

# Travaux Dirigés d'Electronique Numérique

Licence d'Electronique

Hiver 2007



## TD 1 : Algèbre de Boole

---

### 1. Algèbre de Boole

Démontrer les lois de l'algèbre de Boole suivantes en vous aidant des autres lois de cet algèbre :

- $a.1 = a$
- $a+0 = a$
- $a.(a+b) = a$
- $a+(a.b) = a$
- $\overline{a.b} = \bar{a} + \bar{b}$
- $\overline{a+b} = \bar{a}.\bar{b}$

### 2. Démontrer que :

- $a \oplus 1 = \bar{a}$
- $a \oplus b \oplus (a.b) = a + b$
- $a \oplus a = 0$

### 3. Mettez sous forme canonique, disjonctive et conjonctive, les fonctions $s$ dont les tables de vérités sont données par les tables 1.

a	b	c	s
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

a	b	c	s
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Table 1: tables de vérité

### 4. Simplifiez les expressions suivantes sans utiliser les tableaux de Karnaugh :

- $s = a.b.c + \overline{a.b}.c + \bar{a}.b.c + a.\bar{b}.c$
- $s = \overline{a.b}.c + a.b.c + \overline{a.b} + c$
- $s = a.b.c.d + \bar{a}.b.c.d + a.\bar{b}.c.d + a.b.\bar{c}.d + \bar{a}.\bar{b}.c.d + a.\bar{b}.\bar{c}.d + \bar{a}.\bar{b}.\bar{c}.d + \bar{a}.b.\bar{c}.d$

## TD 1 : Algèbre de Boole

- $s = b.d + c.d + \bar{c}.d + \bar{a}.b.\bar{c}.d + \bar{a}.b.c$
- $s = \bar{a}.c + a.b.\bar{c} + a.b.d + a.b.c.\bar{d}$

5. Ecrire les équations booléennes simplifiées issues des tableaux de Karnaugh de la figure 1.

a	b	0	0	1	1
c	d	0	1	1	0
0	0	1			
0	1		1		
1	1		1	1	X
1	0			1	X

a	b	0	0	1	1
c	d	0	1	1	0
0	0	1	1		X
0	1				X
1	1		1		X
1	0				X

a	b	0	0	1	1
c	d	0	1	1	0
0	0		1		
0	1		1		1
1	1		1		1
1	0			1	1

a	b	0	0	1	1
c	d	0	1	1	0
0	0	1			1
0	1	1	1	1	
1	1		1	1	
1	0	1			1

Figure 1: Tableaux de Karnaugh

6. A l'aide de Karnaugh simplifiez les expressions suivantes

- $s = a.b.c + \bar{a}.\bar{b}.c + \bar{a}.b.\bar{c} + a.\bar{b}.\bar{c}$
- $s = a.b.\bar{c} + \bar{a}.\bar{b}.c$  et sachant que la combinaison  $a = b = c = 1$  ne se produit jamais

7. Le feu de croisement

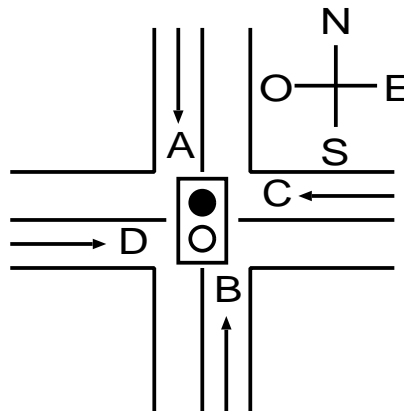


Figure 2: Feu de croisement

La figure 2 montre l'intersection entre une route principale et une route secondaire. Des capteurs de voitures ont été placés le long des voies C et D (route principale) et des voies A et B (route secondaire). Les sorties de ces capteurs sont à l'état logique 0 quand il n'y a pas de voitures et à l'état logique 1 quand il y en a.

## TD 1 : Algèbre de Boole

Le feu de circulation se trouvant à cette intersection est commandé par les règles suivantes:

- Le feu E-O est vert quand il y a des voitures dans les deux voies C et D.
- Le feu E-O est vert quand il y a des voitures dans C ou D et quand il y en a dans A ou B (ou pas du tout) mais pas dans les deux.
- Le feu N-S est vert quand il y a des voitures dans les voies A et B et qu'il y en a dans C ou dans D mais pas dans les deux.
- Le feu N-S est aussi vert quand il y a des voitures dans A ou B et qu'il n'y a pas de voitures dans C et D.
- Le feu E-O est vert quand il n'y a pas de voiture du tout.

On se propose, en utilisant les sorties logiques des capteurs A,B,C et D comme entrées, de concevoir un système numérique qui commande le feu de circulation. Ce circuit a deux sorties, E-O et N-S, qui prennent la valeur logique 1 quand le feu doit être vert.



