

Répondez dans le cahier de réponses.

Question #1 – 15%

Construisez un circuit qui prend en entrée un entier non signé sur 8 bits et qui le transforme en entier signé sur 16 bits.

Question #2 – 10%

L'instruction `beq` s'exécute plus rapidement sur un processeur multi-cycles que sur le processeur à un cycle contrairement à l'instruction `lw` qui s'exécute plus rapidement sur un processeur à un cycle que sur le processeur multi-cycles. Expliquez cette différence.

Question #3 – 10%

L'unité arithmétique et logique calcule le complément à 2 d'un nombre lorsqu'il doit effectuer une soustraction. Expliquez comment l'UAL effectue le complément à 2 d'un nombre.

Question #4 – 10%

La valeur stockée dans le champ Address de l'encodage de type J correspond à une adresse mémoire encodée sur 26 bits. Si les adresses mémoires avec MIPS font 32 bits, où va-t-on chercher les 6 bits manquants de l'adresse?

Question #5 – 15%

Construisez un circuit qui prend en entrée un nombre signé sur 8 bits et qui retourne vrai si le nombre donné est pair et faux lorsque le nombre donné est impair. Expliquez votre démarche.

Question #6 – 10%

L'instruction `beq` possède un encodage de type I. Expliquez clairement ce qui est placé dans chaque champ de l'encodage de type I lorsqu'on encode cette instruction.

Type I : 6 bits pour opcode, 5 bits pour rs, 5 bits pour rt, 16 bits pour la valeur immédiate.

Question #7 – 10%

En respectant la norme IEEE 754 sur l'encodage des points flottants, encodez le nombre réel -44.5. Expliquez clairement votre démarche. (Note : 1 bit de signe, 8 bits d'exposant, 23 bits de mantisse)

Question #8 – 20%

Construisez le circuit simplifié de la fonction suivante. Décrivez votre démarche.

Petit rappel sur les règles de simplification :

$$ab+ac=a(b+c)$$

$$\bar{a}\bar{b}=\overline{a+b}$$

$$\bar{a}+b=\overline{a\bar{b}}$$

$$\bar{a}b+a\bar{b}=a\oplus b$$

<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>S</u>
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
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	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1