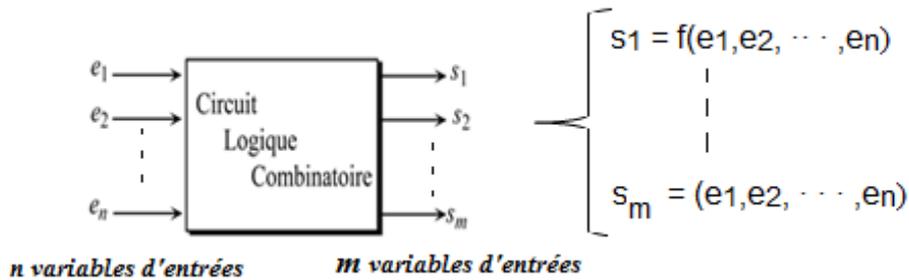


## Chapitre 4 : Les circuits combinatoires logiques

### 4-1 Introduction

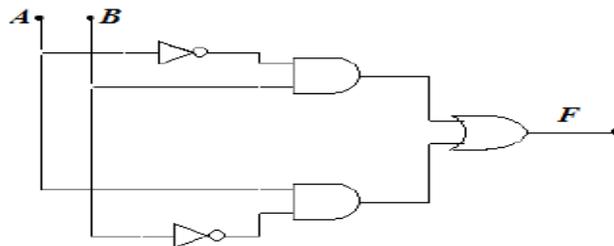
Un circuit gouverné par les règles de la logique combinatoire possède une ou plusieurs entrées, et une ou plusieurs sorties, et obéit à la propriété suivante : L'état de la (ou des) sortie(s) à un instant donné ne dépend que du circuit et de la valeur des entrées à cet instant.



#### 4-1-1 Réalisation matérielle d'une fonction logique

En combinant entre elles les différentes portes logiques on peut a priori réaliser n'importe quelle fonction logique. On appelle logigramme la réalisation d'une fonction complexe à l'aide des portes de base. A titre d'exemple réalisation la fonction **OU-EXCLUSIF** (**XOR**) en utilisant que des portes **NON**, **ET**, **OU**.

On réalise la fonction  $F = \bar{A} \cdot B + A \cdot \bar{B}$ , telle qu'elle est écrite. Obtient alors le schéma suivant :

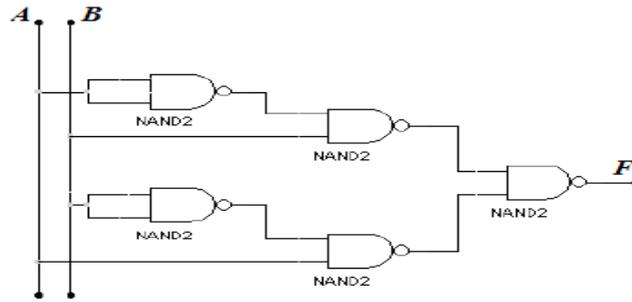


Cette solution n'est pas satisfaisante pour au moins deux raisons :

- On n'a pas cherché à minimiser le nombre de portes utilisées.
- On utilise 3 types de portes différents donc 3 boîtiers différents sur le montage, dont on peut facilement gagner de la place en utilisant qu'un seul type de portes, car un boîtier contient plusieurs portes.

Pour simplifier la réalisation, on cherche à n'utiliser qu'un seul type de portes ; par exemple en reprenant la fonction F et on la réalise qu'à base de porte NAND uniquement :

$$\begin{aligned} F &= \bar{A} \cdot B + A \cdot \bar{B} \\ &= \overline{\overline{\bar{A} \cdot B} \cdot \overline{A \cdot \bar{B}}} \end{aligned}$$



**Réalisation de la fonction OU-Exclusif uniquement avec des portes NAND**

Remarque : on comparant entre les circuits, dans le deuxième on peut le réaliser qu'avec un seul boîtier (circuit intégré) de porte NAND.

