Module: Architecture des ordinateurs

1ère MI S2

Circuits Logiques

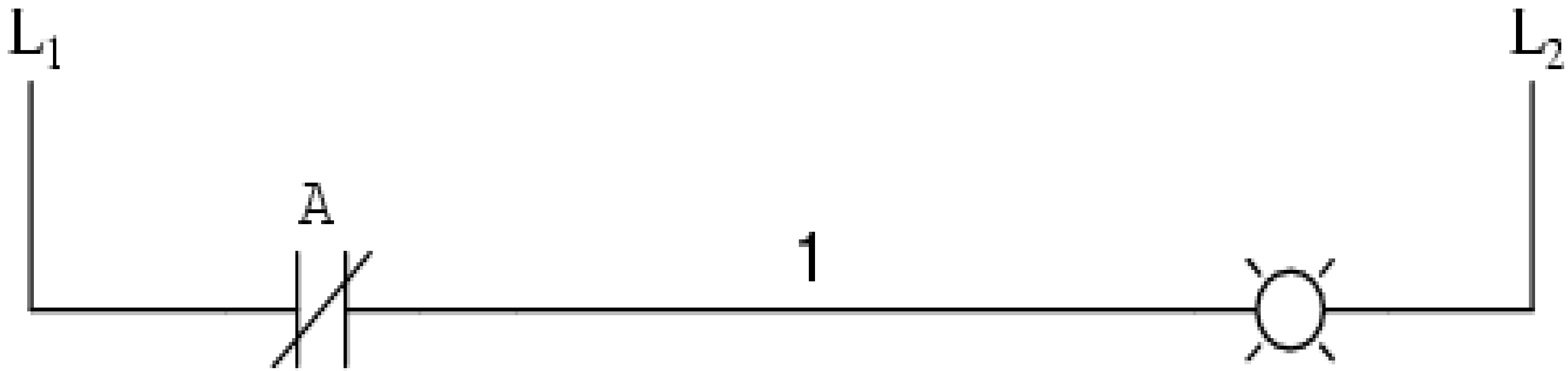
الدارات المنطقية

Taha Zerrouki

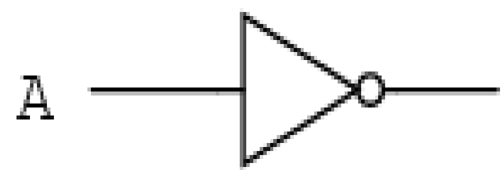
Taha.zerrouki@gmail.com

Circuits de Base

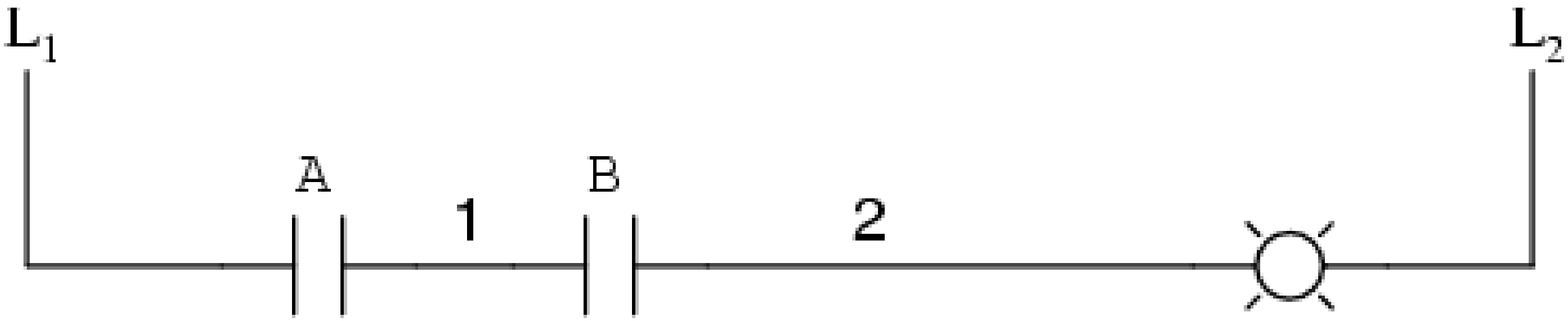
Inverseur (NON)



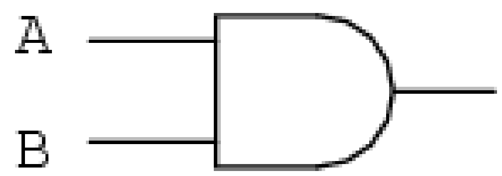
A	Output
0	1
1	0



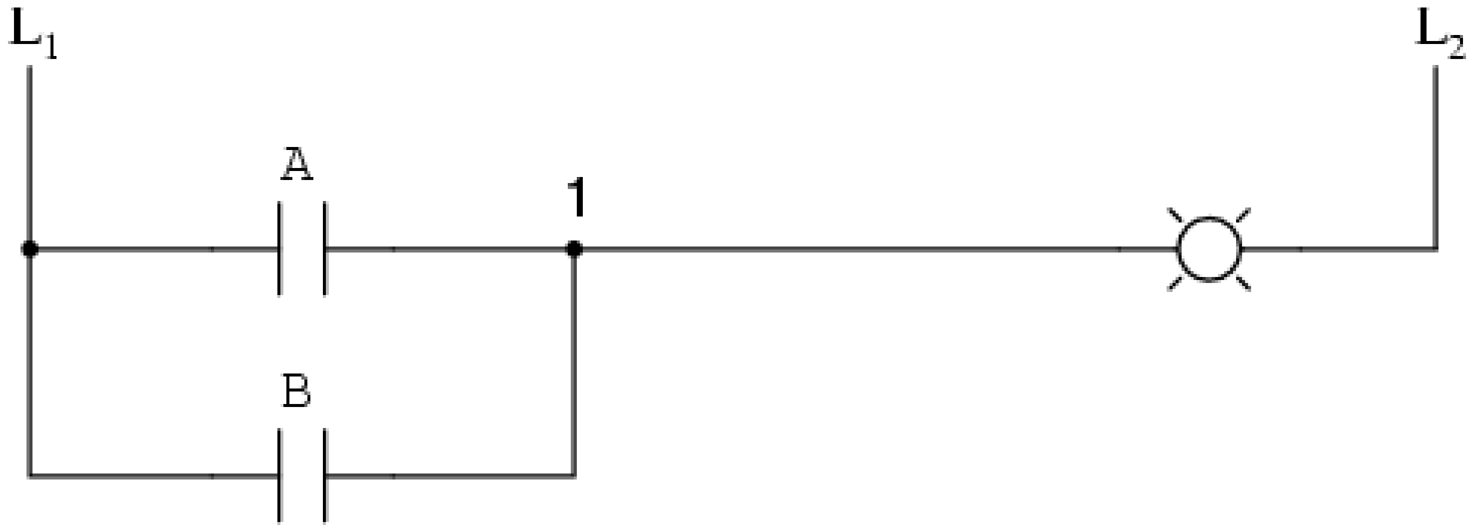
Conjonction ET (AND)



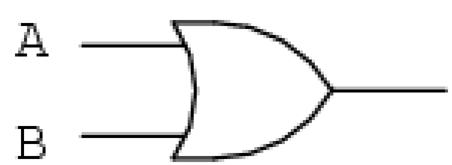
A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1



Disjonction (OU) (OR)



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1



Circuits combinés

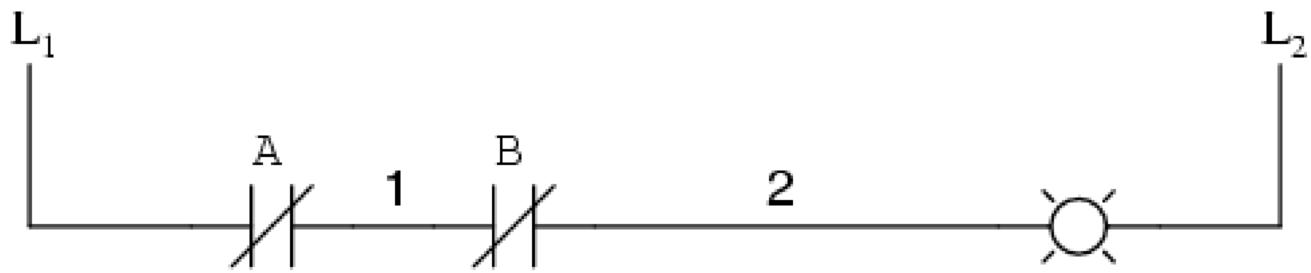
7.3 NOR (NON OU)

$$F(A,B) = \overline{A + B}$$

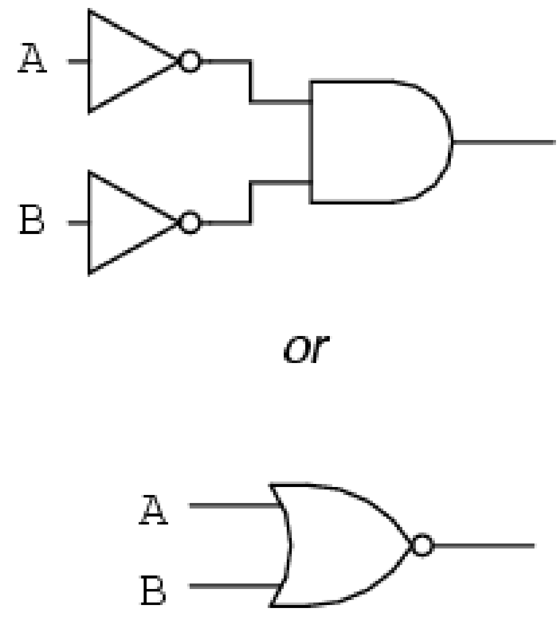
$$F(A,B) = A \downarrow B$$

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

Non-OU (NAND)



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0



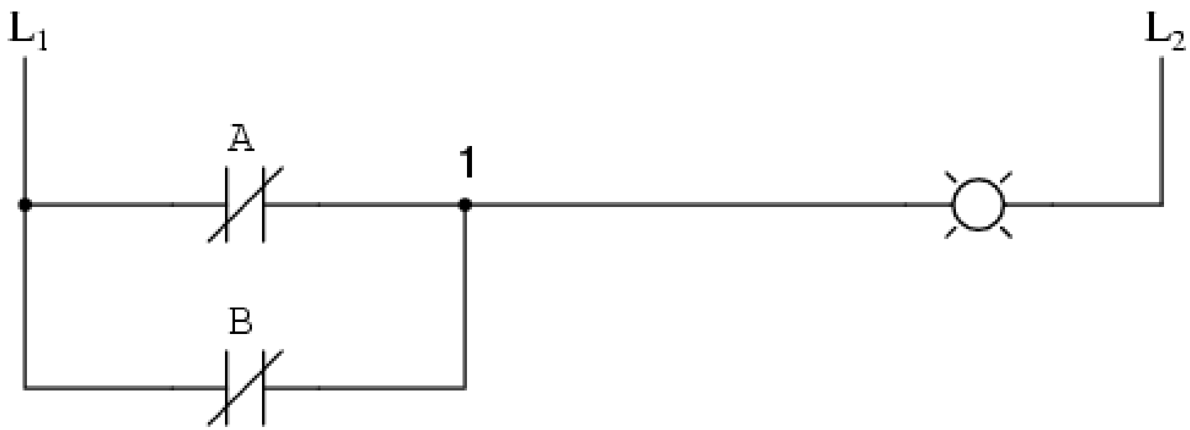
7.2 NAND (NON ET)

$$F(A,B) = \overline{A \, . \, B}$$

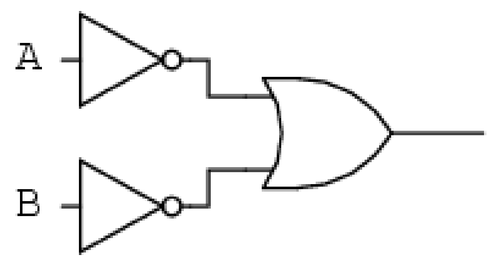
$$F(A,B) = A \uparrow B$$

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

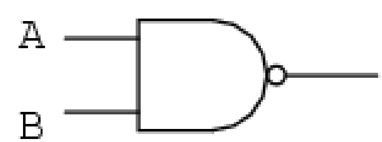
NON-ET (Nand)



