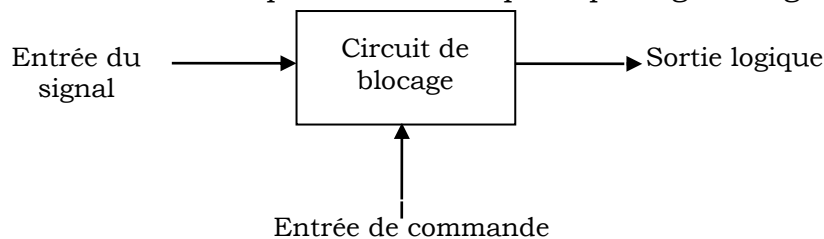


Chapitre III : Les circuits combinatoires.

1. la fonction de blocage
2. la fonction de comparaison
3. les fonctions arithmétiques (addition, soustraction)
4. les fonctions de codage, de décodage et de transcodage
5. la fonction d'aiguillage d'information (multiplexage) et démultiplexage

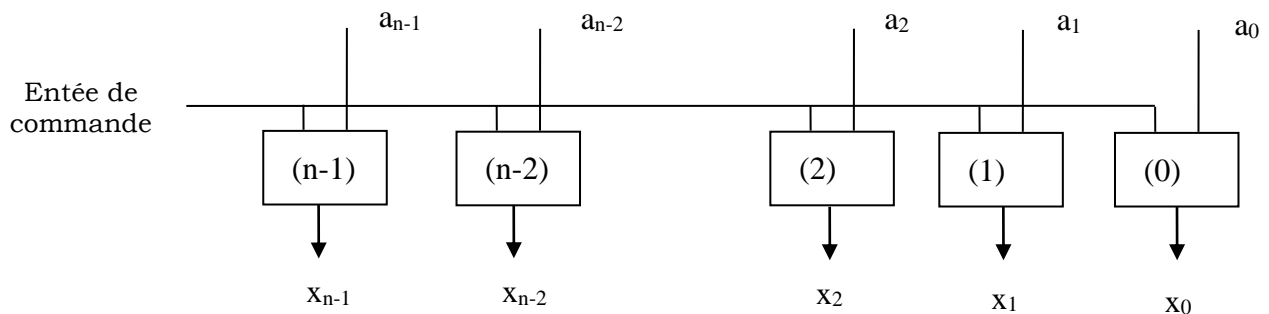
1. la fonction de blocage :

Le circuit associé à la fonction de blocage comporte à son entrée en plus d'une entrée logique, une entrée de commande qui valide ou bloque le passage du signal.



Remarque

Une information écrite sur n bits $a_{n-1}, \dots, a_2, a_1, a_0$ commandée vers la sortie $x = x_{n-1}x_{n-2} \dots x_2x_1x_0$, en utilisant la même entrée de commande pour tous les bits.



2. La fonction de comparaison :

Un comparateur de mots permet la comparaison d'un mot binaire A, avec un autre mot B de même longueur. Si les deux mots coïncident bit à bit, la sortie du comparateur est à '1'.

a/ cas d'un comparateur de mots d'un bit A_0 et B_0 :

A_0	B_0	S_0
0	0	1
0	1	0
1	0	0
1	1	1

$$S_0 = \overline{A_0 \cdot B_0} + A_0 \cdot B_0 = A_0 \oplus B_0$$

b/ Cas d'un comparateur de mots de deux bits A_1A_0 et B_1B_0 :

On compare A_1 avec B_1 et A_0 avec B_0 . On désigne par S_1 le résultat de la 1^{ière} comparaison et par S_0 celui de la seconde. La sortie du comparateur est à '1' si S_1 et S_2 sont les 2 à '1'.

