

## Chapitre 4 : 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**.

1

## 1. Les Circuits combinatoires

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

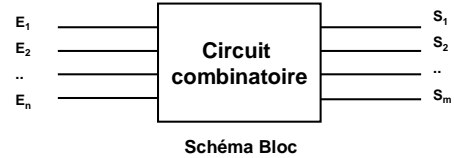


Schéma Bloc

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

2

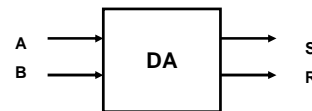
## Exemple de Circuits combinatoires

- Demi Additionneur
- Additionneur complet
- Compareur
- Multiplexeur
- Demultiplexeur
- Encodeur
- Décodeur

3

## 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 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

4

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

$$\left\{ \begin{array}{l} 0+0 = 00 \\ 0+1 = 01 \\ 1+0 = 01 \\ 1+1 = 10 \end{array} \right.$$

- La table de vérité associée :

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

De la table de vérité on trouve :

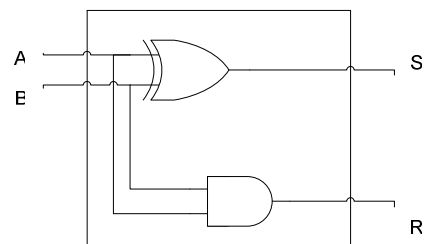
$$R = A.B$$

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

5

$$R = A.B$$

$$S = A \oplus B$$



6

### 3. L'additionneur complet

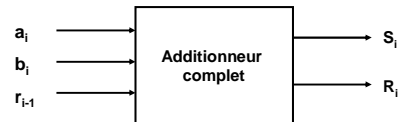
- En binaire lorsque on fait une addition il faut tenir en compte de la **retenue entrante**.

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

7

### 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.
  - $r_{i-1}$  : la retenue entrante sur un bit.
- Il possède deux sorties :
  - $S_i$  : la somme
  - $R_i$  la retenue sortante



8

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

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

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

9

Si on veut simplifier les équations on obtient :

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

$$S_i = \overline{A_i} \cdot (\overline{B_i} \cdot R_{i-1} + B_i \cdot \overline{R_{i-1}}) + A_i \cdot (\overline{B_i} \cdot \overline{R_{i-1}} + B_i \cdot R_{i-1})$$

$$S_i = \overline{A_i} (B_i \oplus R_{i-1}) + A_i (\overline{B_i} \oplus \overline{R_{i-1}})$$

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

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

$$R_i = R_{i-1} \cdot (\overline{A_i} \cdot B_i + A_i \cdot \overline{B_i}) + A_i B_i (\overline{R_{i-1}} + R_{i-1})$$

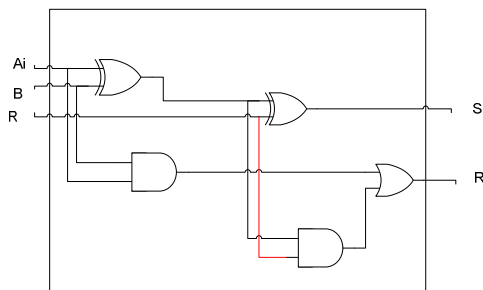
$$R_i = R_{i-1} \cdot (A_i \oplus B_i) + A_i B_i$$

10

### 3.3 Schéma d'un additionneur complet

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

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



11

### 3.4 En utilisant des Demi Additionneurs

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

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

Si on pose  $X = A_i \oplus B_i$  et  $Y = A_i B_i$

On obtient :

$$R_i = Y + R_{i-1} \cdot X$$

$$S_i = X \oplus R_{i-1}$$

et si on pose  $Z = X \oplus R_{i-1}$  et  $T = R_{i-1} \cdot X$

On obtient :

$$R_i = Y + T$$

$$S_i = Z$$

• On remarque que X et Y sont les sorties d'un demi additionneur ayant comme entrées A et B

