Licence d'informatique – 2001/2002

Architecture Élémentaire: Fiche de TD 1

Logique & Arithmétique Binaire

octobre 2001

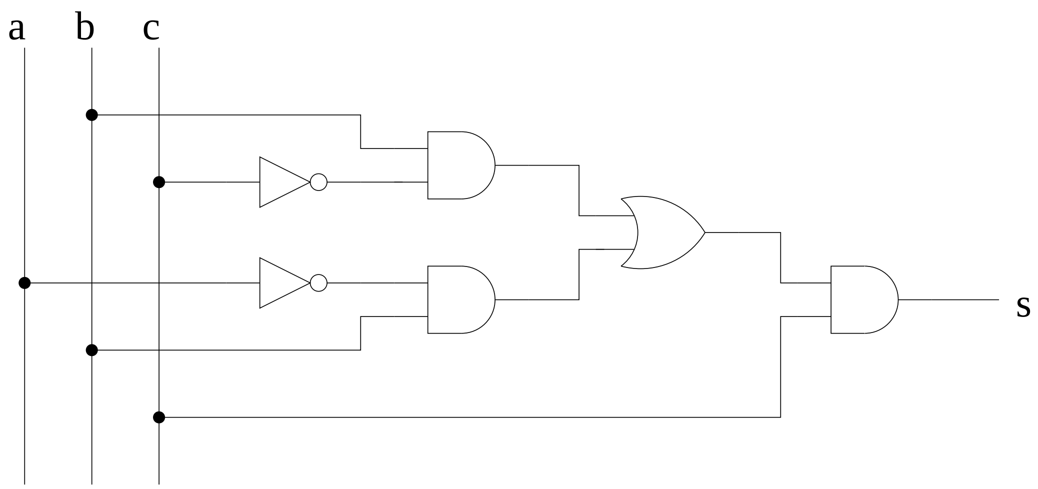
1 Fonctions logiques de base

Question 1.1 : *Remplissez le tableau suivant :*

		NON *NOT*	*OU* *OR*	*ET* *AND*	*NON ET* *NAND*	*NON OU* *NOR*	*OU exclusif* *XOR*	*équivallence* *XNOR*
a	b	$\overline{a}$	$a+b$	$a.b$	$a \uparrow b$	$a \downarrow b$	$a \oplus b$	$a \otimes b$
0	*0*							
0	*1*							
1	*0*							
1	*1*							
		7406	*7432*	*7408*	*7400*	*7402*	*7486*	*74266*

Question 1.2 :

1. *Calculer la table de vérité du circuit logique suivant :*



2. *Donner une expression logique simple pour cette fonction logique.*
3. *Dessiner son circuit logique.*

Question 1.3 : *Montrer, en dessinant les circuits logiques, que toutes les fonctions logiques à deux variables peuvent être exprimées avec uniquement des AND, OR et NOT.*

2 Algèbre de Boole

Les axiomes de l'algèbre de Boole sont notés :

commutativité :	(A1)	$a.b = b.a$	(A2)	$a + b = b + a$
associativité :	(A3)	$a.(b.c) = (a.b).c$	(A4)	$a + (b + c) = (a + b) + c$
distributivité :	(A5)	$a.(b + c) = a.b + ac$	(A6)	$a + (b.c) = (a + b).(a + c)$
élément neutre :	(A7)	$1.a = a.1 = a$	(A8)	$0 + a = a + 0 = a$
complément :	(A9)	$a.\overline{a} = 0$	(A10)	$a + \overline{a} = 1$

Les propriétés sont notées :

élément absorbant :	(P1)	$a.0 = 0$	(P2)	$a + 1 = 1$
absorption :	(P3)	$a.(a + b) = a$	(P4)	$a + (a.b) = a$
idempotence :	(P5)	$a.a = a$	(P6)	$a + a = a$
involution :	(P7)	$\overline{\overline{a}} = a$		
théorème de De Morgan :	(DM1)	$\overline{a.b} = \overline{a} + \overline{b}$	(DM2)	$\overline{a + b} = \overline{a}.\overline{b}$

