

Fiche de TD N° 3 Architecture des Ordinateurs (AO)

Exercice 1

- I. Une mémoire possède 13 lignes d'adresses et 8 lignes de données :
1. Combien de mots binaires peut-on emmagasiner dans cette mémoire et combien de bits possède ce mot ?
 2. Quelle est la capacité totale de cette mémoire (en bits) ?
 3. Donner l'architecture réduite de cette mémoire.
- II. Lesquelles de ces organisations de mémoire sont plausibles ou envisageables ?
- a. Registre d'adresses de 10 bits, 1024 cellules, 08 bits par cellule
 - b. Registre d'adresses de 10 bits, 1024 cellules, 12 bits par cellule
 - c. Registre d'adresses de 9 bits, 1024 cellules, 10 bits par cellule
 - d. Registre d'adresses de 11 bits, 1024 cellules, 10 bits par cellule
 - e. Registre d'adresses de 10 bits, 10 cellules, 1024 bits par cellule

Exercice 2

Soit une machine avec un registre d'adresse mémoire (R@M) comporte 32 bits, calculer :

1. Le nombre de mots adressable si un mot= 1octet, et si 1 mot= 16 bits. (Dans les deux cas)
2. La plus haute adresse possible dans les deux cas.
3. La capacité de la mémoire centrale dans les deux cas.
4. Quelle est la taille des registre CO, RI et ACC dans les deux cas.

Exercice 3

Soit une machine dotée d'une mémoire centrale de 256 Méga bits et un registre de donnée 16 bits.

1. Combien de valeur différente peut prendre un mot de cette mémoire ?
2. Donner la taille du bus d'adresse qui permet d'accéder à cette mémoire.
3. Est-ce que cette mémoire peut avoir un mot mémoire de 8 bits ? justifier votre réponse
4. Supposant que processeur travaillant à une fréquence de 2000 MHz, donner le taux de transfert (théorique) d'un bus de donnée d'un processeur en MOcts/s.
5. Donner l'architecture réduite de cette mémoire.

Exercice 4

Le chargement des données depuis le disque dur vers la mémoire centrale s'effectue via un bus de 64 bits et la capacité de cette mémoire est de 32 Giga-octets.

1. Donner le nombre de lignes de donnée.
2. Donner le nombre de ligne d'adresse.
3. Donner la taille des registres : RI (Registre d'instruction) et CO (Compteur Ordinal).
4. Donner l'adresse du dernier mot en Binaire et en Hexadécimal.
5. On veut stocker sur cette mémoire des nombres réels. Calculer en Hexadécimale l'adresse du dixième nombre sachant que le premier nombre est stocké à partir de l'adresse $1F_{(16)}$ et que chaque nombre est représenté sur 8 Octets.

Exercice 5

1. Nous disposons d'une mémoire principale de 64 M mots de 32 bits.
 - a. Calculer sa capacité en Méga-octets.
 - b. Déterminer la largeur du bus de données.
 - c. Déterminer le nombre de bits du bus d'adresse et du registre d'adresse du microprocesseur associé à cette mémoire.
 - d. Nous voulons étendre la mémoire principale à 1Go avec des puces de capacité 512Ko. Calculer le nombre de puces nécessaire à l'extension de cette mémoire.
2. Dans cette mémoire nous avons stocker un tableau de données de 30 éléments, où le premier élément occupe l'adresse $(6F)_{16}$ et le troisième élément se trouve à l'adresse $(73)_{16}$.
 - a. Déduire le nombre de mots mémoire occupé par un élément du tableau.
 - b. Calculer l'adresse du dernier élément du tableau.

Exercice 6

1. Si l'intervalle des adresses d'une mémoire va de 0000H à FFFE H. Combien cette mémoire a de cases ?
2. Si une mémoire possède 5120 emplacements en mémoire. Donner l'intervalle (début et fin) de ces adresses exprimées en hexadécimal.
3. Si l'intervalle des adresses d'une mémoire va de 0531H à F20DH. Combien cette mémoire a de cases ?

Fiche de TD N° 3 Architecture des Ordinateurs (AO) (Solution)

Exercice 1

I. Une mémoire possède 13 lignes d'adresses et 8 lignes de données :

- Combien de mots binaires peut-on emmagasiner dans cette mémoire et combien de bits possède ce mot ?