**4-2 Additionneur et soustracteur**

**4-2-1 Demi-additionneur**

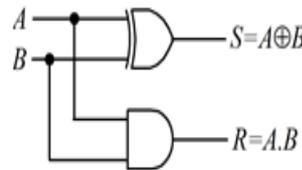
L'additionneur binaire portant sur un bit unique mène aux 4 cas notés dans la table de vérité suivante :

A	B	S (Somme)	R (Retenue)	Mintermes
0	0	0	0	$\bar{A} \bar{B}$
0	1	1	0	$\bar{A} B$
1	0	1	0	$A \bar{B}$
1	1	0	1	$A B$

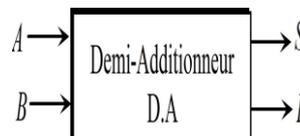
Les équations caractéristiques sont :  $S = \bar{A} \cdot B + A \cdot \bar{B}$

$S = A \oplus B$  et La retenue :  $R = A \cdot B$

Le logigramme correspondant :

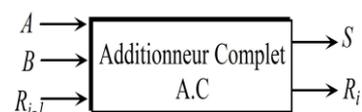


N.B : Le demi-additionneur ne tient pas compte de la retenue précédente.



**4-2-2 Additionneur complet**

La représentation de l'additionneur complet est donnée par le schéma suivant, où  $R_i$  indique la retenue et  $R_{i-1}$  la retenue précédente.



L'analyse du fonctionnement de ce dernier est illustrée par la table de vérité suivante :

A	B	R <sub>i-1</sub>	S	R <sub>i</sub>
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Pour remplir cette table on se base sur l'opération suivante :  $A+B+R_{i-1}$  et les propriétés de l'addition binaire

Exemple :

$$\begin{array}{r}
 A \quad 0 \\
 + B \quad 1 \quad 1 \\
 + R_{i-1} \quad 1 \quad (s_i + 1) \\
 \hline
 S=0, R_i = 1
 \end{array}$$

$s_i=1, r=0$   
 $s=0, r=1$

Les équations logiques des sorties S et R<sub>i</sub> basées sur les mintermes (les sorties sont à l'état 1) :

$$S = \bar{A} \cdot \bar{B} \cdot R_{i-1} + \bar{A} \cdot B \cdot \bar{R}_{i-1} + A \cdot \bar{B} \cdot \bar{R}_{i-1} + A \cdot B \cdot R_{i-1}$$

$$R_i = \bar{A} \cdot B \cdot R_{i-1} + A \cdot \bar{B} \cdot R_{i-1} + A \cdot B \cdot \bar{R}_{i-1} + A \cdot B \cdot R_{i-1}$$

Simplification des fonctions correspondante a S et R<sub>i</sub>

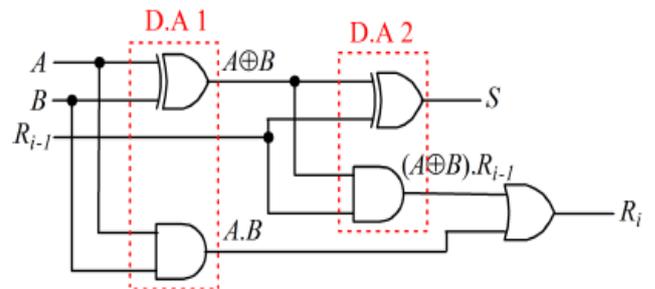
- $$\begin{aligned}
 S &= A \cdot \bar{B} \cdot \bar{R}_{i-1} + A \cdot \bar{B} \cdot R_{i-1} + \bar{A} \cdot B \cdot \bar{R}_{i-1} + \bar{A} \cdot B \cdot R_{i-1} \\
 &= (\bar{A} \cdot \bar{B} + A \cdot B) R_{i-1} + (\bar{A} \cdot B + A \cdot \bar{B}) \bar{R}_{i-1} \\
 &= \overline{(A \oplus B)} R_{i-1} + (A \oplus B) \bar{R}_{i-1}
 \end{aligned}$$

