Suite chapitre II : circuit logiques combinatoires

2.8 Additionneur et soustracteur

2.8.1 Demi-additionneur

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

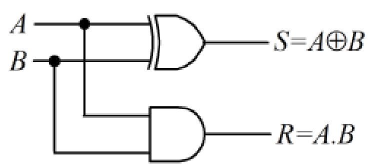
A (Cumulande)	B (Cumulateur)	S (Somme)	R (Retenue)	Mintermes
0	0	0	0	
0	1	1	0	$\overline{A} \cdot B$
1	0	1	0	$A \cdot \overline{B}$
1	1	0	1	$A \cdot B$

Les équations caractéristiques sont :

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

$$R = A \cdot B$$

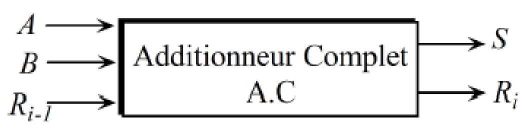
Le logigramme correspondant :



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

2.8.2 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_{i-1}	S	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

Les équations logiques des sorties S et R_i basées sur les mintermes :

$$S = \overline{A} \cdot \overline{B} \cdot R_{i-1} + \overline{A} \cdot B \cdot \overline{R_{i-1}} + A \cdot \overline{B} \cdot \overline{R_{i-1}} + A \cdot B \cdot R_{i-1}$$

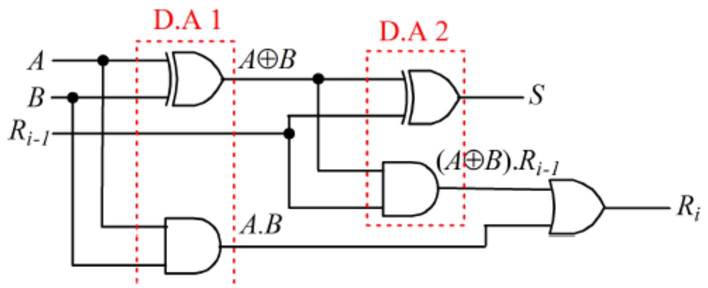
$$R_i = \overline{A} \cdot B \cdot R_{i-1} + A \cdot \overline{B} \cdot R_{i-1} + A \cdot B \cdot \overline{R_{i-1}} + A \cdot B \cdot R_{i-1}$$

Simplification de ces dernières :

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

$$\begin{aligned} R_i &= \overline{A} \cdot B \cdot R_{i-1} + A \cdot \overline{B} \cdot R_{i-1} + A \cdot B \cdot \overline{R_{i-1}} + A \cdot B \cdot R_{i-1} \\ &= (\overline{A} \cdot B + A \cdot \overline{B}) \cdot R_{i-1} + A \cdot B \cdot (R_{i-1} + \overline{R_{i-1}}) \\ \Rightarrow R_i &= (A \oplus B) \cdot R_{i-1} + A \cdot B \end{aligned}$$

Logigramme de l'additionneur complet :



2.8.3 Demi-soustracteur

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

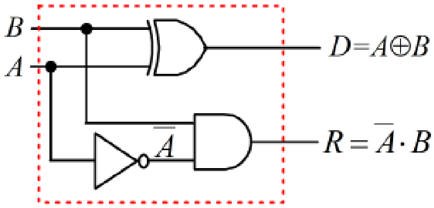
A	B	D (Différence)	R (Retenue)	Mintermes
0	0	0	0	
0	1	1	1	$\overline{A} \cdot B$
1	0	1	0	$A \cdot \overline{B}$
1	1	0	0	

Les équations logiques sont :

$$D = \overline{A} \cdot B + A \cdot \overline{B} = A \oplus B$$

$$R = \overline{A} \cdot B$$

Le logigramme du demi-soustracteur :



Demi-soustracteur

2.8.4 Soustracteur complet

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

A	B	R_{i-1}	D	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

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

$$D = \overline{A} \cdot \overline{B} \cdot R_{i-1} + \overline{A} \cdot B \cdot \overline{R_{i-1}} + A \cdot \overline{B} \cdot \overline{R_{i-1}} + A \cdot B \cdot R_{i-1}$$

