

## Chapitre 3

### Architecture des FPGA

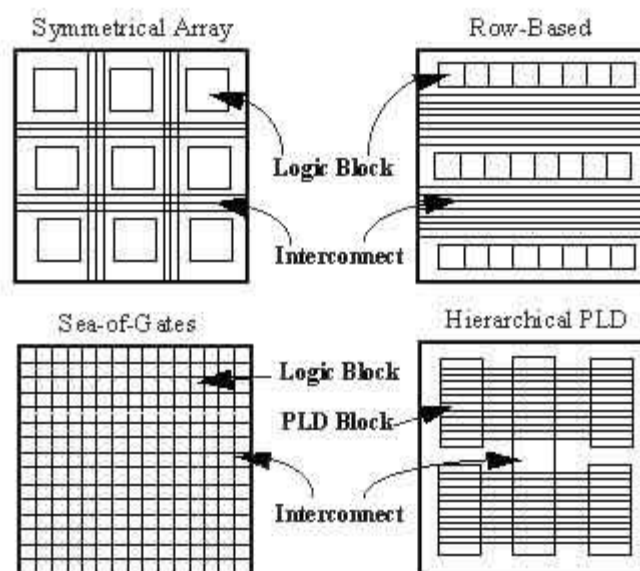
**Présentation rapide:** Les FPGA (Field Programmable Gate Arrays ou "réseaux logiques programmables") sont des composants **VLSI** entièrement reconfigurables ce qui permet de les reprogrammer à volonté afin d'accélérer notablement certaines phases de calculs. L'avantage de ce genre de circuit est sa grande souplesse qui permet de les réutiliser à volonté dans des algorithmes différents en un temps très court. Le progrès de ces technologies permet de faire des composants toujours plus rapides et à plus haute intégration, ce qui permet de programmer des applications importantes.

**Introduction:** Les circuits FPGA sont constitués d'une matrice de blocs logiques programmables entourés de blocs d'entrée sortie programmable. L'ensemble est relié par un réseau d'interconnexions programmable.

Les FPGA sont bien distincts des autres familles de circuits programmables tout en offrant le plus haut niveau d'intégration logique.

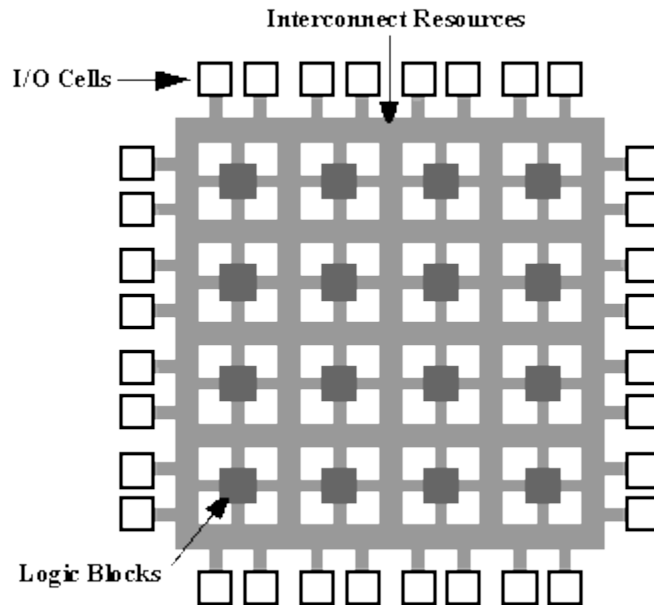
Il y a 4 principales catégories disponibles commercialement:

- Tableau symétrique.
- En colonne.
- Mers de portes.
- Les PLD hiérarchique.



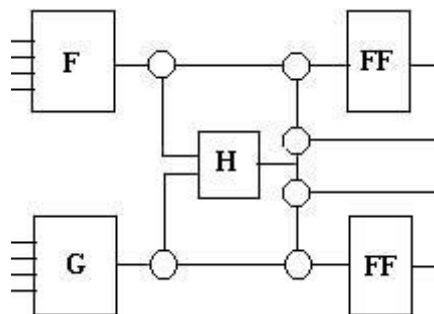
*Les différentes classes de FPGA.*

Voici la structure interne d'un FPGA de type matrice symétrique. Il s'agit de l'architecture que l'on retrouve dans les FPGA de la série XC4000 de chez Xilinx.



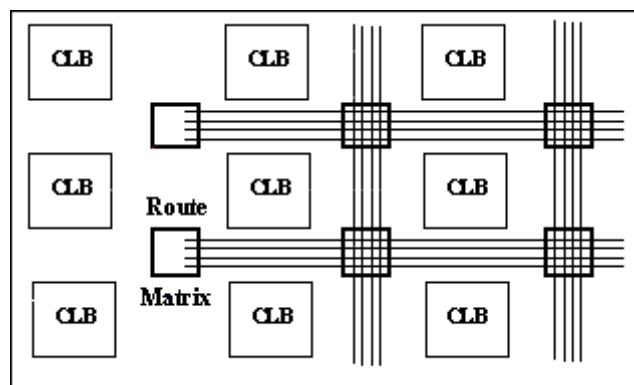
*Structure interne d'un FPGA*

L'utilisateur peut programmer la fonction réalisée par chaque cellule (appelée CLB par Xilinx: Configurable Logic Block):



*Schéma bloc d'une cellule*

On programme aussi les interconnexions entre les cellules:



*Les interconnexions entre les cellule d'un FPGA*

Ainsi que les entrées et sorties du circuit. L'avantage des FPGA est de pouvoir être configuré sur place, sans envoi du circuit chez le fabricant, ce qui permet de les utiliser quelques minutes après leur conceptions. Les FPGA les plus récents sont configurables en une centaine

de millisecondes. Les FPGA sont utilisés pour un développement rapide et bon marché des **ASIC**.

Inventés par la société Xilinx, le FPGA, dans la famille des **ASICs**, se situe entre les **réseaux logiques programmables** et les **prédifusés**. C'est donc un composant standard combinant la densité et les performances d'un prédifusé avec la souplesse due à la reprogrammation des PLD. Cette configuration évite le passage chez le fondeur et tous les inconvénients qui en découlent.

