

Chapitre 1

Mémoires à semi-conducteur

1.1 Introduction

La mémoire constitue un composant fondamental des systèmes informatiques, assurant le stockage des informations sous diverses formes, qu'elles soient électriques, magnétiques ou optiques. Dans un système reposant sur un microprocesseur μP , lequel s'appuie sur une logique numérique, les données sont codées sous forme binaire (0 et 1). Les mémoires remplissent ainsi une fonction critique, non seulement dans l'exécution des programmes, mais également dans la préservation des informations sur le long terme.

Parmi les différentes technologies disponibles, les mémoires à semi-conducteurs occupent une place prépondérante dans les architectures à base de microprocesseurs. Leur adoption massive s'explique par plusieurs avantages déterminants : fiabilité élevée, rapidité d'accès, coût compétitif et modularité en termes d'extension de capacité. Par ailleurs, leur conception favorise une miniaturisation accrue tout en optimisant la densité de stockage, répondant ainsi aux exigences croissantes des applications contemporaines, qu'il s'agisse d'appareils mobiles ou de systèmes embarqués. Ces propriétés ont conduit à leur suprématie face aux technologies obsolètes, telles que les bandes magnétiques et les disques mécaniques, désormais supplantés dans la majorité des cas d'usage.

1.2 Rappels sur les registres à décalage

Un registre à décalage est un circuit séquentiel capable de stocker et de décaler des données binaires. Il est constitué de bascules D connectées en série.

1.2.1 *Types et Fonctionnement*

1. **Entrée série / Sortie série (SISO) (SISO)** : Les bits sont chargés un par un et ressortent dans le même ordre.
2. **Entrée série / Sortie parallèle (SIPO) (SIPO)** : Les bits sont chargés séquentiellement puis lus en parallèle.
3. **Entrée parallèle / Sortie série (PISO) (PISO)** : Les bits sont chargés en parallèle puis décalés en série.
4. **Entrée parallèle / Sortie parallèle (PIPO) (PIPO)** : Fonctionne comme un registre classique.

1.2.2 *Applications des Registres à décalage*

- Conversion série/parallèle et parallèle/série.
- Temporisation et génération de séquences.

1.3 Unité de capacité mémoire

La capacité mémoire est exprimée en :

- **Bits (b)** : Plus petite unité (0 ou 1).
- **Octets (O)** : 1 octet = 8 bits.
- **Kilooctets (KO)** : 1 Ko = 1024 octets.
- **Mégaoctets (MO)** : 1 Mo = 1024 Ko.
- **Gigaoctets (GO)** : 1 Go = 1024 Mo.
- **Téraoctets (TO)** : 1 To = 1024 Go.

1.4 Accès mémoire

1.4.1 *Accès séquentiel*

1. Les données sont lues ou écrites dans un ordre prédéfini (ex : bande magnétique).
2. Lent car nécessite un parcours séquentiel.

1.4.2 *Accès aléatoire*

1. Les données sont accessibles directement via une adresse.

2. Temps d'accès constant quelle que soit la localisation.

1.5 Différentes technologies de mémoires

Différentes technologies ont été utilisées pour concevoir des mémoires. la table 1.1 résume les différentes technologies de mémoires.

Tableau 1.1: Les différentes technologies de mémoires

| Technologie | Description | Exemples |
|------------------|--|----------------------------------|
| Magnétique | Stockage basé sur l'orientation de champs magnétiques. | Disques durs, bandes magnétiques |
| Semi-conducteurs | Utilisation de transistors et de circuits intégrés. | RAM, ROM, Flash |
| Électro-optiques | Stockage par modification optique (laser). | CD (CD) , DVD (DVD), Blu-ray |

1.6 Mémoires à semi-conducteur

1.6.1 Cellule mémoire (*Bascule*)

La bascule occupe une position centrale en tant qu'élément cellulaire fondamental dans les architectures de mémoires à semi-conducteurs, où elle assure la fonction primordiale de stockage de l'information binaire. Plus précisément, une bascule, à l'instar de la bascule D, présente la capacité de mémoriser un bit de données, matérialisé sous la forme d'un état logique 0 ou 1. Cette propriété de rétention d'information, caractérisée par le maintien d'un état stable indépendamment des variations ultérieures des signaux d'entrée, en fait un composant essentiel dans la conception des mémoires numériques. En effet, chaque cellule mémoire repose typiquement sur une structure à base de bascules, garantissant ainsi la conservation fiable des données binaires au niveau élémentaire.