• On remarque que Z et T sont les sorties d'un demi additionneur ayant comme entrées X et  $R_{i-1}$

12

$$X = A_i \oplus B_i$$

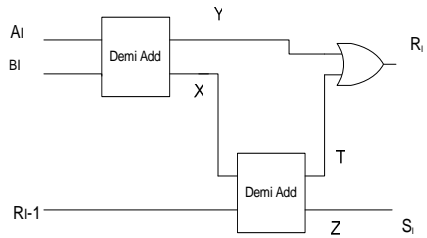
$$Y = A_i B_i$$

$$Z = X \oplus R_{i-1}$$

$$T = R_{i-1} \cdot X$$

$$R_i = Y + T$$

$$S_i = Z$$



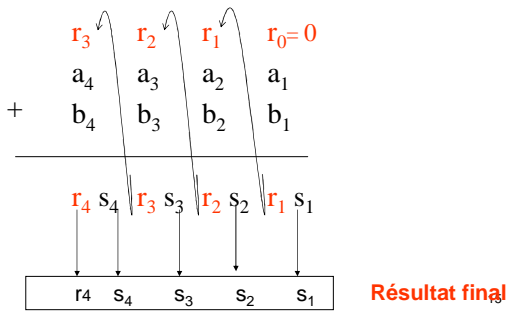
13

### 3.4 Additionneur sur 4 bits

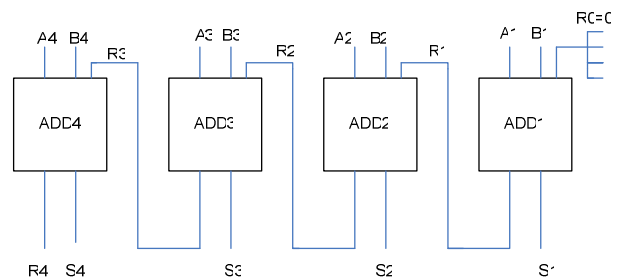
- Un additionneur sur 4 bits est un circuit qui permet de faire l'addition de deux nombres A et B de 4 bits chacun
  - $A(a_3a_2a_1a_0)$
  - $B(b_3b_2b_1b_0)$
  - En plus il tient en compte de la retenue entrante
- En sortie on va avoir le résultat sur 4 bits ainsi que la retenue ( 5 bits en sortie )
- 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 ?

14

• Lorsque on fait l'addition en binaire , on additionne **bit par bit** en commençant à partir du poids faible 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.



### 3.4.1 Additionneur 4 bits ( schéma )



16

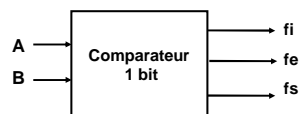
### Exercice

- 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 .

17

### 4. Le Comparateur

- C'est un circuit combinatoire qui permet de **comparer** entre deux nombres binaire A et B.
- Il possède 2 entrées :
  - A : sur un bit
  - B : sur un bit
- Il possède 3 sorties
  - fe : égalité ( A=B)
  - fi : inférieur ( A < B)
  - fs : supérieur ( A > B)



