

Corrigé de l'examen Architecture des ordinateursExercice 1 (6 points)

Soit **K bits** la taille du registre d'adresses (**RA**) et **N bits** la taille d'un mot mémoire.

- Exprimer la capacité de la mémoire centrale en **octets** et en **mots**.
Capa = $N \cdot 2^K$ [bits] = $N \cdot 2^K / 8$ [octets] = $N \cdot 2^K / 32$ [mots]
- Supposons que **K = 32** et **N = 32**. Déduire la capacité mémoire en octets et en mots.
Capa = $32 \cdot 2^{32} / 8 = 2^{34}$ [octets]
Capa = $32 \cdot 2^{32} / 32 = 2^{32} = 4\text{Gmots}$.
- On veut réaliser cette mémoire en utilisant des blocs mémoires de taille **4M mots (1M = 2^{20} mots)**. Combien de blocs mémoires faut-il pour réaliser cette mémoire.

$$\text{Nombre de blocs} = \text{Capa} / 4\text{Mmots} = 2^{32} / 4 \cdot 2^{20} = 2^{10} \text{ blocs}$$

- Supposons que la lecture d'un mot mémoire se fait en **4 cycles** et qu'un seul cycle est égal à **0.5ns** et que son traitement par le processeur nécessite **5 cycles**. Combien de temps faut-il pour traiter toutes les données contenues dans la mémoire?

$$\text{Nombre de Cycles} = 4 + 5 = 9 \text{ cycles}$$

$$T_{\text{traitement}} = 9 \cdot 0.5 \cdot 2^{32} = 19,32 \text{ nano seconde}$$

Exercice 2 (6 points)

- Une donnée est stockée en mémoire (voir ci-contre). Donner cette donnée après lecture dans les deux cas suivants :

- Les données sont rangées en Little-endian.

$$\text{Little endian} = 0x554E4958$$

- Les données sont rangées en Big-endian.
Big Endian = 0x58494E55

0xbfffa7c	58
0xbfffa7d	49
0xbfffa7e	4E
0xbfffa7f	55

- Lors de l'exécution d'une instruction avec le simulateur MARS4.5, une erreur s'affiche. Par cette instruction, on voulait lire un mot mémoire. Sachant que l'adresse mémoire donnée dans cette instruction est la suivante **0x abcd755e**. Donner la cause de cette erreur.

L'adresse n'est pas alignée (n'est pas divisible par 32)

Exercice 3 (8 points)

- On considère l'instruction suivante codées en hexadécimale

Instructions	opcode
0x1310FFFD	beq : 000100

Sachant que le PC contient l'adresse suivante : 0x004004BC.

Bon courage

- Donner la forme d'assemblage des cette instruction en précisant la valeur de l'adresse cible.

$$(0x1310FFFD)_{16} = (\underbrace{000100}_{\text{Format I}} \underbrace{11000}_{\text{Opcode}} \underbrace{10000}_{\text{rs}} \underbrace{1111111111111101}_{\text{rt}} \underbrace{\hspace{1cm}}_{\text{immédiat}})_{2}$$

beq \$24,\$16, @cible

$$\text{@cible} = \text{PC} + 4 * (\text{offset})$$

Offset = -3 l'inverse du complement à deux de (immédiat = 1111111111111101)

$$\text{Donc : @cible} = 0x004004BC + 4 * (-3)$$

$$\text{@cible} = 0x004004B0$$

2. On considère les instructions d'assemblage suivantes

Adresses	Instructions	opcode
0x0040002C	lw \$18, 144(\$6)	lw : 100011 ;
0x00400030	addi \$29,\$29,-4	addi : 001000 ;
0x00400034	sw \$8, 100(\$5)	sw : 101011 ;

- Écrire le code hexadécimal de ces trois instructions.

Instruction	Form at	Code en binaire	Code Hexa
lw \$18, 144(\$6)	I	<u>100011</u> <u>00110</u> <u>10010</u> <u>0000</u> <u>0000</u> <u>1001</u> <u>0000</u> opcode Rs Rt Imm	0x8CD20090
addi \$29,\$29,-4	I	<u>001000</u> <u>11101</u> <u>11101</u> <u>1111</u> <u>1111</u> <u>1111</u> <u>1100</u> opcode Rs Rt Imm	0x23BDFFFC
sw \$8, 100(\$5)	I	<u>101011</u> <u>00101</u> <u>01000</u> <u>0000</u> <u>0000</u> <u>01100100</u> opcode Rs Rt Imm	0xCA80064