A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0



or



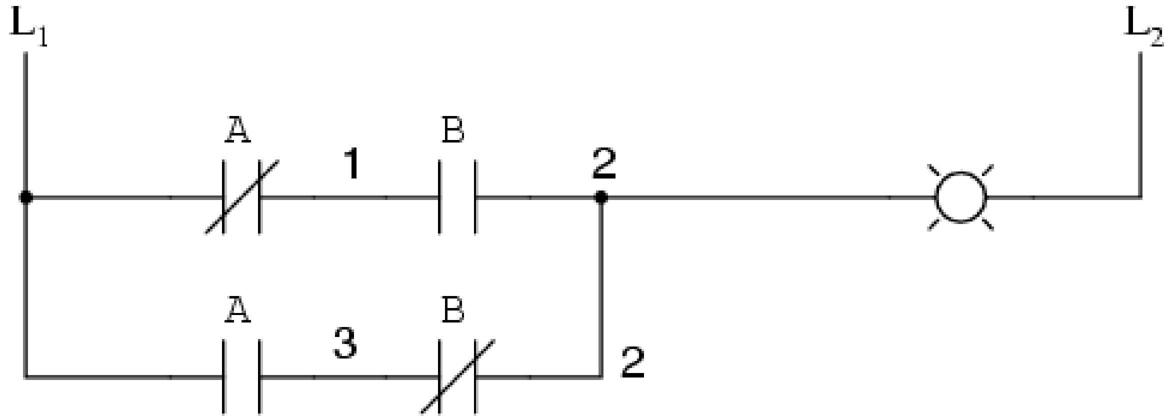
OU exclusif (XOR)

$$F(A,B) = A \oplus B$$

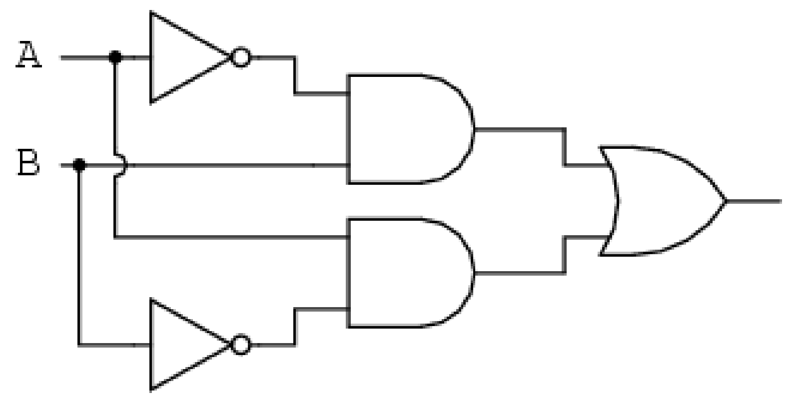
$$A \oplus B = \overline{A}.B + A.\overline{B}$$

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

OU exclusif (XOR)



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0



or

Exercice 1 : Donner l'équation de F ?

Les circuits combinatoires

Les circuits combinatoires

Objectifs

- Apprendre la structure de quelques circuits combinatoires souvent utilisés (demi additionneur , additionneur complet,……..).

- Apprendre comment utiliser des circuits combinatoires pour concevoir d'autres circuits plus complexes.

Les Circuits combinatoires

- Un circuit combinatoire est un circuit numérique dont les sorties dépendent uniquement des entrées.
- $S_i=F(E_i)$
- $S_i=F(E_1,E_2,\ldots.,E_n)$

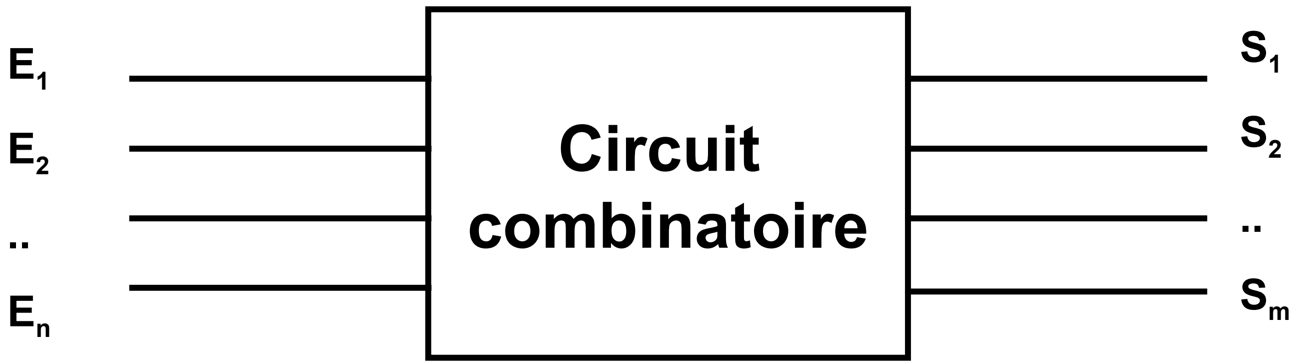


Schéma Bloc

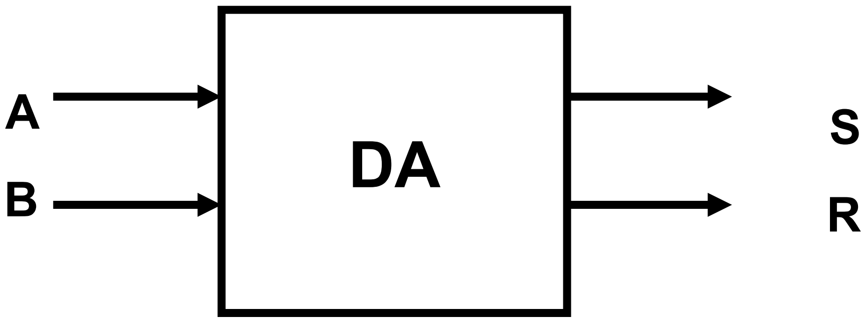
- C'est possible d'utiliser des circuits combinatoires pour réaliser d'autres circuits plus complexes.

Exemple de Circuits combinatoires

1. **Multiplexeur**
2. **Demultiplexeur**
3. **Encodeur**
4. **Décodeur**
5. **Transcodeur**
6. **Demi Additionneur**
7. **Additionneur complet**
8. **Comparateur**

2. 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 sotie 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

- En binaire l'addition sur un seul bit se fait de la manière suivante:

$$\begin{cases} 0+0 &= 00 \\ 0+1 &= 01 \\ 1+0 &= 01 \\ 1+1 &= 10 \end{cases}$$

:La table de vérité associée •

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

:De la table de vérité on trouve

$$R = A.B$$

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

$R = A.B$

$S = A \oplus B$

3. L'additionneur complet

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

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

3.1 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} : le retenue entrante sur un bit.
- Il possède deux sorties :
  - S_i : la somme
  - R_i la retenue sortante

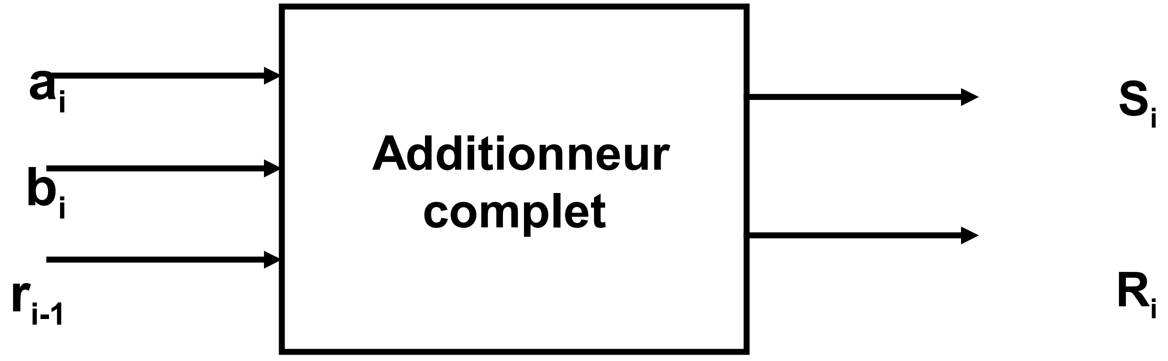


Table de vérité d'un additionneur complet sur 1 bit

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

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

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

3.3 Schéma d'un additionneur complet

$$R_i = A_i.B_i + R_{i-1}.(B_i \oplus A_i)$$

$$S_i = A_i \oplus B_i \oplus R_{i-1}$$

3.4 Additionneur sur 4 bits

- Un additionneur sur 4 bits est un circuit qui permet de faire l'addition de deux nombres A et B de 4 bits chacun
  - $A(a_3a_2a_1a_0)$
  - $B(b_3b_2b_1b_0)$

  En plus il tient en compte de la retenu entrante

- En sortie on va avoir le résultat sur 4 bits ainsi que la retenu (5 bits en sortie)

- Donc au total le circuit possède 9 entrées et 5 sorties.

- Avec 9 entrées on a $2^9=512$ combinaisons !!!!!! Comment faire pour représenter la table de vérité ?????

- Il faut trouver une solution plus facile et plus efficace pour concevoir ce circuit ?

•Lorsque on fait l'addition en binaire , on additionne bit par bit en commençant à partir du poids fiable et à chaque fois on propage la retenue sortante au bit du rang supérieur.

L'addition sur un bit peut se faire par un additionneur complet sur 1 bits.

$r_0 = 0$	r_1	r_2	r_3	
a_1	a_2	a_3	a_4	
b_1	b_2	b_3	b_4	$+$
$r_1\ s_1$	$r_2\ s_2$	$r_3\ s_3$	$r_4\ s_4$	

r4	s_4	s_3	s_2	s_1

Résultat final

3.4.1 Additionneur 4 bits (schéma)

Exercice

- Soit une information binaire sur 5 bits ($i_4i_3i_2i_1i_0$). Donner le circuit qui permet de calculer le nombre de 1 dans l'information en entrée en utilisant uniquement des additionneurs complets sur 1 bit ?
- Exemple :

Si on a en entrée l'information ($i_4i_3i_2i_1i_0$) =(10110) alors en sortie on obtient la valeur 3 en binaire (011) puisque il existe 3 bits qui sont à 1 dans l'information en entrée .

Multiplexage

Question?

- Quel est l’unité de mesure de la mémoire?

Question?

- Quel est l'unité de mesure de débit?