$S_1 (A_1=B_1)$	$S_0 (A_0=B_0)$	S_2
0	0	0
0	1	0
1	0	0
1	1	1

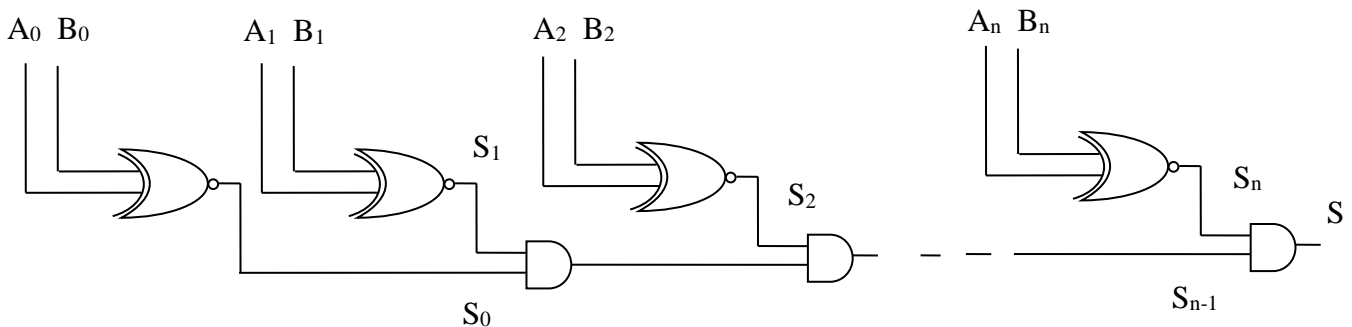
$$S_0 = S_1 \cdot S_2 = A_1 \oplus B_1 \cdot A_0 \oplus B_0$$

Cas général d'un comparateur de (n+1) bits

S_{n-1}	S_n	S
0	0	0
0	1	0
1	0	0
1	1	1

$$S = S_{n-1} \cdot S_n$$

Le logigramme d'un tel système est donc le suivant



Comparateur de mots (n+1) bits

3. Les fonctions arithmétiques :

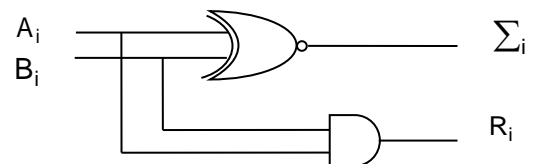
3.1 Demi-additionneur et additionneur complet :

Demi-additionneur :

Un demi-additionneur fait la somme de deux bits A_i et B_i de même poids

La table de vérité de ce circuit est la suivante :

A_i	B_i	Σ_i	R_i
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Logigramme d'un demi-additionneur

$$\Sigma_i = \bar{A}_i \cdot B_i + A_i \cdot \bar{B}_i = A_i \oplus B_i$$

$$R_i = A_i \cdot B_i$$

Additionneur complet :

L'additionneur complet fait l'addition de deux bits A_i et B_i ainsi que la retenue de l'étage précédent R_{i-1}

A_i	B_i	R_{i-1}	Σ_i	R_i
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

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

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

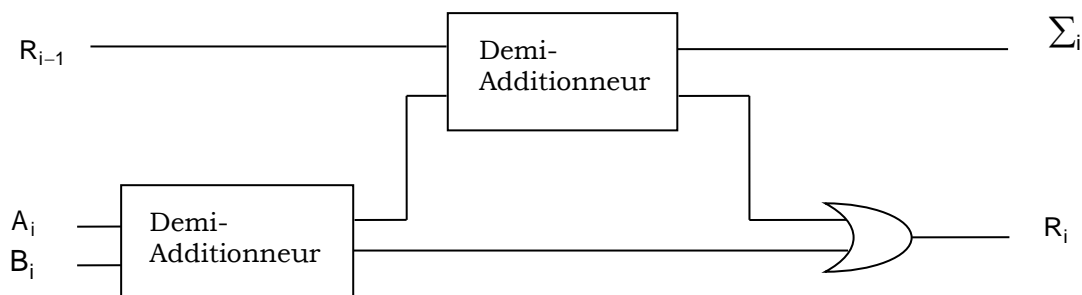
$$\Sigma_i = (\overline{A_i \oplus B_i}) R_{i-1} + (A_i \oplus B_i) \overline{R_{i-1}}$$

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

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

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

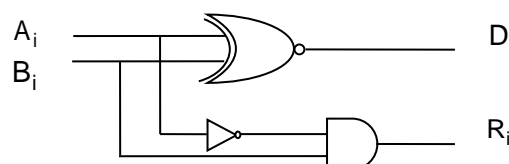
Donc, l'additionneur complet représente deux demi-additionneurs câblés en cascade.