$$S = A \oplus B \oplus R_{i-1}$$

- $$\begin{aligned}
 R_i &= \bar{A} \cdot B \cdot R_{i-1} + A \cdot \bar{B} \cdot R_{i-1} + A \cdot B \cdot \bar{R}_{i-1} + A \cdot B \cdot R_{i-1} \\
 &= (\bar{A} \cdot B + A \cdot \bar{B}) R_{i-1} + A \cdot B (\bar{R}_{i-1} + R_{i-1})
 \end{aligned}$$

$$R_i = (A \oplus B) R_{i-1} + A \cdot B$$

**Logigramme de l'additionneur complet :**



**Remarque :** un Additionneur complet est constitué de deux demi Additionneur (D.A1 et D.A2).

### 4-2-3 Demi-soustracteur

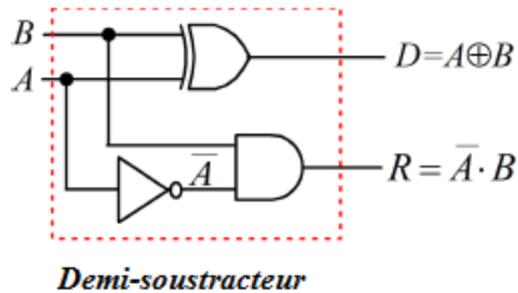
Le soustracteur binaire portant sur un bit unique mène aux 4 cas présentés par la table de vérité suivante :

A	B	D (différence)	R(Retenue)	Mintermes
0	0	0	0	$\bar{A} \bar{B}$
0	1	1	1	$\bar{A} B$
1	0	1	0	$A \bar{B}$
1	1	0	0	$A B$

Les équations logiques sont :  $D = \bar{A} \cdot B + A \cdot \bar{B}$

$D = A \oplus B$  et la retenue :  $R = \bar{A} \cdot B$

Le logigramme du demi- soustracteur :



### 4-2-4 Soustracteur complet

L'analyse de fonctionnement du soustracteur complet est illustrée par la table de vérité suivante

A	B	R <sub>i-1</sub>	D	R <sub>i</sub>
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	1	1	1

Pour remplir cette table on se base sur l'opération suivante :  $A - B - R_{i-1}$  et les propriétés de la soustraction binaire

Les équations logiques des sorties D et Ri en utilisant les mintermes :

$$D = \bar{A} \cdot \bar{B} \cdot R_{i-1} + \bar{A} \cdot B \cdot \bar{R}_{i-1} + A \cdot \bar{B} \cdot \bar{R}_{i-1} + A \cdot B \cdot R_{i-1}$$

$$R_i = \bar{A} \cdot \bar{B} \cdot R_{i-1} + \bar{A} \cdot B \cdot \bar{R}_{i-1} + \bar{A} \cdot B \cdot R_{i-1} + A \cdot B \cdot R_{i-1}$$

Simplification d'équations de D et Ri obtenues via la table de vérité précédente :

$$D = \bar{A} \cdot \bar{B} \cdot R_{i-1} + \bar{A} \cdot B \cdot \bar{R}_{i-1} + A \cdot \bar{B} \cdot \bar{R}_{i-1} + A \cdot B \cdot R_{i-1}$$

$$= (\bar{A} \cdot \bar{B} + A \cdot B) R_{i-1} + (\bar{A} \cdot B + A \cdot \bar{B}) \bar{R}_{i-1}$$

$$= (\overline{A \oplus B}) R_{i-1} + (A \oplus B) \bar{R}_{i-1}$$

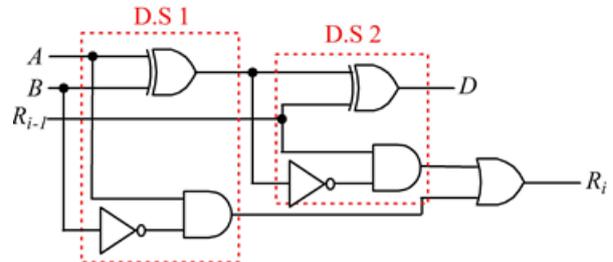
$$D = A \oplus B \oplus R_{i-1}$$

$$R_i = \bar{A} \cdot \bar{B} \cdot R_{i-1} + \bar{A} \cdot B \cdot \bar{R}_{i-1} + \bar{A} \cdot B \cdot R_{i-1} + A \cdot B \cdot R_{i-1}$$

