

# Etude des circuits logiques programmables

Faite par:  
Imane El Midaoui  
Bouchra Sahel  
Tasnime Amrani Nejjar

Encadré par:  
DR.P Ammour Alae

# Plan:

1/Introduction

2/CPLD

3/FPGA

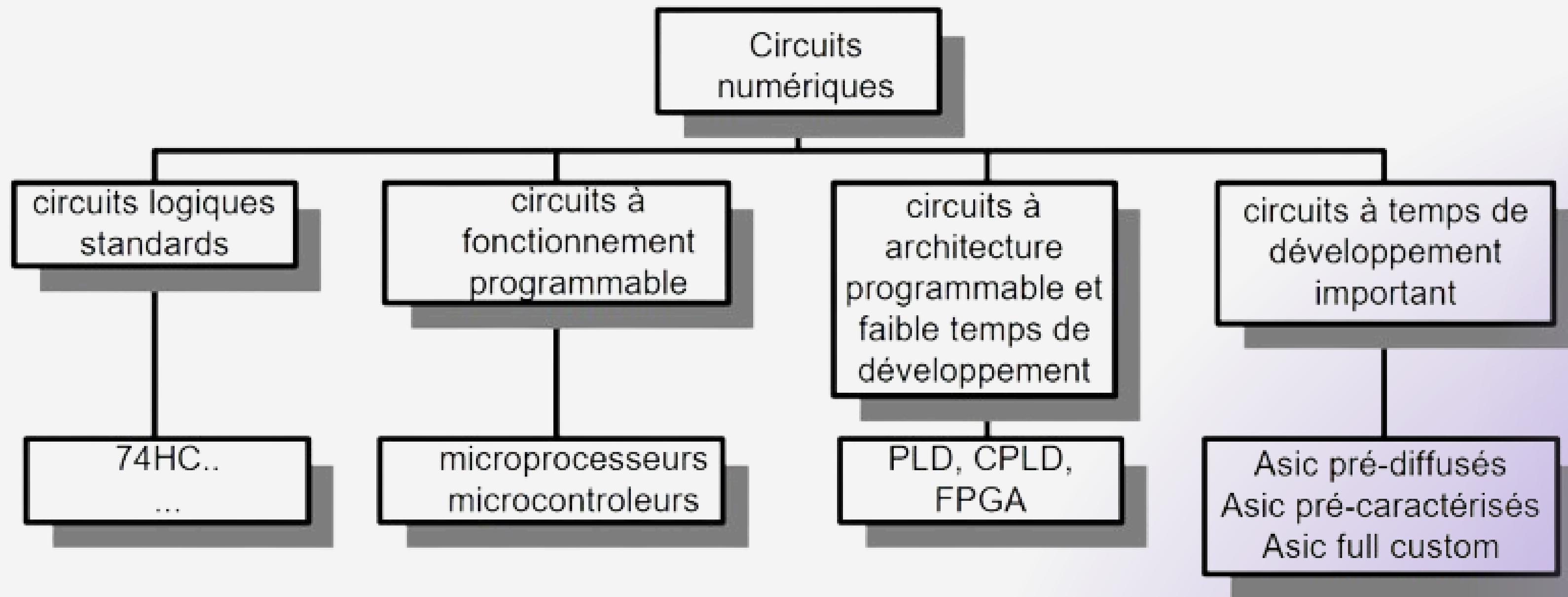
4/FPGA dans le marché

5/Comparaison des CPLD et  
des FPGA

6/Comparaison des  
Microprocesseur, des  
Microcontrôleur et des FPGA

7/VHDL

# Introduction



# CPLD

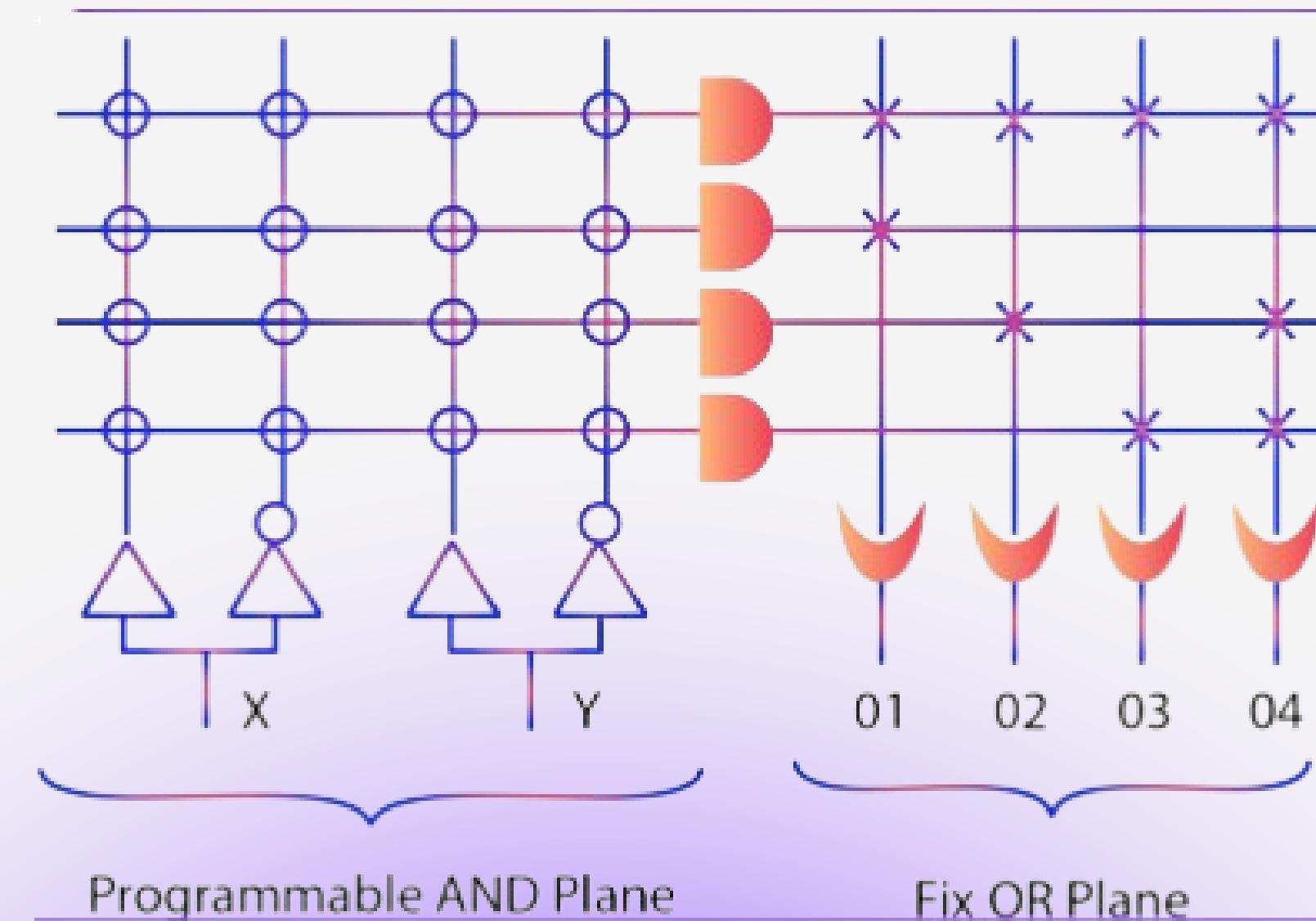
## Définition:

Un CPLD (Complex Programmable Logic Device) est un dispositif logique programmable avancé capable d'implémenter des structures importantes de logique combinatoire et séquentielle programmables sur le terrain.



