

Gestion des mémoires (RAM, ROM, centrale) et systèmes d'exploitation



INF34207 – Séance du
31 janvier 2024

Lise Boudreault, chargée de cours

Plan du cours:

1. Cours précédent: processus et algorithmes d'ordonnancement¹

1.1. Shortest Job First (SJF)

Mémoires:

2. caractéristiques

3. classification

4. vives

5. mortes

6. centrale

1. Gestion des processus: Shortest Job First (SJF)

- Dans cet algorithme les différents processus sont rangés dans la file d'attente des processus prêts selon **un ordre croissant** de leurs temps d'exécution. Le processus se trouvant en tête de la file d'attente a **le plus petit cycle processeur comparativement aux autres**
- Quand **le processeur devient libre**, le processus se trouvant en tête de la file d'attente est exécuté. Si deux processus ont la **même longueur de cycle**, l'algorithme First Come First Served **s'applique soit celui placé en tête d'exécution** 😊!

1. Gestion des processus : Shortest Job First (SJF)

- Cet algorithme peut être **avec ou sans réquisition**:
 - **Sans réquisition**: Le processus élu ne quitte qu'à la fin de la rafale;
 - **Avec réquisition**: Si un nouveau processus arrive dont la durée prévue est inférieure au temps restant d'exécution du processus en cours, ce nouveau processus obtient le contrôle du CPU (on parle alors de SRTF (**Shortest Remaining Time First**)).

1. Gestion des processus: Shortest Job First (SJF)

- **L'algorithme SJF** choisit de façon prioritaire les processus ayant **les plus courts temps d'exécution** sans réellement tenir compte de leurs dates d'arrivée. Toutefois, la condition initiale suppose que l'on compare les TR_i entre eux pour ensuite les ordonner de manière croissante.

QUESTION:

- Comment peut-on connaître le temps d'exécution d'un processus à l'avance TR_i ?

1. Gestion des processus: Shortest Job First (SJF)

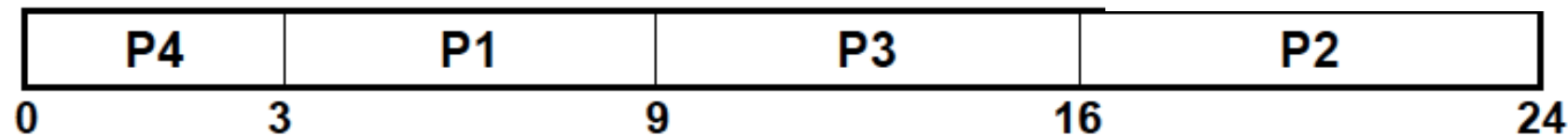
Le temps de cycle correspondant au temps qui sépare le début effectif de l'opération de lecture/ écriture et sa fin, soit le temps d'attente moyen (TAM)

| Processus | Durée d'exécution | Date d'arrivée |
|-----------|-------------------|----------------|
| P1 | 6 | 0 |
| P2 | 8 | 2 |
| P3 | 7 | 3 |
| P4 | 3 | 4 |

– Le **TAM (Temps d'Attente Moyen)** qui décrit la moyenne des délais d'attente pour commencer une exécution:

$$TAM = \sum_{i=1}^n TA_i / n, \text{ avec } TA_i = TR_i - \text{temps d'exécution}$$

Diagramme de Gantt:



$$TAM = ((3-4-3)+(9-0-6)+(16-3-7)+(24-2-8))/4 = 4.75$$

$$\text{Si on utilise FCFS, } TAM = ((6-0-6)+(14-2-8)+(21-3-7)+(24-4-3))/4 = 8$$

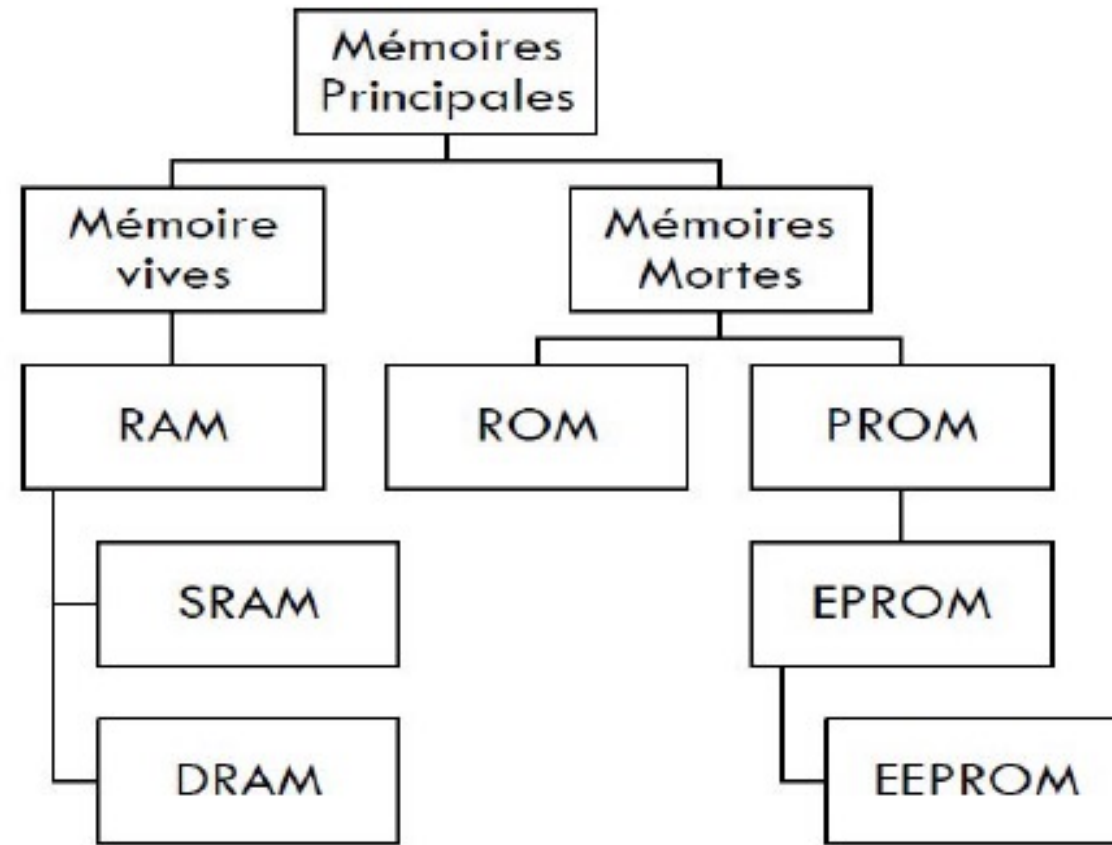
2. Les caractéristiques d'une mémoire

- 🕒 **Le format des données** : c'est le nombre de bits que l'on peut mémoriser par case mémoire ou encore largeur du mot mémorisable.
- 🕒 **La capacité** : c'est le nombre total de bits que peut contenir la mémoire. Il s'exprime aussi souvent en octet.
- 🕒 **Le temps d'accès** : c'est l'intervalle de temps entre l'instant du lancement d'une opération de lecture/écriture en mémoire et l'instant de la disponibilité de la première information sur le bus de données.
- 🕒 **Le temps de cycle** : il représente l'intervalle minimum séparant deux demandes successives de lecture ou d'écriture.

2. Les caractéristiques d'une mémoire

- 🕒 **Le débit** : c'est le nombre maximum d'informations lues ou écrites par seconde.
 - 🕒 **Volatilité** : elle caractérise la permanence des informations dans la mémoire. L'information stockée est volatile si elle risque d'être altérée par un défaut d'alimentation électrique et non volatile dans le cas contraire.
 - 🕒 **Le mode opératoire** : mode aléatoire (indexé, mode séquentiel.
 - 🕒 **Type d'accès** : lecture/écriture ou lecture seule.
- Les principaux critères à retenir sont** : La capacité, La vitesse, La consommation, Le coût.

