Informatique-Electronique - AGIL Elec B

Electronique numérique :
Circuits Combinatoires

Renato Martins, ICB UMR CNRS - Univ. Bourgogne
UFR Sciences & Techniques - IEM, 2024

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.)

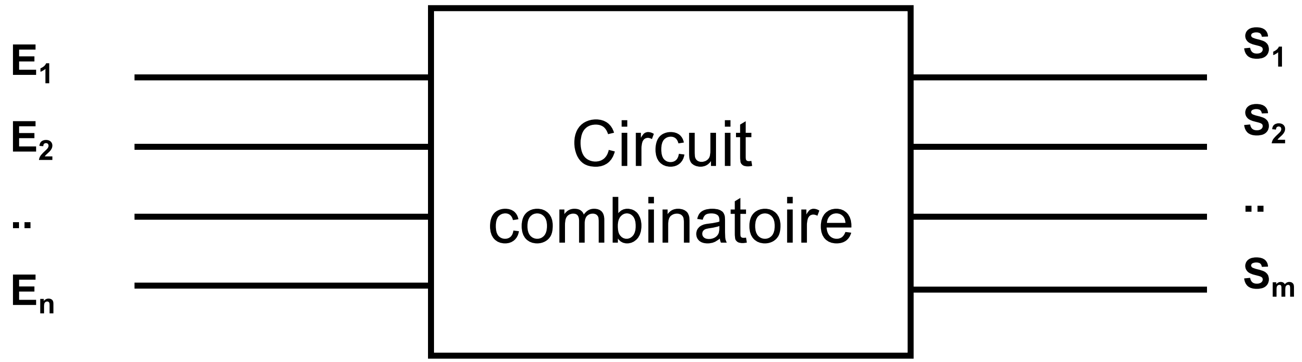
Table des Matières

Chapitre 5 :
Circuits Combinatoires Elementaires

Circuits Combinatoires Elementaires

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

$$S_i = F(E_1,E_2,....,E_n)$$



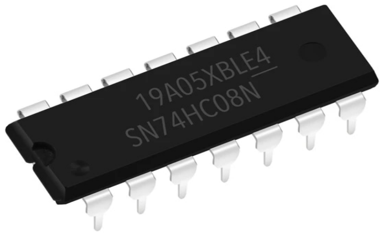
- C'est possible d'utiliser des circuits combinatoires pour réaliser d'autres circuits plus complexes.

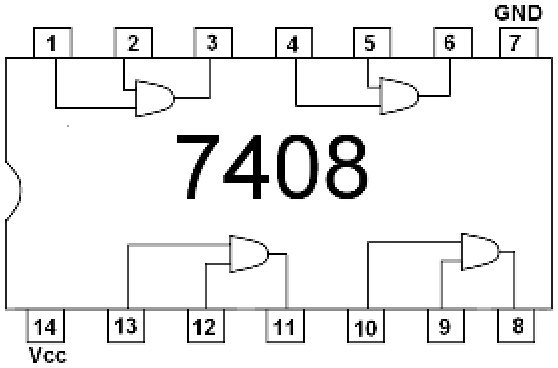
Circuits Combinatoires Elementaires

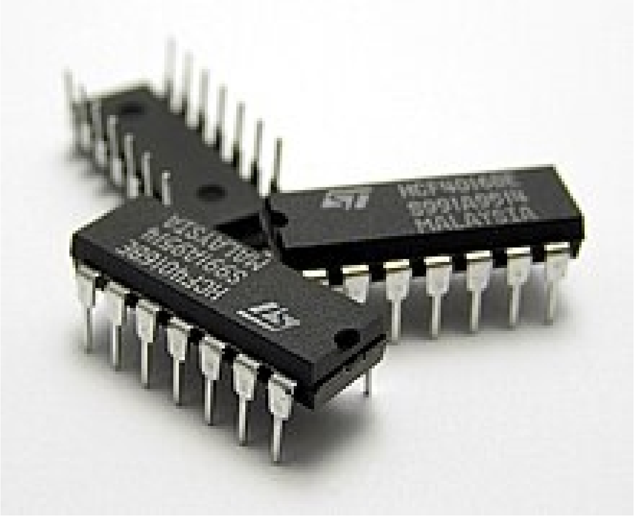
Pour réaliser un circuit logique combinatoire, le concepteur doit utiliser plusieurs portes logiques élémentaires.

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

PAR EXEMPLE EN PEUT PRESENTER LA PORTE LOGIQUE AND AVEC CIRCUIT INTEGRE 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

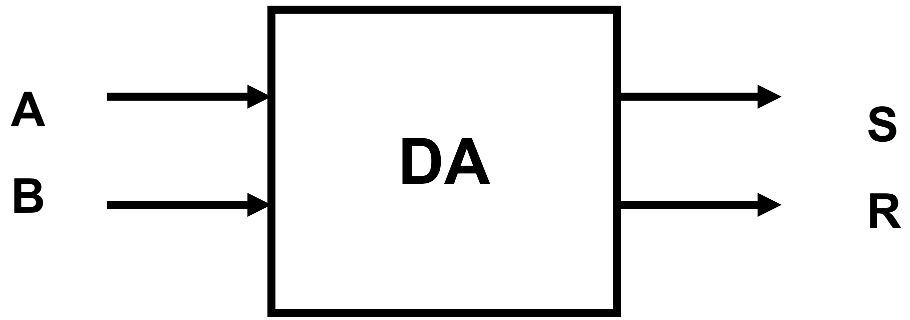
Circuits Combinatoires Elementaires

- **Les opérateurs arithmétiques**
  - les additionneurs
  - les multiplieurs
  - les unités arithmétiques et logiques
- **Les opérateurs d'aiguillage**
  - les multiplexeurs
  - les démultiplexeurs
- **Les opérateurs de comparaison**
- **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 retenu 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 :

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

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

$$R = A.B$$

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

Demi-additionneur

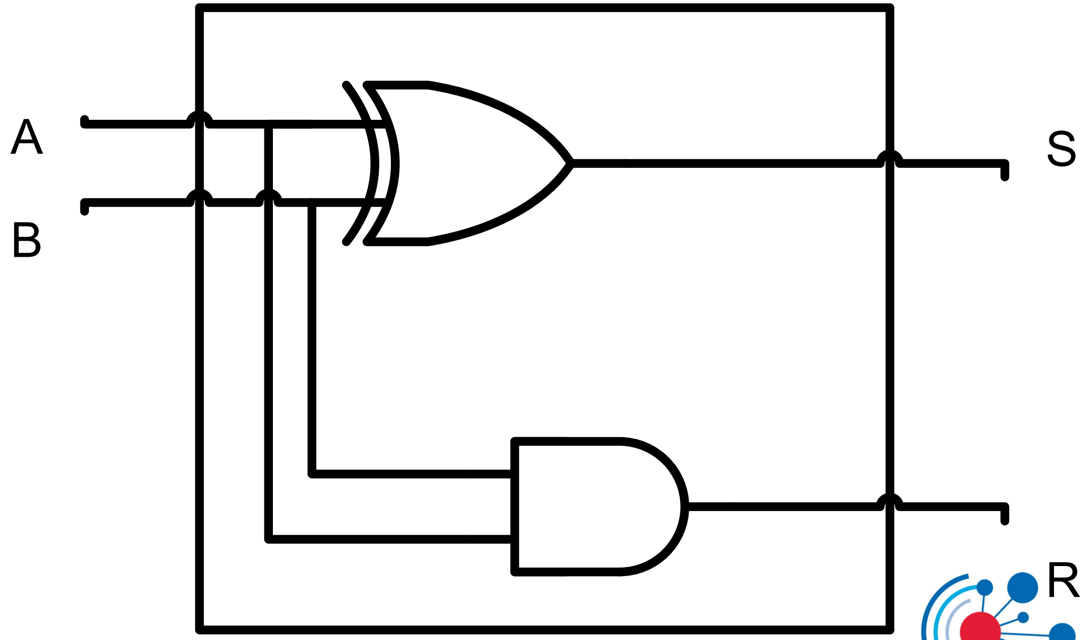
- Table de vérité associée :

