

## Université Sultan Moulay Slimane Ecole Supérieure de Technologie Beni Mellal



Matière : Architecture des ordinateurs

DUT-ARI – 1ere année – S1

# Chapitre 2 : Composants principaux d'un ordinateur

Cours donné par:

Dr. Abderrahmane ELBALAOUI

# Plan

- La carte mère
- 2 Le processeur
- 3 Les Bus
- 4 Les mémoires

#### 1. Carte mère

- La carte mère (en anglais « Motherboard ») est un circuit imprimé constituant le support principal des éléments essentiels d'un ordinateur. Elle permet les échanges d'informations entre les différents composants vitaux d'un PC.
- La carte mère est le centre nerveux d'un ordinateur ou lieu d'échange de données et de calcul.

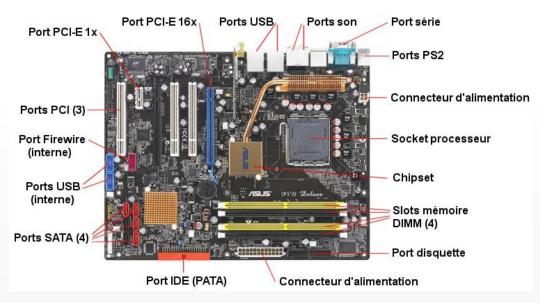


Figure: Principaux composants d'une carte mère

#### Fabricants populaires

- Intel
- ASUS
- AOpen
- UN PEU
- Biostar
- Gigaoctet
- MSI

Une carte mère comporte de nombreux composants électroniques soudés sur la carte et reliés entre eux par les connexions du circuit imprimé et par de multiples connecteurs:

## Le support du processeur

La carte mère possède un emplacement pour accueillir le processeur, appelé support de processeur (Socket).

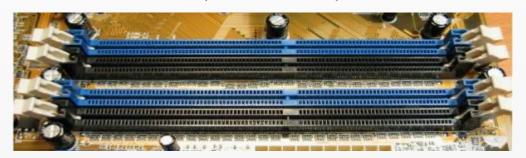
Socket: il s'agit d'un connecteur carré possédant un grand nombre de petits connecteurs sur lequel le processeur vient directement s'enficher.



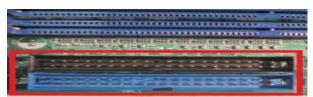
- Chipset: circuit électronique chargé de coordonner les échanges de données entre les divers composants de l'ordinateur (processeur, mémoire...).
- CMOS: C'est un type de puce capable de stocker des informations et de les conserver même quand l'ordinateur est éteint. Leur contenu est maintenu par un faible courant électrique fourni par une pile.



Les connecteurs de mémoire vive (RAM): les connecteurs mémoire (slots en anglais) se trouvent au nombre de deux, quatre, six ou plus rarement huit.

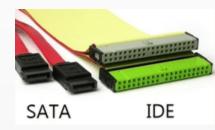


- Les connecteurs de stockage : sont des connecteurs spécifiques présents sur toutes les cartes mères, permettant de lui adjoindre des périphériques de stockage de masse (disque dur, lecteur de disque optique, disque SSD). On en trouve trois types :
  - Le connecteur Floppy : permet de connecter un lecteur de disquettes à la carte mère.
  - Les connecteurs IDE (Integrated Drive Electronics): Les connecteurs IDE (aussi appelé PATA pour Parallel ATA), destinées à connecter les anciens disques à la norme IDE.



Les connecteurs SATA : est un bus standard permettant la connexion de périphériques de stockage haut débit.





Les connecteurs d'entrées/sorties : La carte mère de l'ordinateur possède un certain nombre de connecteurs d'entrées-sorties ou panneau d'entrées/sorties regroupés sur le « panneau arrière».

- Les ports USB: Les ports USB (Universal Serial Bus) permettent de connecter la quasi totalité du matériel récent (clés USB, imprimantes,...).
- Le connecteur RJ45

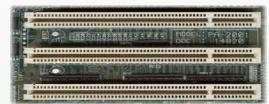


- Le connecteur VGA: Est un standard d'affichage pour ordinateurs.
- Le connecteur DVI: Est un type de connexion vidéo numérique.
- Le connecteur HDMI: est une norme et interface audio/vidéo totalement numérique.
- les connecteurs PCI: est un standard de bus local permettant de connecter des cartes d'extension.



