



## C10 : Modélisation de la chaîne d'information des systèmes C10-1 : Modélisation des systèmes logiques numériques

Émilien DURIF

Lycée La Martinière Monplaisir Lyon  
Classe de MPSI  
7 Juin 2022



# Plan

## 1 Introduction

- Rôle d'un système de commande
- Informations binaires et numériques
- Systèmes asservis numériques

## 2 Manipulation de l'information binaire

- Bases de logiques combinatoires
- Numération

## 3 Applications technologiques

- Réalisation Technologique



# Plan

## 1 Introduction

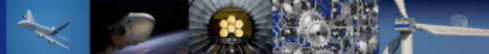
- Rôle d'un système de commande
- Informations binaires et numériques
- Systèmes asservis numériques

## 2 Manipulation de l'information binaire

- Bases de logiques combinatoires
- Numération

## 3 Applications technologiques

- Réalisation Technologique

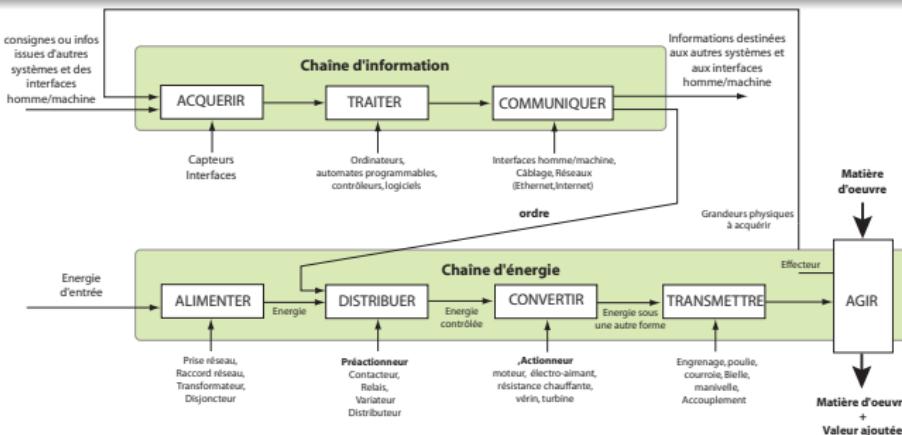


## Système numérique de contrôle-commande

### Système numérique de contrôle-commande

Un **système numérique de contrôle-commande** est une structure programmable (SNCC ou DCS pour *distributed control system* en Anglais) utilisée pour le pilotage d'un procédé industriel ou d'un système mécanique. Un SNCC est composé :

- cartes de commande : fonction **traiter** ;
- interface homme-machine : fonction **communiquer** et/ou **acquérir** ;
- ensemble de capteurs : fonction **acquérir** ;



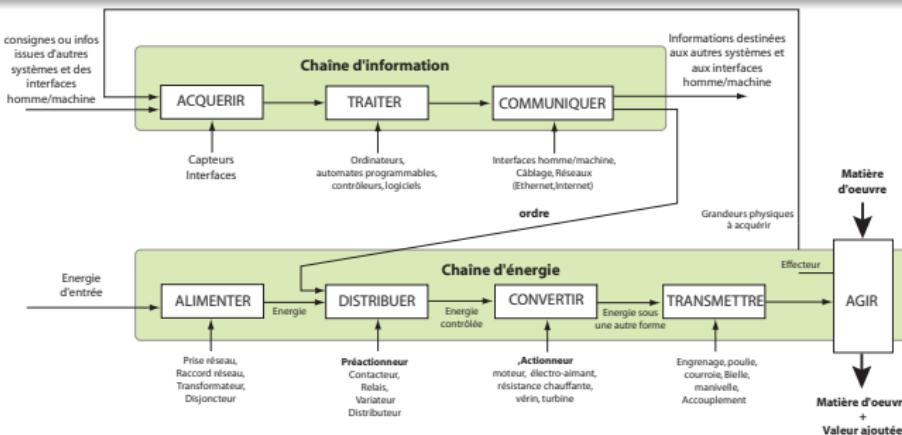


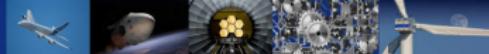
## Système numérique de contrôle-commande

### Système numérique de contrôle-commande

Un **système numérique de contrôle-commande** est une structure programmable (SNCC ou DCS pour *distributed control system* en Anglais) utilisée pour le pilotage d'un procédé industriel ou d'un système mécanique. Un SNCC est composé :

- cartes de commande : fonction **traiter** ;
- interface homme-machine : fonction **communiquer** et/ou **acquérir** ;
- ensemble de capteurs : fonction **acquérir** ;



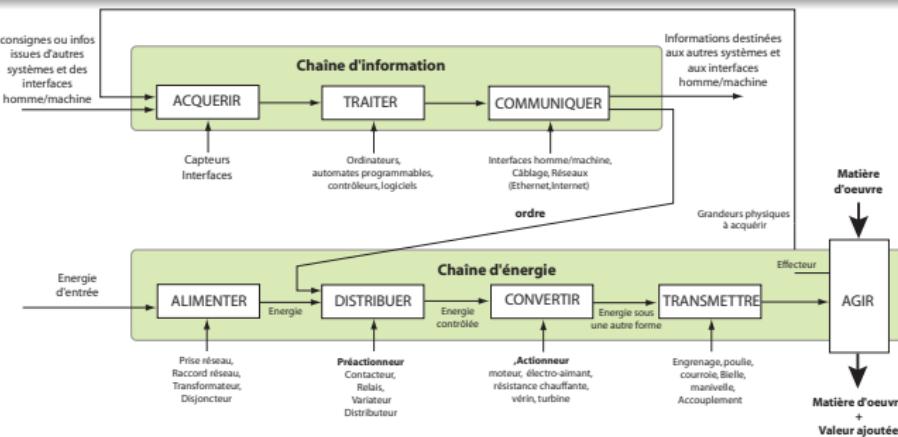


## Système numérique de contrôle-commande

### Système numérique de contrôle-commande

Un **système numérique de contrôle-commande** est une structure programmable (SNCC ou DCS pour *distributed control system* en Anglais) utilisée pour le pilotage d'un procédé industriel ou d'un système mécanique. Un SNCC est composé :

- cartes de commande : fonction **traiter** ;
- interface homme-machine : fonction **communiquer** et/ou **acquérir** ;
- ensemble de capteurs : fonction **acquérir** ;





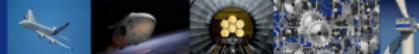
## Système numérique de contrôle-commande

### Système numérique de contrôle-commande

Un **système numérique de contrôle-commande** est une structure programmable (SNCC ou DCS pour *distributed control system* en Anglais) utilisée pour le pilotage d'un procédé industriel ou d'un système mécanique. Un SNCC est composé :

- cartes de commande : fonction **traiter** ;
- interface homme-machine : fonction **communiquer** et/ou **acquérir** ;
- ensemble de capteurs : fonction **acquérir** ;



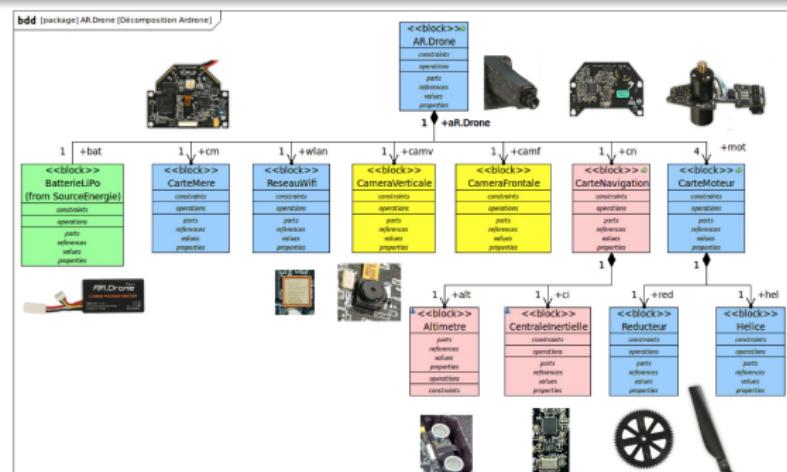


# Système numérique de contrôle-commande

## Système numérique de contrôle-commande

Un **système numérique de contrôle-commande** est une structure programmable (SNCC ou DCS pour *distributed control system* en Anglais) utilisée pour le pilotage d'un procédé industriel ou d'un système mécanique. Un SNCC est composé :

