

Ministère de l'Enseignement Supérieur de la Recherche Scientifique et de la Technologie

Institut Supérieur des Etudes Technologiques de Kasserine

Département Génie Electrique

LABORATOIRE SLMP

TP N°2

Les circuits combinatoires

Préparés par :

***RKHISSI YOSRA
MHAMDI TAOUFIK***

TP-II: LES CIRCUITS COMBINATOIRES

Objectifs

Ce TP à pour objectifs:

- Câbler un comparateur, un codeur et un décodeur à partir des portes logiques de base
- Connecter et comprendre le fonctionnement des circuits intégrés suivants : 7485, 74147, 7442 et 7448.
- Remplir les tables de vérité et indiquer les états de sorties des comparateurs, codeurs et décodeurs utilisés au cours de ce TP.

A. Le circuit comparateur

1. Etude théorique

Un comparateur binaire à un bit possède deux entrées A et B et trois sorties possibles :

$A > B$, $A = B$ et $A < B$ comme l'indique la figure 2.1.

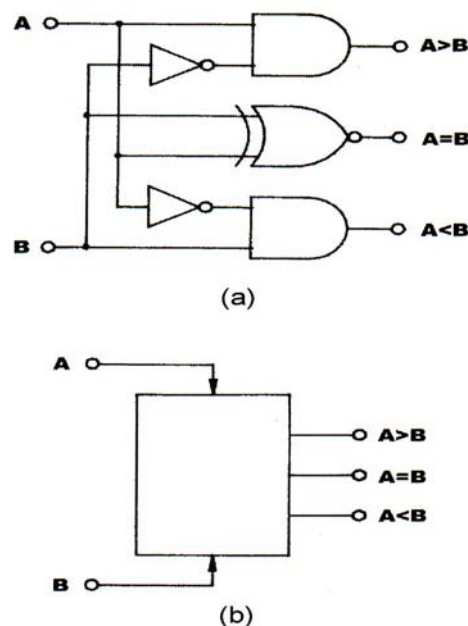


Fig. 2-1 : Comparateur

Dans cette application, on utilise le circuit TTL 7485 qui est un comparateur de deux nombres binaires à 4 bits **A** (A_3, A_2, A_1 et A_0) et **B** (B_3, B_2, B_1 et B_0) comme indiqué dans la Fig 2.1 (b) dont le schéma interne est illustré par la Fig 2.1 (a)

Chaque bit est représenté par son poids : 2^0 pour A_0, B_0 , 2^1 pour A_1, B_1 et 2^2 pour A_2, B_2 , 2^3 pour A_3, B_3 .

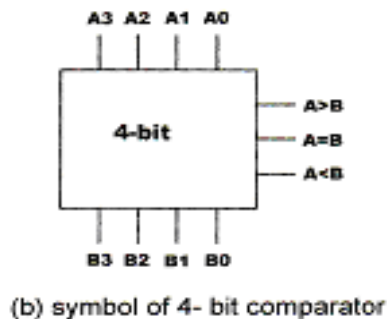
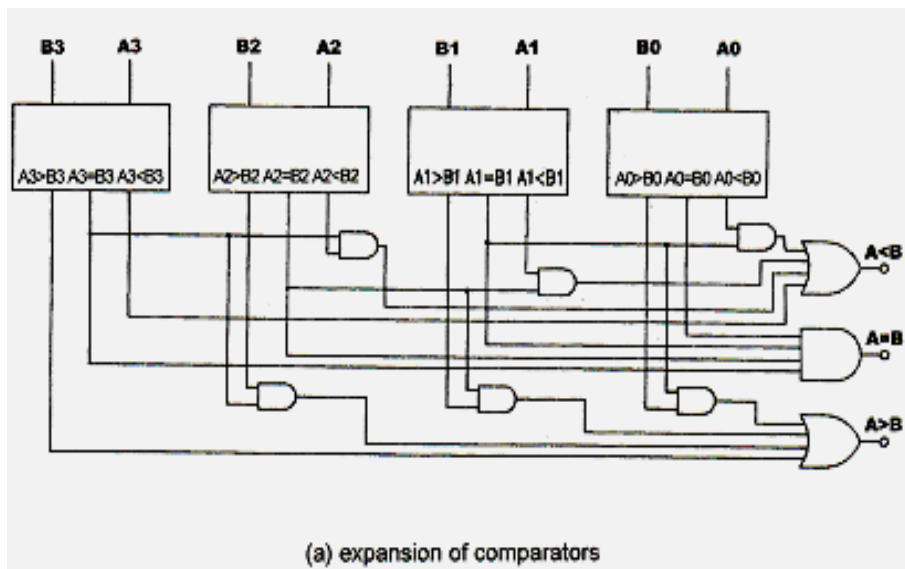