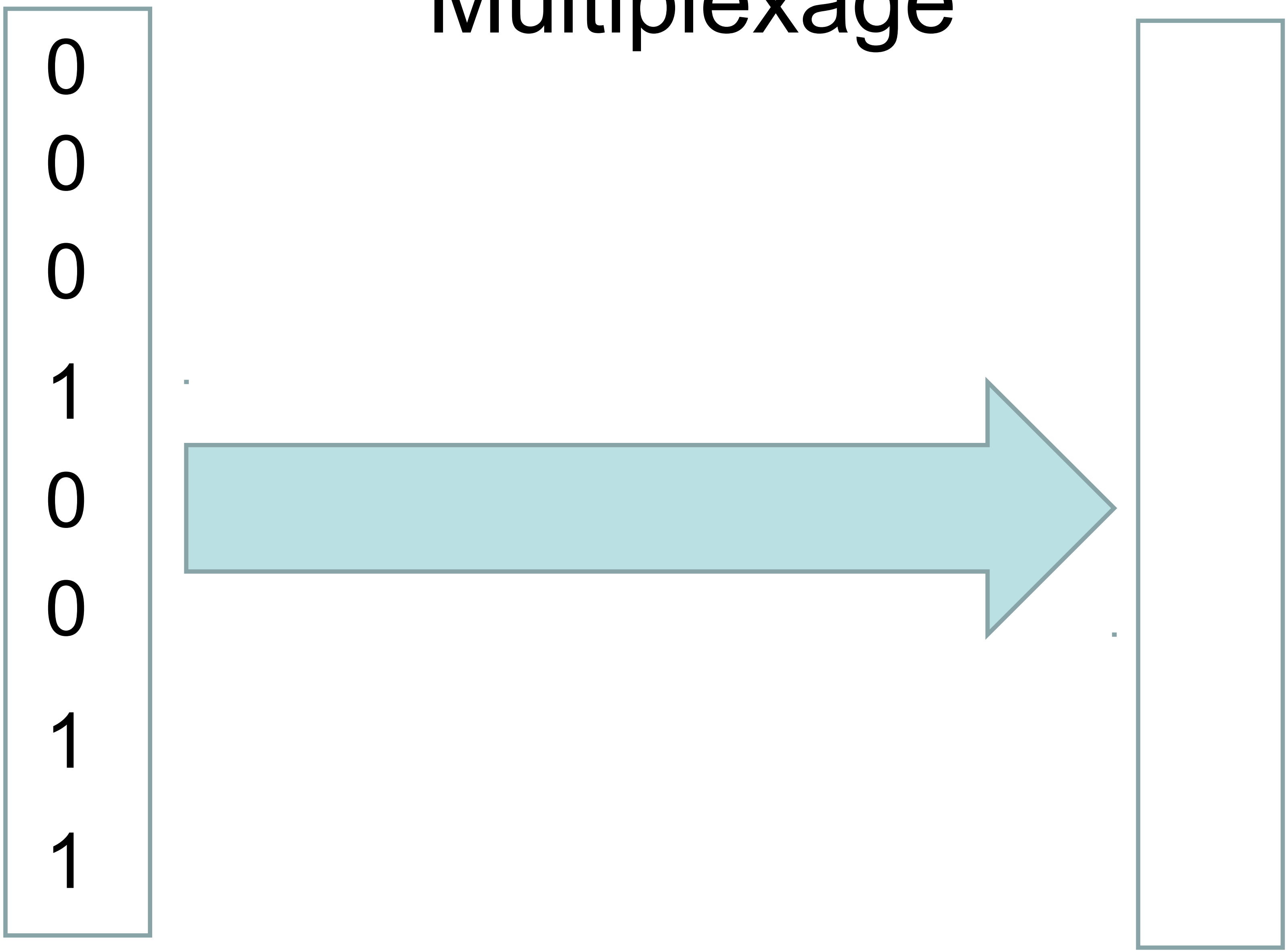
Question?

- Comment transmettre un octet par bits?

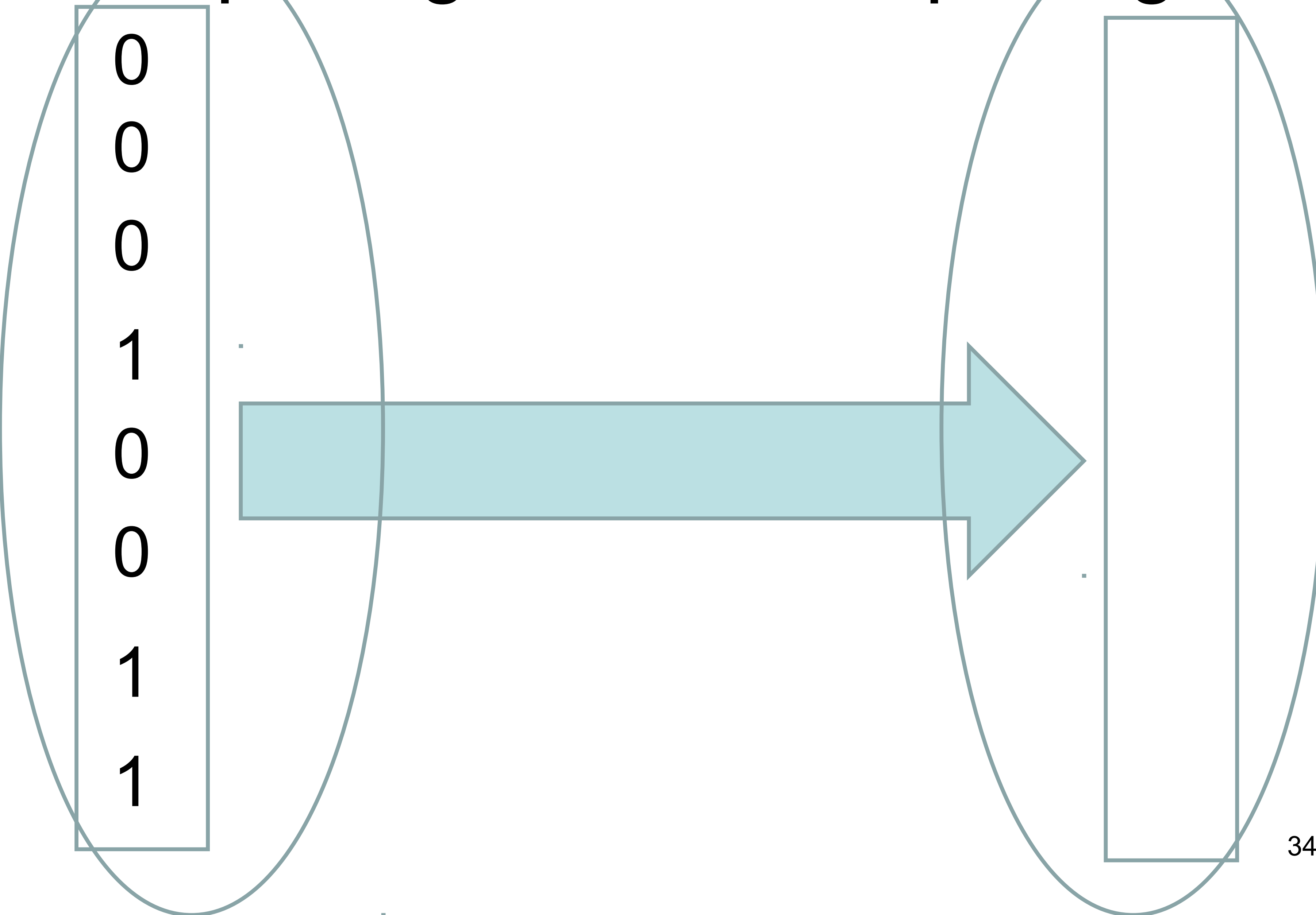
0
0
0
1
0
0
1
1

0
0
0
1
0
0
1
1

Multiplexage

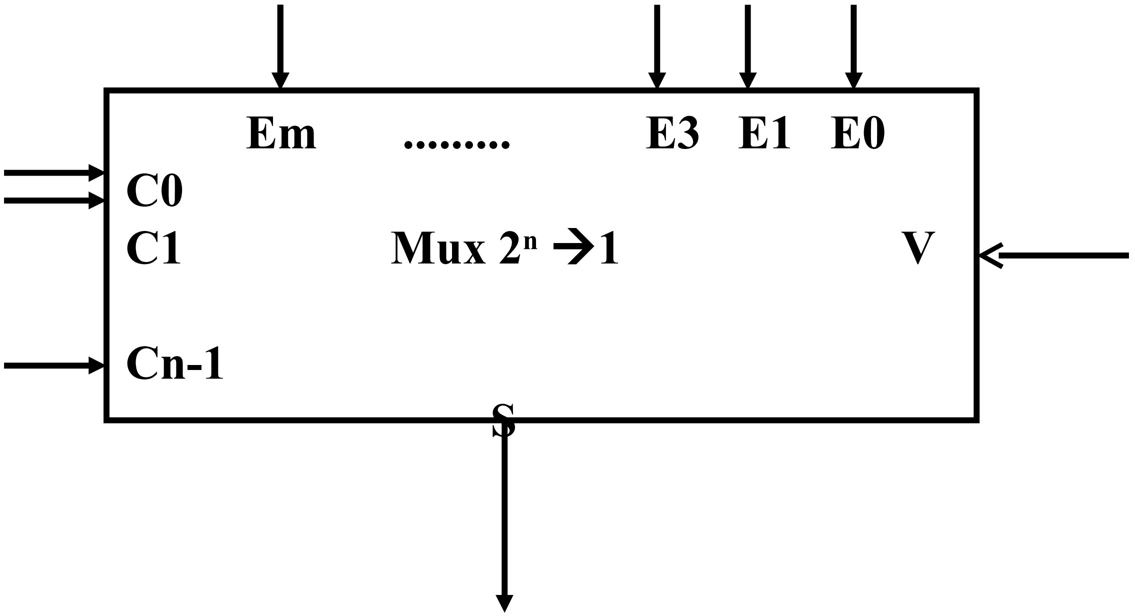


Multiplexage Démultiplexage



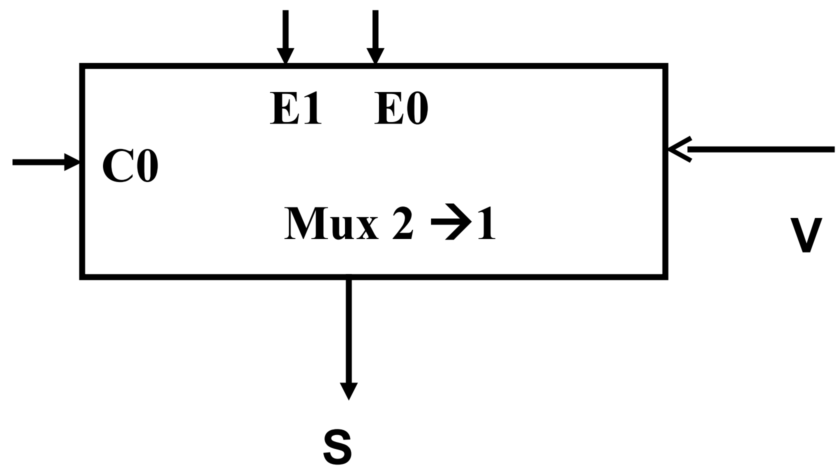
Le Multiplexeur

- Un multiplexeur est un circuit combinatoire qui permet de sélectionner une information (1 bit) parmi 2^n valeurs en entrée.
- Il possède :
  - 2^n entrées d'information
  - Une seule sortie
  - N entrées de sélection (commandes)



