

Etude des circuits logiques programmables

Les FPGA

Fabrice CAIGNET
LAAS - CNRS
fcaignet@laas.fr

Contenu :

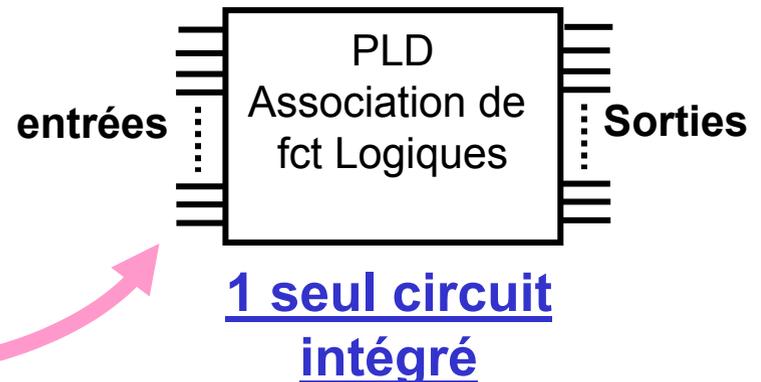
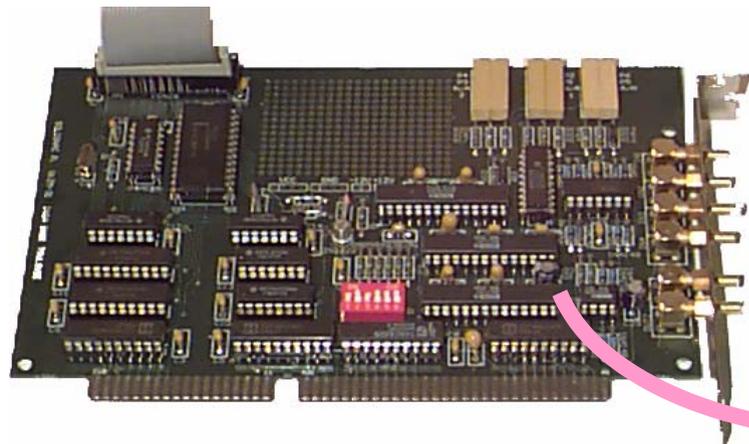
- I. Les Réseaux Logiques Programmables : PLD**
- II. Les technologies des éléments programmables**
- III. Les FPGAs**
- IV. Les outils de développement**
- V. Les tendances des composants programmables**
- VI. Le langage VHDL**

I. Les Réseaux Logiques Programmables : PLD

PLD = Programmable Logic Devices

Circuits logiques + ou - complexes remplaçant l'association de plusieurs boîtes logiques

- Simplification de câblage
- Encombrement réduit
- Diminution des coûts
- Facilité d'utilisation
- Diminution du risque des pannes

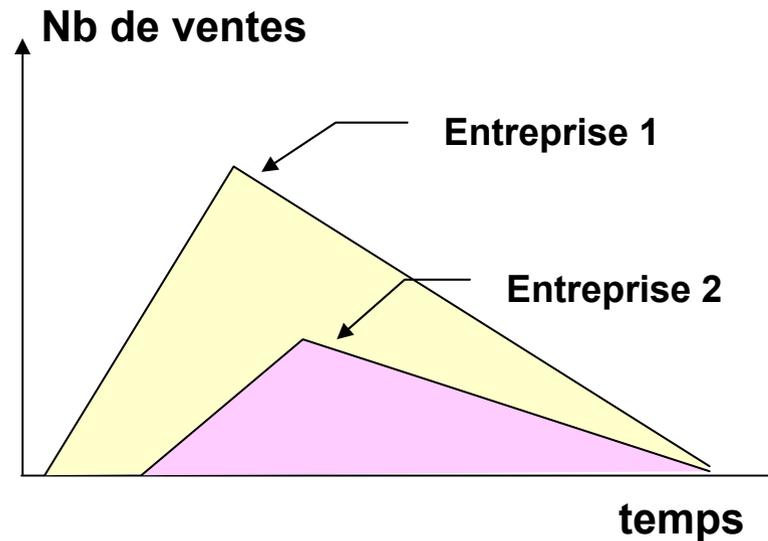


I. Les Réseaux Logiques Programmables : PLD

Importance du «Time To Market »

Le cahier des charges du développement d'un système doit prendre en compte les paramètres suivants :

- Le coût de production
- Les performances
- La consommation
- L'intégration
- L'évolution du produit
- La maintenance
- La protection industrielle



Deux alternatives existent :

- Les ASIC (Application Specific integrated Circuits)
- Les Circuits à réseaux programmables

I. Les Réseaux Logiques Programmables : PLD

Quelles différences entre un ASIC et un PLD

ASIC

Application Specific Integrated Circuits

Choix du fondeur

Conception du circuit
(full-custom - bibliothèques)

Très grand niveau d'intégration

Fabrication
à très grand nombre
d'exemplaire

- Un circuit dédié à une application
- Choix de la technologie
→ Maximum de performances

PLD

Programmable Logic Device

Choix du circuit

Programmation du circuit
(logiciel + interface circuit)

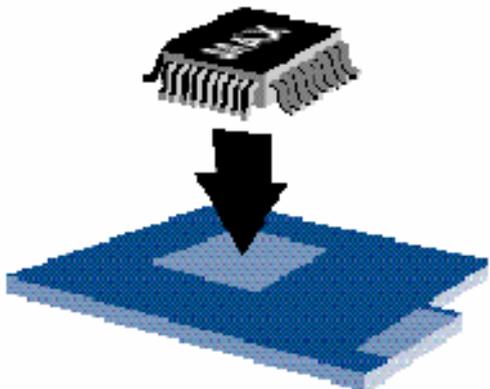
Intégration limitée

Implémentation
nombre d'exemplaires
limité

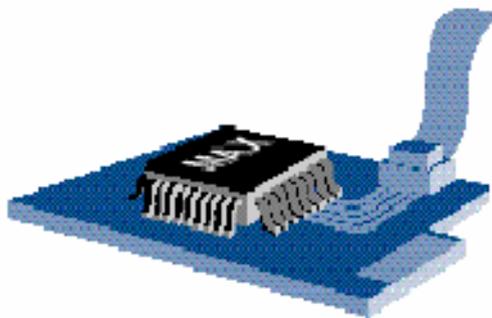
- Un circuit programmé pour une application
- Technologie figée mais :
→ Facilité de programmation

I. Les Réseaux Logiques Programmables : PLD

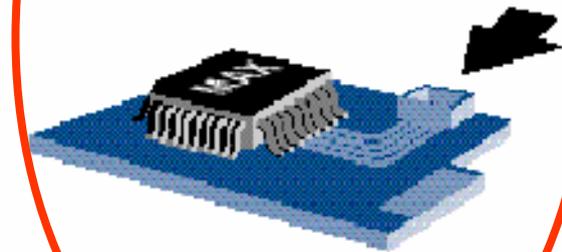
montage du circuit
non programmé



programmation
sur site



reprogrammation
éventuelle



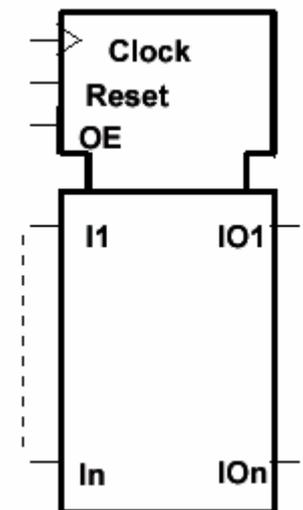
I. Les Réseaux Logiques Programmables : PLD

Avantages des PLD

- Le coût et temps de développement.
- Le prototypage et temps de simulation accéléré.
- La prise de risque est réduite. (modifications possibles, risque d'inventaire)
- Des supports logiciels peu chers.
- Des produits éprouvés. (pas de vecteurs de test structurel)

Inconvénients

- Un niveau d'intégration moindre aux ASIC.
- Une consommation plus élevée.
- Des prix importants et une faible disponibilité pour de grandes séries .

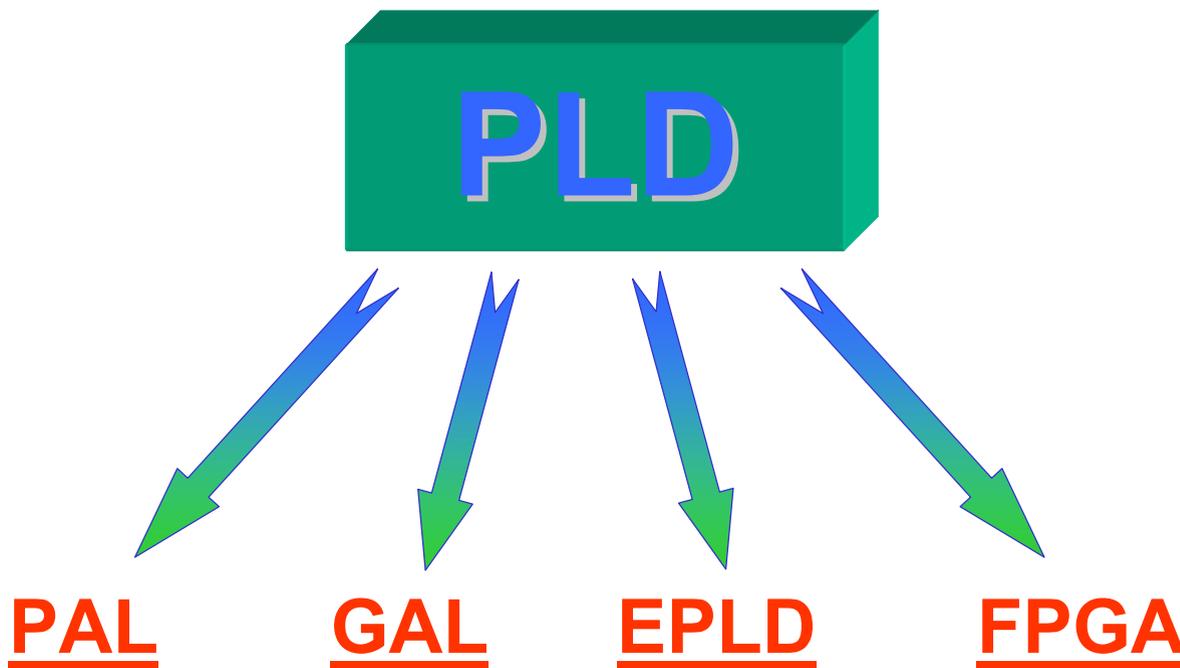


Symbolisation
Normalisée

I. Les Réseaux Logiques Programmables : PLD

Quatre Familles de PLD

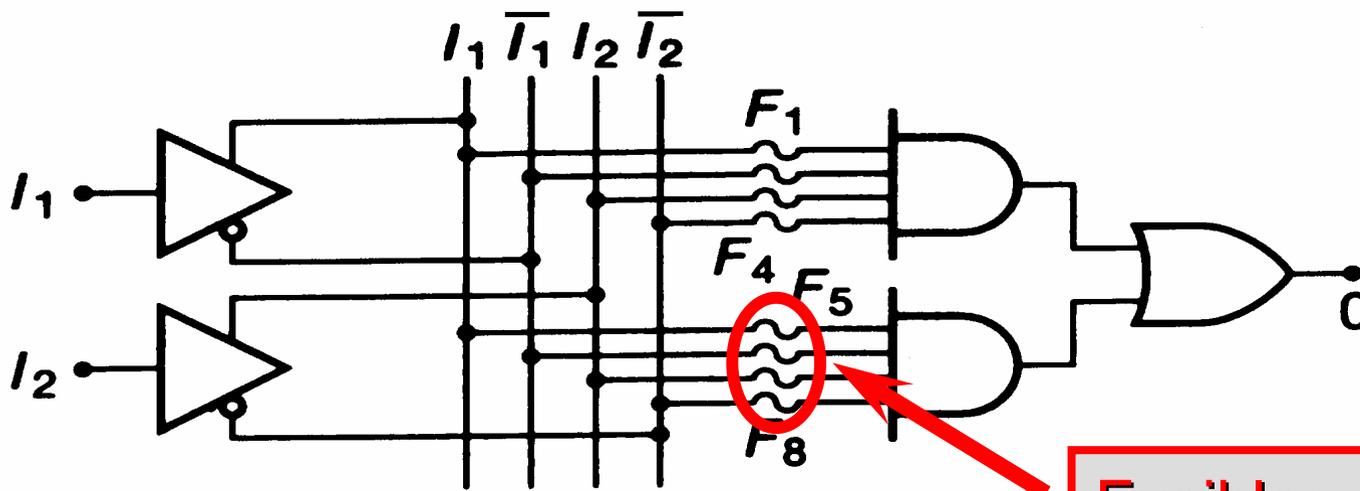
boîtes logiques



I. Les Réseaux Logiques Programmables : PLD

PAL (Programmable Array Logic) : réseaux logiques programmables

- Développés au début des années 70 par MMI (ex-AMD)
- La programmation se fait par destruction de fusibles
- Aucun fusible n'est grillé à l'achat de la PAL



Structure simplifiée d'un PAL
(2 entrées et 1 sortie)

Fusibles

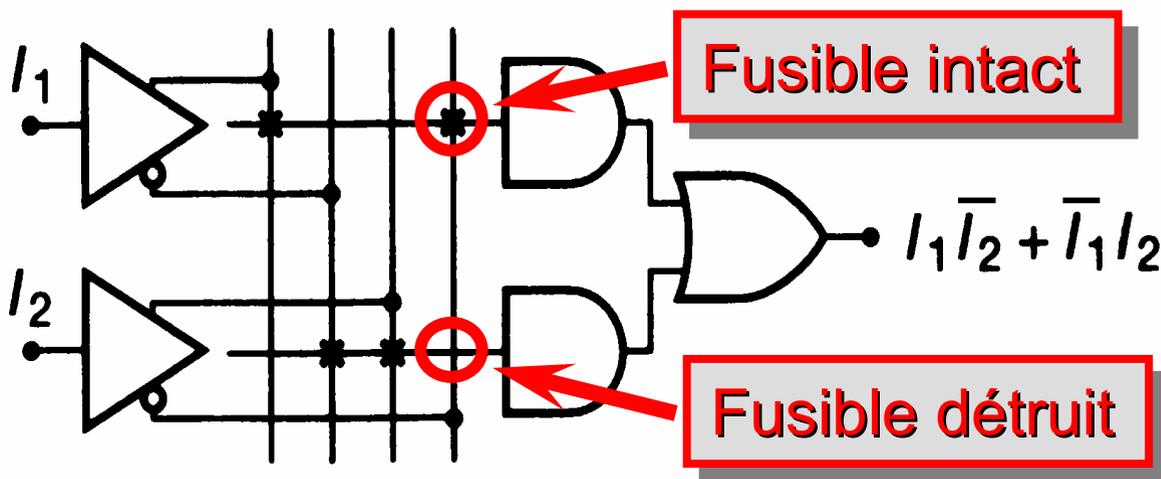
Les fonctions **ET** sont programmables

I. Les Réseaux Logiques Programmables : PLD

PAL (Programmable Array Logic) : réseaux logiques programmables

Représentation :

- les fusibles intacts sont représentés par une connexion
- les fusibles détruits sont représentés par une absence de connexion



Exemple d'un "OU EXCLUSIF"

I. Les Réseaux Logiques Programmables : PLD

PAL (Programmable Array Logic) : réseaux logiques programmables

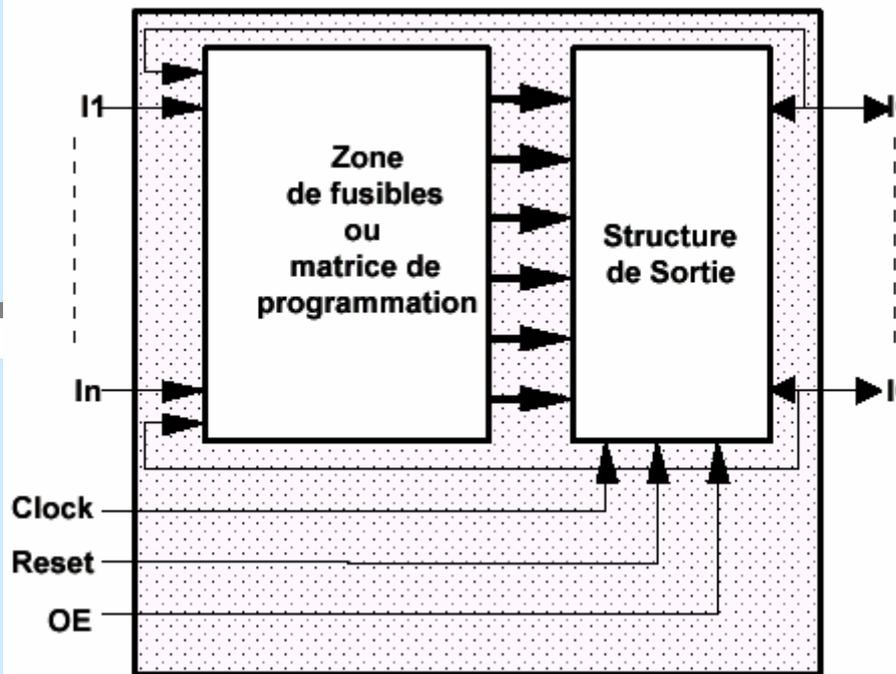
➡ Que peut-on trouver dans une PAL??

Tout *P.A.L.* est constitué :

- D'entrées (Input): I_1 à I_n avec $8 < n < 20$.
- De sorties (Output) Ou d'entrées / sorties (I/O) de type Totem Pôles ou Trois Etats : O_1 à O_n ou IO_1 à IO_n ($2 < n < 15$).

On trouve aussi :

- Une entrée d'horloge (Clock): Clk ou Clock.
- Une entrée de validation des sorties trois états: OE (Output Enable) ou Enable.
- Une entrée de remise à zéro des registres: RESET.

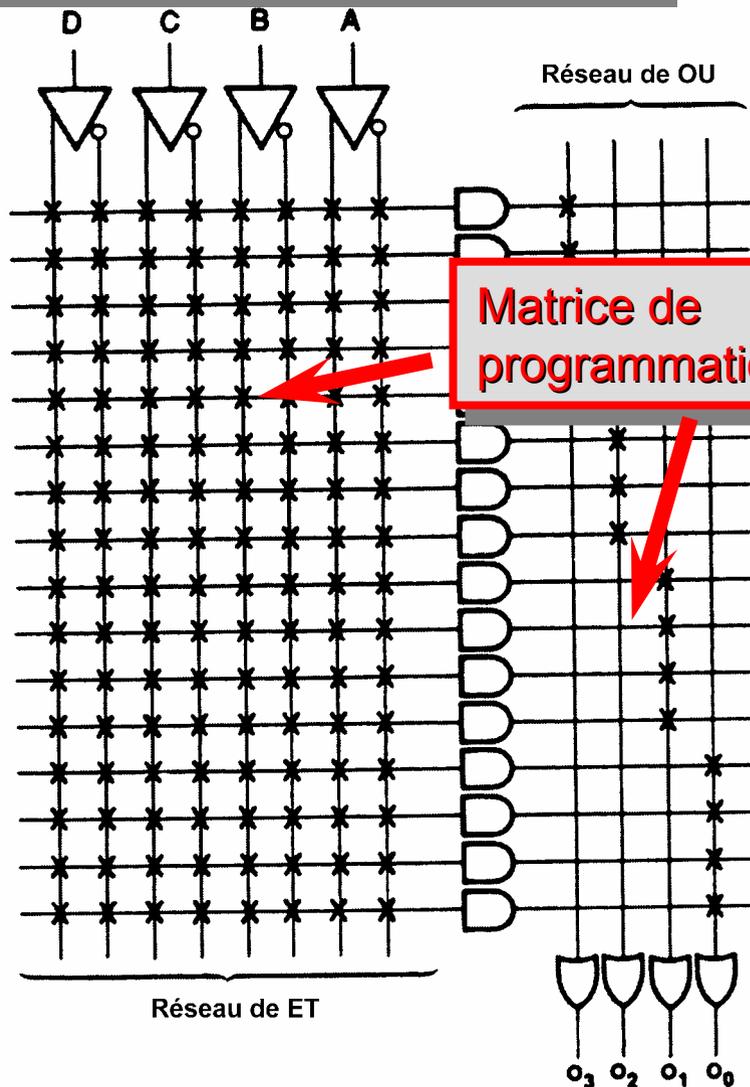


I. Les Réseaux Logiques Programmables : PLD

PAL : Matrice de programmation

La plupart des *P.A.L.* sont constitués :

- Un ensemble de portes « ET » sur lesquelles viennent se connecter les variables d'entrée et leurs compléments.
- Un ensemble de portes « OU » sur lesquelles les sorties des opérateurs « ET » sont connectées les variables d'entrée.



I. Les Réseaux Logiques Programmables : PLD

PAL : Structure de sortie

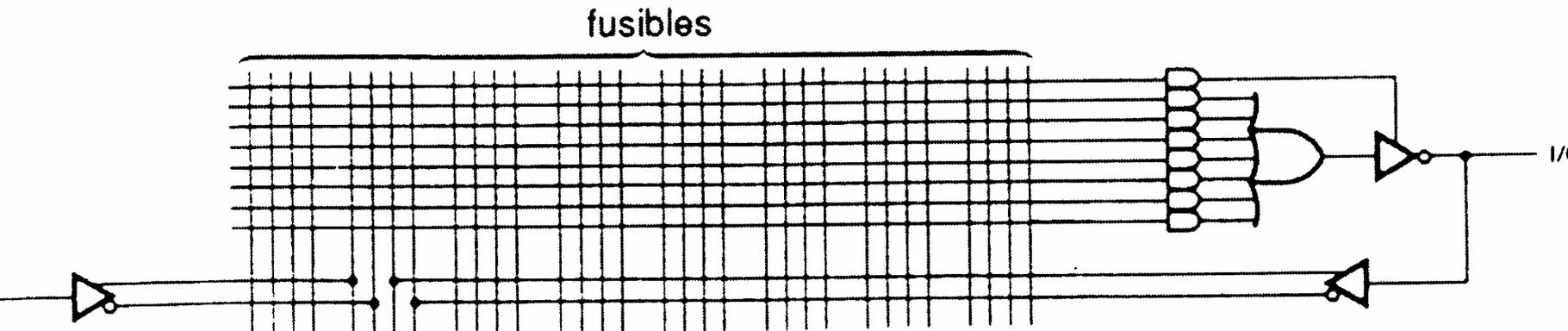
Il existe 3 structures de sortie qui peuvent-être aussi de entrées/sorties :

- Combinatoire
- Séquentielle
- versatiles

- Entrées / Sorties combinatoires

Sortie 3 états rebouchée vers la matrice

- ➔ une sortie peut servir de variable intermédiaire
- ➔ une sortie peut servir d'entrée (en mode haute impédance)



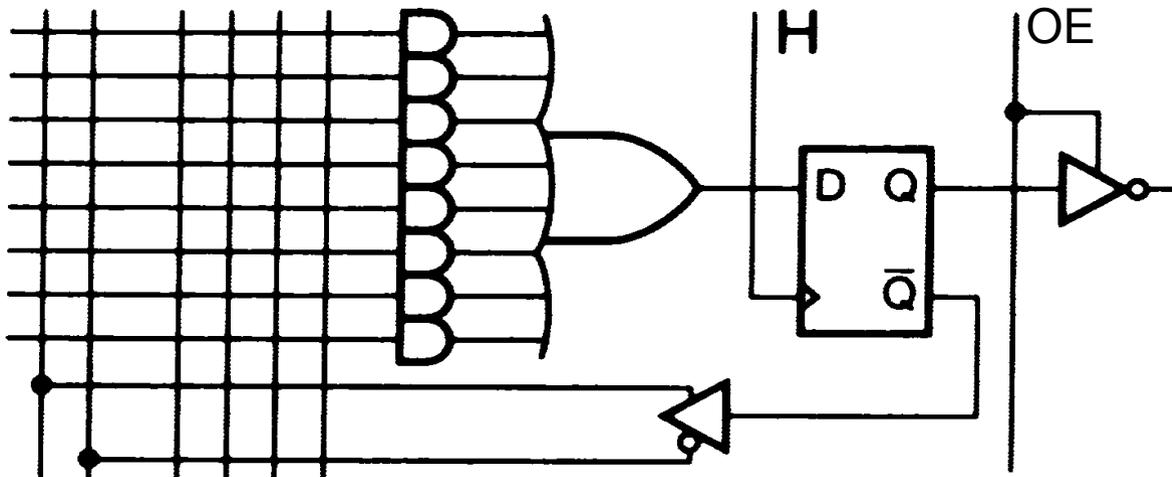
I. Les Réseaux Logiques Programmables : PLD

