

# SYS2041 – Électronique numérique

## Cours 3 : Les portes logiques

Alexandre BRIÈRE



Les bases de l'algèbre de Boole sont :

- Deux valeurs logiques possibles : VRAI et FAUX (1 et 0)
- Variables booléennes ne prenant qu'une de ces deux valeurs
- Opérateurs permettant de faire des calcul avec ces variables
- Équations booléennes utilisant ces variables et ces opérateurs

Comment passer du monde théorique de l'algèbre de Boole à la réalité ?

⇒ Passer de l'équation booléenne au circuit électronique correspondant

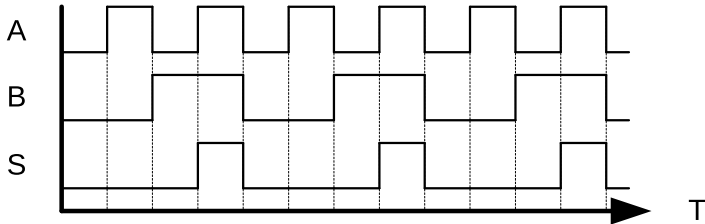
⇒ En utilisant les portes logiques correspondant aux opérateurs

# Comment décrire de tels systèmes ?

- Équation booléenne  $\Leftarrow$  Cours 2
- Table de vérité  $\Leftarrow$  Cours 2
- Schéma logique  $\Leftarrow$  représentation graphique
- Chronogramme  $\Leftarrow$  représentation temporelle
- VHDL  $\Leftarrow$  langage de description matériel

# Chronogramme

Représentation temporelle de plusieurs signaux et de leurs variations les uns par rapport aux autres :



VHDL - VHSIC Hardware Description Language

VHSIC - Very High Speed Integrated Circuits

3 formalismes possibles :

- Flot de données  
⇒ Écriture explicite des fonctions booléennes implémentées
- Structurel  
⇒ On décrit le circuit comme un assemblage de boîtes réalisant elles même des fonctions de complexité variable
- Comportemental  
⇒ On décrit le fonctionnement attendu à la manière des langages de programmation (conditions, boucles, etc.)

Un opérateur unaire :

- NON/NOT

Deux opérateurs (au moins) binaires :

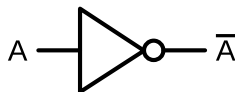
- ET/AND
- OU/OR

Attention, il existe des versions de ces portes avec plus de 2 entrées !

# Opérations de base : NON

La porte logique correspondante est appelée "INVERSEUR"

$A$	$\bar{A}$
0	1
1	0



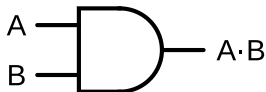
```
entity Inverseur is
port(
    A: in bit;
    S: out bit);
end entity Inverseur;

architecture DataFlow of Inverseur is
begin
    S <= not A;
end architecture DataFlow;
```



La porte logique correspondante est appelée "PORTE ET"

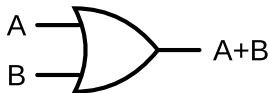
$A$	$B$	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1



# Opérations de base : OU

La porte logique correspondante est appelée "PORTE OU"

$A$	$B$	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1



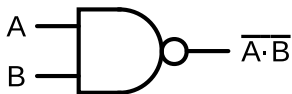
...et les portes correspondantes :

- NON-ET
- NON-OU
- OU-exclusif
- NON-OU-exclusif

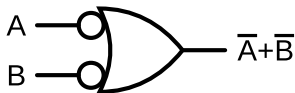
# Opérateur composé et porte associée : NON-ET

Cette porte est plus communément appelée NAND

$A$	$B$	$\overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

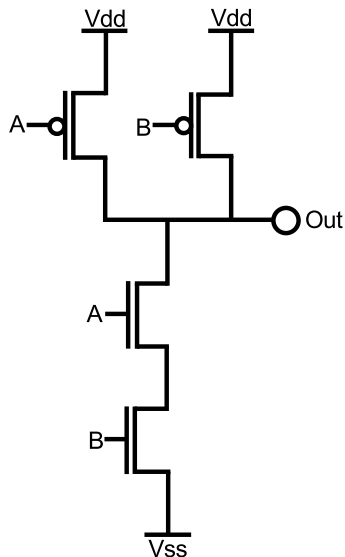


Cette porte a un fonctionnement équivalent à une porte OU négatif.



Attention, il existe des portes avec plus de 2 entrées !

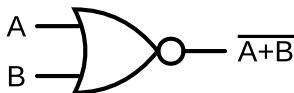
# Implémentation CMOS d'une porte NAND



# Opérateur composé et porte associée : NON-OU

Cette porte est plus communément appelée NOR

A	B	$\overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0



Cette porte a un fonctionnement équivalent à une porte ET négatif :



Attention, il existe des portes avec plus de 2 entrées !

