

# ORDRE DES INGÉNIEURS DU QUÉBEC

SESSION DE MAI 2023

Toute documentation permise  
 Calculatrices : modèles autorisés seulement  
 Durée de l'examen : 3 heures

## 16-EL-A4 – Systèmes numériques et ordinateurs

**Question 1 (40 points)** Soit cette table de vérité pour un circuit logique ayant pour entrées A, B, C, D et pour sortie S :

A	B	C	D	S
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

- Donnez l'équation canonique de ce circuit (10 points)
- Donnez la table de Karnaugh représentant ce circuit (10 points)
- Tracez le diagramme comportant **le plus petit nombre de portes logiques** pour réaliser ce circuit en incluant l'équation booléenne correspondante. Les portes ET et OU peuvent avoir plus de deux entrées. (20 points)

**Question 2 (30 points)** Vous devez concevoir une machine à états finis de Moore capable de compter par multiples de 3 le nombre de bits « 1 » dans une séquence de longueur indéfinie. La machine devra avoir un seul bit de sortie qui sera « 1 » au cycle suivant celui où un multiple de trois a été atteint et « 0 » dans tous les autres cas. Voici un exemple du résultat attendu sur une séquence de 20 cycles :

T	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Entrée	0	1	1	0	1	0	0	1	1	1	1	0	1	0	1	0	1	0	1	0
Sortie	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0

- a) Tracez le diagramme de votre machine à états finis. **Attention** : la machine doit être conçue pour recevoir un seul bit à la fois (et non un groupement de bits) (10 points)
- b) Donnez la table d'états-transitions de la machine. La table devra respecter le format suivant dans votre cahier réponse, et vous pouvez utiliser autant d'états que vous voulez. (5 points)

État	Entrée 0	Entrée 1	Sortie
0	(état suivant si 0)	(état suivant si 1)	(sortie de l'état)
1	...	...	...
2	...	...	...
...	...	...	...

- c) Donnez la séquence de sortie du circuit pour l'entrée suivante (5 points) :
- « 100011100101 »

- d) Traduisez le circuit en un programme en C où l'entrée peut être obtenue par la fonction `int E(void)` et la sortie changée par la fonction `void S(int)`. (10 points).

Par exemple, si la séquence d'entrée est « 100 » et que la sortie devrait être « 000 », la fonction `E()` devrait agir ainsi :

```
int v = 0;
v = E();      // v = 1
v = E();      // v = 0
v = E();      // v = 0
```

Le programme doit fonctionner indéfiniment. Il pourrait donc avoir une structure de la sorte :

```
while (true) {
    // *** À faire : traitement de l'entrée avec E() ***
    int out = 0;    // À changer selon l'état en cours
    S(out);
}
```

(Dans l'exemple précédent, sans modifications, la sortie serait toujours à 0).

**Question 3 (20 points) :** Soit le programme en C suivant :

```
int t0  = 0;
int t1  = 5;
int *t2 = 0x2FC0;

do {
    t0 += 2;
while (t1 >= t0);

*t2 = a;
```

- a) Donnez le contenu de l'adresse 0x2FC0 à la fin de l'exécution (5 points)
  
- b) Traduisez ce programme en assembleur MIPS (15 points)

**Question 4 (10 points)** Comment est-ce que deux processus ayant chacun un espace de mémoire virtuel privé peuvent partager un espace commun ? Expliquez comment un gestionnaire de mémoire (*Memory Management Unit*) peut-être configuré pour réaliser ceci.