Figure 3: Chronogramme à compléter

- Etablir la table de vérité pour les sorties E-O et N-S.
- Dessinez les tableaux de Karnaugh pour les sorties E-O et N-S.
- A partir des tableaux de Karnaugh, établissez l'équation logique la plus simple possible pour chacune des sorties.
- Dessinez le schéma logique pour les sorties E-O et N-S avec les entrées A,B,C et D.
- Compléter les chronogrammes de la figure 3

## TD 2 : Codage

---

### 1. Codage Base 2

Coder en base 2 les nombres décimaux suivants:

- 1024
- 345
- 12

### 2. Codage Base 16

Coder en base 16 les nombres binaires suivants:

- 100101
- 11110000
- 10101010

### 3. Codage Base 16

Coder en base 16 les nombres décimaux suivants:

- 10
- 16
- 458

### 4. Codage DCB

Coder en DCB les nombres

- 5
- 55
- 128

Comparer ces nombres avec leurs homologues codés en binaire standard.

### 5. Codage ASCII : nombres

Sachant que le code ASCII de 0 vaut  $30_H$  coder en ASCII les nombres

- 31
- 3
- 458

### 6. Codage ASCII : lettres

Sachant que le code ASCII de  $a$  vaut  $61_H$  coder en ASCII les mots

## TD 2 : Codage

---

- jour
- nuit
- unité

### 7. Codage en complément à 2

Coder en complément à deux sur 8 bits les nombres suivants:

- 34
- -23
- -128
- 0

### 8. Doigt vers Binaire

Lorsque les enfants sélénites, qui n'ont que 4 doigts (le majeur, l'annulaire, l'auriculaire et le pouce), apprennent à compter sur les doigts, la notion d'ordre n'est pas leur priorité. Cela a pour conséquence que, pour eux, lever en même temps le majeur et l'annulaire ou bien l'auriculaire et le pouce produit le même résultat, à savoir ici la représentation du nombre 2. Si ils lèvent simultanément le pouce, le majeur et l'auriculaire ou bien l'annulaire, le pouce et le majeur cela représentera pour eux le même nombre, à savoir 3.

Il vous ait demandé ici de réaliser un circuit qui a en entrée 4 bits représentant les 4 doigts de la main d'un enfant sélénite et en sortie N bits codant l'information représentée par ces doigts en base 2.

Un bit en entrée du circuit est à 1 lorsque le doigt correspondant est levé, sinon il est à 0.

- Déterminez le nombre N de bits nécessaire en sortie du circuit.
  - Dressez la (ou les) table(s) de vérité réalisant ce codage.
  - A l'aide de tableaux de Karnaugh, déterminez les équations booléennes minimales de toutes les sorties du circuit.
  - Faites le schéma du circuit.
9. Code binaire naturel et binaire réfléchi (code Gray) Pour des entiers naturels allant de 0 à 7, représenter sur deux colonnes adjacentes le codage de ces entiers en binaire naturel (cba) et en binaire réfléchi ou code Gray (gba). Exprimer et simplifier les variables logiques g, b et a en fonction des variables c, b et a. En déduire une méthode simple de transcription d'un chiffre codé en binaire naturel en code binaire réfléchi. Même questions en intervertissant gba et cba.

## TD 3 : Fonctions Logiques Complexes

---

### 1. Etude d'un comparateur logique

On veut comparer 2 mots de 4 bits  $A$  et  $B$ . Si on considère les deux bits de même poids  $A_i$  et  $B_i$ , trois solutions sont possibles : soit  $A_i$  est plus grand que  $B_i$ , soit  $A_i$  est plus petit que  $B_i$ , soit ils sont égaux.

- Donner l'expression des fonctions logiques  $E_i, G_i$  et  $P_i$  exprimant respectivement le fait que  $A_i = B_i$ ,  $A_i > B_i$  et  $A_i < B_i$ . On cherchera pour  $E_i$  une expression simple. Le comparateur étudié comporte, outre les 2 mots de 4 bits à comparer, 3 entrées qui résultent d'une éventuelle comparaison de bits de poids inférieur. Ces entrées sont les suivantes :
  - $(A < B)_{in}$  indique que la comparaison des bits de poids inférieur donne  $A < B$
  - $(A > B)_{in}$  indique que la comparaison des bits de poids inférieur donne  $A > B$
  - $(A = B)_{in}$  indique que tout les bits de poids inférieur sont égaux.

On admettra qu'une seule de ces entrées peut être active.