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

$$R = A.B$$

$$S = A \oplus B$$



- Comment trouver ce circuit ?

Additionneur Complet

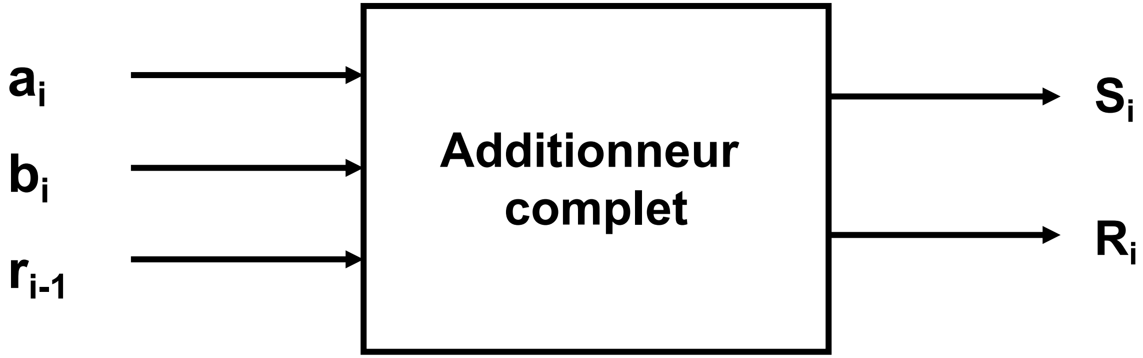
- En binaire lorsqu'on fait une addition il faut tenir en compte de la retenue entrante.

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

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

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

$$S_i = \overline{A_i}.\overline{B_i}.R_{i-1} + \overline{A_i}.B_i.\overline{R_{i-1}} + A_i.\overline{B_i}.\overline{R_{i-1}} + A_i.B_i.R_{i-1}$$

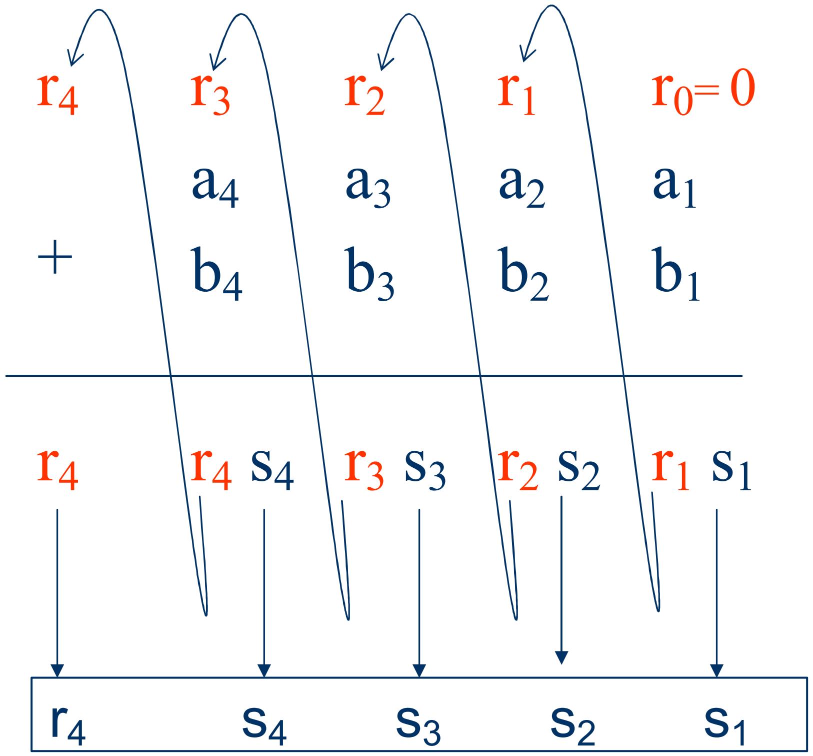
$$R_i = \overline{A_i}B_iR_{i-1} + A_i\overline{B_i}R_{i-1} + A_iB_i\overline{R_{i-1}} + A_iB_iR_{i-1}$$

Exercice

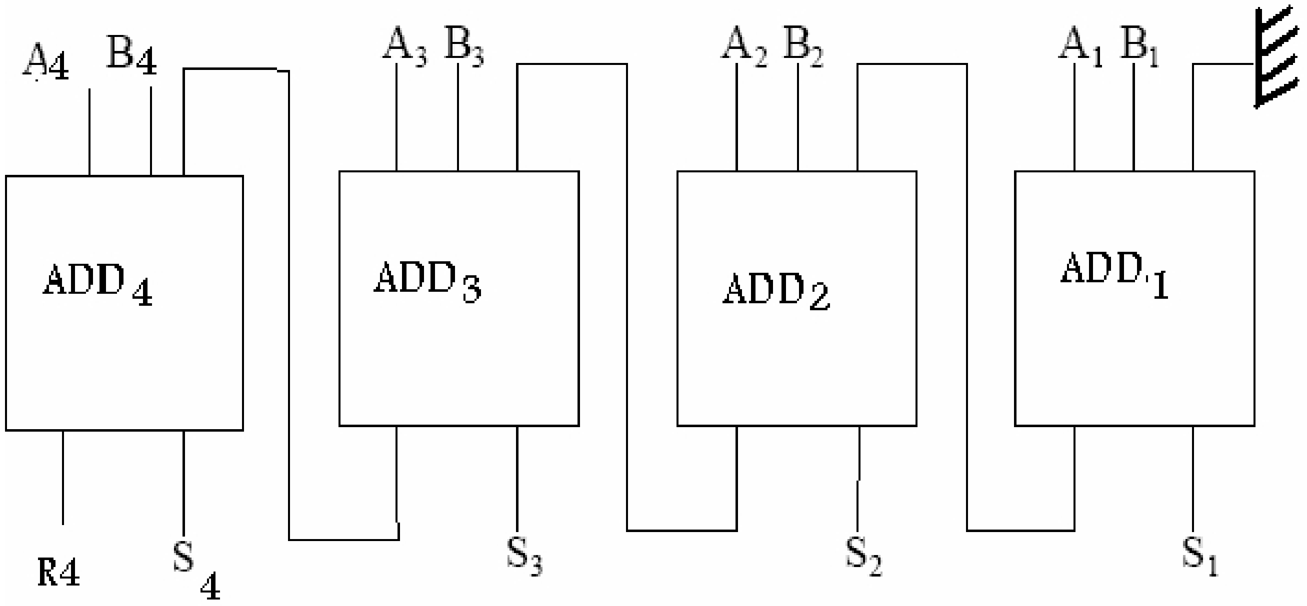
- Pour l'additionneur complet sur 1 bit, trouver la fonction logique simplifiée et son logigramme

