EN201: Technologie des circuits numériques
«Les circuits FPGA»

Bertrand LE GAL
[bertrand.legal@ims-bordeaux.fr]

Filière Electronique - 2ème année
ENSEIRB-MATMECA - Bordeaux INP
Talence, France

Then a miracle occurs...

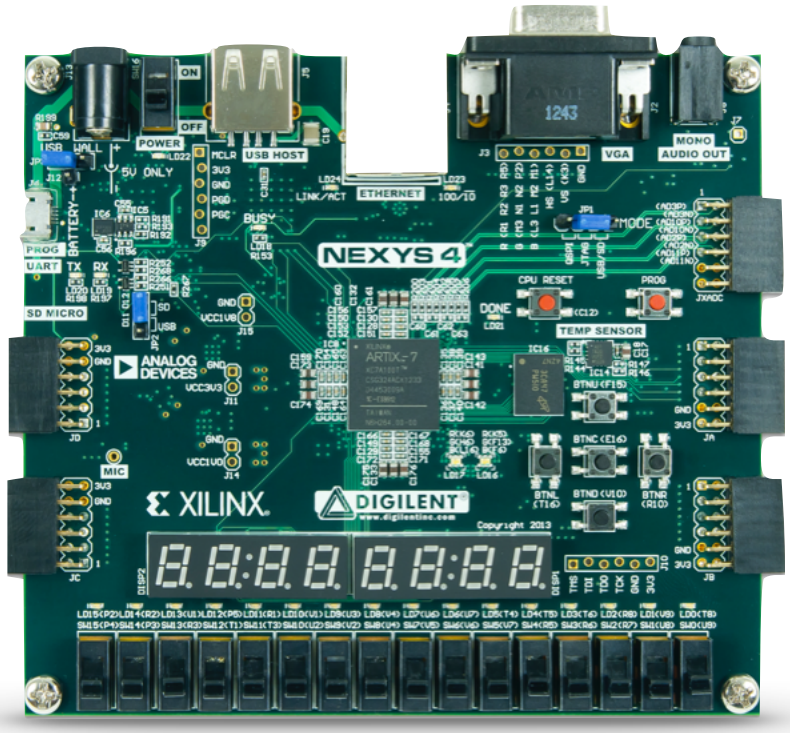
Description VHDL

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;

entity dff is
  port (
    clk : in  std_logic;
    d   : in  std_logic;
    q   : out std_logic
    );
end entity;

architecture arc of dff is
begin
  process (clk)
  begin
    if rising_edge (clk) then
      q <= d;
    end if;
  end process;
end arc;
```

Réalisation sur
FPGA (bitstream)

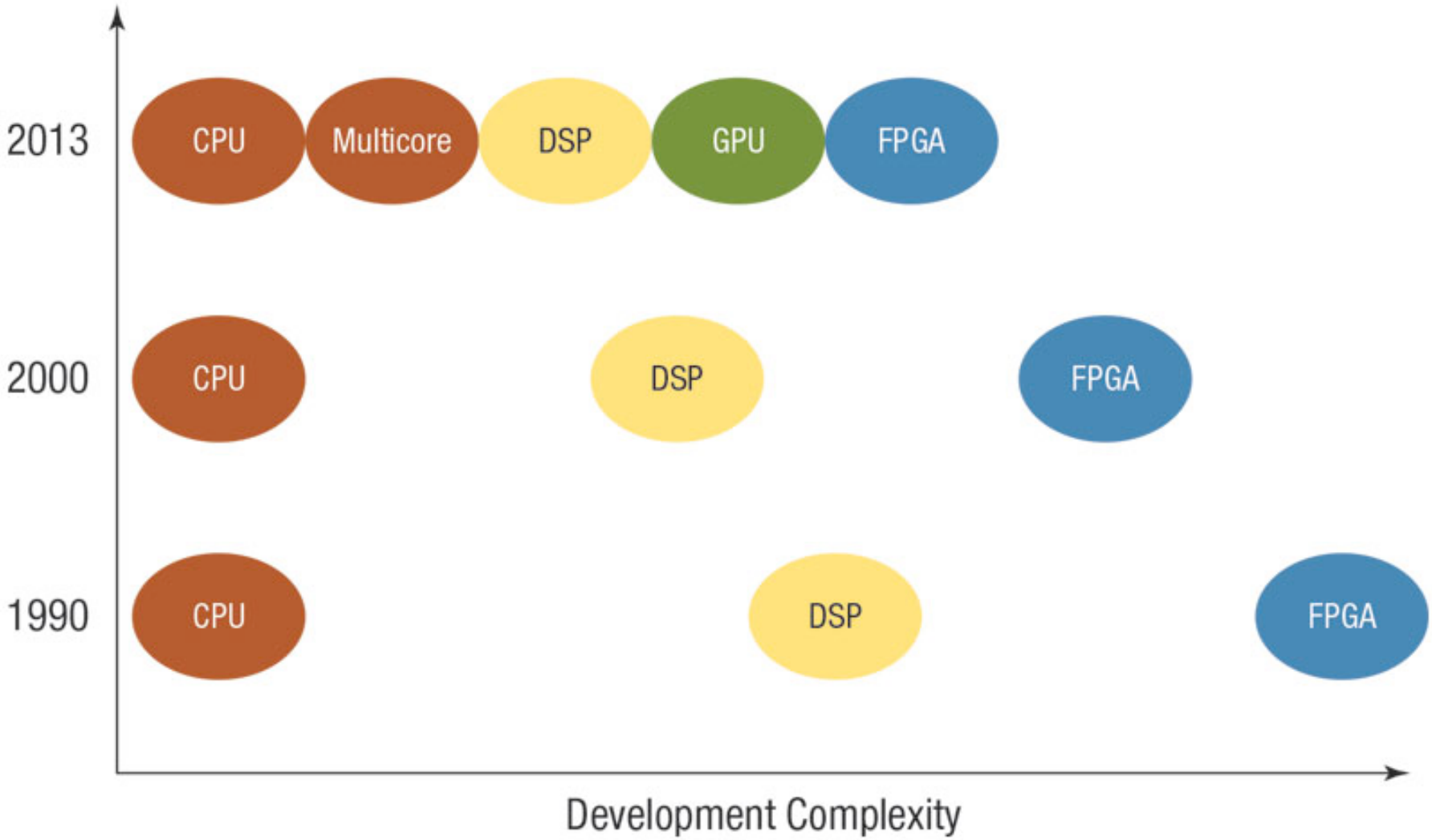
Plan du cours

- Introduction
  - Classification des circuits (re)configurables
  - Evolution des circuits FPGAs,
  - Contexte d'utilisation des FPGAs.
- Architecture de circuit FPGA
  - L'élément configurable de base,
  - Les blocs dédiés (DSP, RAM).
- La configuration d'un circuit FPGA
  - Différentes techniques,
  - Différents types.
- Les fabricants de circuit FPGA
  - Les circuits FPGA chez Xilinx,
  - Les circuits FPGA chez Altera.
- Le flot de conception d'un circuit FPGA
- Les Systems sur Puce Programmables (SOPC)



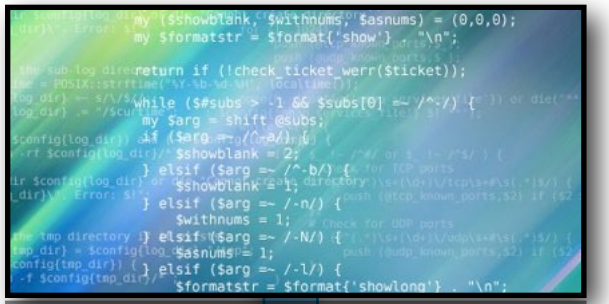
Les cibles technologiques d'intégration numérique

Evolution de la complexité d'intégration

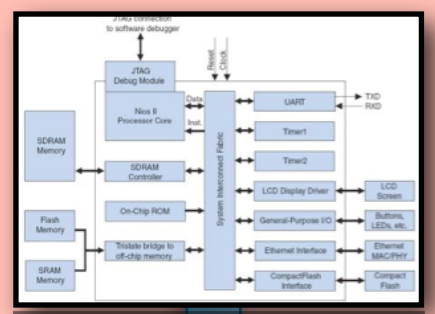


Les principales familles de cibles technologiques

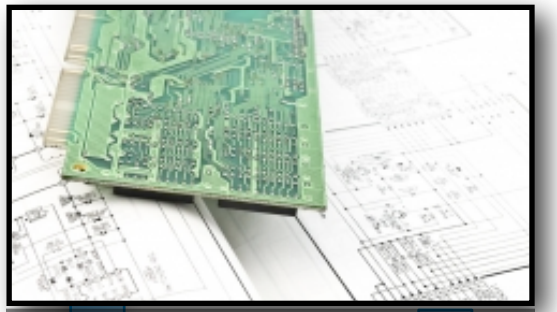
Implantation logicielle

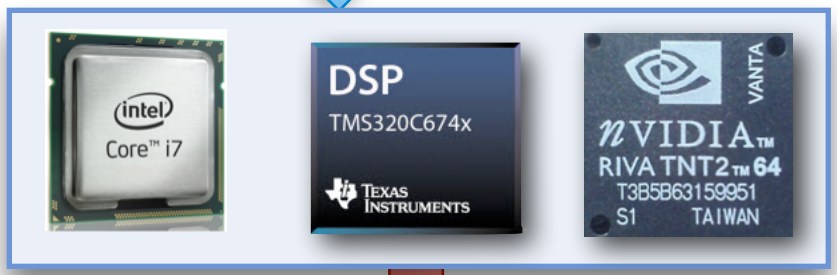


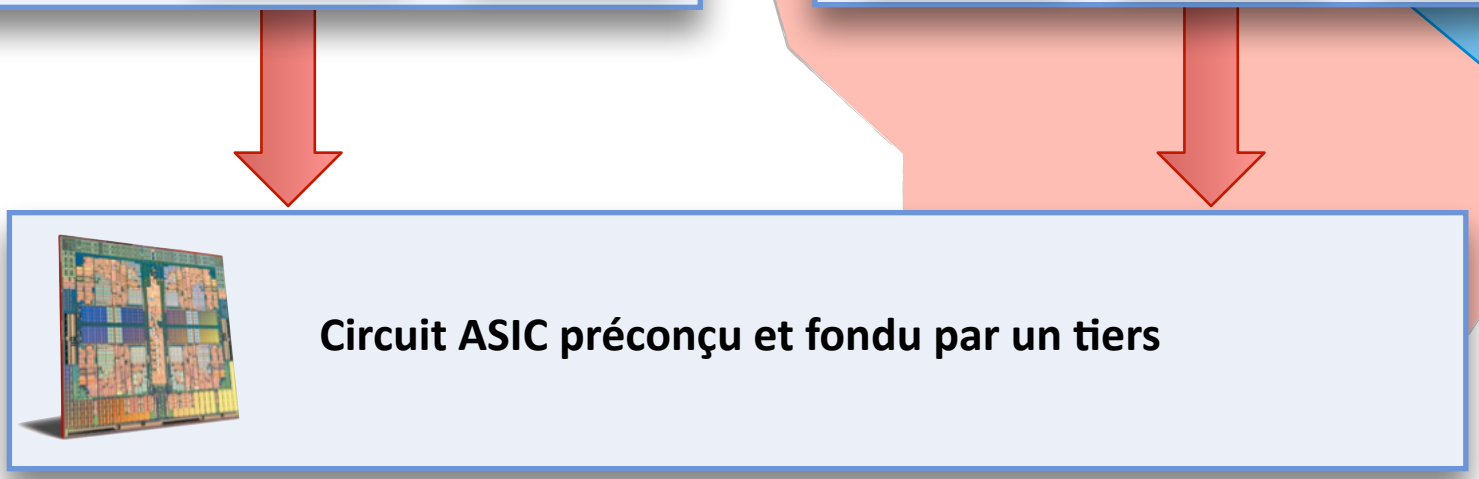
Codesign & softcores

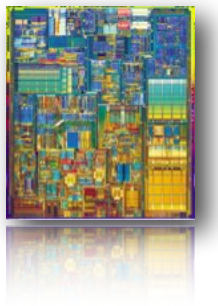


Architecture dédiée





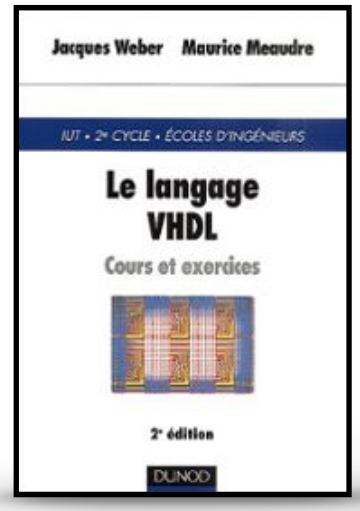




Introduction aux FPGA

Définition: Application Specific Integrated Circuit (ASIC)

Langage de description

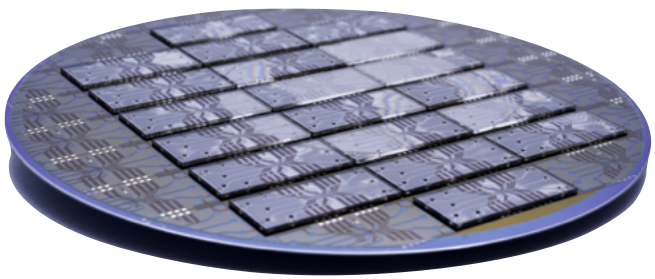


Design du circuit

Fonderie (Intel, TSMC, etc.)

Matière première (sable)

Circuit dédié (ASIC)

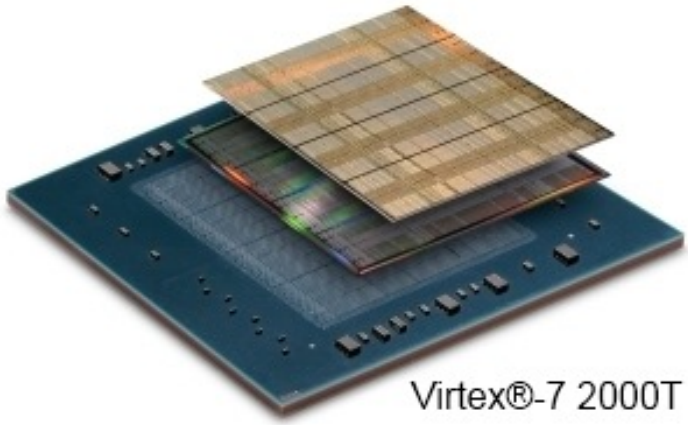


Classification des circuits reconfigurables

- Les circuits programmables ne sont pas vraiment des ASICs :
  - Circuits directement disponibles chez le fournisseur,
  - Développement et programmation «rapide»,
  - Pas besoin de fonderie.
- Différentes familles de circuit (re)programmable :
  - PLD (*Programmable Logic Device*); circuits de 100-200 portes,
  - CPLD (*Complex Programmable Logic Device*); circuits de 5-10 K portes,
  - FPGA (*Field Programmable Gate Array*); circuits atteignant $+10^6$ portes.
- Différentes techniques de programmation :
  - Cellules à fusible ou à antifusible,
  - Mémoire effaçable électriquement (EEPROM & Flash EPROM),
  - Mémoire SRAM.

Classification des circuits reconfigurables

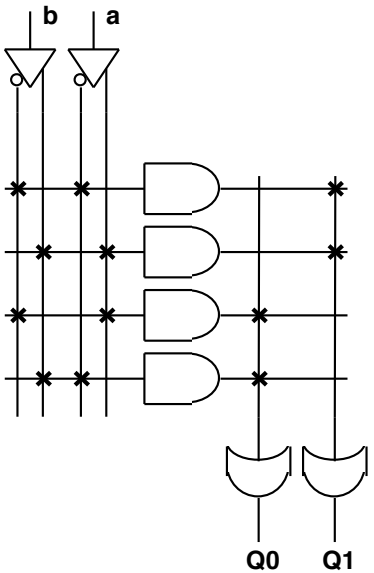
- Différentes techniques de programmation :
  - Cellules à fusible ou à antifusible,
  - Mémoire effaçable électriquement (EEPROM & Flash EPROM),
  - Mémoire SRAM.

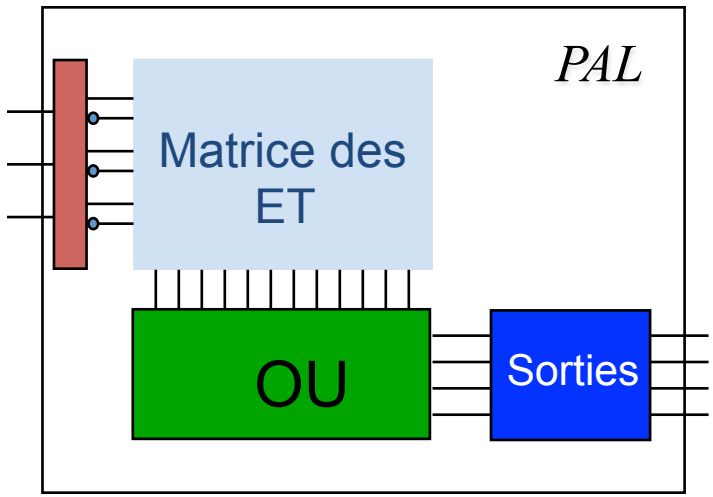

Virtex®-7 2000T

	Antifusible	**EPROM**	**SRAM**
Facilité de programmation	*faible*	*très importante*	*importante*
Densité	*faible*	*élevé*	*très élevé*
Famille de circuit	*FPGA*	*PLD CPLD*	*CPLD FPGA*
Reprogrammabilité	*non*	*oui*	*oui*

Les circuits configurables de type PAL

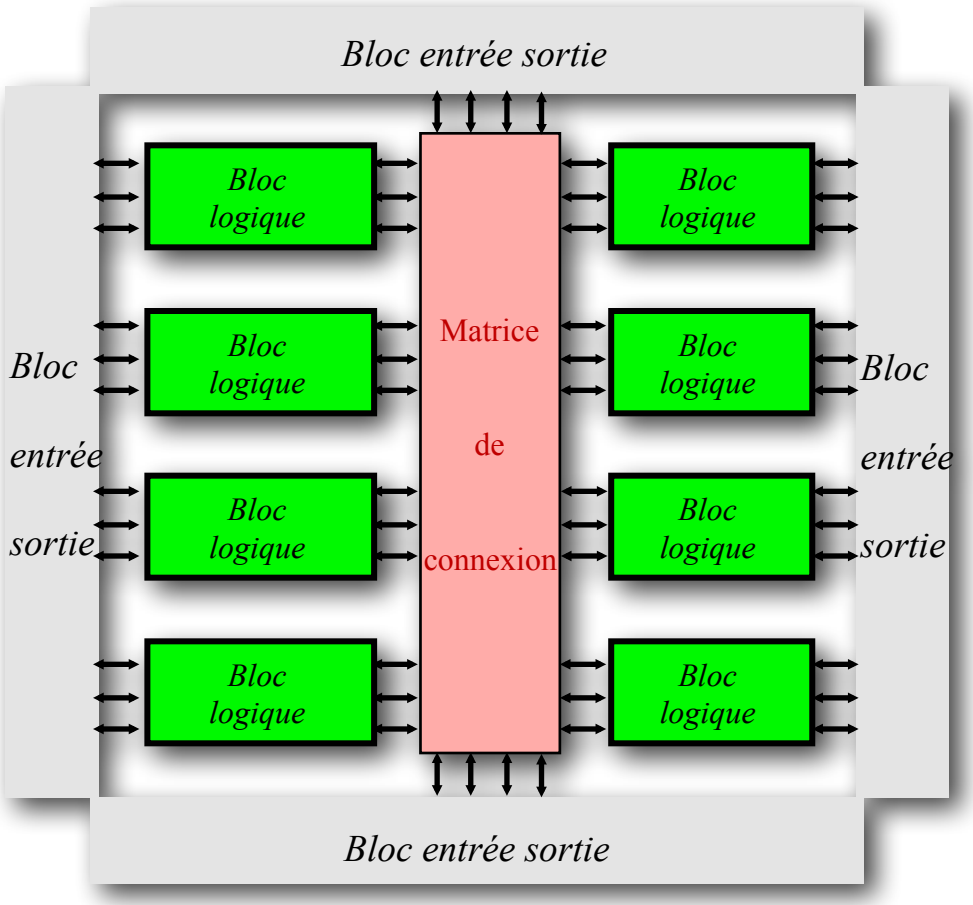
- La plupart des PALs ont la structure suivante :
  - Un ensemble d'opérateurs « ET » sur lesquels viennent se connecter les variables d'entrée et leurs compléments,
  - Un ensemble d'opérateurs « OU » sur lesquels les sorties des opérateurs « ET » sont connectées,
  - Une éventuelle structure de sortie (Portes inverseurs, logique 3-états, registres, etc.).
- Les interconnexions de ces matrices sont programmables.





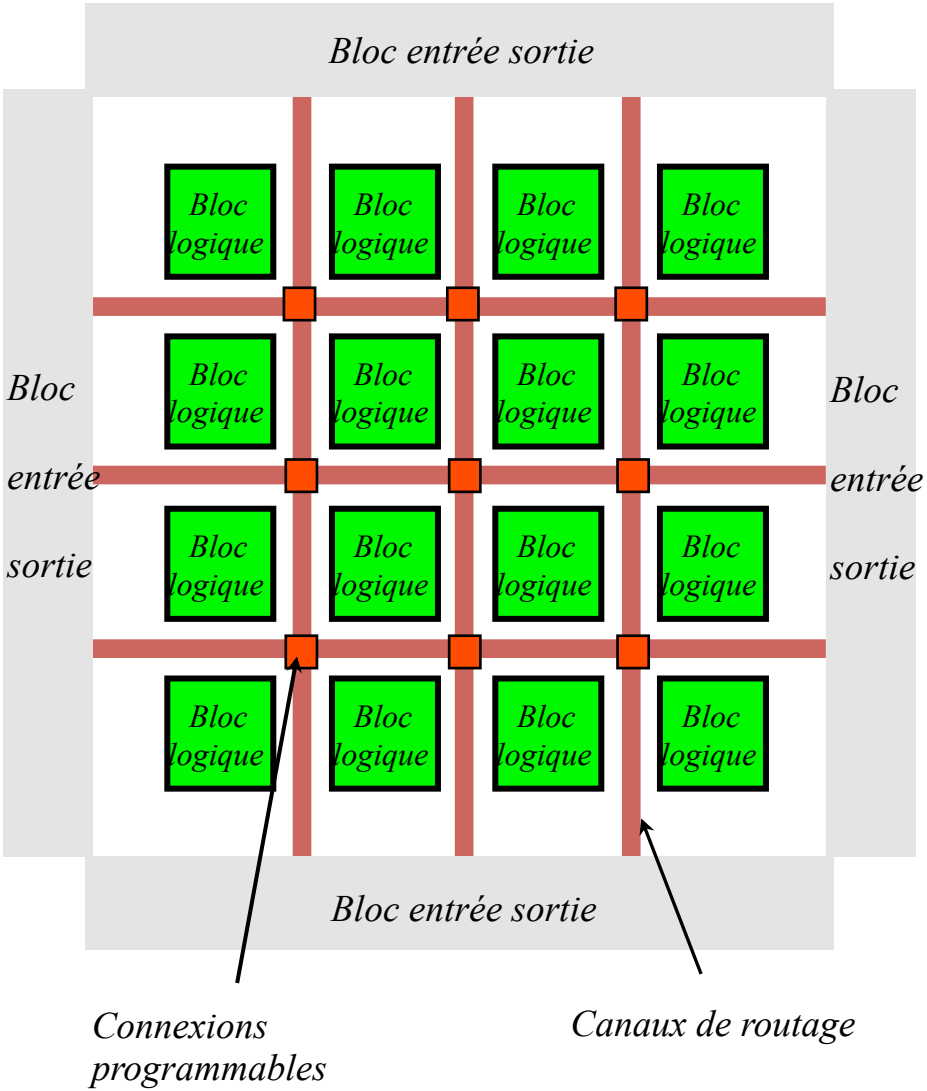
Les circuits configurables de type CPLD

- Un CPLD se présente comme un ensemble de fonctions de type PAL reliées à l'aide d'une matrice de connexion.
- Intérêt
  - ➡ Développement de circuits numériques de faible complexité (5000 à 100000 portes)
- Exemples
  - ➡ Altera MAXII: 240 à 2200 éléments logiques,
  - ➡ Xilinx CoolRunner2: 32 à 512 macrocells.

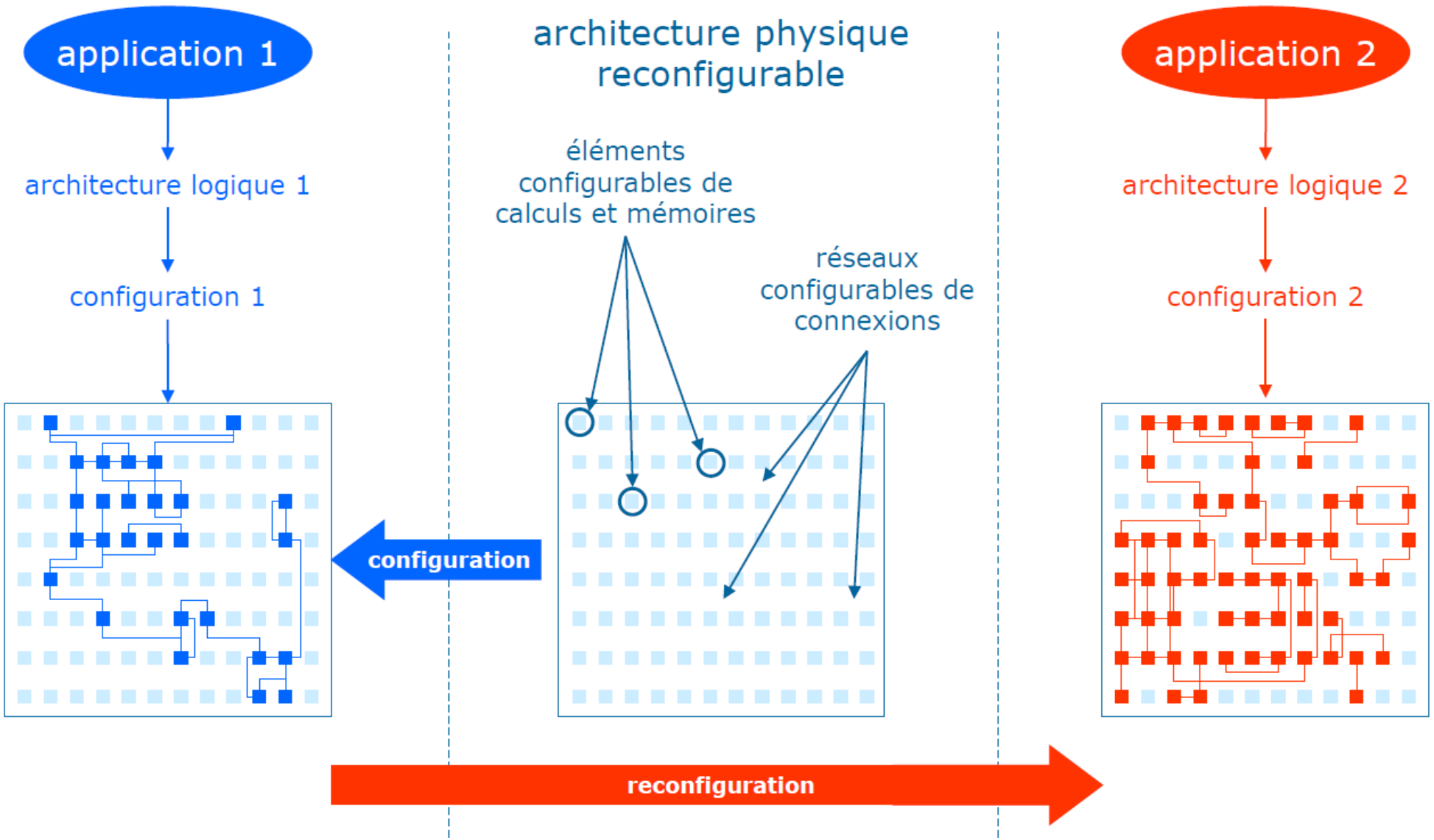


Les circuits configurables de type FPGA

- Les FPGAs comprennent:
  - De nombreux modules logiques,
  - Un réseau d'interconnexions entre les modules logiques,
    - Réseau non centralisé.
- Deux familles de FPGA :
  - FPGA de type SRAM (reprogrammable),
  - FPGA de type antifusible (non reprogrammable).
- Les FPGAs ont des capacités d'intégration élevées.