Question 2.1 : *On considère un ensemble à deux éléments notés "0" et "1", muni de deux opérateurs binaires notés "." et "+" et une fonction unaire "–". Montrer que si les axiomes sont vérifiés alors on a forcément les propriétés qui sont vérifiées.*

Question 2.2 : *On rappelle qu'un système de fonctions logiques est dit complet s'il permet de calculer toutes les fonctions logiques.*

1. *Montrer que {AND, OR, NOT} est complet.*
2. *Montrer que {AND, NOT} et {OR, NOT} sont complets.*
3. *Montrer que {NAND} et {NOR} sont complets.*

Question 2.3 : *Construire les fonctions de la question 1 avec des NAND. En sachant qu'un NAND prend deux transistors, calculer le coût correspondant si l'on n'utilise que des NAND.*
Même question mais avec des NOR...

Question 2.4 : *Montrer en utilisant les formules de l'algèbre de Boole que :*

$$\overline{a}.(b + \overline{b}.c) + a.\overline{b} = \overline{a}.b + (c + a.\overline{c}).\overline{b}$$

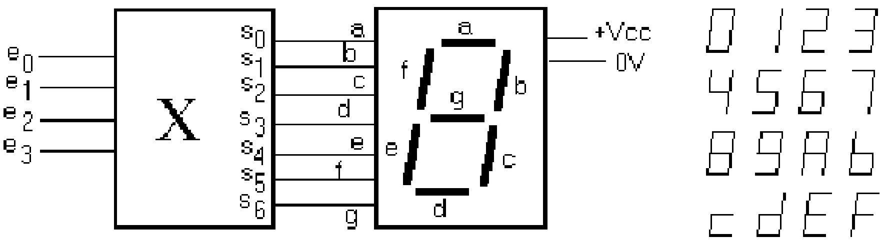
3 Formes normales

Question 3.1 : *Donner les formes normales conjonctives et disjonctives de la fonction*

$$(a.\overline{b} + \overline{c}).(\overline{a} + b)$$

4 Afficheur sept segments

On considère le circuit suivant :



Il est composé de deux parties :

- Un afficheur sept segments avec sept entrées logiques $a, b, \ldots, g$. **Un segment de l'afficheur est allumé si et seulement si l'entrée correspondante est à 0** ;
- Le circuit logique X que l'on doit réaliser. Il prend quatre entrées e_0, e_1, e_2, e_3 et doit donner en les sorties $s_0, s_1, \ldots, s_6$ telles que l'affichage soit le suivant :

e_3	e_2	e_1	e_0	affichage
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	b
1	1	0	0	c
1	1	0	1	d
1	1	1	0	E
1	1	1	1	F

Question 4.1 : *Trouver les tables de vérité de $s_0, s_1, \ldots, s_6$.*

Question 4.2 : *Trouver les expressions logiques pour chacune des sorties et dessiner le circuit complet.*

5 Représentation des nombres

Question 5.1 : *Combien de nombres peut-on coder sur 4, 8, 16 et 32 bits ? Quelles sont les bornes ?*

Question 5.2 : *Écrire les nombres suivants en binaire : 48 ; 10,75 ; 7,6875 et 23,9. Les écrire en hexadécimal.*

6 Nombres binaires en virgule fixe

Question 6.1 : *Remplissez le tableau suivant avec les entiers ou entiers relatifs correspondants ("M&S" signifie "signé module et signe", "Cà1" signifie "signé complément à 1" et "Cà2" signifie "signé complément à 2") :*

	non-signé	*M&S*	*Cà1*	*Cà2*
0000				
0001				
0010				
0011				
0100				
0101				
0110				
0111				
1000				
1001				
1010				
1011				
1100				
1101				
1110				
1111				

Question 6.2 : *Dessiner un circuit logique testant l'égalité de deux nombres binaires de 4 bits en non-signé, M&S, Cà1, Cà2.*

