

TP 4 : Étude et Réalisation de Fonctions Logiques Combinatoires Usuelles – Codeur/Décodeur -

La préparation du TP

- *La préparation du TP est une étape essentielle qui doit être réalisée avant la séance.*
- *Chaque étudiant est tenu d'élaborer la partie théorique du TP de manière rigoureuse, structurée et propre.*
- *La préparation, strictement individuelle, doit être rédigée de façon numérique imprimée et inclure des éléments essentiels tels que des tables de vérité, des logigrammes numérotés et toute autre représentation graphique nécessaire à la compréhension du travail.*

Objectifs du TP

Dans ce TP, vous allez apprendre à :

- Comprendre et Vérifier le comportement des codeurs et décodeurs numériques et des afficheurs 7 segments.

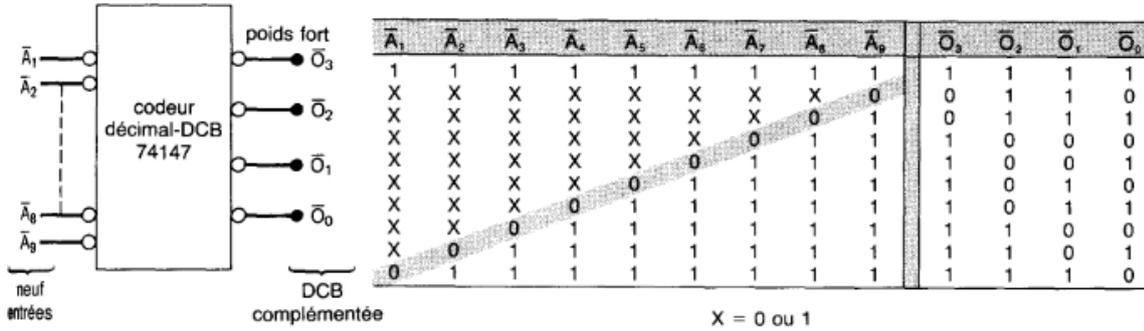
Partie 1 : Codeur (encodeur) avec priorité (Circuit 74147)

1. Présentation du CI 74147

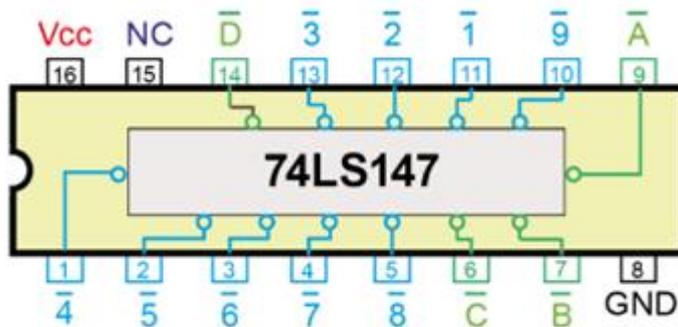
Le 74147 est un codeur de priorité de la famille TTL, qui convertit un chiffre décimal (1 à 9) en une valeur binaire codée en décimal (BCD). Il fonctionne sur le principe de priorité, c'est-à-dire que si plusieurs entrées sont activées simultanément, celle ayant la plus grande valeur (poids le plus fort) est prioritaire.

- **Le CI 74147 est déclenché par le niveau bas (0) des entrées ainsi que des sorties.**
- Il possède 9 entrées ($\overline{A1}$ à $\overline{A9}$) et 4 sorties BCD (\overline{A} , \overline{B} , \overline{C} , \overline{D}).

Codeur de priorité décimal-DCB 74147.



2. Travail à faire :



| A_9 | A_8 | A_7 | A_6 | A_5 | A_4 | A_3 | A_2 | A_1 | A | B | C | D |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | |
| 0 | 0 | 1 | 1 | 0 | 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 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 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 | 0 | 0 | 1 | 1 | 1 | 1 | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | | | | |

Tableau 1

1. Remplir la table de vérité (tableau 1) d'un codeur de priorité 10 – BCD.
2. Utiliser le CI 74147 pour réaliser le codeur 10-4, Pour obtenir le code B.C.D à partir des sorties de 74147, il faut ajouter un inverseur à chacune des sortie.
3. Relier les entrées aux sortie TTL des commutateurs (Data switch) DIP1 et DIP2 et les sorties aux indicateurs logiques.
4. Suivre les séquences d'entrée de la table de vérité et ensuite vérifier les états des sorties.

