**3η ΣΕΙΡΑ ΑΣΚΗΣΕΩΝ**

Παπαδόπουλος Χαράλαμπος

03120199

|  |  |  |  |
| --- | --- | --- | --- |
| **Address** | **Address** | **Hit/Miss** | **Miss Type** |
| 0x184 | 0 110 000 100 | Miss | Compulsory |
| 0x182 | 0 110 000 010 | Hit |  |
| 0x188 | 0 110 001 000 | Miss | Compulsory |
| 0x0EE | 0 011 101 110 | Miss | Compulsory |
| 0x18A | 0 110 001 010 | Miss | Conflict |
| 0x2F4 | 1 011 110 100 | Miss | Compulsory |
| 0x180 | 0 110 000 000 | Hit |  |
| 0x2F8 | 1 011 111 000 | Miss | Compulsory |
| 0x0EF | 0 011 101 111 | Miss | Conflict |

Εφόσον το 0x182 είναι hit καταλαβαίνουμε πως έχουμε τουλάχιστον 3 bit offset.

Το 0x188 όμως είναι miss, άρα το offset είναι 3-bit.

Αφού το 0x18A είναι miss, το index είναι το πολύ 1 ή 2-bit, διαφορετικά θα είχαμε hit λόγω του 0x188.

Εάν όμως το index ήταν 1-bit, τότε το 0x180 θα ήταν miss λόγω του 0x2F4.

Άρα το index είναι 2 bit και το tag 5.

Εφόσον έχουμε index = 2 => 4 blocks και offset = 3 => 8B δεδομένων ανά μπλοκ.

Άρα, tag array = 4 \* 5 = 20-bit

Cache = 4 \* 8 = 32B



Α)

32B μέγεθος block => 5-bit offset

256/32 = 8 block => 3-bit index

Άρα, 24 bit tag

Πρώτο στοιχείο Α: 101010111010101110101011 101 00000

Πρώτο στοιχείο Β: 110011101110110011001110 101 00000

Β)

Κάθε block θα περιέχει 4 στοιχεία.

Α[0][0] m B[0] m A[0][0] m

A[0][1] h B[1] m A[0][1] m

A[0][2] h B[2] m A[0][2] h

A[0][3] h B[3] m A[0][3] h

A[0][4] m B[4] m A[0][4] m

A[0][5] h B[5] m A[0][5] m

A[0][6] h B[6] m A[0][6] m

A[0][7] h B[7] m A[0][7] m

…

A[1][0] m B[16] m A[1][0] m

A[1][1] h B[17] m A[1][1] m

…

Ανά 4 σειρές έχω 9 misses και 3 hits.

Οπότε συνολικά 16 \* 2 \* 4 \* (9m + 3h) = 1.152m + 384h

Γ)

Τώρα που η cache είναι write-no-allocate δεν θα περνιούνται τα blocks στην cache μετά από κάθε miss. Οπότε οι γραμμές θα είναι πλέον:

A[0][0] m B[0] m A[0][0] m

A[0][1] m …

Δηλαδή δεν θα έχω hit read μετά από miss write ανά 4 στοιχεία.

Τελικά, 1.536 misses και 0 hits