## Les différentes technologies utilisées pour les FPGA:

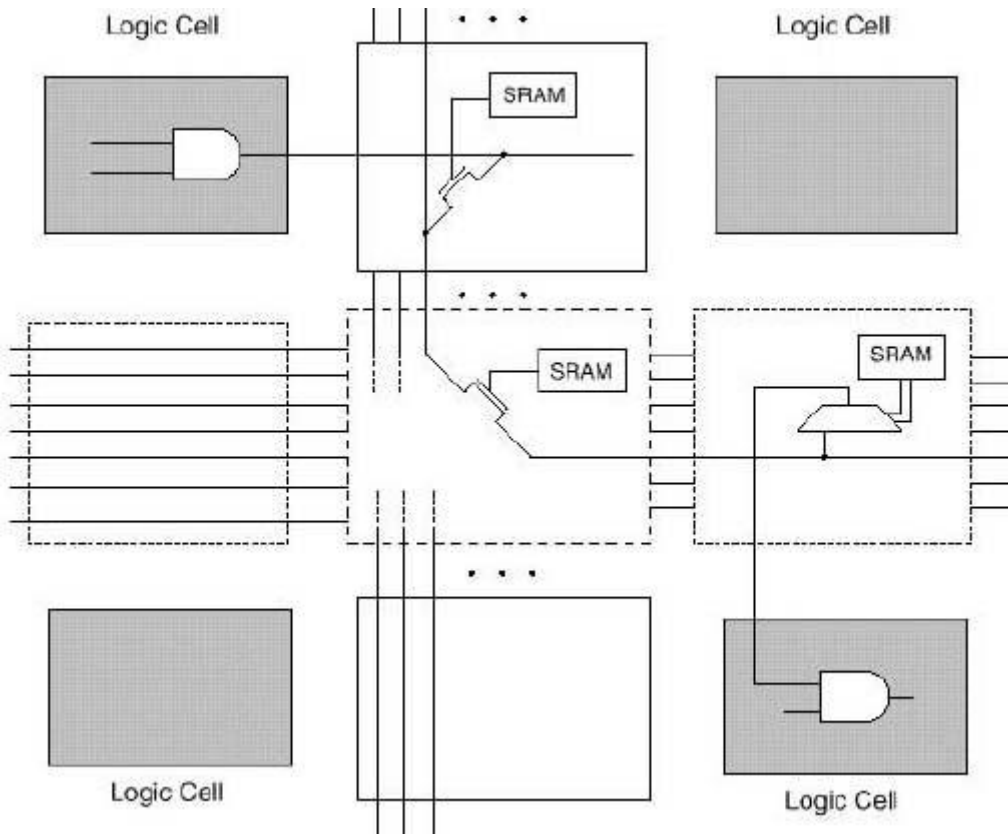
Elles sont au nombre de quatre:

- **Static RAM:** Les connexions sont réalisées en rendant les transistors passant. L'avantage de cette technologie est qu'elle permet une reconfiguration rapide au sein même du circuit. Le principal désavantage est la surface nécessaire pour la SRAM. Ils nécessitent l'utilisation d'une mémoire standard chargée à l'initialisation. Ils font appel à la technologie CMOS. Résistance d'environ 1 Kohm et capacité de quelques dizaines de femtofarad.

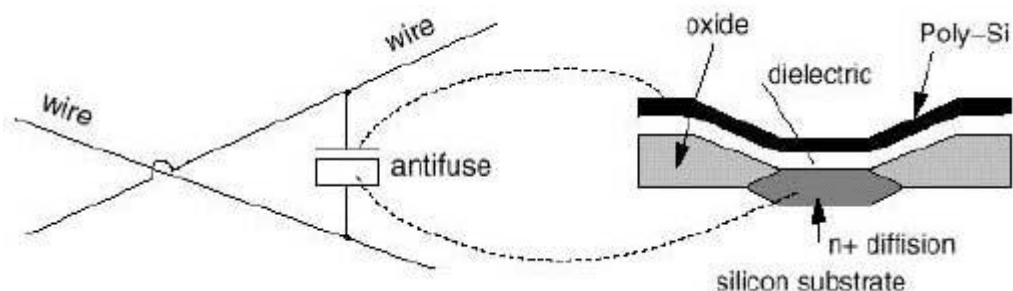
Voici quelques exemples de FPGA utilisant la technologie SRAM:

- **Xilinx XC3000**
- **Xilinx XC5200**
- **Xilinx XC4000**
- **Xilinx Spartan**

Les FPGA comportant le plus grand nombre de portes se trouvent dans les séries Virtex-II. Leurs densités s'étendent de 40 mille à 8 million de portes système. Ce qui est énorme sachant que l'on estime qu'un FPGA de 100 mille portes système suffit amplement à émuler complètement un microcontrôleur de type 8051. Ils possèdent une fréquence d'horloge de maximum 420 MHz et utilisent un processus de 0.15 microns à 8 couches de métal.



- La technologie anti-Fuse: Un état anti-fuse réside en un état d'haute impédance. Il peut être programmé dans un état de faible impédance ou état "fused". Il s'agit d'une technologie moins chère que la SRAM, elle permet d'atteindre des vitesses plus élevées et occupe moins de place sur le circuit. Par contre, un tel FPGA ne peut être programmé qu'une seule fois. Performances électriques supérieures à la technologie SRAM (minimisation des effets RC due à la faible surface). Résistance de l'ordre d'une centaine d'ohm et effet capacitif de quelques femtofarads.



- EPROM/EEPROM: Cette méthode est la même que celle utilisée dans les mémoires EPROM. L'EEPROM est reprogrammable. La puce fonctionne seule. La surface moyenne et les caractéristiques électriques sont semblables à la SRAM.

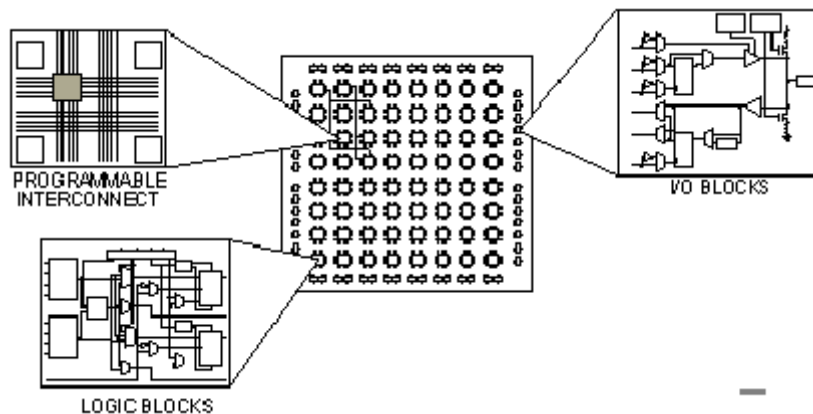
Résumé des différentes technologies			
Nom:	Reprogrammable:	Volatile:	Technologie:
Fuse	Non	Non	Bipolaire
EPROM	Oui, en dehors du circuit	Non	UVCMOS
EEPROM	Oui, dans le circuit	Non	EECMOS
SRAM	Oui, dans le circuit	Oui	CMOS
Antifuse	Non	Non	CMOS

## Architecture détaillée de la série XC4000 de Xilinx:

**Introduction:** L'architecture, retenue par Xilinx, se présente sous forme de deux couches :

- une couche appelée circuit configurable,
- une couche réseau mémoire SRAM.