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



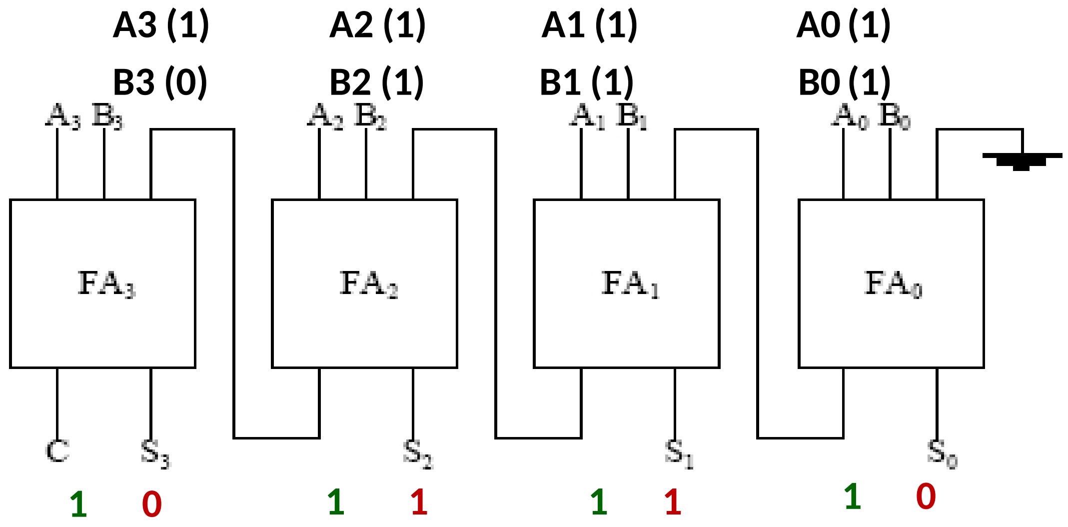
Additionneur Complet 4 bits (schema)



Additionneur Complet 4 bits

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

	1	1	1	1
	0	1	1	1
1	0	1	1	0



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.

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 En. Il présente deux sorties Dn et En+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

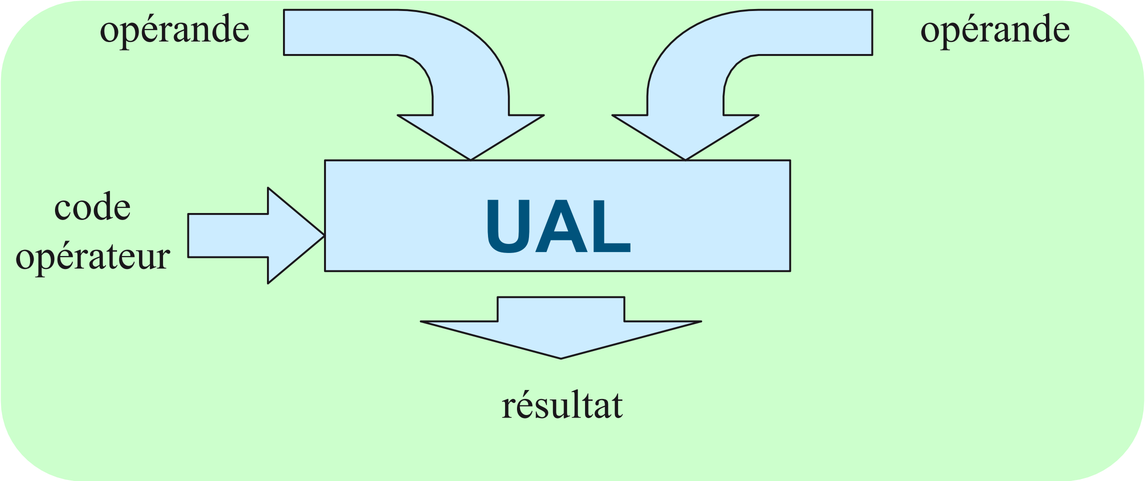
- Exercice : Trouver sa fonction logique en forme SDP, sa forme simplifié et le logigramme.

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 :

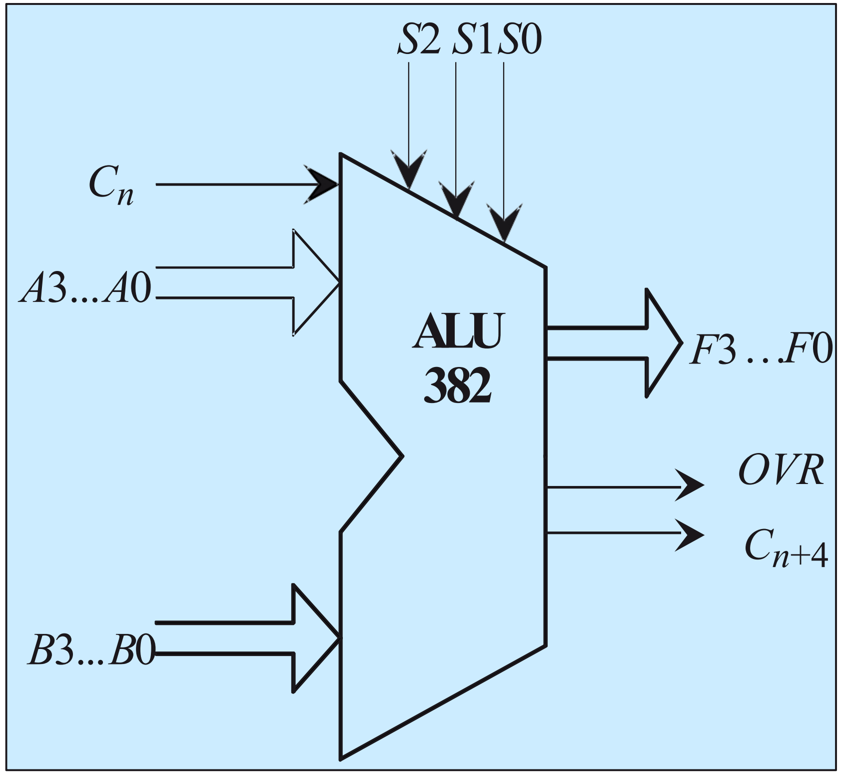
  - opérations logiques de base
  - comparaison et décalage
  - addition et soustraction
  - multiplication et division



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: XOR(*A*,*B*), *A* ou *B*, *A* et *B*, Mise à 0 (Clear), Mise à 1 (Preset)



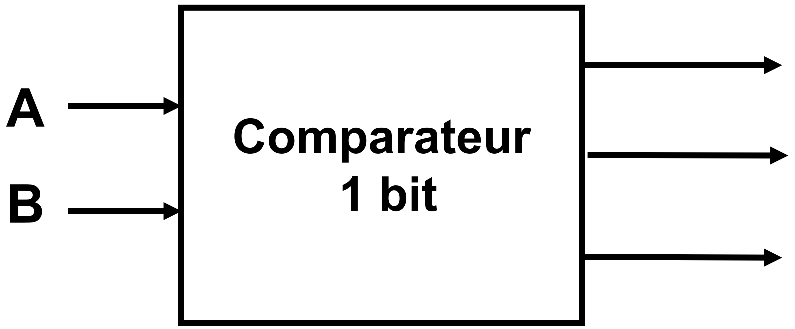
- Opérandes: A et B sur 4 bits
- Cn : retenue entrante; Cn+4 : retenue sortante
- OVR (Overflow): indicateur de dépassement de capacité.