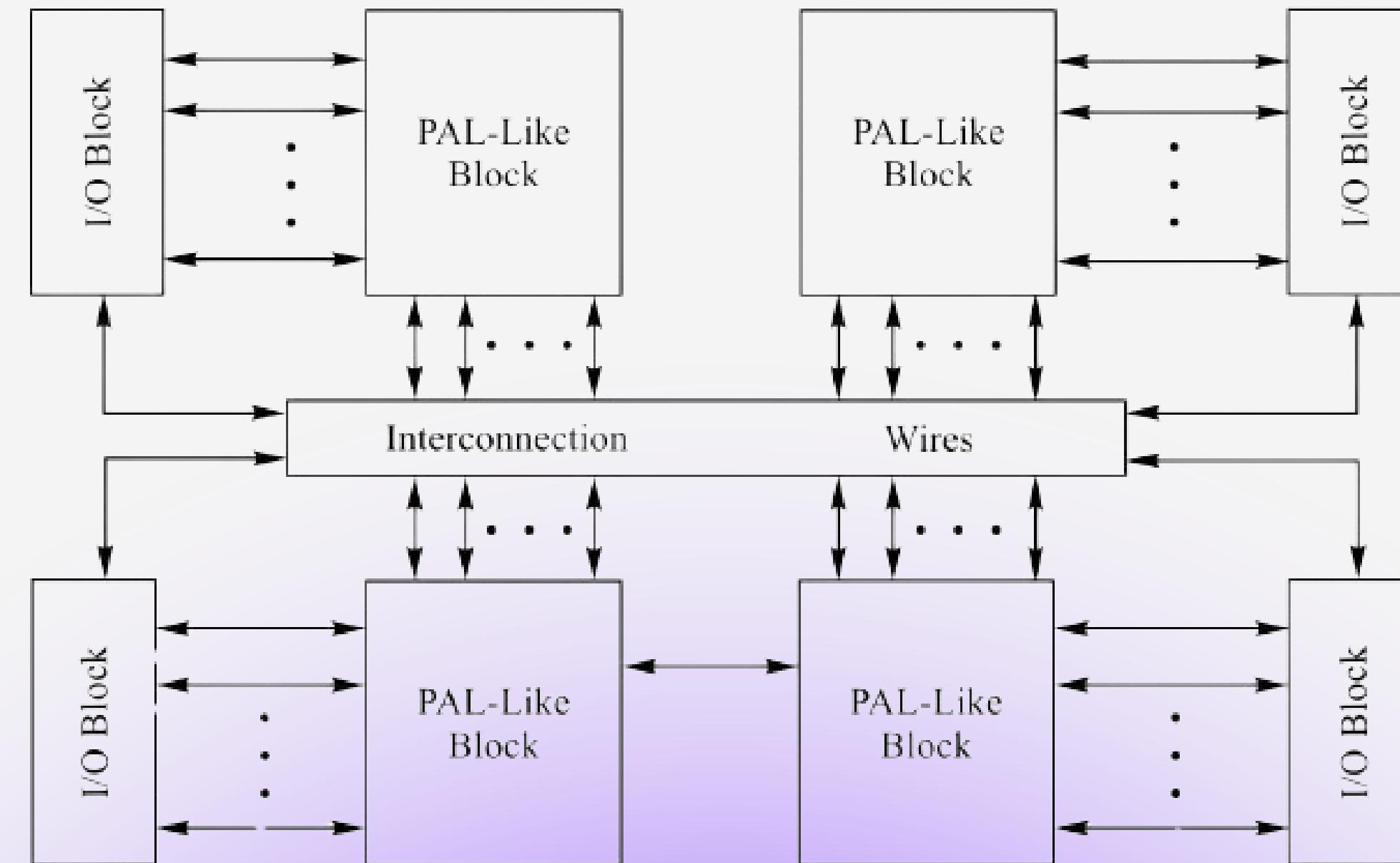
# Rappel

PAL:



# CPLD

## Architecture:



# CPLD

## Avantages:

- Le temps de propagation des signaux sont constants et prédictibles
- une robustesse de la programmation (une fois programmée, le CPLD est instantanément opérationnel à sa mise en marche)
- une faible consommation

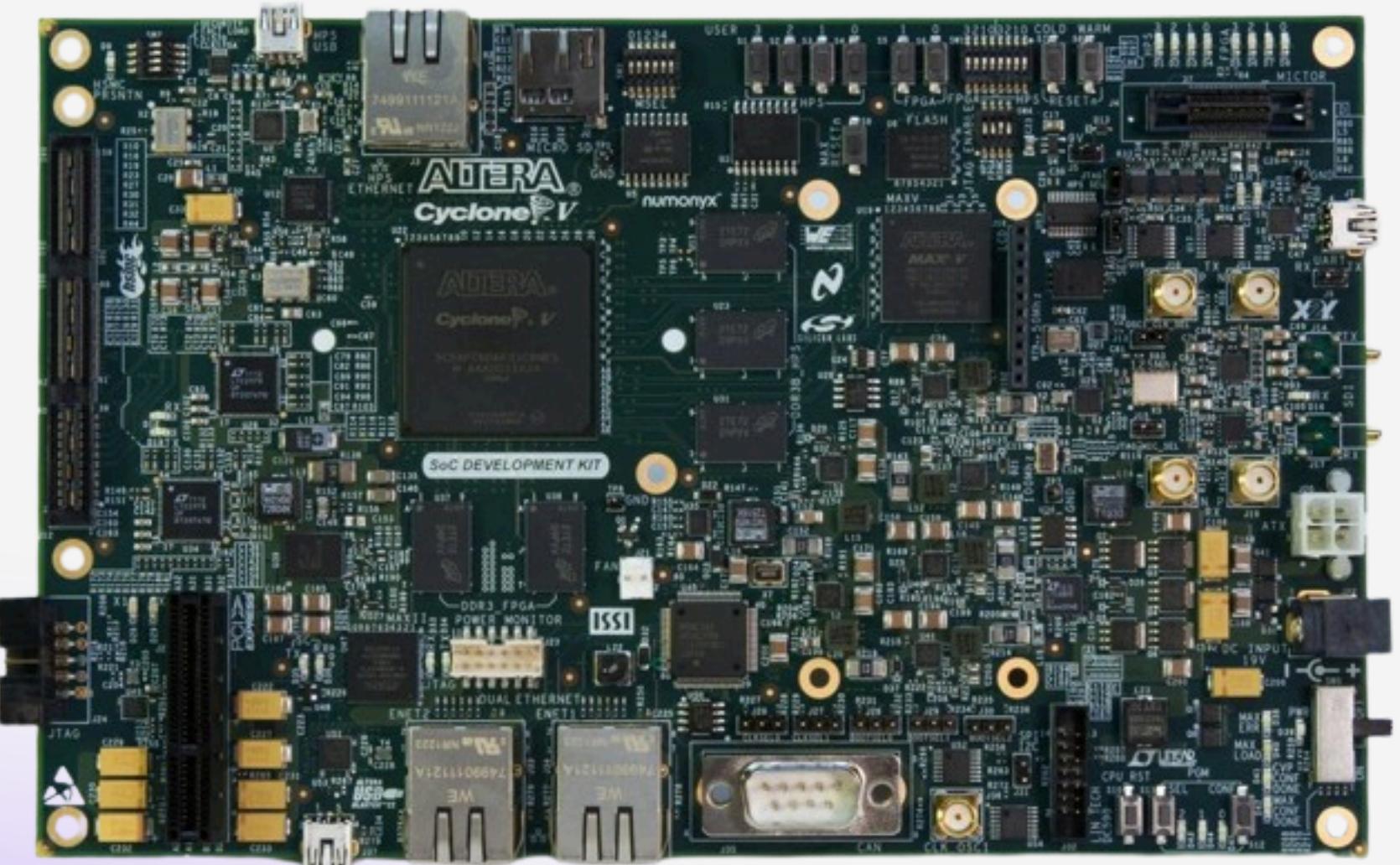
## Inconvénients:

- un nombre de cellules assez limité
- des difficultés à effectuer des traitements complexes
- l'utilisation des ressources n'est pas optimale (tout terme non utilisé dans une équation logique équivaut à des portes perdues), avec des taux d'utilisation d'environ 25 %.

# FPGA

## Définition:

FPGA est l'abréviation de « Field-programmable gate array ». C'est un circuit intégré fait pour être (re)programmé par l'utilisateur après sa fabrication en utilisant un langage informatique spécifique, donc sans modifier le matériel.