La Figure 1.1 illustre le schéma fonctionnel d'une bascule D (ou flip-flop D), un circuit logique séquentiel largement utilisé en électronique numérique. Une analyse détaillée de ses composants et de son principe de fonctionnement peut être formulée comme suit :

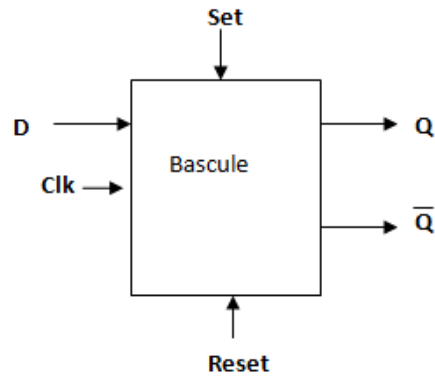


Figure 1.1: Diagramme bloc d'une Bascule D

1. **Entrée D (Data) :**

- L'entrée D est l'entrée de **données**. Dans une bascule D, la valeur présente sur cette entrée est capturée et transférée à la sortie **Q** lorsque le signal d'horloge (Clk) change d'état (montée d'horloge, typiquement). ;
- En fonction de l'état de cette entrée, la bascule stockera soit un "0" soit un "1" lorsque l'horloge sera activée.

2. **Entrée Clk (Clock) :**

- L'entrée d'horloge **Clk** est utilisée pour synchroniser le changement d'état de la bascule. La bascule D ne change d'état qu'à un certain moment du cycle de l'horloge, généralement lors d'une transition montante (front montant) ou parfois lors d'une transition descendante (selon la conception).
- Lorsque l'horloge passe de 0 à 1 (front montant), la valeur présente à l'entrée D est capturée et transférée vers la sortie Q.

3. **Entrée Set (S) :**

- L'entrée Set est une entrée asynchrone qui force la sortie Q à 1, indépendamment de l'état des autres entrées. Cela signifie que lorsque le signal Set est activé, la bascule se fixe à l'état haut (1) immédiatement, sans attendre un cycle d'horloge.
- Cette fonction est souvent utilisée pour initialiser la bascule dans un état défini au démarrage.

4. **Entrée Reset (R) :**

- L'entrée Reset est une autre entrée asynchrone. Lorsqu'elle est activée, elle force la sortie Q à 0, quel que soit l'état des autres entrées.
- Comme le Set, cette entrée est utilisée pour forcer la bascule dans un état spécifique (dans ce cas, à 0), indépendamment du signal d'horloge.

5. **Sorties Q et \overline{Q}** : - Q est la sortie principale de la bascule, qui reflète l'état capturé à l'entrée D lors du front d'horloge. Si la valeur à l'entrée D était 1 au moment de la transition de l'horloge, Q sera également 1. Si D était 0, Q sera 0. (μP
- \overline{Q} est la sortie complémentaire, qui est toujours l'inverse de la sortie Q. Si Q est 1, \overline{Q} sera 0, et inversement.

1.6.1.1 Fonctionnement de la bascule D

1. **Mode synchrone** : La bascule D est un élément de mémoire contrôlé par l'horloge. Elle "prend en compte" la valeur de l'entrée D seulement au moment de la transition de l'horloge (généralement le front montant). Tant que l'horloge ne change pas, la sortie Q reste inchangée.
2. **Mode asynchrone** : Les entrées Set et Reset permettent de forcer la bascule dans un état donné (1 ou 0) sans attendre un cycle d'horloge. Ces entrées sont souvent utilisées pour initialiser ou réinitialiser les bascules au démarrage d'un système.