$$= (\bar{A} \cdot \bar{B} + A \cdot B) R_{i-1} + A \cdot B (R_{i-1} + \bar{R}_{i-1})$$

$$R_i = (A \oplus B) R_{i-1} + A \cdot B$$

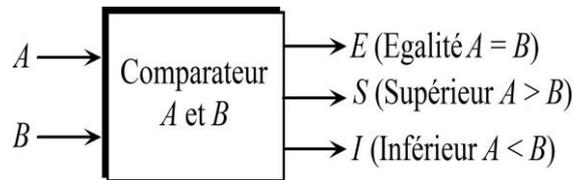
Logigramme du soustracteur complet :



**Remarque** : un Additionneur complet est constitué de deux demi Additionneur (D.S1 et D.S2).

### 4-3 Comparateur

La représentation du comparateur entre 2 nombres A et B est donné par le schéma suivant :



La table de vérité suivante donne l'analyse du fonctionnement d'un **comparateur à 1 bit** :

A	B	E	S	I
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

Les équations caractéristiques des sorties E, S et I en prenant les mintermes :

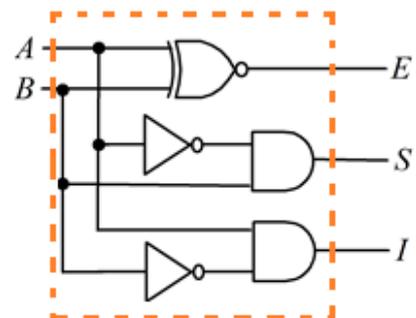
$$E = \bar{A} \cdot \bar{B} + A \cdot B$$

$$E = (A \oplus B)$$

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

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

Le logigramme correspondant au comparateur à 1 bit :



#### 4-4 Transcodeur, codeur et décodeur

##### 4-4-1 Transcodeur

Un transcodeur est un circuit combinatoire permettant de passer d'un code à un autre.

**Exemple :** Concevoir un transcodeur binaire vers Gray à 4 bits

Les variables A, B, C et D représentent le nombre en code binaire, et X, Y, Z et T représentent le même nombre en code Gray.

Analyse par la table de vérité :

N° Décimal	ABCD	XYZT
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Les équations caractéristiques des sorties en fonctions des entrées (en utilisant la table de Karnaugh)

➤  $X = F(A, B, C, D)$

X

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

L'équation caractéristique de la sortie X :

✓  $X = A$

➤  $Y = F(A, B, C, D)$

Y

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

   $\bar{A}B$   
   $A\bar{B}$

L'équation caractéristique de la sortie Y :

$Y = \bar{A}B + A\bar{B}$

✓  $Y = A \oplus B$

➤  $Z = F(A, B, C, D)$

Z

L'équation caractéristique de la sortie Z :

$$Z = \overline{B}C + B\overline{C}$$

✓  $Z = B \oplus C$

AB\CD	00	01	11	10	$\overline{B}C$
00	0	0	1	1	
01	1	1	0	0	
11	1	1	0	0	
10	0	0	1	1	

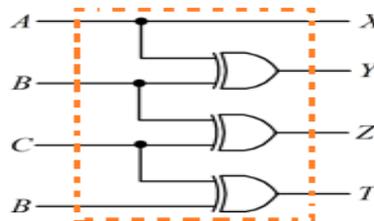
➤  $T = F(A, B, C, D)$

$$T = \overline{C}D + C\overline{D}$$

✓  $T = C \oplus D$

AB\CD	00	01	11	10	$\overline{C}D$
00	0	1	0	1	
01	0	1	0	1	
11	0	1	0	1	
11	0	1	0	1	

Le logigramme est le suivant :



#### 4-4-2 Codeur et décodeur

##### 4-4-2-1 Codeur

Un codeur (ou encodeur) reçoit un niveau valide a l'une de ses entrées, représentant par exemple un chiffre, une lettre, etc. Il le convertit en une sortie codée (par exemple en binaire ou en BCD).