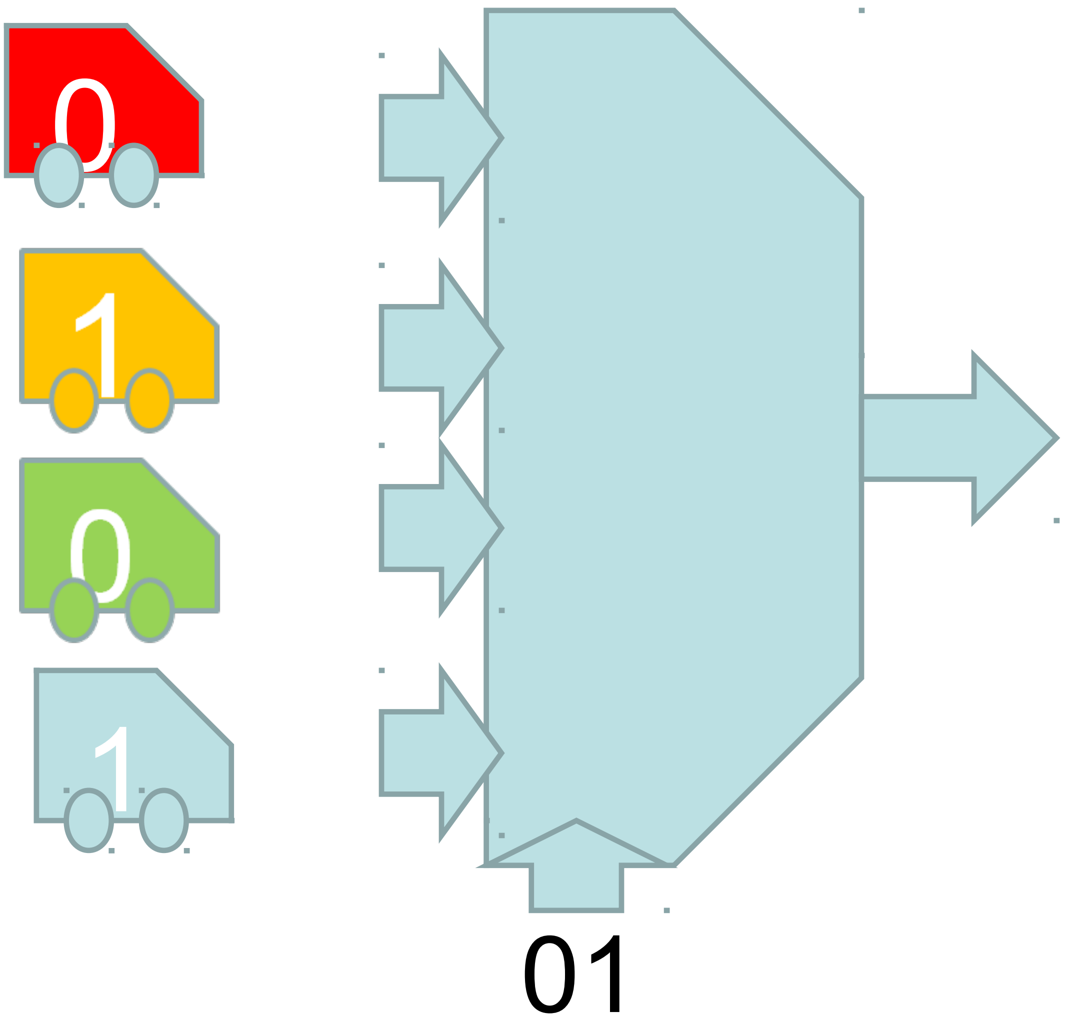
Multiplexeur 2 →1

S		C_0	V
0		X	0
E0		0	1
E1		1	1

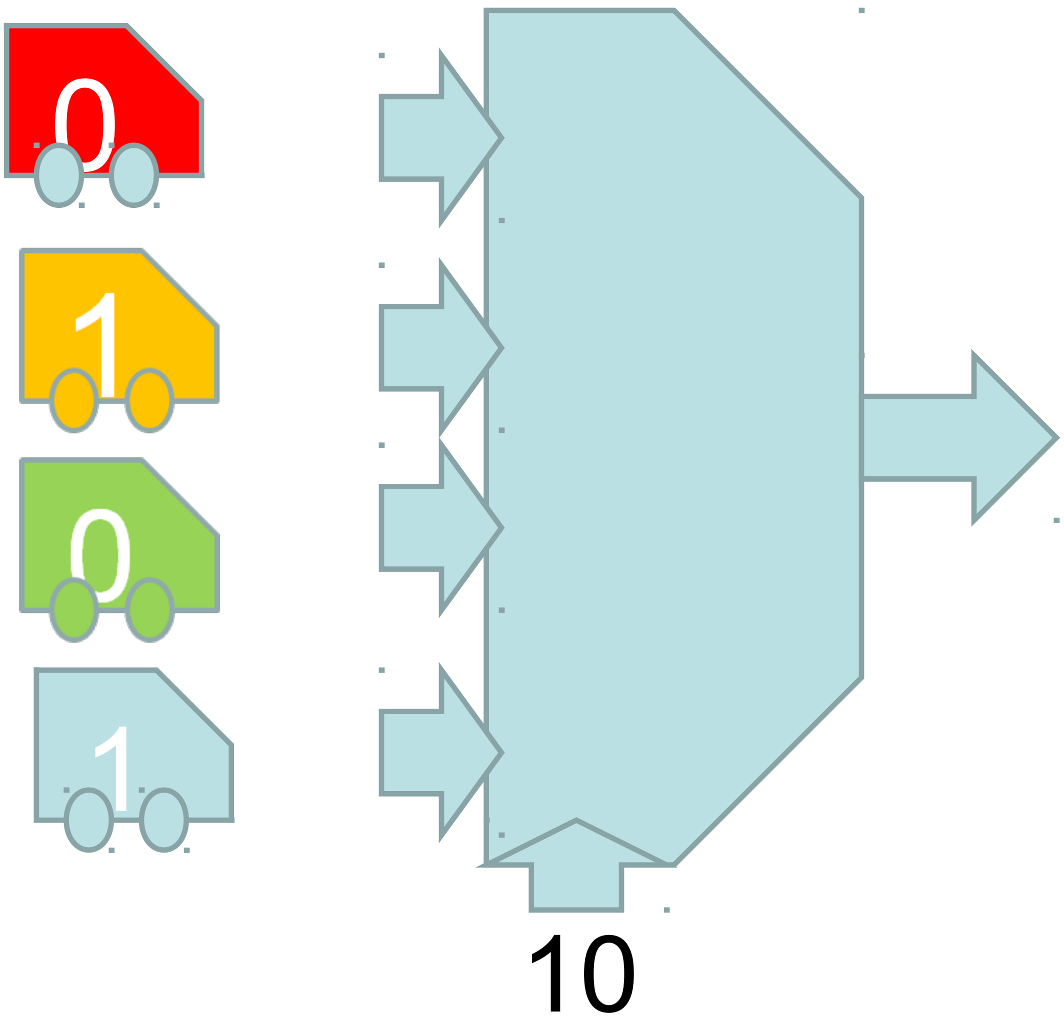


$$S = V.(\overline{C_0}.E0 + C_0.E1)$$

MultiPlexeur 4 → 1

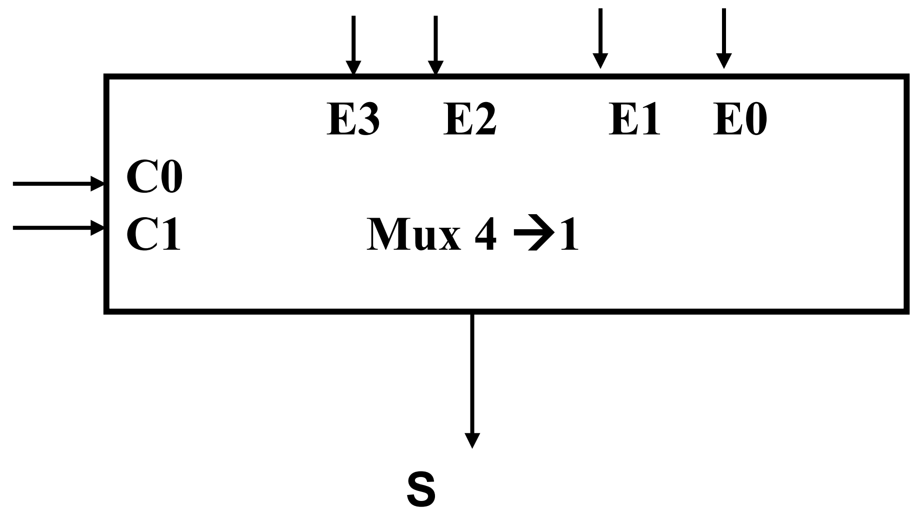


MultiPlexeur 4 → 1



Multiplexeur 4 →1

S		C0	C1
E0		0	0
E1		1	0
E2		0	1
E3		1	1



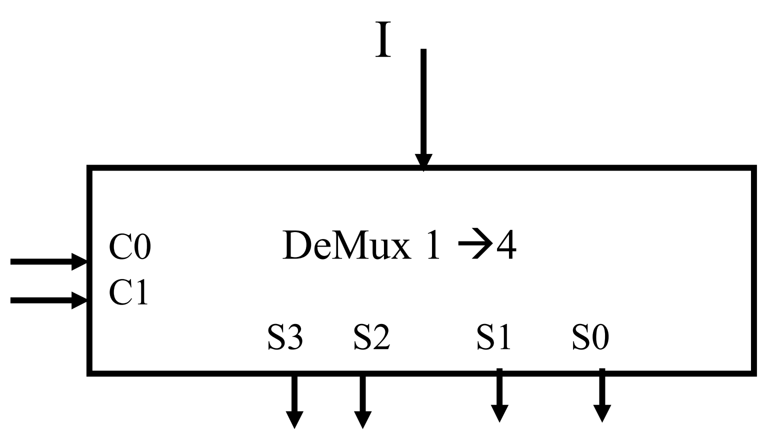
$$S = \overline{C1}.\overline{C0}.(E0) + \overline{C1}.C0.(E1) + C1.\overline{C0}.(E2) + C1.C0.(E3)$$

Exercice

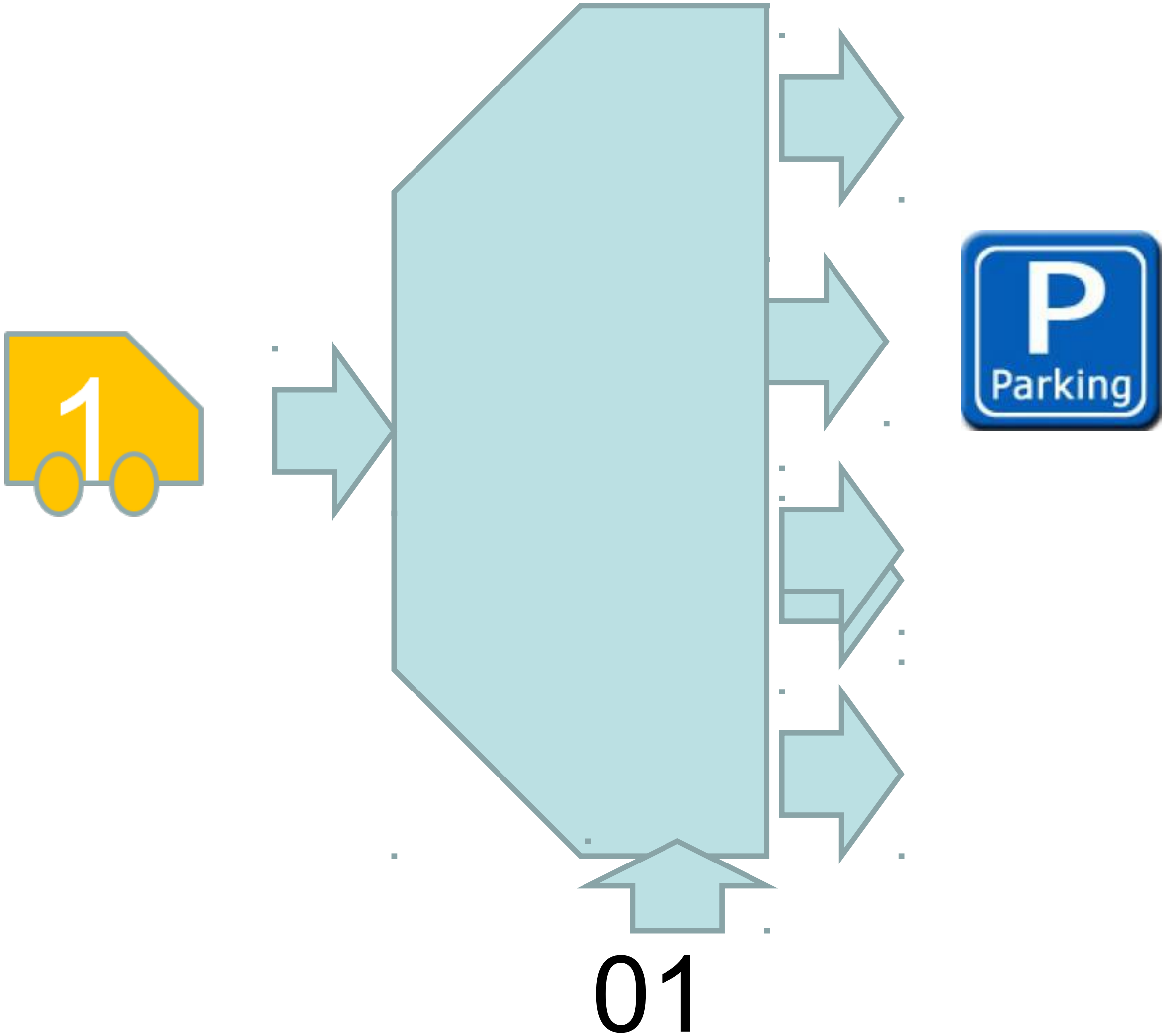
- Donner la table de vérité d'un multiplexeur 8➔1
- Donner le schéma bloc

Demultiplexeurs

- Il joue le rôle inverse d'un multiplexeurs, il permet de faire passer une information dans l'une des sorties selon les valeurs des entrées de commandes.
- Il possède :
  - une seule entrée
  - 2^n sorties
  - N entrées de sélection (commandes)



