

Conception conjointe matérielle/logicielle. Matériels libres pour l'embarqué

email : kadionik@enseirb-matmeca.fr
web : <http://kadionik.vvv.enseirb-matmeca.fr>

Patrice NOUEL, Patrice KADIONIK
ENSEIRB-MATMECA

ME357 : Conception conjointe matérielle/logicielle



Défit technologique

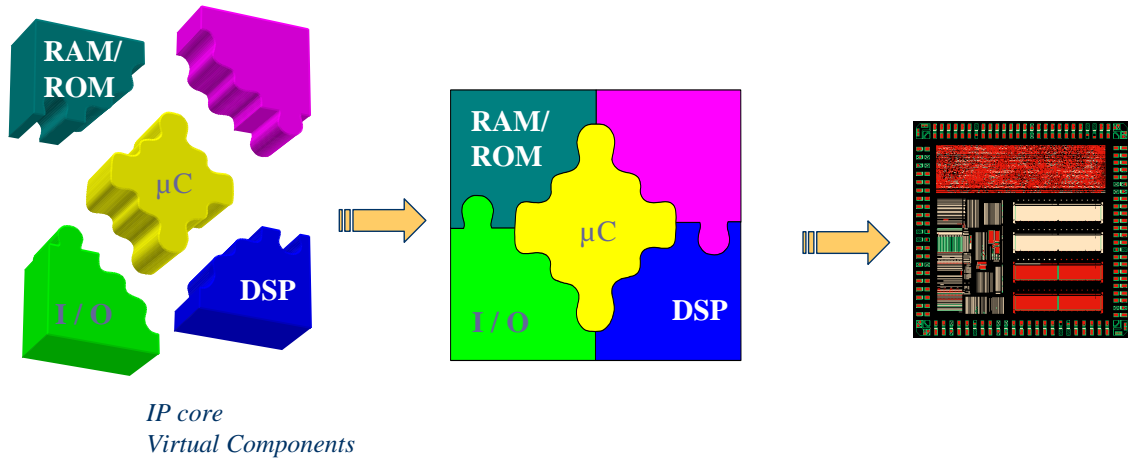
- Intégrer ensemble :
 - 0.10 μm CMOS pour le numérique
 - 0.12 μm Bipolaire pour la RF
 - 0.18 μm DMOS pour la puissance
 - DRAM
 - Mémoires Flash
 - Mems, capteurs, actionneurs électromécaniques

ME357 : Conception conjointe matérielle/logicielle



System on Chip

- Un SoC comporte au moins un processeur :



ME357 : Conception conjointe matérielle/logicielle



IP : composants virtuels

- Pour être capable d'intégrer un système, il faut :
 - Réutiliser les blocs IP déjà conçus (*Design Reuse*)
 - Acheter ou trouver des blocs génériques chez des fournisseurs extérieurs
- Les IP sont des blocs de propriété intellectuelle encore appelés :
 - IP, IP *block*
 - *Core*
 - *System Level block* (SLB)
 - *Macro, System Level Macro* (SLM)
 - *Virtual Component* (VC)

ME357 : Conception conjointe matérielle/logicielle



IP : classification

- **Soft** : le source VHDL ou Verilog doit être synthétisé et optimisé au niveau physique
 - Avantage : indépendance vis à vis du choix technologique
 - Inconvénient : travail d'optimisation peut être long
- **Hard** : description ciblée technologiquement
 - Avantage : performances escomptées
 - Inconvénient : impossible de changer de cible
- **Firm** : Niveau portes (*netlist*)
 - Avantages : flexible et prédictible

Design Reuse

- Méthodologie de conception des circuits
- Règles à respecter pour la conception des blocs IP
- La qualification garantit le respect de ces règles
- VSIA (*Virtual Socket Interface Alliance*) : groupement de sociétés s'ayant donné pour but d'unifier le monde SoC

<http://www.vsi.org>

IP : fournisseurs

- De nombreux fournisseurs de blocs IP (impossible à dénombrer) :
 - <http://www.opencores.org>
 - <http://www.design-reuse.com>
 - <http://www.mentor.com>
 - <http://www.xilinx.com>
 - <http://www.altera.com>

IP : dominantes

- Traitement d'images
- Filtres numériques
- Circuits d'interface : USB, CAN...
- Chiffrement de données
- Processeurs et périphériques
- Communications sans fil
- DSP
- Mémoires