**Exemple :** Codeur décimal - BCD

Il permet d'exprimer un nombre décimal en son équivalent binaire.

La table de vérité d'un codeur est la suivante (N)<sub>10</sub> et Le nombre (ABCD)<sub>BCD</sub>

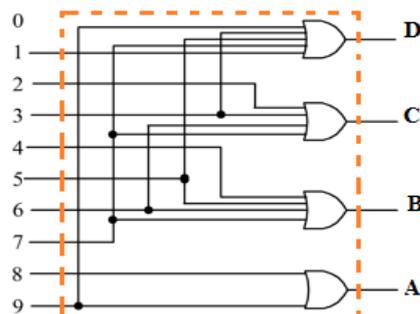
Les expressions logiques sont :

$A = 8+9, \quad B = 4 +5+6+7$

$C = 2+3+6+7, \quad D = 1+3+5+7+9$

Le schéma logique correspondant est donné par la figure

suivante :



N	ABCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

### 4-4-2-2 Décodeur

Le décodeur BCD-décimal est un circuit logique qui traduit un nombre binaire vers son équivalent décimal.

La table de vérité d'un décodeur est la suivante (ABCD)<sub>BCD</sub> et Le nombre (N)<sub>10</sub>

N	ABCD	S <sub>0</sub>	S <sub>1</sub>	S <sub>2</sub>	S <sub>3</sub>	S <sub>4</sub>	S <sub>5</sub>	S <sub>6</sub>	S <sub>7</sub>	S <sub>8</sub>	S <sub>9</sub>
0	0000	1	0	0	0	0	0	0	0	0	0
1	0001	0	1	0	0	0	0	0	0	0	0
2	0010	0	0	1	0	0	0	0	0	0	0
3	0011	0	0	0	1	0	0	0	0	0	0
4	0100	0	0	0	0	1	0	0	0	0	0
5	0101	0	0	0	0	0	1	0	0	0	0
6	0110	0	0	0	0	0	0	1	0	0	0
7	0111	0	0	0	0	0	0	0	1	0	0
8	1000	0	0	0	0	0	0	0	0	1	0
9	1001	0	0	0	0	0	0	0	0	0	1

Les expressions logiques sont :

Afin d'obtenir les équations de sortie, on établit un tableau de Karnaugh de 16 cases pour chaque S<sub>i</sub> (i = 0, . . . ,9). On complète les cases allant de 10 à 15 par des ∅ qu'on choisit librement (0 ou 1) afin d'avoir l'équation la plus simplifiée. On aura:

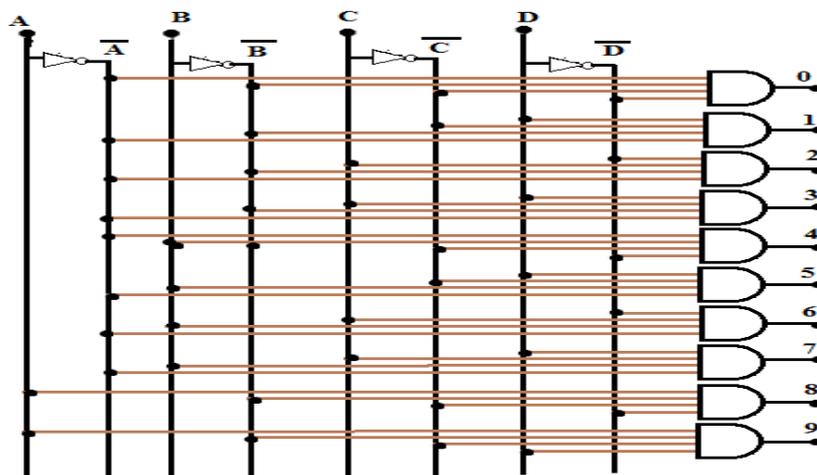
Par exemple pour le cas S<sub>2</sub> correspond à nombre 2 :

