

Jean-luc.dekeyser@lifl.fr
Version 2013

ARCHITECTURE ÉVOLUÉE DES ORDINATEURS

12 semaines

cours	TD	TP
1 FPGA		
2 Langage de description	Carte nexys 3	ISE 1
3 VHDL synthésable	vhdl structure du langage	ISE 2
4 Homade	vhdl synthèse de fsm	ISE 3
5 Modèles d'exécution	Homade et son asm	Homade binaire post fixé et IO nexys3
6 Modèles mémoire distri + hiérarchique	SIMD asm loop	Hmd fibo + asm + Tic
7 CC 5 questions bonus à 2 pt	MIMD SPMD + mem distribuée	Hmd RND , en asm + creation d'un lp
8 Modèles de communication	mémoire cache	Hmd jeu rouletteavec switch led et btn
9 Modèles de routage	multi stage + batcher	TP 1 noté sur fibo et roulette
10 Pipe line	multi cast sur grille et multi stage	Hmd spmd 4 slave 1 btn / slave et 2 led / slave
11 Processeur embarqué , RISC	pipe line	TP2 noté sur spmd
12 Supercomputer	multi scalaire et dependance	
	Cray chainage + systolique	

Contrôle et notation

- Examen en janvier
- Une note de contrôle continue
 - 2 TP sur carte nexys 3 (/20) ou une contribution intégrée au processeur Homade (20/20)
 - Une interro un mardi (5 questions bonus 1 point)
- Règle du sup!

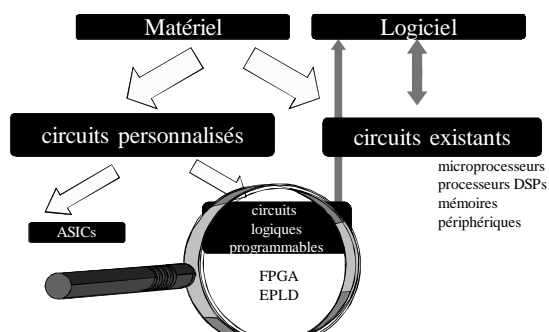
LES FPGAS

Une alternative aux ASICs
(*Application-Specific Integrated Circuit*)

On qualifie les gros **ASIC** de **SoC** (*system-on-chip*, ou système sur silicium), lorsqu'ils intègrent processeur(s), interfaces, mémoires, etc.

http://en.wikipedia.org/wiki/Field-programmable_gate_array

Position du problème



Position du problème

L'intérêt des CLP (Circuit Logique Programmable) se justifie (depuis 30 ans) par :

