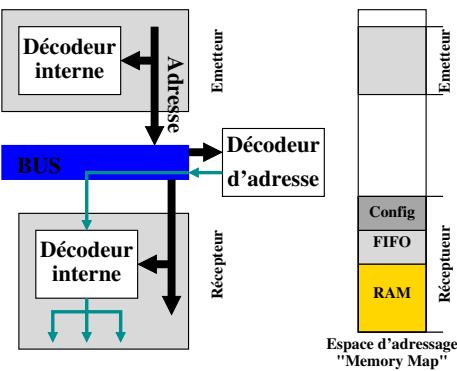
S. Mancini

Plan

- Bus système
- Eléments d'un bus système
- Infrastructure de communication
 - Topologie
 - DMA
 - Schémas communs
- Exemple de bus
- Conclusion

17- Architecture des SoC- Infrastructure de communication

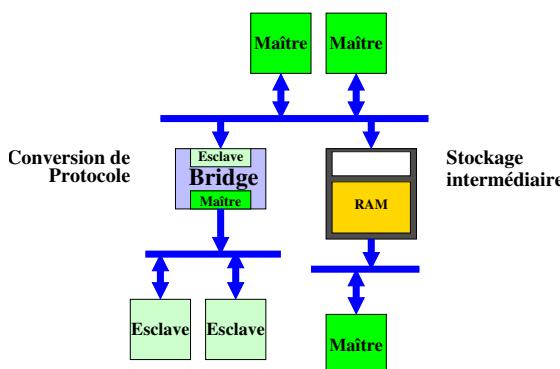
Espace d'adressage



18- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Topologie

S. Mancini

Hiérarchie de bus



19- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Topologie

S. Mancini

Plan

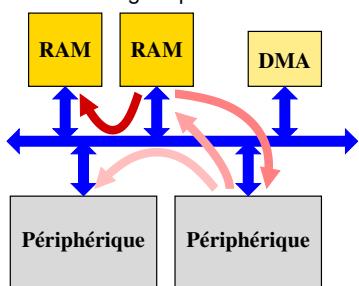
- Bus système
- Eléments d'un bus système
- Infrastructure de communication
 - Topologie
 - DMA
 - Schémas communs
- Exemple de bus
- Conclusion

20- Architecture des SoC- Infrastructure de communication

DMA : "Direct Memory Access"

Un DMA est un automate de transfert automatique de données

Il est programmé et configuré par affectation de registres.



21- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- DMA

S. Mancini

Caractéristiques d'un DMA

- ★ Nombre de canaux
- ★ Gestion du bus
- ★ Transferts
 - Groupés (burst)
 - Auto-incrémentés
 - Canaux chaînés
- ★ Priorité des canaux
- ★ Mode de transfert
 - FIFO
 - Taille des données
- ★ Interruptions
 - Fin de transfert
 - Erreur

22- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- DMA

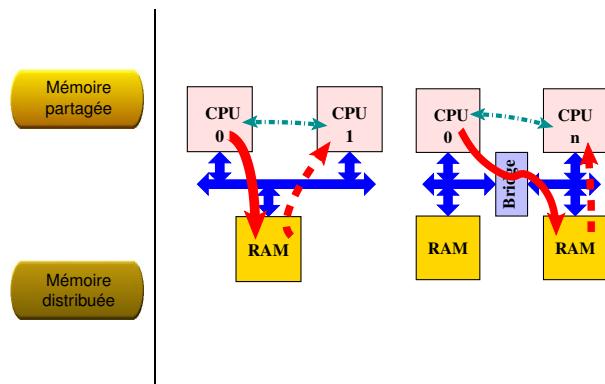
S. Mancini

Plan

- Bus système
- Eléments d'un bus système
- Infrastructure de communication
 - Topologie
 - DMA
 - Schémas communs
- Exemple de bus
- Conclusion

23- Architecture des SoC- Infrastructure de communication

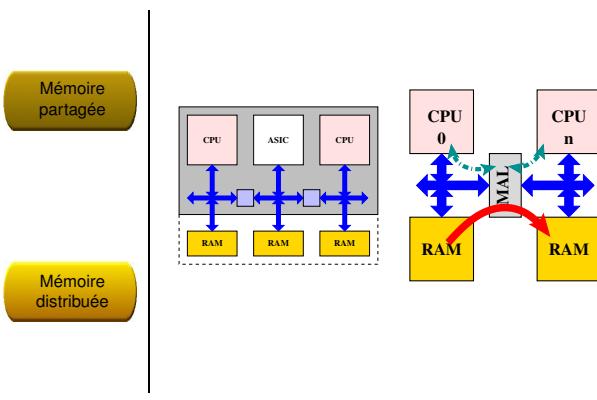
De CPU à CPU



24- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Schémas communs

S. Mancini

De CPU à CPU

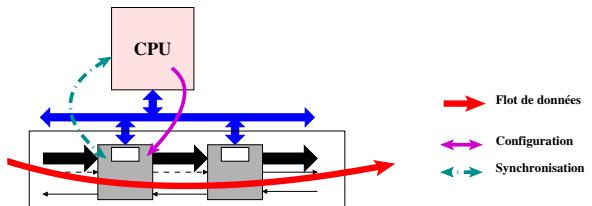


25- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Schémas communs

S. Mancini

Contrôle d'un chemin de données

Un CPU contrôle un chemin de données en le configurant dynamiquement.

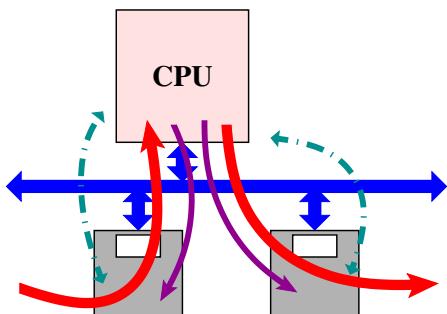


26- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Schémas communs

S. Mancini

Terminal

Un CPU communique vers l'extérieur à travers un chemin de données.

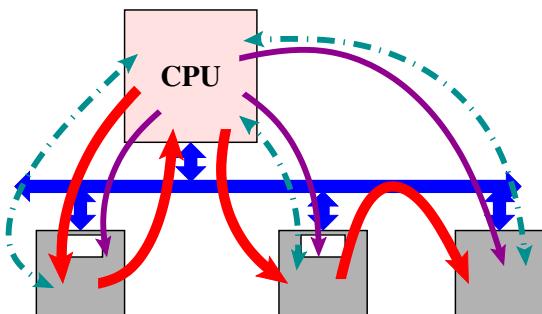


27- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Schémas communs

S. Mancini

Coopération

Un CPU participe aux calculs. Il se sert des calculateurs comme des accélérateurs.

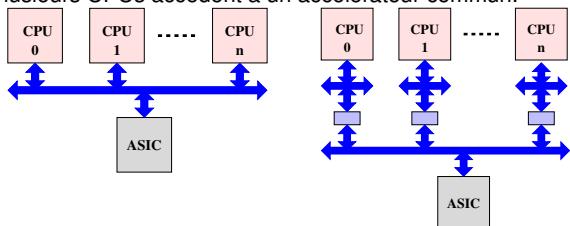


28- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Schémas communs

S. Mancini

Accès à une ressource commune

Plusieurs CPUs accèdent à un accélérateur commun.



Mémoire partagée

Mémoire distribuée

la ressource doit être visible dans tous les espaces d'adresses.

29- Architecture des SoC- Infrastructure de communication
Infrastructure de communication- Schémas communs

S. Mancini

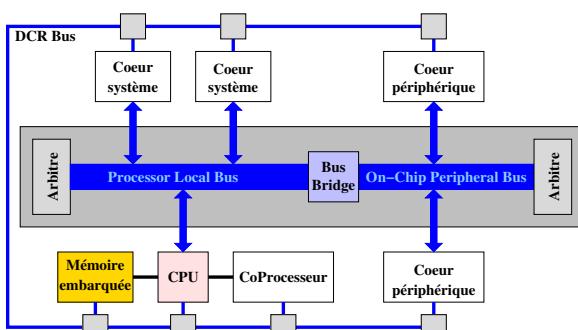
Plan

- Bus système
- Eléments d'un bus système
- Infrastructure de communication
- Exemple de bus
- Coreconnect
- AMBA
- Conclusion

30- Architecture des SoC- Infrastructure de communication

Architecture

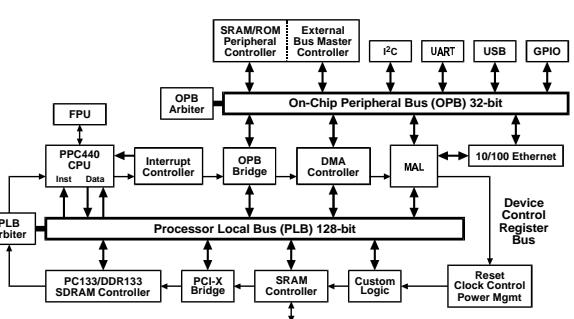
CoreConnect est une architecture de bus proposé par IBM



31- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini

CoreConnect - Exemple



32- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini

CoreConnect - Contrôle

CoreConnect propose un cœur d'arbitre de bus PLB

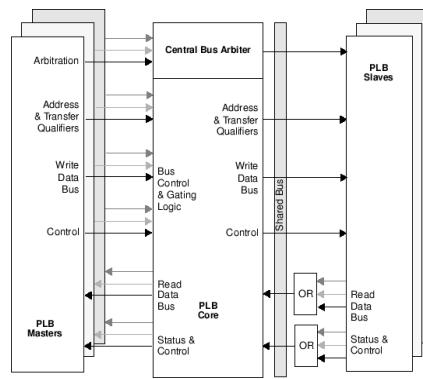
Caractéristiques :

- 8/16 Maîtres
- 8/16 jeux de priorité, sélectionnables dynamiquement
- 2 déclinaisons d'arbitrage
 - Fixe
 - "Juste"

33- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

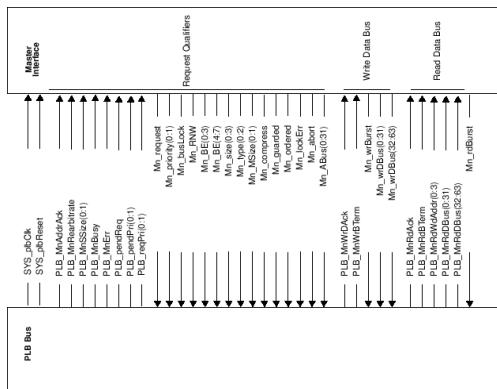
PLB Signaux



34- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

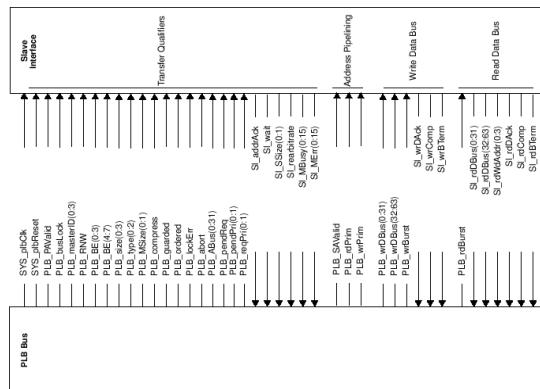
PLB Master



35- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

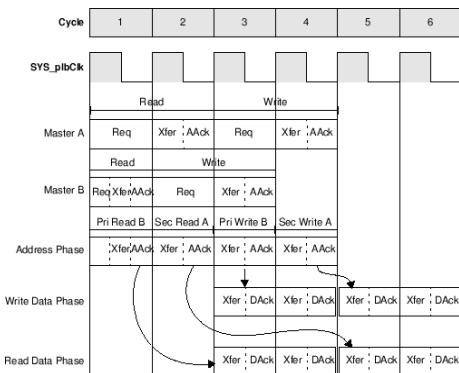
PLB Slave



36- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

PLB & pipeline de transferts



37- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

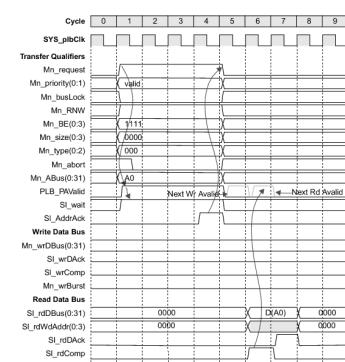
Exercice : interface à une BRAM

Faire le schéma et proposer une machine à état d'une interface entre un bus PLB et une mémoire RAM.
L'interface doit être capable de gérer les accès simple et les burst de longueur fixé par le maître.

PLB read

Figure 10 shows the operation of a single read data transfer on the PLB. The slave asserts SI_wait to indicate to the PLB arbiter that the address is valid but is unable to latch the address or transfer qualifiers at this time. The PLB arbiter will continue to drive the PLB_PAValid signal as well as the address and transfer qualifier signals until the slave device asserts the SI_addrAck signal. The slave then asserts the SI_rdComp signal in the clock cycle preceding the data acknowledge phase to indicate that the transfer will complete in the following clock cycle and that the arbiter may arbitrate the next read request.complete.

PLB Read



39- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

40- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

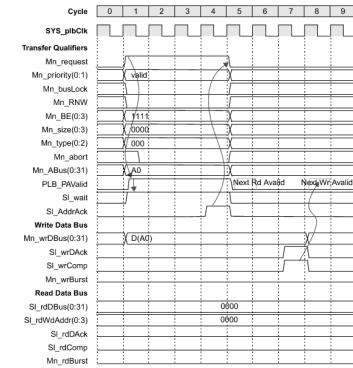
PLB Write

Figure 11 shows the operation of a single write data transfer on the PLB. The slave asserts the SI_wait signal to indicate to the PLB arbiter that the address is valid but that the slave is unable to latch the address or transfer qualifiers at this time. The PLB arbiter will continue to drive the PLB_PAValid signal as well as the address and transfer qualifier signals until the slave device asserts the SI_addrAck signal. The slave then asserts the SI_wrComp and SI_wrDAck to indicate that data is valid on the bus and that the transfer is complete. Note that the write data bus must be valid at the time Mn_request is first asserted and held until the end of the clock cycle in which SI_wrDAck signal is asserted.

41- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

PLB Write



42- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

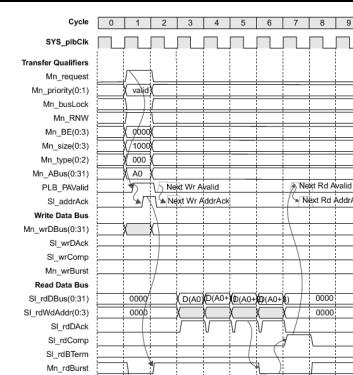
PLB burst read

Figure 19 shows the operation of a burst read from a slave device on the PLB. A master may request a burst transfer across the bus if it needs to read two or more sequential memory locations. The address bus and transfer qualifiers are latched by the slave when the SI_addrAck signal is asserted. The slave will internally increment the address sequentially for each data transfer and will continue to fetch data until it detects a low value on the Mn_rdBurst signal. The burst transfer is then completed by the slave device asserting the SI_rdComp in the data acknowledge phase of the last data transfer cycle following the negation of the Mn_rdBurst signal.

43- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

PLB burst read



44- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

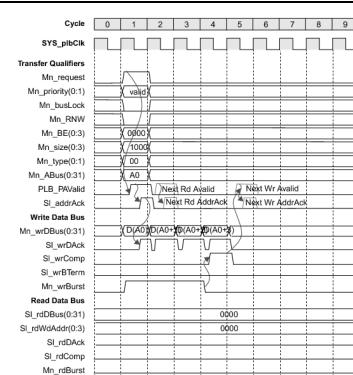
PLB burst write

Figure 21 shows the operation of a burst write to a slave device on the PLB. A master may request a burst write transfer across the bus if it needs to write two or more sequential memory locations. The address bus and transfer qualifiers are latched by the slave when the SI_addrAck signal is asserted. The slave will internally increment the address sequentially for each data transfer. Once the slave detects a low value on the Mn_wrBurst signal, the slave will assert the SI_wrComp signal during the data acknowledge phase for the next (and last) data transfer cycle to indicate to the PLB arbiter that the burst transfer is complete.

45- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

S. Mancini 

PLB burst write



46- Architecture des SoC- Infrastructure de communication
Exemple de bus- Coreconnect

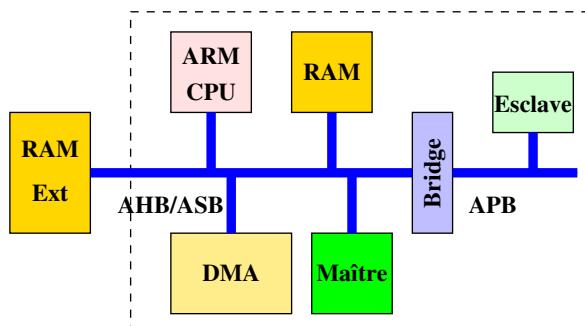
S. Mancini 

Plan

- Bus système
- Eléments d'un bus système
- Infrastructure de communication
- Exemple de bus
 - Coreconnect
 - AMBA
- Conclusion

47- Architecture des SoC- Infrastructure de communication

AMBA - Architecture



48- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

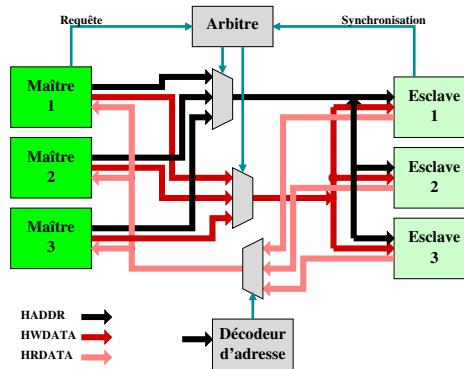
S. Mancini 

Signaux

- ★ Transfert de données
- ★ Contrôle
 - Type de transfert
 - Niveau de protection
 - Gestion des caches
- ★ Arbitrage
 - Requête
 - Temporisation des réponses

49- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

S. Mancini



50- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

S. Mancini

Contrôle

AMBA ne prévoit pas de politique d'arbitrage spécifique

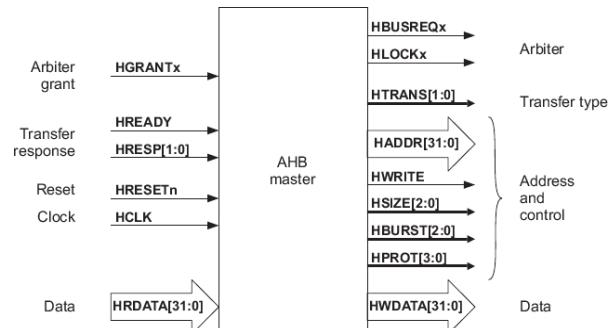
AMBA prévoit la préemption :

- Taille de paquet maximum
- Réponse de l'esclave différée

51- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

S. Mancini

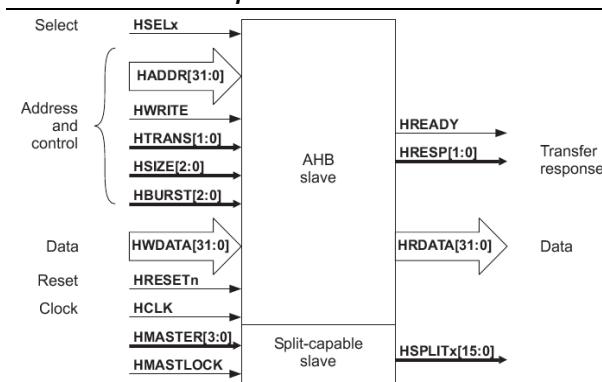
Exemple : AMBA AHB master



52- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

S. Mancini

Exemple : AMBA AHB slave



53- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

S. Mancini

Exemple : AMBA AHB protection

HPROT[3] cacheable	HPROT[2] bufferable	HPROT[1] privileged	HPROT[0] data/opcode	Description
-	-	-	0	Opcode fetch
-	-	-	1	Data access
-	-	0	-	User access
-	-	1	-	Privileged access
-	0	-	-	Not bufferable
-	1	-	-	Bufferable
0	-	-	-	Not cacheable
1	-	-	-	Cacheable

54- Architecture des SoC- Infrastructure de communication
Exemple de bus- AMBA

S. Mancini

Plan

- ❑ Bus système
- ❑ Eléments d'un bus système
- ❑ Infrastructure de communication
- ❑ Exemple de bus
- ✖ Conclusion

Les bus systèmes des Systèmes Intégrés ont des spécificités :

- ❑ Haute fréquence de fonctionnement
- ❑ Grands débits et largeur de bus
- ❑ Latence relativement élevée
- ❑ Pipeline des accès
- ❑ Signaux de contrôle complexes
- ❑ Topologie complexe
- ❑ Cohabitation de différentes technologies

Les bus systèmes sont adaptés aux besoins !!!!