Intérêt des circuits (re)configurables de type FPGA



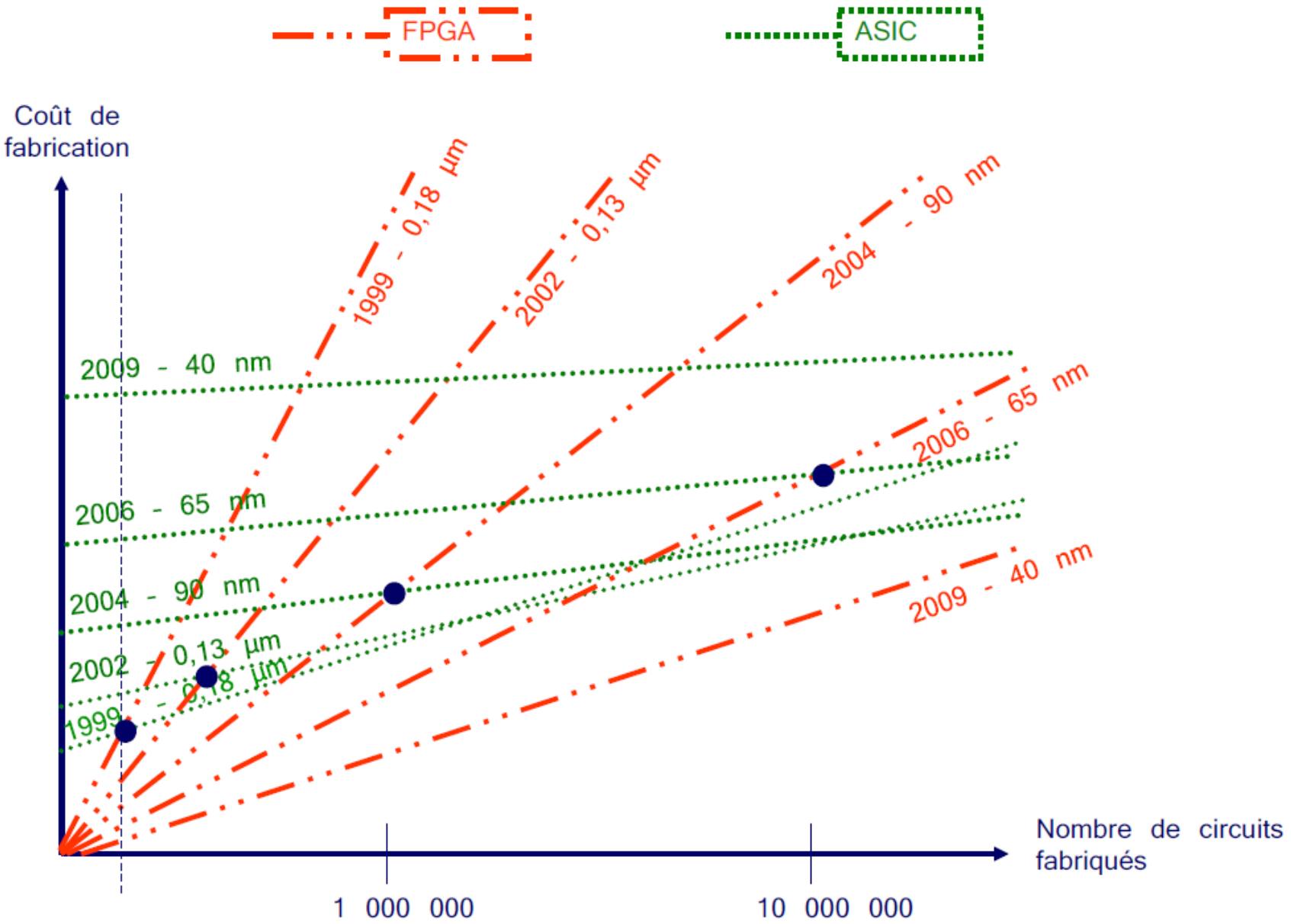
Comparaison des circuits (re)configurables versus les ASICs

Caractéristiques	Circuits Configurables		ASIC	
	FPGA SRAM	CPLD	Standard Cell	Full Custom
Densité (portes/m²)	1500 à 6500 0,13µm 90 nm	Faible o,18 µm	Grande 90 nm	Grande 90 nm
Performance (Hrz, Bits/s)	500 MHz 10 Gbits/s	200 MHz	qq Ghz 12 Gbits/s	qq Ghz 20 Gbits/s
Consommation (Watt)	Grande	Grande	Faible	Faible
Flexibilité	Grande	Grande	Aucune	Aucune
Temps de conception	Court	Court	Long	Très Long
Utilisation des outils	Facile	Facile	Complexe	Complexe
Cout de conception	Faible	Faible	Elevé	Très élevé
Volume de production	Faible et moyen	Faible et moyen	> 1 000 000 de pièces	> 1 000 000 de pièces

Technologie :	0.6 µ trois couches de métal	0.5 µ trois couches de métal	0.35 µ quatre couches de métal	0.25 µ cinq couches de métal	0.18 µ six couches de métal
Taille relative :	100%	78%	32%	19%	12%
Année :	1994	1995	1996	1998	1999

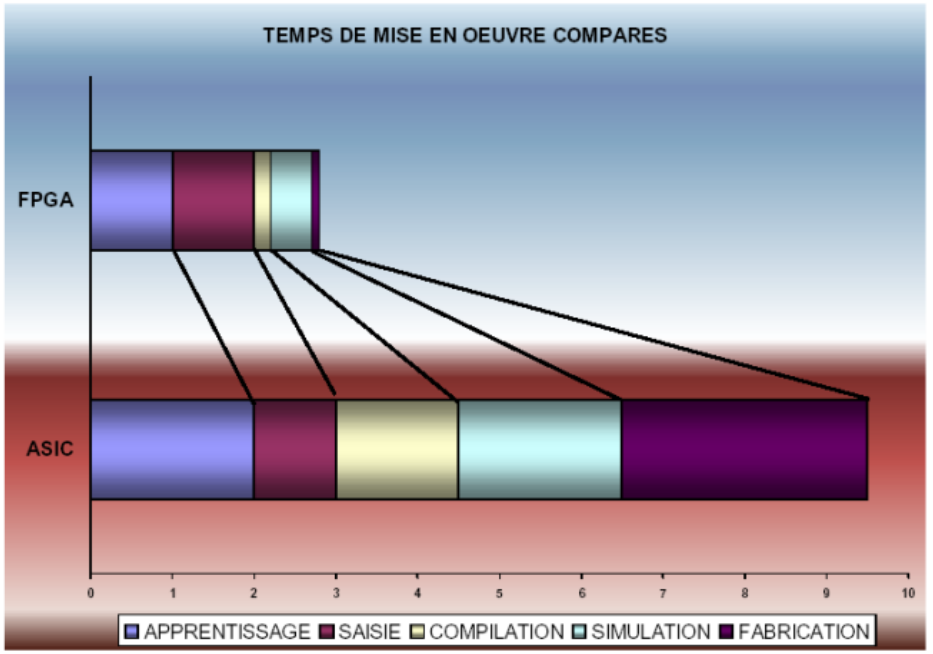
Actuellement (2012) : technologie 0.020 µm avec 11 couches de cuivre

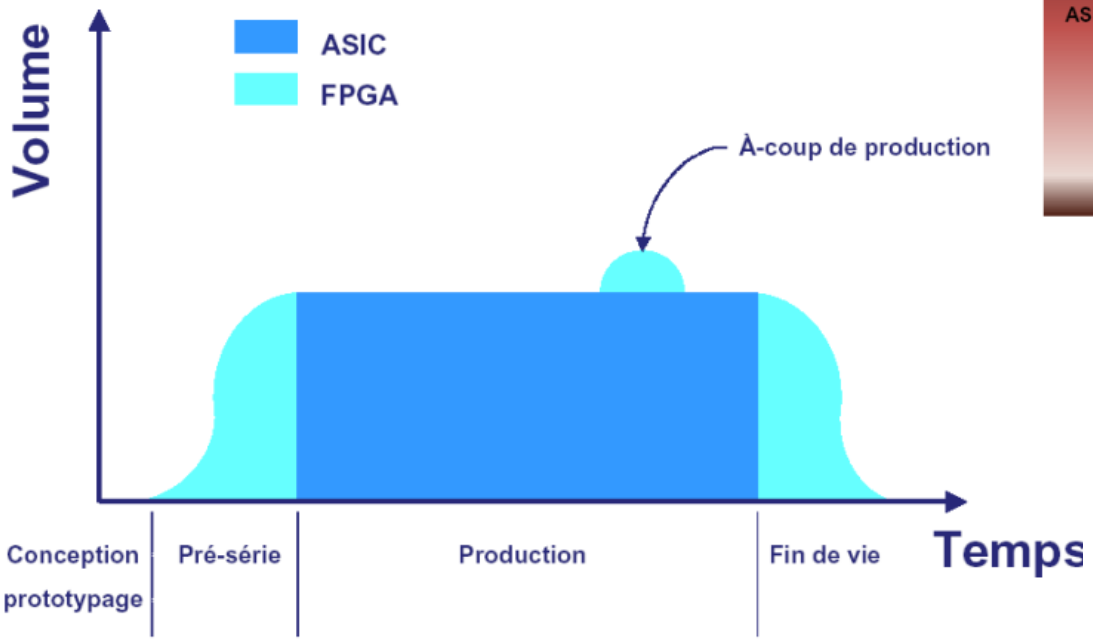
Evolution des couts de production des circuits



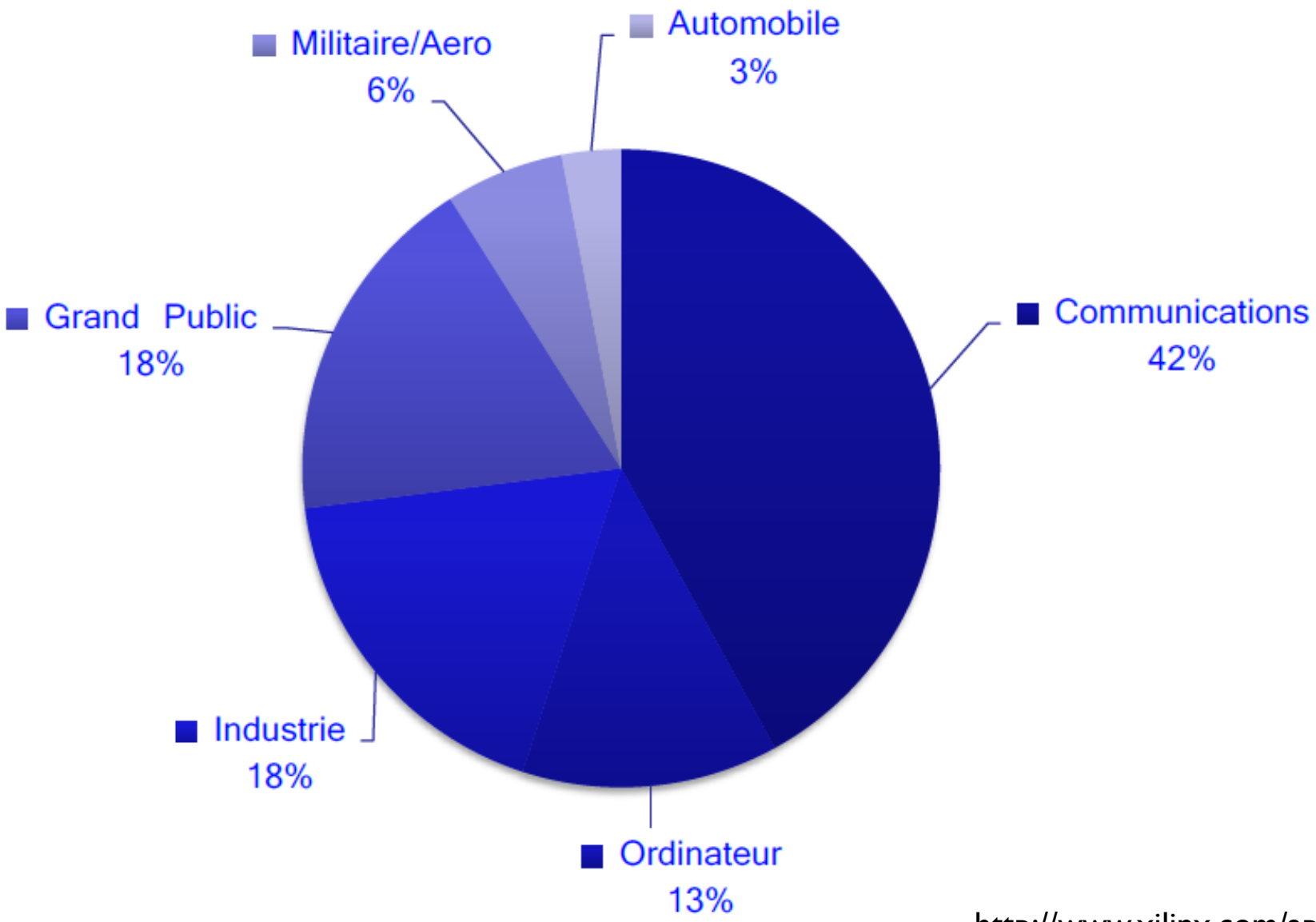
Contexte d'utilisation des circuits FPGA

Contexte d'utilisation des FPGA dans le cas d'une production de masse



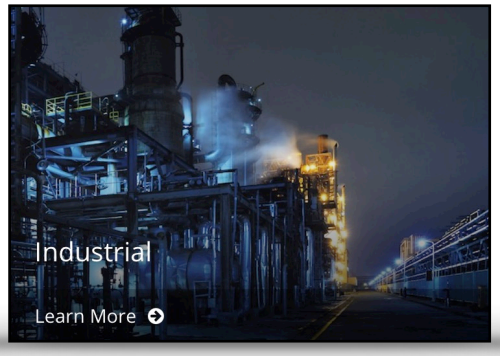


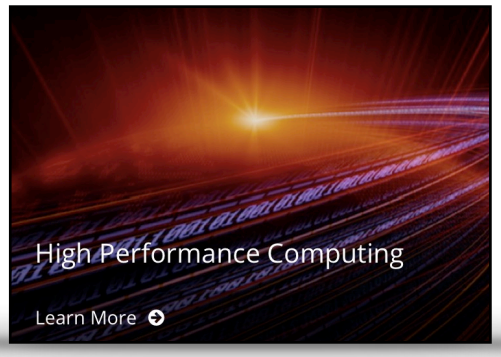
Les domaines d'utilisation des FPGA dans l'industrie (2008)

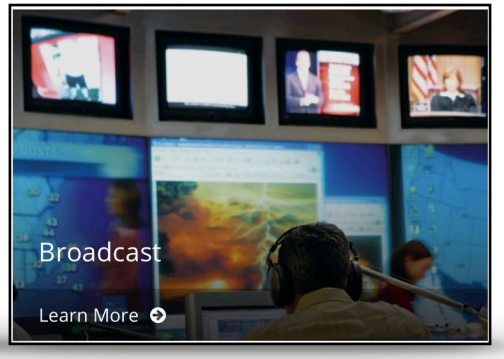


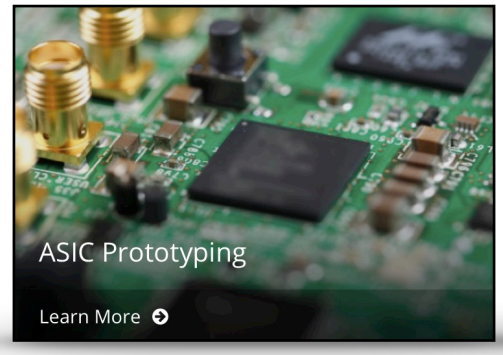
http://www.xilinx.com/applications/index.htm

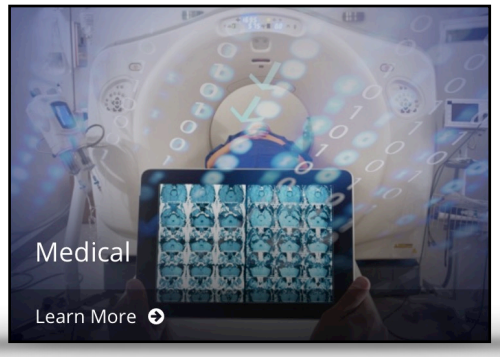
Les domaines d'utilisation des FPGA dans l'industrie (2018)

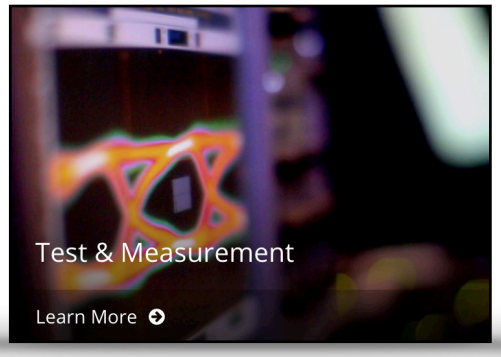


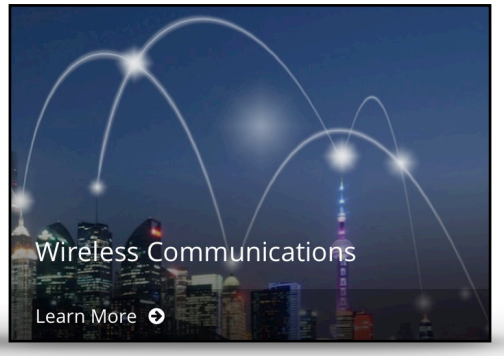


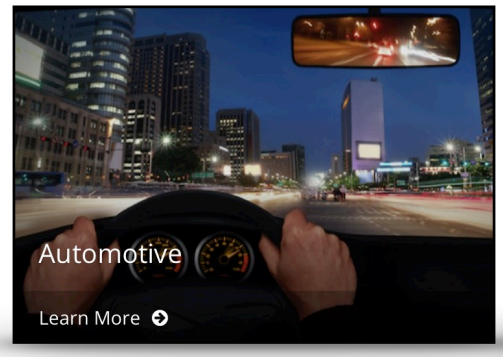


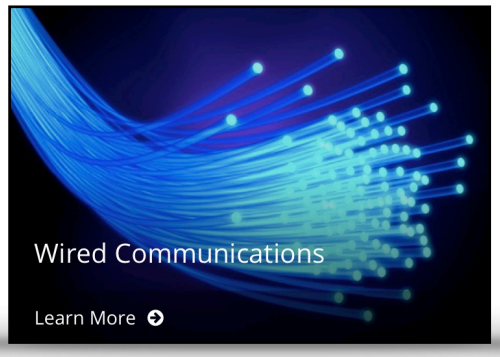


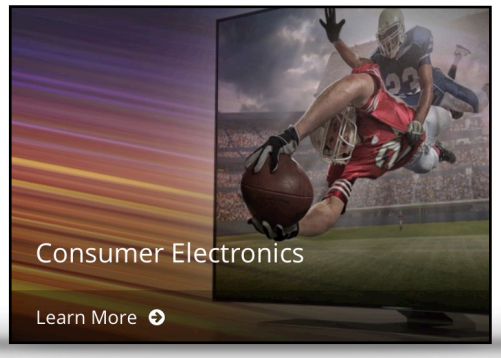


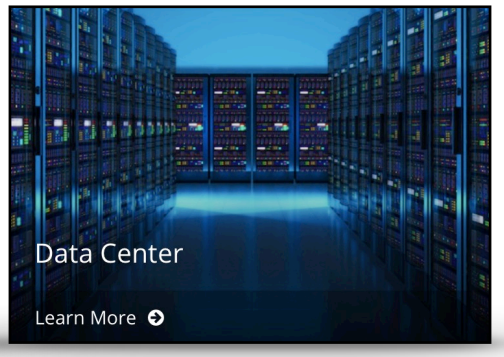












Classement des vendeurs de FPGA

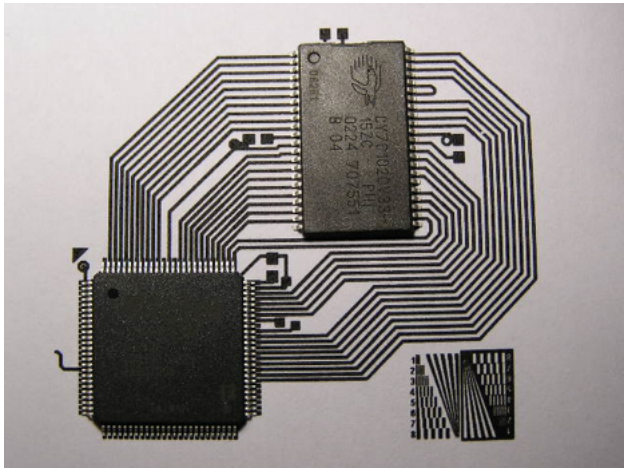
Worldwide FPGA/PLD vendor revenues and rankings, 2007-2008

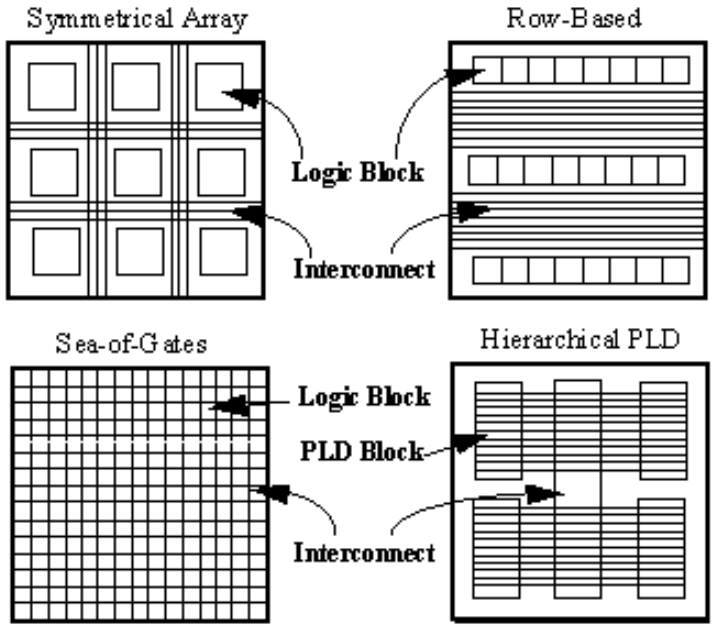
Rank 2007	Rank 2008	Company	Revenue ($M) 2007	Revenue ($M) 2008	Revenue Change 2007-2008	Market Share 2008
1	1	Xilinx	1,809	1.906	5.4%	51.2%
2	2	Altera	1,216	1,323	8.8%	35.5%
3	3	Lattice Semiconductor	229	222	–3.1%	6.0
4	4	Actel	196	218	11.2%	5.9%
6	5	QuickLogic	28	23	–17.9%	0.6%
5	6	Cypress Semiconductor	32	21	–34.4%	0.6%
7	7	Atmel	14	9	–35.7%	0.2%
8	8	Chengdu Sino Microelectronics System	4	3	–25.0%	0.1%
		Others	0	0	NM	0.0%
		Total Market	**3,528**	**3,725**	**5.6%**	**100.0%**

Source: Gartner

La classification des FPGA

- Classification suivant le mode de configuration,
  - ➡ FPGA à anti-fusibles (programmable),
    - le circuit est figé physiquement au niveau des connexions
  - ➡ FPGA à cellules SRAM (reprogrammable),
    - le circuit est reconfigurable à l'aide des cellules SRAM (modification des interconnexions).
- Classification en fonction de l'architecture interne,
  - ➡ Architecture en ilots de calcul,
  - ➡ Architecture hiérarchique,
  - ➡ Architecture logarithmique,



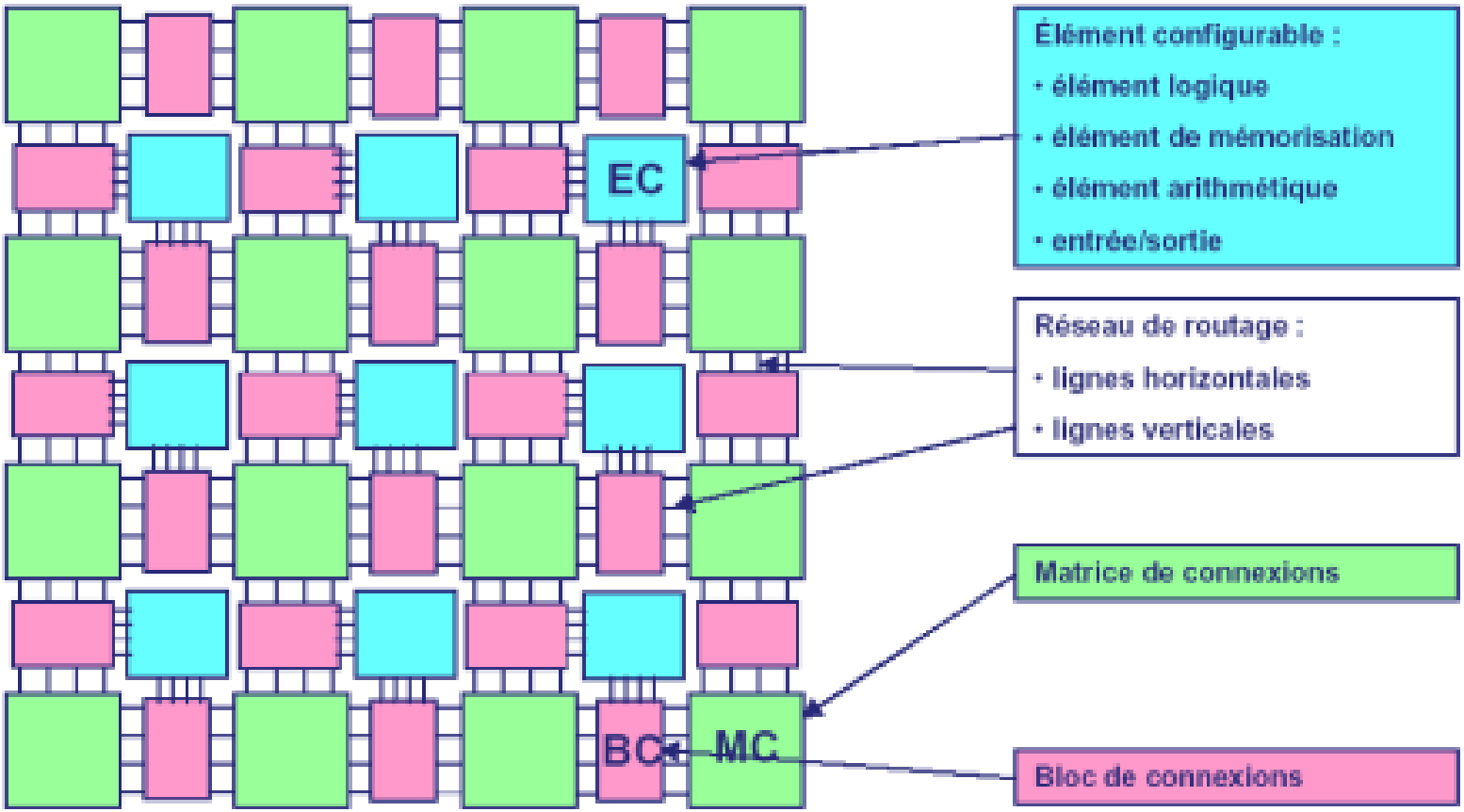


Architecture interne des circuits FPGA

Etude des FPGA de chez Xilinx

Les domaines d'utilisation des FPGA dans l'industrie (2008)

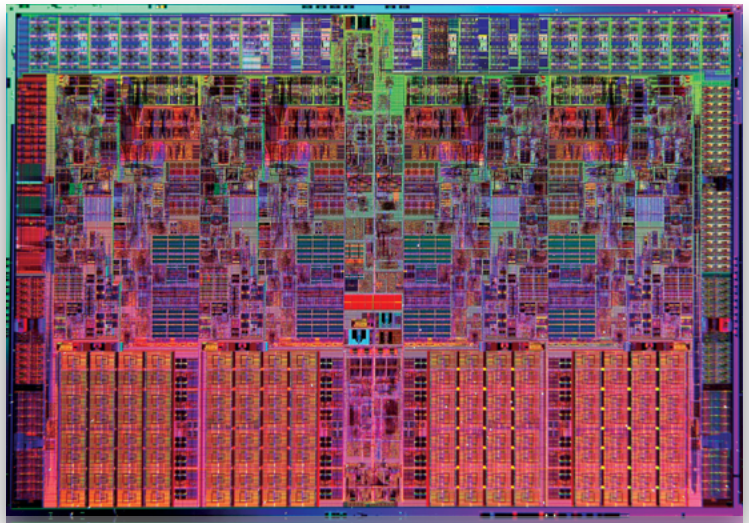
Les éléments fonctionnels (logique, mémoire, IO) sont regroupés sous forme de matrice.



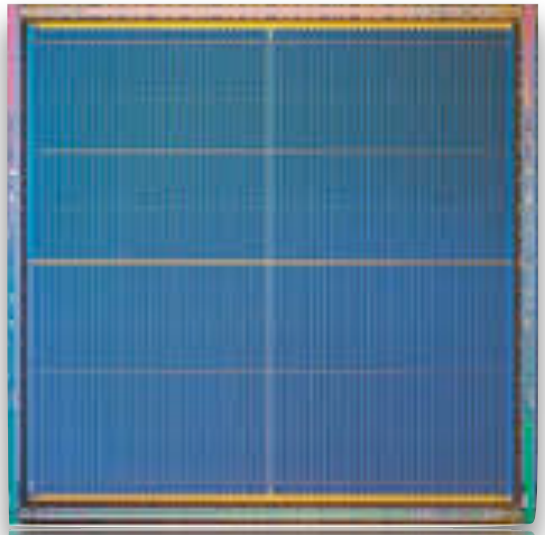
Type d'architecture employé chez Xilinx et Atmel

La régularité de l'architecture vue du silicium

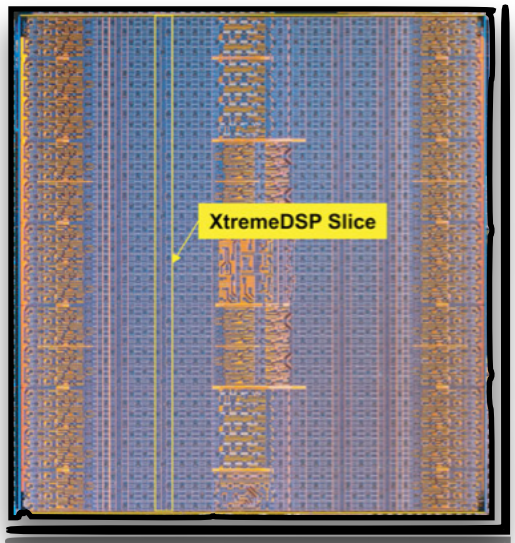
Intel Processor Core-i7



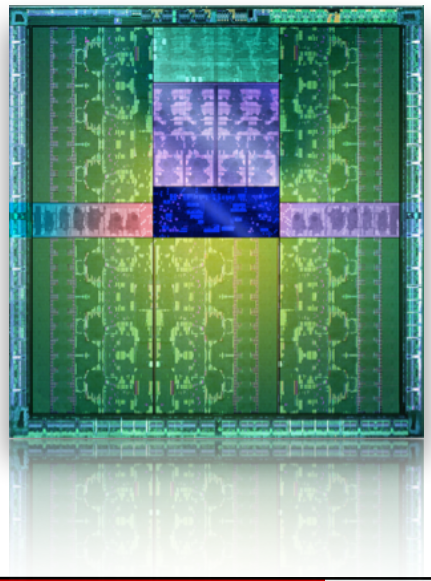
XILINX Virtex II



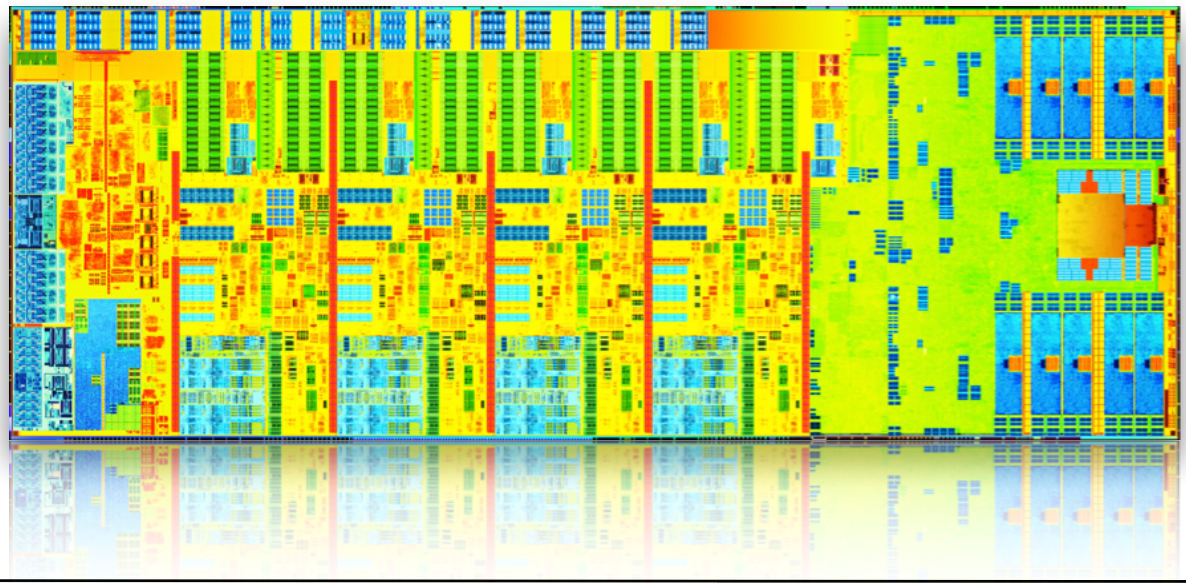
XILINX Virtex 4



NVIDIA K110 GPU

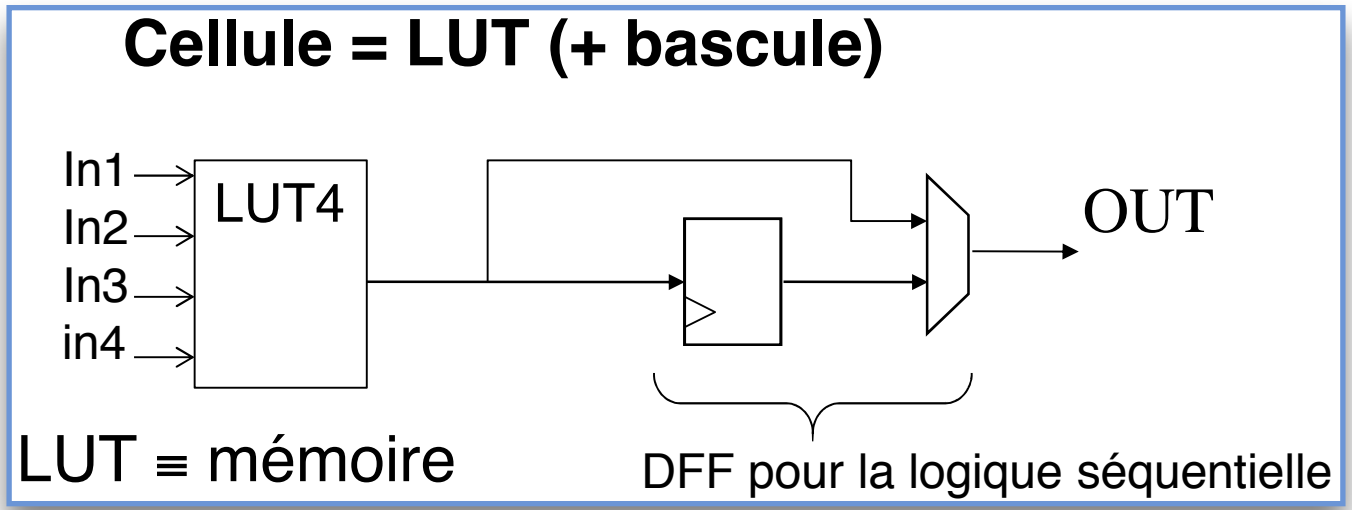


Intel Processor Quad-core Haswell



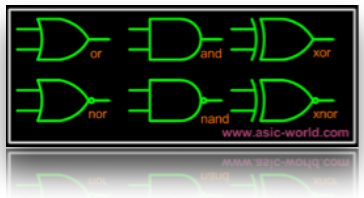
Elément logique de base (cellule élémentaire), qu'est ce donc ?

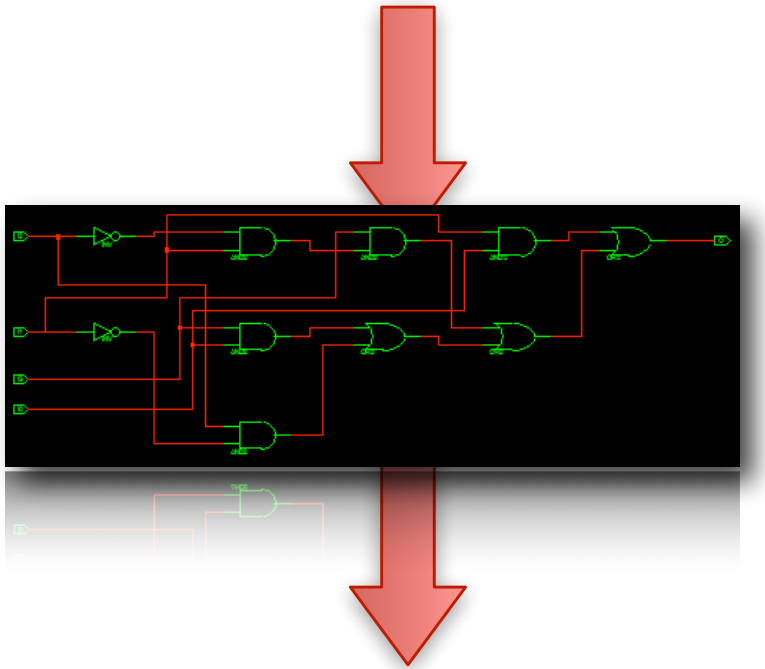
- Etudions le principe sur une cellule de base, composée d'une LUT4 entrées et d'une Flip-Flop.
  - ➡ Les cellules réelles sont un peu plus complexes...
- Une LUT4 à 4 entrées et correspond à une mémoire de 16 bits.
- Comment implanter une fonction logique quelconque sur cette architecture ?



Cas d'étude [O = E3.E1 + E2.E0 + E1.E0] - Solution ASIC

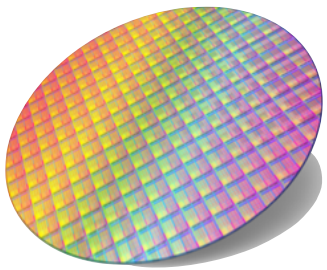
Equation logique +





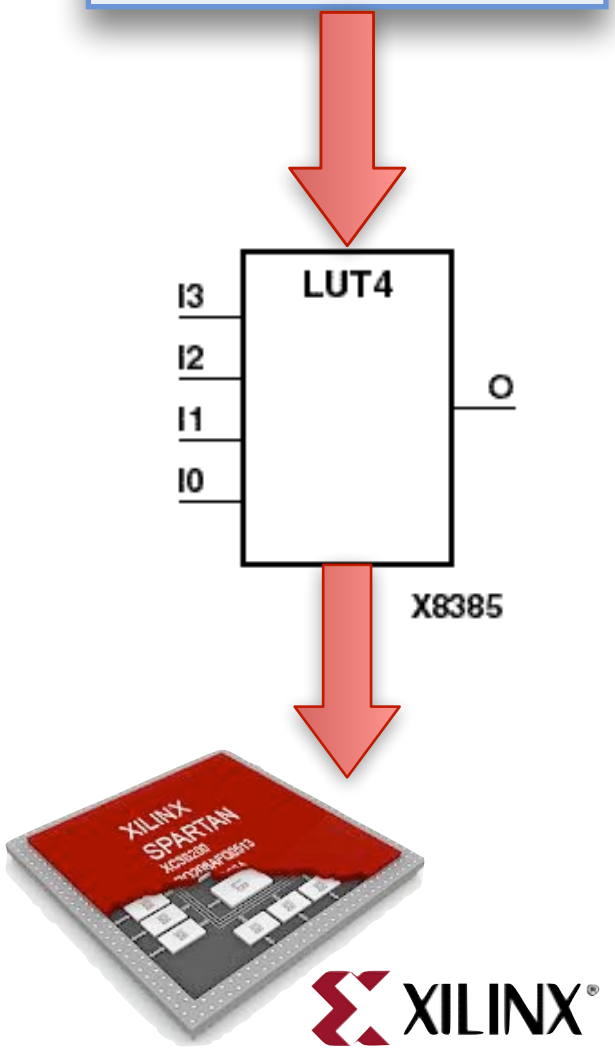
Le processus d'intégration est contraint par la bibliothèque de portes logiques fournie par le fondeur.

tsmc ®



Cas d'étude [O = E3.E1 + E2.E0 + E1.E0] - Solution FPGA

Equation logique



Le processus d'intégration est contraint par les ressources disponibles dans le FPGA.

LUT à 4 entrées = mémoire 16-bits.

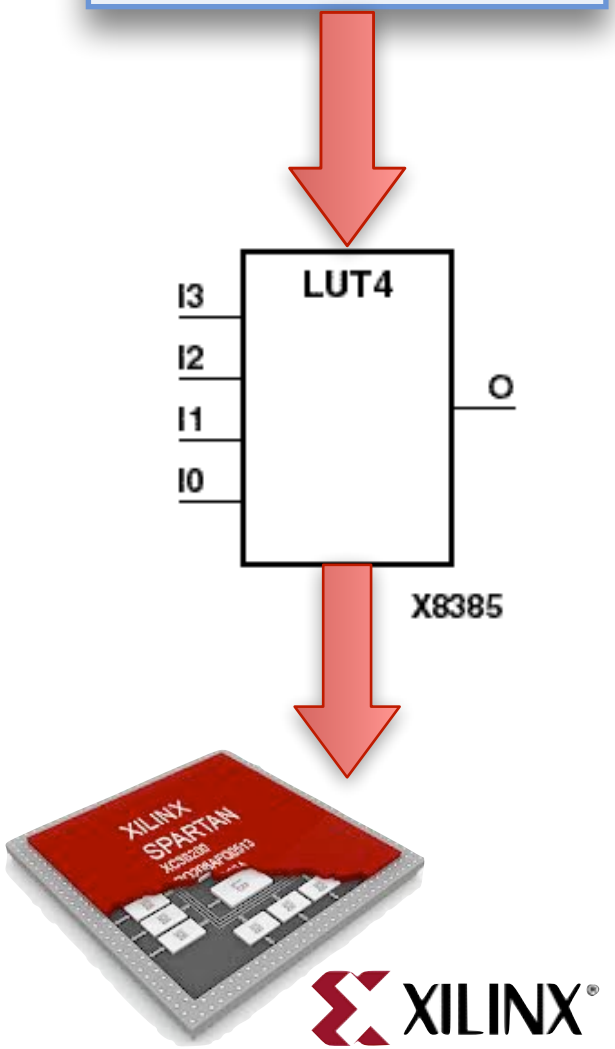
Il faut précalculer le contenu de la mémoire à partir de la fonction logique à intégrer.

Peu importe la complexité de la fonction logique (4 entrées max.) => 1 LUT4.

Toutefois... le pendant est une fonction logique à 1 entrée (i.e. NOT) => 1 LUT4.

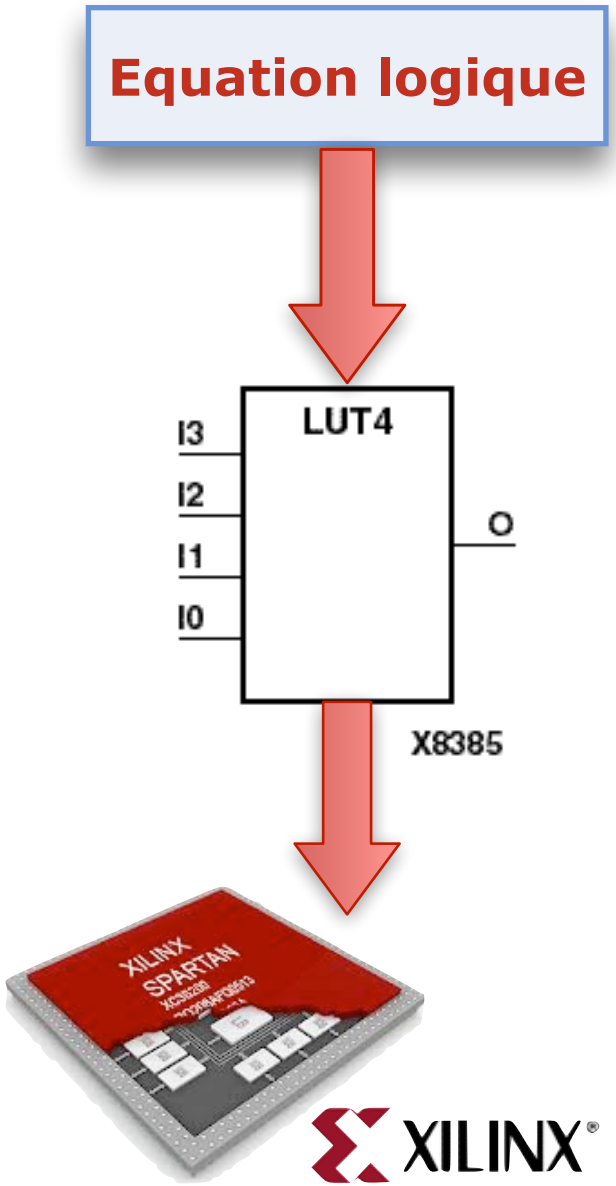
Cas d'étude [O = E3.E1 + E2.E0 + E1.E0] - Solution FPGA

Equation logique



I3	I2	I1	I0	O
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Cas d'étude [O = E3.E1 + E2.E0 + E1.E0] - Solution FPGA



I3	I2	I1	I0	O	INIT
0	0	0	0	0	8
0	0	0	1	0	
0	0	1	0	0	
0	0	1	1	1	B
0	1	0	0	1	
0	1	0	1	1	
0	1	1	0	0	
0	1	1	1	1	E
1	0	0	0	0	
1	0	0	1	1	
1	0	1	0	1	
1	0	1	1	1	B
1	1	0	0	1	
1	1	0	1	1	
1	1	1	0	0	
1	1	1	1	1	

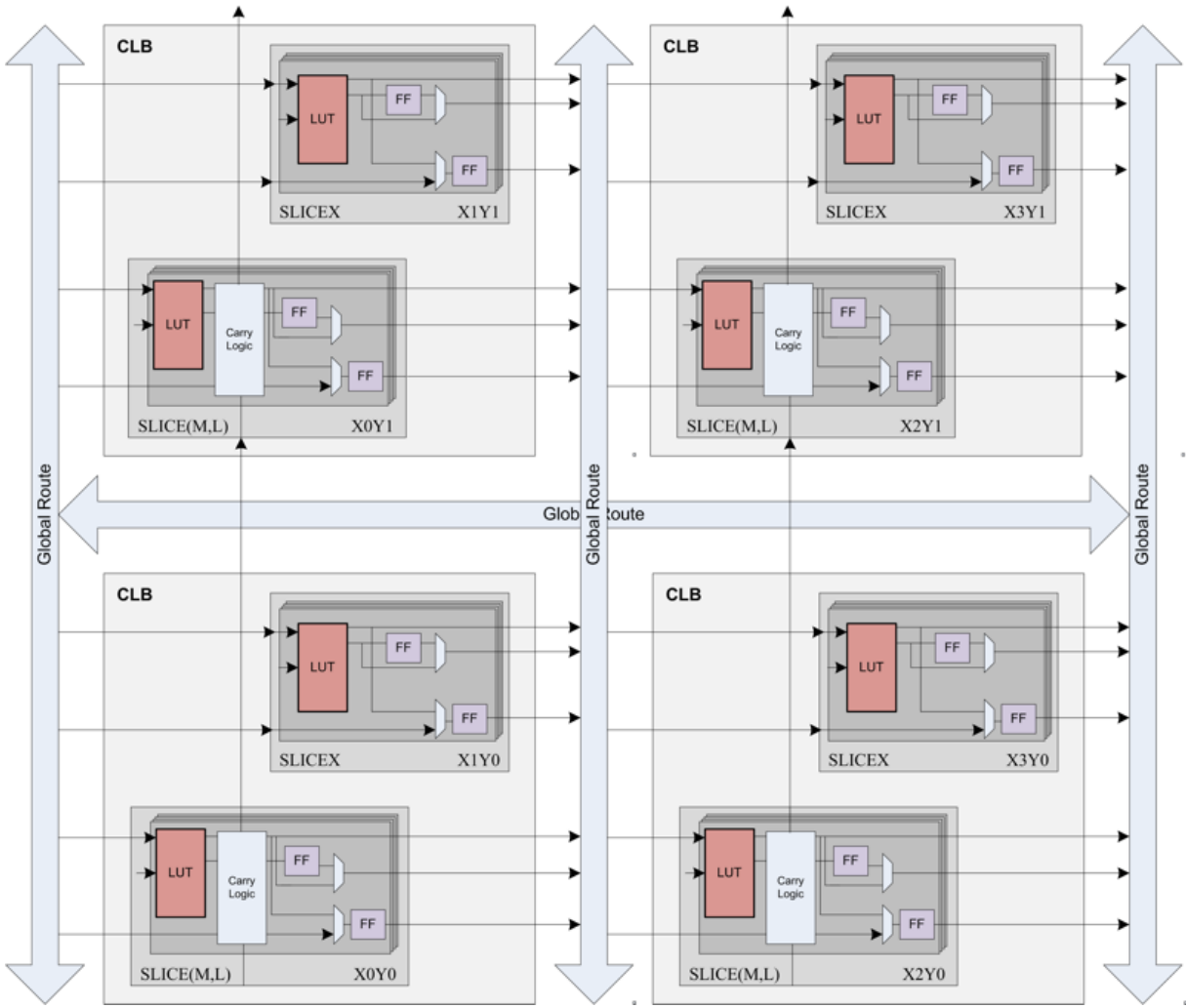
Cas d'étude [O = E3.E1 + E2.E0 + E1.E0] - Solution FPGA

I3	I2	I1	I0	O	INIT
0	0	0	0	0	
0	0	0	1	0	8
0	0	1	0	0	
0	0	1	1	1	
0	1	0	0	1	
0	1	0	1	1	B
0	1	1	0	0	
0	1	1	1	1	
1	0	0	0	0	
1	0	0	1	1	E
1	0	1	0	1	
1	0	1	1	1	
1	1	0	0	1	
1	1	0	1	1	B
1	1	1	0	0	
1	1	1	1	1	

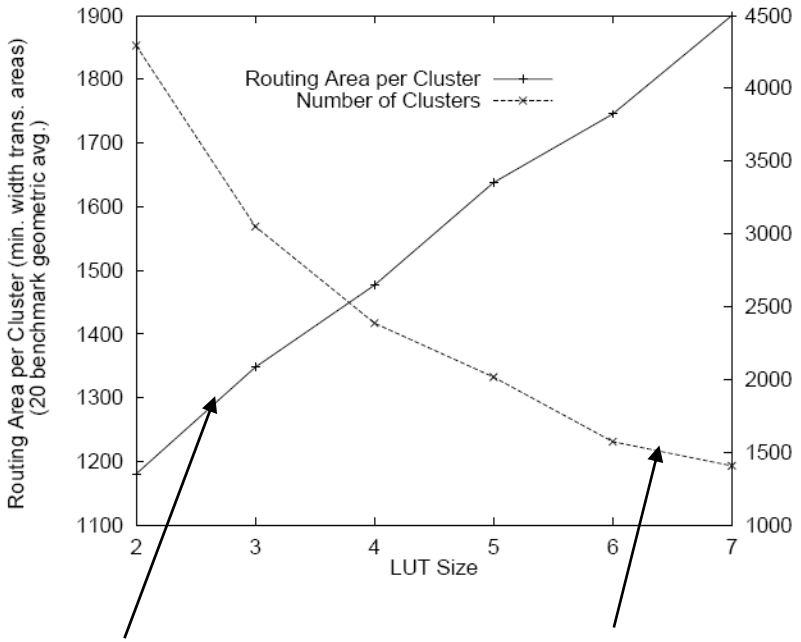
```
LIBRARY UNISIM;
USE UNISIM.Vcomponents.ALL;
.