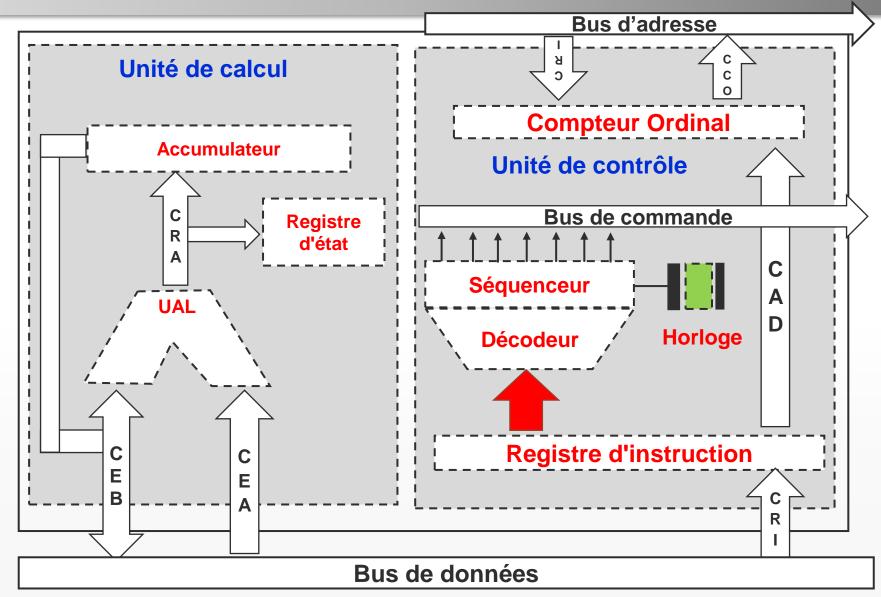






- Le processeur ou CPU (Central Processing Unit) est le composant de votre ordinateur qui exécute les instructions qui lui sont données par votre système d'exploitation.
- C'est l'un des composants les plus importants d'un ordinateur. C'est, en quelque sorte, le cerveau de l'ordinateur.
- Le terme microprocesseur désigne un processeur fait d'un seul circuit intégré. On les trouve généralement dans les machines au format réduit (les smartphones, les tablettes ...)
- Les processeurs récents sont généralement dotés de plusieurs cœurs. Ils peuvent alors traiter simultanément plusieurs instructions.

Plus un processeur a de cœur, plus la machine sera rapide, et plus elle pourra traiter de logiciels en même temps. Structure interne d'un processeur



- Structure interne d'un processeur : Le processeur est constitué d'un ensemble d'unités fonctionnelles reliées entre elles :
  - L'unité de contrôle : Est la partie la plus complexe du processeur. Elle se décompose en plusieurs parties dont elle doit assurer la coordination :
    - Registre d'instruction : Le processeur exécute une à une les instructions stockées dans la mémoire centrale. Pour cela, les instructions doivent être chargées dans le processeur. Le registre d'instruction contient donc l'instruction courante à exécuter.
    - Le décodeur : l'instruction à exécuter, chargée dans le registre d'instruction, est interprétée par le décodeur.
    - Le séquenceur : est alors capable d'ordonnancer les diverses opérations élémentaires du processeur, nécessaires pour exécuter l'instruction, grâce à un microprogramme.

- Le compteur ordinal : est un registre particulier qui contient à tout instant, l'adresse de l'instruction suivante à exécuter.
- **L'horloge :** est un dispositif qui détermine le rythme dans lequel sont exécutées les instructions. Elle fournit un signal régulier au processeur.
- Le registre d'état représente à tout moment l'état du processeur. En effet, divers événements peuvent créer des situations « exceptionnelles ». Par exemple, l'addition de deux valeurs peut dépasser les capacités de représentation du processeur. Dans ce cas, un bit du registre d'état signale que le processeur est dans l'état « overflow ».

- L'unité de calcul: Comme son nom l'indique, effectue tous les calculs au sein du processeur. À côté des opérations arithmétiques, elle peut aussi procéder à des opérations logiques.
  - l'unité arithmétique et logique : (en anglais : Arithmetic and Logic Unit ALU) se charge de réaliser les opérations arithmétiques et logiques.
  - L'accumulateur: est le registre de calcul par excellence. C'est par lui en effet que transitent toutes les données devant faire l'objet d'une opération ainsi que tous les résultats produits par calcul arithmétique.
  - l'accumulateur est secondé par un nombre variable de registres généraux destinés à la réalisation matérielle des calculs.

- Les bus de communication : Pour communiquer avec son environnement, le processeur dispose d'un ensemble de « bus » :
  - Le bus d'adresses : permet de désigner l'adresse d'un ou de plusieurs octets à charger dans le processeur
  - Le bus de données : permet de désigner la valeur à charger dans le processeur ou à stocker en mémoire.
  - Le bus de commande : permet au processeur de désigner l'opération à effectuer, chargement dans le processeur.

- Les registres : L'unité centrale comprend un certain nombre de registres :
  - **L'accumulateur :** Un accumulateur est un registre de travail de 8 ou 16 bits qui sert:
    - à stocker une opérande au début d'une opération arithmétique et le résultat à la fin de l'opération.
    - à stocker temporairement des données en provenance de l'extérieur du microprocesseur avant leur reprise pour être rangées en mémoire.
    - à stocker des données provenant de la mémoire ou de l'UAL pour les présenter vers l'extérieur du microprocesseur.
  - Le compteur de programme (CP): Le compteur de programme contient l'adresse (décalage ou offset) de l'instruction suivante en mémoire qui doit être exécutée. Autrement dit, il doit indiquer au processeur la prochaine instruction à exécuter
  - Le registre d'instruction :

Lorsqu'une instruction est récupérée en mémoire pour être exécuté dans le processeur, elle est mémorisée dans un registre spécial, le registre d'instruction (RI).