PAL : Structure de sortie

- Entrées / Sorties séquentielles, à registre

Une bascule D permet la logique séquentielle

- ➔ sorties séquencées sur une horloge H
- ➔ une sortie ne peut pas servir d'entrée
- ➔ une commande OE (Output Enable) permet de désactiver la sortie



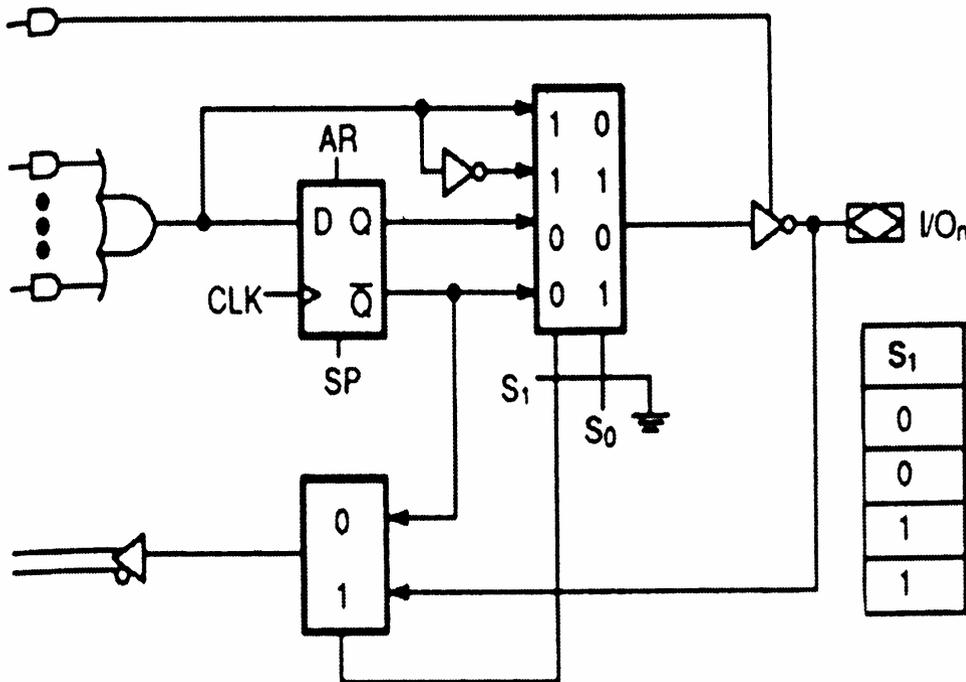
Sortie à REGISTRE

I. Les Réseaux Logiques Programmables : PLD

PAL (Programmable Array Logic) : Structure de sortie

- Entrées / Sorties versatiles (VPAL)

➔ On peut configurer par programmation le mode d'utilisation de la broche de sortie



S ₁	S ₀	Configuration des sorties
0	0	Registre, actif bas
0	1	Registre, actif haut
1	0	Combinatoire, actif bas
1	1	Combinatoire, actif haut

Entrée / Sortie VERSATILE



I. Les Réseaux Logiques Programmables : PLD

[PAL \(Programmable Array Logic\) : Exemple de PAL22V10 \(AMD\)](#)

I. Les Réseaux Logiques Programmables : PLD

GAL (Généríc Array Logic), « LATICE Semiconductor »

Réseau Logique Reprogrammable

Un **GAL** est un
PAL effaçable
électriquement

I. Les Réseaux Logiques Programmables : PLD

EPLD

Erasable Programmable Logic Device

Circuit Logique
reprogrammable

de grande capacité

I. Les Réseaux Logiques Programmables : PLD

FPGA

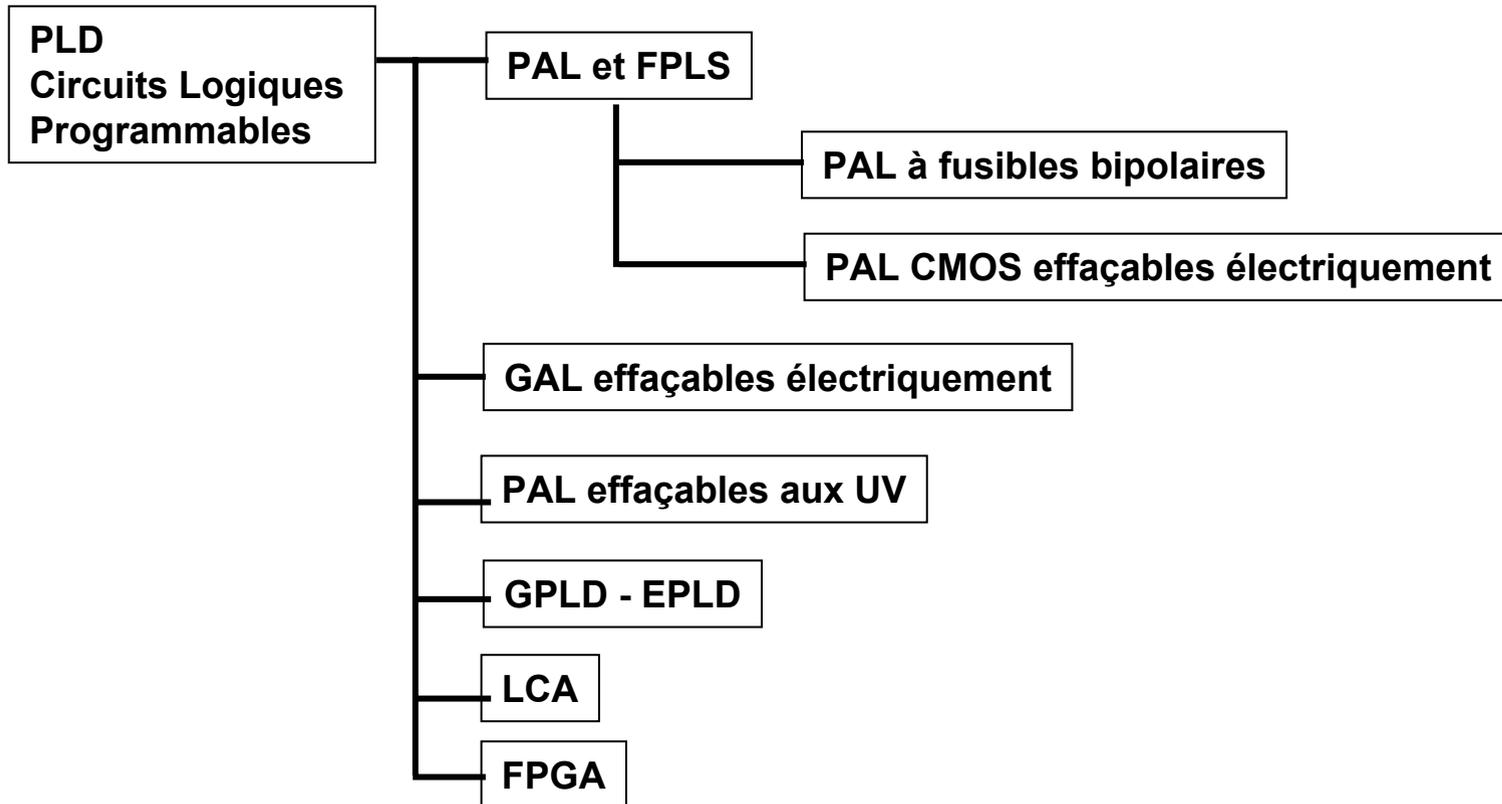
Forecasting Programmable Gate Array

La densité croissante des circuits programmables actuels, notamment des FPGA (Field Programmable Gate Array), permet le prototypage rapide des circuits numériques à grande complexité. Aussi, il est possible de tester rapidement la validité de concepts architecturaux nouveaux: l'implémentation complète d'un processeur sur des circuits FPGA est aujourd'hui à notre portée, entraînant ainsi plus de possibilités d'évaluation que celles offertes par des simulateurs logiciels.

De plus, la reprogrammabilité de certains circuits FPGA a ouvert de nouvelles voies de recherche: des méthodologies de conception des systèmes reconfigurables, capables d'évoluer ou de s'adapter à des environnements ou à des contraintes variables.

I. Les Réseaux Logiques Programmables : PLD

Résumé graphique des familles de *P.L.D.*



I. Les Réseaux Logiques Programmables : PLD

Les principaux critères de choix des *P.L.D.* sont:

- *Vitesse de fonctionnement*
- *Nombre de portes*
- *Consommation*
- *Technologie*
- *Prix*

	circuits MSI (à titre de comparaison)	PLD	CPLD	FPGA
nombre de portes (ordre de grandeur)	100	150	40 000	4 000 000
vitesse de fonctionnement (ordre de grandeur)	100 MHz	200 MHz	240 MHz	100 MHz
technologie de connexion		MOS à grille flottante	MOS à grille flottante	SRAM et antifusible
codage des fonctions		PLA	PLA et LUT	LUT et MUX

II. Les technologies des éléments programmables

Généralité sur les technologies des éléments programmables

On trouve les éléments programmables dans les blocs logiques des PLDs, afin de leur donner une fonctionnalité, mais aussi dans les matrices d'interconnexions entre ces blocs. Un élément programmable peut être considéré comme un interrupteur. Afin de respecter les contraintes imposées à l'ingénieur, les éléments programmables doivent posséder plusieurs qualités :

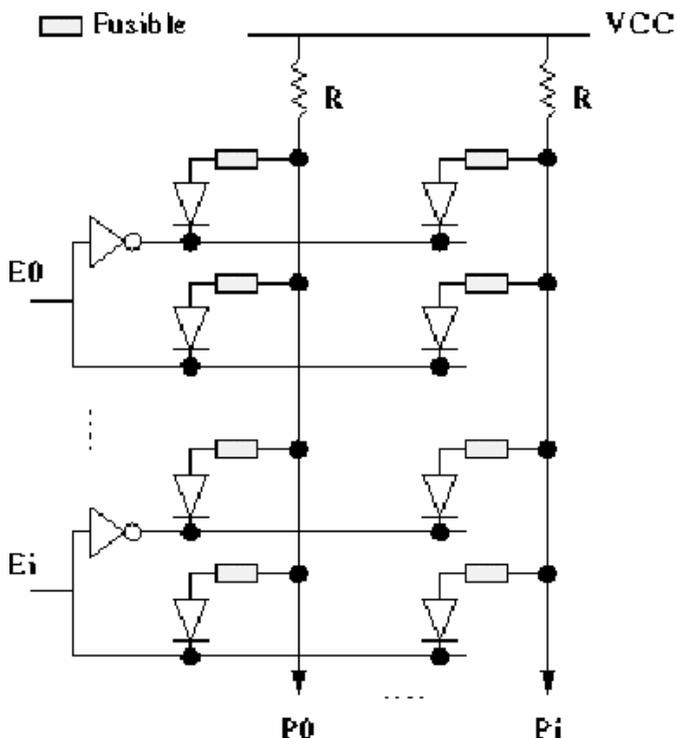
- ➔ - Ils doivent occuper une surface la plus petite possible (Ce point s'explique pour des raisons évidentes de coût. Ceci est d'autant plus vrai que l'on désire en disposer d'un grand nombre).
- ➔ - Ils doivent posséder une résistance de passage faible et une résistance de coupure très élevée.
- ➔ - Ils doivent apporter un minimum de capacité parasite.

Les deux derniers points s'expliquent quant à eux pour des raisons de performance en terme de fréquence de fonctionnement du PLD. Plus la résistance et la capacité sur le chemin d'un signal sont faibles, plus la fréquence de ce signal peut être élevée (RC effet).

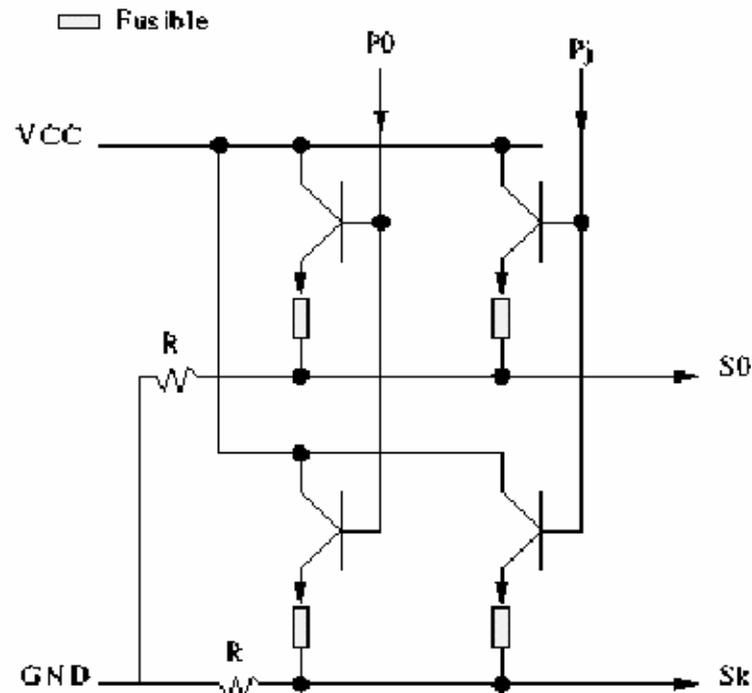
II. Les technologies des éléments programmables

Les technologies à fusibles

Les fusibles sont grillés en appliquant des tensions élevées (12V)



Technologie fusible-diode



Technologie fusible-transistor

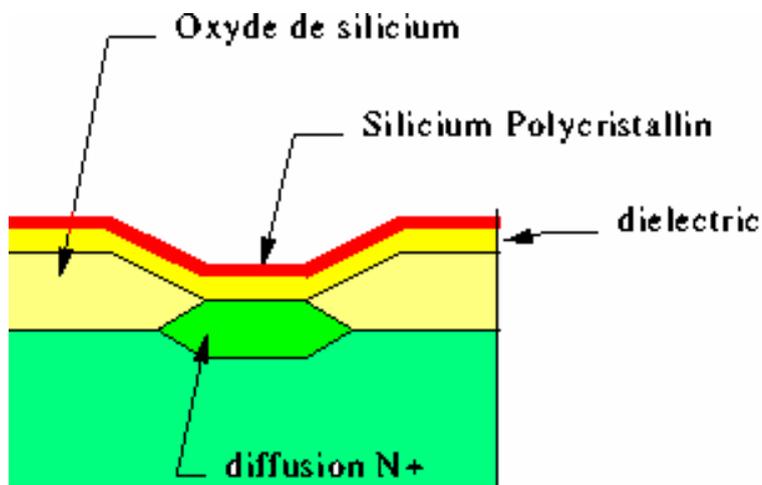
Technologies maintenant abandonnées pour des raisons de manque de fiabilité.
Le fait de "griller" les fusibles provoque des perturbations qui peuvent affecter le reste du circuit.
De plus, cette programmation est irréversible et ne permet donc pas la re-programmabilité.

II. Les technologies des éléments programmables

Les technologies à anti-fusibles (Actel, QuickLogic, Crosspointet Xilinx)

un anti-fusible est un élément programmable qui à l'inverse des fusibles n'est passant qu'après programmation.

La connexion s effectue en détruisant un diélectrique

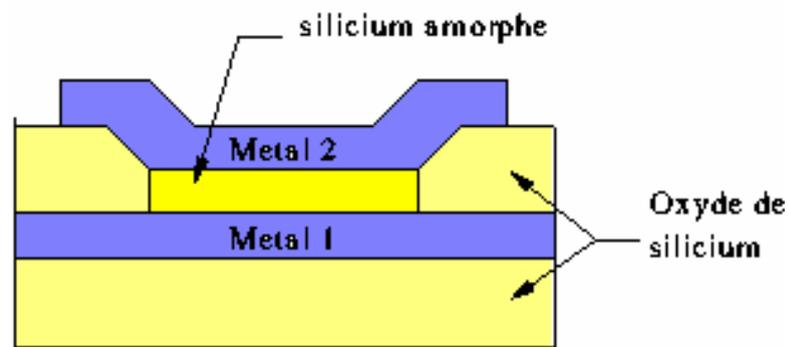


Technologie anti-fusible PLICE



Petite taille (grande capacité d'intégration)

Base impédance (rapidité)



Technologie anti-fusible VIA-Link



Technologie difficile à maîtriser

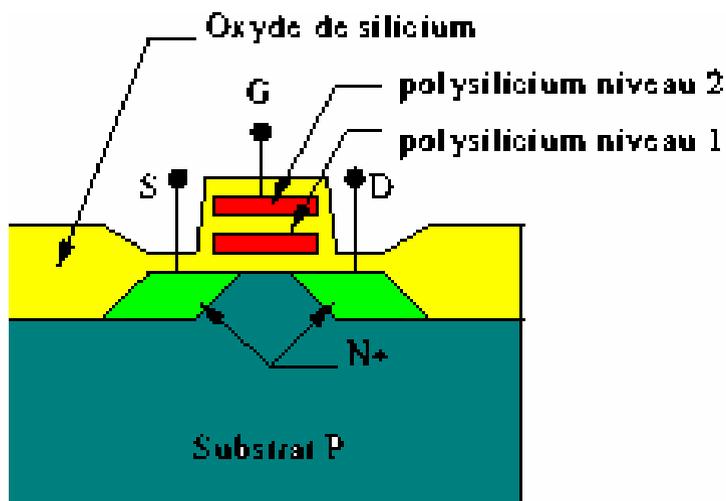
ne permettent pas la reprogrammation

II. Les technologies des éléments programmables

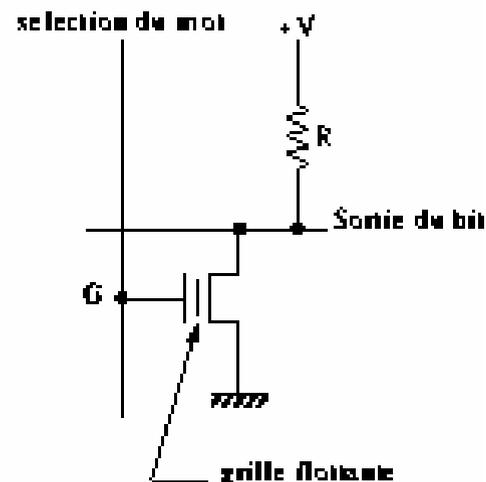
Les technologies à EPROM/FLASH (Actel, AMD...)

Il existe deux variantes de l'EPROM,

- Erasable Programmable Read Only Memory classique (EPROM)
- Electrically Erasable Programmable Read Only Memory (EEPROM).



Cellule à grille flottante d'une EPROM



Système d'interconnexion

Petite taille (grande capacité d'intégration)

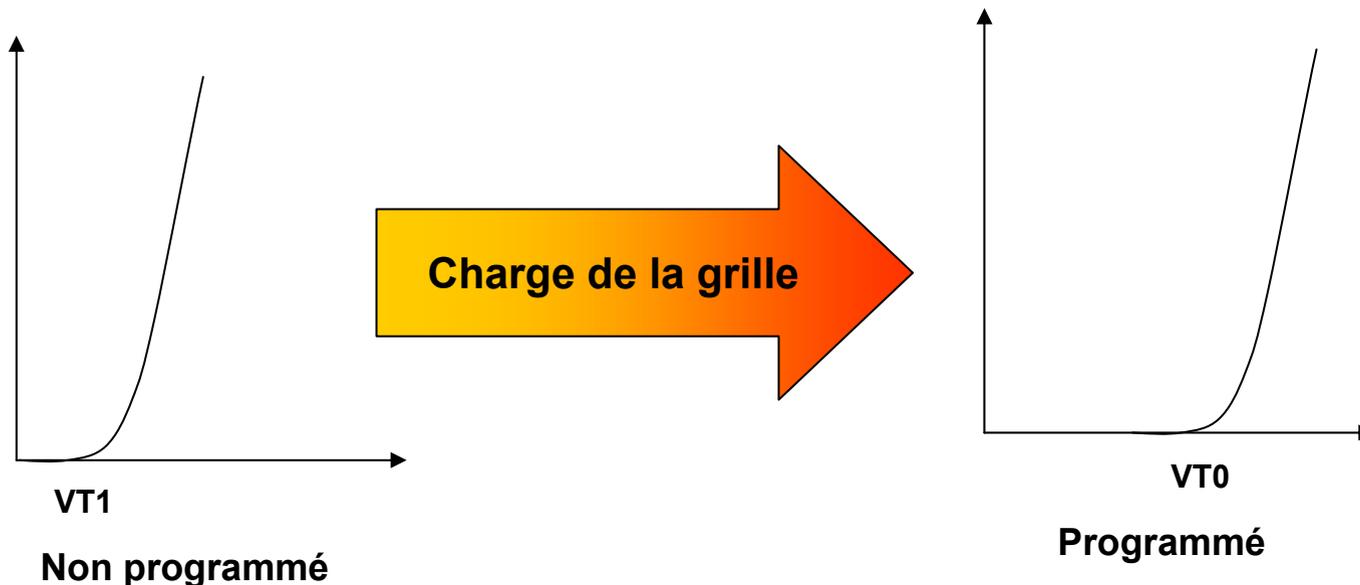
Base impédance (rapidité)

Possibilité de reprogrammation à loisir

II. Les technologies des éléments programmables

Les technologies à EPROM/FLASH (Actel, AMD...)

Le système à Grille flottante :



➡ **Etat non programmé** : il n'existe pas de charge dans la grille flottante et le transistor peut être utilisé normalement.

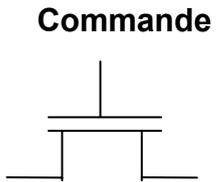
➡ **En cours de programmation** (effet tunnel, passage d'un fort courant entre la source et le drain), une partie de la charge est retenue dans la grille flottante. Le seuil du transistor augmente,

➡ Le transistor est bloqué jusqu'à ce qu'il soit "effacé".

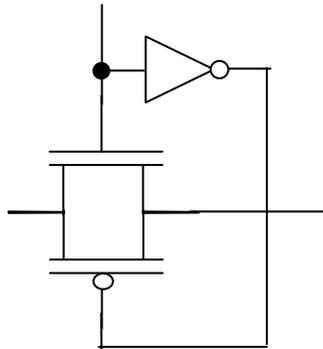
II. Les technologies des éléments programmables

Les technologies à RAM statique -SRAM (XILINX, ALTERA...)

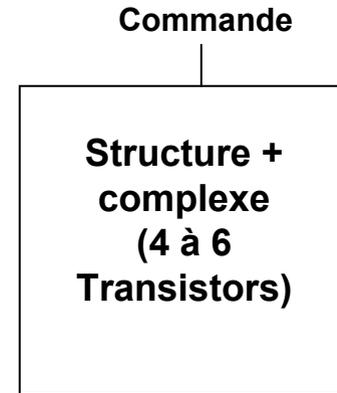
Elle fait appel à une phase de chargement de la configuration depuis l'extérieur



Transistor de passage



Portes de passage



Portes de passage



Grande possibilité de programmation
Technologie de type CMOS (très bien maîtrisée)

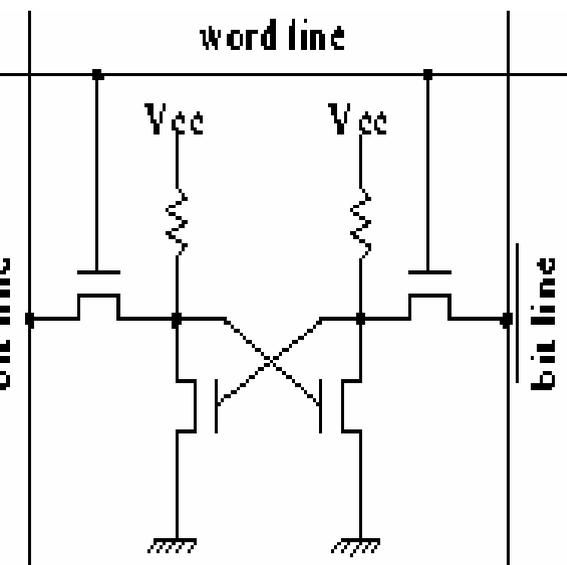


Place importante
Mémoire volatile (reprogrammation obligatoire)

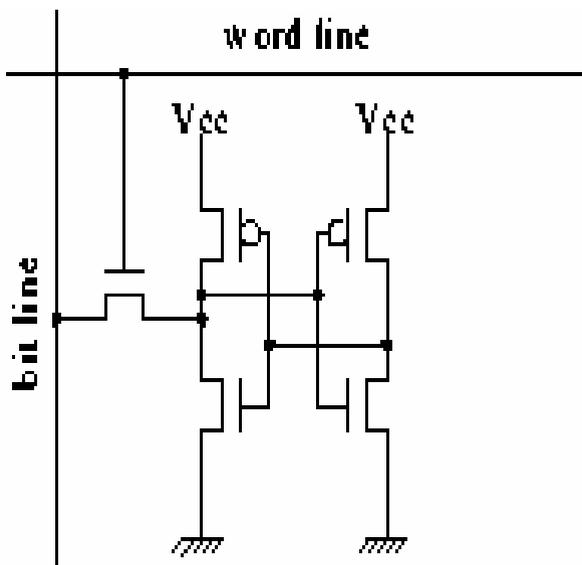
II. Les technologies des éléments programmables

Les technologies à RAM statique -SRAM (XILINX, ALTERA...)