- Exprimer en fonction des fonctions  $E_i, G_i$  et  $P_i$  définies plus haut, les différentes possibilités qui imposeront un niveau actif sur la sortie  $(A > B)_{out}$  puis sur la sortie  $(A < B)_{out}$ . On pourra considérer les bits de poids le plus fort puis descendre jusqu'aux bits de poids faible.
- Donner l'expression de la sortie  $(A = B)_{out}$ .  
On désire maintenant étudier la génération de la sortie  $(A > B)_{out}$ . La démarche pour réaliser cette fonction peut s'exprimer de la manière suivante :  $A$  est plus grand que  $B$  si  $A$  n'est pas plus petit que  $B$  et que  $A$  n'est pas égal à  $B$ .
- Donner l'expression de  $(A > B)_{out}$  en adoptant cette démarche.
- Décrivez en VHDL un composant qui compare 2 mots de 4 bits et qui répond à la description précédente.
- Comment peut on associer 2 circuits de ce type pour comparer 2 mots de 8 bits?
- Décrivez en VHDL un composant qui compare 2 mots de 8 bits en utilisant des comparateurs 4 bits.

### 2. Etude d'un décodeur

Ce circuit permet, lorsque les signaux de validation sont actifs, de sélectionner une sortie  $Y_i$  parmi 8 suivant la valeur des bits de sélection  $C, B$  et  $A$ . La table de vérité du décodeur que nous désirons étudier est la suivante :

## TD 3 : Fonctions Logiques Complexes

$\overline{G_1}$	$\overline{G_2}$	$G_3$	A	B	C	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$
1	X	X	X	X	X	1	1	1	1	1	1	1	1
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	0	X	X	X	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1
0	0	1	1	0	0	1	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	1	1	1
0	0	1	1	1	0	1	1	1	0	1	1	1	1
0	0	1	0	0	1	1	1	1	1	0	1	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	1
0	0	1	0	1	1	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0

- Quels sont les niveaux des signaux de validation  $G_1$  et  $G_2$  qui vont imposer un niveau en sortie rendant toutes les sorties  $Y_i$  inactives ( les sorties  $Y_i$  sont actives au niveau bas ).
- En admettant maintenant que tout les signaux de validation sont inactifs, donner par exemple l'expression de la sortie  $Y_5$  déduite de la table de vérité.
- Réalisez le schéma logique de cette fonction.
- Décrivez en VHDL un composant qui permette de calculer la sortie  $Y_5$  en fonction des entrées  $A, B, C, G_1, G_2$  et  $G_3$ .



## TD 4 : Arithmétique Combinatoire

---

### 1. Additionneurs Soustracteurs

On désire réaliser un montage effectuant l'addition de 2 mots de 4 bits  $A$  et  $B$  en tenant compte d'une éventuelle retenue  $C_{in}$ . Le résultat de cette opération est disponible sur  $S$  qui est un mot de 4 bits et sur  $C_{out}$  qui est la retenue de cette addition.

#### (a) Additionneur en tranche de 1 bit

- La première solution consiste à mettre en cascade plusieurs cellules additionneur 1 bit avec retenue. Les entrées d'une telle cellule sont les 2 bits des mots à additionner  $A_i$  et  $B_i$ , et la retenue de l'addition des bits de poids inférieur  $C_i$ . Les deux sorties sont la somme de ces 3 bits  $S_i$  et la retenue  $C_{i+1}$ .
- Exprimer les fonctions logiques  $S_i$  et  $C_{i+1}$  en fonction de  $A_i$ ,  $B_i$  et  $C_i$ .
- Dessiner le schéma complet d'un additionneur 1 bit avec retenue.
- Si on considère qu'une couche logique a un temps d'établissement  $T$  entre le moment où ses entrées sont stables et le moment où sa sortie est valide, donner le temps de calcul de la somme  $S_i$  et de la retenue  $C_{i+1}$ .
- Donner le schéma d'un additionneur 4 bits complet à partir de cellule additionneur 1 bit.
- Quel est alors le temps d'établissement de  $S_0$ ,  $S_1$ ,  $S_2$ ,  $S_3$  et  $C_4$ ?
- Quel est le principal défaut d'une telle conception?
- Décrivez en VHDL un additionneur 1 bit.
- Décrivez en VHDL un additionneur 4 bits à propagation de retenue.

#### (b) Additionneur avec anticipation de retenue

Pour palier au défaut du montage précédent, on décide de changer la conception de notre additionneur 4 bits. La démarche est la suivante : chaque retenue est directement calculée à l'aide des variables d'entrées sans attendre la propagation de la retenue de l'addition des bits de poids inférieur.

- Donner l'expression de  $C_1$ ,  $C_2$ ,  $C_3$  et  $C_4$  en fonction des entrées  $A_i, B_i$  et  $C_0$ .
- Sachant que l'on peut utiliser des portes logiques à entrées multiples, quel est le temps d'établissement des  $S_i$  et des  $C_i$ ?

### 2. Additionneur 16 bits

On réalise un additionneur 16 bits à l'aide d'additionneurs 4 bits de la première puis de la deuxième question.

