

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΕΜΠ

“Αρχιτεκτονική Υπολογιστών”

Ορφανουδάκης Φίλιππος 03113140

Άσκηση

Μας δίνεται ο ακόλουθος κώδικας :

```
int i, j;
double A[8][8], B[64], C[4][4];
for (i = 0; i < 8; i++)
{
    for (j = 0; j < 8; j++)
    {
        A[i][j] = B[8*i+j] + C[i%4][j%4];
    }
}
```

- Κάθε πίνακας έχει στοιχεία 8 bytes το καθένα
- 1 επίπεδο κρυφής μνήμης
- Τα χαρακτηριστικά της είναι : direct-mapped, write-allocate, write-back , χωρητικότητα 512 bytes
- Το μέγεθος του block είναι 64 bytes.
- Οι πίνακες είναι αποθηκευμένοι στην κύρια μνήμη κατά γραμμές και διαδοχικά, δηλαδή ο ένας μετά τον άλλο. Το πρώτο στοιχείο του πίνακα A βρίσκεται στη διεύθυνση 0x00008000.

A)

Κάθε block είναι 64 bytes και συνολικά η μνήμη έχει 512 bytes άρα μιλάμε για $\#blocks=8$, επομένως το **index = 3 bit**.

Επίσης block size=64 bytes , άρα το **offset = 6 bit**.

Χειριζόμαστε διευθύνσεις 32 bit , επομένως **tag=32-3-6=23 bit**

B)

- Κάθε block έχει χωρητικότητα 64 bytes και κάθε στοιχείο του πίνακα είναι 8 bytes , επομένως σε κάθε block θα φέρνω 8 στοιχεία του πίνακα.
- Ο πίνακας A ξεκινάει από το 00000000000000001000000 000 000000.
Έχει 64 στοιχεία , κάθε στοιχείο είναι 8 bytes άρα θα τελειώνει στο :
00000000000000001000000 111 111111.
- Ο πίνακας B ξεκινάει από το 00000000000000001000001 000 000000.
Έχει 64 στοιχεία , κάθε στοιχείο είναι 8 bytes άρα θα τελειώνει στο :
00000000000000001000001 111 111111.
- Ο πίνακας C ξεκινάει από το 00000000000000001000010 000 000000.
Έχει 16 στοιχεία , κάθε στοιχείο είναι 8 bytes άρα θα τελειώνει στο :
00000000000000001000010 011 111111

Επομένως όλα ανταγωνίζονται για το πρώτο block, αφού έχουν ίδιο index.

Block 0 : A[0][0]-A[0][7]

Block 1: A[1][0]-A[1][7]

κτλ

Block 0 : B[0]-B[[7]

Block 1: B[8]-B[15]

κτλ

Block 0: C[0][0]-C[1][3]

Block 1: C[2][0]-C[3][3]

κτλ

- Αφού είναι write-allocate τότε θα γίνεται η εγγραφή και στη κρυφή μνήμη καθώς θα φέρνουμε το block.
- m + = compulsory miss.

Επομένως θα έχω το εξής.

Για i=0

Στοιχεία	Block					Miss/Hit		
----------	-------	--	--	--	--	----------	--	--

B[0]	C[0][0]	A[0][0]	0	0	0	m +	m +	m +
B[1]	C[0][1]	A[0][1]	0	0	0	m	m	m
B[2]	C[0][2]	A[0][2]	0	0	0	m	m	m
B[3]	C[0][3]	A[0][3]	0	0	0	m	m	m
B[4]	C[0][0]	A[0][4]	0	0	0	m	m	m
B[5]	C[0][1]	A[0][5]	0	0	0	m	m	m
B[6]	C[0][2]	A[0][6]	0	0	0	m	m	m
B[7]	C[0][3]	A[0][7]	0	0	0	m	m	m

Για i=1 έχω :