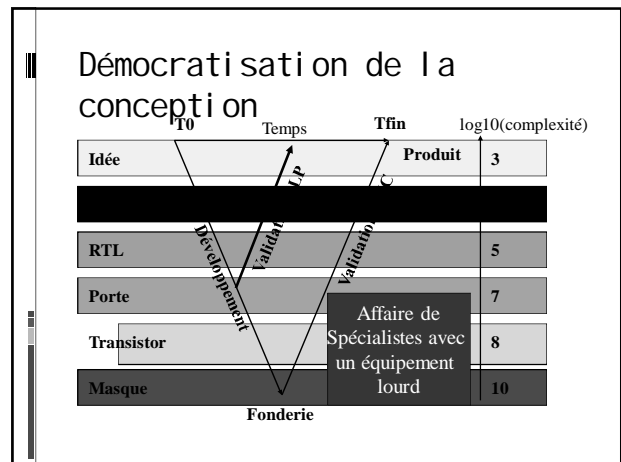
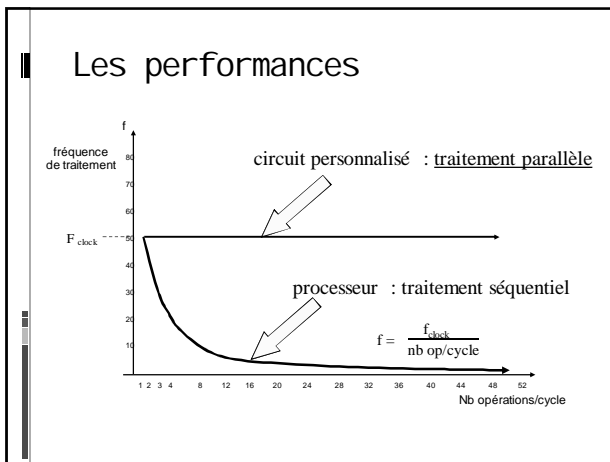
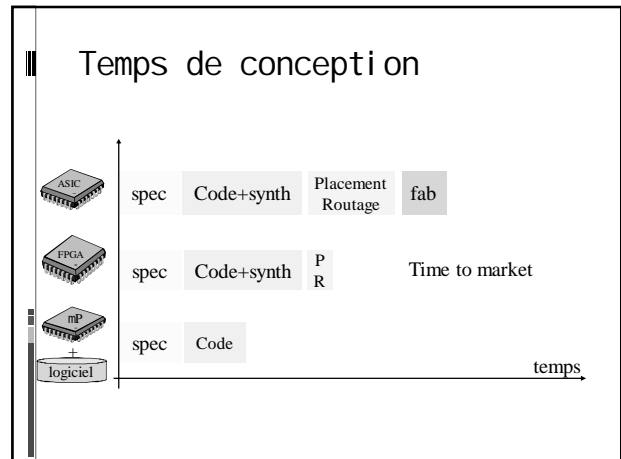
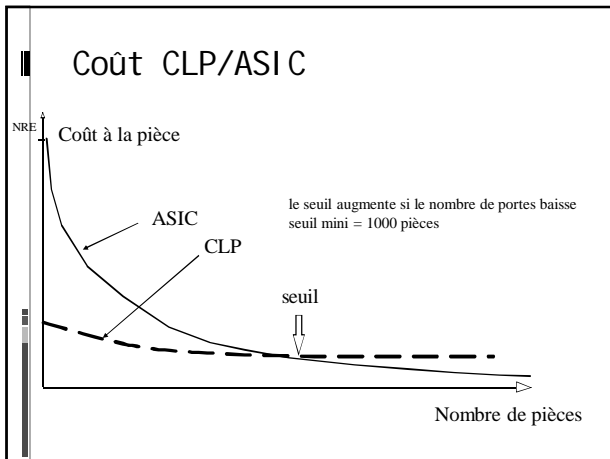
Le coût très important des ASICs

Le temps de développement d'un ASIC

Les performances faibles du logiciel

La non flexibilité des solutions ASIC

« Démocratisation » des FPGA



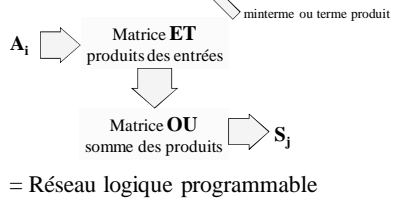
- ### Intérêts propres aux CLPs
- prototypage
 - conception rapide de démonstrateurs
 - autonomie (pas de fondeur)
 - Reprogrammation
 - maintenance in situ
 - Performance de plus en plus proche des processeurs
 - 2 générations de retard sur le taux d'intégration

- ### Comment faire des CLP ?
- Un circuit classique contient :
- des portes logiques ;
 - des connections entre les portes logiques ;
 - des éléments de mémorisation (registre et/ou mémoire) ;
 - des entrées-sorties.
 - une (ou des) horloges
 - une structure hiérarchique
- Le CLP doit donc avoir les mêmes fonctionnalités, avec la notion de « programmabilité ».

Fonction Logique simple : Réseau Logique

Toute fonction logique booléenne peut s'exprimer sous forme canonique

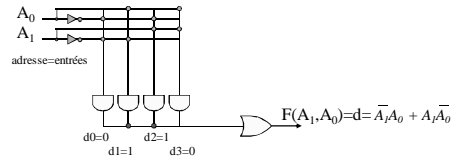
exemple: $S = \bar{A}_2\bar{A}_1\bar{A}_0 + A_2\bar{A}_1\bar{A}_0 + \bar{A}_2A_1A_0 + A_2A_1A_0$