- Comparer les performances des deux additionneurs ainsi réalisés.
- Conclure.

## TD 4 : Arithmétique Combinatoire

---

### 3. Additionneur soustracteur en complément à 2

On désire réaliser un additionneur-soustracteur à l'aide d'un additionneur. Cet additionneur-soustracteur sera contrôlé par une variable extérieure  $C_D$ . Si  $C_D = 0$ ,  $S$  sera le résultat de  $A$  plus  $B$ . Si  $C_D = 1$ ,  $S$  sera égal à  $A$  moins  $B$ .

- Proposer un montage réalisant le complément à 1 de  $B$  commandé par la variable  $C_D$ .
- Après avoir rappelé la différence entre le complément à 1 et la complément à 2, proposer un montage complet d'additionneur-soustracteur commandé par  $C_D$ .

## TD 5 et 6 : Bascules D et Compteurs

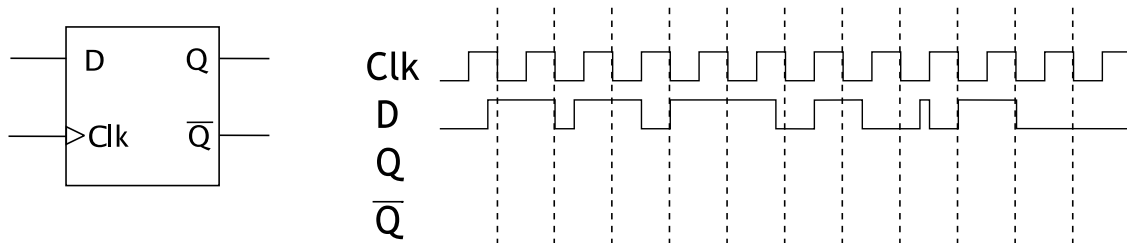


Figure 4: Montage

### 1. Bascules D.

Dessinez sur la figure 4 le chronogramme correspondant aux sorties  $Q$  et  $\bar{Q}$  de la bascule D représentée par le montage de la même figure.

### 2. Compteurs asynchrones.

- Dessinez le schéma d'un diviseur de fréquence par 2 à base de bascules D.
- Dessinez le schéma et le chronogramme de sortie d'un décompteur asynchrone modulo 8 réalisé à partir de bascules D.
- Comment modifier le schéma précédent pour obtenir un compteur asynchrone par 8 ?

### 3. Compteurs synchrones.

On souhaite réaliser un compteur synchrone modulo 5.

- Combien de bascules D sont nécessaires ?
- Etablissez la table de vérité de ce compteur.
- A l'aide de tableaux de Karnaugh, donnez les équations des entrées  $D_i$  des différentes bascules.
- Etudiez l'évolution temporelle du compteur si, lors de la mise sous tension, ce dernier démarre dans les états 6 ou 7.

### 4. Registre à décalage

Un registre à décalage est constitué de  $N$  bascules D mises en série de la manière indiquée sur la figure 5; dans ce cas, le registre comporte 4 bascules, toutes reliées à une horloge commune  $H$  de période  $T$ .

On a accès à l'entrée  $E$ , à la sortie  $S$  et au mot de 4 bit "interne"  $Q = Q_4Q_3Q_2Q_1$  ( $Q_4 = MSB$  et  $Q_1 = LSB$ ). A tout instant, l'état du registre est déterminé par la valeur du mot  $Q$ ; on passe d'un état à un état suivant à chaque front montant d'horloge. Par exemple, de  $Q = 0110$ , on passe à 0011 (si  $E = 0$ ) ou à 1011 (si  $E = 1$ ).

## TD 5 et 6 : Bascules D et Compteurs

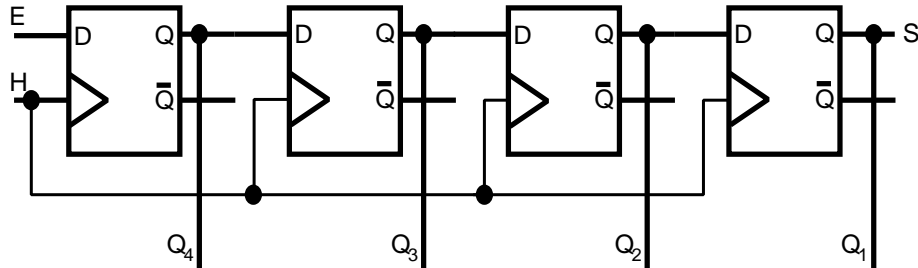


Figure 5: Registre à décalage

- (a) On part de l'état  $Q = 1011$ , et on demande de donner la liste des 5 états suivants dans les 4 cas 4(a)i, 4(a)ii, 4(a)iii et 4(a)iv indiqués ci-dessous. Compléter les tableaux A, B, C et D. Donner la valeur décimale  $V_D$  correspondante à chacun des états obtenus.
- lorsque  $E = 0$
  - lorsque  $E = 1$
  - lorsque  $E = Q_3$
  - lorsque  $E = S$
- (b) Au bout de ces 5 états, quelle est la périodicité observée dans les valeurs successives de  $Q$  dans chacun des cas (en régime permanent) ?

