

TD d'application sur les fonctions logiques

EX1. Etude d'un système technique à base d'opérateurs logiques

Descriptif du fonctionnement attendu du système :

L'étude porte sur un treuil électrique. Ce treuil fonctionne « électriquement » dans les deux sens. Il est commandé par deux boutons poussoirs [enroulement] et [déroulement] et possède un interrupteur général de mise en marche. Un dispositif de sécurité permet de détecter les défauts comme par exemple une surcharge mécanique du treuil, une position de fin de course (butée) etc...

Descriptif des variables logiques associées aux entrées :

- **Marche [M]** : variable d'entrée qui est à l'état logique « 1 » si le système est mis en marche par l'utilisateur.
- **Enroulement [E]** : variable d'entrée qui est à l'état logique « 1 » si l'utilisateur actionne le BP d'enroulement du treuil.
- **Déroulement [D]** : variable d'entrée qui est à l'état logique « 1 » si l'utilisateur actionne le BP de déroulement du treuil.
- **Sécurité [S]** : variable d'entrée qui est à l'état logique « 1 » si un défaut est détecté.

Descriptif des variables logiques associées aux sorties :

- **Sens 1 [S1]** : variable de sortie qui est à l'état logique « 1 » lorsque l'on autorise l'enroulement du treuil.
- **Sens 2 [S2]** : variable de sortie qui est à l'état logique « 1 » lorsque l'on autorise le déroulement du treuil.
- **Voyant [V]** : variable de sortie qui est à l'état logique « 1 » lorsque l'on détecte un défaut de fonctionnement ou des ordres de fonctionnement contradictoires.

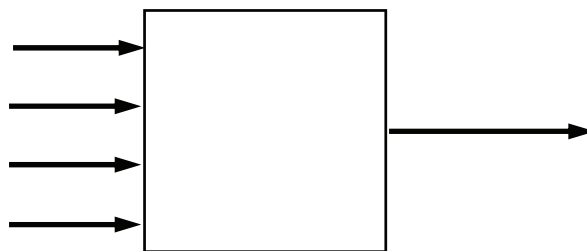
On souhaite réaliser une structure électronique à base d'opérateurs logiques qui permet de *piloter* le moteur du treuil.

Travail à effectuer :

- > Compléter le schéma ci-dessous
- > Etablir la table de vérité de fonctionnement de ce système.
- > Etablir les équations logiques simplifiées des trois sorties.
- > Tracer les logigrammes correspondants aux sorties.

Question subsidiaire :

- > Montrer qu'il est possible de n'utiliser que des opérateurs **ET-NON** pour réaliser ces logigrammes.
- > Intérêt ?



EX2. Comparateur binaire à 2 bits

On souhaite réaliser un comparateur binaire à 2 bits.

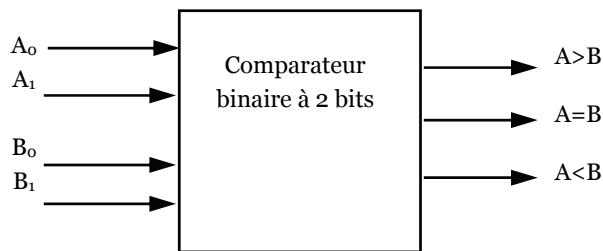
Soit deux nombres binaires A et B ayant 2 bits chacun respectivement (A_1, A_0) et (B_1, B_0) .

On souhaite obtenir en sortie de la fonction logique, le résultat de la comparaison entre ces deux nombres binaires sous la forme

Si $A > B \Rightarrow$ Sortie « $A > B$ » = 1

SI $A = B \Rightarrow$ Sortie « $A = B$ » = 1

SI $A < B \Rightarrow$ Sortie « $A < B$ » = 1



Travail à effectuer :

-> Etablir la table de vérité de fonctionnement de cette fonction (les 3 sorties en fonction des 4 variables binaire d'entrée).

-> Etablir les équations logiques simplifiées des trois sorties.

-> Tracer les logigrammes correspondants aux sorties.

EX3. Nombres Pairs / Impairs

On souhaite réaliser un circuit logique qui affiche une succession de nombres pairs de 0 à 15.

Principe :

En entrée un nombre binaire sur 4 bits (N_3, N_2, N_1, N_0) .

En sortie, si le nombre est PAIR :

-> ce même nombre sous forme d'un quartet.

-> la sortie Nb_Pair à « 1 ».

En sortie, si le nombre est IMPAIR :

-> la sortie « nb_Impair » à « 1 ».

Travail à effectuer :

-> Etablir la table de vérité de fonctionnement de cette fonction. (les 6 sorties en fonction des 4 variables binaire d'entrée).

-> Etablir les équations logiques simplifiées des sorties.

-> Tracer les logigrammes correspondants aux sorties.

EX4. Codeur Binaire -> BCD

On souhaite réaliser un circuit logique qui convertit un nombre binaire sous 4 bits en son équivalent codé en BCD.

Travail à effectuer :

- > Indiquer la valeur binaire maximale qu'il est possible de coder avec un nombre binaire sous 4 bits ?
- > Quel sera alors le format du nombre en sortie du codeur ?
- > Etablir la table de vérité de ce codeur.
- > Etablir les équations logiques simplifiées des sorties.
- > Tracer les logigrammes correspondants aux sorties.

EX5. Codeur BCD -> XS3

On souhaite réaliser un circuit logique qui convertit un nombre BCD (4 bits) en son équivalent codé en code eXceS3.

Remarque : Le code XS3 s'obtient à partir de la valeur décimale du nombre en rajoutant la valeur 3

Exemple : $(0000)_2 = (0011)_{XS3}$

Travail à effectuer :

- > Etablir la table de vérité de ce codeur.
- Nb : - Entrées notées A B C D, sorties notées S3 S2 S1 S0
- Les valeurs indifférentes seront notées X (elles valent indifféremment « 1 » ou « 0 »)
- > Tracer le logigramme correspondant à la sortie S3 en n'utilisant que des opérateurs ET-NON à 2 entrées.
 - > Tracer le logigramme correspondant à la sortie S3 en n'utilisant que des opérateurs OU-NON à 2 entrées.

EX6. Simplification d'équations

Simplifier les équations suivantes (méthode algébrique ou Tableaux de Karnaugh)

$$S1 = a b c d + a b c \bar{d} + a b \bar{c} d + \bar{a} b \bar{c} d + \bar{a} b c d$$

$$S2 = \bar{a} \bar{b} \bar{c} \bar{d} + \bar{a} \bar{b} c \bar{d} + a \bar{b} \bar{c} \bar{d}$$

$$S3 = a b (\bar{c} \bar{d} + \bar{c} d + c \bar{d} + c d) + \bar{a} b \bar{c} \bar{d}$$

$$S4 = a b (d) + a b (c + \bar{d})$$