

ETRS302_ETRS

«Fonctionnement des ordinateurs»

Les mémoires

Définition

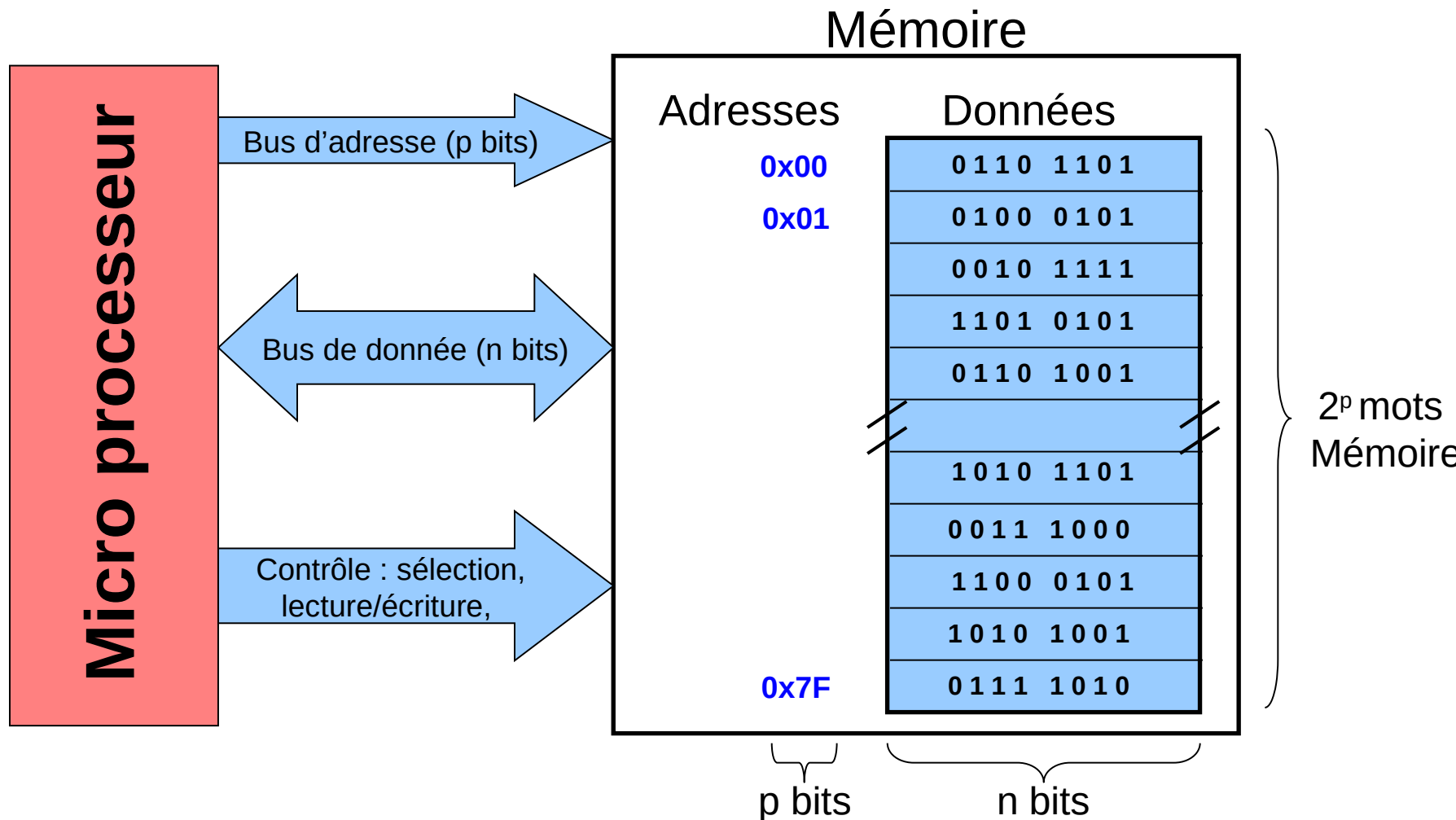
- Une mémoire est un dispositif de stockage dans lequel on peut :
 - Enregistrer de l'information (Ecriture)
 - Restituer une information (Lecture)
- Différentes formes de mémoire
 - A semi-conducteur (RAM, ROM, flash...)
 - Optique (DVD, CD)
 - Magnétique (Disque dur)
- Dans ce cours on utilise uniquement des mémoires à semi-conducteur