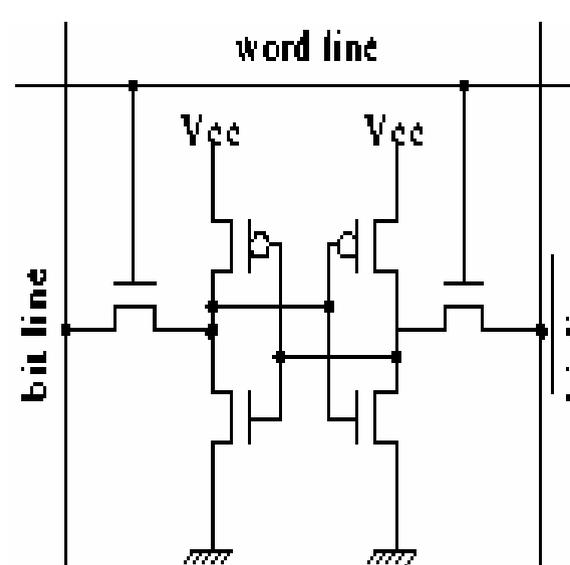
Exemples de structure SRAM implantées



(l) SRAM à 4 transistors



(m) SRAM à 5 transistors



(n) SRAM à 6 transistors

II. Les technologies des éléments programmables

Récapitulatif :

Nom	Re-programmabilité	vitesse de reprogrammation	Volatile	Surface	technologie
fusible	non	-	non	++++	bipolair
anti-fusible	non	-	non	++++	CMOS+
PROM	non	-	non	+++	CMOS
EPROM	oui (hors du circuit)	+	non	+++	UVCMOS
EEPROM	oui (dans le système)	++	non	++	EECMOS
SRAM	oui (dans le système)	+++	oui	+	CMOS

La technologie EPROM

Les PLD à EPROM se programment électriquement et s'effacent aux UV,
Par contre

Les PLD à EEPROM se programment quasi instantanément, et gardent la configuration jusqu'à une nouvelle programmation (même en l'absence de tension)

La technologie EEPROM

Facile et rapide à programmer, la configuration disparaît sans alimentation.

II. Les technologies des éléments programmables

Technologies utilisées par les différents fabricants

ACTEL	ANTIFUSE, SRAM
ALTERA	EPROM, EEPROM, SRAM
AMD	EEPROM
ATMEL	SRAM
LATTICE	EPROM, EEPROM
XILINX	SRAM, ANTIFUSE, EPROM, EEPROM

III. Les FPGA

Rappel des caractéristiques principales

Comme nous venons de le voir, les éléments programmable sont à la base des caractéristiques des FPGA

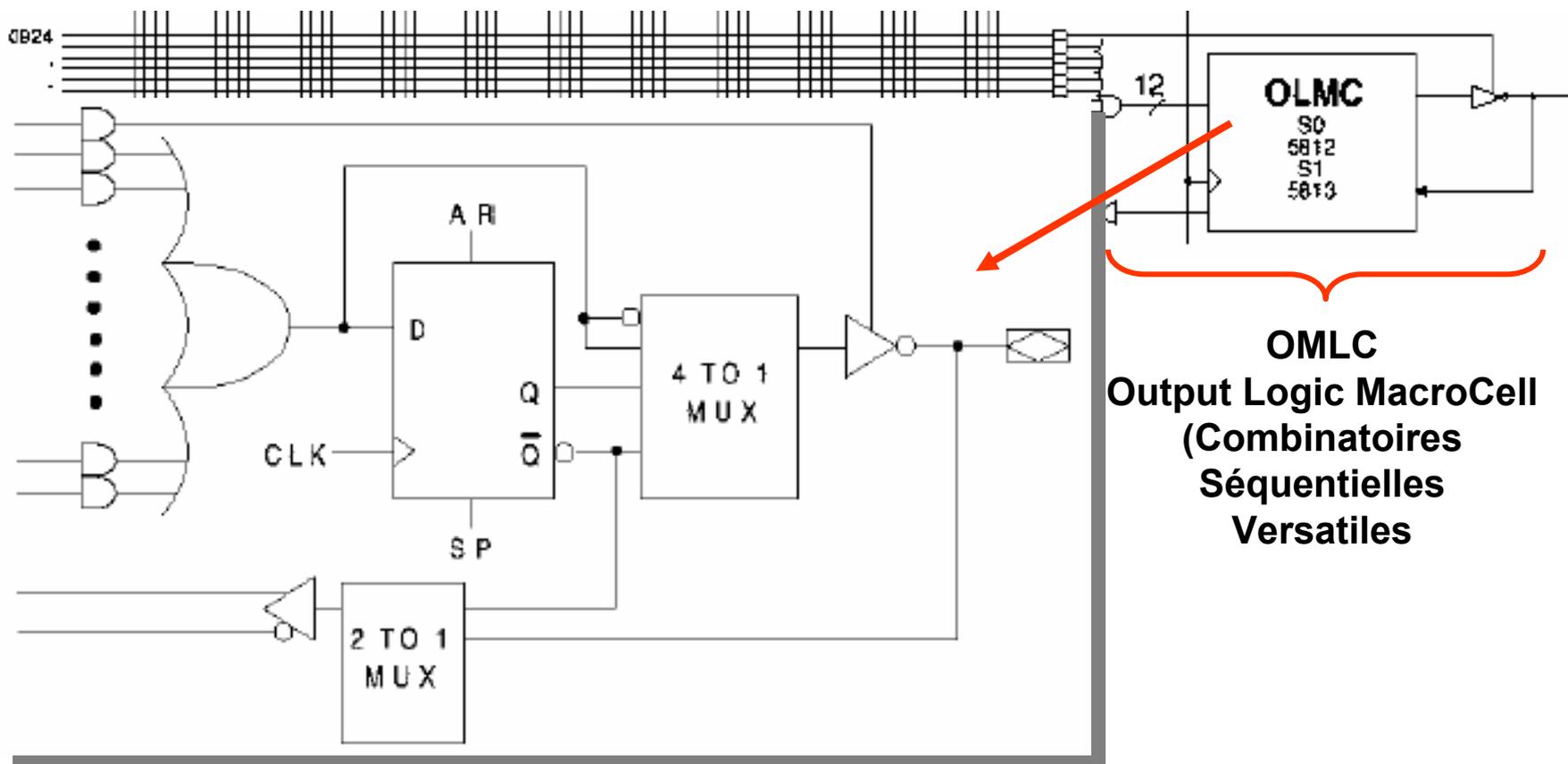
Le choix d'un PLD dépendra dont :

- La densité d'intégration
- De la rapidité de fonctionnement (paramètre directement lié à l'impédance intrinsèque des éléments programmables
- De la facilité de mise en œuvre (programmation, re-programmation...)
- de la possibilité de maintien de l'information

III. Les FPGA

Rappel des caractéristiques principales

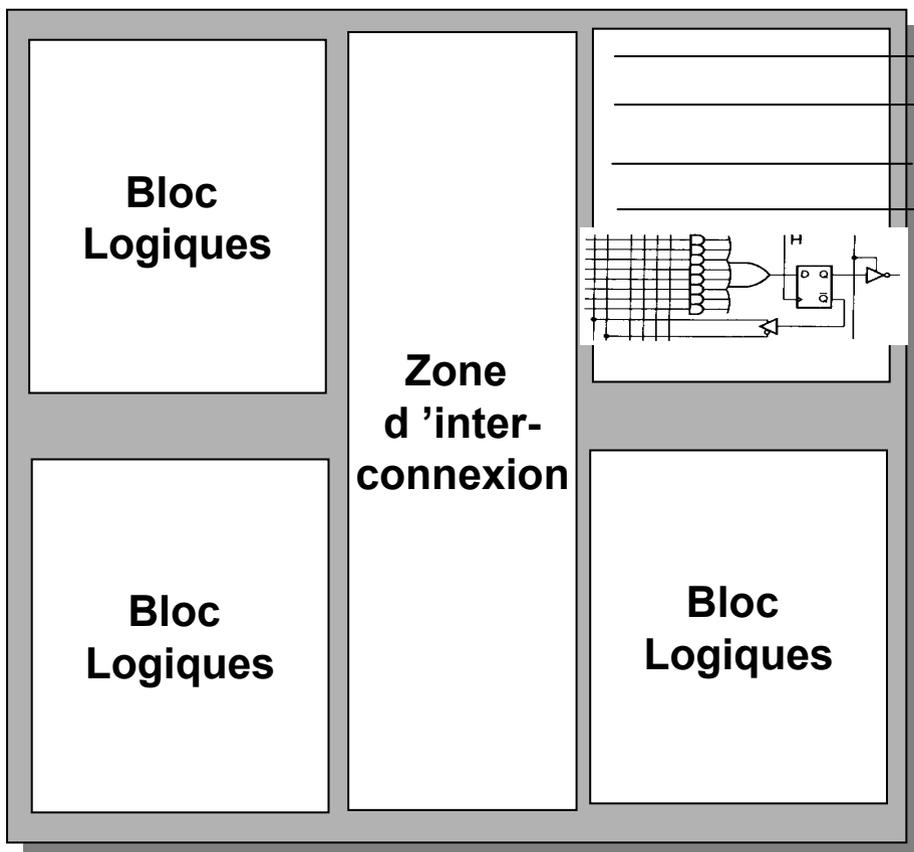
Les structures PAL sont à la base des CPLD, FPGA d'aujourd'hui



III. Les FPGA

Les CPLD (Complexe Programmable Logic Device) :

Ces circuits ont une capacité de portes et de configuration très largement supérieur aux PAL



Macro-cellules composées de :

- une zone de portes logiques
- un bascule

- Architecture identique aux PAL
- équivalent à plusieurs PAL séparées par une zone d'interconnexion

- 100 à 100000 portes
- 16 à 1000 bascules

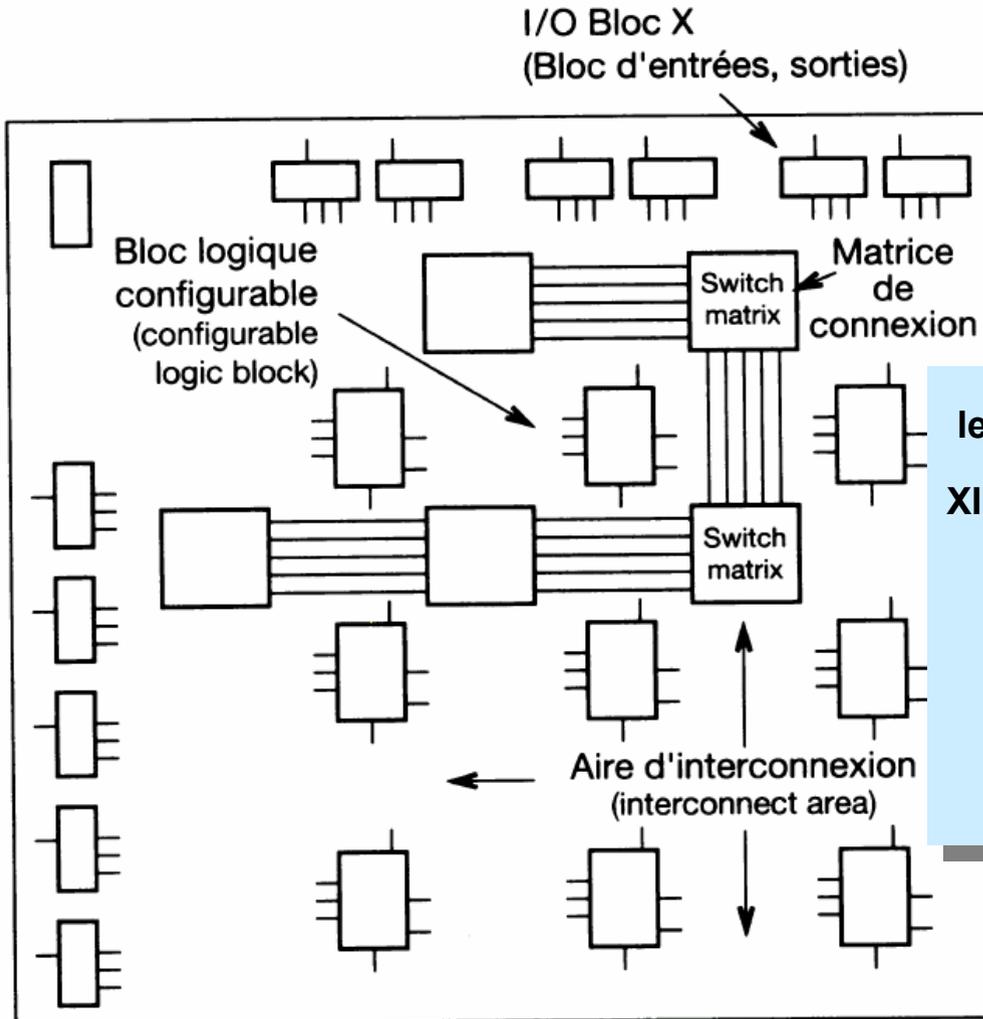
III. Les FPGA

Les FPGA (Fields Programmable Gate Array) : Réseau de portes programmables à la demande - XILINX et ALTERA.

- les FPGA à la différence des CPLD sont assimilables à des A.S.I.C. programmables par l'utilisateur.
- La puissance de ces circuits est telle qu'ils peuvent être composés de plusieurs milliers voir millions de portes logiques et de bascules. Les dernières générations de FPGA intègrent même de la mémoire vive (RAM). Les deux plus grands constructeurs de FPGA sont XILINX et ALTERA.
- Ils sont composés de blocs logiques élémentaires (plusieurs milliers de portes) qui peuvent être interconnectés.
- Critère de choix : vitesse de fonctionnement plus élevées pour les CPLD

III. Les FPGA

Structure interne d'un FPGA type XILINX

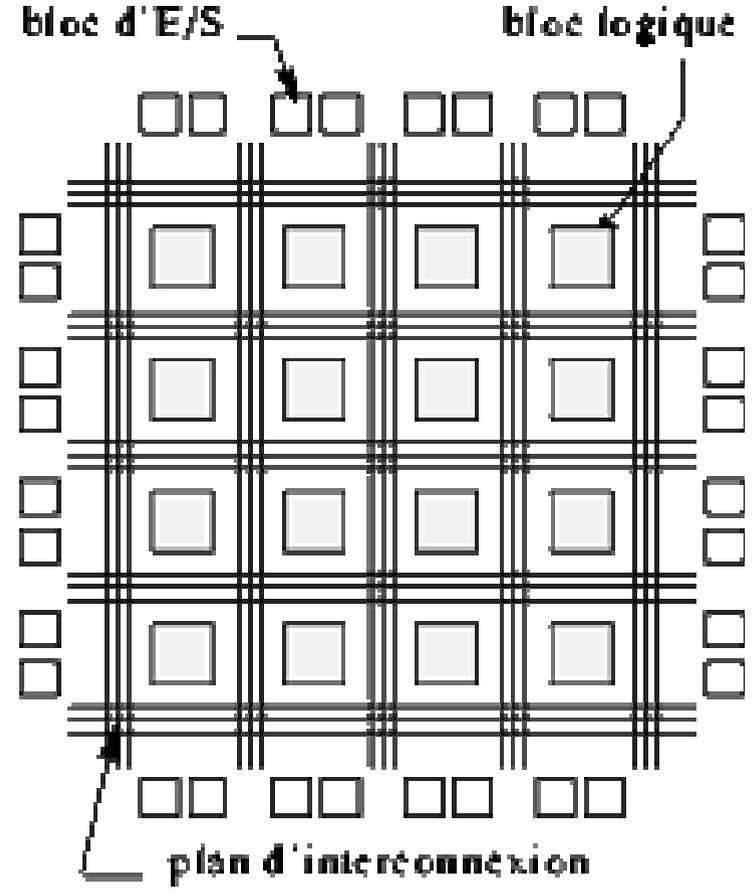
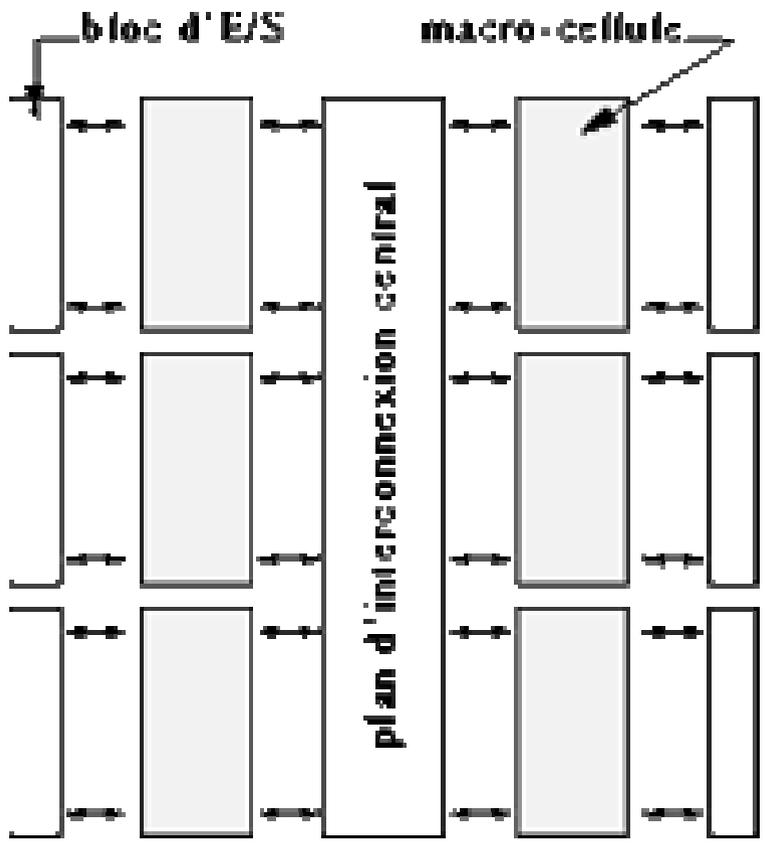


le FPGA est présenté pour la 1er fois par XILINX avec des structure :

- blocs logiques configurables
- blocs d'I/O configurables
- des interconnexions entre bloc configurables

III. Les FPGA

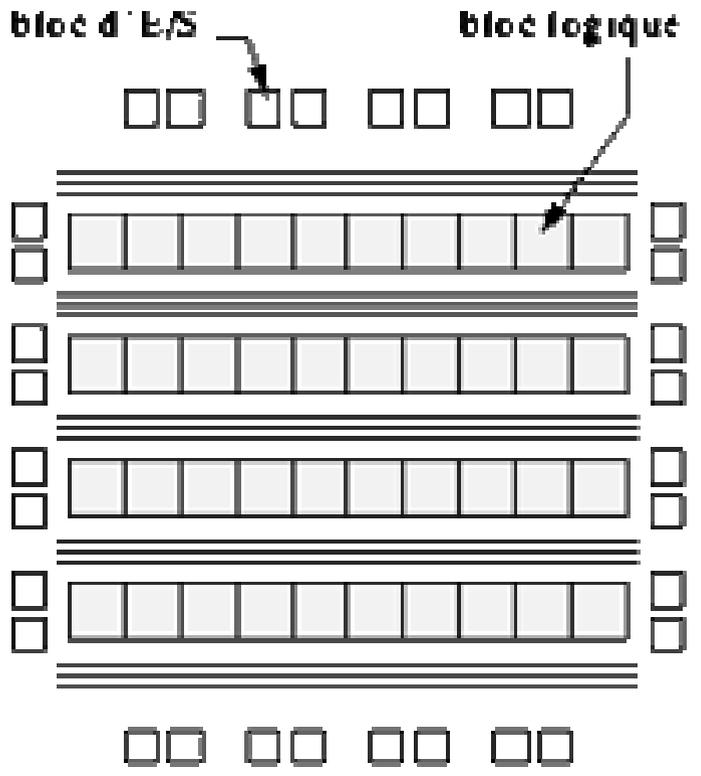
Structure interne des FPGA



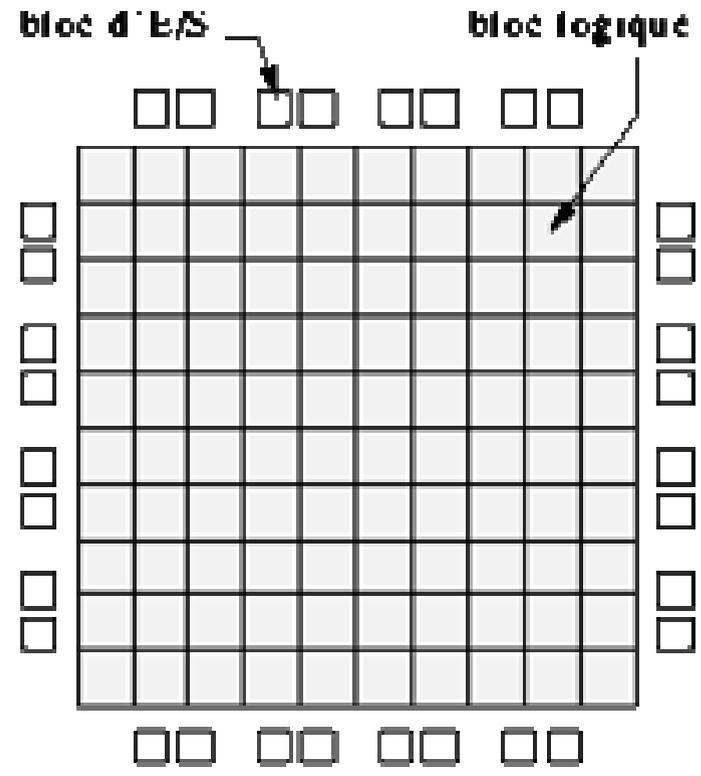
Exemple de structures de FPGA

III. Les FPGA

Structure interne des FPGA



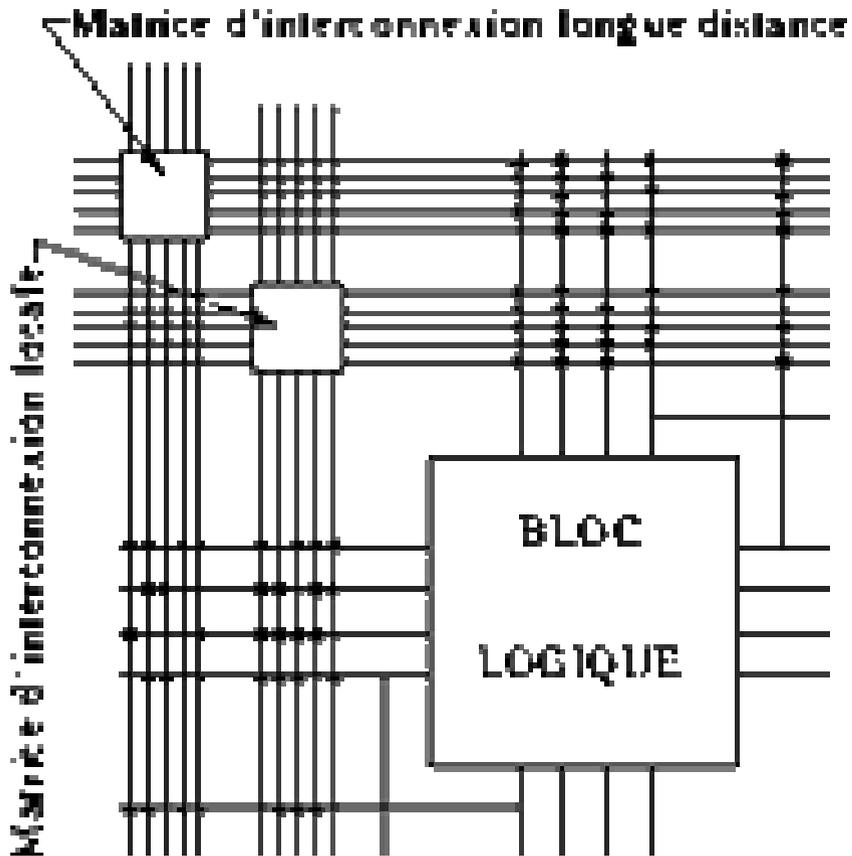
FPGA « Raw-based »



FPGA « Sea of gates »

III. Les FPGA

Les interconnexions programmables dans les FPGA



Dans les FPGA les blocs sont interconnectés de différentes façons :

- des optimisations de routage sont possibles
- sélections de lignes ou colonnes
- canaux de routage rapide (longues distances)

III. Les FPGA

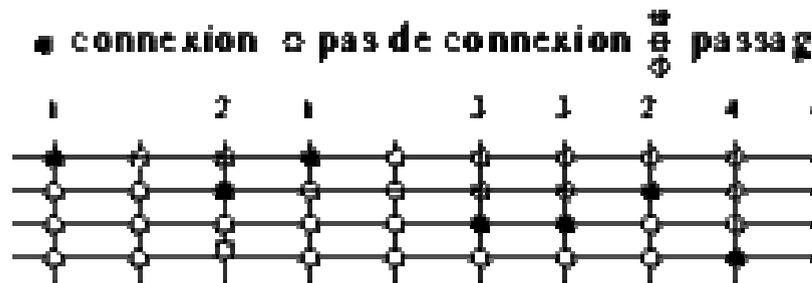
Les interconnexions programmables dans les FPGA

Afin de diminuer le nombre de canaux de routage, ils sont segmentés ou peuvent l'être par programmation. Cette dernière méthode est utilisée dans le cas d'une technologie de programmation par anti-fusibles car les éléments programmables sont de petite taille.