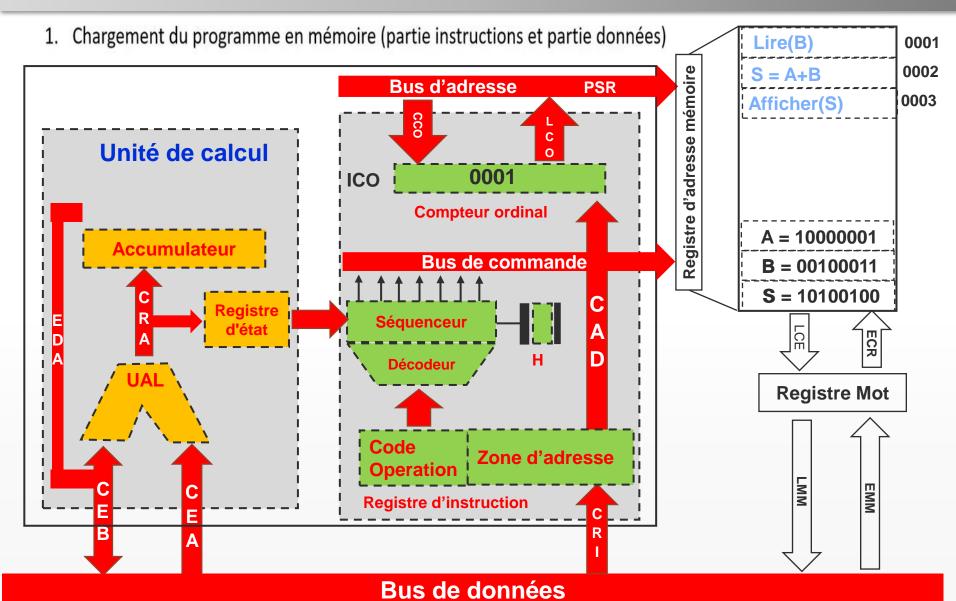
- Les registres : L'unité centrale comprend un certain nombre de registres :
  - **L'accumulateur :** Un accumulateur est un registre de travail de 8 ou 16 bits qui sert:
    - à stocker une opérande au début d'une opération arithmétique et le résultat à la fin de l'opération.
    - à stocker temporairement des données en provenance de l'extérieur du microprocesseur avant leur reprise pour être rangées en mémoire.
    - à stocker des données provenant de la mémoire ou de l'UAL pour les présenter vers l'extérieur du microprocesseur.
  - Le compteur de programme (CP): Le compteur de programme contient l'adresse (décalage ou offset) de l'instruction suivante en mémoire qui doit être exécutée. Autrement dit, il doit indiquer au processeur la prochaine instruction à exécuter
  - Le registre d'instruction : Lorsqu'une instruction est récupérée en mémoire pour être exécuté dans le processeur, elle est mémorisée dans un registre spécial, le registre d'instruction (RI).

- Les registres : L'unité centrale comprend un certain nombre de registres :
  - Le décodeur d'instruction : C'est lui qui va interpréter l'instruction contenue dans le registre d'instruction (RI). C'est-à-dire qu'elle est l'opération à effectuer (Addition, branchement etc...).
  - Registres d'adresses : Ces registres servent à gérer l'adressage de la mémoire.
  - Registre d'état : Le registre d'état sert à contenir l'état de certaines opérations effectuées par le processeur.

Soit le programme suivant :

```
entier A = 10000001
entier B,S
Lire(B)
S = A+B
Afficher(S)
```

Donner les étapes détaillées de l'exécution de ce programme :



**Chapitre 2: Principaux composants d'un ordinateur** 

- 2. Le compteur ordinal contient l'adresse de l'instruction à exécuter (0001);
- 3. L'instruction **Lire(B)** passera par LEC, Registre Mot, Bus de donnée, et puisque c'est une instruction, elle passera par CRI vers le registre d'instruction.
  - 4. Le registre d'instruction contient l'instruction en cours d'exécution (lire(B)),
    Instruction = code opération + adresse opérande, Pendant que l'adresse de l'opérande
    B est envoyée à le bus d'adresse, le code opération est transmis au décodeur qui
    détermine le type d'opération demandée (Lire) et le transmet au séquenceur.
    - La mémoire tampon de clavier est ouverte en écriture;

- 5. La donnée arrivant du clavier passera par EMM, Registre Mot, ECR et sera écrite dans **B**;
- 6. Le compteur ordinal est incrémenté (0002);
- 7. L'instruction **S** = **A**+**B** passera du même chemin sauf que le séquenceur la divisera en quatre opérations :
  - a. Récupérer la valeur de A de la mémoire;
  - b. Récupérer la valeur de **B** de la mémoire;
  - c. Calculer la somme;
  - d. Écrite le résultat de la somme dans S.

# le fonctionnement du processeur

- 7. a. Récupérer la valeur de **A** de la mémoire :
  - Ouvrir A en lecture
  - Sa valeur passe par LEC, Registre Mot, Bus de données, et puisque c'est une donnée, elle passera par CEA a l'UAL.
  - Comme l'opération est une addition, il manque une deuxième opérande, la valeur de A passera par CRA et sera sauvegardée temporairement dans l'accumulateur;
- 7. b. Récupérer la valeur de **B** de la mémoire :

Même chose que 7.a. sauf quand elle arrivera à L'UAL, la valeur de A ( qui est dans l'accumulateur passera par EDA puis CEB.

- 7. c. L'UAL calculi la somme et met le résultat dans l'accumulateur
- 7. d. Écrire le résultat de la somme dans **S**:

Ouvrir S en écriture, la valeur dans l'accumulateur passe par EDA, Bus de données, EMM, Registre Mot, ECR et sera écrite dans **S**;

- 8. Le compteur ordinal est incrémenté (0003);
- 9. Afficher (S) passe jusqu'au registre instruction;
  - La mémoire tampon de l'écran est ouverte en écriture;
  - S passera par LEC, Registre Mot, Bus de données et sera écrite dans la mémoire tampon de l'écran puis affichée.

- LCO: Lecture du Compteur Ordinal. Transfère le contenu du compteur ordinal sur le bus d'adresses.
- CCO: Chargement du Compteur Ordinal. Transfère le contenu du bus d'adresses dans le compteur ordinal.
- **PSR** : Pointage Sur Registre. Transfère le contenu du bus d'adresses dans le registre adresse mémoire.
- LEC: Lecture. Transfère le contenu de l'emplacement mémoire dans le registre mot.
- ECR : Écriture. Transfère le contenu du registre mot mémoire dans l'emplacement mémoire repéré par l'adresse stockée dans le registre adresse mémoires.