55- Architecture des SoC- Infrastructure de communication

Conclusion

S. Mancini

Architecture des SoC

Infrastructure de communication

S. Mancini

Plan Détaillé

* Bus système

- Objectifs
- Problématique
- Physiologie
- Exemple : communication CPU/IP
- Exemple : communication CPU/IP

* Éléments d'un bus système

- ❑ Connectique
 - Bus Trois Etats
 - Bus "Cross bar"
- ❑ Contrôleur de bus
 - Contrôleur de bus
 - Arbitrage
 - Gestion des interblocages
 - Exercice : arbitre de bus

* Exercice : arbitre de bus

- Exercice : arbitre de bus

* Infrastructure de communication

- ❑ Topologie
 - Espace d'adressage
 - Hiérarchie de bus
- ❑ DMA
 - DMA : "Direct Memory Access"
 - Caractéristiques d'un DMA
- ❑ Schémas communs
 - De CPU à CPU
 - Contrôle d'un chemin de données
 - Terminal
 - Coopération
 - Accès à une ressource commune

* Exemple de bus

- ❑ Coreconnect
 - Architecture
 - CoreConnect - Exemple
 - CoreConnect - Contrôle
 - PLB Signaux
 - PLB Master
 - PLB Slave
 - PLB & pipeline de transferts
 - Exercice : interface à une BRAM
 - PLB read
 - PLB Read
 - PLB Write
 - PLB Write

* PLB burst read

- PLB burst read

* PLB burst write

- PLB burst write

* AMBA

- AMBA - Architecture
- Signaux
- Connectique AHB
- Contrôle
- Exemple : AMBA AHB master
- Exemple : AMBA AHB slave
- Exemple : AMBA AHB protection

* Conclusion

- Conclusion

Architecture des SoC

Architecture mémoire

S. Mancini

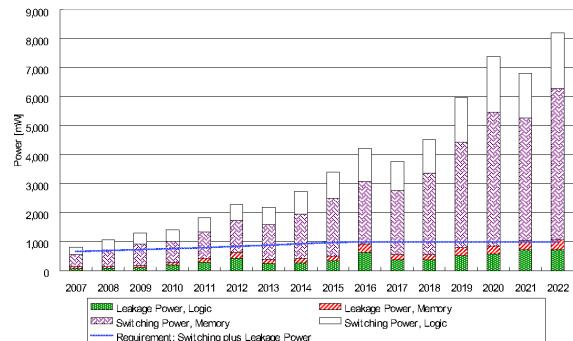


Plan

- ✖ Introduction
 - ❑ Architecture des mémoires
- ❑ Technologie des points mémoire
- ❑ Hiérarchie mémoire
- ❑ Méthodologie
- ❑ TD : estimation de mouvement

1- Architecture des SoC- Architecture mémoire

Impact de l'architecture mémoire



2- Architecture des SoC- Architecture mémoire
Introduction

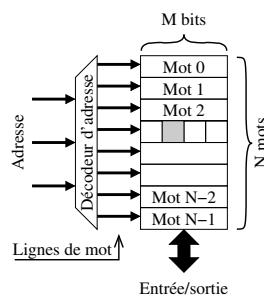
S. Mancini

Architecture des mémoires

Les mémoires sont des tableaux de points mémoire assemblés en lignes de mot.

La technologie du point mémoire détermine les performances :

- ★ Densité (coût par bit)
- ★ Rapidité d'accès

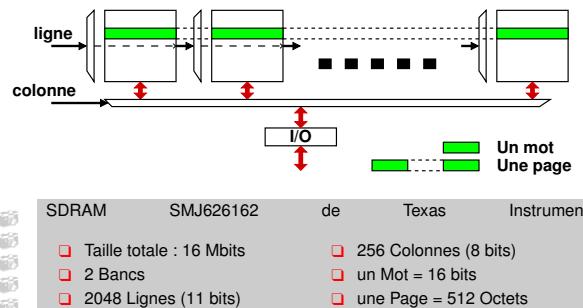


3-Architecture des SoC- Architecture mémoire
Introduction- Architecture des mémoires

S. Mancini

Organisation en colonnes

Les contraintes technologiques imposent de découper les tableaux en blocs, disposés en colonnes

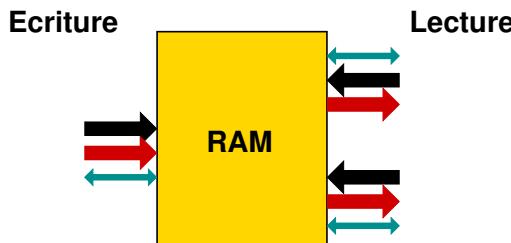


4-Architecture des SoC- Architecture mémoire
Introduction- Architecture des mémoires

S. Mancini

RAM multi-port

Certaines RAM permettent l'accès simultané en écriture et en lecture sur plusieurs ports.



L'utilisateur doit gérer les éventuelles synchronisations entre la lecture et l'écriture

5-Architecture des SoC- Architecture mémoire
Introduction- Architecture des mémoires

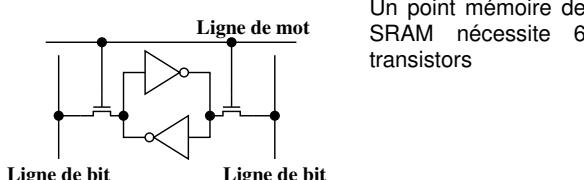
S. Mancini

Plan

- ❑ Introduction
- ✖ Technologie des points mémoire
 - ✓ SRAM
 - ❑ Famille (S)DRAM
 - ❑ Autres points mémoire
 - ❑ Conclusion
- ❑ Hiérarchie mémoire
- ❑ Méthodologie
- ❑ TD : estimation de mouvement

6-Architecture des SoC- Architecture mémoire

Technologie des SRAM



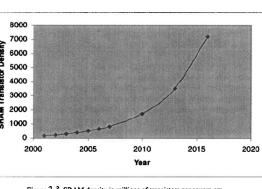
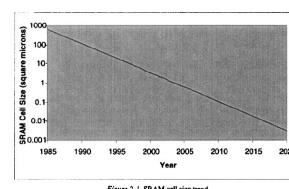
- Rapidité
- Technologie "standard"

- ➡ Densité faible
- ➡ Non synthétisable

7-Architecture des SoC- Architecture mémoire
Technologie des points mémoire- SRAM

S. Mancini

Densité des SRAM



High performance memory testing : design principles, fault modeling, and self test. Par R. Dean Adams

8-Architecture des SoC- Architecture mémoire
Technologie des points mémoire- SRAM

S. Mancini

Performances des SRAM

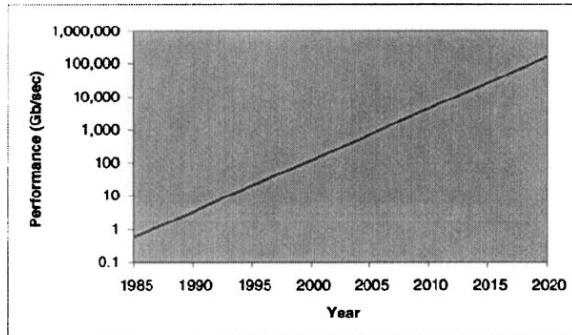


Figure 2-4. SRAM overall performance trend.

9- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- SRAM

S. Mancini

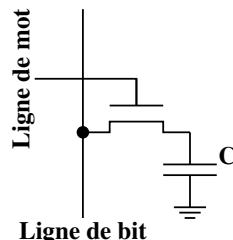
Technologie des DRAM et SDRAM

SDRAM=Synchronous Dynamic RAM

Un point mémoire SDRAM ou DRAM est constitué :

- d'un transistor
- d'une capacité

Les DRAM sont **asynchrones**
les SDRAM possèdent une interface **synchrone**



Grande densité

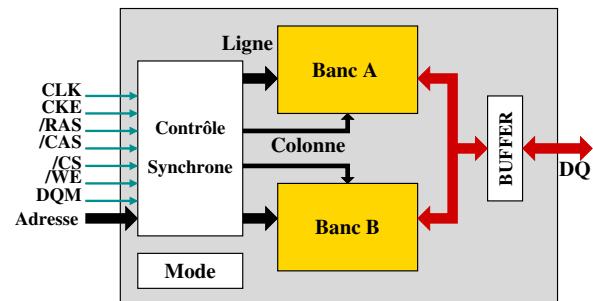
Rafraîchissement

11- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

10- Architecture des SoC- Architecture mémoire

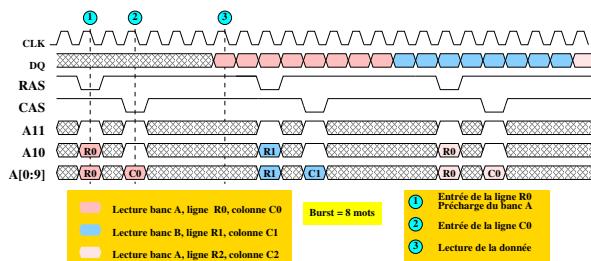
Architecture des SDRAM



12- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

Accès aux SDRAM



13- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

Rafraîchissement des DRAM

Pour maintenir le contenu d'une SDRAM, il faut assurer son rafraîchissement.

Il est réalisé par la lecture des bits mémoire à intervalles régulier.

La DDR3 MT41J512M4 (Micron) nécessite un rafraîchissement à une période de 64 ms.

Il est possible de réaliser un rafraîchissement semi-automatique.

La commande REFR doit être émise au plus toutes les 7.8 µs. Sa durée d'exécution dépend de la fréquence et de la taille de la mémoire :

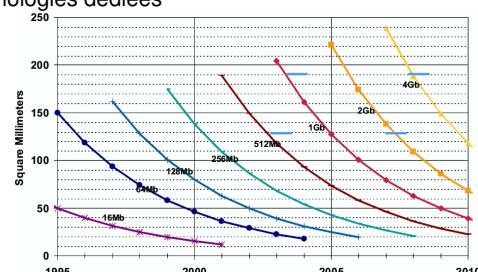
- f=800 Mhz, 1 Gb, RFC=44 cycles
- f=1600 Mhz, 4 Gb, RFC=240 cycles

14- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

Densités des SDRAM

Technologies dédiées



Technologie 90 nm mixte

- 430 KPortes/mm²
- SRAM 1.6 à 1.2 mm² par Mbit
- DRAM 0.5 mm² par Mbit

15- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

Performance des SDRAM

Densité

Mémoire	Bank	Densité	Architecture
DDR	4	256 Mb - 1 Gb	2n-prefetch
DDR2	8	256 Mb - 1 Gb	4n-prefetch
DDR3	8	1-4 Gb	8n-prefetch

Performances

Mémoire	Freq clk	Bit par cycle	Freq bit	CL (cycle)	CL (ns)
DDR	75-200	2	150-400	2-3	≈ 15
DDR2	200-400	2	400-800	3-6	≈ 15
DDR3	400-800	2	800-1600	5-10	≈ 13-15

16- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

En résumé

- Densité élevée
- Débit élevé
- Coût faible
- Grande latence
- Technologie spécifique
- Consommation
- Non synthétisable



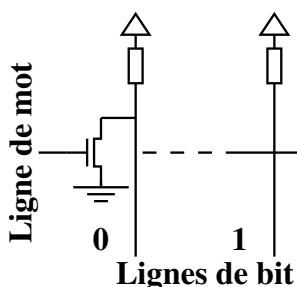
17- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Famille (S)DRAM

S. Mancini

Plan

- Introduction
- Technologie des points mémoire
 - SRAM
 - Famille (S)DRAM
 - Autres points mémoire
 - Conclusion
- Hiérarchie mémoire
- Méthodologie
- TD : estimation de mouvement

ROM



- Petit
- Rapide

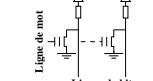
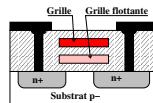
Lecture seule

19- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Autres points mémoire

S. Mancini

FLASH

C'est une mémoire non volatile, programmable in-situ.



- Mémorisation de données hors tension
- Accès sécurisé

On y accède par une séquence d'accès.

Contrôle

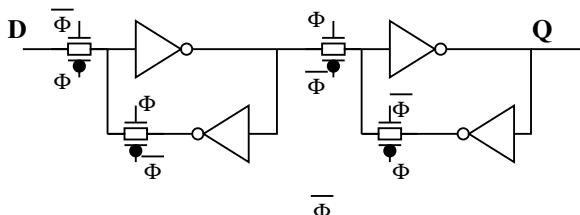
- Logiciel
- Matériel

- Lent
- Technologie spécifique

20- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Autres points mémoire

S. Mancini

Registre



- Rapide
- Utilisable en synthèse logique

Gros

21- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Autres points mémoire

S. Mancini

Plan

- Introduction
- Technologie des points mémoire
 - SRAM
 - Famille (S)DRAM
 - Autres points mémoire
 - Conclusion
- Hiérarchie mémoire
- Méthodologie
- TD : estimation de mouvement

Types de mémoire

Type de mémoire	Taille ≈	Densité	Rapidité
ROM	100 KO		
SRAM	100 KO		
SDRAM	GO		
PROM	KO		
FLASH	KO/GO		
Associative	KO		
Registres	KO		

23- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Conclusion

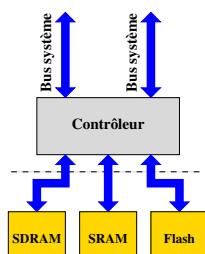
S. Mancini

Contrôleur mémoire

Le contrôleur mémoire :

- Convertit les protocoles des bus système et mémoire
- Gère les accès multiples
- Gère l'espace d'adressage
- Protège les données
- Effectue le rafraîchissement

On trouve des contrôleurs mémoire génériques, paramétrables selon les besoins.



24- Architecture des SoC- Architecture mémoire
Technologie des points mémoire- Conclusion

S. Mancini

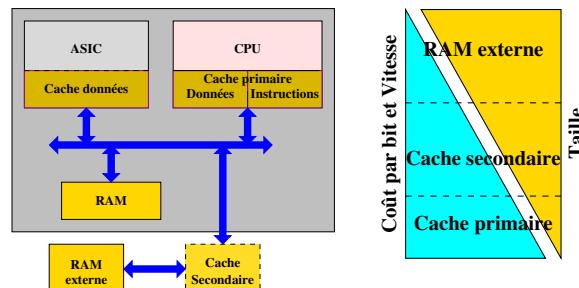
Plan

- Introduction
- Technologie des points mémoire
- ✖ Hiérarchie mémoire
 - ✓ Objectifs
 - ❑ Scratch-Pad RAM et BRAM
 - ❑ Caches standards
- Méthodologie
- TD : estimation de mouvement

25- Architecture des SoC- Architecture mémoire

Objectifs d'une hiérarchie mémoire

L'objectif est d'obtenir les performances de la mémoire la plus rapide au coût de la moins chère.



26- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Objectifs

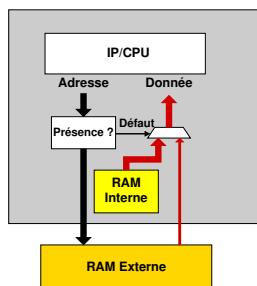
S. Mancini

Principe d'une hiérarchie mémoire

Une hiérarchie mémoire sert à stocker dans la mémoire la plus rapide les données les plus souvent utilisées.

Elle exploite le principe de localité { temporelle
spatiale}

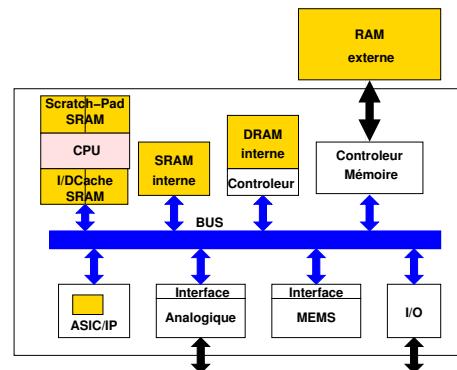
Les mécanismes de copie des données sont plus ou moins spécialisés ou génériques, à gérer par l'application ou automatiques.



27- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Objectifs

S. Mancini

Types de mémoires



28- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Objectifs

S. Mancini

Plan

- Introduction
- Technologie des points mémoire
- ✖ Hiérarchie mémoire
 - ✓ Objectifs
 - ❑ Scratch-Pad RAM et BRAM
 - ❑ Caches standards
- Méthodologie
- TD : estimation de mouvement

29- Architecture des SoC- Architecture mémoire

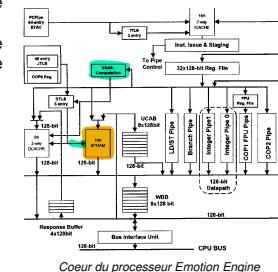
Scratch-Pad RAM

Une **Scratch-Pad RAM (SPRAM)** est une mémoire directement connectée à un cœur de processeur. Elle est visible dans l'espace mémoire.