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

$$0 = \bar{A} \bar{B} \bar{C} \bar{D}; 1 = \bar{A} \bar{B} \bar{C} D; 2 = \bar{A} \bar{B} C \bar{D}; 3 = \bar{A} \bar{B} C D; 4 = \bar{A} B \bar{C} \bar{D}$$

$$5 = \bar{A} B \bar{C} D; 6 = \bar{A} B C \bar{D}; 7 = \bar{A} B C D; 8 = A \bar{B} \bar{C} \bar{D}; 9 = A \bar{B} \bar{C} D$$

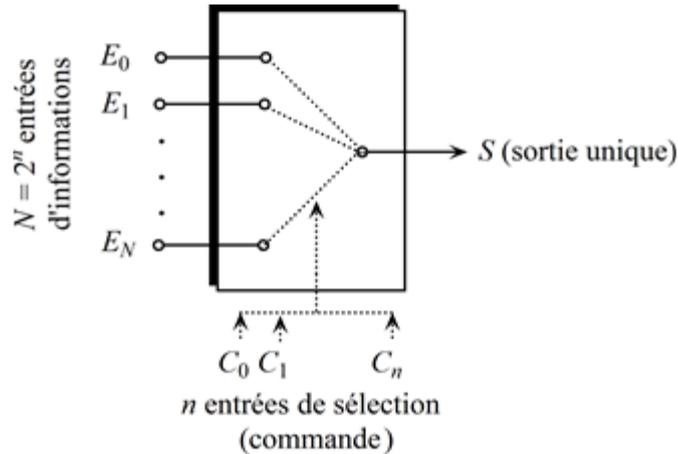
Le schéma logique correspondant est donné par la figure suivante :



### 4-5 Circuits d'aiguillage d'information

#### 4-5-1 Multiplexeur

C'est un circuit combinatoire permettant de réaliser un aiguillage de l'une des entrées en une sortie unique, dont la représentation est donnée par le schéma suivant :



**N.B :** Pour  $N = 2^n$  entrées (avec  $n$  entier positif) correspond  $n$  éléments binaire de commande (sélection).

#### Exemple-1 : Multiplexeur 2 vers 1

Il s'agit d'un multiplexeur à 2 ( $2^1$ ) entrées (qu'on note  $E_0$  et  $E_1$ ), qui nécessite une (1) entrée de commande (qu'on nomme  $C_0$ ) et une seule sortie ( $S$ ).

Son fonctionnement en aiguillage se résume par :

$$\begin{cases} S = E_0 & \text{si } C_0 = 0 \\ S = E_1 & \text{si } C_0 = 1 \end{cases}$$

L'analyse du fonctionnement est portée la table de vérité suivante :

$C_0$	$E_1$	$E_0$	$S$	
0	0	0	0	$S = E_0$
0	0	1	1	
0	1	0	0	
0	1	1	1	
1	0	0	0	$S = E_1$
1	0	1	0	
1	1	0	1	
1	1	1	1	

La table peut être simplifiée

$C_0$	$S$
0	$E_0$
1	$E_1$

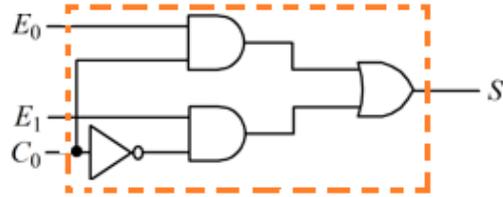
➤ **L'équation caractéristique** est  $S = \bar{C}_0 E_0 + C_0 E_1$

En utilisant le tableau de Karnaugh :

$C_0 \backslash E_1 E_0$	00	01	11	10
0	0	1	1	0
1	0	0	1	1

→  $\bar{C}_0 E_0$  (points to the 00 column)  
→  $C_0 E_1$  (points to the 11 column)

Le logigramme du multiplexeur 2 vers 1 :