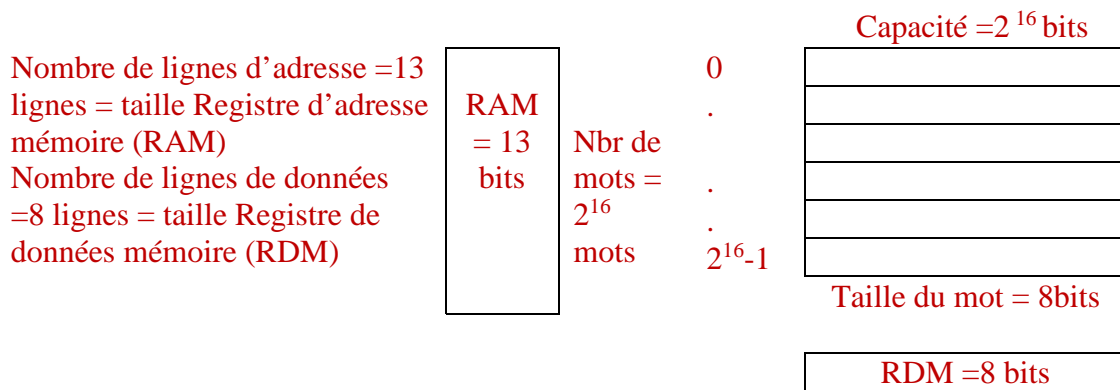
Nombre de mots = $2^{\text{nombre lignes d'@}}$ → donc Le nombre de mots = 2^{13} mots

Et taille du mot = nbre de lignes de données → donc taille du mot = 8 bits

- Quelle est la capacité totale de cette mémoire (en bits) ?

On a capacité = Taille du mot * nombre de mots = $2^{13} \times 8$ → donc Capacité = 2^{16} bits

- Donner l'architecture réduite de cette mémoire.



II. Lesquelles de ces organisations de mémoire sont plausibles ou envisageables ?

Organisation envisageable si $2^{\text{taille du registre d'@}} \geq \text{nbre de cellules}$ → car ce sont les seuls paramètres qu'ils ont une relation entre eux

- Registre d'adresses de 10 bits, 1024 cellules, 08 bits par cellule
 $2^{10} = 1024$ donc envisageable
- Registre d'adresses de 10 bits, 1024 cellules, 12 bits par cellule
 $2^{10} = 1024$ donc envisageable
- Registre d'adresses de 9 bits, 1024 cellules, 10 bits par cellule
 $2^9 < 1024$ donc impossible
- Registre d'adresses de 11 bits, 1024 cellules, 10 bits par cellule
 $2^{10} > 1024$ donc envisageable
- Registre d'adresses de 10 bits, 10 cellules, 1024 bits par cellule
 $2^{10} > 10$ donc envisageable

Exercice 2

Soit une machine avec un registre d'adresse mémoire (R@M) comporte 32 bits, calculer :

- Le nombre de mots adressable si un mot = 1 octet, et si 1 mot = 16 bits. (Dans 2 cas)

1 ^{er} cas : si 1 mot = 8 bits Le nbre de mots = $2^{\text{lignes d'@}}$ => Le nbre de mots = 2^{32} Donc le nombre de mots = 2^{32} mots de 8 bits	2 ^{ème} cas : si 1 mot = 2 octets Donc le nombre de mots = 2^{32} mots de 2 octets
---	--

- La plus haute adresse possible dans les deux cas.

1 ^{er} cas : si 1 mot= 8 bits	2 ^{ème} cas : si 1 mot= 2 octets
La plus haute adresse possible pour ces mots de 8 bits. rep= le nombre de mots -1 = $2^{32} - 1 = \text{FFFFFFFF (16)}$	La plus haute adresse possible pour ces mots de 2 octets. rep= $2^{32} - 1 = \text{FFFFFFFF (16)}$

3- La capacité de la mémoire centrale dans les deux cas.

1 ^{er} cas : si 1 mot= 8 bits	2 ^{ème} cas : si 1 mot= 2 octets
Capacité1= Nbre de mots x Taille du mot = $2^{32} \times 8 = 2^{35}$ bits	Capacité2= $2^{32} \times 2 = 2^{33}$ octets ou = $2^{32} \times 16 = 2^{36}$ bits

4- Quelle est la taille des registre CO, RI et ACC dans les deux cas.