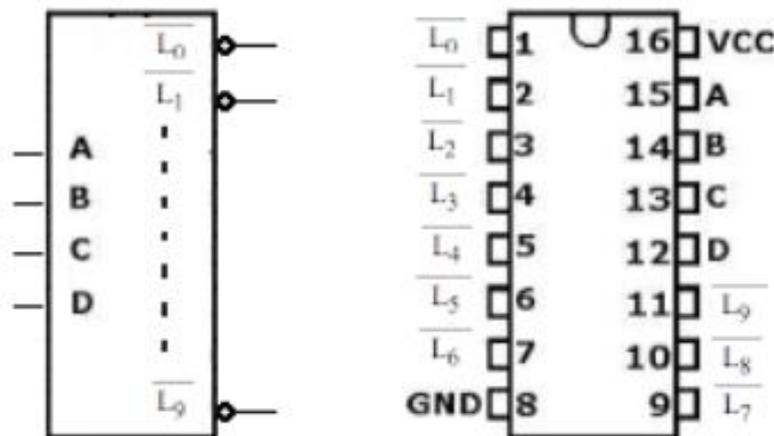
Partie 2 : Décodeur BCD-Décimal (Circuit 7442)

1. Présentation du 7442

Le circuit 7442 est un décodeur qui convertit un code BCD (4 bits en entrée) en un signal de sortie correspondant à l'un des 10 chiffres décimaux (0 à 9). Chaque sortie est active à l'état bas (0 logique).

- Il possède 4 entrées (A, B, C, D) et 10 sorties ($\overline{L_0}$ à $\overline{L_9}$).
- Chaque sortie est active à l'état bas (0 logique), ce qui signifie qu'elle passe à 0 lorsque la combinaison BCD correspondante est appliquée aux entrées.

2. Travail à faire :



Brochage du circuit intégré 7442

1. Compléter et vérifier expérimentalement la table de vérité du décodeur (tableau 2) en appliquant les différentes combinaisons BCD sur les entrées du 7442.

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

Tableau 2

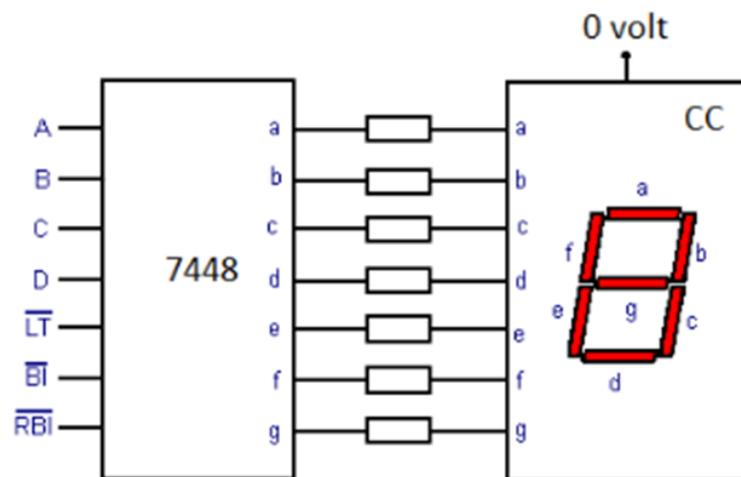
2- Déterminer les expressions logiques de chaque sortie en fonction des entrées.

