

# Syst de Télécom Embarqués

## Chapitre I: Quelques notions



## Introduction

- Electronique enseignée: Analogique et numérique
- Deux sujets étudiés séparément, et les ingénieurs se spécialisent dans l'un des deux
- Au sein de ces deux volets il y a plusieurs spécialités
- De nos jours l'ingénierie logicielle joue un rôle progressivement important dans les systèmes embarqués

## Exemple

Systèmes de contrôle sophistiqués, ordinateurs, téléviseurs numériques, téléphones

## Automatisation de la conception

- Rythme d'évolution rapide des technologies
- Design des produits électroniques doit suivre cette rapidité
- Automatisation du design: Développement informatique
- Une synthèse qui consiste à l'interprétation de la tâche sollicitée
- Une simulation dans laquelle l'étape précédente est exécutée pour vérifier l'exactitude de la synthèse
- Les outils de synthèse et simulation doivent assurer le transfert de l'idée du designer en pratique
- Ceci peut-être tracé sous forme d'un diagramme graphique décrivant le design voulu: méthode schématique
- Autrement, le design peut-être représenté par une forme textuelle (programme) comprise par le matériel: méthode comportementale
- HDL: Hardware Description Language

## Automatisation de la conception

### Exemple

VHDL: VHSIC Hardware Description Language, VHSIC pour Very High Speed Integrated Circuit

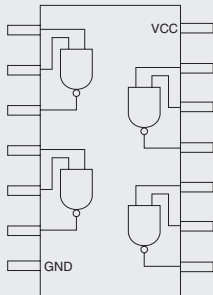
- Les HDLs standards sont très importants
- Ils peuvent être utilisés par différents outils (vendeurs) de développement dédiés à l'automatisation de la conception

## Portes logiques

- La construction des circuits numériques (intégrés) est basée sur les portes logiques
- Une porte logique est un composant électronique
- Un nombre d'entrées et généralement une seule sortie
- Les entrées et sorties prennent deux états logique 0 ou 1
- Ces valeurs sont représentées par des tensions (0V pour 0 et 5V pour 1 par exemple)

## Portes logiques

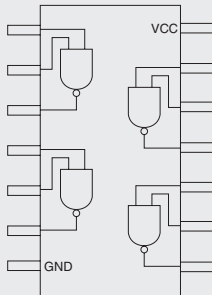
- Il est possible d'acheter un simple circuit intégré contenant par exemple 4 portes logiques



- Vcc et GND correspondent respectivement aux bornes positive et négative de l'alimentation
- Un système numérique peut-être construit en connectant plusieurs circuits pareils

## Portes logiques

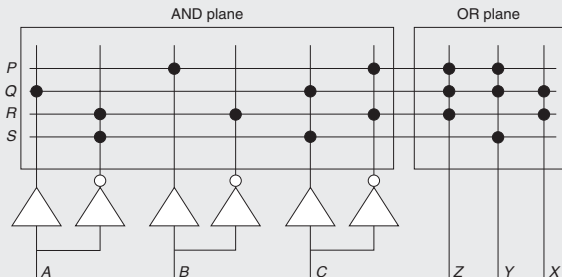
- Il est possible d'acheter un simple circuit intégré contenant par exemple 4 portes logiques



- Vcc et GND correspondent respectivement aux bornes positive et négative de l'alimentation
- Un système numérique peut-être construit en connectant plusieurs circuits pareils

## Vers FPGA

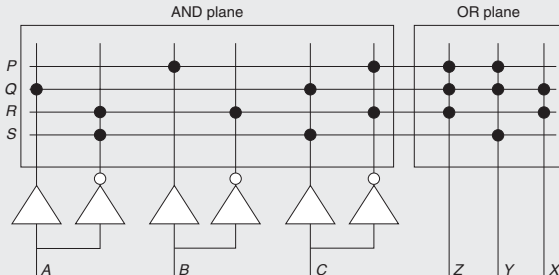
- Au début PLAs (Programmable Logic Arrays) matrices logiques programmables
- Un réseau de portes logiques
- Choix du chemin pour avoir la sortie(s) désirée(s) (Electriquement)
- Retour à l'état initial en exposant le dispositif à la lumière UV