Il est possible d'y accéder en un cycle d'horloge, sans arbitrage.

Une SPRAM doit être gérée par le logiciel. Les données peuvent être lues/écrites par :

- Les instructions LD/ST standards
- Un DMA interne au processeur



30- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Scratch-Pad RAM et BRAM

S. Mancini

BRAM

Une **Block-RAM (BRAM)** est une mémoire accessible par le bus système. C'est une IP esclave accédée par les IPs maîtres.

En général, les BRAM servent aux échanges de données entre les processeurs et IPs maîtres.

Dans un contexte d'architecture à mémoire partagée, les BRAM sont associées à des IP utiles à la synchronisation (gestionnaire d'interruption, etc...).

31- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Scratch-Pad RAM et BRAM

S. Mancini

Plan

- Introduction
- Technologie des points mémoire
- ✖ Hiérarchie mémoire
 - ✓ Objectifs
 - ❑ Scratch-Pad RAM et BRAM
 - ✓ Caches standards
- Méthodologie
- TD : estimation de mouvement

32- Architecture des SoC- Architecture mémoire

Objectifs

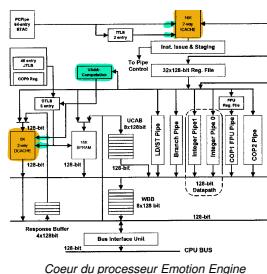
Les cache (antémémoire) sont des mémoires rapides qui disposent d'un mécanisme automatique de copie/écriture des données/instructions de la mémoire principale.

Un cache exploite le principe de localité { temporelle spatiale (en adresse)

Les principaux paramètres d'efficacité sont :

- o La taille mémoire
- o La politique de gestion

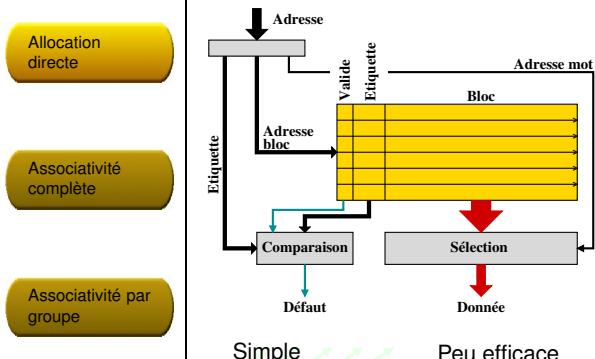
Le principal critère d'efficacité est le **taux de défaut de cache**, qui dépend de ces paramètres et de l'application.



33- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

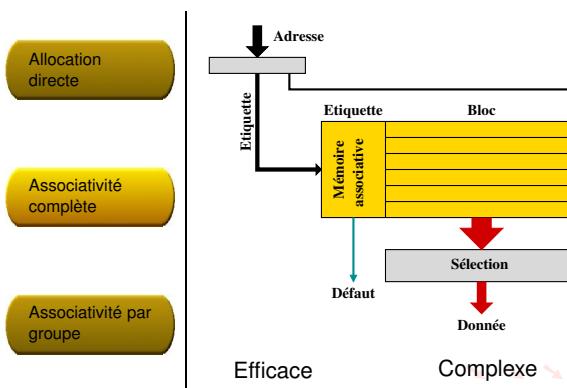
Stratégies d'allocation des blocs



34- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

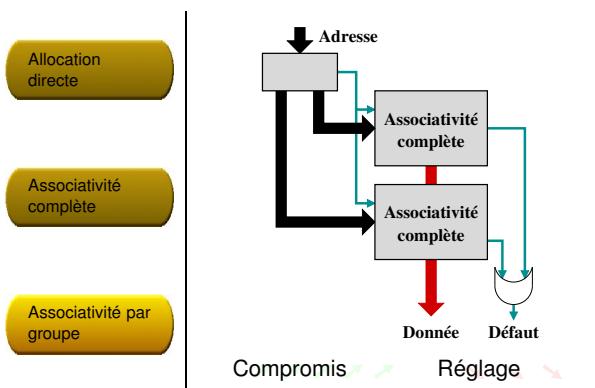
Stratégies d'allocation des blocs



35- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

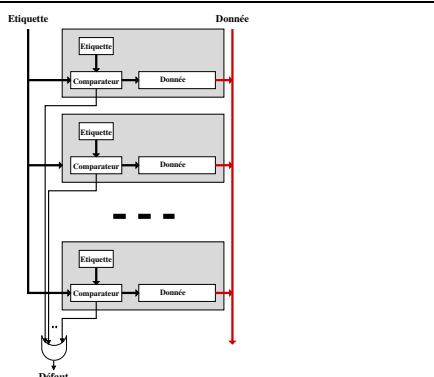
Stratégies d'allocation des blocs



36- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

Mémoire associative



37- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

Temps d'accès aux caches

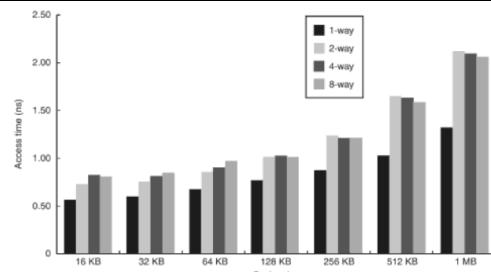


Figure 5.4 Access times as size and associativity vary in a CMOS cache. These data are based on the CACTI model 4.0 by Tarjan, Thoziyor, and Jouppi (2006). They assumed 90 nm feature size, a single bank, and 64-byte blocks. The median ratios of access time relative to the direct-mapped caches are 1.32, 1.39, and 1.43 for 2-way, 4-way, and 8-way associative caches, respectively.

Performance d'un cache

Temps moyen d'accès = Temps hit + Taux de défaut * Pénalité de défaut

Comparer le temps moyen entre un cache 2-way set associative et un 4-way set associative de 8 KB.

On fait l'hypothèse que le 4-way set associative permet de passer d'un Taux de défaut de 0,049 à 0,044. En faisant les hypothèses suivantes :

- ★ le 2-way set associative a un temps de hit de 1 et le 4-way set associative de 1,1
- ★ La Pénalité de défaut du 2-way set associative est de 10 (accès à un cache L2 sans défaut)

Lequel a le meilleur temps moyen d'accès ?

Politique de remplacement

Lors d'un défaut de cache, il faut choisir un bloc dans lequel on va stocker la donnée demandée

★ Aléatoire

Un bloc est choisi au hasard

★ LRU (Least Recently Used)

la durée pendant laquelle un bloc n'a pas été accédé est mesurée

★ LFU (Least Frequently Used)

On mesure le nombre de fois qu'un bloc est utilisé

★ FIFO L'ordre de remplacement est fixé

39- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

40- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

Amélioration du taux de défaut de cache

- ★ Architecture du cache
 - ☞ Taille du cache
 - ☞ Amélioration de l'associativité
 - ☞ Buffer de victimes de remplacement
- ★ Organisation des accès mémoire
 - ☞ Organisation des données
 - ☞ Structure des programmes

41- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

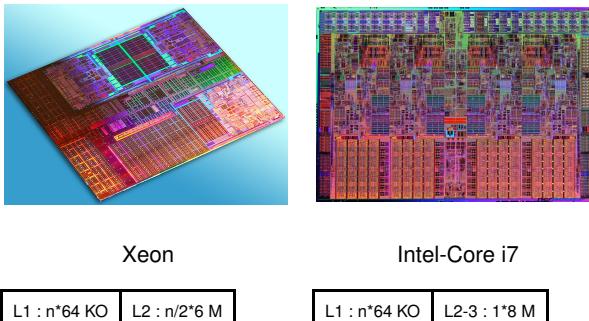
Principaux paramètres d'efficacité

- ★ Taille du cache
- ★ Politique d'allocation
 - Allocation directe
 - Associativité complète
 - Associativité par groupe
- ★ Politique d'écriture
 - Transparent
 - Ecriture différée
- ★ Politique de remplacement
 - LRU (Least Recently Used)
 - Aléatoire
- ★ Addressage
 - Virtuel
 - Physique

42- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

Exemple : les coeurs Intel

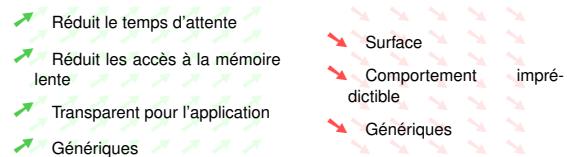


43- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

En résumé

Les différents paramètres des caches doivent être adaptés à l'application.



44- Architecture des SoC- Architecture mémoire
Hiérarchie mémoire- Caches standards

S. Mancini

Plan

- Introduction
- Technologie des points mémoire
- Hiérarchie mémoire
- ✖ Méthodologie
- TD : estimation de mouvement

45- Architecture des SoC- Architecture mémoire

Spécificité des mémoires dans les systèmes intégrés

Les Systèmes Intégrés permettent :

- ★ D'intégrer différents types de mémoires dans un même circuit
- ★ De réaliser des mémoires sur mesure
- ☞ Largeur de mot & Adressage
- ☞ Accès
- ★ De construire des hiéronymies mémoire adaptées à l'application

46- Architecture des SoC- Architecture mémoire
Méthodologie

S. Mancini

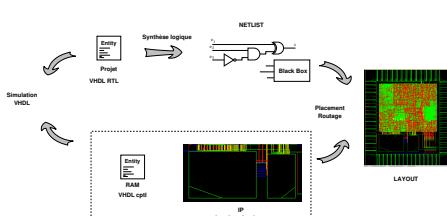
Utilisation des mémoires

Les (D/S)RAM sont proposées sous la forme de macro-blocs.

Les fondeurs fournissent :

- ★ des modèles de simulation
- ★ des masques (vue abstraite)

La synthèse se fait par instanciation de "boîte noire".



47- Architecture des SoC- Architecture mémoire
Méthodologie

S. Mancini

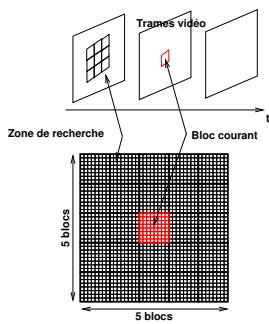
Plan

- Introduction
- Technologie des points mémoire
- Hiérarchie mémoire
- Méthodologie
- ✖ TD : estimation de mouvement

48- Architecture des SoC- Architecture mémoire

Principe de l'estimation de mouvement

La compression vidéo MPEG nécessite le calcul d'une fonction de distance entre un bloc et ses voisins.



49- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Questions

Données : Flux vidéo CCIR 656 à 25 Hz, trames de dimension 720*525, 8 bit par pixel. Les macro-blocs font 8*8 pixels. La zone de recherche fait 5*5 blocs. Les trames sont arrondies (supérieur) sur un multiple de macro-bloc. Les pixels d'une trame sont placés dans l'ordre canonique.

- Ecrire le nid de boucles qui correspond à l'estimation de mouvement de tous les blocs d'une image
- Calculez le débit nécessaire pour réaliser ce traitement en temps réel

51- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Blanc !

50- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Solution

Questions

Sans cache : On ne souhaite pas utiliser de cache, les trames sont stockées en DDR-SDRAM ($f=100$ Mhz, CL=3, 32 bits). Seul le bloc courant est stocké en mémoire interne.

- Ce débit est-il atteignable sans exploiter les burst ni le recouvrement des accès ?
- Proposez un séquencement des calcul pour des burst de 2, 4 et 8 mots
- Calculez le débit atteignable selon la taille de burst
- Comment les performances évoluent-elles en fonction de la latence ?
- Quel ordre des pixels en mémoire permettrait ce débit sans gestion des recouvrements ?

53- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Solution

54- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Questions

- Calculez le taux de réutilisation d'un pixel de l'image de référence

Avec cache standard : On souhaite utiliser un cache standard de CPU pour exploiter au mieux la réutilisation.

Processeur	Taille	Ligne	Set-Associative	Selection set
PowerPC	16 KB	32 B	2 way	[12..5]
TM32	16 KB	64 B	8-way	[10..6]

- Superposez les "set" et macroblocs
- Estimez (à la louche) les performances de ces caches
 - Pour une recherche, le cache étant vide au début
 - Pour la recherche suivante

55- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Solution

56- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini

Questions

Avec cache “maison” : La conception d'un cache maison permet davantage de compromis et nous allons en explorer quelques-un.

- Quelle architecture permet de ne charger que 5 blocs de référence pour chaque nouveau bloc courant ?
 - De quelle quantité mémoire avons nous besoin ?
 - De combien de macro-blocs mémoire (identifiez deux cas) ?
- Même question pour un seul bloc de référence chargé ?

57- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini 

Architecture des SoC

Architecture mémoire

S. Mancini

Plan Détailé

- ✖ **Introduction**
 - ✖ Impact de l'architecture mémoire
 - ❑ Architecture des mémoires
 - ✖ Architecture des mémoires
 - ✖ Organisation en colonnes
 - ✖ RAM multi-port
- ✖ **Technologie des points mémoire**
 - ❑ SRAM
 - ✖ Technologie des SRAM
 - ✖ Densité des SRAM
 - ✖ Performances des SRAM
 - ❑ Famille (S)DRAM
 - ✖ Technologie des DRAM et SDRAM
 - ✖ Architecture des SDRAM
 - ✖ Accès aux SDRAM
- ✖ **Rafraîchissement des DRAM**
 - ✖ Densités des SDRAM
 - ✖ Performance des SDRAM
 - ✖ En résumé
- ❑ **Autres points mémoire**
 - ✖ ROM
 - ✖ FLASH
 - ✖ Registré
- ❑ **Conclusion**
 - ✖ Types de mémoire
 - ✖ Contrôleur mémoire
- ✖ **Hierarchie mémoire**
 - ❑ Objectifs
 - ✖ Objectifs d'une hiérarchie mémoire
 - ✖ Principe d'une hiérarchie mémoire

58- Architecture des SoC- Architecture mémoire
TD: estimation de mouvement

S. Mancini 

✖ Méthodologie

- ✖ Spécificité des mémoires dans les systèmes intégrés
- ✖ Utilisation des mémoires

✖ TD : estimation de mouvement

- ✖ Principes de l'estimation de mouvement
- ✖ Blanc !
- ✖ Questions
- ✖ Solution
- ✖ En résumé

Architecture des SoC

Etudes de cas Multimédia

S. Mancini



Plan

- ✖ Contexte
 - ✓ Présentation
 - ❑ Standards vidéo
 - ❑ 3D
- ❑ Set top Box PNX8500
- ❑ Set top box IBM STB03xxx
- ❑ Plateforme multimédia "portable"
- ❑ Playstation 2 & 3
- ❑ GPU

1- Architecture des SoC- Etudes de cas- Multimédia

Applications

- ★ Internet
- ★ Educatif, jeux, etc ...
- ★ Télévision interactive
- ★ MPEG4
- ★ Téléachat
- ★ Télécommunications
- ★ Téléconférence
- ★ Visiophonie
- ★ Imagerie médicale

S. Mancini

Particularités

- ★ Les traitements bas niveau sont réguliers et nécessitent de grande puissances de calcul :
 - ★ Compression/ décompression
 - ★ Filtrage
 - ★ Composition
 - ★ 3D (textures, etc ...)
- ★ La synchronisation des contenus est complexe et irrégulière.

Aussi les systèmes multimédia doivent être constitués :

- ★ d'opérateurs spécifiques, ...
- ★ et de microprocesseurs

3- Architecture des SoC- Etudes de cas
Multimédia- Contexte- Présentation

S. Mancini

Plan

- ✖ Contexte
 - ❑ Présentation
 - ✓ Standards vidéo
 - ❑ 3D
- ❑ Set top Box PNX8500
- ❑ Set top box IBM STB03xxx
- ❑ Plateforme multimédia "portable"
- ❑ Playstation 2 & 3
- ❑ GPU

4- Architecture des SoC- Etudes de cas- Multimédia

Format vidéo

Format vidéo non compressé pour la télévision numérique haute définition.

- ★ SD - CCIR BT-656
 - . 625 ou 525 lignes faisant jusqu'à 720 pixels.
 - . Format 4 :2 :2 (Cb, Y, Cr, Y, ...)
 - . Débit : 27 MOctets/s
- ★ HD - 1080p
 - . 1920*1080 en 24 et 30 Hz
 - . Format 4 :2 :0
 - . Débit : 52 MPixel/s = 78 MO/s

5- Architecture des SoC- Etudes de cas
Multimédia- Contexte- Standards vidéo

S. Mancini

MPEG2

Format vidéo compressé

MPEG2 définit plusieurs résolutions et débits :

Level	Max. sampling dimensions fps	Pixels/sec	Max. bitrate
Low	352 x 240 x 30	3.05 M	4 Mb/s
Main	720 x 480 x 30	10.40 M	15 Mb/s
High 1440	1440 x 1152 x 30	47.00 M	60 Mb/s
High	1920 x 1080 x 30	62.70 M	80 Mb/s

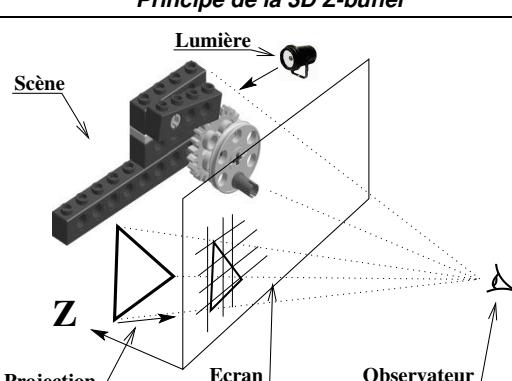
6- Architecture des SoC- Etudes de cas
Multimédia- Contexte- Standards vidéo

S. Mancini

Plan

- ✖ Contexte
 - ❑ Présentation
 - ❑ Standards vidéo
 - ✓ 3D
- ❑ Set top Box PNX8500
- ❑ Set top box IBM STB03xxx
- ❑ Plateforme multimédia "portable"
- ❑ Playstation 2 & 3
- ❑ GPU

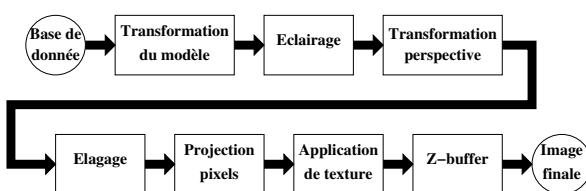
7- Architecture des SoC- Etudes de cas- Multimédia



8- Architecture des SoC- Etudes de cas
Multimédia- Contexte- 3D

S. Mancini

Pipeline 3D



9- Architecture des SoC- Etudes de cas Multimédia- Contexte- 3D

S. Mancini

Performances des accélérateurs 3D

La performance des accélérateurs 3D dépend essentiellement du débit aux mémoires de texture et du nombre de pixels affichés par seconde.

