**5/1/21- EX1**

1. Toutes les va A,B,C,et D sont=1 :Proposition

A convertir en expression logic ?(A=1 et B=1 et C=1 et D=1)

F1= ?

F1=A.B.C.D

Le complément est :

F1’=( A.B.C.D)’

Thm de De Morgan donne :

F1’=( A)’+( B.)’+ (C.)’+ ( D)’

- Toutes les va A,B,C,et D sont=0 :Proposition ?(A=0 et B=0 et C=0 et D=0)

A convertir en expression logic ?

F2= ?

F2=(A+B+C+D)’==( A)’.( B.)’. (C.)’. ( D)’

Thm de De Morgan :Le complément est :

X’’=x ; F2’= A+B+C+D

1. Simplifier 1 express logic par voie de l’Alg. Boole

(1ère méthode)

* F1=(A+B’).C’+A+B’.C

=A.C’+ B’.C’+A+B’.C

=A.(1+C’)+ B’.C’+A+B’.C

= A.(1+C’)+ B’.(C’ +C)

f1=A+B’

-f2=A.C+B.C’+A.B

Concensus doit etre éliminé !

f2=A.C+B.C’+A.B.(C’ +C)

= A.C+B.C’+A.B.C’ + A.B.C

= C.(A+A.B) + C’(B+A.B)

= C.A(1+B) + C’B(1+A)

F2=AC+BC’

1. Calcul des compléments de :

Y=(A+B) (A’+B’)

Y’ =?

Pensez à de De morgan :

Y’ = [(A+B) .(A’+B’)]’

= (A+B)’+(A’+B’)’

=A’.B’+A.B =AxnorB

EX2

Générateur de parité

1. Table de vérité :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| D | C | B | A | P |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 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 | 1 |

1. LOGIGRAMME QUI RéALISE LA FONCTION P ?

**Tabeaux de Karnaugh** : Variante des tables de vérité ;

Organisation en forme de grille bidimensionnelle (Damier) ;

Les cellules voisines, ou adjacentes ( qui ont 1 côté en commun) ne diffèrent

que **d’un bit** dans leurs codes d’entrée **(Code Gray) ;**

 Un **regroupement de cellules voisines**

**ou termes mineurs, est appelé Impliquant** ; il associe 2*m* (1, 2, 4, 8, . . .) minterms ;

**la variable d’entrée qui change, disparaît** **(X+X’=1).** 2 impliquants voisins s’associent pour former un nouveau impliquant

comprenant 2 fois plus de termes mineurs.

**p.ex1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Y Z**  **X** | 0**0** | 0**1** | **1**1 | **1**0 |
| 0 | 0 | 0 | **1** | **1** |
| 1 | 0 | 0 | **1** | **1** |

**F1=Y(Meme res avec les maxterms)**

**p.ex2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Z T**  **XY** | 00 | **0**1 | 11 | 10 |
| 00 | 0 | **1** | 0 | 0 |
| 01 | 0 | **1** | 0 | 0 |
| 11 | 0 | **1** | **1** | **1** |
| 10 | 0 | **1** | **1** | **1** |

**F2=Z’T + ( XT )+ XZ**

**p.ex3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Z **T**   XY | 0**0** | 01 | 11 | 1**0** |
| 00 | **1** | 0 | 0 | **1** |
| 01 | **1** | **1** | 0 | **1** |
| 11 | **1** | **1** | 0 | **1** |
| 10 | **1** | 0 | **1** | **1** |

**X’Y’Z’T’+ XY’Z’T’**

**F3=T’ + YZ’ + XY’Z**

p.ex4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Y Z  X | 00 | 01 | 11 | 10 |
| 0 | 0 | 0 | **1** | **1** |
| 1 | 0 | **1** | 0 | **1** |

**F4=XY’Z + X’Y + YZ’**

p.ex5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Y Z  X | 00 | 01 | 11 | 10 |
| 0 | 0 | **1** | **1** | **1** |
| 1 | **1** | **1** | 0 | **1** |

**F5=XY’ + X’Z + YZ’**

**F5=XZ’ + Y’Z + YX’**

**(équivalentes)**

p.ex5 (2ème version)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Y Z  X | 0**0** | 01 | 11 | 1**0** |
| 0 | 0 | **1** | **1** | **1** |
| 1 | **1** | **1** | 0 | **1** |

