



Examen Final (Corrigé-type)

La séance de consultation des copies d'examen aura lieu Mardi à 11h (Amphi C2)

Questions de cours : (5 points)

15 ‘

1. Pour convertir 8 bits il faudrait 3 chiffres en octal ou 2 chiffres en hexadécimal.	01
2. Une bascule qui change d'état uniquement sur un front d'horloge est une bascule <input checked="" type="checkbox"/> Synchrone <input type="checkbox"/> Asynchrone	01
3. Un registre est un ensemble de composants logiques appelés bascules .	01
4. Le circuit qui permet de sélectionner un mot-mémoire d'après son adresse est appelé décodeur .	01
5. Dans un processeur les commandes sont données par le séquenceur (ou bloc logique de commandes) .	01

Exercice 1 : Logique combinatoire (5 points)

30 ‘

Soit un circuit logique avec la table de vérité ci-contre :

- Entrées : a, b, c,
- Sorties : S1, S2 :

a	b	c	S1	S2
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

1) Déterminer les expressions logiques des 2 sorties. $S1 = \bar{a}bc + a\bar{b}c + ab\bar{c} + abc$ $S2 = \bar{a}bc + a\bar{b}c + ab\bar{c} + abc$	02																																																
2) Simplifier les expressions par le tableau de Karnaugh. <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>S1</p> <table border="1"> <tr><td></td><td>ab</td><td>00</td><td>01</td><td>11</td><td>10</td></tr> <tr><td>c</td><td></td><td></td><td>1</td><td></td><td>1</td></tr> <tr><td></td><td>0</td><td></td><td>1</td><td></td><td>1</td></tr> <tr><td></td><td>1</td><td>1</td><td></td><td>1</td><td></td></tr> </table> </div> <div style="text-align: center;"> <p>S2</p> <table border="1"> <tr><td></td><td>ab</td><td>00</td><td>01</td><td>11</td><td>10</td></tr> <tr><td>c</td><td></td><td></td><td></td><td>1</td><td></td></tr> <tr><td></td><td>0</td><td></td><td></td><td>1</td><td></td></tr> <tr><td></td><td>1</td><td></td><td>1</td><td>1</td><td>1</td></tr> </table> </div> </div> <p>Rien à simplifier (pas de bits adjacents). $S2 = ab + bc + ac$</p>		ab	00	01	11	10	c			1		1		0		1		1		1	1		1			ab	00	01	11	10	c				1			0			1			1		1	1	1	02
	ab	00	01	11	10																																												
c			1		1																																												
	0		1		1																																												
	1	1		1																																													
	ab	00	01	11	10																																												
c				1																																													
	0			1																																													
	1		1	1	1																																												
3) Quel est le rôle de ce circuit ? Ce circuit est un additionneur complet de 2 bits a et b et une retenue c. La sortie S1 représente la somme et S2 la nouvelle retenue.	01																																																



Exercice 2 : Processeurs (5 points)

15 ‘

Dérouler le segment suivant dans les 2 modes sachant que :

$$[ACC] = 200 ; [40] = 120 ; [120] = 10 ; [10] = 40$$

	Immédiat	Direct	
10 ADD 40	$[Acc] = [Acc] + 40 = 200 + 40 = 240$	$[Acc] = [Acc] + [40] = 200 + 120 = 320$	01
11 SUB 120	$[Acc] = [Acc] - 120 = 240 - 120 = 120$	$[Acc] = [Acc] - [120] = 320 - 10 = 310$	01
12 MPY 10	$[Acc] = [Acc] * 10 = 120 * 10 = 1200$	$[Acc] = [Acc] * [10] = 310 * 40 = 12400$	01
13 DIV 120	$[Acc] = [Acc] / 120 = 1200 / 120 = 10$	$[Acc] = [Acc] / [120] = 12400 / 10 = 1240$	01
14 STR 40	$[40] = [Acc] = 10$	$[40] = [Acc] = 1240$	01

Exercice 3 : Mémoires (5 points)

30 ‘

On considère une machine avec la configuration suivante :

- Mémoire centrale de taille 1 MOctets
- Mot Mémoire de taille 2 Octets
- Bus d'adresse (ou registre d'adresse) de taille 20 bits.

Remplir le tableau suivant (**formule + résultat**) :

1) La taille minimale du bus de données = Taille du mot-mémoire = 16 bits	01
2) La taille minimale du bus d'adresses = $\log_2 \left(\frac{C}{n} \right) = \log_2 \left(\frac{2^{20}}{2} \right) = \log_2 (2^{19}) = 19$ bits	01
3) La plage d'adressage de cette mémoire = $[0 - (2^k - 1)] = [0 - (2^{19} - 1)]$	01
4) La taille de cette mémoire = 2^{20} mots de 8 bits = 2^{17} mots de 64 bits	01
5) L'adresse, en hexadécimal du 5ème élément d'un tableau dont l'adresse du premier élément est 69_{16} et dont tous les éléments sont composés de 64 bits = $69_{16} + 4 * 4_{10} = 69_{16} + 10_{16} = 79_{16}$	01

Bon courage.