18

## 4.1 Comparateur sur un bit

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

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

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

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

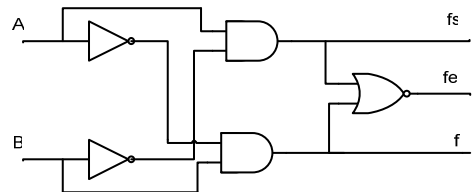
19

## Schéma d'un comparateur dur un bit

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

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

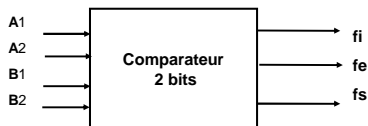
$$fe = \overline{fs + fi}$$



20

## 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.



21

- A=B si

$$A2=B2 \text{ et } A1=B1$$

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

- A>B si

$$A2 > B2 \text{ ou } (A2=B2 \text{ et } A1 > B1)$$

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

- A<B si

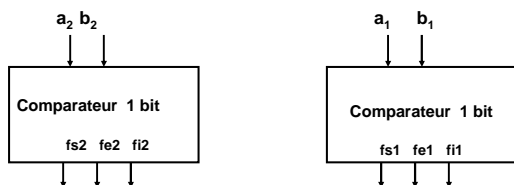
$$A2 < B2 \text{ ou } (A2=B2 \text{ et } A1 < B1)$$

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

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

## 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.



23

- A=B si

$$A2=B2 \text{ et } A1=B1$$

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

- A>B si

$$A2 > B2 \text{ ou } (A2=B2 \text{ et } A1 > B1)$$

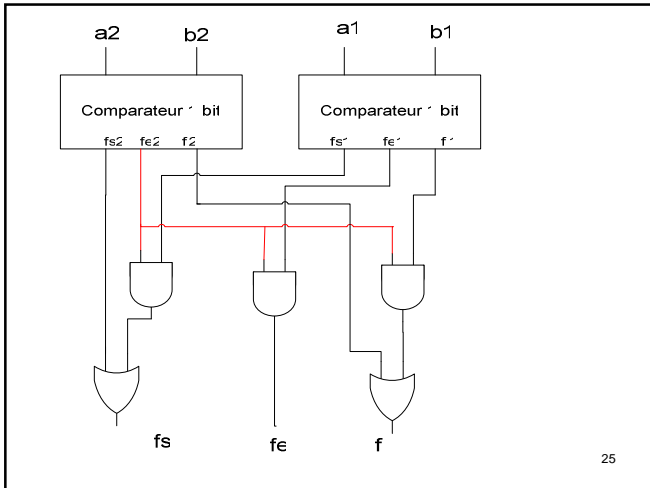
$$fs = A2 \cdot \bar{B2} + \overline{(A2 \oplus B2)} \cdot (A1 \cdot \bar{B1}) = fs2 + fe2 \cdot fs1$$

- A<B si

$$A2 < B2 \text{ ou } (A2=B2 \text{ et } A1 < B1)$$

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

24



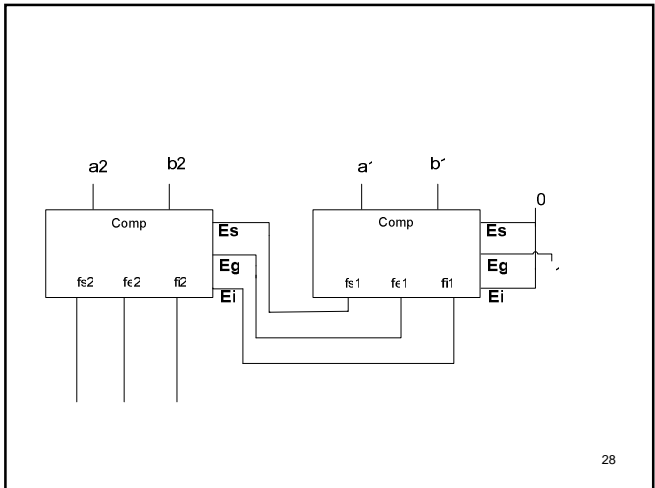
### 4.2.3 Comparateur avec des entrées de mise en cascade

- 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**.