- LMM: Lecture Mot Mémoire. Transfère le contenu du registre mot mémoire sur le bus de données.
- **EMM** : Écriture Mot Mémoire. Transfère le contenu du bus de données dans le registre mot mémoire.
- CRI: Chargement Registre Instruction. Transfère le contenu du bus de données dans le registre instruction.
- CAD: Chargement Adresse. Transfère la partie adresse (opérande) de l'instruction sur le bus d'adresse.
- CRA: Chargement Registre Accumulateur. Le résultat d'une opération dans l'UAL transfère dans l'accumulateur.

- CEA : Chargement Entrée A. Transfère le contenu du bus de données sur l'entrée A de l'UAL.
- CEB : Chargement Entrée B. Transfère le contenu de l'accumulateur sur l'entrée B de l'UAL.
- EDA: Envoi de Données Accumulateur. Transfère le contenu de l'accumulateur sur le bus de données.
  - ICO: Incrémentation (+1) du Compteur ordinal.

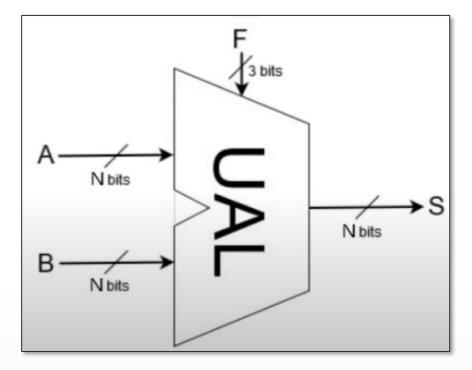
#### Fonctionnement de l'UAL

- L'UAL est un circuit combinatoire responsable des opérations arithmétiques et logiques.
- Elle a 2 entrées (sur N bits) comme opérandes(A,B), et une sortie de résultat (D).
- Elle possède aussi une entrée de commande (F)
  permettant de choisir l'opération à faire.
- Une table des codes des opérations pour l'entrée des commandes est associée a l'UAL
- Elle met par ailleurs à jour des indicateurs (ou drapeaux ou flags) en fonction du résultat de l'opération effectuée :
- F R
- Zéro flag (Z) : indicateur mis à 1 si le résultat de l'opération est 0.
- Négative flag (N): indicateur mis à 1 pour un résultat négatif
- Carry flag (C) : mis à 1 en cas de retenue
- Overflow flag (V): mis à 1 en cas de débordement en contexte signé

#### Construction de L'UAL

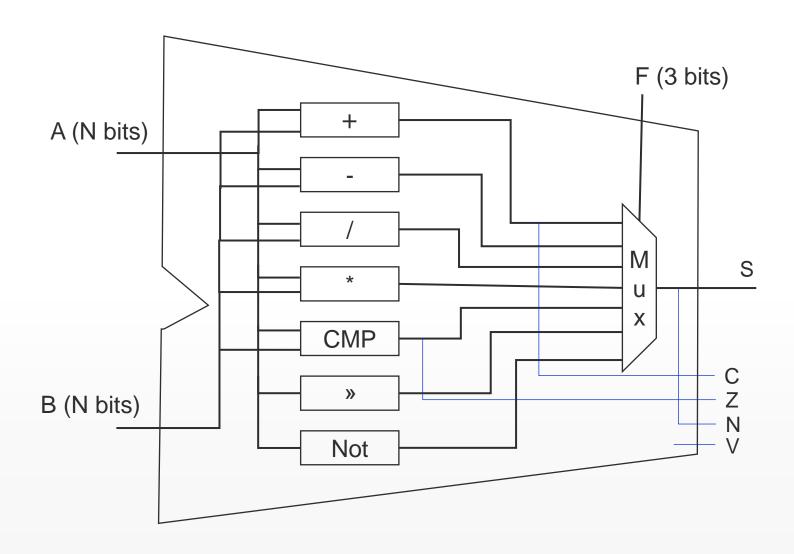
- Pour construire un UAL il faut créer un circuit combinatoire pour chaque opération
- Les arranger en parallèle l'un à cote de l'autre, et utiliser un multiplexeur pour choisir la sortie de l'opération sélectionnée par la commande.
- Rajouter les sorties des Flags, ce sont des indicateurs d'état sur l'opération actuelle.

# Fonctionnement de l'UAL



F3	F2	F1	fonctions
0	0	0	+
0	0	1	-
0	1	0	1
0	1	1	*
1	0	0	CMP
1	0	1	<b>»</b>
1	1	0	<b>«</b>
1	1	1	Not

# Construction de L'UAL



**Chapitre 2: Principaux composants d'un ordinateur** 

# Caractéristiques d'un processeur

#### Les cœurs

Les cœurs permettent d'effectuer des travails en parrallèle. Cela rend le processeur plus rapide et plus efficace.

## > Fréquence

La fréquence s'exprime en GigaHertz (GHz), elle signifie le nombre d'opérations que fait le processeur en une seconde. 3GHz = 3 milliards d'opération à la seconde. Actuellement, les processeurs tournent entre 1.5 et 3.5 GHz.