.
ARCHITECTURE ArcTop OF top IS

BEGIN

  i_LUT : LUT4
    GENERIC MAP(
      INIT => x"BEB8"
      )
    PORT MAP(
      I0 => I0,
      I1 => I1,
      I2 => I2,
      I3 => I3,
      O  => O
      );
```

Utilité de la cellule de base étudiée

- Que peut on faire avec une LUT à 4 entrées ?
  - Un additionneur 1-bit,
  - Une mémoire 16-bits,
  - Un registre à décalage,
  - Toutes les équations logiques à 4 entrées,
  - Etc…
- Limitations ?
  - Les équations logiques à **n** entrées (**n** > 4).
    - Nécessaire d'utiliser plusieurs LUT4 interconnectée les unes avec les autres...

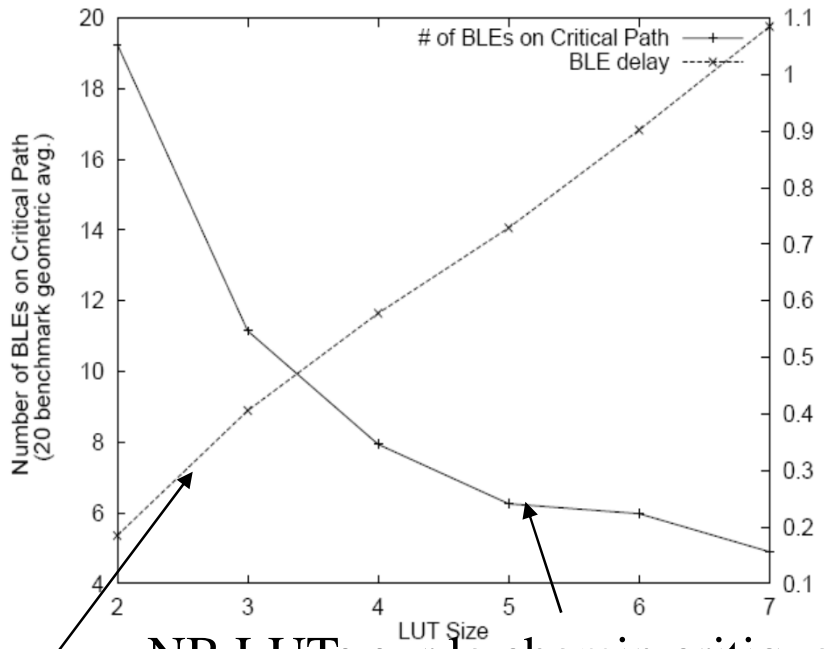


Pourquoi utiliser des LUTs à 4 entrées ?



interconnexion

Nb LUTs



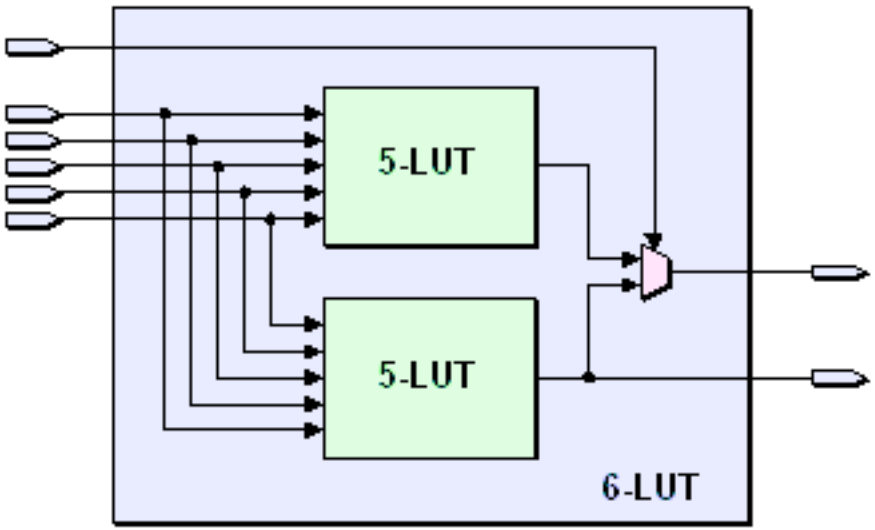
Temps critique

NB LUTs sur le chemin critique

Elias Ahmed, Jonathan Rose: The effect of LUT and cluster size on deep-submicron FPGA performance and density. IEEE Trans. VLSI Syst. 12(3): 288-298 (2004)

Toutefois, aujourd'hui les LUTs ont [6, 8] entrées et [1, 2] sorties

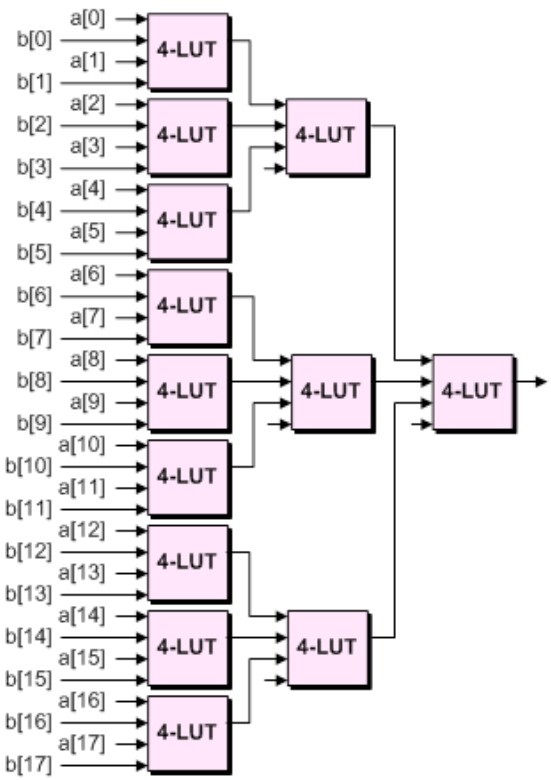
Evolution de la taille des LUTs chez Xilinx



Le passage des LUT4 (Virtex-4) aux LUT6 (Virtex-5,6,7) a été réalisée an factorisant 2 LUTs de taille inférieure.

Réduction du nombre d'étages de logique (LUT) a traverser pour implanter des fonctions complexes (chemin critique).

Risque d'inefficacité silicium : une LUT6 pour implanter une fonction B = **non** A.
=> Fonctionnement en mode 2 sorties.

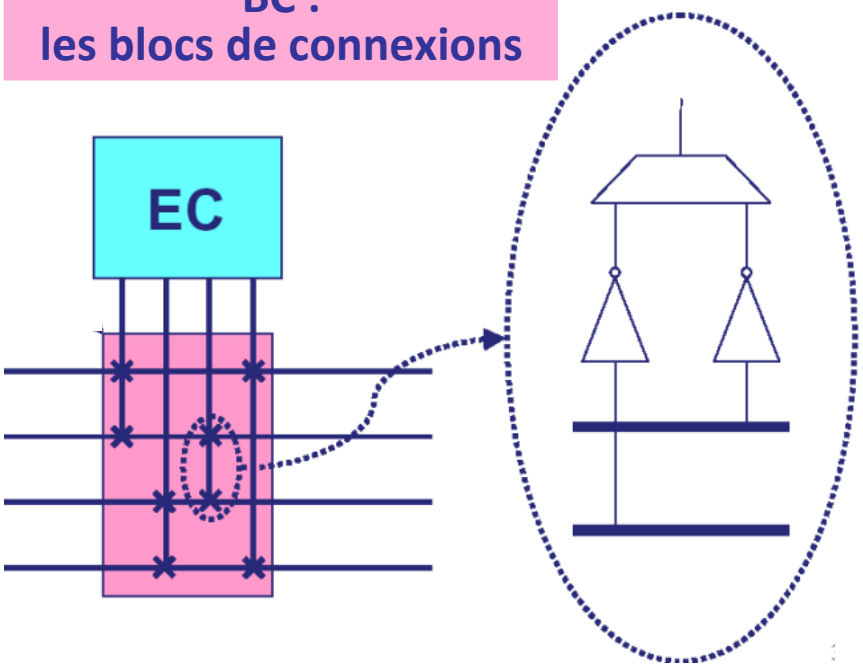


(a) Virtex-4: 13 LUTs, 3 levels

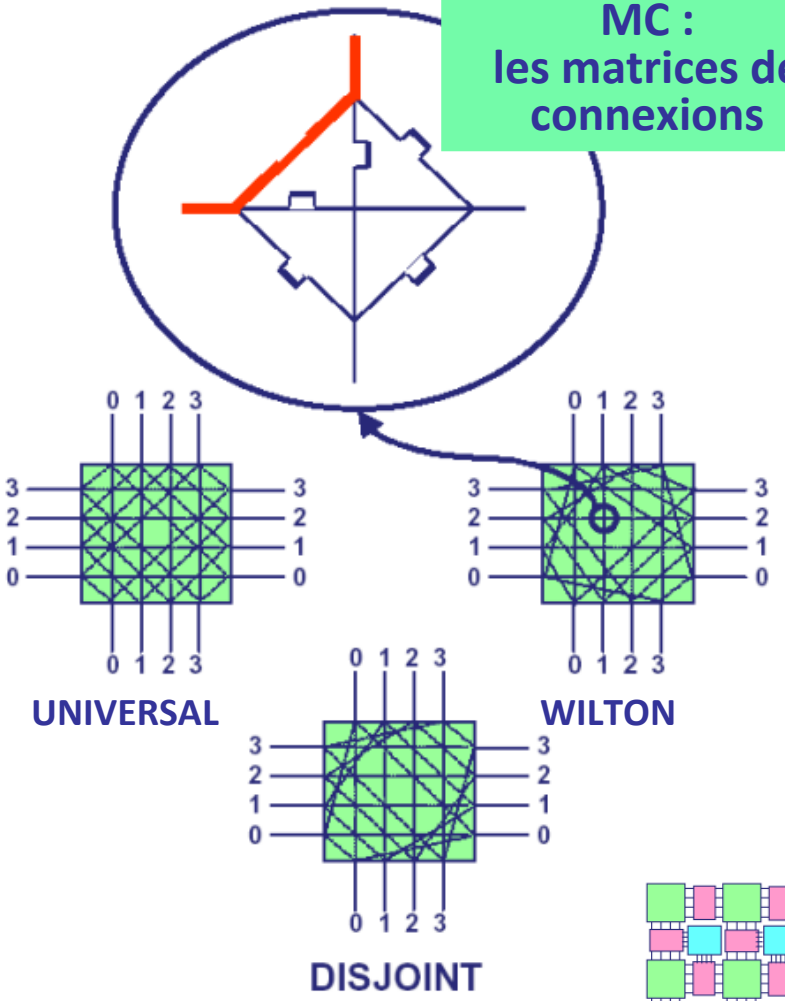
(b) Virtex-5: 7 LUTs, 2 levels

Interconnexion des éléments configurables

BC :
les blocs de connexions

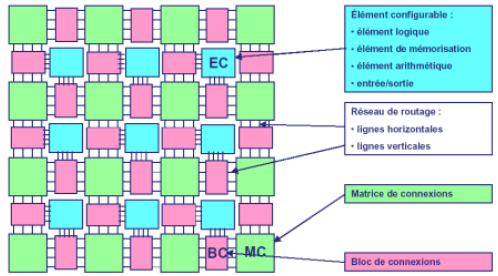


MC :
les matrices de connexions



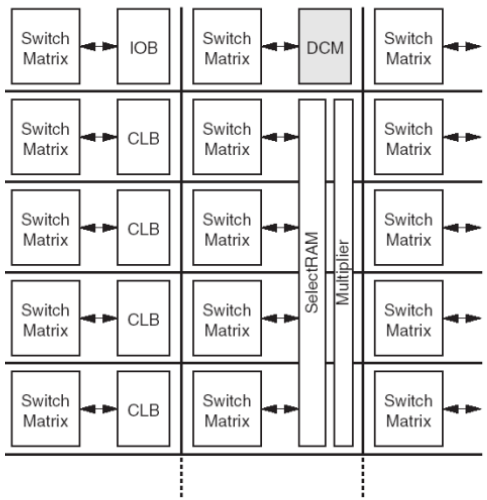
Les **blocs de connexions** assurent la connexion des éléments configurables

Les **matrices de connexions** assurent la connexion des blocs de connexions



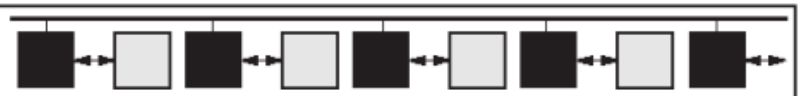
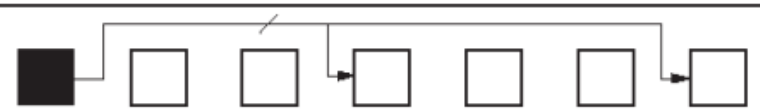
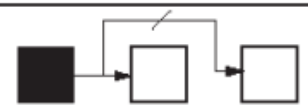
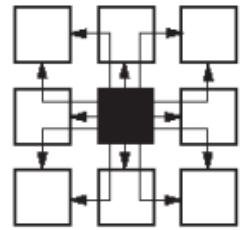
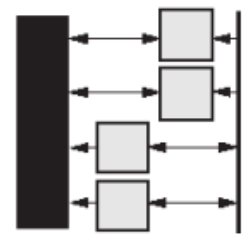
Interconnexion des éléments configurables

Le réseau de routage



L'architecture peut être vue comme un tableau de matrice de connexion

Ressources de routage hiérarchiques

24 Horizontal Long Lines 24 Vertical Long Lines	
120 Horizontal Hex Lines 120 Vertical Hex Lines	
40 Horizontal Double Lines 40 Vertical Double Lines	
16 Direct Connections (total in all four directions)	
8 Fast Connects	

Il existe également des ressources de routage dédiées aux horloges et aux retenues

La structure interne des FPGA est plus complexe

- En théorie, toutes les fonctions numériques peuvent être intégrées à l'aide de LUTx,
  - ➡ Certaines implantations s'avèrent toutefois inefficaces !
- Ajout de ressources spécialisées dans le FPGA,
  - ➡ Bloc mémoire (RAM 18/36kbits)
  - ➡ Blocs DSP (MAC)
- D'autres blocs dédiés peuvent être disponibles,
  - ➡ PLL, PCIe, Ethernet, etc.

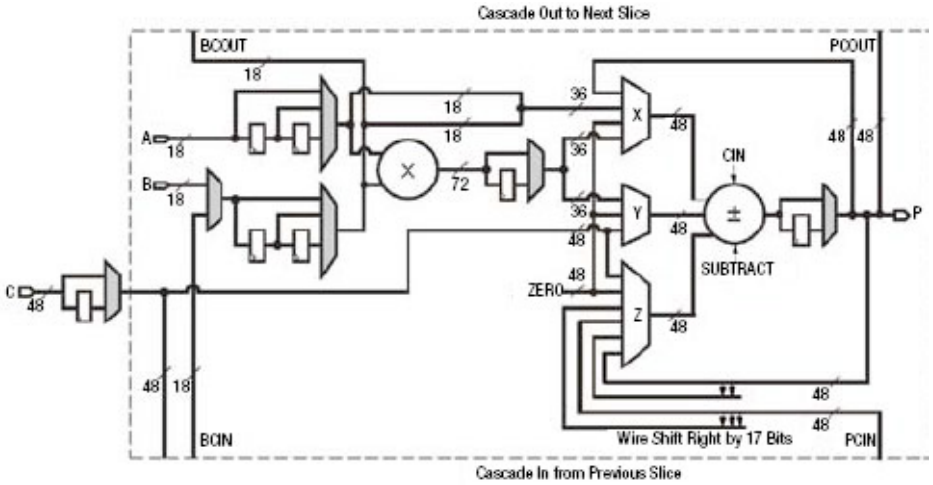
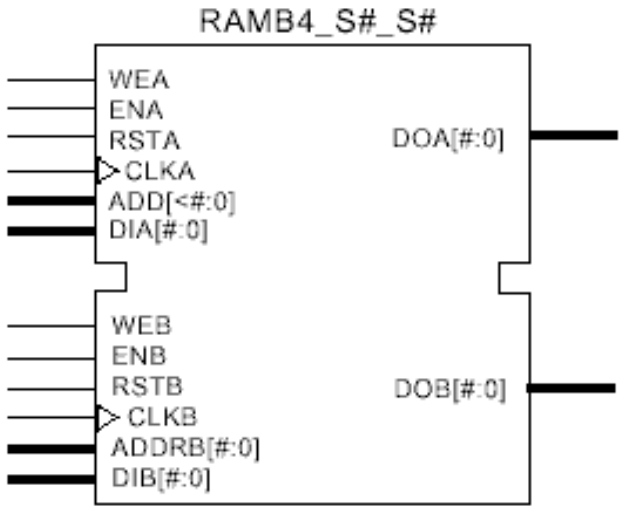


Figure 1 A single Xilinx V-4 Xtreme DSP48 Slice. Two DSP slices are combined with logic to form a DSP tile.

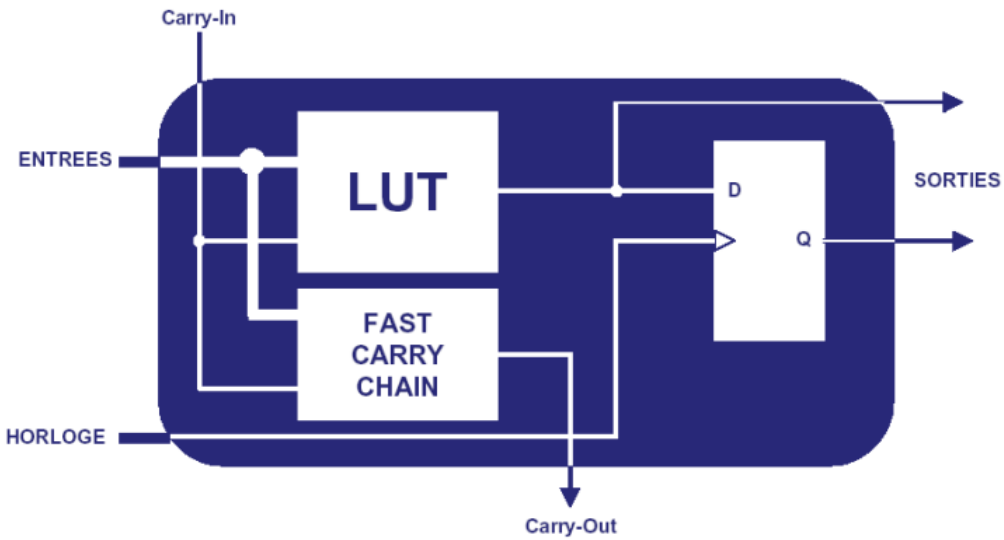


Décomposition de l'élément logique de base (CLB)

- Le bloc de logique CLB (Configurable Logic Block) correspond à l'élément logique (configurable) de base chez Xilinx.
- Un CLB est construit à partir de cellules élémentaires (de 2 à 4 cellules en règle générale).

Cellule élémentaire

- Un générateur de fonction LUT (Look Up Table),
- Une chaîne de propagation rapide de la retenue,
- Une bascule D.



Terminologie dans la structure des FPGA de chez Xilinx

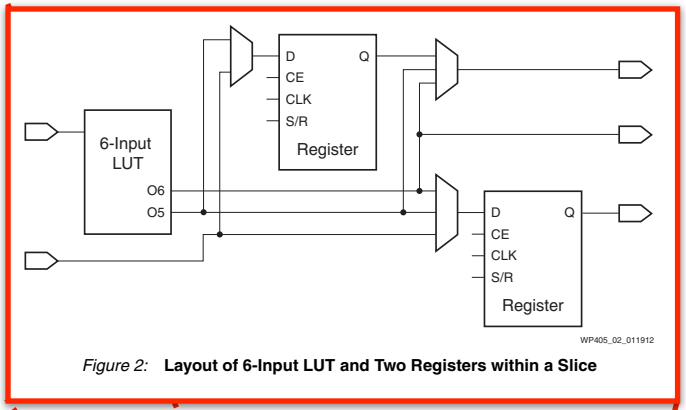
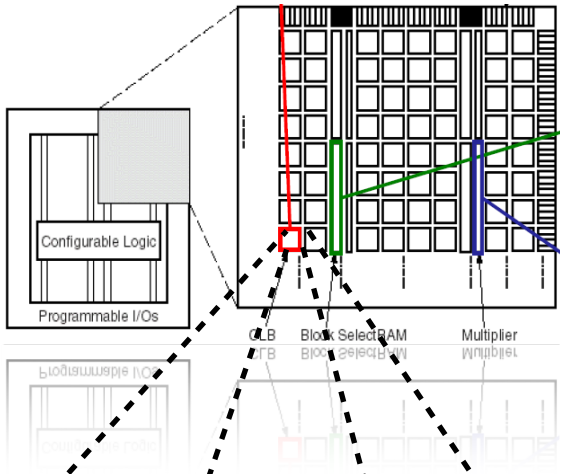


Figure 2: **Layout of 6-Input LUT and Two Registers within a Slice**

Element de base



Xilinx FPGA structure

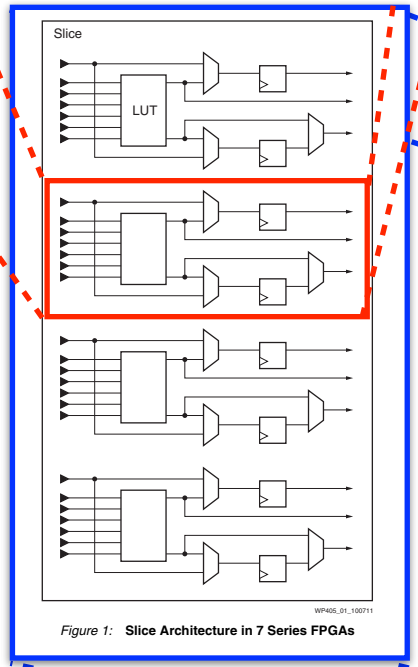
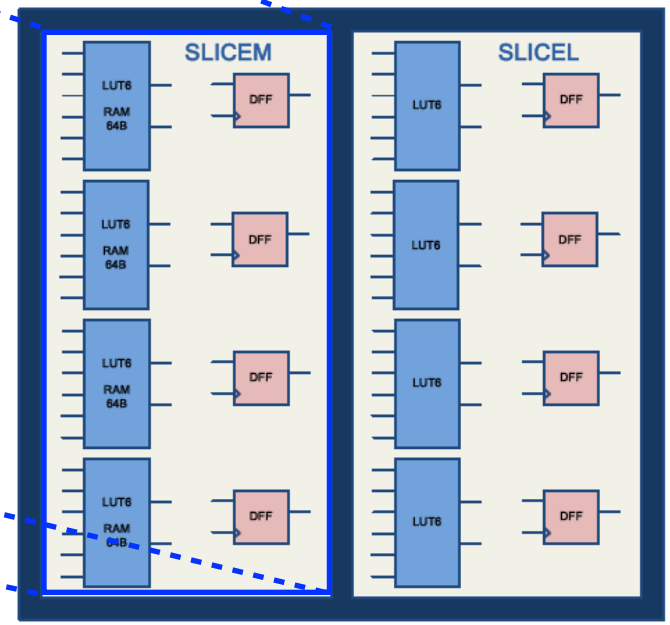


Figure 1: **Slice Architecture in 7 Series FPGAs**

Slice (M ou L)

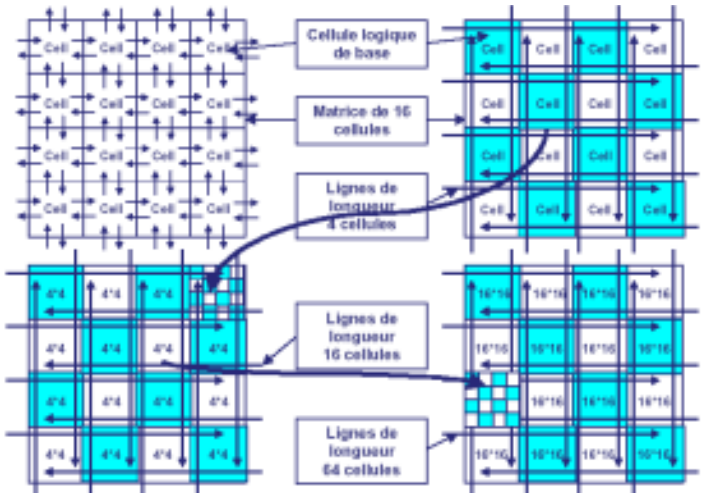


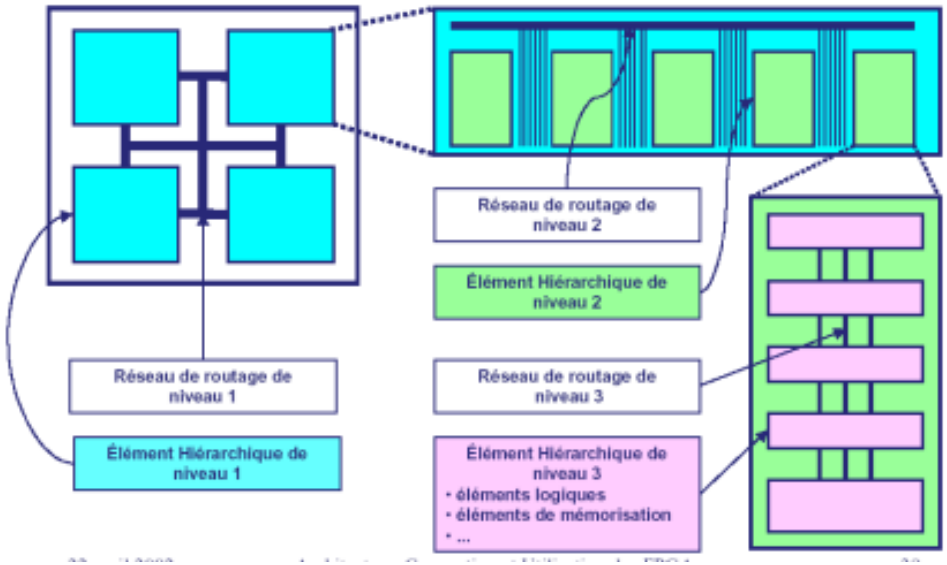
Configurable Logic Block (CLB)

Les différents types d'architectures de FPGA

Architecture hiérarchique où chaque niveau i correspond à une matrice de 4^{2i} cellules.

Type d'architecture employé chez Xilinx.





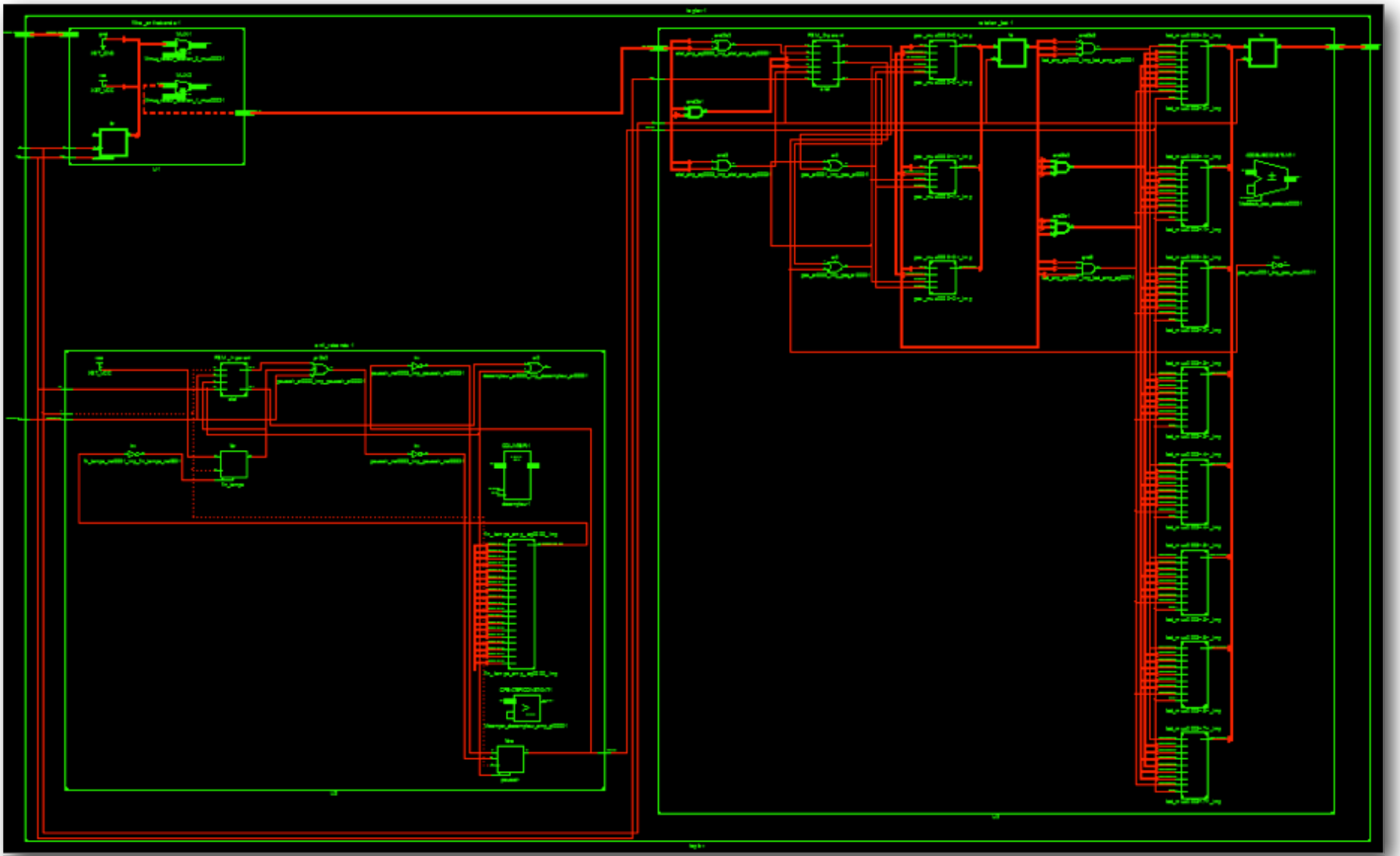
Les réseaux de routage d'une architecture hiérarchique dépendent du niveau de hiérarchie.

Architecture utilisée dans les FPGA de chez Altera et Lattice

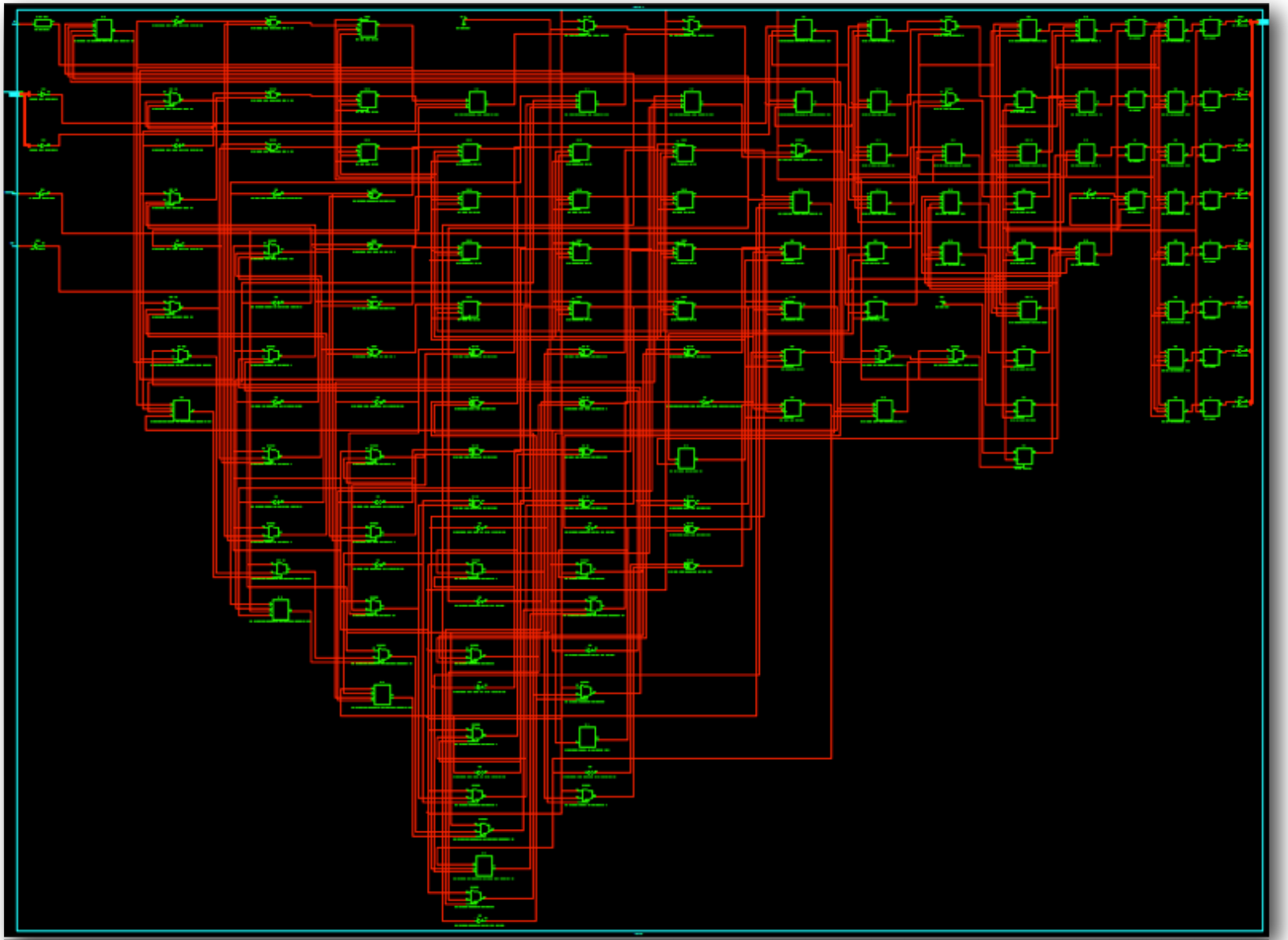
Back to the Past: votre premier TP de VHDL cette année...

