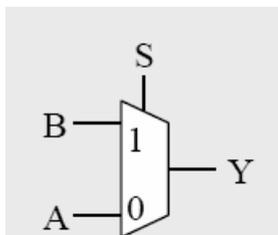


### 1- Exercice 1 : (Centrale)

La fonction Multiplexeur 2 vers 1, dont le symbole logique est représenté Figure 1, correspond à la définition donnée Figure 2.



- Figure 1 -

### Définition

Si le signal de contrôle  $S$  vaut 0  
alors le multiplexeur transmet le  
signal d'entrée  $A$  vers la sortie  $Y$ ,  
sinon,  
le multiplexeur transmet le  
signal d'entrée  $B$  vers la sortie  $Y$ .

- Figure 2 -

1- Après avoir établi la Table de Vérité puis le tableau de KARNAUGH de cette fonction Multiplexeur, donner l'expression de la sortie  $Y$  en fonction des entrées  $A$  et  $B$  et du signal de sélection  $S$ .

- en effectuant une synthèse sur les 1, ( expression sous la forme d'une somme de produits),
- en effectuant une synthèse sur les 0, ( expression sous la forme d'un produit de sommes),
- vérifier l'égalité des 2 expressions.

2- Etablir le schéma logique du multiplexeur 2 vers 1 en utilisant successivement.

- des portes AND, OR et NOT,
- des portes NOR,
- des portes NAND.



### 2- Exercice 2 : (centrale)

#### 1- Etude logique de l'addition arithmétique

1-1- Compléter le Tableau 1 avec le résultat, codé par les bits  $C_0$  et  $S$ , de l'addition arithmétique des 3 bits  $A$ ,  $B$  et  $C_i$ .

1-2- Etablir les équations logiques complètes de  $S$  et de  $C_0$  En fonction de  $A$ ,  $B$  et  $C_i$ .

1-3- L'équation logique de  $S$  peut-elle être réduite ?

1-4- L'équation logique de  $C_0$  peut-elle être réduite ?

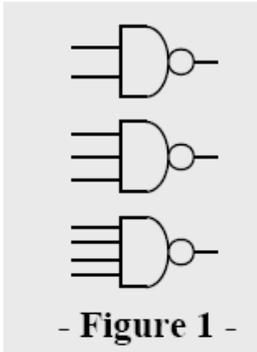
#### 2- Etude matérielle de l'additionneur

Pour réaliser l'addition de 3 bits on dispose de portes NAND à 2 entrées, 3 entrées et 4 entrées, voir Figure 1.

2-1- Transformer en conséquence les équations logiques, éventuellement réduites, de  $S$  et  $C_0$ .

2-2- Etablir les schémas logiques correspondants.

$C_i$	$B$	$A$	$C_0$	$S$
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		



### 3- Exercice 3 :(comparateur)

Déterminer la structure des circuits réalisant les fonctions suivantes entre 2 nombre  $A$  et  $B$  de 4 bits :

1 :  $S = 1$  si  $A = B$

2 :  $S = 1$  si  $A > B$

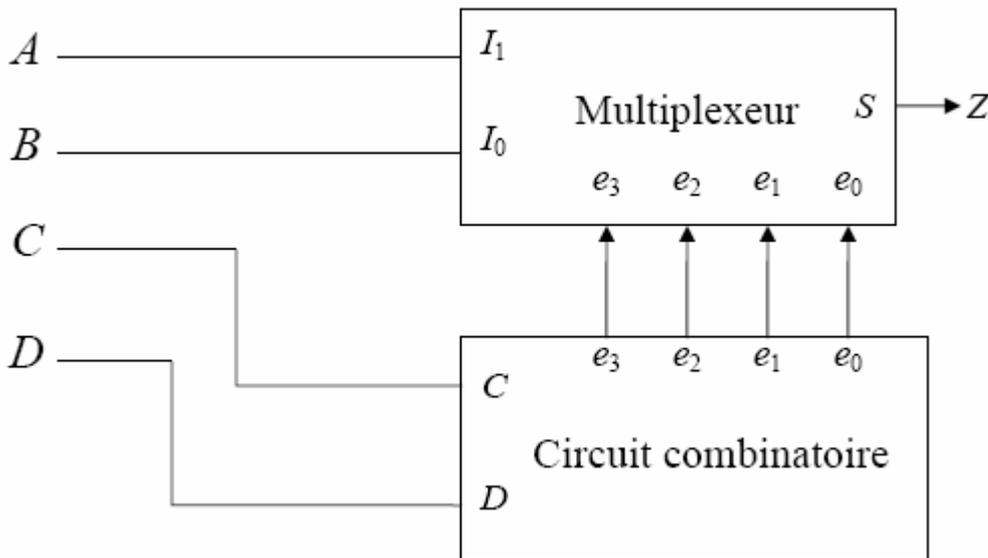
### 4- Exercice 4 : (Polytechnique Montréal)

Soit la fonction :

$$Z = \overline{A}\overline{B}\overline{C}D + \overline{A}\overline{B}C\overline{D} + \overline{A}B\overline{C}D + \overline{A}BC\overline{D} + A\overline{B}\overline{C}D + A\overline{B}C\overline{D} + ABCD$$

a) Représenter cette fonction par une table de Karnaugh à variable inscrite où  $C$  et  $D$  sont les variables inscrites.

b) Vous devez réaliser cette fonction à l'aide d'un multiplexeur à quatre entrées ( $e_0, e_1, e_2$  et  $e_3$ ) commandé par deux bits de sélection ( $I_1, I_0$ ). Les variables  $A$  et  $B$  seront les bits de sélection comme l'illustre la figure suivante



Votre tâche consiste à concevoir le circuit combinatoire à deux entrées ( $C$  et  $D$ ) et dont les quatre sorties sont branchées au multiplexeur. Liste du matériel à votre disposition :

- Des inverseurs,
- Des portes logiques à deux entrées (ET et OU seulement).

c) réaliser maintenant la fonction  $Z$  avec **uniquement des multiplexeurs à deux entrées** (1 bit de sélection  $I_0$  et deux entrées  $e_0$  et  $e_1$ ). Aucun autre élément (inverseur ou porte logique) n'est disponible. Considérer que **vous avez accès aux entrées ( $A, B, C, D$ ) et leurs valeurs inverses**. Donner le schéma du circuit.

