

Chapitre 2 Les mémoires

Table des matières

Les mémoires

- 2.1 Mémoires RAM et ROM
- 2.2 Schéma fonctionnel d'une mémoire
- 2.3 Interfaçage microprocesseur/mémoire
- 2.4 Chronogramme de lecture/ écriture
- 2.5 Construction de plusieurs boîtiers mémoire
- 2.6 Décodage d'adresse
- 2.7 Classification des mémoires

Les mémoires

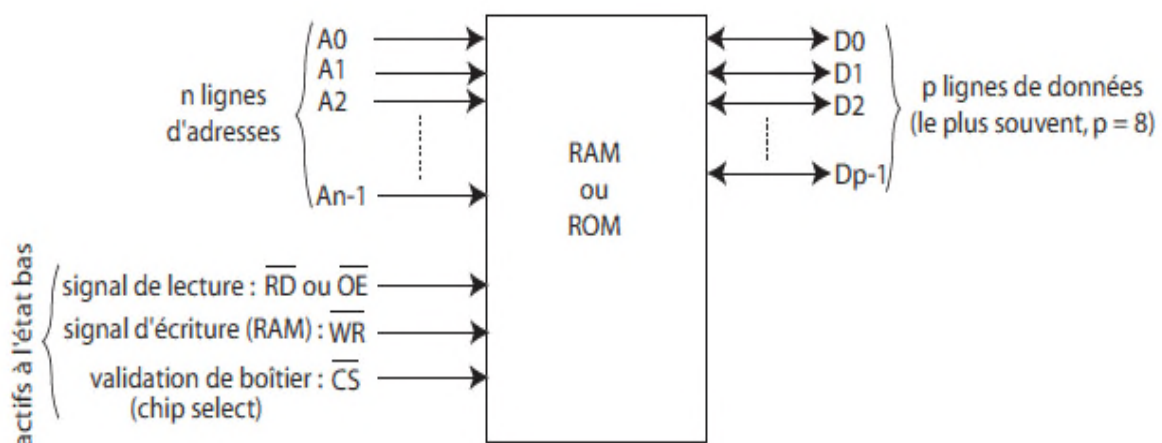
2.1 Mémoires RAM et ROM

On distingue deux types de mémoires :

- les **mémoires vives** (RAM : Random Access Memory) ou mémoires volatiles. Elles perdent leur contenu en cas de coupure d'alimentation. Elles sont utilisées pour stocker temporairement des données et des programmes. Elles peuvent être lues et écrites par le microprocesseur ;
- les **mémoires mortes** (ROM : Read Only Memory) ou mémoires non volatiles. Elles conservent leur contenu en cas de coupure d'alimentation. Elles ne peuvent être que lues par le microprocesseur (pas de possibilité d'écriture). On les utilise pour stocker des données et des programmes de manière définitive.

Les mémoires sont caractérisées par leur **capacité** : nombre total de cases mémoire contenues dans un même boîtier.