Modèle d'une mémoire

	Adresses	Contenu (données)
0x00	0 0 0 0 0 0 0 0	0 1 1 0 1 1 0 1
0x01	0 0 0 0 0 0 0 1	0 1 0 0 0 1 0 1
	0 0 0 0 0 0 1 0	0 0 1 0 1 1 1 1
	0 0 0 0 0 0 1 1	1 1 0 1 0 1 0 1
	0 0 0 0 0 1 0 0	0 1 1 0 1 0 0 1
	// //
	0 1 1 1 1 0 1 1	1 0 1 0 1 1 0 1
	0 1 1 1 1 1 0 0	0 0 1 1 1 0 0 0
	0 1 1 1 1 1 0 1	1 1 0 0 0 1 0 1
	0 1 1 1 1 1 1 0	1 0 1 0 1 0 0 1
0x7F	0 1 1 1 1 1 1 1	0 1 1 1 1 0 1 0

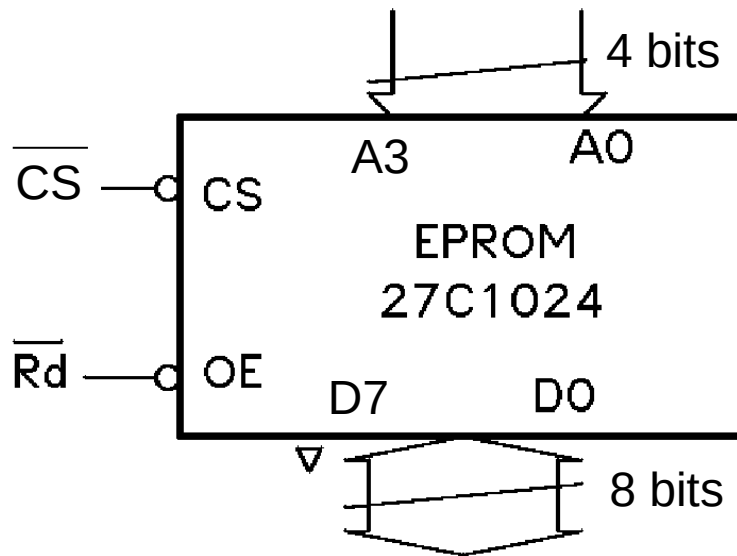
Taille : $2^{\text{nbr bits d'adresse}}$ X nbr bits de données

Interconnexion entre le processeur et la mémoire



Fonctionnement

Symbol d'un composant mémoire

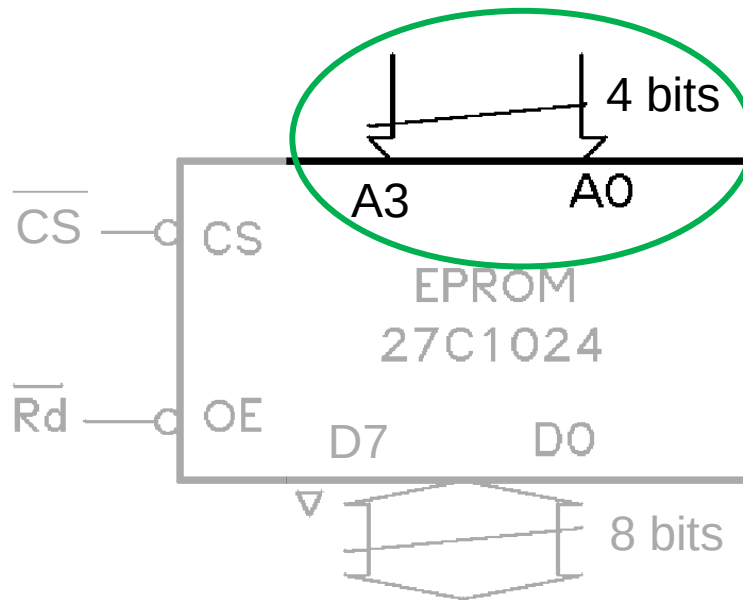


Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 0 1 0 1 1 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 0 0 0 0 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Fonctionnement