Codesign

- Le *codesign* est un projet mariant une conception conjointe matérielle et logicielle
- On peut être amené dans ce cadre à effectuer des cosimulations (*debugging* d'un programme sur une plateforme matérielle simulée)
- Un projet SoC mêle trois aspects :
 - Architecture matérielle
 - Technologie
 - Logiciel

Outils

- Vérification système. A partir du moment où l'on peut rapidement construire un gros système avec des blocs IP disparates, la difficulté se reporte sur la vérification conjointe matérielle et logicielle. C'est dans ce domaine que les outils se développent le plus à l'heure actuelle.
- Partition logiciel/matériel : définition de l'architecture idéale du système

Les cœurs de processeurs

- Ils sont très nombreux (plusieurs centaines) et sont inventoriés sur le site <http://www.design-reuse.com>
- Le leader dans ce domaine est le processeur ARM. Plus de 50 sociétés possèdent 140 licences ARM (principalement des applications faible consommation).

ARM

ME357 : Conception conjointe matérielle/logicielle



Critères de choix

- Le processeur :
 - RISC ou CISC
 - 8, 16, 32, 64 bits
 - IP *soft* ou *hard*
- Le système de bus
- Les périphériques existantes
- Le système d'exploitation
- Les outils disponibles pour le *codesign* :
 - Implantation du cœur et des blocs IP
 - Vérifications et tests

ME357 : Conception conjointe matérielle/logicielle



Pourquoi utiliser un processeur softcore ?

- Durée de vie du produit
- Indépendance de la technologie
- Prototypage en vue de réaliser un ASIC
- Propriété et maîtrise des sources
- Souplesse d'évolution (VHDL est standard)

Processeurs disponibles

- Cœur PowerPC en dur dans un circuit FPGA Virtex de Xilinx
- *Hardcores* (ARM et MIPS par exemple)
- *Softcores* commerciaux multi-cibles ASIC ou FPGA
- *Softcores* commerciaux à cible propriétaire :
 - NIOS II pour Altera
 - Microblaze et Picoblaze pour Xilinx
 - ..
- *Softcore opensource*
 - LEON Sparc
 - OpenRisc

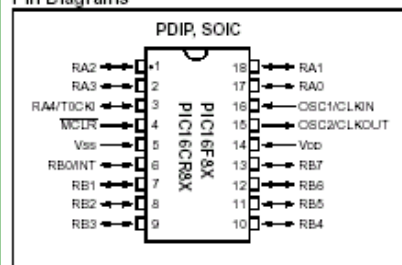
Exemple : cœur P1684

- Projet PPX16mcu (<http://www.opencores.org>)
- 10 fichiers VHDL

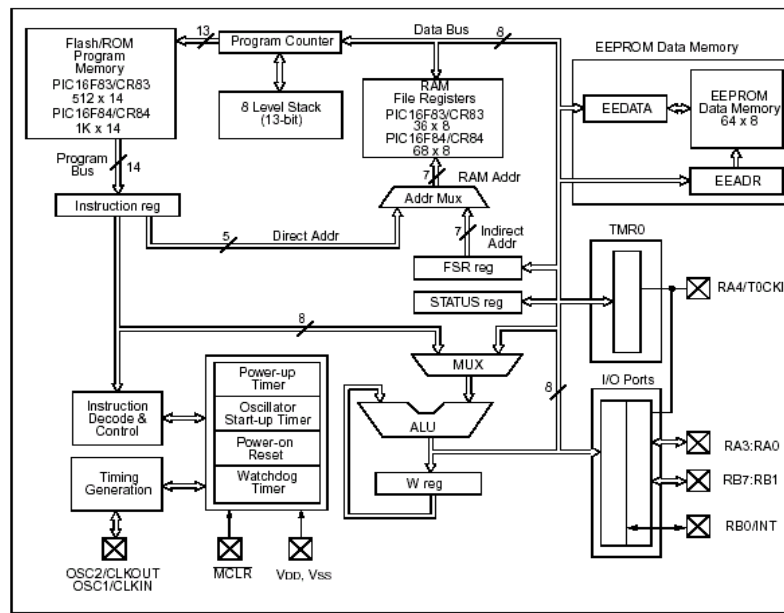
Exemple : cœur P1684

```
ENTITY P16F84 IS
    GENERIC(
        SyncReset : boolean := true);
    PORT(
        Clk : IN std_logic;
        Reset_n : IN std_logic;
        T0CKI : IN std_logic;
        INT : IN std_logic;
        Port_A : INOUT std_logic_vector(7 DOWNTO 0);
        Port_B : INOUT std_logic_vector(7 DOWNTO 0)
    );
END P16F84;
```