### 5- Exercice 5 : (comparateur)

Un comparateur  $n$  bits est un circuit logique contenant  $2n$  entrées,  $a_0, a_1, \dots, a_{n-1}$  et  $b, b, \dots, b_{n-1}$  et deux sorties nommées  $s$  et  $e$ , on aura :

$e=1$  et  $s=0$  si  $A=B$

$e=0$  et  $s=1$  si  $A>B$

$e=1$  et  $s=1$  si  $A<B$

Donner les équations d'un comparateur 1 bit puis le représenter en utilisant des portes élémentaires à deux entrées.

### 6- Exercice 6 : (codeur/décodeur)

1. Dans un décodeur 1 parmi 16, avec deux entrées de validation, quel est le nombre total de voies d'entrée et de voies de sortie ?

2. Donner une implémentation d'un décodeur 1 parmi 4 à base de portes ET. Transformez cette structure pour obtenir une implémentation en technologie NAND. Modifier la structure pour intégrer une entrée de validation  $en$  validant les entrées si  $en=1$ .



# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING

3. A l'aide d'un décodeur, donner une implémentation possible d'un comparateur de deux nombres de 2 bits A et B,  $f(A_1, A_0, B_1, B_0)$  délivrant 3 valeurs booléennes: GT, EQ, LT pour respectivement indiquer que A est respectivement strictement supérieur, égal ou strictement inférieur à B.

4. Donner une implémentation du circuit codeur de priorité à partir de la table de vérité fournie en **Erreur ! Source du renvoi introuvable.**

### 7- Exercice 7 : (Mux/demux)

1. Donner une implémentation possible du MUX à 4 voies simples en logique à deux niveaux.

2. A l'aide de Mux, modifier la structure de l'additionneur parallèle de manière à ce qu'il puisse fonctionner comme opérateur additionneur si le signal de sélection  $a/s=0$ , comme soustracteur si  $a/s=1$ .

3. Montre comment implémenter la table de vérité du comparateur à 2 à l'aide d'un MUX à 2 entrées de sélection.

4. Implémenter la table de vérité du FA à l'aide d'un double MUX 4 voies doubles et un simple inverseur.

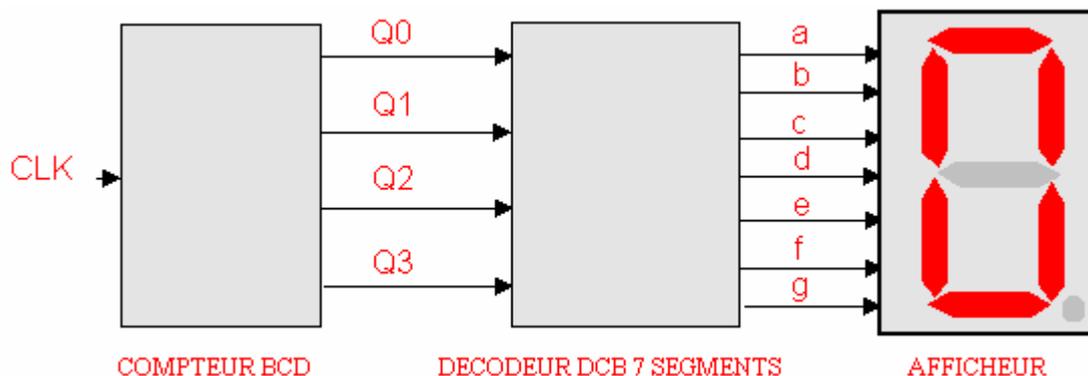
### 8- Exercice 8 : (Additionneur)

1. Donner une solution de réalisation possible d'un additionneur complet à partir d'une logique à deux niveaux en technologie NAND.

2. Montrez que  $\overline{C_{out}}(X + Y + C_{in}) = \overline{X}C_{in}Y + \overline{X}Y\overline{C_{in}} + X\overline{Y}C_{in}$ . De cette égalité, déduisez une nouvelle implémentation du FA à partir d'une logique multi niveaux. Proposer un schéma.

### 9- Exercice 9 : (Décodeur)

Construire le décodeur qui permet de faire afficher en valeur décimale sur un afficheur la valeur binaire présente à son entrée.





# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING

A partir de la documentation technique donnée en annexe, proposer un câblage convenable de décodeur.

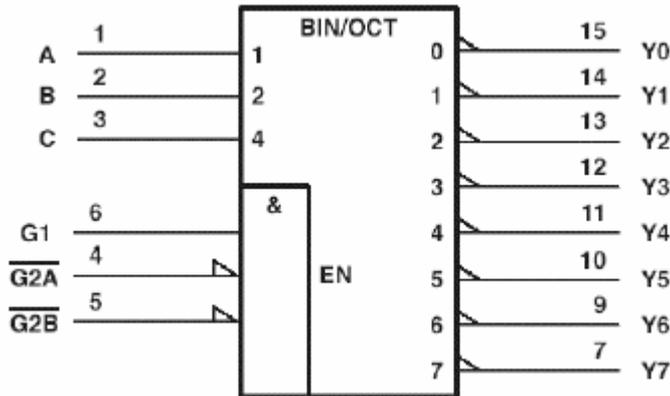
### 10- Exercice 10 : (Décodeur)

On dispose d'un jeu de décodeurs/démultiplexeurs de type 74ALS138 à 3 entrées

- Quelle est l'entrée correspondant au bit de poids fort ?
- Comment faut-il assembler de tels décodeurs pour réaliser un seul décodeur à 4 entrées ?
- Même question pour obtenir un décodeur à 5 entrées.