Gamme	Horloge MHz	Débit mémoire	Pixels/s
Geforce 3 (2001)	175	2,4 GB/s	800 M
Geforce 6 (2004)	500	16 GB/s	4 G
Geforce 7 (2005)	600	51 GB/s	10 G
Geforce 8 (2007)	612/1500	51.2 GB/s	15 G
Geforce 9 (2008)	700/1500	51 GB/s	15 G ??
Console (2001)	300	10 GB/s	4 G
Console (2006)	3200	25.6GB/s + 20GB/s	16 G
Professionnel (2001)	500	32*11,2 GB/s	32*6,1 G

10- Architecture des SoC- Etudes de cas Multimédia- Contexte- 3D

S. Mancini

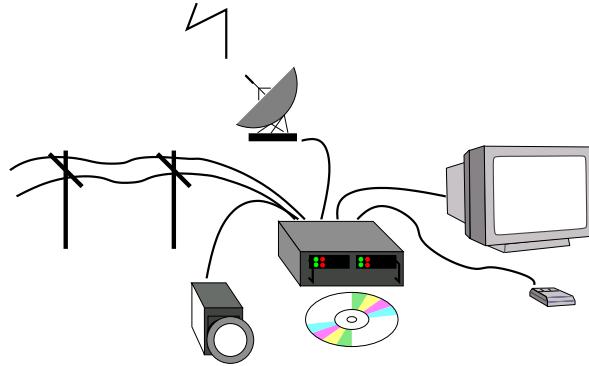
Plan

- Contexte
- Set top Box PNX8500
 - Environnement
 - Les processeurs
 - Bus, mémoire et 3D
 - Architecture logicielle
 - Exemple d'application
 - Le circuit
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

11- Architecture des SoC- Etudes de cas- Multimédia

S. Mancini

Environnement



12- Architecture des SoC- Etudes de cas Multimédia- Set top Box PNX8500- Environnement

S. Mancini

Classe d'application

Le PNX8500 est dédié aux applications multimédia.

Il permet de gérer des **flux vidéo** en réalisant des opérations comme la conversion, la décompression et la composition de flux.

La présence de CPUs facilite la mise en oeuvre d'**environnements interactifs**.

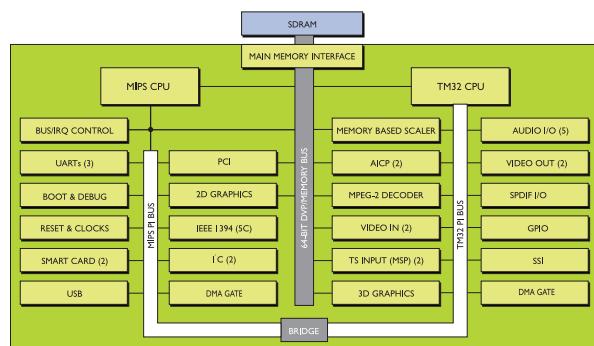
L'accélération 2D autorise des systèmes de fenêtrages.

Une **accélération de la 3D** permet l'utilisation de scènes de synthèse.

13- Architecture des SoC- Etudes de cas Multimédia- Set top Box PNX8500- Environnement

S. Mancini

Architecture générale



14- Architecture des SoC- Etudes de cas Multimédia- Set top Box PNX8500- Environnement

S. Mancini

Plan

- Contexte
- Set top Box PNX8500
 - Environnement
 - Les processeurs
 - Bus, mémoire et 3D
 - Architecture logicielle
 - Exemple d'application
 - Le circuit
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

15- Architecture des SoC- Etudes de cas- Multimédia

S. Mancini

MIPS

Le MIPS 32 bits est dédié à la gestion globale du système.

Parties opératives :

- ★ ALU
- ★ Multiplieur-Accumulateur et Diviseur entier

Cache | I. :16 KB
 | D.:32 KB

Pipeline de 6 étages à 150 MHz,

Il adapté à l'utilisation d'un OS : il contient un module de gestion de la mémoire (MMU).

16- Architecture des SoC- Etudes de cas Multimédia- Set top Box PNX8500- Les processeurs

S. Mancini

Trimédia- parties opératives

Il est dédié aux traitement de la vidéo et du son.
C'est un VLIW 32 bits à 5 opérations par instruction, à 200 MHz. Il dispose de 27 opérateurs.

Functional Unit	Quantity	Latency ^b /Delay ^b	Recovery ^c	Slot Assignment				
				1	2	3	4	5
Constant	5	1	1	✓	✓	✓	✓	✓
Integer ALU	5	1	1	✓	✓	✓	✓	✓
Load/Store	2	3	1					
DSP ALU	2	2	1	✓		✓		
DSP MUL	2	3	1	✓	✓			
Shifter	2	1	1	✓	✓			
Branch	3	3	1	✓	✓	✓		
Int/Float MUL	2	3	1	✓	✓			
Float ALU	2	3	1	✓				
Float Compare	1	1	1			✓		
Float sqrt/div	1	17	16	✓				

17- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Les processeurs

S. Mancini 

Trimédia-Architecture mémoire

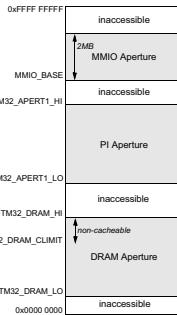
Le trimédia accède à deux bus distincts :

- ❑ DRAM
- ❑ MMIO (sur bus PI)

Cache I. :32 KB
D.:16 KB

Associatif par groupes, à 8 voies de 64 octets par bloc.

La zone MMIO est non-cachable.



Les instructions sont compressées en mémoire externe.

18- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Les processeurs

S. Mancini 

Trimédia-Mécanismes systèmes

Le TM32 implémente un mécanisme de **sémaphores matériels**.

Ils sont accessibles en tant que registres, depuis le cœur ou le bus PI.

Procédure d'accès :

```
write ID to SEM (use 32 bit store, with ID in 12 LSB)
retrieve SEM (use 32 bit load, it returns 0x00000nnn)
if (SEM == ID) {
    performs a short critical section action
    write 0 to SEM
}
else
"try again later, or loop back to write"
```

19- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Les processeurs

S. Mancini 

Plan

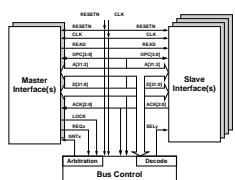
- ❑ Contexte
- ✖ Set top Box PNX8500
 - ❑ Environnement
 - ❑ Les processeurs
 - ✓ Bus, mémoire et 3D
 - ❑ Architecture logicielle
 - ❑ Exemple d'application
 - ❑ Le circuit
- ❑ Set top box IBM STB03xxx
- ❑ Plateforme multimédia "portable"
- ❑ Playstation 2 & 3
- ❑ GPU

20- Architecture des SoC- Etudes de cas- Multimédia

Les bus

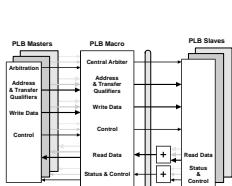
PI Bus

Bus Trois Etats
Bus système
32 bits à 50 MHz ≈ 200 MB/s



DVP Memory bus

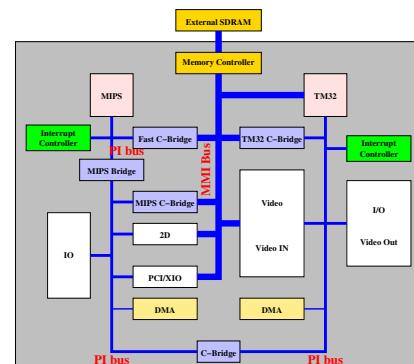
Bus Point à Point
Bus mémoire
64 bits à 143 MHz ≈ 1 GB/s



21- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Bus, mémoire et 3D

S. Mancini 

Architecture des Bus

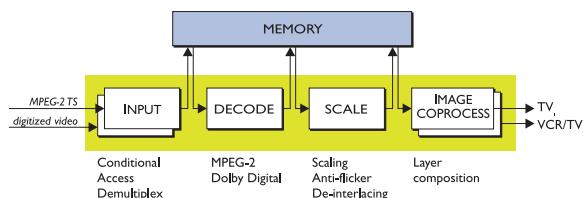


22- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Bus, mémoire et 3D

S. Mancini 

Architecture Mémoire

Les transferts de données entre les différents opérateurs se font à travers la mémoire partagée principale.



Grande souplesse

Les débits sur le bus sont doublés

23- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Bus, mémoire et 3D

S. Mancini 

Architecture "Système"

La présence d'une mémoire partagée nécessite l'utilisation de matériel spécifique aux mécanismes systèmes tels que les sémaphores.

- ★ Le bus PI implémente la fonction "lock"
- ★ Le TM32 implémente des sémaphores
- ★ Les deux processeurs ont des zones non-cachables

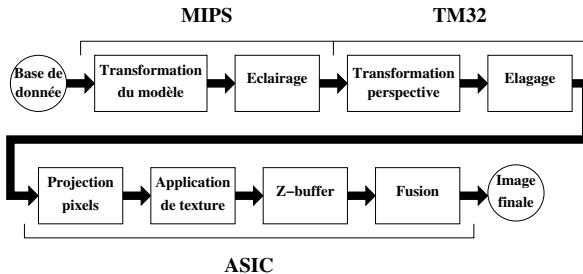
Les deux processeurs peuvent communiquer en construisant des communications par "tube", réalisées par les OS.

24- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Bus, mémoire et 3D

S. Mancini 

Partitionnement de la 3D

60 Millions de Pixels par seconde



25- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Bus, mémoire et 3D

S. Mancini

Plan

- Contexte
- Set top Box PNX8500
 - Environnement
 - Les processeurs
 - Bus, mémoire et 3D
 - Architecture logicielle
 - Exemple d'application
 - Le circuit
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

27- Architecture des SoC- Etudes de cas- Multimédia

Streaming Software

Le Trimedia Streaming Software Architecture (TSSA) permet de simplifier la gestion de la synchronisation des fonctions de traitement.

L'utilisateur décrit les fonctions de traitement et leurs connexions.

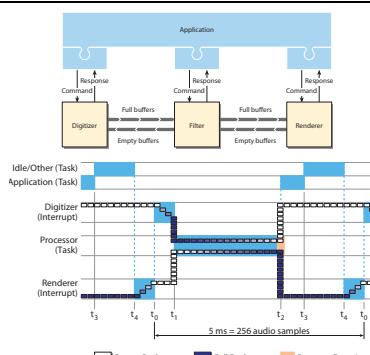
TSSA gère les déplacements, créations et destructions des données.

La gestion de la mémoire n'est pas laissée à l'OS pour éviter la fragmentation de l'espace mémoire.

29- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Architecture logicielle

S. Mancini

Exemple TSSA simple



31- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Architecture logicielle

S. Mancini

Vidéo

- ★ Décodage des "Transport Stream" par 3 MSP (MPEG System Processor)
 - ☞ un MSP est un processeur RISC 16 bits
- ★ Les traitements vidéo sont réalisés par des ASICs
 - ☐ Décodage MPEG2
 - ☐ Homotétrie
 - ☐ Composition d'image

26- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Bus, mémoire et 3D

S. Mancini

Systèmes d'exploitations

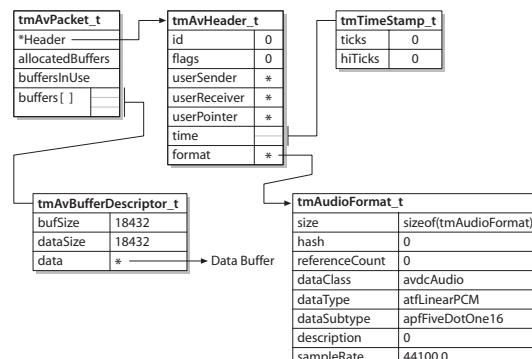
Les deux processeurs peuvent supporter un système d'exploitation.

- ★ Le **MIPS** permet d'utiliser des systèmes d'exploitation à processus "protégés".
- ☞ Un MMU permet de disjoindre les espaces mémoire des processus.
Inconvénients : les changements de contexte peuvent être lents.
- ★ L'OS du **TM32** doit être rapide pour effectuer la synchronisation des flux.
Pour réduire le temps changement de contexte, une petite partie des 128 registres est sauvegardée.

28- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Architecture logicielle

S. Mancini

Données standardisées gérées par TSSA



30- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Architecture logicielle

S. Mancini

Description de l'application

```

// Initialisation de chaque module
...
// Initialisation des connexions
iodescl = fct ( taille paquet, etc ....)

// Connexion des modules
digSetup ->outputDescriptors[0] = iodescl;
procSetup ->inputDescriptors[0] = iodescl;
...

// Lancement de l'application
tmolDigitizerStart( digInst );
tmolProcessStart ( procInst );
tmolRendererStart ( rendInst );
...
  
```

32- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Architecture logicielle

S. Mancini

Plan

- Contexte
- Set top Box PNX8500
 - Environnement
 - Les processeurs
 - Bus, mémoire et 3D
 - Architecture logicielle
 - Exemple d'application
 - Le circuit
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

33- Architecture des SoC- Etudes de cas- Multimédia

Objectifs

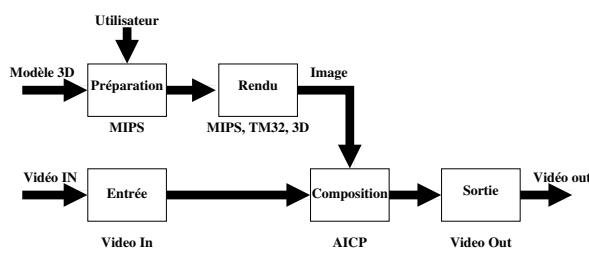
Une image 3D animée incrustée dans un flux vidéo.



34- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Exemple d'application

S. Mancini

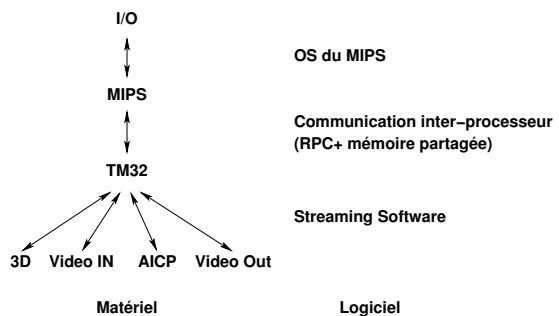
Organisation de l'application



35- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Exemple d'application

S. Mancini

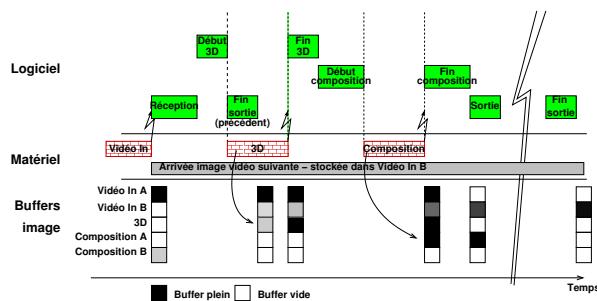
Organisation du contrôle



36- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Exemple d'application

S. Mancini

Déroulement de l'application



37- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Exemple d'application

S. Mancini

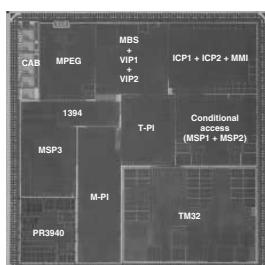
Plan

- Contexte
- Set top Box PNX8500
 - Environnement
 - Les processeurs
 - Bus, mémoire et 3D
 - Architecture logicielle
 - Exemple d'application
 - Le circuit
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

38- Architecture des SoC- Etudes de cas- Multimédia

Le circuit

- ★ 35 Millions de transistors
- ★ Techno 0.18 μm
- ★ 82 Domaines d'horloges
 - TM32 : 200 MHz
 - PR3940 : 150 MHz
 - SDRAM : 143 MHz
- ★ 243 Mémoires : 750 Kbits
- ★ Les périphériques sont regroupés :
 - M-PI, T-PI
- ★ Les blocs sont connectés par boutement



39- Architecture des SoC- Etudes de cas
Multimédia- Set top Box PNX8500- Le circuit

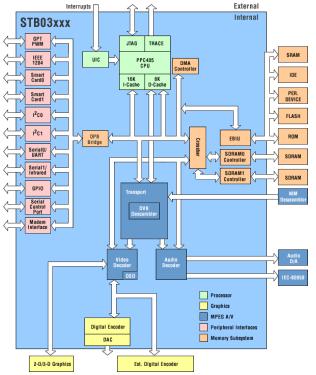
S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

40- Architecture des SoC- Etudes de cas- Multimédia

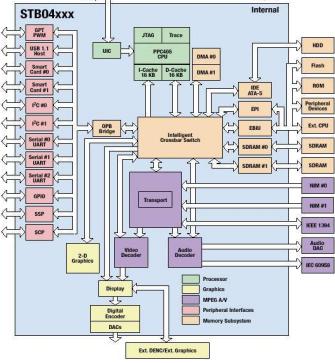
Set top box IBM STB03xxx



41- Architecture des SoC- Etudes de cas Multimédia- Set top box IBM STB03xxx

S. Mancini

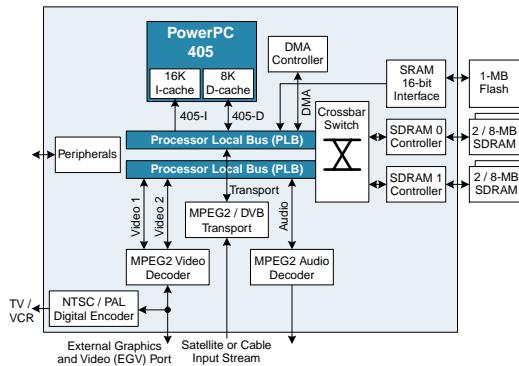
Set top box IBM STB04xxx



42- Architecture des SoC- Etudes de cas Multimédia- Set top box IBM STB04xxx

S. Mancini

Architecture mémoire



43- Architecture des SoC- Etudes de cas Multimédia- Set top box IBM STB03xxx

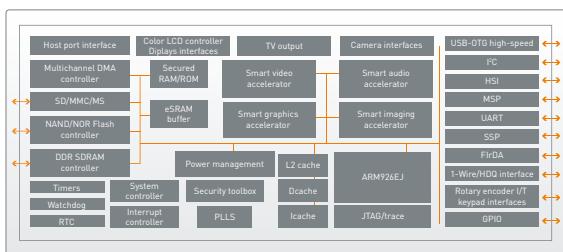
S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia “portable”
- Playstation 2 & 3
- GPU

