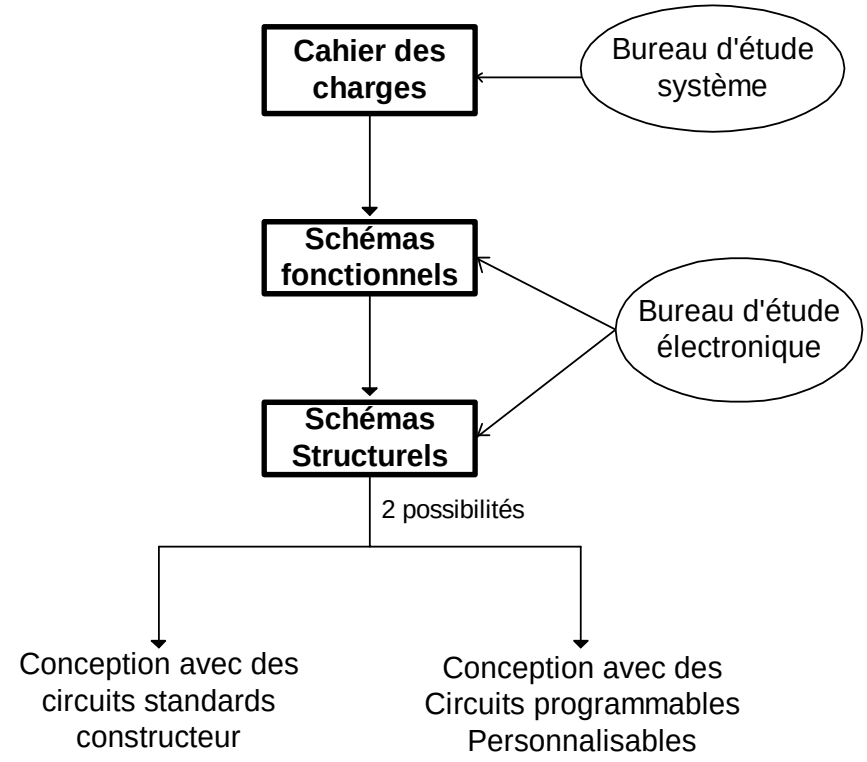
Les circuits programmables

1) Conception



Les circuits programmables

Avantages :

- simulation directe par logiciel sans maquette
- Gain de place sur le circuit imprimé
- Consommation fiabilité et vitesse de fonctionnement plus intéressantes, car les connexions sont réduites au minimum dans un seul boîtier

Piratage presque impossible

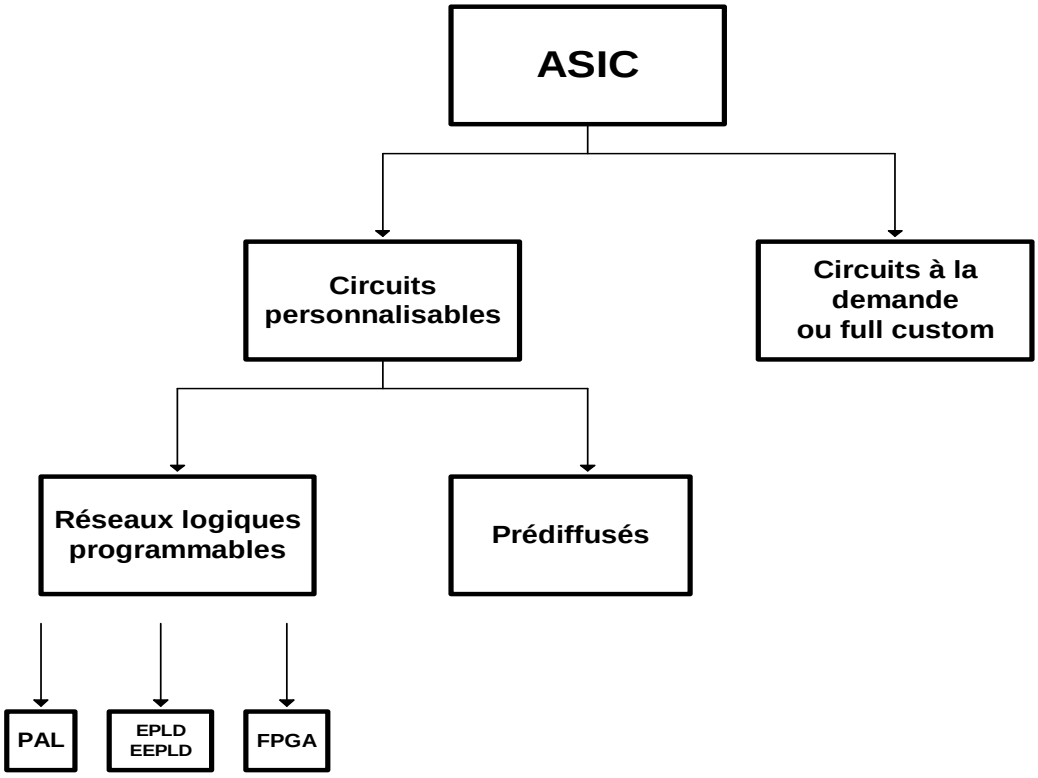
Inconvénients :