3. La classification des mémoires



3. La classification des mémoires pour conserver les données: (non volatiles ou vives)

Une mémoire est un dispositif capable : d'enregistrer une information, de la conserver (memoriser) et de la restituer (possible de la lire ou la récupérer par la suite). Les mémoires peuvent être classées de deux manières

⌚ Selon la technologie utilisée : on distingue trois catégories

o Mémoire à semi-conducteur (mémoire centrale, ROM, PROM,.....) : très rapide mais de taille réduite.

o Mémoire magnétique (disque dur, disquette,...) : moins rapide mais stocke un volume d'informations très grand

o Mémoire optique (DVD, CDROM,...)

⌚ Selon son emplacement : Interne ou externe. La mémoire peut être dans le processeur (des registres), interne (Mémoire centrale ou principale) ou externe (Mémoire secondaire).

4. Les mémoires vives

La mémoire vive (RAM) est utilisable pour écrire ou lire des informations. Elle constitue la plus grande partie de la mémoire principale d'un ordinateur. C'est une mémoire à accès aléatoire et volatile. Il existe deux grandes familles de mémoires RAM :

les RAM statiques (SRAM) et les RAM dynamiques (DRAM).



4. Les mémoires vives

TABLEAU 1 TABLEAU COMPARATIF ENTRE DRAM ET SRAM

| Caractéristiques | DRAM | SRAM |
|-----------------------------------|--------------|-------------|
| Données enregistrées dans | Condensateur | Transistor |
| A besoin d'être rafraichie | Oui | Non |
| vitesse | Lente | Rapide |
| température | Froide | Chaude |
| Prix | Bas | haute |

5. Les mémoires mortes

- Les mémoires mortes ne sont normalement accessibles qu'en lecture. Ces mémoires sont non volatiles. On distingue 5 différents types de circuits de mémoires mortes :
- ROM
- PROM
- EPROM
- EEPROM
- FLASH EPROM

5. La mémoire morte ROM

- Elle est non volatile, un inconvénient principal est de ne pas pouvoir être manipulable ou modifiée, de plus le coût est élevé comparativement aux mémoires volatiles (RAM)

5. Les mémoires mortes : PROM

C'est une ROM qui peut être programmée une seule fois par l'utilisateur (Programmable ROM). Les PROM sont en fait des ROM "vierges" qui contiennent toutes les connexions possibles et sur lesquelles un appareil spécial, le programmeur de PROM permet de détruire certains fusibles internes (technique de claquage).

5. Les mémoires mortes : EPROM

- Pour faciliter la mise au point d'un programme ou tout simplement permettre une erreur de programmation, il est intéressant de pouvoir reprogrammer une PROM. L'EPROM (Erasable programmable ROM) est une PROM qui peut être effacée. L'avantage de l'utilisation des EPROM c'est qu'il s'agit d'un outil de stockage reprogrammable et non volatile

INCONVÉNIENT: il n'est pas possible de sélectionner une seule cellule à effacer

5. Electrically Erasable PROM (EEPROM)

Similaire au fonctionnement que le EPROM avec pour différence que...

l'effacement se fait à l'aide de signaux électriques, ce qui est plus rapide et pratique. Cette mémoire se comporte comme une RAM non volatile avec la possibilité d'une programmation et effacement mot par mot.

Inconvénients c'est qu'elle est très lente pour une utilisation en RAM et son coût de réalisation est assez élevé.

5. Les mémoires FLASH

Les mémoires FLASH sont similaires aux mémoires EEPROM, mais l'effacement peut se faire par blocs et ne nécessite pas le démontage du circuit.

6. La mémoire centrale

Pour pouvoir effectuer les opérations sur les données et exécuter des programmes, le processeur (unité centrale) dispose d'une mémoire centrale qui sert d'espace de traitement.

6. La mémoire centrale

La performance d'un ordinateur est généralement évaluée selon deux critères:

- La vitesse de traitement de l'information par le processeur central
- La capacité de rangement (stockage) de l'information en mémoire centrale.