La couche dite 'circuit configurable' est constituée d'une matrice de blocs logiques configurables CLB permettant de réaliser des **fonctions combinatoires** et des **fonctions séquentielles**. Tout autour de ces blocs logiques configurables, nous trouvons des blocs entrées/sorties IOB dont le rôle est de gérer les entrées-sorties réalisant l'interface avec les modules extérieurs (cf. figure). La programmation du circuit FPGA appelé aussi LCA (logic cells arrays) consistera par le biais de l'application d'un potentiel adéquat sur la grille de certains transistors à effet de champ à interconnecter les éléments des CLB et des IOB afin de réaliser les fonctions souhaitées et d'assurer la propagation des signaux. Ces potentiels sont tout simplement mémorisés dans le réseau mémoire SRAM.



*Architecture interne d'un FPGA.*

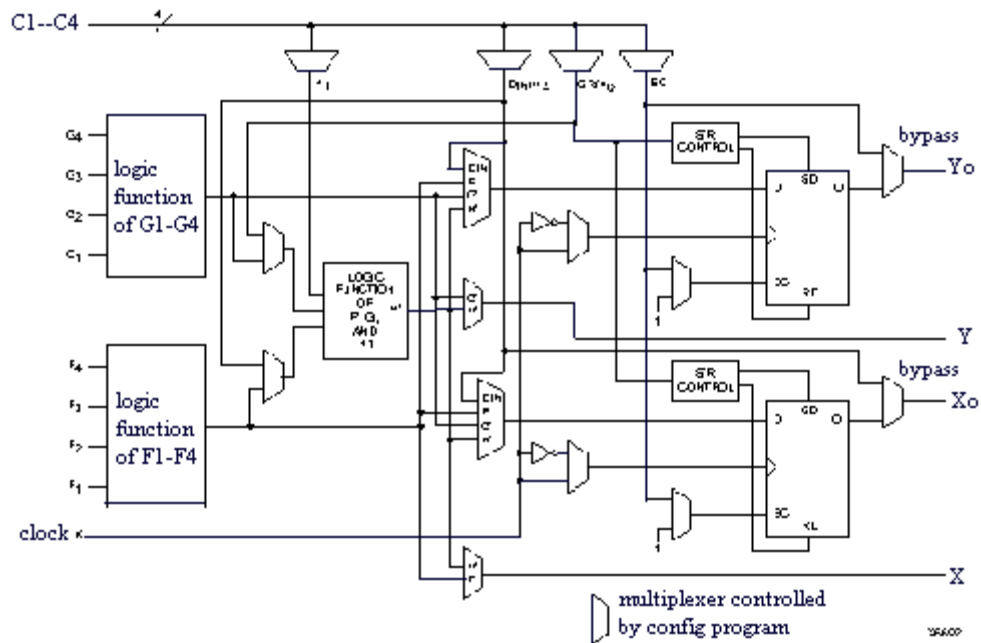
Les circuits FPGA du fabricant Xilinx utilisent deux types de cellules de base :

- les cellules d'entrées/sorties appelés IOB (input output bloc),

- les cellules logiques appelées CLB (configurable logic bloc). Ces différentes cellules sont reliées entre elles par un réseau d'interconnexions configurable.

### I. Les CLB (configurable logic bloc):

Les blocs logiques configurables sont les éléments déterminants des performances du FPGA. Chaque bloc est composé d'un bloc de logique combinatoire composé de deux générateurs de fonctions à quatre entrées et d'un bloc de mémorisation synchronisation composé de deux bascules D. Quatre autres entrées permettent d'effectuer les connexions internes entre les différents éléments du CLB. La figure ci-dessous, nous montre le schéma d'un CLB.



Cellule logique (CLB).

Voyons d'abord le bloc logique combinatoire qui possède deux générateurs de fonctions F' et G' à quatre entrées indépendantes (F1...F4, G1...G4), lesquelles offrent aux concepteurs une flexibilité de développement importante car la majorité des fonctions aléatoires à concevoir n'excède pas quatre variables. Les deux fonctions sont générées à partir d'une table de vérité câblée inscrite dans une zone mémoire, rendant ainsi les délais de propagation pour chaque générateur de fonction indépendants de celle à réaliser. Une troisième fonction H' est réalisée à partir des sorties F' et G' et d'une troisième variable d'entrée H1 sortant d'un bloc composé de quatre signaux de contrôle H1, Din, S/R, Ec. Les signaux des générateurs de fonction peuvent sortir du CLB, soit par la sortie X, pour les fonctions F' et G', soit Y pour les fonctions G' et H'. Ainsi un CLB peut être utilisé pour réaliser:

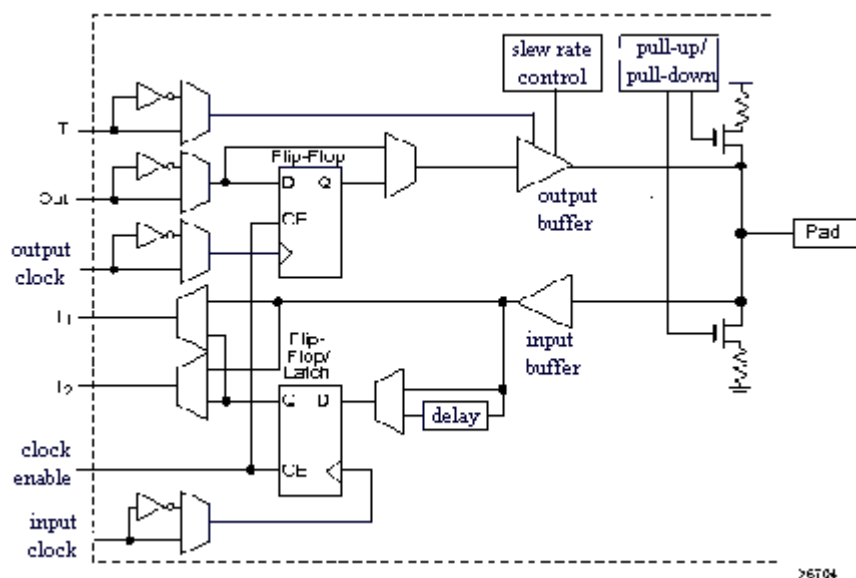
- deux fonctions indépendantes à quatre entrées indépendantes, plus une troisième fonction de trois variables indépendantes
- ou toute fonction à cinq variables
- ou toute fonction à quatre variables et une autre avec quelques fonctions à six variables
- ou certaines fonctions jusqu'à neuf variables.