1 ^{er} cas : si 1 mot= 8 bits	2 ^{ème} cas : si 1 mot= 2 octets
CO=Nbre de ligne d'adresse= RAM=32 bits RI= Nbre de ligne de données= RDM =taille mots= 8bits ACC= Nbre de ligne de données= 8bits	CO =32 bits RI= 2 octets = 16 bits ACC= 2 octets = 16 bits

Exercice 3

Soit une machine dotée d'une mémoire centrale de 256 Méga bits et un registre de donnée 16 bits.

1. Combien de valeur différente peut prendre un mot de cette mémoire ?

Un mot mémoire = taille du registre de donnée = 16 bits

Nombre de valeur différente d'un mot mémoire= $2^{\text{taille du mot}} = 2^{16}$ valeur différente

2. Donner la taille du bus d'adresse qui permet d'accéder à cette mémoire.

Nbre de mots = capacité/taille du mot → Nbre de mots = $256 \text{ Méga} / 16 = 2^8 * 2^{20} / 2^4 = 2^{24}$ mots

Nombre de mots = $2^{\text{nombre de lignes d'adresse}}$ sachant que nbre de lignes d'@ = taille du bus d'@

Donc taille du bus d'adresse = 24 bits

3. Est-ce que cette mémoire peut avoir un mot mémoire de 8 bits ? justifier votre réponse

Mot mémoire = 8 bits donc **oui** car **taille du mot mémoire** =< **taille registre de donnée**

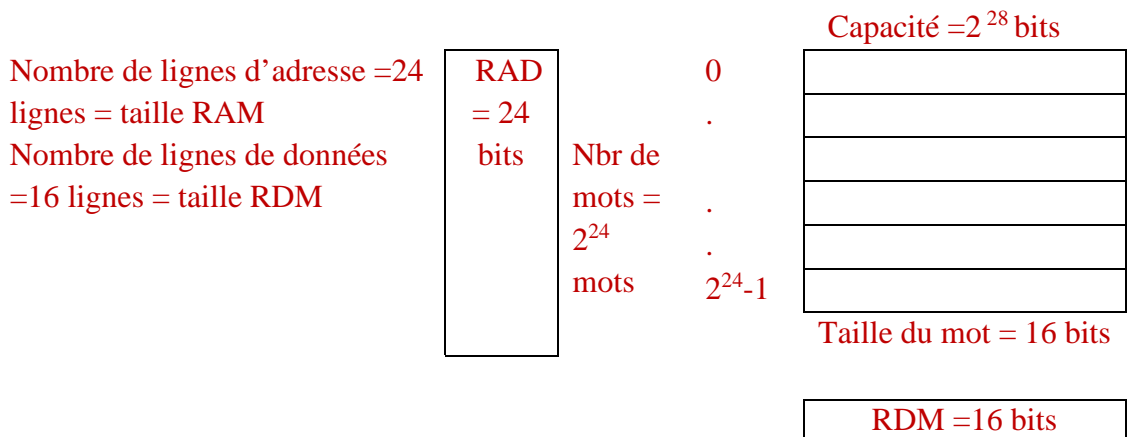
4. Supposant que processeur travaillant à une fréquence de 2000 MHz, donner le taux de transfert (théorique) d'un bus de donnée d'un processeur en MOctets/s.

Taux de transfert (Mo/s) = Fréquence (en MHz) × Largeur du bus (en octets)

Largeur du bus= Taille d'un bus de données = nombre de lignes de données=16 bits = **2 octets**

Taux de transfert (Mo/s) = $2000 * 2 = 4000 \text{ Mo/s}$

5. Donner l'architecture réduite de cette mémoire.



Exercice 4

Le chargement des données depuis le disque dur vers la mémoire centrale s'effectue via un bus de 64 bits et la capacité de cette mémoire est de 32 Giga-octets.

