

Architecture avancée : TD7

Exercice 1 : Alignement

Sujet

On considère un cache à correspondance directe (direct mapped), à 2^p lignes stockant chacune 2^m octets. Le système qui l'utilise adresse la mémoire sur 32 bits. On appellera mot de la mémoire un ensemble de 4 octets.

1. Détailler la translation (adresse en mémoire centrale) \rightarrow (ligne de la mémoire cache) en fonction de m et p , i.e., le découpage en offset, index et tag.
2. Supposons $m = 7$: chaque ligne contient 128 octets de la mémoire centrale. Quelle est, dans ce cas, la longueur (en nombre de bits), de l'offset d'une adresse ?
3. Donner l'adresse en mémoire centrale des premiers mots contenus dans les lignes contenant les mots d'adresses :
 - 0xA23847EC
 - 0x7245E824
 - 0xEEFABCD8
4. On suppose maintenant $m = 7$, que $p = 9$ et que le cache est toujours à correspondance directe. Quelle est, dans ce cas, la longueur (en nombre de bits), de l'index d'une adresse ?

Résolution

Question 1

- 2^p lignes \Rightarrow Index sur p bits
- 2^m octets \Rightarrow Offset sur m bits
- \Rightarrow Tag sur $32 - p - m$ bits

Question 2

- Longueur de l'offset : 128

Question 3

- 0xA23847EC

- EC -> 1110 1100 Offset sur 7 bits donc on met à 0 ces 7 bits, premier mot sur 1000 0000 -> 80
- 0xA2384780
- 0x7245E824
 - 0x7245E800
- 0xEEFABCD8
 - 0xEEFABC80

Question 4

- 7 bits
-

Exercice 2 : Accès mémoire

Sujet

Un processeur utilise des adresses sur 16 bits et dispose d'une mémoire cache direct mapped, de capacité totale de 16 mots, avec des blocs de 4 mots. Un mot mémoire est représenté sur 32 bits.

1. Détailler la translation (adresse en mémoire centrale) → (mémoire cache), i.e., le découpage en offset, index et tag. Un programme utilise des variables stockées aux adresses suivantes : 0x74, 0xA0, 0x78, 0x38C, 0xAC, 0x84, 0x88.
2. Calculez le nombre de défauts dans la mémoire cache.

Question 1

- 0x74
 - tag : 0000 0000 00
 - index : 11
 - offset : 0100