44- Architecture des SoC- Etudes de cas- Multimédia

Nomadik vidéo portable



45- Architecture des SoC- Etudes de cas Multimédia- Plateforme multimédia “portable”

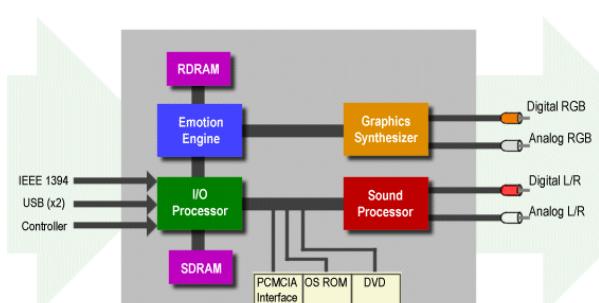
S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia “portable”
- Playstation 2 & 3
 - PS2 : contexte
 - EE : architecture matérielle
 - EE : architecture logicielle
 - EE : le circuit
 - PS3 & Cell
- GPU

46- Architecture des SoC- Etudes de cas- Multimédia

Architecture générale



47- Architecture des SoC- Etudes de cas Multimédia- Playstation 2 & 3- PS2 : contexte

S. Mancini

Performances

Système tri-processeurs à 250 MHz

- 1 processeur 128 bits
- 2 processeurs vectoriels
- 6.2 Giga Flops

Architecture mémoire mixte partagée/distribuée

- Débit à la mémoire principale = 500 MB/s
- Débit interne = 2 GB/s

Performances 3D

- 75 Millions de polygones par seconde
- 2,4 GPixels/s

48- Architecture des SoC- Etudes de cas Multimédia- Playstation 2 & 3- PS2 : contexte

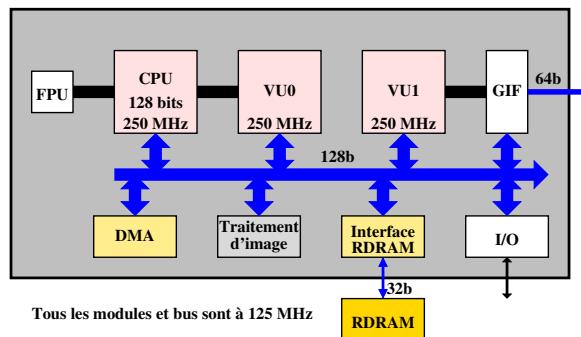
S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia “portable”
- * Playstation 2 & 3
 - PS2 : contexte
 - ✓ EE : architecture matérielle
 - EE : architecture logicielle
 - EE : le circuit
 - PS3 & Cell
- GPU

49- Architecture des SoC- Etudes de cas- Multimédia

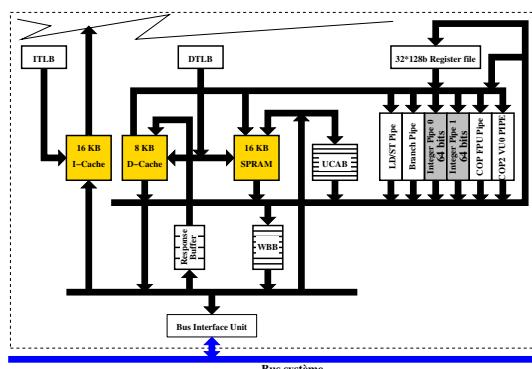
Vue d'ensemble



50- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3 - EE: architecture matérielle

S. Mancini

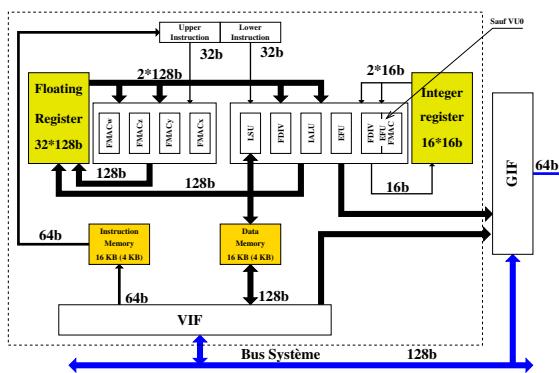
Processeur principal



51- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3 - EE: architecture matérielle

S. Mancini

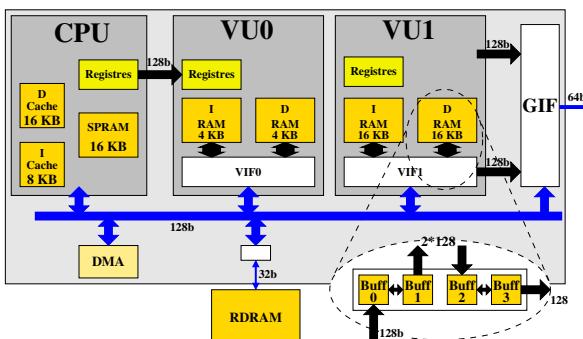
Architecture de VU1 (et VU0)



52- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3 - EE: architecture matérielle

S. Mancini

Architecture mémoire

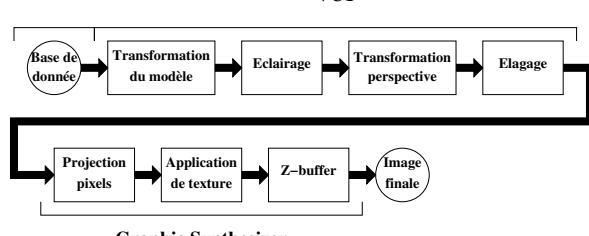


53- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3 - EE: architecture matérielle

S. Mancini

Partitionnement de la 3D

CPU+VU0



VU1

Graphic Synthesizer

54- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3 - EE: architecture matérielle

S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia “portable”
- * Playstation 2 & 3
 - PS2 : contexte
 - ✓ EE : architecture matérielle
 - ✓ EE : architecture logicielle
 - EE : le circuit
 - PS3 & Cell
- GPU

55- Architecture des SoC- Etudes de cas- Multimédia

Modèle de programmation

VPU0 peut fonctionner soit comme un coprocesseur du CPU soit de façon autonome.
VPU1 fonctionne exclusivement de façon autonome.

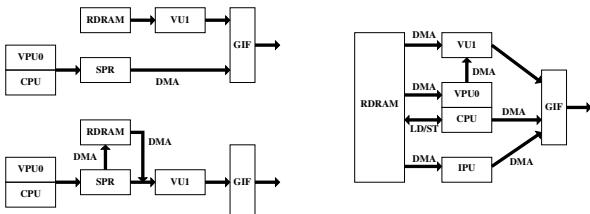
Le VIF de chaque VU reçoit des paquets de données à stocker soit dans la mémoire programme soit dans la mémoire donnée associée, selon un code d'identification. Ce code peut indiquer l'adresse d'un programme à lancer. tV

Les programmes destinés aux VPU0 et VPU1 sont écrits en assembleur.

56- Architecture des SoC- Etudes de cas- Multimédia- Playstation 2 & 3 - EE: architecture logicielle

S. Mancini

Flot de données



57- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3- EE: architecture logicielle

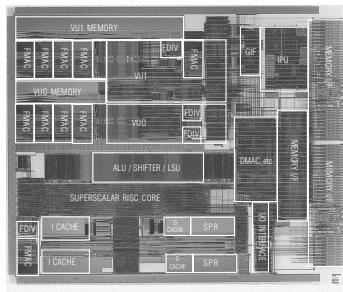
S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
 - PS2 : contexte
 - EE : architecture matérielle
 - EE : architecture logicielle
 - EE : le circuit
 - PS3 & Cell
- GPU

58- Architecture des SoC- Etudes de cas- Multimédia

Le circuit



- 10,5 Millions de transistors
- 17x14.1 mm² en 0.18 µm
- 15 W
- 250 MHz (CPU, VU0, VU1)
- 125 MHz

59- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3- EE: le circuit

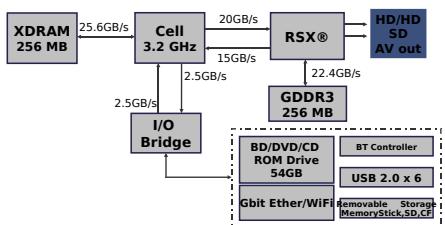
S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
 - PS2 : contexte
 - EE : architecture matérielle
 - EE : architecture logicielle
 - EE : le circuit
 - PS3 & Cell
- GPU

60- Architecture des SoC- Etudes de cas- Multimédia

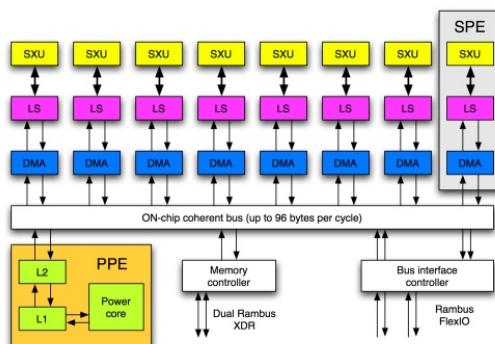
Architecture de la PS3



61- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3- PS3 & Cell

S. Mancini

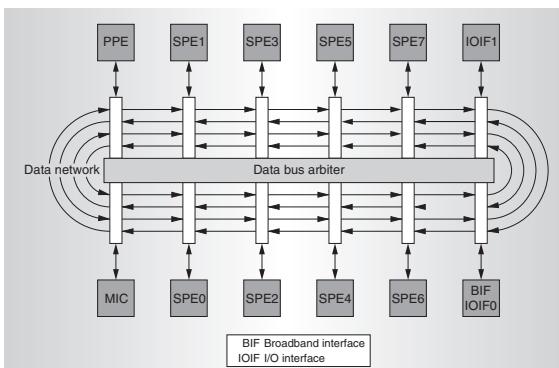
Cell : schéma bloc



62- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3- PS3 & Cell

S. Mancini

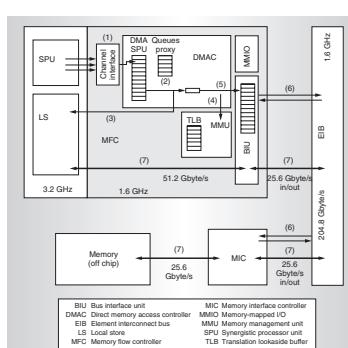
Cell : EIB



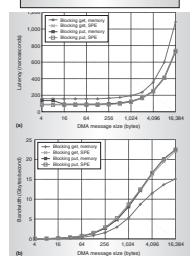
63- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3- PS3 & Cell

S. Mancini

Cell : performance DMA



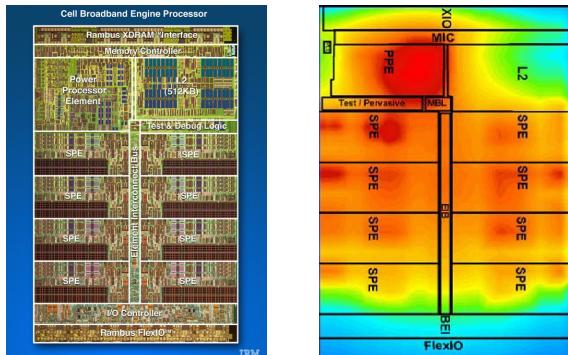
Latency component	Cycles	Nanoseconds
DMA to EIB	30	0.375
Copy to EIB	60	0.750
Coherence protocol	100	1.250
Data transfer for inter-SPE put	140	1.875
Copy	200	2.500



64- Architecture des SoC- Etudes de cas
Multimédia- Playstation 2 & 3- PS3 & Cell

S. Mancini

Cell : le circuit



65- Architecture des SoC- Etudes de cas Multimédia- Playstation 2 & 3- PS3 & Cell

S. Mancini

Plan

- Contexte
- Set top Box PNX8500
- Set top box IBM STB03xxx
- Plateforme multimédia "portable"
- Playstation 2 & 3
- GPU

Objectifs des (GP)-GPU

- ★ Unifier les processeurs qui interviennent dans le pipeline graphique
- ★ Permettre leur programmation pour un usage général

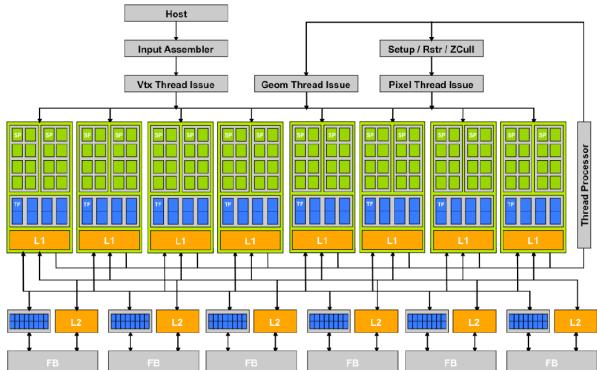
Leur architecture reste conditionnée par le pipeline graphique :

- ★ Parallélisme massif pour le calcul des pixel
- ★ Hiérarchie mémoire adaptée
- ★ Intégration d'opérations systématiques (interpolation et filtrage de texture, pixelisation des triangles, etc ...)

67- Architecture des SoC- Etudes de cas Multimédia- GPU

S. Mancini

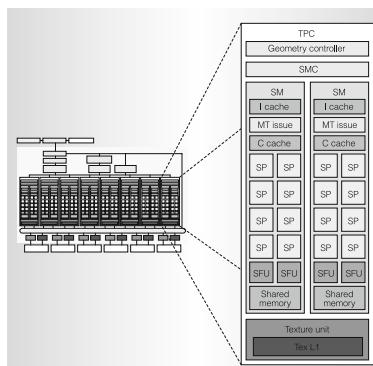
GeForce 8, 9 et GT200 (Tesla) : Architecture globale



68- Architecture des SoC- Etudes de cas Multimédia- GPU

S. Mancini

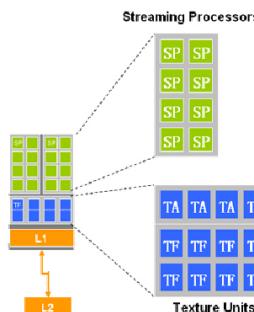
GeForce Tesla : Architecture TPC



69- Architecture des SoC- Etudes de cas Multimédia- GPU

S. Mancini

GeForce Tesla : Architecture SP

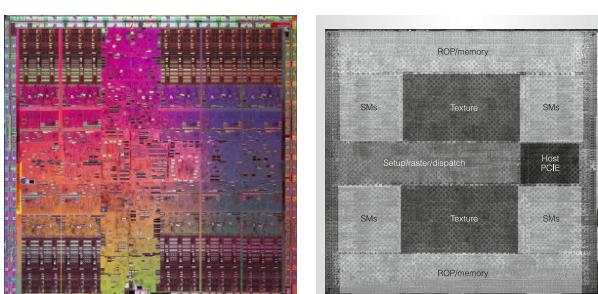


- **SP = Streaming Processors**
- **TF = Texture Filtering Unit**
- **TA = Texture Address Unit**
- **L1/L2 = Caches**

70- Architecture des SoC- Etudes de cas Multimédia- GPU

S. Mancini

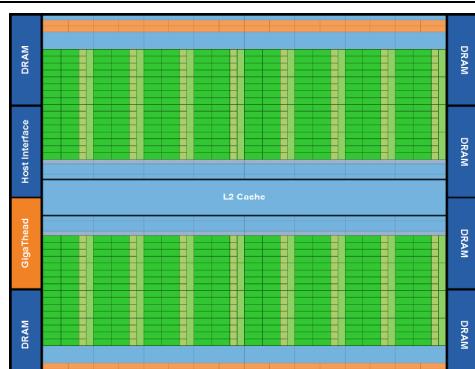
GeForce Tesla : Circuit



71- Architecture des SoC- Etudes de cas Multimédia- GPU

S. Mancini

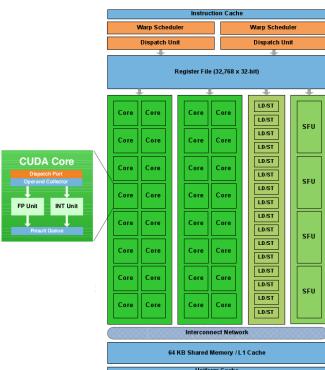
Fermi : Architecture globale



72- Architecture des SoC- Etudes de cas Multimédia- GPU

S. Mancini

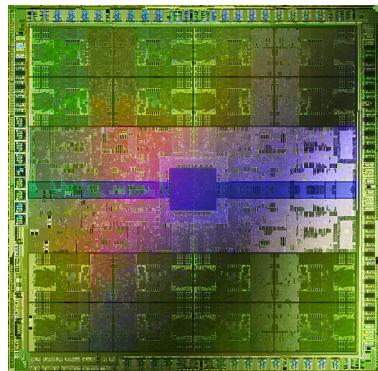
Fermi : Architecture SM



73- Architecture des SoC- Etudes de cas
Multimédia- GPU

S. Mancini

Fermi : Circuit



74- Architecture des SoC- Etudes de cas
Multimédia- GPU

S. Mancini

Architecture des SoC Etudes de cas- Multimédia S. Mancini

Plan Détailié

- ✖ Contexte
 - ❑ Présentation
 - ★ Applications
 - ★ Particularités
 - ❑ Standards vidéo
 - ★ Format vidéo
 - ★ MPEG2
 - ❑ 3D
 - ★ Principe de la 3D Z-buffer
 - ★ Pipeline 3D
 - ★ Performances des accélérateurs 3D
- ✖ Set top Box PNX8500
 - ❑ Environnement
 - ★ Environnement
 - ★ Classe d'application
 - ✖ Architecture générale
 - ❑ Les processeurs
 - ★ MIPS
 - ★ Trimédia- parties opératives
 - ★ Trimédia-Architecture mémoire
 - ★ Trimédia-Mécanismes systèmes
 - ❑ Bus, mémoire et 3D
 - ★ Les bus
 - ★ Architecture des Bus
 - ★ Architecture Mémoire
 - ★ Architecture "Système"
 - ★ Partionnement de la 3D
 - ★ Vidéo
 - ❑ Architecture logicielle
 - ★ Systèmes d'exploitations
 - ★ Streaming Software

