

Chapitre III : Systèmes logiques itératifs

I- Introduction

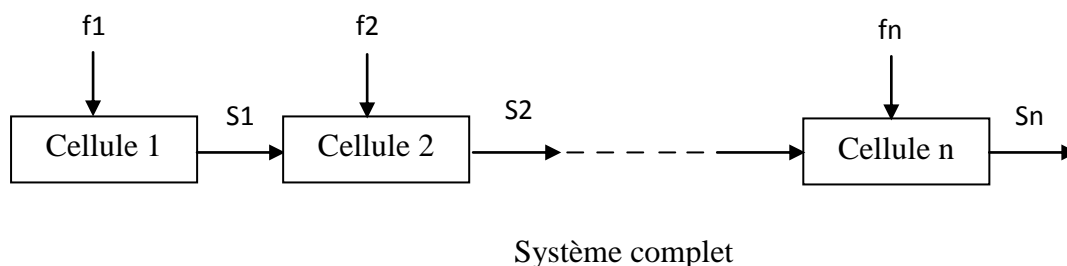
La simplification des fonctions logiques par la méthode du tableau de Karnaugh devient difficile pour un nombre de variables plus de 6. Or dans les systèmes les nombres sont généralement représentés sur 8, 16 ou 32 bits et même plus. La conception des systèmes combinatoires réalisant l'addition, la soustraction ou la comparaison de tels nombres donne des fonctions logiques de 16, 32 ou 64 variables.

On dépose alors le système en plusieurs sous systèmes et chacun peut être étudié par les méthodes (de simplification) habituelles.

II-Définition

Un système logique itératif est un système combinatoire décomposé en plusieurs sous-systèmes identiques entre eux.

Le sous système est appelé cellule.



III- Demi-additionneur et additionneur complet

II-1 Demi-additionneur

Un demi-additionneur fait la somme de deux bits a, b de même poids. La table de vérité de ce circuit est la suivante :

a	B	Si	Ri
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Si représente la somme et Ri représente la retenue.

$$Si = \bar{a}b + a\bar{b} = a \oplus b$$

$$Ri = a \cdot b$$

III-2 Additionneur complet

L'additionneur complet fait l'addition de deux bits a et b ainsi que la retenue précédente.

La table de vérité est comme suit :

a	b	Ri-1	Si	Ri
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

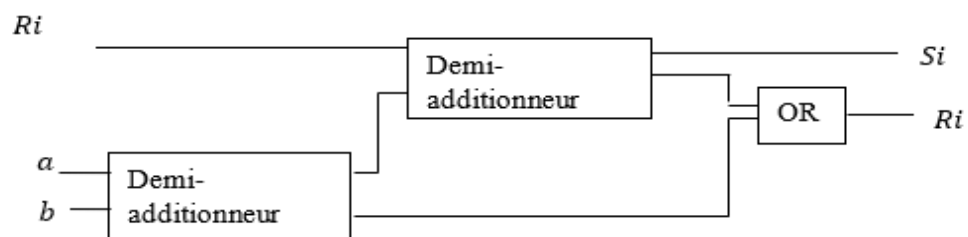
$$Si = \overline{a}b\overline{Ri-1} + abRi-1 + \overline{a}\overline{b}Ri-1 + a\overline{b}\overline{Ri-1}$$

$$Si = Ri-1 \oplus a \oplus b$$

$$Ri = \overline{a}\overline{b}\overline{Ri-1} + abRi-1 + \overline{a}bRi-1 + a\overline{b}Ri-1$$

$$Ri = ab + Ri-1(a \oplus b)$$

L'additionneur complet représente donc deux demi-additionneurs montés en cascade.