L'intégration de fonctions à nombreuses variables diminue le nombre de CLB nécessaires, les délais de propagation des signaux et par conséquent augmente la densité et la vitesse du circuit. Les sorties de ces blocs logiques peuvent être appliquées à des bascules au nombre de deux ou directement à la sortie du CLB (sorties X et Y). Chaque bascule présente deux modes de fonctionnement : un mode **'flip-flop'** avec comme donnée à mémoriser, soit l'une des fonctions F', G', H' soit l'entrée directe DIN et un mode **latch**. La donnée peut être mémorisée sur un **front montant ou descendant** de l'horloge (CLK). Les sorties de ces deux bascules correspondent aux sorties du CLB XQ et YQ. Un mode dit de "verrouillage" exploite une entrée S/R qui peut être programmée soit en mode SET, mise à 1 de la bascule, soit en Reset, mise à zéro de la bascule. Ces deux entrées coexistent avec une autre entrée laquelle n'est pas représentée sur la figure, appelée le global Set/Reset. Cette entrée initialise le circuit FPGA à chaque mise sous tension, à chaque configuration, en commandant toutes les bascules au même instant soit à '1', soit à '0'. Elle agit également lors d'un niveau actif sur le fil RESET lequel peut être connecté à n'importe quelle entrée du circuit FPGA.

Un mode optionnel des CLB est la configuration en mémoire RAM de 16x2bits ou 32x1bit ou 16x1bit (en mode DUAL port). Les entrées F1 à F4 et G1 à G4 deviennent des lignes d'adresses sélectionnant une cellule mémoire particulière. La fonctionnalité des signaux de contrôle est modifiée dans cette configuration, les lignes H1, DIN et S/R deviennent respectivement les deux données D0, D1 (RAM 16x2bits) d'entrée et le signal de validation d'écriture WE. Le contenu de la cellule mémoire (D0 et D1) est accessible aux sorties des générateurs de fonctions F' et G'. Ces données peuvent sortir du CLB à travers ses sorties X et Y ou alors en passant par les deux bascules.

## II. Les IOB (input output bloc):

La figure présente la structure de ce bloc. Ces blocs entrée/sortie permettent l'interface entre les broches du composant FPGA et la logique interne développée à l'intérieur du composant. Ils sont présents sur toute la périphérie du circuit FPGA. Chaque bloc IOB contrôle une broche du composant et il peut être défini en entrée, en sortie, en signaux bidirectionnels ou être inutilisé (haute impédance).



**III. Configuration en entrée:**

Premièrement, le signal d'entrée traverse un buffer qui selon sa programmation peut détecter soit des seuils TTL ou soit des seuils CMOS. Il peut être routé directement sur une entrée directe de la logique du circuit FPGA ou sur une entrée synchronisée. Cette synchronisation est réalisée à l'aide d'une bascule de type D, le changement d'état peut se faire sur un front montant ou descendant. De plus, cette entrée peut être retardée de quelques nanosecondes pour compenser le retard pris par le signal d'horloge lors de son passage par l'amplificateur. Le choix de la configuration de l'entrée s'effectue grâce à un multiplexeur (program controlled multiplexer). Un bit positionné dans une case mémoire commande ce dernier.

**IV. Configuration en sortie:**

Nous distinguons les possibilités suivantes :

- inversion ou non du signal avant son application à l'IOB,
- synchronisation du signal sur des fronts montants ou descendants d'horloge,
- mise en place d'un " pull-up " ou " pull-down " dans le but de limiter la consommation des entrées sorties inutilisées,
- signaux en logique trois états ou deux états. Le contrôle de mise en haute impédance et la réalisation des lignes bidirectionnelles sont commandés par le signal de commande Out Enable lequel peut être inversé ou non. Chaque sortie peut délivrer un courant de 12mA. Ainsi toutes ces possibilités permettent au concepteur de connecter au mieux une architecture avec les périphériques extérieurs.

**V. Les différents types d'interconnexions:**

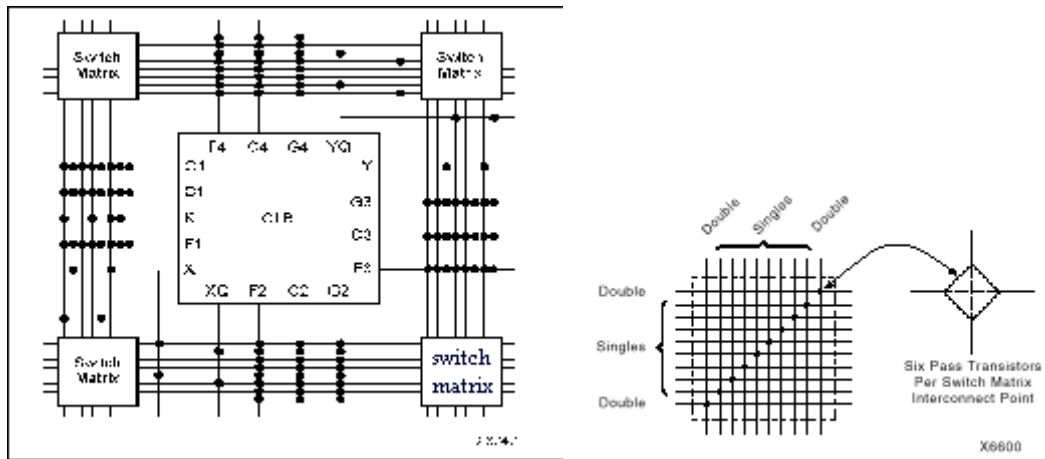
Les connexions internes dans les circuits FPGA sont composées de segments métallisés. Parallèlement à ces lignes, nous trouvons des matrices programmables réparties sur la totalité du circuit, horizontalement et verticalement entre les divers CLB. Elles permettent les connexions entre les diverses lignes, celles-ci sont assurées par des transistors MOS dont l'état est contrôlé par des cellules de mémoire vive ou RAM. Le rôle de ces interconnexions est de relier avec un maximum d'efficacité les blocs logiques et les entrées/sorties afin que le taux d'utilisation dans un circuit donné soit le plus élevé possible. Pour parvenir à cet objectif, Xilinx propose trois sortes d'interconnexions selon la longueur et la destination des liaisons. Nous disposons :

- d'interconnexions à usage général,
- d'interconnexions directes,
- de longues lignes.

**VI. Les interconnexions à usage général:**

Ce système fonctionne en une grille de cinq segments métalliques verticaux et quatre segments horizontaux positionnés entre les rangées et les colonnes de CLB et de l'IOB.



