

Université 8 mai 1945 de Guelma
Faculté des Mathématiques, D'Informatique
et des Sciences de la Matière
Département : Informatique

Série N°3

Exercice N°1

Soit une Mémoire principale de 64 Mi mots de 32 bits, nous avons associé à cette mémoire un cache totalement associatif (Fully associative) d'une capacité de 1Mio avec des lignes de 128 octets.

1. Calculer le nombre de lignes du cache.
2. Calculer le nombre de blocs de la MP.
3. Calculer le nombre de bits d'adressage.
4. Calculer l'offset et le Tag et en déduire le format de l'adresse mémoire.
5. Donnez l'adresses du premier mot dans les lignes contenant les adresses suivantes : 0x35617A6, 0x285E832, 0x3B025BA.

Exercice N°2

Un système de cache est utilisé pour optimiser l'accès à la mémoire. Le cache en question a une taille totale de **4 Ko** (4096 octets), et les blocs de données sont de **64 octets**. Pour chaque type de cache effectuer les calculs suivants :

1. Cache Direct
 - a. Calculez le **nombre de blocs** dans le cache.
 - b. Déterminez le **nombre de bits pour l'index**, l'offset et le **tag**.
 - c. Donnez la structure de l'adresse (en bits) et **décomposez l'adresse donnée 0x1F3A7C80 en Tag, Index et Offset**.
2. Cache Totalement Associatif
 - a. Déterminez le **nombre de bits pour l'offset** et le **tag**.
 - b. Donnez la structure de l'adresse (en bits).
 - c. **Décomposez** l'adresse donnée 0x1F3A7C80 en **Tag et Offset**.
3. Cache Associatif par Ensemble (8 voies)
 - a. Calculez le **nombre d'ensembles** et le **nombre de voies par ensemble**.
 - b. Déterminez le **nombre de bits pour l'index**, l'offset et le **tag**.
 - c. Donnez la structure de l'adresse (en bits).
 - d. **Décomposez** l'adresse donnée 0x1F3A7C80 en **Tag, Set Index et Offset**.
4. Donner vos conclusions.

Exercice N°3

Considérons un système avec un cache de **4 blocs seulement**. Lorsqu'un bloc doit être ajouté au cache et que celui-ci est plein, un remplacement doit être effectué. La politique de remplacement utilisée est **LFU ou LRU**.

On donne la séquence suivante d'accès mémoire (en termes de numéros de blocs) : 2,3,2,1,5,2,4,5,3,2,4,6.

1. Effectuez pour chaque politique de remplacement (LFU et LRU) le suivant :
 - a. Sur un tableau récapitulatif simuler le comportement du cache a chaque accès effectué :
 - En déterminant s'il s'agit d'un **hit** (succès cache) ou d'un **miss** (échec cache).
 - En désignant les blocs remplacés si le cache est plein.

- b. Calculez le nombre total et les taux de **hit** et de **miss**.
2. Comparez et commentez les résultats obtenus ?

Remarque :

- Notons que le cache est vide au début.
- Dans la politique LFU, en cas d'égalité de fréquences, utiliser la politique FIFO pour l'arbitrage.

Exercice N°4

Nous cherchons à optimiser une mémoire cache ayant une capacité totale de **8 octets**. Trois conceptions de cache **direct** sont possibles selon la taille des blocs :

Conception 1 : Cache avec des blocs de **1 octet et temps d'accès : 2 cycles**

Conception 2 : Cache avec des blocs de **2 octets et temps d'accès : 3 cycles**

Conception 3 : Cache avec des blocs de **4 octets et temps d'accès : 5 cycles**

Notons que le temps d'attente en cas de défaut (*miss stall time*) : **25 cycles**

On donne la séquence suivante d'accès mémoire (en terme d'adresse) : **1, 134, 212, 1, 135, 213, 162, 161, 2, 44, 41, 221.**

Remarques

- Chaque accès à la mémoire cache (un succès ou un défaut) nécessite un temps d'accès.
- Les adresses ont été converties en format binaire pour simplifier la tâche.

Faire pour chaque conception le suivant :

1. Décomposez les adresses en **bits de tag**, **bits d'index**, et **bits d'offset**, selon la taille des blocs.
2. Complétez le tableau (ci-dessous) des accès en identifiant si chaque adresse entraîne un **hit** ou un **miss**.
3. Calculer le taux de défaut (miss rate) et en déduire la configuration qui minimise les défauts ?
4. Calculer le nombre de cycle totale et en déduire la conception de cache optimale.

Adresse (en décimale)	Adresse (en binaire) 8 bits LS de l'adresse	Index (...bits)	Offset (... bits)	Hit/Miss
1	0000 0001			
134	1000 0110			
212	1101 0100			
1	0000 0001			
135	1000 0111			
213	1101 0101			
162	1010 0010			
161	1010 0001			
2	0000 0010			
44	0010 1100			
41	0010 1001			
221	1101 1101			