Fig. 2-2 : Comparateur 4 bits

Ainsi, on peut comparer des nombres de **8, 12, 16 bits....**

Le brochage de ce circuit est donné à la figure 2-3, tandis que la figure 2-4 représente son schéma logique.

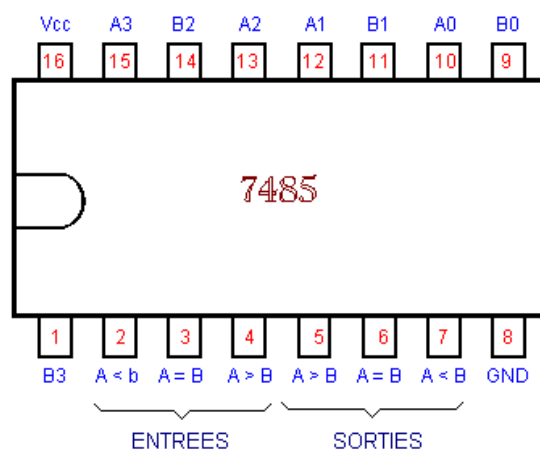


Fig. 2-3 : Brochage du circuit intégré 7485

De plus, il dispose de 3 entrées notées **A = B**, **A > B** et **A < B** qui autorisent la mise en cascade de plusieurs circuits comparateurs du même type.

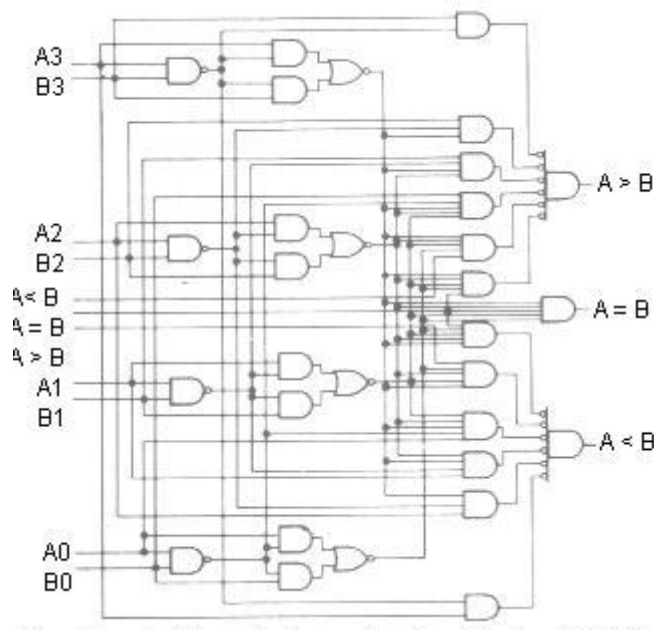


Fig. 2-4 : Schéma logique du circuit intégré 7485

La table de vérité Tableau I met en évidence l'action des entrées $A > B$, $A < B$ et $A = B$.

Entrées des nombres				Entrées cascadables			Sorties		
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	1	0	0
A3 < B3	X	X	X	X	X	X	0	1	0
A3 = B3	A2 > B2	X	X	X	X	X	1	0	0
A3 = B3	A2 < B2	X	X	X	X	X	0	1	0
A3 = B3	A2 = B2	A1 > B1	X	X	X	X	1	0	0
A3 = B3	A2 = B2	A1 < B1	X	X	X	X	0	1	0
A3 = B3	A2 = B2	A1 = B1	A0 > B0	X	X	X	1	0	0
A3 = B3	A2 = B2	A1 = B1	A0 < B0	X	X	X	0	1	0
A3 = B3	A2 = B2	A1 = B1	A0 = B0	1	0	0	1	0	0
A3 = B3	A2 = B2	A1 = B1	A0 = B0	0	1	0	0	1	0
A3 = B3	A2 = B2	A1 = B1	A0 = B0	0	0	1	0	0	1
A3 = B3	A2 = B2	A1 = B1	A0 = B0	X	X	1	0	0	1
A3 = B3	A2 = B2	A1 = B1	A0 = B0	1	1	0	0	0	0
A3 = B3	A2 = B2	A1 = B1	A0 = B0	0	0	0	1	1	0