- Disposition d'outils informatiques puissants et conviviaux
- Maîtrise des outils

2) Les différents types de circuits programmables

2.1) Famille D'ASIC

Les circuits programmables font partie des ASIC (application specific integrated circuit signifiant circuit intégré spécifique à une application). Ils se partagent en plusieurs familles suivant la complexité de la fonction que l'on désire réaliser (de simples portes logiques jusqu'au microprocesseur).



Circuit à la demande ou full custom :

Le concepteur utilise dans la phase de conception une bibliothèque de cellules fonctionnelles précaractérisées électriquement qu'il va assembler. Le fabricant devra tout intégrer sur le silicium et rendre un circuit testé.

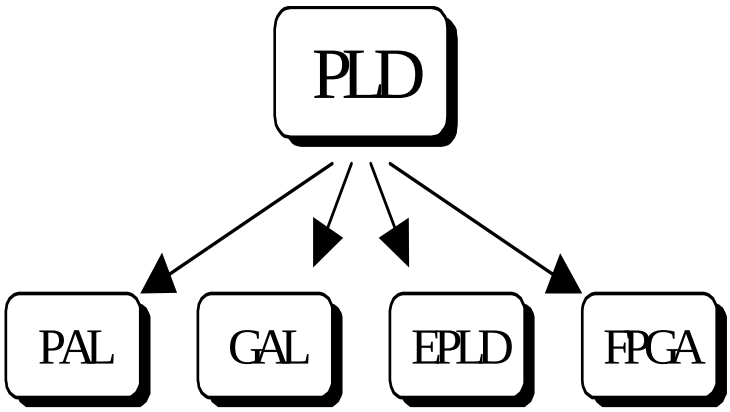
Circuits personnalisables

Prédiffusés

Ce sont des tranches de silicium comportant des réseaux de portes logiques ou des fonctions logiques plus complexes déjà diffusées sur la puce mais non connectées. Le câblage final sera réalisé à la demande du client par une ou plusieurs couches d'aluminium, cette dernière opération n'est réalisable que par la société qui commercialise ce type de produit.

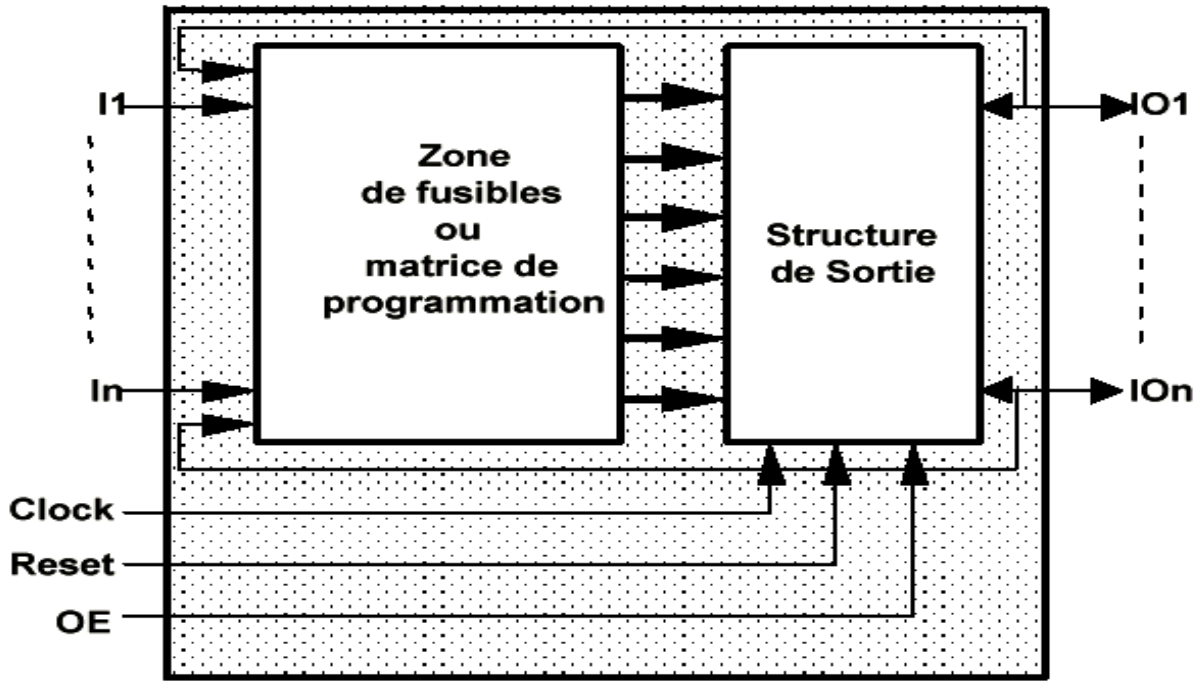
3) Les circuits logiques programmables (PLD)