IV- Demi-soustracteur et soustracteur complet

IV-1 Demi-soustracteur

Un demi-soustracteur fait la soustraction de deux bits a et b de même poids. La table de vérité de ce circuit est la suivante :

a	b	Di	Ri
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Di représente le résultat de la soustraction et Ri le report.

$$Di = \bar{a}b + a\bar{b} = a \oplus b$$

$$Ri = \bar{a}b$$

IV-2 Soustracteur complet

Le soustracteur complet fait la soustraction entre deux bits a et b ainsi que le report de l'étage précédent.

a	b	Ri-1	Di	Ri
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

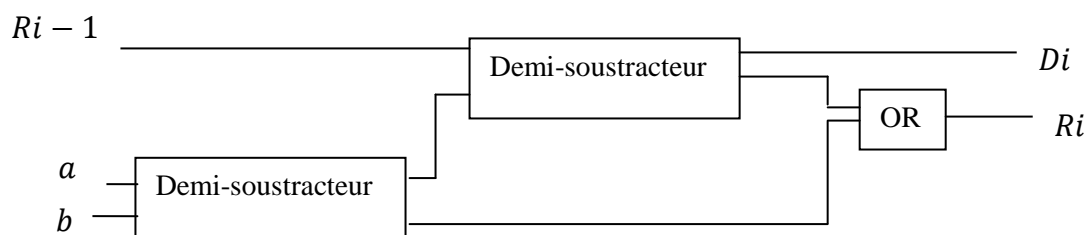
$$Di = \bar{a}\bar{b}Ri - 1 + \bar{a}b\overline{Ri - 1} + a\bar{b}\overline{Ri - 1} + abRi - 1$$

$$Di = Ri - 1 \oplus a \oplus b$$

$$Ri = \bar{a}\bar{b}Ri - 1 + \bar{a}b\overline{Ri - 1} + \bar{a}bRi - 1 + abRi - 1$$

$$Ri = \bar{a}b + Ri - 1(\overline{a \oplus b})$$

Comme pour l'additionneur complet, le soustracteur complet est l'équivalent de deux demi-soustracteurs mis en cascade.



V- Additionneur-soustracteur en complément à deux.

La soustraction de deux nombres en complément à deux revient à faire la somme :

$$a + \text{complément à 2 de } b = a + \bar{b} + 1$$

Un même circuit peut faire l'addition et la soustraction de deux nombres binaires a et b suivant un signal de commande X .

- On pose pour $X=1$ le circuit fait l'addition.
- Pour $X=0$ le circuit fait une soustraction.

VI- Circuit comparateur

a- Pour faire la comparaison entre deux bits a et b on considère les trois cas :

- Sup : dans le cas où $a > b$
- Inf : dans le cas où $a < b$
- Égal : dans le cas où $a = b$

La table de vérité du circuit est donc :

<i>a</i>	<i>b</i>	<i>sup</i>	<i>inf</i>	<i>egal</i>
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

$$sup = a\bar{b}$$

$$inf = \bar{a}b$$

$$egal = \bar{a}\bar{b} + ab = \overline{a \oplus b}$$

b- Pour réaliser un comparateur de mots de deux bits chacun a_0a_1 et b_0b_1 on doit comparer a_0 avec b_0 et a_1 avec b_1 . Si les deux mots coïncident la sortie du comparateur S2 vaut 1.

S1(a1=b1)	S0(a0=b0)	S2
0	0	0
0	1	0
1	0	0
1	1	1

- S1 sortie du comparateur des bits a_1 et b_1 . Donc $s_1=1$ si $a_1=b_1$.
- S0 sortie du comparateur des bits a_0 et b_0 . Donc $s_0=1$ si $a_0=b_0$.
- S2 représente la sortie du comparateur. $S_2=1$ si $s_1=1$ et $s_0=1$

$$s_1 = \overline{a_1 \oplus b_1}$$

$$s_0 = \overline{a_0 \oplus b_0}$$

$$s_2 = (\overline{a_1 \oplus b_1}).(\overline{a_0 \oplus b_0})$$

Dans le cas général où on veut comparer $n+1$ bits la sortie du système est :

$$s = sn - 1.sn$$

VII- Circuit contrôleur de parité

La sortie d'un circuit de contrôle de parité impaire est égale à 1 si le mot binaire contient un nombre impair de « 1 » et égale à 0 si le mot contient un nombre pair de « 1 ».

a- Cas de deux bits :

Soit A un nombre binaire constitué de deux bits tel que $A=a_0a_1$. Pour une parité impaire nous avons la table de vérité suivante :

a0	a1	S1
0	0	0
0	1	1
1	0	1
1	1	0

$$s1 = \overline{a_0}a_1 + a_0\overline{a_1} = a_0 \oplus a_1$$

b- Cas de trois bits

Pour un nombre codé sur trois bits, on prend le troisième bit ainsi que la sortie (s1) pour contrôler la parité des deux premiers bits.