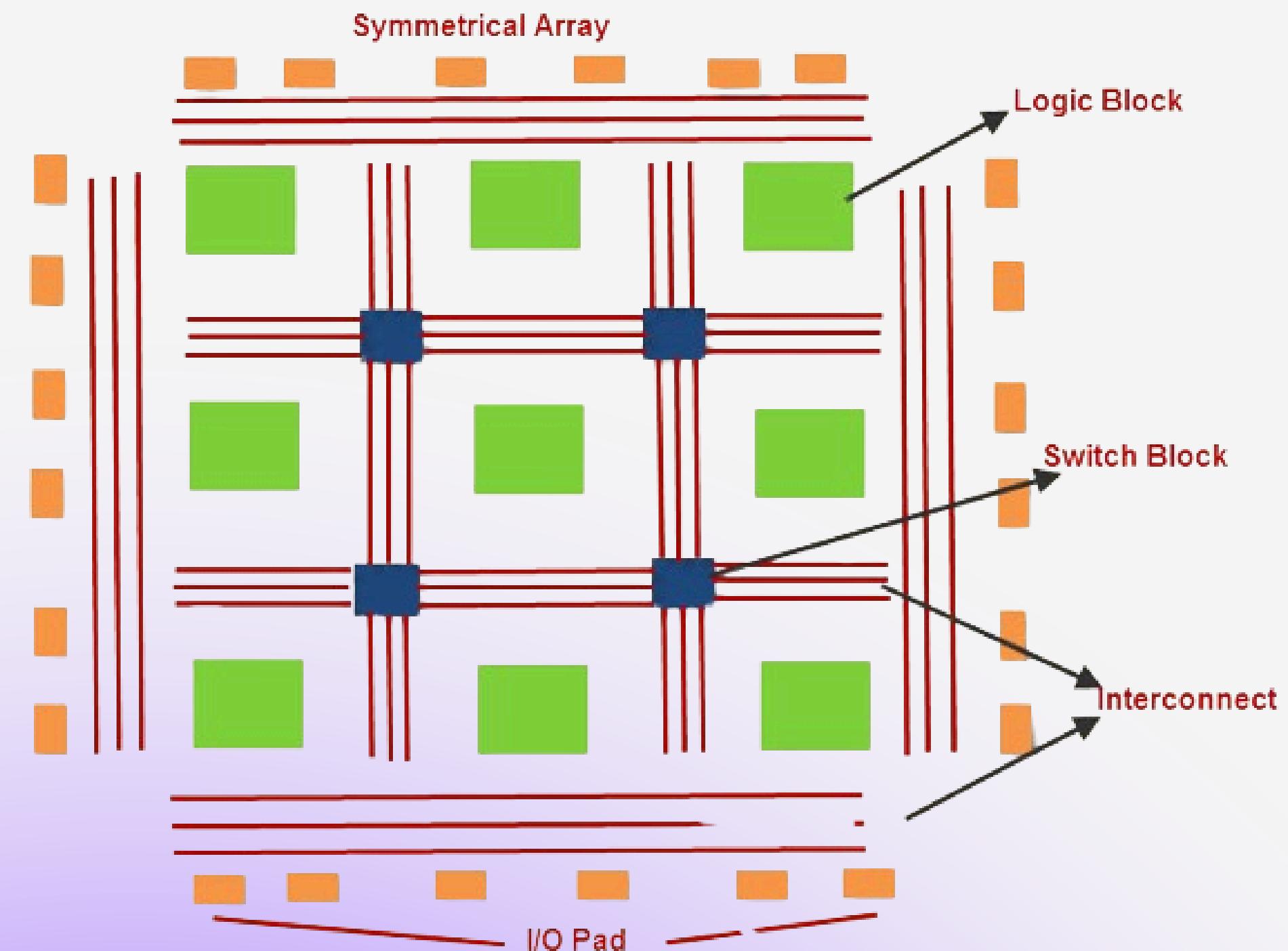


# FPGA

## Architecture:

Les FPGA se composent de :

- Bloc d'I/O
- Bloc logique configurable (CLB)
- Interconnexions
- Matrice de commutation

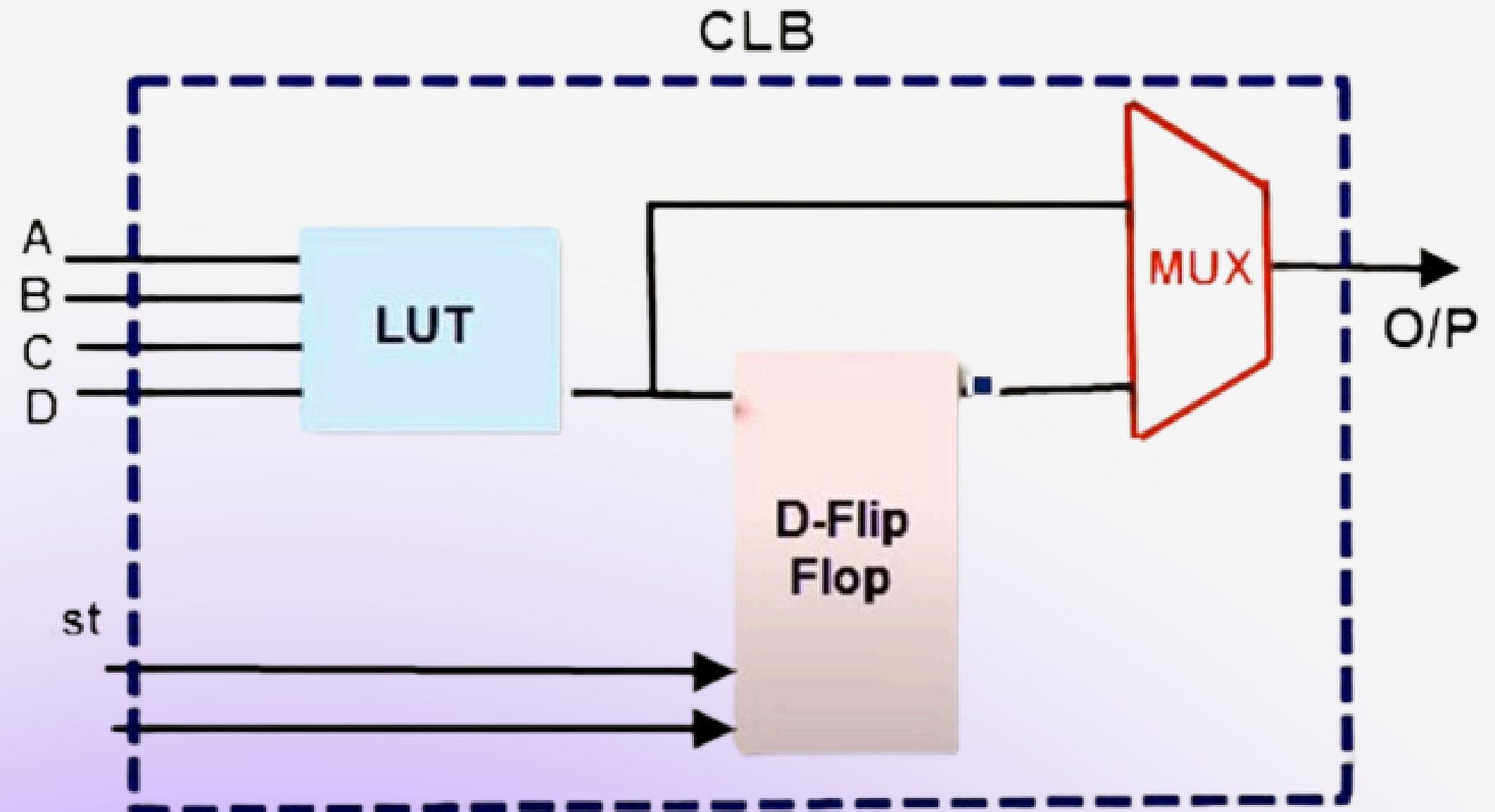


# CLB

## Architecture:

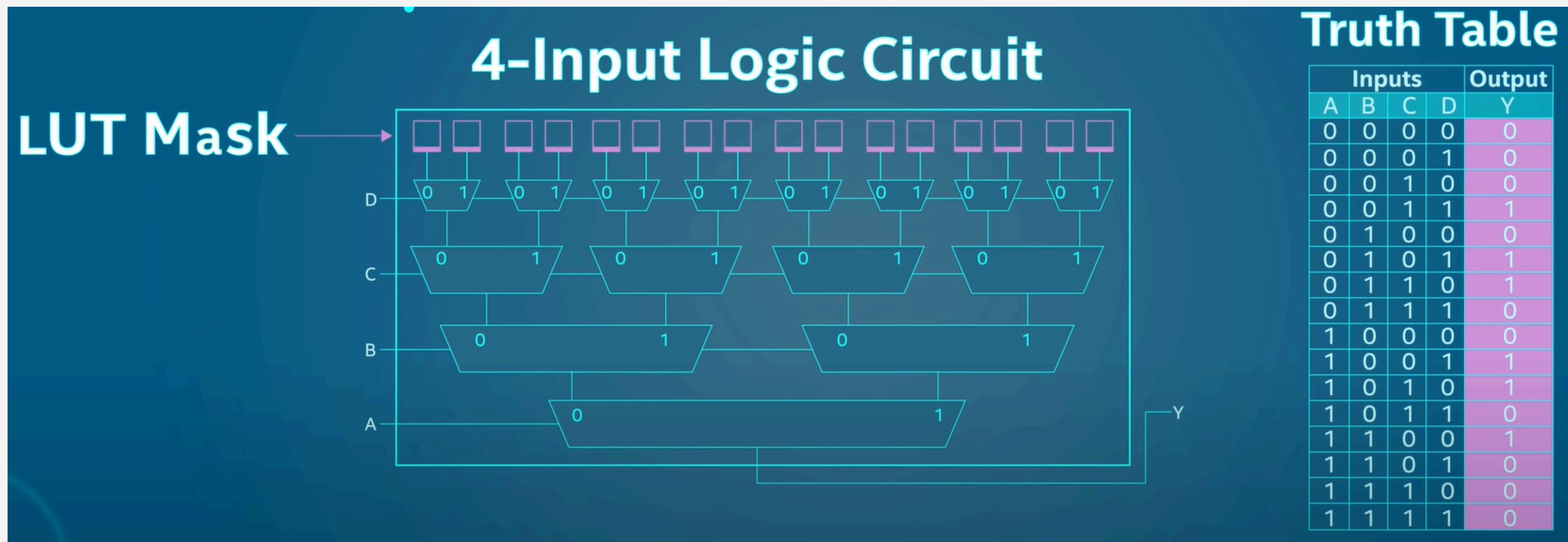
Un CLB se compose de :

- LUT (table de consultation)
- DFF (bascule D)
- Mux (multiplexeur)



# LOOK UP TABLE (LUT)

Architecture:



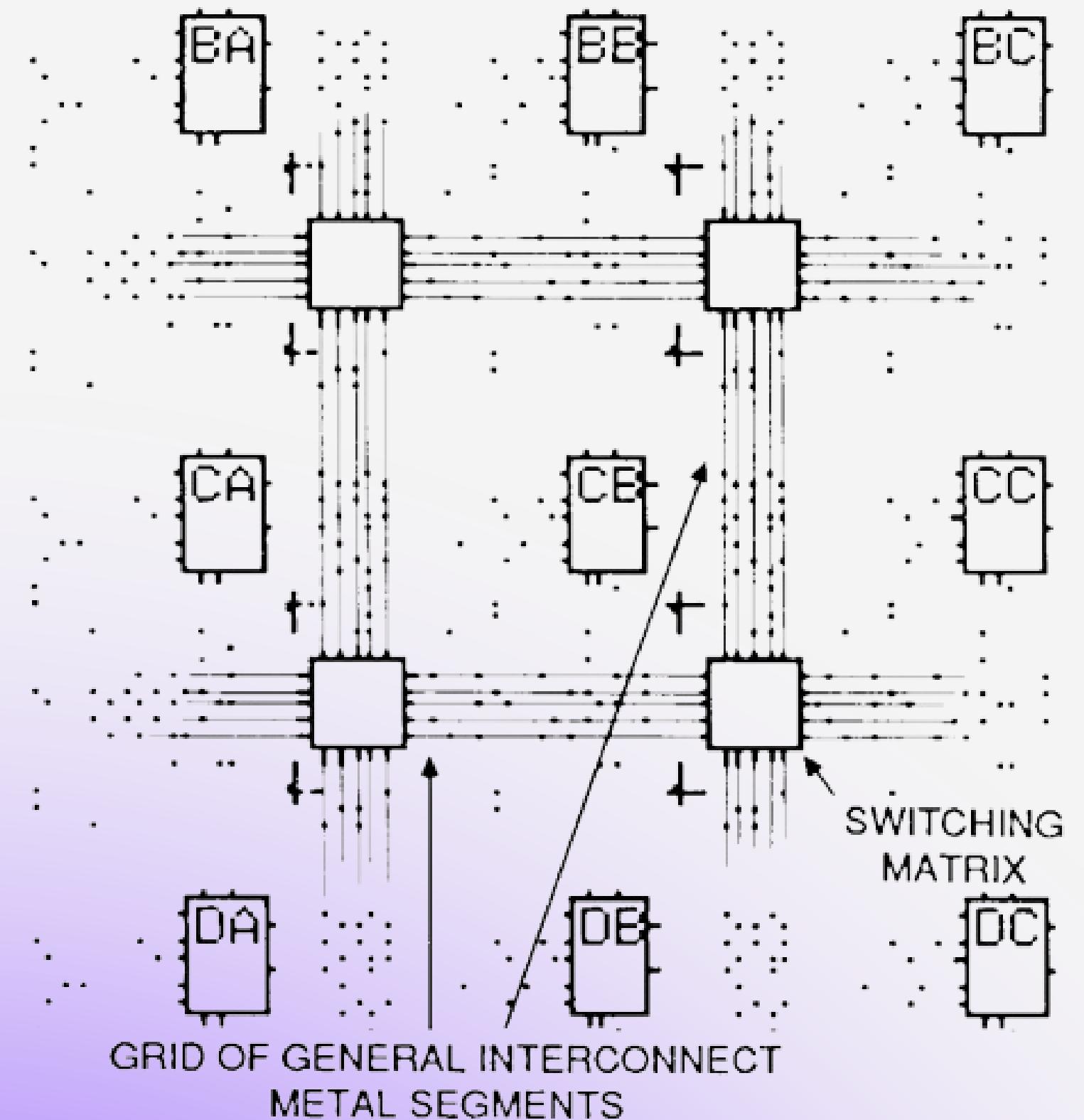
# Interconnexions

se réfèrent aux chemins de connexion qui permettent aux différents blocs logiques et éléments du FPGA de communiquer entre eux. Il existe 3 types:

- interconnexions à usage générale.
- interconnexion directes.
- de longues lignes.