Tableau I : Table de vérité du circuit intégré 7485

- Si l'on souhaite que la sortie $A = B$ passe à l'état 1 chaque fois que les deux nombres binaires sont égaux, il suffit de porter l'entrée $A = B$ à l'état 1, l'état des entrées $A < B$ et $A > B$ n'ayant alors pas d'importance.

► Si l'on souhaite que la sortie $A > B$ passe à l'état **1** également dans le cas où les deux nombres binaires sont égaux, il suffit de porter l'entrée $A > B$ à l'état **1** et de porter les entrées $A < B$ et $A = B$ à l'état **0**.

Dans cette configuration de l'état des entrées $A > B$, $A < B$ et $A = B$, la sortie $A > B$ est à l'état **1** lorsque le nombre binaire A est supérieur au nombre binaire B ou quand ces deux nombres sont égaux. Elle indique donc si $A \geq B$.

► De même, en portant l'entrée $A < B$ à l'état **1** et les entrées $A > B$ et $A = B$ à l'état **0**, la sortie $A < B$ indique le nombre binaire A est inférieur ou égal au nombre binaire B .

En mettant en série deux comparateurs **7485**, on peut comparer deux nombres de **8 bits**. Il suffit de relier la sortie $A = B$ du premier comparateur à l'entrée correspondante du second et de faire de même avec les sorties $A > B$ et $A < B$. Les liaisons à effectuer sont indiquées à la figure 2.

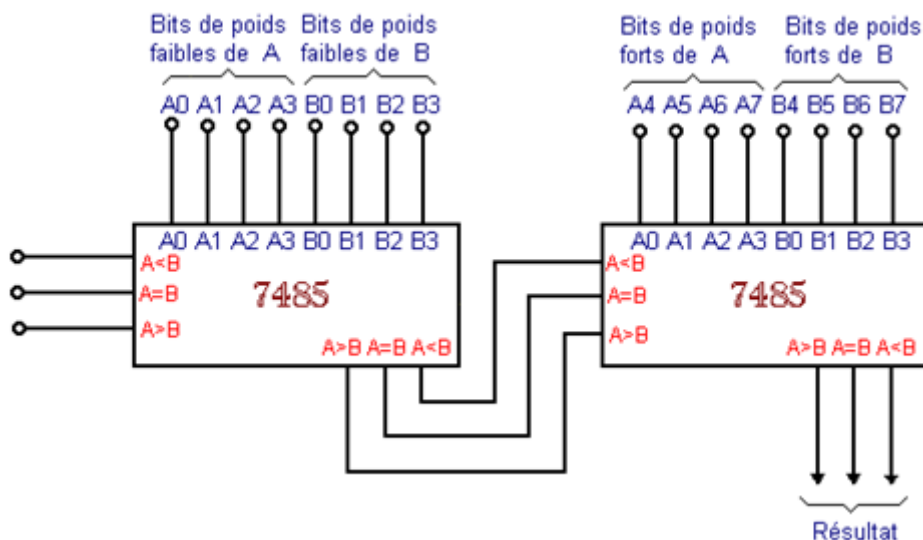


Fig. 2-5 Mise en cascade de deux circuits intégré 7485

Ainsi, on compare le nombre A formé des 8 bits $A7$ à $A0$ ($A7 = \text{MSB}$ et $A0 = \text{LSB}$) et le nombre B formé des 8 bits $B7$ à $B0$ ($B7 = \text{MSB}$ et $B0 = \text{LSB}$).

Le premier circuit compare les poids faibles de A avec le poids faibles de B . Le résultat de cette comparaison est transmis aux entrées $A < B$, $A = B$ et $A > B$ du deuxième circuit. Celui-ci compare les poids forts de A avec les poids forts de B et, en fonction du résultat de la comparaison des bits de poids faibles de A et B , indique sur ses sorties $A > B$, $A = B$ et $A < B$ le résultat de la comparaison des nombres A et B .