On notera les compromis qu'il existe entre nombre de canaux de routage, nombre d'éléments de programmation et temps de propagation.



(s) Routage sans contraintes (par masques)



(i) Canaux non segmentés

Les figures illustrent les différents types de segmentation pour un même exemple de routage

III. Les FPGA

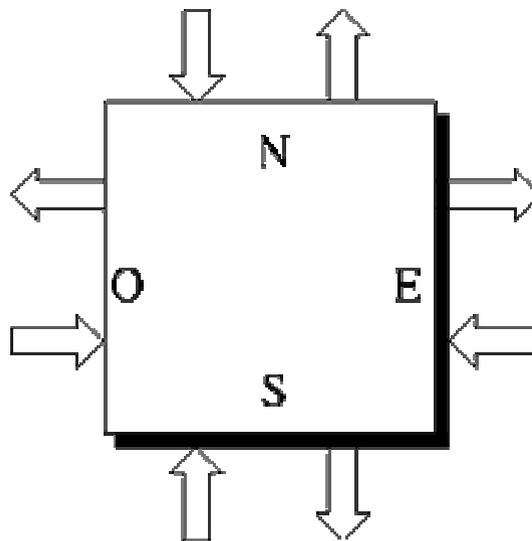
Les types de blocs logiques dans les FPGA

Il existe 4 types de blocs logiques :

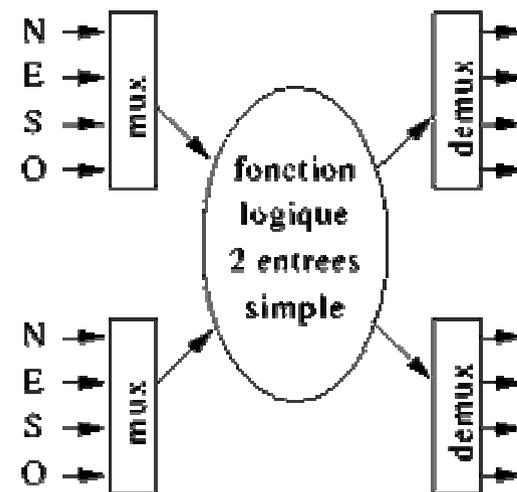
- Les macrocellules.
- Les blocs à multiplexeurs.
- Les LUT.
- Les cellules symétriques.

Cellules symétriques :

Ces blocs de petite taille servent à réaliser des fonctions logiques simples mais aussi comme ressources de connexion



(x) schématique



(y) fonctionnement

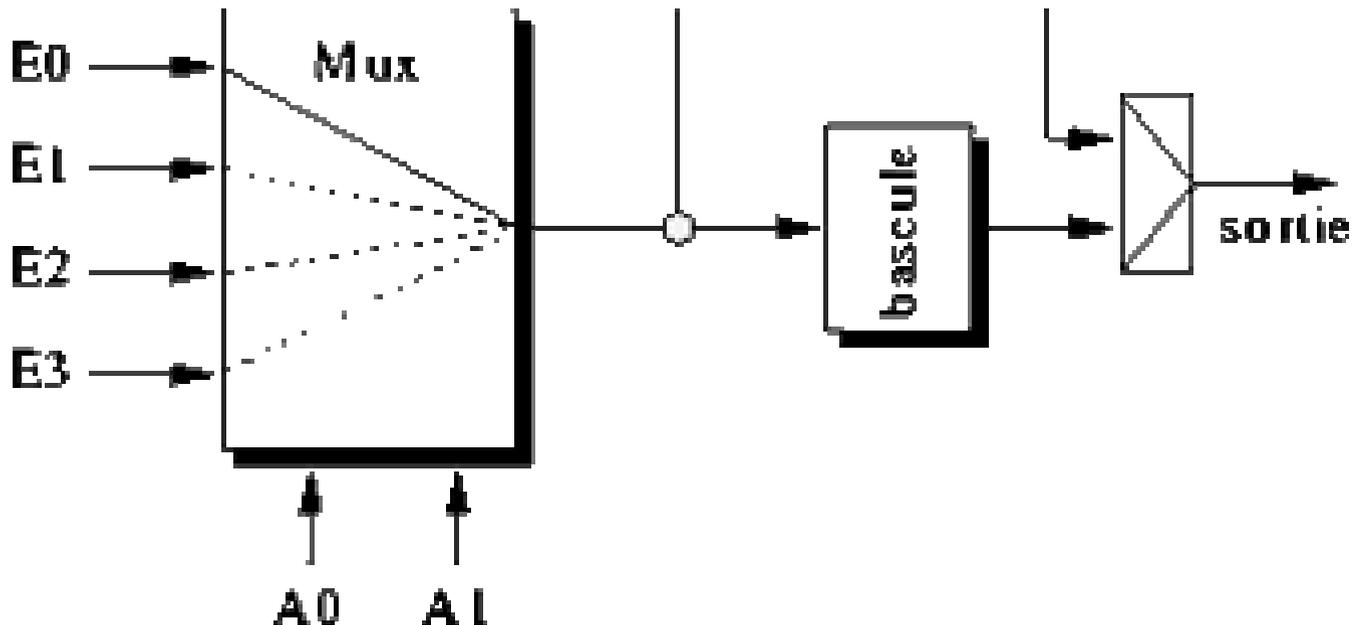
III. Les FPGA

Les types de blocs logiques dans les FPGA

Les Multiplexeurs :

En venant programmer les entrées du multiplexeur et en pilotant les signaux de sélection, il est possible de réaliser toutes les fonctions logiques (à autant d'entrées que de signaux de sélection).

Leur très petite taille est particulièrement adaptée à la technologie anti-fusible.



III. Les FPGA

Les types de blocs logiques dans les FPGA

Les LUT (Look-Up Table) :

Les Look-Up Table sont un cas particulier des cellules à multiplexeur, avec de 2 à 9 entrées et dont la technologie du point mémoire est une technologie SRAM.

Une Look Up Table de N entrées est une mémoire qui peut implémenter n'importe quelle fonction booléenne de N variables.

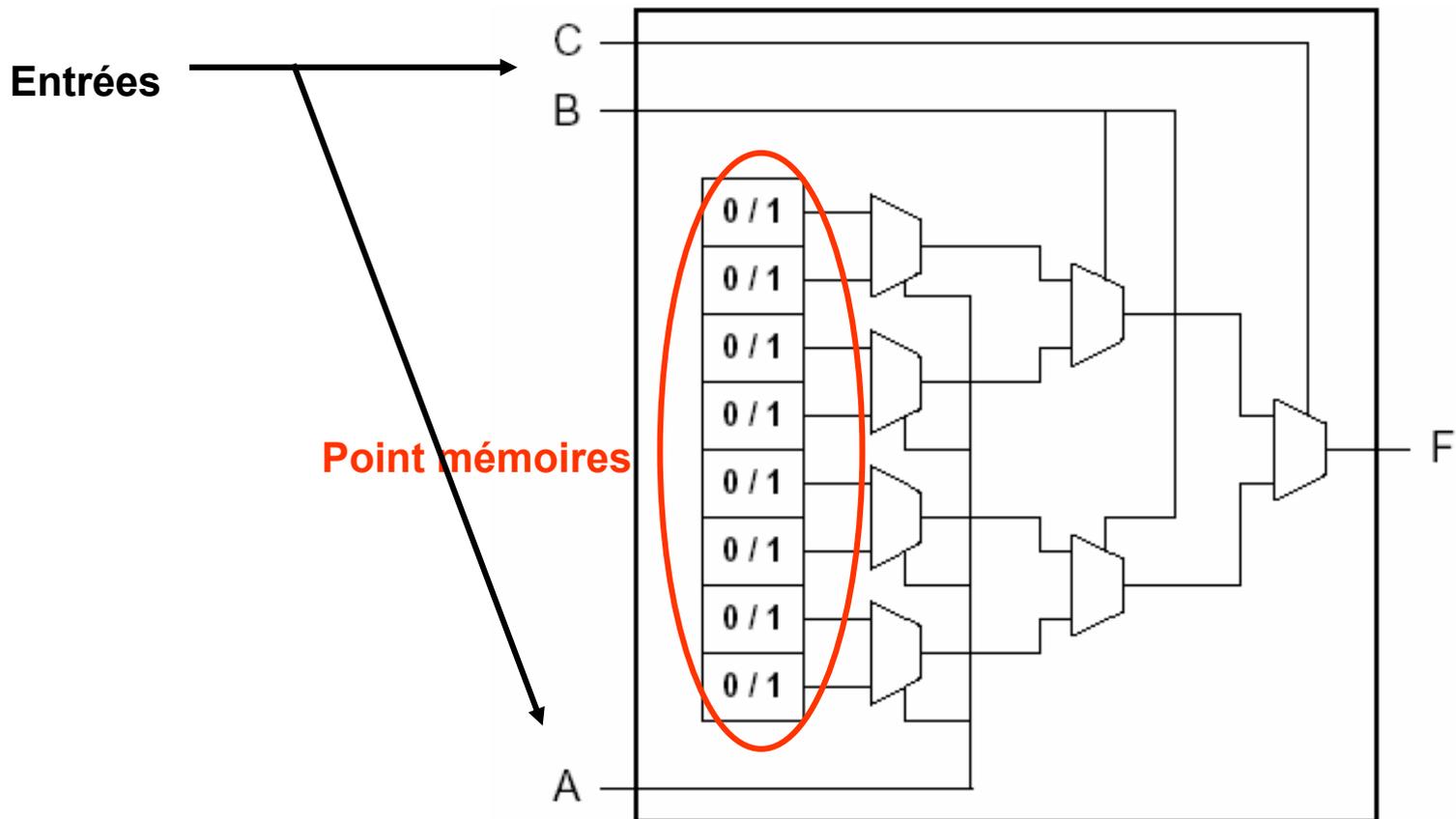
Les N entrées sont utilisées comme adresse d'une mémoire de 2^n bits qui code la fonction booléenne à réaliser. On peut donc réaliser 2^n fonctions différentes avec une LUT à N entrées.

Les Look-Up Tables sont des blocs logiques de très petite granularité dans un CLP. Comme Les Look-Up Table possèdent une bascule de sortie, les architectures à base de Look-Up Table sont beaucoup plus riches en bascules que les architectures à macro-cellules.

III. Les FPGA

Les types de blocs logiques dans les FPGA

Les LUT (Look-Up Table) :



III. Les FPGA

Les types de blocs logiques dans les FPGA

Les Macro-cellules :

On retrouve dans certaines macro-cellules toute la complexité d'un PAL.

Il y a en général un nombre réduit de macro-cellules dans un EPLD, car ces cellules occupent une grande surface.

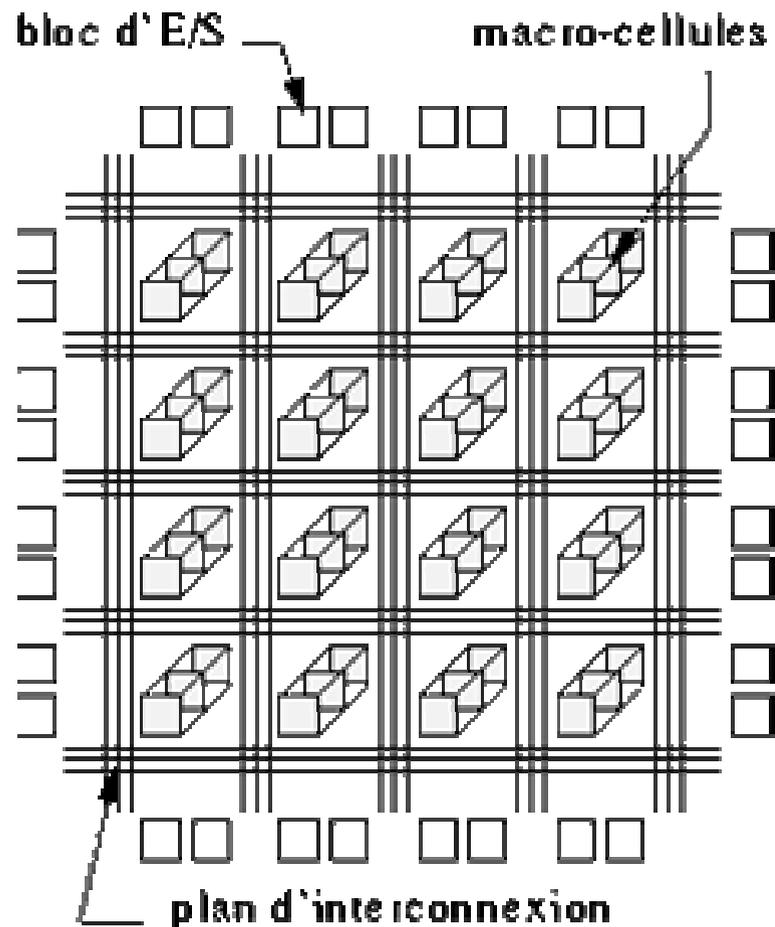
la macro-cellule présente plusieurs intérêts :

- grand nombre de variables d'entrées possibles

possibles

- grand nombre de termes de produits

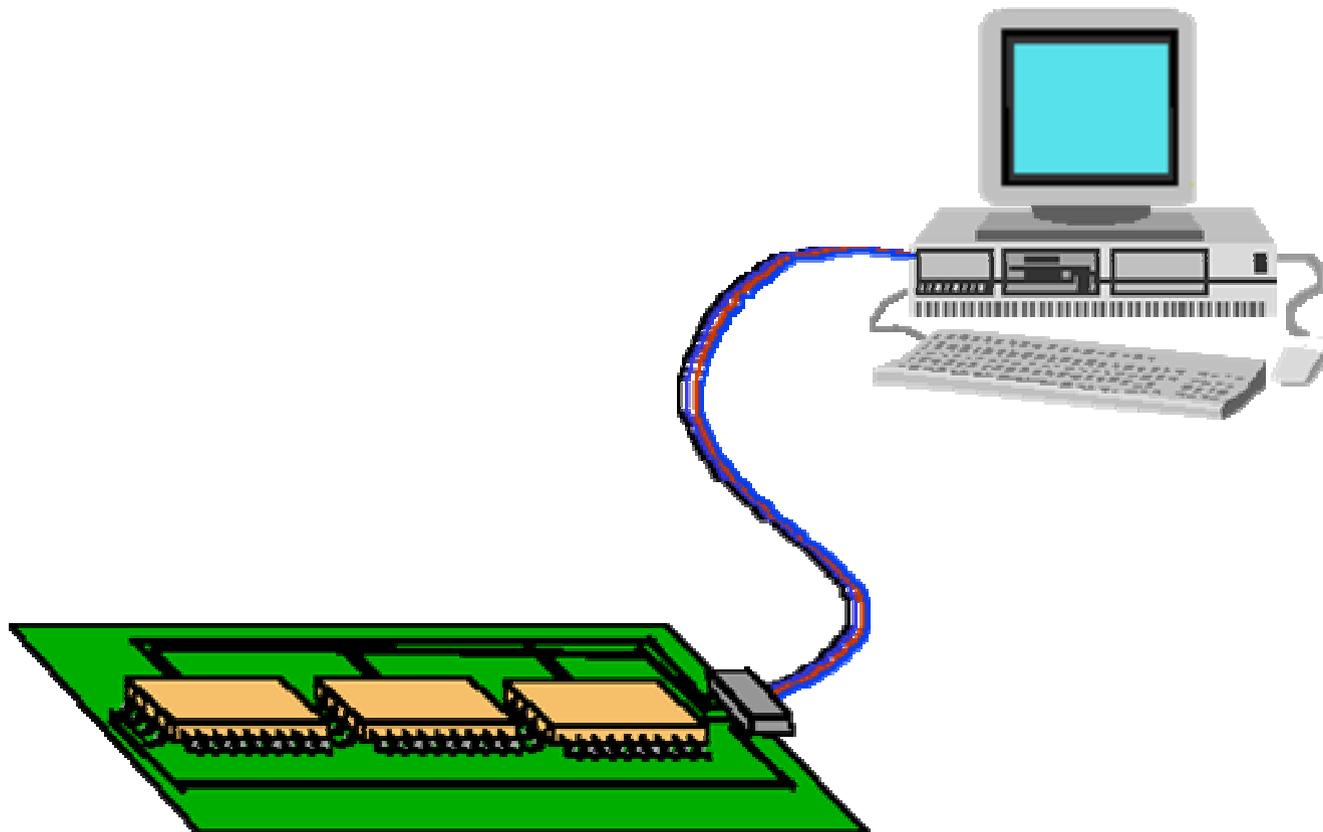
possibles



IV. Les outils de développement

Deux Outils sont nécessaires à la programmation des FPGA :

- Le système de développement
- Le programmeur



IV. Les outils de développement

Les systèmes de développement

Ces systèmes produisent une table représentant les fusibles à détruire en fonction des équations logiques, diagramme d'états et tables de vérités écrit dans le langage propre au système, c'est le rôle du compilateur ou synthétiseur.

La description du fonctionnement des circuits peut se faire de plusieurs façons, soit :

- Par un schéma à base de fonctions logiques élémentaires (Portes ET,OU,NON, ... bascules, compteurs, registres à décalages).
- En utilisant un langage de description comportementale H.D.L. (Hardware Description Language). Les plus anciens sont *PALASM*, *ORCAD/PLD* et le plus connu et utilisé est sans conteste *ABEL* (utilisé par la plus part des systèmes de développements). Enfin les langages dit de haut niveau, VHDL (Very high speed Hardware Description Language) et VERILOG sont en général utilisés pour des circuits complexes. Le langage VHDL est très utilisé en Europe.

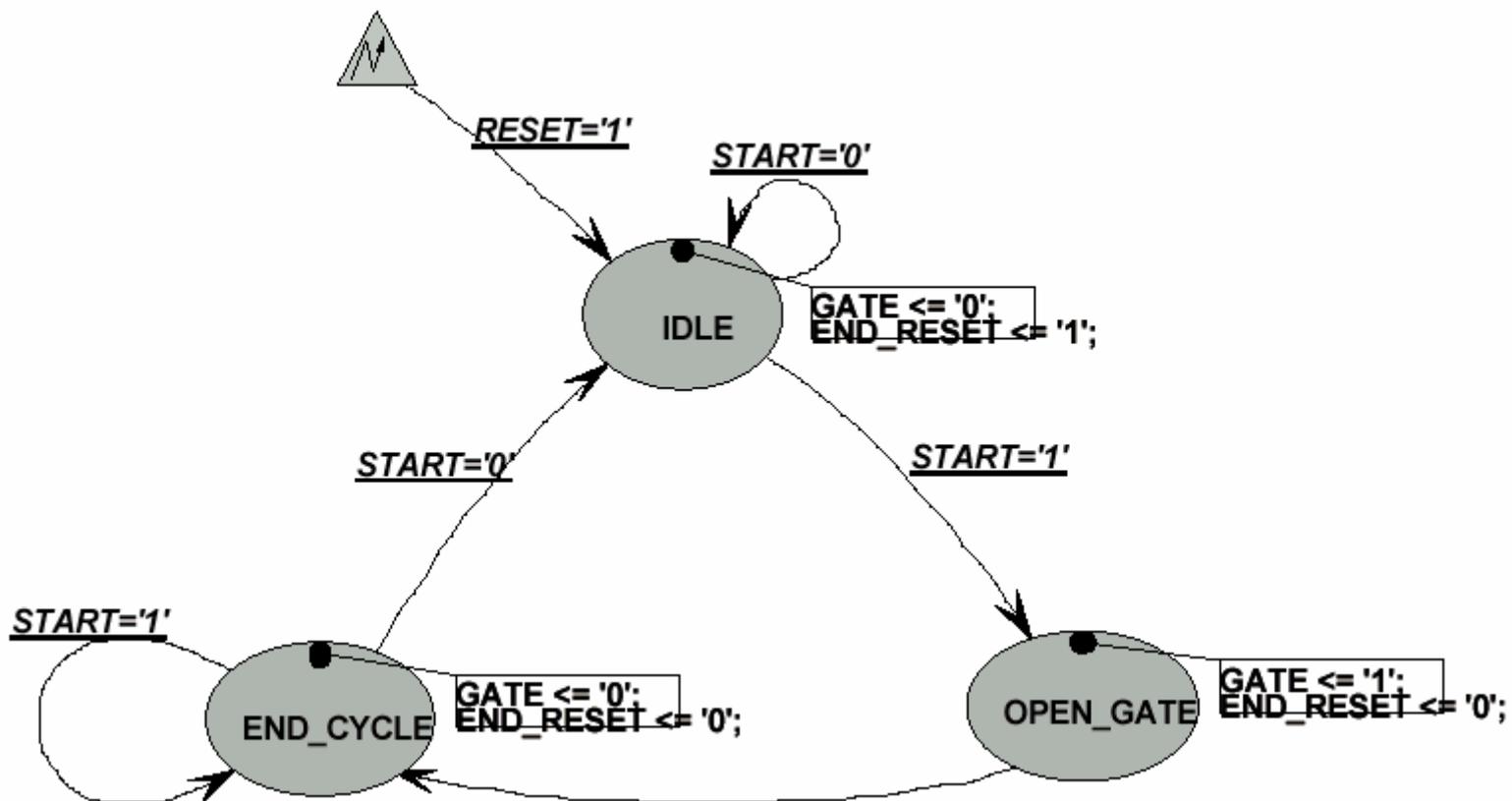


Voir 2ème fichier

IV. Les outils de développement

Les systèmes de développement

- Par l'utilisation de graphes d'états (Flow States Machines, FSM)



V. Le FPGA M7000 de ALTERA

Presentation générale

- **Éléments programmables de type EEPROM, basé sur le principe de seconde génération de matrice programmables.**
- **5.0V, IEEE 1149.1 test standard**
- **600 à 5000 portes**
- **5 ns Pin to pin delay (175MHz)**

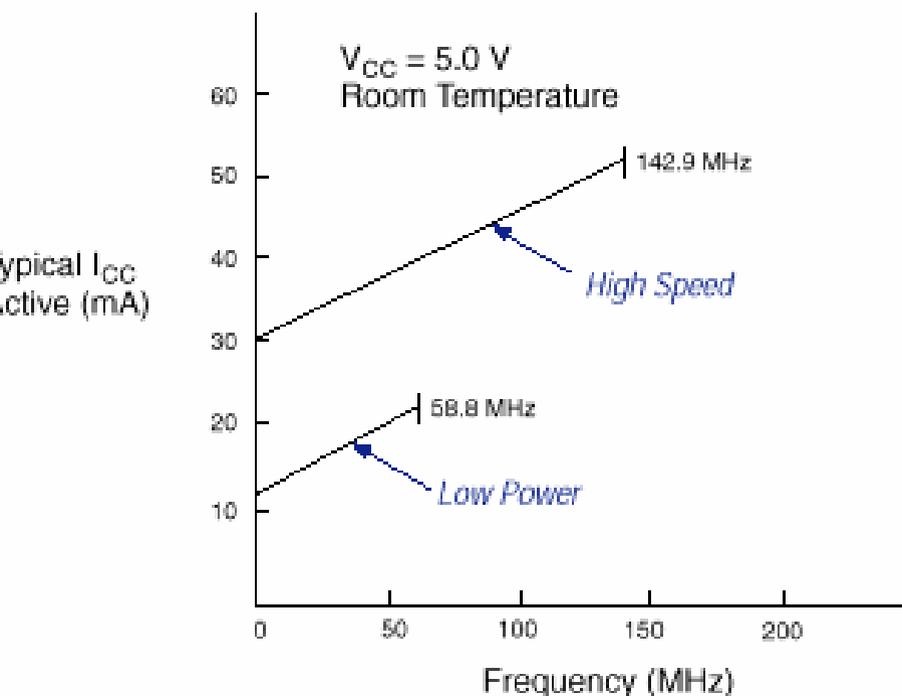
Table 2. MAX 7000S Device Features

Feature	EPM7032S	EPM7064S	EPM7128S	EPM7160S	EPM7192S	EPM7256S
Usable gates	600	1,250	2,500	3,200	3,750	5,000
Macrocells	32	64	128	160	192	256
Logic array blocks	2	4	8	10	12	16
Maximum user I/O pins	36	68	100	104	124	164
t_{PD} (ns)	5	5	6	6	7.5	7.5
t_{SU} (ns)	2.9	2.9	3.4	3.4	4.1	3.9
t_{FSU} (ns)	2.5	2.5	2.5	2.5	3	3
t_{CO1} (ns)	3.2	3.2	4	3.9	4.7	4.7
f_{CNT} (MHz)	175.4	175.4	147.1	149.3	125.0	128.2

