

Informatique-Electronique - AGIL Elec B

# Électronique numérique : Circuits Combinatoires

Renato Martins, ICB UMR CNRS  
Université Bourgogne Europe, 2025



# Information

---

- 5 séances (dont 2 séances de correction et exercices)
- Contrôle terminal: 10 décembre 2025



# Remerciements

---



- Les slides du cours sont basés pour la plupart sur le support gentillement mis à disposition par **Amira Bousselmi** et par de nombreuses autres personnes.
- Je n'ai pas crédité ces personnes dans la plupart des slides (ce qui n'est pas bien...mes excuses.)

# Table des Matières

---

- **Chapitre 1** : Numérisation et arithmétique binaire
- **Chapitre 2** : Portes et logigrammes
- **Chapitre 3** : Bases de l'algèbre de Boole
- **Chapitre 4** : Simplification des fonctions logiques
- **Chapitre 5** : Circuits combinatoires de base
- **Chapitre 6** : Circuits séquentiels

---

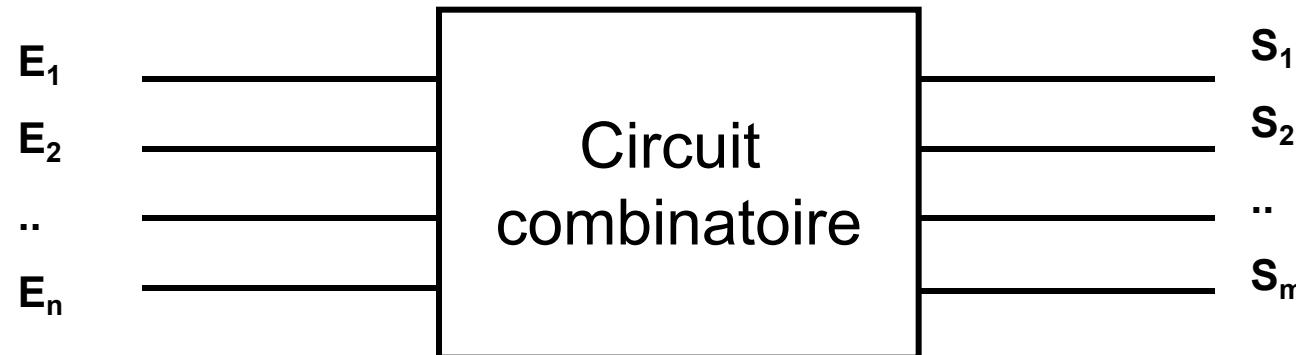
# Chapitre 5 :

# Circuits Combinatoires Elementaires

# Circuits Combinatoires

- Un circuit combinatoire est un circuit numérique dont **les sorties** dépendent uniquement des entrées
  - **Pas de mémorisation de l'état précédent**

$$S_i = F(E_1, E_2, \dots, E_n)$$



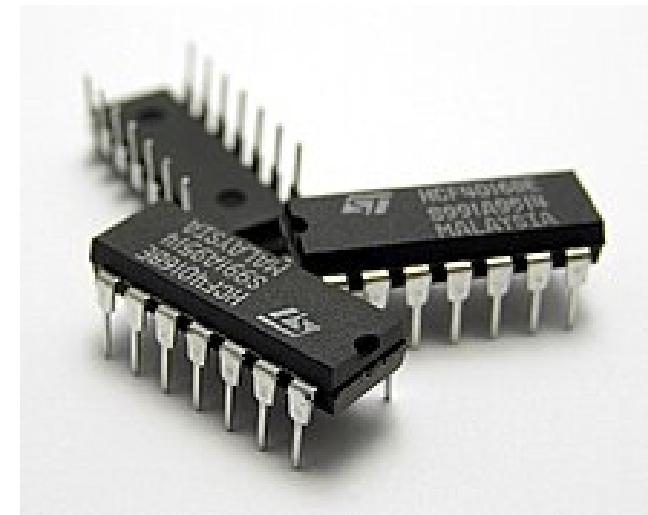
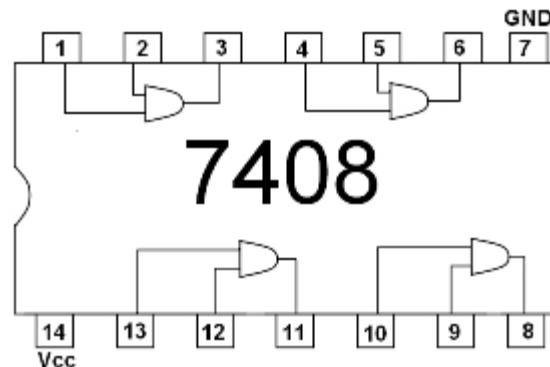
- C'est possible d'utiliser des circuits combinatoires pour réaliser d'autres circuits **plus complexes**.

# Circuits Combinatoires Elementaires

Pour réaliser un circuit logique combinatoire, le concepteur doit utiliser plusieurs portes logiques élémentaires.

Pour faciliter sa tâche, les fabricants fournissent des circuits sous forme intégrés.

PAR EXEMPLE EN PEUT PRÉSENTER LA PORTE LOGIQUE AND AVEC CIRCUIT INTEGRÉ 7408.



Il existe plusieurs dispositifs logiques combinatoires couramment utilisé dans les systèmes numériques. On peut citer les codeurs, les décodeurs, les multiplexeurs, les démultiplexeurs, les comparateurs ... Nous allons :

- Etudier les principaux circuits logiques combinatoires utilisés dans les systèmes numériques (tels que : les circuits arithmétiques, les codeurs, les transcodeurs, ...)

# Circuits Combinatoires Elementaires

---

- La transmission de données nécessite fréquemment des opérations arithmétiques, de conversion, de transposage et d'aiguillage.
- On utilise pour cela des circuits combinatoires :
  - Additionneur
  - Soustracteur
  - Multiplexeur : une des X entrées vers 1 sortie
  - Démultiplexeur : 1 entrée vers une des X sorties
  - Décodeur : active une des X sorties selon un code en entrée
  - Codeur : pour 1 entrée active, fournit un code
  - Transcodeur : pour un code A fournit un code B

# Circuits Combinatoires Elementaires

---

## ■ Les opérateurs arithmétiques

- les additionneurs
- les multiplicateurs
- les unités arithmétiques et logiques

## ■ Les opérateurs d'aiguillage

- les multiplexeurs
- les démultiplexeurs

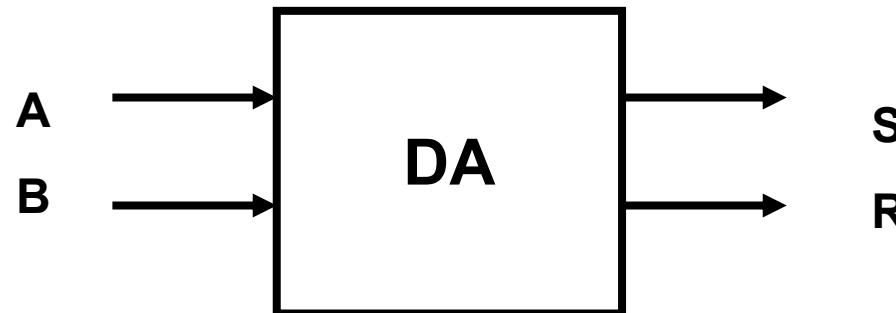
## ■ Les opérateurs de transcodage

- les codeurs
- les décodeurs
- les transcodeurs

# Opérateurs Arithmétiques

## Demi-additionneur

- Le **demi additionneur** est un circuit combinatoire qui permet de réaliser la **somme arithmétique** de deux nombres A et B chacun sur **un bit**.
- A la sortie on va avoir la **somme S et la retenue R** ( Carry ).