Mémoire

fonction logique 2 entrées : $f(A_1, A_0) = f(0,0)A_1\bar{A}_0 + f(0,1)A_1A_0 + f(1,0)A_1\bar{A}_0 + f(1,1)A_1A_0$

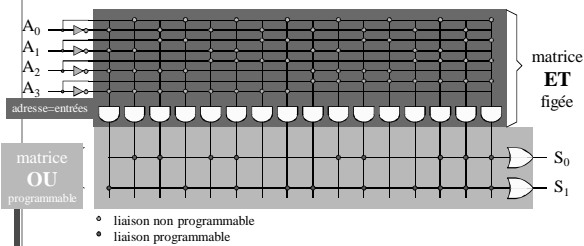
mémoire 4 mots de 1 bit : $d = d_0\bar{A}_1\bar{A}_0 + d_1\bar{A}_1A_0 + d_2A_1\bar{A}_0 + d_3A_1A_0$



- liaison non programmable
- liaison programmable

Mémoire

mémoire 16 mots de 2 bits => 2 fonctions logiques de 4 entrées

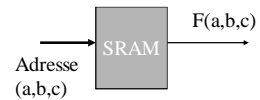


LOOK UP TABLE (LUT)

Une mémoire contient la table de vérité d'une fonction logique :

Ex : Fonction $F(a,b,c) = a \text{ xor } b \text{ xor } c$

000 0
001 1
010 0
011 1
100 0
101 1
110 0
111 1



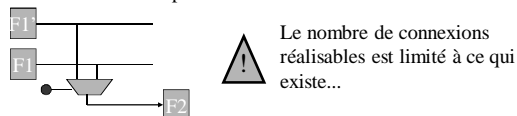
=> Mode de lecture différent du résultat

Programmation des interconnexions

Par connexion ou non de deux fils



Par l'utilisation de multiplexeur



Gestion de la complexité

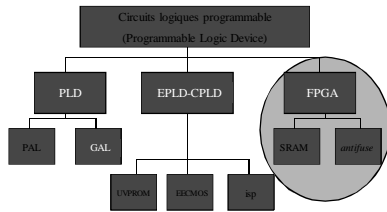
Pour les CLP complexes, il est possible de définir des niveaux de hiérarchie du matériel.

- => Au niveau des interconnexions
- => Au niveau du regroupement de fonctions logiques

Il est aussi possible de diviser le circuit en fonctionnalités différentes :

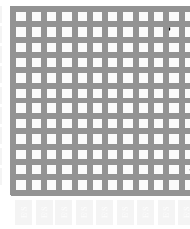
- => Zone de mémoire RAM
- => Zone de CLP
- => Zone spécialisée

Taxinomie des CLPs (PLD)



Les noms peuvent changer selon l'auteur, le fondateur ...

Architecture gate-array (FPGA)



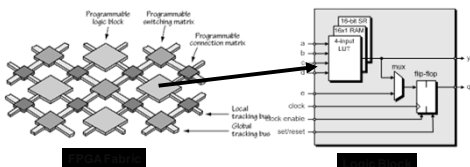
- Ensemble de cellules logiques disposées en matrice et séparées par des canaux de routage organisés en Rangées/Colonnes. A chaque intersection se trouve un commutateur.
- Le temps de propagation dépend du routage choisi pour réaliser l'interconnexion.

canaux de routage

cellule logique

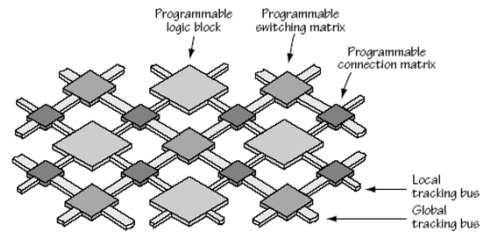
Construction en blocs

- Fonctions logiques dans Lookup Table LUTs
- Multiplexers (selection 1 / N inputs)
- Flip-Flops. Registers. Clocked Storage elements.