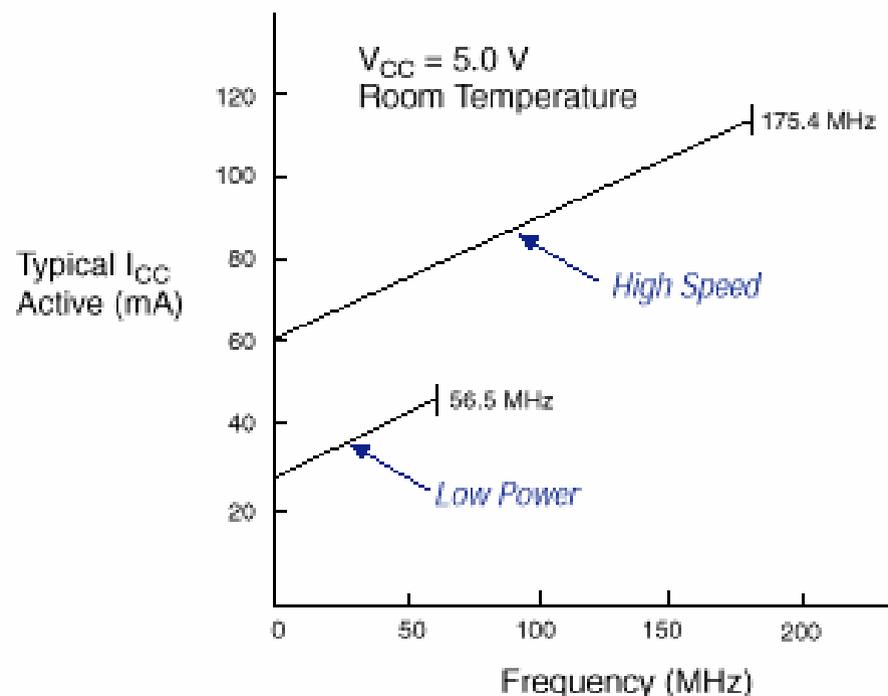
V. Le FPGA M7000 de ALTERA

Performances de certains FPGA parmi les MAX7000

EPM7032S



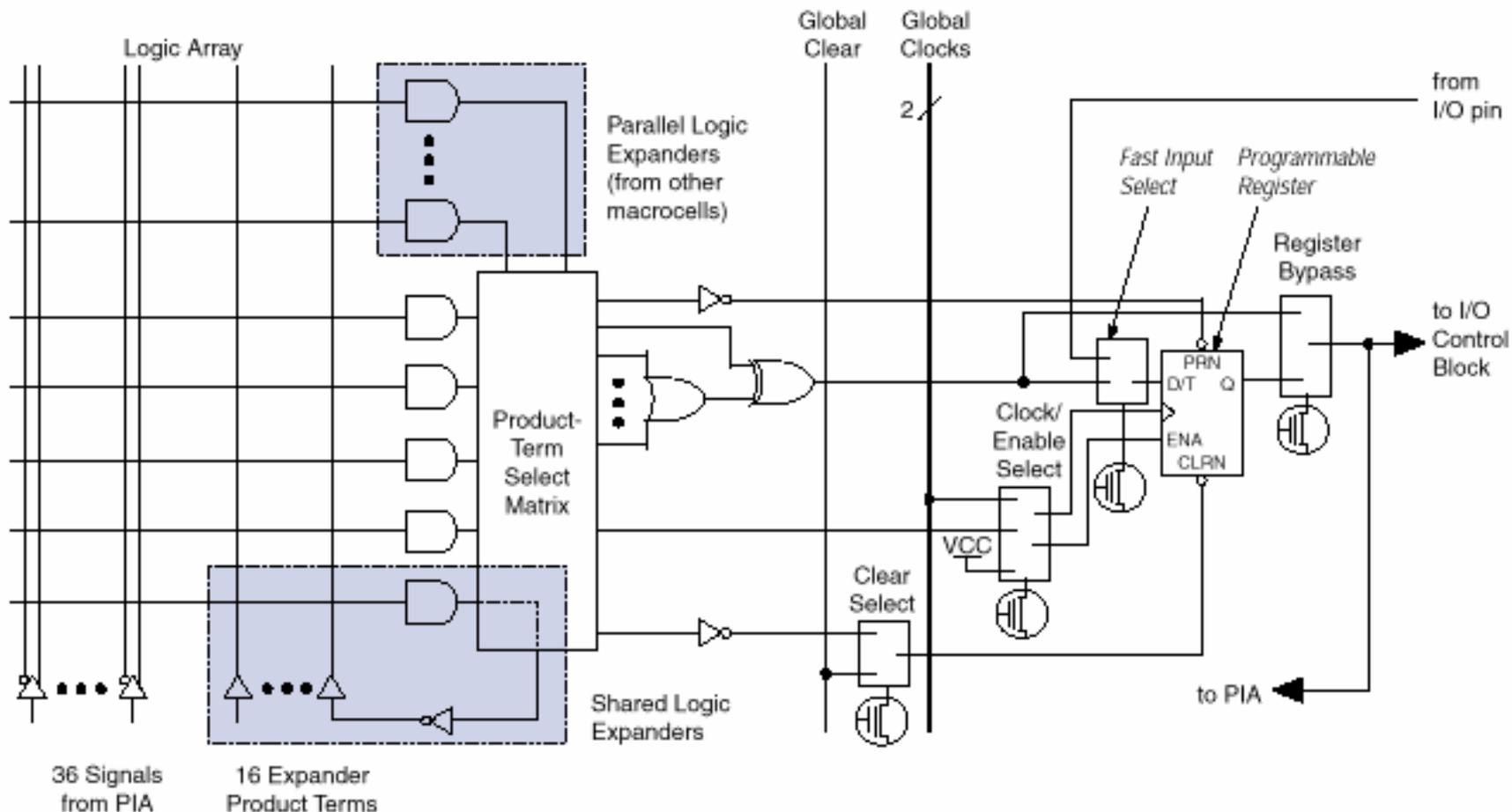
EPM7064S



V. Le FPGA M7000 de ALTERA

Les Macrocells

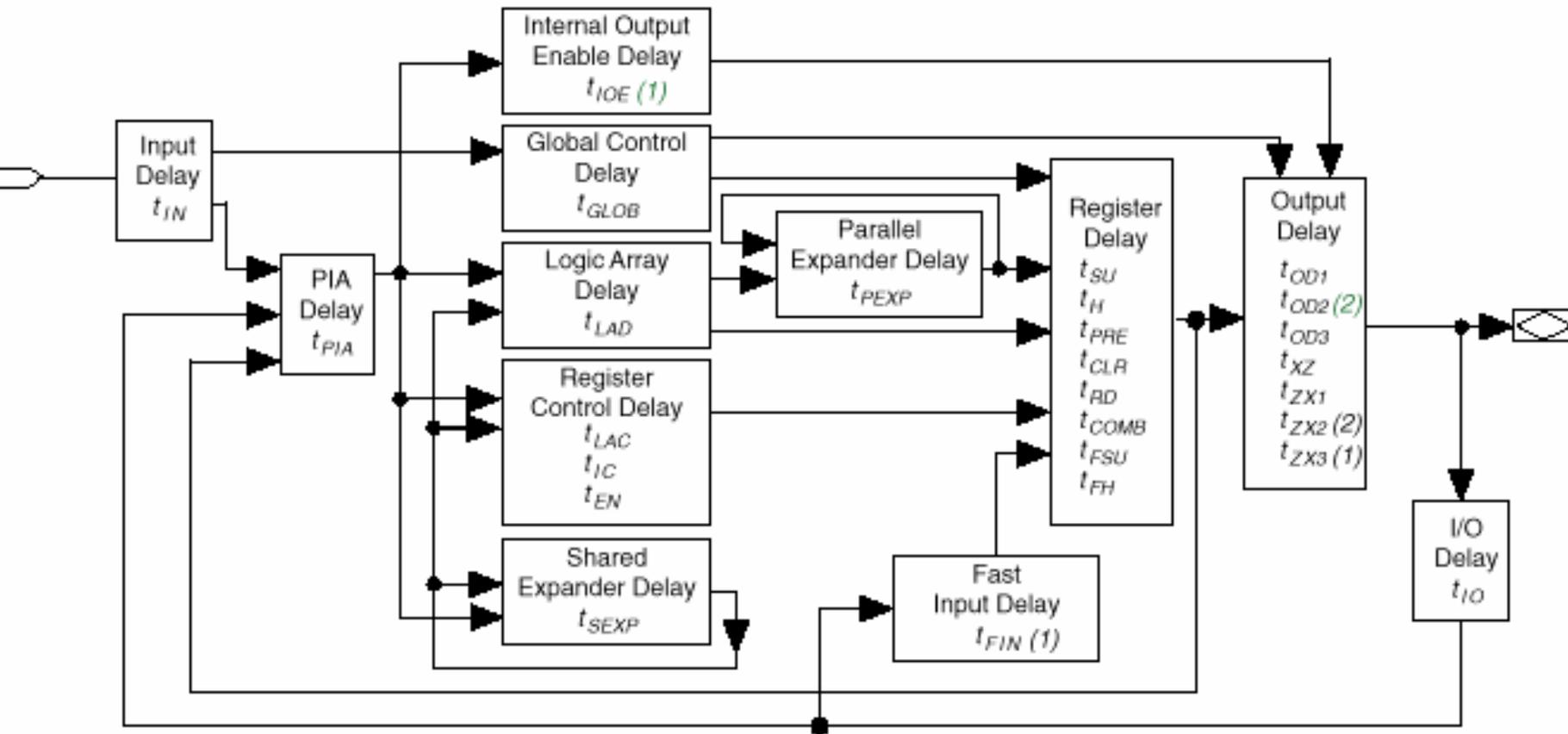
Figure 4. MAX 7000E & MAX 7000S Device Macrocell



V. Le FPGA M7000 de ALTERA

Les contraintes de temps (timing constraints)

Chaque élément intervient et doit être pris en compte dans l'évaluation des temps de propagation





V. Le FPGA M7000 de ALTERA

Les contraintes de temps (timing constrains)

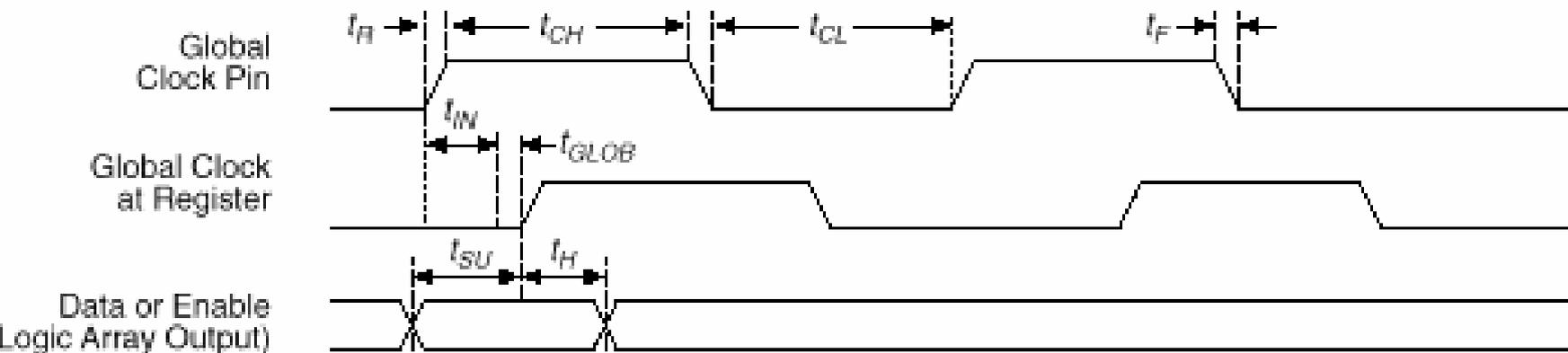
Exemple d'application à un cas simple de logique combinatoire

V. Le FPGA M7000 de ALTERA

Les contraintes de temps (timing constraints)

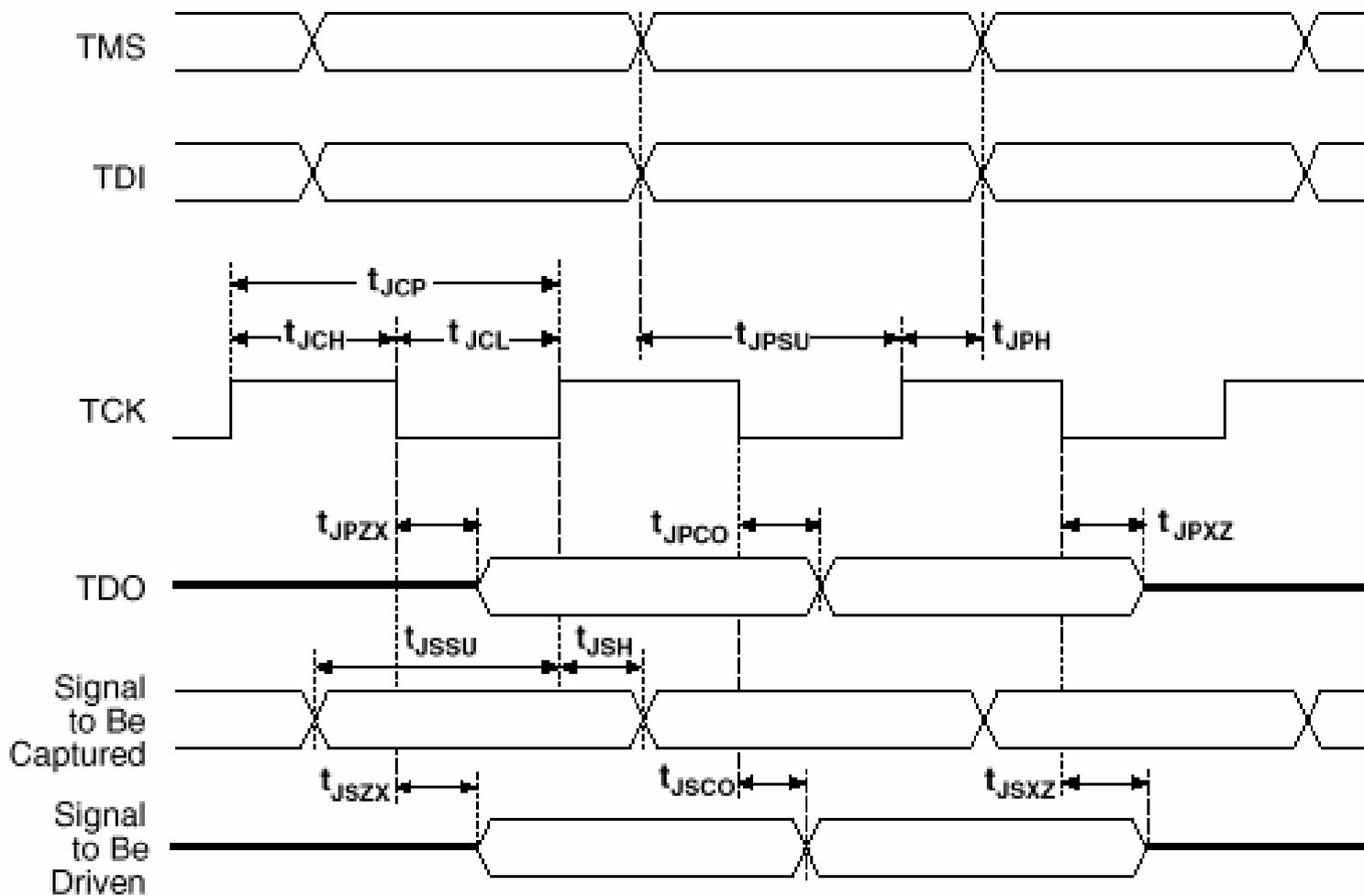
Exemple d'application au cas séquentiels

Global Clock Mode

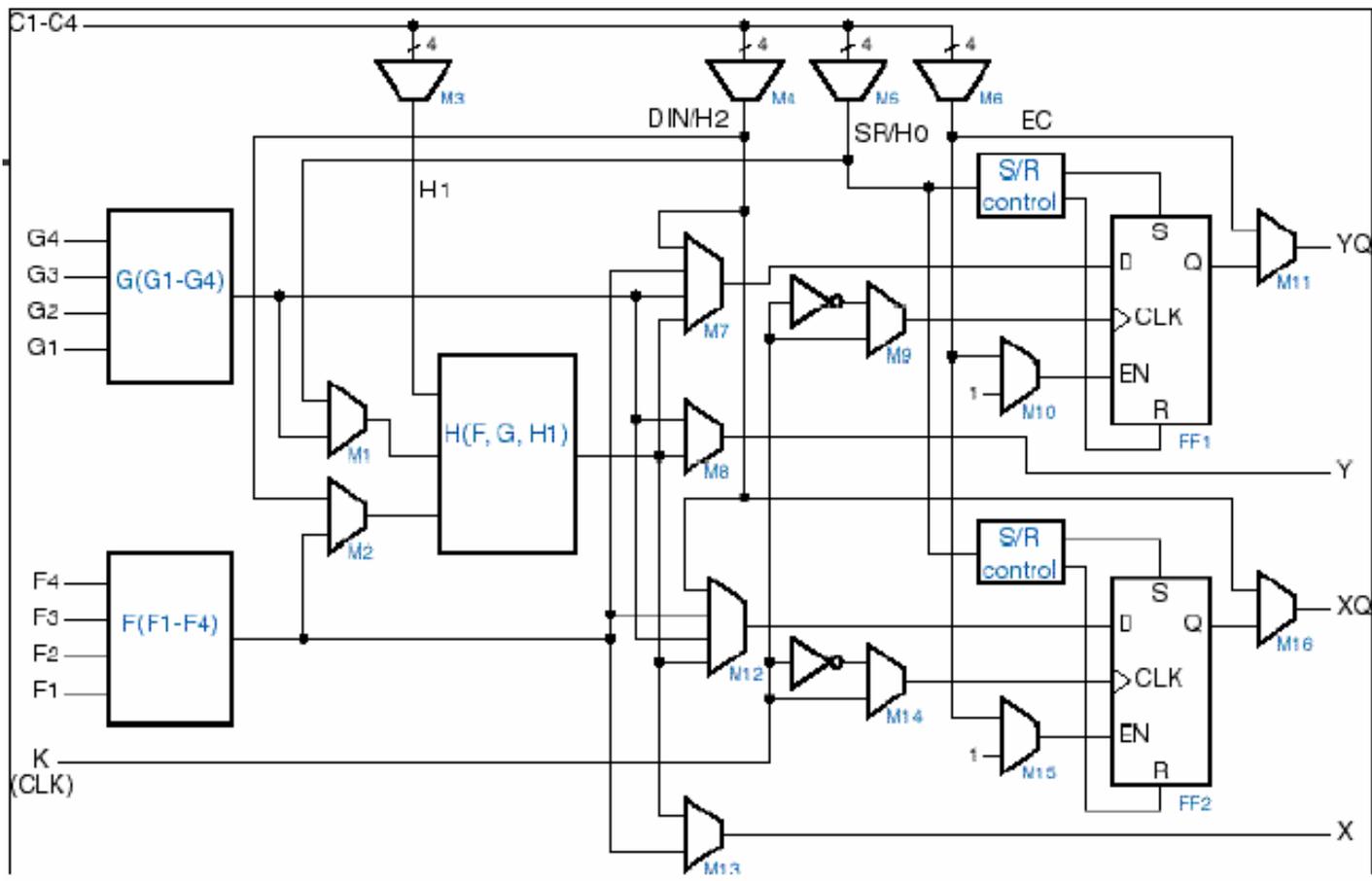


V. Le FPGA M7000 de ALTERA

Le format de test JTAG : IEEE 1149.1 (the boundary scan standard)

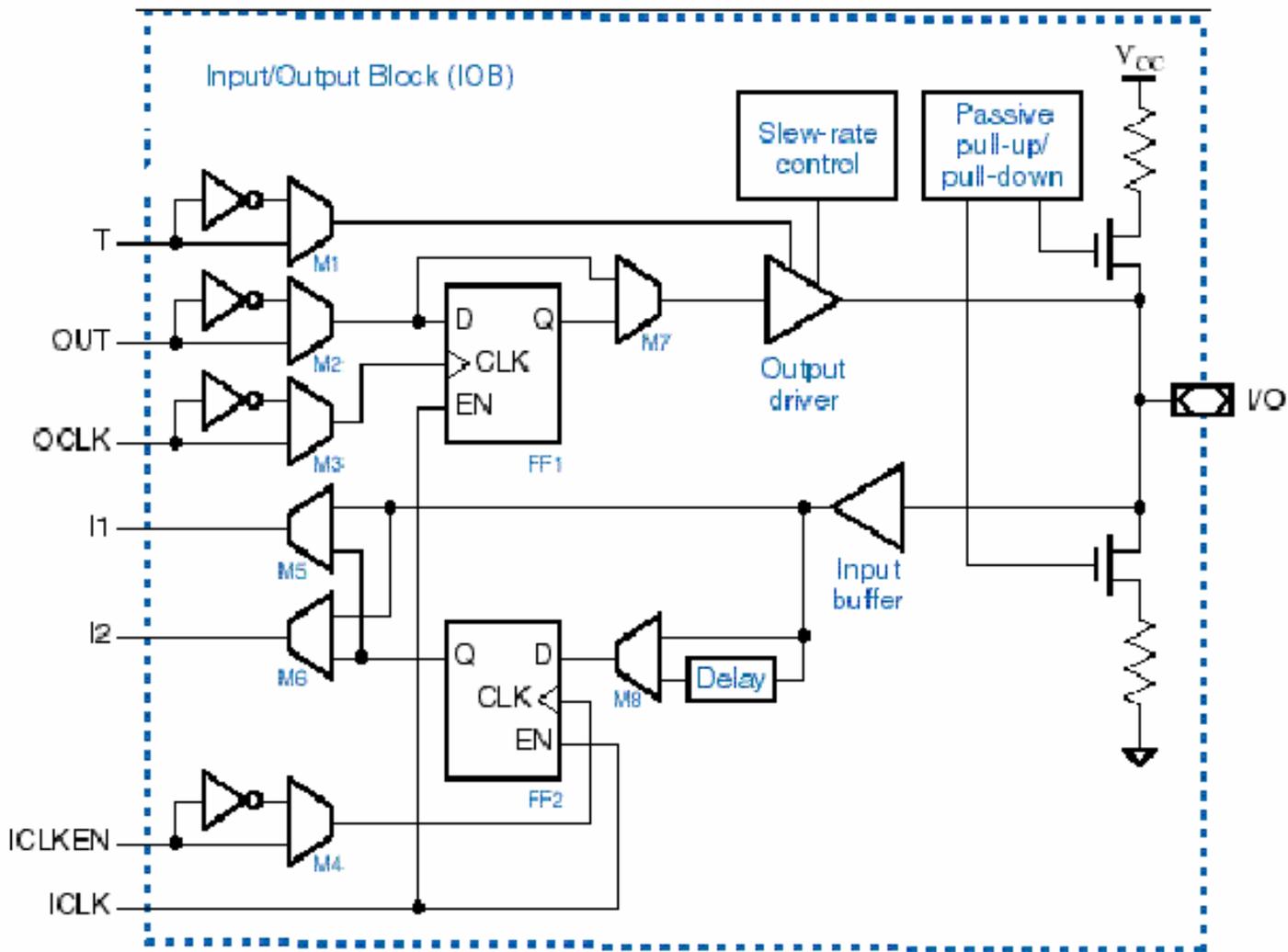


V. Le FPGA CX4000 de Xilinx



XC4000 Configurable Logic Block
Fig 10-44 w.p. 884

V. Le FPGA CX4000 de Xilinx



V. Le FPGA CX4000 de Xilinx

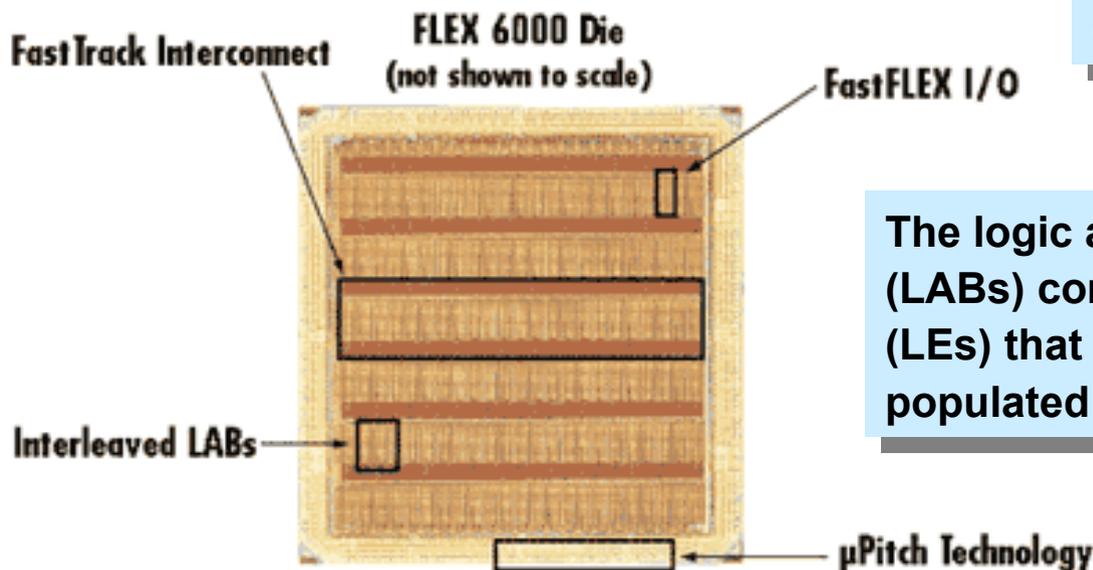
Device	Logic Cells	Max logic gates (no RAM)	Max RAM bits (no gates)	Typical gate range (Logic and RAM)	Matrix	Total CLBs	Number of Flip-Flops	Max user I/O
XC4002XL	152	1,600	2,048	1,000 - 3,000	8 x 8	64	256	64
XC4003E	238	3,000	3,200	2,000 - 5,000	10 x 10	100	360	80
XC4005E/XL	466	5,000	6,272	3,000 - 9,000	14 x 14	196	616	112
XC4006E	608	6,000	8,192	4,000 - 12,000	16 x 16	256	768	128
XC4008E	770	8,000	10,368	6,000 - 15,000	18 x 18	324	936	144
XC4010E/XL	950	10,000	12,800	7,000 - 20,000	20 x 20	400	1,120	160
XC4013E/XL	1368	13,000	18,432	10,000 - 30,000	24 x 24	576	1,536	192
XC4020E/XL	1862	20,000	25,088	13,000 - 40,000	28 x 28	784	2,016	224
XC4025E	2432	25,000	32,768	15,000 - 45,000	32 x 32	1,024	2,560	256
XC4028EX/XL	2432	28,000	32,768	18,000 - 50,000	32 x 32	1,024	2,560	256
XC4036EX/XL	3078	36,000	41,472	22,000 - 65,000	36 x 36	1,296	3,168	288
XC4044XL	3800	44,000	51,200	27,000 - 80,000	40 x 40	1,600	3,840	320
XC4052XL	4598	52,000	61,952	33,000 - 100,000	44 x 44	1,936	4,576	352
XC4062XL	5472	62,000	73,728	40,000 - 130,000	48 x 48	2,304	5,376	384
XC4085XL	7448	85,000	100,352	55,000 - 180,000	56 x 56	3,136	7,168	448