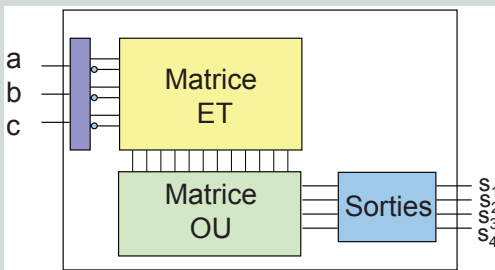
## Vers FPGA

- Au début PLAs (Programmable Logic Arrays) matrices logiques programmables
- Un réseau de portes logiques
- Choix du chemin pour avoir la sortie(s) désirée(s) (Electriquement)
- Retour à l'état initial en exposant le dispositif à la lumière UV



### PLDs

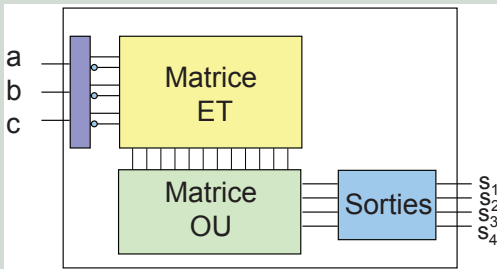
- Apparition des PLDs (Programmable Logic device)
- Réseaux de ET et OU permettant de réaliser n'importe quelle fonction logique



- Tendence vers:
  - CPLDs: Complex Programmable Logic Devices
  - FPGAs: Field Programmable Gate Arrays

### PLDs

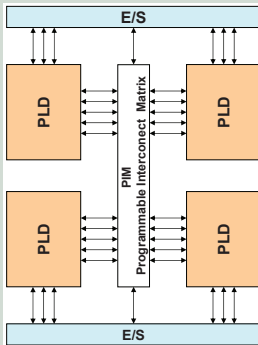
- Apparition des PLDs (Programmable Logic device)
- Réseaux de ET et OU permettant de réaliser n'importe quelle fonction logique



- Tendance vers:
  - CPLDs: Complex Programmable Logic Devices
  - FPGAs: Field Programmable Gate Arrays

### CPLDs

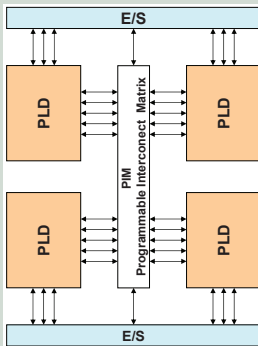
- Les CPLDs regroupent plusieurs PLD interconnectés par un réseau de connexions programmables
- Les CPLDs sont les prémisses des premiers FPGAs



- Ces circuits tendent à être remplacés par les FPGAs

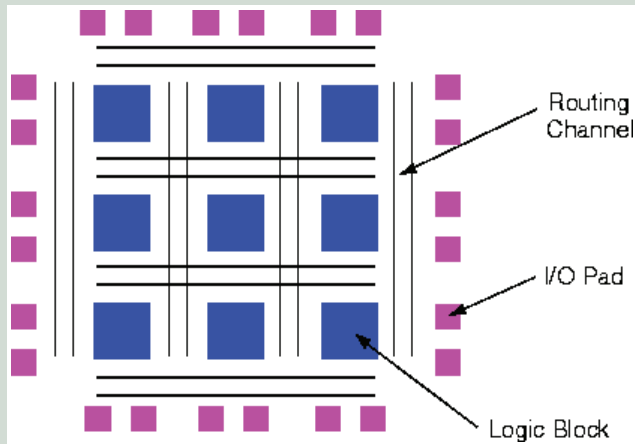
### CPLDs

- Les CPLDs regroupent plusieurs PLD interconnectés par un réseau de connexions programmables
- Les CPLDs sont les prémisses des premiers FPGAs