Pin Diagrams



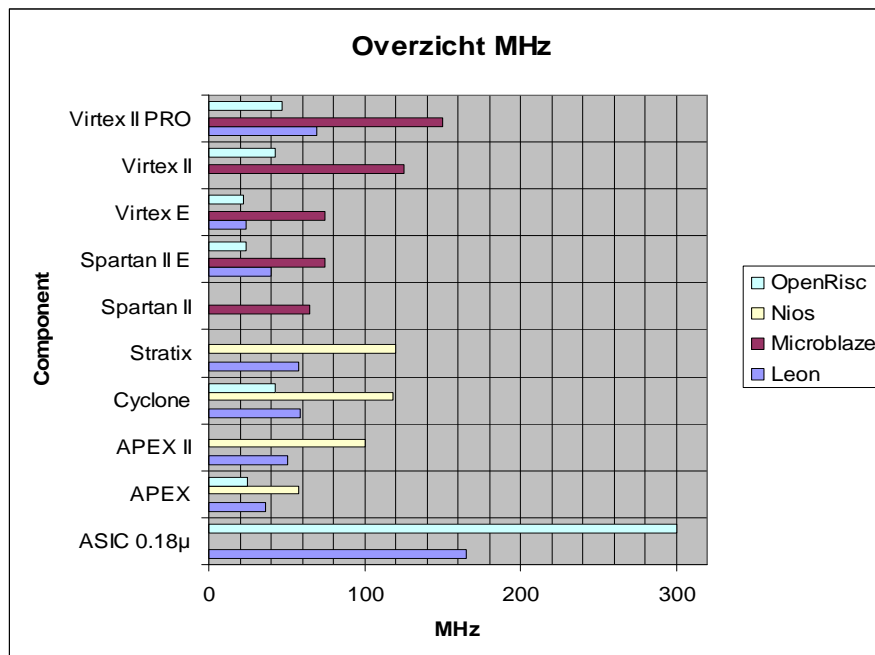
Exemple : cœur P1684



ME357 : Conception conjointe matérielle/logicielle



Performances de processeurs softcore



D'après Patrick Pelgrims De Nayer Instituut

ME357 : Conception conjointe matérielle/logicielle



Performances de processeurs softcore

MicroBlaze™

Device Family	Timing	D-MIPS	MUL.	RAM	Cache [kB]	
					i	d
Virtex II 1000 -4	100 MHz	62	HW	blockRAM	NA	NA
Virtex II 1000 -4	100 MHz	45	SW	blockRAM	NA	NA
Virtex II 1000 -4	50 MHz	4	HW	SDRAM	NA	NA
Virtex II 1000 -4	50 MHz	6	HW	SDRAM	2	2
Virtex II 1000 -4	50 MHz	13	HW	SDRAM	8	8

Nios™

Device Family	Timing	D-MIPS	RAM	Cache [kB]	
				i	d
Altera APEX20k200-2X	33 MHz	13	SRAM	0	0
Altera APEX20k1000E-2X	40 MHz	9	SDRAM	0	0
Altera Cyclone EP1C20	50 MHz	17	SRAM	4	4
Altera Cyclone EP1C20	50 MHz	15	SDRAM	4	4
Altera Stratix EP1S10	50 MHz	17	SRAM	4	4
Altera Stratix EP1S10	50 MHz	15	SDRAM	4	4

OpenRisc 1200 [Richard Herveille]

Device Family	Timing	D-MIPS	RAM
Virtex-E 1600	25 MHz	20	SRAM

LeoSPARC™

Device Family	Timing	D-MIPS	RAM	Cache [kB]	
				i	d
Xilinx Virtex-E 1000E-6	25 MHz	21	SRAM	16	8
Altera APEX20k200-2X	26 MHz	21	SRAM	2	2
Altera APEX20k1000E-2X	20 MHz	4	SDRAM	8	8
Xilinx Virtex II 6000-5	40 MHz	27	SRAM	4	4
Altera Cyclone EP1C20	50 MHz	33	SRAM	4	4

ME357 : Conception conjointe matérielle/logicielle