Gestion du bouton rotatif
et allumage des LEDs

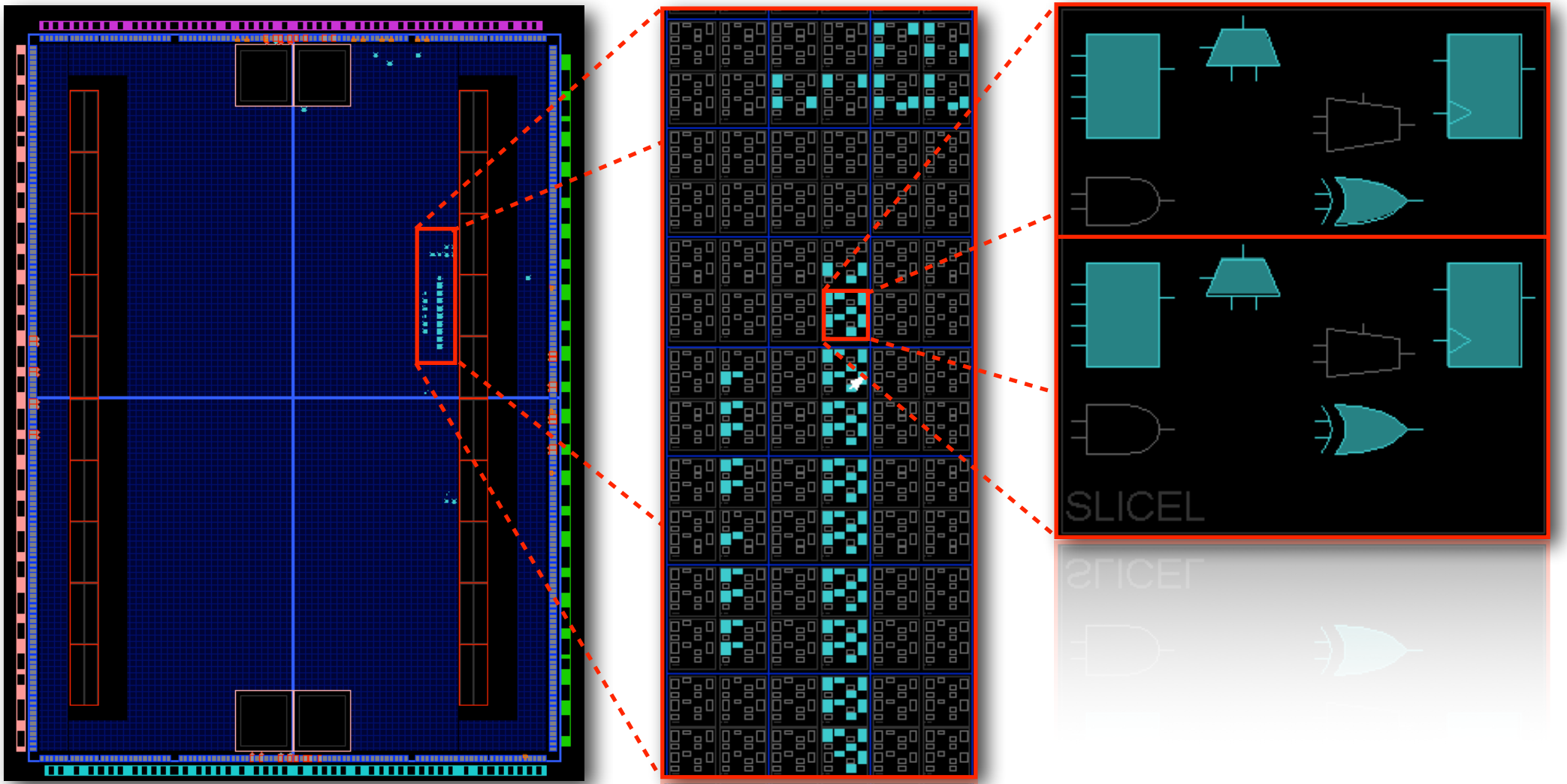
Observation des résultats post-synthèse (RTL)



Observation des résultats post-synthèse (technologie)



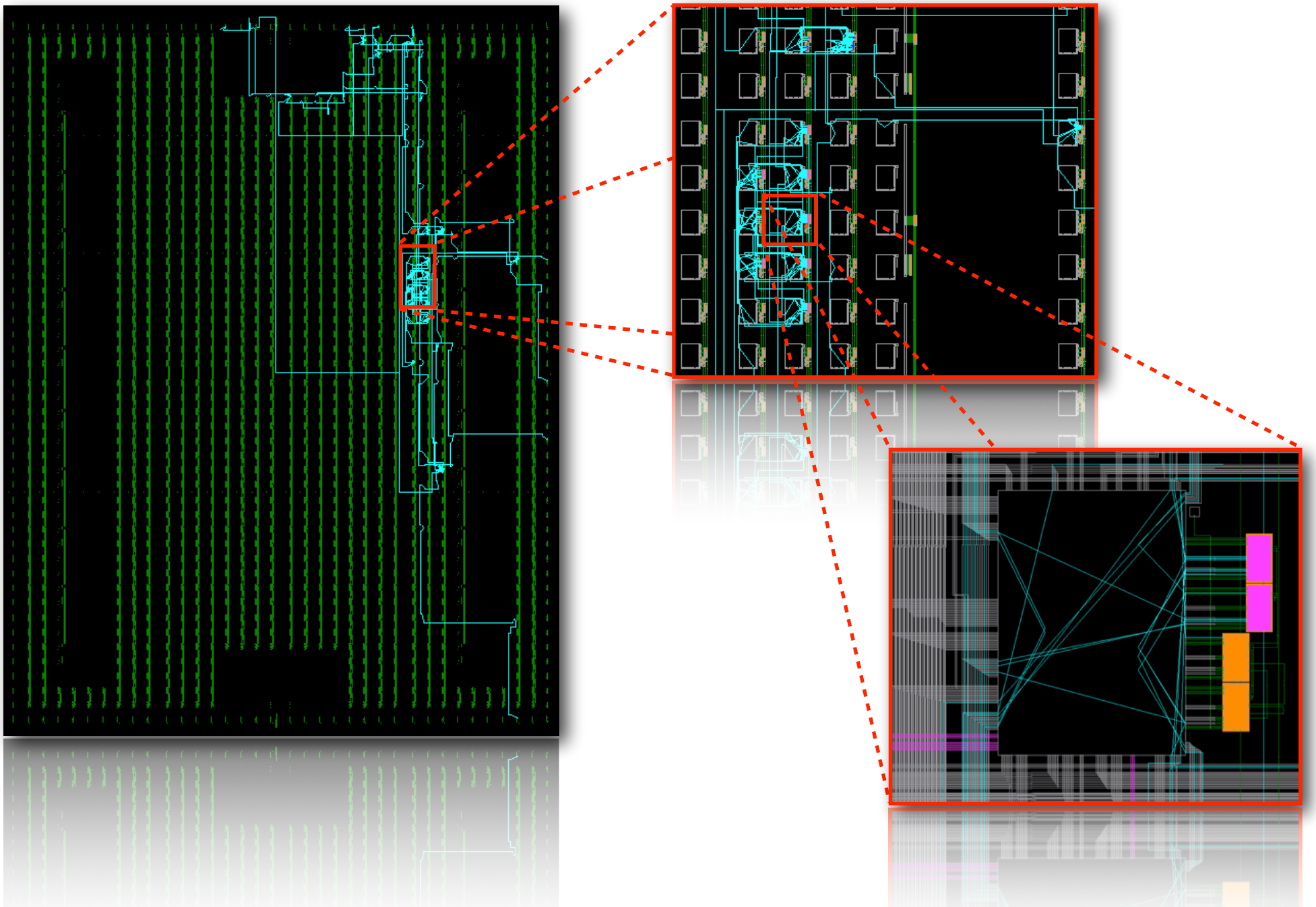
Observation des résultats post-placement & routage (1/2)



Logic Utilization:

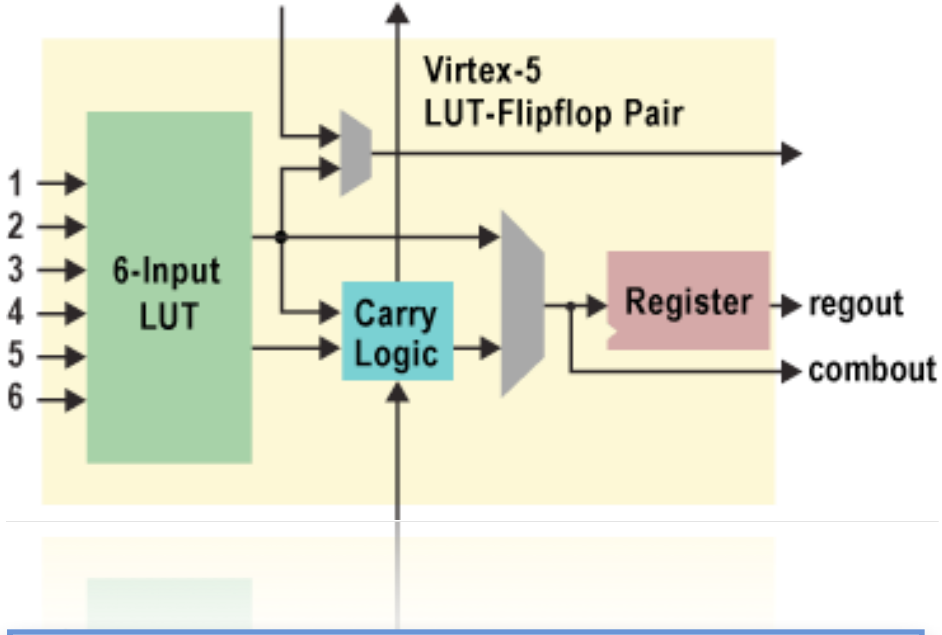
Number of Slice Flip Flops:	39 out of	9,312	1%
Number of 4 input LUTs:	53 out of	9,312	1%

Observation des résultats post-placement & routage (2/2)



Etude des FPGA de chez Altera

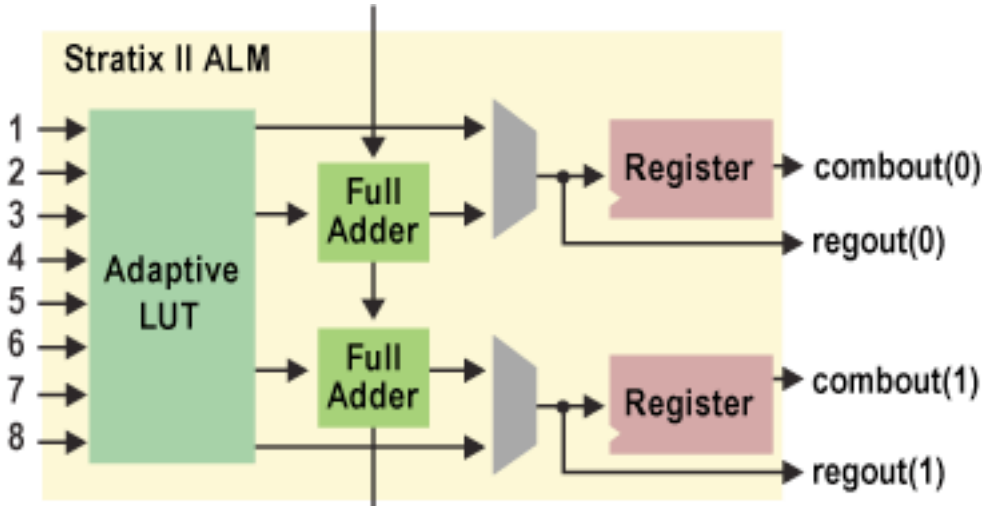
Décomposition de l'élément logique configurable chez Altera



L'association chez Xilinx d'une LUTn + d'une Flip-flop constitue une cellule de base.

Chez Altera, la cellule de base est plus complexe: ALUT + 2 registers + 2 FA

Adaptative LUT: une LUT8 à plusieurs sorties qui peut être découpée en LUTs de tailles inférieures pour maximiser l'occupation.



Décomposition de l'élément logique configurable chez Altera

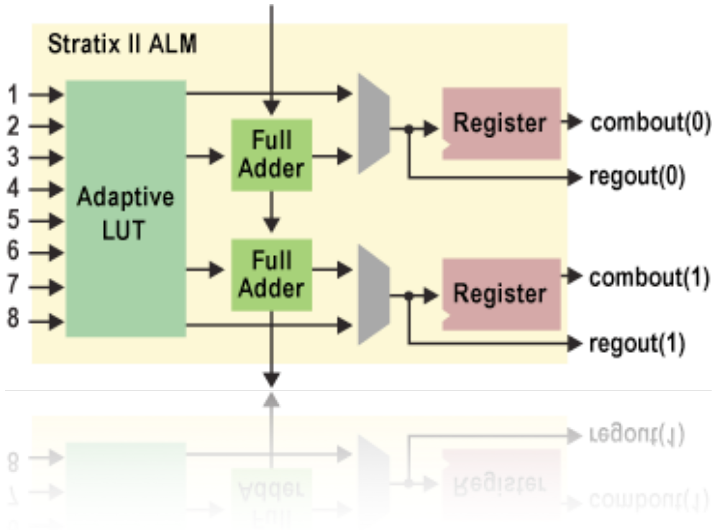
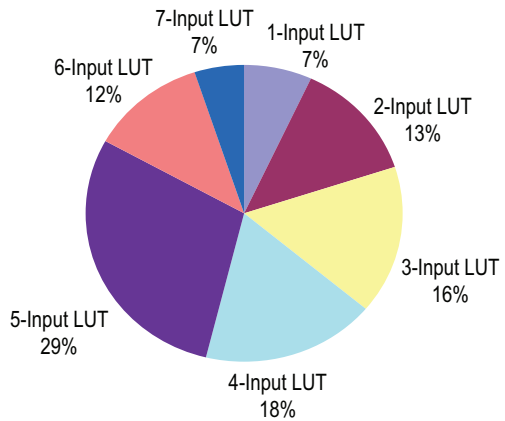
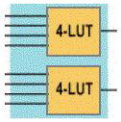
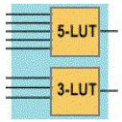
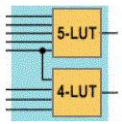
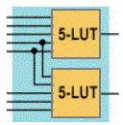
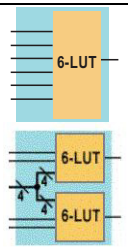
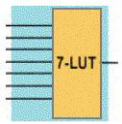


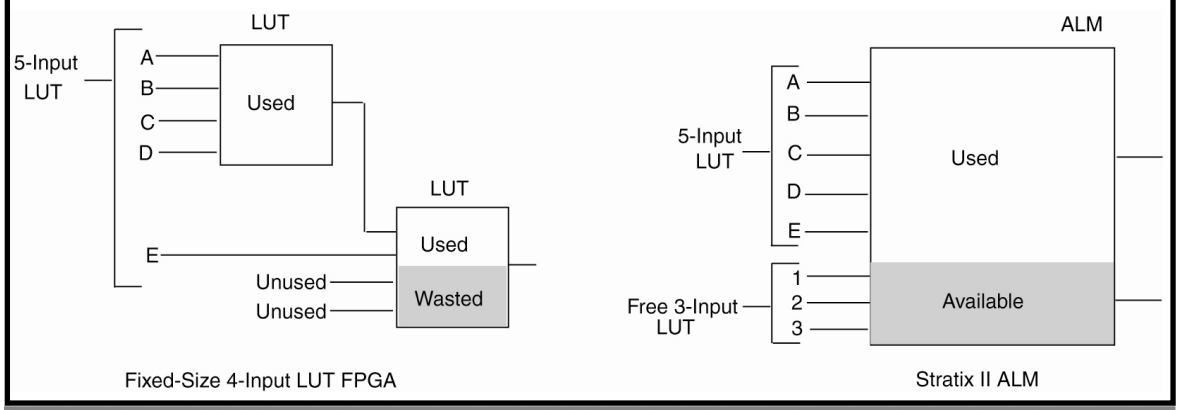
Figure 4. LUT Distribution Analysis



Configuration	Description
	One Stratix II ALM can be configured to implement two independent four-input (or smaller) LUTs. This configuration can be viewed as the "backward-compatibility" mode. Designs that are optimized for the traditional four-input LUT (4-LUT) FPGAs can easily be migrated to the Stratix II family.
	One Stratix II ALM can be configured to implement a five-input LUT (5-LUT) and a three-input LUT (3-LUT). The inputs to the two LUTs are independent of each other. The 3-LUT can be used to implement any logic function that has three or fewer inputs. Therefore, 5-LUT-2-LUT is also allowed.
	One Stratix II ALM can be configured to implement a five-input LUT and a four-input LUT. One of the inputs is shared between the two LUTs. The 5-LUT has up to four independent inputs. The 4-LUT has up to three independent inputs. The sharing of inputs between LUTs is very common in FPGA designs, and the Quartus II software automatically seeks logic functions that are structured in this manner.
	One Stratix II ALM can be configured to implement two five-input LUTs (5-LUTs). Two of the inputs between the LUTs are common, and up to three independent inputs are allowed for each 5-LUT.
	A Stratix II ALM can implement any six-input function (6-LUT). If there are two six-input functions that have the same logic operation and four shared inputs, then the two six-input functions can be implemented in one Stratix II ALM For example, a 4x2 crossbar switch that has four data input lines and two sets of unique select signals requires four LEs in the Stratix family. In the Stratix II family, this function only requires one ALM. Another example is a six-input AND gate. An ALM can implement two six-input AND gates that have four common inputs. The same function would require three LEs if implemented in a Stratix device.
	One Stratix II ALM in the extended mode can implement a subset of a seven-variable function. The Quartus II software automatically recognizes the applicable seven-input function and fits it into an ALM. Refer to the *Stratix II Device Handbook* for detailed information about the types of seven-input functions that can be implemented in an ALM.

Intérêt de la structure modulaire de la LUT8 de chez Altera

Figure 5. Five-Variable Function Implementation Comparison Between a Fixed-Size Four-Input LUT FPGA & Stratix II Devices



Virtex-5 LUT6

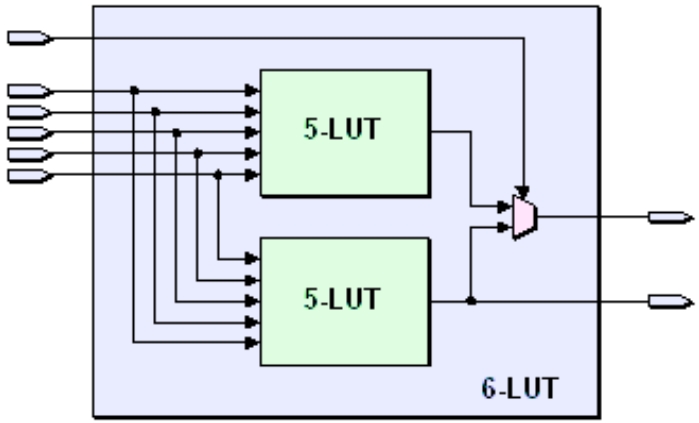
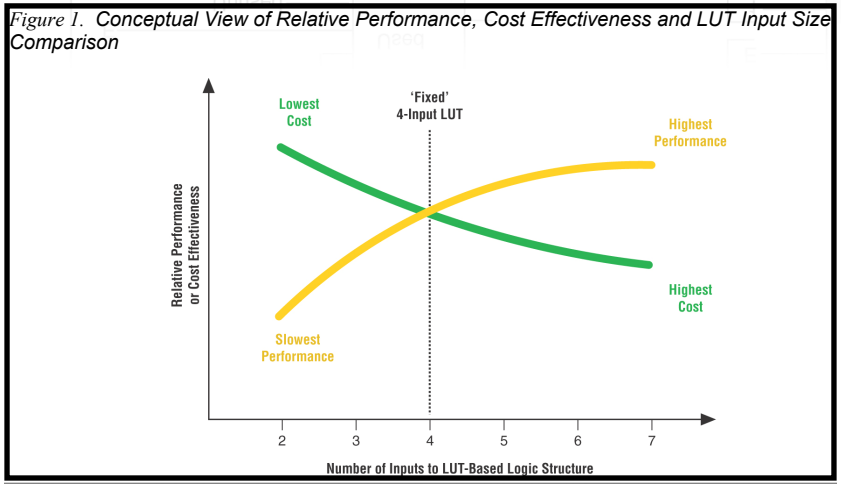


Figure 1. Conceptual View of Relative Performance, Cost Effectiveness and LUT Input Size Comparison



La LUT8 de chez Altera permet un meilleur taux d'utilisation des ressources versus les LUT4 et LUT6 (en théorie).

Etude plus fine de la LUT6 de chez Xilinx

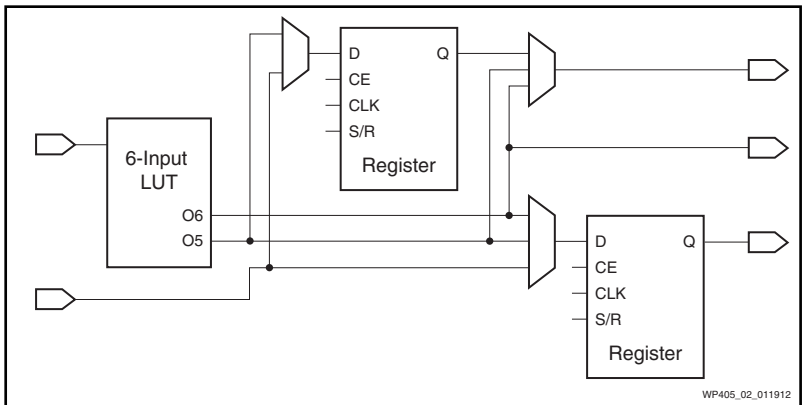


Figure 2: **Layout of 6-Input LUT and Two Registers within a Slice**

La LUT6 de chez Xilinx peut se décomposer en 2 LUT5 si les entrées sont identiques (différent chez Altera).

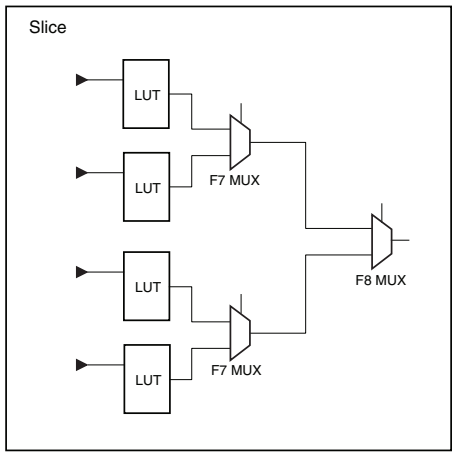


Figure 6: **Wide Logic Functions Using F7 and F8 Multiplexers**

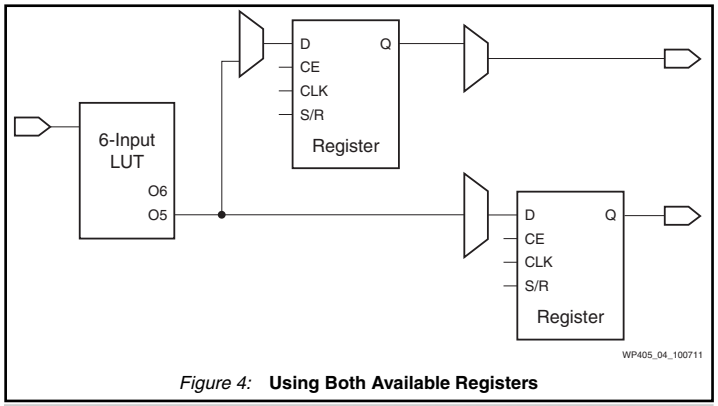


Figure 4: **Using Both Available Registers**

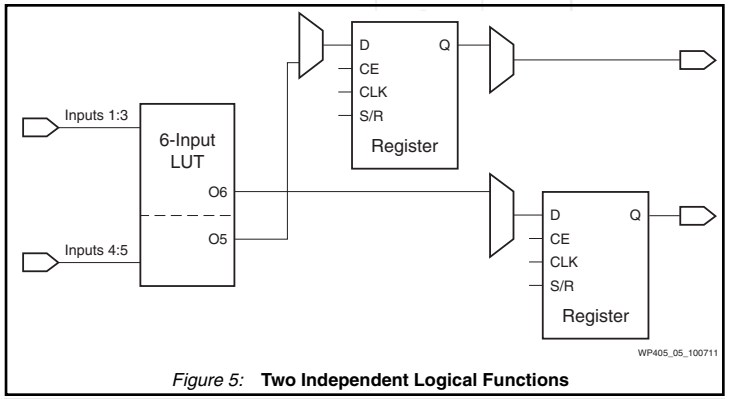


Figure 5: **Two Independent Logical Functions**

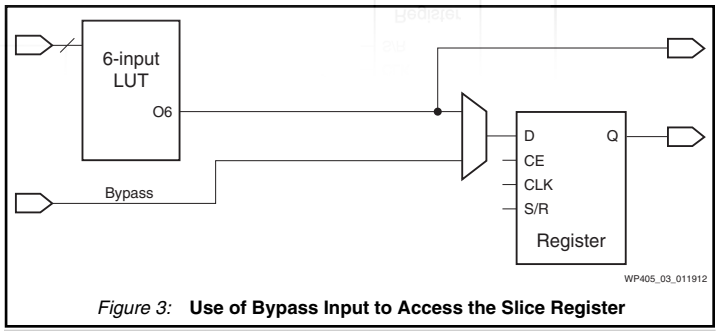


Figure 3: **Use of Bypass Input to Access the Slice Register**

Structuration du Stratix II, FPGA de chez Altera

1 LAB = 8 ALM

Figure 2–1. Stratix II Block Diagram

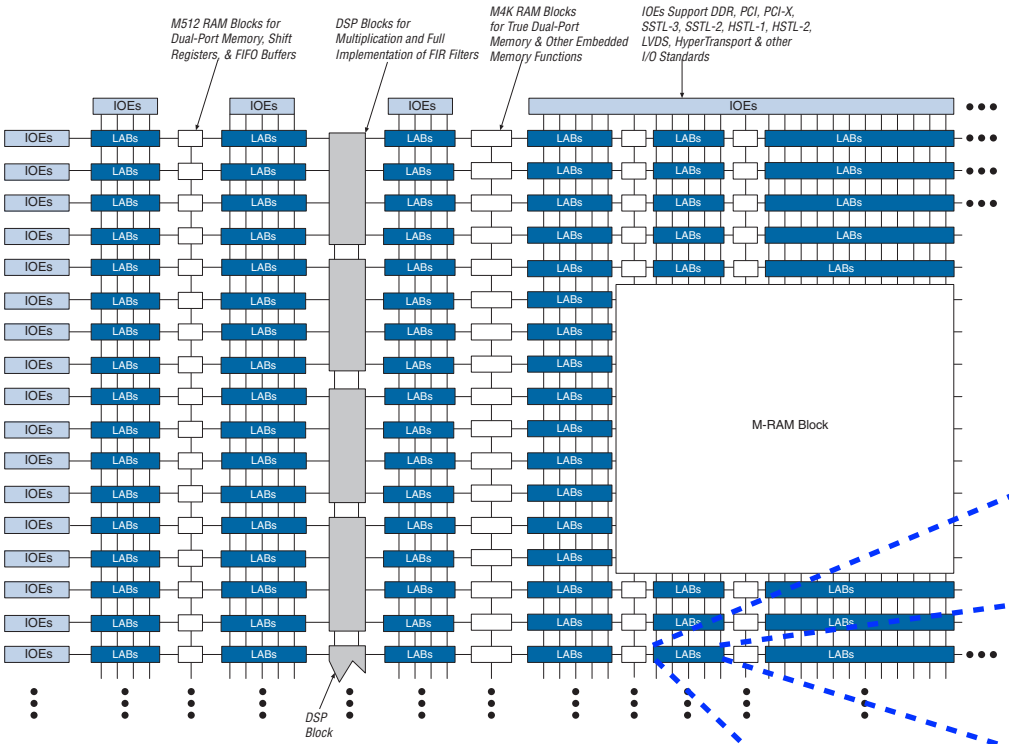


Figure 2–5. High-Level Block Diagram of the Stratix II ALM

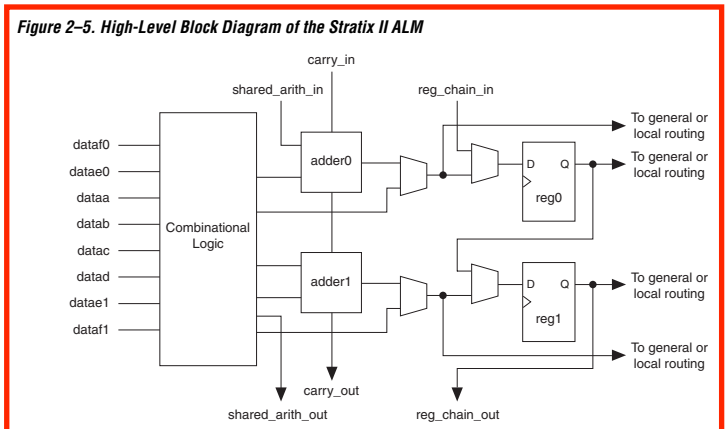


Figure 2–2. Stratix II LAB Structure

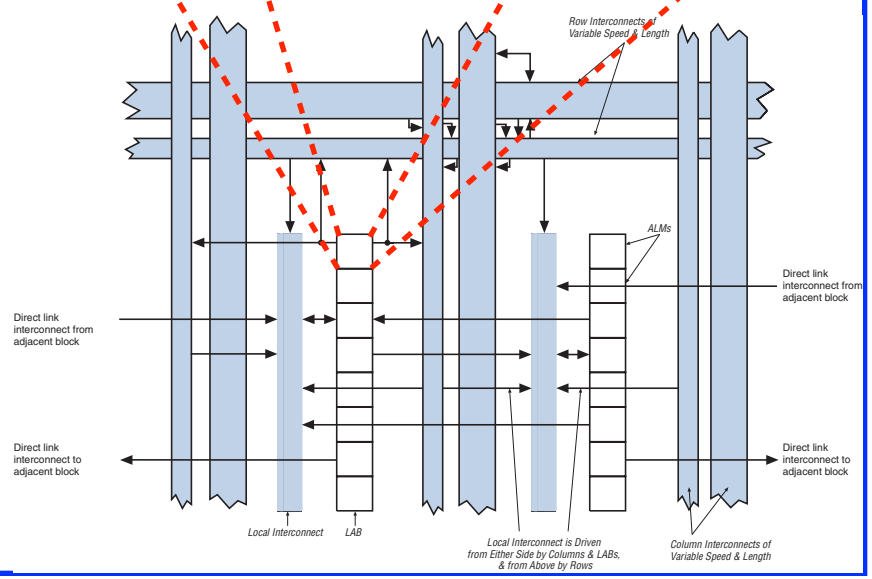
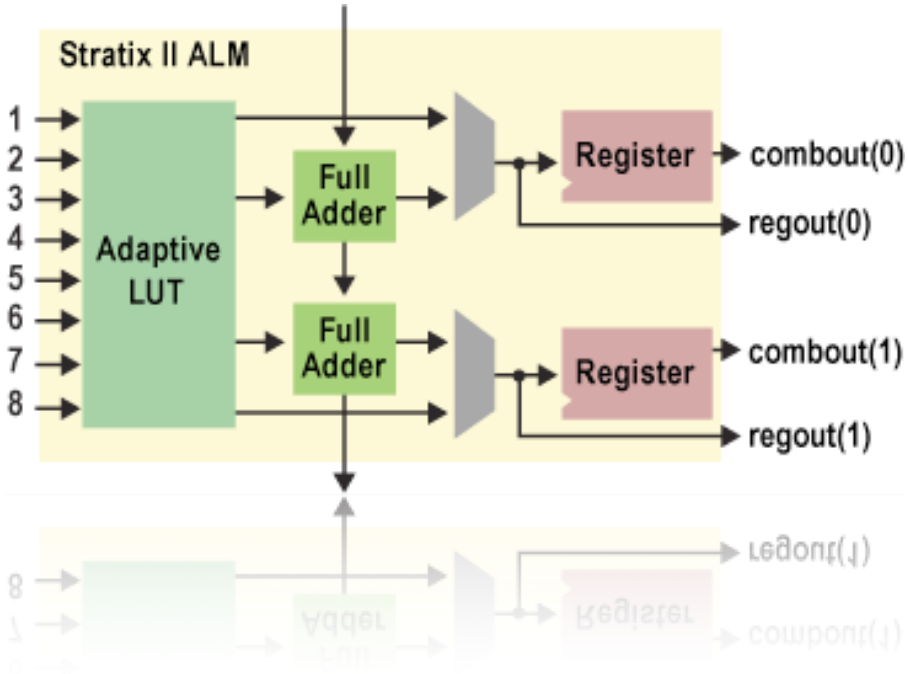


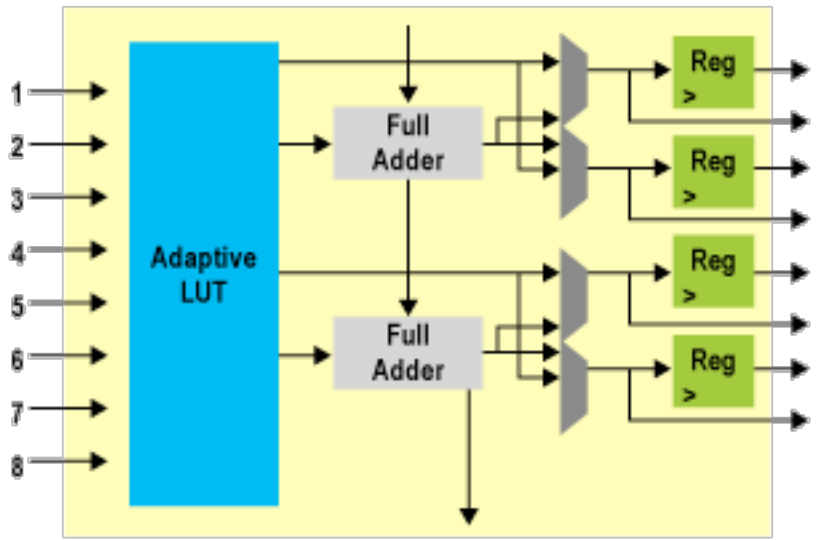
Table 2–1. Stratix II Device Resources

Device	M512 RAM Columns/Blocks	M4K RAM Columns/Blocks	M-RAM Blocks	DSP Block Columns/Blocks	LAB Columns	LAB Rows
EP2S15	4 / 104	3 / 78	0	2 / 12	30	26
EP2S30	6 / 202	4 / 144	1	2 / 16	49	36
EP2S60	7 / 329	5 / 255	2	3 / 36	62	51
EP2S90	8 / 488	6 / 408	4	3 / 48	71	68
EP2S130	9 / 699	7 / 609	6	3 / 63	81	87
EP2S180	11 / 930	8 / 768	9	4 / 96	100	96

Evolution de la structure de l'ALM (Stratix II vs Stratix V)



Stratix V (ALM structure)

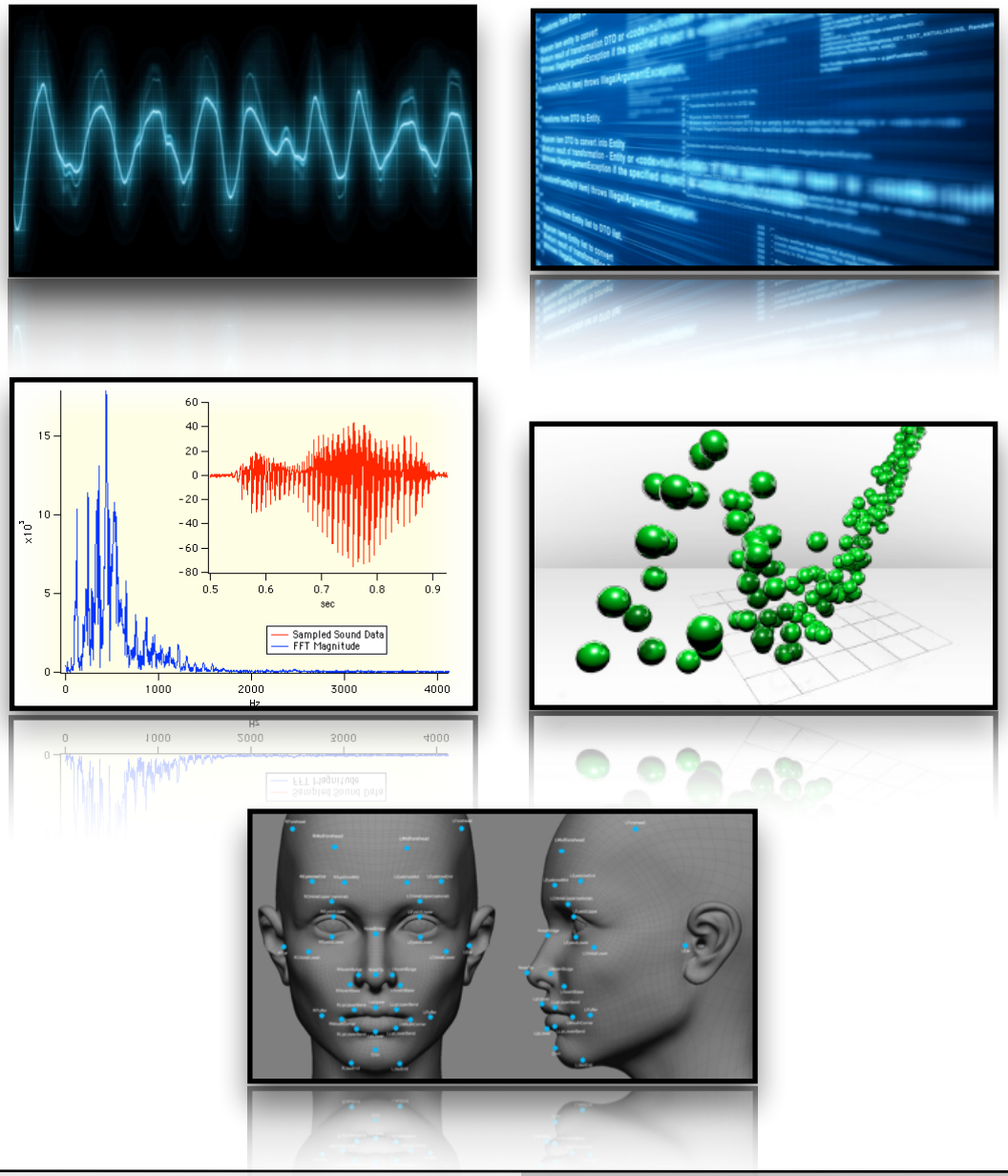


Augmentation du nombre de registre au sein de l'ALM (2 => 4)

Les blocs dédiés : Etude des DSP blocs

Présentation du besoin, contexte général

- Les FPGA sont utilisés pour réaliser des calculs intensifs,
  - ➡ Calcul scientifique, traitement vidéo, communication numérique, traitement du signal, etc.
- Cette applications sont consommatrices d'opérations arithmétiques (mult, mac, etc.).
- Une forte complexité calculatoire implique beaucoup de ressources de calcul,
  - ➡ Les structures à base de LUTs sont inefficaces...