- ✖ Données standardisées gérées par TSSA
 - ★ Exemple TSSA simple
 - ★ Description de l'application
- ❑ Exemple d'application
 - ✖ Objectifs
 - ✖ Organisation de l'application
 - ✖ Organisation du contrôle
 - ✖ Déroulement de l'application
- ❑ Le circuit
 - ✖ Architecture de VU1 (et VU0)
 - ✖ Architecture mémoire
 - ✖ Partionnement de la 3D
- ✖ EE : architecture logicielle
 - ✖ Modèle de programmation
 - ✖ Flot de données
- ❑ EE : le circuit
 - ✖ Le circuit
- ✖ Set top box IBM STB03xxx
 - ✖ Set top box IBM STB03xxx
 - ✖ Set top box IBM STB04xxx
 - ✖ Architecture mémoire
- ✖ Plateforme multimédia "portable"
 - ✖ Nomadik vidéo portable
- ✖ Playstation 2 & 3
 - ❑ PS2 : contexte
 - ✖ Architecture générale
 - ✖ Performances
 - ❑ EE : architecture matérielle
 - ✖ Vue d'ensemble
 - ✖ Vue d'ensemble
 - ✖ Processeur principal
- ✖ GPU
 - ✖ Objectifs des (GP)-GPU
 - ✖ GeForce 8, 9 et GT200 (Tesla) : Architecture globale
 - ✖ GeForce Tesla : Architecture TPC
 - ✖ GeForce Tesla : Architecture SP
 - ✖ GeForce Tesla : Circuit
 - ✖ Fermi : Architecture globale
 - ✖ Fermi : Architecture SM
 - ✖ Fermi : Circuit

Architecture des SoC

Etudes de cas Téléphonie mobile

S. Mancini



Plan

- Contexte
- GSM
- WCDMA
- Station de base

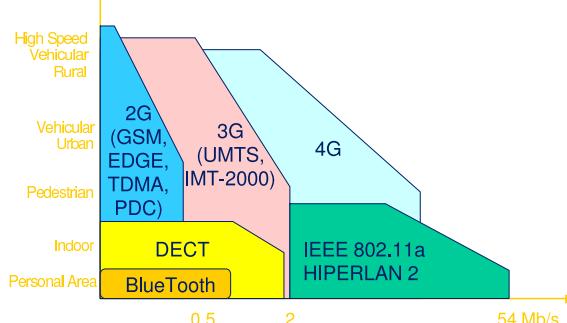
1- Architecture des SoC- Etudes de cas- Téléphonie mobile

Evolution de la téléphonie mobile

Génération	2	2.5	3
Applications	Voix SMS	Internet (WAP) GPS Données	Multimédia Interactivité
Standards	GSM CDMA (IS 95A) PDC	GPRS/EDGE CDMA (IS 95 B)	UMTS CDMA 2000
Produits	Téléphones portables	"Smart Phones" PDA	Multimédia Laptop

S. Mancini

Performances



3- Architecture des SoC- Etudes de cas
Téléphonie mobile- Contexte

S. Mancini

Puissances de calcul

- | | |
|------------------------------------|-----------|
| <input type="checkbox"/> DECT | 10 MIPS |
| <input type="checkbox"/> Bluetooth | 20 MIPS |
| <input type="checkbox"/> GSM | 100 MIPS |
| <input type="checkbox"/> GPRS | 350 MIPS |
| <input type="checkbox"/> EDGE | 1200 MIPS |
| <input type="checkbox"/> UMTS | 5800 MIPS |

S. Mancini

Plan

- Contexte
- GSM
 - Présentation
 - ML 2000
 - Plateformes mobiles multimédia
 - Conclusion
- WCDMA
- Station de base

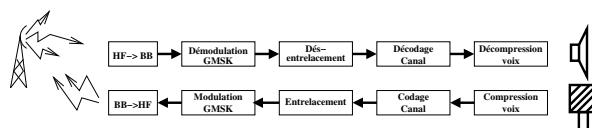
5- Architecture des SoC- Etudes de cas- Téléphonie mobile

Principe général

Bande RF : $\begin{cases} \text{SM vers SB : } 890 - 915 \text{ MHz} \\ \text{SB vers SM : } 935 - 960 \text{ MHz} \end{cases}$

La bande de 24 MHz est découpée en 124 paires de canaux physiques de 200 kHz (FDMA).

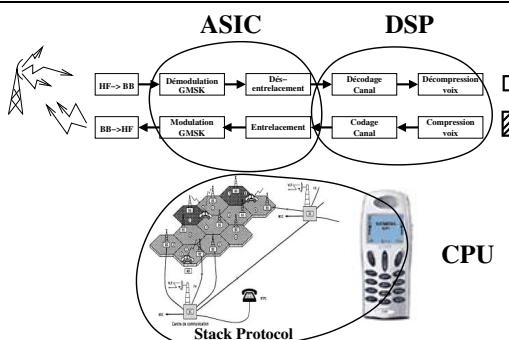
Chaque canal physique est accédé en TDMA selon une organisation temporelle en trames.



6- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- Présentation

S. Mancini

Partitionnement type



La réduction de la **consommation** est un objectif essentiel.

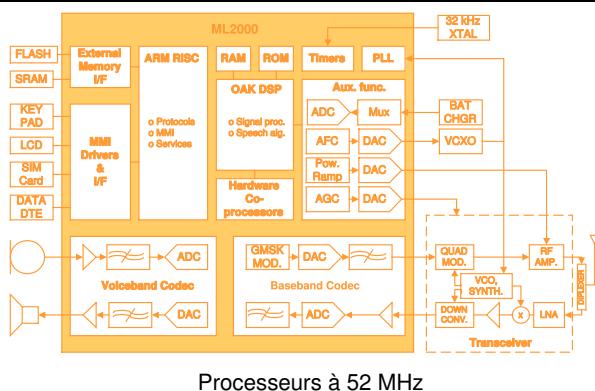
7- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- Présentation

Plan

- Contexte
- GSM
 - Présentation
 - ML 2000
 - Plateformes mobiles multimédia
 - Conclusion
- WCDMA
- Station de base

8- Architecture des SoC- Etudes de cas- Téléphonie mobile

Architecture générale ML2000 (Mobilink)

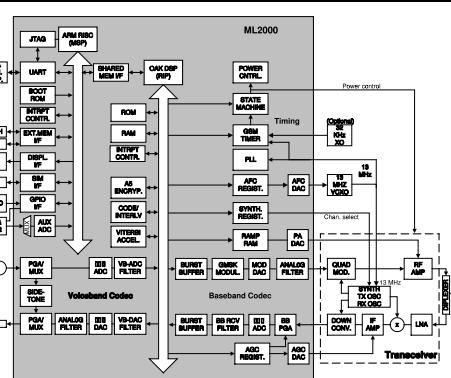


Processeurs à 52 MHz

9- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- ML 2000

S. Mancini

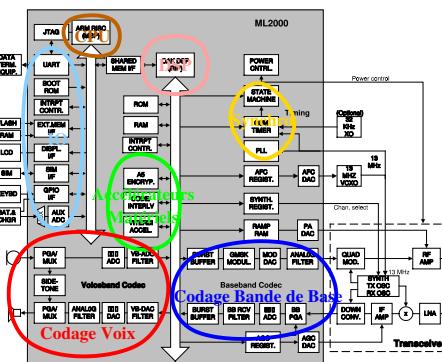
Détails



10- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- ML 2000

S. Mancini

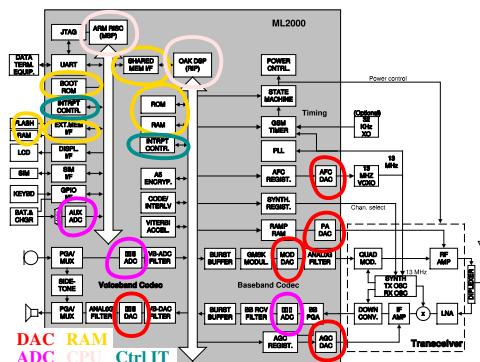
Blocs fonctionnels



11- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- ML 2000

S. Mancini

Modules "système"



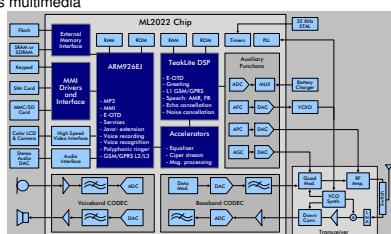
12- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- ML 2000

S. Mancini

Version "Multimédia" et GPRS de ML2000

- ★ ARM
 - 104 MHz
 - Cache
 - Instruction 16K
 - Données 16K
 - 5 Mbits internal SRAM
 - Interfaces multimédia

- ★ DSP
 - 104 MHz
 - Accélérateurs matériels gèrent 4 canaux



13- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- ML 2000

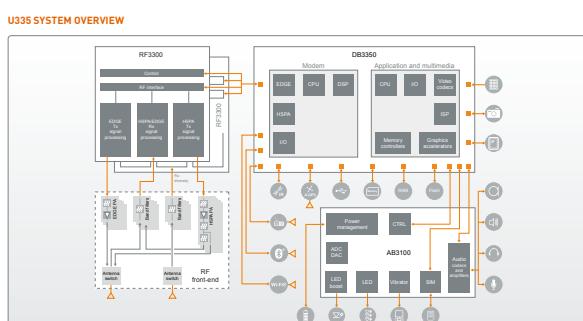
S. Mancini

Plan

- Contexte
- ★ GSM
 - Présentation
 - ML 2000
 - ✓ Plateformes mobiles multimédia
 - Conclusion
- WCDMA
- Station de base

14- Architecture des SoC- Etudes de cas- Téléphonie mobile

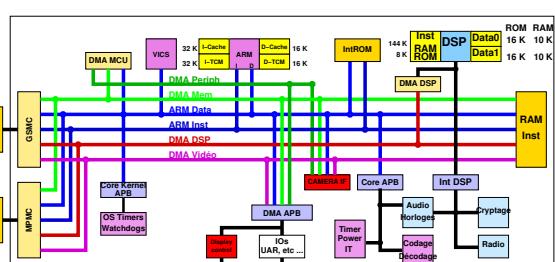
Exemple : nomadik



15- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- Plateformes mobiles multimédia

S. Mancini

Plateforme GPRS multimédia : détail des bus



16- Architecture des SoC- Etudes de cas
Téléphonie mobile- GSM- Plateformes mobiles multimédia

S. Mancini

Plan

- Contexte
- GSM
 - Présentation
 - ML 2000
 - Plateformes mobiles multimédia
 - Conclusion
- WCDMA
- Station de base

17- Architecture des SoC- Etudes de cas- Téléphonie mobile

Conclusion

L'objectif essentiel de l'architecture des plateformes de téléphonie mobile est la réduction de la **consommation** :

- Mise en oeuvre d'un système de veille
- Réalisation d'accélérateurs matériels

La **conception mixte** numérique-analogique devient incontournable.

- l'analogique RF est encore dans un circuit séparé.

S. Mancini 

Plan

- Contexte
- GSM
- WCDMA
 - Complexité
 - Architecture du récepteur
- Station de base

19- Architecture des SoC- Etudes de cas- Téléphonie mobile

Objectifs

Services supplémentaires : 

Débits envisagés :

- 144 Kbps : Zone rurale
- 384 Kbps : Zone semi-urbaine
- 2 Mbps : Couverture limitée

Gestion réseau :

- Débits multiples
- Débits variables

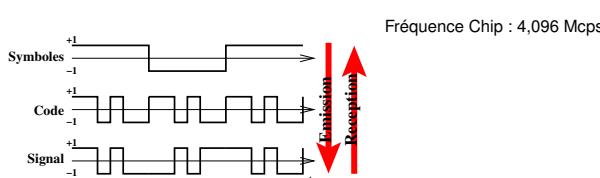
20- Architecture des SoC- Etudes de cas- Téléphonie mobile- WCDMA- Complexité

S. Mancini 

Modulation CDMA

CDMA = Code Division Multiple Access

Le CDMA permet de faire cohabiter plusieurs canaux physiques sur un même bande de fréquence (5 MHz).
L'émetteur et le récepteur partagent un séquence de codage des symboles.



21- Architecture des SoC- Etudes de cas- Téléphonie mobile- WCDMA- Complexité

S. Mancini 

Fonctionnalités d'un terminal 3G

Fonctionnalités :

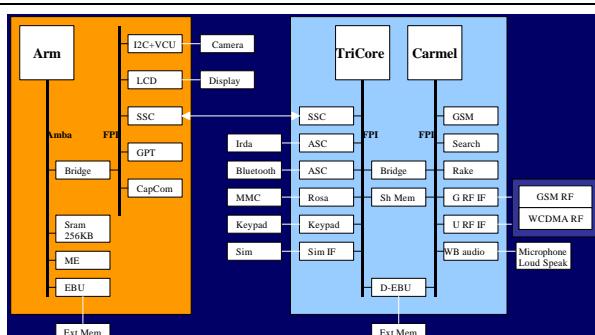
- Multi-band, multi-mode
- BlueTooth, IrDA
- Accès internet
- Images fixes et animées (MPEG4)
- Interface graphique
- Reconnaissance vocal
- Audio (MP3)
- Affichage couleur
- PDA (Personal Digital Assistant)

- Grande puissance de calcul
- Basse consommation
- Mémoire importante

22- Architecture des SoC- Etudes de cas- Téléphonie mobile- WCDMA- Complexité

S. Mancini 

Partitionnement Système



Modem et application sont séparés

23- Architecture des SoC- Etudes de cas- Téléphonie mobile- WCDMA- Complexité

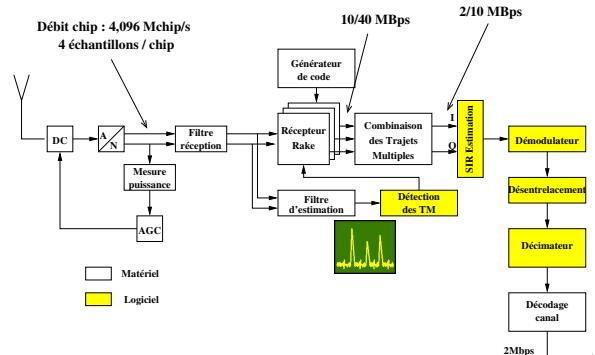
S. Mancini 

Plan

- Contexte
- GSM
- WCDMA
 - Complexité
 - Architecture du récepteur
- Station de base

24- Architecture des SoC- Etudes de cas- Téléphonie mobile

Récepteur mono-utilisateur



25- Architecture des SoC- Etudes de cas
Téléphonie mobile- WCDMA- Architecture du récepteur

S. Mancini

Complexité des récepteurs mobiles

Pour un débit de 384 Kbps

Tâches	MIPS
Filtres numériques	3600
Acquisition et synchronisation	1500
Rake recevoir	650
Combinaison des MP	24
Estimation du canal	12
Contrôle de gain	10
Désentrelacement	14
Turbo décodage	52
Total	5 860

26- Architecture des SoC- Etudes de cas
Téléphonie mobile- WCDMA- Architecture du récepteur

S. Mancini

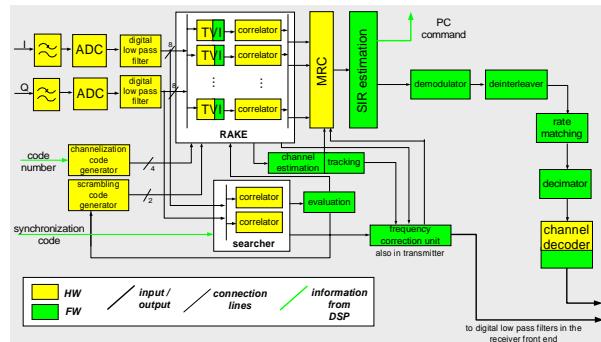
Partitionnement logiciel/matériel

Tâches	Partitionnement
Filtres numériques	HW
Acquisition et synchronisation	HW
Rake recevoir	HW
Combinaison des MP	HW
Estimation du canal	12
Contrôle de gain	10
Désentrelacement	14
Turbo décodage	HW

27- Architecture des SoC- Etudes de cas
Téléphonie mobile- WCDMA- Architecture du récepteur

S. Mancini

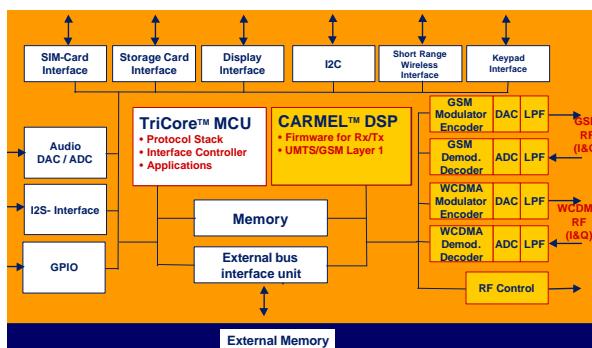
Partitionnement Matériel/Logiciel de M-gold



28- Architecture des SoC- Etudes de cas
Téléphonie mobile- WCDMA- Architecture du récepteur

S. Mancini

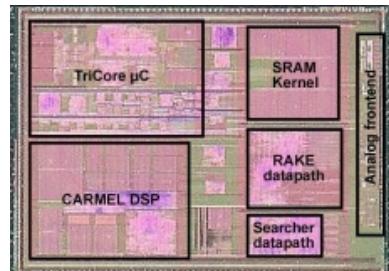
Plateforme UMTS M-gold



29- Architecture des SoC- Etudes de cas
Téléphonie mobile- WCDMA- Architecture du récepteur

S. Mancini

Le circuit



- ❑ 40 Millions de transistors
- ❑ Techno 0.18 μ m
- ❑ Conception mixte numérique/analogique

30- Architecture des SoC- Etudes de cas
Téléphonie mobile- WCDMA- Architecture du récepteur

S. Mancini

Plan

- ❑ Contexte
- ❑ GSM
- ❑ WCDMA
- ✖ Station de base

La station de base doit gérer les différentes stations mobiles dans sa zone de couverture.

La problématique n'est plus de réduire la consommation mais d'avoir assez de puissance de calcul pour gérer jusqu'à une cinquantaine d'utilisateurs.

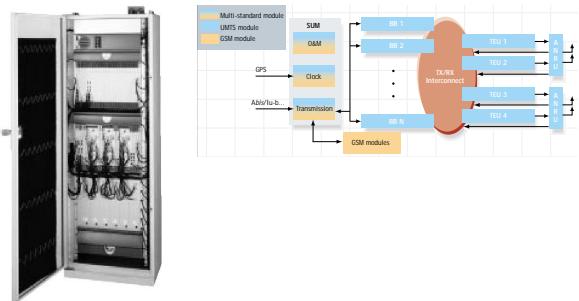
Les économies d'échelle sont plus faibles que pour les stations mobiles : une à 50 stations par Km^2 .