**F5=XZ’ + Y’Z + YX’**

p.ex6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Z T  XY | 00 | 01 | 11 | 10 |
| **0**0 | 0 | 0 | **Փ=0** | **Փ=0** |
| **0**1 | 0 | 0 | 0 | 0 |
| **1**1 | **1** | **1** | **1** | **1** |
| **1**0 | **1** | **1** | **Փ=1** | **Փ=1** |

**(don’t happen or care : X)**

**F6=X**

**p.ex7**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Z T  XY | 00 | 01 | 11 | 10 |
| 00 | **1** | 0 | **Փ=0** | **Փ=1** |
| 01 | **1** | 0 | 0 | **1** |
| 11 | **0** | **1** | **1** | **0** |
| 10 | **0** | **1** | **Փ=1** | **Փ=0** |

**X’T’+XT=porte xnor**

**F7=X** *xnor* **T**

**Opération inverse :**

**Expression suivie de TK**

**Ex3**

**1.**

**Y1= (A+B )A’ + A’B’**

**TK**

**2va ce qui implique 2puiss2=4 cases**

|  |  |  |
| --- | --- | --- |
| **A B** | **0** | **1** |
| **0** | **1** | **1** |
| **1** | **0** | **0** |

**Y1= A’**

**Y1= A’B’+A’C’+BC’**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BC  A | 0**0** | 01 | 11 | 1**0** |
| 0 | 1 | **1** | **1** | **1** |
| 1 | **0** | **0** | 0 | **1** |

**F5=BC’ + A’**

**F5=CA’ + A’C’+ABC’**

**(Nécessite simplification)**

**Multiplexeurs et génération de fonctions combinatoires**

**Produit : généré avec mux2v1**

**F1=e1.e2**

**T.V :**

|  |  |  |
| --- | --- | --- |
| **e1=sel** | **e2** | **F1** |
|  0 0 | 0 |  0 0 |
| 0 | 1 | 0 |
|  1 1 | 0 | 0 |
| **1** | **1** | 1 |

Circuit est :

 **GND=0**

 **e2** F**1=0 ; e2**

 **e1=0**

 **=1**

**Imparité : générée avec mux2v1**

**F1=e1 *xor* e2**

**T.V :**

|  |  |  |
| --- | --- | --- |
| **e1** | **e2** | **F2** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Circuit est :

 **e2**

 **e2’** F**2**

 **e1**

**F3= ∑ m(0,1,4,7) : générée avec mux4v1.**

**T.V :**

|  |  |  |  |
| --- | --- | --- | --- |
| **e1** | **e2** | **e3** | **F3** |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

**1**

**0**

**e3’**

**e3**

D’où le Circuit :

 **1**

 0 F**3**

 **e3’**

 **e3**

 **e1 e2**

code gray ?

00***0***

00***1***

**01*1***

**01*0***

**110**

**111**

**101**

**100**

Multiplexeur :

Mission : **Acheminer au choix 1 entrée parmi +s vers la sortie.**

Utilise des signaux de contrôle :sel,S, c…

**Ex1 : Mux 2(entrées)vers1(sortie)**

**1 seul signal de contrôle : C**

Circuit symbole est :

 **e1**

 **e2** F

 **C**

**TV :**

|  |  |
| --- | --- |
| Sel | F |
| 0 | **e1** |
| 1 | **e2** |

Express de F ?

**F=SEl’.e1+SEl.e2**

**(2 AND, 1 OR,1 Not)**

**Décodeur : mission inverse de multiplexeur sauf que e=1**

**Ds ce cas, 1sortie parmi +s qui sera active, les autres sont mises à 0.**

**P.ex : Decodeur 2(signaux de control) fois4(2 puiss2) sorties**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **C1** | **C0** | **S3** | **S2** | **S1** | **S0** |
| **0** | **0** | **0** | **0** | **0** | **1** |
| **0** | **1** | **0** | **0** | **1** | **0** |
| **1** | **0** | **0** | **1** | **0** | **0** |
| **1** | **1** | **1** | **0** | **0** | **0** |

**Chercher logigramme.**

**Générer également des fonctions àpartir de TV**

**P.ex : F=π M(1,6)=∑m(0,2,3,4,5,7)**

**(Voir cours)**