Les circuits de types réseaux programmables qui permettent de réaliser des fonctions combinatoires et séquentiel . L'utilisateur adapte le produit à son application par une programmation électrique. Cette programmation s'effectue par ces propres soins
Le terme **PLD** regroupe 4 familles de composants :

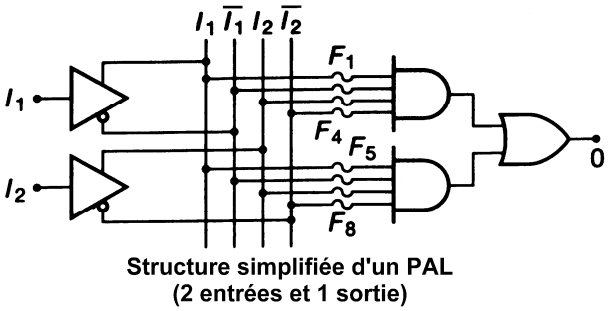


3.1) Les PAL (programmable array logic)

Ce sont les circuits logiques programmables les plus anciens. Les PAL sont programmés par destruction de fusibles. Ils ne sont donc **programmables qu'une fois**, ce qui peut être gênant en phase de développement. Un PAL permet de remplacer jusqu'à 10 boîtiers SSI ou 2 à 3 boîtiers MSI.
D'un point de vue fonctionnel un *P.A.L* est constitué d'une *zone d'entrée de fusibles ou matrice de programmation* et une *structure de sortie non programmable déterminant le type de circuit programmable* voir schéma ci-dessous**.**



3.1.1) Principe d'un PAL

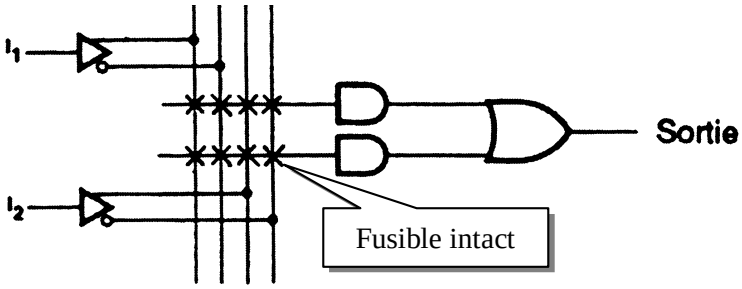


Structure simplifiée d'un PAL
(2 entrées et 1 sortie)

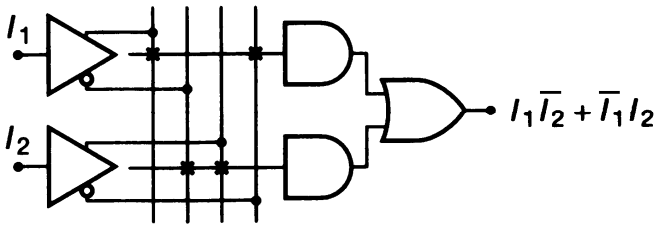
Ce PAL simplifié comporte 2 entrées I_1 et I_2 et une sortie O. Huit fusibles (F_1 à F_8) permettent de réaliser diverses fonctions logiques. La programmation va consister à faire sauter les fusibles nécessaires afin de réaliser la fonction voulue.