- Pour trouver la structure ( le schéma ) de ce circuit on doit en premier dresser sa table de vérité.

# Opérateurs Arithmétiques

## Demi-additionneur

- Table de vérité associée :

A	B	R	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

	somme	retenue
0+0	0	0
0+1	1	0
1+1	0	1

$$R = A \cdot B$$

$$S = \overline{A} \cdot B + A \cdot \overline{B} = A \oplus B$$

# Opérateurs Arithmétiques

## Demi-additionneur

- Table de vérité associée :

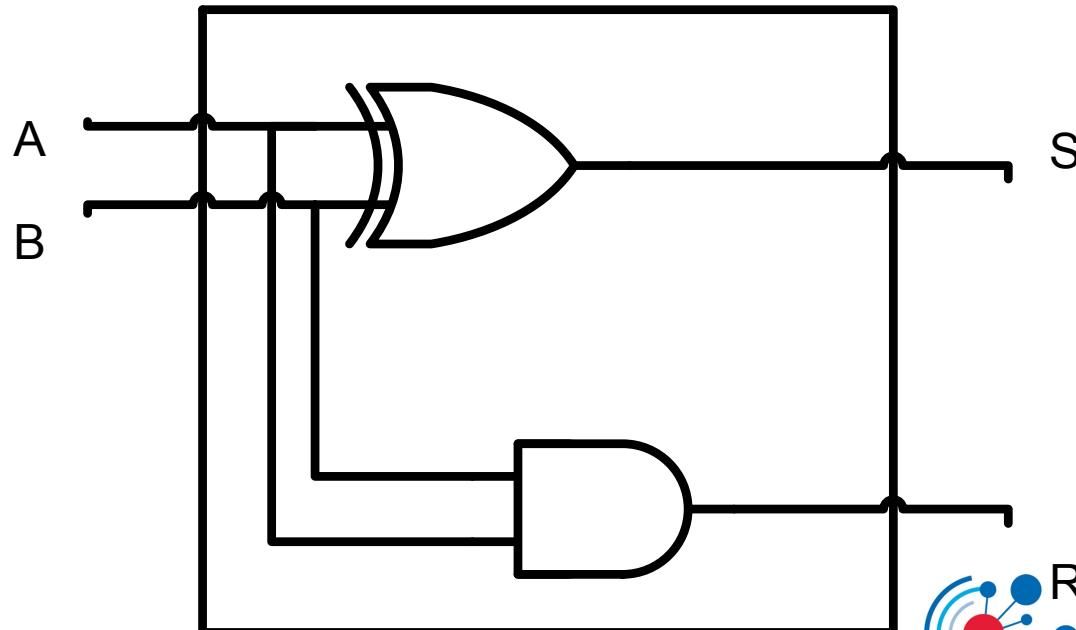
A	B	R	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

- Comment trouver ce circuit ?

$$R = A \cdot B$$

$$S = A \oplus B$$

	somme	retenue
0+0	0	0
0+1	1	0
1+1	0	1



# Opérateurs Arithmétiques

## Additionneur Complet

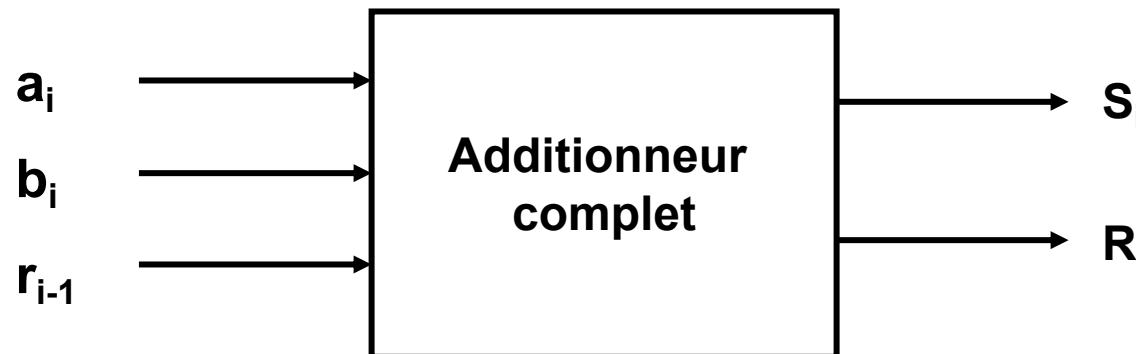
- En binaire lorsqu'on fait une addition il faut tenir en compte de la **retenue entrante**.

$$\begin{array}{ccccccccc} & r_4 & r_3 & r_2 & r_1 & r_0=0 & & r_{i-1} \\ & a_4 & a_3 & a_2 & a_1 & & & A_i \\ + & b_4 & b_3 & b_2 & b_1 & & + & b_i \\ \hline & r_4 & s_4 & s_3 & s_2 & s_1 & & r_i & S_i \end{array}$$

# Opérateurs Arithmétiques

## Additionneur Complet 1 bit

- L'additionneur complet **un bit** possède 3 entrées :
  - $a_i$  : le premier nombre sur un bit.
  - $b_i$  : le deuxième nombre sur un bit.
  - $r_{i-1}$  : la retenue entrante sur un bit.
- Il possède deux sorties :
  - $s_i$  : la somme
  - $R_i$  : la retenue sortante



# Additionneur Complet 1 bit

- Table de vérité d'un additionneur complet sur 1 bit

$$S_i = \overline{A}_i \cdot \overline{B}_i \cdot R_{i-1} + \overline{A}_i \cdot B_i \cdot \overline{R}_{i-1} + A_i \cdot \overline{B}_i \cdot \overline{R}_{i-1} + A_i \cdot B_i \cdot R_{i-1}$$

$$R_i = \overline{A}_i B_i R_{i-1} + A_i \overline{B}_i R_{i-1} + A_i B_i \overline{R}_{i-1} + A_i B_i R_{i-1}$$

<b>a<sub>i</sub></b>	<b>b<sub>i</sub></b>	<b>r<sub>i-1</sub></b>	<b>r<sub>i</sub></b>	<b>s<sub>i</sub></b>
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