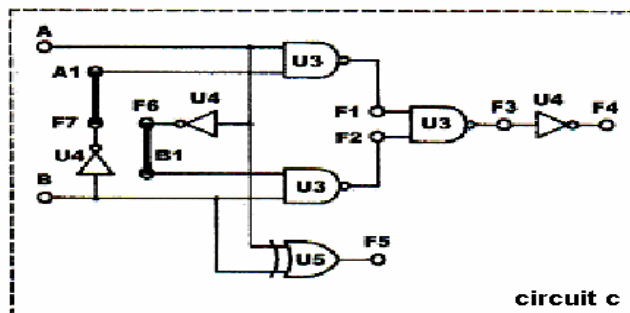
2. Matériel utilisé

DLLT-1300 Digital Logic Lab Trainer; Module DLLT-EM02: « Assembled Logic Circuits (1) Experiment »

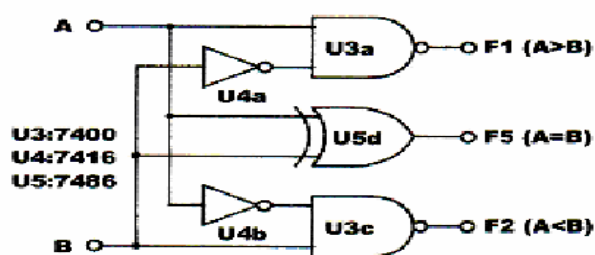
3. Manipulation

a - Comparateur construit par des fonctions logiques

1. Câbler le circuit de la Figure. 2-6 (a) comme le montre la figure. 2-6 (b).



(a)



(b)

Fig. 2-6: Comparateur à un bit

2. Connecter les entrées A et B au interrupteurs SW1 et SW2. Brancher F1, F2 et F5 à L1, L2 et L3 respectivement.
3. Compléter le Tableau II en indiquant l'état des sorties F1, F2 et F5.

INPUT			OUTPUT		
SW2(B)	SW1(A)		F1	F2	F5
0	0	A=B			
0	1	A>B			
1	0	A<B			
1	1	A=B			

Tableau II

b - Comparateur construit par le circuit TTL 7485

- 1- Connecter les entrées cascables A>B à SW1 et F1; A=B à SW2 et F2; A<B à SW3 et F3. Connecter les entrées A1 à A4 et B1 à B4 du CI 7485 aux Interrupteurs de la maquette DLLT -1300
- 3- A1 à A4 représentent le nombre A, et B1 à B4 représentent le nombre B. Compléter alors le tableau de comparaison suivant dans le cas où **les deux mots A et B sont égaux**.

INPUT			OUTPUT		
SW3	SW2	SW1			
A>B	A=B	A<B	A<B	A=B	A>B
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	1			

Tableau III

4- Mettre SW3 à 0; Sw2 à 1 et SW1 à 0. Observer et noter les sorties du comparateur dans les **conditions suivantes des mots A et B** :

(1) A>B

(2) A=B

(3) A<B

5- Débrancher A1~ A4 et B1~ B4 des Interrupteurs puis les connecter aux DIP Switch DIP1.0 ~ DIP1.3 et DIP2.0 ~ DIP 2.3 respectivement. Refaire la question 4 et conclure.

B. Le circuit Codeur

1. Etude théorique

Un codeur ou encodeur est un circuit logique qui possède 2^N voies d'entrées dont une seule est **active** et N voies de sorties comme illustré sur la figure 2-7.

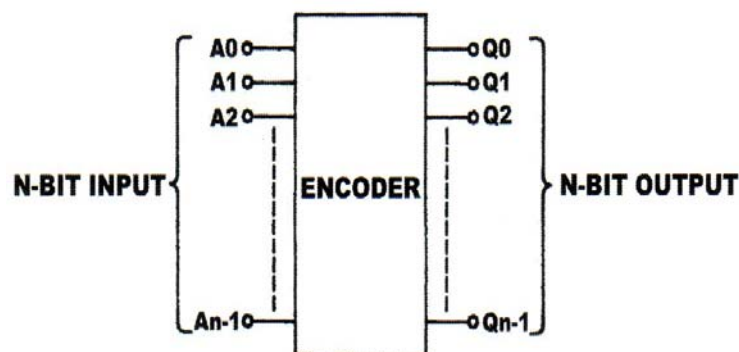


Fig. 2-7 : Codeur

➤ Codeur de priorité

C'est un dispositif qui réalise le codage du numéro le plus élevé dans le cas où plusieurs entrées seraient actionnées.

