

## Chapitre 2 : Principaux composants d'un ordinateur

### Exercice 1 :

Associer chaque composant de la liste suivante avec une des définitions proposées :

| Composant                            | Définition   |
|--------------------------------------|--|
| a. Carte mère                        | 1. Ensemble de fils permettant de lier et faire communiquer les composants d'un ordinateur afin d'assurer la transmission du même type d'information (données, adresses ou commandes). |
| b. Chipset                           | 2. Permettant l'exécution des instructions du programme et des calculs sur les données.  |
| c. Bus                               | 3. Circuit électronique chargé de coordonner les échanges de données entre les divers composants de l'ordinateur (processeur, mémoire...).   |
| d. Séquenceur                        | 4. Identifier l'instruction à exécuter qui se trouve dans le Registre d'Instruction (RI).  |
| e. Software                          | 5. Permettent l'échange d'informations avec les dispositifs extérieurs.  |
| f. Processeur                        | 6. Dispositif de stockage de données et de programme.  |
| g. Hardware                          | 7. Constituée de l'ensemble des programmes pouvant être un programme d'application ou un programme de pilotage ou de base d'une machine  |
| h. Mémoire                           | 8. Constituée de l'ensemble des composants physique d'une machine.   |
| i. Périphérique d'entrée / de sortie | 9. Socle permettant la connexion de l'ensemble des éléments essentiels de l'ordinateur.  |
| g. Décodeur                          | 10. Chargé de synchroniser l'exécution des instructions au rythme d'une horloge.   |

### Solution :

| Composant                         | Définition |
|-----------------------------------|------------|
| a. Carte mère                     | 9          |
| b. Chipset                        | 3          |
| c. Bus                            | 1          |
| d. Séquenceur                     | 10         |
| e. Software                       | 7          |
| f. Processeur                     | 2          |
| g. Hardware                       | 8          |
| h. Mémoire                        | 6          |
| i. Périphérique d'entrée / sortie | 5          |
| g. Décodeur                       | 4          |

**Exercice 2**

Quelle est la signification de chacun des acronymes suivants :

- 1) UAL : .....
- 2) CPU : .....
- 3) RAM : .....
- 4) ROM : .....
- 5) USB : .....
- 6) VGA : .....
- 7) HDMI : .....

**Solution :**

- 1) UAL : **Unité Arithmétique et Logique**
- 2) CPU : **Central Processing Unit**
- 3) RAM : **Random Access Memory**
- 4) ROM : **Read Only Memory**
- 5) USB : **Universal Serial BUS**
- 6) VGA : **Video Graphics Array**
- 7) HDMI : **High Definition Multimedia Interface**

**Exercice 3**

Un ordinateur est équipé d'un processeur Pentium 4 à 3,6 GHz fonctionnant à une fréquence de carte mère de 800 MHz.

- Calculer le taux de transfert maximal du bus processeur sachant que la quantité de données pouvant être transférées simultanément est de 64 bits.

**Solution :**

Pour calculer le taux de transfert du bus processeur, il faut multiplier la quantité de données pouvant être transférées simultanément (64 bits) par la fréquence d'horloge du bus (identique à la fréquence du processeur avant multiplication).

Pour obtenir le taux de transfert, on doit utiliser la formule suivante :

$$\text{Taux de transfert (Mo/s)} = \text{Fréquence (en MHz)} \times \text{Largeur du bus (en octet)}$$

D'où **Taux de transfert (Mo/s) = 800 MHz × 8 octets (64 bits) = 6400 Mo/s.**

D'où le taux de transfert instantané maximal est de **6400 Mo/s.**

**Exercice 4**

Soit un bus PCI 64 bits tournant à 64 Mhz

1. Calculer le taux maximum (théorique) de transfert.
2. Exprimer le résultat obtenu en Méga Byte/s et Méga Octets/s.

**Solution :****1. Le taux maximum (théorique) de transfert**

$$\text{Taux de transmission ou débit (en bits/s)} = \text{largeur bus (en bits)} \times \text{Fréquence (MHz)}$$