Symbol d'un composant mémoire



Ici, le bus d'adresse a 4 bits : A0, A1, A2 et A3
Sur ces 4 bits, on peut coder $2^4=16$ adresses
La première adresse est 0, la dernière est 15
Comme on écrit les adresses en hexadécimal, la dernière adresse s'écrit F ou 0xF (0x=hexadécimal)

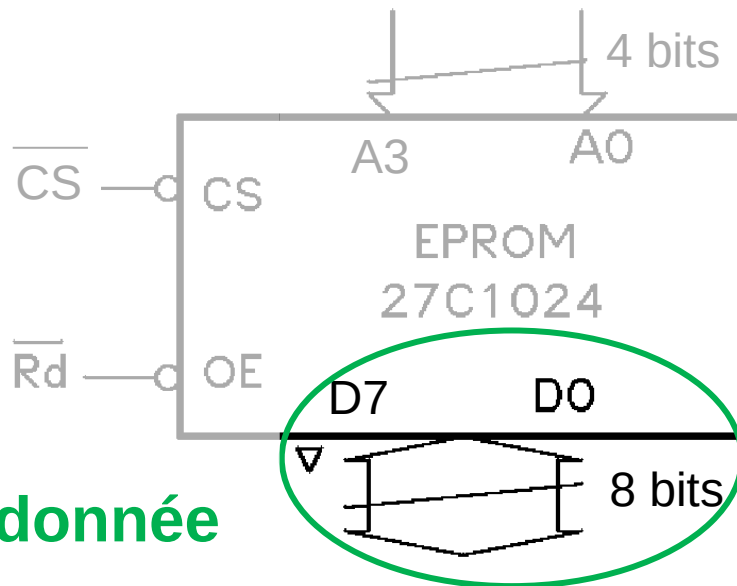
Bus d'adresse

Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 0 1 0 1 1 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 0 0 0 0 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Fonctionnement

Symbol d'un composant mémoire



Bus de donnée

Ici, le bus d'adresse a 8 bits de D0 à D7
L'ensemble des 8 bits forment un mot
Chaque mot est associé à 1 adresse

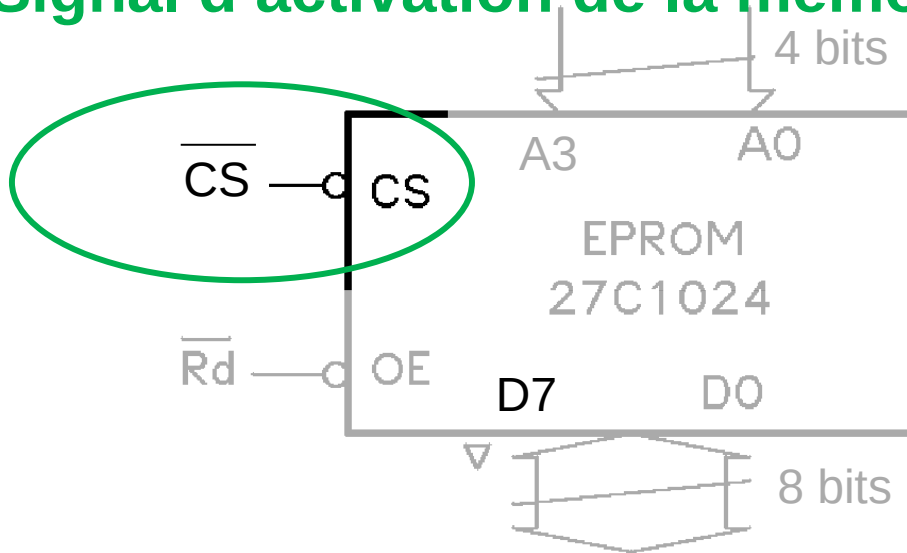
Contenu de la mémoire

Contenu de la mémoire	
Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 0 1 0 1 1 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 0 0 0 0 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Fonctionnement

Symbol d'un composant mémoire

Signal d'activation de la mémoire



Le signal \overline{CS} (Chip Select) permet d'activer la mémoire ou de la mettre en veille :

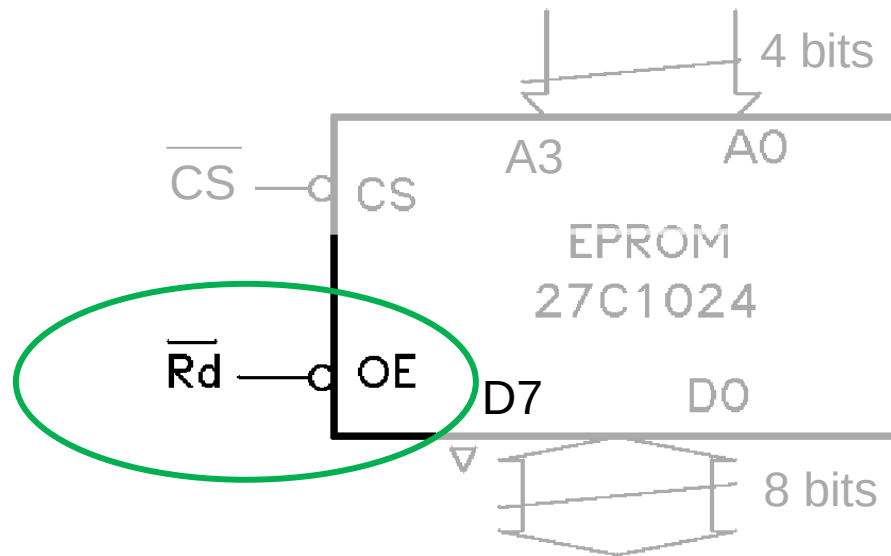
- Si $\overline{CS}=0$, la mémoire est active
- Si $\overline{CS}=1$, la mémoire est en veille, les données sont à l'état Z (haute impédance = déconnecté)

Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 0 1 0 1 1 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 0 0 0 0 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Fonctionnement

Symbol d'un composant mémoire



Signal lecture/écriture

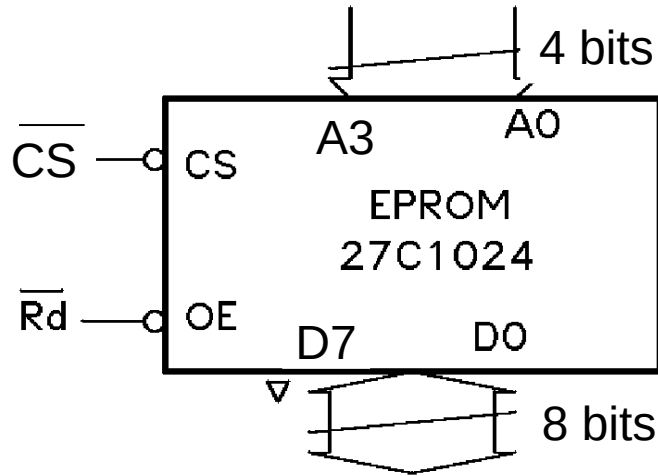
Le signal \overline{Rd} permet de choisir entre lecture et écriture (\overline{Rd} = read = lecture)

- Si $\overline{Rd}=0$, la mémoire est mode lecture
- Si $\overline{Rd}=1$, la mémoire est en mode écriture

Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 0 1 0 1 1 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 0 0 0 0 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Mode lecture



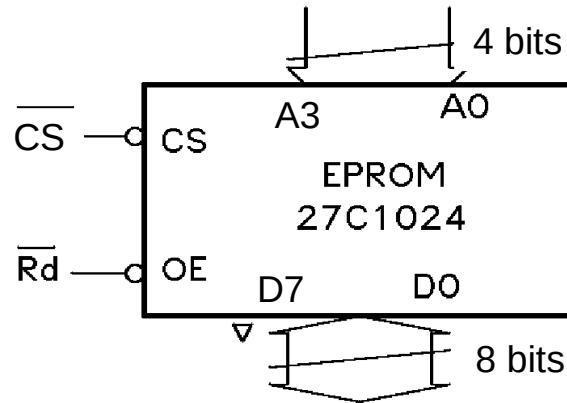
Si je veux lire le mot enregistré à l'adresse 0xA :

- Je dois demander l'adresse 0xA en envoyant A en hexa, c'est-à-dire 1010 en binaire sur le bus d'adresse
- Je dois mettre en mode lecture : $\overline{Rd} = 0$
- Je dois activer la mémoire : $\overline{CS} = 0$
- La mémoire envoie sur le bus de donnée 0010 0100 en binaire = 0x24 en hexa

Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 0 1 0 1 1 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 1 0 0 1 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Mode écriture



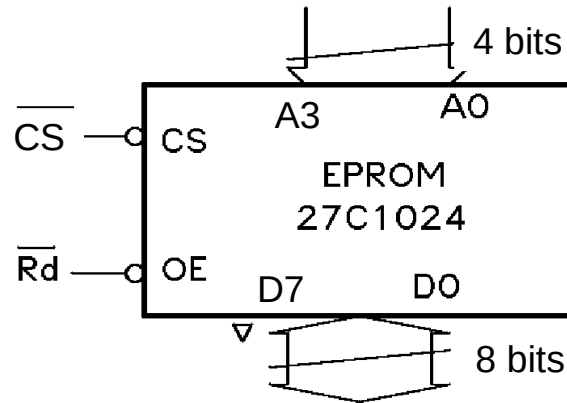
Si je veux écrire le mot 0xF1 à l'adresse 0x5 :

- Je dois demander l'adresse 0x5 en envoyant 5 en hexa, c'est-à-dire 0101 en binaire sur le bus d'adresse
- Je dois envoyer 0xF1 sur le bus de donnée, c'est à dire 1111 0001 en binaire
- Je dois mettre en mode écriture : $\overline{Rd} = 1$
- Je dois activer la mémoire : $\overline{CS} = 0$
- La mémoire enregistre 0xF1 à l'adresse 0x5

Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 1 1 1 0 0 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 1 0 0 1 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Taille de la mémoire



Cette mémoire a 8 bits de donnée

Elle a 4 bits d'adresse

Elle a donc $2^4 = 16$ adresses

A chaque adresse on enregistre 1 donnée de 8 bits

La mémoire contient donc $16 \times 8 = 128$ bits

Sa taille est donc 128 bits

Comme 1 octet = 8 bits, on peut aussi dire que sa taille est de $128 / 8 = 16$ octets