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 :
  - fe = 1 si égalité (A = B)
  - fi = 1 si inférieur (A < B)
  - fs = 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)

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

$$fs = A.\overline{B}$$

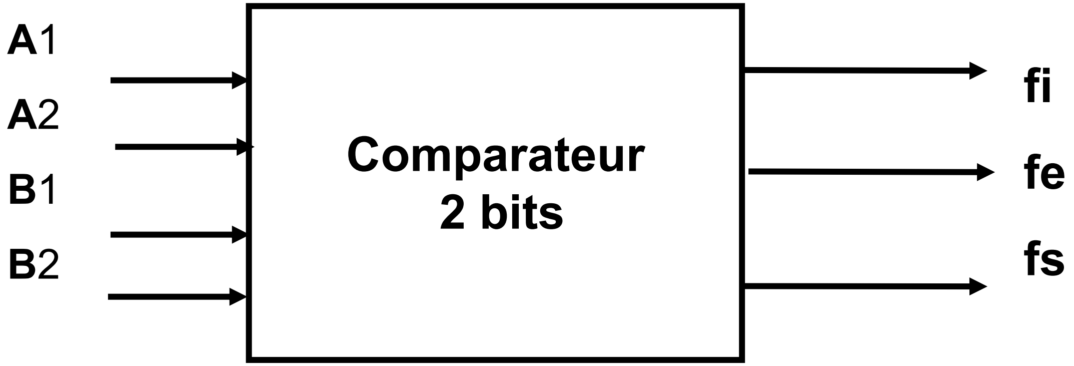
$$fi = \overline{A}B$$

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

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

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.

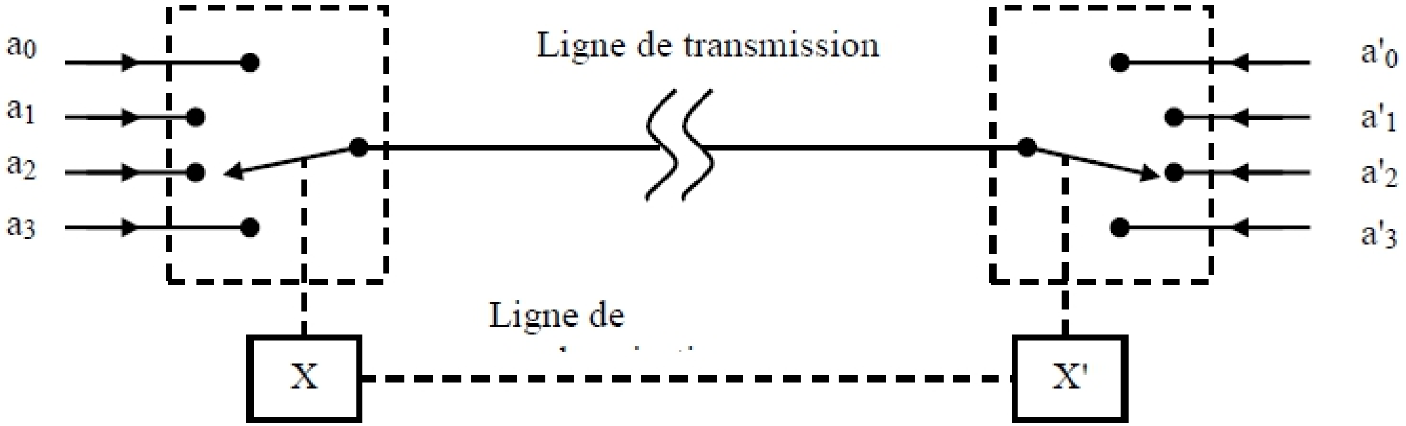


- **Exercice :** Construire la table de verité, la fonction logique en forme SDP, la fonction simplifiée et son logigramme.

Opérateurs d'Aiguillage (Multiplexeur)

- Pour transmettre des informations en parallèle, cela exige autant de lignes d'informations.

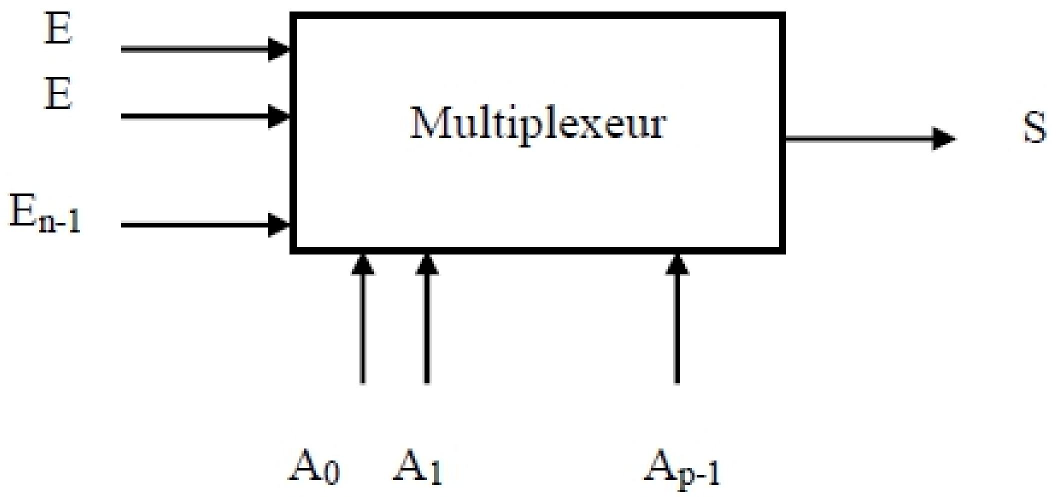
- Pour simplifier la liaison ou pour la rendre moins coûteuse, on réunit au départ les informations sur une seule ligne, c'est **le multiplexage**, et à l'arrivée, on répartit ces informations sur plusieurs lignes, c'est le **démultiplexage**.



- En synchronisant les commandes des sélecteurs X et X', on peut transmettre les informations a0, a1, a2 et a3 respectivement vers a'0, a'1, a'2 et a'3.