- $\bar{L}_0 = \dots\dots\dots$
- $\bar{L}_1 = \dots\dots\dots$
- $\bar{L}_2 = \dots\dots\dots$
- $\bar{L}_3 = \dots\dots\dots$
- $\bar{L}_4 = \dots\dots\dots$
- $\bar{L}_5 = \dots\dots\dots$
- $\bar{L}_6 = \dots\dots\dots$
- $\bar{L}_7 = \dots\dots\dots$
- $\bar{L}_8 = \dots\dots\dots$
- $\bar{L}_9 = \dots\dots\dots$

Partie 3 : Décodeur BCD-7 Segments (Circuit 7448)

1. Présentation du 7448

- Les afficheurs 7 segments sont couramment utilisés pour représenter les chiffres décimaux sous forme visuelle.
- Le circuit 7448 est un décodeur BCD-7 segments qui convertit un nombre en code BCD (4 bits) en un affichage numérique sur un afficheur 7 segments.
 - Il possède 4 entrées (A, B, C, D) pour le code BCD.
 - Il contrôle 7 sorties (a, b, c, d, e, f, g) qui commandent l'éclairage des segments d'un afficheur 7 segments.



Brochage d'un afficheur 7 segments commandé par un décodeur

2. Types d'affichage :

Deux types d'afficheurs :

- Afficheur à cathode commune (CC) : les cathodes de toutes les LEDs sont connectées ensemble et reliées à 0 volt.
- Afficheur à anode commune (AC) : les anodes de toutes les LEDs sont connectées ensemble et reliées à Vcc.

Remarque : Pour ce TP, nous utiliserons un afficheur à cathode commune avec le 7448.

3. Travail à faire :

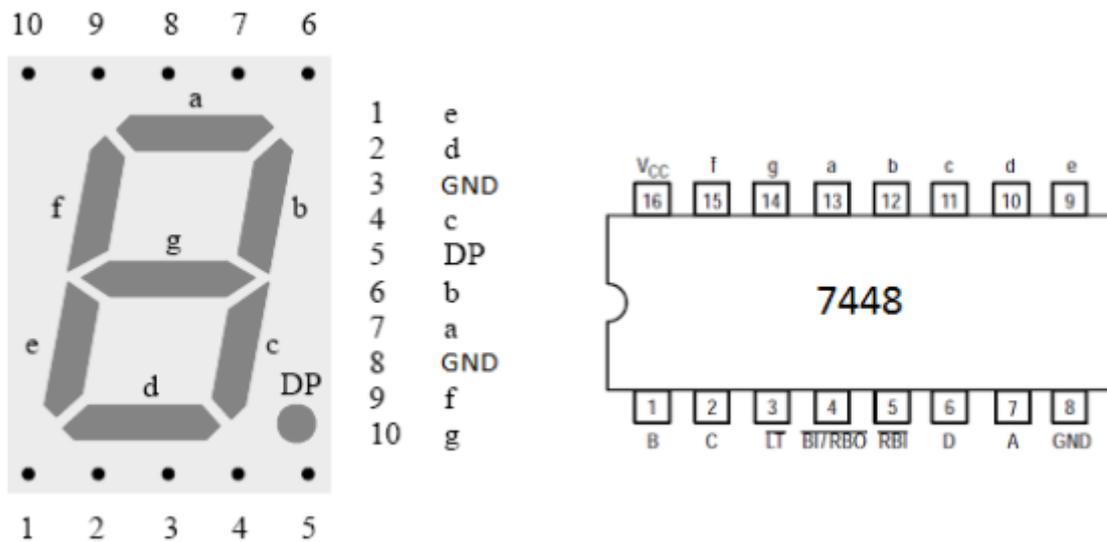


Fig 4. Afficheurs 7 segments, décodeur à cathode commune

1. Câbler le décodeur 7448 avec un afficheur 7 segments en respectant les connexions des segments : (fig 4).
 - Connecter les entrées A, B, C, D 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 3 et observer les sorties de l'afficheur 7 segments. Est-il possible d'afficher le nombre 10 et comment ?
3. Mettre « LT » à « 0 » et garder « RBI » à l'état « 1 ». Refaire l'étape 2. Quelles sont les différences par rapport à l'étape 2 ?

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.

| D | C | B | A | a | b | c | d | e | f | g | Affichage |
|---|---|---|---|---|---|---|---|---|---|---|-----------|
| 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 3