**Exemple-2 : Multiplexeur 4 vers 1.**

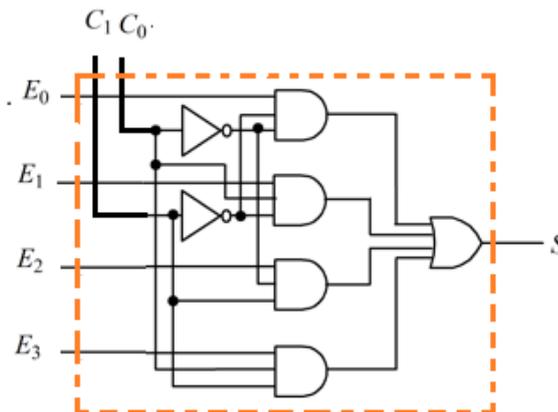
C'est un multiplexeur à 4 ( $2^2$ ) entrées ( $E_0, E_1, E_2$  et  $E_3$ ), qui nécessite 2 entrées de commande ( $C_0$  et  $C_1$ ) et une seule sortie ( $S$ ).

Son fonctionnement est donné par la table de vérité suivante :

$C_1$	$C_0$	<b>S</b>
0	0	<b><math>E_0</math></b>
0	1	<b><math>E_1</math></b>
1	0	<b><math>E_2</math></b>
1	1	<b><math>E_3</math></b>

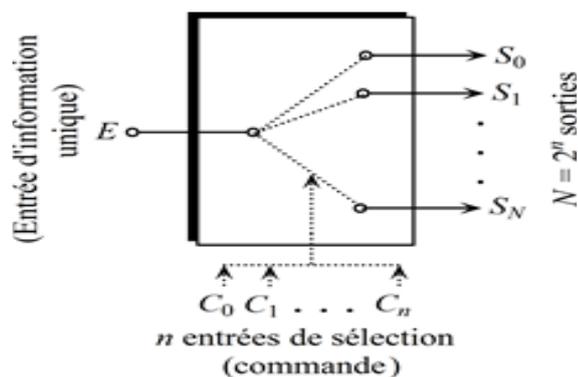
L'expressions logique est :  $S = \bar{C}_1 \bar{C}_0 E_0 + \bar{C}_1 C_0 E_1 + C_1 \bar{C}_0 E_2 + C_1 C_0 E_3$

Le logigramme du multiplexeur 4 vers 1 :



**4-5-2 Démultiplexeur**

Le démultiplexeur réalise l'opération inverse de celle du multiplexeur. Il comporte une seule entrée d'information (ou de données)  $E$ ,  $n$  entrées de commande  $C_i$  avec  $i = 0, 1, \dots, n$  (appelées aussi entrées d'adresse ou de sélection) et  $N = 2^n$  sorties ( $S_0, S_1, \dots, S_N$ ). Le schéma représentatif du démultiplexeur est illustré par la figure suivante :



**Exemple : Démultiplexeur 1 vers 4**

C'est un démultiplexeur a 4 ( $2^2$ ) sorties ( $S_0$ ,  $S_1$ ,  $S_2$  et  $S_3$ ), qui nécessite 2 entrées de commande ( $C_0$  et  $C_1$ ) et une seule entrée( $E$ ).

Le fonctionnement du démultiplexeur est donné par la table de vérité simplifiée ( $E=0$  c'est-à-dire aucune information) suivante :

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

Les équations caractéristiques des diverse sorties en prenant les "1" de la table de vérité précédente :

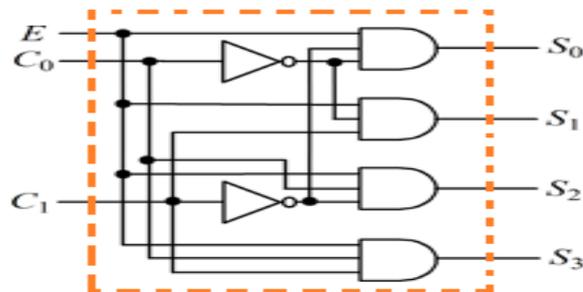
$$S_0 = \bar{C}_1 \bar{C}_0 E,$$

$$S_1 = \bar{C}_1 C_0 E$$

$$S_2 = C_1 \bar{C}_0 E$$

$$S_3 = C_1 C_0 E$$

Le logigramme du démultiplexeur 4 vers 1 est représenté par la figure ci-après :