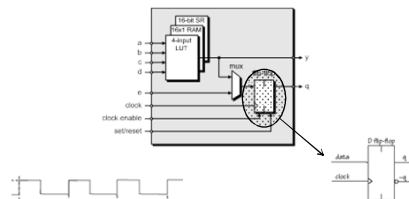
Sur plusieurs blocs

- Connection de plusieurs blocs pour des fonctions logiques plus complexe



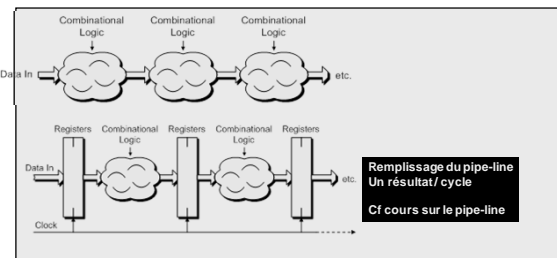
Une logique avec horloge

- Flip Flops outputs. CLOCKED elements.
- Sequential Logic Functions (cf Combinational Logic LUTs)
- Pipelines. Synchronous Logic Design

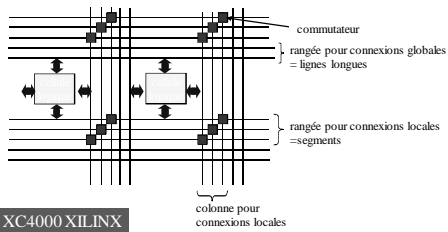


Design Synchronous Logi c

- Un pipeline de fonction logique
- Des registres mémorisent les résultats

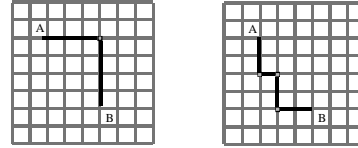


roulage dans un FPGA



exemple : XC4000 XILINX

roulage dans un FPGA



grande adaptabilité du roulage

Comment choisir un FPGA ?

Caractéristiques techniques du FPGA

- Performances (complexité, vitesse, conso.)
- Nombre E/S et interface
- Tension d'alimentation
- Boîtier

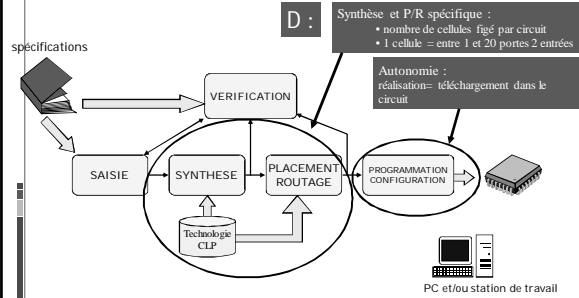
Qualité des softs associés au FPGA (CAO et IP)

- Technique de programmation
- Technique de debug et de test
- Critères économiques

- prix, disponibilité
- pérennité et flexibilité de la famille.

CAO pour conception de CLP

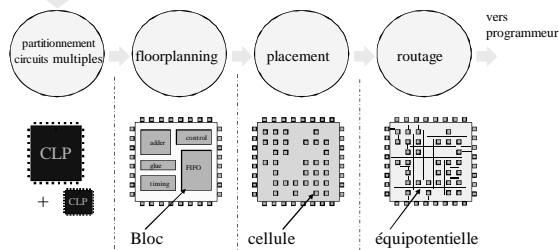
Très similaires à celle des circuits VLSI :



placement routage

Chaque vendeur de CLP fournit son outil de P/R dédié

"Netlist" après synthèse



La qualité des outils

Les FPGAs deviennent très complexes et très flexibles.
=> L'optimisation du P/R devient très complexe
=> Logiciels performants

Or les logiciels de CAO pour les CLPs manquent de maturités :

- => très lents (24 h pour un placement routage)
- => peu stables (machine à rebouter)
- => peu efficaces et non déterministes :
 - a) temps de propagation trop long
 - b) en échec (pas de solution trouvée).
 => il faut alors guider « à la main » le placement

Synthèse sur FPGA

Pour optimiser la synthèse, il faut la guider :

- => Directive globale de compilation :
Effort d'optimisation, critères de synthèse
- => Directive locale de compilation dans le texte
-- « mot clef » « directive compilation »
- => Directive hiérarchique de compilation
tel bloc, telle méthode (insertion libre de registres, méthode A).
- => Utilisation d'opérateurs « optimisés par le fondeur »
appels de macro-blocs additionneurs, multiplieurs,... paramétrables

PROBLEME : Le code VHDL devient spécifique au FPGA

Placement sur FPGA

Problème NP-complet qui conditionne :

- la possibilité de trouver une solution de routage
- le chemin critique

Solution :

- Indiquer « à la main » le placement des blocs dans le floor plan.
- Placement routage incrémental

Assignation manuelle ou automatique des PAD E/S sur les entrées/sorties du composant VHDL.