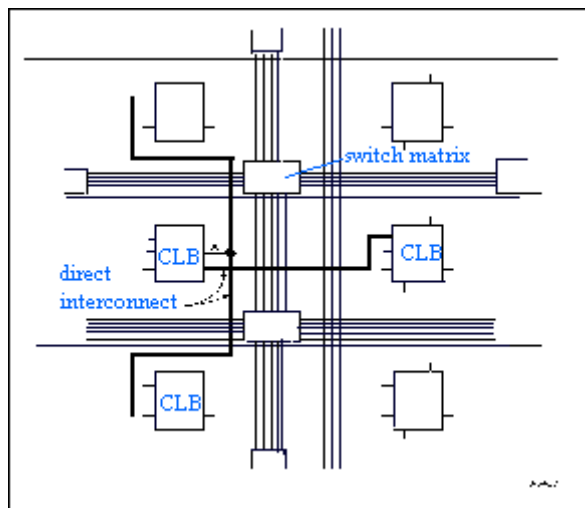


*Connexions à usage général et détail d'une matrice de commutation.*

Des aiguilleurs appelés aussi matrices de commutation sont situés à chaque intersection. Leur rôle est de raccorder les segments entre eux selon diverses configurations, ils assurent ainsi la communication des signaux d'une voie sur l'autre. Ces interconnexions sont utilisées pour relier un CLB à n'importe quel autre. Pour éviter que les signaux traversant les grandes lignes ne soient affaiblis, nous trouvons généralement des buffers implantés en haut et à droite de chaque matrice de commutation.

## VII. Les interconnexions directes:

Ces interconnexions permettent l'établissement de liaisons entre les CLB et les IOB avec un maximum d'efficacité en terme de vitesse et d'occupation du circuit. De plus, il est possible de connecter directement certaines entrées d'un CLB aux sorties d'un autre.



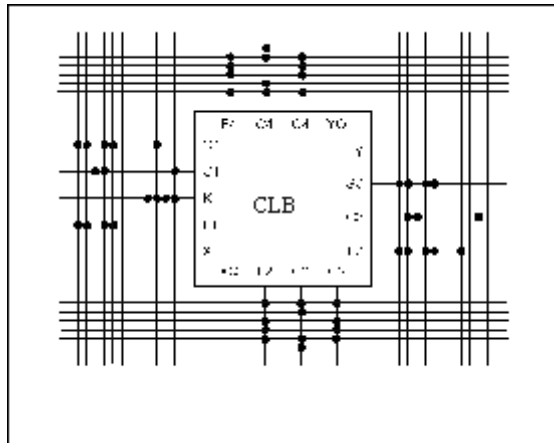
*Les interconnexions directes.*

Pour chaque bloc logique configurable, la sortie X peut être connectée directement aux entrées C ou D du CLB situé au-dessus et les entrées A ou B du CLB situé au-dessous. Quant à la sortie Y, elle peut être connectée à l'entrée B du CLB placé immédiatement

à sa droite. Pour chaque bloc logique adjacent à un bloc entrée/sortie, les connexions sont possibles avec les entrées I ou les sorties O suivant leur position sur le circuit.

### VIII. Les longues lignes:

Les longues lignes sont de longs segments métallisés parcourant toute la longueur et la largeur du composant, elles permettent éventuellement de transmettre avec un minimum de retard les signaux entre les différents éléments dans le but d'assurer un synchronisme aussi parfait que possible. De plus, ces longues lignes permettent d'éviter la multiplicité des points d'interconnexion.



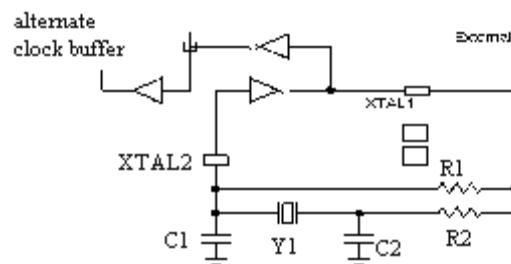
*Les longues lignes.*

### IX. Performances des interconnexions:

Les performances des interconnexions dépendent du type de connexions utilisées. Pour les interconnexions à usage général, les délais générés dépendent du nombre de segments et de la quantité d'aiguilleurs employés. Le délai de propagation de signaux utilisant les connexions directes est minimum pour une connectique de bloc à bloc. Quant aux segments utilisés pour les longues lignes, ils possèdent une faible résistance mais une capacité importante. De plus, si on utilise un aiguilleur, sa résistance s'ajoute à celle existante.

### X. L'oscillateur à quartz:

Placé dans un angle de la puce, il peut être activé lors de la phase de programmation pour réaliser un oscillateur. Il utilise deux IOB voisins, pour réaliser l'oscillateur dont le schéma est présenté ci-dessous. Cette oscillateur ne peut être réalisé que dans un angle de la puce où se trouve l'amplificateur prévu à cet effet. Il est évident que si l'oscillateur n'est pas utilisé, les deux IOB sont utilisables au même titre que les autres IOB.