Taux de transmission = 64 X 64 = **4096 M bits/s**

**2. Le résultat obtenu en Méga Byte/s et Méga Octets/s**

Taux de transmission = 4096 M bits/s = 4096/8 = **512 M byte/s** (car 1 byte = 8bit)

Taux de transmission = 4096 M bits/s = 4096/8 = **512 M octets/s** (car 1 octet = 8 bits)

### Exercice 5

Calculer les taux de transferts pour les bus mémoires suivants :

|                                 | DRAM | SDRAM | SDRAM<br>PC100 | DDR SDRAM<br>PC2100 |
|---------------------------------|------|-------|----------------|---------------------|
| <b>Largeur du bus (bits)</b>    | 32   | 64    | 64             | 64                  |
| <b>Fréquence du bus (MHz)</b>   | 66   | 66    | 100            | 133                 |
| <b>Taux de transfert (Mo/s)</b> |      |       |                |                     |

### Solution :

Les taux de transferts pour les bus mémoires donnés sont calculés à l'aide de :

$$\text{Taux de transmission (en Mo/s)} = (\text{largeur bus (en bits)} \times \text{Fréquence (en MHz)}) / 8$$

|                                 | DRAM       | SDRAM      | SDRAM<br>PC100 | DDR SDRAM<br>PC2100                |
|---------------------------------|------------|------------|----------------|------------------------------------|
| <b>Largeur du bus (bits)</b>    | 32         | 64         | 64             | 64                                 |
| <b>Fréquence du bus (MHz)</b>   | 66         | 66         | 100            | 133                                |
| <b>Taux de transfert (Mo/s)</b> | <b>264</b> | <b>528</b> | <b>800</b>     | <b>2128 (1064x2 pour PC 2x100)</b> |

### Exercice 6

Calculer les taux de transfert pour les bus périphériques suivants :

|                                 | ISA  | EISA | PCI   | AGP   | AGP 4x |
|---------------------------------|------|------|-------|-------|--------|
| <b>Largeur du bus (bits)</b>    | 16   | 32   | 32    | 32    | 32     |
| <b>Fréquence du bus (MHz)</b>   | 8.33 | 8.33 | 33.33 | 66.66 | 66.66  |
| <b>Taux de transfert (Mo/s)</b> |      |      |       |       |        |

### Solution :

Les taux de transfert pour les bus périphériques donnés sont calculés à l'aide de :

$$\text{Taux de transmission (en Mo/s)} = (\text{largeur bus (en bits)} \times \text{Fréquence (en MHz)}) / 8$$

|                                 | ISA          | EISA         | PCI           | AGP           | AGP 4x                        |
|---------------------------------|--------------|--------------|---------------|---------------|-------------------------------|
| <b>Largeur du bus (bits)</b>    | 16           | 32           | 32            | 32            | 32                            |
| <b>Fréquence du bus (MHz)</b>   | 8.33         | 8.33         | 33.33         | 66.66         | 66.66                         |
| <b>Taux de transfert (Mo/s)</b> | <b>16.66</b> | <b>33.32</b> | <b>133.32</b> | <b>266.64</b> | <b>1066.56</b><br>= 266.64 x4 |

**Exercice 7**

A quoi servent les registres suivants du processeur :

1. PC/IP (ou CO/PI)
2. IR (ou RI)
3. SP (ou PP)
4. Accumulateur (Acc)
5. PSW (ou RE)

**Solution :**

1. PC/IP (ou CO/PI) : Program Counter /Instruction Pointer (Compteur Ordinal/Pointeur d'instruction) pointe vers l'instruction à exécuter (suivante)
2. IR (ou RI) : Le registre d'instruction (Instruction Register) contient l'instruction en cours d'exécution
3. SP (ou PP) : Le pointeur de pile (Stack Pointer) pointe le sommet de la pile.
4. Accumulateur (Acc) : L'accumulateur stocke le résultat de l'UAL.
5. PSW (ou RE) : le Processor Status Word (Registre d'état) permettant de stocker des indicateurs sur l'état du système (retenue, dépassement, etc.).

**Exercice 8**

Préciser la fonction de la RAM dans un ordinateur. Quelles sont ses deux caractéristiques principales ?