Attention toutefois, on ne peut comparer la fréquence que de 2 processeurs conçus sur la même architecture ! Par exemple, un Core i5 750 à 2.66GHz est plus puissant qu'un Core i3 650 à 3GHz

# Caractéristiques d'un processeur

#### > Le cache

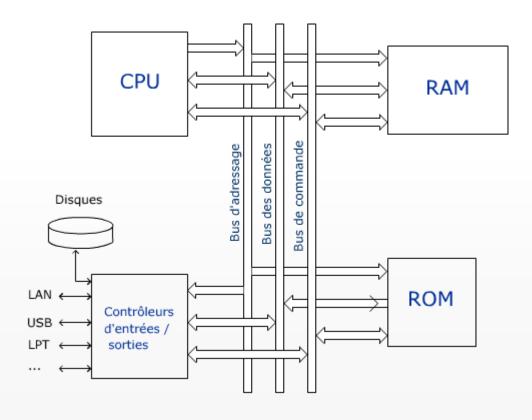
- La mémoire cache est un type de mémoire vive (RAM) à laquelle le processeur peut accéder plus rapidement qu'à la mémoire RAM habituelle. Généralement, cette mémoire cache est directement intégrée dans la puce de l'unité centrale (UC) ou placée sur une puce distincte dotée d'une interconnexion par bus à l'UC.
- C'est une petite quantité de mémoire très rapide qui est utilisée pour le stockage temporaire. Plus le cache d'un processeur est volumineux, plus il pourra stocker de fichiers pour cette récupération rapide.

#### > Le socket

La compatibilité des sockets est une préoccupation majeure lors de l'achat d'un processeur. La compatibilité socket active l'interface entre une carte mère et son CPU.

#### 3. Les bus

Un bus est un 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)



**Chapitre 2: Principaux composants d'un ordinateur** 

# Caractéristiques d'un bus

La largeur : un bus est caractérisé par le volume d'informations transmises simultanément (exprimé en bits). La largeur désigne le nombre de bits qu'un bus peut transmettre simultanément.

1 bus à n fils = 1 bus à n bits

**Exemple**: une nappe de 32 fils permet ainsi de transmettre 32 bits en parallèle.

La vitesse : est le nombre de paquets de données envoyés ou reçus par seconde. Elle est également définie par sa fréquence (exprimée en Hertz). On parle de cycle pour désigner chaque envoi ou réception de données.

Cycle mémoire (s) = 1/ fréquence

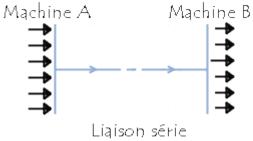
➤ Le débit : Le débit maximal du bus (ou le taux de transfert maximal) est la quantité de donnée qu'il peut transfert par unité de temps ; en multipliant sa largeur par sa fréquence.

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

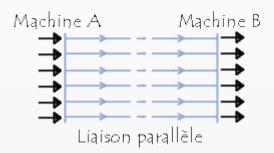
# Types de bus de données

Il existe deux grands types de bus de données selon le type de transmission :

Les bus séries : ils permettent des transmissions sur de grandes distances. Ils utilisent une seule voie de communication sur laquelle les bits sont envoyés les uns à la suite des autres. (câble SATA)



Les bus parallèles : sur un bus parallèle plusieurs bits sont transmis simultanément. Ils sont utilisés sur des distances courtes ; par exemple ; pour relier le processeur ; la mémoire. (câble IDE)

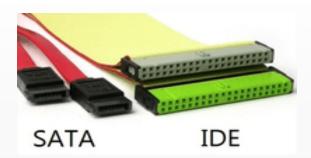


# Les principaux bus

On distingue généralement sur un ordinateur deux principaux bus :

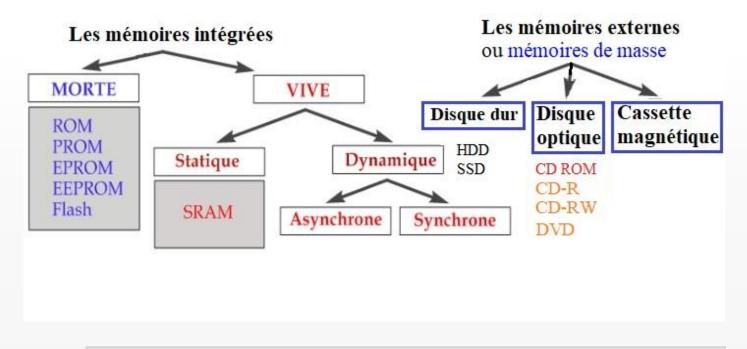
Le bus système (bus interne) permet au processeur de communiquer avec la mémoire centrale du système (mémoire vive ou RAM).

Le bus d'extension permet aux divers composants liés à la carte-mère (USB, série, parallèle, cartes branchées sur les connecteurs PCI, disques durs, lecteurs et graveurs de CD-ROM, etc.) de communiquer entre eux. Il permet aussi l'ajout de nouveaux périphériques grâce aux connecteurs d'extension qui lui y sont raccordées.



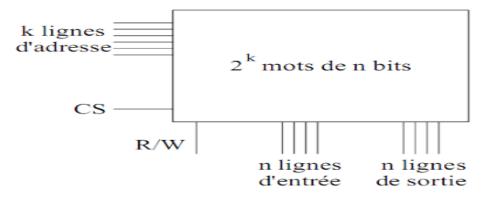
#### 3. La mémoire d'un ordinateur

- Une mémoire est un circuit à semi-conducteur permettant d'enregistrer, de conserver et de restituer des informations (instructions et variables). C'est cette capacité de mémorisation qui explique la polyvalence des systèmes numériques et leur adaptabilité à de nombreuses situations. Les informations peuvent être écrites ou lues. Il y a écriture lorsqu'on enregistre des informations en mémoire, lecture lorsqu'on récupère des informations précédemment enregistrées.
- On distingue ainsi deux grandes catégories de mémoires :



### Organisation d'une mémoire

On peut donc schématiser un circuit mémoire par la figure suivante :



- La taille d'un bloc mémoire est donc  $2^k$ , le premier mot se situant à l'adresse 0 et le dernier à l'adresse  $2^k$  1.
- Une ligne de commande (R/W) indique si la mémoire est accédée en écriture (l'information doit être mémorisée) ou en lecture (l'information doit être restituée).
- Sur ce schéma on distingue deux canaux de n lignes en entrée et en sortie, mais dans d'autres cas les accès en entrée et en sortie peuvent être confondus en un seul canal bidirectionnel.
- Une entrée de sélection du circuit (CS).