6. La mémoire centrale

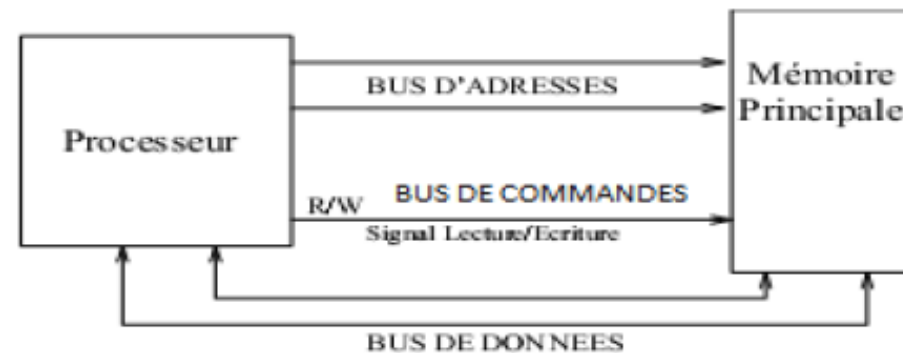
- ⌚ La mémoire centrale est une mémoire vive : accès en lecture et écriture.
- ⌚ La mémoire centrale est dite à accès aléatoire (RAM : Random Acces Memory) c'est-à-dire que le temps d'accès à l'information est indépendant de sa place en mémoire.
- ⌚ La mémoire centrale est volatile : la conservation de son contenu nécessite la permanence de son alimentation électrique.
- ⌚ Un temps d'accès à une mémoire centrale est moyen mais plus rapide que les mémoires magnétiques.
- ⌚ La capacité d'une mémoire centrale est limitée mais il y a toujours une possibilité d'une extension.

6. La mémoire centrale et le microprocesseur

Relation entre la mémoire centrale et le microprocesseur

La relation entre la mémoire centrale et le microprocesseur est établie via un câblage de bus spécialisés. Un bus est simplement un ensemble de n fils conducteurs, utilisés pour transporter n signaux binaires.

On distingue trois types de bus véhiculant des informations entre la mémoire centrale et le processeur : bus de données, bus d'adresse et le bus de contrôle.



6. Mémoire centrale, bus de données/adresses

Bus de données, bus de contrôle et bus d'adresses

- **Le bus de données :** Le bus de données est un bus bidirectionnel, son rôle consiste à transporter les données à lire ou à écrire de la mémoire centrale vers le microprocesseur ou du microprocesseur vers la mémoire centrale. Lors d'une lecture, c'est la mémoire qui envoie un mot sur le bus (le contenu de l'emplacement demandé) ; lors d'une écriture, c'est le processeur qui envoie la donnée. La taille du bus de données est égale au TCM.

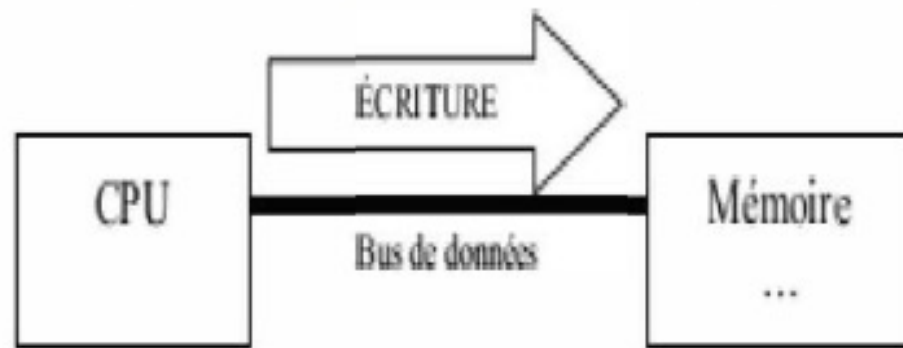
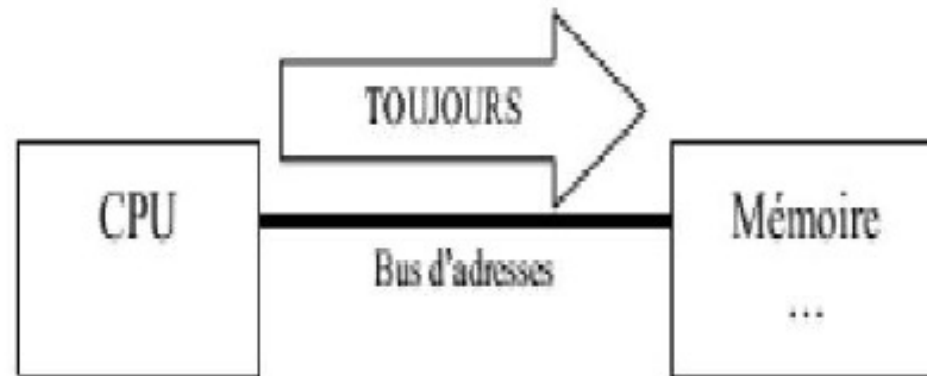


