

Licence d'Education- Informatique

Travaux dirigés de l'électronique numérique

Série n°3: Circuits logiques combinatoires

Comparateur

Exercice 1 :

On veut concevoir un circuit permettant de comparer deux nombres A et B de 4 bits, $A = a_3a_2a_1a_0$ et $B = b_3b_2b_1b_0$. Le circuit a deux sorties : S valant 1 si $A > B$ et E valant 1 si $A = B$. Pour ce faire, on doit réaliser d'abord un comparateur d'un seul bit. Soit a et b deux nombres de 1 bit. Le comparateur à 1 bit doit avoir deux sorties : s valant 1 si $a > b$ et e valant 1 si $a = b$.

1. établir les expressions logiques de s et e et donner le circuit correspondant.

En utilisant ce comparateur à 1 bit, on peut implémenter le comparateur à 4 bits. Il y a deux possibilités :

a. La mise en parallèle des comparateurs 1 bit

2. En utilisant le comparateur de la question précédente ainsi que des portes logiques, concevez un circuit permettant de comparer deux nombres de 4 bits.

b. La mise en cascade des comparateurs 1 bit

Pour cette solution, il faut disposer d'un comparateur cascadeable. Pour ce faire, on ajoute au comparateur de 1 bit deux entrées en cascade s_{i-1} et e_{i-1} , destinées à recueillir les valeurs des sorties s et e du comparateur 1 bit précédent, c.-à-d. le résultat de la comparaison des bits du rang inférieur.

3. Déterminer les équations logiques du comparateur à 1 bit cascadeable.

4. Décrire schématiquement l'architecture d'un comparateur 4 bits à partir de ce comparateur 1 bit cascadeable.

5. Comparer entre les deux comparateurs 4 bits en termes de nombres de portes et temps de propagation. On suppose que le passage d'un signal électrique dans une porte nécessite 10 nanosecondes.

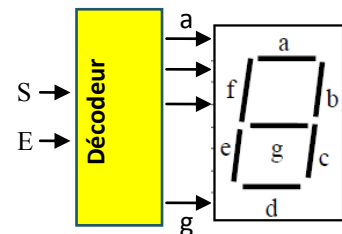
Décodeur/codeur

Exercice 2

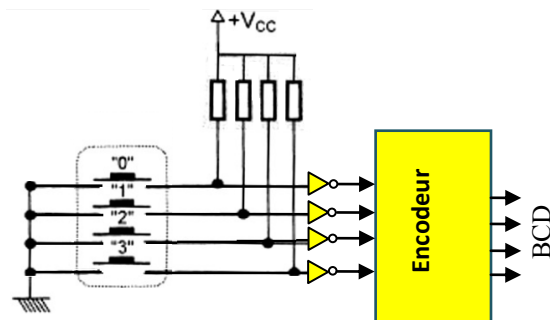
On souhaite afficher le résultat de la comparaison de l'exercice 1 sur un afficheur 7 segments à cathode commune.

1. Concevoir un décodeur qui permet de commander l'afficheur pour afficher :

- S (supérieur) si $A > B$,
- E (égal) si $A = B$
- I (inférieur) si $A < B$.



Pour bien comprendre l'utilité d'un codeur, on donne l'exemple d'application ci-contre. On désire réaliser un codeur qui permet de donner le code binaire des chiffres décimaux inscrits sur des interrupteurs. Si on appuie sur un interrupteur, on doit obtenir en sortie le code BCD du chiffre correspondant. Les 4 interrupteurs peuvent être ceux d'un clavier d'une machine (on peut ajouter d'autres interrupteurs pour réaliser, par exemple, le clavier d'une calculatrice). Lorsque les interrupteurs sont ouverts, les entrées des inverseurs sont au niveau haut par l'intermédiaire des résistances de tirage.



L'appui sur un interrupteur entraîne l'application d'un niveau haut sur l'entrée du codeur correspondante.

2. Donner le logigramme optimal de ce codeur après avoir établi sa table de vérité.

Le défaut de ce codeur se situe dans le fait que, si on appuie accidentellement sur 2 interrupteurs au même temps, les sorties sont erronées. Pour éviter ce problème, on peut utiliser un codeur de priorité qui donne aux entrées des priorités différentes. Considérons par exemple un codeur qui donne la priorité à l'interrupteur de chiffre le plus grand.

3. Établir sa table de vérité. En déduire ses fonctions simplifiées puis son logigramme.

Multiplexeur/démultiplexeur

Exercice 3

On souhaite réaliser un multiplexeur $2 \rightarrow 1$.

1. Après avoir établi sa table de vérité, donner l'expression de la sortie Y en fonction des entrées E_1 et E_2 et de sélection C. En déduire son logigramme.
2. A l'aide de ce multiplexeur, monter comment peut-on utiliser l'additionneur pour effectuer l'addition et la soustraction à la fois.
3. On dispose de 2 multiplexeurs $2 \rightarrow 1$, avec lesquels on aimerait réaliser un multiplexeur $4 \rightarrow 1$. Montrer comment faut-il les assembler.

Générations de fonctions logiques

Exercice 4

La fonction majorité est une fonction qui vaut 1 si le nombre des 1 est supérieur au nombre des 0, et vaut 0 dans le cas inverse. On désire réaliser cette fonction pour des données sur 3 bits.

1. Après avoir établi la table de vérité de cette fonction, donner son expression sous forme SDP canonique.
2. Réaliser cette fonction logique avec un multiplexeur $8 \rightarrow 1$.
3. Peut-on optimiser la réalisation de cette fonction en utilisant un multiplexeur $4 \rightarrow 1$.
4. Réaliser la fonction à l'aide d'un démultiplexeur $1 \rightarrow 8$
5. Réaliser la fonction en utilisant un décodeur $3 \rightarrow 8$. Comparer le décodeur et le démultiplexeur.

Réalisation d'une unité arithmétique logique UAL simple

Exercice 5

L'UAL est un circuit logique combinatoire formé de portes logiques prenant deux nombres en entrée et générant un nombre en sortie en fonction de signaux de commande indiquant l'opération arithmétique ou logique à effectuer. Une UAL peut être caractérisée par sa taille, ses possibilités et sa vitesse. Sa taille correspond au nombre maximum de bits que l'UAL peut traiter. Les possibilités de l'UAL correspondent aux différentes opérations qu'elle peut effectuer (opérations logiques, addition, soustraction, ...etc.).

Dans cet exercice, on désire réaliser une UAL 1 bit simplifiée qui est capable de calculer le ET, le OU et la somme de 2 bits d'entrée, le NON du second bit. Le choix parmi ces quatre opérations se fait via deux lignes de commandes f_0 et f_1 .