Demi-soustracteur et soustracteur complet :

Demi- soustracteur fait la soustraction entre deux bits A_i et B_i de même poids.

A_i	B_i	D_i	R_i
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



D_i : La différence et R_i : Le rapport.

$$D_i = \overline{A_i} \cdot B_i + A_i \cdot \overline{B_i} = A_i \oplus B_i$$

$$R_i = \overline{A_i} \cdot B_i$$

Demi-soustracteur

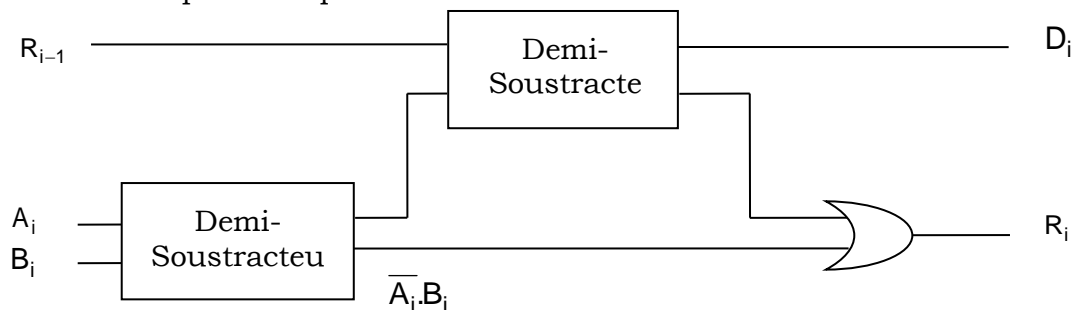
Soustracteur complet :

Le soustracteur complet fait la soustraction entre deux bits de même poids A_i , B_i et le rapport de l'étage précédent R_{i-1} .

A_i	B_i	R_{i-1}	D_i	R_i
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	0	0	0
1	1	1	1	1

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

Le soustracteur complet est équivalent à 2 demi-soustracteurs mis en cascade.



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

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

Et

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

Additionneur-soustracteur en complet à 2 :

On sait que la soustraction $A-B$ en complet à 2, revient à faire :

$$A + (\text{en complet à 2}) = A + \bar{B} + 1$$

Un même circuit peut faire alors, l'addition et la soustraction de deux nombres binaires, suivant un signal de commande x .

Si on pose que : pour $x = 0$; le circuit doit faire l'addition

Et pour $x = 1$; il doit faire la soustraction.

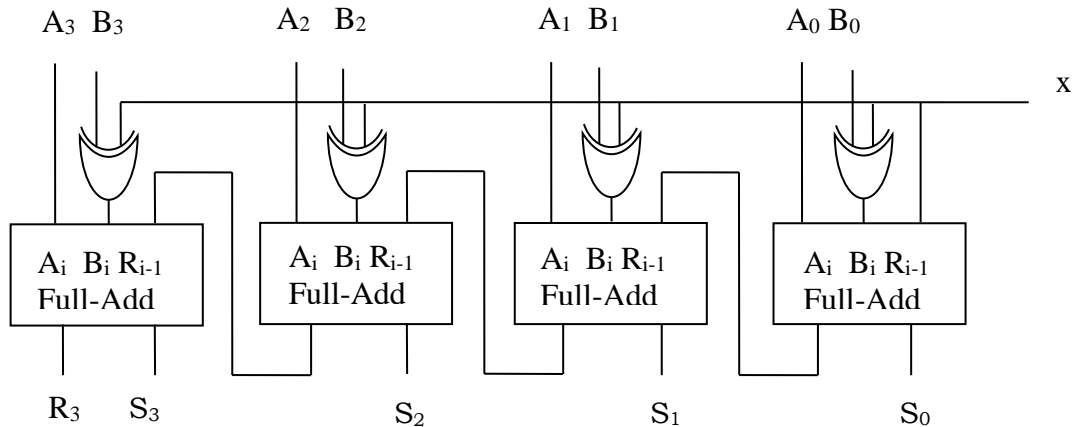
Alors les entrées du circuit réalisé avec des additionneurs, seront A et Y , avec $Y = B$ si $x = 0$, $Y =$ le complément de B si $x = 1$.