FIGURE 3 OPERATION D'ECRITURE AVEC LE BUS DE DONNEES

6. Mémoire centrale et bus d'adresses

- **Le bus d'adresses** : Le bus d'adresse est un bus unidirectionnel, son rôle consiste à transporter l'adresse de la case mémoire à lire ou à écrire. Seul le processeur envoie des adresses. Il est composé de m fils ; on utilise donc des adresses de m bits. La mémoire peut posséder au maximum 2^m emplacements (adresses 0 à $2^m - 1$).



6. Mémoire centrale et bus de commandes (contrôle)

- **Le bus de commandes (ou de contrôle)** : Le bus d'adresse est un bus unidirectionnel, son rôle consiste à transporter les commandes de lecture et écriture. Le bus de contrôle est souvent composé des fils suivants :
 - **CS (Chip Select) ou CE (Chip Enable)** : Permet de signaler que le composant est sélectionné et que les ordres qu'il reçoit sur les autres fils lui sont destinés. Cette entrée est généralement active au niveau bas.
 - **OE (Output Enable)** : Permet de demander au composant de valider ses sorties, car les fils de données ont en général la particularité d'être à trois états (niveau bas, niveau haut ou haute impédance). Cette entrée est généralement active au niveau bas.
 - **RD/W (Read/Write) ou en français L/E** : Permet au microprocesseur d'informer un composant de la nature de l'échange : lecture ou écriture. La dénomination RD/W a pour signification :
 - RD : Read, accès en lecture lorsque ce fil est à l'état 1.
 - /W : Write, accès en écriture lorsque l'état est 0.

6. La mémoire centrale

L'adresse basse dite aussi adresse de début : représente l'adresse de début de la case numéro 1 elle est représentée en binaire par une suite de m bits (nombre de bits d'adresse) de 0

L'adresse haute ou aussi l'adresse de fin : représente l'adresse de début de la case numéro $N-1$ elle est représentée en binaire par une suite de m bits (nombre de bits d'adresse) de 1

6. La mémoire centrale

la plus petite composante électronique, le bit représente deux états

0

1

Vrai

Faux

Oui

Non

Pair

Impair

Blanc

Noir

...

...