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

A2 B2

$fs = (A2 > B2) \text{ ou } (A2 = B2).Es$   
 $fi = (A2 < B2) \text{ ou } (A2 = B2).Ei$   
 $fe = (A2 = B2).Eg$

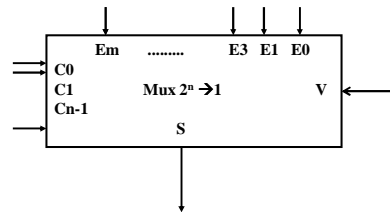


### Exercice

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

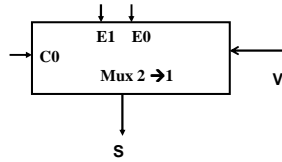
### 5. Le Multiplexeur

- 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)



### 5.1 Multiplexeur 2 → 1

V	C <sub>0</sub>	S
0	X	0
1	0	E <sub>0</sub>
1	1	E <sub>1</sub>

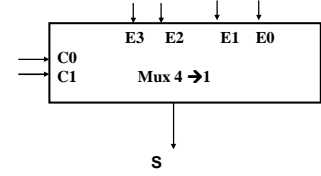


$$S = V \cdot (\overline{C_0} \cdot E_0 + C_0 \cdot E_1)$$

31

### 5.2 Multiplexeur 4 → 1

C1	C0	S
0	0	E <sub>0</sub>
0	1	E <sub>1</sub>
1	0	E <sub>2</sub>
1	1	E <sub>3</sub>

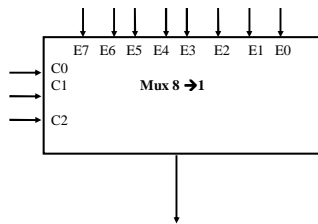


$$S = \overline{C_1} \cdot \overline{C_0} \cdot (E_0) + \overline{C_1} \cdot C_0 \cdot (E_1) + C_1 \cdot \overline{C_0} \cdot (E_2) + C_1 \cdot C_0 \cdot (E_3)$$

32

### 5.3 Multiplexeur 8 → 1

C2	C1	C0	S
0	0	0	E <sub>0</sub>
0	0	1	E <sub>1</sub>
0	1	0	E <sub>2</sub>
0	1	1	E <sub>3</sub>
1	0	0	E <sub>4</sub>
1	0	1	E <sub>5</sub>
1	1	0	E <sub>6</sub>
1	1	1	E <sub>7</sub>



$$S = \overline{C_2} \cdot \overline{C_1} \cdot \overline{C_0} \cdot (E_0) + \overline{C_2} \cdot \overline{C_1} \cdot C_0 \cdot (E_1) + \overline{C_2} \cdot C_1 \cdot \overline{C_0} \cdot (E_2) + \overline{C_2} \cdot C_1 \cdot C_0 \cdot (E_3) + C_2 \cdot \overline{C_1} \cdot \overline{C_0} \cdot (E_4) + C_2 \cdot \overline{C_1} \cdot C_0 \cdot (E_5) + C_2 \cdot C_1 \cdot \overline{C_0} \cdot (E_6) + C_2 \cdot C_1 \cdot C_0 \cdot (E_7)$$

33

### Exemple : Réalisation d'un additionneur complet avec des multiplexeurs 8 → 1

• Nous avons besoin d'utiliser deux multiplexeurs : Le premier pour réaliser la fonction de la somme et l'autres pour donner la retenue.

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

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

34

#### Réalisation de la fonction de la somme

$$S_i = \overline{A_i} \cdot \overline{B_i} \cdot \overline{R_{i-1}}(0) + \overline{A_i} \cdot \overline{B_i} \cdot R_{i-1}(1) + \overline{A_i} \cdot B_i \cdot \overline{R_{i-1}}(1) + \overline{A_i} \cdot B_i \cdot R_{i-1}(0) + A_i \cdot \overline{B_i} \cdot \overline{R_{i-1}}(1) + A_i \cdot \overline{B_i} \cdot R_{i-1}(0) + A_i \cdot B_i \cdot \overline{R_{i-1}}(0) + A_i \cdot B_i \cdot R_{i-1}(1)$$

$$S = \overline{C_2} \cdot \overline{C_1} \cdot \overline{C_0} \cdot (E_0) + \overline{C_2} \cdot \overline{C_1} \cdot C_0 \cdot (E_1) + \overline{C_2} \cdot C_1 \cdot \overline{C_0} \cdot (E_2) + \overline{C_2} \cdot C_1 \cdot C_0 \cdot (E_3) + C_2 \cdot \overline{C_1} \cdot \overline{C_0} \cdot (E_4) + C_2 \cdot \overline{C_1} \cdot C_0 \cdot (E_5) + C_2 \cdot C_1 \cdot \overline{C_0} \cdot (E_6) + C_2 \cdot C_1 \cdot C_0 \cdot (E_7)$$

