Partie 2

1. L’adresse mémoire est d’une taille de 20 bits.

Réparti comme suis :

|  |  |  |  |
| --- | --- | --- | --- |
|  | Tag | Ensemble | Word |
| Direct | 6 | 10 | 4 |
| Associatif par ensemble de 2 blocs | 7 | 9 | 4 |
| Associatif par ensemble de 4 blocs | 8 | 8 | 4 |

2.

Toutes les réponses sont en hexadécimal

**Tableau de différents accès mémoire**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Direct | | | | 2 blocs | | | | 4 blocs | | | |
| Accès | Tag | Set | Hit | w-b | Tag | Set | Hit | w-b | Tag | Set | Hit | w-b |
| WR 0x5EF1D | 17 | 2F1 |  |  | 2F | F1 |  |  | 5E | F1 |  |  |
| WR 0x19C7C | 6 | 1C7 |  |  | C | 1C7 |  |  | 19 | C7 |  |  |
| RD 0x5EF1B | 17 | 2F1 | x |  | 2F | F1 | x |  | 5E | F1 | x |  |
| RD 0x8CDB0 | 23 | DB |  |  | 46 | DB |  |  | 8C | DB |  |  |
| WR 0x3CDB3 | F | DB |  |  | 1E | DB |  |  | 3C | DB |  |  |
| WR 0x5EF15 | 17 | 2F1 | x |  | 2F | F1 | x |  | 5E | F1 | x |  |
| RD 0x68DBF | 1A | DB |  |  | 34 | DB |  |  | 68 | DB |  |  |
| WR 0xCAF1C | 32 | 2F1 |  | x | 65 | F1 |  |  | CA | F1 |  |  |
| RD 0x39C7E | E | 1C7 |  |  | 1C | 1C7 |  |  | 39 | C7 |  |  |
| WR 0xCAF1A | 32 | 2F1 | x |  | 65 | F1 | x |  | CA | F1 | x |  |



**État de la cache avec version de placement direct**

|  |  |
| --- | --- |
| Set | Tag0 |
| 2F1 | 32\* |
| 1C7 | 06 |
| 0DB | 0E |

**État de la cache avec version de placement associative par ensemble de 2 blocs**

|  |  |  |
| --- | --- | --- |
| Set | Tag0 | Tag1 |
| 0F1 | 2F\* | 65\* |
| 1C7 | 0C | Vide |
| 0DB | 1E | Vide |

**État de la cache avec version de placement associative par ensemble de 4 blocs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Set | Tag0 | Tag1 | Tag2 | Tag3 |
| 0F1 | 5E\* | CA\* | Vide | Vide |
| 0C7 | 19 | Vide | Vide | Vide |
| 0DB | 3C | Vide | Vide | Vide |

2.4

h\*tp + (1-h)\*ts = temps d’accès effectif

h = 7/10

1-h = 3/10

Tp = 8ns

Ts = 100ns

Pour la version de placement direct :

3/10 \* (8ns) + 7/10 \*(100 ns) + 1/10 (100ns) = 82.4 ns

Pour la version de placement associative par ensemble de 2 blocs :

3/10 \* (8ns) + 7/10 \*(100ns) = 72.4 ns

Pour la version de placement associative par ensemble de 4 blocs :

3/10 \* (8ns) + 7/10 \*(100ns) = 72.4 ns

2.5

Il n’y aurait plus eu d’ensemble dans l’adresse mémoire, le tag aurait occupé 16 bits sur les 20 bits de l’adresse et octet les 4 bits restant. Ainsi, dans l’adresse on ne retrouverait que le tag et word.