Le circuit 74147 est un codeur de priorité BCD à 9 entrées et 4 sorties. La priorité est établie en ordre décroissant c'est à dire que l'entrée 9 a la priorité la plus élevée tandis que l'entrée 1 a la priorité la plus faible.

Le tableau VI représente une partie de la table de vérité du codeur 74147 :

INPUT									OUTPUT			
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

Tableau IV : Fonctionnement du codeur 74147

Les entrées et les sorties sont actives à l'état bas.

Dans le cas ou, par exemple, l'entrée 8 est active et l'entrée 9 inactive, l'entrée 8 a la priorité. On aura alors à la sortie le nombre 8 affiché en BCD.

2. Matériel utilisé

DLLT-1300 Digital Logic Lab Trainer; Module DLLT-EM05: «Assembled Logic Circuits (4) Experiment Module »

3. Manipulation

a- Construction d'un codeur 4-à-2 à l'aide de portes logiques

1. Etablir le schéma de la figure 2.8. .Reproduire les séquences de la table de vérité et noter les états des sorties sur la table V. Quel est le type de conversion réalisé ?

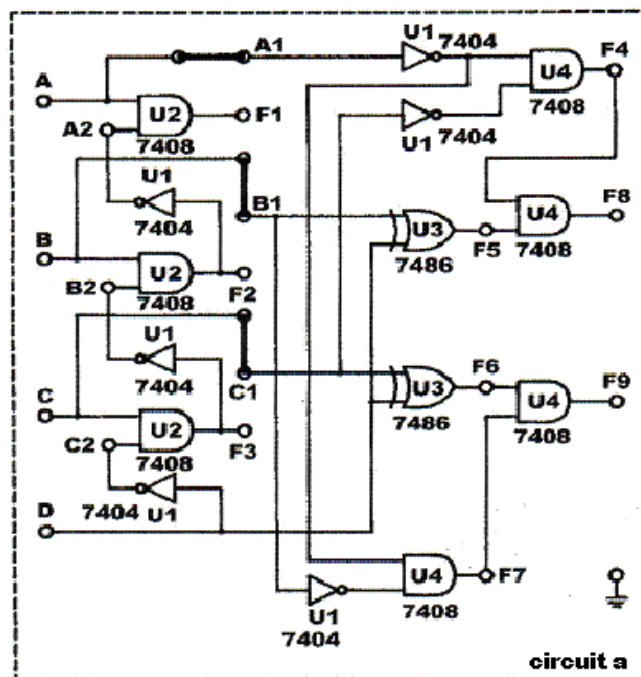


Fig. 2-8 : Module DLLT-EM05 : Circuit a

D	C	B	A	F8	F9
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

Tableau V

2. Etablir le schéma de la figure 2.9. Reproduire les séquences de la table de vérité et noter les états des sorties sur la table VI.