- cartes de commande : fonction **traiter** ;
- interface homme-machine : fonction **communiquer** et/ou **acquérir** ;
- ensemble de capteurs : fonction **acquérir** ;





## Système numérique de contrôle-commande

### Système numérique de contrôle-commande

Un **système numérique de contrôle-commande** est une structure programmable (SNCC ou DCS pour *distributed control system* en Anglais) utilisée pour le pilotage d'un procédé industriel ou d'un système mécanique. Un SNCC est composé :

- cartes de commande : fonction **traiter** ;
- interface homme-machine : fonction **communiquer** et/ou **acquérir** ;
- ensemble de capteurs : fonction **acquérir** ;

### Rôle des informations logiques dans un système

- Un certain nombre d'informations logiques doivent être collectées et traitées sur ce genre de système (Arrêt, détection d'obstacle, etc..).
- L'objet de ce cours est la présentation des bases nécessaires à la manipulation des informations logiques.



## Informations binaires

### Informations binaires

- Un signal **binnaire ou logique** est une grandeur physique qui ne peut prendre que deux états : "Vrai ou Faux".
- On dit aussi qu'un signal binaire est un signal "Tout ou Rien", "présent ou absent".
- Par convention, on donne le code numérique 1 au signal quand il est à l'état "vrai" et 0 quand il est à l'état "faux".
- En pratique, il ne peut y avoir de discontinuité au niveau d'un signal physique. Le passage de 0 à 1 est appelé "**front montant**" et le passage de 1 à 0 "**front descendant**". Les informations traitées par un système combinatoire sont des signaux binaires.



## Informations binaires

### Informations binaires

- Un signal **binnaire ou logique** est une grandeur physique qui ne peut prendre que deux états : "Vrai ou Faux".
- On dit aussi qu'un signal binaire est un signal "Tout ou Rien", "présent ou absent".
- Par convention, on donne le code numérique 1 au signal quand il est à l'état "vrai" et 0 quand il est à l'état "faux".
- En pratique, il ne peut y avoir de discontinuité au niveau d'un signal physique. Le passage de 0 à 1 est appelé "front montant" et le passage de 1 à 0 "front descendant". Les informations traitées par un système combinatoire sont des signaux binaires.



## Informations binaires

### Informations binaires

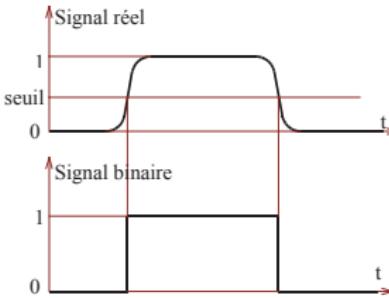
- Un signal **binnaire ou logique** est une grandeur physique qui ne peut prendre que deux états : "Vrai ou Faux".
- On dit aussi qu'un signal binaire est un signal "Tout ou Rien", "présent ou absent".
- Par convention, on donne le code numérique 1 au signal quand il est à l'état "vrai" et 0 quand il est à l'état "faux".
- En pratique, il ne peut y avoir de discontinuité au niveau d'un signal physique. Le passage de 0 à 1 est appelé "front montant" et le passage de 1 à 0 "front descendant". Les informations traitées par un système combinatoire sont des signaux binaires.



# Informations binaires

## Informations binaires

- Un signal **binnaire ou logique** est une grandeur physique qui ne peut prendre que deux états : "Vrai ou Faux".
- On dit aussi qu'un signal binaire est un signal "Tout ou Rien", "présent ou absent".
- Par convention, on donne le code numérique 1 au signal quand il est à l'état "vrai" et 0 quand il est à l'état "faux".
- En pratique, il ne peut y avoir de discontinuité au niveau d'un signal physique. Le passage de 0 à 1 est appelé "**front montant**" et le passage de 1 à 0 "**front descendant**". Les informations traitées par un système combinatoire sont des signaux binaires.





## Informations numériques

### Informations numériques

Une **information numérique** se représente à partir d'une combinaison d'**informations binaires**. La taille d'une information numérique est directement liée au nombre d'informations binaires qui la compose.

- un *bit* ou *digit* est composé d'un nombre binaire ;
- un *octet* est composé de *8 bits* ;
- un *kilo octet (ko)* est composé de *1000 octets* et donc de *8000 bits*.



Code Barres



QR Code



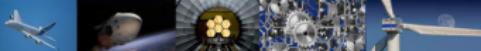
## Systèmes asservis numériques

### Definition

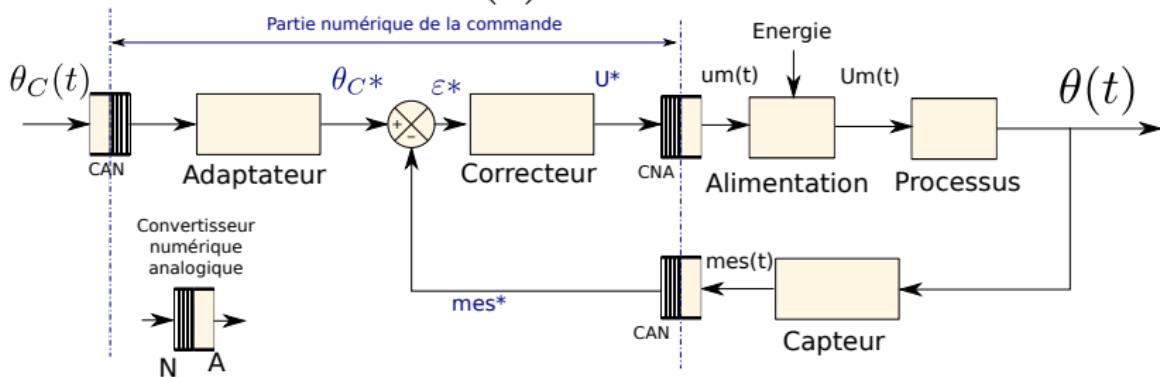
Représentation physique et modèle numérique

Le rôle d'une carte informatique est d'assurer les fonctions de calcul de l'écart et de la valeur de commande. Il faut donc convertir les grandeurs physique continues en grandeurs numériques :

- dans un sens avec un **Convertisseur Analogique Numérique** : CAN ;
- dans l'autre sens avec un **Convertisseur Numérique Analogique** : CNA.



## Systèmes asservis numériques





# Plan

## 1 Introduction

- Rôle d'un système de commande
- Informations binaires et numériques
- Systèmes asservis numériques

## 2 Manipulation de l'information binaire

- Bases de logiques combinatoires
- Numération

## 3 Applications technologiques

- Réalisation Technologique



## Algèbre de Boole

### Algèbre de Boole

Soit un ensemble  $B\{0,1\}$  constitué de deux éléments représentants les deux états logiques vrai ou faux. Cet ensemble est muni d'une structure d'algèbre avec les trois lois suivantes :

Complémentarités : **NON**

$$\begin{aligned} B &\longmapsto B \\ a &\longmapsto \text{NON}(a) = \bar{a}. \end{aligned} \tag{1}$$

$\bar{a}$  est à 1 si  $a$  est à 0 et réciproquement.