1.6.1.2 Applications de la bascule D

1. **Stockage de données** : Une bascule D peut être utilisée pour stocker un bit de donnée. Un ensemble de bascules D peut former un registre pour stocker plusieurs bits (un octet, un mot, etc.).
2. **Compteurs** : Les bascules D peuvent être utilisées dans des circuits de comptage pour créer des registres à décalage, des compteurs binaires, etc.
3. **Synchronisation** : Elles sont souvent utilisées pour synchroniser des signaux dans des systèmes synchrones où les opérations doivent être cadencées par une horloge.

1.6.2 Schéma fonctionnel d'une mémoire à semi conducteur

La figure 1.2 représente une simplification d'une mémoire à semi-conducteurs, constituée d'un ensemble de **cases mémoire**, où chacune de ces cases contient plusieurs **cellules mémoire**. Ces cellules mémoire sont généralement réalisées à l'aide de bascules (flip-flops), qui sont des circuits logiques séquentiels capables de stocker un bit d'information. Chaque case mémoire regroupe un ensemble de cellules, ce qui permet de stocker plusieurs bits, formant ainsi un octet ou un mot selon la taille de la case.

Dans une mémoire à semi-conducteurs, chaque case mémoire est identifiée par **une adresse unique**. Cette adresse permet au processeur ou à un autre contrôleur de lire

ou d'écrire des données dans cette case spécifique. Le processus d'adressage est crucial pour accéder aux données dans une mémoire de grande capacité. Les cases sont disposées dans une structure matricielle où chaque ligne et colonne peut être sélectionnée pour une lecture ou une écriture via des circuits de gestion d'adressage.

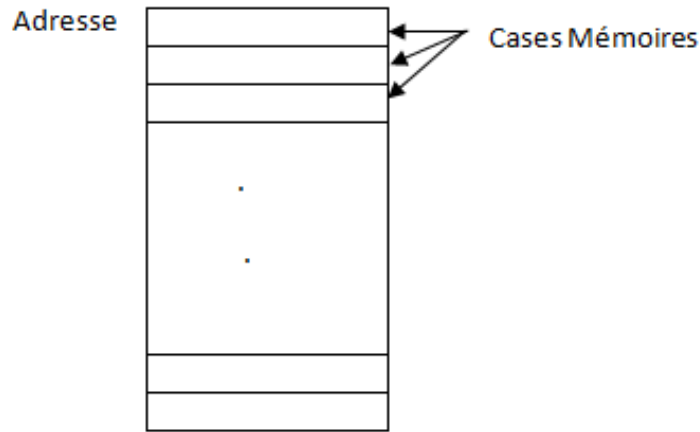


Figure 1.2: Diagramme bloc simplifié d'une mémoire à semi conducteur

1.6.3 Architecture externe d'une mémoire à semi conducteur

Le schéma de la Figure 1.3 représente **une vue simplifiée de l'architecture externe** d'une **mémoire à semi-conducteurs (MSC)**, montrant les connexions principales pour l'accès aux données. Ce type de mémoire est commun dans les systèmes numériques et est utilisé pour stocker temporairement ou de façon permanente des données. Voici une explication détaillée de chaque élément du schéma :