# Organisation d'une mémoire

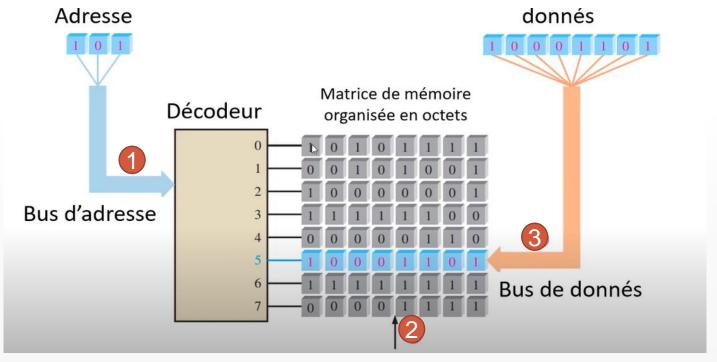
- La Ram peut être vue comme un large tableau de mots;
- Un mémoire stocke une information sur n bits;
- Chaque mot possède sa propre adresse;
- une adresse est un numéro unique qui permet d'accéder a un mot mémoire;
- Les adresses sont séquentielles (consécutives);
- La taille de l'adresse ( le nombre de bits)
   dépend de la capacite de la RAM.

		_	_	_	
0000					
0001					
0010					
0011					
0100					
0101					
0110					
0111					
1000					
1001					
1010					
1011					
1100					
1101					
1110					
1111					

# Organisation d'une mémoire

Une opération de lecture ou d'écriture s'effectue avec les étapes suivantes :

- − la sélection de l'adresse ;
- − le choix entre lecture et écriture (niveau appliqué sur R/W) ;
- la sélection du circuit (niveau appliqué sur CS) ;
- la lecture ou l'écriture de la donnée.



Écriture

# Caractéristiques d'une mémoire

La capacité : c'est le nombre total de bits que contient la mémoire. Elle s'exprime aussi souvent en octet.

Capacité (en bits) =  $2^{nombres de lignes d'adresse}$  \* nombre de lignes de donnée

Le format des données : c'est le nombre de bits que l'on peut mémoriser par case mémoire. On dit aussi la largeur du mot mémorisable (mot de 8, 16, 32 ou 64 bits).

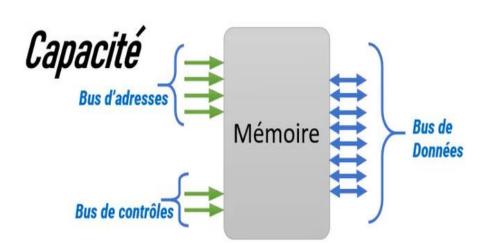
Le temps d'accès : c'est le temps qui s'écoule entre l'instant où a été lancée une opération de lecture/écriture en mémoire et l'instant où la première information est disponible sur le bus de données

Le temps de cycle : il représente l'intervalle minimum qui doit séparer deux demandes successives de lecture ou d'écriture.

Le débit : c'est le nombre maximum d'informations lues ou écrites par seconde.

### 3. La mémoire d'un ordinateur

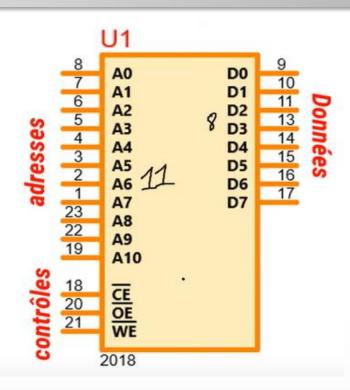
# Exemple 1 d'une mémoire :



Capacité =  $8*2^4$  = 128 bits

			_	_	_	
0000						
0001						
0010						
0011						
0100						
0101						
0110						
0111						
1000						
1001						
1010						
1011						
1100						
1101						
1110						
1111						

# Exemple 2 d'une mémoire :



### La mémoire vive :

- La mémoire vive RAM (Random Access Memory): est la mémoire informatique dans laquelle peuvent être stockées, puis effacées, les informations traitées par un appareil informatique.
- On distingue généralement deux grandes catégories de mémoires vives :
  - Les mémoires dynamiques (DRAM, Dynamic Random Access Memory)
  - Les mémoires statiques (SRAM, Static Random Access Memory)

### La mémoire vive :

#### Pour SRAM :

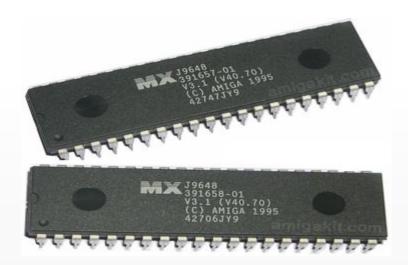
- Dans la mémoire vive statique ou **SRAM** la cellule de base est constituée par une bascule de transistors.
- Le terme de statique fait référence à leur fonctionnement interne. Elles ne nécessitent quasiment pas de rafraichissement.
- Dans la mesure où ce rafraichissement à un coût en temps, cela explique pourquoi ce type de mémoire est très rapide, entre 6 et 15 ns, mais assez chère.
- On utilisera donc essentiellement pour des mémoires de faible capacité comme dans la mémoire cache pour les microprocesseurs.