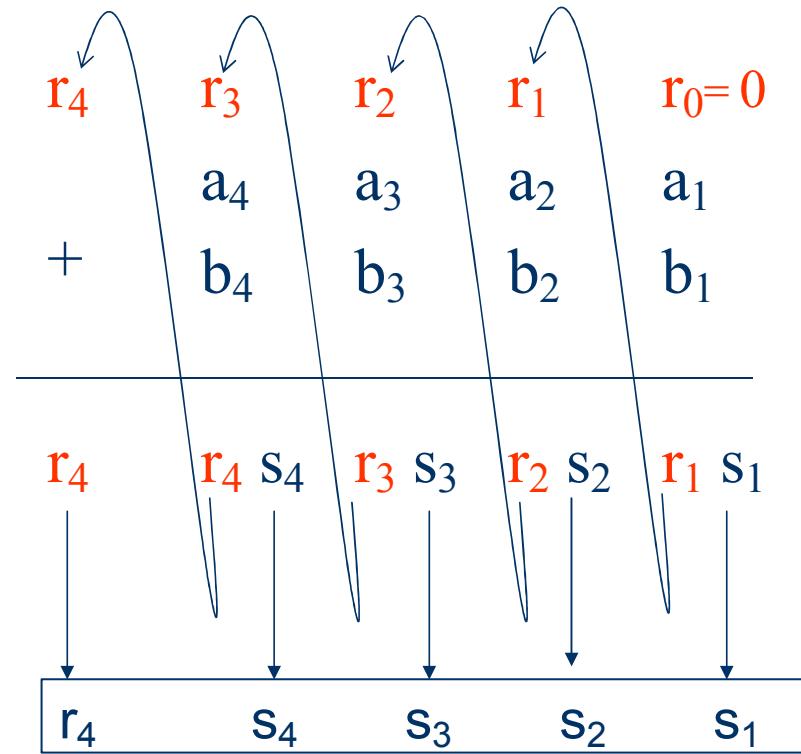
# Exercice

---

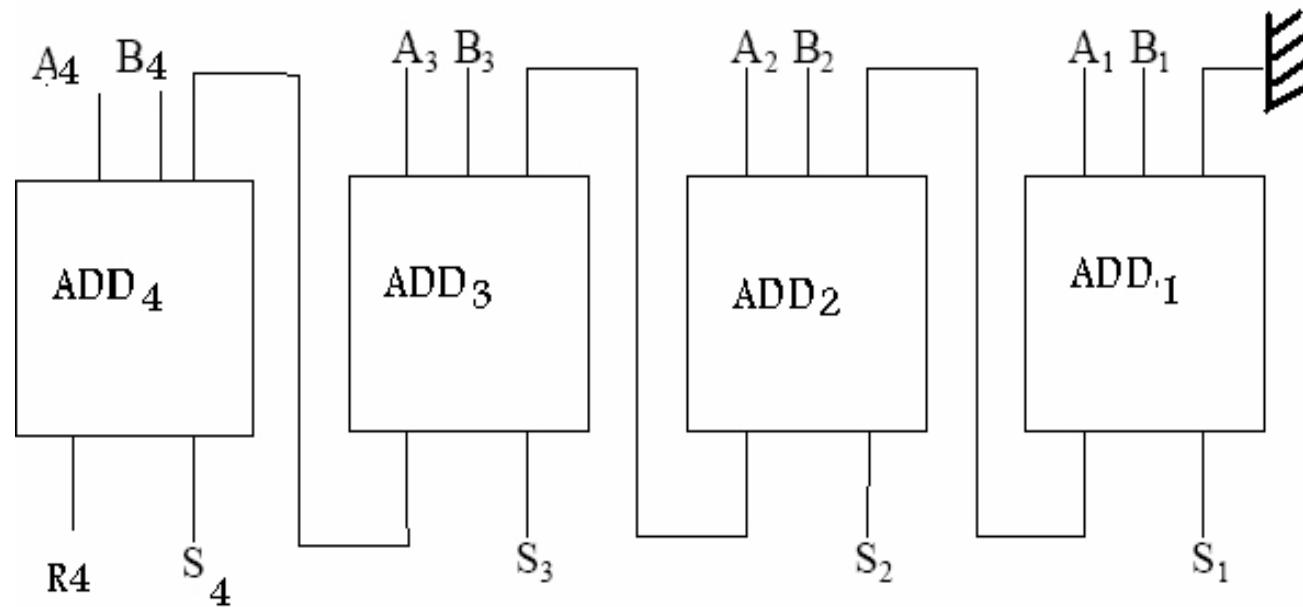
- Pour l'additionneur complet sur 1 bit, trouver la fonction logique simplifiée et son logigramme

$a_i$	$b_i$	$r_{i-1}$	$r_i$	$s_i$
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

# Additionneur Complet 4 bits

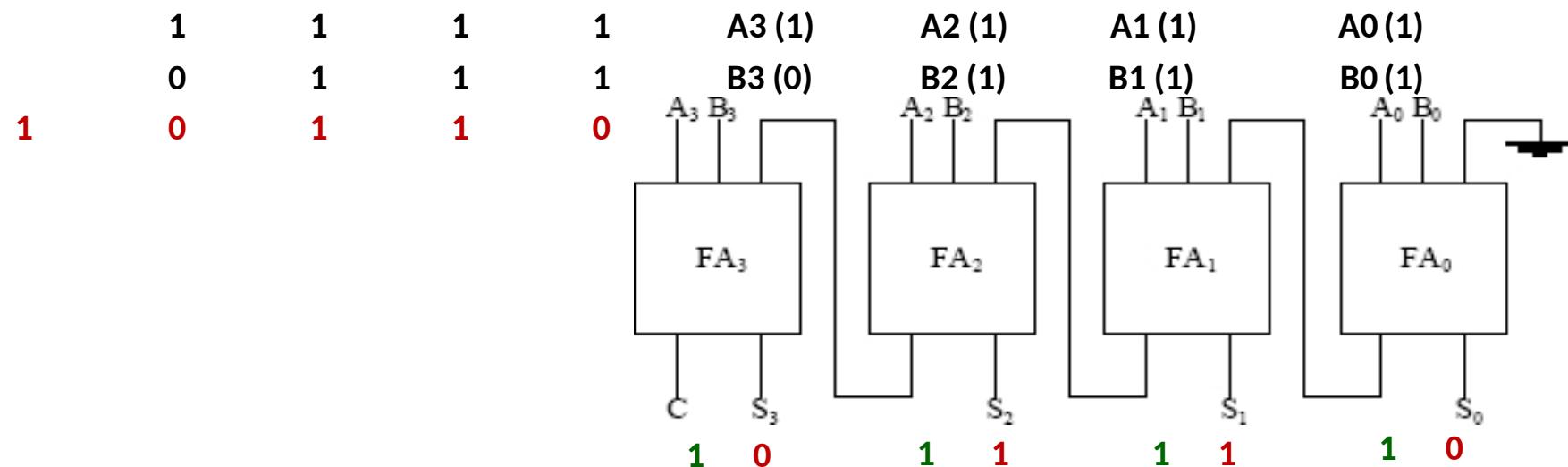


# Additionneur Complet 4 bits (schema)



# Additionneur Complet 4 bits

- Exemple: Additionneur de deux nombres à 4 bits (**Binary Parallel Adder**)



# Opérateurs Arithmétiques

## Soustraction (demi-soustracteur)

- Il obéit aux quatre opérations de la soustraction binaire et possède deux sorties: la différence des entrées A et B ( $A-B$ ) et un empreint E
- Table de vérité :

A	B	D	E
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0
- Exercice : Trouver sa fonction logique en forme SDP, sa forme simplifié et le logigramme.