*L'oscillateur à quartz.*

# Programmation des FPGA de chez Xilinx:

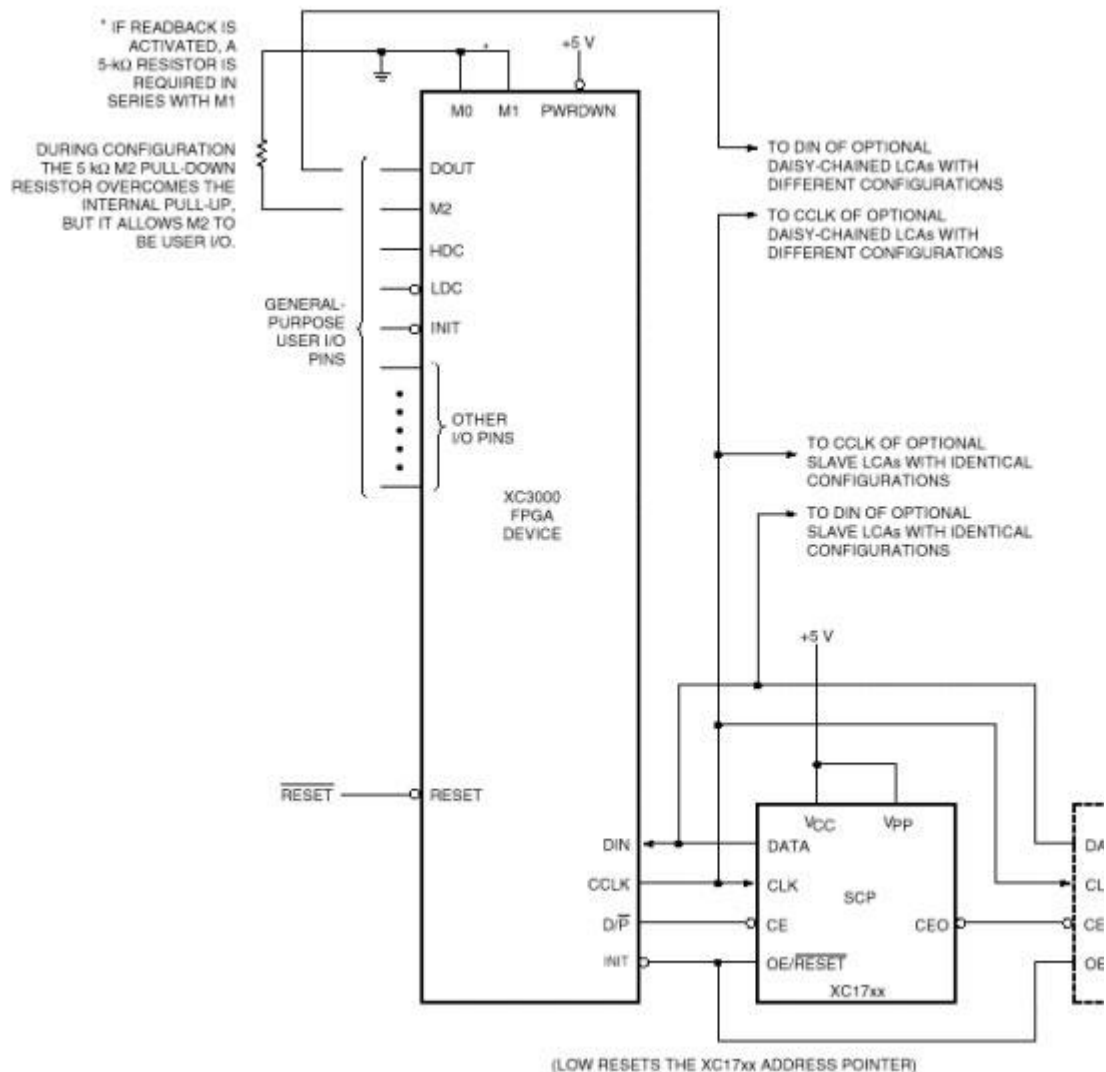
**Introduction:** La configuration du circuit est mémorisée sur la couche réseau SRAM et stockée dans une ROM externe. Un dispositif interne permet à chaque mise sous tension de charger la SRAM interne à partir de la ROM. Ainsi on conçoit aisément qu'un même circuit puisse être exploité successivement avec des ROM différentes puisque sa programmation interne n'est jamais définitive. On voit tout le parti que l'on peut tirer de cette souplesse en particulier lors d'une phase de mise au point. Une erreur n'est pas rédhibitoire, mais peut aisément être réparée.

**Technique de programmation des FPGA:** Les circuits FPGA ne possèdent pas de programme résident. A chaque mise sous tension, il est nécessaire de les configurer. Leur configuration permet d'établir des interconnexions entre les CLB et IOB. Pour cela, ils disposent d'une RAM interne dans laquelle sera écrit le fichier de configuration. Le format des données du fichier de configuration est produit automatiquement par le logiciel de développement sous forme d'un ensemble de bits organisés en champs de données. Le FPGA dispose de quatre modes de chargement et de trois broches M0, M1, M2 lesquelles définissent les différents modes comme l'indique le tableau ci-dessous. Ces modes définissent les différentes méthodes pour envoyer le fichier de configuration vers le circuit FPGA, selon deux approches complémentaires :

- configuration automatique, le circuit FPGA est autonome,
- configuration externe, l'intervention d'un opérateur est nécessaire.
- M0 M1 M2 Mode sélectionné
  - 0 0 0 maître série
  - 0 0 1 maître parallèle bas
  - 0 1 1 maître parallèle haut
  - 1 0 1 périphérique
  - 1 1 1 esclave série

## I. **Mode maître série:**

Le mode maître série utilise une mémoire à accès série de type registre à décalage commercialisée par le fabricant Xilinx. Le programme est préalablement chargé par le système de développement utilisé pour le circuit FPGA. Le FPGA génère tous les signaux de dialogue nécessaires pour la copie du contenu de la PROM dans sa RAM interne, lorsque la copie est terminée, il bascule le signal DONE pour le signaler au circuit. Comme nous pouvons le remarquer sur la figure, une seule PROM peut configurer plusieurs circuits FPGA avec la même configuration ou plusieurs PROM peuvent configurer plusieurs FPGA en chaîne où le premier des circuits FPGA est le maître et génère l'horloge. Les données en provenance des PROM sont envoyées aux autres circuits FPGA par la sortie DOUT de ce premier.



On dispose aussi d'un mode maître-parallèle où le FPGA est relié en parallèle à une EPROM classique, de même qu'un mode passif type périphérique dans lequel le FPGA est considéré comme un périphérique de  $\mu P$  et peut-être configuré à partir de celui-ci.

- II. **Mode esclave:** Dans ce mode, le programme de configuration peut être envoyé à partir d'un PC, d'une station de travail ou à partir d'un autre circuit FPGA. Le circuit FPGA maître peut être interfacé à une mémoire en mode parallèle ou série sans apporter aucune modification au niveau du câblage des circuits FPGA esclaves. C'est souvent le mode exploité pour la mise au point d'une configuration.

**Programmation des FPGA:** On commence par décrire le design soit en utilisant un langage de description matériel (tel VHDL, Verilog, ABEL,...) soit en rentrant directement le schématique. Le synthétiseur va générer la netlist, ensuite il faudra placer tous ces composants (si c'est possible, en effet, certains FPGA ne permettent pas d'émuler des Latches) dans un FPGA et effectuer le routage entre les différentes cellules logiques. Au terme de ces étapes, le synthétiseur aura généré le bitstream qui sera prêt à être envoyé vers le FPGA. C'est seulement à ce moment là que la programmation proprement dite pourra avoir lieu.