S1	a2	S2
0	0	0
0	1	1
1	0	1
1	1	0

$$s2 = \overline{s1}a_2 + s1\overline{a_2} = a_2 \oplus s1$$

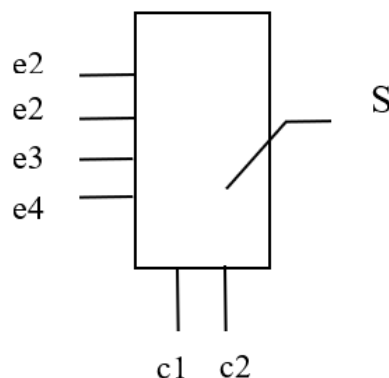
- Si on veut maintenant contrôler la parité d'un nombre codés sur plus de trois bits on suit le même raisonnement.
- Dans le cas d'une parité paire nous avons $s=1$ si le mot contient un nombre pair de « 1 » et égale à 0 dans le cas contraire.

VIII- Circuit multiplexeur

Un multiplexeur permet de sélectionner une donnée (entrée) parmi plusieurs. Chaque donnée est spécifique par une adresse ou variable de sélection. La donnée sélectionnée est conduite à l'unique sortie.

Un multiplexeur qui contient n variables de sélection (ou entrées de commande) possède 2^n variable d'entrées. Par exemple pour un multiplexeur à deux variables de sélection ($c1$ et $c2$) on trouve 4 entrées, un multiplexeur à trois variables de sélection on trouve 8 entrées...etc.

Exemple d'un multiplexeur à deux entrées de sélection $c1$ et $c2$:



☞ Exemple

Réaliser la fonction suivante à l'aide d'un multiplexeur à trois entrées de sélections.

$$f = \sum(0,1,2,6,7)$$

La sortie du multiplexeur est déterminée par les états des sélecteurs a, b, c . Quand $abc = 000, 001, 010, 110, 111$ la sortie f vaut 1. Pour tous les autres états la sortie est égale à 0.

$$f = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}b\bar{c} + ab\bar{c} + abc$$

$$f = (\bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c) + (\bar{a}b\bar{c} + ab\bar{c}) + (abc + \textcolor{red}{ab}\bar{c})$$

$$f_{min1} = \bar{a}\bar{b} + b\bar{c} + ab$$

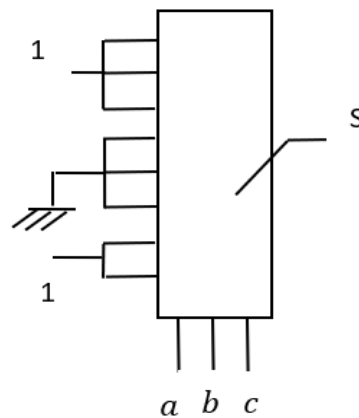
Ou bien :

$$f = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}b\bar{c} + ab\bar{c} + abc$$

$$f = (\bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c) + (\bar{a}b\bar{c} + \textcolor{red}{a}\bar{b}\bar{c}) + (ab\bar{c} + abc)$$

$$f_{min2} = \bar{a}\bar{b} + \bar{a}\bar{c} + ab$$

La représentation de la fonction sur un multiplexeur à trois entrées de sélection est telle que :



IX- Circuit démultiplexeur

Une sortie est activée parmi plusieurs celle correspondante à l'adresse. Les autres sorties sont forcément désactivées.

X- Circuit transcodeur

La fonction d'un transcodeur est la conversion d'un mot exprimé dans un code A en son équivalent exprimé dans un code B. on parle alors d'un transcodeur A/B.