2.2 Schéma fonctionnel d'une mémoire

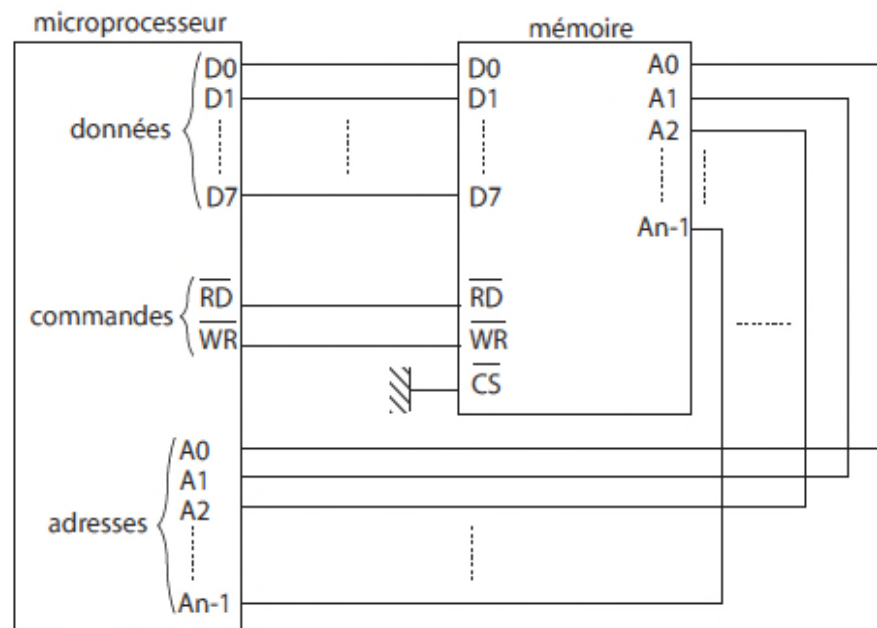


Le nombre de lignes d'adresses dépend de la capacité de la mémoire : n lignes d'adresses permettent d'adresser 2^n cases mémoire : 8 bits d'adresses permettent d'adresser 256 oc-

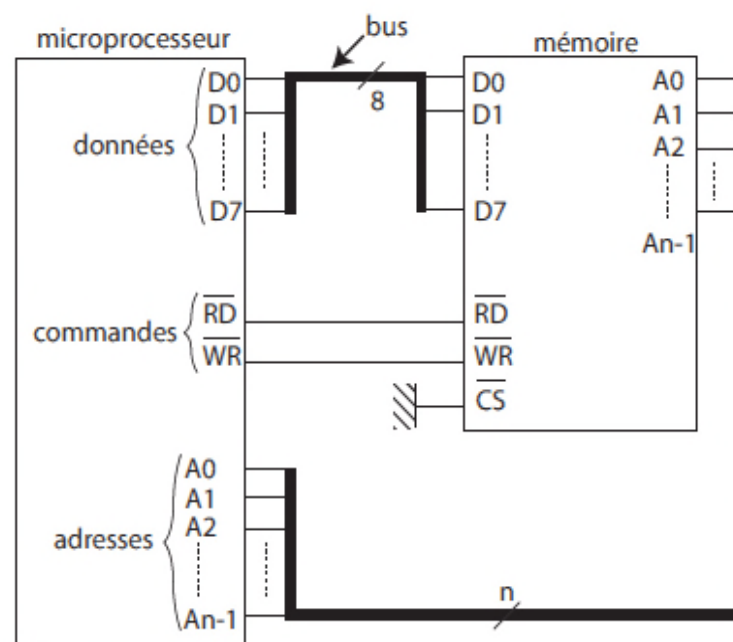
tets, 16 bits d'adresses permettent d'adresser 65536 octets (= 64 Ko), ...

Exemple : mémoire RAM 6264, capacité = $8K \times 8$ bits : 13 broches d'adresses A0 à A12, $2^{13} = 8192 = 8$ Ko.