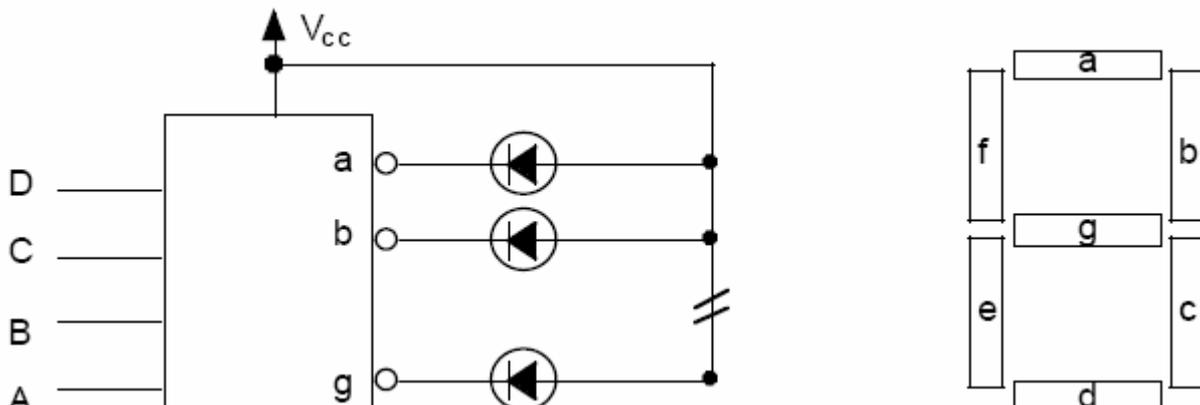
FUNCTION TABLE

INPUTS			OUTPUTS												
ENABLE			SELECT												
G1	$\overline{G2A}$	$\overline{G2B}$	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7		
X	H	X	X	X	X	H	H	H	H	H	H	H	H		
X	X	H	X	X	X	H	H	H	H	H	H	H	H		
L	X	X	X	X	X	H	H	H	H	H	H	H	H		
H	L	L	L	L	L	L	H	H	H	H	H	H	H		
H	L	L	L	L	H	H	L	H	H	H	H	H	H		
H	L	L	L	H	L	H	H	L	H	H	H	H	H		
H	L	L	L	H	H	H	H	L	H	H	H	H	H		
H	L	L	H	L	L	H	H	H	H	L	H	H	H		
H	L	L	H	H	L	H	H	H	H	H	L	H	H		
H	L	L	H	H	H	H	H	H	H	H	H	L	H		
H	L	L	H	H	H	H	H	H	H	H	H	H	L		



### 11- Exercice 11 : (Décodeur)

Le circuit suivant est un pilote/décodeur BCD-7 segments destiné à être utilisé avec un affichage LED à 7 segments. L'entrée D correspond au bit le moins significatif.



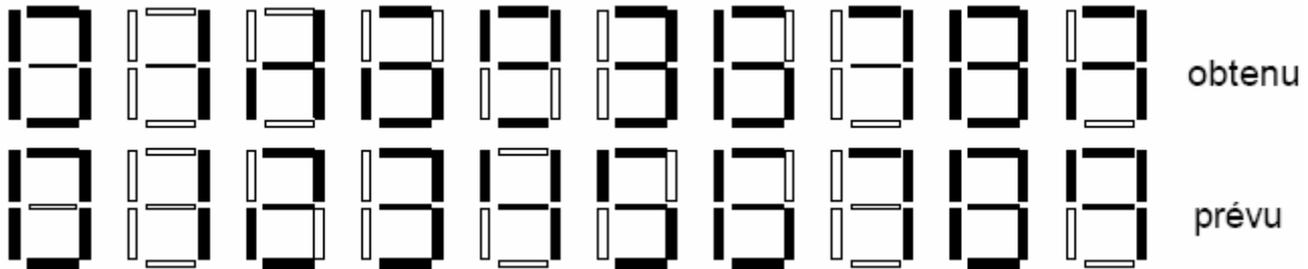


# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING

- Dresser la table de vérité de ce circuit pour les valeurs d'entrée allant de 0 à 9.
- Au cours d'un contrôle on obtient l'affichage ci-dessous. Quelle est l'origine de ce problème ? Les causes peuvent être : entrée/sortie mal connectée ou court-circuitée avec une autre, connexion court-circuitée avec la masse, entrées ou sorties permutées, ...



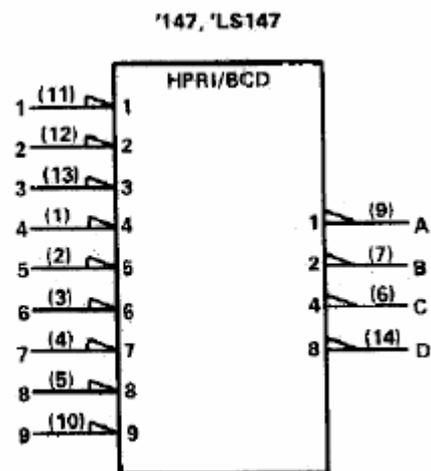
### 12- Exercice 12 : (Codeur)

Le 74LS147 est un codeur prioritaire décimal-BCD dont les caractéristiques sont :

**'147, 'LS147**  
**FUNCTION TABLE**

INPUTS									OUTPUTS			
1	2	3	4	5	6	7	8	9	D	C	B	A
H	H	H	H	H	H	H	H	H	H	H	H	H
X	X	X	X	X	X	X	X	L	L	H	H	L
X	X	X	X	X	X	X	L	H	L	H	H	H
X	X	X	X	X	L	H	H	H	H	L	L	H
X	X	X	L	H	H	H	H	H	H	L	H	L
X	X	L	H	H	H	H	H	H	H	H	L	L
X	L	H	H	H	H	H	H	H	H	H	L	H
L	H	H	H	H	H	H	H	H	H	H	H	L

H = high logic level, L = low logic level, X = irrelevant



- Quel sera son état de sortie s'il reçoit en entrée le nombre 1 1 1 0 1 0 1 1 ?
- Lors d'un contrôle, toutes les entrées sont mises à l'état haut, puis les entrées 1 à 9 sont successivement et une-à-une mises à l'état bas. La lecture des sorties fournit dans ce cas : (1111), (1110), (1101), (1111), (1011), (1010), (1001), (1000), (0111), (0110). Quelles sont les causes possibles de ce dysfonctionnement ?

### 13- Exercice 13 : (Codeur)

Esquisser le schéma interne d'un codeur à 2 bits et à 4 entrées.