N	$Q_4$	$Q_3$	$Q_2$	$Q_1$	$V_D$
-	1	0	1	1	
1					
2					
3					
4					
5					
tableau A : $E = 0$					

N	$Q_4$	$Q_3$	$Q_2$	$Q_1$	$V_D$
-	1	0	1	1	
1					
2					
3					
4					
5					
tableau B : $E = 1$					

N	$Q_4$	$Q_3$	$Q_2$	$Q_1$	$V_D$
-	1	0	1	1	
1					
2					
3					
4					
5					
tableau C : $E = Q_3$					

N	$Q_4$	$Q_3$	$Q_2$	$Q_1$	$V_D$
-	1	0	1	1	
1					
2					
3					
4					
5					
tableau D : $E = S$					

## TD 5 et 6 : Bascules D et Compteurs

N	E	$Q_4$	$Q_3$	$Q_2$	$Q_1$
-		1	0	1	1
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
tableau E : $E = Q_4 + Q_3 + Q_2$					

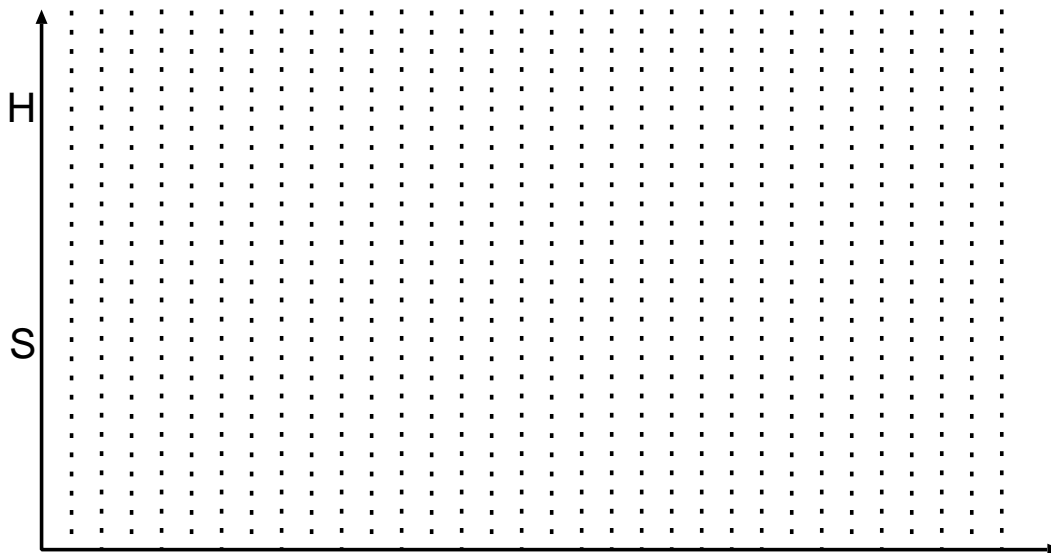


Figure 6: Diagramme de  $H$  et  $S$

- (c) On part toujours de  $Q = 1011$ , mais cette fois-ci on câble avec  $E = \overline{Q_4 + Q_3 + Q_2}$ .
- Faire le schéma logique du montage avec un (ou des) OU à 3 entrées. Donner la série des états obtenus dans le tableau E.
  - Montrer qu'au bout d'un certain nombre d'états (combien ?), il ne reste plus qu'un

## TD 5 et 6 : Bascules D et Compteurs

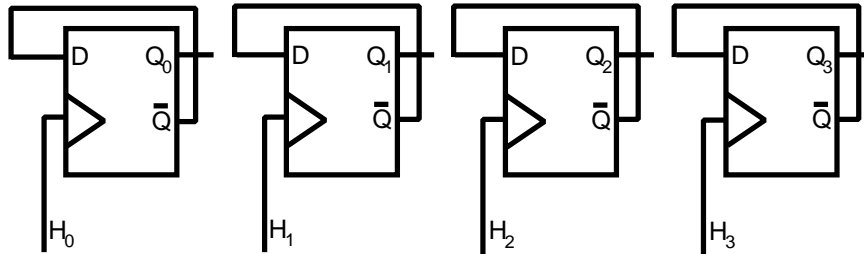


Figure 7: Compteur Modulo 10

seul "1" qui "tourne" dans le registre. Porter alors sur le diagramme de la figure 6 les signaux  $H$  et  $S$ . En régime permanent, quelle est la période de  $S$  ?