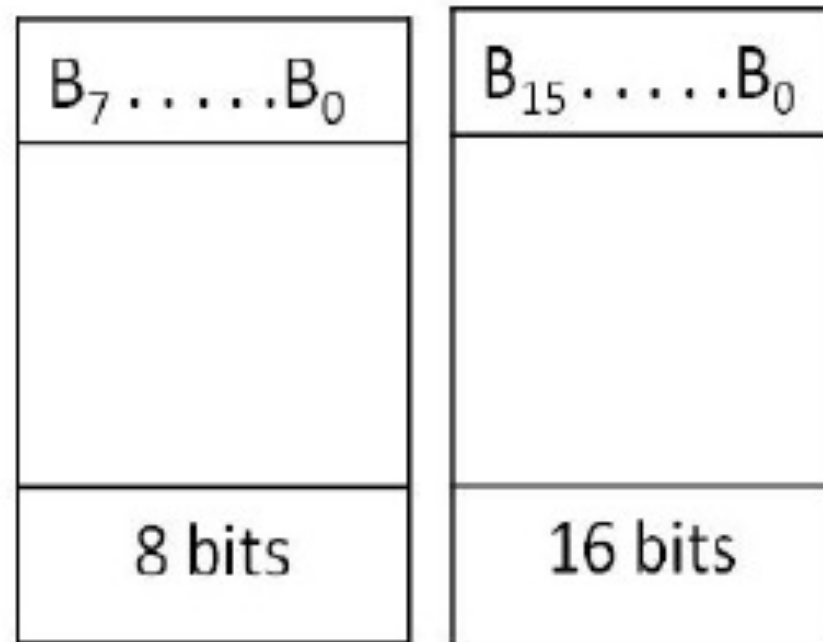
6. La mémoire centrale

La mémoire centrale peut être vue comme un large tableau divisé en emplacements (cases) de taille fixe utilisés pour stocker instructions et données. La taille d'un emplacement mémoire noté TCM pourrait être quelconque (toujours multiple de 2) doit être la même pour toutes les cases d'un ordinateur. En fait, la plupart des ordinateurs utilisent des emplacements mémoire d'un octet (1 byte = 8 bits).

Tightly Coupled Memory (TCM) provides low-latency memory accesses that the core can use without the unpredictability of access time that is a feature of caches

6. La mémoire centrale

Exemple : une mémoire centrale avec une TCM= 8 bits et une autre avec TCM=16 bits

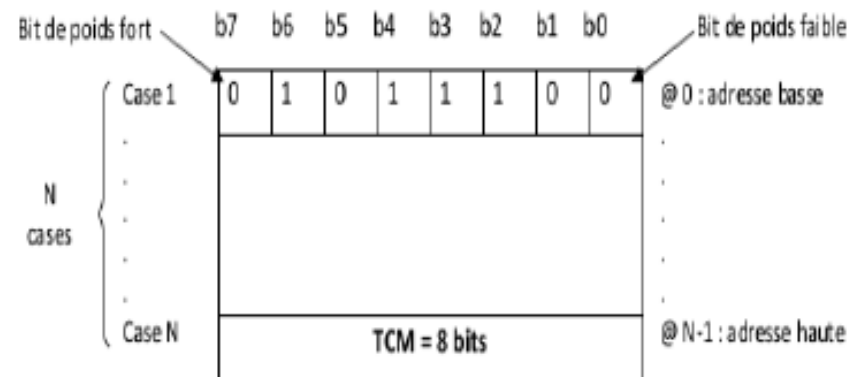


Chaque case mémoire est référencée par un numéro unique appelé: **adresse mémoire** (généralement écrite en hexadécimal). Les adresses sont séquentielles (consécutives).

6. La mémoire centrale

Exemple D'UNE STRUCTURE DE MEMOIRE CENTRALE AVEC TCM = 8 BITS

Toutes les adresses des cases mémoires nécessitent le même nombre de bits pour les écrire : m bits, ou m est considéré comme le nombre de bits d'adresse (nombre de lignes d'adresse). Le nombre de bit m dépend du nombre des cases dans la mémoire centrale. Si la mémoire contient N cases alors les adresses de ces cases sont numérotées de 0 à $N-1$.