#### Pour SRAM:

- Dans la mémoire vive dynamique ou **DRAM**; la cellule de base est constituée par un condensateur et un transistor
- Son inconvénient réside dans les courants de fuite des pico-condensateurs (Petit condensateur permettant de gérer un bit de mémoire) : l'information disparaît à moins que la charge des condensateurs ne soit rafraîchie avec une période de quelques millisecondes d'où le terme de dynamique.

## Les mémoires mortes (ROM)

- Comme son nom l'indique, les données de la mémoire ne peuvent être lues que par l'ordinateur.
- La **ROM** (Read-Only Memory) est une mémoire non volatile (il n'oublie pas les données même si l'alimentation est retirée), elle sert à stocker le micrologiciel du matériel qui n'obtient pratiquement aucune mise à jour, par exemple le **BIOS**.



# Les Types de ROM:

**ROM** : Elle est programmée par le fabricant et son contenu ne peut plus être ni modifié, ni effacé par l'utilisateur.

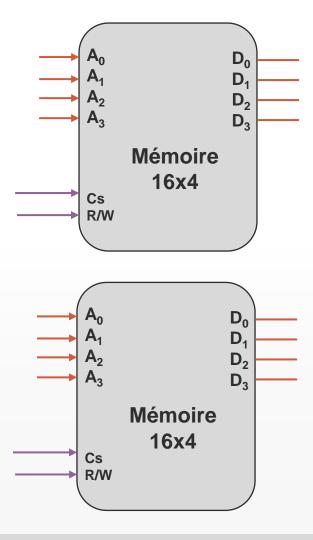
**PROM (Programmable Read-only Memory)**: C'est une ROM qui peut être programmée une seule fois par l'utilisateur (Programmable ROM). La programmation est réalisée à partir d'un programmateur spécifique.

**EPROM (Erasable Programmable Read-only Memory)**: les données de cette puce de mémoire non volatile peuvent être effacées en l'exposant à une lumière UV de haute intensité.

**EEPROM** (flash) (Electrically Erasable Programmable Read-only Memory): les données de cette puce de mémoire non volatile peuvent être effacées électriquement à l'aide de l'émission d'électrons sur le champ.

### Exercice 1 d'application :

Réaliser une mémoire 16x8 (bits) en utilisant deux mémoires de 16x4



**Chapitre 2: Principaux composants d'un ordinateur** 

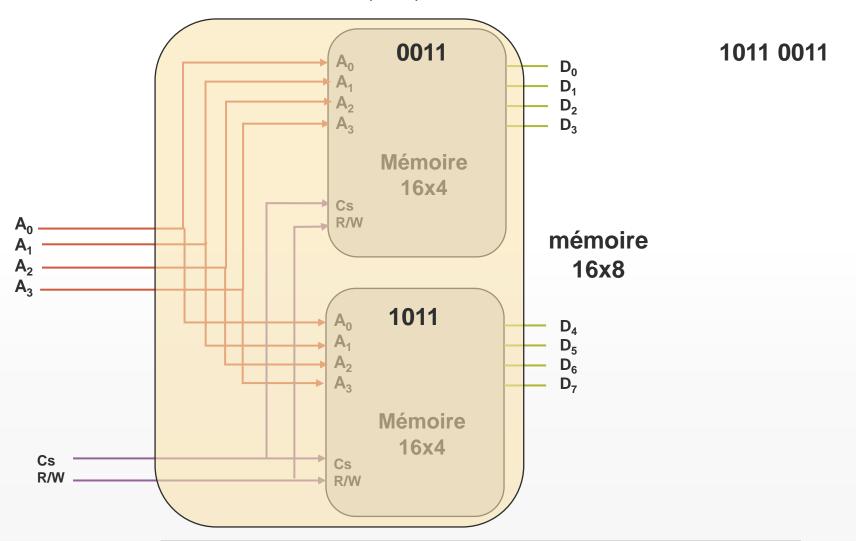
Réaliser une mémoire 16x8 (bits) en utilisant deux mémoires de 16x4

Capacité (en bits) =  $2^{nombres de lignes d'adresse}$  \* nombre de lignes de donnée

$$2^{nombres\ de\ lignes\ d'adresse} = 16 = 2^4$$

$$nombre\ de\ lignes\ d'adresse = 4$$

Réaliser une mémoire 16x8 (bits) en utilisant deux mémoires de 16x4



**Chapitre 2: Principaux composants d'un ordinateur** 

- 1. Réaliser une mémoire 2048x8 (bits) en utilisant les mémoires de 256x8.
- 2. On veut sauvegarder les informations 00111111 11111011 sur la mémoire n°7 a partir de la première ligne. Quelle est l'adresse appropriée en hexadécimal?
- 3. Donner sur une tableau plage d'adresse en hexadécimal pour chaque mémoire.