Convention de représentation :

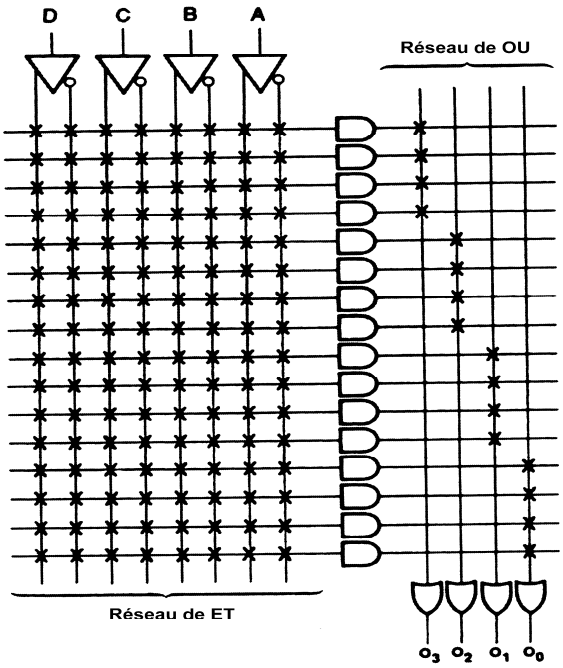
La représentation simplifiée ne montre pas tous les fusibles, les entrées de la porte ET sont regroupées sur une seule ligne. **Une croix représente un fusible intact.**

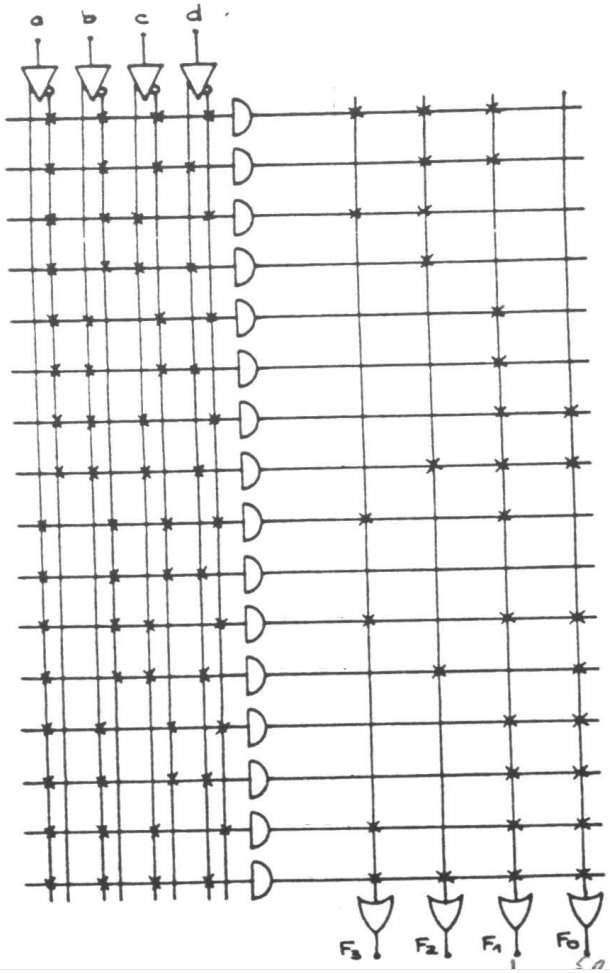


Exemple de programmation d'un PAL
On souhaite réaliser une fonction OU EXCLUSIF : $\mathbf{O = I_1 \oplus I_2 = I_1 \cdot \overline{I_2} + \overline{I_1} \cdot I_2}$