## Algèbre de Boole

### Algèbre de Boole

Soit un ensemble  $B\{0,1\}$  constitué de deux éléments représentants les deux états logiques vrai ou faux. Cet ensemble est muni d'une structure d'algèbre avec les trois lois suivantes :

Produit booléen : **ET**

$$\begin{aligned} B \times B &\longmapsto B \\ (a, b) &\longmapsto a \text{ ET } b = a \cdot b. \end{aligned} \tag{2}$$

*a · b est à 1 si et seulement si a est à 1 ET b est à 1.*



## Algèbre de Boole

### Algèbre de Boole

Soit un ensemble  $B\{0,1\}$  constitué de deux éléments représentants les deux états logiques vrai ou faux. Cet ensemble est muni d'une structure d'algèbre avec les trois lois suivantes :

Somme booléenne : **OU**

$$\begin{aligned} B \times B &\longmapsto B \\ (a, b) &\longmapsto a \text{ OU } b = a + b. \end{aligned} \tag{3}$$

*a + b est à 1 si et seulement si a est à 1 OU b est à 1.*



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$
- Inclusion :  
$$a + a \cdot b = a \cdot (1 + b) + a \cdot b = a + a \cdot b + a \cdot b = a + (a + a) \cdot b = a + 1 \cdot b = a + b$$



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$
- Inclusion :  
$$a + a \cdot b = a \cdot (1 + b) + a \cdot b = a + a \cdot b + a \cdot b = a + (a + a) \cdot b = a + 1 \cdot b = a + b$$

## Propriétés : manipulation de l'algèbre de Boole

	<b>Loi OU</b>	<b>Loi ET</b>
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$
- Inclusion :  
$$a + a \cdot b = a \cdot (1 + b) + a \cdot b = a + a \cdot b + a \cdot b = a + (a + a) \cdot b = a + 1 \cdot b = a + b$$

## Propriétés : manipulation de l'algèbre de Boole

	<b>Loi OU</b>	<b>Loi ET</b>
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$
- Inclusion :  
$$a + a \cdot b = a \cdot (1 + b) + a \cdot b = a + a \cdot b + a \cdot b = a + (a + a) \cdot b = a + 1 \cdot b = a + b$$



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$
- Inclusion :  
$$a + a \cdot b = a \cdot (1 + b) + a \cdot b = a + a \cdot b + a \cdot b = a + (a + a) \cdot b = a + 1 \cdot b = a + b$$



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$
- Inclusion :  
$$a + a \cdot b = a \cdot (1 + b) + a \cdot b = a + a \cdot b + a \cdot b = a + (a + a) \cdot b = a + 1 \cdot b = a + b$$

## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$ .
  - Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$ .
  - Inclusion :
 
$$a + \bar{a} \cdot b = a \cdot (1 + b) + \bar{a} \cdot b = a + a \cdot b + \bar{a} \cdot b = a + (a + \bar{a}) \cdot b = a + 1 \cdot b = a + b.$$

## Propriétés : manipulation de l'algèbre de Boole

	<b>Loi OU</b>	<b>Loi ET</b>
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$ .
  - Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$ .
  - Inclusion :  

$$a + \bar{a} \cdot b = a \cdot (1 + b) + \bar{a} \cdot b = a + a \cdot b + \bar{a} \cdot b = a + (a + \bar{a}) \cdot b = a + 1 \cdot b = a + b.$$



## Propriétés : manipulation de l'algèbre de Boole

	Loi OU	Loi ET
Commutativité	$a + b = b + a$	$a \cdot b = b \cdot a$
Distributivité	$a + (b \cdot c) = (a + b) \cdot (a + c)$	$a \cdot (b + c) = (a \cdot b) + (a \cdot c) = a \cdot b + a \cdot c$
Associativité	$a + (b + c) = (a + b) + c = a + b + c$	$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$
Élément neutre 0	$a + 0 = a$	$a \cdot 0 = 0$
Élément neutre 1	$a + 1 = 1$	$a \cdot 1 = a$
Complémentarité	$a + \bar{a} = 1$	$a \cdot \bar{a} = 0$
Idempotence	$a + a = a$	$a \cdot a = a$

- Double complément :  $\bar{\bar{a}} = a$ .
- Absorption :  $a + a \cdot b = a \cdot (1 + b) = a \cdot 1 = a$ .
- Inclusion :  
$$a + \bar{a} \cdot b = a \cdot (1 + b) + \bar{a} \cdot b = a + a \cdot b + \bar{a} \cdot b = a + (a + \bar{a}) \cdot b = a + 1 \cdot b = a + b.$$



## Théorème de De Morgan

### Théorème de De Morgan

- Le complément d'un **OU** est le **ET** des compléments :

$$\overline{(a + b)} = \bar{a} \cdot \bar{b}. \quad (4)$$

- Le complément d'un **ET** est le **OU** des compléments :

$$\overline{(a \cdot b)} = \bar{a} + \bar{b}. \quad (5)$$



## Théorème de De Morgan

### Théorème de De Morgan

- Le complément d'un **OU** est le **ET** des compléments :

$$\overline{(a + b)} = \bar{a} \cdot \bar{b}. \quad (4)$$

- Le complément d'un **ET** est le **OU** des compléments :

$$\overline{(a \cdot b)} = \bar{a} + \bar{b}. \quad (5)$$



## Définition d'une fonction par sa table de vérité

### Table de vérité

Une **table de vérité** permet de représenter le comportement logique d'une fonction de plusieurs variables logiques. Pour une fonction de  $n$  variables la table comporte :

- $n$  colonnes plus une pour le résultat.
- $2^n$  lignes pour les différentes combinaisons des  $n$  variables écrites en **binaire naturel** ou **binaire réfléchi** à partir de 0.

$a$	$b$	$c$	$S$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Code binaire naturel

$a$	$b$	$c$	$S$
0	0	0	0
0	0	1	1
0	1	1	1
0	1	0	1
1	1	0	0
1	1	1	1
1	0	1	0
1	0	0	0

Code binaire réfléchi

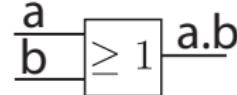


## Opérateurs logiques fondamentaux

Opérateur logique NON																	
Informations	Table de vérité	Symbolisation															
C'est l'opérateur qui transforme une variable logique $a$ en son complément $\bar{a}$ (a barre).	<table border="1"> <tr> <td><math>a</math></td><td><math>\bar{a}</math></td></tr> <tr> <td>0</td><td>1</td></tr> <tr> <td>1</td><td>0</td></tr> </table>	$a$	$\bar{a}$	0	1	1	0										
$a$	$\bar{a}$																
0	1																
1	0																
Opérateur logique ET																	
Informations	Table de vérité	Symbolisation															
Le résultat R de l'opération "ET" entre deux variables binaires $a$ et $b$ est égal à 1 si et seulement si les deux variables sont vraies. On note $a \cdot b = R$ . On lit	<table border="1"> <tr> <td><math>a</math></td><td><math>b</math></td><td><math>a \cdot b</math></td></tr> <tr> <td>0</td><td>0</td><td>0</td></tr> <tr> <td>0</td><td>1</td><td>0</td></tr> <tr> <td>1</td><td>0</td><td>0</td></tr> <tr> <td>-</td><td>-</td><td>-</td></tr> </table>	$a$	$b$	$a \cdot b$	0	0	0	0	1	0	1	0	0	-	-	-	
$a$	$b$	$a \cdot b$															
0	0	0															
0	1	0															
1	0	0															
-	-	-															