DéMultiPlexeur 1 ➔4



6.1 Demultiplexeur 1→4

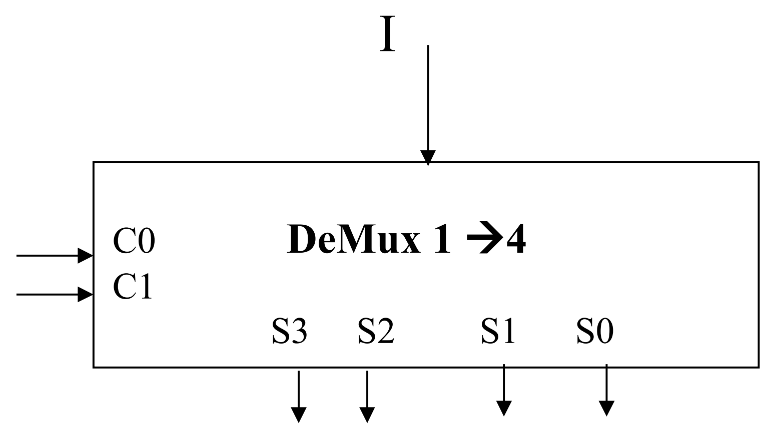
S0	S1	S2	S3		C0	C1
i	0	0	0		0	0
0	i	0	0		1	0
0	0	i	0		0	1
0	0	0	i		1	1

$$S0 = \overline{C1}.\overline{C0}.(I)$$

$$S1 = \overline{C1}.C0.(I)$$

$$S2 = C1.\overline{C0}.(I)$$

$$S3 = C1.C0.(I)$$



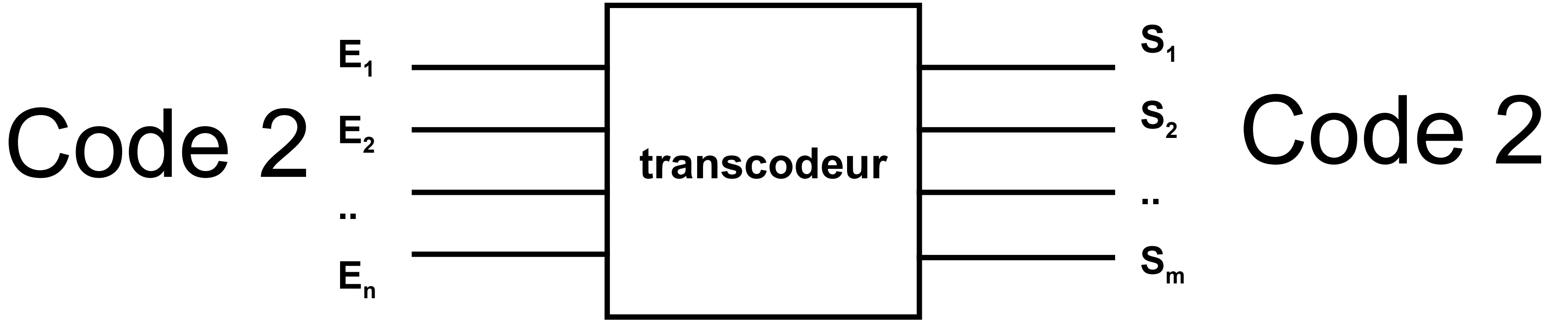
Exercice

- Donner la table de vérité d’un d démultiplexeur 1→8
- Donner le schéma bloc

Transcodage

Transcodage

- Les circuits combinatoires de transcodage
- (appelés aussi convertisseurs de code).

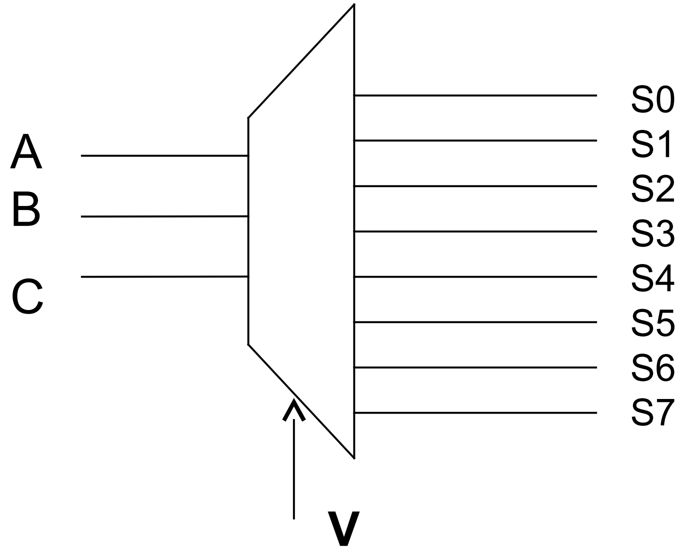


Transcodage

- **CODEUR**
  - 2^n entrées
  - n sorties
- **DECODEUR**
  - n entrées
  - 2^n sorties dont une seule est validée à la fois
- **TRANSCODEUR**
  - p entrées
  - k sorties.

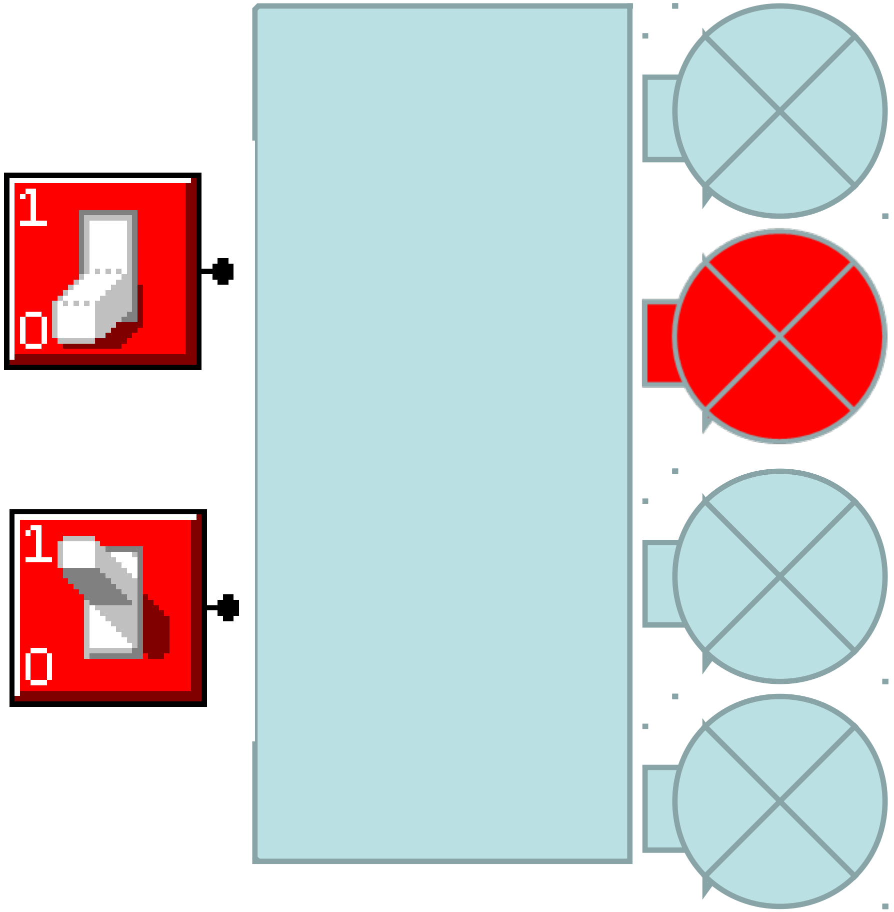
Le décodeur binaire

- C'est un circuit combinatoire qui est constitué de :
  - N : entrées de données
  - 2^n sorties
  - Pour chaque combinaison en entrée une seule sortie est active à la fois

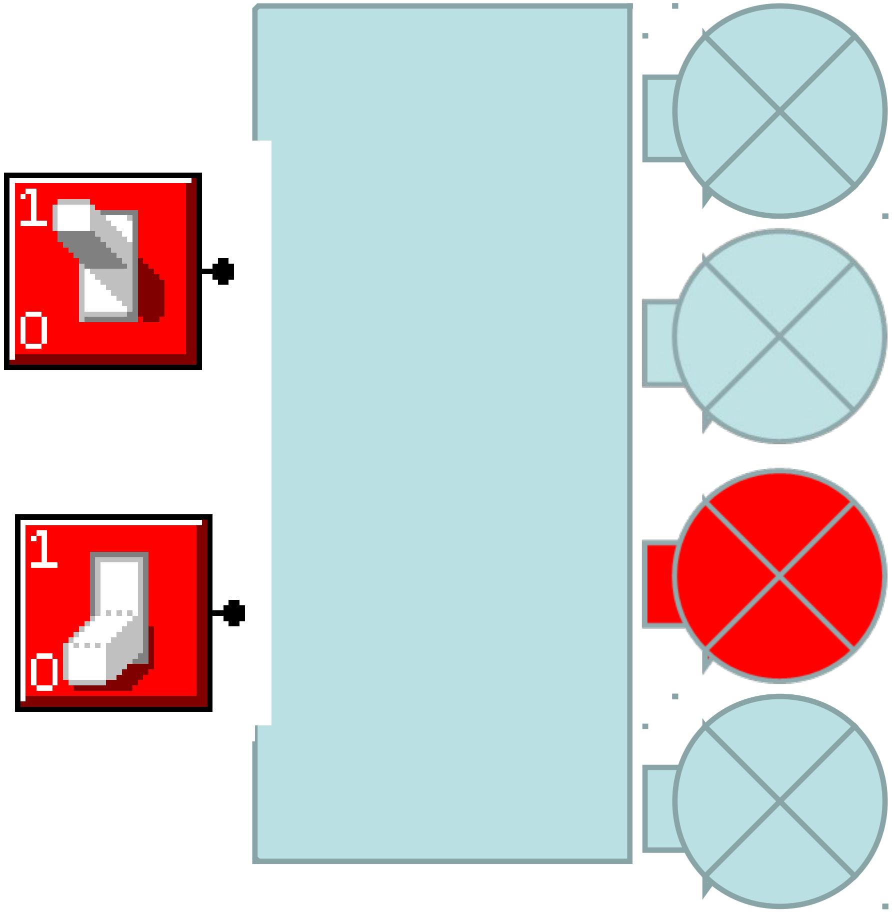


Un décodeur 3→8

Décodeur 2 ➔4

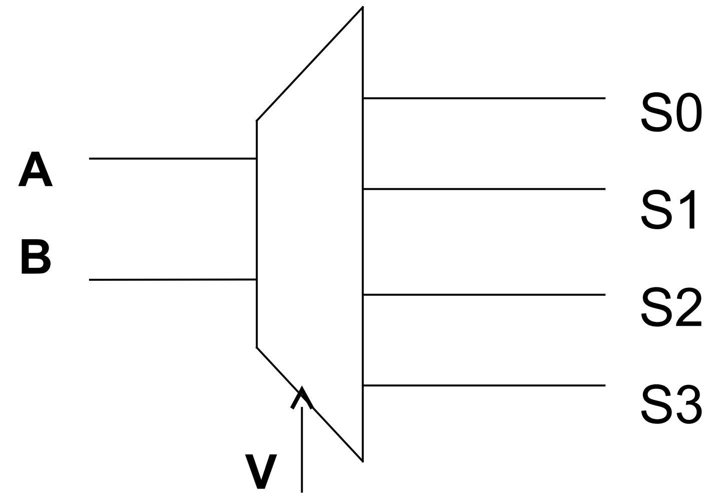


Décodeur 2 → 4



Décodeur 2→4

S3	S2	S1	S0		B	A	V
0	0	0	0		X	X	0
0	0	0	1		0	0	1
0	0	1	0		1	0	1
0	1	0	0		0	1	1
1	0	0	0		1	1	1



$$S_0 = (\overline{A}.\overline{B}).V$$

$$S_1 = (\overline{A}.B).V$$

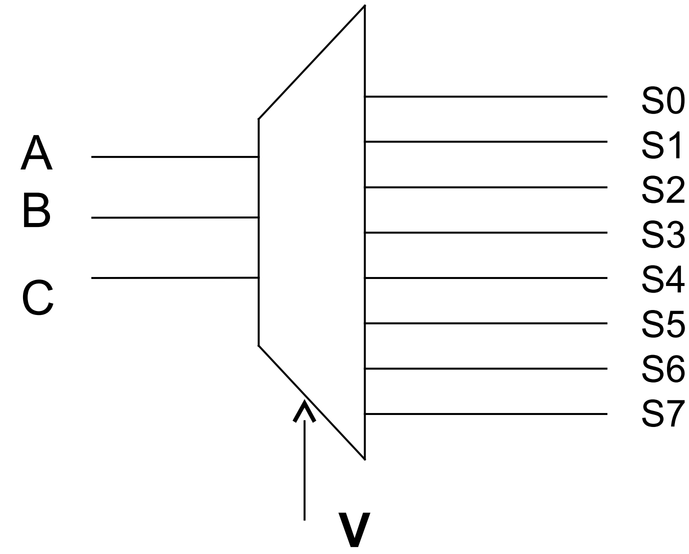
$$S_2 = (A.\overline{B}).V$$

$$S_3 = (A.B).V$$

Exercice

- Donner la table de vérité d'un décodeur 4→16
- Donner le schéma bloc

Décodeur 3→8



S7	S6	S5	S4	S3	S2	S1	S0		C	B	A
0	0	0	0	0	0	0	1		0	0	0
0	0	0	0	0	0	1	0		1	0	0
0	0	0	0	0	1	0	0		0	1	0
0	0	0	0	1	0	0	0		1	1	0
0	0	0	1	0	0	0	0		0	0	1
0	0	1	0	0	0	0	0		1	0	1
0	1	0	0	0	0	0	0		0	1	1
1	0	0	0	0	0	0	0		1	1	1