3.1.2) Structure d'un PAL à 4 entrées et 4 sorties





$F_0 = \bar{a}\,b\,c\,\bar{d} + \bar{a}\,b\,c\,d +$

$+ \qquad + \qquad +$

$+ \qquad + \qquad .$

$F_1 =$

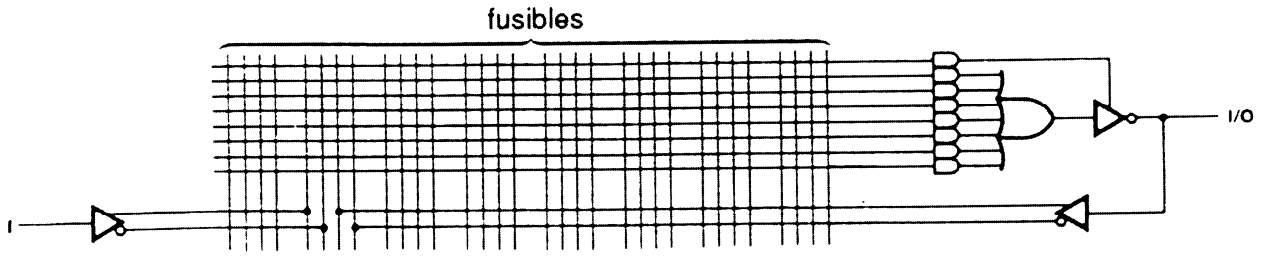
$F_2 =$

$F_3 =$

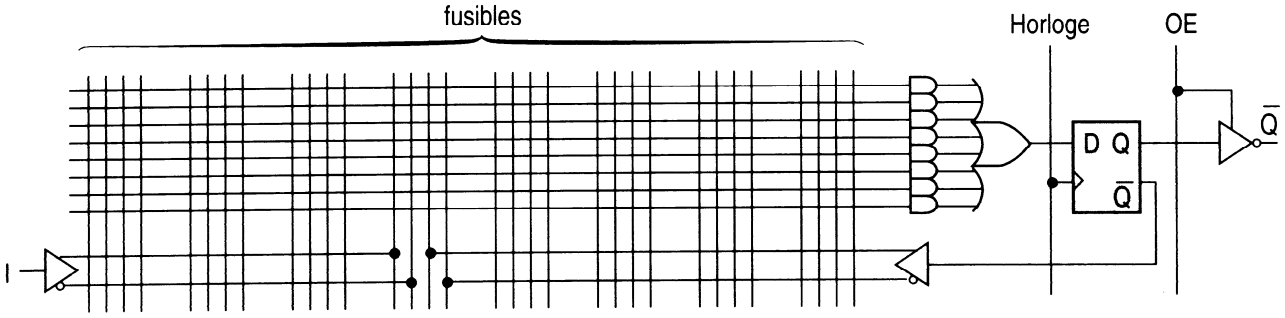
3.1.3) Les différents types d'entrées / sorties

On distingue 3 principes utilisés pour les sorties. Selon le modèle, un ou plusieurs types de sorties peuvent être utilisés sur un même PAL.

☞ **Entrées / Sorties combinatoires :**
Ces sorties 3 états sont rebouclées vers la matrice de fusibles. Une sortie peut donc servir de variable intermédiaire. En mode haute impédance (la sortie étant inhibée), on peut utiliser une broche de sortie comme étant une entrée. On parle alors d'entrée / sortie (I/O).

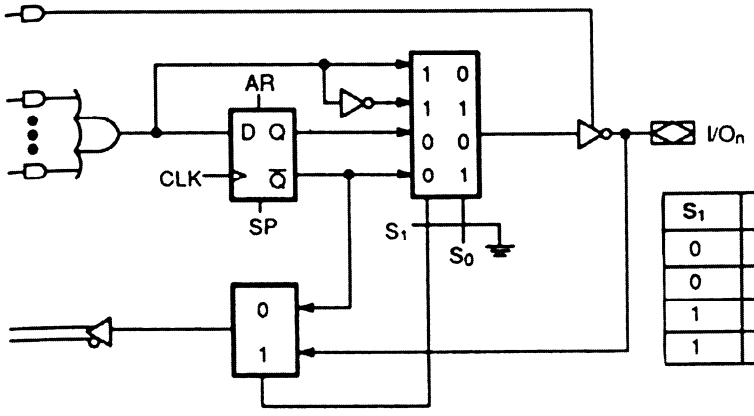


☞ **Sorties à registres :**
Ces sorties utilisent une bascule D qui permet la logique séquentielle. Par contre, une sortie à



registre ne peut pas être utilisée comme entrée.

☞ **Entrées / Sorties des PAL versatiles (VPAL) :**
Le bloc de sortie des PAL versatiles permet de configurer (par programmation) le mode



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

d'utilisation de la broche de sortie :
On remarque sur les 3 schémas que l'on peut reboucler la variable de sortie vers les réseaux combinatoires.
A partir de la documentation du PALCE22V10, remplir les caractéristiques essentielles du circuit dans le tableau suivant:

Type de sortie	
Nombre de macrocellules en sortie	
Nombre de portes logiques ET	
Nombre de portes logiques ou	
Nombre d'entrées pour les portes logiques ET	
Technologie du circuit (CMOS, TTL, ECL --)	
Nombres d'entrées maximales pour les portes ou	