# Opérateurs Arithmétiques

## Soustraction - soustracteur complet 1 bit

- Il s'agit d'effectuer la différence  $A-B$  de deux nombres  $A$  et  $B$  à un seul bit en tenant compte d'un empreint antérieur  $E_n$ . Il présente deux sorties  $D_n$  et  $E_{n+1}$ .

- Table de vérité :

A	B	$E_n$	$D_n$	$E_{n+1}$
0	0	0	0	0
1	0	0	1	0
0	1	0	1	1
1	1	0	0	0
0	0	1	1	1
1	0	1	0	0
0	1	1	0	1
1	1	1	1	1

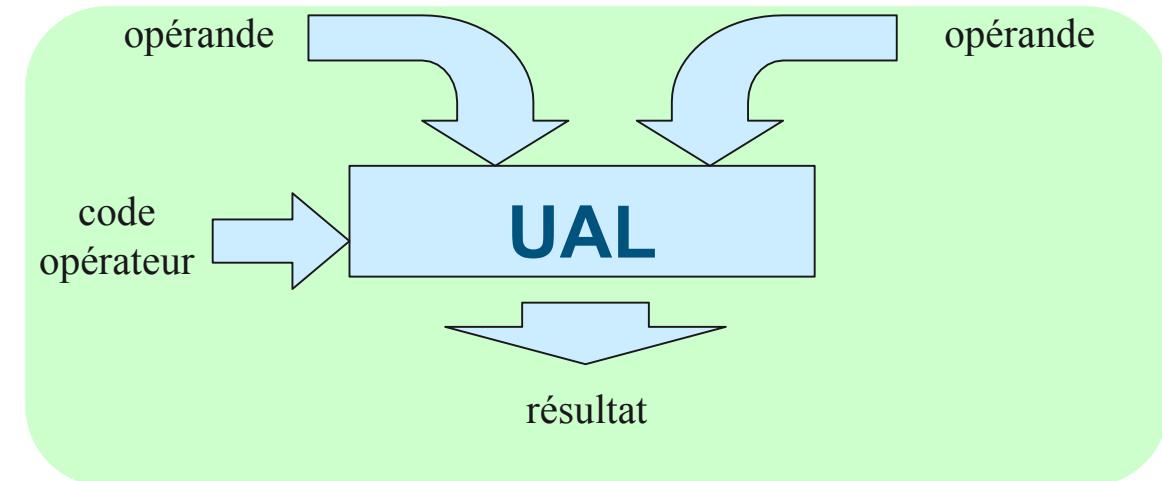
- Exercice : Trouver sa fonction logique en forme SDP, sa forme simplifié et le logigramme.

# Unités Arithmétiques et Logiques (UALs)

## Unités Arithmétiques et Logiques (ou ALU)

- Circuits capables d'effectuer un ensemble d'opérations arithmétiques. Nous pouvons distinguer 4 types de fonction :

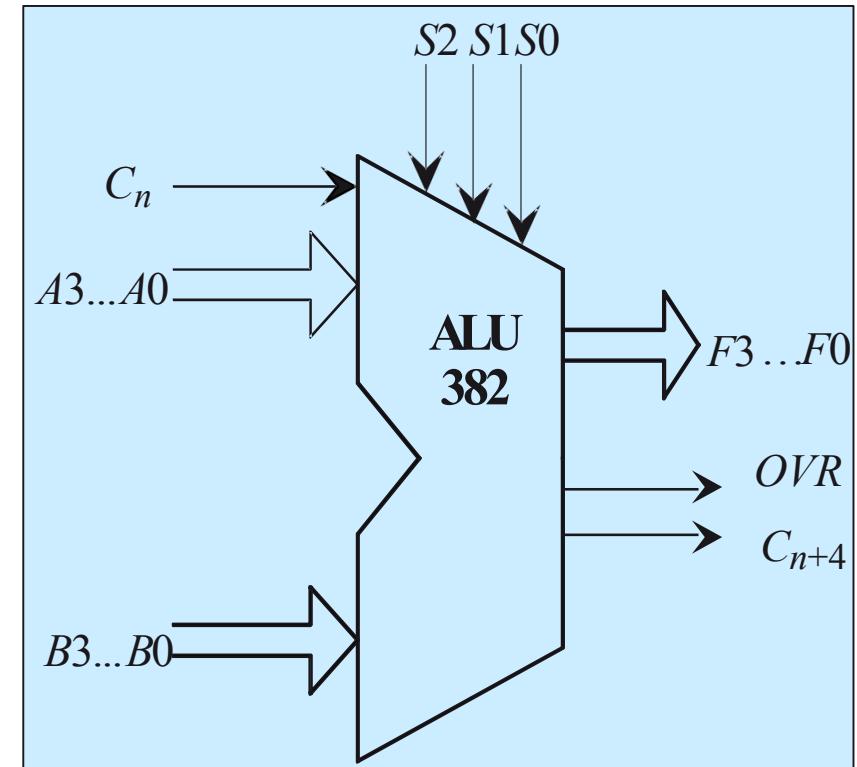
- opérations logiques de base
- comparaison et décalage
- addition et soustraction
- multiplication et division



# Unités Arithmétiques et Logiques (UALs)

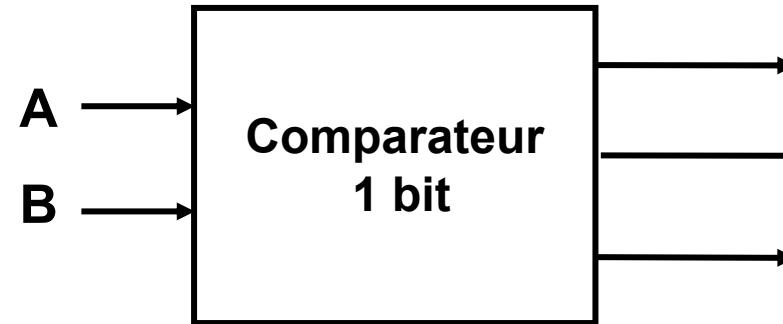
Exemple : le circuit xx382

- Les entrées de commande  $S2 S1 S0$  permettent de sélectionner une opération parmi 8.
  - Opérations arithmétiques:  $A$  plus  $B$ ,  $A$  moins  $B$ ,  $B$  moins  $A$
  - Opérations logiques:  $\text{XOR}(A, B)$ ,  $A$  ou  $B$ ,  $A$  et  $B$ , Mise à 0 (Clear), Mise à 1 (Preset)
- Opérandes:  $A$  et  $B$  sur 4 bits
- $C_n$  : retenue entrante;  $C_{n+4}$  : retenue sortante
- OVR (Overflow): indicateur de dépassement de capacité.