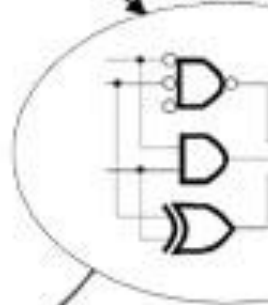
## VHDL Source Code

```
entity leddcd is
  port(
    d: in std_logic_vector(3 downto 0);
    s: out std_logic_vector(6 downto 0);
  );
end;

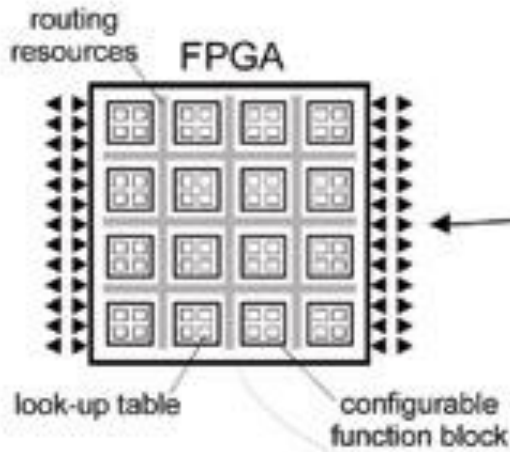
architecture leddcd_arch of leddcd is
begin
  s <= "1110111" when d="0000" else
       "0010010" when d="0001" else
       "1101101";
end leddcd_arch;
```

Synthesize

Netlist



Map, Place & Route



Generate Bitstream

Bitstream

```
10101001010
01011010101
01011010010
01010100101
10101010100
11011011010
11010010101
00101100101
01010110100
01100110001
10101010011
```

Download and Test

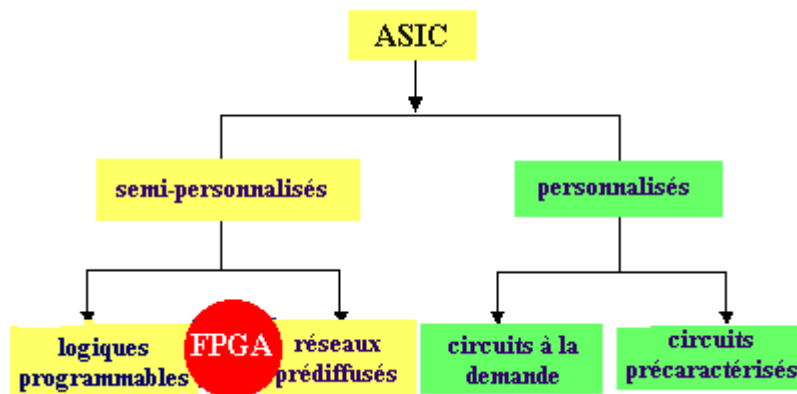
XSA Board



# Les ASICs:

Les circuits ASIC constituent la troisième génération de circuits intégrés qui a vu le jour au début des années 80. En comparaison des circuits intégrés standards et figés proposés par les fabricants, l'ASIC présente une personnalisation de son fonctionnement, selon l'utilisation, accompagnée d'une réduction du temps de développement, d'une augmentation de la densité d'intégration et de la vitesse de fonctionnement. En outre sa personnalisation lui confère un autre avantage industriel, c'est évidemment la confidentialité. Ce concept d'abord développé autour du silicium s'est ensuite étendu à d'autres matériaux pour les applications microondes ou très rapides (GaAs par exemple).

Par définition, les circuits ASIC regroupent tous les circuits dont la fonction peut être personnalisée d'une manière ou d'une autre en vue d'une application spécifique, par opposition aux circuits standards dont la fonction est définie et parfaitement décrite dans le catalogue de composants. Les ASIC peuvent être classés en plusieurs catégories selon leur niveau d'intégration [LUCA94], en fait un ASIC est défini par sa structure de base (réseau programmable, cellule de base, matrice, etc.). Sous le terme ASIC deux familles sont regroupées, les semi-personnalisés et les personnalisés.



*Cliquez sur le graphique pour approfondir un sujet*

**Avantages et inconvénients de l'utilisation d'ASIC:** D'une manière générale l'utilisation d'un ASIC conduit à de nombreux avantages provenant essentiellement de la réduction de la taille des systèmes. Il en ressort :

- Réduction du nombre de composants sur le circuit imprimé. La consommation et l'encombrement s'en trouvent considérablement réduits.
- Le concept ASIC par définition assure une optimisation maximale du circuit à réaliser. Nous disposons alors d'un circuit intégré correspondant réellement à nos propres besoins.
- La personnalisation du circuit donne une confidentialité au concepteur et une protection industrielle.
- Enfin, ce type de composant augmente la complexité du circuit, sa vitesse de fonctionnement et sa fiabilité.
- Dans l'approche des circuits prédiffusés et personnalisés, l'inconvénient majeur réside dans le fait du passage obligatoire chez le fondeur ce qui implique des frais de

développement élevés du circuit. En général le fondeur ne souhaite pas intervenir dans la phase de conception ; sa tâche est de réaliser le composant à partir des masques. Dans le but de réduire les surcoûts dus aux modifications, il s'avère nécessaire d'être rigoureux lors de la phase de développement de telle sorte que le circuit prototype fonctionne dès les premiers essais : c'est réussi dans environ 60% des cas. De plus dans de nombreuses applications, l'utilisateur doit concevoir les programmes de testabilité.

## Cartes de prototypage:

La société Xess offre aux universités ainsi qu'aux étudiants la possibilité d'acheter des cartes de prototypage, à base de FPGA Xilinx, à bas prix.

Toutes leurs cartes se programment via le port parallèle d'un pc. Tous les programmes nécessaires à la programmation des cartes sont fournis. Par contre les programmes nécessaires à la synthèse du langage de description hardware ainsi qu'à la génération du "bitstream" ne sont pas fournis.

Les modèles les plus simples comprennent un microcontrôleur de chez Winbond (**W87C32C**) compatible avec les microcontrôleurs 80C32. Il s'agit d'un microcontrôleur spécialement optimisés pour les fréquences élevées (DC->40MHz). Ce microcontrôleur est également programmable par le port parallèle. Il sert au transfert du "bitstream" du PC vers le FPGA.

Chaque pin du FPGA et du microcontrôleur sont accessibles par la carte est la mettant sur une breadboard. L'oscillateur servant d'horloge au FPGA est programmable et permet de régler une horloge de 100 MHz jusqu'à 48.8 KHz (100 MHz divisé par 2048).

## Conclusion:

### Avantages des FPGAs:

- Délai de conception court.
- Reconfiguration à chaud.
- Prototypage rapide.
- Délai de fabrication court.
- Facilité de test.

### Inconvénients des FPGAs:

- Prix unitaire trop élevé pour la très grande série.
- Performance électriques inférieures aux puces spécialisées (notamment en fréquence).
- Faible taux d'utilisation du circuit.

# Glossaire:

**ASIC:** Application-Specific Integrated Circuits. Les ASIC fournissent précisément la fonctionnalité nécessaire pour une tâche spécifique. Les ASIC sont conçus pour une tâche bien précise et cela leur permet d'être plus petits, moins chers, plus rapide et de consommer moins de puissance qu'un processeur programmable (tel un Intel© Pentium IV). Une puce graphique faite sur mesure pour PC, par exemple, peut tracer des lignes ou des images 10 à 100 fois plus rapidement qu'un processeur central à usage générique.