31- Architecture des SoC- Etudes de cas- Téléphonie mobile

32- Architecture des SoC- Etudes de cas- Téléphonie mobile- Station de base

S. Mancini

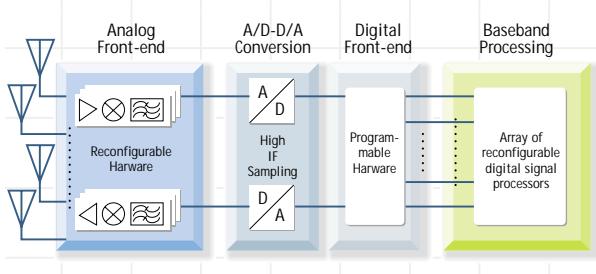
Exemple d'architecture



33- Architecture des SoC- Etudes de cas
Téléphonie mobile- Station de base

S. Mancini

Perspectives : SDR (Software Defined Radio)



34- Architecture des SoC- Etudes de cas
Téléphonie mobile- Station de base

S. Mancini

Architecture des SoC Etudes de cas- Téléphonie mobile S. Mancini

Plan détaillé

- * Contexte
 - ★ Evolution de la téléphonie mobile
 - ★ Performances
 - ★ Puissances de calcul
- * GSM
 - Présentation
 - ★ Principe général
 - ★ Partitionnement type
 - ML 2000
 - ★ Architecture générale ML2000 (Mobilink)
 - ★ Détails
 - ★ Blocs fonctionnels
 - ★ Modules "système"
 - ★ Version "Multimédia" et GPRS de ML2000
 - Plateformes mobiles multimédia
 - ★ Exemple : nomadik
 - ★ Plateforme GPRS multimédia : détail des bus
 - Conclusion
- * WCDMA
 - Complexité
 - ★ Objectifs
 - ★ Modulation CDMA
 - ★ Fonctionnalités d'un terminal 3G
 - ★ Partitionnement Système
 - Architecture du récepteur
 - ★ Récepteur mono-utilisateur
 - ★ Complexité des récepteurs mobiles
 - ★ Partitionnement logiciel/matériel
 - ★ Partitionnement Matériel/Logiciel de M-gold
 - ★ Plateforme UMTS M-gold
 - ★ Le circuit
- * Station de base
 - ★ Objectifs
 - ★ Exemple d'architecture
 - ★ Perspectives : SDR (Software Defined Radio)
- * Conclusion

Architecture des SoC

Etudes de cas Network Processor

S. Mancini

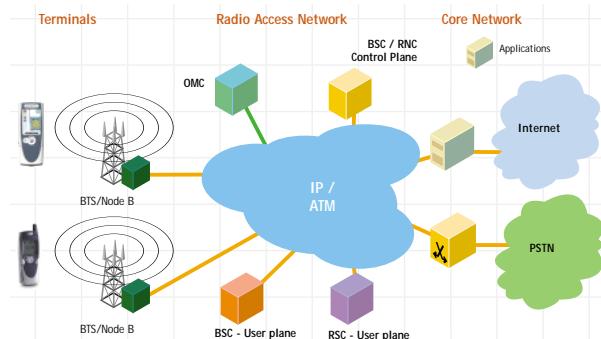


Plan

- ✖ Contexte
- ❑ IXP1200
- ❑ C-Port
- ❑ IBM NP4GS3
- ❑ Conclusion

1- Architecture des SoC- Etudes de cas- Network Processor

Architecture des réseaux



2- Architecture des SoC- Etudes de cas
Network Processor- Contexte

S. Mancini

Couches OSI

7	Application
6	Présentation
5	Session
4	Transport
3	Réseau
2	Liaison
1	Physique

Transcodages entre machines hétérogènes

Ex: ftp avec reprise de transfert

Message complets et découpage des messages en paquets

Routage de paquets de données

Maintien/interruption d'une liaison, correction erreur

Support physique, matériel (cable, etc ...)

3- Architecture des SoC- Etudes de cas
Network Processor- Contexte

S. Mancini

Performances types

La difficulté principale est d'obtenir de grands débits.

Protocole	Débit
xDSL	8 Mbps
Ethernet	100 Mbps
SONET	51 Mbps à 40 Gbps
GSM	10 Kbps
UMTS	2 Mbps

Fonctions principales :

- ★ Gestion de réseau (débits, sûreté)
- ★ Gestion de fluxes
- ★ Routage
- ★ Conversion de protocole

4- Architecture des SoC- Etudes de cas
Network Processor- Contexte

4- Architecture des SoC- Etudes de cas
Network Processor- Contexte

S. Mancini

Objectifs des network-processor

Objectifs

- ★ Réduire le temps de développement.
- ★ Accroître la durée de vie des produits en les rendant programmable.

Solutions

- ★ Utiliser des microprocesseurs
- ★ possédant des fonctions spécifiques
- ★ et des ASIC pour les traitements coûteux

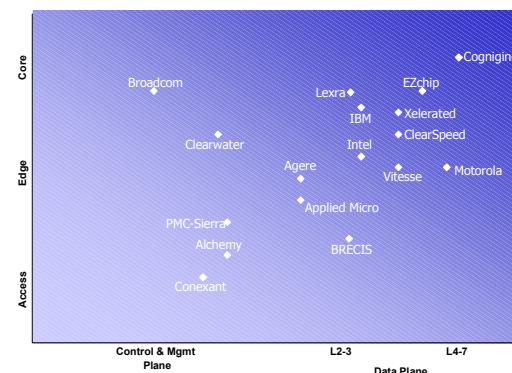
Inconvénients

- ★ Trouver des architectures assez souples.
- ★ Difficulté de programmation.

5- Architecture des SoC- Etudes de cas
Network Processor- Contexte

S. Mancini

Classification des Network Processor



6- Architecture des SoC- Etudes de cas
Network Processor- Contexte

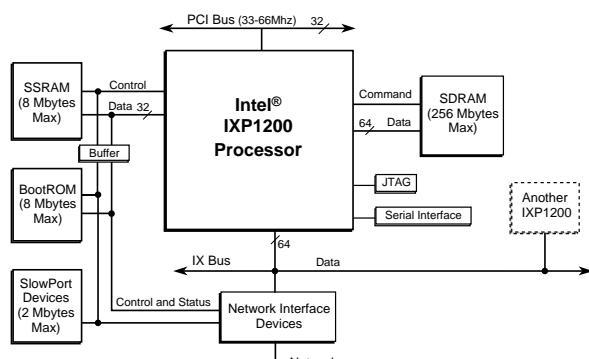
S. Mancini

Plan

- ❑ Contexte
- ✖ IXP1200
 - ✓ Introduction
 - ❑ Modules
 - ❑ Architecture mémoire et bus
 - ❑ Communication inter-modules
 - ❑ Architecture logicielle
 - ❑ IXP2800
 - ❑ Conclusion
- ❑ C-Port
- ❑ IBM NP4GS3
- ❑ Conclusion

7- Architecture des SoC- Etudes de cas- Network Processor

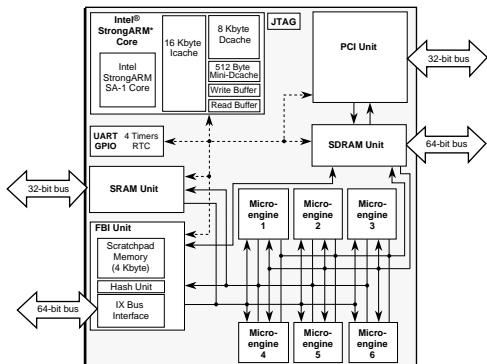
Environnement



8- Architecture des SoC- Etudes de cas- Network Processor- IXP1200- Introduction

S. Mancini

Vue d'ensemble



9- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Introduction

S. Mancini

Plan

- Contexte
- IXP1200
 - Introduction
 - Modules
 - Architecture mémoire et bus
 - Communication inter-modules
 - Architecture logicielle
 - IXP2800
 - Conclusion
- C-Port
- IBM NP4GS3
- Conclusion

10- Architecture des SoC- Etudes de cas- Network Processor

StrongArm

Processeur RISC 32 bits à 232 MHz

Cache	I.: 16 KO	+ 1 KB de Micro-Data cache
	D.: 8 KO	

Cache : lignes de 32 Octets. Cache 32 way set-associative

Le StrongArm est en charge de :

- La synchronisation des micro-engine
- La gestion des flux de données ie les buffers stockés en RAM

11- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

Micro Engine

Ces sont des RISC à 166 MHz capables de gérer **4 tâches** en parallèles.

Spécificités :

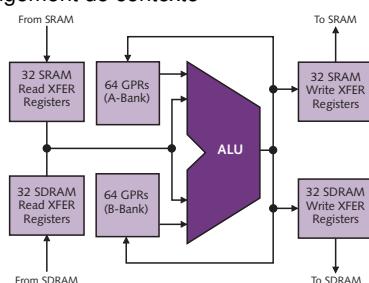
- Gestion du multitâches
 - 4 Compteurs de programmes
 - Commutation de tâche instantanée
 - Adressage des registre global et/ou dépendant du contexte
- Gestion des données
 - Accès mémoires optimisés
 - Organisation spécifique des registres
- Accès directs aux accélérateurs matériels

12- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

Micro Engine registres et accès mémoires

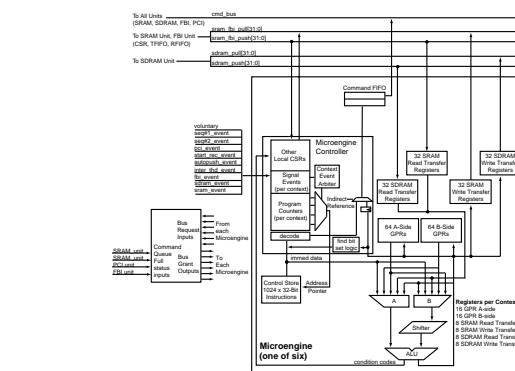
- 256 registres organisés en deux bancs
- La latence des accès mémoire peut être compensée par un changement de contexte



13- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

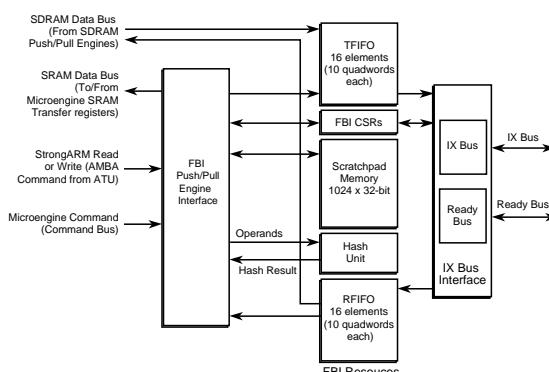
Architecture des Micro Engines



14- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

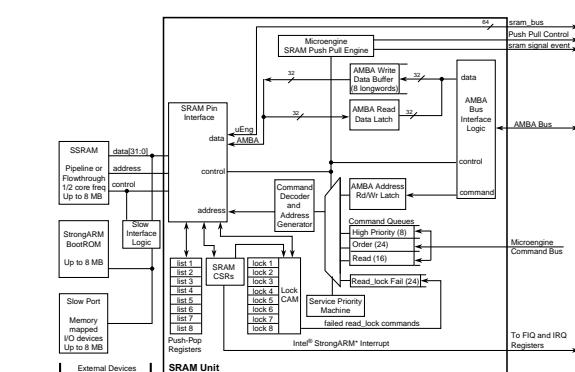
Module FBI



15- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

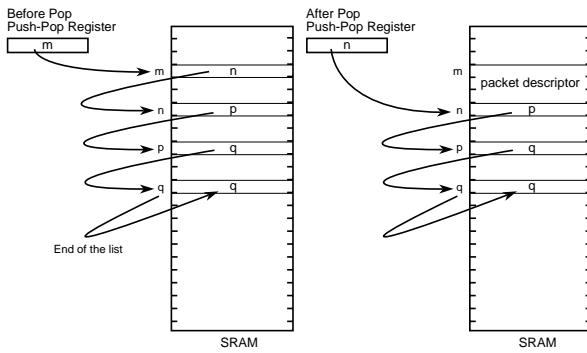
Le contrôleur de SRAM



16- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

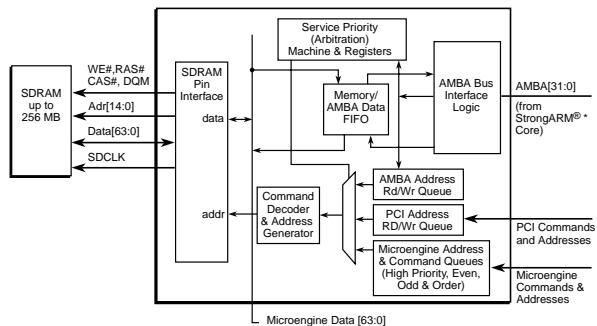
Gestion des listes chaînées



17- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

Le contrôleur SDRAM



18- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Modules

S. Mancini

Plan

□ Contexte

✗ IXP1200

- ❑ Introduction
- ❑ Modules
- ✓ Architecture mémoire et bus**
- ❑ Communication inter-modules
- ❑ Architecture logicielle
- ❑ IXP2800
- ❑ Conclusion

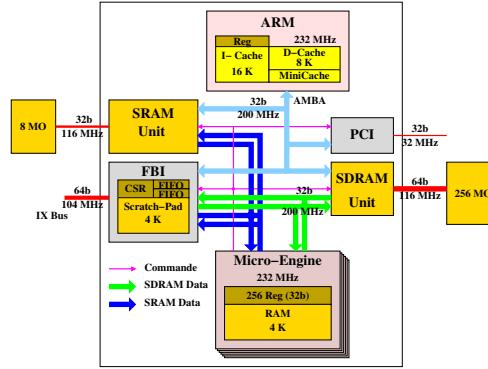
✗ C-Port

✗ IBM NP4GS3

✗ Conclusion

19- Architecture des SoC- Etudes de cas- Network Processor

Architecture mémoire et bus



20- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Architecture mémoire et bus

S. Mancini

Carte mémoire (extrait)

Device 0 PCI UNIT		Device 0 SRAM UNIT	
4000 0000h		Address Range	Description
4000 0000h - 4000 FFFFh	PCI Type 0 Configuration Cycle Access		*Push
5300 0000 - 53FF FFFF	PCI Type 1 Configuration Cycle Access		
4200 0200 - 51FF FFFF	Reserved		
4200 0000 - 4200 0400	Local PCI Configuration Space and PCI Unit CSRs		
4000 0000 - 41FF FFFF	Reserved		
1000 0000h - 1000 FFFFh	SRAM Unit		
0000 0000h - 000F FFFFh			

Address Range Description *Push

Range	Function	Descriptor	Operations	Operations
3800 0000 - 3800 0007	Command FIFO	Pop List	0	2000 0000 2400 0000
3800 0008 - 3800 0028	SRAM CSRs		1	2080 0000 2480 0000
2400 0000 (see Table 1-1)	Pop Command		2	2100 0000 2500 0000
2000 0000 (see Table 1-1)	Push Command		3	2180 0000 2580 0000
1980 0000 - 19FF FFFF	Bit Test & Set		4	2200 0000 2600 0000
1960 0000 - 197F FFFF	Bit Test & Clear		5	2280 0000 2680 0000
1800 0000 - 187F FFFF	Bit Write Clear		6	2360 0000 2760 0000
1600 0000 - 167F FFFF	CAM Unlock		7	2380 0000 2780 0000
1400 0000 - 147F FFFF	Write Lock			
1200 0000 - 127F FFFF	Read Lock			
1000 0000 - 107F FFFF	Read/Write			
0000 0000 - 007F FFFF	BootROM			

*To push descriptor write to base addr + pointer.
(data written is has no effect)

21- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Architecture mémoire et bus

S. Mancini

Plan

□ Contexte

✗ IXP1200

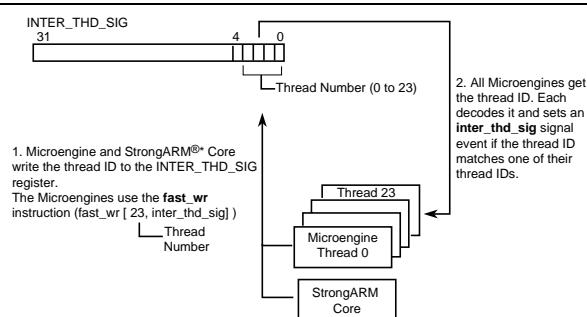
- ❑ Introduction
- ❑ Modules
- ❑ Architecture mémoire et bus
- ✓ Communication inter-modules**
- ❑ Architecture logicielle
- ❑ IXP2800
- ❑ Conclusion

✗ C-Port

✗ IBM NP4GS3

✗ Conclusion

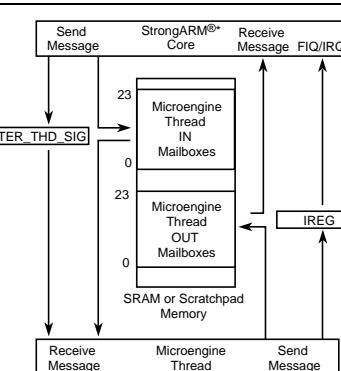
Micro-Engines - Micro-Engines



23- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Communication inter-modules

S. Mancini

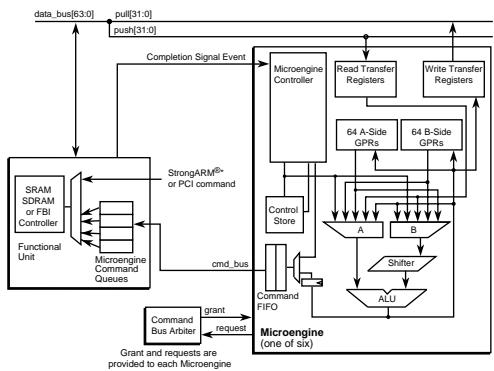
Micro-Engine - ARM



24- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Communication inter-modules

S. Mancini

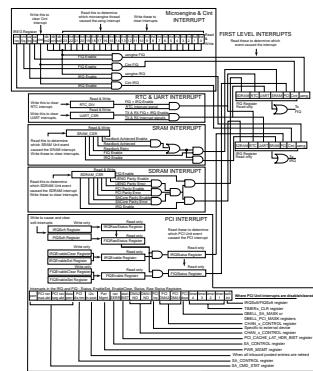
Micro-Engines - Accélérateur



25- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Communication inter-modules

S. Mancini

ARM - Accélérateur



26- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Communication inter-modules

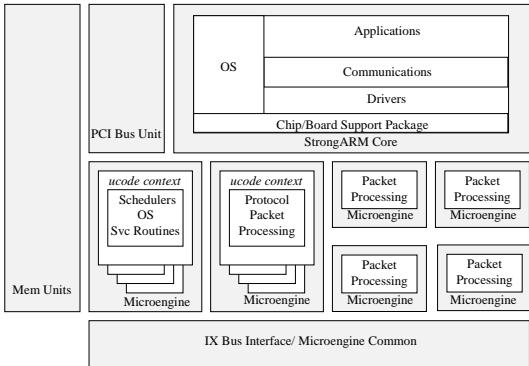
S. Mancini

Plan

- ❑ Contexte
- ✖ IXP1200
 - ❑ Introduction
 - ❑ Modules
 - ❑ Architecture mémoire et bus
 - ❑ Communication inter-modules
 - ✓ Architecture logicielle
 - ❑ IXP2800
 - ❑ Conclusion
- ❑ C-Port
- ❑ IBM NP4GS3
- ❑ Conclusion