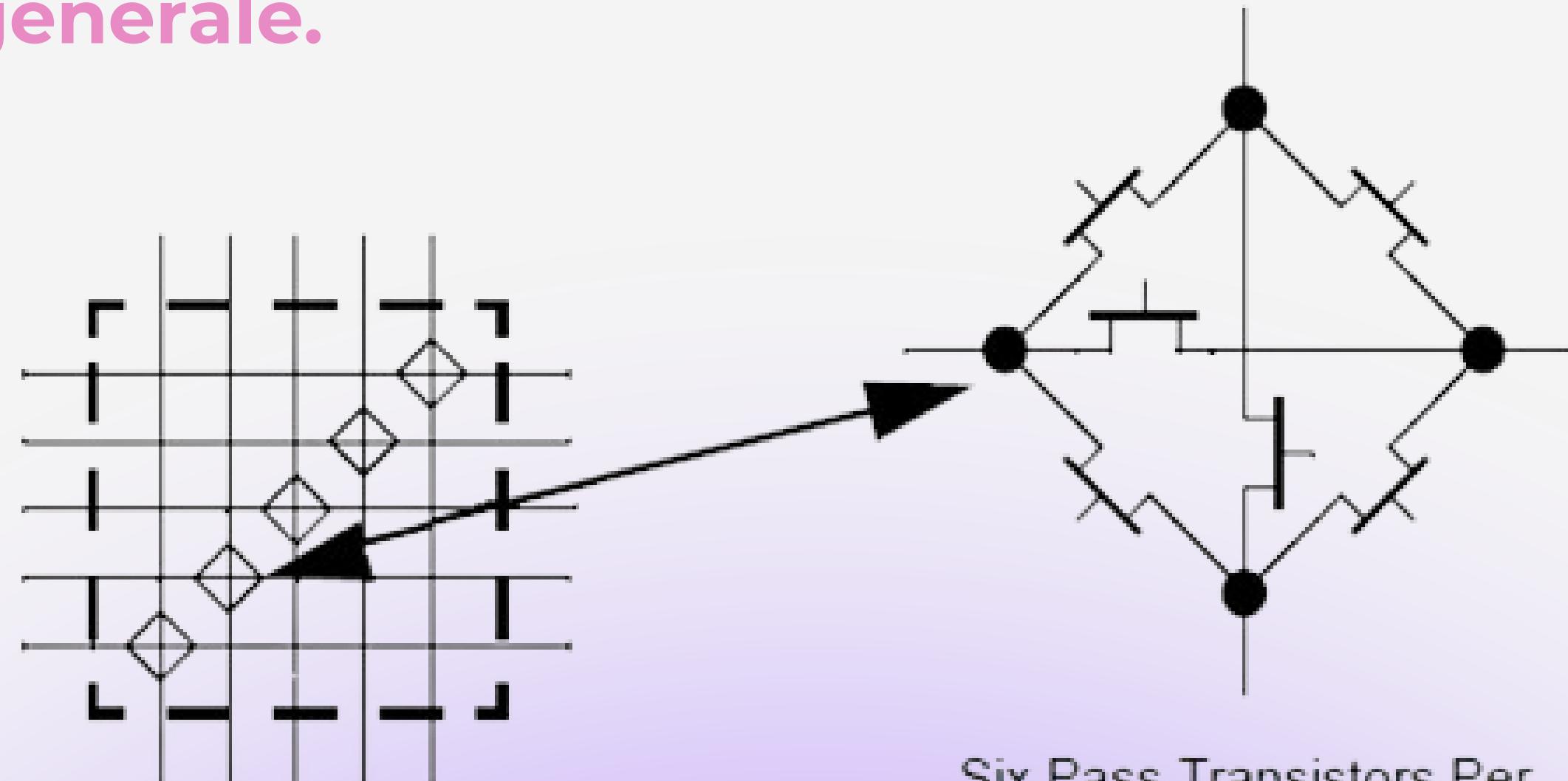
# Interconnexions

## 1. à usage générale.



# Interconnexions

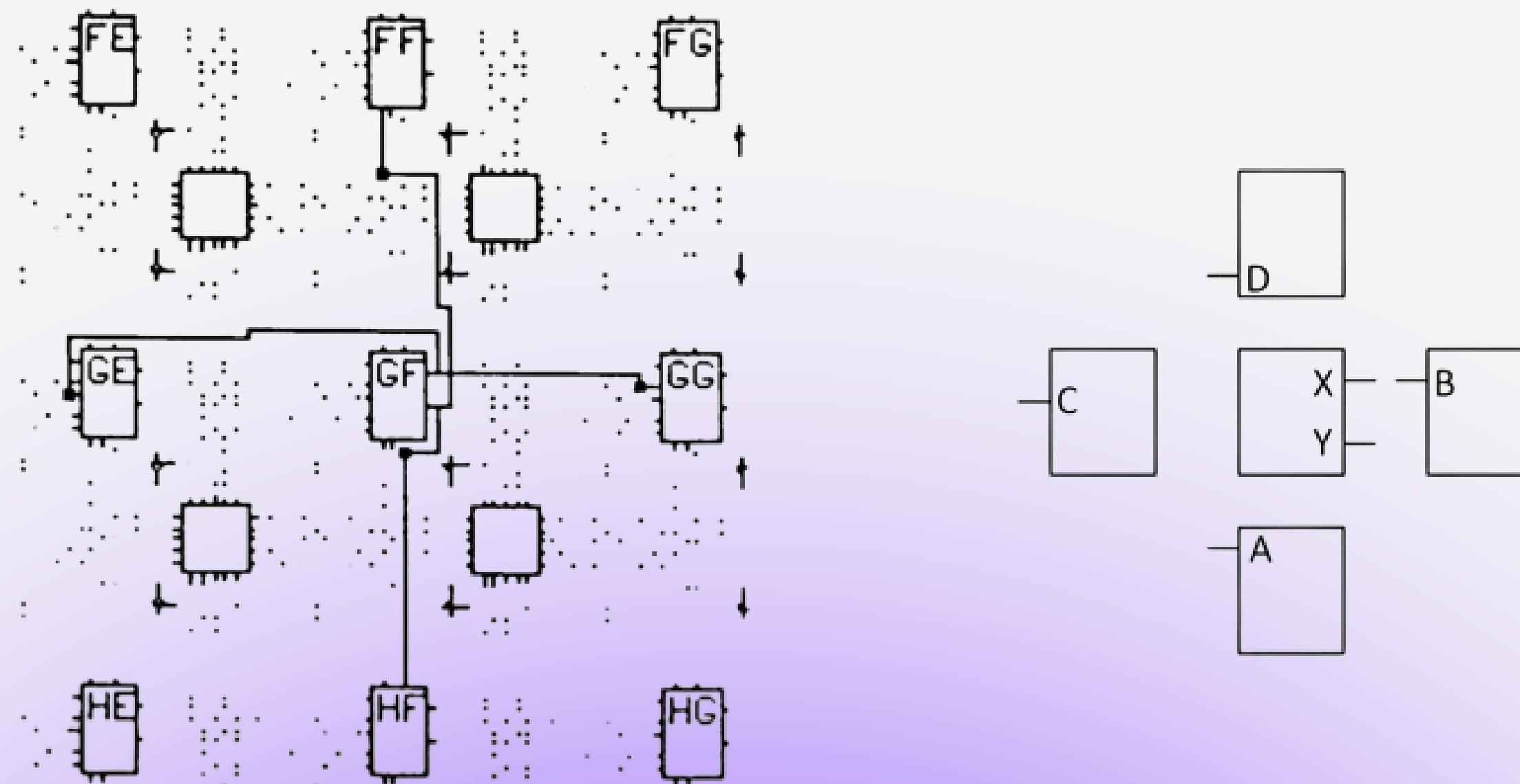
## 1. à usage générale.