Définir le terme 44 * 132 pour le réseau de portes et

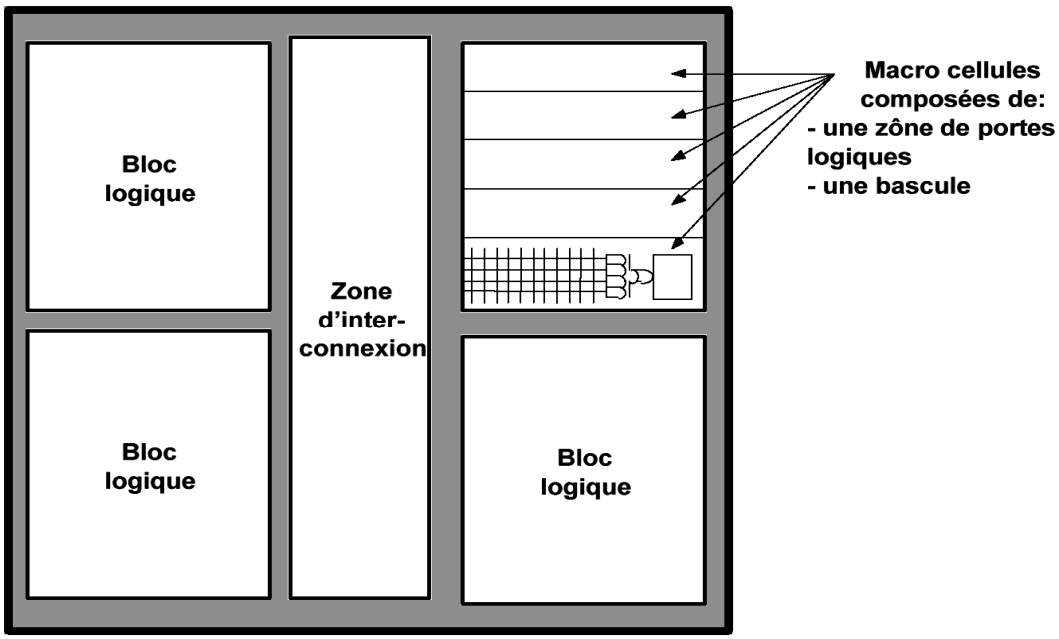
3.2) Les GAL

L'appellation GAL est une marque déposée de LATTICE SEMICONDUCTOR qui a été la première société à proposer sur le marché ce type de produits. D'autres marques proposent des équivalents (compatibles) commercialisés sous le nom de PAL CMOS, E2PAL ou encore PAL EECMOS.
Les GAL sont des **PAL effaçables électriquement**, qui utilisent la technologie CMOS.
La quasi totalité des GALs sont du type EEPLD et permettent donc une programmation et un effacement électrique.
Finis donc les composants gaspillés pour une erreur dans le développement, le GAL est là pour contourner agréablement ce problème.
Le revers de la médaille est toutefois le prix, puisqu'un GAL coûte en moyenne quatre fois plus cher qu'un PAL de possibilités équivalentes.
Il est également important de noter que le cycle de programmation / effacement n'est pas infini, mais de l'ordre de 100 !.

3.3) LES C.P.L.D.s (Complex Programmable Logic Device).

Ces circuits ont une capacité en nombre de portes et en possibilités de configuration très supérieure à celle des PALs. Leurs architectures sont basées sur celles des PALS. Un CPLD c'est l'équivalent de plusieurs PALs mis dans le même circuit associé à une zone d'interconnexion. Le nombres de portes peut varier entre 100 et 100 000 portes logiques et entre 16 et 1000 bascules voir plus.

Structure générale d'un CPLD. Structure générale d'un CPLD.



le cycle de programmation / effacement est de l'ordre de 1000 !.
A Partir de la documentation constructeur du CY37032, donner les caractéristiques essentielles du circuit en remplissant le tableau suivant :

Nombre de bloc logique	
Nombre de macrocellules d'un bloc logique	
Nombre de portes logiques ET	
Nombre de portes logiques ou	
Nombre d'entrées pour les portes logiques ET	
Technologie du circuit (CMOS, TTL, ECL --)	
Nombres de cycle de reprogrammation minimum	