Στοιχεία	Block					Miss/Hit		
----------	-------	--	--	--	--	----------	--	--

B[8]	C[1][0]	A[1][0]	1	0	1	m +	m	m +
B[9]	C[1][1]	A[1][1]	1	0	1	m	h	m
B[10]	C[1][2]	A[1][2]	1	0	1	m	h	m
B[11]	C[1][3]	A[1][3]	1	0	1	m	h	m
B[12]	C[1][0]	A[1][4]	1	0	1	m	h	m
B[13]	C[1][1]	A[1][5]	1	0	1	m	h	m
B[14]	C[1][2]	A[1][6]	1	0	1	m	h	m
B[15]	C[1][3]	A[1][7]	1	0	1	m	h	m

για $i=2$ έχω :

Στοιχεία			Block			Miss/Hit		
B[16]	C[2][0]	A[2][0]	3	1	3	m +	m+	m +
B[17]	C[2][1]	A[2][1]	3	1	3	m	h	m
B[18]	C[2][2]	A[2][2]	3	1	3	m	h	m
B[19]	C[2][3]	A[2][3]	3	1	3	m	h	m
B[20]	C[2][0]	A[2][4]	3	1	3	m	h	m
B[21]	C[2][1]	A[2][5]	3	1	3	m	h	m
B[22]	C[2][2]	A[2][6]	3	1	3	m	h	m
B[23]	C[2][3]	A[2][7]	3	1	3	m	h	m

Βλέπουμε ότι αυτό το μοτίβο συνεχίζεται , επομένως μπορούμε να πούμε ότι έχουμε 24 misses στην πρώτη επανάληψη.

Και στις επόμενες έχουμε $7 \cdot 16 = 112$ misses απο τους πίνακες A,B και +2 απο τον πίνακα C , συνολικά $24 + 114 = 138$ misses από αυτά τα 18 είναι compulsory και τα 120 είναι conflict .

Επομένως έχω $192 - 138 = 54$ hits , άρα $54/192 = 28,1\%$ hit rate.

Γ)

i) Αντι για direct - mapped χρησιμοποιώ τεχνική 2-way set associative , επομένως έχω set και κάθε set έχει 2 blocks επομένως έχω 4 set.

Άρα :

set0 : block0,block1

set1 : block2,block3

κτλ

Η τεχνική αντικατάστασης είναι LRU δηλαδή θα φύγει το μπλοκ που χρησιμοποιήθηκε παλαιότερα.

Άρα για $i=0$ έχω :

Στοιχεία	Set			Miss/Hit
----------	-----	--	--	----------

B[0]	C[0][0]	A[0][0]	0	0	0	m+	m+	m+
B[1]	C[0][1]	A[0][1]	0	0	0	m	m	m
B[2]	C[0][2]	A[0][2]	0	0	0	m	m	m
B[3]	C[0][3]	A[0][3]	0	0	0	m	m	m
B[4]	C[0][0]	A[0][4]	0	0	0	m	m	m
B[5]	C[0][1]	A[0][5]	0	0	0	m	m	m
B[6]	C[0][2]	A[0][6]	0	0	0	m	m	m
B[7]	C[0][3]	A[0][7]	0	0	0	m	m	m

Για i=1 έχω :

Στοιχεία	Set			Miss/Hit
----------	-----	--	--	----------

B[8]	C[1][0]	A[1][0]	1	0	1	m +	m	m +
B[9]	C[1][1]	A[1][1]	1	0	1	h	h	h
B[10]	C[1][2]	A[1][2]	1	0	1	h	h	h
B[11]	C[1][3]	A[1][3]	1	0	1	h	h	h
B[12]	C[1][0]	A[1][4]	1	0	1	h	h	h
B[13]	C[1][1]	A[1][5]	1	0	1	h	h	h
B[14]	C[1][2]	A[1][6]	1	0	1	h	h	h
B[15]	C[1][3]	A[1][7]	1	0	1	h	h	h

Αυτό το μοτίβο της επανάληψης θα συνεχιστεί και στις υπόλοιπες επαναλήψεις
Ο λόγος είναι ότι επειδή η αναθέτηση πάει κυκλικά θα υπάρχει ανταγωνισμός παί στη 5 επανάληψη για το set 0 ,
Άρα $2 \cdot (21) = 42$ misses + 16 = 58 misses απο αυτά τα 18 είναι compulsory και τα 36 είναι conflict.