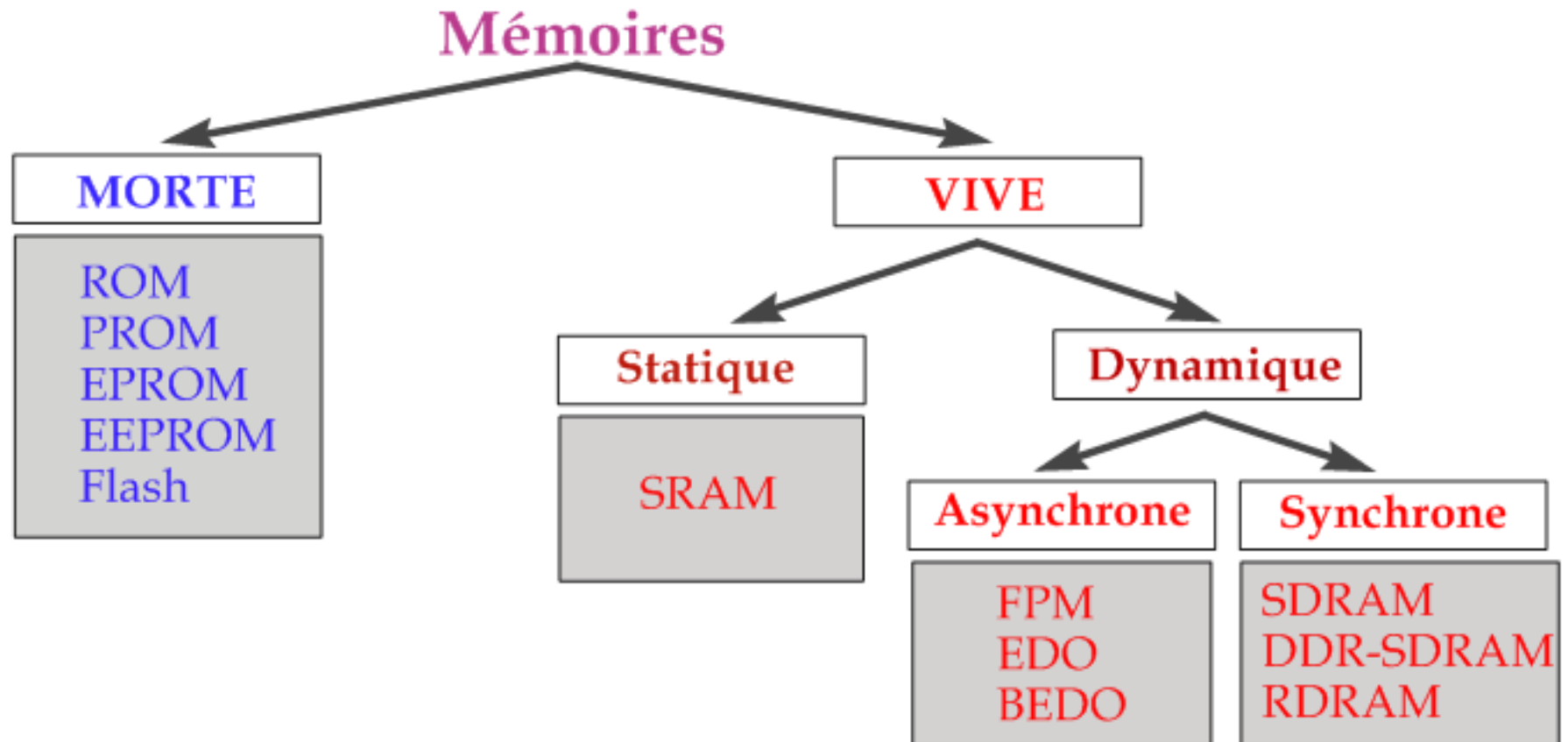
Contenu de la mémoire

Adresses	Données
0	0 1 1 0 1 1 0 1
1	0 1 0 0 0 1 0 1
2	0 0 1 0 1 1 1 1
3	1 1 0 1 0 1 0 1
4	0 1 1 0 1 0 0 1
5	1 1 1 1 0 0 0 1
6	0 0 1 1 1 0 0 0
7	1 1 0 0 0 1 0 1
8	0 1 1 1 1 0 1 0
9	1 0 1 0 1 0 0 1
A	0 0 1 0 0 1 0 0
B	1 1 1 1 1 1 1 1
C	0 1 1 1 1 0 1 0
D	0 1 0 0 0 1 0 1
E	1 1 0 0 0 1 0 1
F	0 1 1 1 1 0 1 0

Caractéristique des mémoires

- La **capacité** = sa taille
- Le **temps d'accès** = intervalle de temps entre la demande de lecture/écriture et la disponibilité de la donnée.
- Le **temps de cycle** = intervalle de temps minimum entre deux accès successifs
- La **non volatilité** : une mémoire volatile perd son contenu si on coupe son alimentation électrique. La RAM est volatile.

Les types de mémoires



Les mémoires vives

Ce sont des mémoires volatiles accessible en Lecture/ Ecriture
On distingue 2 catégories :

- SRAM = Static RAM
 - L'information maintenue spontanément sous tension
- DRAM = Dynamique RAM
 - L'information maintenue par rafraîchissement (balayage régulier de toutes les cases mémoires)
 - SDRAM : Synchronous DRAM
 - Synchronisation sur l'horloge de la carte mère
 - DDR SDRAM : Double Data Rate Synchronous SDRAM
 - Synchronisation sur l'horloge de la carte mère

Les mémoires mortes

- La **mémoire morte**, est un type de mémoire permettant de conserver les informations qui y sont contenues même lorsque la mémoire n'est plus alimentée électriquement. On distingue plusieurs types :
 - Les ROM (**R**ead **O**nly **M**emory) : Le contenu est défini lors de la fabrication.
 - Les PROM (**P**rogrammable **R**ead **O**nly **M**emory) : Elles sont programmables par l'utilisateur, mais une seule fois.
 - Les EPROM (**E**rasable **P**rogrammable **R**ead **O**nly **M**emory) : Elles sont effaçables et programmables par l'utilisateur par bombardement d'ultra violet
 - Les EEPROM (**E**rasable **P**rogrammable **R**ead **O**nly **M**emory) : Elles sont effaçables et programmables électriquement par l'utilisateur
 - Les mémoires flash

Activité moodle

Faire le test moodle sur les mémoires