Six Pass Transistors Per  
Switch Matrix Interconnect Point

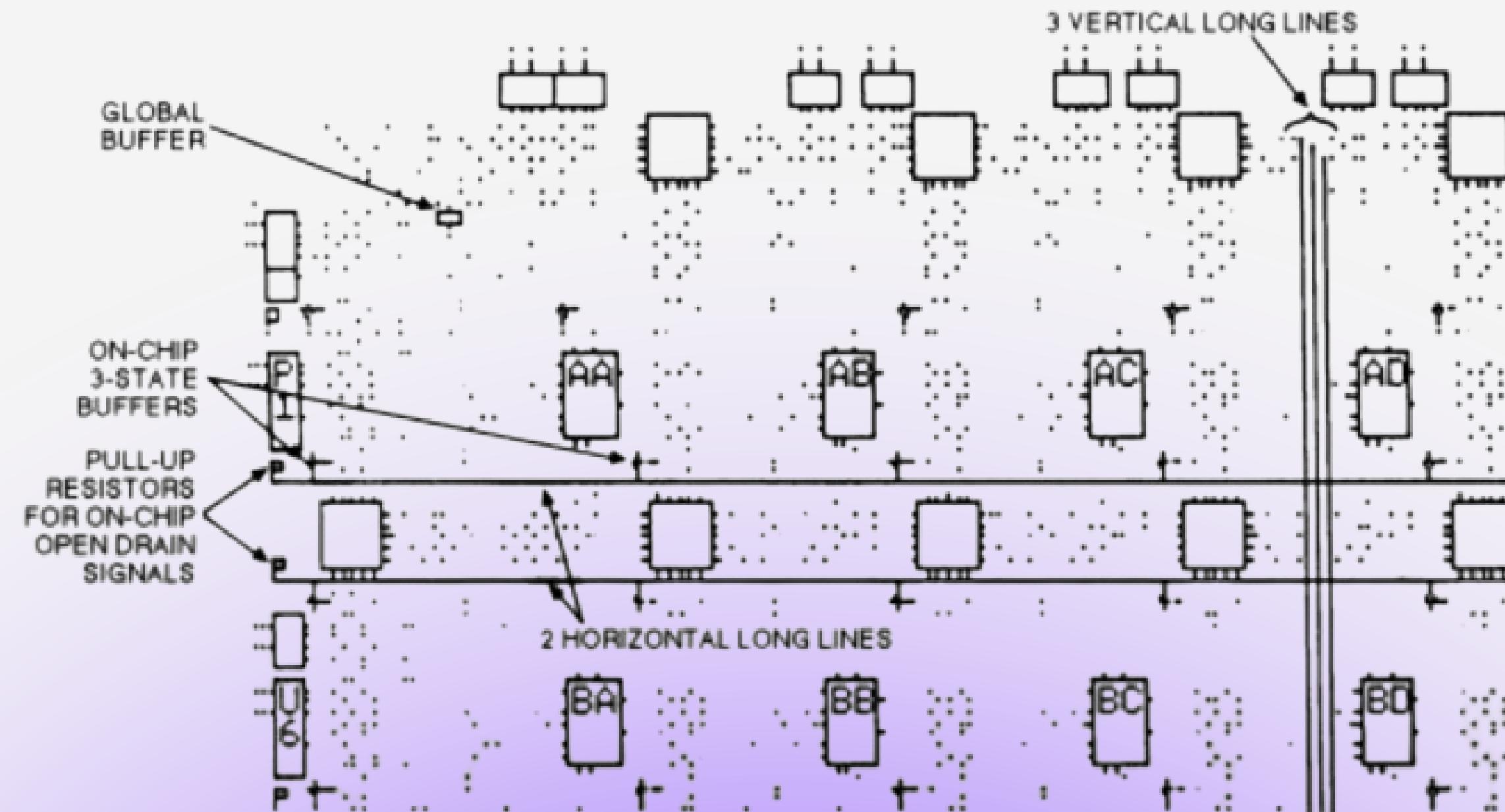
# Interconnexions

## 2. directes.

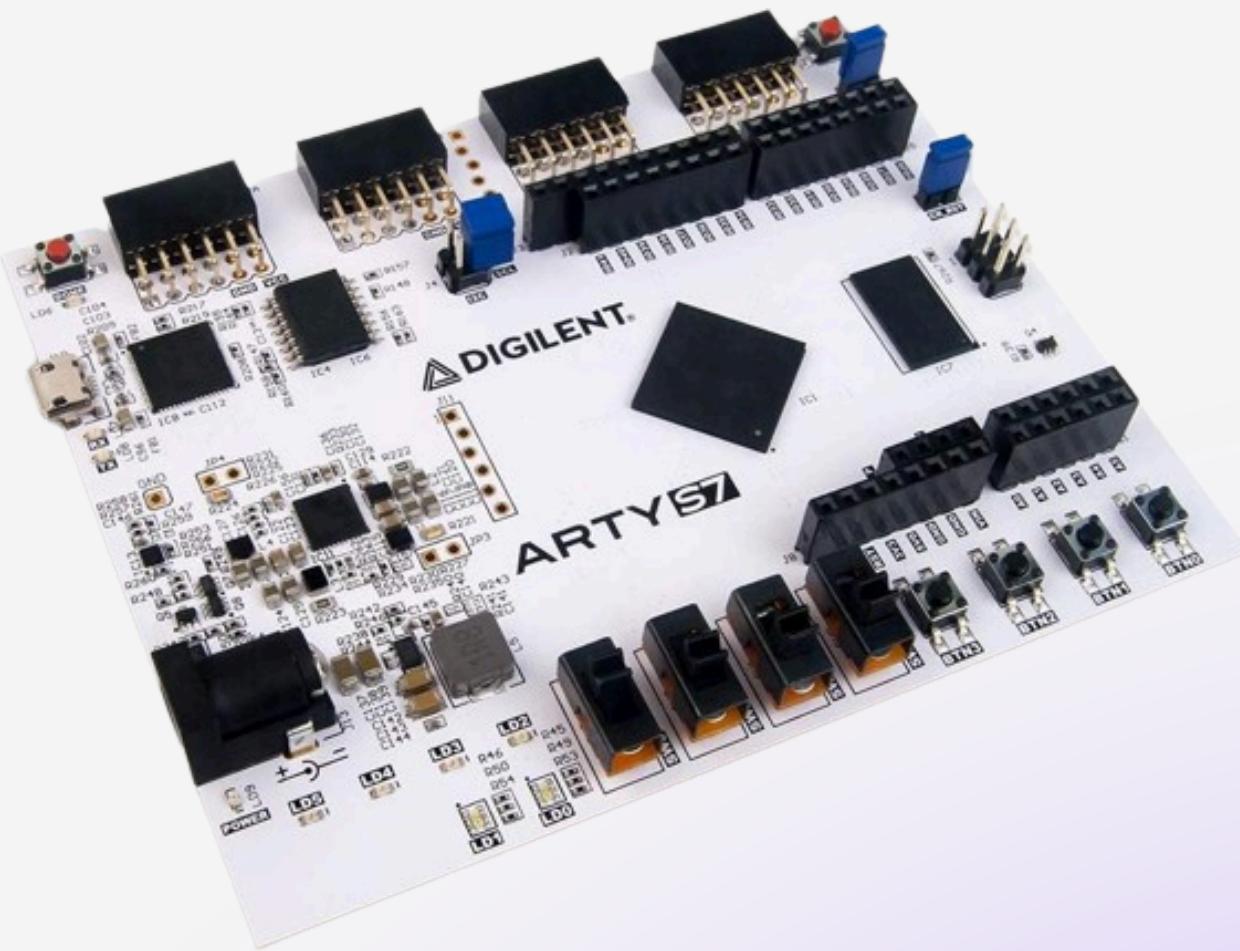


# Interconnexions

## 3. des longues lignes.



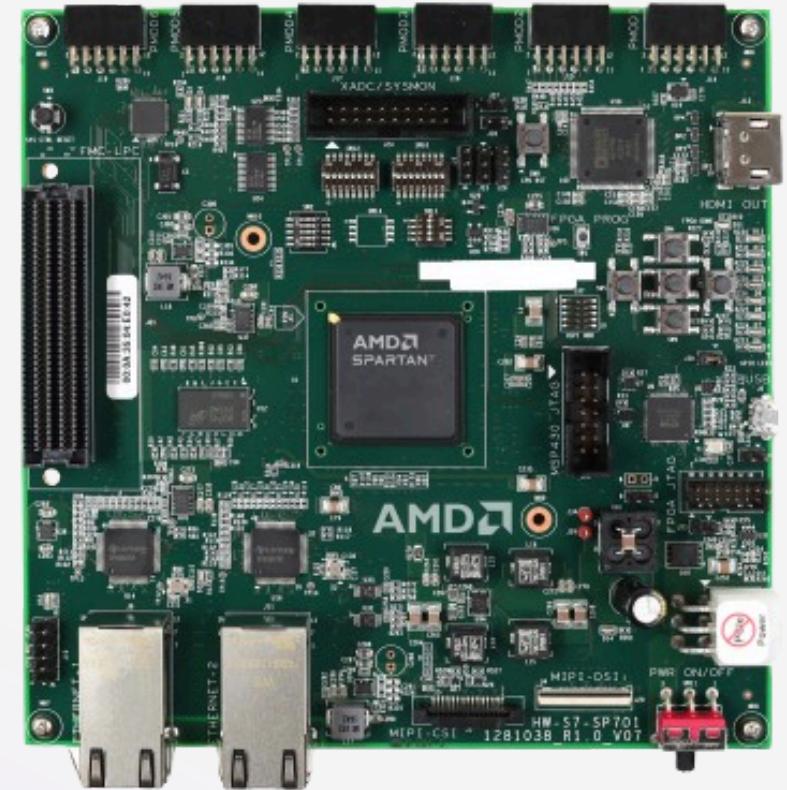
# FPGA dans le marché



Digilent Arty S7: Spartan-7



Intel/Altera Cyclone IV



Xilinx Spartan 7 SP701

# Comparaison des CPLD et des FPGA

	CPLD	FPGA
<b>le nombre de portes</b>	quelques milliers de portes logiques	plusieurs millions
<b>La mémoire</b>	une mémoire morte modifiable électriquement	une mémoire volatile
<b>Consommation d'Énergie</b>	énergie inférieure par rapport aux FPGA	plus élevée
<b>Coût</b>	moins chers	plus coûteux

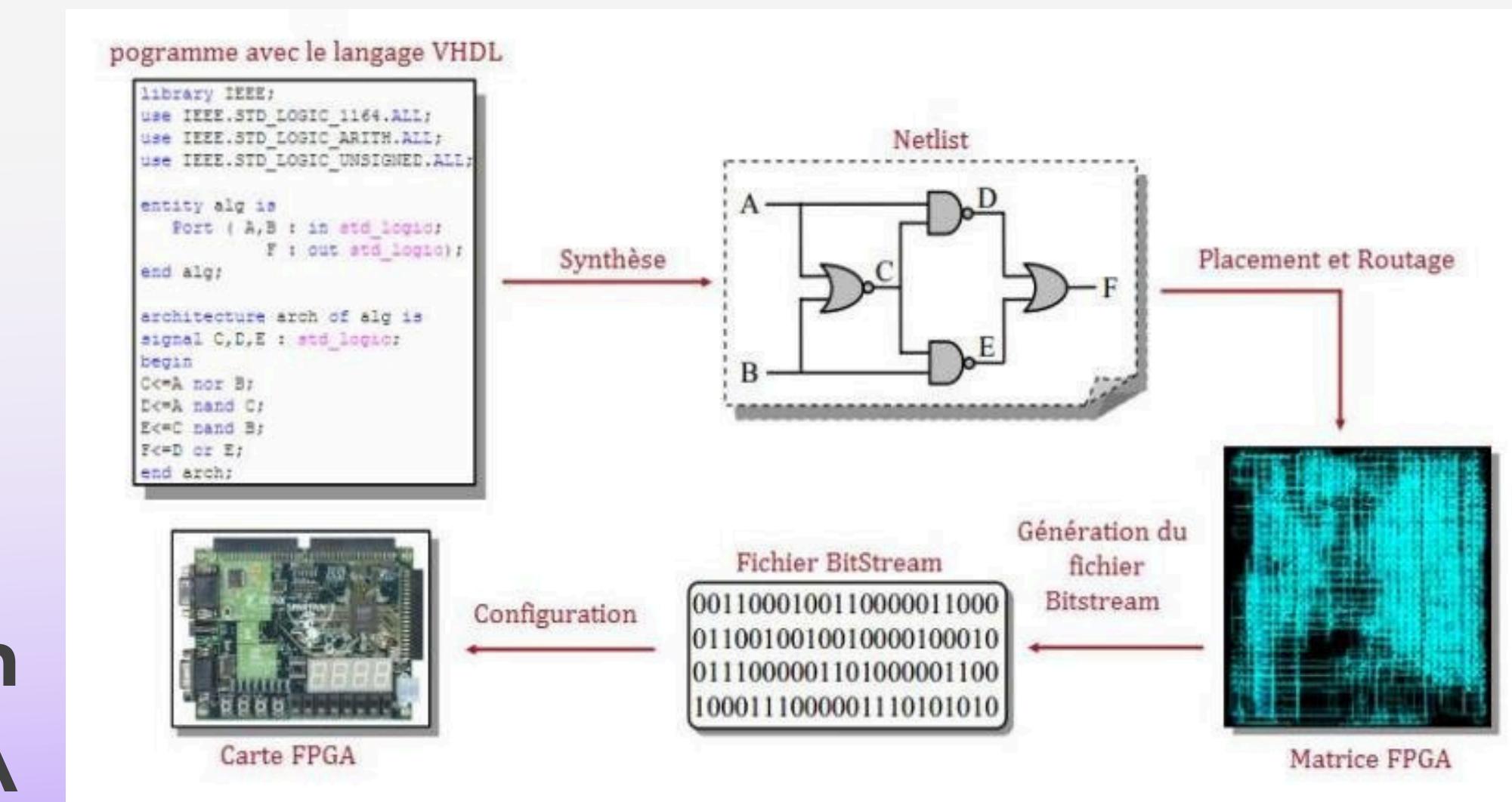
# Microprocesseur VS Microcontrôleur VS FPGA