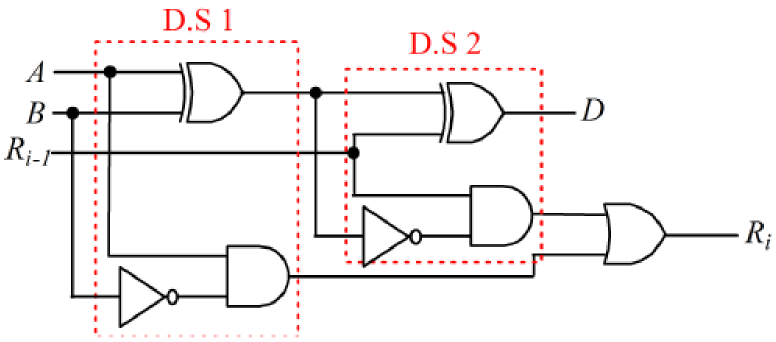
$$R_i = \overline{A} \cdot \overline{B} \cdot R_{i-1} + \overline{A} \cdot B \cdot \overline{R_{i-1}} + \overline{A} \cdot B \cdot R_{i-1} + A \cdot B \cdot R_{i-1}$$

Simplification de équations précédentes:

$$\begin{aligned}
D &= \overline{A} \cdot \overline{B} \cdot R_{i-1} + \overline{A} \cdot B \cdot \overline{R_{i-1}} + A \cdot \overline{B} \cdot \overline{R_{i-1}} + A \cdot B \cdot R_{i-1} \\
&= (\overline{A} \cdot \overline{B} + A \cdot B) \cdot R_{i-1} + (\overline{A} \cdot B + A \cdot \overline{B}) \cdot \overline{R_{i-1}} \\
&= (\overline{A \oplus B}) \cdot R_{i-1} + (A \oplus B) \cdot \overline{R_{i-1}} \\
\Rightarrow D &= A \oplus B \oplus R_{i-1}
\end{aligned}$$

$$\begin{aligned}
R_i &= \overline{A} \cdot \overline{B} \cdot R_{i-1} + \overline{A} \cdot B \cdot \overline{R_{i-1}} + \overline{A} \cdot B \cdot R_{i-1} + A \cdot B \cdot R_{i-1} \\
&= (\overline{A} \cdot \overline{B} + A \cdot B) \cdot R_{i-1} + \overline{A} \cdot B \cdot (R_{i-1} + \overline{R_{i-1}}) \\
\Rightarrow R_i &= (\overline{A \oplus B}) \cdot R_{i-1} + \overline{A} \cdot B
\end{aligned}$$

eur



3- Transcodeur, codeur et décodeur

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

Exemple 2.13 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 e représentent le même nombre en code Gray.

Analyse par la table de vérité :

N° Décimal	$A\ B\ C\ D$	$X\ Y\ Z\ T$
0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1
2	0 0 1 0	0 0 1 1
3	0 0 1 1	0 0 1 0
4	0 1 0 0	0 1 1 0
5	0 1 0 1	0 1 1 1
6	0 1 1 0	0 1 0 1
7	0 1 1 1	0 1 0 0
8	1 0 0 0	1 1 0 0
9	1 0 0 1	1 1 0 1
10	1 0 1 0	1 1 1 1
11	1 0 1 1	1 1 1 0
12	1 1 0 0	1 0 1 0
13	1 1 0 1	1 0 1 1
14	1 1 1 0	1 0 0 1
15	1 1 1 1	1 0 0 0

1. $X = f(A,B,C,D)$?

AB \ CD	0 0	0 1	1 1	1 0
0 0	0	0	0	0
0 1	0	0	0	0
1 1	1	1	1	1
1 0	1	1	1	1

A

$$X = A$$

2. $Y = f(A,B,C,D)$?

$$Y = G_1 + G_2 = \overline{A} \cdot B + A \cdot \overline{B} = A \oplus B$$

AB \ CD	0 0	0 1	1 1	1 0
0 0	0	0	0	0
0 1	1	1	1	1
11	0	0	0	0
1 0	1	1	1	1

3. $Z = f(A,B,C,D)$?