7 Addition

Question 7.1 : *On considère des nombres binaires de 6 bits sans virgule en complément à 2. Effectuez les additions "6+5", "12+21", "5+(-8)" et "(-12)+(-30)".*

Question 7.2 : *Donner la condition pour détecter un overflow lors d'une addition de deux nombres signés en complément à 2.*

Question 7.3 : *Donner le circuit d'un additionneur 4 bits en utilisant des additionneurs élémentaires. Supposons qu'un additionneur élémentaire mette 50 ns pour que la sortie "retenue" soit stable et 90 ns pour que la sortie "somme" soit stable. Combien de temps faut-il pour que le résultat à la sortie de l'additionneur 4 bits soit valide ?*

Question 7.4 : *Ajouter au circuit de la question précédente la fonction de détection d'overflow en complément à 2.*

8 Anticipation de retenue

On définit les fonctions suivantes (dans le cas de l'addition de deux bits a_i et b_i) :

- la génération de retenue : $g_i = a_i.b_i$;

- la propagation de retenue : $p_i = a_i \oplus b_i$;
- la retenue : $c_i = g_i + (p_i.c_{i-1})$.
- la somme : $s_i = p_i \oplus c_{i-1}$;

On veut réaliser un additionneur 3bits utilisant la méthode d'anticipation de retenue.

Question 8.1 : *Donner les formules de s_0, c_0, s_1, c_1, s_2 et c_2 en fonction de g_0, g_1, g_2, p_0, p_1, p_2 et c_e (c_e est la retenue provenant de l'étage précédent).*

Question 8.2 : *Compléter le schéma du circuit de la figure 1 et justifier le gain par rapport à l'additionneur parallèle de la question 7.*

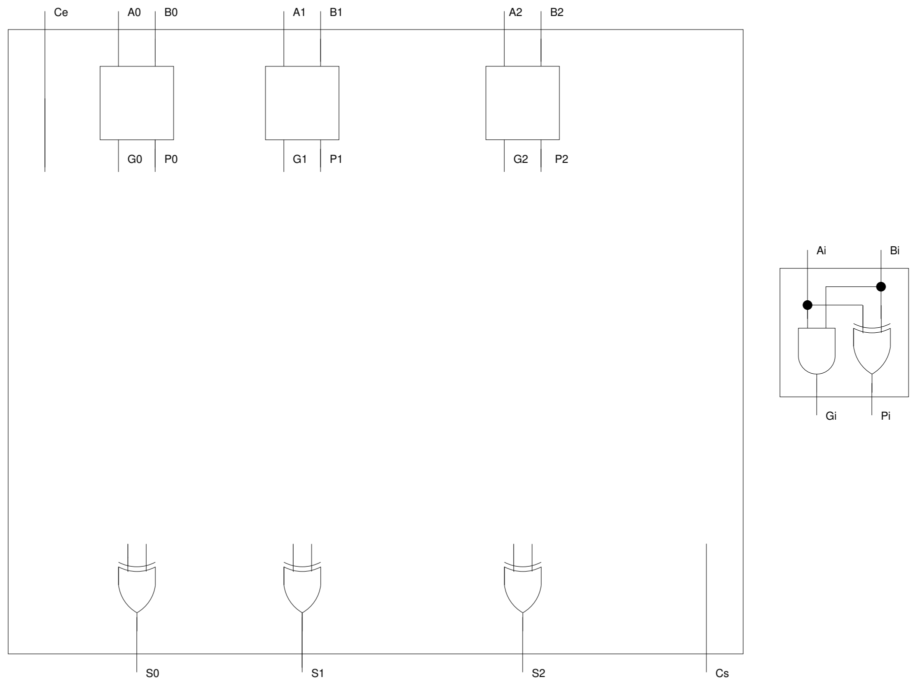


Figure 1: Additionneur binaire en complément à 2 avec anticipation de retenue