V. Le FPGA FLEX6000 de ALTERA

Informations générales

Informations générales

- Nb de portes : 5000 - 24000



The logic array contains logic array blocks (LABs) composed of 10 logic elements (LEs) that communicate through a fully populated local interconnect structure

Table 1. FLEX 6000 Device Features

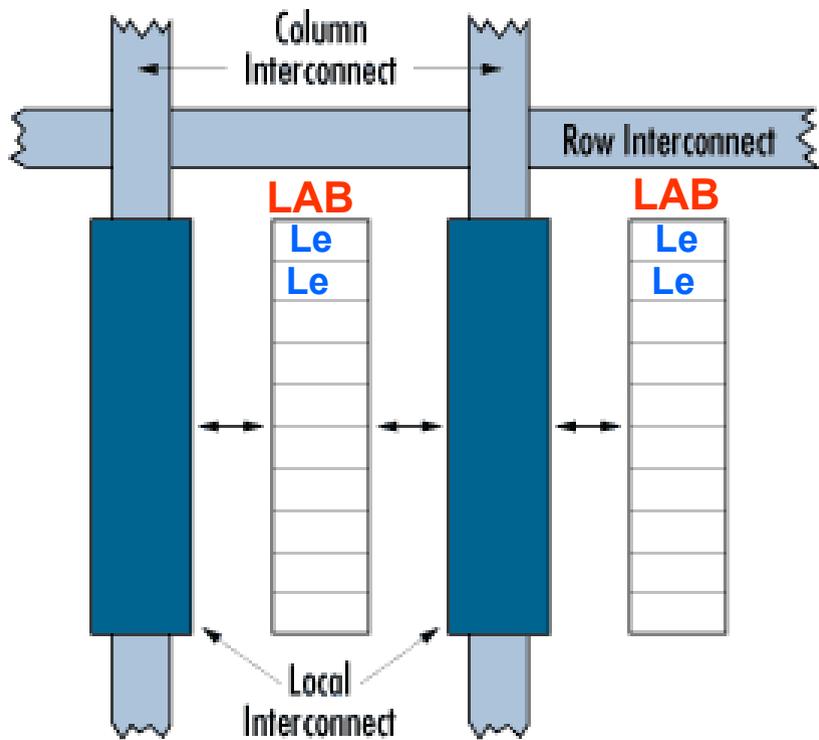
Feature	EPF6010A	EPF6016	EPF6016A	EPF6024A
Typical gates (1)	10,000	16,000	16,000	24,000
Logic elements (LEs)	880	1,320	1,320	1,960
Maximum I/O pins	102	204	171	218
Supply voltage (V_{CCINT})	3.3 V	5.0 V	3.3 V	3.3 V

Note:

(1) The embedded IEEE Std. 1149.1 JTAG circuitry adds up to 14,000 gates in addition to the listed typical gates.

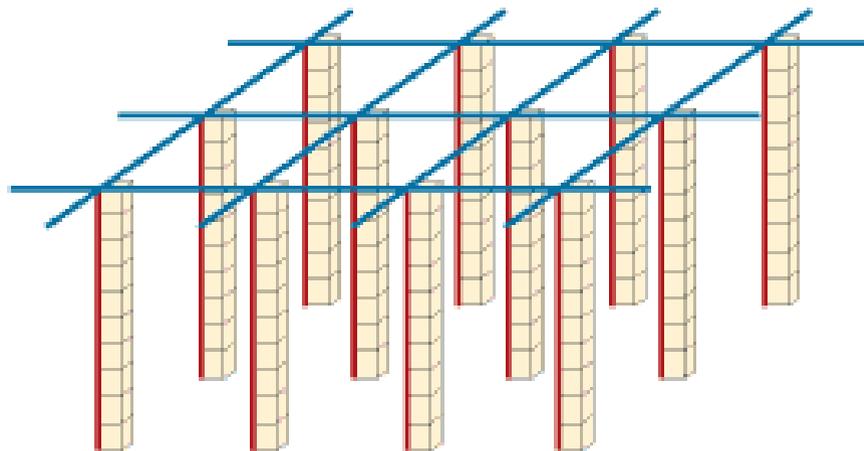
V. Le FPGA FLEX6000 de ALTERA

Informations générales



Interleaved LABs

Chaque Le d'un même LAB peuvent être interconnectés ainsi qu'à leurs plus proches voisins



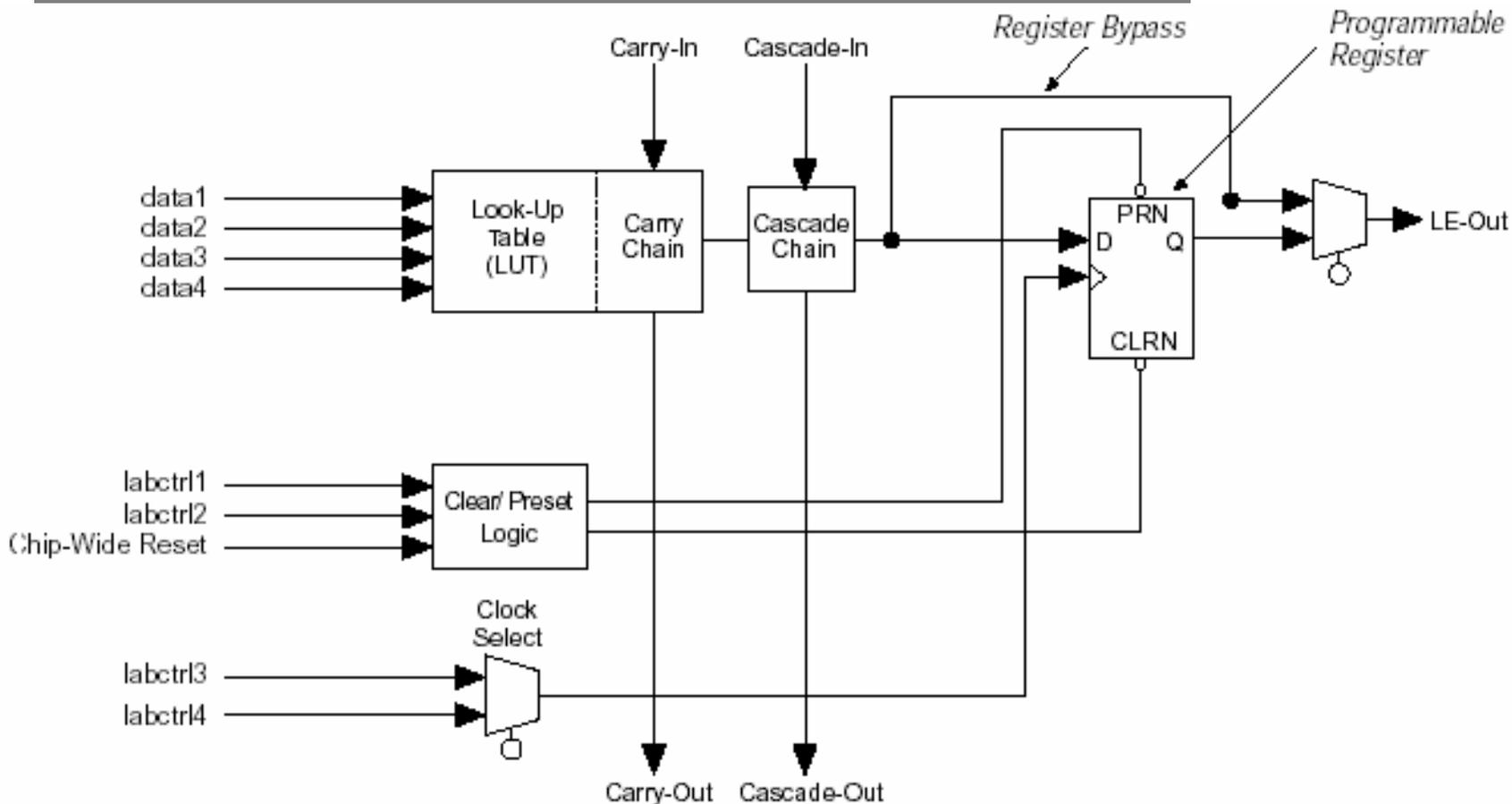
FastTrack Interconnect

Les LAB peuvent être interconnectés grâce à un réseau d'interconnexions longue distance

V. Le FPGA FLEX6000 de ALTERA

Informations générales : que sont les LE (logic Element)?

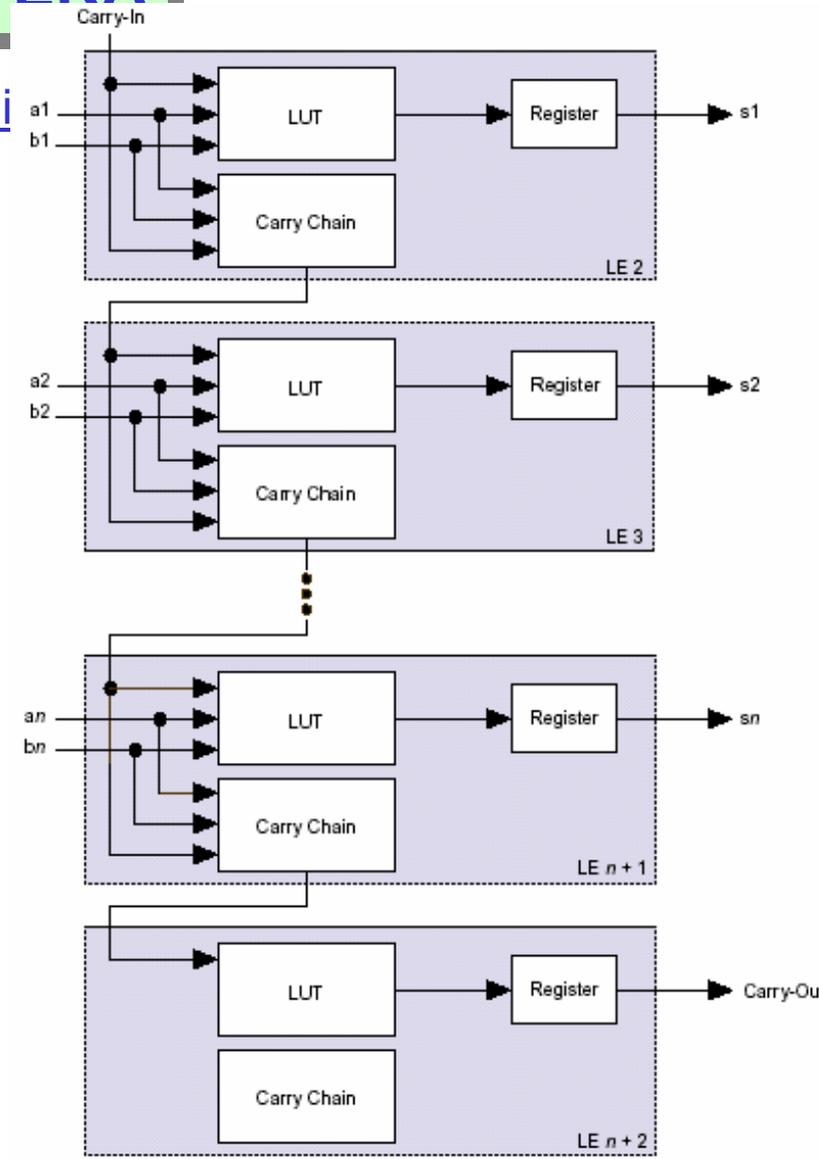
Chaque LE (Logic Element) contient 1 LUT a 4 entrees



V. Le FPGA FLEX6000 de ALTERA

Informations générales : que sont les LE (logi

Le Carry Chain

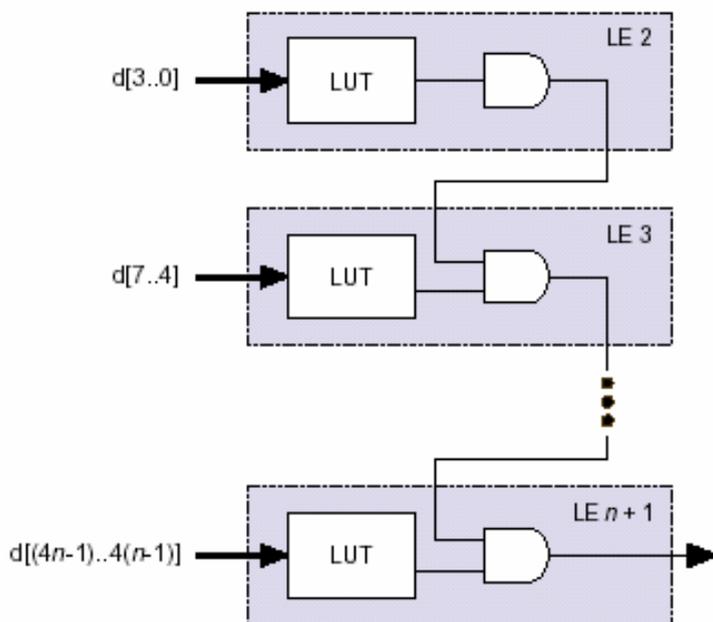


V. Le FPGA FLEX6000 de ALTERA

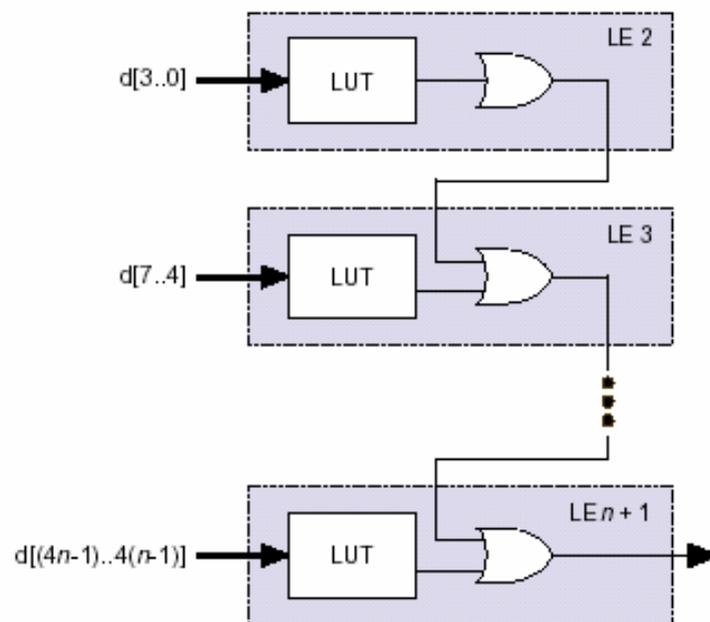
Informations générales : que sont les LE (logic Element)?

Le Cascade chain

AND Cascade Chain



OR Cascade Chain



V. Le FPGA FLEX6000 de ALTERA

Informations générales : performances

Table 3. FLEX 6000 Device Performance for Common Designs

Application	LEs Used	Performance			Units
		-1 Speed Grade	-2 Speed Grade	-3 Speed Grade	
16-bit loadable counter	16	172	153	133	MHz
16-bit accumulator	16	172	153	133	MHz
24-bit accumulator	24	136	123	108	MHz
16-to-1 multiplexer (pin-to-pin) (1)	10	12.1	13.4	16.6	ns
16 × 16 multiplier with a 4-stage pipeline	592	84	67	58	MHz

Table 4. FLEX 6000 Device Performance for Complex Designs *Note (1)*

Application	LEs Used	Performance			Units
		-1 Speed Grade	-2 Speed Grade	-3 Speed Grade	
8-bit, 16-tap parallel finite impulse response (FIR) filter	599	94	80	72	MSPS
8-bit, 512-point fast Fourier transform (FFT) function	1,182	75	89	109	μS
		63	53	43	MHz
a16450 universal asynchronous receiver/transmitter (UART)	487	36	30	25	MHz
PCI bus target with zero wait states	609	56	49	42	MHz

Note:

(1) The applications in this table were created using Altera MegaCore™ functions.

V. Le FPGA FLEX10K de ALTERA

Informations générales

Premier circuit programmable dit:

embedded programmable logic device (PLD) family,
providing System-on-a-Programmable-Chip (SOPC)

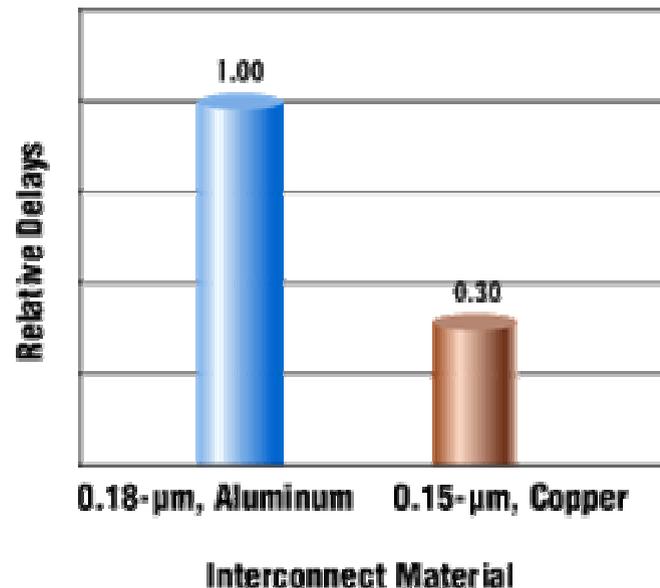
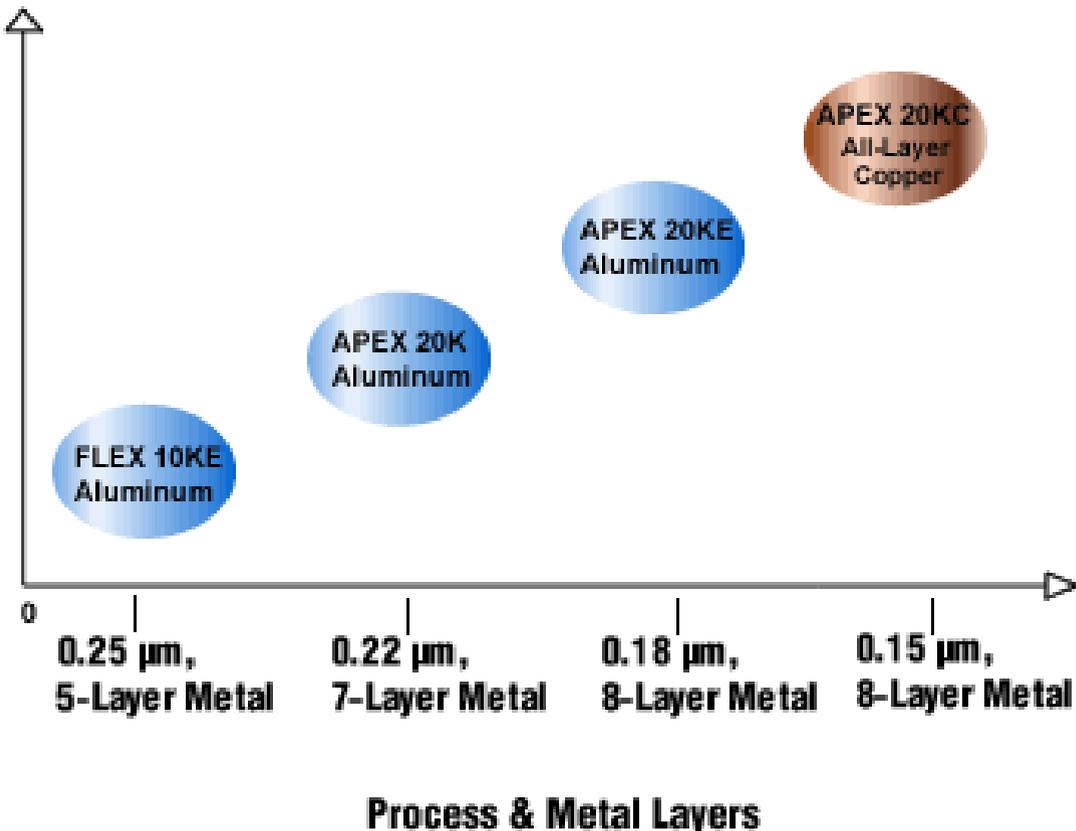
De 10000 à 250000 portes

Feature	EPF10K10 EPF10K10A	EPF10K20	EPF10K30 EPF10K30A	EPF10K40	EPF10K50 EPF10K50V
Typical gates (logic and RAM) (1)	10,000	20,000	30,000	40,000	50,000
Maximum system gates	31,000	63,000	69,000	93,000	116,000
Logic elements (LEs)	576	1,152	1,728	2,304	2,880
Logic array blocks (LABs)	72	144	216	288	360
Embedded array blocks (EABs)	3	6	6	8	10
Total RAM bits	6,144	12,288	12,288	16,384	20,480
Maximum user I/O pins	150	189	246	189	310