Opérateurs d'Aiguillage (Multiplexeur)

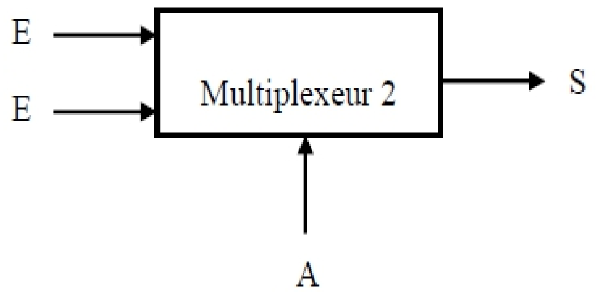
- Le multiplexeur est un circuit possédant plusieurs entrées et **une seule sortie**. Suivant la valeur de l'adresse, une seule entrée est transmise en sortie.



- 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 $n = 2^p$.

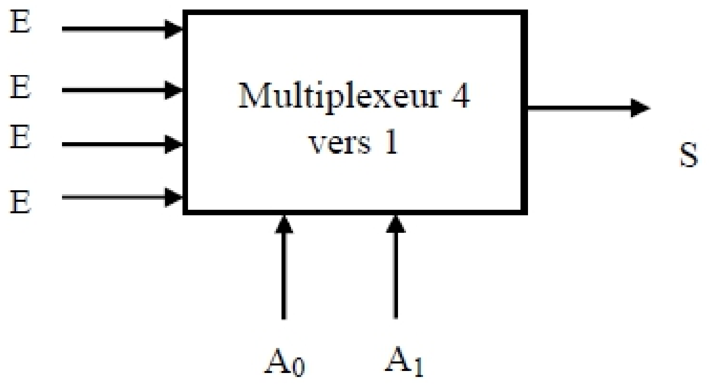
Multiplexeur (2 entrées 1 sortie)

- 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 verité, la fonction logique en forme SDP, la fonction simplifiée et son logigramme.

Multiplexeur (4 entrées 1 sortie)



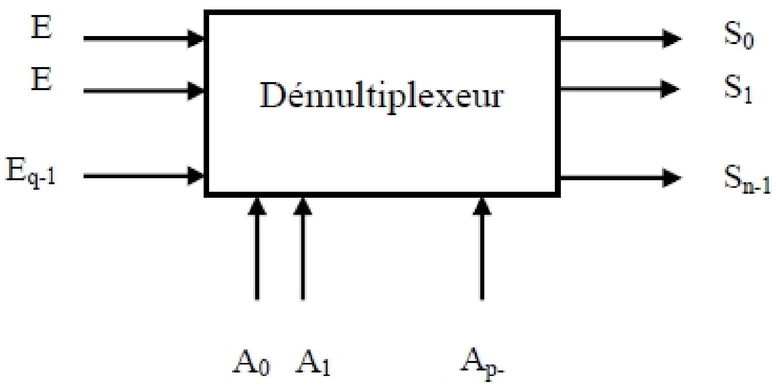
A_0	A_1	S
0	0	E_0
1	0	E_1
0	1	E_2
1	1	E_3

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

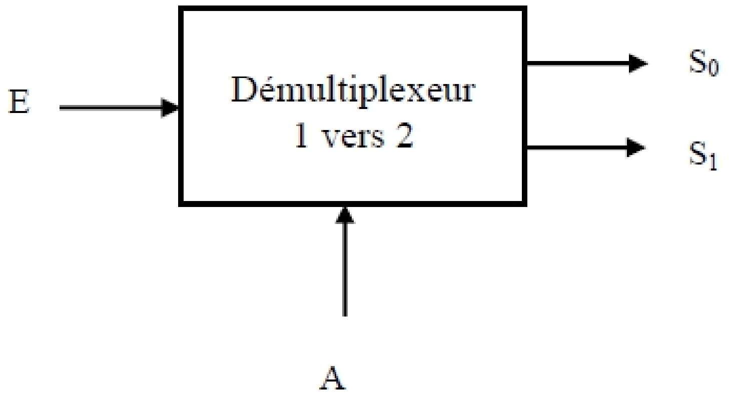
- **Exercice :** Construire la table de verité, la fonction logique en forme SDP, la fonction simplifiée et son logigramme.

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)



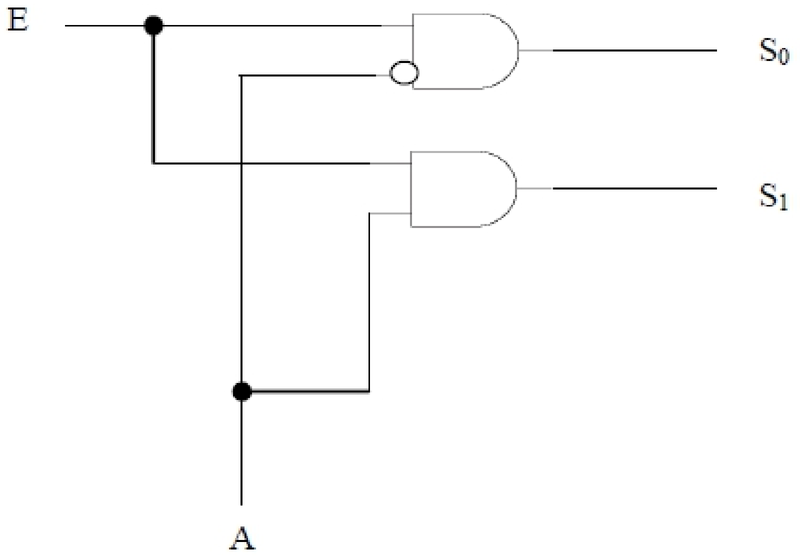
- Suivant la valeur de l'adresse A, l'entrée E est transmise vers l'une des deux sorties S_0 et S_1.
- Supposons: si $A=0$ alors $S_0=E$
  si $A=1$ alors $S_1=E$

E	A	S_0	S_1
0	0	0	0
0	1	0	0
1	0	1	0
1	1	0	1

Demultiplexeur (1 entrée 2 sorties)

- Logigramme :

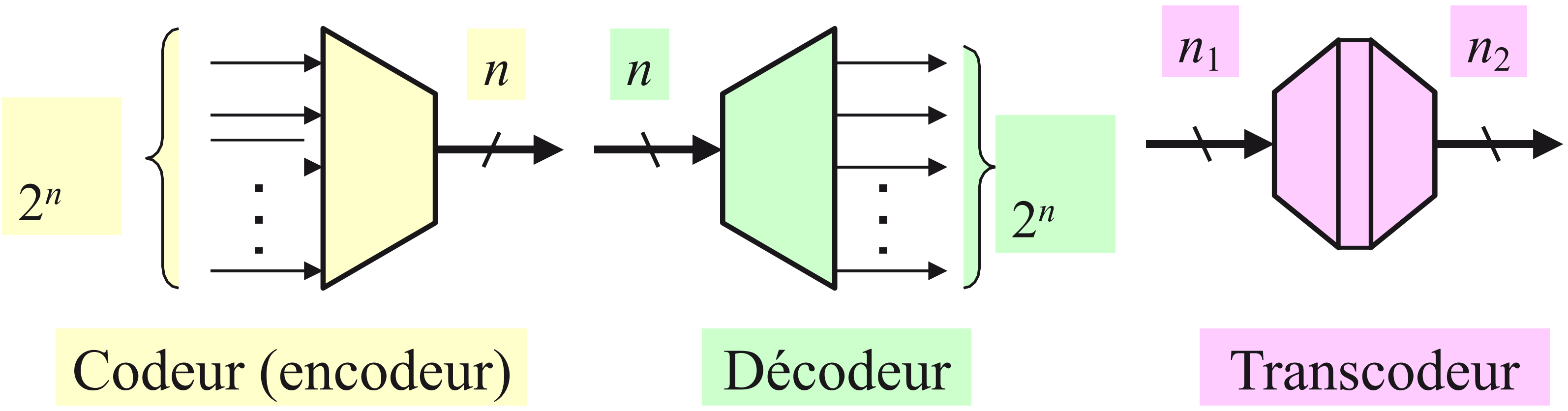
$$S_0 = E \bullet \overline{A}$$
$$S_1 = E \bullet A$$