**Solution :**

La mémoire vive, mémoire système ou mémoire volatile aussi appelée **RAM** de l'anglais Random Access Memory (mémoire à accès aléatoire) est la mémoire informatique dans laquelle un ordinateur place les données lors de leur traitement. Ses principales caractéristiques sont :

- 1) Sa rapidité d'accès (essentielle pour fournir rapidement les données au processeur)
- 2) Sa volatilité (cette volatilité implique que les données sont perdues dès que l'ordinateur cesse d'être alimenté en électricité).

**Exercice 9**

Quelles sont les principales différences entre la RAM et la ROM ? Où utilise-t-on la ROM ?

**Solution :**

La RAM est en lecture/écriture mais volatile alors la ROM est en lecture seule et non volatile.

On utilise en général la ROM pour les données du BIOS pour le démarrage de la machine.

**Exercice 10**

Pourquoi utilise-t-on des mémoires caches ? Pour qu'elle raison la mémoire cache est très couteuse ?

**Solution :**

Une mémoire cache en informatique est une mémoire qui enregistre temporairement des copies de données provenant de la RAM, afin de diminuer le temps d'accès (en lecture ou en écriture) d'un matériel informatique (en général, un processeur) à ces données. La

mémoire cache est plus rapide et plus proche du matériel informatique qui demande la donnée, mais plus petite que la mémoire pour laquelle elle sert d'intermédiaire.  
La mémoire cache est très couteuse car elle est plus rapide et elle est miniaturisée)

**Exercice 11**

Classer les mémoires suivantes par taille et par rapidité : RAM, registres, disques durs, cache L1, cache L2, CD-ROM. Que constatez-vous ?

**Solution :**

**Classement par taille :** Registres < L1 < L2 < RAM < CD < DD.

**Classement par vitesse :** Registres > L1 > L2 > RAM > DD > CD.

**Remarque :** Plus la taille de la mémoire est petite plus elle est rapide.

**Exercice 12**

Soit une machine où tous les registres sont sur 32 bits

- 1- Calculer la capacité maximale de la mémoire
- 2- Donner le nombre de mot réel de cette mémoire si on suppose que sa capacité =  $2^{32}$  bits
- 3- Proposer une architecture réduite pour cette mémoire (nombre de lignes d'adresses ; nombre de lignes de données ; taille du registre de mémoire de données ; taille du registre de mémoire d'adresse)

**Solution :**

- 1- La capacité maximale de la mémoire

$$\text{Capacité} = 2^{\text{lignes d'@}} \times \text{nbre de lignes de données} = 2^{32} \times 32 = \mathbf{2^{37} \text{ bits}}$$

- 2- Le nombre de mot réel de cette mémoire si on suppose que sa capacité =  $2^{32}$  bits

$$\mathbf{\text{Nbre de mots} = \text{capacité} / \text{taille du mot}}$$

$$\text{La taille du mot (en bits)} = \text{nbre de lignes de données} \Rightarrow 2^{32} / 32 = \mathbf{2^{27} \text{ mots}}$$

- 3- Une architecture réduite pour cette mémoire (nombre de lignes d'adresses ; nombre de lignes de données ; taille du registre de mémoire de données ; taille du registre de mémoire d'adresse)

$$\text{Le nbre de mots} = 2^{\text{lignes d'@}} \text{ et la taille du mot (en bits)} = \text{nbre de lignes de données.}$$

$$\text{Nombre de lignes d'adresses} = 27 = \text{taille RAM}$$

$$\text{Nombre de lignes de données} = 32 = \text{taille RDM}$$

$$\text{Capacité} = 2^{32} \text{ bits}$$

$$\begin{array}{ll} \text{RAD} & 0 \\ = 27 & . \\ \text{bits} & . \end{array}$$

$$\begin{array}{c} . \\ . \\ 2^{27}-1 \end{array}$$

---


$$32 \text{ bits}$$

$$\text{RDM} = 32 \text{ bits}$$

**Exercice 13**

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) ?