# Opérateurs Arithmétiques (Comparateur)

- C'est un circuit combinatoire qui permet **de comparer** deux nombres binaires A et B.
- Le circuit possède 2 entrées :
  - A : sur un bit
  - B : sur un bit
- Et 3 sorties :
  - $f_e = 1$  si égalité ( $A = B$ )
  - $f_i = 1$  si inférieur ( $A < B$ )
  - $f_s = 1$  si supérieur ( $A > B$ )



# Opérateurs Arithmétiques (Comparateur)

- C'est un circuit combinatoire qui permet **de comparer** deux nombres binaires A et B.
- Le circuit possède 2 entrées :
  - A : sur un bit
  - B : sur un bit
- Et 3 sorties :
  - fe = 1 si égalité ( A = B)
  - fi = 1 si inférieur ( A < B)
  - fs = 1 si supérieur (A > B)
- Exercice : Trouver sa fonction logique en forme SDP, sa forme simplifiée et le logigramme.

A	B	fs	fe	fi
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

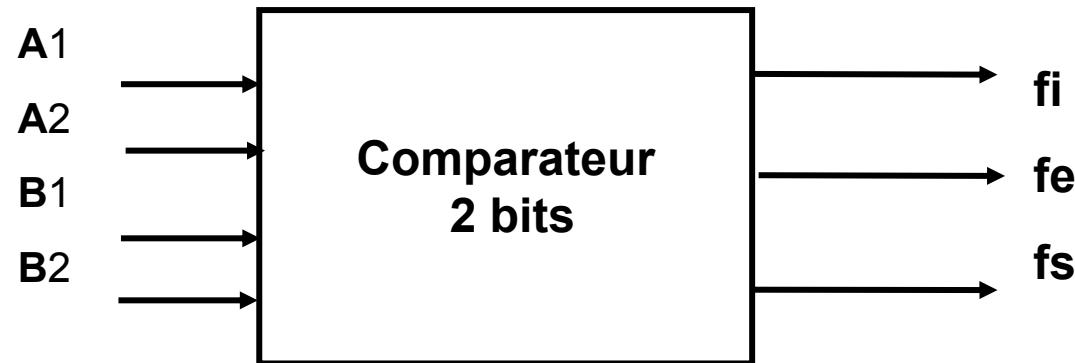
$$fs = A \cdot \bar{B}$$

$$fi = \bar{A}B$$

$$fe = \overline{\bar{A}B} + AB = \overline{\overline{A} \oplus B} = \overline{fs + fi}$$

# Opérateurs Arithmétiques (Comparateur 2 bits)

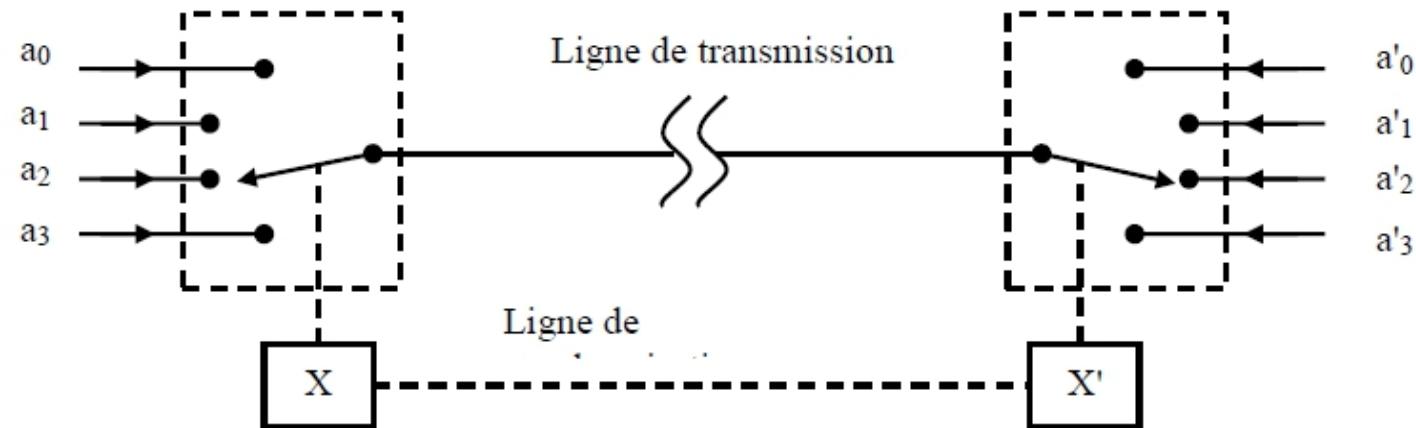
- Circuit qui permet de faire la comparaison entre deux nombres A( $a_2a_1$ ) et B( $b_2b_1$ ) chacun sur deux bits.



- **Exercice :** Construire la table de vérité, la fonction logique en forme SDP, la fonction simplifiée et son logigramme.

# Opérateurs d'Aiguillage (Multiplexeur)

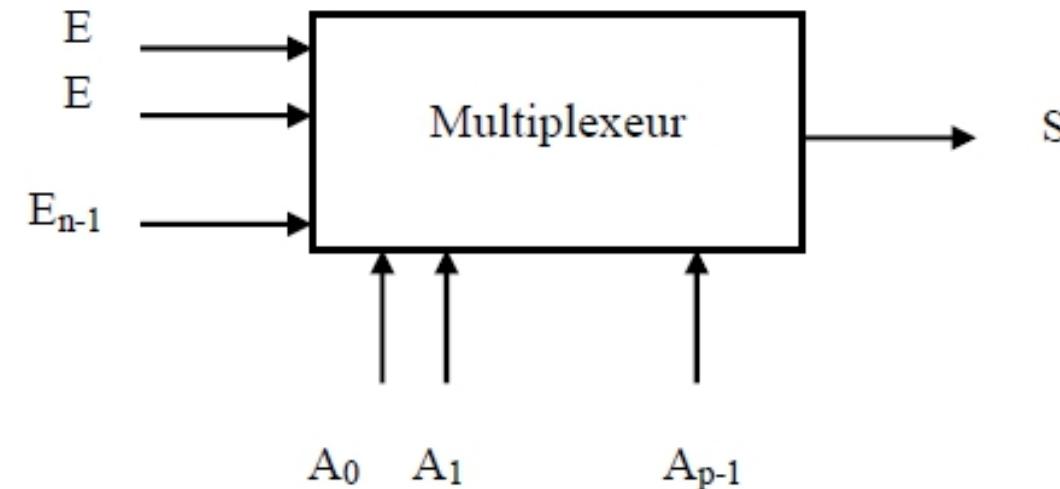
- Pour transmettre des informations en parallèle, cela exige autant de lignes d'informations.
- Pour simplifier la liaison ou pour la rendre moins coûteuse, on réunit au départ les informations sur une seule ligne, c'est le **multiplexage**, et à l'arrivée, on répartit ces informations sur plusieurs lignes, c'est le **démultiplexage**.



- En synchronisant les commandes des sélecteurs  $X$  et  $X'$ , on peut transmettre les informations  $a_0, a_1, a_2$  et  $a_3$  respectivement vers  $a'_0, a'_1, a'_2$  et  $a'_3$ .