2.3 Interfaçage microprocesseur / mémoire



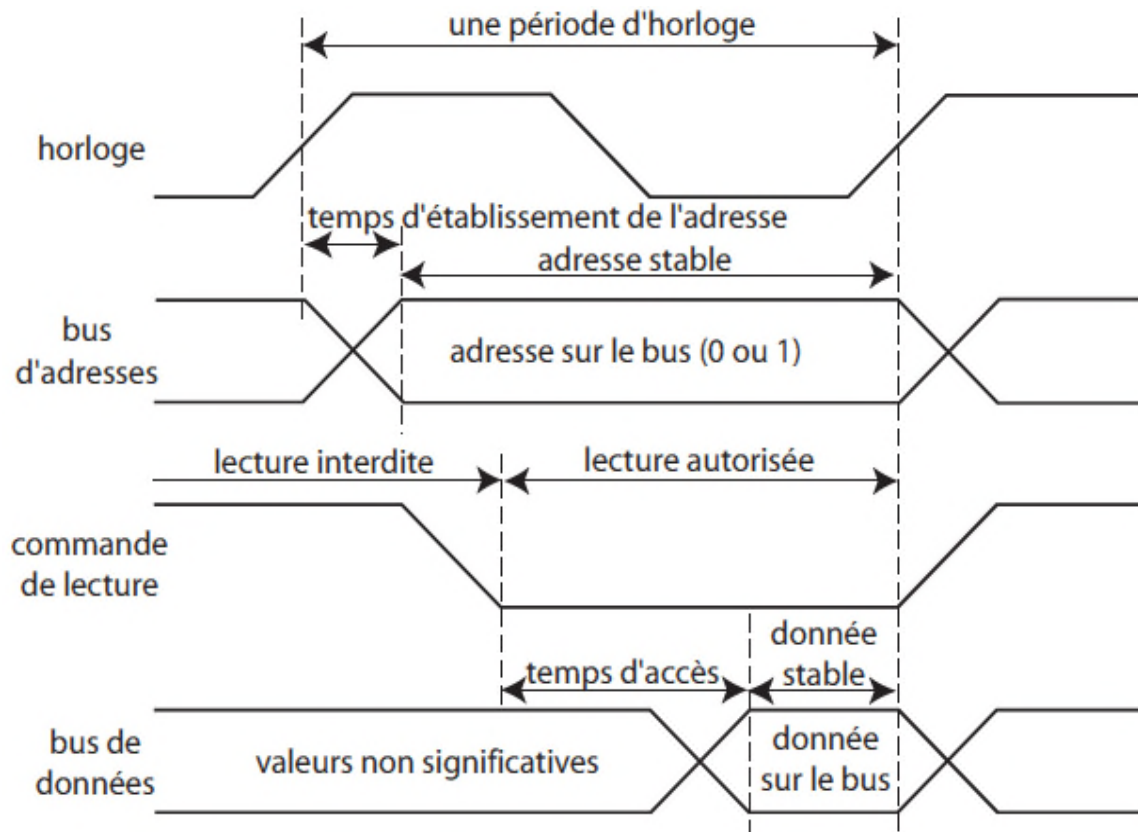
Représentation condensée (plus pratique) :



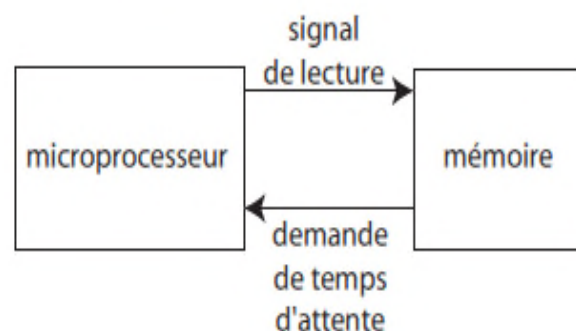
2.4 Chronogramme de lecture / écriture en mémoire

Une caractéristique importante des mémoires est leur **temps d'accès** : c'est le temps qui s'écoule entre l'instant où l'adresse de la case mémoire est présentée sur le bus d'adresses et celui où la mémoire place la donnée demandée sur le bus de données. Ce temps varie entre 50 ns (mémoires rapides) et 300 ns (mémoires lentes).