- iii. Quelle fonction  $S(H)$  réalise le registre ?
- iv. Quelle serait la fonction réalisée si le registre comportait 5 bascules ? Donner une application possible d'un tel montage.

### 5. compteur modulo 10.

On désire réaliser un compteur modulo 10 à l'aide des 4 bascules D de la figure 7. Chaque bascule est commandée par une horloge  $H_j$  qui sera réalisée à partir d'une horloge principale  $H$ .

Les dix états du compteur sont définis par

$$\sum_{k=0}^3 Q_k 2^k = 0, 1, \dots, 9$$

et leur succession temporelle s'effectue dans l'ordre 0, 1, 2, ..., 9, 0, 1 etc.. Le signal d'horloge  $H_j$  de la  $j$ ème bascule est  $H_j = H + K_j$  où  $H$  est le signal d'horloge du compteur qui est actif sur le front montant.

- (a) Quelles sont les valeurs des différentes grandeurs logiques  $K_j$  pour les dix états du compteur ?
- (b) Donner les expressions des  $K_j$  en fonction des sorties des bascules  $D$  à l'aide des diagrammes de Karnaugh. Pour la grandeur  $K_3$ , on trouve trois solutions notées  $K_3$ ,  $K_3'$ ,  $K_3''$ . Etudier l'évolution temporelle du compteur si lors de la mise sous tension ce dernier démarre dans l'un des états suivants : 10, 11, 12, 13, 14, 15.
- (c) Donner ensuite un schéma de câblage du compteur à l'aide de portes NAND et OU.
- (d) Décrivez en VHDL ce compteur.

## TD 7 : Arithmétique pipelinée

### 1. Multiplieur

- Etablissez la table de vérité d'un multiplieur 1 bit ayant en entrée deux variables a et b et en sortie une variable p
- A l'aide de multiplieurs 1 bit et d'additionneurs 1 bit, réalisez le schéma d'un multiplieur 3 bits ayant en entrée une variable A et une variable B de 3 bits chacune et une variable P de 6 bits. On utilisera ici la technique de propagation de retenue.
- Si  $t_p$  est le temps de propagation d'un multiplieur 1 bit et d'un additionneur 1 bit, déterminez le temps de calcul d'une multiplication 3 bits avec le multiplieur précédemment défini.
- Déterminez le temps de calcul de deux multiplications successives avec ce multiplieur.

### 2. Multiplieur Pipeliné

- Pour accélérer le traitement, modifiez le schéma pour incorporer des éléments de mémorisation permettant de réaliser un multiplieur 3 bits pipeliné. On appellera l'horloge *CLK*.
- Déterminez en fonction de  $t_{clk}$  le temps de calcul d'une multiplication.
- Déterminez en fonction de  $t_{clk}$  le temps de calcul de deux multiplications successives.
- Si  $t_{clk}$  est pris proche de  $t_p$ , estimez l'accélération de ce multiplieur par rapport au multiplieur de la question 1
- Généralisez pour un multiplieur  $N$  bits.

### 3. Multiplication Accumulation Pipeliné

On désire réaliser le calcul suivant

$$va = |a - b|$$

va, a et b étant des nombres binaires sur 3 bits.

Pour cela on utilise le montage du schéma de la figure 8.

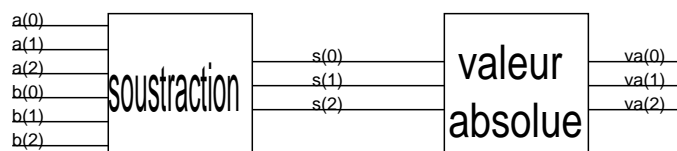


Figure 8: montage calcul valeur absolue

## TD 7 : Arithmétique pipelinée

---

Dans ce montage les temps de propagation de chaque opérateur sont :

- Opérateur Soustraction : 4 ns
  - Opérateur Valeur Absolue : 6 ns
- (a) On désire pipeliner cet opérateur en le découpant en 2 étages, un étage soustraction et un étage valeur absolue. Modifiez le schéma de la figure 8 afin d'obtenir un opérateur pipeliné
- (b) Caculer la fréquence maximale à laquelle pourra être cadencé cet opérateur
- (c) A l'aide d'un multiplexeur 2 vers 1 ayant 2 entrées et une sortie, toutes sur 3 bits, réalisez le schéma d'un opérateur de calcul de valeur absolue



## TD 8 : Technologie

1. Soit le schéma en technologie CMOS de la Figure 9 qui comprend 2 entrées  $A$  et  $B$  et une sortie  $S$ . Déterminer la fonction logique  $S$  de ce circuit