# Opérateurs d'Aiguillage (Multiplexeur)

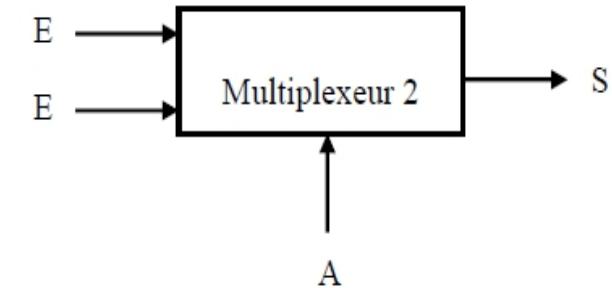
- Le multiplexeur est un circuit possédant plusieurs entrées et **une seule sortie**. Suivant la valeur de l'adresse, une seule entrée est transmise en sortie.



- $p$  est le nombre d'adresses (entrées de sélection)
- $n$  est le nombre d'entrées d'informations
- Un nombre  $p$  d'adresse permet le multiplexage de  $n$  entrées d'informations tel que  $n = 2^p$ .

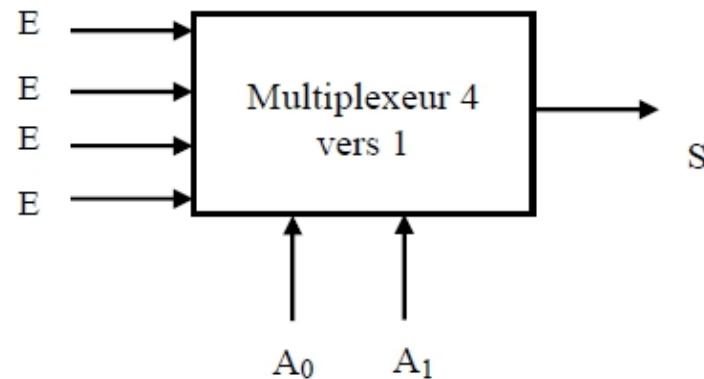
# Multiplexeur (2 entrées 1 sortie)

- On suppose:
  - Si  $A=0$ , alors  $S=E0$ : on transmet le donnée  $E0$
  - Si  $A=1$ , alors  $S=E1$ : on transmet le donnée  $E1$



- **Exercice :** Construire la table de vérité, la fonction logique en forme SDP, la fonction simplifiée et son logigramme.

# Multiplexeur (4 entrées 1 sortie)



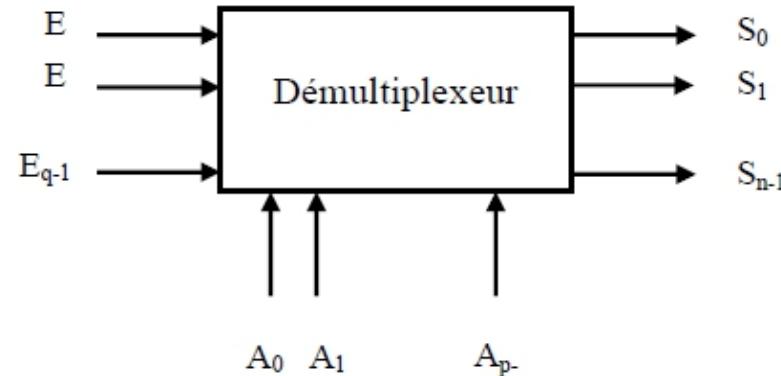
A <sub>0</sub>	A <sub>1</sub>	S
0	0	E <sub>0</sub>
1	0	E <sub>1</sub>
0	1	E <sub>2</sub>
1	1	E <sub>3</sub>

$$S = E_0 \cdot \overline{A_0} \cdot \overline{A_1} + E_1 \cdot A_0 \cdot \overline{A_1} + E_2 \cdot \overline{A_0} \cdot A_1 + E_3 \cdot A_0 \cdot A_1$$

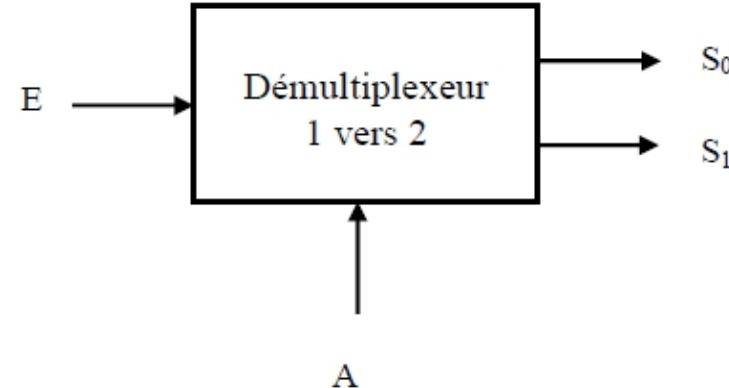
- Exercice : Construire la table de vérité, la fonction logique en forme SDP, la fonction simplifiée et son logigramme.

# Opérateurs d'Aiguillage (Demultiplexeurs)

- Le **démultiplexeur** est un circuit possédant une ou plusieurs entrées et plusieurs sorties. Suivant la valeur de l'adresse, une entrée est transmise vers l'une des sorties.



# Demultiplexeur (1 entrée 2 sorties)



- Suivant la valeur de l'adresse A, l'entrée E est transmise vers l'une des deux sorties S<sub>0</sub> et S<sub>1</sub>.
- Supposons: si  $A=0$  alors  $S_0=E$   
si  $A=1$  alors  $S_1=E$

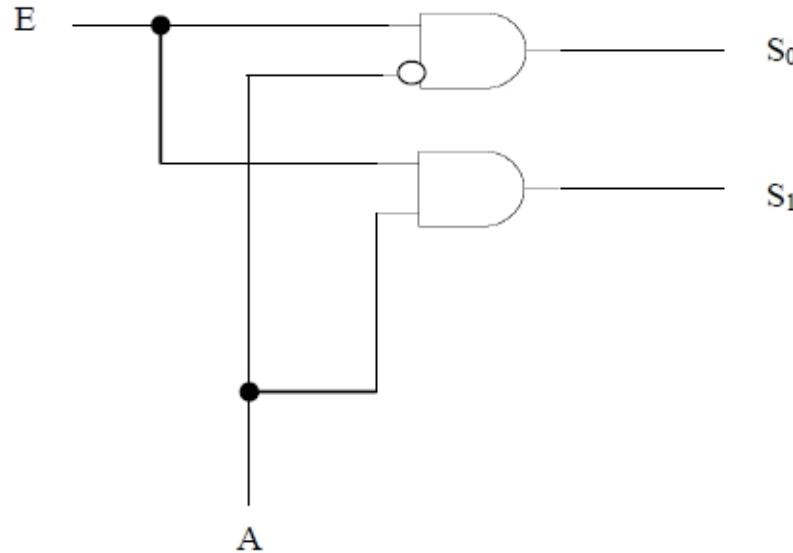
E	A	S <sub>0</sub>	S <sub>1</sub>
0	0	0	0
0	1	0	0
1	0	1	0
1	1	0	1