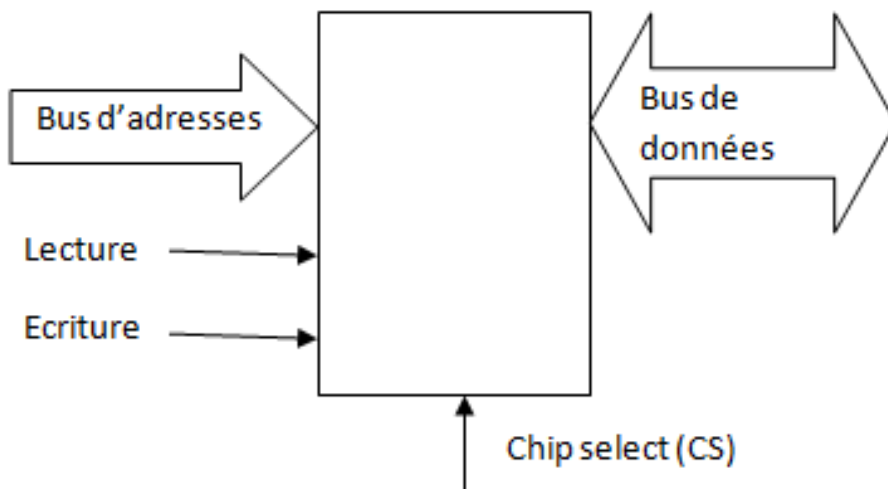


Figure 1.3: Architecture Externe d'une mémoire à semi conducteur

1. **Bus d'adresses :**

- Le **bus d'adresses** est un ensemble de lignes unidirectionnelles entrantes permettant au processeur ou à un contrôleur d'envoyer une adresse à la mémoire. Cette adresse permet de sélectionner une case mémoire spécifique à lire ou à écrire.
- Chaque case mémoire dans le circuit est identifiée par une adresse unique, et le bus d'adresses véhicule cette information pour accéder aux données souhaitées.

2. **Bus de données :**

- Le **bus de données** est également un ensemble de lignes bidirectionnelles , mais il sert à transférer les données entre la mémoire et le processeur (ou un autre dispositif).
- Lorsque le processeur effectue une opération de lecture, les données sont récupérées de la mémoire et envoyées via le bus de données au processeur. Dans le cas d'une écriture, les données transitent du processeur vers la mémoire via ce même bus.

3. **Lecture (Read) :**

- La ligne de **lecture** est un signal de contrôle envoyé par le processeur à la mémoire pour indiquer qu'une opération de lecture est en cours. Lorsque cette ligne est active, la mémoire place les données demandées sur le bus de données pour que le processeur ou l'unité de commande puisse les lire.
- Ce signal est essentiel pour distinguer une lecture d'une écriture.

4. **Écriture (Write) :**

- De manière similaire à la lecture, la ligne d'écriture permet de signaler à la mémoire qu'une opération **d'écriture** est en cours.
- Lorsque ce signal est actif, les données présentes sur le bus de données sont stockées dans la case mémoire correspondant à l'adresse spécifiée.

5. **Chip Select (CS) :**

- Le signal **Chip Select (CS)** est utilisé pour activer ou désactiver la mémoire. Ce signal permet de sélectionner spécifiquement un composant mémoire lorsque plusieurs mémoires sont présentes dans un système.
- Lorsque le signal **CS** est activé, la mémoire est "sélectionnée" et peut répondre aux commandes de lecture ou d'écriture. Si ce signal est désactivé, la mémoire est mise en mode inactif et ignore les commandes.

Fonctionnement global :

Lorsqu'un dispositif (comme un processeur) souhaite accéder à la mémoire, il envoie une adresse spécifique via le **bus d'adresses** et un signal indiquant s'il s'agit d'une opération de **lecture** ou **d'écriture**. Si une **lecture** est demandée et que le signal **Chip Select** est activé, la mémoire place les données stockées à l'adresse correspondante sur le **bus de données**, qui est ensuite récupéré par le processeur. Si une **écriture** est effectuée, les données envoyées via le **bus de données** sont stockées à l'adresse mémoire spécifiée.

1.6.4 *Capacité mémoires*

La capacité mémoire d'un système à mémoire à semi conducteur peut être calculée en fonction de la taille du bus d'adresse (BA) et du bus de données (BD). Le bus d'adresse détermine le nombre total d'adresses uniques que le processeur peut utiliser pour accéder à la mémoire. En effet, un bus d'adresse de taille n bits permet de générer 2^n adresses distinctes, ce qui correspond au nombre de cellules mémoire que le processeur peut atteindre. Le bus de données, quant à lui, définit la quantité de données que le processeur peut traiter en une seule fois, mesurée en bits. La capacité totale de mémoire adressable se calcule ainsi :