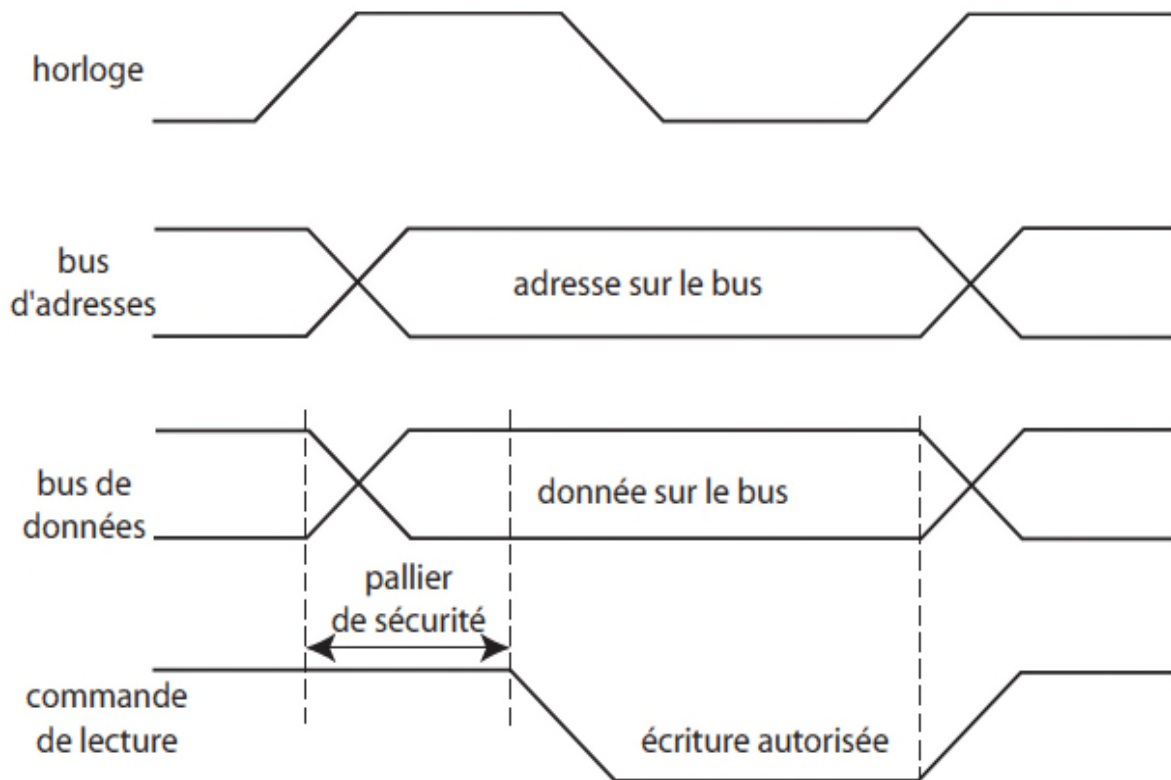
Chronogramme de lecture en mémoire :



Remarque : si le temps d'accès d'une mémoire est supérieur à une période d'horloge (mémoire lente), le microprocesseur peut accorder à la mémoire un temps supplémentaire (une ou plusieurs périodes d'horloge), à la demande de celle-ci. Ce temps supplémentaire est appelé **temps d'attente** (wait time : T_W) :