Pour réaliser le transcodeur il faut trouver le circuit combinatoire approprié, c'est-à-dire trouver les équations de sorties (du code B) en fonction des variables d'entrées du code (A).

XI- Mise en équation d'un système combinatoire

Prenons un exemple pour cette section :

☞ Exemple

Dans un centre culturel, considérer les expressions suivantes :

- Rachid ne joue jamais aux échecs.
- Karim joue si et seulement si ses supporteurs Rachid et Ali sont présent.
- Salah joue sans conditions, s'il est seul il joue contre l'ordinateur du centre.
- Ali joue si et seulement si son porte malheur Rachid n'est pas présent.

Si R, K, A, S désignent les présences de Rachid, Karim, Ali et Salah respectivement déterminer la fonction F définie par personne des quatre ne joue aux échecs dans le centre.

Exprimer cette fonction sous forme littéraire d'une phrase.

XI-1 Mise en équation directe :

Prenons \bar{f} qui est le complément de f qui exprime le fait qu'il ya au moins quelqu'un qui joue au centre.

$$\bar{f} = S + KRA + A\bar{R} = S + KRA + A\bar{R}(k + \bar{k}) + A\bar{R}K$$

$$\bar{f} = S + KA(R + \bar{R}) + A\bar{R} = S + KA + A\bar{R}$$

$$f = \overline{S + KA + A\bar{R}} = \bar{S} \cdot \overline{(KA + A\bar{R})}$$

$$f = \bar{S} \cdot \overline{A(K + \bar{R})} = \bar{S} \cdot (\bar{A} + \overline{(K + \bar{R})})$$

$$f = \bar{S} \cdot (\bar{A} + \bar{K}R)$$

Nous pouvons donc dire que personne ne joue au centre lorsque Salah est absent et que soit Ali est absent soit Rachid est présent avec Karim absent.

XI-2 Mise en équation à partir de la table de vérité :

On écrit toutes les combinaisons d'entrées possibles et on prend comme monômes de la fonction recherchée la somme de tous les monômes qui réalise la fonction.

R	K	A	S	f
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

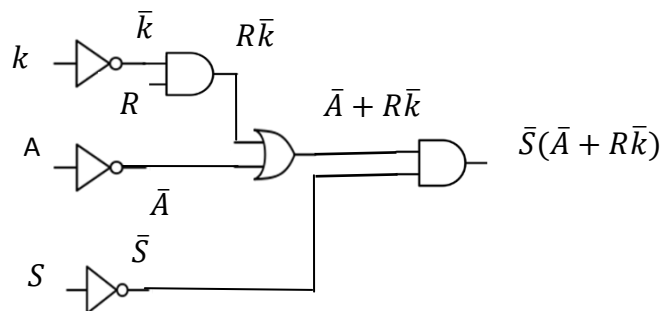
La sortie f est donc :

$$\begin{aligned}
 f &= \bar{R}\bar{k}\bar{A}\bar{S} + \bar{R}K\bar{A}\bar{S} + R\bar{k}\bar{A}\bar{S} + R\bar{k}A\bar{S} + RK\bar{A}\bar{S} \\
 &= \bar{R}\bar{A}\bar{S} + K\bar{A}\bar{S} + R\bar{k}\bar{S} + \bar{k}\bar{A}\bar{S} \\
 &= \bar{A}\bar{S} + \bar{A}\bar{S}\bar{R} + R\bar{k}\bar{S} = \bar{A}\bar{S} + R\bar{k}\bar{S} \\
 f &= \bar{S}(\bar{A} + R\bar{k})
 \end{aligned}$$

XI-3 Mise en équation à partir d'un logigramme :

La méthode consiste à écrire l'équation logique à la sortie de chaque opérateur logique.

Donc : $f = \bar{S}(\bar{A} + R\bar{k})$ la représentation de la fonction est :



⊗ Exercice

Réaliser la fonction f à l'aide d'un multiplexeur trois entrées de sélection AKR.