On pose :

$$C_2 = A_i$$

$$C_1 = B_i$$

$$C_0 = R_{i-1}$$

$$E_0 = 0, E_1 = 1, E_2 = 1, E_3 = 0, E_4 = 1, E_5 = 0, E_6 = 0, E_7 = 1$$

35

#### Réalisation de la fonction de la retenue

$$R_i = \overline{A_i} \cdot \overline{B_i} \cdot \overline{R_{i-1}}(0) + \overline{A_i} \cdot \overline{B_i} \cdot R_{i-1}(0) + \overline{A_i} \cdot B_i \cdot \overline{R_{i-1}}(0) + \overline{A_i} \cdot B_i \cdot R_{i-1}(0) + A_i \cdot \overline{B_i} \cdot \overline{R_{i-1}}(0) + A_i \cdot \overline{B_i} \cdot R_{i-1}(0) + A_i \cdot B_i \cdot \overline{R_{i-1}}(0) + A_i \cdot B_i \cdot R_{i-1}(0)$$

$$S = \overline{C_2} \cdot \overline{C_1} \cdot \overline{C_0} \cdot (E_0) + \overline{C_2} \cdot \overline{C_1} \cdot C_0 \cdot (E_1) + \overline{C_2} \cdot C_1 \cdot \overline{C_0} \cdot (E_2) + \overline{C_2} \cdot C_1 \cdot C_0 \cdot (E_3) + C_2 \cdot \overline{C_1} \cdot \overline{C_0} \cdot (E_4) + C_2 \cdot \overline{C_1} \cdot C_0 \cdot (E_5) + C_2 \cdot C_1 \cdot \overline{C_0} \cdot (E_6) + C_2 \cdot C_1 \cdot C_0 \cdot (E_7)$$

On pose :

$$C_2 = A_i$$

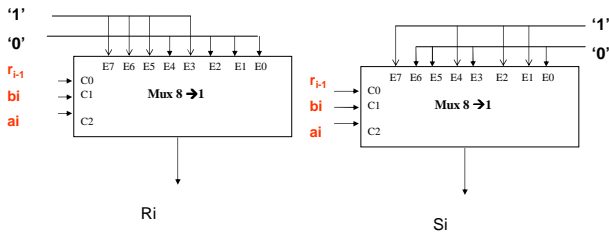
$$C_1 = B_i$$

$$C_0 = R_{i-1}$$

$$E_0 = 0, E_1 = 0, E_2 = 0, E_3 = 1, E_4 = 0, E_5 = 1, E_6 = 1, E_7 = 1$$

36

### Réalisation d'un additionneur complet avec des multiplexeurs 8→1



37

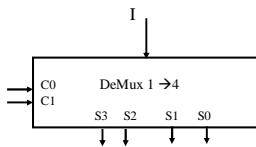
### Exercice

- Réaliser le circuit qui permet de trouver le maximum entre deux nombres A et B sur un Bit en utilisant le minimum de portes logiques et de circuits combinatoires?