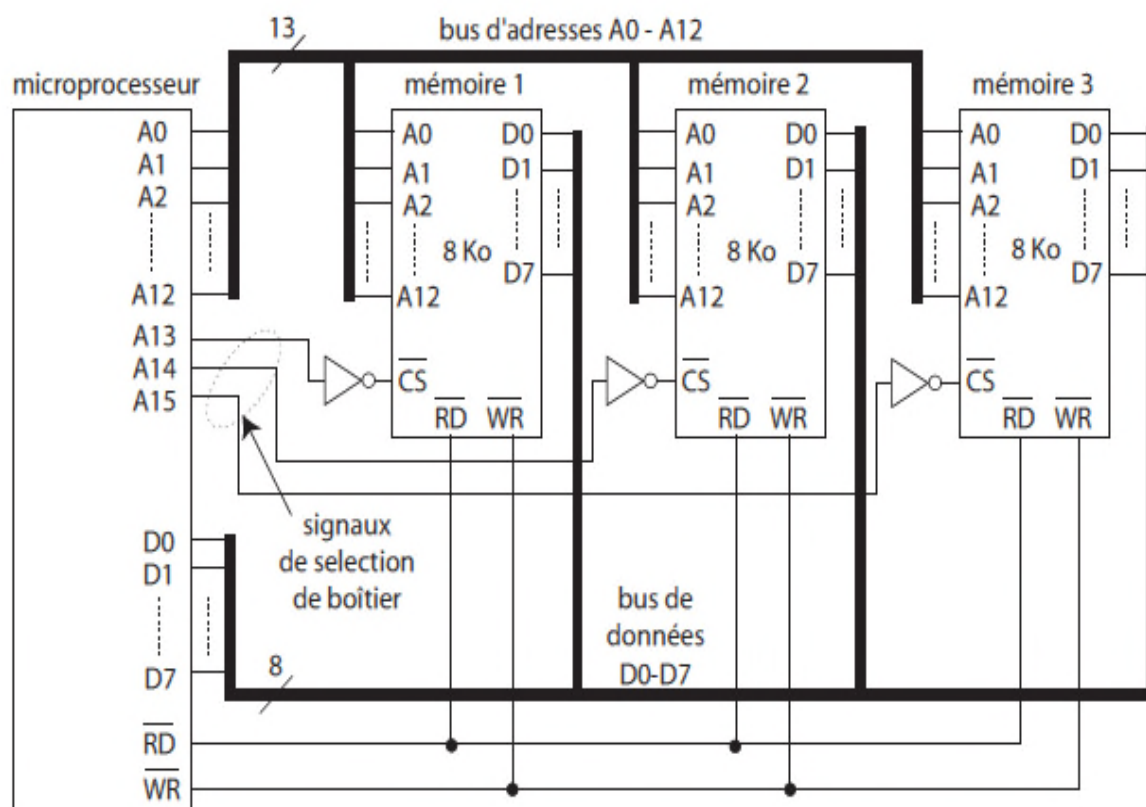
Chronogramme d'écriture en mémoire :



2.5 Connexion de plusieurs boîtiers mémoires sur le bus d'un microprocesseur

Les boîtiers mémoire possèdent une broche notée $\overline{CS}$: Chip Select. Lorsque cette broche est active (état bas), le circuit peut être lu ou écrit. Lorsqu'elle est inactive (état haut), le circuit est exclu du service : ses broches de données D0 à D7 passent à l'état de haute impédance : tout se passe comme si la mémoire était déconnectée du bus de données du microprocesseur, d'où la possibilité de connecter plusieurs boîtiers mémoire sur un même bus : un seul signal $\overline{CS}$ doit être actif à un instant donné pour éviter les conflits entre les différents boîtiers.

Exemple : connexion de trois boîtiers mémoire d'une capacité de 8 Ko chacun (13 lignes d'adresses) sur un bus d'adresse de 16 bits :



Dans un même boîtier, une case mémoire est désignée par les bits d'adresses A0 à A12 :

A12	A11	...	A1	A0		A12	A11	...	A1	A0
0	0	...	0	0	à	1	1	...	1	1
0000H						1FFFH				

Pour atteindre la mémoire n°1, il faut mettre à 1 le bit A13 et à 0 les bits A14 et A15. La plage d'adresses occupée par cette mémoire est donc :