### 14- Exercice 14 : (Décodeur/Codeur)

Pour chacun des énoncés qui suivent, dites si cela concerne un codeur, un décodeur ou les deux :

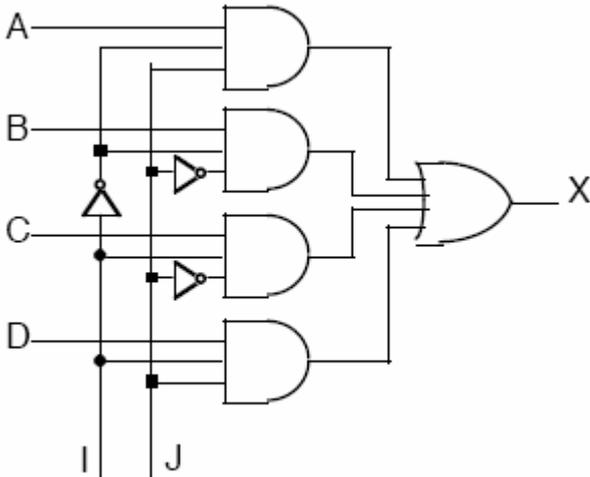
- il possède plus de sorties que d'entrées.
- une seule sortie peut être activée à la fois.
- il sert à traduire en binaire l'enfoncement de touches.
- on peut utiliser ses sorties pour alimenter d'autres circuits qui nécessitent des courants d'intensité relativement élevée (par exemple, diodes LED).



- en mode de fonctionnement normal, plusieurs entrées peuvent être activées simultanément.

### 15- Exercice 15 : (Multiplexeur)

Établir la table de vérité du circuit suivant et déterminer sa fonction



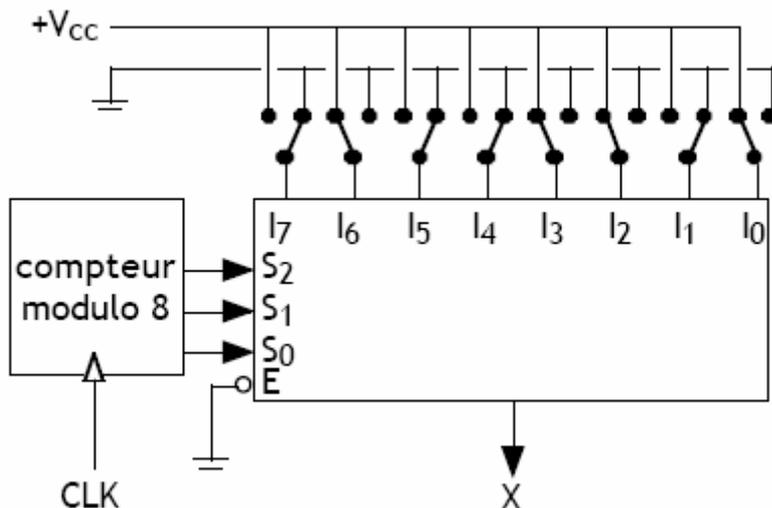
### 16- Exercice 16 : (Multiplexeur)

Dessiner le schéma interne d'un multiplexeur à 3 entrées.

### 17- Exercice 17 : (Multiplexeur)

Un multiplexeur peut être utilisé pour générer des formes d'onde périodiques. Dans le circuit ci-dessous, on utilise un multiplexeur à 8 entrées dont le code d'entrée est fourni par un compteur modulo 8 (la valeur décimale de S vaut successivement 0, 1, 2, 3, 4, 5, 6, 7, 0, 1, ...). Le compteur s'incrémente chaque fois que le signal d'horloge (CLK) passe d'un état bas à un état haut.

- Tracez le chronogramme de la sortie X
- Quelle est la périodicité de la forme d'onde ainsi obtenue ?





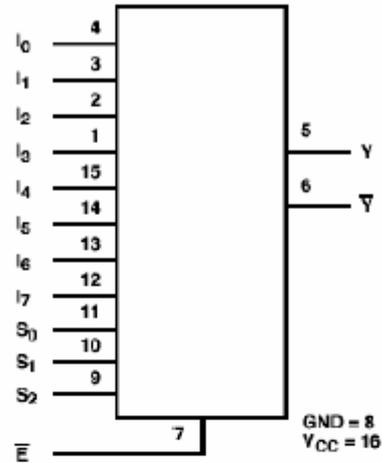
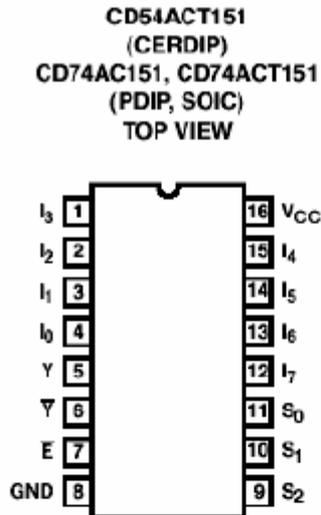
# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING

### 18- Exercice 18 : (Multiplexeur)

On dispose d'une série de multiplexeurs de type 74AC151 à 8 entrées, avec lesquels on aimerait réaliser un seul multiplexeur à 16 entrées. Combien en faudra-t-il ? Montrez comment il faut les assembler.



TRUTH TABLE

INPUTS												OUTPUTS	
$\bar{E}$	$S_2$	$S_1$	$S_0$	$I_0$	$I_1$	$I_2$	$I_3$	$I_4$	$I_5$	$I_6$	$I_7$	$\bar{Y}$	Y
H	X	X	X	X	X	X	X	X	X	X	X	H	L
L	L	L	L	L	X	X	X	X	X	X	X	H	L
L	L	L	L	H	X	X	X	X	X	X	X	L	H
L	L	L	H	X	L	X	X	X	X	X	X	H	L
L	L	L	H	X	H	X	X	X	X	X	X	L	H
L	L	H	L	X	X	L	X	X	X	X	X	H	L
L	L	H	L	X	X	H	X	X	X	X	X	L	H
L	L	H	H	X	X	X	L	X	X	X	X	H	L
L	L	H	H	X	X	X	H	X	X	X	X	L	H
L	H	L	L	X	X	X	X	L	X	X	X	H	L
L	H	L	L	X	X	X	X	H	X	X	X	L	H
L	H	L	H	X	X	X	X	X	L	X	X	H	L
L	H	L	H	X	X	X	X	X	H	X	X	L	H
L	H	H	L	X	X	X	X	X	X	L	X	H	L
L	H	H	L	X	X	X	X	X	X	H	X	L	H
L	H	H	H	X	X	X	X	X	X	X	L	H	L
L	H	H	H	X	X	X	X	X	X	X	H	L	H