Aspect	Microprocesseur	Microcontrôleur	FPGA
<b>Fonction Principale</b>	<ul style="list-style-type: none"><li>Calcul, exécution, traitement.</li></ul>	<ul style="list-style-type: none"><li>Contrôle embarqué, tâches spécifiques</li></ul>	<ul style="list-style-type: none"><li>Logique configurable, conception matérielle</li></ul>
<b>Composants Intégrés</b>	<ul style="list-style-type: none"><li>CPU</li></ul>	<ul style="list-style-type: none"><li>CPU, mémoire, périphériques</li></ul>	<ul style="list-style-type: none"><li>Matrice de blocs logiques configurables</li></ul>
<b>Programmabilité (structure matérielle)</b>	<ul style="list-style-type: none"><li>Jeu d'instructions fixe</li></ul>	<ul style="list-style-type: none"><li>Fixe ou limitée</li></ul>	<ul style="list-style-type: none"><li>Programmable via un langage matériel</li></ul>
<b>Vitesse d'Exécution</b>	<ul style="list-style-type: none"><li>Élevée</li></ul>	<ul style="list-style-type: none"><li>Modérée à faible selon la complexité</li></ul>	<ul style="list-style-type: none"><li>Peut varier selon la configuration</li></ul>
<b>Consommation d'Énergie</b>	<ul style="list-style-type: none"><li>Élevée</li></ul>	<ul style="list-style-type: none"><li>Basse</li></ul>	<ul style="list-style-type: none"><li>Très élevée (selon la configuration)</li></ul>
<b>Flexibilité</b>	<ul style="list-style-type: none"><li>Taches spécifiques</li></ul>	<ul style="list-style-type: none"><li>Taches spécifiques</li></ul>	<ul style="list-style-type: none"><li>N'importe quelle tache (reconfigurable)</li></ul>
<b>Exécution des taches</b>	<ul style="list-style-type: none"><li>exécute les taches en séquence</li></ul>	<ul style="list-style-type: none"><li>exécute les taches une par une</li></ul>	<ul style="list-style-type: none"><li>exécute toutes les taches en parallèle.</li></ul>
<b>Coût</b>	<ul style="list-style-type: none"><li>Modéré à élevé</li></ul>	<ul style="list-style-type: none"><li>Faible à modéré</li></ul>	<ul style="list-style-type: none"><li>Modéré à élevé, selon la taille</li></ul>
<b>Langage de programmation</b>	<ul style="list-style-type: none"><li>C, C++, Assembleur</li></ul>	<ul style="list-style-type: none"><li>C, C++, Assembleur</li></ul>	<ul style="list-style-type: none"><li>Verilog ou VHDL</li></ul>

# Les étapes d'implémentation d'un circuit sur un FPGA

1. Écriture du Code Source
2. Compilation
3. Simulation
4. Synthèse
5. Placement
6. Routage
7. Génération du Bitstream
8. Chargement sur le FPGA



# Synthèse

## Définition :

Traduire un modèle logique en une configuration matérielle réelle : netlist.

## Outil de synthèse:

- Xilinx vivado
- Synopsis Design compiler

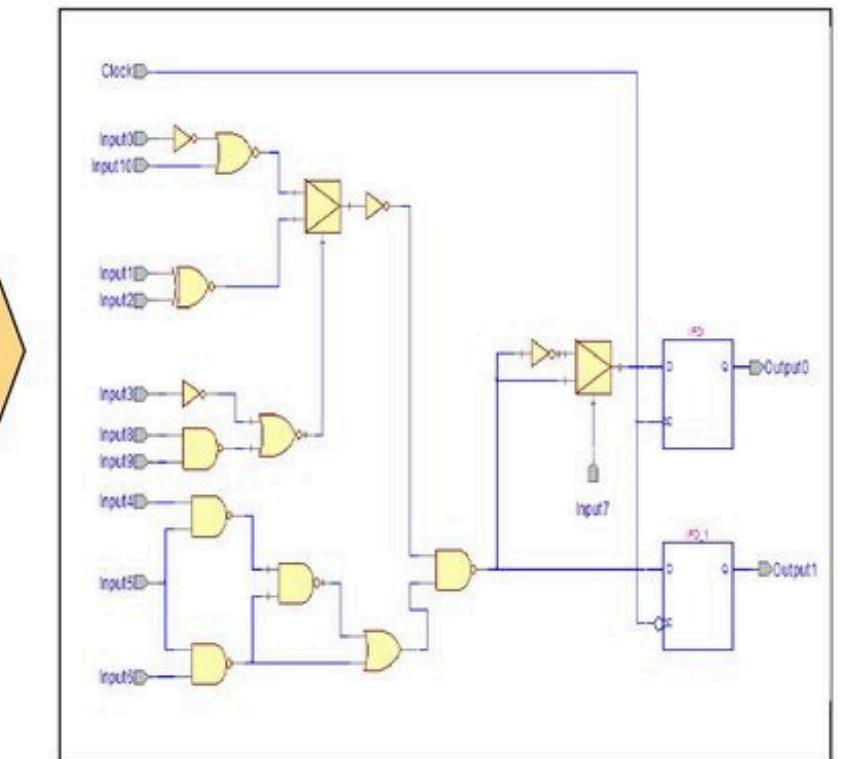
## VHDL description

```
architecture MLU_DATAFLOW of MLU is
begin
    A1<=A when (NEG_A='0') else
        not A;
    B1<=B when (NEG_B='0') else
        not B;
    Y1<=Y1 when (NEG_Y='0') else
        not Y1;

    MUX_0<=A1 and B1;
    MUX_1<=A1 or B1;
    MUX_2<=A1 xor B1;
    MUX_3<=A1 xnor B1;

    with (L1 & L0) select
        Y1<=MUX_0 when "00",
        MUX_1 when "01",
        MUX_2 when "10",
        MUX_3 when others;
end MLU_DATAFLOW;
```

## Circuit netlist



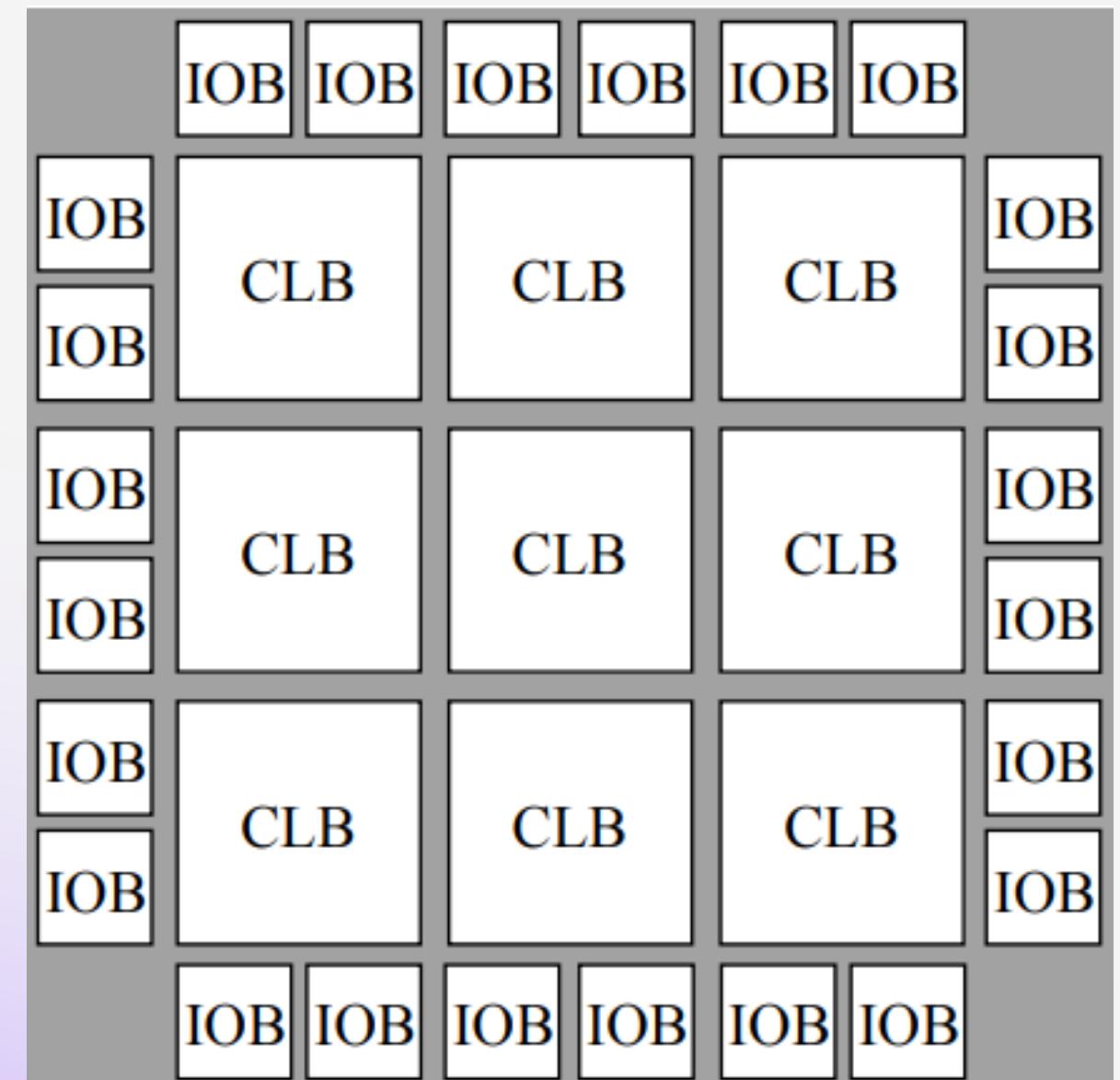
# Placement

## Définition et objectifs:

Attribuer des blocs logiques à des emplacements spécifiques sur la puce.