A15	A14	A13	A12	...	A0		A15	A14	A13	A12	...	A0
0	0	1	0	...	0	à	0	0	1	1	...	1
2000H							3FFFH					

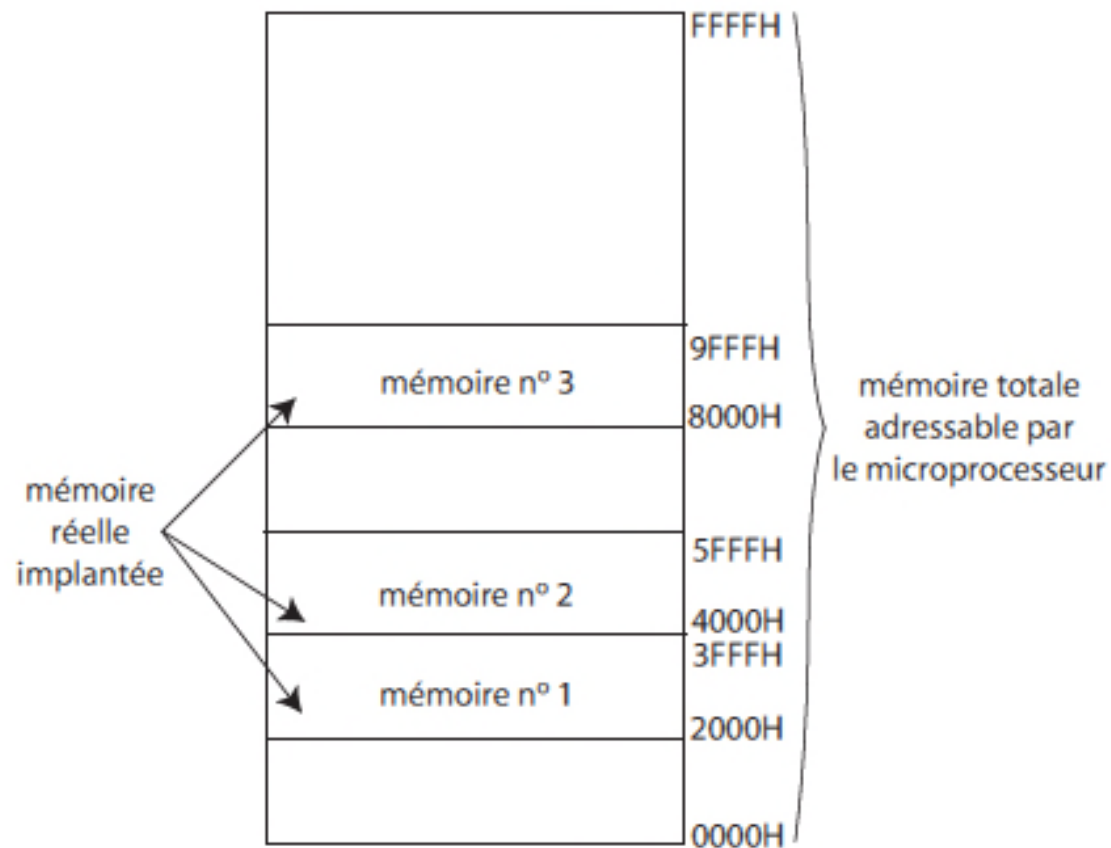
De même, pour la mémoire n°2, on doit avoir A13 = 0, A14 = 1 et A15 = 0 d'où la plage d'adresses occupée cette mémoire :

A15	A14	A13	A12	...	A0		A15	A14	A13	A12	...	A0
0	1	0	0	...	0	à	0	1	0	1	...	1
4000H							5FFFH					

Pour la mémoire n°3, on doit avoir A13 = 0, A14 = 0 et A15 = 1 d'où la plage d'adresses occupée cette mémoire :

A15	A14	A13	A12	...	A0		A15	A14	A13	A12	...	A0
1	0	0	0	...	0	à	1	0	0	1	...	1
8000H							9FFFH					

On en déduit la **cartographie** ou **mapping** de la mémoire visible par le microprocesseur :



2.6 Décodage d'adresses

Les trois bits A13, A14 et A15 utilisés précédemment fournissent en fait 8 combinaisons, de 000 à 111, d'où la possibilité de connecter jusqu'à 8 boîtiers mémoire de 8 Ko sur le bus. La mémoire totale implantée devient donc de $8 \times 8 \text{ Ko} = 64 \text{ Ko}$: valeur maximale possible avec 16 bits d'adresses.