# Opérateur composé et porte associée : OU-exclusif

Aussi appelée disjonction exclusive

- 5 notations possible :

- ▶  $a \text{ XOR } b$
- ▶  $a \oplus b$
- ▶  $a \underline{\vee} b$
- ▶  $a \neq b$
- ▶  $a \neq b$

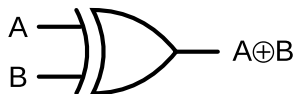
$a$	$b$	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

- Commutative :

$$a \oplus b = b \oplus a$$

- Associative :

$$a \oplus (b \oplus c) = (a \oplus b) \oplus c = a \oplus b \oplus c$$

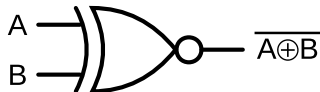


Pour l'instant, nous ne considérons que la version à 2 entrées !

# Opérateur composé et porte associée : NON-OU-exclusif

Cette porte est plus communément appelée NXOR

A	B	$\overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1



Pour l'instant, nous ne considérons que la version à 2 entrées !



# Problème n° 1

Une maison possède 1 porte et 4 fenêtres, chacune est reliée à un détecteur donnant un niveau HAUT quand elle est ouverte.

Comment réaliser son système d'alarme ?

Vous devez définir les variables d'entrées et vous pouvez utiliser des composants "magiques" pour la sirène déclenchée par un niveau HAUT.

Comment réaliser le complément à 1 d'un nombre binaire sur 4 bits ?

Soit deux circuits identiques en parallèle, ces circuits doivent donc produire des sorties identiques.

Comment réaliser un circuit permettant de détecter la défaillance de l'un des deux circuits ?

Une usine utilise deux réservoirs pour stocker des produits chimiques liquides. Chaque réservoir est muni d'un capteur qui détecte quand le niveau de la cuve descend à 25% de sa capacité totale. Pour ce faire, il produit un niveau HAUT quand le niveau est à plus de 25%.

Comment réaliser un circuit indiquant par une diode verte que les deux réservoirs sont remplis à plus de 25% ?

Comment réaliser un circuit indiquant par une diode rouge quand au moins l'un des deux réservoirs descend en dessous de 25% de sa capacité ?

Une voiture possède deux détecteurs, le premier indique un niveau HAUT quand le moteur est allumé, le deuxième indique un niveau BAS quand la ceinture du conducteur est bouclée.

Comment réaliser le circuit qui déclenche une alarme si le conducteur n'a pas bouclé sa ceinture un certain temps après avoir mis le contact ?

Vous devez définir les variables d'entrées et vous pouvez utiliser des composants "magiques" pour le temps et le signal sonore.

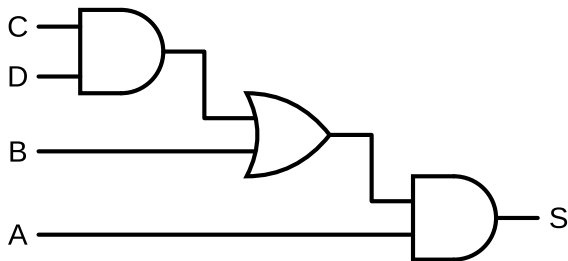
Un avion possède trois trains d'atterrissage. Chaque train d'atterrissage est équipé d'un détecteur qui produit un niveau BAS quand celui-ci est sorti.

Réalisez le circuit possédant les caractéristiques suivantes :

- une DEL verte s'allume si les 3 trains d'atterrissage sont déployés
- une DEL rouge s'allume si au moins l'un des 3 n'est pas déployé

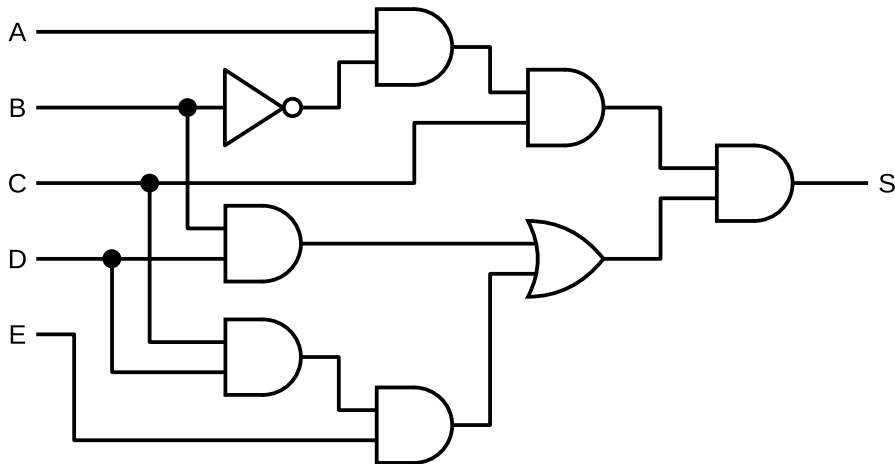
## Problème n° 7

Donnez l'équation booléenne et la table de vérité de ce circuit logique :



## Problème n° 8

Simplifiez le circuit logique suivant :





- [1] Sébastien GAGEOT et Franck CRISON :  
*SYS2041 : Systèmes numériques (Laval)*.
- [2] Thomas FLOYD :  
*Systèmes numériques*.  
Éditions Reynald Goulet, 2018.