Présentation du besoin, quelques chiffres (Virtex-4)

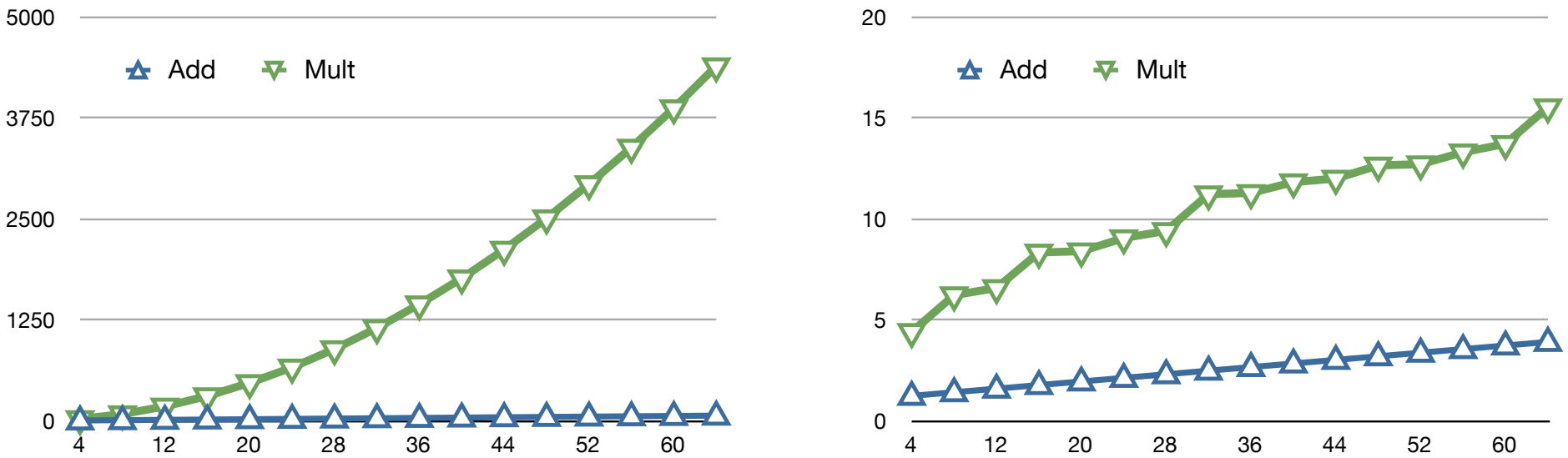
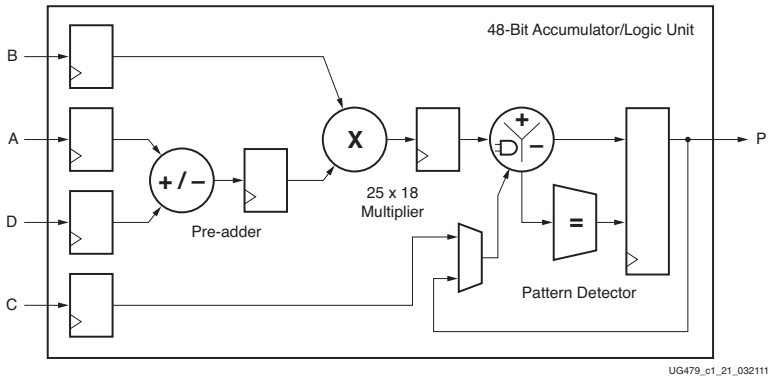


Figure 2 : *Évolution de caractéristiques des opérateurs (Add/Mult) entiers sur FPGA*

(a) Occupation en nombre de LUTs (b) Evolution du chemin critique en nanosecondes

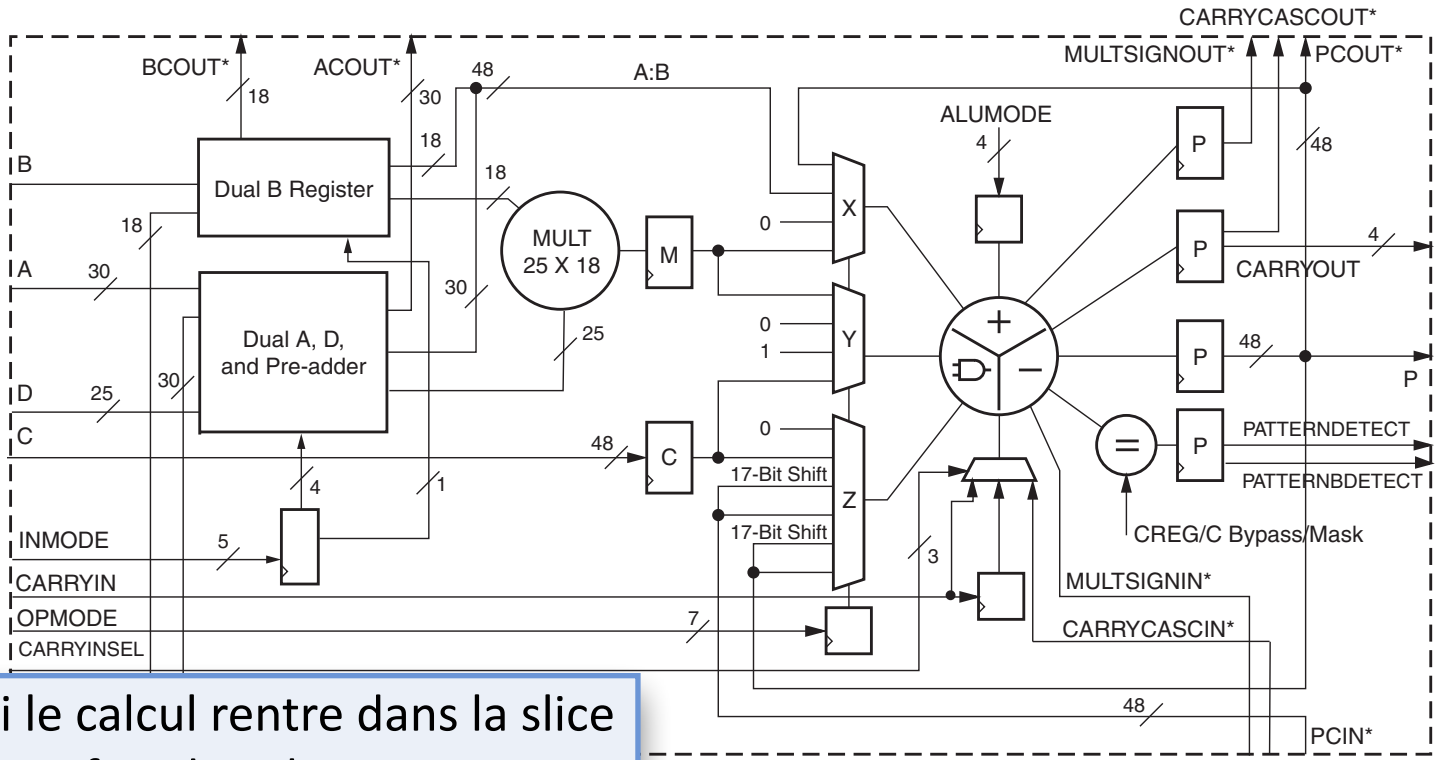
Implanter des opérations arithmétiques (i.e. les multiplications) à l'aide de LUTs devient vite inefficace. Cela est du au nombre de LUTs utilisées et au délais introduits par le routage (entre les LUTs).

Les blocs DSP48 de Xilinx (Virtex-4 => Virtex-7)



Développé pour optimiser les applications fortement calculatoires (signal processing).

Fonctionne jusqu'à 600MHz. Multiplieur 25x18 et accumulateur 48-bits.



Performances optimales si le calcul rentre dans la slice DSP (pipeline possible). Toutefois dans le cas contraire les performances chuttent vite à cause du routage...

Evaluation des performances des blocs DSP (multiplication)

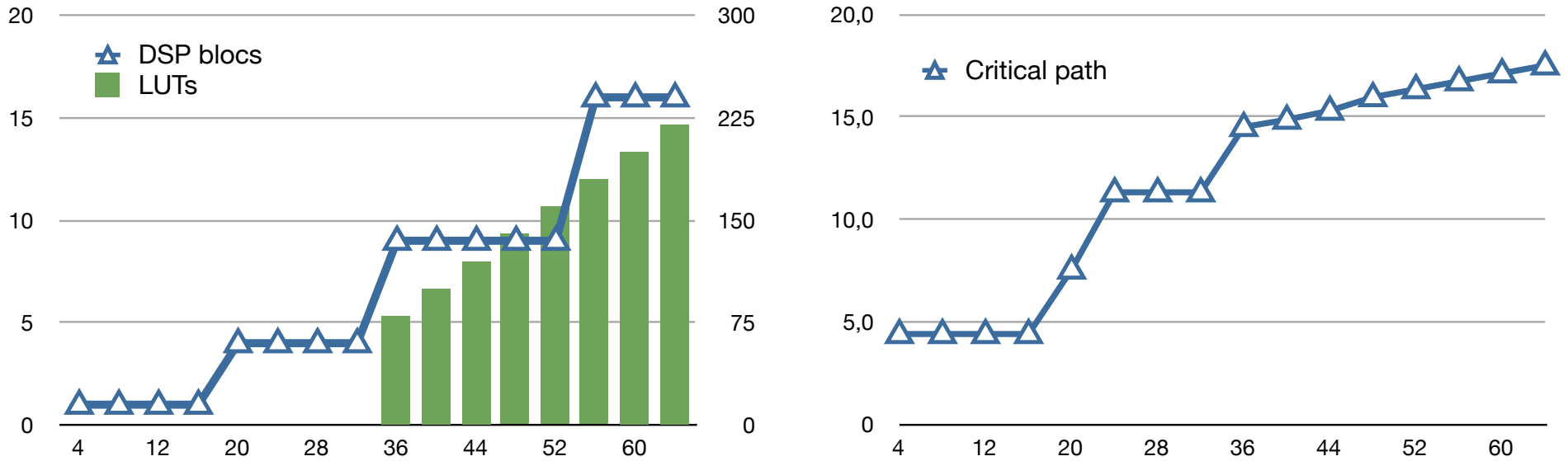
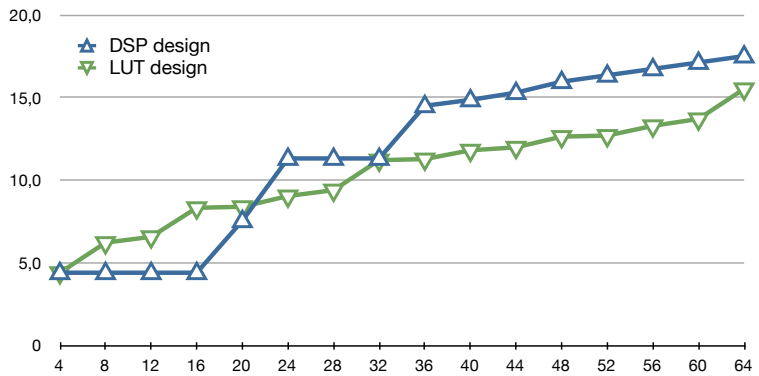


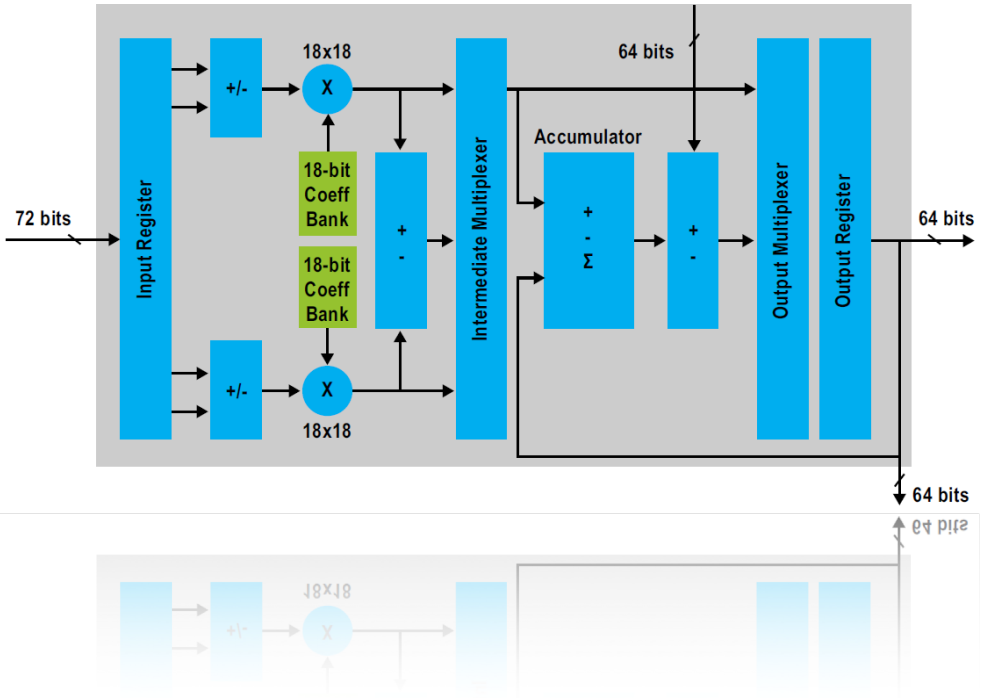
Figure 3 : *Évolution des caractéristiques des opérateurs de multiplication entiers implantés sur FPGA à l'aide de blocs DSP (a) Occupation en surface (b) Évolution du chemin critique en nanosecondes*



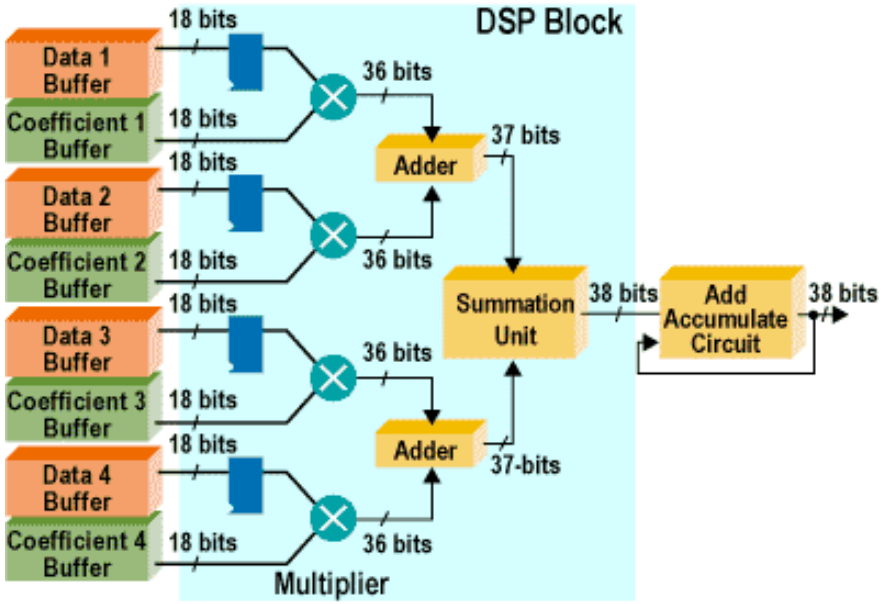
Lorsque les multiplications ont des entrées bien dimensionnées, les DSP48 sont très efficaces.

Figure 4 : *Comparaison des chemins critiques (nanosecondes) entre une implantation à l'aide de logique câblée versus l'utilisation de blocs matériels précablés en fonction de la dynamique des calculs.*

Le bloc DSP présent dans les PGA de chez Altera



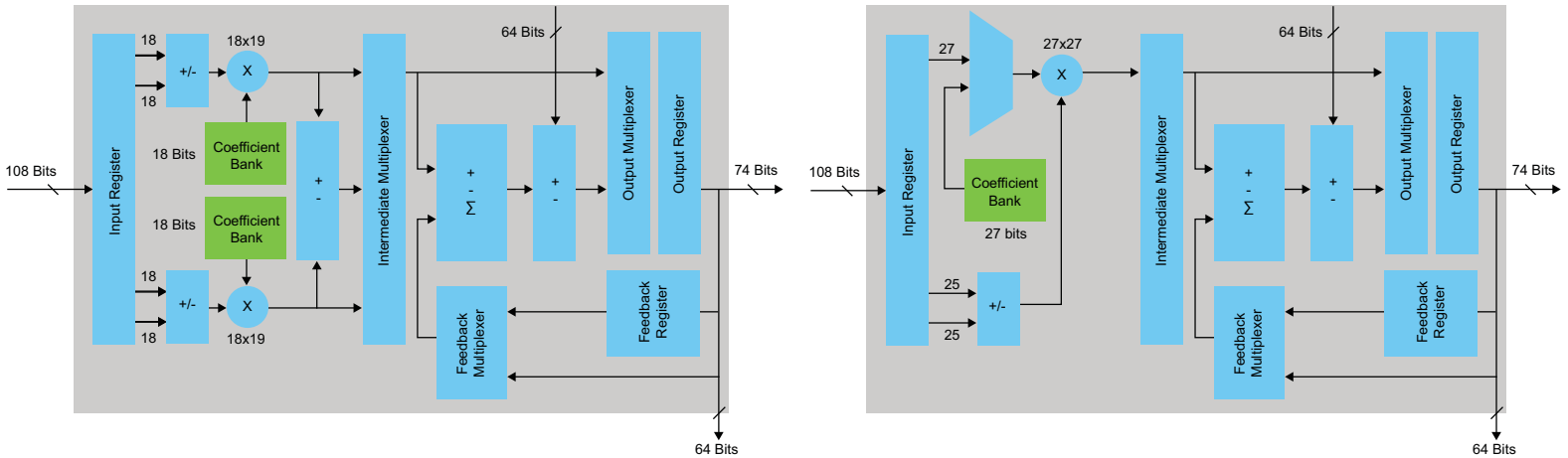
Les blocs DSP sont basés sur des multiplieurs 18x18.
4 multiplieurs par bloc DSP
(=>2 par ½ bloc)



Un bloc DSP permet d'implanter efficacement un filtre FIR 4 points
(si le format de données est compatible).

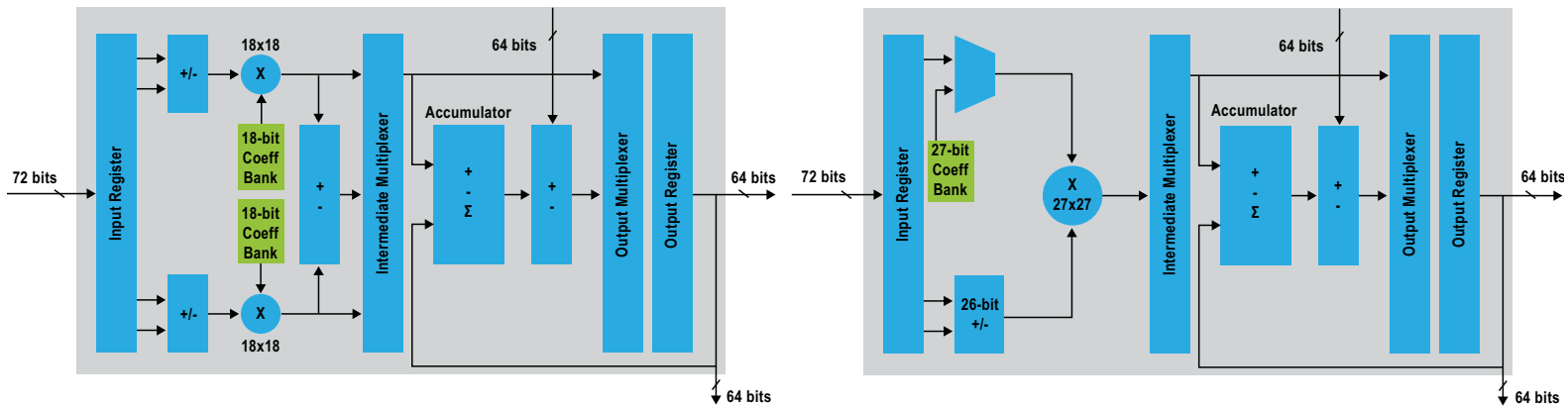
Evolution du bloc DSP dans les différentes familles de FPGA

Figure 3. Arria V and Cyclone V 18-Bit Precision and High-Precision Modes



18-Bit Precision Mode High-Precision Mode

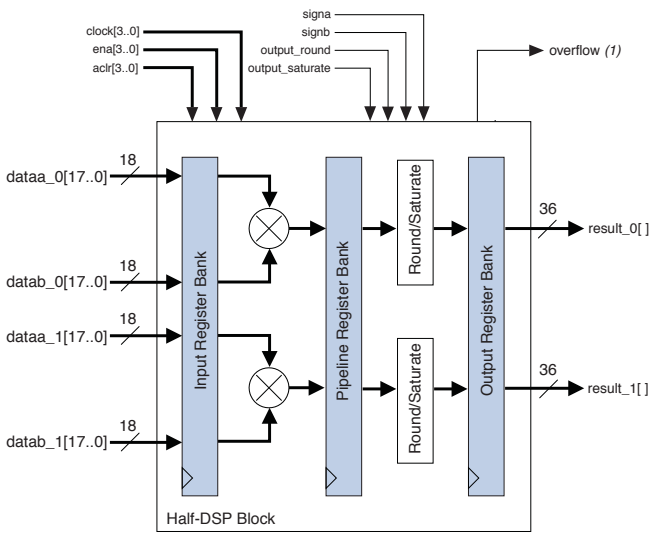
Figure 4. Stratix V 18-bit Precision and High-Precision Modes



18-Bit Precision Mode High-Precision Mode

Partitionnement du bloc DSP en fonction des applications

Figure 4–8. 18-Bit Independent Multiplier Mode Shown for a Half DSP Block



Note to Figure 4–8:

(1) Block output for accumulator overflow and saturate overflow.

Figure 4–9. 12-Bit Independent Multiplier Mode Shown for a Half DSP Block

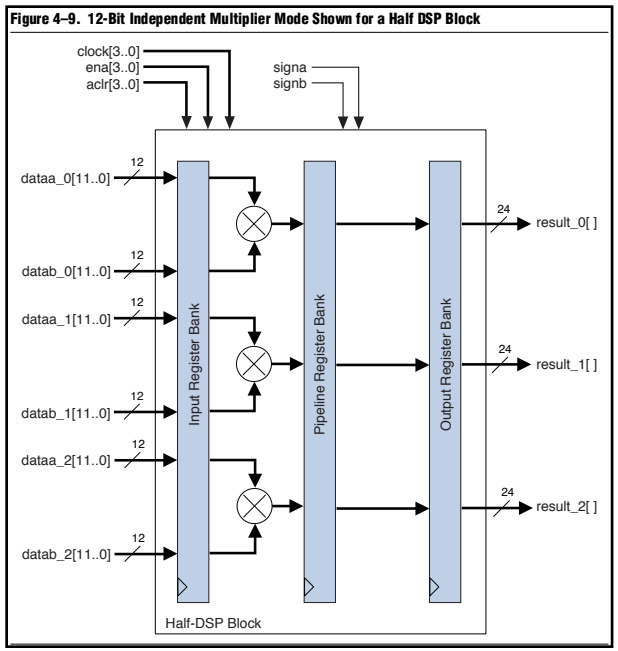
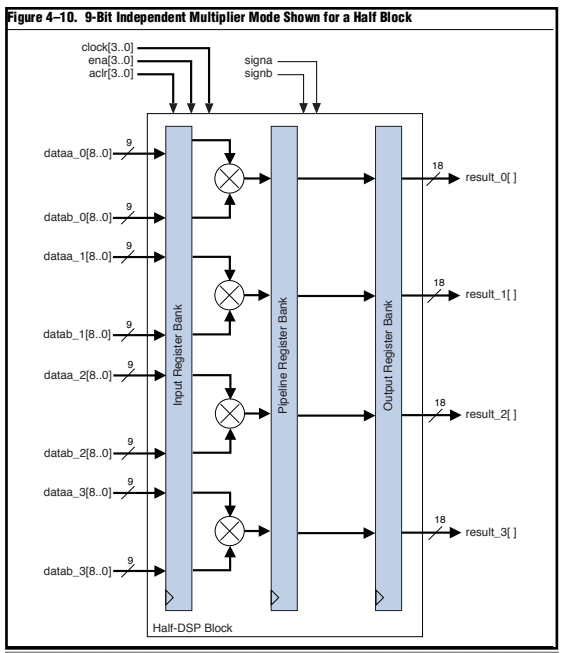


Figure 4–10. 9-Bit Independent Multiplier Mode Shown for a Half Block



Afin de s'adapter aux applications (format de données), les DSP blocs peuvent être scindés en **N** opérateurs de taille inférieur (multiplication).

Les blocs dédiés : Etude des blocs RAM

Les blocs mémoire, étude des blocs RAM du Spartan-6

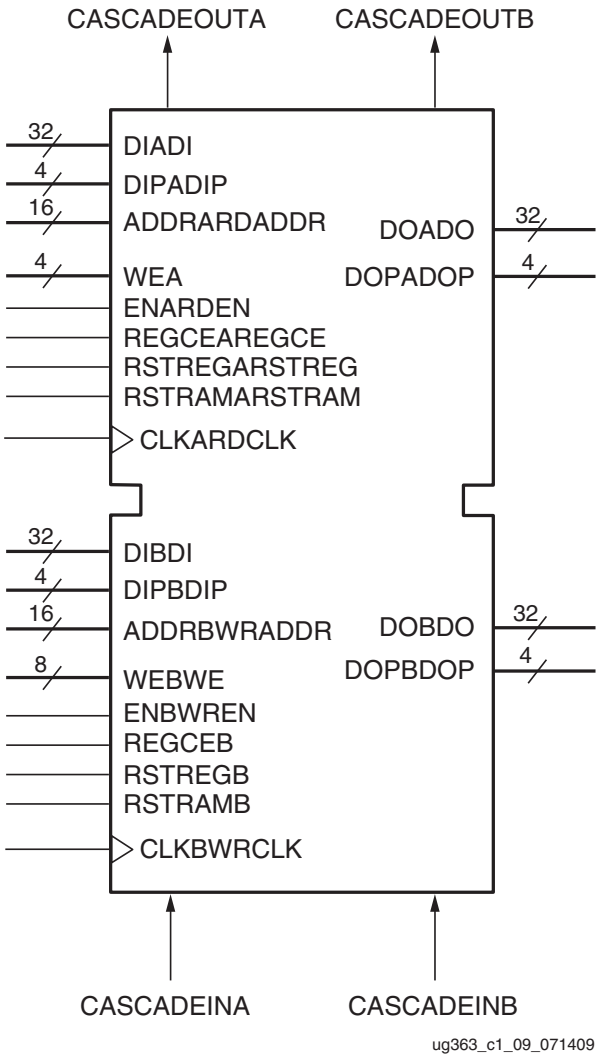


Figure 1-9: **Block RAM Port Signals (RAMB36E1)**

Les DSP48 ont été introduits pour augmenter l'éfficacité calculatoire.

Les blocs RAM ont été ajouté pour permettre une mémorisation efficace des informations.

Les blocs RAM sont répartis en colonne dans la matrice du FPGA.

La taille des blocs varie en fonction des FPGAs ainsi que la capacité totale.

SPARTAN-6: 12 à 268 blocs de 18Kb (max. 4820kb).

Les blocs mémoire, étude des blocs RAM du Spartan-6

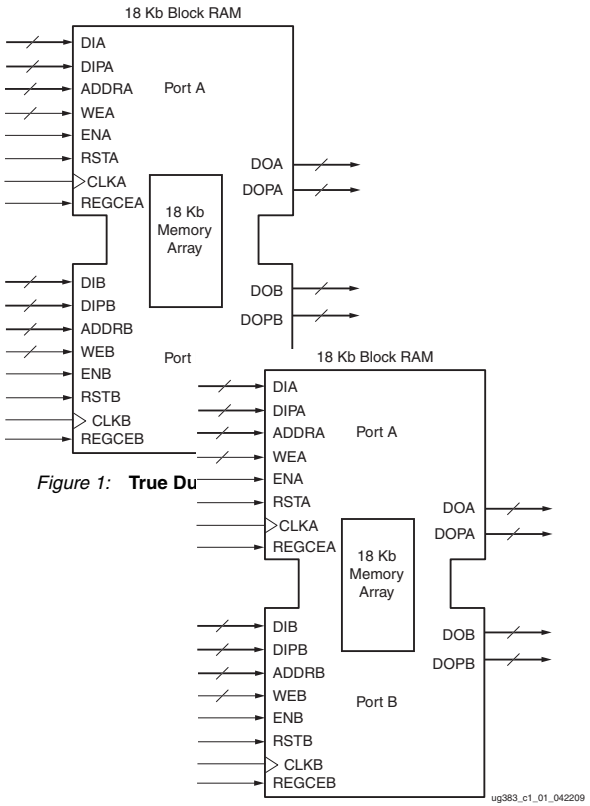


Figure 1: **True Dual-Port Data Flows**

Combinations	Memory Depth
9 Kb Block RAM With and W	
256 x 32(1)	256
256 x 36(1)	256
512 x16	512
512 x18	512
1K x 8	1024
1K x 9	1024
2K x 4	2048
4K x 2	4096
8K x 1	8192
18 Kb Block RAM With and V	
512 x 32	512
512 x 36	512
1K x16	1024
1K x18	1024
2K x 8	2045
2K x 9	2048
4K x 4	4096
8K x 2	8192
16K x 1	16384

Blocs configurables (width x depth) et assocciables(dual-port ou capacité supérieure).

Table 2: **True Dual-Port Mode Allowed Combinations for 9 Kb Block RAM**

		Port A						
		No Parity Bits					With Parity Bits	
		8K x 1	4K x 2	2K x 4	1K x 8	512 x 16	1K x 9	512 x 18
Port B	8K x 1	All Allowed					None Allowed	
	4K x 2							
	2K x 4							
	1K x 8							
	512 x 16							
	1K x 9	None Allowed					All Allowed	
	512 x 18							

Table 1: **Simple Dual-Port Mode Allowed Combinations for 9 Kb Block RAM**

			Port A								
			No Parity Bits						With Parity Bits		
			8K x 1	4K x 2	2K x 4	1K x 8	512 x 16	256 x 32	1K x 9	512 x 18	256 x 36
Port B	No parity bits	8K x 1	Use true dual-port mode					None Allowed	None Allowed		
		4K x 2									
		2K x 4									
		1K x 8									
		512 x 16									
		256 x 32	None Allowed					Allowed			
	With parity bits	1K x 9	None Allowed						Use true dual-port mode		None Allowed
		512 x 18									
		256 x 36							None Allowed		Allowed

Table 3: **True Dual-Port Mode Allowed Combinations for 18 Kb Block RAM**

		Port A								
		No Parity Bits						With Parity Bits		
		16K x 1	8K x 2	4K x 4	2K x 8	1K x 16	512 x 32	2K x 9	1K x 18	512 x 36
Port B	16K x 1	All Allowed						None Allowed		
	8K x 2									
	4K x 4									
	2K x 8									
	1K x 16									
	512 x 32									
	2K x 9	None Allowed						All Allowed		
	1K x 18									
	512 x 36									

Les blocs mémoire, étude des blocs RAM du Virtex-6

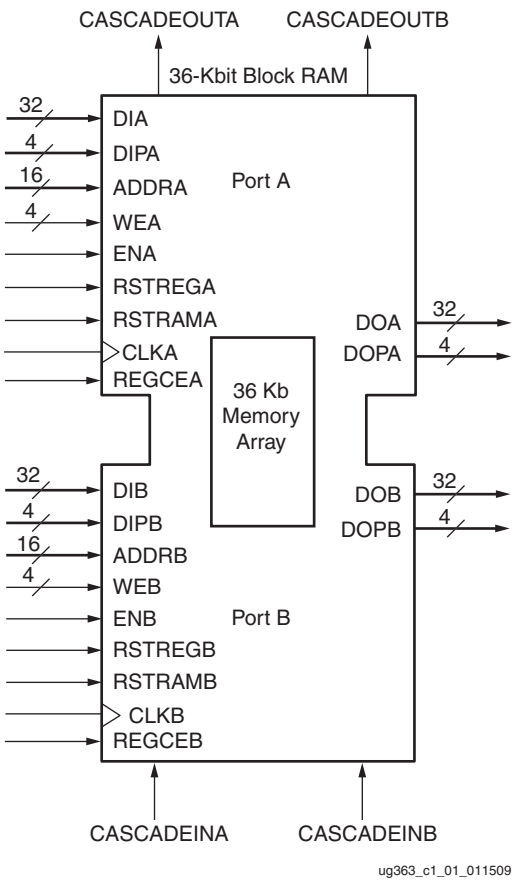


Figure 1-1: **True Dual-Port Data Flows for a RAMB36**

Table 1-5: **Virtex-6 FPGA Block RAM and FIFO Primitives**

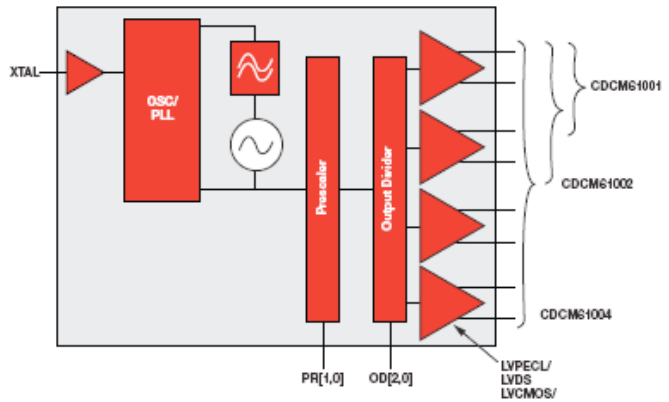
Primitive	Description
RAMB36E1	In TDP mode, supports port widths of x1, x2, x4, x9, x18, x36 In SDP mode, the Read or Write port width is x64 or x72. Alternate port is x1, x2, x4, x9, x18, x36. In ECC mode, supports 64-bit ECC encoding and decoding
RAMB18E1	In TDP mode, supports port widths of x1, x2, x4, x9, x18 In SDP mode, the Read or Write port width is x32 or x36. Alternate port is x1, x2, x4, x9, x18.
FIFO36E1	In FIFO36 mode, supports port widths of x4, x9, x18, x36 In FIFO36_72 mode, port width is x72, optional ECC support.
FIFO18E1	In FIFO18 mode, supports port widths of x4, x9, x18 In FIFO18_36 mode, port width is x36

La taille des blocs mémoire du Virtex-6 est 2 fois supérieure à celle du Spartan-6.
Cette approche est avantageuse ou pas en fonction des besoins mémoire...
Jusqu'à 912 blocs RAM36kb disponiles => 32832kb.

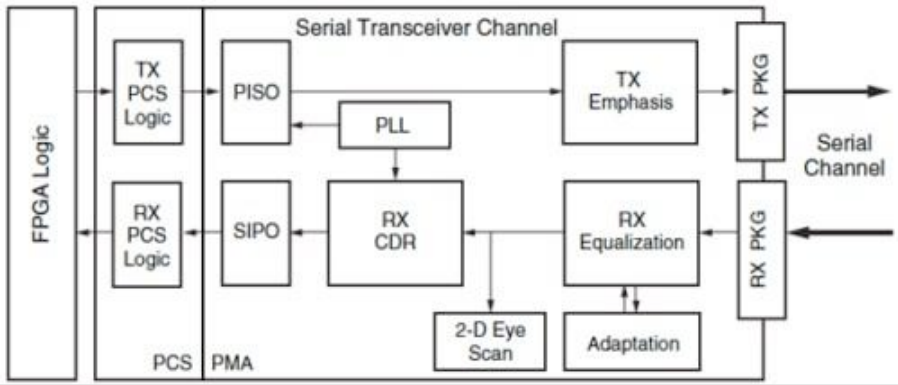
Les blocs dédiés : autres ressources

Les blocs usuels dans les FPGA actuels

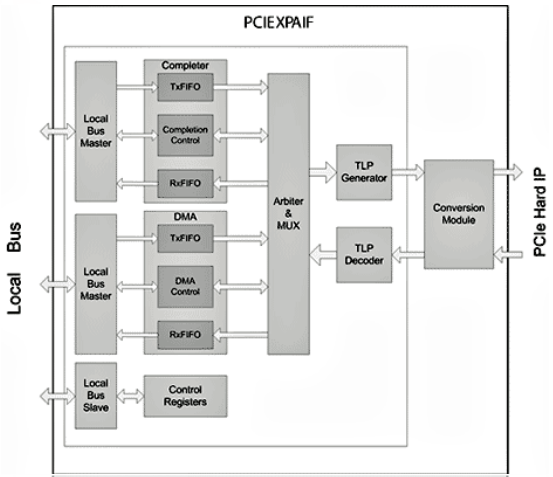
Horloge (PLL)



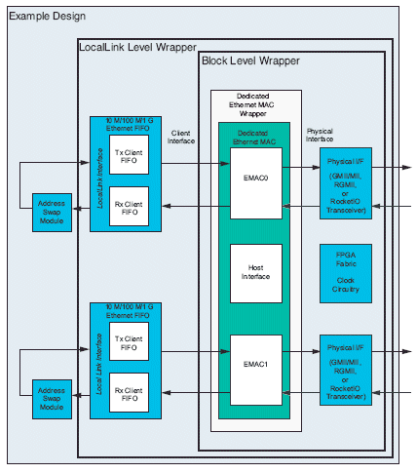
Lien rapide: SERDES



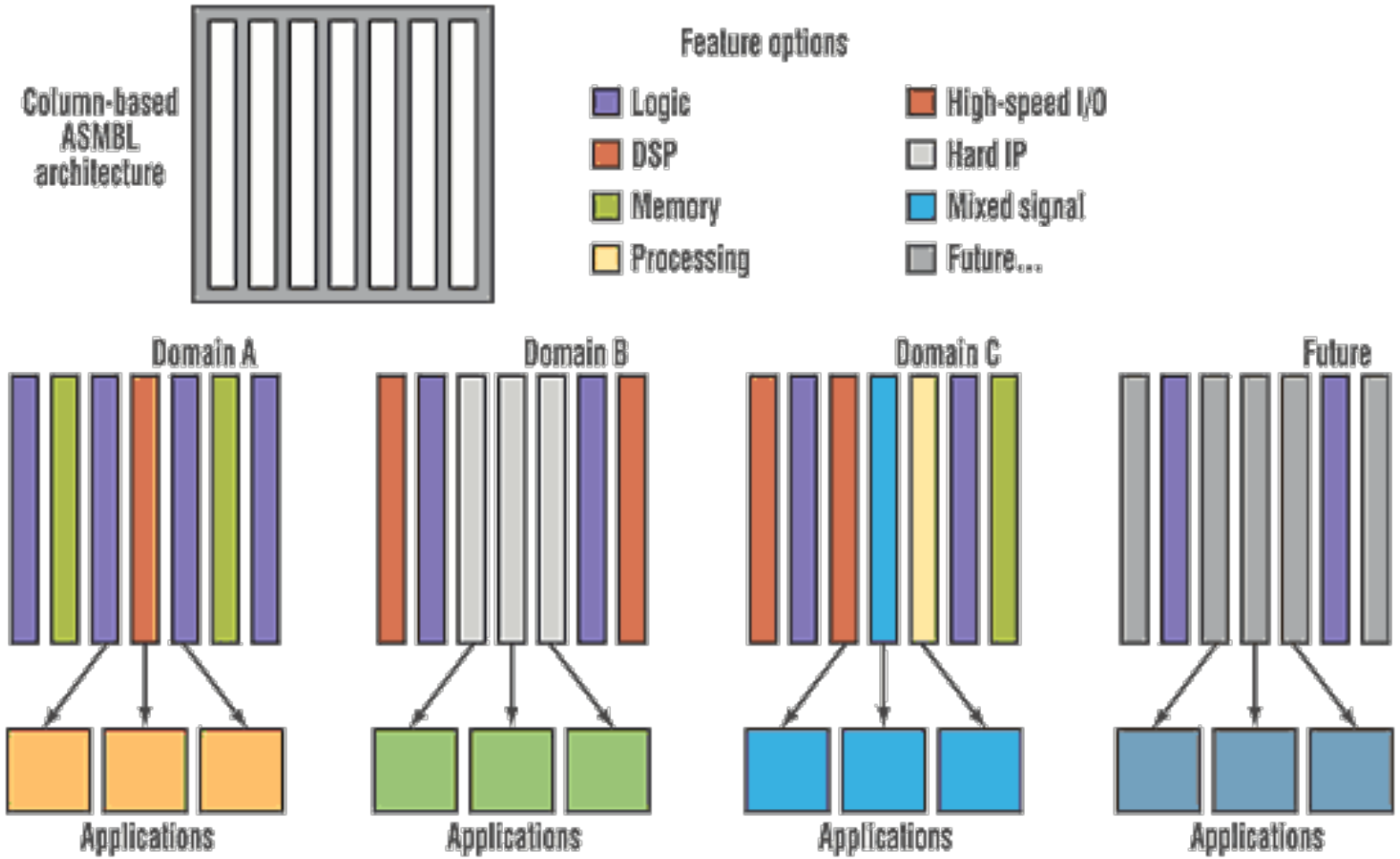
Interface PCIe



Interface Ethernet



La différenciation des FPGA en fonction des besoins applicatifs



With the modular architecture developed by Xilinx for its ASMBL family of FPGAs, function-specific stripes of logic can be incorporated into the silicon to optimize the FPGA for an application domain.