## Objectifs:

- minimiser la distance de routage et permettre ainsi un routage réussi.
- Respecter les contraintes de temporisation sur les signaux critiques



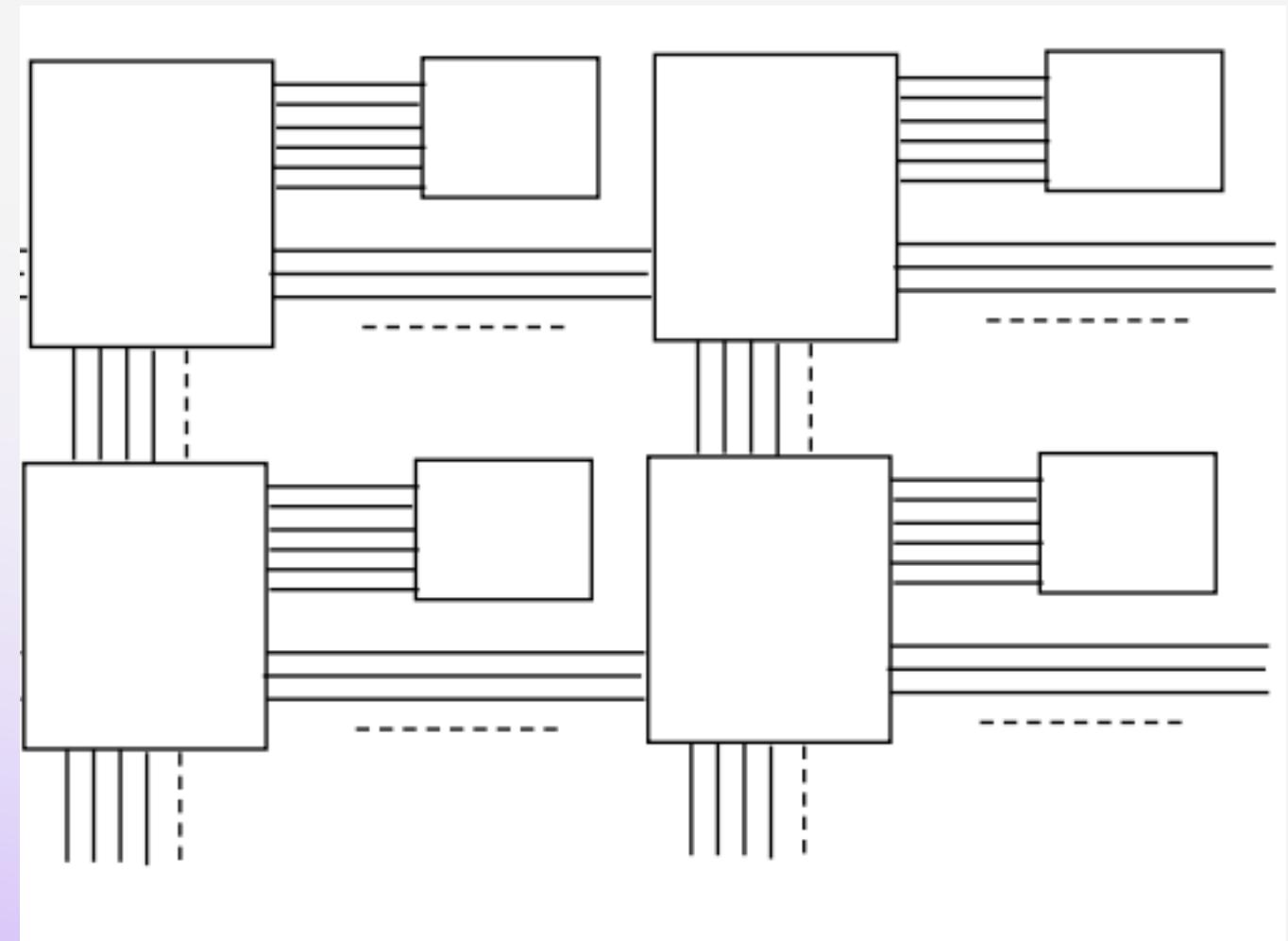
# Routage

## Définition et objectifs:

- Déterminer comment les signaux doivent être routés entre les différents éléments du circuit en utilisant les interconnexions disponibles sur le FPGA.

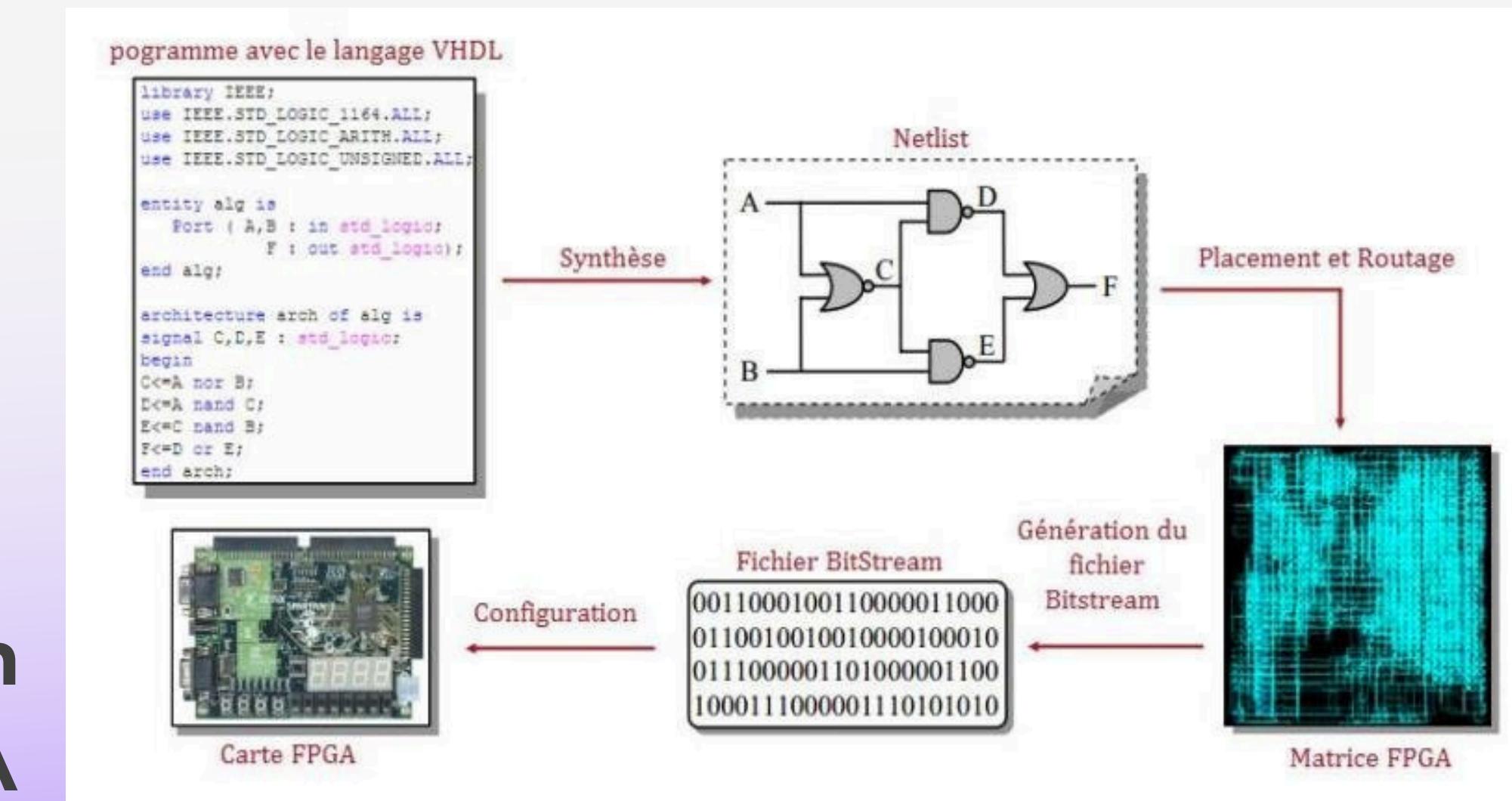
## Objectifs:

- Trouver un routage réalisable pour tous les signaux
- Optimiser le retard des réseaux critiques
- Emprunter des routes plus directes pour les réseaux critiques.
- Les réseaux non critiques peuvent emprunter des routes plus longues.



# Les étapes d'implémentation d'un circuit sur un FPGA

1. Écriture du Code Source
2. Compilation
3. Simulation
4. Synthèse
5. Placement
6. Routage
7. Génération du Bitstream
8. Chargement sur le FPGA



# Qu'est ce qu'il fait d'un FPGA reprogrammable ?

- FPGA adaptable grâce à un code VHDL personnalisé
- Placement et routage ajustés pour chaque exigence.