V. Le FPGA APEX de ALTERA

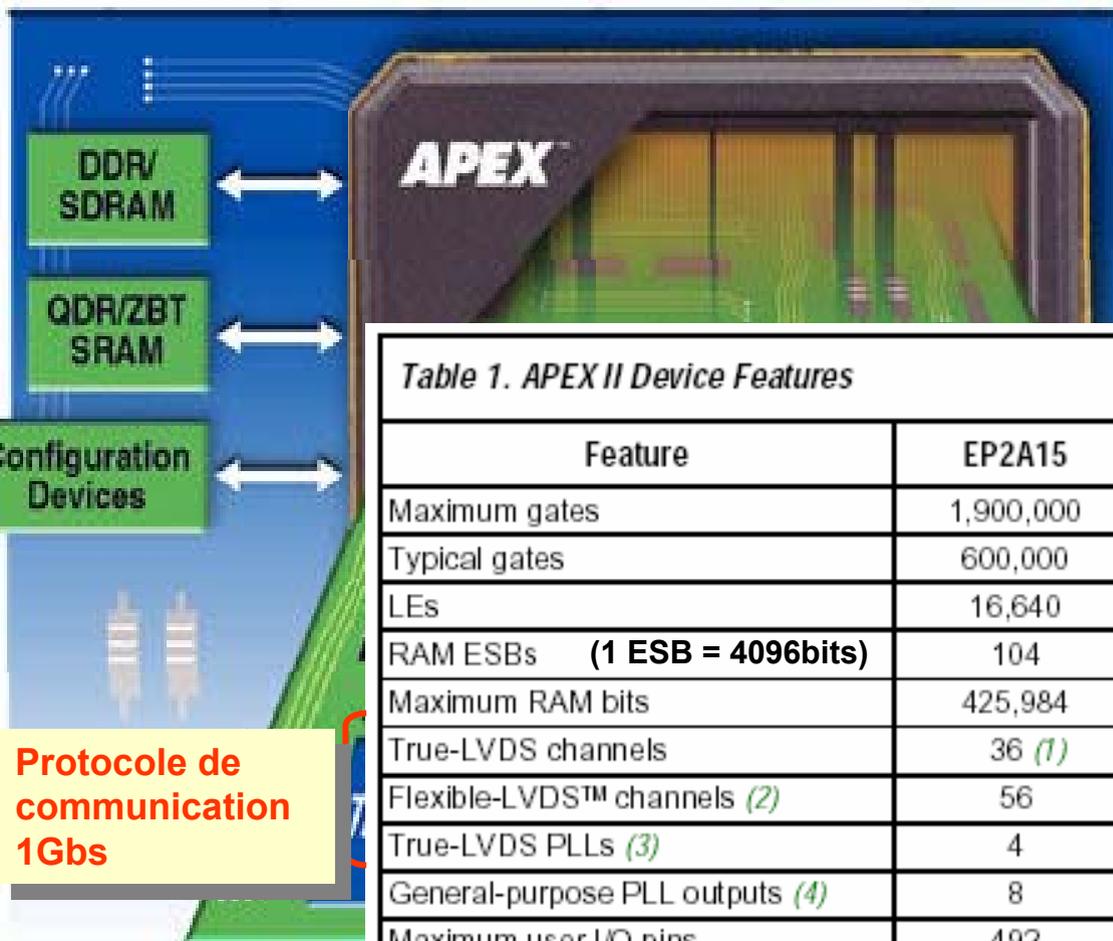
Informations générales

The APEX™ device family ranges from 30,000 to over 1.5 million gates (113,000 to over 2.5 million system gates) and ships on 0.22- μm , 0.18- μm , and 0.15- μm processes. Introduced in 1999 with all copper interconnect layers



V. Le FPGA APEX de ALTERA

Informations générales



**Introducing
APEX 20KC
Devices!**

Table 1. APEX II Device Features

Feature	EP2A15	EP2A25	EP2A40	EP2A70
Maximum gates	1,900,000	2,750,000	3,000,000	5,250,000
Typical gates	600,000	900,000	1,500,000	3,000,000
LEs	16,640	24,320	38,400	67,200
RAM ESBs (1 ESB = 4096bits)	104	152	160	280
Maximum RAM bits	425,984	622,592	655,360	1,146,880
True-LVDS channels	36 (1)	36 (1)	36 (1)	36 (1)
Flexible-LVDS™ channels (2)	56	56	88	88
True-LVDS PLLs (3)	4	4	4	4
General-purpose PLL outputs (4)	8	8	8	8
Maximum user I/O pins	492	612	735	1,060

V. Le FPGA APEX de ALTERA

Informations générales

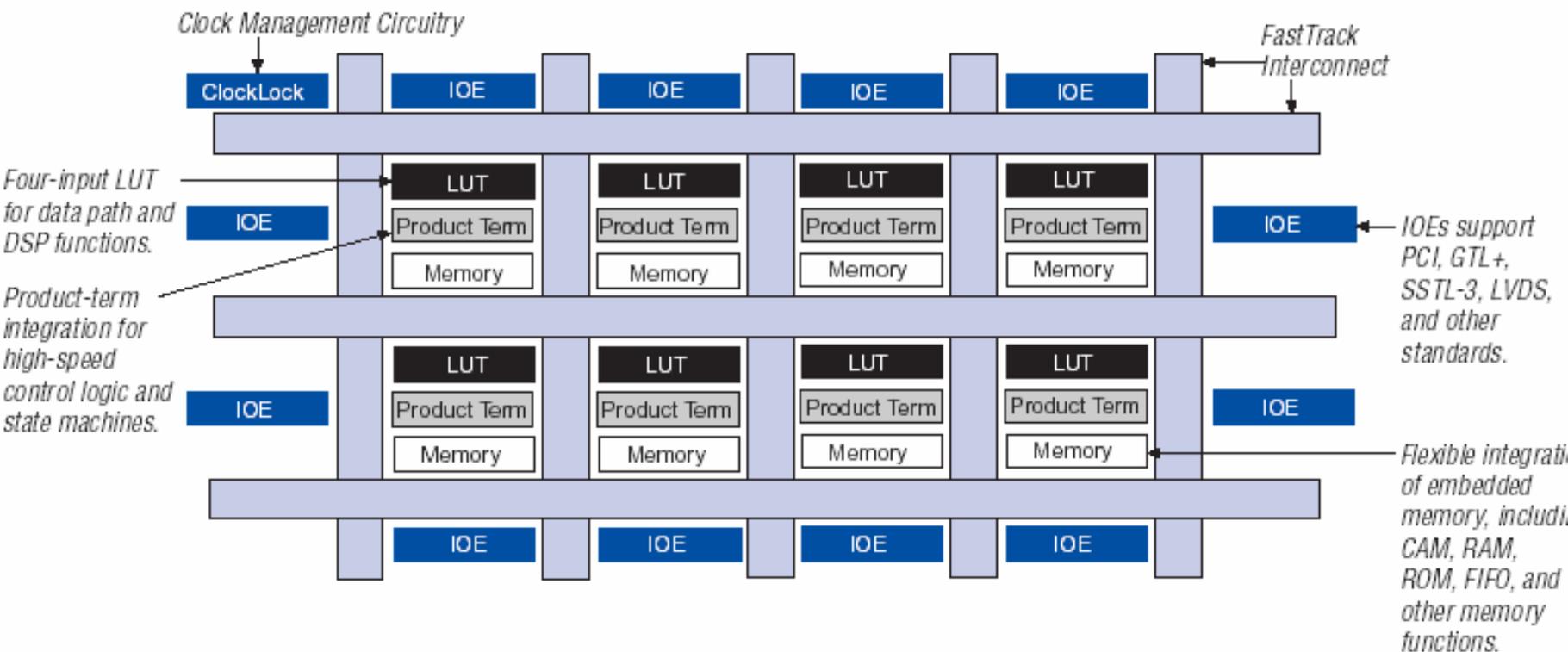
Package informations

Feature	672-Pin FineLine BGA	724-Pin BGA	1,020-Pin FineLine BGA	1,508-Pin FineLine BGA
EP2A15	492	492		
EP2A25	492	536		
EP2A40	492	536	735	
EP2A70		536		1,060

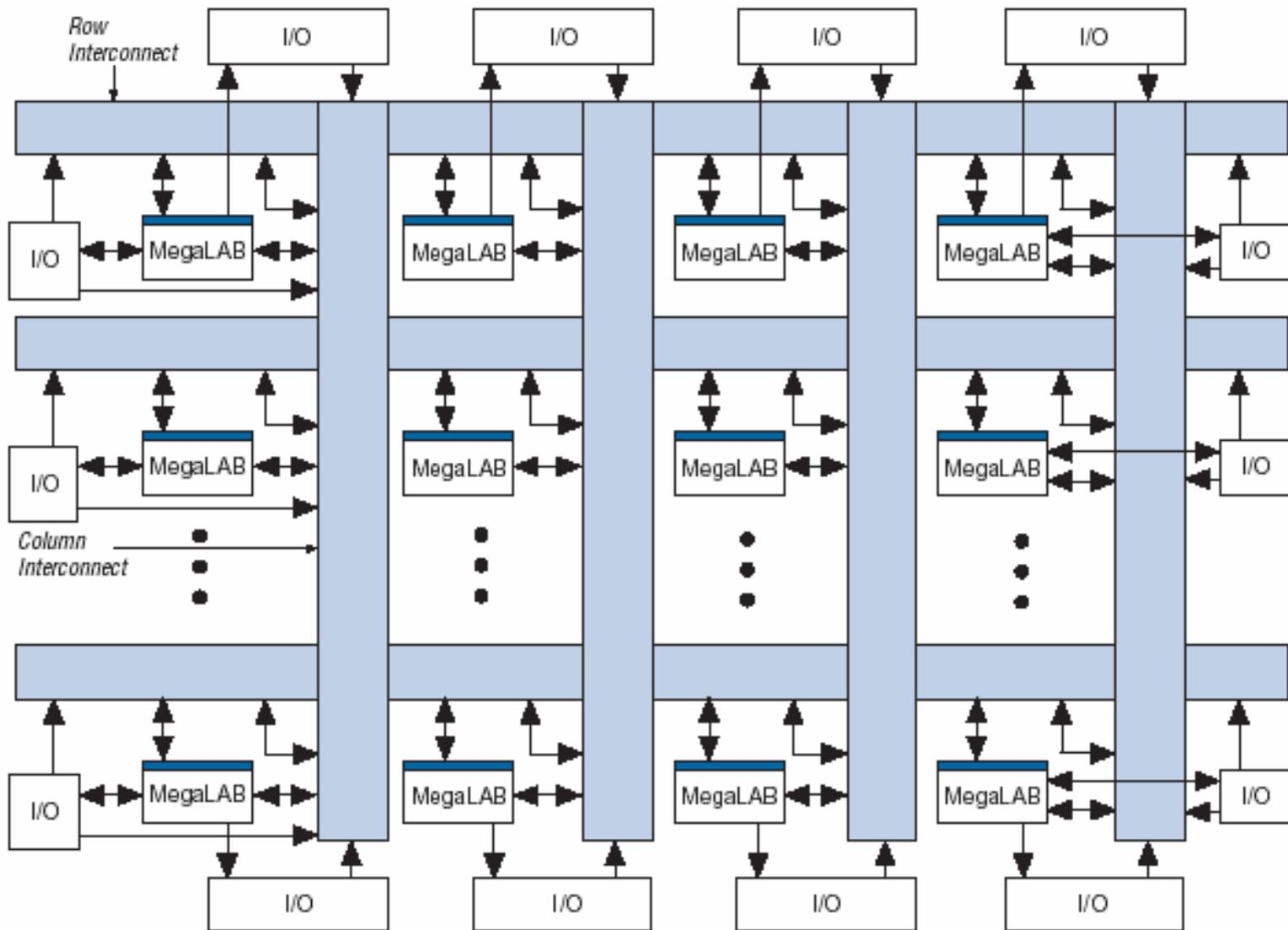
V. Le FPGA APEX de ALTERA

Informations générales

Figure 1. APEX II Device Block Diagram



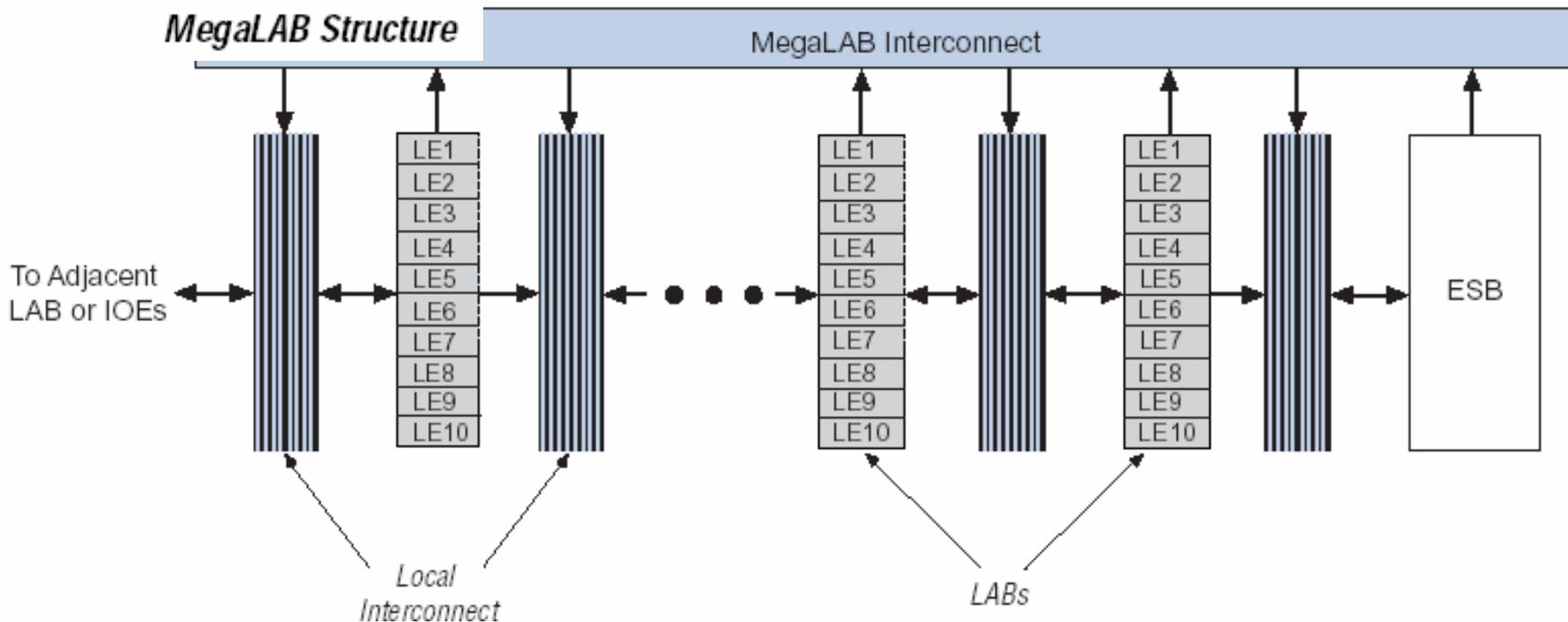
V. Le FPGA APEX de ALTERA



V. Le FPGA APEX de ALTERA

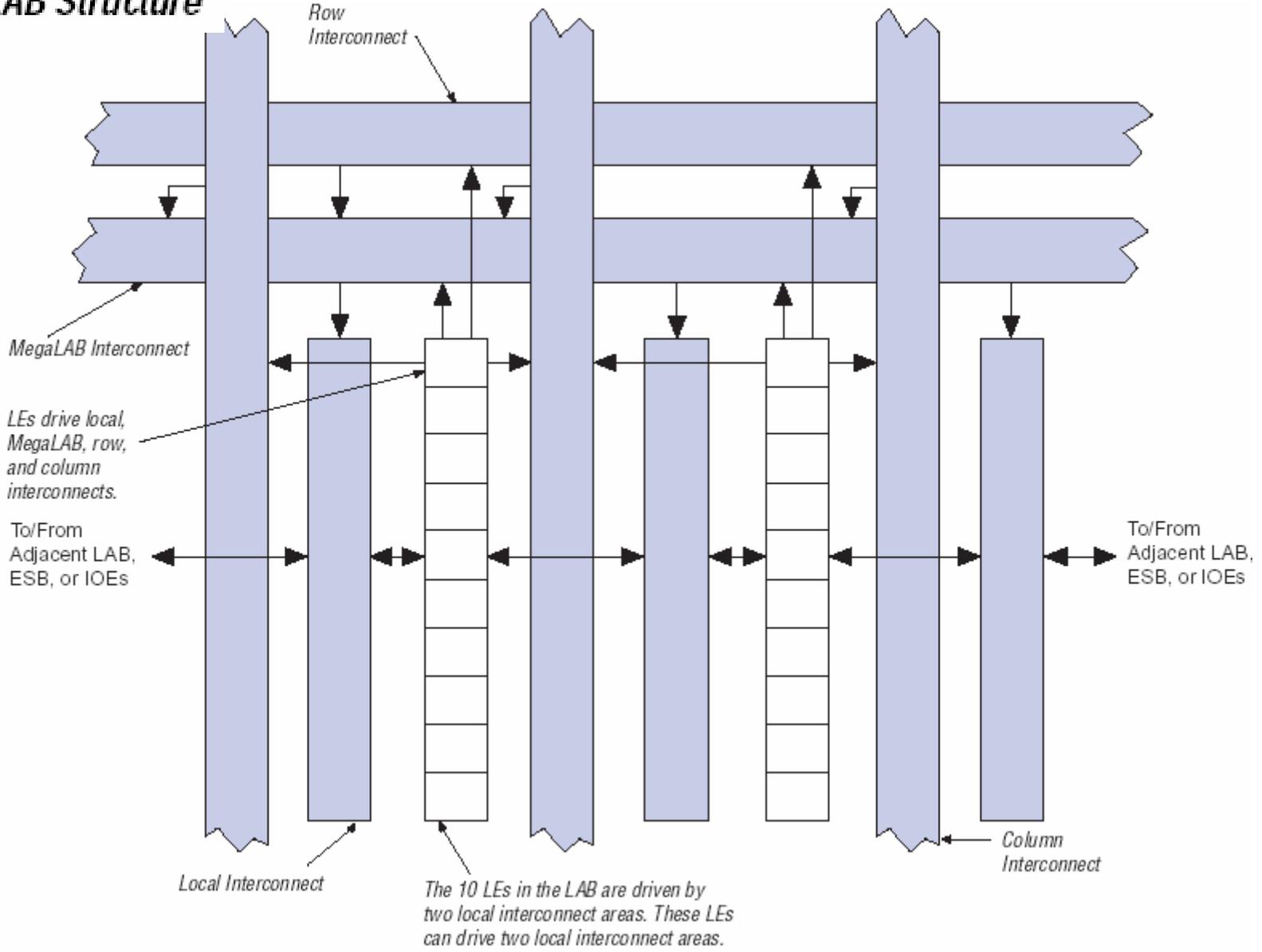
Structure

Système de communication entre les LABs



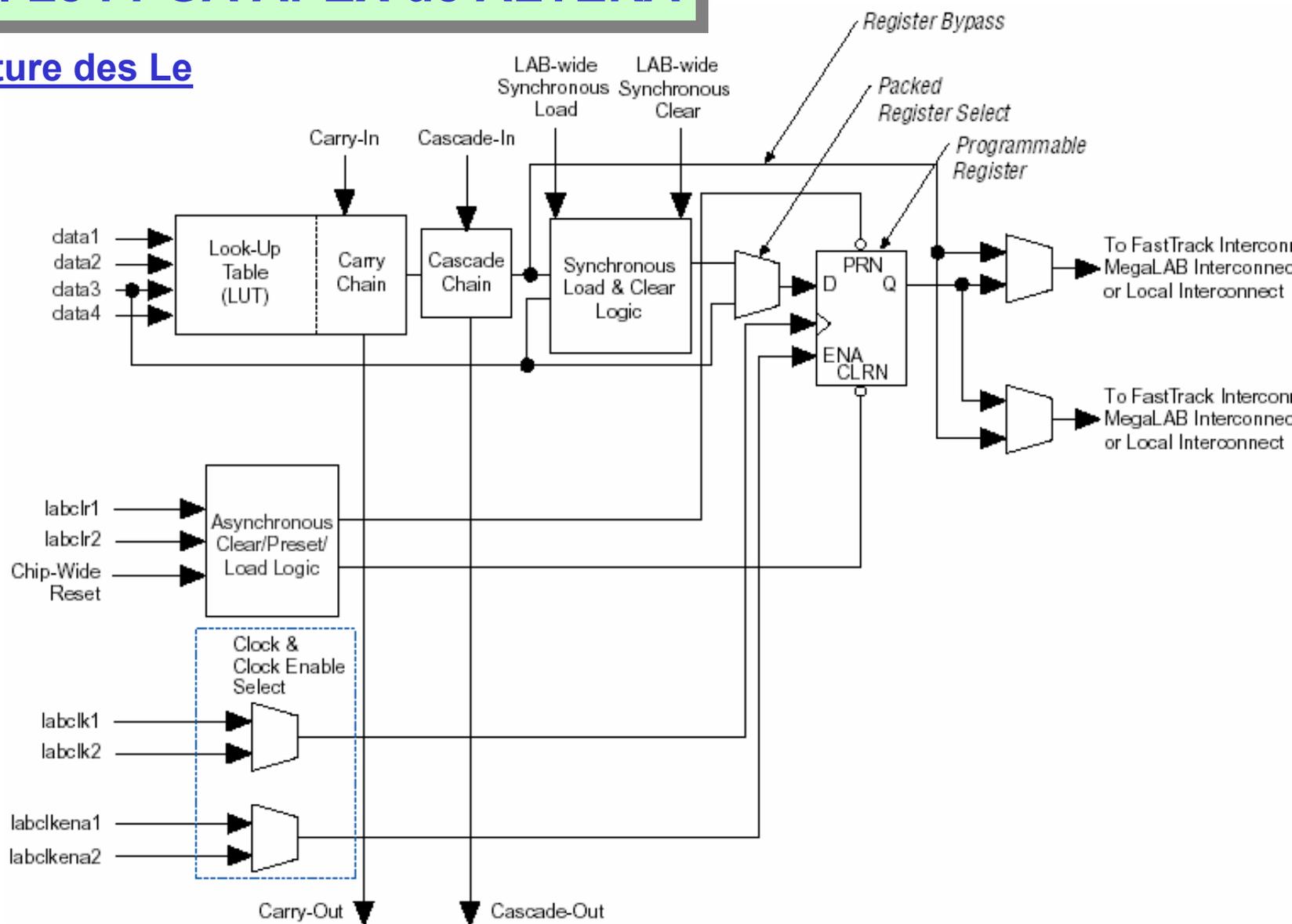
Chaque MegaLAB contient un groupe de 16 ou 24 LABs

APEX II LAB Structure



V. Le FPGA APEX de ALTERA

Structure des Le

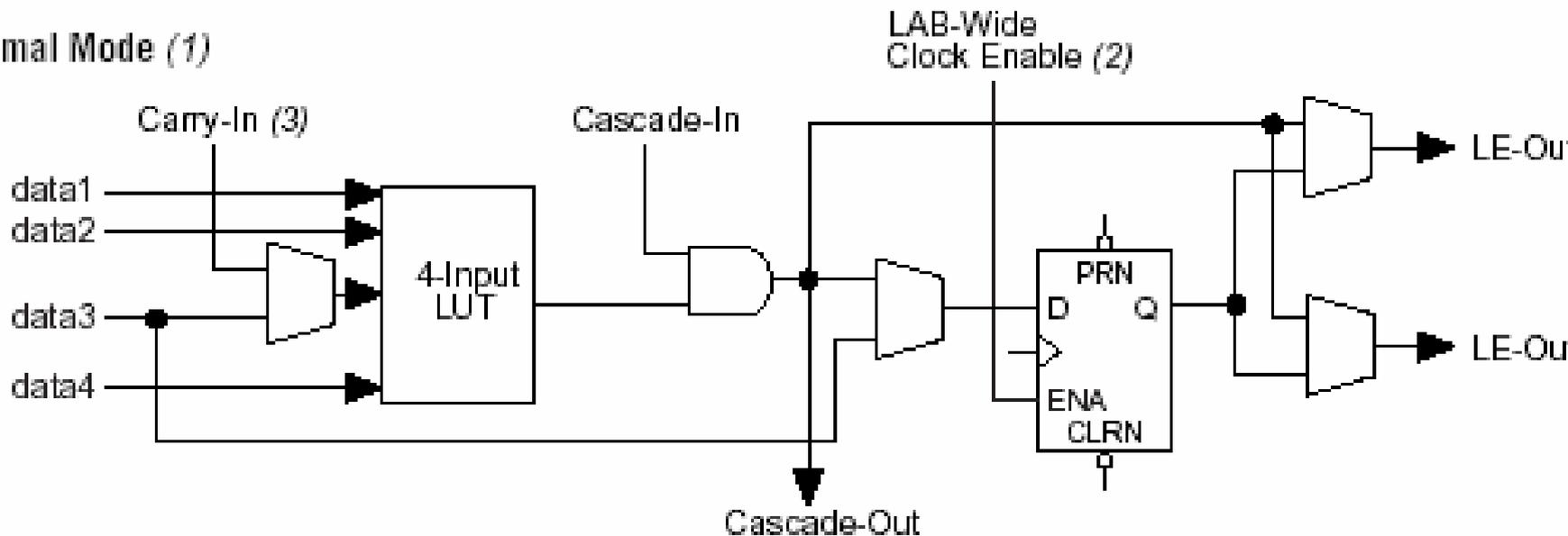


V. Le FPGA APEX de ALTERA

Mode de programmation des Le

- Normal mode
- Arithmetic mode
- Counter mode

Normal Mode (1)