x	B_i	Y_i
0	0	0
0	1	1
1	0	1
1	1	0

$$Y_i = \bar{x} \cdot B_i + x \cdot \bar{B}_i = x_i \oplus B_i$$



L'entrée retenue, de l'additionneur de poids le plus faible, sera prise égale à x.
 L'additionneur-soustracteur complément à 2, pour 2 nombres à 4 bits A et B est:

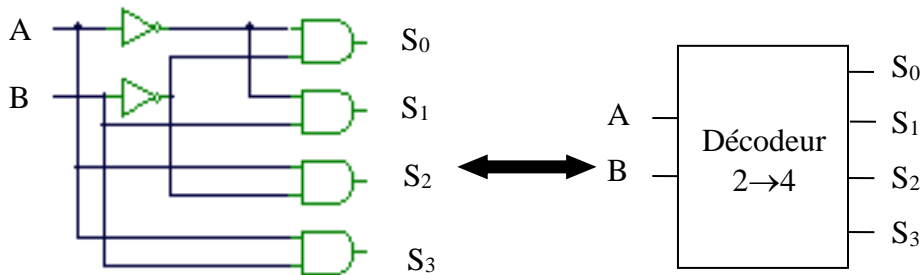


3.2. Les décodeurs et encodeurs :

Un décodeur est un circuit, qui à N entrées et 2^N sorties, dont une seule est active à la fois. En logique positive, la sortie active est à "1" et les autres sorties sont à "0" alors qu'en logique négative on a l'inverse.

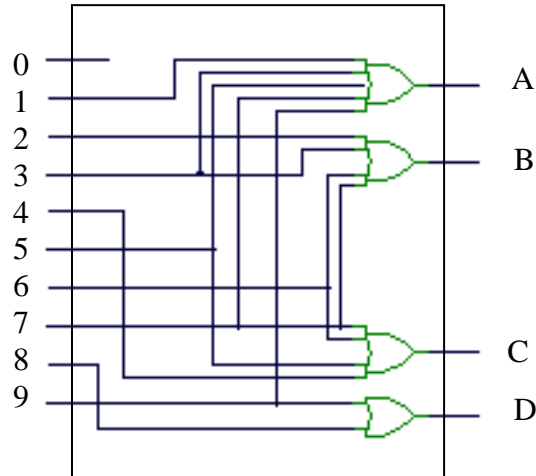
Exemple: Un décodeur $2 \rightarrow 4$, fonctionnant en logique positive.

Entrées		Sorties			
A	B	S ₀	S ₁	S ₂	S ₃
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



Le fonctionnement d'un encodeur est l'inverse de celui d'un décodeur. Il sert à générer une sortie codée en binaire ou en BCD, à partir d'une seule entrée active à la fois. Le tableau de fonctionnement et le logigramme d'un encodeur décimal-BCD sont :

Entrées décimale s	Sorties			
	D A	C	B	
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