**4-5-3 Application des multiplexeurs / démultiplexeurs**

Si l'on envoie à distance les informations issues d'un grand nombre de sources différentes ; on multiplexe ces informations pour les transmettre en série sur une seule ligne (conversion parallèle- série). A l'autre extrémité de la ligne ; il faut démultiplexeur ; c'est-à-dire restituer les informations sur les récepteurs homologues des émetteurs ; le démultiplexeur est réalisé à l'aide des décodeurs.

Référence de quelques circuits intégrés existant

<p><b>Fonction ET</b>                      7408 : quatre portes logiques ET à deux entrées                      7409 : quatre portes logiques ET à deux entrées avec sortie à collecteur ouvert                      7411 : trois portes logiques ET à trois entrées                      7415 : trois portes logiques ET à trois entrées avec sortie à collecteur ouvert                      7421 : ET à quatre entrées                      74800 : ET et NON-ET à quatre entrées                      74808 : Six portes logiques ET à deux entrées</p>	<p><b>Fonction NON</b>                      7404 : six portes logiques                      7405 : six portes logiques                      7406 : six portes logiques                      7414 : six portes logiques                      7416 : six portes logiques                      7419 : six portes logiques                      74619 : Huit portes logiques</p>
<p><b>Fonction NON-ET (ou NAND)</b>                      7400 : quatre portes logiques à deux entrées                      7401 : quatre portes logiques à deux entrées                      7403 : quatre portes logiques à deux entrées                      7410 : trois portes logiques à trois entrées                      7412 : trois portes logiques NON-ET à trois entrées avec sortie à collecteur ouvert                      7413 : deux portes logiques à quatre entrées                      7418 : deux portes logiques à quatre entrées                      7430 : une porte NON-ET à huit entrées</p>	<p><b>Décodage et conversion</b>                      7441 : décodeur BCD vers décimal                      7442 : décodeur BCD vers décimal                      7443 : décodeur 4 bits                      7444 : décodeur 4 bits                      7445 : décodeur BCD                      7446 : décodeur BCD                      7447 : décodeur BCD à 7 segments                      7448 : décodeur BCD à 7 segments                      7449 : décodeur BCD à 7 segments                      74184 : Convertisseur BCD vers binaire                      74185 : Convertisseur binaire vers BCD</p>
<p><b>Fonction OU</b>                      7432 : quatre portes logiques à deux entrées                      74802 : Triple portes logiques à quatre entrées                      744075 : Trois portes logiques OU à trois entrées</p>	<p><b>Fonction NON-OU (ou NOR)</b>                      7423 : Double porte à 4 entrées                      7425 : Double porte à 4 entrées avec                      7427 : trois portes à trois entrées                      7428 : quatre portes à deux entrées                      7433 : quatre portes à deux entrées                      7436 : quatre portes à deux entrées                      74260 : deux portes à cinq entrées                      74802 : Triple portes à quatre entrées                      74805 : Six portes à deux entrées</p>
<p><b>Fonction OU EXCLUSIF (XOR)</b>                      7486 : Quatre portes logiques à deux entrées                      74136 : Quatre portes logiques à deux entrées</p>	<p><b>Fonction compteurs</b>                      7458 : deux compteurs décimaux sur 4 bits                      7459 : deux compteurs binaires sur 4 bits                      7482 : additionneur complet de 2 bits                      7483 : additionneur complet de 4 bits                      7490 : compteur BCD                      7492 : compteur diviseur par 12                      7493 : compteur binaire de 4 bits</p>

Pour plus d'information voir : [https://www.electronique-et-informatique.fr/Electronique-et-Informatique/Digit/Data\\_book.php#FAMILLE%20TTL](https://www.electronique-et-informatique.fr/Electronique-et-Informatique/Digit/Data_book.php#FAMILLE%20TTL)