Άρα έχω $192 - 54 = 138$ hits , άρα έχω $138 / 192 = 71,8\%$ hit rate.

ii)

Εδώ έχω μια ιδιαίτερη περίπτωση όπου χρησιμοποιείται write - back μαζί με write-no-allocate αυτό σημαίνει ότι η εγγραφή θα γίνεται μόνο στη κύρια μνήμη σε περίπτωση miss και δεν θα έρχεται το block στη κρυφή , θα μένει το προηγούμενο περιεχόμενο, σε περίπτωση hit συνήθως υπάρχει ένα dirty bit έτσι ώστε να ενημερωθεί μετά η κύρια μνήμη αφού έχουμε ετεροχρονισμένη εγγραφή.

direct - mapped

αρα για i=0 έχω :

Στοιχεία			Block			Miss/Hit		
B[0]	C[0][0]	A[0][0]	0	0	-	m +	m +	m(+)
B[1]	C[0][1]	A[0][1]	0	0	-	m	m	m(+)
B[2]	C[0][2]	A[0][2]	0	0	-	m	m	m(+)
B[3]	C[0][3]	A[0][3]	0	0	-	m	m	m(+)
B[4]	C[0][0]	A[0][4]	0	0	-	m	m	m(+)
B[5]	C[0][1]	A[0][5]	0	0	-	m	m	m(+)
B[6]	C[0][2]	A[0][6]	0	0	-	m	m	m(+)
B[7]	C[0][3]	A[0][7]	0	0	-	m	m	m(+)

Για i=1

Στοιχεία			Block			Miss/Hit		
B[8]	C[1][0]	A[1][0]	1	0	-	m +	h	m(+)
B[9]	C[1][1]	A[1][1]	1	0	-	h	h	m(+)
B[10]	C[1][2]	A[1][2]	1	0	-	h	h	m(+)
B[11]	C[1][3]	A[1][3]	1	0	-	h	h	m(+)

B[12]	C[1][0]	A[1][4]	1	0	-	h	h	m(+)
B[13]	C[1][1]	A[1][5]	1	0	-	h	h	m(+)
B[14]	C[1][2]	A[1][6]	1	0	-	h	h	m(+)
B[15]	C[1][3]	A[1][7]	1	0	-	h	h	m(+)

Πλέον ο πίνακας A δεν έρχεται με το block του μαζί , αλλά απευθυνόμαστε στη κύρια μνήμη κάθε φορά για να γράψουμε . Επομένως κάθε φορά έχουμε αστοχία για τον πίνακα A το οποίο μπορεί να θεωρηθεί και compulsory με τις συγκεκριμένες προδιαγραφές,

Το μοτίβο συνεχίζεται και επομένως θα έχω 54 misses απο τον πίνακα A, $8+7=15$ απο τον πίνακα B και $8+2=10$ από τον πίνακα C.Συνολικά 79 misses.

από αυτά τα 10 είναι compulsory καθώς αφορούν block στη κύρια μνήμη στα οποία δεν έχει γίνει προηγουμενως αναφορά και τα υπόλοιπα είναι conflict καθώς και αστοχίες που αφορούν τον πίνακα A.

Επομένως $192-79=113$ hits. Άρα $113/192=58,8\%$ hit rate.

Και οι δύο τεχνικές βελτώνουν αισθητά το hit rate σε σχέση με την αρχιτεκτονική που χρησιμοποιήθηκε στο B).