H = HIGH voltage level, L = LOW voltage level, X = Don't Care



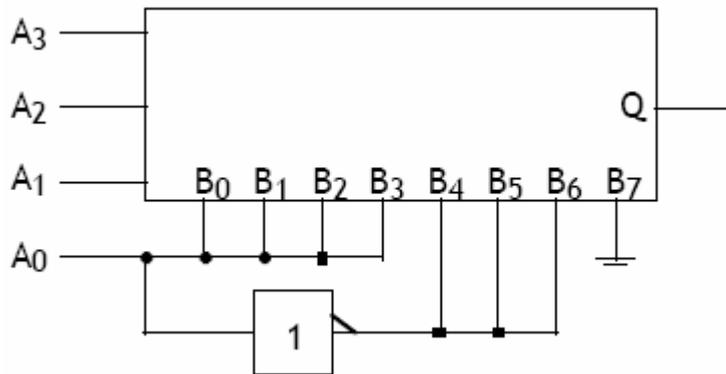
# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING

### 19- Exercice 20 : (Multiplexeur)

Le circuit ci-dessous est constitué d'un multiplexeur qui reçoit en entrée (A) le code binaire correspondant aux 12 mois de l'année (janvier=1, ..., décembre=12); A0 est le bit de poids faible. Etablissez la table de vérité donnant l'état de la sortie Q pour chaque mois. Quelle fonction ce circuit remplit-il ?



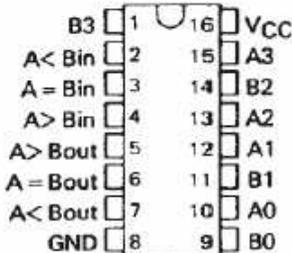
### 20- Exercice 20 : (Comparateur)

Le 74LS85 est un comparateur de grandeurs binaires à 4 bits, ayant trois entrées en cascade et trois sorties. Montrez comment câbler plusieurs circuits intégrés pour réaliser les opérations suivantes :

- comparer deux nombres de 3 bits chacun.
- comparer deux nombres se 10 bits chacun.
- comparer trois nombres de 4 bits chacun (état haut si A=B=C, état bas sinon).

FUNCTION TABLE

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A3, B3	A2, B2	A1, B1	A0, B0	A>B	A<B	A=B	A>B	A<B	A=B
A3>B3	X	X	X	X	X	X	H	L	L
A3<B3	X	X	X	X	X	X	L	H	L
A3=B3	A2>B2	X	X	X	X	X	H	L	L
A3=B3	A2<B2	X	X	X	X	X	L	H	L
A3=B3	A2=B2	A1>B1	X	X	X	X	H	L	L
A3=B3	A2=B2	A1<B1	X	X	X	X	L	H	L
A3=D3	A2=D2	A1=D1	A0>B0	X	X	X	H	L	L
A3=B3	A2=B2	A1=B1	A0<B0	X	X	X	L	H	L
A3=B3	A2=B2	A1=B1	A0=B0	H	L	L	H	L	L
A3=B3	A2=B2	A1=B1	A0=B0	L	H	L	L	H	L
A3=B3	A2=B2	A1=B1	A0=B0	H	H	L	L	L	L
A3=B3	A2=B2	A1=B1	A0=B0	L	L	L	H	H	L
A3=B3	A2=B2	A1=B1	A0=B0	X	X	H	L	L	H

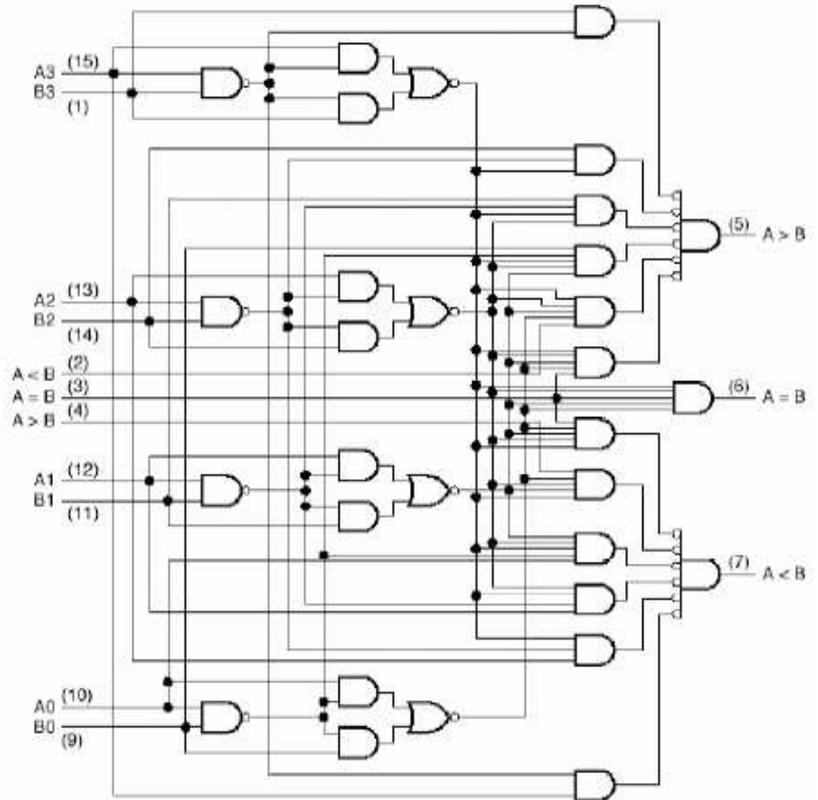
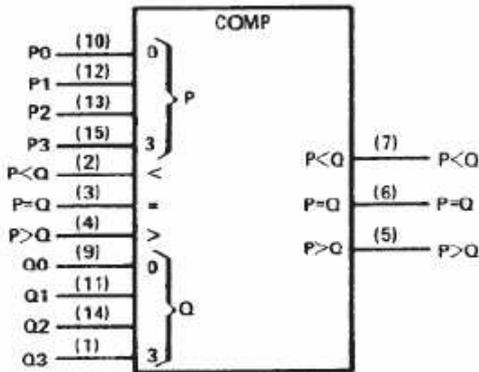




# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

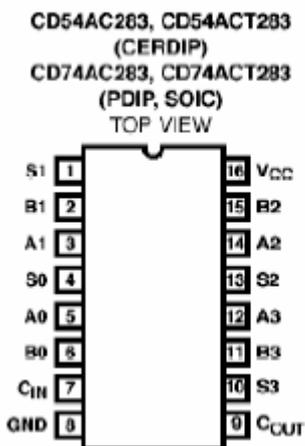
EPMI Cergy  
1AING



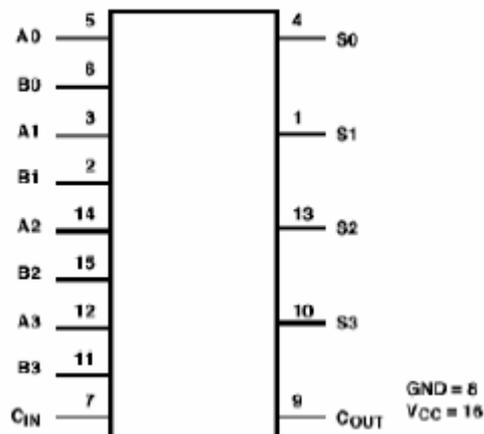
### 21- Exercice 21 : (additionneur)

Montrer comment assembler des additionneurs parallèles à 4 bits pour faire la somme de deux nombres de 7 bits chacun. Voici le brochage du circuit intégré 74AC283

#### Pinout



#### Functional Diagram





### BCD to 7-segment latch/decoder/driver

### HEF4511B MSI

**DESCRIPTION**

The HEF4511B is a BCD to 7-segment latch/decoder/driver with four address inputs ( $D_A$  to  $D_D$ ), an active LOW latch enable input ( $\overline{EL}$ ), an active LOW ripple blanking input ( $\overline{BI}$ ), an active LOW lamp test input ( $\overline{LT}$ ), and seven active HIGH n-p-n bipolar transistor segment outputs ( $O_a$  to  $O_g$ ).

When  $\overline{EL}$  is LOW, the state of the segment outputs ( $O_a$  to  $O_g$ ) is determined by the data on  $D_A$  to  $D_D$ . When  $\overline{EL}$  goes HIGH, the last data present on  $D_A$  to  $D_D$  are stored in the latches and the segment outputs remain stable. When  $\overline{LT}$  is LOW, all the segment outputs are HIGH independent of all other input conditions. With  $\overline{LT}$  HIGH, a LOW on  $\overline{BI}$  forces all segment outputs LOW. The inputs  $\overline{LT}$  and  $\overline{BI}$  do not affect the latch circuit.

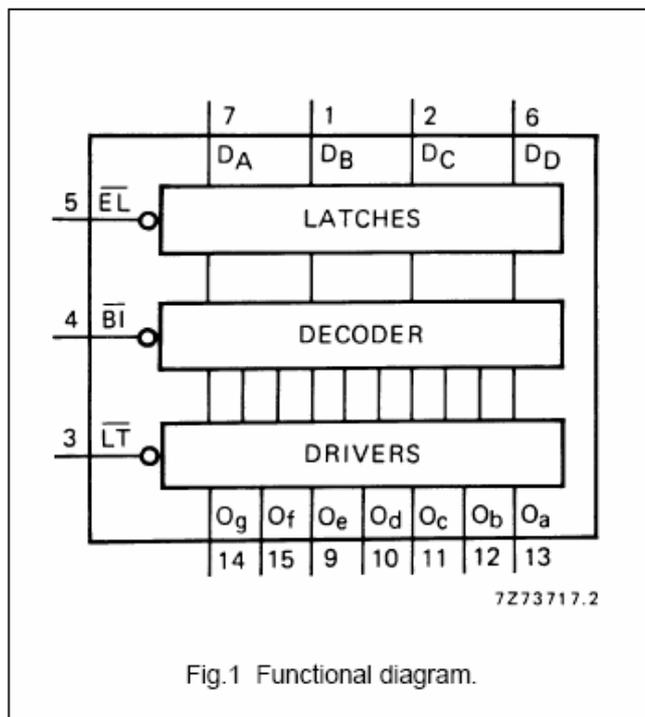


Fig.1 Functional diagram.

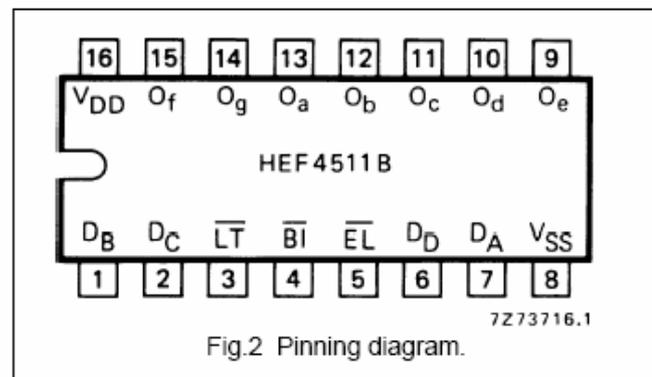


Fig.2 Pinning diagram.

- HEF4511BP(N): 16-lead DIL; plastic (SOT38-1)
- HEF4511BD(F): 16-lead DIL; ceramic (cerdip) (SOT74)
- HEF4511BT(D): 16-lead SO; plastic (SOT109-1)
- ( ): Package Designator North America