Nombre de mémoires = 2048/256 = 8 mémoires

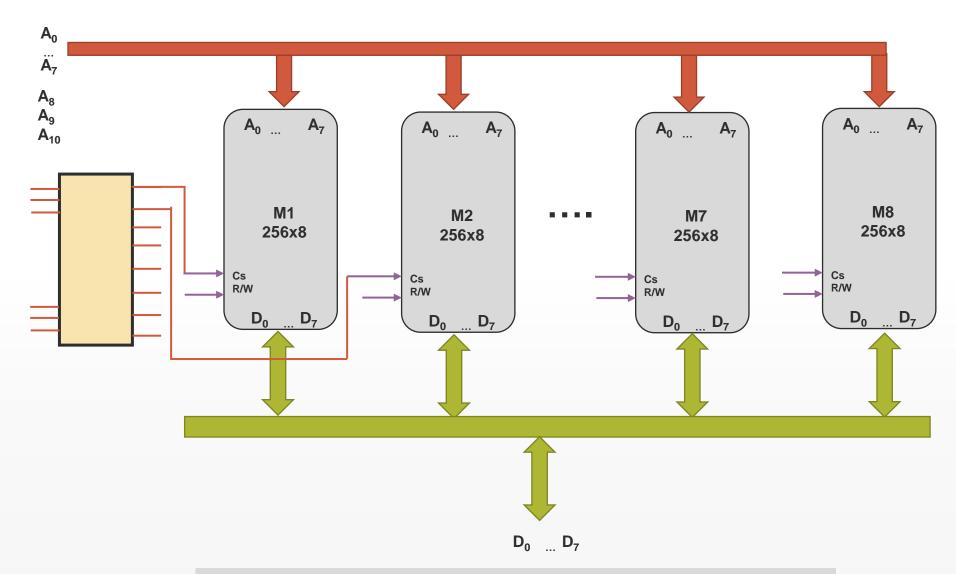
Capacité (en bits) =  $2^{nombres de lignes d'adresse}$  \* nombre de lignes de donnée

### Pour la mémoire 256x8:

 $2^{nombres\ de\ lignes\ d'adresse} = 256 = 2^{8}$ nombre de lignes d'@ = 8

#### Pour la mémoire 2048x8:

 $2^{nombres\ de\ lignes\ d'adresse} = 2048 = 2^{11}$ nombre de lignes d'@ = 11



**Chapitre 2: Principaux composants d'un ordinateur** 

2. On veut sauvegarder les informations 00111111 11111011 sur la mémoire n°7 a partir de la première ligne. Quelle est l'adresse appropriée en hexadécimal?

		<b>A0</b>	<b>A1</b>	<b>A2</b>	<b>A3</b>	<b>A4</b>	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
	600	0	0	0	0	0	0	0	0	0	1	1
RAM 7												
	0FF	1	1	1	1	1	1	1	1	0	1	1

L'adresse appropriée sur la mémoire n°7 pour sauvegarder les info. 00111111 : **600 h** 

L'adresse appropriée sur la mémoire n°7 pour sauvegarder les info. 11111011 : **601 h** 

3. Donner sur une tableau plage d'adresse en hexadécimal pour chaque mémoire.

### La mémoire n°1:

		<b>A</b> 1	<b>A2</b>	<b>A3</b>	<b>A</b> 4	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
000 DAM	0	0	0	0	0	0	0	0	0	0	0
RAM											
OFF	1	1	1	1	1	1	1	1	0	0	0

#### La mémoire n°2:

			<b>A1</b>	A2	<b>A3</b>	<b>A</b> 4	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
	100	0	0	0	0	0	0	0	0	1	0	0
- RAM 1	455											
J	1FF	1	1	1	1	1	1	1	1	1	0	0

3. Donner sur une tableau plage d'adresse en hexadécimal pour chaque mémoire.

La mémoire n°3:

		<b>A0</b>	<b>A</b> 1	<b>A2</b>	<b>A3</b>	<b>A4</b>	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
	200	0	0	0	0	0	0	0	0	0	1	0
- RAM 1	055											
	2FF	1	1	1	1	1	1	1	1	0	1	0

La mémoire n°4:

			<b>A</b> 1	<b>A2</b>	<b>A3</b>	<b>A</b> 4	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
	300	0	0	0	0	0	0	0	0	1	1	0
- RAM 1	055											
ل	3FF -	1	1	1	1	1	1	1	1	1	1	0

3. Donner sur une tableau plage d'adresse en hexadécimal pour chaque mémoire.

La mémoire n°5:

	<b>A0</b>	<b>A</b> 1	<b>A2</b>	<b>A3</b>	<b>A</b> 4	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
100	0	0	0	0	0	0	0	0	0	0	1
- RAM 1											
4FF	1	1	1	1	1	1	1	1	0	0	1

La mémoire n°6:

		<b>A0</b>	<b>A</b> 1	<b>A2</b>	<b>A3</b>	<b>A4</b>	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
RAM	500	0	0	0	0	0	0	0	0	1	0	1
1 (7 (17)												
J	5FF -	1	1	1	1	1	1	1	1	1	0	1

3. Donner sur une tableau plage d'adresse en hexadécimal pour chaque mémoire.

La mémoire n°7:

		<b>A0</b>	<b>A</b> 1	<b>A2</b>	<b>A3</b>	<b>A4</b>	<b>A5</b>	<b>A6</b>	A7	<b>A8</b>	<b>A9</b>	A10
	600	0	0	0	0	0	0	0	0	0	1	1
⊢ RAM 1	٥٥٥											
J	6FF -	1	1	1	1	1	1	1	1	0	1	1

La mémoire n°8:

A10	<b>A9</b>	<b>A8</b>	A7	<b>A6</b>	<b>A5</b>	<b>A</b> 4	<b>A3</b>	A2	<b>A</b> 1			
1	1	1	0	0	0	0	0	0	0	0	700	
											<b></b>	- RAM 1
1	1	1	1	1	1	1	1	1	1	1	7FF	