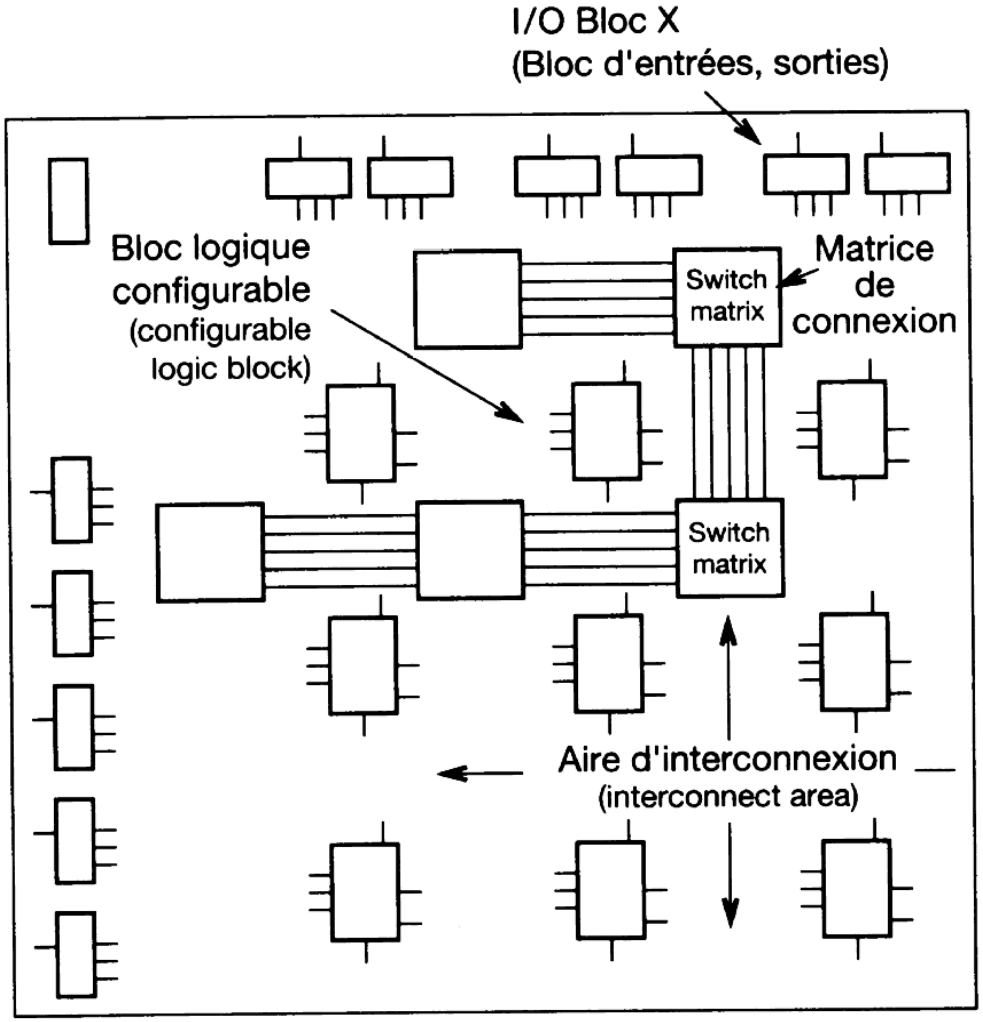
3.4) LES F.P.G.A.s (Fields Programmable Gate Array).

les FPGAs à la différence des CPLDs sont assimilables à des A.S.I.C. (Application Specific Integrated Circuit) programmables par l'utilisateur.
La puissance de ces circuits est telle qu'ils peuvent être composés de plusieurs milliers voire 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.

Chaque **bloc configurable** est constituée de **réseau de portes logiques ou des fonctions logiques complexes (compteur, multiplexeur etc...).**

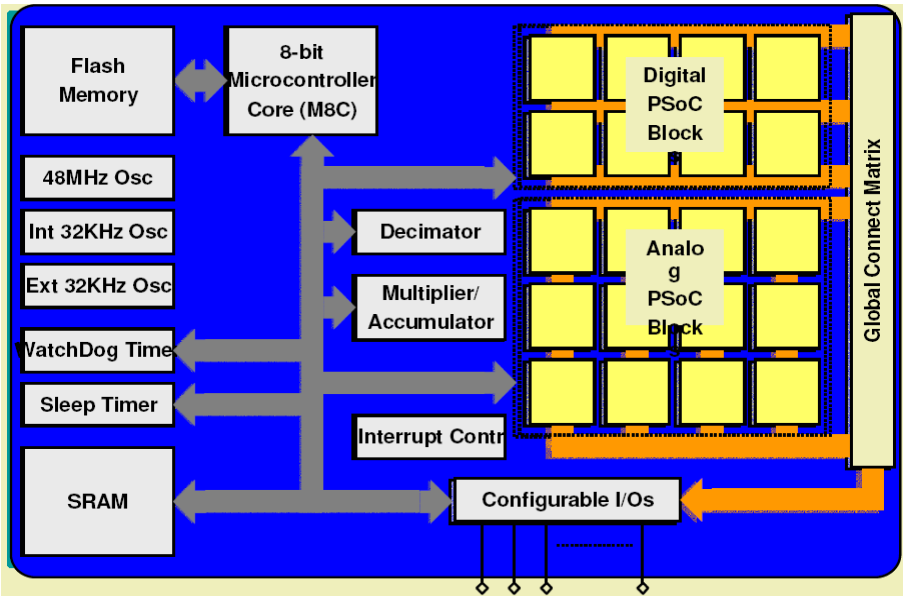
Par une simple programmation électrique (d'une mémoire SRAM) on peut :
- configurer un bloc logique ou plusieurs
- interconnecter entre eux les blocs grâce à une matrice de connexion