1. Donner le nombre de lignes de donnée.

Taille bus de donnée = Nbre de ligne de données = 64 lignes

2. Donner le nombre de ligne d'adresse.

Nbre de mots = Capacité / Nbre de ligne de données = $32 * 2^{30} * 2^3 / 2^6 = 2^{38} / 2^6 = 2^{32}$ mots

Nombre de mots = $2^{\text{nombre de lignes d'adresse}}$ → donc nbre de lignes d'adresse = 32 lignes

3. Donner la taille des registres : RI (Registre d'instruction) et CO (Compteur Ordinal).

RI = Nbre de ligne de données = 64 bits

CO = Nbre de ligne d'adresse = 32 bits

4. Donner l'adresse du dernier mot en Binaire et en Hexadécimal.

L'adresse du dernier mot = le nombre de mots - 1 = $2^{32} - 1$

1111111 ... 1111 (2) sur 32 bits = FFFFFFFF (16)

5. On veut stocker sur cette mémoire des nombres réels. Calculer en Hexadécimale l'adresse du dixième nombre sachant que le premier nombre est stocké à partir de l'adresse 1F₍₁₆₎ et que chaque nombre est représenté sur 8 Octets.

Nombre de case mémoire pour chaque nombre réel = taille du nombre réel / taille du mot

= 8 octets / 64 bits = 64 / 64 = 1 mot

@10^{ème} nbre réel = @1^{er} nbre réel + (nombre de mots pour chaque nbre réel * (n-1))

Donc @10^{ème} nbre réel = @1^{er} nbre réel + (10-1)

@10^{ème} nbre réel = (1F)₍₁₆₎ = 00011111 = 31₍₁₀₎

@10^{ème} nbre réel = 31 + 9 = 40₍₁₀₎ = 28₍₁₆₎

Exercice 5

1. Nous disposons d'une mémoire principale de 64 M mots de 32 bits.

- a. Calculer sa capacité en Méga-octets.

C = 64 M * 32 bits = 256 MO

- b. Déterminer la largeur du bus de données.

Largeur Bus de données = taille du mot = 32 bits

- c. Déterminer le nombre de bits du bus d'adresse et du registre d'adresse du microprocesseur associé à cette mémoire.

Le bus d'adresse et Le registre d'adresse ont tous les deux la même taille.

Le nbre de mots = 64 MO = $2^6 * 2^{20} = 2^{26}$ → bus d'adresse et registre d'adresse = 26 bits

- d. Nous voulons étendre la mémoire principale à 1Go avec des puces de capacité 512Ko.

Calculer le nombre de puces nécessaire à l'extension de cette mémoire.

L'extension est : 1GO - 256 MO = 2^{30} octets - $256 * 2^{20}$ octets

= 2^{20} (1024 - 256) octets = 2^{20} (768) octets

Puce disponible : 512 K octet

Nombre de puce nécessaire : 2^{20} (768) octets / $512 * 2^{10}$ octets = 1536 puces

2. Dans cette mémoire nous avons stocker un tableau de données de 30 éléments, où le premier élément occupe l'adresse (6F)₁₆ et le troisième élément se trouve à l'adresse (73)₁₆.

a. Déduire le nombre de mots mémoire occupé par un élément du tableau.

@ 1^{er} élément + (nb mot par élément*(3-1)) = @ 3^{ème} élément

⇒ nb mot par élément = (@ 3^{ème} élément - @ 1^{er} élément) / 2

nb mot par élément = ((73)₁₆ - (6F)₁₆) / 2 = (115-111)/2=2

→ Un élément du tableau occupe 2 Mots mémoires

b. Calculer l'adresse du dernier élément du tableau.

Ecart : 30 - 1 = 29 * 2 = (58)₁₀ = (3A)₁₆

Adresse du 30^{ème} élément = adresse du 1^{er} élément + Ecart = (6F)₁₆ + (3A)₁₆ = (A9)₁₆

Exercice 6

1. Si l'intervalle des adresses d'une mémoire va de 0000H à FFFEh. Combien cette mémoire a de cases ?

Le nombre des cases mémoires est : (FFFEh - 0000h) + 1 = FFFFh = (65535)₁₀

2. Si une mémoire possède 5120 emplacements en mémoire. Donner l'intervalle (début et fin) de ces adresses exprimées en hexadécimal.

@ de début est 0000h. @ de fin est : 5120 - 1 = (5119)₁₀ = 13FFh

3. Si l'intervalle des adresses d'une mémoire va de 0531H à F20DH. Combien cette mémoire a de cases ?

Le nombre de cases est : (F20Dh - 0531h) + 1h = ECDDh = (60637)₁₀