**Design asynchrone:** S'oppose au design synchrone. Prohibé par les règles de l'art de la conception des ASIC et des FPGA. Ce manque de synchronisme signifie que la sortie de certains bistables ne change pas sur un flanc montant de l'horloge mais à n'importe quel moment. Dès lors, en fonction du routage, les délais entre cellules étant différents, le fonctionnement du circuit risque de varier lui aussi! Ceci peut être très gênant si des précautions particulières ne sont pas prises. De plus, la compréhension du schéma en est compliquée. Les symptômes d'un circuit asynchrone sont les Gated Clocks, les resets et sets asynchrones, l'horloge mère comme entrée des portes logiques.

**Les circuits semi-personnalisés:** Les semi-personnalisés sont des réseaux prédéfinis de transistors ou de fonctions logiques qui nécessitent une personnalisation de l'utilisateur pour réaliser la fonction désirée. Cette famille comprend :

- les réseaux logiques programmables,
- les réseaux prédiffusés.

**Les réseaux logiques programmables:** Ce composant ne nécessite aucune étape technologique supplémentaire pour être personnalisé. Nous y trouvons les PAL/PLD, ce sont des circuits standards programmables par l'utilisateur grâce à différents outils de développement. La programmation consiste à établir des connexions en imposant un courant supérieur aux courants de fonctionnement normaux (claquage de fusibles ou de jonctions). Les circuits logiques programmables incluent un grand nombre de solutions, toutes basées sur des variantes de l'architecture des portes ET OU.

Nous y trouvons :

- PAL (Programmable Array Logic) matrice ET programmable, matrice OU figée),
- PLA (Programmable Logic Array) matrice ET ou matrice OU programmable,
- EPLD (Erasable PLD) effaçables par rayons ultraviolet, ils peuvent être reprogrammer,
- EEPLD (Electrically Erasable PLD) programmables et effaçables électriquement, ils peuvent être reprogrammés sur site. Les limites de l'architecture du PLD résident dans le nombre de bascules, le nombre de signaux d'entrées/sorties, la rigidité du plan logique ET OU et des interconnexions. Précisons que ces composants très souples d'emploi sont limités à des fonctions numériques et adaptés à des productions de petites séries et ne présentent aucune garantie quant à la confidentialité.



**Fonction combinatoire:** sa sortie ne dépend que des états présents de ses entrées et non des ses états passés.

**Fonction séquentielle:** ses sorties dépendent à la fois des états présents et des états passés de ses entrées.

**Les prédifusés:** Les réseaux prédifusés sont des circuits partiellement préfabriqués. L'ensemble des éléments (transistors, diodes, résistances, capacités, etc.) est déjà implanté sur le circuit suivant une certaine topologie, mais les éléments ne sont pas connectés entre eux (sauf au niveau diffusion). La réalisation des connexions dans le but de définir la fonction souhaitée est la tâche du concepteur, pour cela il dispose de bibliothèques de macrocellules et d'outils logiciels d'aide à la conception. A partir de cette liste d'interconnexions (netlist) le fondeur n'aura que quelques étapes technologiques à effectuer pour achever le circuit, c'est à dire le dépôt d'une ou plusieurs couches de métallisation.

Cette technique est intéressante sur le plan de la conception et de la fabrication, par contre elle présente l'inconvénient de ne pas permettre une optimisation en terme de densité de composants puisque les éléments de base sont préimplantés et pas forcément utilisés et que leur positionnement a priori n'est pas forcément optimal pour le but recherché.

**Les circuits personnalisés:** Ce sont des circuits non préfabriqués. Pour chaque application on optimise le circuit intégré, ce qui conduit à la création de son propre composant. Cette famille comprend :

- les circuits à la demande
- les circuits précaractérisés

**Les circuits à la demande:** Les solutions circuit à la demande (qu'on appelle full custom en anglais) présentent l'avantage d'autoriser une meilleure optimisation duplacement puisque celui-ci n'est pas prédéfini. On dispose d'une bibliothèque de modèles mathématiques de comportement et via un "compilateur de silicium" logiciel très sophistiqué on peut concevoir toute l'architecture du circuit en faire une validation logicielle (simulation logique) puis dans une avant dernière étape en déduire le dessin des divers masques de fabrication.

Toutes les opérations, de la conception à la fabrication, sont effectuées de façon spécifique adaptées aux exigences de l'utilisation. L'ensemble des critères techniques est au choix du concepteur, que ce soit la taille du composant, le nombre de broches, le placement du moindre transistor. C'est l'ASIC le plus optimisé car aucune contrainte ne lui est imposée. Le placement des blocs fonctionnels et le routage des interconnexions, même si ces opérations sont assistées par ordinateur, sont effectuées avec beaucoup plus d'interventions manuelles pour atteindre l'optimisation au niveau de chaque transistor. Cependant, les phases de mise au point sont longues et onéreuses, il va de soi que la rentabilisation des investissements de développement nécessite un fort volume de production.

**Les circuits précaractérisés:** Pour la réalisation de circuits précaractérisés on dispose d'une bibliothèque de circuits élémentaires que le fondeur sait fabriquer et dont il peut garantir les caractéristiques et on les associe pour réaliser le circuit à la demande. Ici encore on utilisera en fabrication des masques personnalisés pour chacune des couches diffusées et des métallisations.

Le concept est très semblable de celui des circuits à la demande. La seule différence réside dans la réalisation du schéma puisque l'on accède à une bibliothèque de cellules prédéfinies générant de très nombreuses fonctions élémentaires ou élaborées. Cette dernière constitue un véritable catalogue dans lequel le concepteur se sert pour constituer son schéma. Il existe trois types de cellules :

- les cellules standards (standard cells) correspondent à la logique classique,
- les mégacellules (megacells) peuvent être des blocs du type microprocesseur, périphérique,
- les cellules compilées (compilable cells) dont les blocs RAM ou ROM. Il est nécessaire de personnaliser complètement la diffusion, et par conséquent de créer tous les masques. Cependant un avantage évident en découle : alors qu'il est impossible avec les prédiffusés d'utiliser à 100% le réseau de cellules ou portes, ce qui se traduit par une perte de silicium, les précaractérisés permettent d'exploiter complètement la surface du circuit.

**VLSI**: Very Large Scale Integration: circuits intégrés à très haut niveau d'intégration. Typiquement, on estime qu'un circuit intégré VLSI comprend entre 10.000 et 99.999 portes. Néanmoins, on utilise aussi cette appellation pour les circuits intégrés comprenant plus de portes.