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

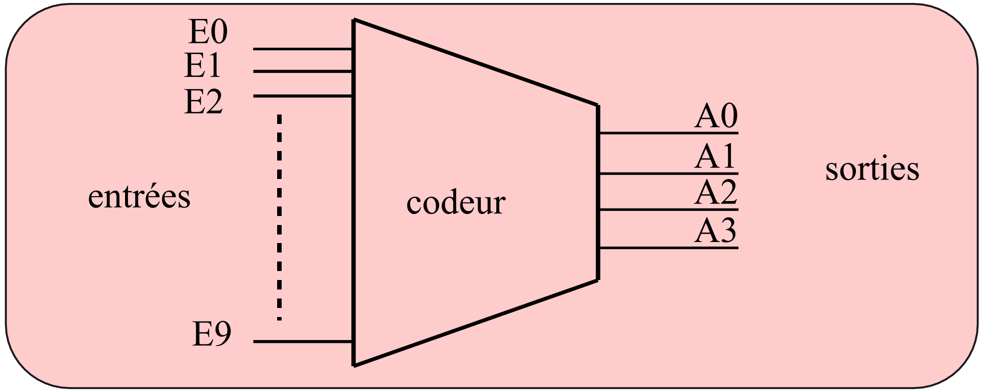


Opérateurs Transcodeurs (Codeur)

Codeur

Lorsqu'une entrée (sur les *N*) 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 E5=1 et Ei=0 pour toutes les autres entrées, alors les sorties affichent (A3,A2,A1,A0)=(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.