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

**Solution :**

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 ?

$$\text{Nbre mots} = 2^{13}, \text{ taille du mot} = 8 \text{ bits}$$

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

$$\text{Capacité} = 2^{13} \times 8 = 2^{16} \text{ bits}$$

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

Organisation envisageable si  $2^{\text{taille du registre d'@}} \geq \text{nbre de cellules}$

- a) Registre d'adresses de 10 bits, 1024 cellules, 08 bits par cellule  $\Rightarrow$  envisageable
- b) Registre d'adresses de 10 bits, 1024 cellules, 12 bits par cellule  $\Rightarrow$  envisageable
- c) Registre d'adresses de 9 bits, 1024 cellules, 10 bits par cellule  $\Rightarrow$  impossible
- d) Registre d'adresses de 11 bits, 1024 cellules, 10 bits par cellule  $\Rightarrow$  envisageable
- e) Registre d'adresses de 10 bits, 10 cellules, 1024 bits par cellule  $\Rightarrow$  envisageable

**Exercice 14**

Soit une machine dotée d'une mémoire centrale de 1024 K mot de 32 bits.

- 1- Combien de bits, d'octets, de Kilo octets et de Méga octets contient cette mémoire
- 2- Combien de valeur différente peut prendre un mot de cette mémoire ?
- 3- Déterminer la plage d'adressage de cette mémoire (en hexadécimal/ base 16)
- 4- On veut stocker sur cette mémoire des nombres réels et chaque nombre est représenté sur 64 bits. Calculer l'adresse du 9<sup>ème</sup> nombre sachant que le premier est stocké à l'adresse FF<sub>(16)</sub>

**Solution :**

1- Combien de bits, d'octets, de Kilo octets et de Méga octets contient cette mémoire

$$\text{Capacité (bits)} = \text{Nombre de mots} \times \text{taille du mot}$$

$$\text{Capacité (bits)} = 1024 \text{ K} \times 32 = 2^{10} \times 2^{10} \times 2^5 = 2^{25} \text{ bits}$$

$$\text{Capacité (octets)} = 2^{25} / 2^3 = 2^{22} \text{ octets}$$

$$\text{Capacité (Koctets)} = 2^{22} / 2^{10} = 2^{12} \text{ Koctets}$$

$$\text{Capacité (Moctets)} = 2^{12} / 2^{10} = 2^2 \text{ Moctets}$$

2- Combien de valeur différente peut prendre un mot de cette mémoire ?

$$2^{32} \text{ valeurs}$$

3- La plage d'adressage de cette mémoire (en hexadécimal/ base 16)

**Nombre de mots** =  $2^{\text{nombre de lignes d'adresses}}$

Nombre de mots =  $1024 \text{ K} = 2^{10} * 2^1 = 2^{20}$

⇒ **Nombre de lignes d'adresse** = 20

**Adresse minimale** ( 0000000000.. ) sur 20 bits = **00000**<sub>(16)</sub>

et **adresse maximale** ( $2^{20} - 1$ ) = 1111111 ....1111 sur 20 bits = **FFFFFF**<sub>(16)</sub>

- 4- On veut stocker sur cette mémoire des nombres réels et chaque nombre est représenté sur 64 bits. Calculer l'adresse du 9<sup>ème</sup> nombre sachant que le premier est stocké à l'adresse FF<sub>(16)</sub>

Chaque nombre dans la mémoire prend 2 mots mémoire (taille du nombre réel / taille du mot =  $64/32 = 2$  mots)

**@n<sup>ème</sup> nombre** = **@1<sup>er</sup> nombre** + (nombre de mots de chaque nombre) \* (n-1)

**@1<sup>er</sup> nombre** = FF<sub>(16)</sub> =  $(15 * 16 + 15)_{(10)} = 255_{(10)}$

**@9<sup>ème</sup> nombre** = **@1<sup>er</sup> nombre** + (8 \* 2) =  $255_{(10)} + 16_{(10)} = 271_{(10)} = 10F_{(16)}$

### Exercice 15

Soit une mémoire cache de niveau L1 ayant les caractéristiques suivantes :

- 32 mots par lignes (mots de 2 octets) - Taille de 32ko - L1 et L2 sont inclusifs - 4-associatifs - Remplacement LRU - Association par poids faible - Taille de bus d'adresse : 32bits

- 1- Combien y-a-t-il de lignes dans cette mémoire cache ?
- 2- Combien y-a-t-il de blocs associatifs dans cette mémoire cache ?
- 3- Si la mémoire cache de niveau L2 a une taille de 2 Mo, combien y-a-t-il de blocs de la mémoire cache L2 par bloc de la mémoire cache L1 ?
- 4- Si la mémoire fait 1Go, combien d'adresses correspondront à un bloc du cache L1 ?
- 5- Si un bloc n'est pas présent en cache L1, combien de lignes de L1 aura-t-on parcouru ?