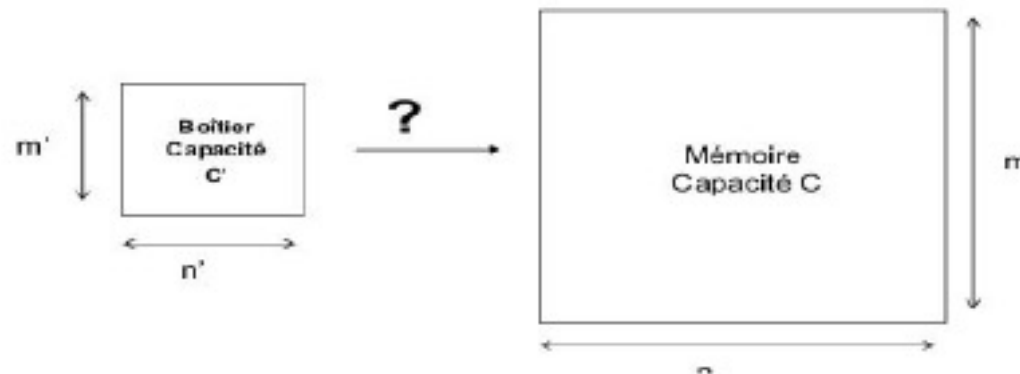
6. La mémoire centrale

Conception d'une mémoire centrale

Le problème avec la mémoire centrale c'est qu'on veut réaliser une mémoire de capacité C , mais nous disposons uniquement de boîtiers de taille inférieure.

Les techniques d'intégration ne permettent pas d'obtenir des boîtiers ayant des capacités ou des formats suffisants pour toutes les applications. Il est alors nécessaire d'associer plusieurs boîtiers pour augmenter la longueur des mots ou le nombre de mots. L'association de plusieurs blocs peut permettre d'améliorer les performances temporelles de la mémoire en faisant fonctionner plusieurs blocs en parallèle.

$$\text{Nombre de puces} = \text{Taille mémoire} / \text{Taille puce}$$



6. La mémoire centrale

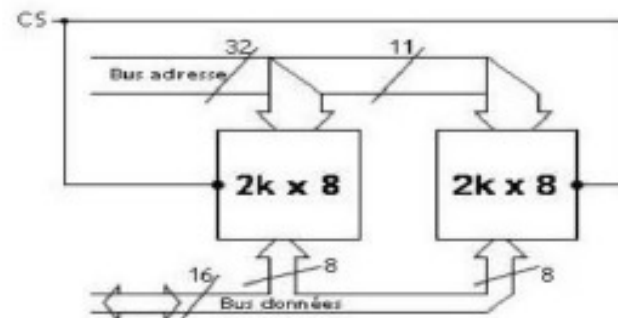
Assemblage Horizontal

(puces en Parallèle placées sur la même ligne → nombre de puces parallèles = nombre de colonnes)
Augmentation de taille case mémoire

On effectue une mise en parallèle des blocs mémoires élémentaire afin d'augmenter la largeur des mots, c'est-à-dire la largeur du bus de données, le bus d'adresse commun aux blocs élémentaire restant inchangé. On pourrait par exemple associer en parallèle 4 blocs mémoires de 8 bits pour former des mots mémoires de 32 bits.

Pour déterminer le nombre de boîtier nécessaire pour obtenir la taille de la case de la mémoire M (extension cases ou extension colonnes) ou les puces placées en parallèles on applique la formule suivante.

$$\text{Nombre de puces parallèles} = TCM / TCP$$



6. La mémoire centrale

Assemblage Vertical

(Puces en série placées sur la même colonne → nombre de puces séries = nombre de lignes) : Augmentation espace d'adressage

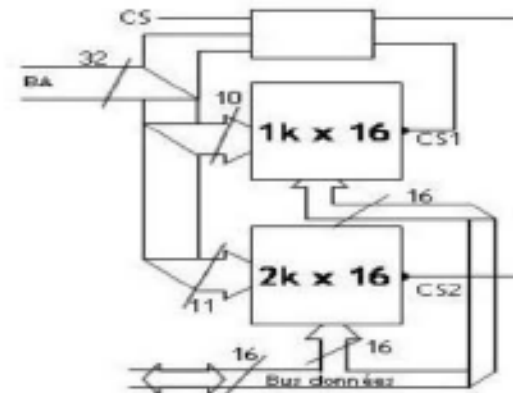
On effectue une association série des blocs mémoires pour augmenter le nombre de mots mémoires, c'est-à-dire la capacité de la zone adressable. Pour ce faire on augmente la largeur du bus d'adresse.