$$S_0 = \overline{A}.\overline{B}.\overline{C}$$

$$S_1 = \overline{A}.\overline{B}.C$$

$$S_2 = \overline{A}.B.\overline{C}$$

$$S_3 = \overline{A}.B.C$$

$$S_4 = A.\overline{B}.\overline{C}$$

$$S_5 = A.\overline{B}.C$$

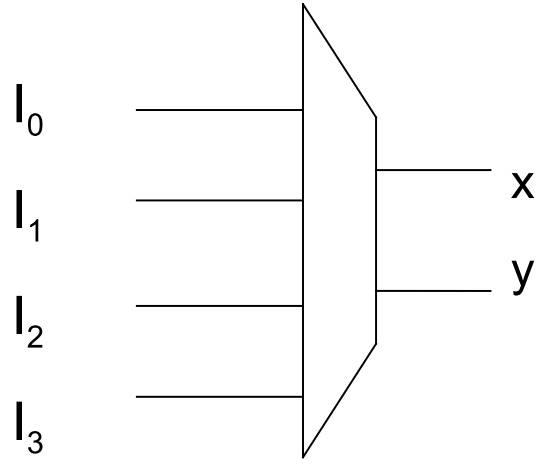
$$S_6 = A.B.\overline{C}$$

$$S_7 = A.B.C$$

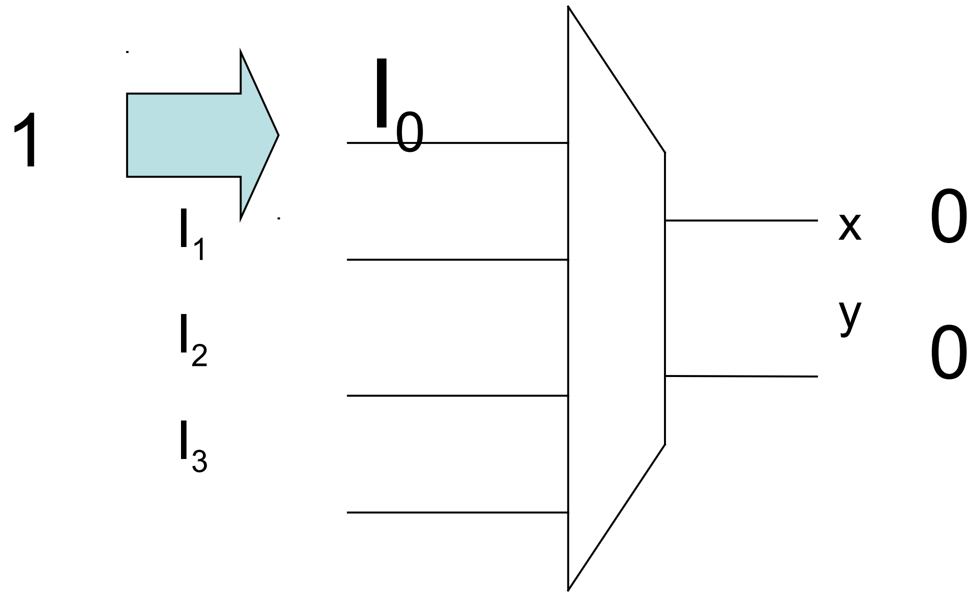
8. L'encodeur binaire

- Il joue le rôle inverse d'un décodeur
  - Il possède 2^n entrées
  - N sortie
  - Pour chaque combinaison en entrée on va avoir sont numéro (en binaire) à la sortie.

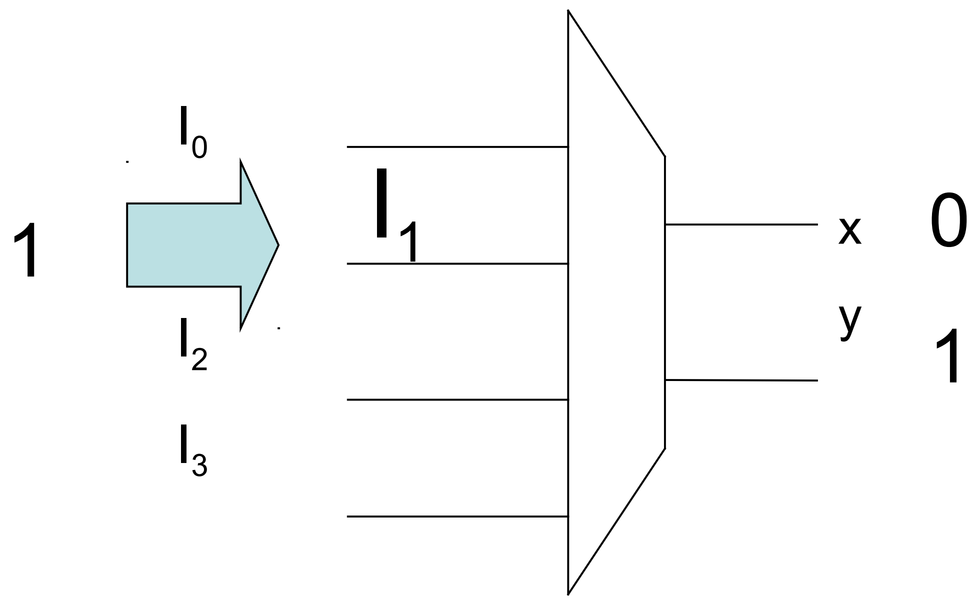
Encodeur 4→2



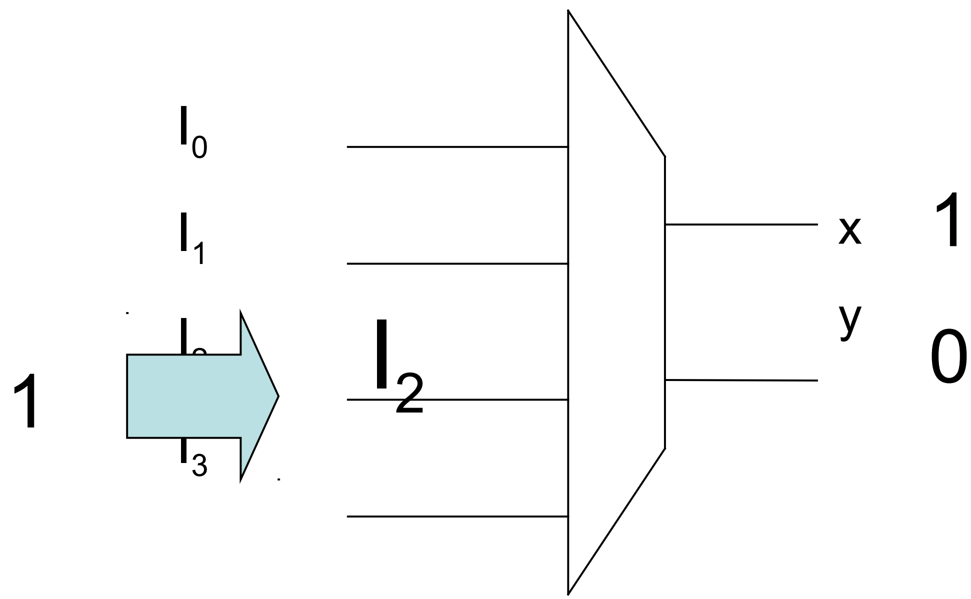
L'encodeur binaire (4→2)



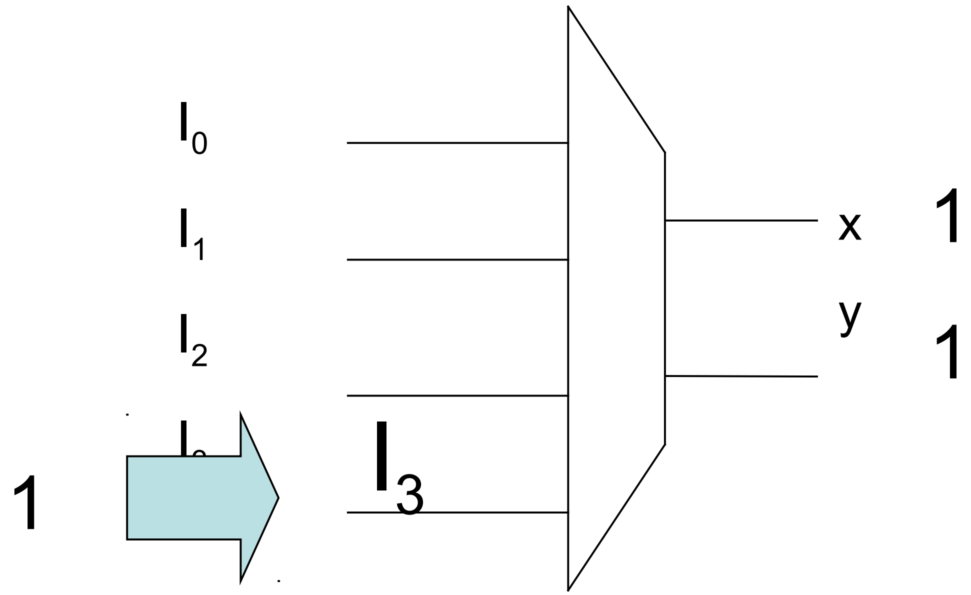
L’encodeur binaire (4→2)



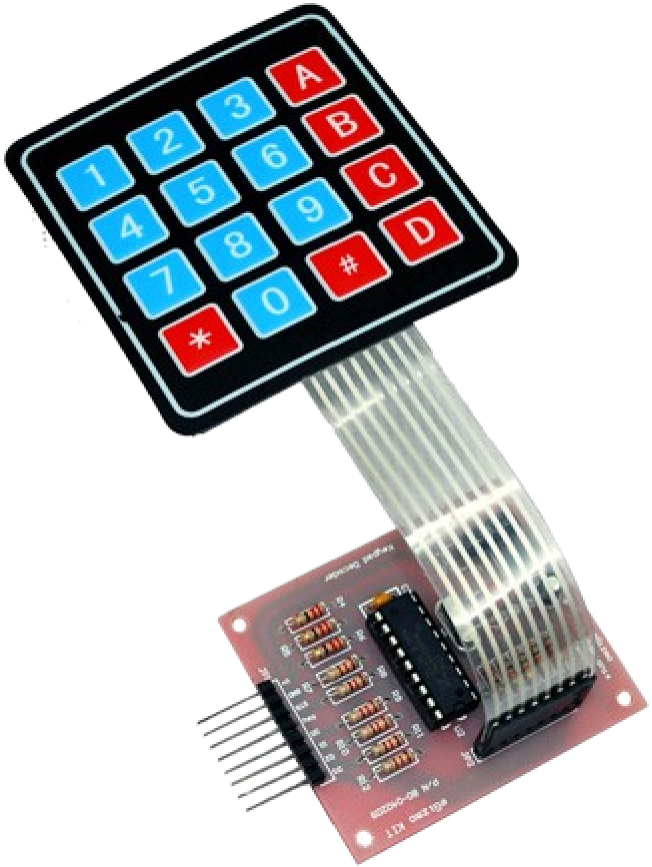
L'encodeur binaire (4→2)