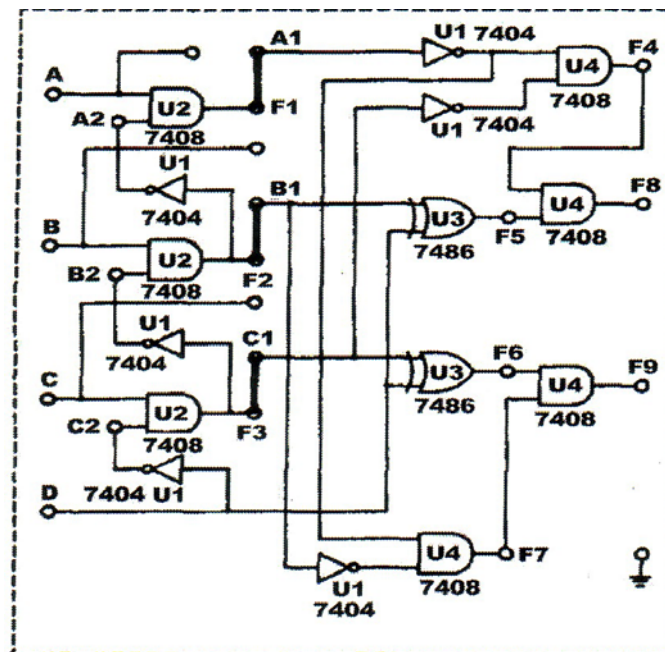


Fig. 2-9 : Module DLLT-EM05, circuit a

D	C	B	A	F8	F9
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

Tableau VI

3. Comparer l'état des sorties dans chacun des tableaux V et VI. Quelle est la différence ?

b- Construction d'un codeur 10-à-4 avec des circuits intégrés TTL

1. On va travailler avec le circuit intégré 74147 (U7) comme illustré sur la figure 2.10.

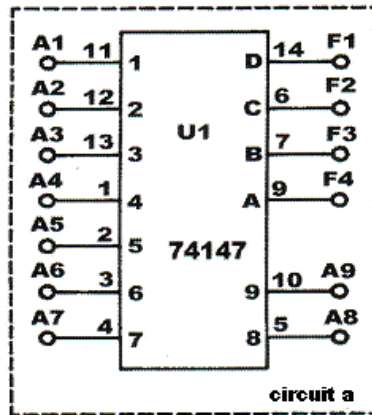


Fig. 2-10 : DLLT-EM06, circuit a

Reproduire les séquences de la table de vérité et noter les états des sorties sur la table VII.

A9	A8	A7	A6	A5	A4	A3	A2	A1	F4	F3	F2	F1
0	1	1	1	1	1	1	1	1				
0	0	1	1	1	1	1	1	1				
1	1	1	1	1	1	1	1	0				
1	1	1	1	1	1	1	0	0				
1	1	1	1	1	1	0	1	1				
1	1	1	1	1	0	0	0	0				
1	1	1	1	0	1	1	1	1				
1	1	1	1	0	0	0	1	1				
1	1	1	0	1	1	1	0	0				
1	1	0	1	1	0	1	1	0				
1	1	0	0	0	1	1	1	1				
1	0	0	0	0	0	1	1	1				

Tableau VII

2. Indiquer et Interpréter l'état des sorties dans le cas ou $A_1=A_2=A_3=A_4=0$ et $A_5=A_6=A_7=A_8=A_9=0$?

C. Circuit Décodeur

1. Etude théorique

1.1. Définition

Un codeur est un circuit numérique qui possède N entrées et 2N sorties. Pour chacune des combinaisons possibles des entrées, seule une ligne de sortie est validée. Les décodeurs sont souvent dotés d'une ou plusieurs entrées de validation E qui servent à valider son fonctionnement. Le schéma d'un décodeur à N bits d'entrée est donné par la figure suivante :

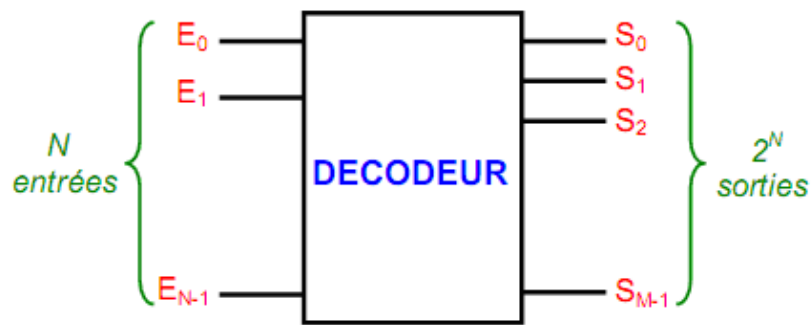


Fig. 2-11 : Schéma fonctionnel d'un décodeur

1.2. Décodeur BCD-7 segments 7448

Ce décodeur dispose, comme illustré sur la figure 2.12 de :

- Variables d'entrée:

- D, C, B et A, constitutifs d'un nombre binaire codé en BCD où **D** est le bit le plus significatif de poids 2^3 , tandis que celui le moins significatif a un poids de 2^0 .
- LT: entrée de test de l'état des segments
- RBI: entrée de blanchiment des zéros pas significatifs.

- Variables de sortie:

- 7 variables de "a" à "g" correspondant aux 7 segments,
- RBO: sortie de blanchiment des zéros pas significatifs des décades inférieures.