27- Architecture des SoC- Etudes de cas- Network Processor

Architecture logicielle



28- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Architecture logicielle

S. Mancini

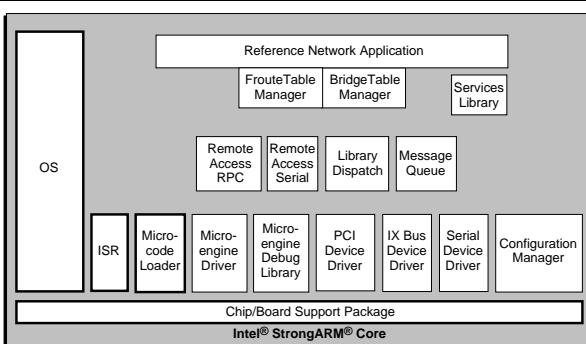
Plan

- ❑ Contexte
- ✖ IXP1200
 - ❑ Introduction
 - ❑ Modules
 - ❑ Architecture mémoire et bus
 - ❑ Communication inter-modules
 - ❑ Architecture logicielle
 - ✓ IXP2800
 - ❑ Conclusion
- ❑ C-Port
- ❑ IBM NP4GS3
- ❑ Conclusion

29- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- Architecture logicielle

S. Mancini

Architecture logicielle de l'ARM



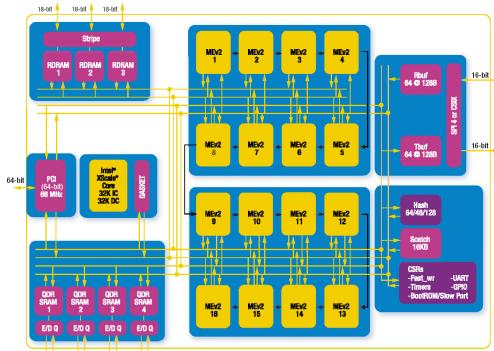
30- Architecture des SoC- Etudes de cas- Network Processor

Plan

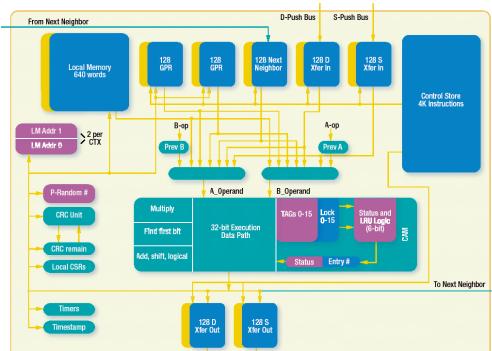
31- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- IXP2800

S. Mancini

Architecture générale



Micro Engine



32- Architecture des SoC- Etudes de cas
Network Processor- IXP1200- IXP2800

S. Mancini

Plan

- Contexte
- * IXP1200
 - Introduction
 - Modules
 - Architecture mémoire et bus
 - Communication inter-modules
 - Architecture logicielle
 - IXP2800
 - ✓ Conclusion
- C-Port
- IBM NP4GS3
- Conclusion

33- Architecture des SoC- Etudes de cas- Network Processor

S. Mancini 

Conclusion

Les caractéristiques principales des processeurs de la famille IXP sont :

- ★ Utilisation de micro-processeurs spécialisés pour les traitements bas niveau
 - Multi-tâche
 - Programmation spécifique
- ★ Accélérateurs matériels
 - Gestion des buffers et listes chaînées
 - Tables de "hash"
 - Communications avec l'extérieur
- ★ Fort couplage micro-engine/ accélérateurs
- ★ Un processeur généraliste de gestion du système

34- Architecture des SoC- Etudes de cas- Network Processor

34- Architecture des SoC- Etudes de cas- Network Processor- IXP1200- Conclusion

S. Mancini 

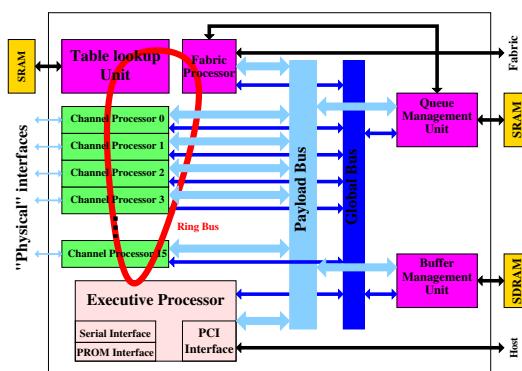
Plan

- Contexte
- IXP1200
- * C-Port
 - ✓ Présentation
 - Processeurs et accélérateurs
 - Les Bus
 - Logiciel
 - Résumé
- IBM NP4GS3
- Conclusion

35- Architecture des SoC- Etudes de cas- Network Processor

S. Mancini 

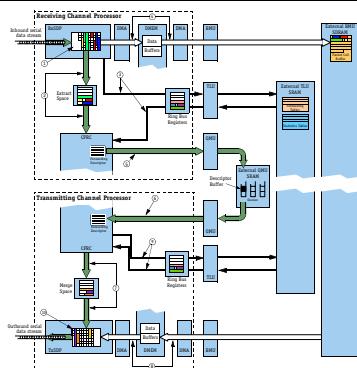
Vue d'ensemble



36- Architecture des SoC- Etudes de cas- Network Processor- C-Port- Présentation

S. Mancini 

Exemple de flux de données



37- Architecture des SoC- Etudes de cas- Network Processor- C-Port- Présentation

S. Mancini 

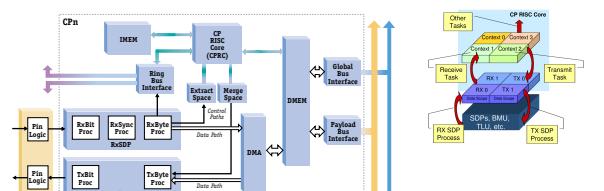
Plan

- Contexte
- IXP1200
- * C-Port
 - ✓ Présentation
 - Processeurs et accélérateurs
 - Les Bus
 - Logiciel
 - Résumé
- IBM NP4GS3
- Conclusion

38- Architecture des SoC- Etudes de cas- Network Processor

S. Mancini 

Processeur de canal

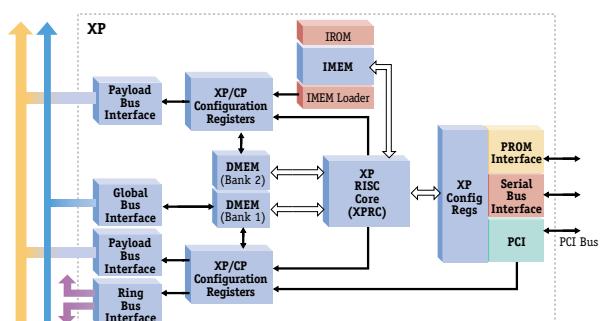


- Architecture Harvard
- IMEM (24K) 4 ports
- DMEM (48K) 4 ports
- 4 jeux de registres indépendants
- 4 espaces mémoires séparés
- Les CPn sont regroupés en "clusters"

39- Architecture des SoC- Etudes de cas- Network Processor- C-Port- Processeurs et accélérateurs

S. Mancini 

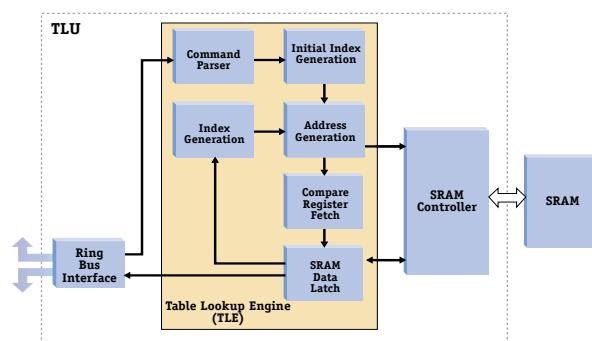
Executive Processor



40- Architecture des SoC- Etudes de cas- Network Processor- C-Port- Processeurs et accélérateurs

S. Mancini 

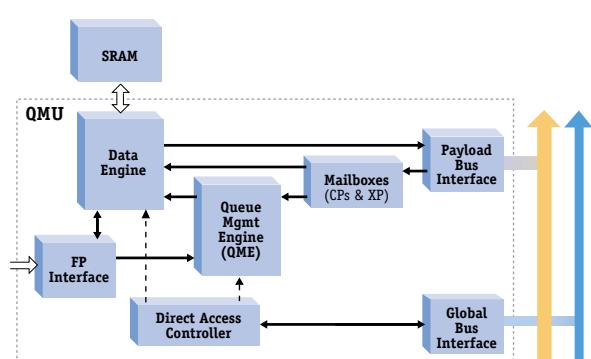
Table Lookup Unit



41- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Processseurs et accélérateurs

S. Mancini

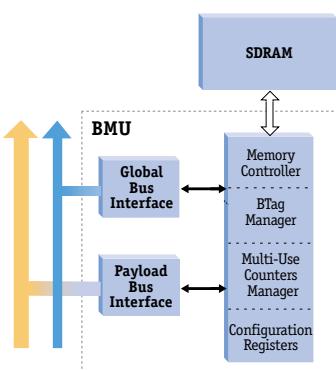
Queue Management Unit



42- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Processseurs et accélérateurs

S. Mancini

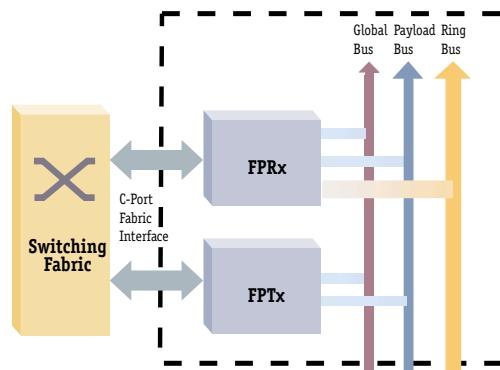
Buffer Management Unit



43- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Processseurs et accélérateurs

S. Mancini

Fabric Processor



44- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Processseurs et accélérateurs

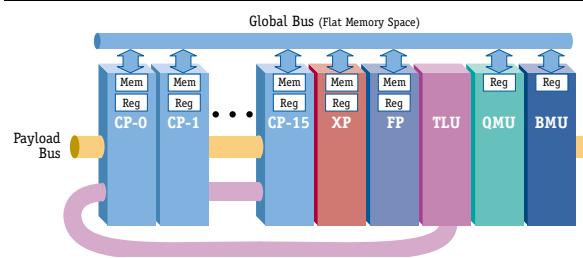
S. Mancini

Plan

- Contexte
- IXP1200
- C-Port
 - Présentation
 - Processeurs et accélérateurs
 - Les Bus
 - Logiciel
 - Résumé
- IBM NP4GS3
- Conclusion

45- Architecture des SoC- Etudes de cas- Network Processor

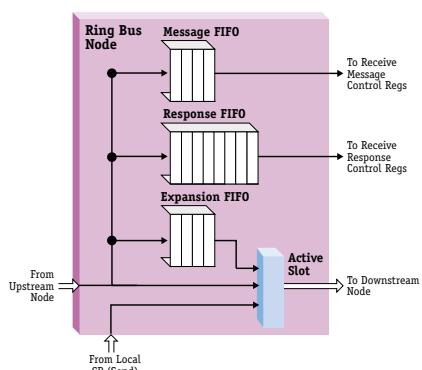
Topologie



46- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Les Bus

S. Mancini

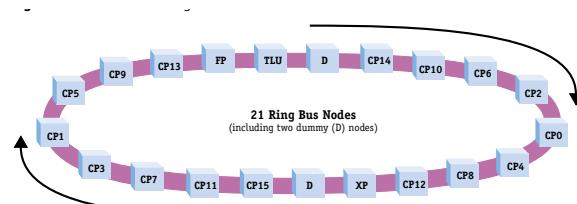
Noeuds du Ring Bus



47- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Les Bus

S. Mancini

Topologie du Ring Bus



48- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Les Bus

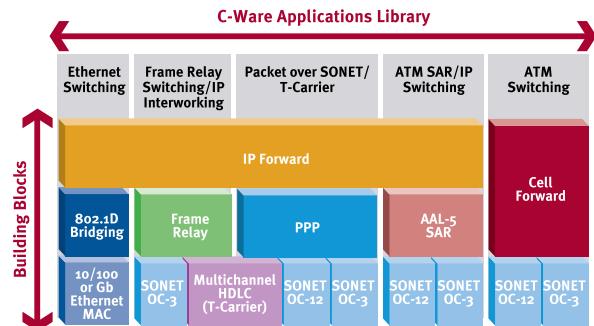
S. Mancini

Plan

- Contexte
- IXP1200
- C-Port
 - Présentation
 - Processeurs et accélérateurs
 - Les Bus
 - Logiciel
 - Résumé
- IBM NP4GS3
- Conclusion

49- Architecture des SoC- Etudes de cas- Network Processor

Architecture Logicielle



50- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Logiciel

S. Mancini

Plan

- Contexte
- IXP1200
- C-Port
 - Présentation
 - Processeurs et accélérateurs
 - Les Bus
 - Logiciel
 - Résumé
- IBM NP4GS3
- Conclusion

51- Architecture des SoC- Etudes de cas- Network Processor

Conclusion

L'architecture du C-port repose sur :

- Un processeur de contrôle global
 - Des processeurs d'entrée/sortie
 - Des opérateurs spécialisés
 - Des bus très haut débit
- Les principales caractéristiques sont :
- La **synchronisation souple** entre CPU et opérateurs
 - ⇒ perte d'efficacité.
 - Les **processeurs de canaux limitent le débit**
connexion directe aux MAC (16*200 Mbps).
 - Les **processeurs sont facilement programmable**
 - ⇒ rapidité de développement et évolutivité

52- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Résumé

S. Mancini

Le circuit

- 56M Transistors 0.18 um CMOS process
- 200 Mhz
- 17 coeurs RISC
- 32 Processeurs de donnée série
- 5 Co-processeurs
- SRAM intégrée

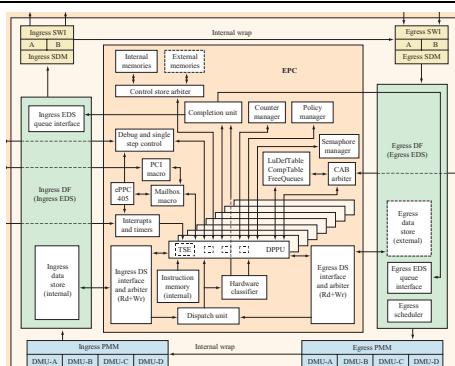
53- Architecture des SoC- Etudes de cas
Network Processor- C-Port- Résumé

Plan

- Contexte
- IXP1200
- C-Port
- IBM NP4GS3
- Conclusion

54- Architecture des SoC- Etudes de cas- Network Processor

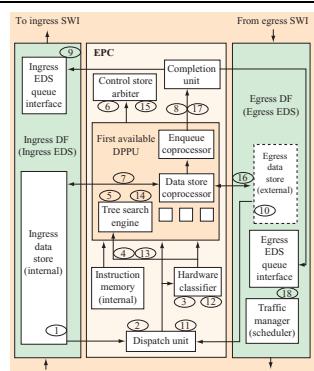
IBM NP4GS3



55- Architecture des SoC- Etudes de cas- Network Processor- IBM NP4GS3

S. Mancini

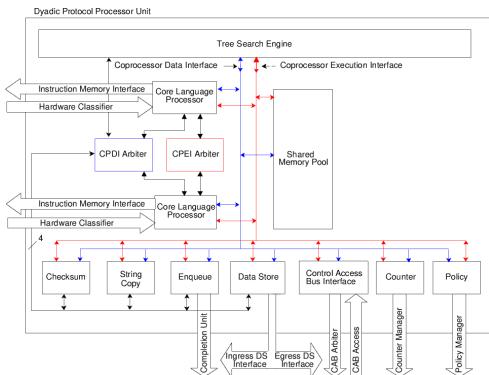
Flot de données



56- Architecture des SoC- Etudes de cas- Network Processor- IBM NP4GS3

S. Mancini

Architecture d'un DPPU



57- Architecture des SoC- Etudes de cas
Network Processor- IBM NP4GS3

S. Mancini

Plan

- Contexte
- IXP1200
- C-Port
- IBM NP4GS3
- Conclusion

58- Architecture des SoC- Etudes de cas- Network Processor

Conclusion

Les network processor sont constitués :

- ❖ de processeurs généralistes,
- ❖ de processeurs spécialisés,
- ❖ et d'accélérateurs matériels.

Les cas étudiés se distinguent par

- ❖ Le niveau de parallélisme
- ❖ Le degré de couplage entre les différents éléments

❖ Cela revient à un compromis entre la facilité d'utilisation et la performance

59- Architecture des SoC- Etudes de cas
Network Processor- Conclusion

S. Mancini

Architecture des SoC Etudes de cas- Network Processor

S. Mancini

Plan Détailé

- Contexte
 - ❖ Architecture des réseaux
 - ❖ Couches OSI
 - ❖ Performances types
 - ❖ Objectifs des network-processor
 - ❖ Classification des Network Processor
- IXP1200
 - Introduction
 - ❖ Nœuds du Ring Bus
 - ❖ Topologie du Ring Bus
 - Modules
 - ❖ StrongArm
 - ❖ Micro Engine
 - ❖ Micro Engine registres et accès mémoires
 - ❖ Architecture des Micro Engines
 - Communication inter-modules
 - ❖ Micro-Engines - Micro-Engines
 - ❖ Micro-Engine - ARM
 - ❖ Micro-Engines - Accélérateur
 - ❖ ARM - Accélérateur
 - Architecture logicielle
 - ❖ Architecture logicielle
 - ❖ Architecture logicielle de l'ARM
 - IXP2800

- ❖ Architecture générale
- ❖ Micro Engine
- Conclusion
- Conclusion
- C-Port
 - Présentation
 - ❖ Vue d'ensemble
 - ❖ Exemple de flot de données
 - Processeurs et accélérateurs
 - ❖ Processeur de canal
 - ❖ Executive Processor
 - ❖ Table Lookup Unit
 - ❖ Queue Management Unit
 - ❖ Buffer Management Unit
 - ❖ Fabric Processor
 - Les Bus
- IBM NP4GS3
 - ❖ IBM NP4GS3
 - ❖ Flot de données
 - ❖ Architecture d'un DPPU
- Conclusion
 - Conclusion