La différenciation des FPGA en fonction des besoins applicatifs

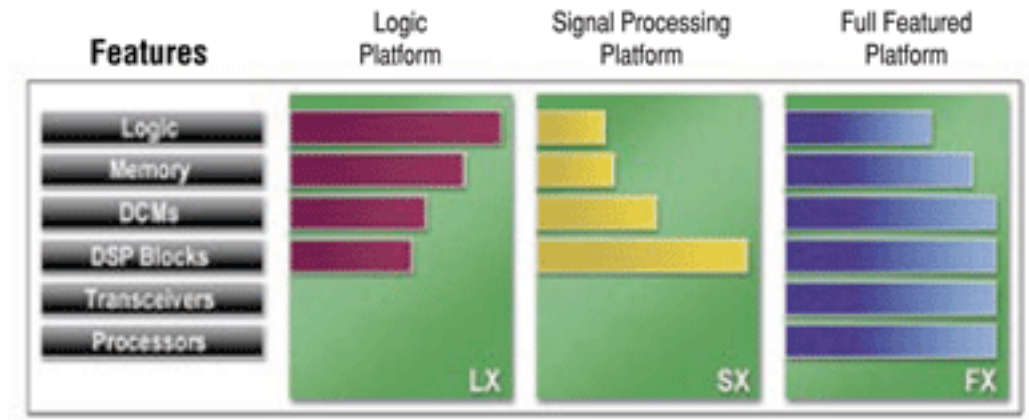
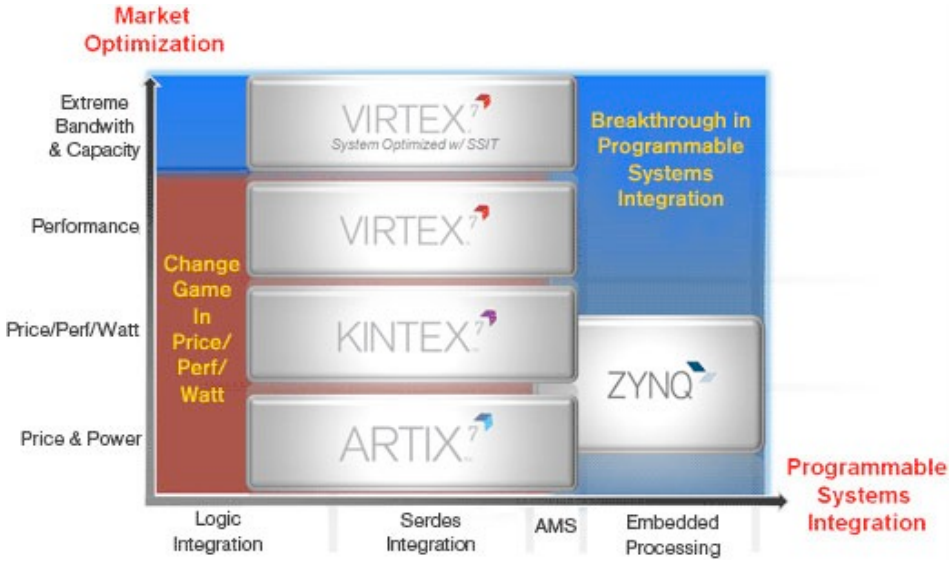
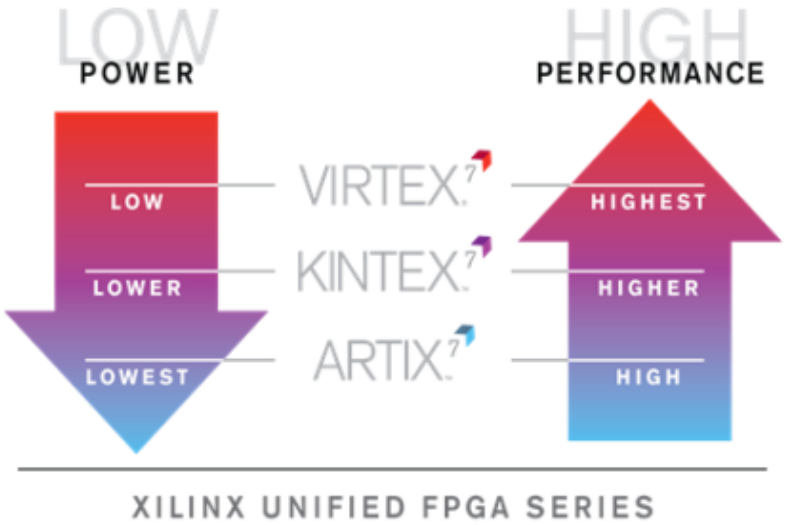


Figure 2 The ASMBL-enabled FPGA currently offers three design platforms, addressing high-density logic, DSP-centric, and embedded and high-speed serial application domains.



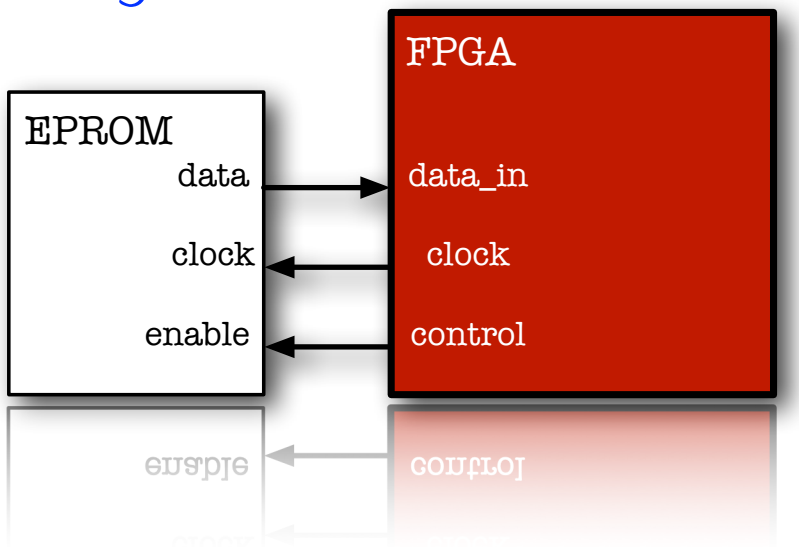


Processus de configuration d'un circuit FPGA

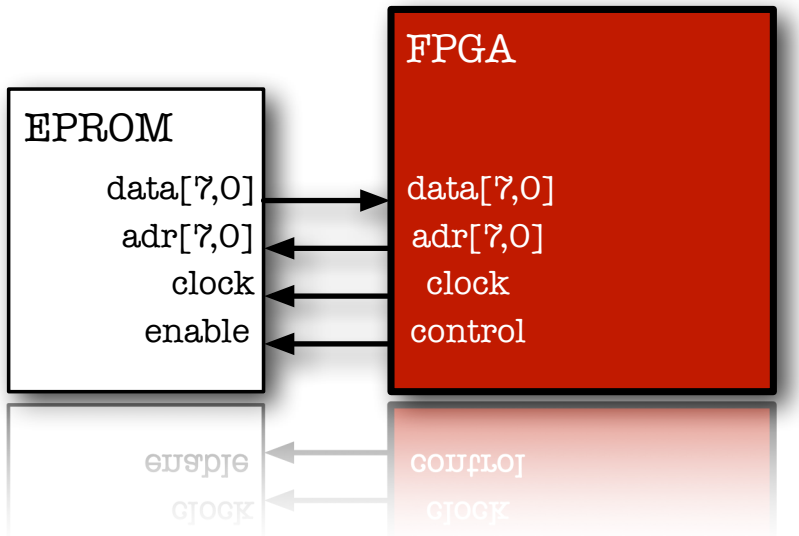
Les modes de configuration d'un FPGA

- Le FPGA en mode «Master»
  - Le FPGA est maitre de sa configuration,
- Au démarrage, il vient charger les données à partir d'une mémoire externe,
  - La mémoire (EPROM) contient le fichier bitstream,
  - Le chargement du bitstream peut être série ou parallèle.
- Mode de configuration généralement utilisé en production (on board).

Chargement série des données

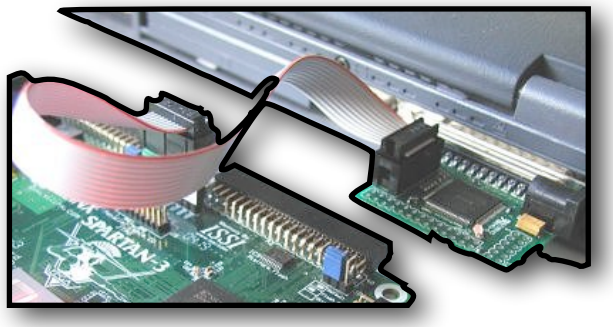


Chargement parallèle des données

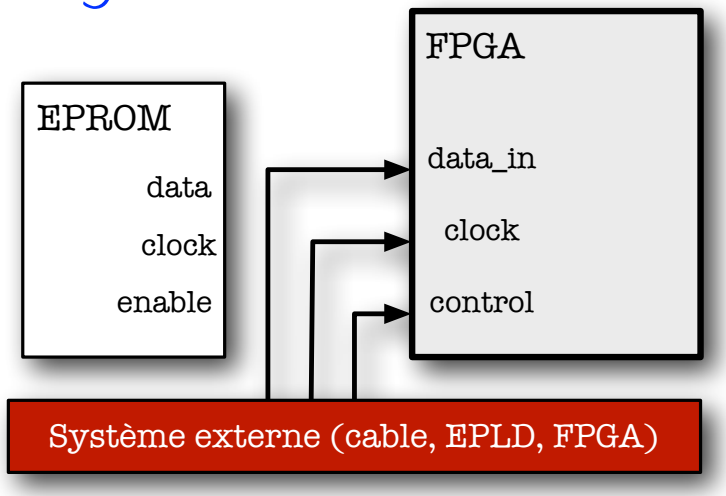


Les modes de configuration d'un FPGA

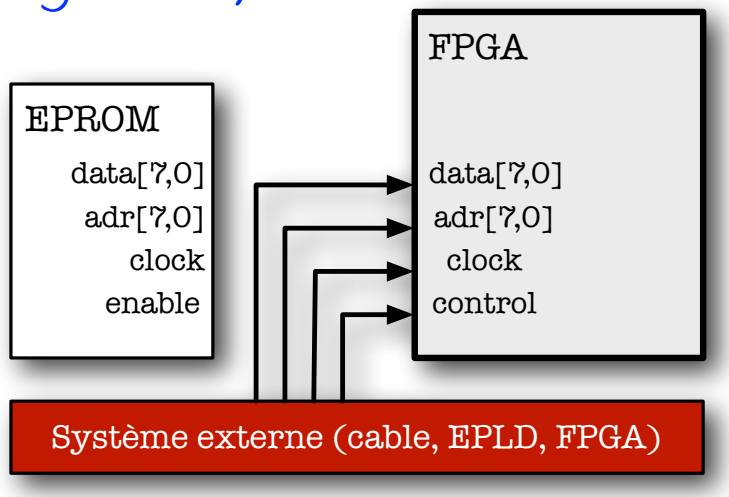
- Le mode «Slave»
  - ➡ Le FPGA est subit la configuration,
- Au cours du temps un acteur extérieur configure le FPGA,
  - ➡ Le bitstream est transmis depuis l'acteur extérieur vers le FPGA,
- Mode de configuration utilisé en phase de développement,
  - ➡ Ce que vous faites en TP...



Chargement série des données

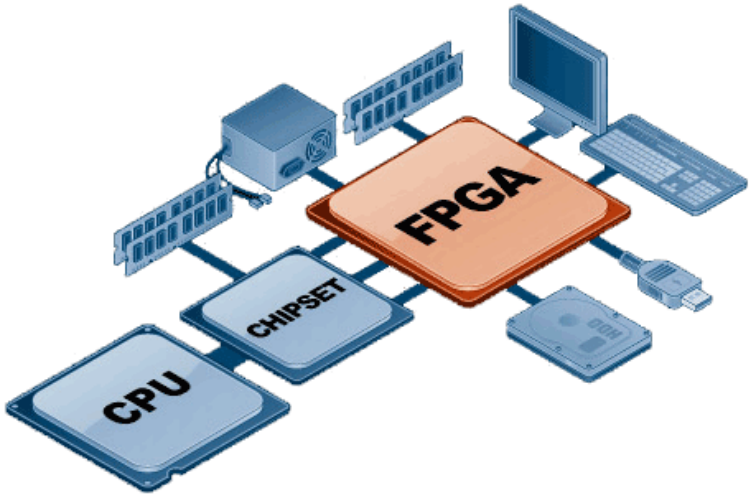


Chargement parallèle des données

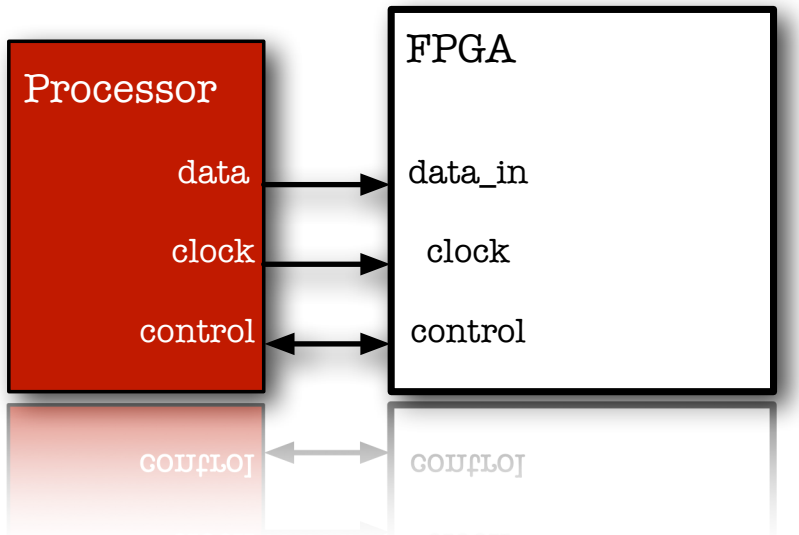


Les modes de configuration d'un FPGA

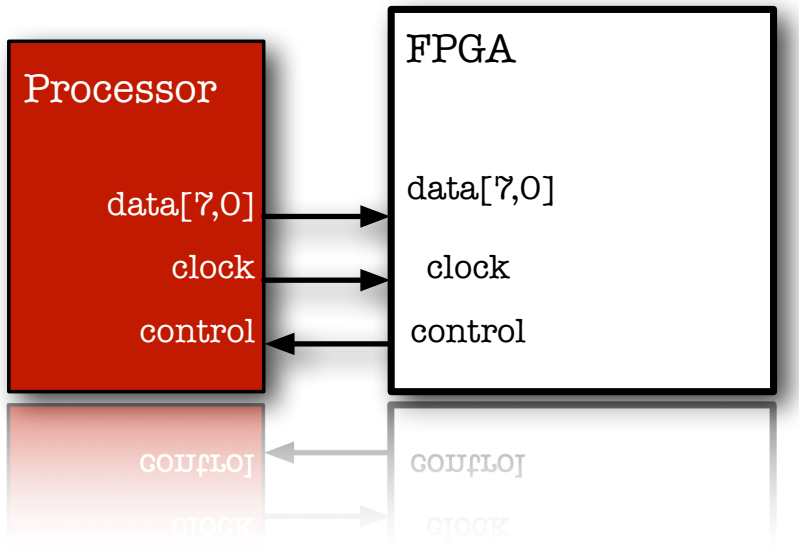
- Le mode «Périphérique»
  - Le FPGA est vu comme un périphérique du système embarqué dans lequel il est intégré,
- Au cours du temps le processeur configure le FPGA,
  - Le bitstream est transmis le processeur vers le FPGA,



Chargement série des données

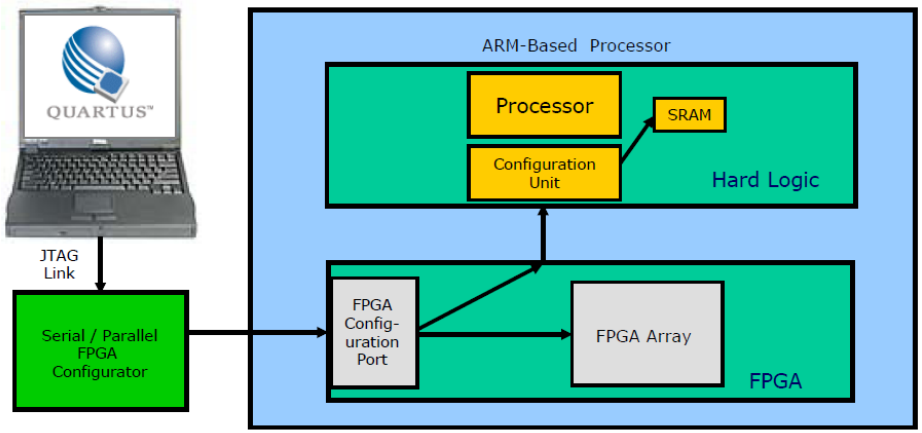


Chargement parallèle des données



La configuration des FPGAs : configuration et cœurs de proc.

Configuration centrée sur le FPGA (Exemple Altera Excalibur)

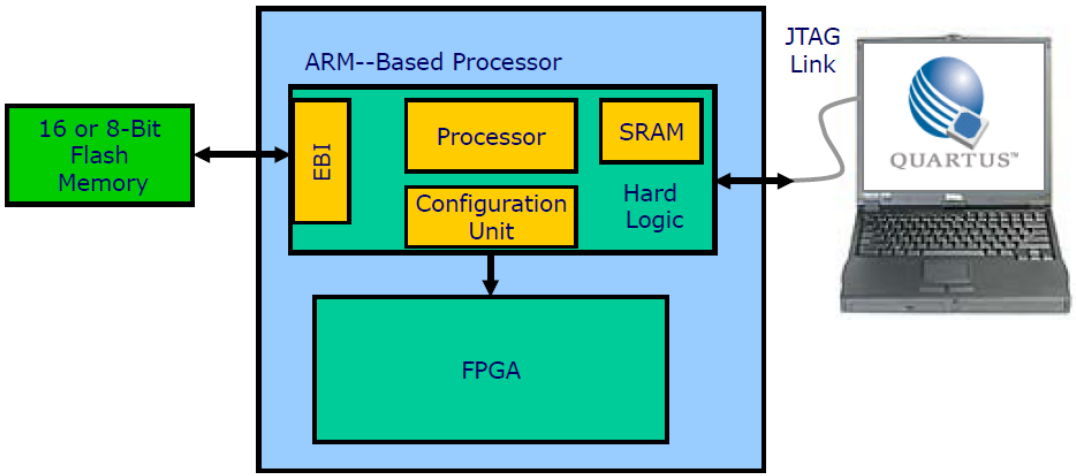


Le FPGA est esclave pour sa configuration

mais il est maître de la configuration du processeur

Deux approches complémentaires

Configuration centrée sur le Processeur

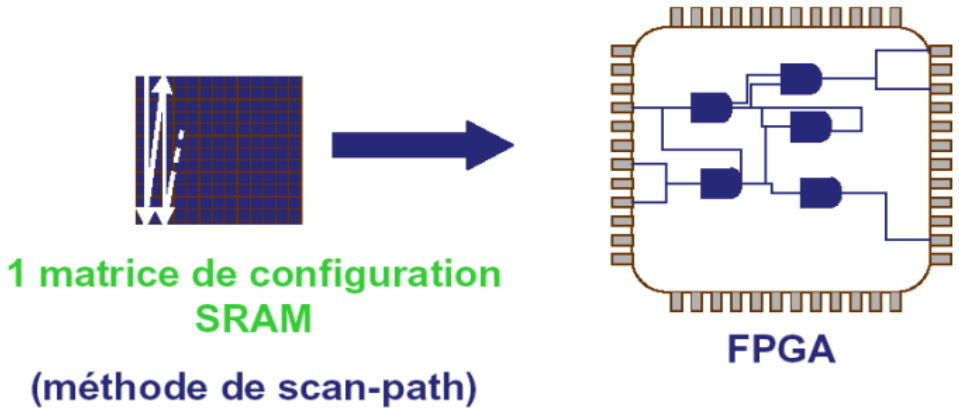


Le processeur est maître de la configuration du FPGA

La configuration des FPGAs : types de configuration

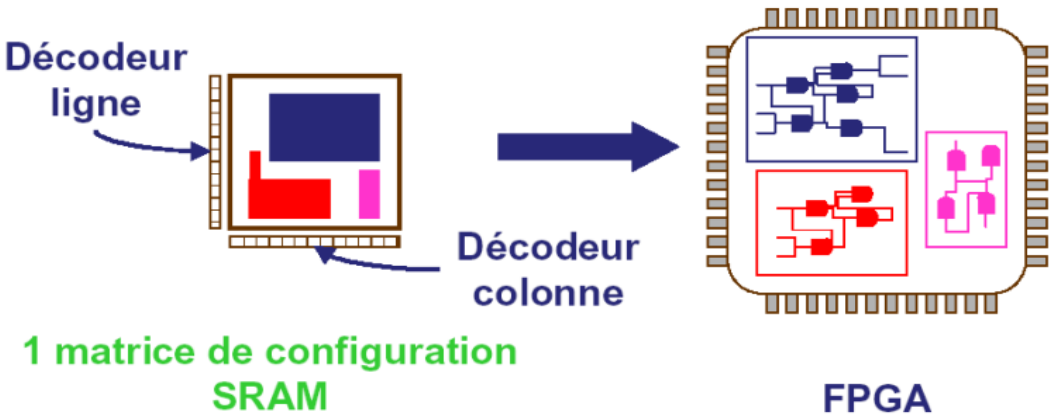
Configuration simple contexte

L'ensemble du FPGA est configuré à partir d'une matrice (configuration la plus courante)



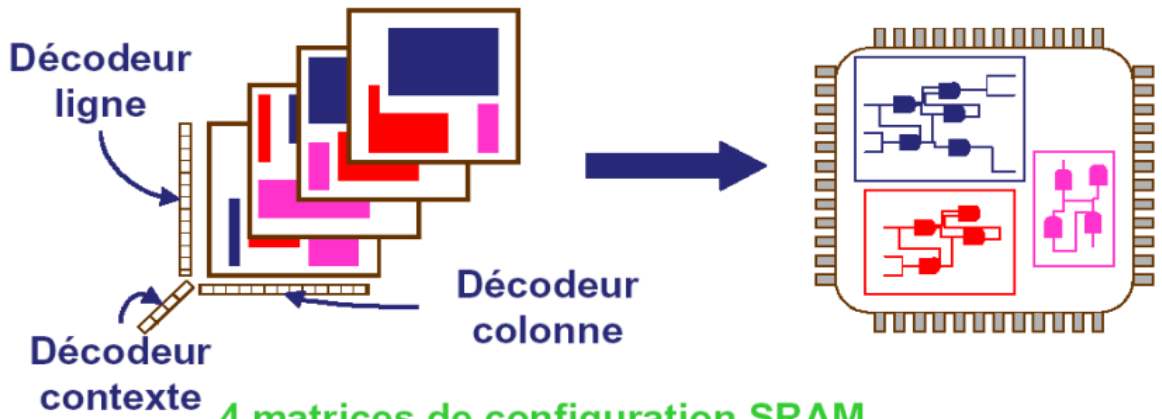
Configuration partielle simple contexte

Le FPGA est configuré partiellement (modification par partie)



Configuration partielle multi-contextes

Le FPGA est configuré partiellement, passage d'un contexte à l'autre



Auto-reconfiguration partielle

Développement d'architectures partiellement ou reconfigurables de manière autonome

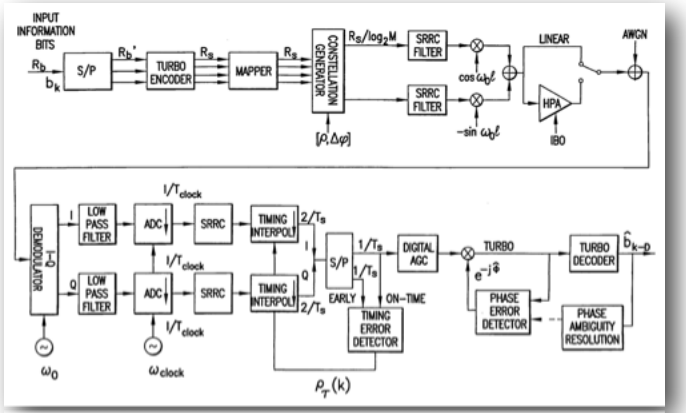
Reconfiguration dynamique

Développement d'architectures partiellement ou totalement reconfigurables dynamiquement (en cours d'exécution)

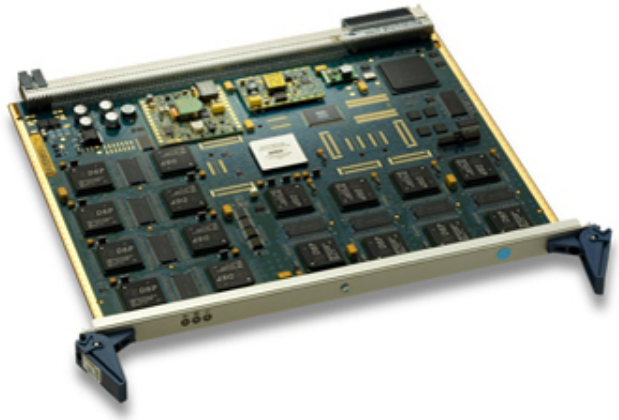
L'auto-reconfiguration partielle : Expression du besoin

Comm. numériques

Une chaine et diff. algorithmes



Multiples composants ou IPs



Reconf. dynamique partielle

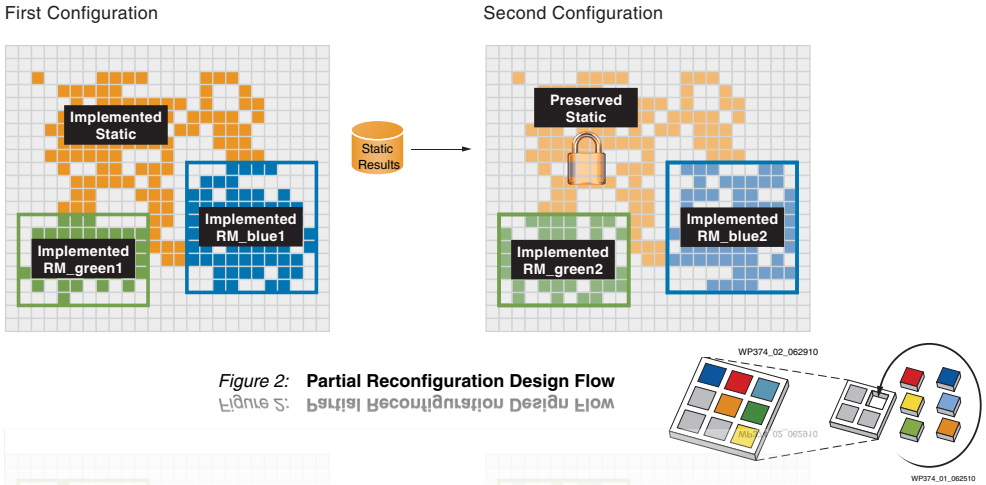
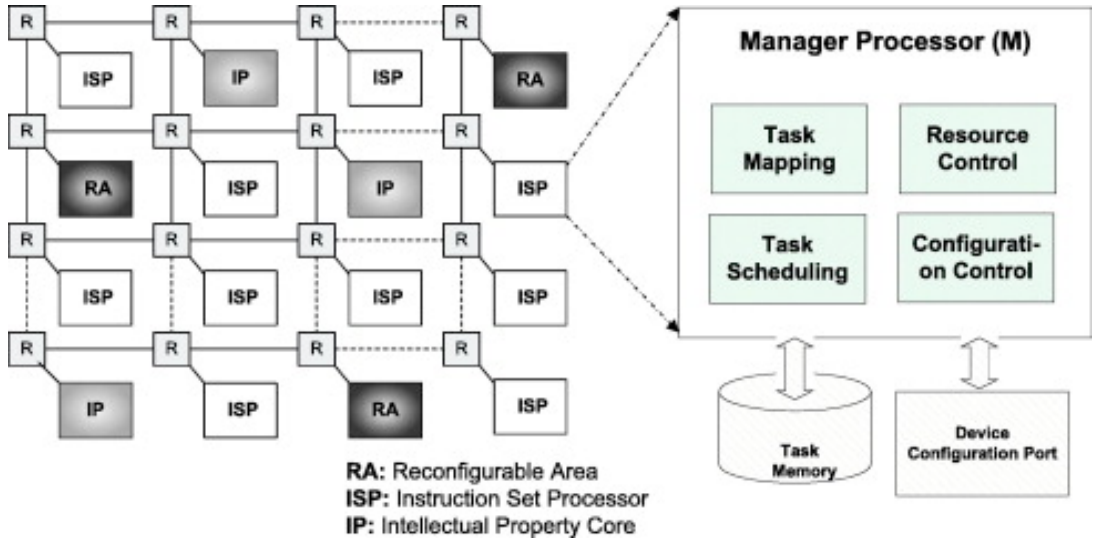
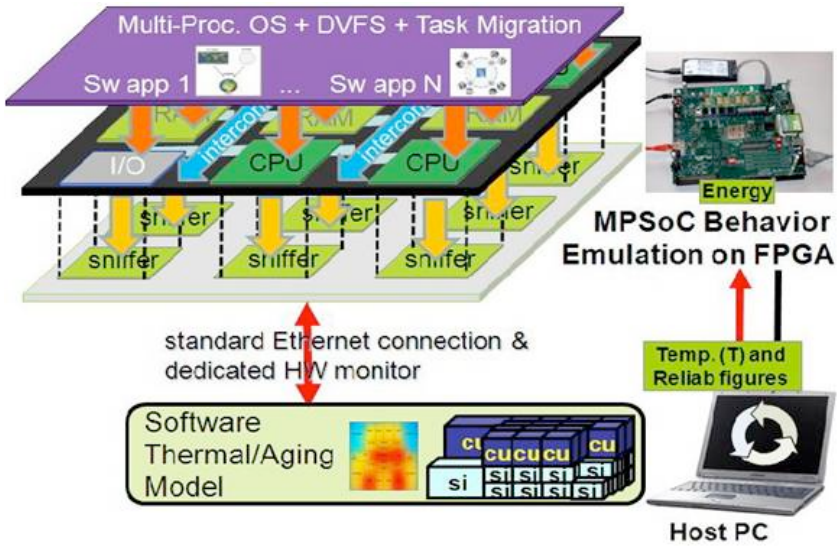


Figure 2: **Partial Reconfiguration Design Flow**

Au delà du SoPC, il y a le MPSoC dans le SoPC...

Dans un FPGA, on peut mettre ce qu'on désire... Plusieurs coeurs de processeur ?!



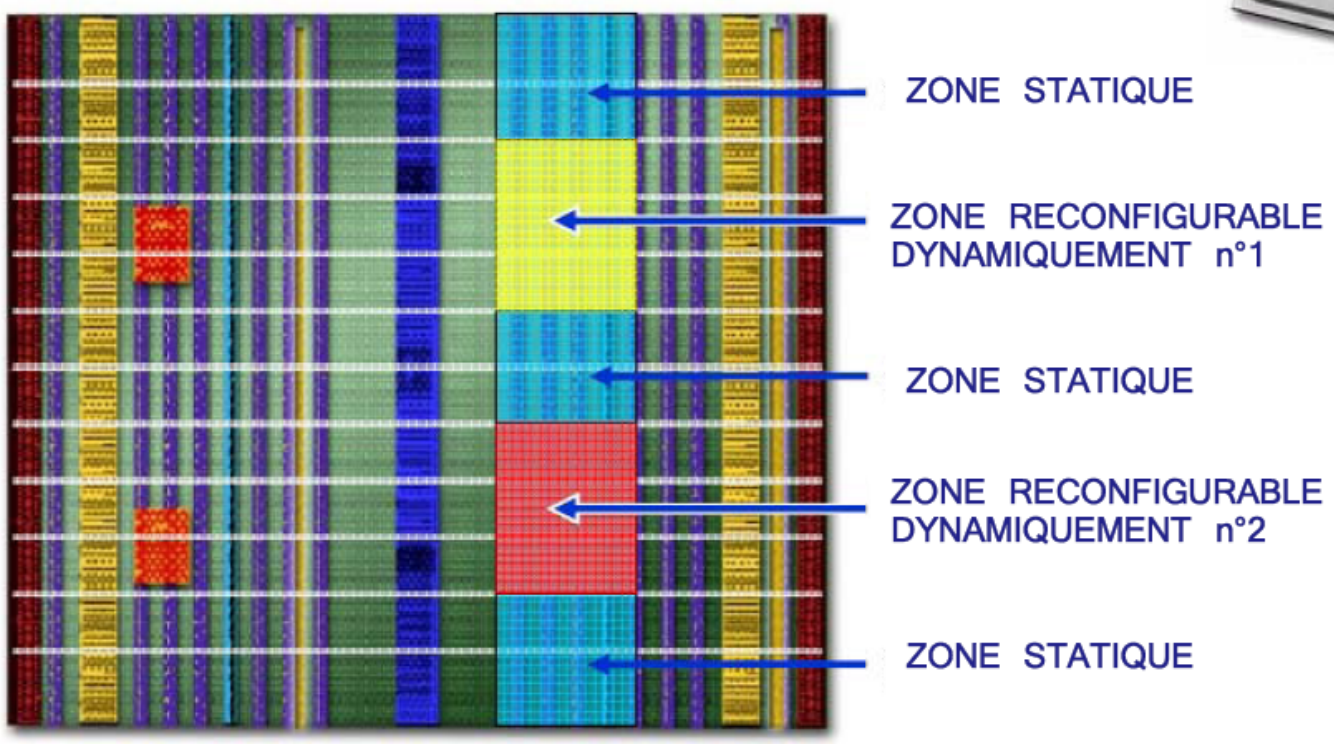


Comment peut on gérer cela efficacement ?! Et avec la reconfiguration dynamique partielle cela donne quoi ???