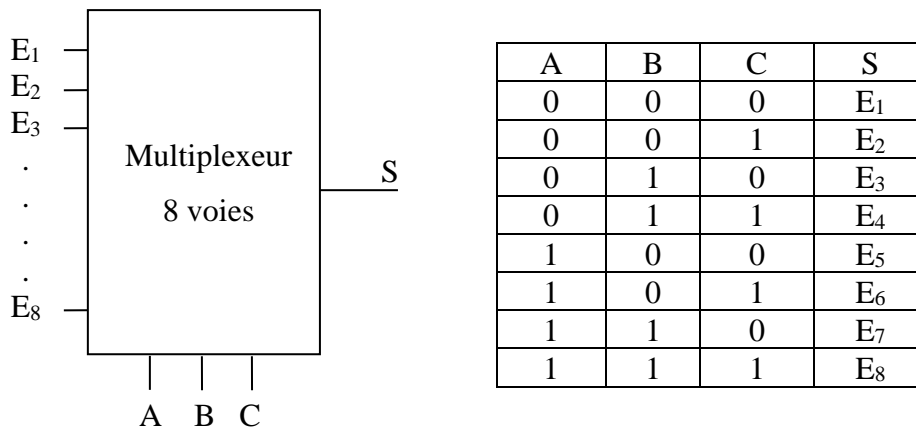


$$D=8+9 ; C=4+5+6+7 ; B=2+3+6+7 ; A=1+3+5+7+9.$$

3.3. Les multiplexeurs et démultiplexeurs :

Le multiplexage est la technique de sélection et de transmission de signaux émanant de plusieurs sources vers une seule voie. Un multiplexeur est un circuit qui a 2^N entrées dites d'information, N entrées adresses dites de commande et une seule sortie. L'état de la sortie indique la valeur de l'entrée d'information qui correspond à la valeur des entrées adresses.

Exemple: un multiplexeur à 8 entrées ($E_i, i=1\dots,8$) possède 3 lignes de commande (A, B, C) et une sortie S (Le MUX 74151). Le fonctionnement d'un tel circuit est défini par la table de vérité suivante:



$$S = \overline{A}\overline{B}\overline{C}E_1 + \overline{A}\overline{B}CE_2 + \overline{A}B\overline{C}E_3 + \overline{A}BCE_4 + A\overline{B}\overline{C}E_5 + A\overline{B}CE_6 + AB\overline{C}E_7 + ABCE_8$$

Un démultiplexeur fait la fonction inverse d'un multiplexeur. C'est un circuit qui possède une entrée E et 2^N sorties. Le transfert de l'entrée vers l'une des sorties se fait grâce à N signaux de commandes. La table de vérité suivante montre, le fonctionnement d'un démultiplexeur 4 voies.

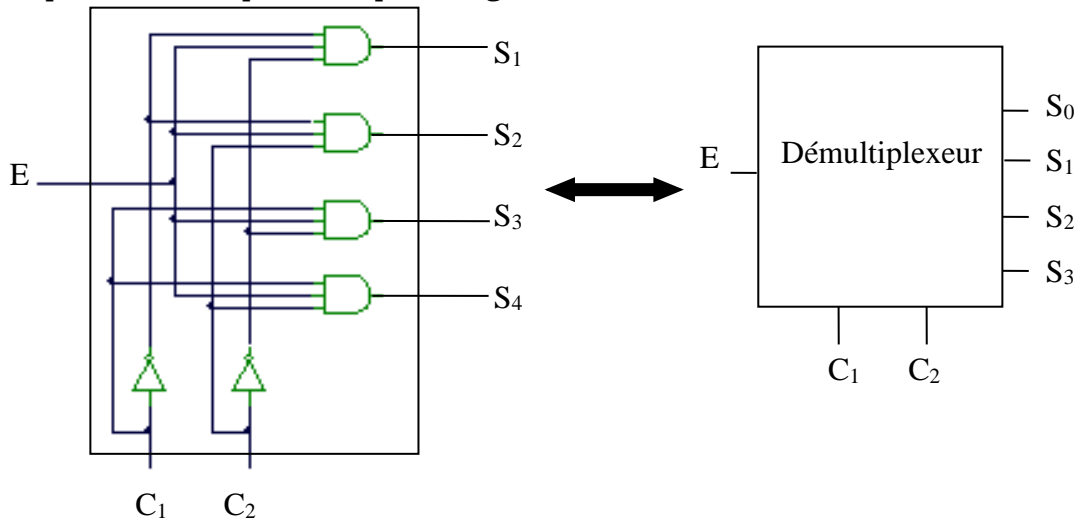
Table de vérité d'un démultiplexeur 4 voies:

C ₁	C ₂	S ₁	S ₂	S ₃	S ₄
0	0	E	0	0	0
0	1	0	E	0	0
1	0	0	0	E	0
1	1	0	0	0	E

Les expressions des sorties en fonction des entrées sont alors définies comme suit:

$$S_1 = \bar{C}_1 \cdot \bar{C}_2 \cdot E, S_2 = \bar{C}_1 \cdot C_2 \cdot E, S_3 = C_1 \cdot \bar{C}_2 \cdot E \text{ et } S_4 = C_1 \cdot C_2 \cdot E$$

Ce démultiplexeur est représenté par la figure ci-dessous:



La plus grande utilité du multiplexeur et de son inverse le démultiplexeur se trouve dans les montages de transmission d'information, où il est question de transformation parallèle-série et inversement.

3.4. Les transcodeurs ou convertisseurs de code:

Un transcodeur permet la conversion d'un mot exprimé dans un code donné sur p bits, en son équivalent dans un autre code donné sur k bits.

Le lien entre ces codes est donné par les fonctions logiques, calculées à partir de la table de vérité en utilisant les méthodes habituelles.

On peut trouver sur le marché des transcodeurs disponibles sous forme de circuits intégrés, comme par exemple le transcodeur excédant3-décimal (7442), le transcodeur décimal-excédant3 (7443) et le transcodeur excédant3-GRAY (7444).

Transcodeur Gray-Binaire:

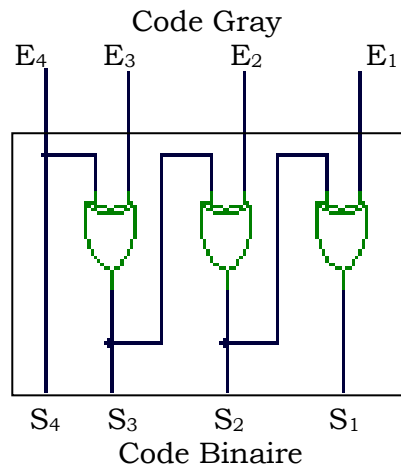
E ₄	E ₃	E ₂	E ₁	S ₄	S ₃	S ₂	S ₁
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

Après simplification on obtient:

$$S_4 = E_4, S_3 = E_3 \oplus E_4, S_2 = E_2 \oplus E_3 \oplus E_4$$

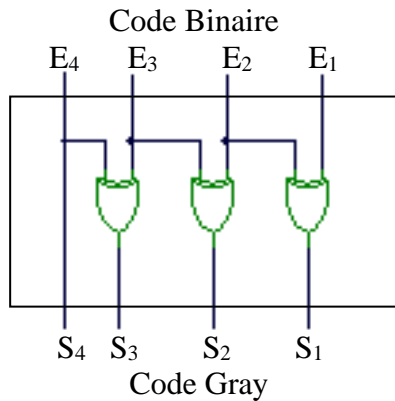
$$\text{et } S_1 = E_1 \oplus E_2 \oplus E_3 \oplus E_4.$$

Ce qui donne le logigramme suivant:



Transcodeur Binaire-Gray:

De la même façon, en inversant les entrées/sorties on obtient le schéma suivant:



La table de vérité donne les fonctions logiques de sorties suivantes:

$$S_4 = E_4, S_3 = E_3 \oplus E_4, S_2 = E_2 \oplus E_3, S_1 = E_1 \oplus E_2$$

Transcodeur Gray-Binaire/Binaire-Gray:

La conversion Gray-Binaire ou Binaire-Gray peut être facilement faite par un seul circuit commandé par un signal désignant le sens de conversion.

Ce circuit représente une combinaison des deux logigrammes obtenus précédemment avec des multiplexeurs, comme indiqué ici:

