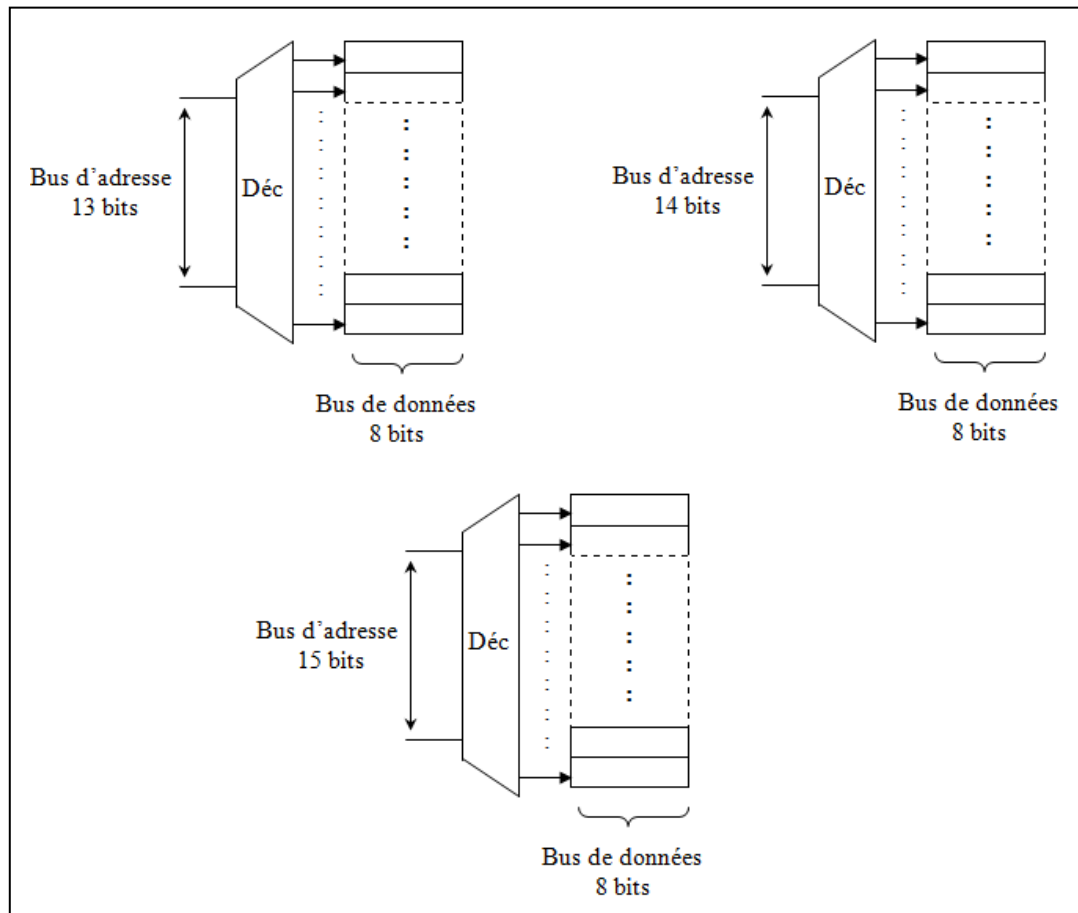


## Annexe Chapitre 3

### Exercice d'application:

Soient les circuits RAM suivants :



- Calculer la capacité totale de chaque mémoire.
- Calculer l'adresse de début et de fin de chaque mémoire.
- On veut associer à ces mémoires un processeur admettant 16 bits d'adresse et 8 bits de données. Donner la capacité totale adressable par le processeur.
- Dresser la table des adresses, sachant que les mémoires sont disposées successivement à partir de la première adresse (adresse 0) selon l'ordre suivant : Mémoire 1, Mémoire 2 et Mémoire 3.
- Calculer les équations de sélection  $CS_1$ ,  $CS_2$  et  $CS_3$ .
- Expliquer pourquoi la capacité totale adressable par le processeur permet d'introduire une quatrième puce mémoire. Calculer la taille maximale de la puce qu'on peut y ajouter.

## Annexe Chapitre 3

### Rappel

- Abréviations et terminologies**

Capacité d'Adressage (**CA**) = Nombre de cases mémoire

Capacité Mémoire (**CM**) = Taille de la Mémoire Centrale

Taille du Mot de Donnée (**TMD**) = Taille de la Case Mémoire

Nombre de lignes d'adresse (**n**) = nombre de bits d'adresse

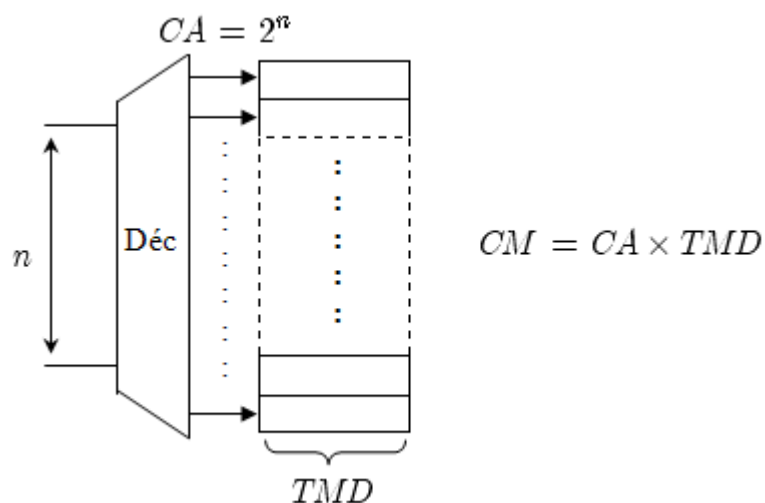
- Formules**

$$CA = 2^n$$

$$n = \log_2(CA)$$

$$CM = CA \times TMD = 2^n \times TMD$$

- Schéma**



### Correction de l'exercice d'application

**1.**

Mémoire 1:

$$\begin{aligned}
 CM_1 &= 2^{13} \times 8 \text{ bits} = (2^3 \times 2^{10}) \times 1 \text{ octet} \\
 &= 8 \times 1k \times 1 \text{ octet} = 8k \text{ octet}
 \end{aligned}$$

Mémoire 2 :

$$\begin{aligned}
 CM_2 &= 2^{14} \times 8 \text{ bits} = (2^4 \times 2^{10}) \times 1 \text{ octet} \\
 &= 16 \times 1k \times 1 \text{ octet} = 16k \text{ octet}
 \end{aligned}$$

Mémoire 3 :

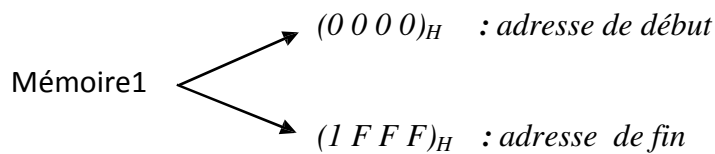
$$\begin{aligned}
 CM_3 &= 2^{15} \times 8 \text{ bits} = (2^5 \times 2^{10}) \times 1 \text{ octet} \\
 &= 32 \times 1k \times 1 \text{ octet} = 32k \text{ octet}
 \end{aligned}$$

## 2.

- Mémoire 1 : (13 bits d'adresse)

→ L'adresse de début =  $(0)_{10} = (0000000000000)_2 = (0000)_H$

→ L'adresse de fin =  $(2^{13}-1)_{10} = (1111111111111)_2 = (1FFF)_H$



- Mémoire 2 : (14 bits d'adresse)

→ L'adresse de début =  $(00000000000000)_2 = (0000)_H$

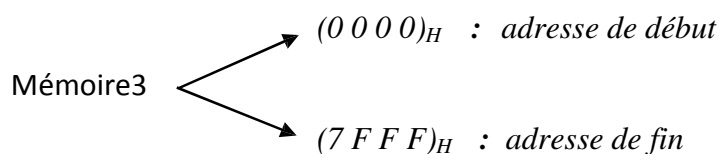
→ L'adresse de fin =  $(11111111111111)_2 = (3FFF)_H$



- Mémoire 3 : (15 bits d'adresse)

→ L'adresse de début =  $(000000000000000)_2 = (0000)_H$

→ L'adresse de fin =  $(111111111111111)_2 = (7FFF)_H$



## 3.

Le processeur admet 16 bits d'adresse et 8 bits de données, la capacité de la mémoire adressable par ce processeur est donc :

$$\begin{aligned}
 CM_{Totale} &= 2^{16} \times 8 \text{ bits} = (2^6 \times 2^{10}) \times 1 \text{ octet} \\
 &= 64 \times 1k \times 1 \text{ octet} = 64k \text{ octet}
 \end{aligned}$$

4.

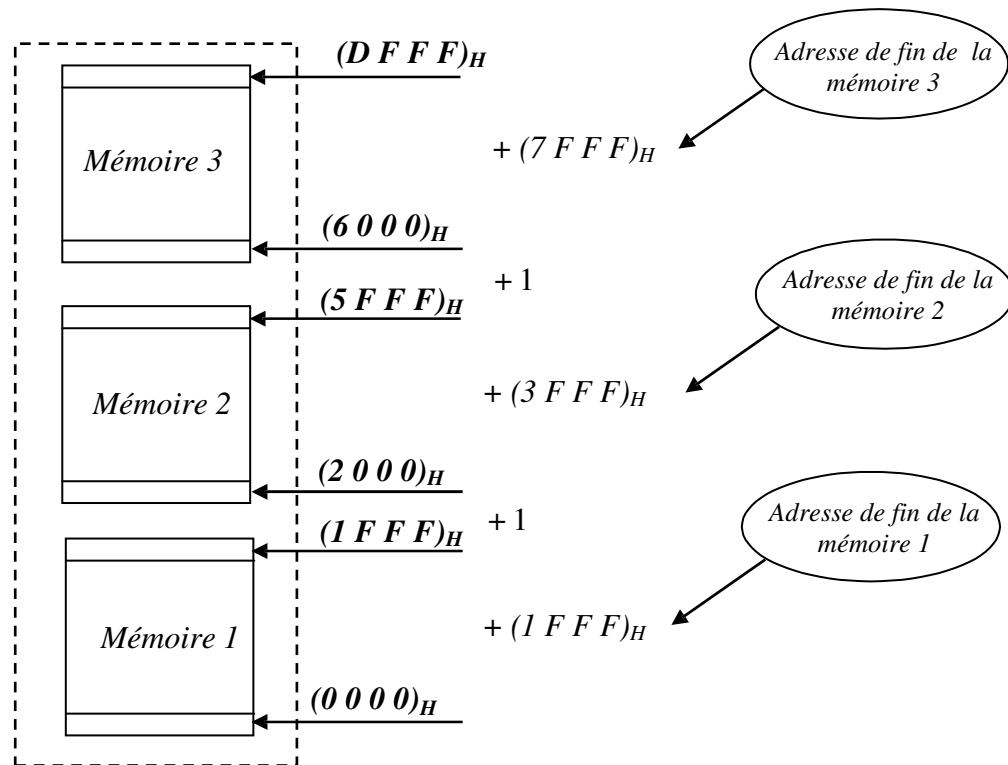


Table des adresses :

$a_{15}$	$a_{14}$	$a_{13}$	$a_{12}$	$a_{11}$	$a_{10}$	$a_9$	$a_8$	$a_7$	$a_6$	$a_5$	$a_4$	$a_3$	$a_2$	$a_1$	$a_0$	@Hex	Mémoires
1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	DFFF	Mémoire 3
0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6000	
0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	5FFF	Mémoire 2
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2000	
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFF	Mémoire 1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	

5. En exploitant la table des adresses, on peut déduire les équations de sélection de chaque mémoire.

$$CS_1 = \overline{a_{15}} \cdot \overline{a_{14}} \cdot \overline{a_{13}}$$

$$CS_2 = \overline{a_{15}} \cdot \overline{a_{14}} \cdot a_{13} + \overline{a_{15}} \cdot a_{14} \cdot \overline{a_{13}} = \overline{a_{15}} (a_{14} \oplus a_{13})$$

$$CS_3 = \overline{a_{15}} \cdot a_{14} \cdot a_{13} + a_{15} \cdot \overline{a_{14}} \cdot \overline{a_{13}} + a_{15} \cdot \overline{a_{14}} \cdot a_{13} + a_{15} \cdot a_{14} \cdot \overline{a_{13}} = \overline{a_{15}} \cdot a_{14} \cdot a_{13} + a_{15} \cdot \overline{a_{14}} \cdot \overline{a_{13}} + a_{15} (a_{14} \oplus a_{13})$$

6. La capacité totale adressable par processeur permet d'introduire une quatrième puce mémoire. Ceci est expliqué par le fait que la capacité adressable par le processeur est encore plus grande que la capacité mémoire totale des trois mémoires disponibles.

La taille de la puce = 64kocet - (8kocet + 16kocet + 32 kocet) = **8 koctet**.