Pour déterminer le nombre de puces en série (extension lignes) nous appliquons la formule suivante :

$$\text{Nombre de puces séries} = \text{Nbr total de puces} / \text{Nbr de puces parallèles}$$

Ou

$$\text{Nombre de puces séries} = \text{espace d'@ Zone} / \text{espace d'@ Zone puce}$$



6. La mémoire centrale

Pour une mémoire avec 5 ligne d'adresse

@début=@basse = m '0' = 5 '0' = 000002 = 00/H

@fin=@haute = m '1' = 5 '1' = 111112 = 1F/H

On effet on peut représenter le nombre de cases mémoire par 2^m cases, on dit que le nombre de case mémoire directement adressable est de 2^m (espace d'adressage directe)

6. La mémoire centrale

la taille de la mémoire centrale noté **TMC** dépend du l'espace d'adressage et de la taille d'une casse mémoire on obtien donc la formule suivante :

$$\text{Taille de la mémoire centrale} = \text{espace d'adressage} \times \text{TCM}$$

$$\text{TMC} = 2^m \times \text{TCM}$$

La taille de la mémoire centrale est généralement exprimée en kilo ø, Méga ø, Giga ø, rappelons ces mesures

| | |
|------------|---------------------------------------|
| 1 K (Kilo) | $2^{10} = 1024$ bites |
| 1 M (Mega) | $2^{20} = 1\,048\,576$ bits |
| 1 G (Giga) | $2^{30} = 1\,073\,741\,824$ bits |
| 1 T (Téra) | $2^{40} = 1\,099\,511\,627\,776$ bits |

6. La mémoire centrale

Exercice 1

Si $N = 4$ cases alors les adresses varient de 0 à 3

L'espace d'adressage direct $= 4 = 2^2$

Alors une adresse s'écrit sur 2 bits

$TMC = 2^2 \times 1 \text{ } \emptyset = 4 \text{ } \emptyset$

@début = 2 bits de 0 $= 00_2 = 0_{16}$

@fin = 2 bits de 1 $= 11_2 = 3_{16}$

6. La mémoire centrale

Exercice 2

Soit une mémoire centrale de 64 M \emptyset et TCM = 4 \emptyset

Déterminer la capacité d'adressage de cette mémoire

6. La mémoire centrale

On sait que

$$\text{TMC} = N \times \text{TCM} \quad 64 \text{ M}\emptyset = N \times 4 \emptyset \rightarrow N = \frac{64 \text{ M}\emptyset}{4 \emptyset} = \frac{2^6 \times 2^{20} \times 2^3}{2^2 \times 2^3} = 2^{24} \text{ cases}$$

m=24 lignes d'adresse

@début = 24 bits de 0 = 00000₁₆

@fin = 24 bits de 1 = FFFFFFF₁₆

6. La mémoire centrale

Exercice 3

Soit une mémoire centrale de 224 ø

Déterminer la capacité d'adressage de cette mémoire ainsi que le nombre de lignes d'adresse.

6. La mémoire centrale

On sait que $TMC = N \times TCM$

$$224 \text{ } \varnothing = N \times 1 \text{ } \varnothing \rightarrow N = \frac{224 \text{ } \varnothing}{1 \text{ } \varnothing} = 224 \text{ cases}$$

Nombre de lignes d'adresse $2^7 \leq m \leq 2^8$ c'est-à-dire $128 \leq m \leq 256$

D'où $m = 8$ bits

6. La mémoire centrale

Exercice récapitulatif :

Soit une mémoire centrale de 48 k d'octets adressable par 8 bits, on dispose des puces mémoires suivantes : 8k*16, 32k*4, 8k*8, 8k*2 et 16k*32

1. Proposer une organisation de cette mémoire
2. Déterminer l'adresse de fin et de début de cette mémoire
3. Déterminer l'adresse de début et de fin de chaque zone.
4. Dresser la carte d'adressage et déterminer le nombre de bite fixe pour chaque zone et déduire le décodeur à utiliser

Je vous remercie de votre attention

Rappel: Travail #1 remise 14 février 23h59

Préparation pour la semaine prochaine:
Système d'exploitation et
gestion des fichiers et machines virtuelles

-Lecture recommandée:

Laurent Bloch

Bonne semaine!