Intellectual Property (IP)

La qualité d'une CLPs dépend aussi des IPs disponibles pour accélérer la conception (de même que la qualité d'un système d'exploitation dépend des logiciels existant).

- Les structures matérielles deviennent du consommable
- Achat ou location d'un design paramétrable
- Licence ou royalties
- Développement et échange d'IP grâce aux langages standards

IP MegaStore: Search Engine

The Portfolio (XILINX, ALTERA)

Communications	Bus Interface	Digital Signal Processing	Processor, Peripheral
ADPCM (a-law, a-law)	CAN Bus	Adaptive Filter	2910
ATM Controller	IIC Slave	Binary Pattern Correlator	49410
Cell Delimitation	IIC Master	Biorthogonal Wavelet Filter	6402 UART
CRC	IEEE 1394	Complex Multiplier Mixer	6850 UART
Ethernet MAC (10/100 Gigabit)	PCI Target	CSC	16450/550 UARTs
HDLC Protocol Core	PCI Master/Target	Decimating Filter	8251 UART
IMA Controller	PCI-X Master/Target	Digital Modulator	8237 DMA Controller
Intermediate Data Rate Frame/DeFrame	PowerPC Bus Arbiter	Discrete Cosine Transform	8255 Peripheral Interface
Multi-Channel ATM TC	PowerPC Bus Master	Early/Late Gate Symbol	8259 Interrupt Controller
Packet Over SONET Controller	PowerPC Bus Slave	FFT	8254 Timer/Counter
Rank Order Filter	USB Function Controller	FIR Compiler	8051
SpeedBridge	USB Host Controller	FIR Filter Library	8052
Telephony Tone Generator		Floating Point Adder/Divider	6502
Utopia Level II/III Master and Slave		IIR Compiler	Z80
		Image Processing Library	Excalbur NIOS Processor
		Integer Divider	LX-4080R
		NCO	Xtensa 32-bit Processor
		Reed Solomon Compiler	SDRAM Controller
		Square Root Operator	
		Symbol Interleaver/Deinterleaver	
		Viterbi Decoder	

Critère économique

Coût du composant : évidemment...

Coût du développement : dépend de la culture de l'entreprise

- => Savoir faire
- => Réutilisation de code déjà existant
- => Existence ou non des logiciels de CAO (conditionne aussi le temps de développement)

Coût de maintenance :

- => extension de la famille FPGA : nouvelle génération de produit

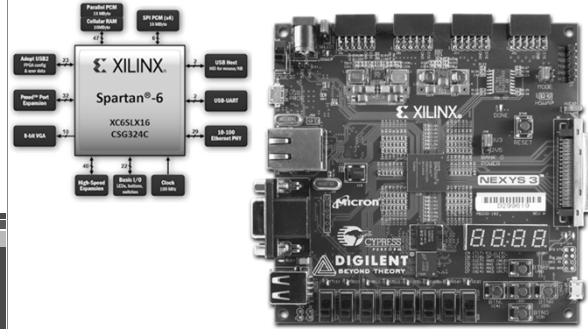
L'environment Xilinx

Outils (Xilinx)

- Spartan-3 Starter Board
- ISE WebPack



La nexys 3



L'environment ISE



Conception

Etapes

- Edition des sources
- Edition du fichier UCF : « User Constraints File »
 - Répartition des signaux de l'entité principale sur des pins
 - Contraintes de timing
 - IO Standards (LVTTTL, LVCMOS33, LVCMOS25...)
- Synthèse et génération du fichier programme
- Génération de l'image EEPROM
- Flashage de l'EEPROM

Le top Virtex 7