- Ces circuits tendent à être remplacés par les FPGAs

## FPGAs



## Evolution des données

- La majorité des systèmes est séquentielle: Etats
- Les sorties dépendent de l'état précédent
- Tendance des systèmes numériques à être synchrone
- Changement d'état se déclenche en fonction d'une ou plusieurs horloges
- Le design d'un système utilisant des portes logiques évolue comme suit:

## Evolution des données

- 1 Ecriture de la fiche technique (cahier de charge)
- 2 Si possible répartir le design en "sous design" puis faire 1
- 3 Faire montrer les états, leurs entrées et les sorties qui leurs correspondent
- 4 Réduire le nombre des états le maximum possible (optionnel)
- 5 Affecter les variables booléennes pour chaque état
- 6 Extraction de l'état suivant et sa sortie logique
- 7 Optimisation du nombre de portes logiques (redondance)
- 8 Bon choix de l'emplacement des portes pour la réalisation d'un circuit imprimé simple à interconnecter
- 9 Conception du routage entre les circuits intégrés

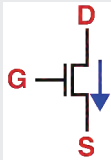


## Evolution des données

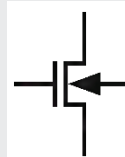
- ➊ Généralement les étapes 1 et 2 dépendent de la créativité du concepteur
- ➋ Les étapes de 3 à 7 font l'objet de tout les livres portant sur la conception numérique
- ➌ Les étapes 8 et 9 sont réalisables manuellement
- ➍ Possibilité de simuler le design étape par étape
- ➎ Satisfaire 8 et 9 revient à simuler toutes les étapes ( $\preceq 7$ )
- ➏ Une fois 9 est achevée, il faut revoir la structure du circuit. Pourquoi ?
- ➐ Vérifier l'existence des retards duent aux résitances et capacités
- ➑ Réduire ces retard puis simuler de nouveau (temps réel)

## Les transistors MOS

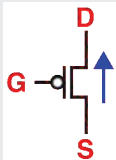
- Transistors NMOS: Un courant d'électrons est susceptible de passer dans le canal, de la source vers le drain



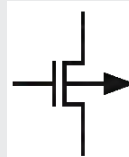
ou



- Transistors PMOS: Un courant d'électrons est susceptible de passer dans le canal, du drain vers la source



ou

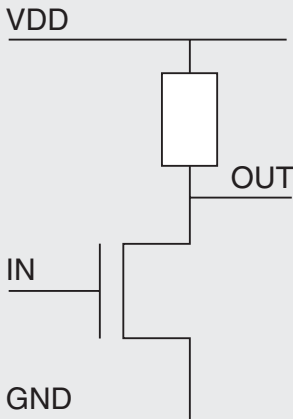


- Transistors CMOS: Combinaison de NMOS et PMOS

## MOS inverseur

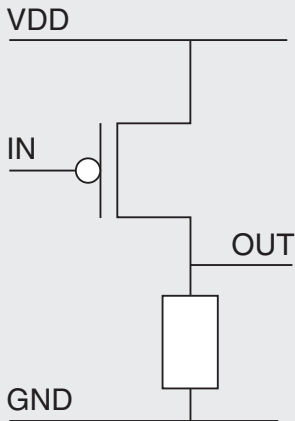
Un inverseur est un circuit à une entrée et une sortie qui réalise l'opération booléenne de négation

- NMOS inverseur:



## MOS inverseur

- PMOS inverseur:

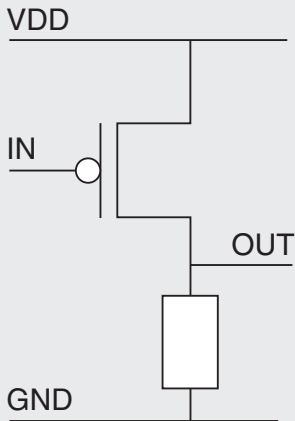


## CMOS

La technologie CMOS utilise à la fois les transistors NMOS et PMOS

## MOS inverseur

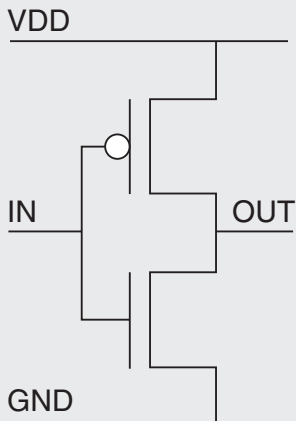
- PMOS inverseur:



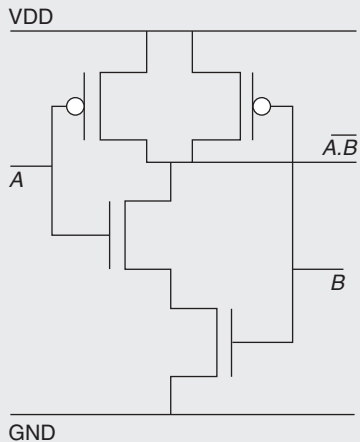
## CMOS

La technologie CMOS utilise à la fois les transistors NMOS et PMOS

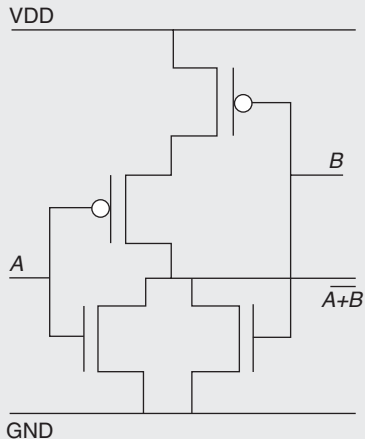
## CMOS inverseur



## CMOS NAND

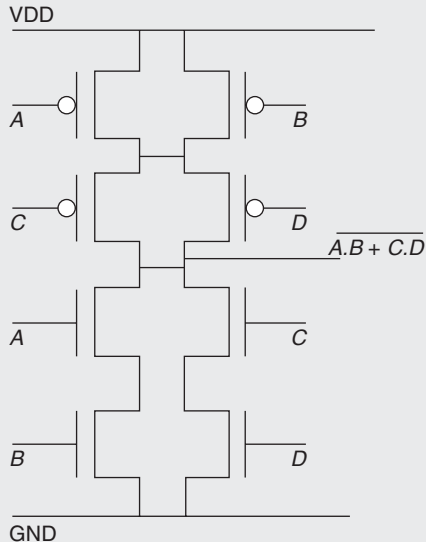


## CMOS NOR





## CMOS AND-OR inverseur



## TTL et CMOS

- Ce sont deux familles technologiques utilisées pour les circuits logiques en électronique
- Circuit intégré contenant des portes logiques telles que des OR, AND, NAND etc...

### Famille TTL

- Transistor-Transistor Logic
- Réalisée avec des transistors bipolaires
- L'alimentation doit être précise à 5V (+ ou - 5 %) sinon on risque de détruire le circuit
- Du fait qu'elle est réalisée avec des transistors bipolaires elle consomme pas mal de courant
- La famille TTL tend à être remplacée par la technologie CMOS

## TTL et CMOS

- Ce sont deux familles technologiques utilisées pour les circuits logiques en électronique
- Circuit intégré contenant des portes logiques telles que des OR, AND, NAND etc...

### Famille TTL

- Transistor-Transistor Logic
- Réalisée avec des transistors bipolaires
- L'alimentation doit être précise à 5V (+ ou - 5 %) sinon on risque de détruire le circuit
- Du fait qu'elle est réalisée avec des transistors bipolaires elle consomme pas mal de courant
- La famille TTL tend à être remplacée par la technologie CMOS

## TTL et CMOS

### Famille CMOS

- Complementary Metal Oxide Semi-conductor
- Cette famille est réalisée avec des transistors à effet de champs
- L'alimentation peut aller de 3V à 18V

Application: Donner le schéma électrique correspondant à:

- $S = \overline{E_1 E_2 + E_3 E_4}$
- $S = \overline{(E_1 + E_2) \cdot (E_3 + E_4)}$