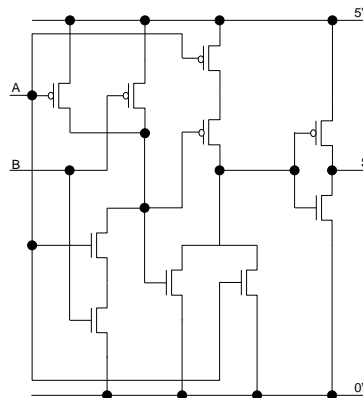


Figure 9: Schéma en Technologie CMOS

2. Réaliser le schéma en technologie CMOS des fonctions logiques

(a)  $S = \overline{A \cdot B} + C$

(b)  $S = A \oplus B$

(c)  $S = \overline{A} + \overline{B} \cdot \overline{C}$

3. Compléter le schéma en technologie CMOS de la figure 10

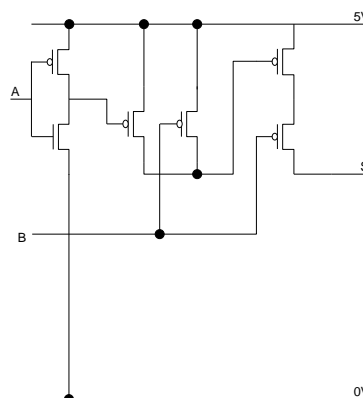


Figure 10: Schéma en Technologie CMOS à compléter

## TD 9 : CAN - CNA

---

### 1. CAN à rampe

On veut convertir une tension continue  $U_0$  variant de 0 à  $U_{0max}$ , en un mot binaire codé sur quatre bits. Pour cela, on compare cette tension à une rampe linéaire. Tant que la rampe reste inférieure à  $U_0$ , on incrémente régulièrement un compteur tout les  $T_0$ . Si la rampe dépasse  $U_0$ , on arrête alors le comptage. Ceci est réalisé par le schéma de la figure 11.

Fonctionnement général

- En quoi ce montage constitue-t-il un convertisseur Analogique - Numérique ?
- Où se trouve le résultat final ?
- Identifier sur la figure les différentes parties précitées.

Un deuxième compteur de la même capacité que le premier assure le séquençement du montage total.

- A quel moment le résultat final doit-il être mémorisé ?
- Quelle partie du montage de la figure réalise le décodage de cet instant ?
- Quelles autres fonctions assure-t-elle ?

Calcul des composants

- Exprimer  $X$  en fonction de  $H_0$  et des  $Q_i$  du compteur  $A$ . Compléter sur la figure 12 le chronogramme de  $X$ .
- Exprimer  $V_+$  en fonction de  $t$ ,  $I_0$ ,  $R_0$ ,  $C_0$  et  $T_0$  (période de l'horloge  $H_0$ ) quand l'interrupteur est ouvert ( $X = 0$ ) et quand il est fermé ( $X = 1$ ).
- En déduire l'expression de  $C_0$  en fonction de  $I_0$ ,  $T_0$ ,  $U_{0max}$  et du nombre de bits  $N$  pour avoir une résolution optimale. A.N :  $I_0 = 200mA$  ;  $T_0 = 10ms$  ;  $U_{0max} = 1V$  et  $N = 4$  bits.
- Quel est alors le temps de conversion ?
- Calculer  $R_0$  pour avoir une tension en fin de décharge inférieure à 1% du maximum atteint.
- Avec les valeurs précédentes, compléter les chronogrammes de la fig 12 pour  $U_0 = 0,61V$ .
- Ecrire la table donnant la valeur en volt déduite de  $NC$  (c'est-à-dire la valeur décodée de  $NC$  lue sur ce voltmètre) en fonction de  $NC$ . Quelle est la précision de ce CAN ?