## Opérateur logiques fondamentaux

Informations	Opérateur logique OU	Symbolisation															
	Table de vérité																
<p>Le résultat R de l'opération "OU" entre deux variables binaires <math>a</math> et <math>b</math> est égal à 1 si au moins une des deux variables est vraie. On note <math>a + b = R</math>. On lit "a OU b égale R"</p>	<table border="1"><thead><tr><th data-bbox="434 428 530 489"><math>a</math></th><th data-bbox="530 428 625 489"><math>b</math></th><th data-bbox="625 428 830 489"><math>a + b</math></th></tr></thead><tbody><tr><td data-bbox="434 489 530 550">0</td><td data-bbox="530 489 625 550">0</td><td data-bbox="625 489 830 550">0</td></tr><tr><td data-bbox="434 550 530 622">0</td><td data-bbox="530 550 625 622">1</td><td data-bbox="625 550 830 622">1</td></tr><tr><td data-bbox="434 622 530 694">1</td><td data-bbox="530 622 625 694">0</td><td data-bbox="625 622 830 694">1</td></tr><tr><td data-bbox="434 694 530 765">1</td><td data-bbox="530 694 625 765">1</td><td data-bbox="625 694 830 765">1</td></tr></tbody></table>	$a$	$b$	$a + b$	0	0	0	0	1	1	1	0	1	1	1	1	
$a$	$b$	$a + b$															
0	0	0															
0	1	1															
1	0	1															
1	1	1															



## Opérateurs logiques fondamentaux

Opérateur NON-ET (ou NAND)			
Règle	Table de vérité		
La sortie est à 1 si au moins une des entrées est à 0. $S = \bar{a} \cdot \bar{b} = \bar{a} + \bar{b}$		a	b
		0	0
		0	1
		1	0
		1	1
			S
Opérateur NON-OU (ou NOR)			
Règle	Table de vérité		
La sortie est à 1 si toutes les entrées sont à 0. $S = \bar{a} + \bar{b} = \bar{a} \cdot \bar{b}$		a	b
		0	0
		0	1
		1	0
		1	1
			S



## Fonctions logiques de base

Opérateur INHIBITION				
Règle	Table de vérité			Symbolisation
La sortie est à 1 si l'entrée inhibition est à 0 et les autres à 1. $S = a \cdot \bar{b}$		a	b	S
		0	0	0
		0	1	0
		1	0	1
		1	1	0
Opérateur IDENTITÉ				
Règle	Table de vérité			Symbolisation
La sortie est à 1 si toutes les entrées sont au même état. $S = a \otimes b = a \cdot b + \bar{a} \cdot \bar{b}$		a	b	S
		0	0	1
		0	1	0
		1	0	0
		1	1	1



## Opérateurs logiques fondamentaux

Opérateur OU EXCLUSIF				
Règle	Table de vérité			Symbolisation
La sortie est à 1 si une et une seule des entrées est à 1. $S = a \oplus b = a \cdot \bar{b} + \bar{a} \cdot b$		$a$	$b$	$S$
		0	0	0
		0	1	1
		1	0	1
		1	1	0
		$a$	$b$	$=1$

### Remarque

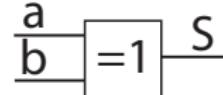
$$a \oplus b = a \oplus b$$

$$a \oplus \bar{b} = a \oplus b$$

$a \oplus b \oplus c$  ne veut pas dire une seule variable à 1, il faut lire  $a \oplus (b \oplus c)$  ou  $(a \oplus b) \oplus c$ .



## Opérateurs logiques fondamentaux

Opérateur OU EXCLUSIF				
Règle	Table de vérité			Symbolisation
La sortie est à 1 si une et une seule des entrées est à 1. $S = a \oplus b = a \cdot \bar{b} + \bar{a} \cdot b$		$a$	$b$	$S$
		0	0	0
		0	1	1
		1	0	1
		1	1	0
				

### Remarque

- $\overline{a \oplus b} = a \otimes b$  ;
- $\overline{a \otimes b} = a \oplus b$  ;
- $a \oplus b \oplus c$  ne veut pas dire une seule variable à 1, il faut lire  $a \oplus (b \oplus c)$  ou  $(a \oplus b) \oplus c$  .



## Opérateurs logiques fondamentaux

Opérateur OU EXCLUSIF				
Règle	Table de vérité			Symbolisation
La sortie est à 1 si une et une seule des entrées est à 1. $S = a \oplus b = a \cdot \bar{b} + \bar{a} \cdot b$		$a$	$b$	$S$
		0	0	0
		0	1	1
		1	0	1
		1	1	0
		$a$	$b$	$=1$

### Remarque

- $\overline{a \oplus b} = a \otimes b$  ;
- $\overline{a \otimes b} = a \oplus b$  ;
- $a \oplus b \oplus c$  ne veut pas dire une seule variable à 1, il faut lire  $a \oplus (b \oplus c)$  ou  $(a \oplus b) \oplus c$  .



## Opérateurs logiques fondamentaux

Opérateur OU EXCLUSIF				
Règle	Table de vérité			Symbolisation
La sortie est à 1 si une et une seule des entrées est à 1. $S = a \oplus b = a \cdot \bar{b} + \bar{a} \cdot b$		$a$	$b$	$S$
		0	0	0
		0	1	1
		1	0	1
		1	1	0
		$a$	$b$	$=1$

### Remarque

- $\overline{a \oplus b} = a \otimes b$  ;
- $\overline{a \otimes b} = a \oplus b$  ;
- $a \oplus b \oplus c$  ne veut pas dire une seule variable à 1, il faut lire  $a \oplus (b \oplus c)$  ou  $(a \oplus b) \oplus c$  .

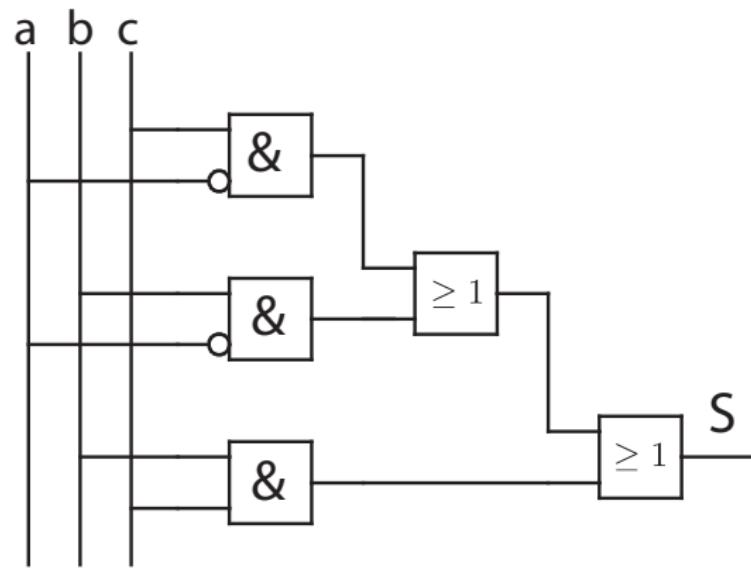


## Logigrammes

### Logigramme

Les logigrammes utilisent les symboles des opérateurs et des fonctions de base.

Exemple de logigramme pour  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$



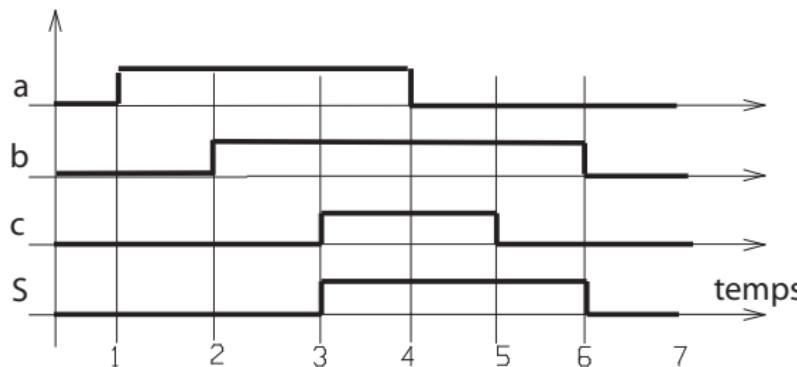


## Chronogrammes

### Chronogramme

Le **chronogramme** est un graphe qui permet de définir l'état d'une variable de sortie à partir de l'état des variables d'entrée définies chronologiquement.

Exemple de chronogramme pour  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$



### Remarque

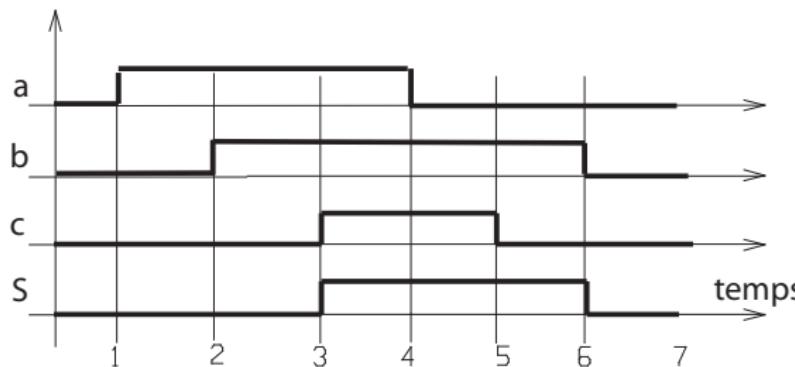
Deux variables ne peuvent pas changer d'état simultanément.

## Chronogrammes

### Chronogramme

Le **chronogramme** est un graphe qui permet de définir l'état d'une variable de sortie à partir de l'état des variables d'entrée définies chronologiquement.

Exemple de chronogramme pour  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$



### Remarque

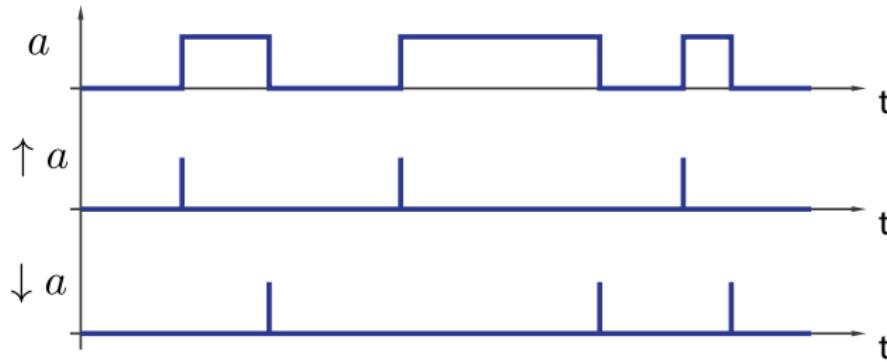
Deux variables ne peuvent pas changer d'état simultanément.



## Notions de fronts montants et descendants

### Fronts montants et descendants

- Le **front montant** d'une variable  $a$  est vrai à l'instant pour lequel la variable passe de son état logique 0 à son état logique 1. Il est noté  $\uparrow a$ .
- Le **front descendant** d'une variable  $a$  est vrai à l'instant pour lequel la variable passe de son état logique 1 à son état logique 0. Il est noté  $\downarrow a$ .
- Un front est d'une durée théorique nulle. Il sera représenté par une impulsion de Dirac sur un chronogramme.

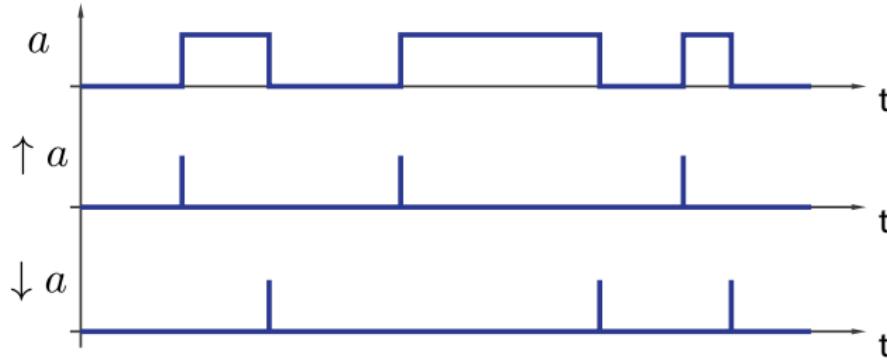




## Notions de fronts montants et descendants

### Fronts montants et descendants

- Le **front montant** d'une variable  $a$  est vrai à l'instant pour lequel la variable passe de son état logique 0 à son état logique 1. Il est noté  $\uparrow a$ .
- Le **front descendant** d'une variable  $a$  est vrai à l'instant pour lequel la variable passe de son état logique 1 à son état logique 0. Il est noté  $\downarrow a$ .
- Un front est d'une durée théorique nulle. Il sera représenté par une impulsion de Dirac sur un chronogramme.

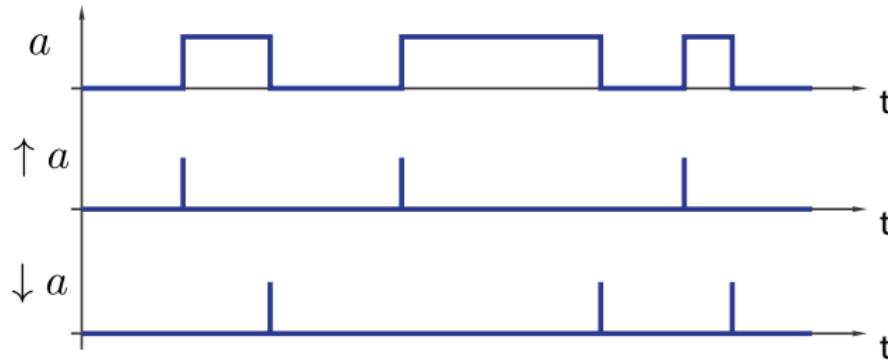




## Notions de fronts montants et descendants

### Fronts montants et descendants

- Le **front montant** d'une variable  $a$  est vrai à l'instant pour lequel la variable passe de son état logique 0 à son état logique 1. Il est noté  $\uparrow a$ .
- Le **front descendant** d'une variable  $a$  est vrai à l'instant pour lequel la variable passe de son état logique 1 à son état logique 0. Il est noté  $\downarrow a$ .
- Un front est d'une durée théorique nulle. Il sera représenté par une impulsion de Dirac sur un chronogramme.





## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base (12345<sub>10</sub>).
- La notation en langage C utilise un préfixe (0b ou 0x) :
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base ( $12345_m$ ).
- La notation en langage C utilise un préfixe (0b ou 0x) :
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base (12345<sub>10</sub>).
- La notation en langage C utilise un préfixe (0b ou 0x) :
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base ( $12345_{10}$ ).
- La notation en langage C utilise un préfixe (0b ou 0x) :
  - 0b10110010 pour un nombre binaire,
  - 0xF3 pour un nombre en hexadécimal.
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base ( $12345_{10}$ ).
- La notation en langage C utilise un préfixe (0b ou 0x) :
  - 0b0111010 pour un nombre binaire,
  - 0x6F3 pour un nombre en hexadécimal.
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base ( $12345_{10}$ ).
- La notation en langage C utilise un préfixe (0b ou 0x) :
  - 0b0111010 pour un nombre binaire,
  - 0x6F3 pour un nombre en hexadécimal.
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base ( $12345_{10}$ ).
- La notation en langage C utilise un préfixe (0b ou 0x) :
  - 0b0111010 pour un nombre binaire,
  - 0xF3 pour un nombre en hexadécimal.
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Numération binaire

- Les microprocesseurs et les ordinateurs travaillent en base 2.
- Cette base ne comporte que deux éléments : 0 et 1.

### Écriture sur une base N

L'écriture d'un nombre  $A$  dans une base  $N$  est :

$$A = \sum_{i=0}^k c_i N^i, \quad (6)$$

où les  $c_i$  sont des éléments de la base  $N$ . Par exemple

$$12345 = 1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$$

### Remarque

- Pour reconnaître dans quelle base est écrit un nombre, on note en indice la base ( $12345_{10}$ ).
- La notation en langage C utilise un préfixe (0b ou 0x) :
  - 0b0111010 pour un nombre binaire,
  - 0xF3 pour un nombre en hexadécimal.
- La base décimal n'a pas de notation particulière puisqu'il s'agit de la base par défaut.



## Conversion bases 2, 10 et 16

La table suivante donne la conversion entre les bases 2, 10 et 16 :

Base 10	Base 2	Base 16
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Pour convertir un nombre on peut utiliser des divisions euclidiennes ou des factorisations successives. Conversion d'un nombre décimal en binaire par la méthode des divisions successives.



## Numération binaire

conversion du nombre 47 ?

$$(47)_{10} = (001011)_2 = (2F)_{16} = (10025)_{32}$$



## Numération binaire

conversion du nombre 47 ?

$$(47)_{10} = (001011)_2 = (2F)_{16} = (1002F)_{32}$$



## Numération binaire

conversion du nombre 47 ?

$$(47)_{10} = (001011)_2 = (25)_{16} = (0025)$$



## Numération binaire

conversion du nombre 47 ?

$$(47)_{10} = (101111)_2 = (2F)_{16} = (0x2F)$$



## Numération binaire

conversion du nombre 47 ?

$$(47)_{10} = (101111)_2 = (2F)_{16} = (0x2F)$$



## Codage ASCII

### Code ASCII

Le **code ASCII** (American Standard Code for Information Interchange) est un code défini sur 8 bits permettant de définir 256 caractères distincts. Chaque caractère possède un nombre associé en hexadécimal.

		MSB	0	1	2	3	4	5	6	7
LSB		000	001	010	011	100	101	110	111	
0	0000	NUL	DLE	SP	0	@	P	`	p	
1	0001	SOH	DC1	!	1	A	Q	a	q	
2	0010	STX	DC2	"	2	B	R	b	r	
3	0011	ETX	DC3	#	3	C	S	c	s	
4	0100	EOT	DC4	\$	4	D	T	d	t	
5	0101	ENQ	NAK	%	5	E	U	e	u	
6	0110	ACK	SYN	&	6	F	V	f	v	
7	0111	BEL	ETB	'	7	G	W	g	w	
8	1000	BS	CAN	(	8	H	X	h	x	
9	1001	HT	EM	)	9	I	Y	i	y	
A	1010	LF	SUB	*	:	J	Z	j	z	
B	1011	VT	ESC	+	:	K	[	k	}	
C	1100	FF	FS	,	<	L	\	l	l	
D	1101	CR	GS	-	=	M	]	m	{	
E	1110	SO	RS	.	>	N	^	n	~	
F	1111	SI	US	/	?	O	_	o	DEL	



# Plan

## 1 Introduction

- Rôle d'un système de commande
- Informations binaires et numériques
- Systèmes asservis numériques

## 2 Manipulation de l'information binaire

- Bases de logiques combinatoires
- Numération

## 3 Applications technologiques

- Réalisation Technologique



## Technologie électrique câblée

- En logique électrique l'élément technologique de base est le contact à établissement de circuit (contact NO : normalement ouvert).
- La variable complémentée est matérialisée par le contact à interruption de circuit (contact NF : normalement fermé).
- L'opérateur ET est obtenu par la mise en série (câblage) des contacts, l'opérateur OU par la mise en parallèle. (*Voir l'exemple du "va-et-vient" ci-après.*)



## Technologie électrique câblée

- En logique électrique l'élément technologique de base est le contact à établissement de circuit (contact NO : normalement ouvert).
- La variable complémentée est matérialisée par le contact à interruption de circuit (contact NF : normalement fermé).
- L'opérateur ET est obtenu par la mise en série (câblage) des contacts, l'opérateur OU par la mise en parallèle. (*Voir l'exemple du "va-et-vient" ci-après.*)



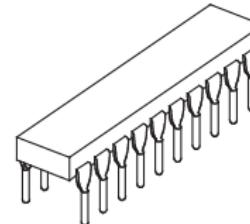
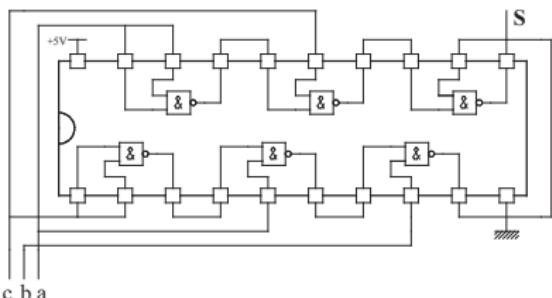
## Technologie électrique câblée

- En logique électrique l'élément technologique de base est le contact à établissement de circuit (contact NO : normalement ouvert).
- La variable complémentée est matérialisée par le contact à interruption de circuit (contact NF : normalement fermé).
- L'opérateur ET est obtenu par la mise en série (câblage) des contacts, l'opérateur OU par la mise en parallèle. (*Voir l'exemple du "va-et-vient" ci-après.*)



## Technologie électronique

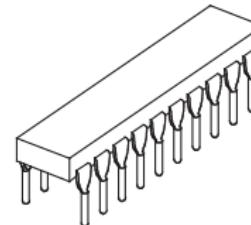
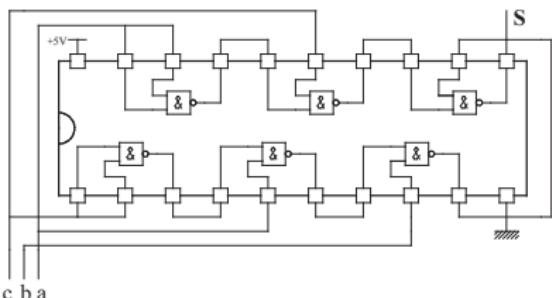
- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\bar{a} \cdot c \cdot \overline{b \cdot a \cdot \bar{c}}}$ .





## Technologie électronique

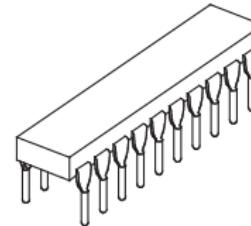
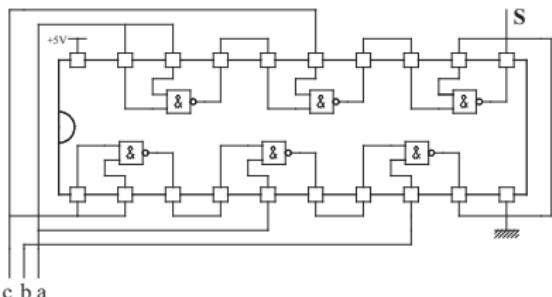
- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\bar{a} \cdot c \cdot b \cdot a \cdot \bar{c}}$ .





## Technologie électronique

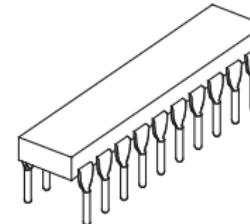
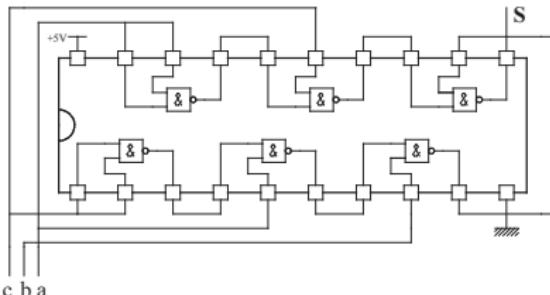
- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\bar{a} \cdot c \cdot b \cdot a \cdot \bar{c}}$ .





## Technologie électronique

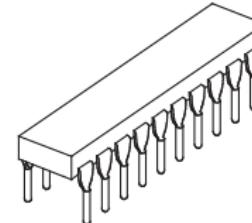
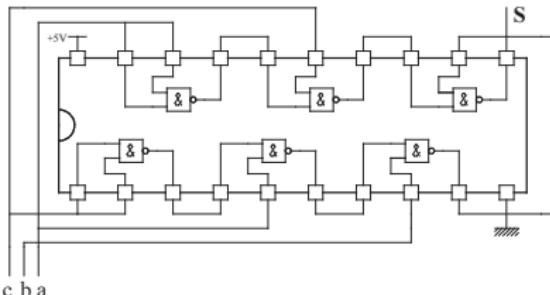
- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\bar{a} \cdot c \cdot b \cdot a \cdot \bar{c}}$ .





## Technologie électronique

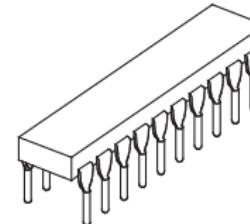
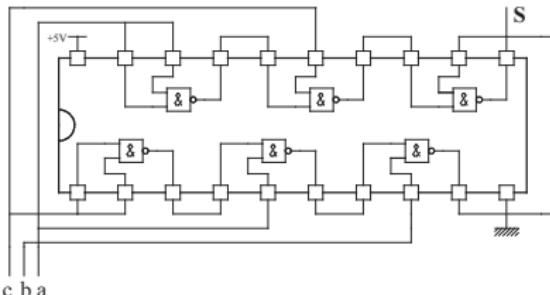
- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\bar{a} \cdot c \cdot b \cdot a \cdot \bar{c}}$ .





## Technologie électronique

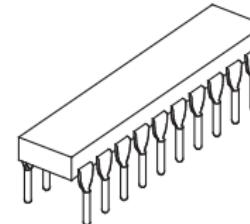
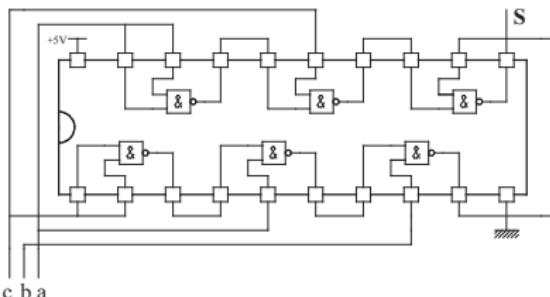
- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\bar{a} \cdot c \cdot b \cdot a \cdot \bar{c}}$ .





## Technologie électronique

- Cette technologie de commande est réservée à des produits fabriqués en grande série.
- Les composants sont soudés sur un circuit imprimé.
- Reprenons la fonction :  $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$  et cherchons à l'implanter avec ce composant. Il faut pour cela la mettre sous la forme de produits complémentés de deux termes.
  - $S = \bar{a} \cdot c + \bar{a} \cdot b + b \cdot c$
  - $S = \bar{a} \cdot c + b \cdot (\bar{a} + c)$
  - $S = \bar{a} \cdot c + b \cdot \overline{a \cdot \bar{c}}$
  - $S = \overline{\overline{\bar{a} \cdot c} \cdot \overline{b \cdot \overline{a \cdot \bar{c}}}}$ .





## Micro-contrôleur et ordinateurs embarqués

### Micro-contrôleur et ordinateurs embarqués

- Les micro-contrôleurs sont constitués de portes logiques et de mémoires mais à la différence des FPGA, leur structure interne se rapproche de celle des ordinateurs avec un microprocesseur.
- De plus ils comportent des périphériques permettant des fonctionnalités avancées (communication série RS232, USB, Ethernet, etc...).
- Une application courante est la commande des hacheurs (variateur électronique de puissance dédiés aux moteurs électriques). Ils se rapprochent des ordinateurs à la différence près que leur performances sont optimisées en fonction de leur utilité dans un système (moins de mémoire et de puissance de calcul).
- La plupart des systèmes intelligents en possèdent un ou plusieurs (voitures modernes, téléphones portables, AR drones, etc...).

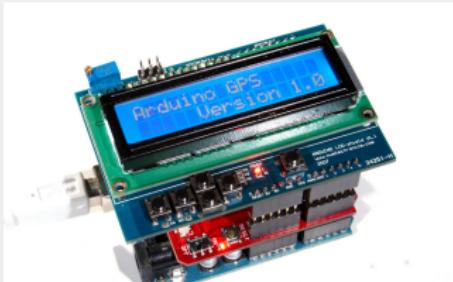




## Micro-contrôleur et ordinateurs embarqués

### Micro-contrôleur et ordinateurs embarqués

- Les micro-contrôleurs sont constitués de portes logiques et de mémoires mais à la différence des FPGA, leur structure interne se rapproche de celle des ordinateurs avec un microprocesseur.
- De plus ils comportent des périphériques permettant des fonctionnalités avancées (communication série RS232, USB, Ethernet, etc...).
- Une application courante est la commande des hacheurs (variateur électronique de puissance dédiés aux moteurs électriques). Ils se rapprochent des ordinateurs à la différence près que leur performances sont optimisées en fonction de leur utilité dans un système (moins de mémoire et de puissance de calcul).
- La plupart des systèmes intelligents en possèdent un ou plusieurs (voitures modernes, téléphones portables, AR drones, etc...).





## Micro-contrôleur et ordinateurs embarqués

### Micro-contrôleur et ordinateurs embarqués

- Les micro-contrôleurs sont constitués de portes logiques et de mémoires mais à la différence des FPGA, leur structure interne se rapproche de celle des ordinateurs avec un microprocesseur.
- De plus ils comportent des périphériques permettant des fonctionnalités avancées (communication série RS232, USB, Ethernet, etc...).
- Une application courante est la commande des hacheurs (variateur électronique de puissance dédiés aux moteurs électriques). Ils se rapprochent des ordinateurs à la différence près que leur performances sont optimisées en fonction de leur utilité dans un système (moins de mémoire et de puissance de calcul).
- La plupart des systèmes intelligents en possèdent un ou plusieurs (voitures modernes, téléphones portables, AR drones, etc...).





## Micro-contrôleur et ordinateurs embarqués

### Micro-contrôleur et ordinateurs embarqués

- Les micro-contrôleurs sont constitués de portes logiques et de mémoires mais à la différence des FPGA, leur structure interne se rapproche de celle des ordinateurs avec un microprocesseur.
- De plus ils comportent des périphériques permettant des fonctionnalités avancées (communication série RS232, USB, Ethernet, etc...).
- Une application courante est la commande des hacheurs (variateur électronique de puissance dédiés aux moteurs électriques). Ils se rapprochent des ordinateurs à la différence près que leur performances sont optimisées en fonction de leur utilité dans un système (moins de mémoire et de puissance de calcul).
- La plupart des systèmes intelligents en possèdent un ou plusieurs (voitures modernes, téléphones portables, AR drones, etc...).

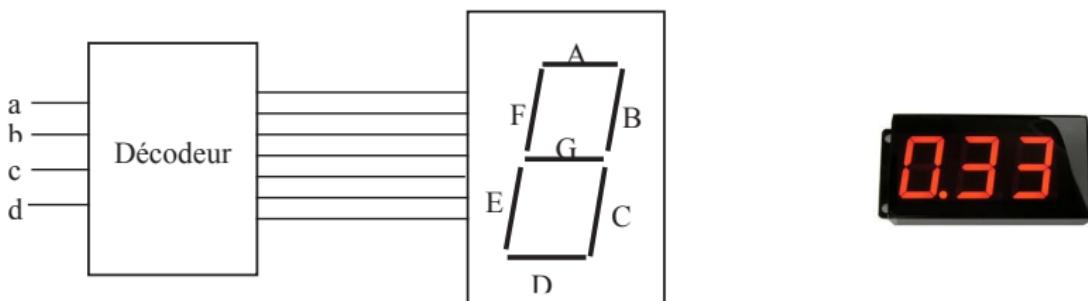




## Codage, décodage, transcodage d'informations

### Commande d'un afficheur 7 segments

- Un afficheur permet d'inscrire un chiffre décimal par l'allumage d'une combinaison de diodes électroluminescentes ou de cristaux liquides.
- Les variables d'entrées représentent le chiffre décimal codé en DCB (décimal codé binaire).

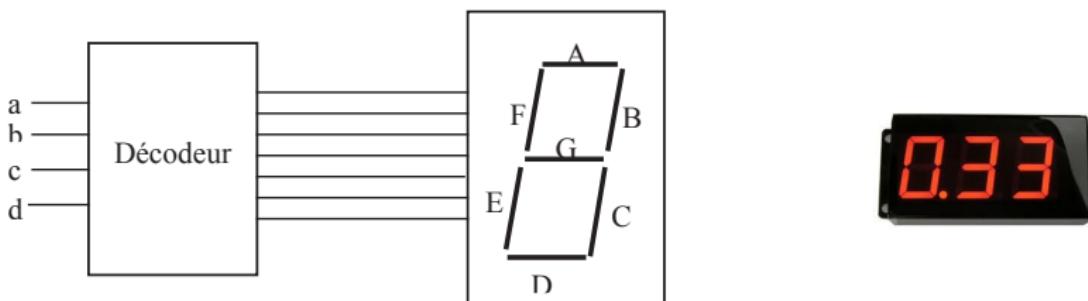




## Codage, décodage, transcodage d'informations

### Commande d'un afficheur 7 segments

- Un afficheur permet d'inscrire un chiffre décimal par l'allumage d'une combinaison de diodes électroluminescentes ou de cristaux liquides.
- Les variables d'entrées représentent le chiffre décimal codé en DCB (décimal codé binaire).





## Codage, décodage, transcodage d'informations

Commande d'un afficheur 7 segments :

	d	c	b	a	A	B	C	D	E	F	G
0	0	0	0	0							
1	0	0	0	1							
2	0	0	1	0							
3	0	0	1	1							
4	0	1	0	0							
5	0	1	0	1							
6	0	1	1	0							
7	0	1	1	1							
8	1	0	0	0							
9	1	0	0	1							



## Codage, décodage, transcodage d'informations

Commande d'un afficheur 7 segments :

	d	c	b	a	A	B	C	D	E	F	G
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

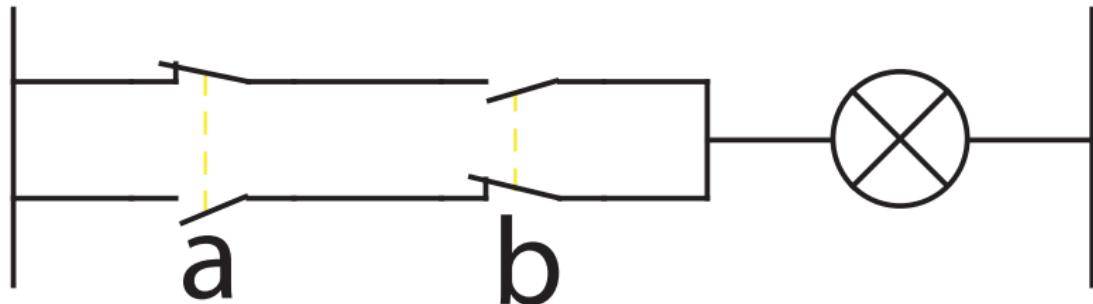
## Commande de systèmes simples

### Definition (Système combinatoire)

Un système est dit combinatoire lorsqu'une même combinaison des entrées donne toujours la même sortie, (la même cause produit toujours le même effet et l'effet disparaît dès que la cause disparaît).



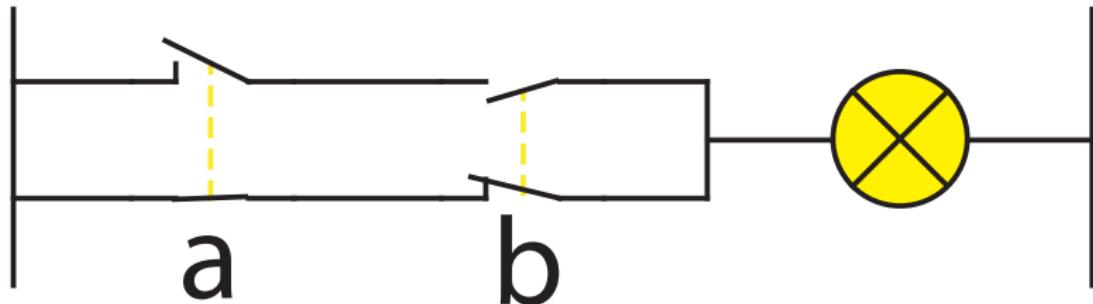
## Commande de systèmes simples



$a$	$b$	$S$
0	0	0
0	1	1
1	0	1
1	1	0

$$S = \bar{a} \cdot b + a \cdot \bar{b} = a \oplus b.$$

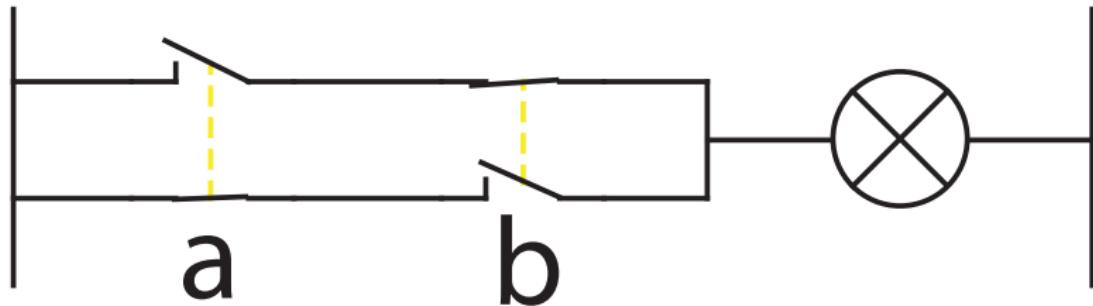
## Commande de systèmes simples



$a$	$b$	$S$
0	0	0
0	1	1
1	0	1
1	1	0

$$S = \bar{a} \cdot b + a \cdot \bar{b} = a \oplus b.$$

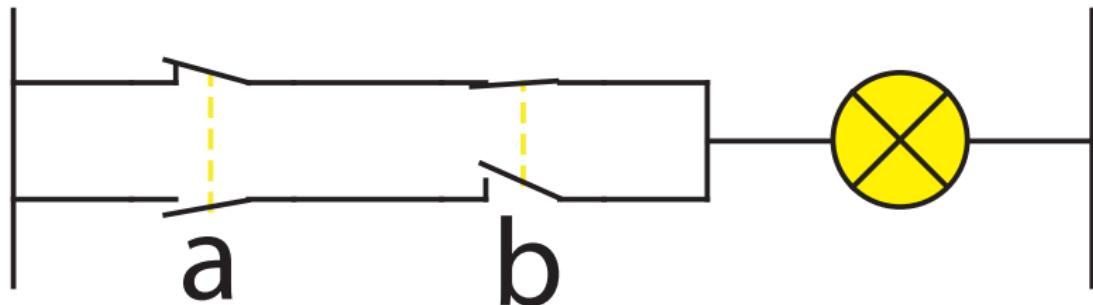
## Commande de systèmes simples



$a$	$b$	$S$
0	0	0
0	1	1
1	0	1
1	1	0

$$S = \bar{a} \cdot b + a \cdot \bar{b} = a \oplus b.$$

## Commande de systèmes simples



$a$	$b$	$S$
0	0	0
0	1	1
1	0	1
1	1	0

$$S = \bar{a} \cdot b + a \cdot \bar{b} = a \oplus b.$$