- Remarque :

Les entrées supplémentaires sont prévues pour :

► **LT** ou «**lamp test**» qui permet de vérifier le fonctionnement de l'afficheur en allumant tous les segments si **BI** est à l'état 1.

► **BI / RBO** ou «**blanking input**» qui permet l'effacement des segments de l'afficheur quelque soit l'état des autres entrées.

► **RBI** ou «**ripple blanking input**» qui permet l'effacement des 0 à gauche si **A, B, C, D** sont à 0.

- Etude de la table de vérité:

a- Test des segments:

La variable LT est prioritaire sur toutes les autres variables d'entrée; elle est active à l'état 0. Quel que soit l'état des autres variables d'entrée, toutes les sorties 7 segments sont mises à l'état 1 (éclairage).

b- Blanchiment des zéros:

La variable RBI permet, lorsqu'elle est active, c'est à dire à l'état logique 0, d'effacer le chiffre zéro sur l'afficheur, et lui seul.

La variable de sortie RBO passe alors à l'état 0, dans le but de commander l'entrée RBI du décodeur suivant, cela afin d'effacer en cascade les zéros pas significatifs, de l'afficheur de poids le plus fort à l'afficheur de poids le plus faible.

c- Décodage BCD/7segments:

Le décodage est assuré de sorte que le circuit intégré fournisse un code 7 segments pour chaque nombre binaire de 0 à 9. Aussi pour les nombres de 10 à 15, certains signes particuliers sont définis, bien qu'ils ne présentent aucune utilité.

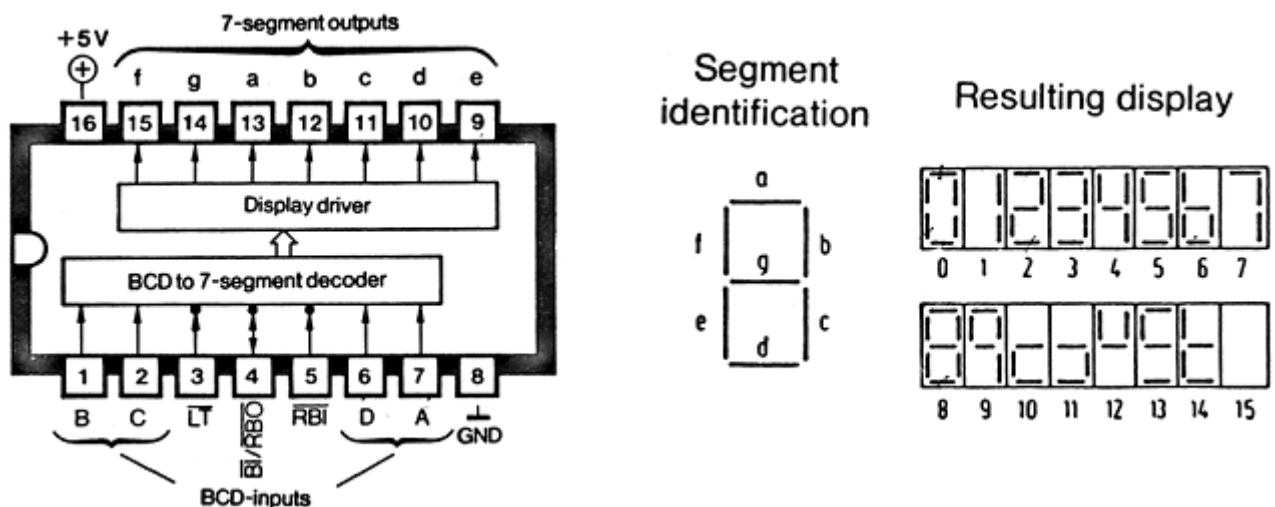


Fig. 2-12 : Brochage d'un décodeur BCD-7segments de type 7448

2. Matériel utilisé

DLLT-1300 Digital Logic Lab Trainer; Module DLLT-EM04: «Assembled Logic Circuits (3) Experiment Module » et DLLT-EM05 : “Assembled Logic Circuits (4) Experiment Module”.