on peut aussi électriquement déprogrammer ce que l'on avait programmé

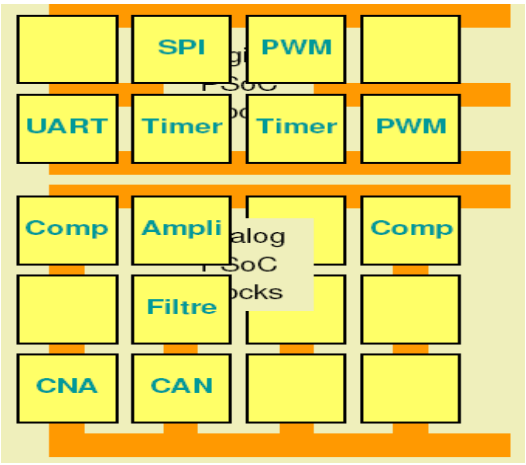


3.5 Les PSOC (Programmable system on chip)

C'est Un composant associant un **microcontrôleur** et des **fonctions logiques et analogiques** configurables



L'architecture intègre des blocs numériques et analogiques banalisés, configurables au gré de l'utilisateur : UART, SPI, timer, PWM, SPI, amplis op.,filtres, convertisseurs, etc.
un seul modèle peut répondre à une multitude d'applications. Le microcontrôleur ne se retrouve pas encombré de fonctions superflues Les blocs sont configurables et reconfigurables en fonctionnement, permettant d'adapter, de modifier, d'optimiser dynamiquement les fonctions réalisées

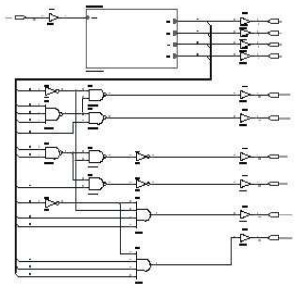


	CY8C29xxx	**CY8C27xxx**	CY8C24xxx	CY8C21xxx
Program Memory (Kbytes)	32	**16**	4	4/8
Data Memory (Bytes)	2K	**256**	256	256/512
Digital Blocs	16	**8**	4	4
Analog Blocs	12	**12**	6	2
I/O pins	24-64	**6-44**	6-24	12-28

4) Comparaison entre CPLD et FPGA

	Avantages	Inconvénients
CPLD	Non volative	Fan-in important
	Compteur et machines d'états rapides	Les ressources de routage sont faibles
	Logique combinatoire ou de controle	Fonction réclamant peu de routage
	Les temps d'arrivés sont déterministes	
FPGA	Architecture microprogrammée, DSP	Les temps d'arrivés dépendent du routage
	Systèmes séquentiels	Reconfiguration par SRAM
	Densité Intégration élevée	Nécessite une PROM (non volatile)

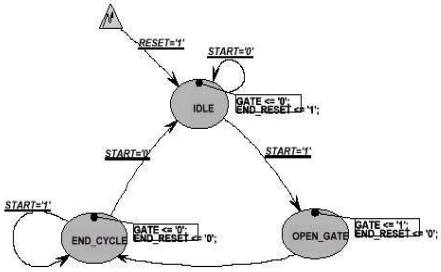
5) Schéma fonctionnel d'un outil de développement de FPGA /CPLD voir PAL.



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

entity CNT_4B is
    port (
        CLK: in STD_LOGIC;
        RESET: in STD_LOGIC;
        ENABLE: in STD_LOGIC;

        FULL: out STD_LOGIC;
        Q: out STD_LOGIC_VECTOR
(3 downto 0)
    );
```



Entrée schématique Viewlogic, Orcad,

Entrée syntaxique langages: - VHDL ou ABEL

Diagramme d'états Statecad

VHDL *VHDL* *VHDL*

Vérification des erreurs: de schéma et de syntaxe

STILMULI ou TESTBENCH

Simulation comportementale

VHDL

SYNTHETISEUR

Il convertit en fonction du ciruit cible (CPLD ou FPGA: suivant le modèle choisit) le projet en portes logiques et bascules de bases.
De plus on peut lui spécifier des contraites de technologies, par exemple la vitesse de fonctionnement

VITAL, VHDL

STILMULI ou TESTBENCH

Simulation après synthèse

SYNTHETISEURS CONNUS:

-FPGA Expres (Synosys).
-EXEMPLAR (Exemplar).
-SYNPLICITY (Synplicity).
- WARP (CYPRESS)

EDIF, XNF

Tient compte des temps de propagation

Outils de placement et routage interne au circuit. Cet outil est propre à chaque fabriquant

Optimisation Placement / routage dans le circuit choisit

VITAL, VHDL, SDF

STILMULI ou TESTBENCH

Simulation temporelle

JEDEC (.jed)

La programmation du circuit peut se faire soit:
- En utilisant un programmateur.
- En téléchragement le code JEDEC directement sur l'objet technique (JTAG ou ISP)