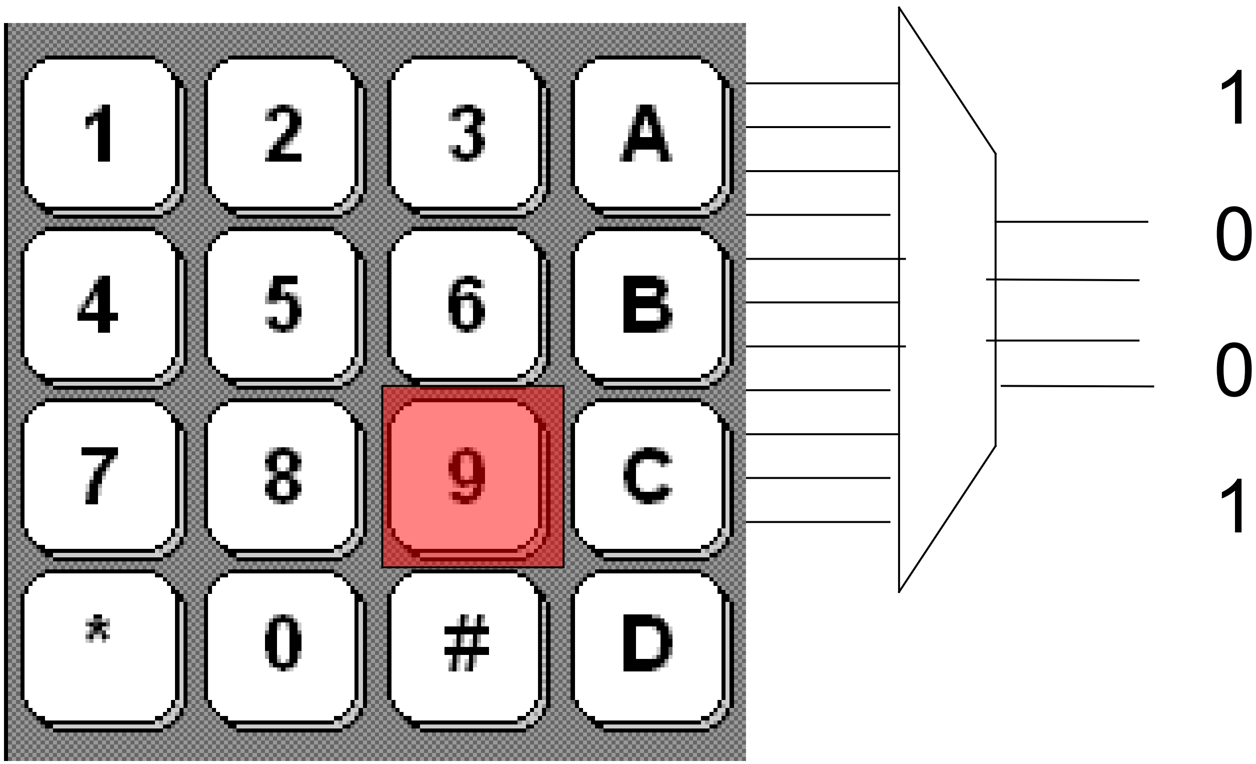
L'encodeur binaire (4→2)



Exemple d’application

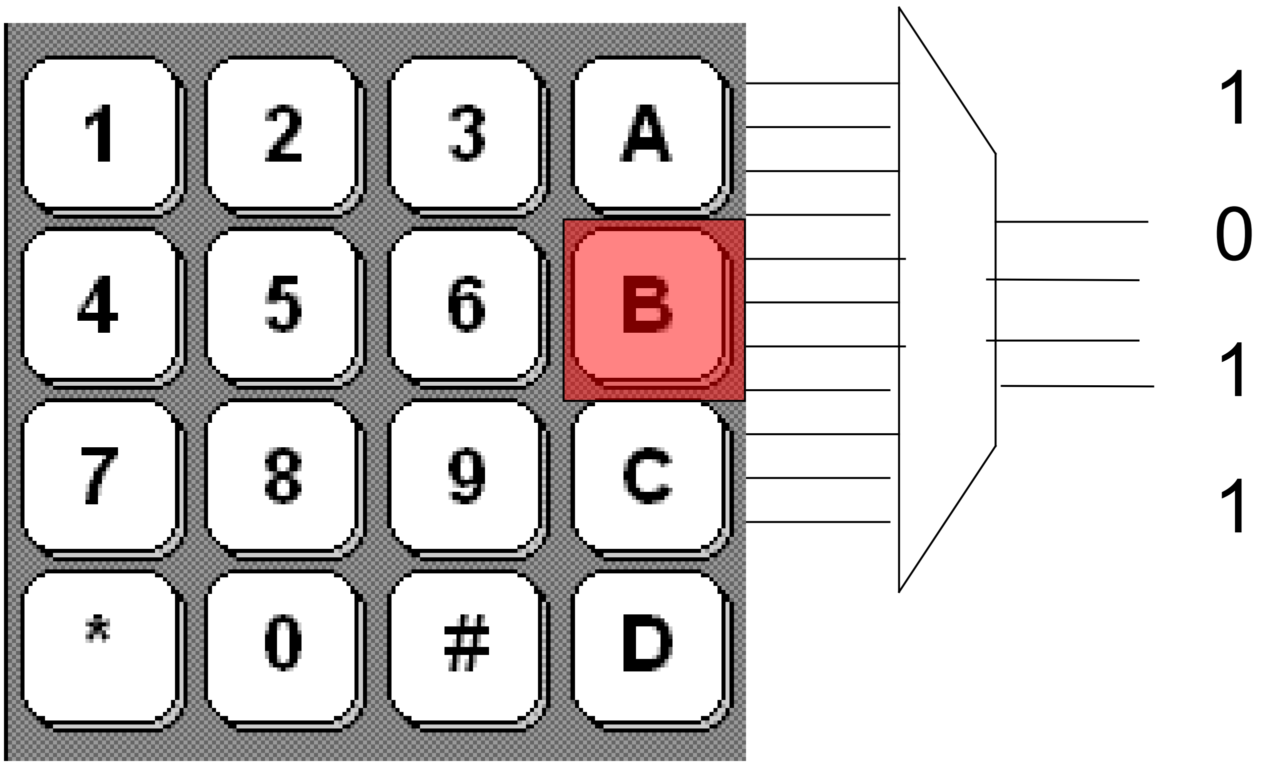


Exemple d'application



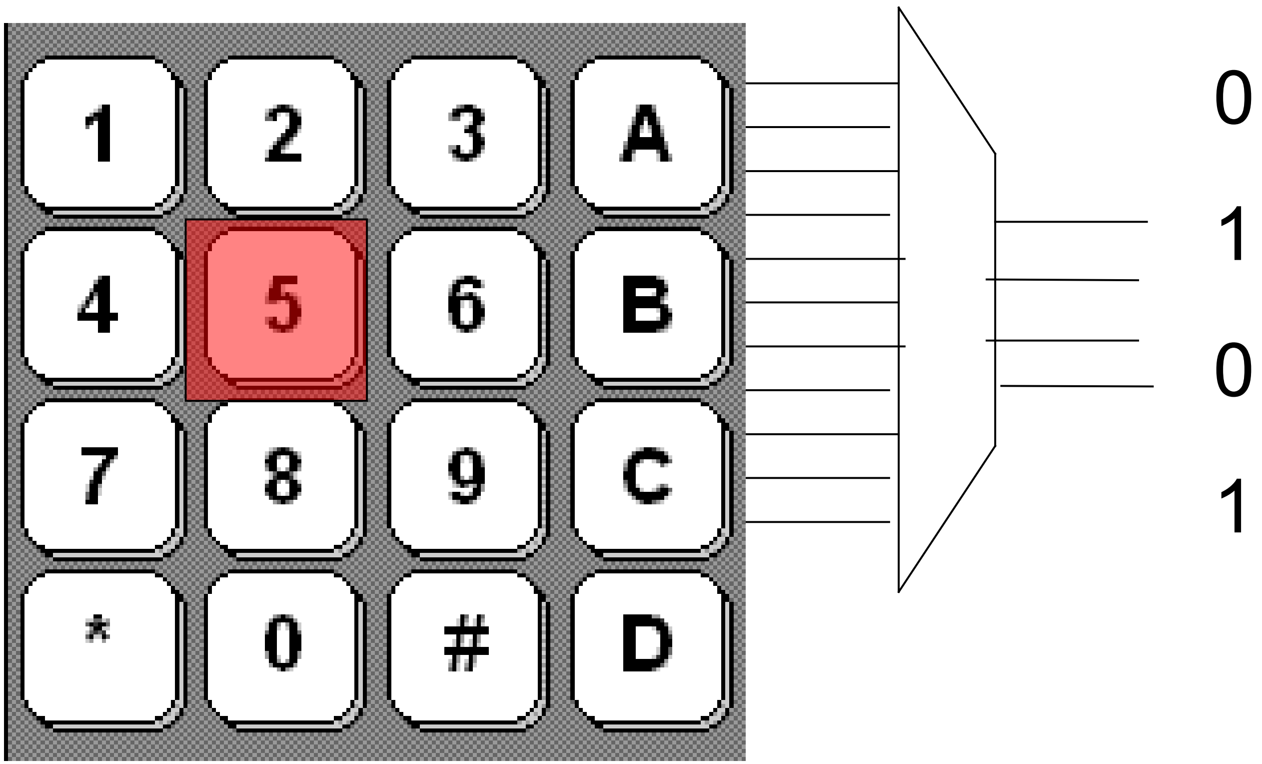
Encodeur 16→4

Exemple d'application



Encodeur 16→4

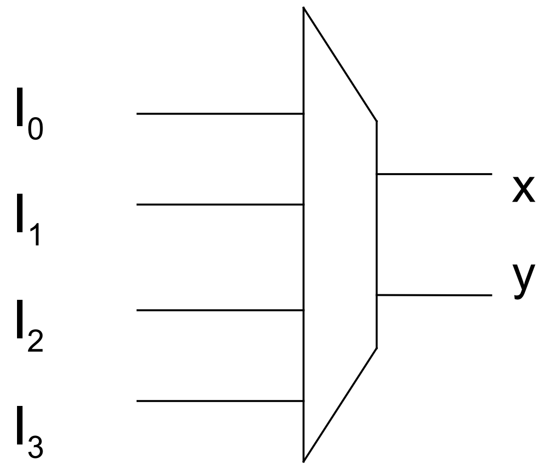
Exemple d'application



Encodeur 16→4

L'encodeur binaire (4→2)

y	x		I_3	I_2	I_1	I_0
0	0		0	0	0	0
0	0		x	x	x	1
1	0		x	x	1	0
0	1		x	1	0	0
1	1		1	0	0	0