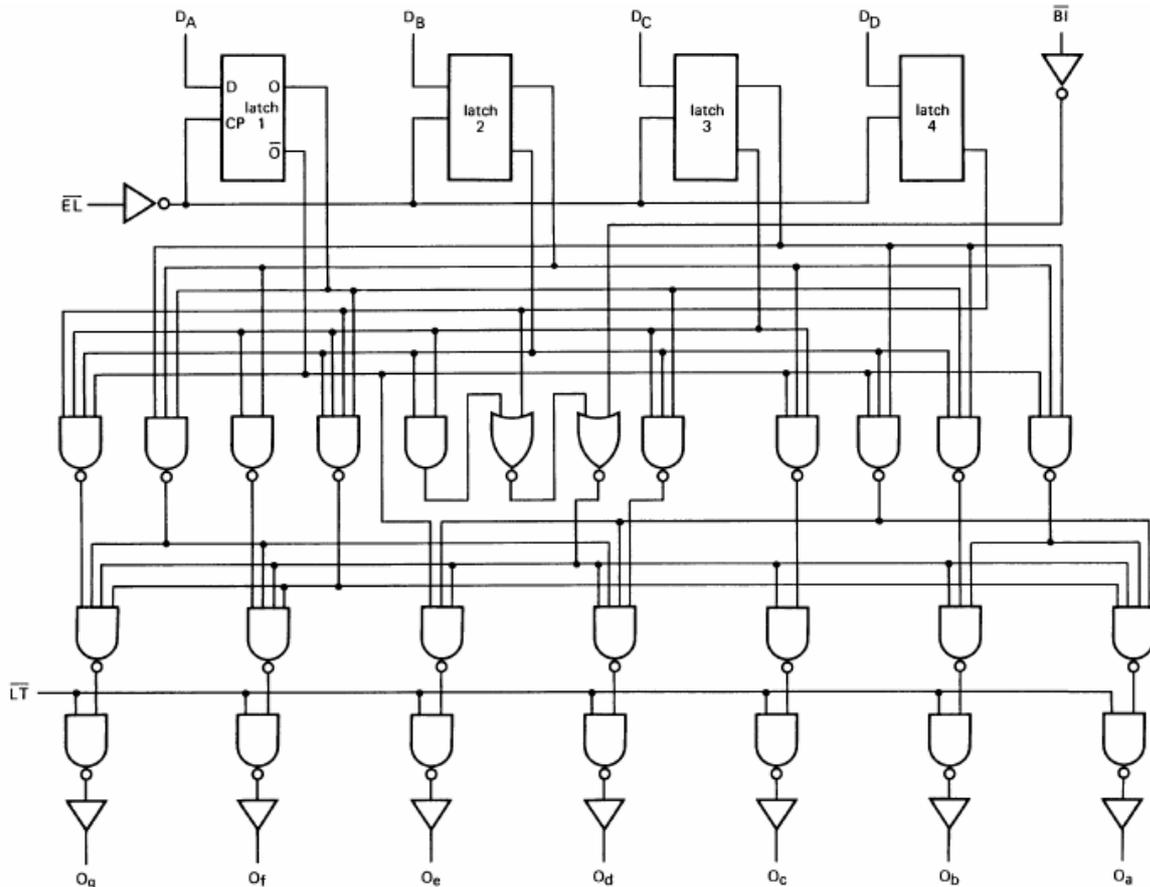
**PINNING**



# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING



FUNCTION TABLE

INPUTS							OUTPUTS							
$\overline{EL}$	$\overline{BI}$	$\overline{LT}$	$D_D$	$D_C$	$D_B$	$D_A$	$O_a$	$O_b$	$O_c$	$O_d$	$O_e$	$O_f$	$O_g$	DISPLAY
X	X	L	X	X	X	X	H	H	H	H	H	H	H	8
X	L	H	X	X	X	X	L	L	L	L	L	L	L	blank
L	H	H	L	L	L	L	H	H	H	H	H	H	L	0
L	H	H	L	L	L	H	L	H	H	L	L	L	L	1
L	H	H	L	L	H	L	H	H	L	H	H	L	H	2
L	H	H	L	L	H	H	H	H	H	L	L	L	H	3
L	H	H	L	H	L	L	L	H	H	L	L	H	H	4
L	H	H	L	H	L	H	H	L	H	H	L	H	H	5
L	H	H	L	H	H	L	L	L	H	H	H	H	H	6
L	H	H	L	H	H	H	H	H	H	L	L	L	L	7
L	H	H	H	L	L	L	H	H	H	H	H	H	H	8
L	H	H	H	L	L	H	H	H	H	L	L	H	H	9
L	H	H	H	L	H	L	L	L	L	L	L	L	L	blank
L	H	H	H	L	H	H	L	L	L	L	L	L	L	blank
L	H	H	H	L	H	H	L	L	L	L	L	L	L	blank
L	H	H	H	H	H	L	L	L	L	L	L	L	L	blank
L	H	H	H	H	H	H	L	L	L	L	L	L	L	blank
H	H	H	X	X	X	X				*				*

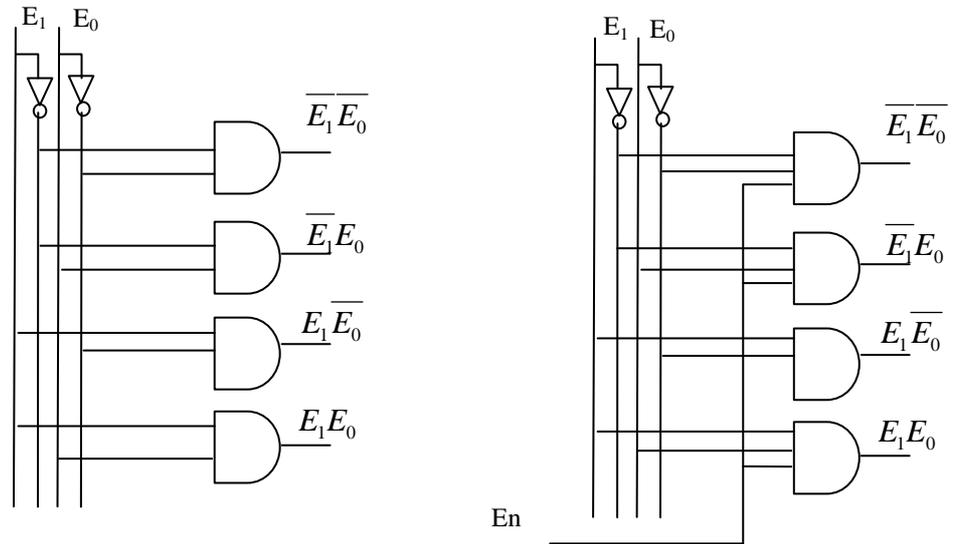


SOLUTIONS EXERCICES CHAPITRE III

**Exo 6 Codeurs/décodeurs**

1-Réponse : 6 entrées ; 16 sorties

2-Réponse :