$$\text{Capacité mémoire (bits)} = 2^{BA} * BD$$

$$\text{Capacité mémoire (Octets)} = 2^{BA} * \frac{BD}{8}$$

Exemple : Une mémoire disposant d'un BA de 10 lignes et d'un BD=8 lignes ;

$$CM(\text{bits}) = 2^{10} * 8 = 2^{10} * 2^3 = 2^{13} \text{ bits}$$

$$CM(\text{octets}) = 2^{10} * \frac{8}{8} = 2^{10} \text{ octets}$$

Règles à retenir : $1k = 2^{10}$

$$1M = 2^{20}$$

$$1G = 2^{30}$$

$$1T = 2^{40}$$

1.7 Catégories des mémoires à semi-conducteur

Les systèmes à microprocesseur utilisent différents types de mémoire à semi-conducteur, chacun jouant un rôle spécifique dans le fonctionnement global du système. Les principaux types incluent la **mémoire ROM (Read-Only Memory)** (ROM) , la **mémoire RAM (Random Access Memory)** (RAM), et des mémoires non volatiles comme la **Flash** et la **EEPROM (Electrically Erasable Programmable Read-Only Memory)**.

1. **La ROM** est une mémoire non volatile qui conserve les données même lorsque l'alimentation est coupée. Elle stocke les instructions de démarrage ou le programme de base, appelé BIOS ou firmware, qui est essentiel pour initialiser le système et lancer le microprocesseur. Comme son contenu est fixe, la ROM assure la stabilité des instructions de démarrage.
2. **La RAM** est une mémoire volatile, utilisée pour le stockage temporaire des données et instructions pendant l'exécution d'un programme. Elle offre un accès rapide aux données, ce qui est crucial pour la performance du microprocesseur dans les calculs en temps réel et le traitement des instructions. La RAM est divisée en deux catégories principales : la SRAM (Static RAM), qui est plus rapide et conserve les données tant qu'elle est alimentée, et la DRAM (Dynamic RAM) (DRAM), qui nécessite un rafraîchissement périodique mais offre une plus grande capacité.
3. **La mémoire Flash et EEPROM (EEROM)** sont des types de mémoires non volatiles, utilisées pour stocker des données qui doivent être modifiables mais conservées après la mise hors tension. La Flash est couramment utilisée pour stocker les programmes dans les microcontrôleurs, car elle permet de réécrire le programme sans matériel supplémentaire. L'EEPROM, bien qu'offrant une capacité plus réduite,

permet de modifier des données individuelles, ce qui est utile pour enregistrer des paramètres configurables par l'utilisateur.

1.8 Critères de choix d'une mémoire

Tableau 1.2: Nombre de paramètres, taille de filtre et nombres de couches convolutives de ces modèles CNNs

| Critère | Description |
|--------------|---|
| Capacité | Nombre de bits stockables. |
| Vitesse | Temps d'accès et fréquence. |
| Volatilité | Conservation des données sans alimentation. |
| Coût | Prix par bit. |
| Consommation | Important pour les systèmes embarqués. |

1.9 Extension de la capacité mémoire

Dans le domaine des mémoires semi-conductrices, l'extension de la capacité mémoire peut être réalisée selon deux configurations distinctes : l'assemblage en série et l'assemblage en parallèle. L'assemblage en série permet d'augmenter le nombre de cases mémoire disponibles, ce qui se traduit par une expansion de l'espace d'adressage global du système. Cette approche est particulièrement utile pour les applications nécessitant une grande quantité de données stockées de manière séquentielle. En revanche, l'assemblage en parallèle vise à accroître la taille du mot mémoire, améliorant ainsi le débit des données en permettant le transfert simultané de plusieurs bits. Cette configuration est privilégiée dans les contextes où la bande passante et les performances en lecture/écriture sont critiques. Ces deux méthodes d'extension offrent des compromis distincts entre capacité, vitesse et complexité de conception, répondant ainsi aux exigences variées des architectures informatiques modernes.