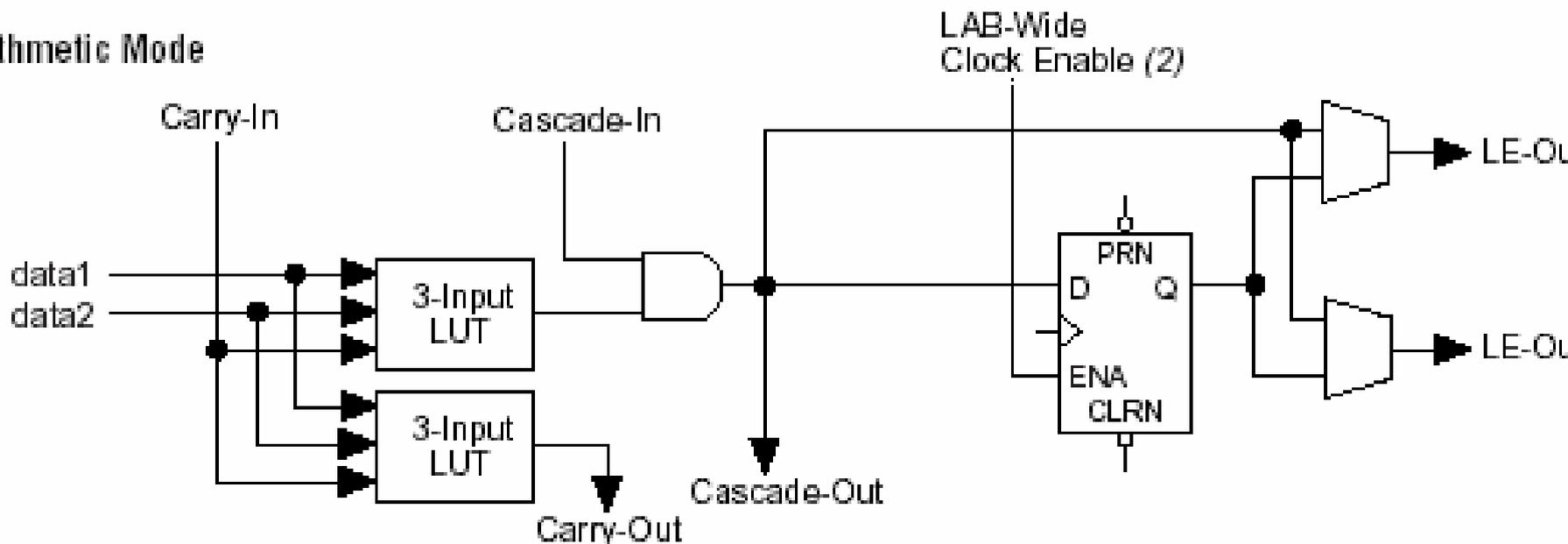


LE mode normal est utilisé pour la synthèse de structures combinatoires ou séquentiel, avec les possibilités de mise en cascade

V. Le FPGA APEX de ALTERA

Mode de programmation des Le

Arithmetic Mode

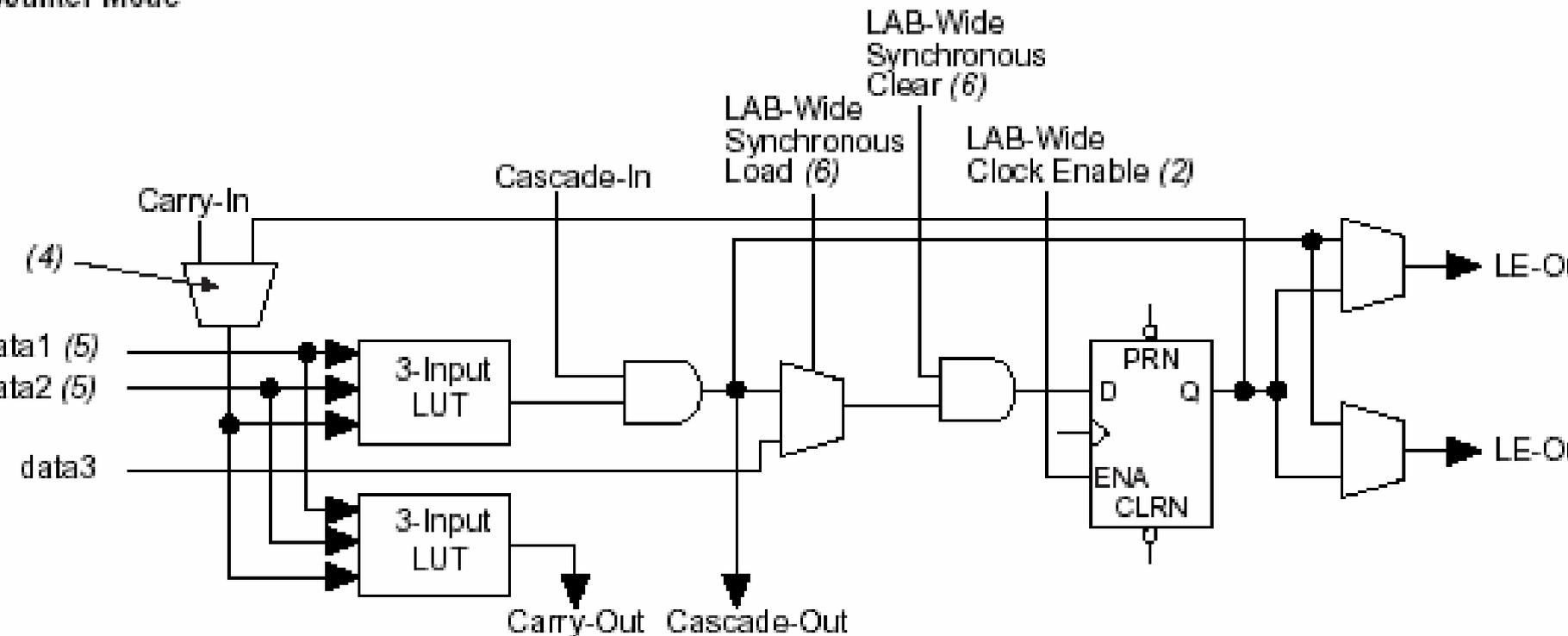


LE mode arithmétique est utilisé pour la synthèse de fonctions addition, comparaison, accumulation...

V. Le FPGA APEX de ALTERA

Mode de programmation des Le

Counter Mode



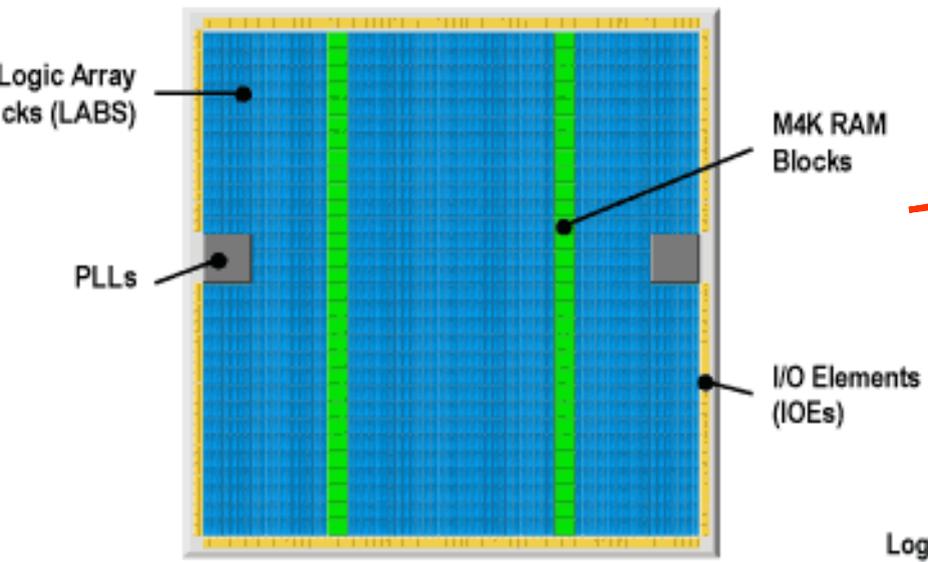
LE mode comptage propose des clock enable, counter enable, synchrone, Asynchrones, up, down, ...

V. Le FPGA Cyclone de ALTERA

Low-Cost FPGAs		<p>Second-generation, lowest-cost family in the Cyclone™ FPGA series for designs where cost concerns outweigh the need for performance or extensive features</p>	<ul style="list-style-type: none"> • Nios® II embedded processor support • Embedded 18x18 digital signal processing (DSP) multipliers • Moderate on-chip memory • Moderate-speed I/O & memory interfaces • Broad intellectual property (IP) portfolio support
		<p>First-generation, lower-density, low-cost family in the Cyclone FPGA series</p>	<ul style="list-style-type: none"> • Nios II embedded processor support • Low to moderate on-chip memory • Low to moderate-speed I/O & memory interfaces • Broad IP portfolio support

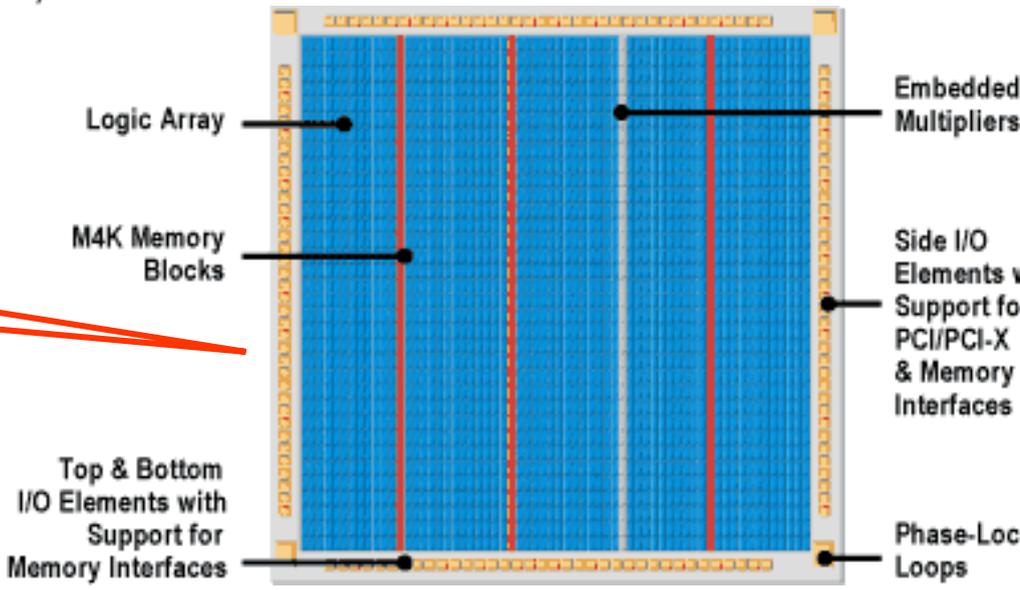
V. Le FPGA Cyclone de ALTERA

EP1C20



Cyclone I

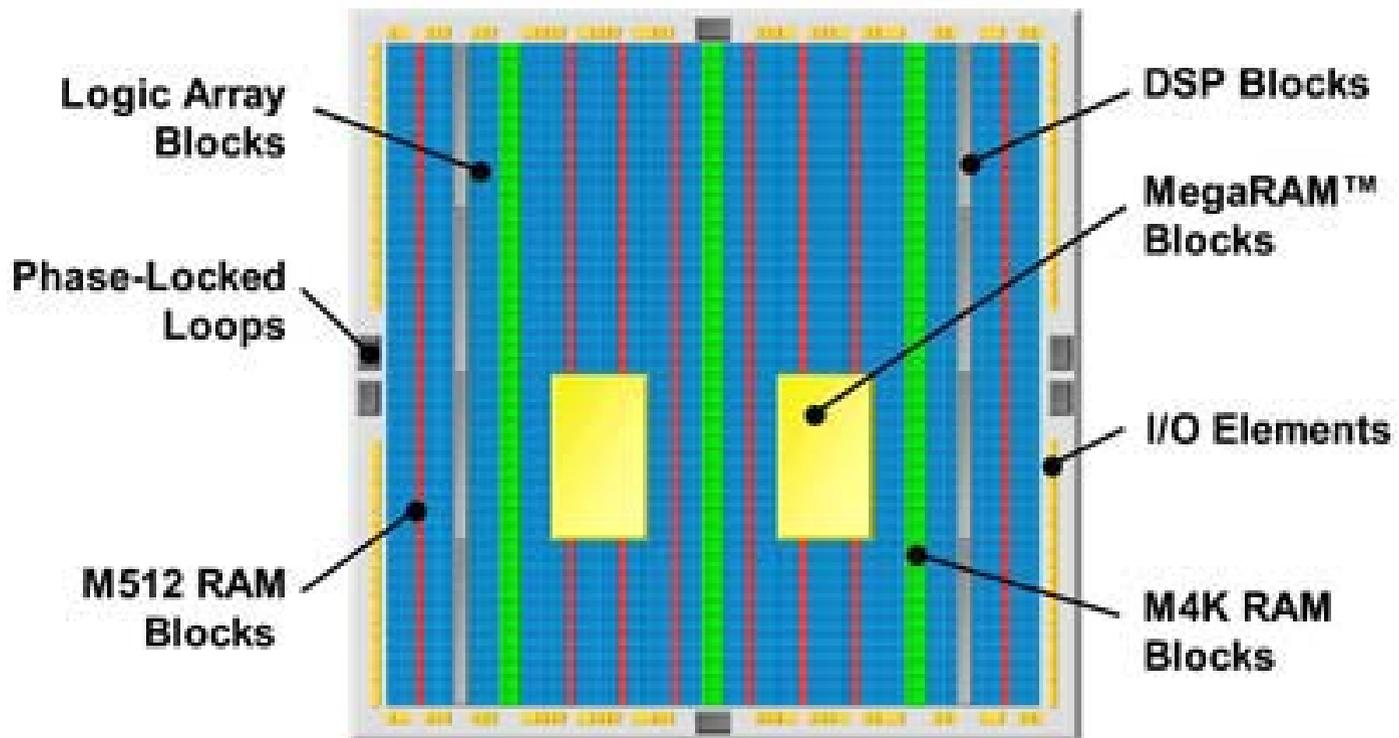
Cyclone II



V. Le FPGA Stratix de ALTERA

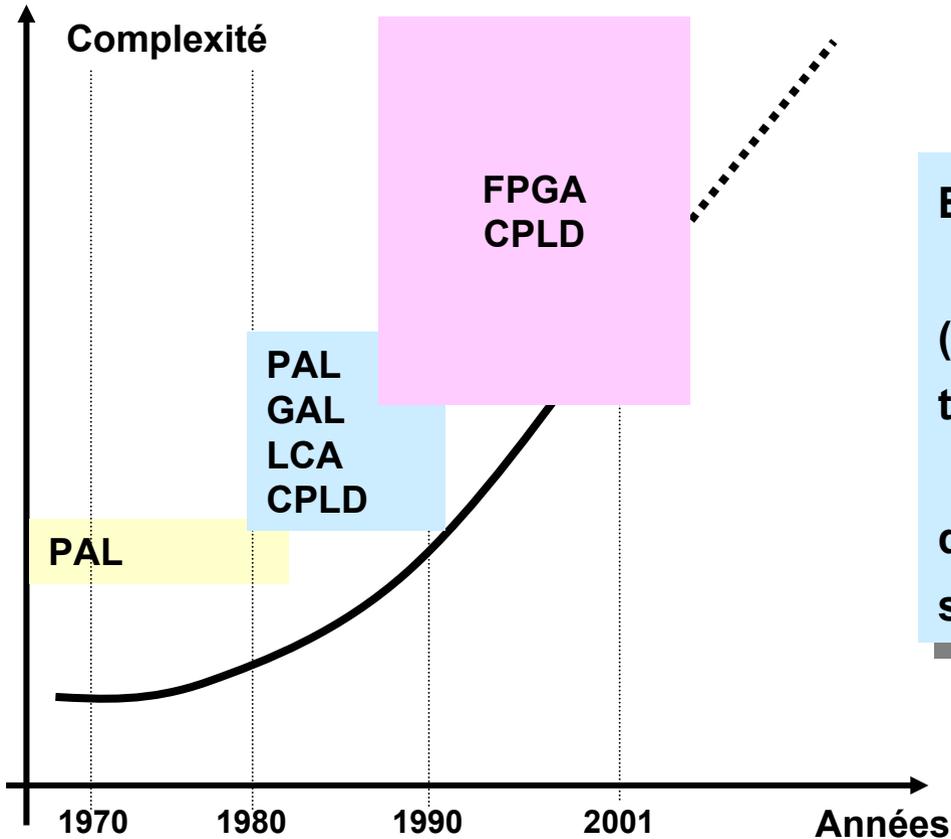
High-Density FPGAs		General-purpose FPGA family with the largest density & fastest performance	<ul style="list-style-type: none"> • Nios II embedded processor support • The most DSP blocks • Large on-chip memory • High-speed I/O & memory interfaces • 1-Gbps dynamic phase alignment (DPA) with source-synchronous signaling • Broad IP portfolio support
		General-purpose, high-performance FPGA family	<ul style="list-style-type: none"> • Nios II embedded processor support • DSP blocks • Large on-chip memory • High-speed I/O & memory interfaces • Broad IP portfolio support
		Stratix [®] architecture with high-speed signaling support	<ul style="list-style-type: none"> • All Stratix features • 3.125-Gbps transceivers • 1-Gbps DPA • Receiver equalization & transmitter pre-emphasis • Broad IP portfolio support

V. Le FPGA Stratix de ALTERA



VI. Les tendances des composants programmables

Evolution des systèmes vers les très grandes complexités

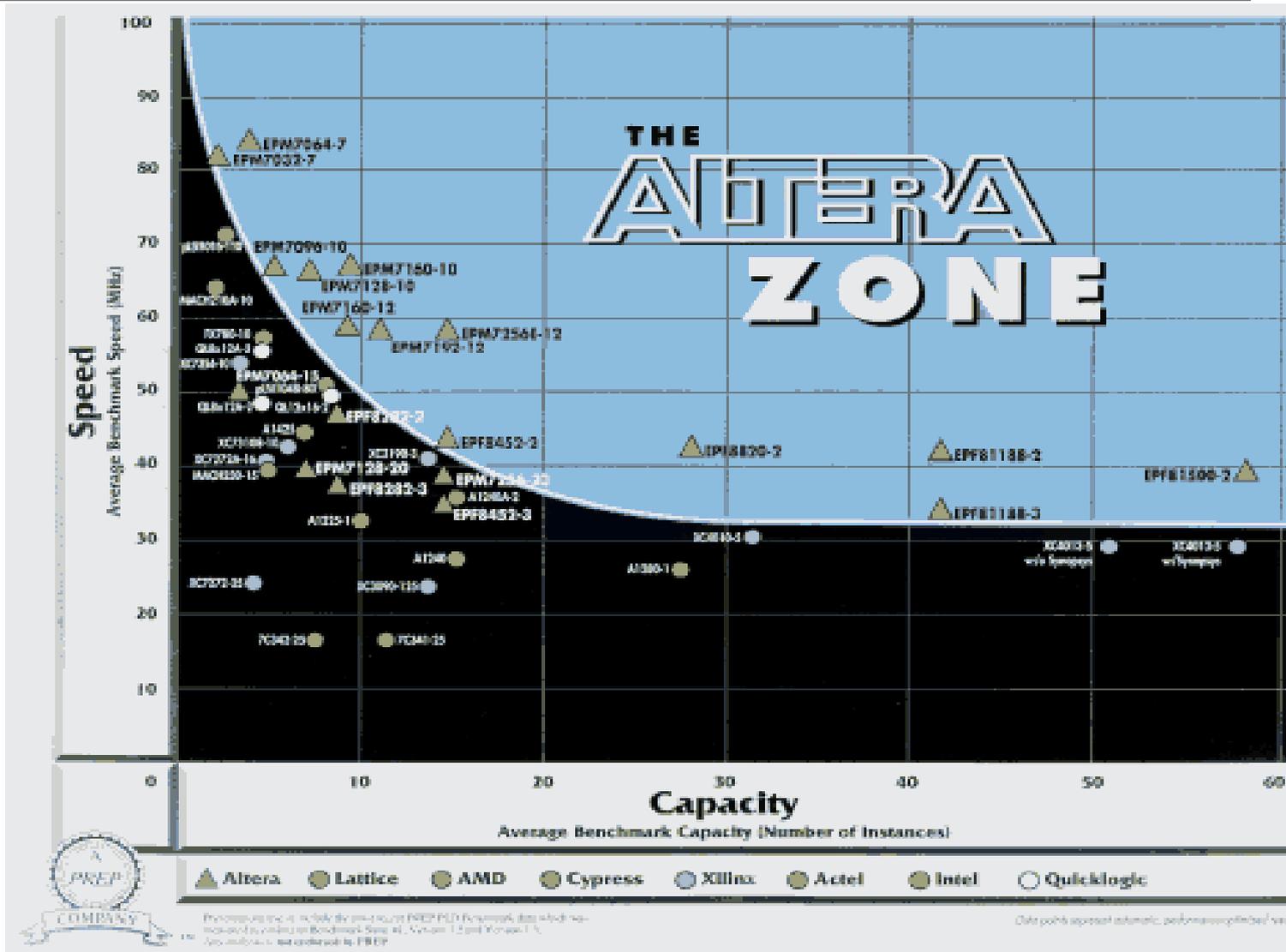


Evolution résultantes :

- des progrès technologiques (réduction des dimensions de transistors - densités d'intégration)
- des progrès logiciel des outils de développement (langage HDL - simplicité de développement)

VI. Les tendances des composants programmables

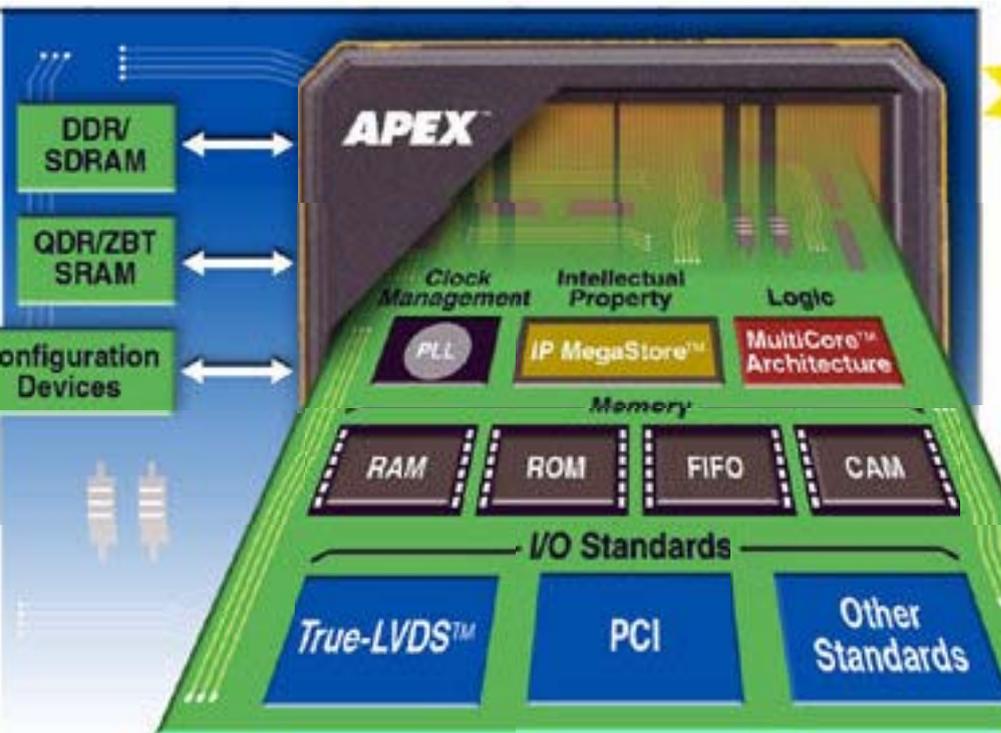
performances :



VI. Les tendances des composants programmables

Evolution vers les systèmes de très grande complexité

APEX Devices: System-on-a-Programmable-Chip Solutions from ALTERA chip



Introducing
APEX 20KC
Devices!

30,000 vers plus de 1.5 million de portes
(113,000 to over 2.5 million
system gates)

Technologie : 0.22- μm , 0.18- μm et
0.09 μm

Ports d'entrée/sortie : 128 à 808 I/O

Introduction des « System-on-chip »
dans les composants programmables
(RAM, ROM, FIFO, CAN, PCI, μP)

Aujourd'hui, près de 50% des FPGA sont des SoC

Introduction des IPs dans les FPGA (Intellectual Properties)

VI. Les tendances des composants programmables

Quels sont les problèmes limitant l'évolution et les performances des FPGA's?

Les interconnexions

Plus les technologies évoluent plus les interconnexions deviennent des éléments actifs des circuits

La densité d'intégration

La quantité de mémoire nécessaire pour accroître les performances des FPGA devient très importante

Les softs (méthodes de programmation) évolution vers les « systemC »

Les logiciels de développement évoluent vers des systèmes types Visual C++. C'est une obligation face à l'augmentation des possibilités de programmation des FPGA