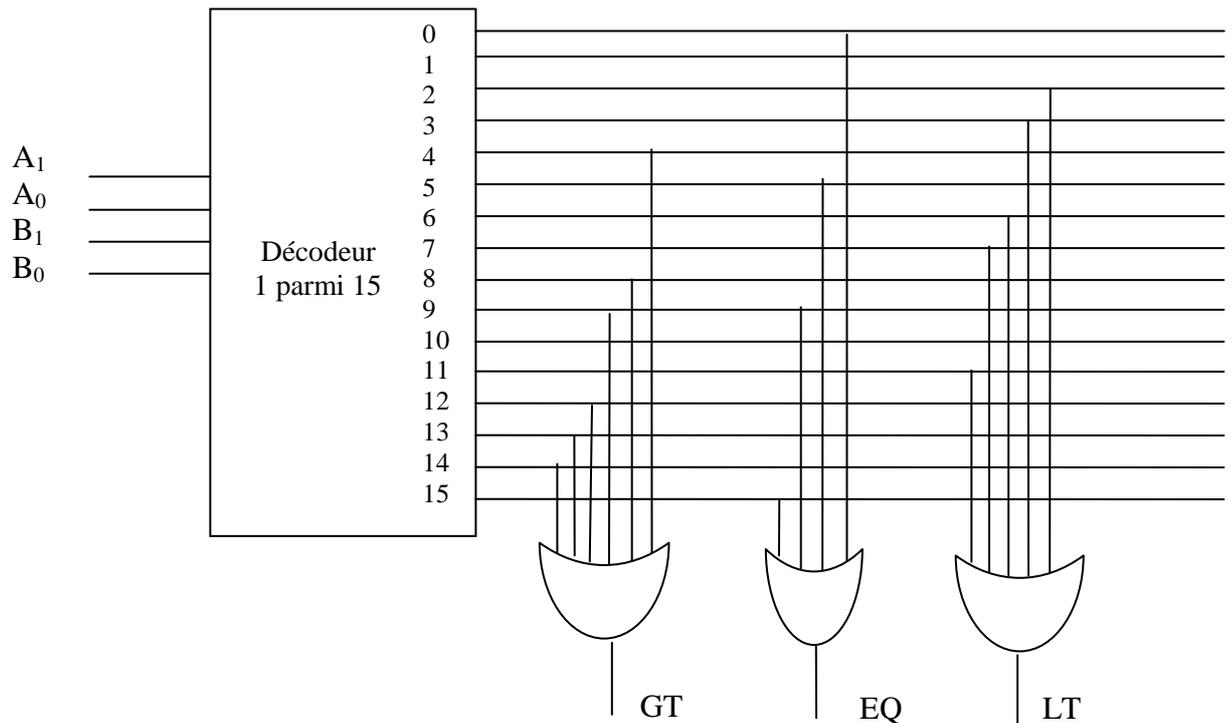


3- Réponse : on peut facilement déduire de la table de vérité les formes canoniques disjonctives suivantes :

$$EQ = \sum m(0,5,10,15)$$

$$GT = \sum m(4,8,9,12,13,14)$$

$$LT = \sum m(1,2,3,6,7,11)$$





# ELECTRONIQUE NUMERIQUE

## Logique combinatoire et multiplexage

EPMI Cergy  
1AING

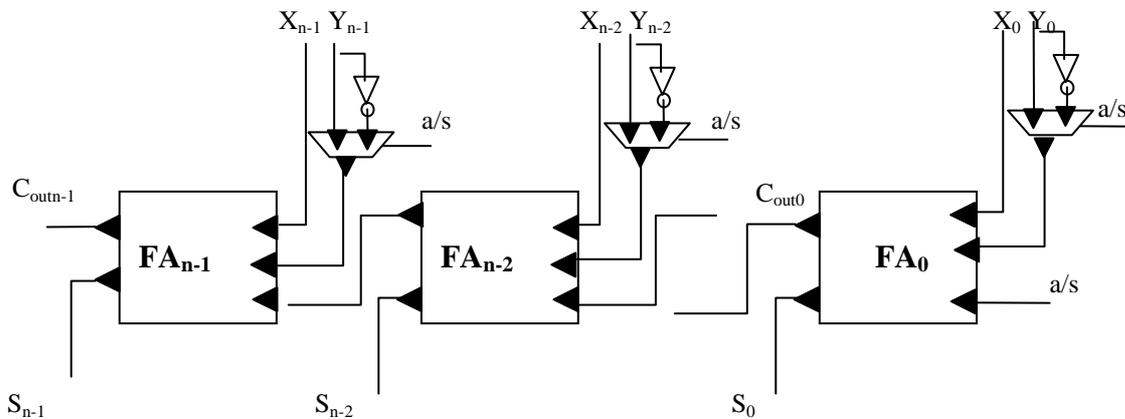
4-Réponse : les équations simplifiées donnent  $S_1 = A + \overline{B}C$  ;  $S_0 = A + B$  ;  $V = A + B + C + D$

### Exercice 7- Multiplexeurs/démultiplexeurs

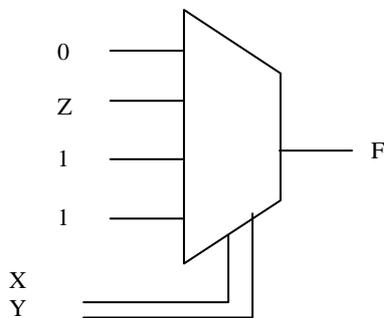
1-Réponse : L'équation de la sortie M se dérive aisément de la table de vérité

$$M = D_0 \overline{S_1} \overline{S_0} + D_1 \overline{S_1} S_0 + D_2 S_1 \overline{S_2} + D_3 S_1 S_0$$

2-Réponse : Il suffit de placer un MUX à deux voies pour sélectionner la valeur de Y ou son complément en n'oubliant pas la retenue d'entrée que l'on fait correspondre à la valeur de a/s..

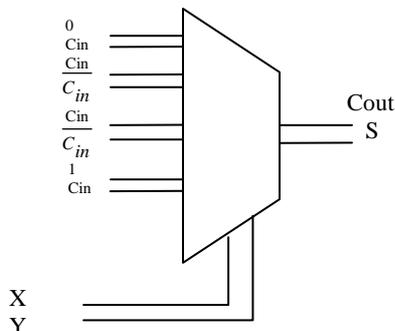


3-Réponse :



4. Implémentez la table de vérité du FA à l'aide d'un double MUX 4 voies doubles et un simple inverseur.

4-Réponse :





### Exercice 8 Circuits arithmétiques

1-Réponse : L'implémentation se déduit des équations simplifiées

$$C_{out} = XY + XC_{in} + YC_{in}$$

$$S = XYC_{in} + \overline{X}YC_{in} + \overline{X}Y\overline{C_{in}} + X\overline{Y}\overline{C_{in}}$$

qui mènent à une logique à deux niveaux ET-OU aisément transformable en logique 2 niveaux NAND.

2-Réponse : L'égalité ci-dessus permet la réécriture de S comme

$$S = \overline{C_{out}}(X + Y + C_{in}) + XYC_{in}$$

qui se traduit par le schéma ci-dessous en logique à 4 niveaux :