L'auto-reconfiguration partielle chez Xilinx

Solution proposée en 2002 par Xilinx pour les Virtex-II Pro

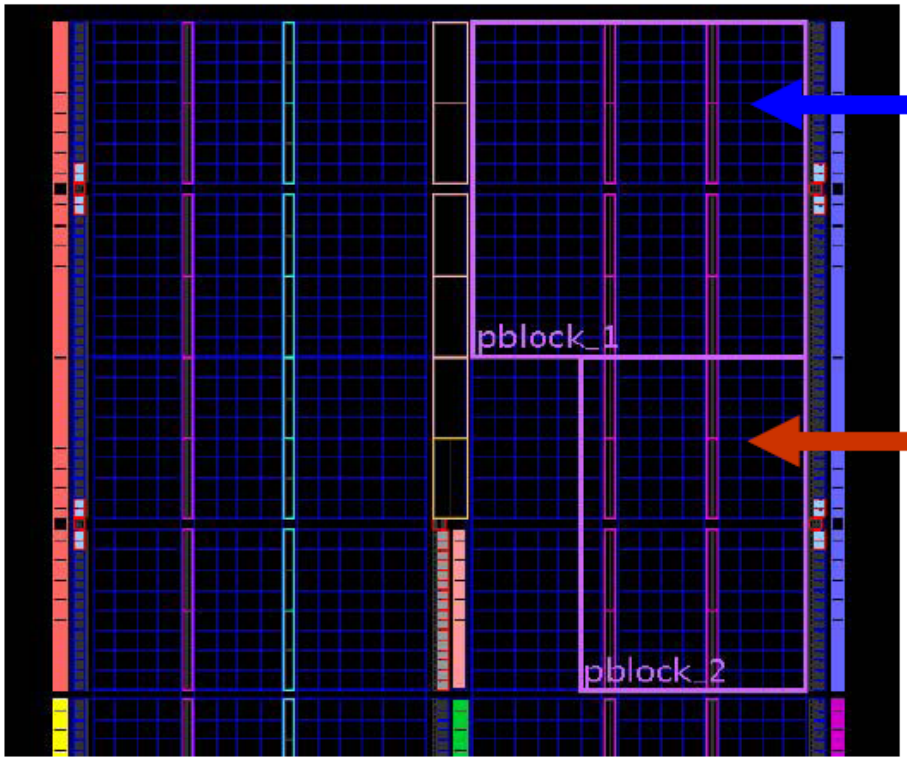
- ➔ Disponible pour : Virtex-II, Virtex-II Pro, Virtex-4, Virtex-5, Virtex-6, Spartan-6
- ➔ Partitionnement du circuit
  - Zones statiques
  - Zones reconfigurables dynamiquement



L'auto-reconfiguration partielle chez Xilinx

Méthodologie de conception

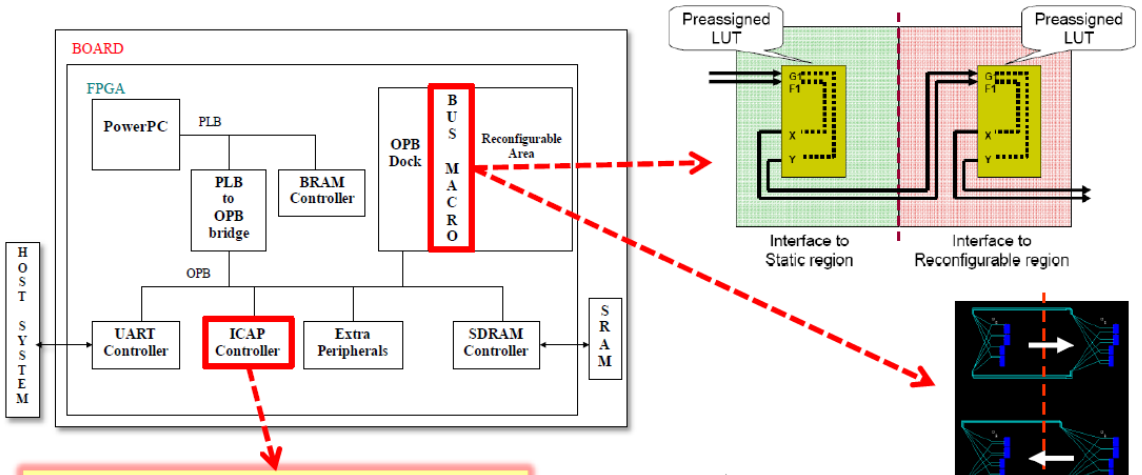
- ➔ Définition des configurations
- ➔ Choix des zones statiques et dynamiques
- ➔ Définition des contraintes de placement et routage



ZONE RECONFIGURABLE DYNAMQIUEMENT N°1

ZONE RECONFIGURABLE DYNAMQIUEMENT N°1

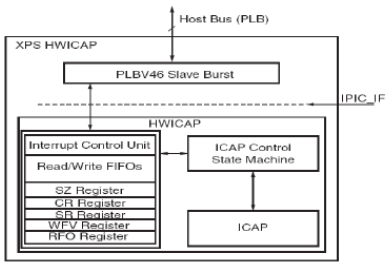
L'auto-reconfiguration partielle chez Xilinx

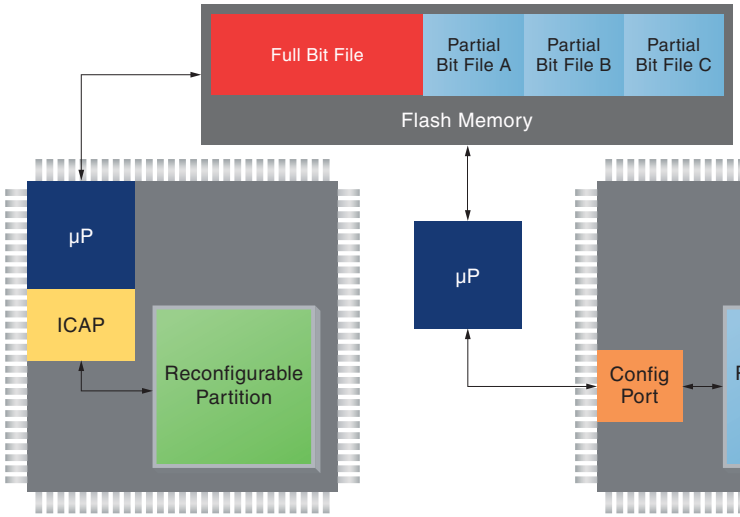


ICAP (*Internal Configuration Access Port*)
32 bits 100MHz(Virtex-4)
32 bits 120MHz (Virtex-5)

Contrôleur :

HWICAP (Virtex-II)
XPS_HWICAP (Virtex-4 et Virtex-5)





Self-Reconfigurable FPGA

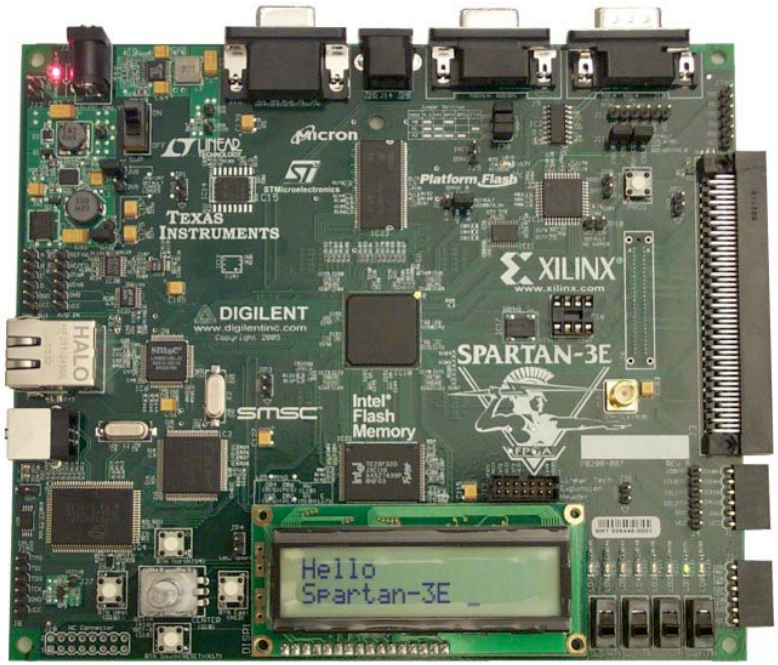
Externally-Reconfigurable FPGA

Les fabricants de circuit FPGA

Historique des PFGA de la société Xilinx

- 1985 : 1ere famille de FPGA Xilinx, le XC2000 (1 500 portes)
- 1998 : Famille Spartan ciblant les petits circuits en facilitant l'utilisation
- 1999 : Famille Virtex dédiée aux applications complexes (4 millions de portes) Technologie 0.22 µm, 5 niveaux d'interconnexion
- 2000 : Famille Spartan II, FPGA de moyenne capacité (100 000 portes) bas coût
- 2001 : Famille Virtex II dédiée aux applications complexes (8 millions de portes) Technologie 0.15 µm, 8 niveaux d'interconnexion
- 2002 : Famille Virtex II Pro (+10 millions de portes), intégration de 0 à 4 cœurs de processeur Power PC405. Technologie 0.13 µm, 9 niveaux d'interconnexion
- 2004 : Famille Spartan-3, FPGA de capacité 50K à 5M portes bas coût. Technologie 0.09 µm
- 2004 : Famille Virtex-4, 3 sous-familles (LX, SX et FX) intégration de 0 à 2 cœurs de processeur. Technologie 0.09 µm, 11 niveaux d'interconnexion
- 2006 : Famille Virtex-5, 4 sous-familles (LX, SX, SXT et FX). Élément logique contenant des LUTs à 6 entrées. Technologie 0.065 µm, 11 niveaux d'interconnexion.
- 2009 : Famille Virtex-6, 3 sous-familles (LX, LXT et SXT). LUTs à 6 entrées. Technologie 0.040 µm, 11 niveaux d'interconnexion. Famille Virtex-6 Technologie 0.045 µm, 9 niveaux d'interco.
- 2010 : Famille Artix-7, Kintex-7, Virtex-7, 3 sous-familles. Élément logique contenant des LUTs à 6 entrées. Technologie 0.028 µm, 11 niveaux d'interconnexion

La famille Spartan-3E de chez Xilinx



Vous connaissez cette carte, son FPGA est un Spartan-3E 500E.

Combien de % du FPGA avez vous utilisé dans vos TPs ?

Device	System Gates	Equivalent Logic Cells	CLB Array (One CLB = Four Slices)				Distributed RAM bits(1)	Block RAM bits(1)	Dedicated Multipliers	DCMs	Maximum User I/O	Maximum Differential I/O Pairs
			Rows	Columns	Total CLBs	Total Slices						
XA3S100E	100K	2,160	22	16	240	960	15K	72K	4	2	108	40
XA3S250E	250K	5,508	34	26	612	2,448	38K	216K	12	4	172	68
XA3S500E	500K	10,476	46	34	1,164	4,656	73K	360K	20	4	190	77
XA3S1200E	1200K	19,512	60	46	2,168	8,672	136K	504K	28	8	304	124
XA3S1600E	1600K	33,192	76	58	3,688	14,752	231K	648K	36	8	376	156

Caractéristiques de la famille Spartan-6

Spartan-6 LX FPGAs
Optimized for Lowest-Cost Logic, DSP, and Memory
(1.2V, 1.0V)

	Part Number	XC6SLX4	XC6SLX9	XC6SLX16	XC6SLX25	XC6SLX45	XC6SLX75	XC6SLX100	XC6SLX150
Logic Resources	Slices[1]	600	1,430	2,278	3,758	6,822	11,662	15,822	23,038
	Logic Cells[2]	3,840	9,152	14,579	24,051	43,661	74,637	101,261	147,443
	CLB Flip-Flops	4,800	11,440	18,224	30,064	54,576	93,296	126,576	184,304
Memory Resources	Maximum Distributed RAM (Kb)	75	90	136	229	401	692	976	1,355
	Block RAM (18 Kb each)	12	32	32	52	116	172	268	268
	Total Block RAM (Kb) [3]	216	576	576	936	2,088	3,096	4,824	4,824
Clock Resources	Clock Management Tiles (CMT) [4]	2	2	2	2	4	6	6	6
I/O Resources	Maximum Single-Ended Pins	132	200	232	266	358	408	480	576
	Maximum Differential Pairs	66	100	116	133	179	204	240	288
Embedded Hard IP Resources	DSP48A1 Slices [5]	8	16	32	38	58	132	180	180
	Endpoint Block for PCI Express®	—	—	—	—	—	—	—	—
	Memory Controller Blocks	0	2	2	2	2	4	4	4
	GTP Low-Power Transceivers	—	—	—	—	—	—	—	—
Speed Grades	Commercial [10]	-1L, -2, -3	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N
	Industrial[10]	-1L, -2, -3	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N	-1L, -2, -3, -3N
Configuration	Configuration Memory (Mb)	2.7	2.7	3.7	6.4	11.9	19.6	26.5	33.8

Caractéristiques de la famille Virtex-6 de chez Xilinx

Device	Logic Cells	Configurable Logic Blocks (CLBs)		DSP48E1 Slices[2]	Block RAM Blocks			MMCMs[4]	Interface Blocks for PCI Express	Ethernet MACs[5]	Maximum Transceivers		Total I/O Banks[6]	Max User I/O[7]
		Slices[1]	Max Distributed RAM (Kb)		18 Kb[3]	36 Kb	Max (Kb)				GTX	GTH		
XC6VLX75T	74,496	11,640	1,045	288	312	156	5,616	6	1	4	12	0	9	360
XC6VLX130T	128,000	20,000	1,740	480	528	264	9,504	10	2	4	20	0	15	600
XC6VLX195T	199,680	31,200	3,040	640	688	344	12,384	10	2	4	20	0	15	600
XC6VLX240T	241,152	37,680	3,650	768	832	416	14,976	12	2	4	24	0	18	720
XC6VLX365T	364,032	56,880	4,130	576	832	416	14,976	12	2	4	24	0	18	720
XC6VLX550T	549,888	85,920	6,200	864	1,264	632	22,752	18	2	4	36	0	30	1200
XC6VLX760	758,784	118,560	8,280	864	1,440	720	25,920	18	0	0	0	0	30	1200
XC6VSX315T	314,880	49,200	5,090	1,344	1,408	704	25,344	12	2	4	24	0	18	720
XC6VSX475T	476,160	74,400	7,640	2,016	2,128	1,064	38,304	18	2	4	36	0	21	840
XC6VHX250T	251,904	39,360	3,040	576	1,008	504	18,144	12	4	4	48	0	8	320
XC6VHX255T	253,440	39,600	3,050	576	1,032	516	18,576	12	2	2	24	24	12	480
XC6VHX380T	382,464	59,760	4,570	864	1,536	768	27,648	18	4	4	48	24	18	720
XC6VHX565T	566,784	88,560	6,370	864	1,824	912	32,832	18	4	4	48	24	18	720

Caractéristiques de la famille 7 de chez Xilinx

Table 1: **7 Series Families Comparison**

Maximum Capability	Artix-7 Family	Kintex-7 Family	Virtex-7 Family
Logic Cells	215K	478K	1,955K
Block RAM(1)	13 Mb	34 Mb	68 Mb
DSP Slices	740	1,920	3,600
Peak DSP Performance(2)	929 GMAC/s	2,845 GMAC/s	5,335 GMAC/s
Transceivers	16	32	96
Peak Transceiver Speed	6.6 Gb/s	12.5 Gb/s	28.05 Gb/s
Peak Serial Bandwidth (Full Duplex)	211 Gb/s	800 Gb/s	2,784 Gb/s
PCIe Interface	x4 Gen2	x8 Gen2	x8 Gen3
Memory Interface	1,066 Mb/s	1,866 Mb/s	1,866 Mb/s
I/O Pins	500	500	1,200
I/O Voltage	1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V	1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V	1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V
Package Options	Low-Cost, Wire-Bond, Lidless Flip-Chip	Low-Cost, Lidless Flip-Chip and High-Performance Flip-Chip	Highest Performance Flip-Chip

Table 6: **Virtex-7 FPGA Feature Summary**

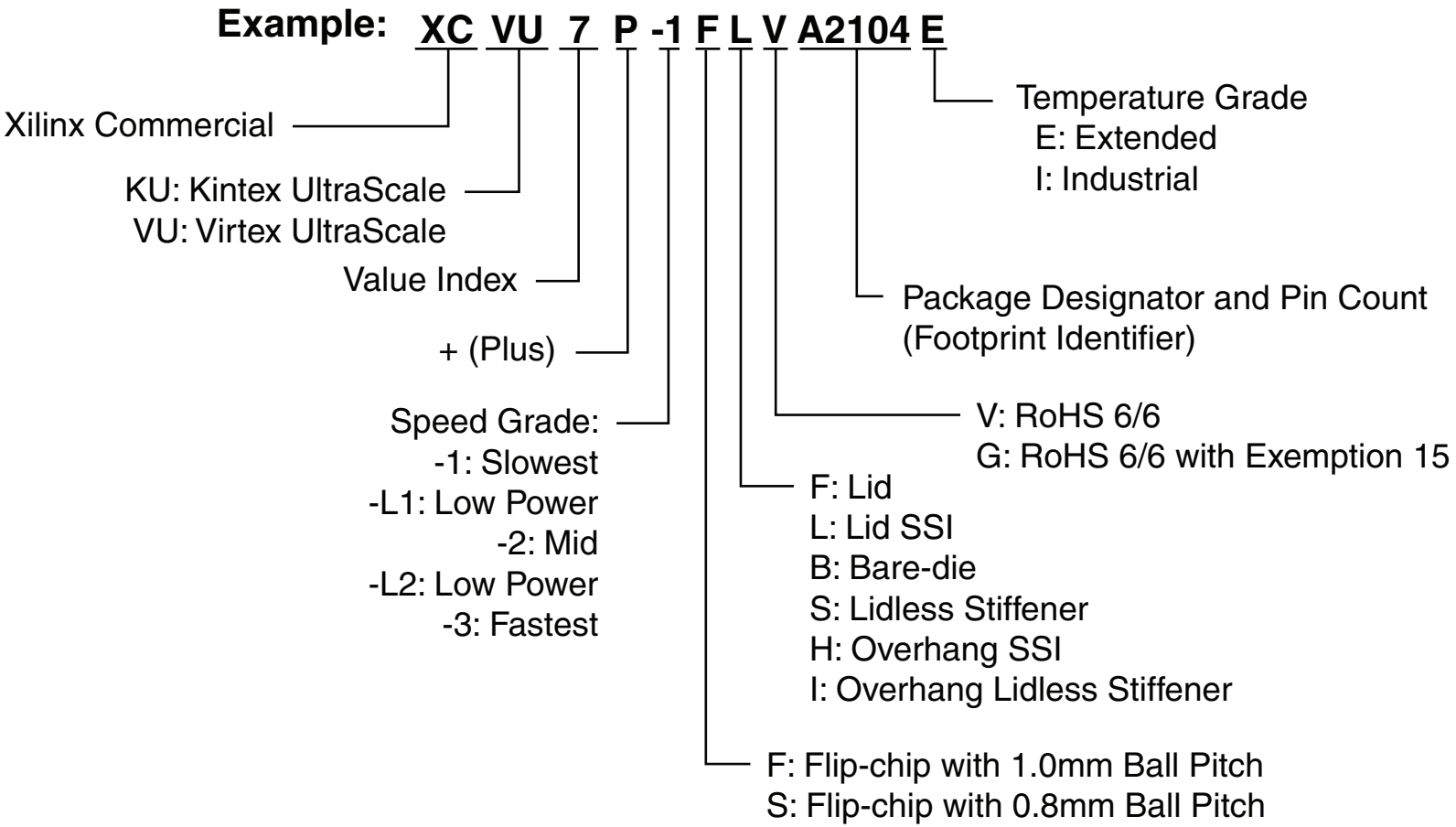
Device(1)	Logic Cells	Configurable Logic Blocks (CLBs)		DSP Slices(3)	Block RAM Blocks(4)			CMTs(5)	PCIe(6)	GTX	GTH	GTZ	XADC Blocks	Total I/O Banks(7)	Max User I/O(8)	SLRs(9)
		Slices(2)	Max Distributed RAM (Kb)		18 Kb	36 Kb	Max (Kb)									
XC7V585T	582,720	91,050	6,938	1,260	1,590	795	28,620	18	3	36	0	0	1	17	850	N/A
XC7V2000T	1,954,560	305,400	21,550	2,160	2,584	1,292	46,512	24	4	36	0	0	1	24	1,200	4
XC7VX330T	326,400	51,000	4,388	1,120	1,500	750	27,000	14	2	0	28	0	1	14	700	N/A
XC7VX415T	412,160	64,400	6,525	2,160	1,760	880	31,680	12	2	0	48	0	1	12	600	N/A
XC7VX485T	485,760	75,900	8,175	2,800	2,060	1,030	37,080	14	4	56	0	0	1	14	700	N/A
XC7VX550T	554,240	86,600	8,725	2,880	2,360	1,180	42,480	20	2	0	80	0	1	16	600	N/A
XC7VX690T	693,120	108,300	10,888	3,600	2,940	1,470	52,920	20	3	0	80	0	1	20	1,000	N/A
XC7VX980T	979,200	153,000	13,838	3,600	3,000	1,500	54,000	18	3	0	72	0	1	18	900	N/A
XC7VX1140T	1,139,200	178,000	17,700	3,360	3,760	1,880	67,680	24	4	0	96	0	1	22	1,100	4
XC7VH580T	580,480	90,700	8,850	1,680	1,880	940	33,840	12	2	0	48	8	1	12	600	2
XC7VH870T	876,160	136,900	13,275	2,520	2,820	1,410	50,760	18	3	0	72	16	1	13	650	3

Zinq UltraScale+ FPGA family

	ZU4EV	ZU5EV	ZU7EV
Application Processing Unit	Quad-core ARM Cortex-A53 MPCore with CoreSight; NEON & Single/Double Precision Floating Point; 32KB/32KB L1 Cache, 1MB L2 Cache		
Real-Time Processing Unit	Dual-core ARM Cortex-R5 with CoreSight; Single/Double Precision Floating Point; 32KB/32KB L1 Cache, and TCM		
Embedded and External Memory	256KB On-Chip Memory w/ECC; External DDR4; DDR3; DDR3L; LPDDR4; LPDDR3; External Quad-SPI; NAND; eMMC		
General Connectivity	214 PS I/O; UART; CAN; USB 2.0; I2C; SPI; 32b GPIO; Real Time Clock; WatchDog Timers; Triple Timer Counters		
High-Speed Connectivity	4 PS-GTR; PCIe Gen1/2; Serial ATA 3.1; DisplayPort 1.2a; USB 3.0; SGMII		
Graphic Processing Unit	ARM Mali-400 MP2; 64KB L2 Cache		
Video Codec	1	1	1
System Logic Cells	192,150	256,200	504,000
CLB Flip-Flops	175,680	234,240	460,800
CLB LUTs	87,840	117,120	230,400
Distributed RAM (Mb)	2.6	3.5	6.2
Block RAM Blocks	128	144	312
Block RAM (Mb)	4.5	5.1	11.0
UltraRAM Blocks	48	64	96
UltraRAM (Mb)	13.5	18.0	27.0
DSP Slices	728	1,248	1,728
CMTs	4	4	8
Max. HP I/O(1)	156	156	416
Max. HD I/O(2)	96	96	48
System Monitor	2	2	2
GTH Transceiver 16.3Gb/s(3)	16	16	24
GTY Transceivers 32.75Gb/s	0	0	0
Transceiver Fractional PLLs	8	8	12
PCIe Gen3 x16 and Gen4 x8	2	2	2
150G Interlaken	0	0	0
100G Ethernet w/ RS-FEC	0	0	0

Every Configurable Logic Block (CLB) in the UltraScale architecture contains 8 LUTs and 16 flip-flops. The LUTs can be configured as either one 6-input LUT with one output, or as two 5-input LUTs with separate outputs but common inputs. Each LUT can optionally be registered in a flip-flop. In addition to the LUTs and flip-flops, the CLB contains arithmetic carry logic and multiplexers to create wider logic functions.

Xilinx FPGA ordering information



1) -L1 and -L2 are the ordering codes for the low power -1L and -2L speed grades, respectively.

DS890_04_092917

Figure 4: **UltraScale+ FPGA Ordering Information**

Quelques caractéristiques de FPGA de chez Altera

Table 1. Stratix Series Introduction

Device Family	Stratix	Stratix GX	Stratix II	Stratix II GX	Stratix III	Stratix IV	Stratix V	Stratix 10
Year of introduction	2002	2003	2004	2005	2006	2008	2010	2013
Process technology	130 nm	130 nm	90 nm	90 nm	65 nm	40 nm	28 nm	14 nm Tri-Gate

Table 2. Stratix Series Common Features

Technology	Stratix IV	Stratix V	Stratix 10
Adaptive Logic Modules	✓	✓	✓
Transceivers	✓	✓	✓
Power	✓	✓	✓
DSP blocks	✓	✓	✓
External memory interfaces	✓	✓	✓
Embedded memory	✓	✓	✓
I/O performance	✓	✓	✓
Design security	✓	✓	✓
Single-event upset mitigation	✓	✓	✓
Remote system upgrades	✓	✓	✓
Partial reconfiguration		✓	✓
3D Tri-Gate Transistor Technology			✓
Hard Processor System			✓
Next-Generation architecture			✓
Heterogeneous 3D Solutions (SRAM, DRAM, and ASICs)			✓
Hard IEEE 754 single precision floating point			✓

Complexité des FPGA de chez Altera

	Maximum Resource Count for Arria 10 GX FPGAs[1]								
	10AX016	10AX022	10AX027	10AX032	10AX048	10AX057	10AX066	10AX090	10AX115
Resources									
ALMs	61,510	81,510	101,620	119,660	182,720	217,080	251,450	339,620	427,700
LEs (K)	160	220	270	320	480	570	660	900	1,150
Registers	246,040	326,040	406,480	478,640	730,880	868,320	1,005,800	1,358,480	1,710,800
M20K memory blocks	440	583	750	891	1,438	1,850	1,964	2,339	2,713
M20K memory (Mb)	9	11	15	17	28	36	39	46	54
MLAB memory (Mb)	1	1.4	2.2	2.9	4.4	5.0	5.7	9.2	12.7
Variable-precision digital signal processing (DSP) blocks	156	192	800	985	1,368	1,612	1,855	1,518	1,518
18 x 19 multipliers	312	384	1,600	1,970	2,736	3,223	3,356	3,036	3,036

	Maximum Resource Count for Cyclone® V E FPGAs (1.1 V)[1]				
	5CEA2	5CEA4	5CEA5	5CEA7	5CEA9
Resources					
ALMs	9,434	18,480	29,080	56,480	113,560
LEs (K)	25	49	77	149.5	301
Registers	37,736	73,920	116,320	225,920	454,240
M10K memory blocks	176	308	446	686	1,220
M10K memory (Kb)	1,760	3,080	4,460	6,860	12,200
MLAB memory (Kb)	196	303	424	836	1,717
Variable-precision DSP blocks	25	66	150	156	342
18 x 18 multipliers	50	132	300	312	684

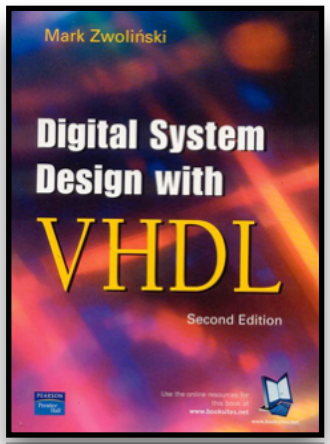
	Maximum Resource Count for Stratix V GX FPGAs (0.85 V)[1]									
	5SGXA3	5SGXA4	5SGXA5	5SGXA7	5SGXA9	5SGXAB	5SGXB5	5SGXB6	5SGXB9	5SGXBB
Resources										
ALMs	128,300	158,500	185,000	234,720	317,000	359,200	185,000	225,400	317,000	359,200
LEs (K)	340	420	490	622	840	952	490	597	840	952
Registers	513,200	634,000	740,000	938,880	1,268,000	1,436,800	740,000	901,600	1,268,000	1,436,800
M20K memory blocks	957	1,900	2,304	2,560	2,640	2,640	2,100	2,660	2,640	2,640
M20K memory (Mb)	19	37	45	50	52	52	41	52	52	52
MLAB memory (Mb)	3.92	4.84	5.65	7.16	9.67	10.96	5.65	6.88	9.67	10.96
Variable-precision DSP blocks	256	256	256	256	352	352	399	399	352	352
18 x 18 multipliers	512	512	512	512	704	704	798	798	704	704

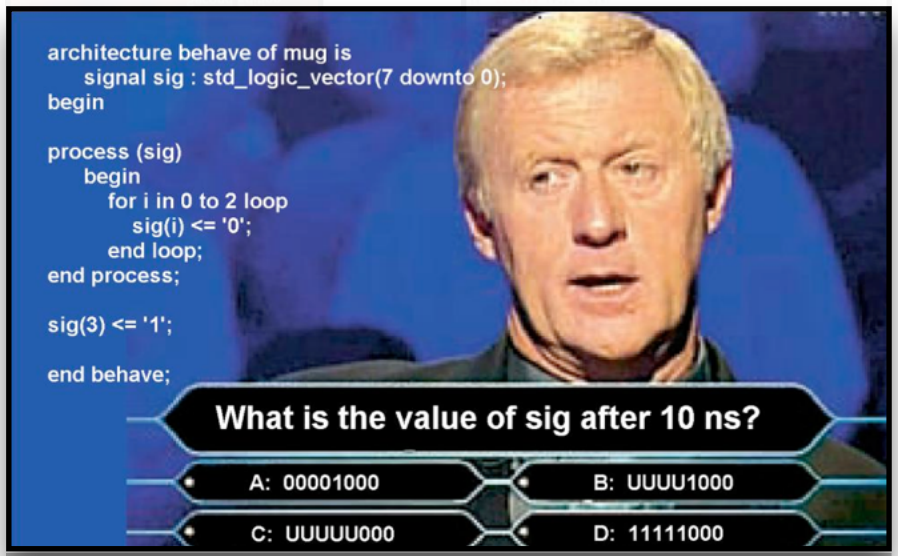
	Maximum Resource Count for Stratix IV GT FPGAs (0.95 V)[1]					
	EP4S40G2	EP4S40G5	EP4S100G2	EP4S100G3	EP4S100G4	EP4S100G5
Resources						
ALMs	91,200	212,480	91,200	116,480	141,440	212,480
LEs (K)	228	531	228	291	354	531
Registers[2]	182,400	424,960	182,400	232,960	282,880	424,960
M9K memory blocks	1,235	1,280	1,235	936	1,248	1,280
M144K memory blocks	22	64	22	36	48	64
MLAB memory (Kb)	2,850	6,640	2,850	3,640	4,420	6,640
Embedded memory (Kb)	14,283	20,736	14,283	13,608	18,144	20,736
18 x 18 multipliers	1,288	1,024	1,288	832	1,024	1,024

Le flot de conception d'un circuit FPGA

Le flot de conception d'un FPGA

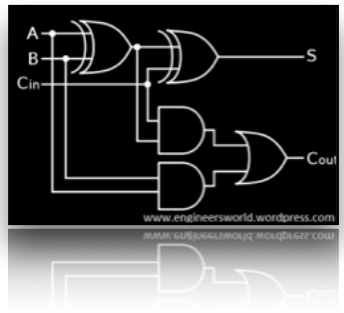
- Very high speed integrated circuit Hardware Description Language
  - Langage de description de structures électroniques numériques.
- Historique :
  - Définition au milieu des années 80 par le Département de la Défense américain (même période que le langage ADA pour le domaine logiciel).
  - Standardisation du langage VHDL en 1987 (norme IEEE 1076.1987),
  - Evolution du langage VHDL en 1993 (norme IEEE 1076.1993),
  - Extension à l'analogique VHDL-AMS (norme IEEE 1076.1999),
  - Extension type numériques (IEEE 1076.2008).

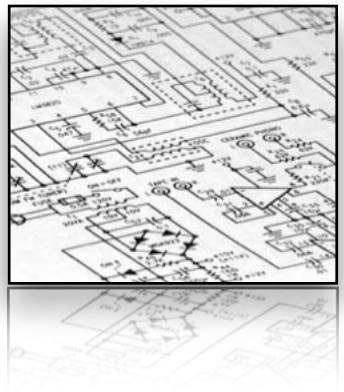


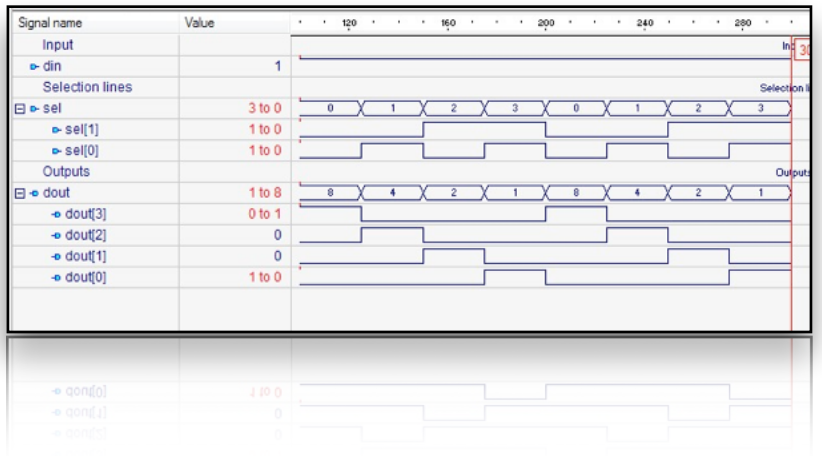


Le flot de conception d'un FPGA

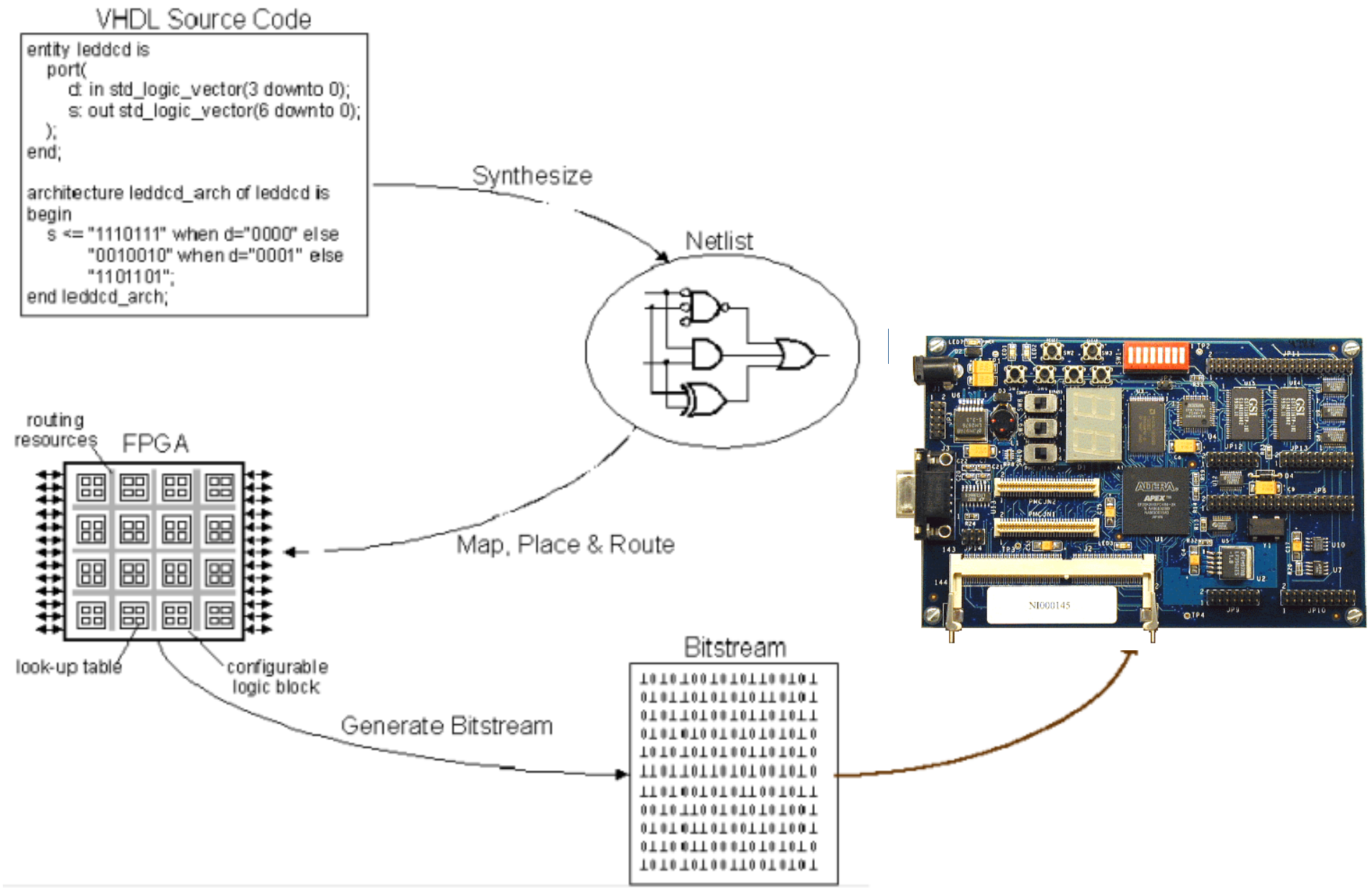
- Un modèle VHDL peut être :
  - Comportemental : décrit la fonctionnalité d'un objet par un algorithme séquentiel, ou table de vérité, sans référence à une structure d'implémentation,
    - Flux de données ou RTL: décrit le flux entre entrée et sortie au niveau bit, par des équations élémentaires.
  - Structurel : décrit la constitution de l'objet en un ensemble d'objets élémentaires interconnectés (proche du schéma).
- Deux jeux d'instructions en VHDL :
  - Instructions séquentielles: elles s'exécutent les unes après les autres,
  - Instructions concurrentes: elles s'exécutent «en même temps».



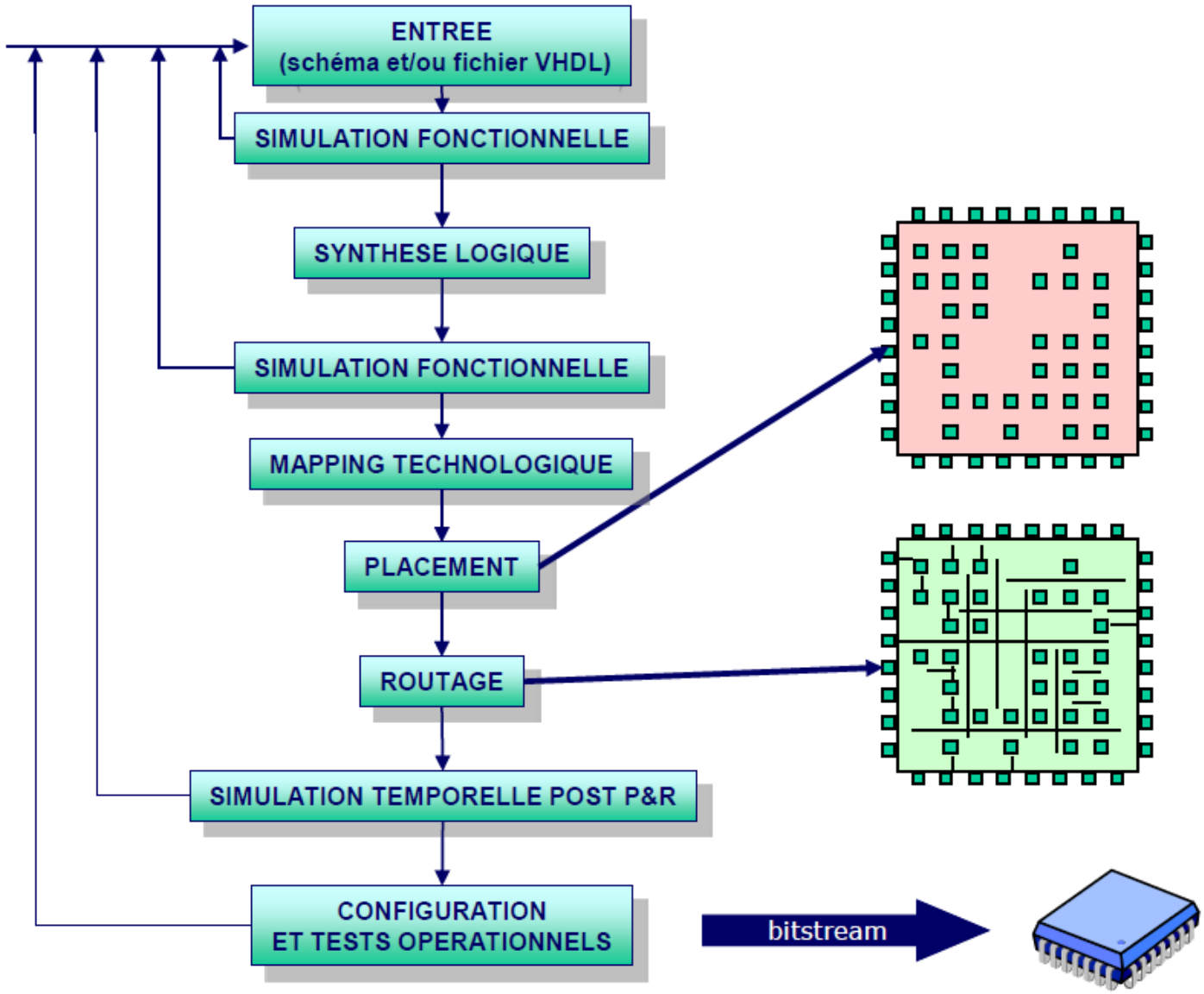




Rappel du flot de synthèse traditionnel



Une vue un peu plus complète du flot de chez Xilinx



Les outils de synthèse ciblant les FPGA

Les fabricants de FPGA proposent des outils de CAO pour la programmation de leurs circuits

ISE-Foundation

Quartus II

Ces outils permettent l'implémentation (placement-routage & configuration).