$$X = \overline{I0}.\overline{I1}.(I2 + I3)$$

$$Y = \overline{I0}.(I1 + .\overline{I2}.I3)$$

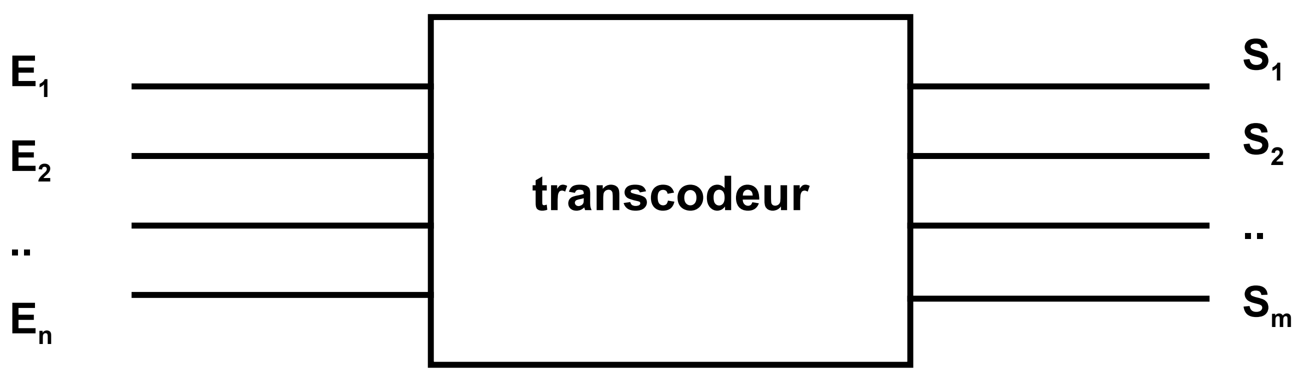
Exercice

- Donner la table de vérité
- d’un encodeur 16 ➔ 4
- Donner le schéma bloc

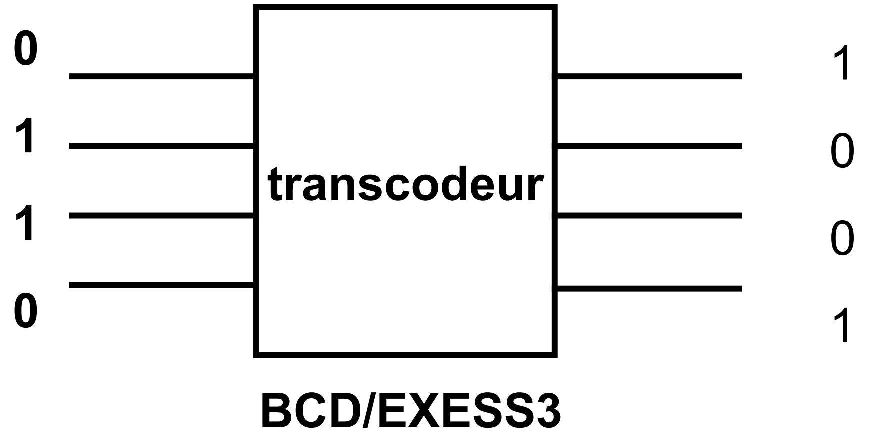
Transcodeurs

9. Le transcodeur

- C’est un circuit combinatoire qui permet de transformer un code X (sur n bits) en entrée en un code Y (sur m bits) en sortie.



transcodeur



- Décimal → BCD
- BCD → décimal
- XS 3 → décimal
- Gray → excédant 3
- DCB → afficheur 7 segments
- binaire 5 bits → DCB
- DCB → binaire 5 bits

Exercice

- Donner la table de vérité
- Transcodeur BCD /Exces 3
- Donner le schéma bloc

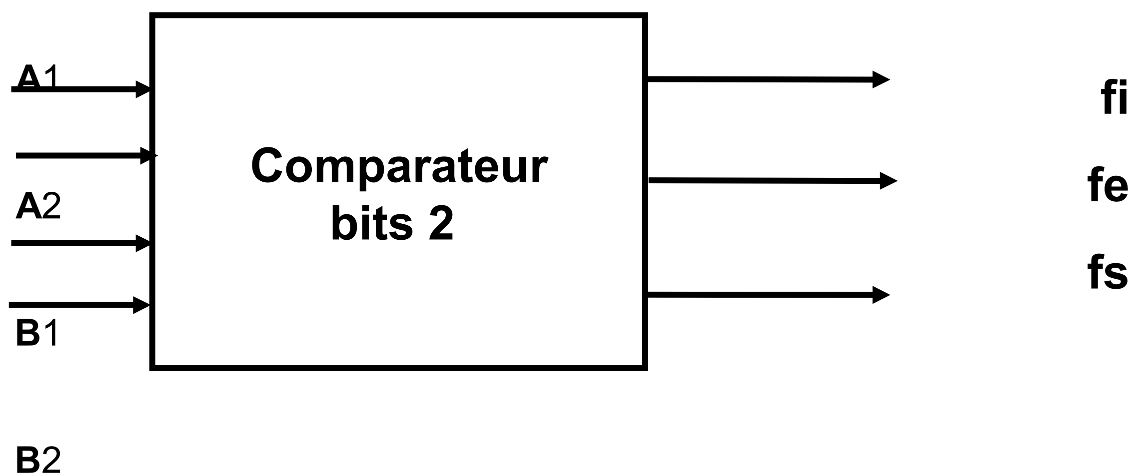
Exemple : Transcodeur BCD/EXESS3

T	Z	Y	X		D	C	B	A
1	1	0	0		0	0	0	0
0	0	1	0		1	0	0	0
1	0	1	0		0	1	0	0
0	1	1	0		1	1	0	0
1	1	1	0		0	0	1	0
0	0	0	1		1	0	1	0
1	0	0	1		0	1	1	0
0	1	0	1		1	1	1	0
1	1	0	1		0	0	0	1
0	0	1	1		1	0	0	1
x	x	x	x		0	1	0	1
x	x	x	x		1	1	0	1
x	x	x	x		0	0	1	1
x	x	x	x		1	0	1	1
x	x	x	x		0	1	1	1
x	x	x	x		1	1	1	1

Comparateur

4.2 Comparateur 2 bits

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



A=B si. 1

A2=B2 et A1=B1

$$fe = (\overline{A2 \oplus B2}).(\overline{A1 \oplus B1})$$

A>B si. 2

)A2 > B2 ou (A2=B2 et A1>B1

$$fs = A2.\overline{B2} + (\overline{A2 \oplus B2}).(A1.\overline{B1})$$

A<B si. 3

)A2 < B2 ou (A2=B2 et A1<B1

$$fi = \overline{A2}.B2 + (\overline{A2 \oplus B2}).(\overline{A1}.B1)$$

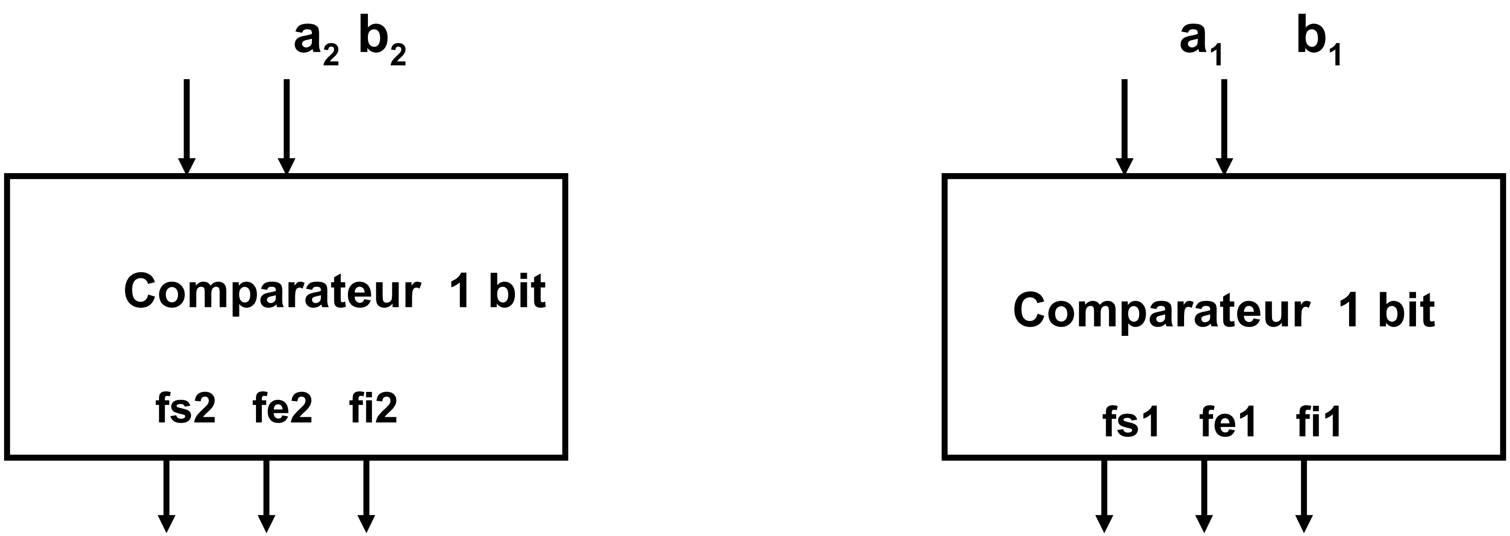
fi	fe	fs		B1	B2	A1	A2
0	1	0		0	0	0	0
1	0	0		1	0	0	0
1	0	0		0	1	0	0
1	0	0		1	1	0	0
0	0	1		0	0	1	0
0	1	0		1	0	1	0
1	0	0		0	1	1	0
1	0	0		1	1	1	0
0	0	1		0	0	0	1
0	0	1		1	0	0	1
0	1	0		0	1	0	1
1	0	0		1	1	0	1
0	0	1		0	0	1	1
0	0	1		1	0	1	1
0	0	1		0	1	1	1
0	1	0		1	1	1	1

4.2.2 comparateur 2 bits avec des comparateurs 1 bit

C’est possible de réaliser un comparateur 2 bits en utilisant des •
.comparateurs 1 bit et des portes logiques

Il faut utiliser un comparateur pour comparer les bits du poids faible •
.et un autre pour comparer les bits du poids fort

Il faut combiner entre les sorties des deux comparateurs utilisés •
.pour réaliser les sorties du comparateur final



A=B si. 1

A2=B2 et A1=B1

$$fe = \overline{(A2 \oplus B2)}.\overline{(A1 \oplus B1)} = fe2.fe1$$

A>B si. 2

)A2 > B2 ou (A2=B2 et A1>B1

$$fs = A2.\overline{B2} + \overline{(A2 \oplus B2)}.(A1.\overline{B1}) = fs2 + fe2.fs1$$

A<B si. 3

)A2 < B2 ou (A2=B2 et A1<B1

$$fi = \overline{A2}.B2 + \overline{(A2 \oplus B2)}.(\overline{A1}.B1) = fi2 + fe2.fi1$$

4.2.3 Comparateur avec des entrées de mise en cascade

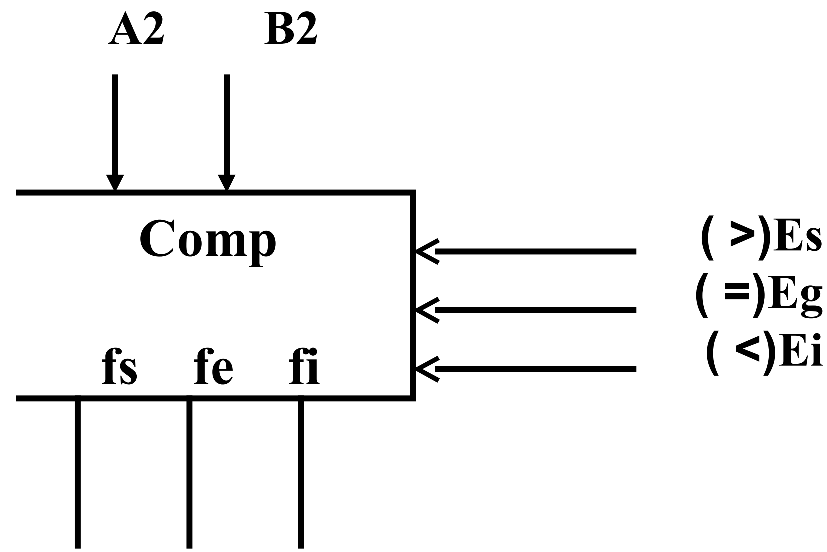
- On remarque que :
  - Si A2 >B2 alors **A > B**
  - Si A2<B2 alors **A < B**

- Par contre si A2=B2 alors il faut tenir en compte du résultat de la comparaison des bits du poids faible.

- Pour cela on rajoute au comparateur des entrées qui nous indiquent le résultat de la comparaison précédente.

- Ces entrées sont appelées des entrées de mise en cascade.

fs	fe	fs		Ei	Eg	Es	B2	A2
0	0	1		X	X	X		A2>B2
1	0	0		X	X	X		A2<B2
0	0	1		0	0	1		
0	1	0		0	1	0		A2=B1
1	0	0		1	0	0		



fs= (A2>B2) ou (A2=B2).Es

fi= (A2<B2) ou (A2=B2).Ei

fe=(A2=B2).Eg

Exercice

- Réaliser un comparateur 4 bits en utilisant des comparateurs 2 bits avec des entrées de mise en cascade?