# Demultiplexeur (1 entrée 2 sorties)

- Logigramme :

$$S_0 = E \bullet \bar{A}$$

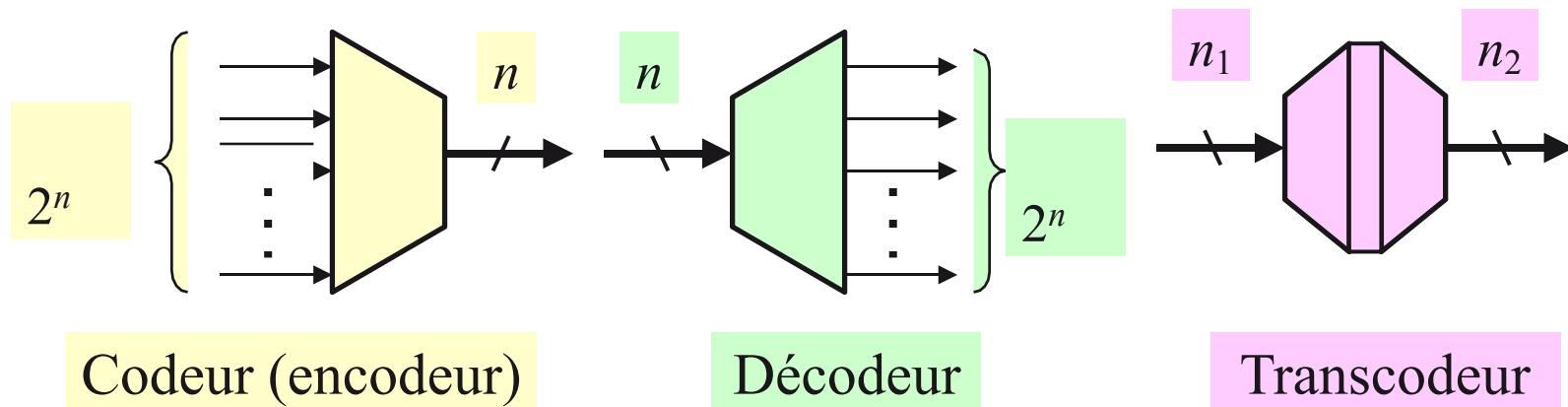
$$S_1 = E \bullet A$$



# Opérateurs Transcodeurs

Un opérateur de transcodage est un circuit transformant une information présente en entrée sous une forme donnée (code 1) en la même information en sortie mais sous une autre forme (code 2)

## Les trois types de transcodeurs

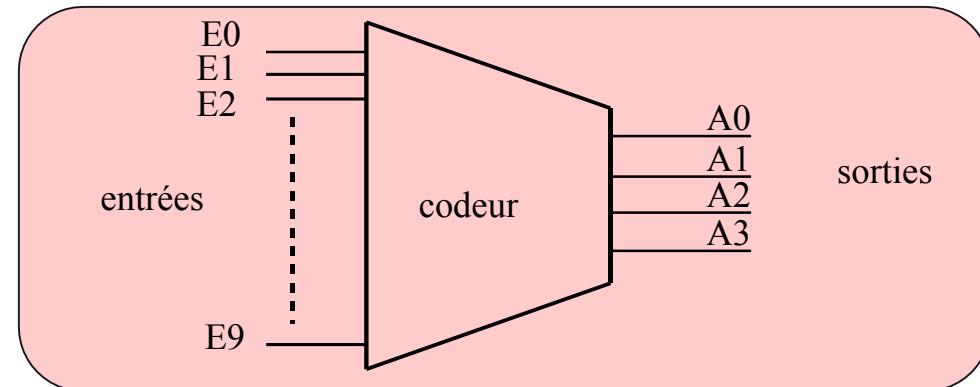


# Opérateurs Transcodeurs (Codeur)

## Codeur

Lorsqu'une entrée (sur les  $M$ ) est activée, les sorties affichent le numéro de l'entrée active dans le code binaire choisi (sur  $n$  bits)

Exemple : codeur décimal vers binaire (10 entrées vers 4 sorties)



Ex: si  $E_5=1$  et  $E_i=0$  pour toutes les autres entrées, alors les sorties affichent  $(A_3, A_2, A_1, A_0)=(0, 1, 0, 1)$ .

# Opérateurs Transcodeurs (Codeur)

## Codeur

C'est un circuit qui traduit les valeurs d'une entrée dans un code choisi. Un codeur (ou encodeur) est un circuit logique qui possède  $2^n$  voies d'entrées dont une seule est activée et N voies de sorties.

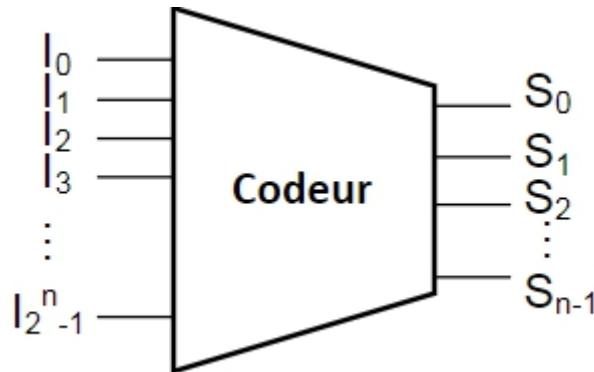
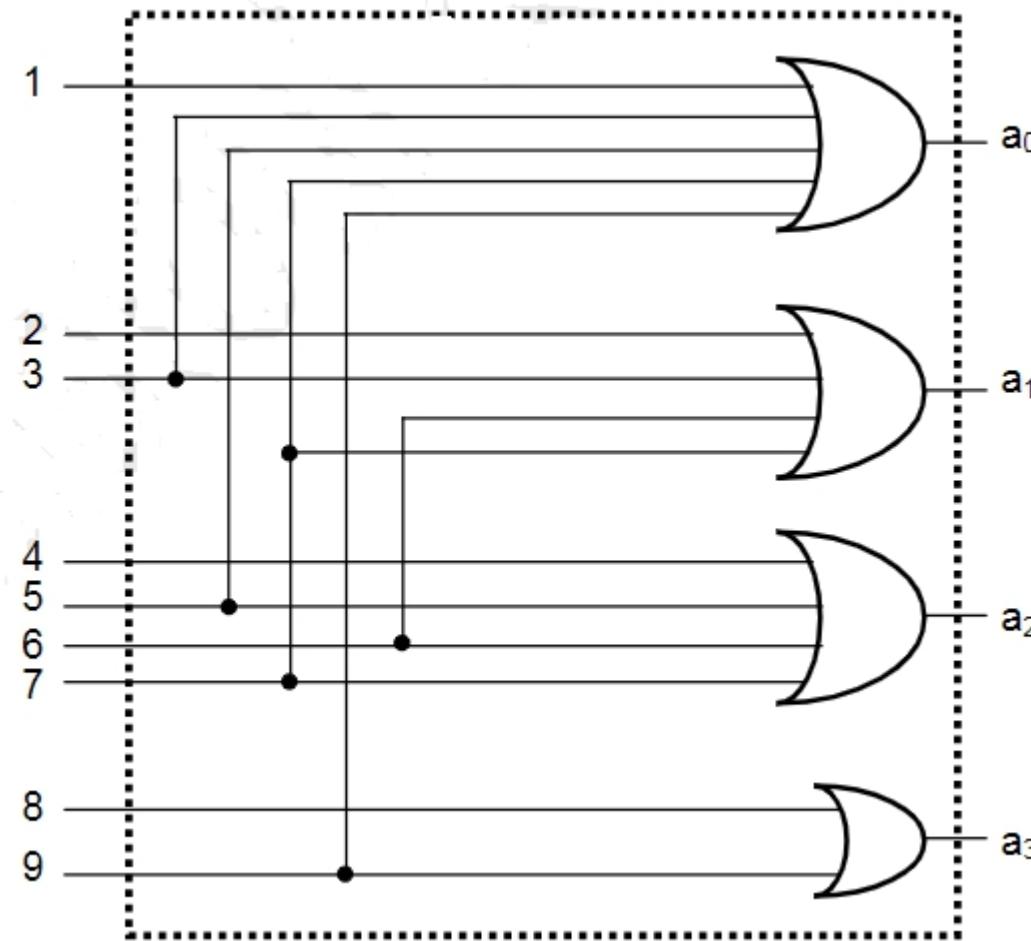