Pour des applications complexes, l'utilisation d'outils de synthèse logique et de simulation VHDL sont nécessaires

Outils commerciaux de synthèse logique FPGA

Outil de synthèse :

DC FPGA

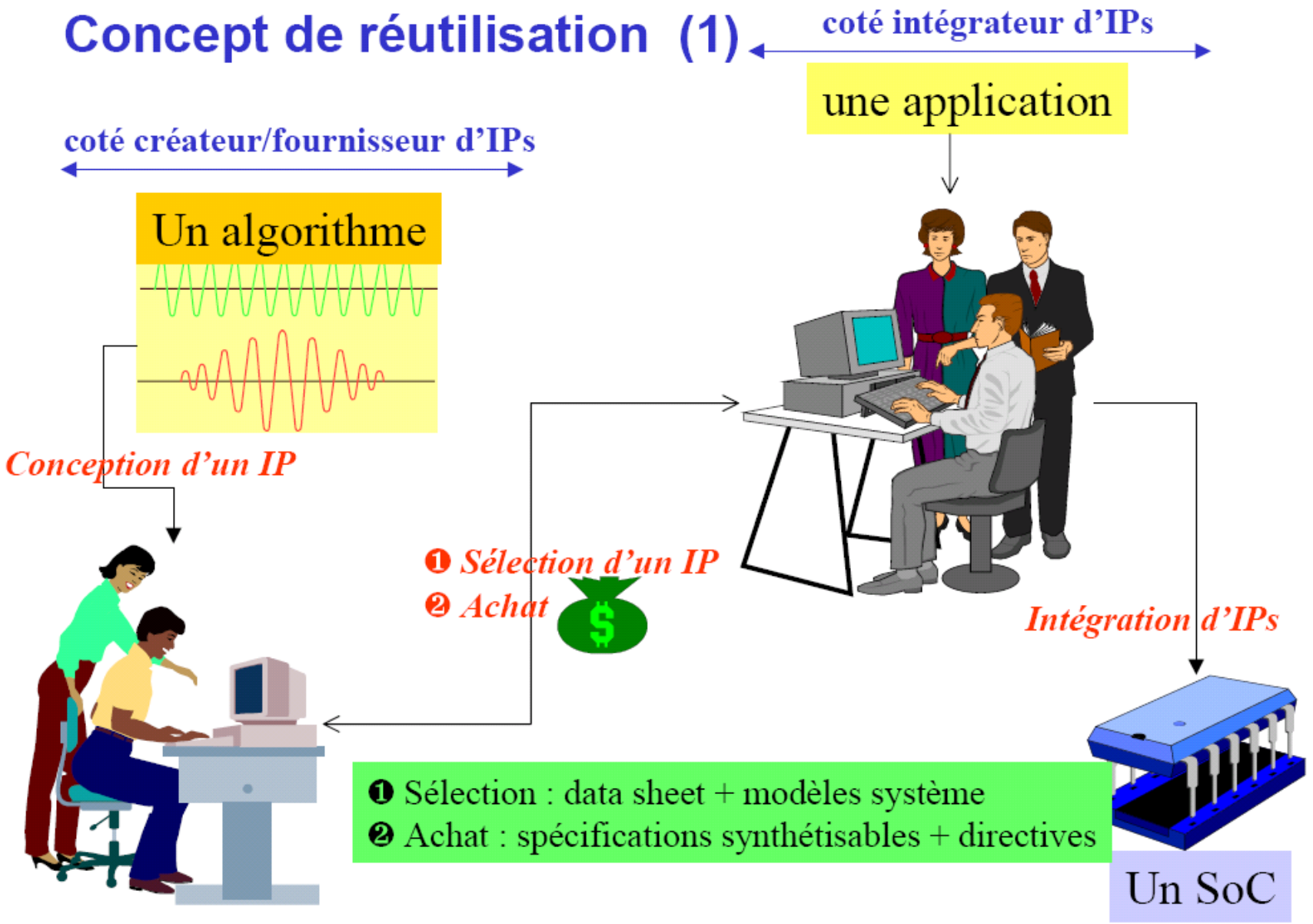
Outil de synthèse :

Precision™ RTL Synthesis

Outil de synthèse :

Synplify Pro

Design for Reuse, arrêter de réinventer la roue...



Quelques exemples de composants virtuels...

Communications	Bus Interface	Digital Signal Processing	Processor, Peripheral
ADPCM (u-law, a-law)	PCI Target	Color Space Converter	Nios™ Processor
ATM Controller	PCI Master-Target	Correlator	Tensilica X-tensa Processor
CRC	PCI-X	Digital Modulator	PalmChip Bus
Ethernet MAC (10/100/Gigabit)	CAN Bus	Discrete Cosine Transform	SDRAM Controller
HDLC Protocol Core	IIC Master & Slave	Fast Fourier Transform	DDR-SDRAM Controller
IMA Controller	IEEE 1394	FIR Compiler	QDR-SDRAM Controller
SONET/SDH Framer	PowerPC Bus Arbiter	IIR Filter	8237 DMA Controller
T3/E3 Framer	PowerPC Bus Master	Image Processing Library	8255 Peripheral Interface
Packet Over SONET Processor	PowerPC Bus Slave	NCO	8259 Interrupt Controller
Telephony Tone Generator	USB Function Controller	Reed Solomon Encoder/Decoder	8254 Timer/Counter
Utopia Master & Slave	USB Host Controller	Interleaver/Deinterleaver	8051, 6502, Z80
POS-PHY Interface		Viterbi Decoder	
		Turbo Decoder	

Les causes de l'augmentation de l'utilisation des IPs

Les outils sont le points faibles des FPGA

- Complexité du problème de placement routage
- Mise à jour et nouvelles version très fréquente
- Ils conditionnent bien souvent le choix du fabricant de FPGA
- Ils sont complexes à utilisés si l'on avec des contraintes particulièrement sérées
- A noter que le résultat de la synthèse et fortement lié à la spécification des applications (écriture du VHDL par exemple)

C'est pourquoi il faut s'appuyer sur des IP

- Estimations et simulations disponibles
- Temps de conception raccourcis
- Maintenance possible

Ils faut impérativement prendre en compte des méthodes de conception même si le composants est reconfigurable !

Les Systems sur Puce Programmables (SoPC)

Introduction au concept de SoC

L'approche SOC (technologie ASIC) répond aux besoins de performances et d'intégration mais :

- ✓ elle s'adapte difficilement aux besoins d'évolution des systèmes
- ✓ elle reste réservée aux grands volumes de production
- ✓ la fabrication et le test sont des étapes longues et coûteuses

Les systèmes sur puce programme répond en partie à ces problèmes :

- ✓ développement et prototypage rapide
- ✓ composants reconfigurables (quelques ms) à volonté

Mais :

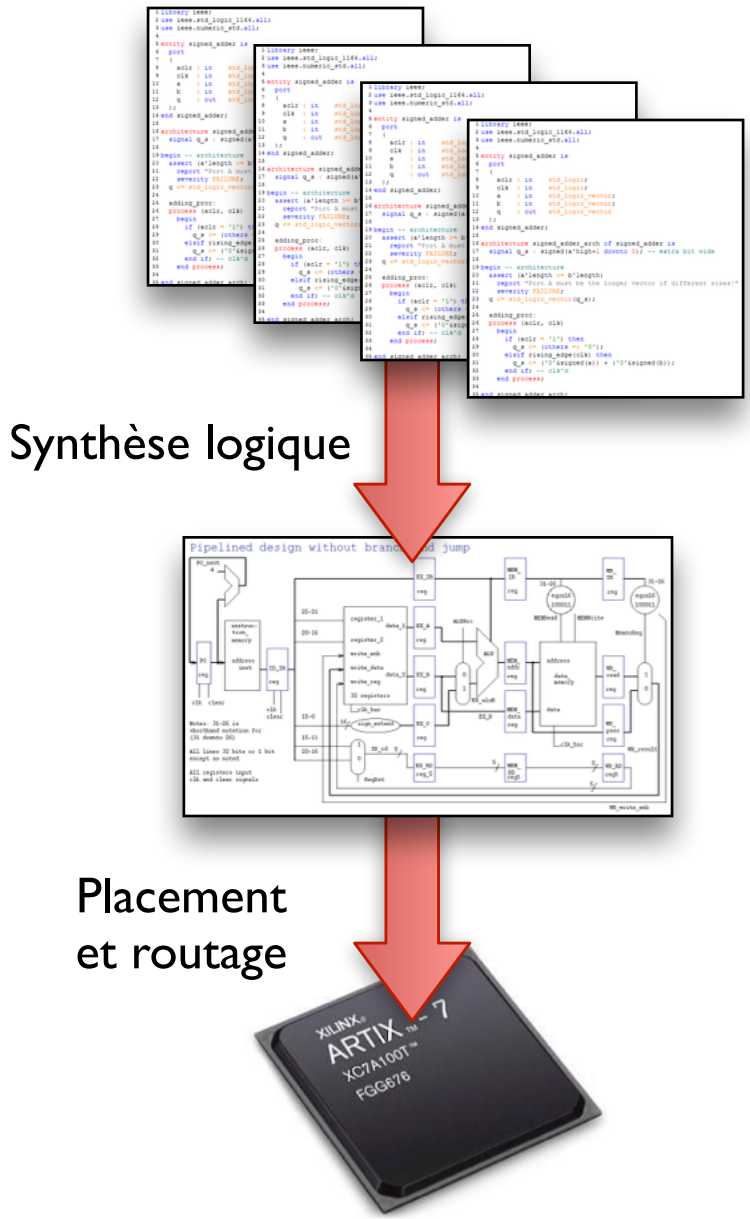
- ✓ la densité d'intégration est moindre (~10 Millions de portes)
- ✓ les performances atteintes sont moins importantes (consommation élevée)

Différentes notations :

- ✓ SOPC (Altera) : System On a Programmable Chip
- ✓ SORC (Xilinx) : System On a Reprogrammable Chip
- ✓ CSOC : Configurable System On Chip

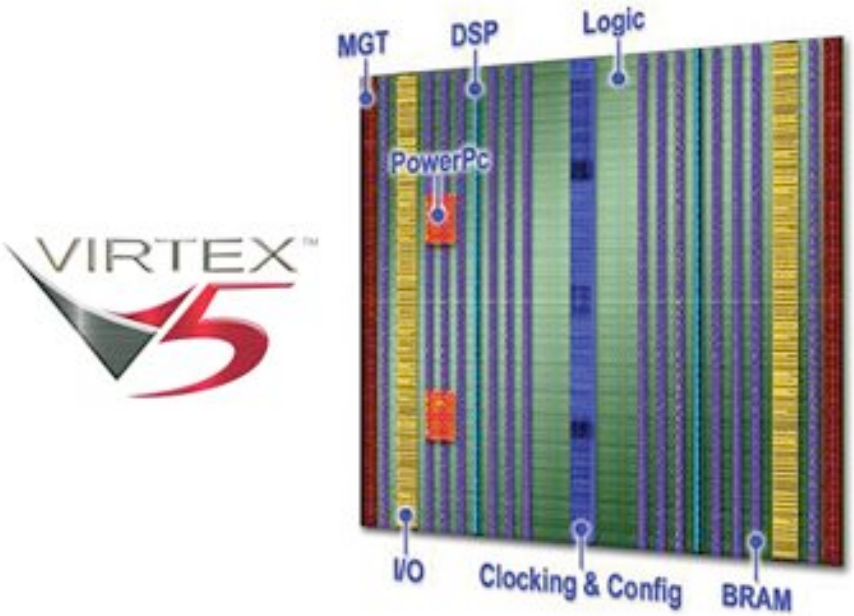
Les différentes familles de coeur de processeur (1/2)

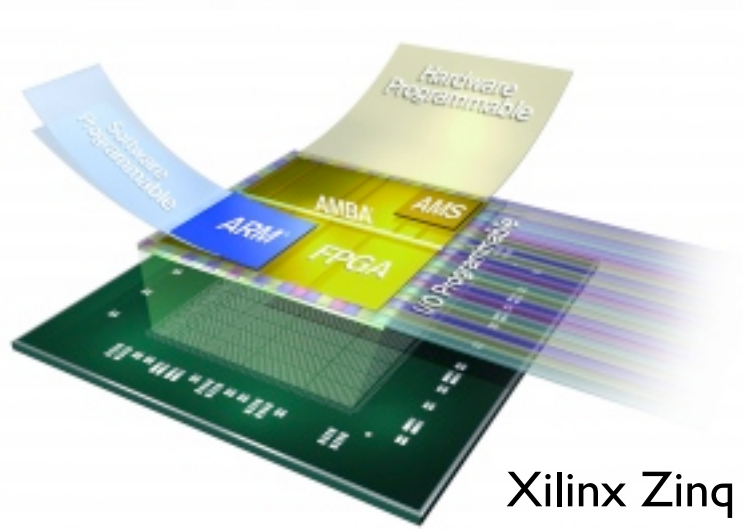
- Les Soft-cores sont des processeurs décrits en VHDL,
  - Modifiables à souhait (configuration ou customisation),
  - Généralement indépendants du FPGA (architecture portable et pérenne),
  - Multiples compromis (cost/perf.)
    - (Xilinx) Microblaze & Picoblaze,
    - (Altera) NIOS II,
    - (MIPS) Plasma,
    - (SPARC) Léon-3, etc.
- Performances en retrait,
  - L'intégration se fait en fonction des ressources du FPGA cible...



Les différentes familles de coeur de processeur (2/2)

- Les Hard-cores sont des processeurs gravés dans le silicium (au sein du FPGA),
  - Implantations performantes (>400MHz),
  - Taille optimale (full custom),
  - Architecture courante (OS, GCC).
- Par contre:
  - La customisation est quasi-inexistante,
  - Approche non pérenne (V5: PowerPC alors que V7:Arm),
  - Interconnexion par bus partagé des blocs développés en VHDL,
  - Les FPGA intégrant des processeurs hard-core sont plus chers à l'achat...

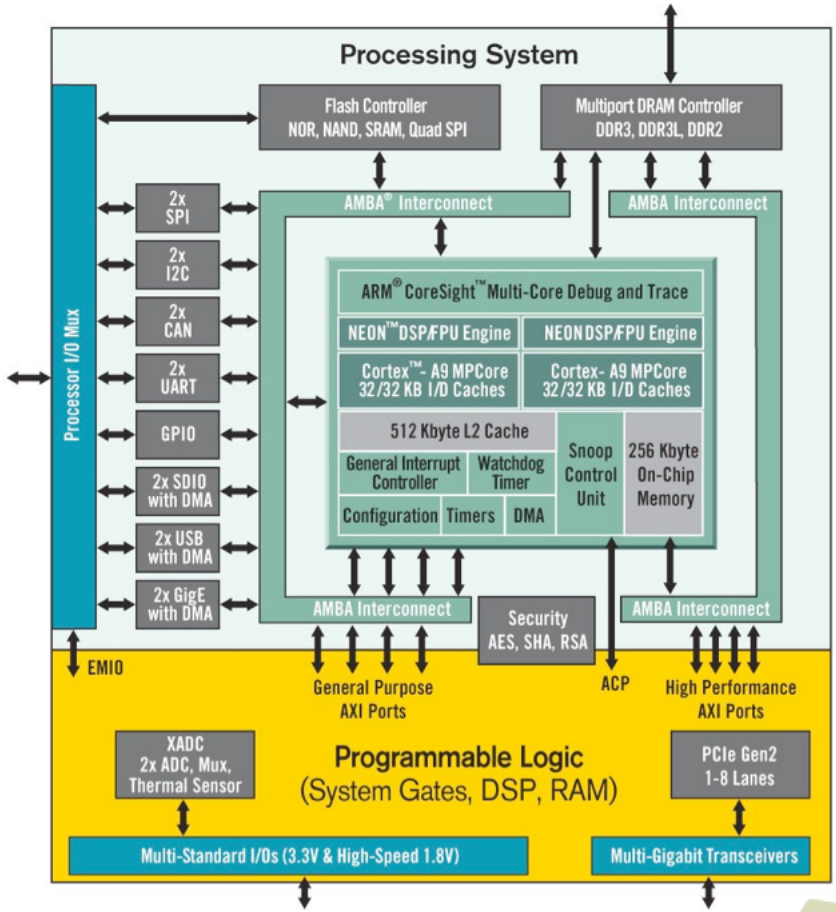


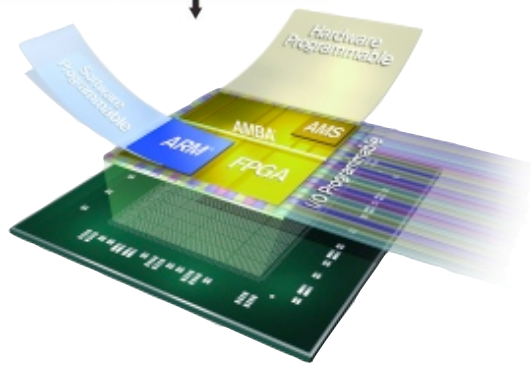


Xilinx Zinq

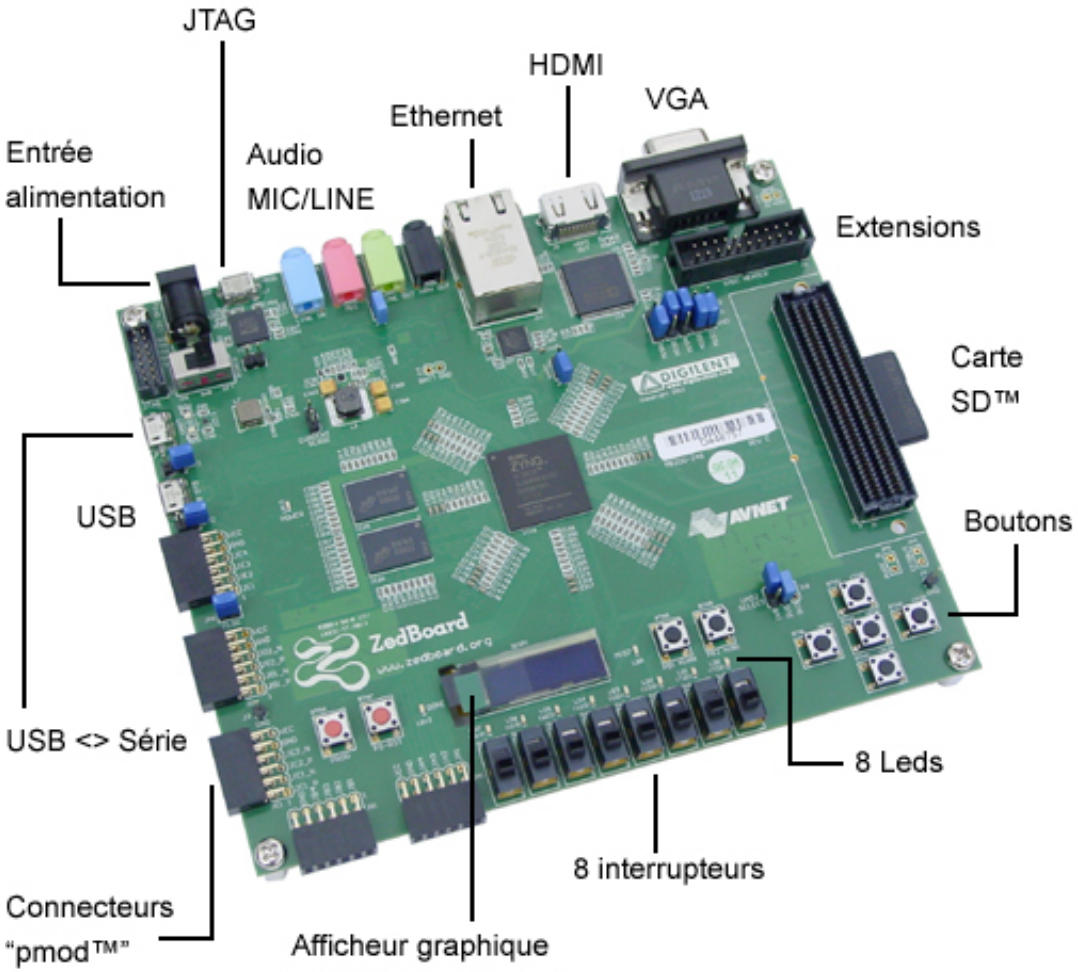
La logique de conception d'une architecture SoPC

- La gamme Zynq-7000 de Xilinx est la première gamme à intégrer un processeur dual-core ARM Cortex-A9 MPCore à une logique programmable jumelée sur une seule puce.
- Les processeurs fournissent:
  - Une manière élégante de gérer le contrôle et les périphériques (ethernet),
  - Un support applicatif (application sur le processeur et les accélérateurs sur la logique programmable),
  - Cortex-A9: librairies optimisées, chaines de compilation, OS embarqués, écosystème.

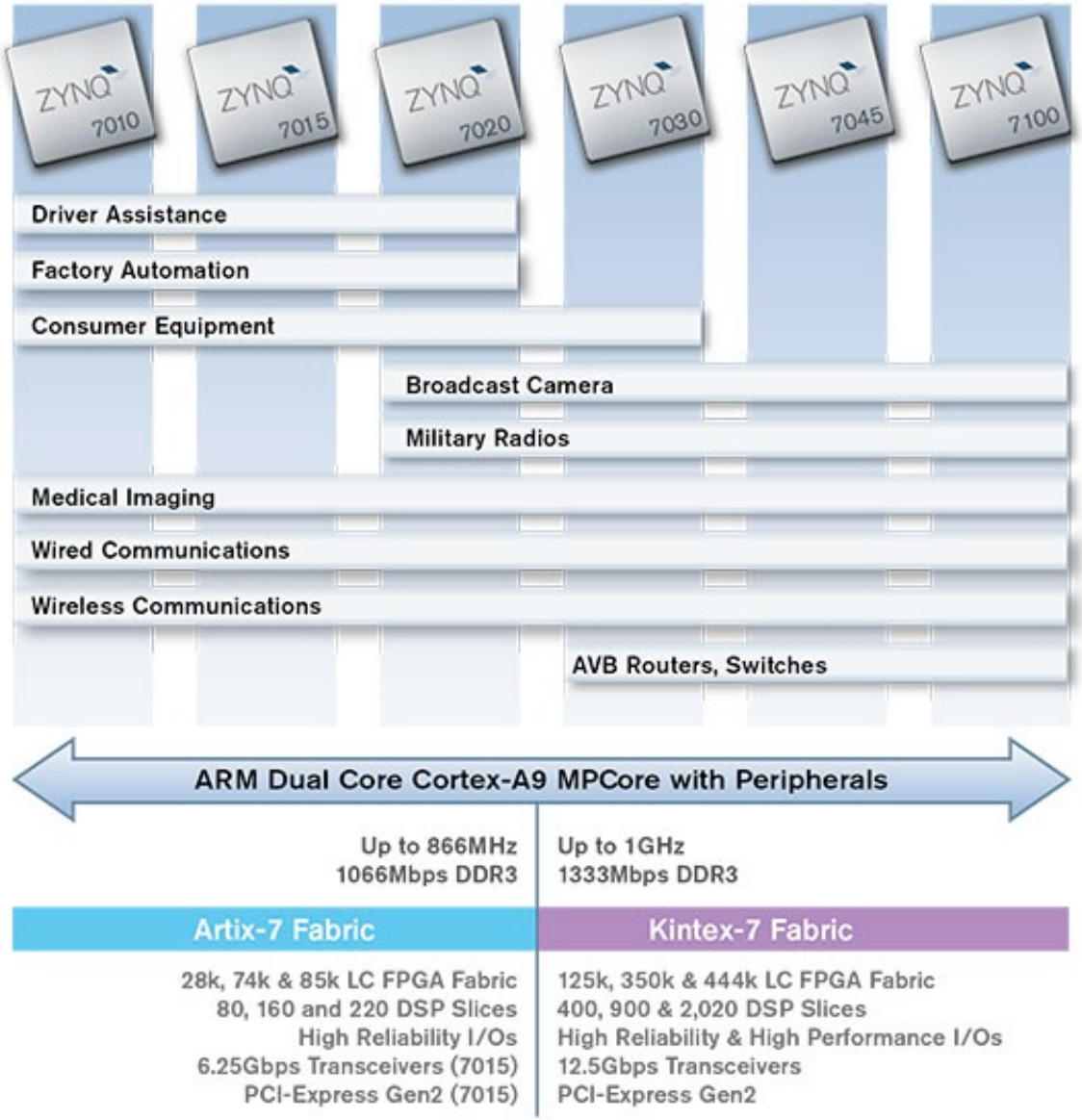




Quelques exemples de cartes de développement (ZedBoard)



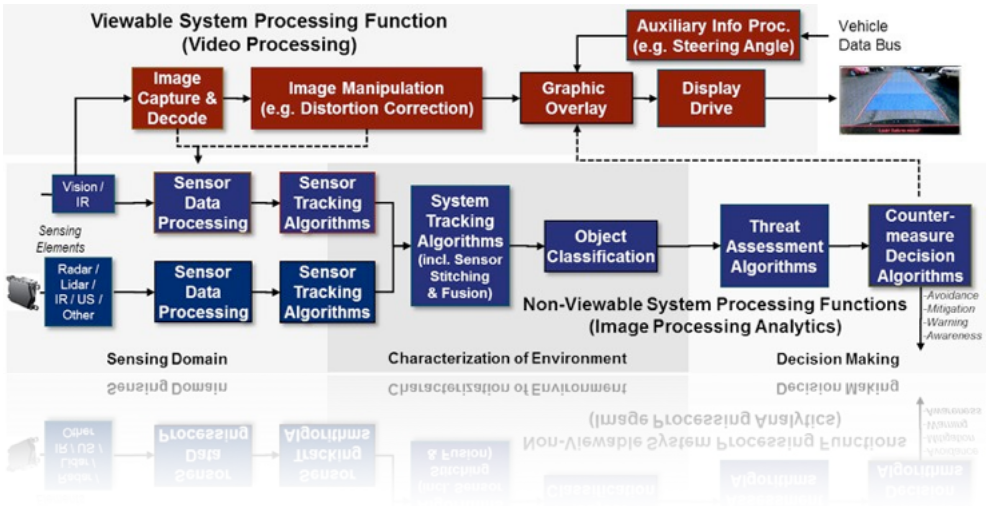
Les domaines d'utilisation des différents FPGA Xilinx

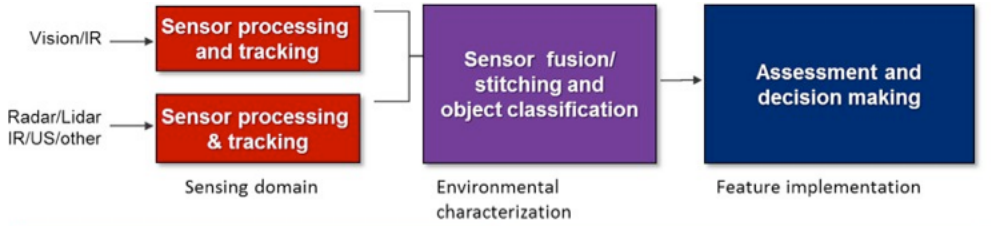


Différents FPGA construits sur la même architecture mais avec des propriétés différentes.

Une architecture SoPC (ARM bi-coeurs) mais avec des matrices de taille différentes (plus ou moins d'accélérateurs matériels).

Présentation de l'intérêt des SoC par Xilinx



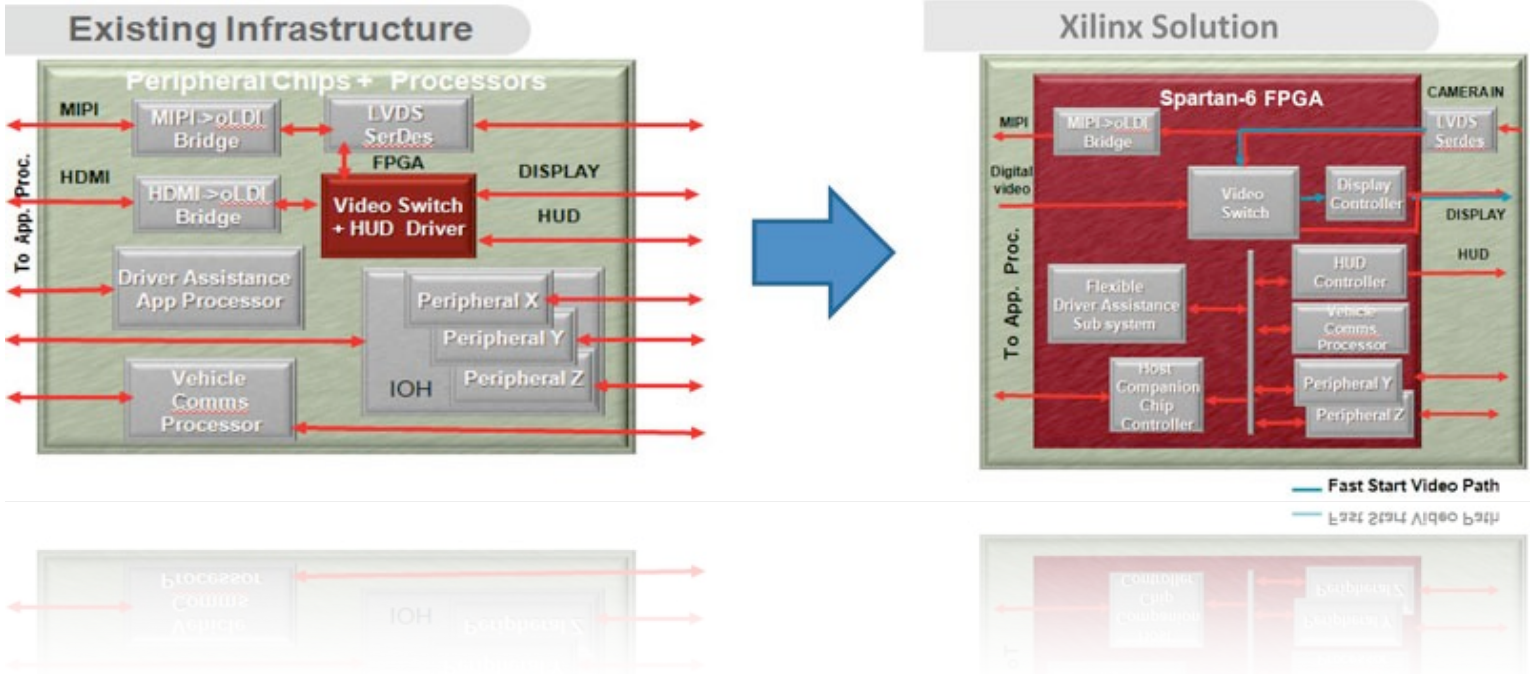


FPGA-based Parallel Hardware Processing

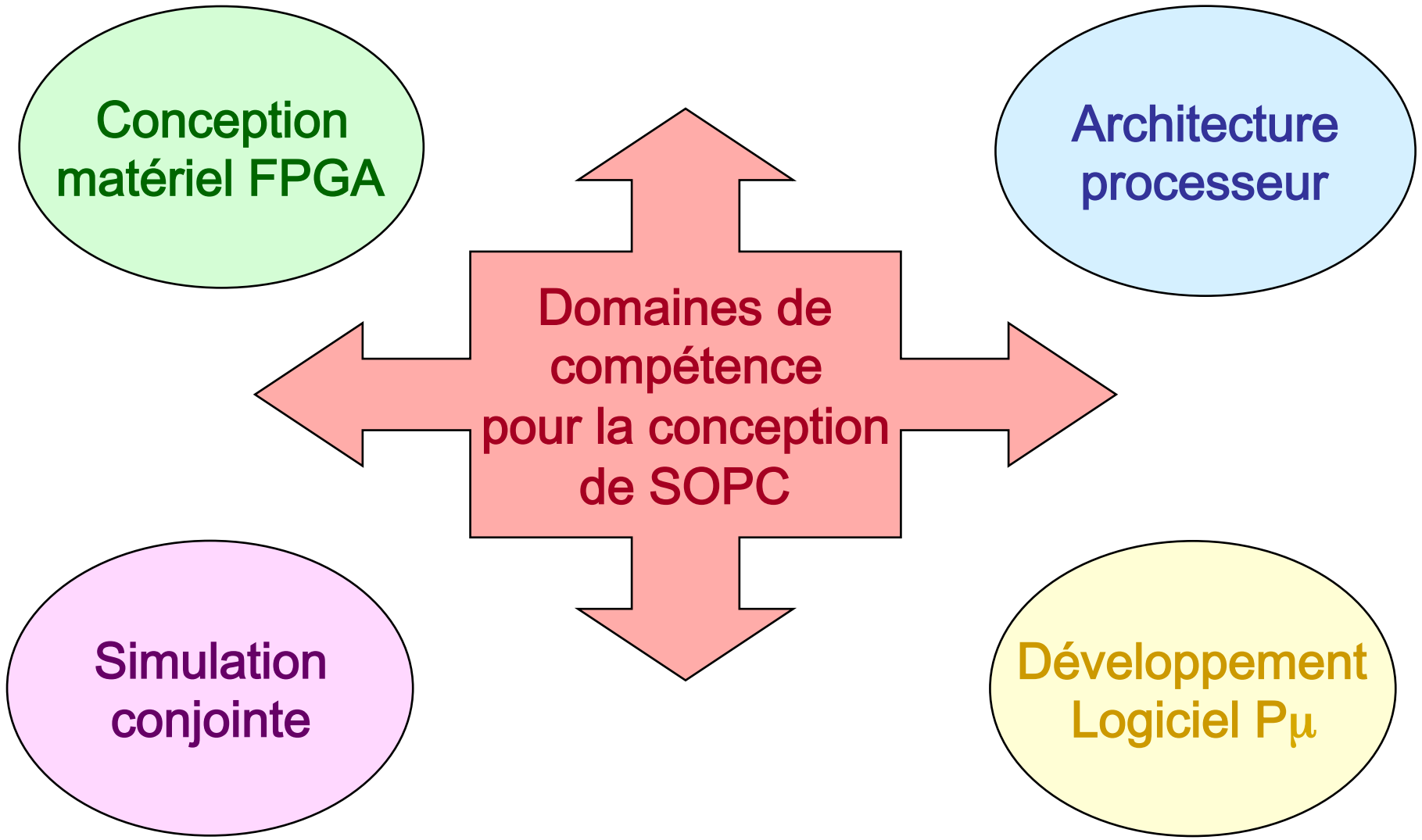
- FPGA fabric supports parallel processing necessary for pixel level analysis
- DSP blocks enable hardware acceleration of real-time sensors input

Processor-based Serial Software Processing

- Processors suited for serial decision making algorithms common in ADAS apps
- Processors enable feature bundling, such as camera sensors used for multiple applications

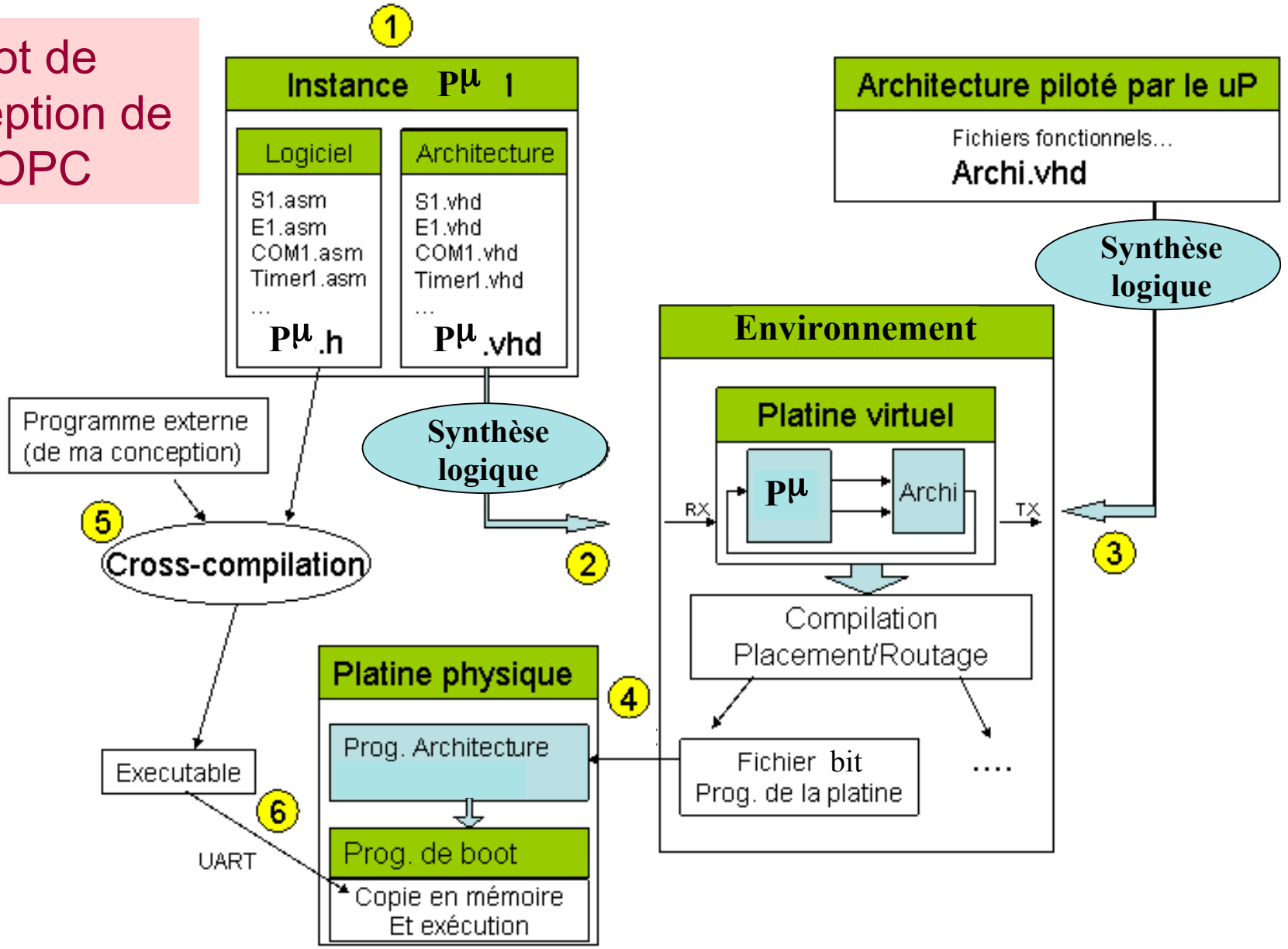


Les domaines de compétences nécessaires

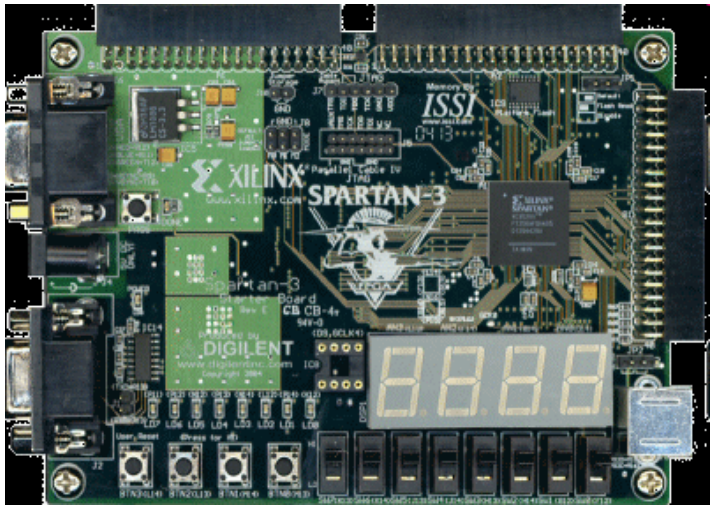


Flot de conception associé au design de SoCP

Flot de conception de SOPC



Protection de la Playstation 3 hacké (janvier 2010)



Carte Digilent avec
Spartan3 100$

George Hotz (né le 2 octobre 1989)
a cracké la PS3 en cinq semaine