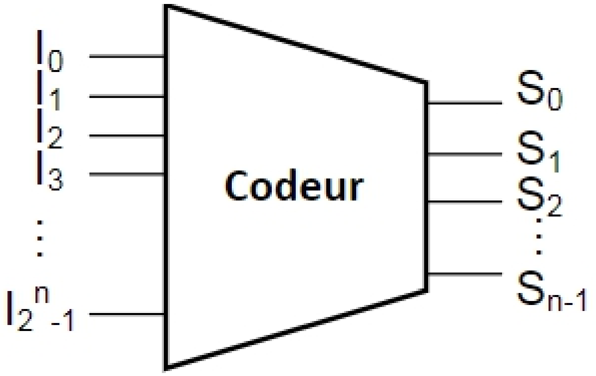


Table de vérité

Entrées	Sorties			
	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

Equation des sorties

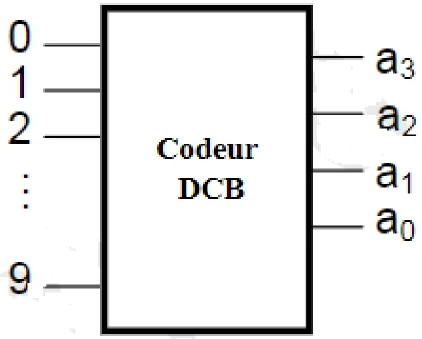
$a_0=1+3+5+7+9$

$a_1=2+3+6+7$

$a_2=4+5+6+7$

$a_3=8+9$

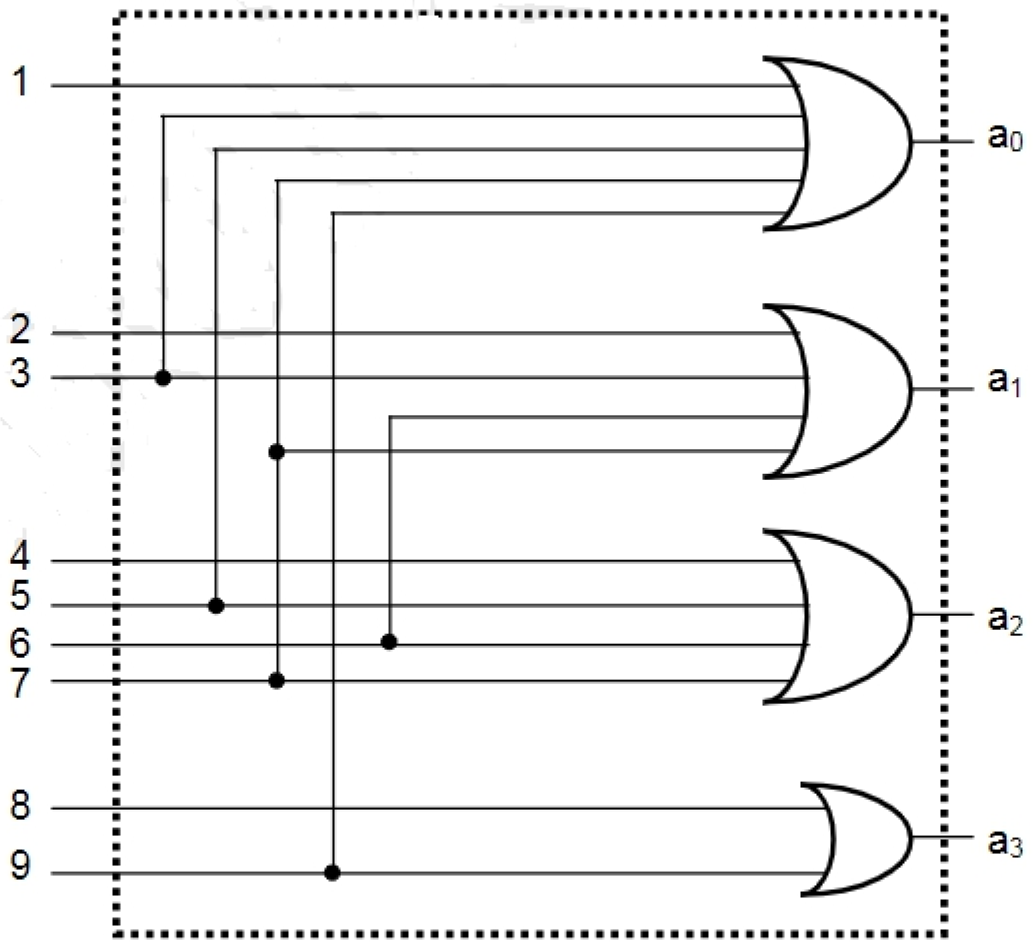
Logigramme



Circuit intégré :
74LS147

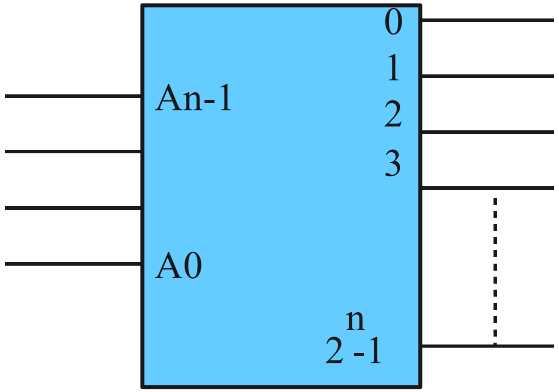
Opérateurs Transcodeurs

Codeurs

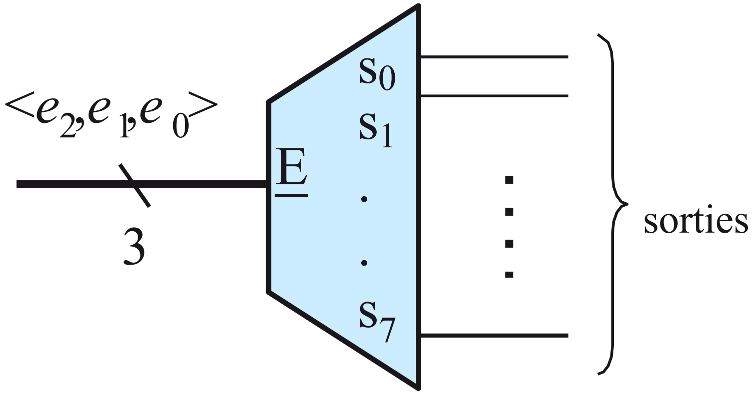


Opérateurs Transcodeurs (Decodeur)

- *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

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

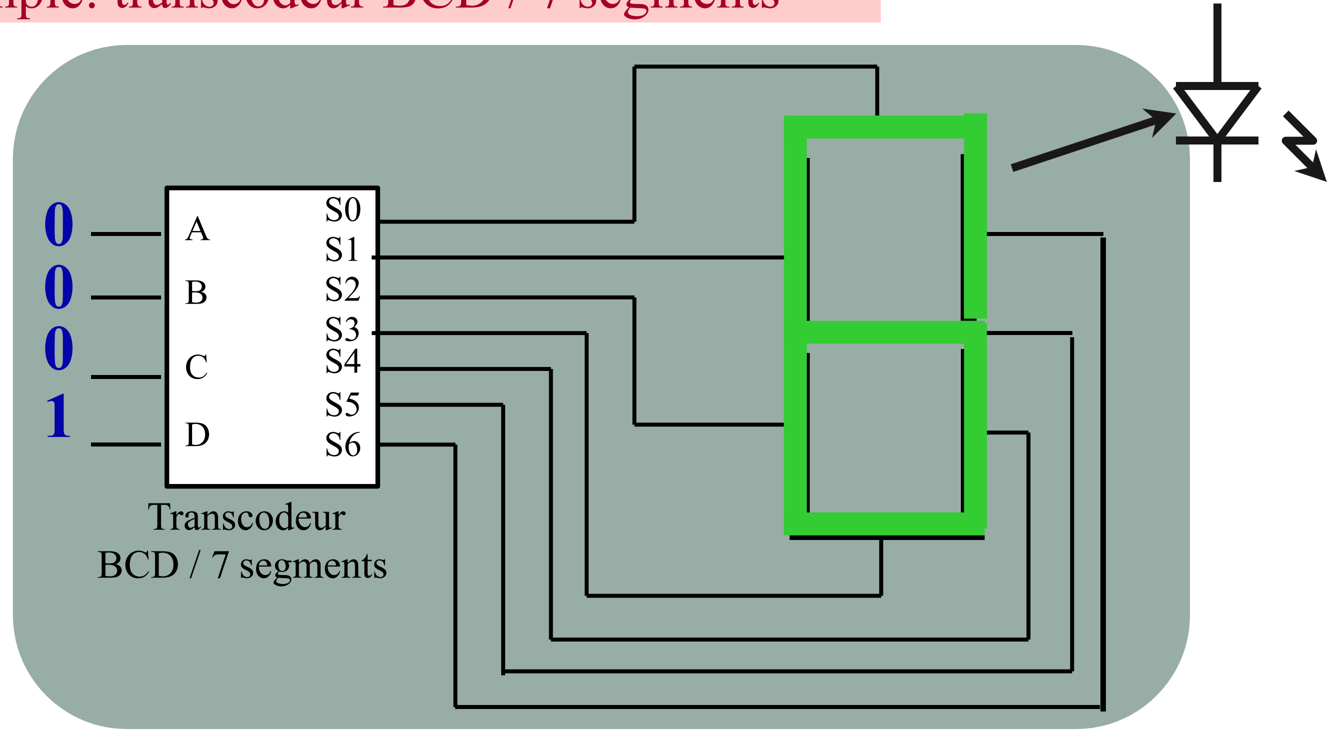
$$s_1 = \overline{e_2}\,\overline{e_1}\,e_0$$

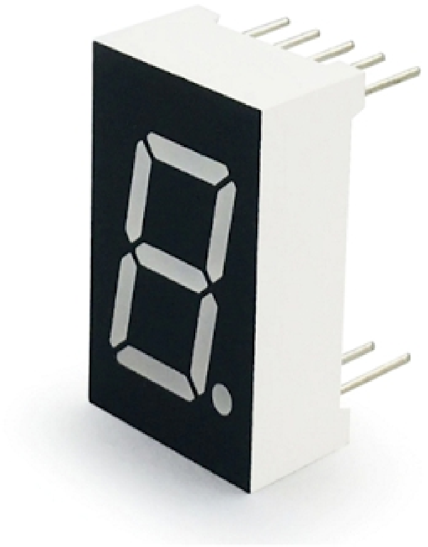
$$s_2 = \overline{e_2}\,e_1\,\overline{e_0}$$

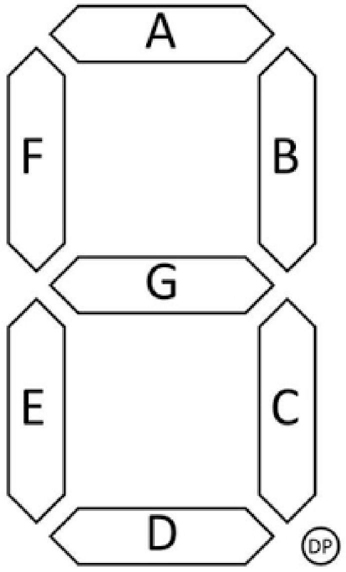
etc.