3. Manipulation

a- Construction d'un décodeur 2-à-4 avec des portes logiques

1. Etablir le schéma de la figure 2.13 et reproduire les séquences de la table de vérité et noter les états des sorties sur la table VIII. Est il possible d'utiliser ce circuit pour faire fonctionner un circuit parmi d'autres ?

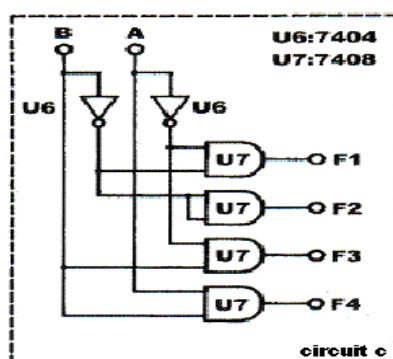


Fig. 2-13 : Module DLLT-EM05, circuit c

B	A	F1	F2	F3	F4
0	0				
0	1				
1	0				
1	1				

Tableau VIII

b- Construction d'un décodeur 4 à 10 avec des circuits intégrés

1. Le circuit intégré U10 (7442) du module DLLT-EM04, comme le montre la figure 2.14, est un décodeur BCD-Decimal.

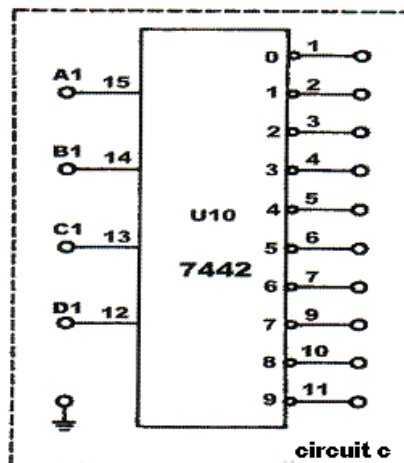


Fig. 2-14 : Module DLLT-EM04, circuit c

Connecter les entrées A1, B1, C1, D1, respectivement, aux sorties BCD du DIP Switch "1", "2", "4", "8".

Les DIP Switchs permettent de convertir un nombre décimal en code BCD.

Reproduire les séquences de la table de vérité et noter les états des sorties sur le tableau IX.

	D	C	B	A	0	1	2	3	4	5	6	7	8	9
0														
1														
2														
3														
4														
5														
6														
7														
8														
9														

Tableau IX

c- Décodeur BCD-7-Segment

La figure 2.15 montre un décodeur de type 7448 et un afficheur 7segments :

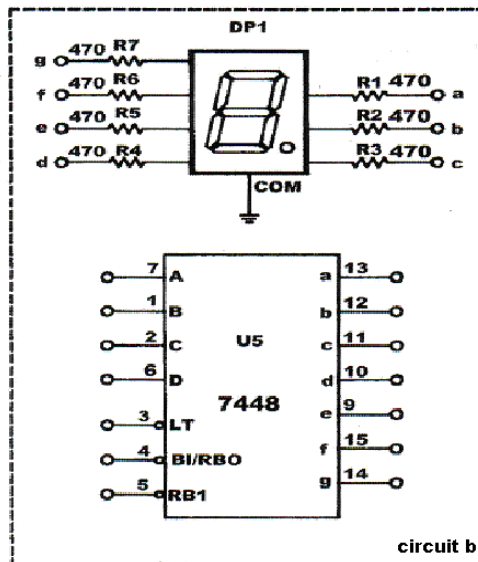


Fig. 2-15 : Module DLLT-EM05, circuit b

1. Connecter les entrées A, B, C, D de U5 (7448) respectivement aux Interrupteurs SW3, SW2, SW1, SW0. Connecter "RBI" et "LT" aux interrupteurs et "BI/RBO" à la led L0. Mettre DIP "RBI" et "LT" à "1".

2. Reproduire les séquences de la table de vérité du tableau X et observer les sorties de l'afficheur 7 segments. Est il possible d'afficher le nombre 10 et comment ?

3. Mettre et "LT" à "0" et garder "RBI" à l'état "1". Refaire l'étape 2. Quelles sont les différences par rapport à l'étape 2?

D	C	B	A	Display
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Tableau X

4. Mettre "RBI" à "0" et "LT" à "1". Refaire l'étape 2. Comparer l'état des sorties par rapport à l'étape 2 quand DCBA=0000-1001. Est ce que les sorties sont différentes ?