38

## 6. 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<sup>n</sup> sorties
  - N entrées de sélection ( commandes)



39

### 6.1 Demultiplexeur 1→4

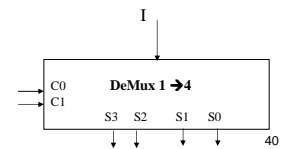
C1	C0	S3	S2	S1	S0
0	0	0	0	0	i
0	1	0	0	i	0
1	0	0	i	0	0
1	1	i	0	0	0

$$S_0 = \overline{C1}.\overline{C0}.(I)$$

$$S_1 = \overline{C1}.C0.(I)$$

$$S_2 = C1.\overline{C0}.(I)$$

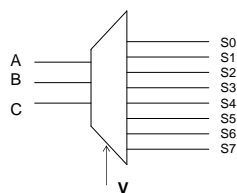
$$S_3 = C1.C0.(I)$$



40

## 7. Le décodeur binaire

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

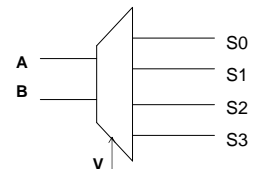


Un décodeur 3→8

41

### Décodeur 2→4

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



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

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

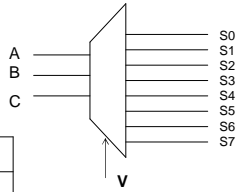
$$S_2 = (A.\overline{B}).V$$

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

42

## Décodeur 3→8

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



$$\begin{aligned}
 S_0 &= \overline{A} \cdot \overline{B} \cdot \overline{C} \\
 S_1 &= \overline{A} \cdot \overline{B} \cdot C \\
 S_2 &= \overline{A} \cdot B \cdot \overline{C} \\
 S_3 &= \overline{A} \cdot B \cdot C \\
 S_4 &= A \cdot \overline{B} \cdot \overline{C} \\
 S_5 &= A \cdot \overline{B} \cdot C \\
 S_6 &= A \cdot B \cdot \overline{C} \\
 S_7 &= A \cdot B \cdot C
 \end{aligned}$$

## Réalisation d'un additionneur complet avec des décodeurs binaire 3→8

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

$$0 \ 0 \ 1 \quad 0 \ 1 \ 0 \quad 1 \ 0 \ 0 \quad 1 \ 1 \ 1$$

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

$$0 \ 1 \ 1 \quad 1 \ 0 \ 1 \quad 1 \ 1 \ 0 \quad 1 \ 1 \ 1$$

On pose  $A=A_i, B=B_i, C=R_{i-1}$

$$S_0 = \overline{A} \cdot \overline{B} \cdot C, S_1 = \overline{A} \cdot B \cdot \overline{C}, S_2 = \overline{A} \cdot B \cdot C, S_3 = A \cdot \overline{B} \cdot \overline{C},$$

$$S_4 = A \cdot \overline{B} \cdot C, S_5 = A \cdot B \cdot \overline{C}, S_6 = A \cdot B \cdot C, S_7 = A \cdot B \cdot C$$

$$R_i = S_3 + S_5 + S_6 + S_7$$

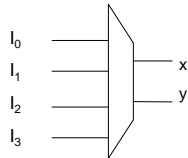
$$S_i = S_1 + S_2 + S_4 + S_7$$

44

## 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.

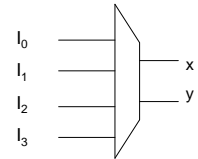
Encodeur 4→2



45

## L'encodeur binaire ( 4→2)

I <sub>0</sub>	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	x	y
0	0	0	0	0	0
1	x	x	x	0	0
0	1	x	x	0	1
0	0	1	x	1	0
0	0	0	1	1	1



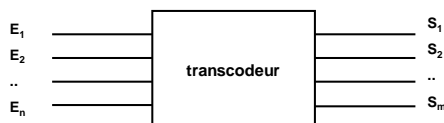
$$X = \overline{I_0} \cdot \overline{I_1} \cdot (I_2 + I_3)$$

$$Y = \overline{I_0} \cdot (I_1 + \overline{I_2} \cdot I_3)$$

46

## 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.



47

## Exemple : Transcodeur BCD/EXESS3

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

48