AB \ CD	0 0	0 1	1 1	1 0
0 0	0	0	1	1
0 1	1	1	0	0
11	1	1	0	0
1 0	0	0	1	1

G_1 G_2

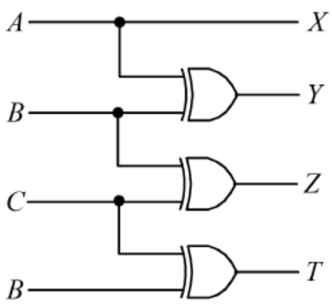
$$Z = G_1 + G_2 = B \cdot \overline{C} + \overline{B} \cdot C = B \oplus C$$

4. $T = f(A,B,C,D)$?

AB \ CD	0 0	0 1	1 1	1 0
0 0	0	1	0	1
0 1	0	1	0	1
11	0	1	0	1
1 0	0	1	0	1

G_1 G_2

$$Z = G_1 + G_2 = \overline{C} \cdot D + C \cdot \overline{D} = C \oplus D$$



logigramme correspondant

3.2 Codeur

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

Exemple 2.14 Codeur décimal - BCD

Il permet de traduire un nombre écrit en décimal, en son équivalent binaire.

La table de vérité est la suivante :

Décimal	BCD			
N°	B_3	B_2	B_1	B_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

Les expressions logiques sont :

$$B_3 = 8 + 9$$

$$B_2 = 4 + 5 + 6 + 7$$

$$B_1 = 2 + 3 + 6 + 7$$

$$B_0 = 1 + 3 + 5 + 7 + 9$$

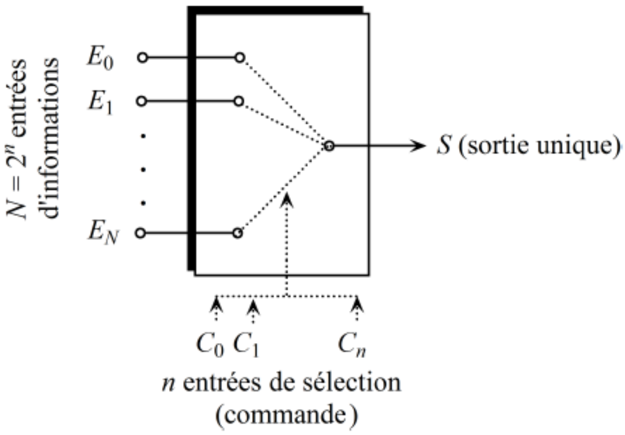
3.3.Décodeur

Un décodeur est un circuit logique qui réalise la fonction inverse du codeur.

Exemple : Décodeur BCD-décimal.

3.4 .Multiplexeur

C'est un circuit combinatoire permettant de réaliser un aiguillage de l'une des entrées en une sortie unique, dont le 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 : MUX 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	

Donc la table de vérité précédente peut être simplifiée :

C_0	S
0	E_0
1	E_1

Par suite, l'équation caractéristique est :

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

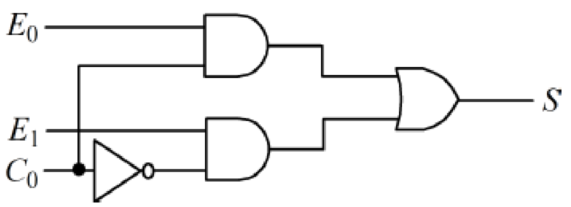
En utilisant le tableau de Karnaugh :

C_0 \ $E_1 E_0$	0 0	0 1	1 1	1 0
0	0	1	1	0
1	0	0	1	1

G_1 G_2

$$S = G_1 + G_2 = \overline{C_0} \cdot E_0 + C_0 \cdot E_1$$

Le logigramme du multiplexeur 2 vers 1 :



Réf :

Logique combinatoire et séquetielle Dr Inel Fouad

Coura élctronique numérique Myriam siadar