Table de vérité					Equation des sorties	Logigramme
Entrées	Sorties				$a_0=1+3+5+7+9$ $a_1=2+3+6+7$ $a_2=4+5+6+7$ $a_3=8+9$	Circuit intégré : 74LS147
	$a_3$	$a_2$	$a_1$	$a_0$		
0	0	0	0	0		
1	0	0	0	1		
2	0	0	1	0		
3	0	0	1	1		
4	0	1	0	0		
5	0	1	0	1		
6	0	1	1	0		
7	0	1	1	1		
8	1	0	0	0		
9	1	0	0	1		

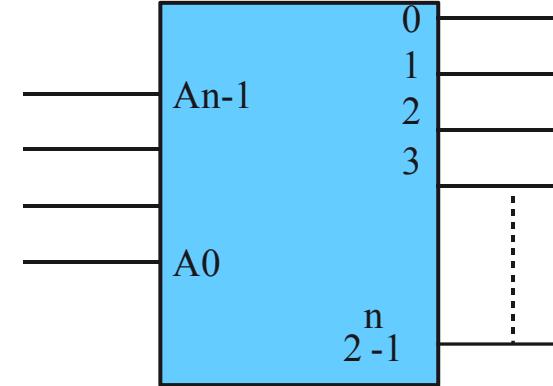
# Opérateurs Transcodeurs

Codeurs

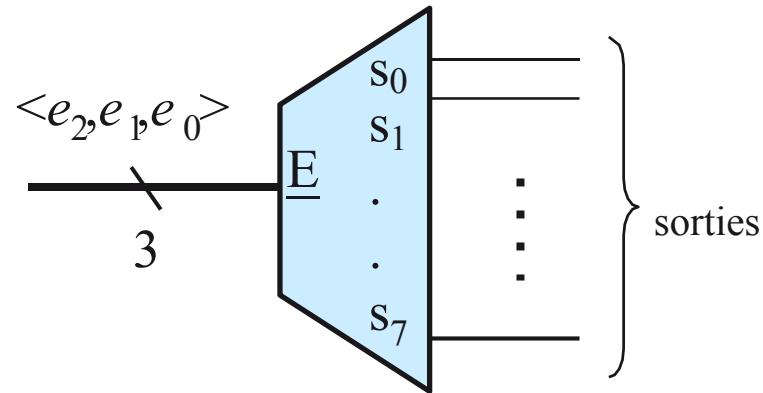


# Opérateurs Transcodeurs (Decodeur)

- $n$  entrées de données
- $N$  sorties avec  $N = 2^n$
- Une seule sortie est active à la fois
- Quand un nombre est codé en binaire pur à l'entrée, c'est la sortie correspondante qui est activée.



Exemple de décodeur binaire "1 parmi 8"



Equations de sortie

$$s_0 = \overline{e_2} \overline{e_1} \overline{e_0}$$

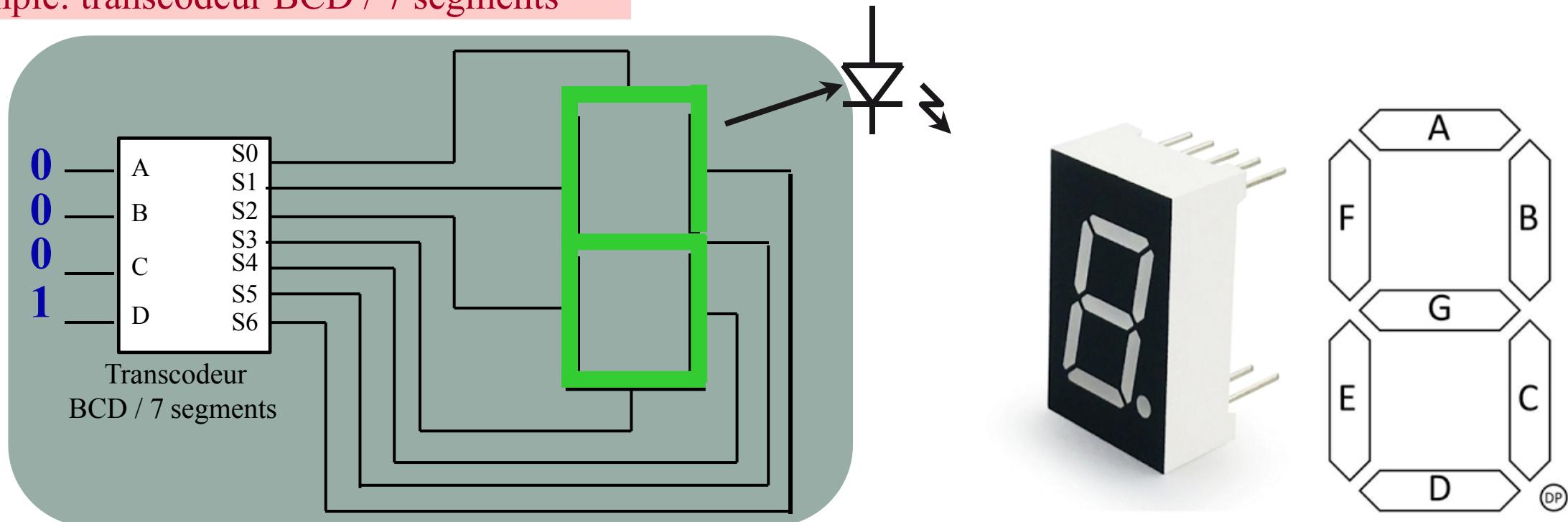
$$s_1 = \overline{e_2} \overline{e_1} e_0$$

$$s_2 = \overline{e_2} e_1 \overline{e_0}$$

etc.

# Opérateurs Transcodeurs

Exemple: transcodeur BCD / 7 segments



Il est souvent nécessaire de visualiser une information codée en binaire sur des afficheurs (7 segments)  
=> convertisseur BCD (*Binary-Coded Decimal*) / 7 segments  
=> convertisseur binaire pur / 7 segments

# Opérateurs Transcodeurs

