**CHAPITRE I**

**LES MEMOIRES A SEMI CONDUCTEURS**

**I.1. Introduction**

• Avec une bascule c’est possible de mémoriser une information sur **1 seul bit**.

• Avec un registre c’est possible de mémoriser une information sur **n bits**.

• Si on veut mémoriser une information de **taille important** \_ il faut utiliser **une mémoire.**

**I.2. Définition**

Les mémoires sont des dispositifs qui permettent de conserver puis de restituer les données binaires. L’enregistrement des données dans la mémoire est appelé opération **d’écriture** et leur restitution est l’opération de **lecture**.

Comme le montre la figure ci-dessous, une mémoire peut être considérée comme un ensemble de cases mémoires; chaque case qui peut contenir un bit ou un ensemble de N bits appelé **mot**, est identifiée par un numéro appelé **adresse**. Donc afin de pouvoir lire ou écrire une information en mémoire, il faut connaître l’adresse à laquelle elle a été stockée ou à laquelle elle va être écrite.

Mémoire

**Adresse**

de la donnée

**Donnée**

entrée (écriture) ou sortie (lecture)

**Contrôle**

* sélection de la mémoire
* sens de la donnée (écriture/lecture)
* ...

Figure II.1 : Architecture d’une mémoire

• Exemple de mémoire :

– La mémoire centrale

– Un disque dure

– Une disquette

– Un flash disque

– …………..

• La mémoire peut être dans le processeur (des registres), interne (Mémoire centrale ou principale) ou externe (Mémoire secondaire).

**I.3. Types de mémoires**

**I.3.1. Les mémoires à lecture seule ROM (Read-Only Memories)**

Une mémoire ROM est d'abord un circuit intégré ; C’est une mémoire à lecture seule ou mémoires mortes. L’écriture nécessite un programmateur ou une procédure plus longue que pour les RAMs. Les informations qu'elle contient sont conservées en permanence, même lors d’une coupure d’alimentation.

Dans cette catégorie de mémoires on trouve :

**\* ROM Masquée** : elle est programmée une fois, par le constructeur, lors de la fabrication.

**\*PROM**: une seule programmation est possible, elle est faite en « brûlant » des fusibles. Elle se programme à l’aide d’un équipement particulier appelé programmateur.

\* **UVPROM(EPROM)** : elle est effaçable aux ultraviolets. Il faut 10 à 20 minutes pour l’effacer (effacement de toute la capacité de la mémoire). Elle est ensuite reprogrammable.

\* **EEPROM ou E2PROM** : Elle est effaçable électriquement (effacement adresse par adresse). Elle est ensuite reprogrammable.

\* **EPROM FLASH** : Elle est effaçable électriquement plus rapidement (effacement de toute la capacité de la mémoire). Son temps de programmation est plus rapide, et son coût de fabrication est plus faible que l’EEPROM.

**I.3.2. Les mémoires lecture-écriture (RAM).**

Une mémoire vive RAM "Random Access Memory" est une mémoire dans laquelle on peut écrire et lire les données. Le principal inconvénient de ce type de mémoire est leur volatilité, c’est à dire que les données qu’elles contiennent sont perdues si la tension d’alimentation est coupée. La RAM peut être réalisée de deux façons :

- En utilisant des bascules bistables ; la mémoire obtenue est dite statique "SRAM".

- En utilisant des capacités dont la charge matérialise l’information, c’est le cas de la mémoire dite dynamique "DRAM".

**\*RAM statiques** (SRAM) dans lesquelles les informations sont mémorisées par une bascule de type D et conservées tant que l'alimentation est présente (mémoire volatile), elles sont réalisées en technologie MOS ou bipolaire.

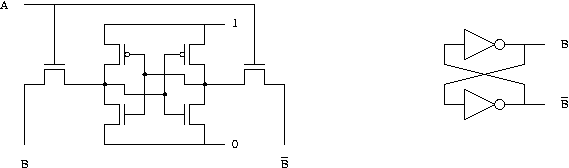


Figure II.2 : Mémoire statique 1 bit

**\*RAM dynamiques** (DRAM) qui utilisent un condensateur comme cellule mémoire (un bit mémorisé) de l'information. Cette information tend à se dégrader à cause des courants de fuites, ce qui nécessite un rafraîchissement périodique.

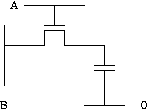


Figure II.3 : Mémoire dynamique 1 bit

****

Figure II.4 : Les mémoires à semi-conducteurs

**I.4. Structure générale d’une mémoire**

On peut donc schématiser un circuit mémoire par la figure suivante où l’on peut distinguer :



Figure II.5 : Structure externe d’une mémoire

*( R/W)* une entrée de sélection de lecture ou d’écriture.

Si *R /W* = 1: c’est une opération de lecture.

Si *R / W* = 0: c’est une opération d’écriture.

La ligne *CS* valide ou non la mémoire.

Si *CS* = 0: la mémoire est validée et fonctionne correctement.

Si *CS* = 1: la mémoire est invalidée, elle ne répond pas

**I.5. Caractéristiques des mémoires**

**Le format des données :**

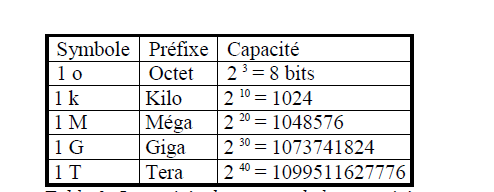
\*c’est la largeur d’un mot mémoire (le nombre de bits par case mémoire)

\*c’est le nombre de fils de données

**Le format d’adresse :** c’est le nombre de fils dans le bus d’adresses

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

• La table suivante résume les préfixes utilisés pour exprimer les capacités des mémoires :



**.**Soit k la taille du bus d’adresses

• Soit n la taille du bus de données (la taille d’un mot mémoire)

• On peut exprimer la capacité de la mémoire centrale soit en nombre de mots mémoire ou en bits (octets, kilo-octets,….)