## TD 9 : CAN - CNA

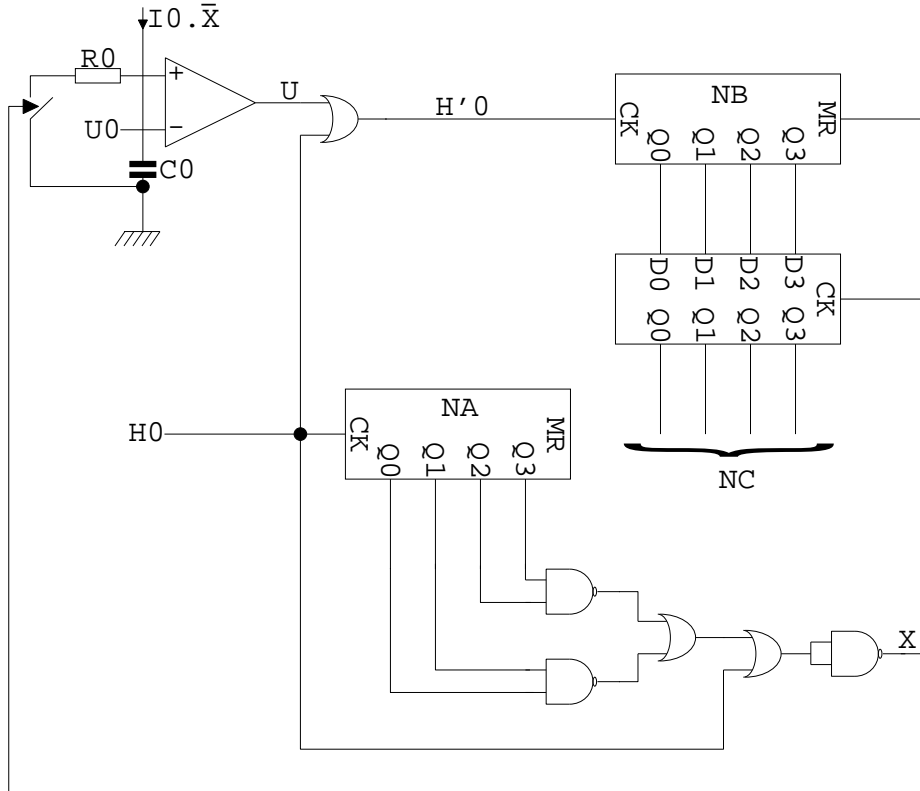


Figure 11: CAN à rampe

### 2. CNA par modification du rapport cyclique

- Rappeler les conditions à vérifier pour qu'un signal logique  $H$  de périodicité  $N * T_0$  et de rapport cyclique  $n/N$  attaquant un filtre RC donne une tension en sortie de ce filtre proportionnelle à  $n/N$  ?
- Expliquer alors le fonctionnement du montage de la figure 13.
- Tracer les chronogrammes de  $H$ ,  $Q_0$  à  $Q_3$ ,  $V_{comp}$  et  $V_s$  dans le cas où le mot à convertir est égal à 1011.
- Comment pourrait-on réaliser, en utilisant le même principe, un Numérique - Analogique sur 8 bits ?
- Quel est le principal défaut de ce montage ?

## TD 9 : CAN - CNA

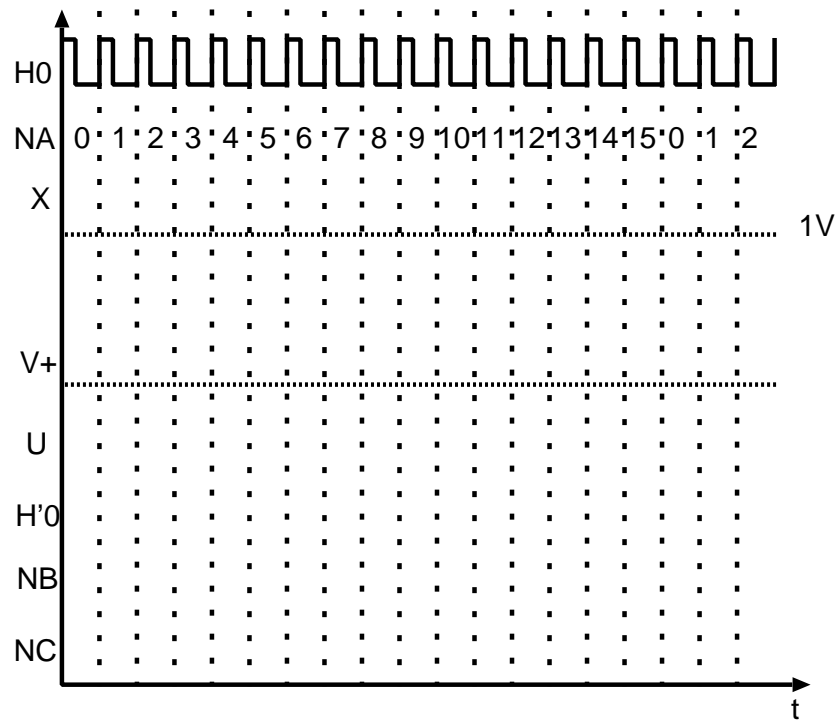


Figure 12: Diagramme CAN à rampe

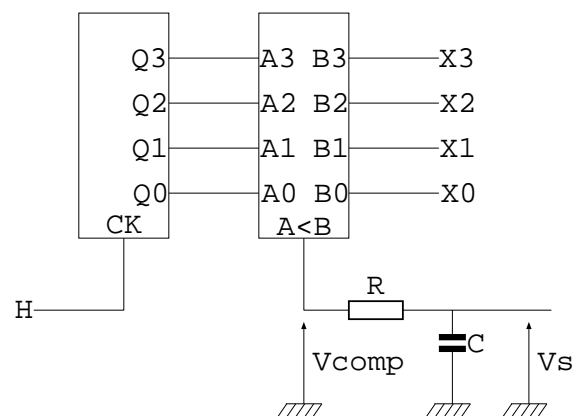


Figure 13: CNA 2