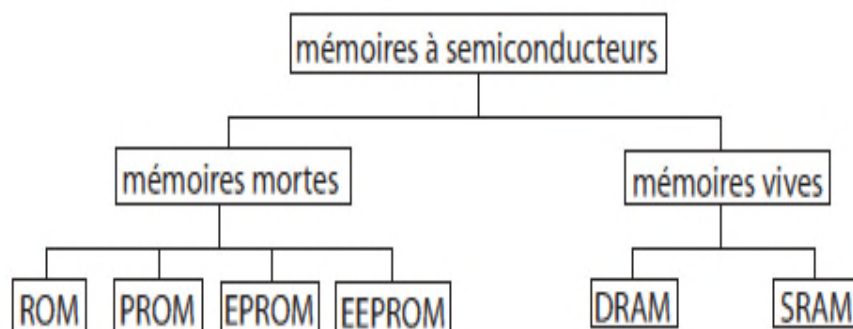
Pour cela, il faut utiliser un **circuit de décodage d'adresses**, dans ce cas : un **décodeur 3 vers 8**.

Le mapping de la mémoire devient ainsi :

mémoire n° 7	FFFFH
	E000H
mémoire n° 6	DFFFH
	C000H
mémoire n° 5	BFFFH
	A000H
mémoire n° 4	9FFFH
	8000H
mémoire n° 3	7FFFH
	6000H
mémoire n° 2	5FFFH
	4000H
mémoire n° 1	3FFFH
	2000H
mémoire n° 0	1FFFH
	0000H

2.7 Classification des mémoires

Jusqu'à la fin des années 1970, on utilisait des mémoires à tores magnétiques, lentes et de faibles capacités. Actuellement, on n'utilise plus que des mémoires à semiconducteurs.



Mémoires mortes :

- **ROM** : Read Only Memory. Mémoire à lecture seule, sans écriture. Son contenu est programmé une fois pour toutes par le constructeur. Avantage : faible coût. Inconvénient : nécessite une production en très grande quantité.
- **PROM** : Programmable Read Only Memory. ROM programmable une seule fois par l'utilisateur (ROM OTP : One Time Programming) en faisant sauter des fusibles. Nécessite un programmeur spécialisé : application d'une tension de programmation (21 ou 25 V) pendant 20 ms.
- **EPROM** : Erasable PROM, appelée aussi UVPROM. ROM programmable électriquement avec un programmeur et effaçable par exposition à un rayonnement ultraviolet pendant 30 minutes. Famille 27nnn, exemple : 2764 (8 Ko), 27256 (32 Ko). Avantage : reprogrammable par l'utilisateur.
- **EEPROM** : Electrically Erasable PROM. ROM programmable et effaçable électriquement. Lecture à vitesse normale (≤ 100 ns). Ecriture (= effacement) très lente (≈ 10 ms). Application : les EEPROM contiennent des données qui peuvent être modifiées de temps en temps, exemple : paramètres de configuration des ordinateurs. Avantage : programmation sans extraction de la carte et sans programmeur. Inconvénient : coût élevé.

Mémoires vives :

- **SRAM** : Static Random Access Memory. Mémoire statique à accès aléatoire, à base de bascules à semiconducteurs à deux états (bascules RS). Famille 62nnn, exemple : 62128 (16 Ko). Avantage : très rapide, simple d'utilisation. Inconvénient : compliqué à réaliser.
- **DRAM** : Dynamic RAM. Basée sur la charge de condensateurs : condensateur chargé = 1, condensateur déchargé = 0. Avantage : intégration élevée, faible coût. Inconvénient : nécessite un rafraîchissement périodique à cause du courant de fuite des condensateurs. Application : réalisation de la mémoire vive des ordinateurs (barettes mémoire SIMM : Single In-line Memory module).