– La capacité = 2k Mots mémoire

– La capacité = 2k \* n Bits

**Exemple 1**

Dans une mémoire la taille du bus d’adresses K=14 et la taille du bus de données n=4. Calculer la capacité de cette mémoire ?

Réponse

C=214 = 16384 Mots de 4 bits

C= 214 \* 4 = 65536 Bits = 8192 Octets = 8 Ko

**Exemple 2**

Soit une mémoire ayant une capacité de 8 KO.

1- Combien a-t-elle de lignes de données ?

2- Combien a-t-elle de lignes d’adresses ?

3- Quelle est sa capacité en octets ?

Réponse

- Cette mémoire possède 8 lignes de données.

- Cette mémoire stocke 8KO =8x1024=8192 mots. Il y a donc 8192 cases mémoires.

Puisque 8192 = 213, il faut donc 13 lignes d’adresse.

- La capacité de cette mémoire est de 8192 octets.

**Le débit :** c’est le nombre 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.

**La vitesse :** deux paramètres principaux caractérisent la vitesse d’une mémoire :

* Le **temps d'accès**, correspondant à l'intervalle de temps entre la demande de lecture/écriture et la disponibilité de la donnée.
* **Temps de Cycle**: temps minimal entre 2 accès mémoire

.

Une opération de lecture ou d’écriture de la mémoire suit toujours le même cycle :

1. sélection de l’adresse

2. choix de l’opération à effectuer (R/W)

3. sélection de la mémoire (CS = 0)

4. lecture ou écriture de la donnée

Remarque : lorsqu’un composant n’est pas sélectionné, ses sorties sont mises à l’état « haute impédance » afin de ne pas perturber les données circulant sur le bus. ( elle présente une impédance de sortie très élevée = circuit ouvert ).



Figure II.6 : Chronogramme d’une mémoire

**I.6. Localisations**

Lorsque le microprocesseur traite des données, il cherche d'abord dans la mémoire cache ; s'il y trouve les instructions (stockées suite à une précédente lecture des données), cela lui évite d'effectuer une lecture des données plus longue dans une mémoire plus volumineuse ou sur d'autres périphériques de stockage.



Figure II.6 : Localisations des mémoires

**I.7. Assemblage des boîtiers mémoire**

**Problème ?**

• On veut réaliser une mémoire de **capacité C,** mais nous disposons uniquement de boîtiers (des circuits) de **taille inférieure** ?

Solution

On associe plusieurs boîtiers mémoires :

* pour augmenter la taille des mots de la mémoire
* pour augmenter le nombre de mots dans la mémoire

Soit M une mémoire de capacité C, tel que m est le nombre de mot et n la taille d’un mot.

Soit M’  un boîtier de capacité C’, tel que m’ le nombre de mot et n’ la taille d’un mot.

On suppose que C > C’ (m >= m’, n >=n’) Pour connaître le nombre de boîtiers nécessaire, il faut calculer les deux facteurs suivants :

– P =m/m’

– Q=n/n’

P : permet de déterminer de nombre de boîtiers M’ nécessaire pour obtenir le nombre de mots de la mémoire M (extension lignes).

Q : permet de déterminer le nombre de boîtier M’ nécessaire pour obtenir la taille de mot de la mémoire M (extension mots ou extension colonnes).

P.Q donne le nombre total de boîtiers nécessaire pour réaliser la mémoire M.

Pour sélectionner les boîtiers on utilise les bits de poids forts d’adresses. Si P est le facteur d’extension lignes alors on prend k bits tel que P=2k.

Les autres bits d’adresses restants sont utilisés pour sélectionner un mot dans un boîtier.

**Exemple 1**

• Réaliser une mémoire de 1Ko (la taille d’un mot est de 8 bits) en utilisant des boîtiers de taille 256 mots de 8 bits ?

• Solution :

(m, n)=(1024,8) taille du bus d’adresses est de 10 bits **A9 0**(A9…A0), taille du bus de données est de 8 bits **D7 0**(D7….D0)

(m’, n’)=(256,8) taille du bus d’adresses est de 8 bits (A7’…A0’), taille du bus de données est de 8 bits (D7’….D0’)

• Calculer les deux facteurs d’extension lignes et colonnes :

P= m/m’ =1024/256=4 (extension lignes)

Q= n/n’ =8/8=1 (extension colonnes)

• Le nombre total de boîtiers P.Q=4

**Exemple 2**

• On veut réaliser une mémoire de 1Ko (la taille d’un mot est de 16 bits) en utilisant des boîtiers de taille 1Ko mots de 4 bits) ?

• Solution :

(m, n)=(1024,16) taille du bus d’adresses est de 10 bits (A9…A0), taille de bus de données est du 16 bits (D15….D0)

(m’, n’)=(1024,4) taille du bus d’adresses est de 10 bits (A9’…A0’), taille de bus de données est du 4 bits (D3’….D0’)

• P=1024/1024=1 (extension lignes)

• Q=16/4=4 (extension colonnes)

• Le nombre total de boîtiers P.Q=4