### Solution :

Soit une mémoire cache de niveau L1 ayant les caractéristiques suivantes :

- 32 mots par lignes (mots de 2 octets) - Taille de 32ko - L1 et L2 sont inclusifs - 4-associatifs - Remplacement LRU - Association par poids faible - Taille de bus d'adresse : 32bits

- 2- Combien y-a-t-il de lignes dans cette mémoire cache ?

**Nombre de lignes** = **Taille cache / (Taille mot \* nombre de mots par ligne)**

$$= 32 \text{ KOct} / (2 \text{ Oct} * 32) = 2^{10}/2 = 2^9 = 512 \text{ lignes}$$

- 6- Combien y-a-t-il de blocs associatifs dans cette mémoire cache ?

**Nombre de blocs** = **Nombre de lignes / Nombre de lignes par bloc**

$$= 512/4 = 2^9/2^2 = 2^7 = 128 \text{ blocs}$$

- 7- Si la mémoire cache de niveau L2 a une taille de 2 Mo, combien y-a-t-il de blocs de la mémoire cache L2 par bloc de la mémoire cache L1 ?

**Taille cache L2 / Taille cache L1** =  $2 \text{ MOct} / 32 \text{ KOct} = 2^{21} / 2^{15} = 2^6 = 64$

- 8- Si la mémoire fait 1Go, combien d'adresses correspondront à un bloc du cache L1 ?

**Nbre d'adresse à un bloc** =

**Taille mémoire / (Nbre blocs \* Taille mot \* Nbre de lignes d'adresse)**

$$= 1 \text{ GOct} / (128 * 2 \text{ Oct} * 32) = 2^{30} / 2^{13} = 2^{17} = 131072$$

Sachant que le nombre de lignes d'adresse = taille du bus d'adresse = 32

- 9- Si un bloc n'est pas présent en cache L1, combien de lignes de L1 aura-t-on parcouru ?

**4 lignes**

**Exercice 16**

**A)** Supposons un programme composé d'une boucle de 10 instructions et que la moitié des instructions se trouvent en mémoire cache et l'autre moitié en mémoire centrale. Si le temps d'accès au cache est de 5 ns et celui de la mémoire centrale est de 20 ns, calculer le temps global d'exécution du programme (sans prise en compte du temps d'exécution des instructions par le processeur).

**B)** Si le temps d'accès au cache est de 5 ns, le temps de pénalité au cache est de 10 ns, et que le temps d'échec du cache est de 20 ns. Calculez le temps de récupération d'une instruction

**Solution :**

**A)** Puisque la mémoire cache contient la moitié des instructions du programme, donc nous avons 5 instructions qui se trouvent en cache.

Chaque instruction dans le cache prend 5 ns pour la récupérer.

Au total pour le temps d'accès aux instructions qui se trouvent en mémoire cache, nous avons  $5 \times 5 = 25$  ns

Nous ajoutons à ce temps, le **temps d'accès aux instructions qui se trouvent en mémoire centrale** et pour lesquelles le **temps d'accès** est de **20 ns**.

Donc, nous avons  $5 \times 20 = 100$  ns

Le **temps global d'exécution** est de  $25 + 100 = 125$  ns

**B)** Le temps de récupération d'une instruction à partir du cache peut être calculé selon la formule de récupération d'une instruction de la façon suivante :

$$\text{Temps de récupération} = T_{\text{échec}} \times T_{\text{pénalité}} + (1 - T_{\text{échec}}) \times T_{\text{accès}}$$

$$\text{Temps de récupération} = 20 \times 10 + (1 - 20) \times 5 = 200 + (-95) = 105 \text{ ns}$$