Opérateurs Transcodeurs

Exemple: transcodeur BCD / 7 segments

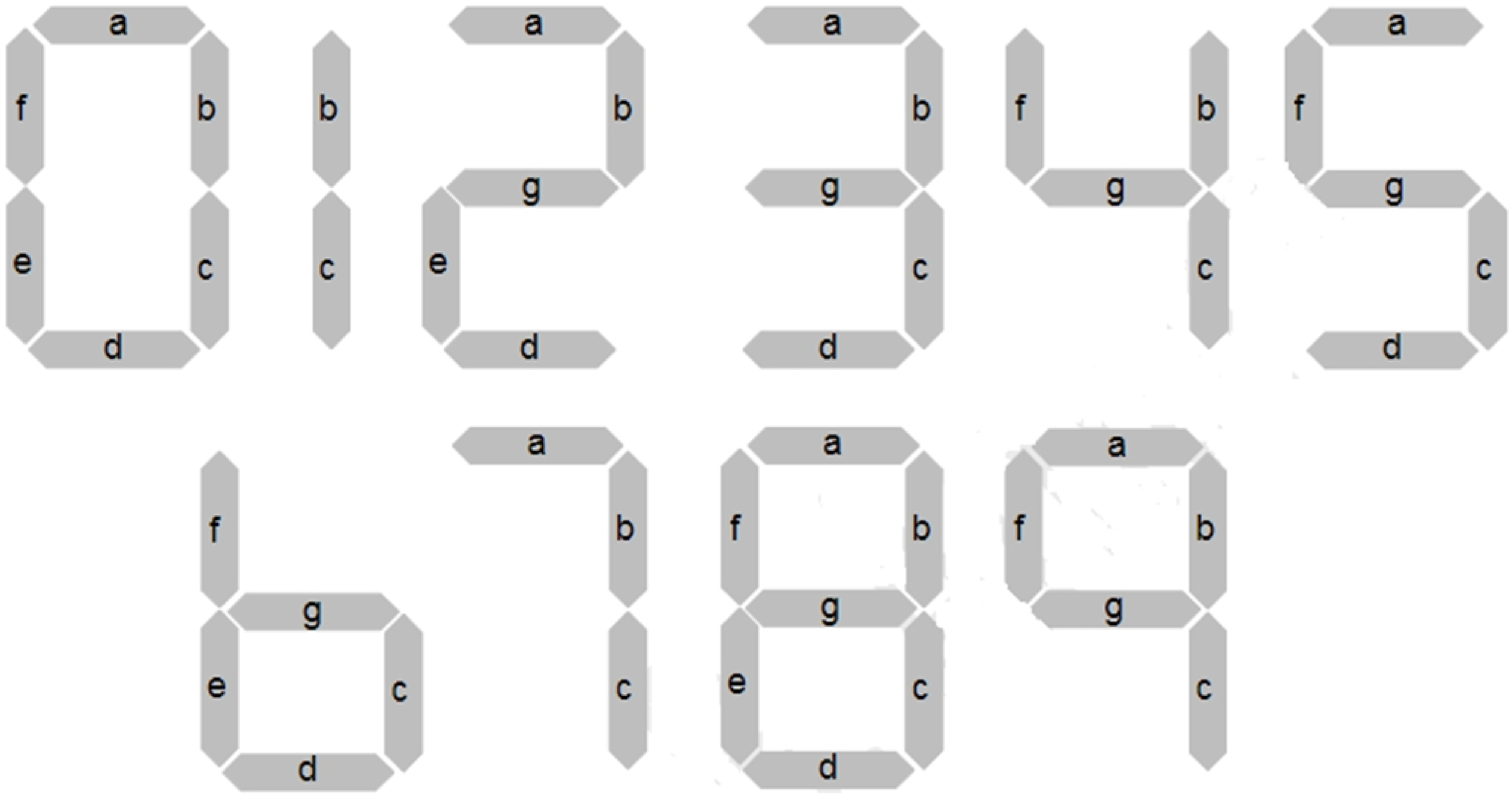






Il est souvent nécessaire de visualiser une information codée en binaire sur des afficheurs (7 segments)
=> convertisseur BCD (*Binary-Coded Decimal*) / 7 segments
=> convertisseur binaire pur / 7 segments

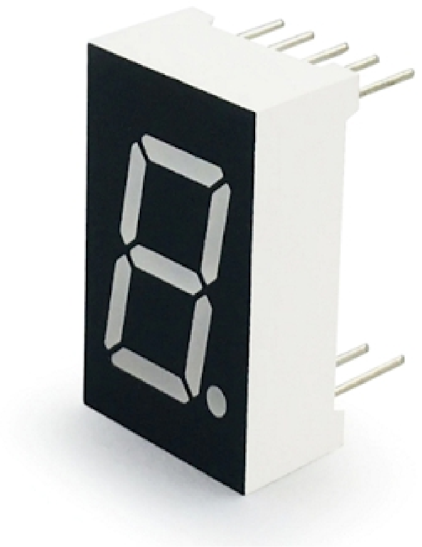
Opérateurs Transcodeurs

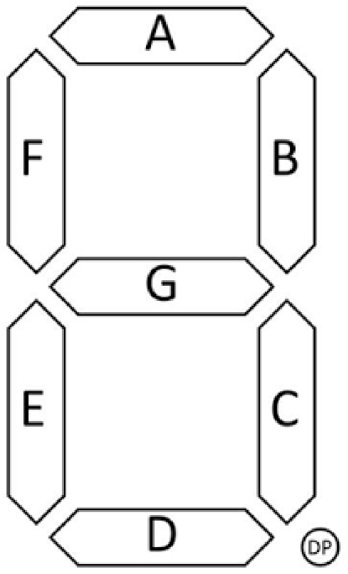


Opérateurs Transcodeurs

DCB 7 segments

Table de vérité											
Nombre BCD	Entrées				Sorties						
	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





NB : Il y a 6 combinaisons 10, 11, 12, 13, 14, 15 sans sortie definies

Opérateurs Transcodeurs

DCB 7 segments

a

AB \ CD	00	01	11	10
00	1	1	X	0
01	1	X	X	1
11	1	X	X	1
10	0	1	X	1

e

AB \ CD	00	01	11	10
00	1	1	X	0
01	1	X	X	1
11	0	X	X	0
10	0	0	X	0

d

AB \ CD	00	01	11	10
00	1	1	X	0
01	1	X	X	1
11	1	X	X	0
10	0	1	X	1

Equations logiques	
$a=B+D+AC+\bar{A}\bar{C}$	$e=\bar{A}B+\bar{A}\bar{C}$
$b=\bar{C}+AB+\bar{A}\bar{B}$	$f=D+\bar{A}\bar{B}+\bar{B}C+C\bar{A}$
$c=A+C+\bar{B}$	$g=D+\bar{A}C+C\bar{B}+B\bar{C}$
$d=D+\bar{A}B+B\bar{C}+\bar{C}\bar{A}+A\bar{B}C$	

b

AB \ CD	00	01	11	10
00	1	1	X	1
01	1	X	X	0
11	1	X	X	1
10	1	1	X	0

f

AB \ CD	00	01	11	10
00	1	1	X	1
01	0	X	X	1
11	0	X	X	0
10	0	1	X	1

c

AB \ CD	00	01	11	10
00	1	1	X	1
01	0	X	X	1
11	1	X	X	1
10	1	1	X	1

g

AB \ CD	00	01	11	10
00	0	1	X	1
01	1	X	X	1
11	1	X	X	0
10	0	1	X	1