

Αρχιτεκτονική Διαλέξη 12

Μια RAM 256 byte, μέγεθος block 4 bytes

Η cache 64 byte, οργανωμένη σε σύνολα 4 δρόμων.

Διευθύνσεις που αντιστοιχίζονται στο σύνολο 2

Κάθε γραμμή 4 byte

0	set 0	$\frac{64}{4} = 16$ γραμμές 4 δρόμοι: 4 γραμμές / σύνολο						
1								
2								
3								
4	set 1	Κάθε σετ μπορεί να υποδεχτεί οποιαδήποτε μπλοκ που αντιστοιχεί σε αυτό. Δεν υπάρχει διευθυνσιοδότηση γραμμής. Υπάρχει διευθυνσιοδότηση set. Κάθε set πρέπει να υποδεχτεί οποιαδήποτε block που αντιστοιχεί σε αυτό. Δεν έχω για κάθε block της RAM 1 γραμμή της cache. Έχω τόσες όσοι οι δρόμοι. Το πλήθος συγκριζών που απαιτούνται = πλήθος δρόμων. Κάθε block της RAM που αντιστοιχίζεται σε 1 σετ, μπορεί να πάει οπουδήποτε μέσα στο σετ.						
5								
6								
7								
8	set 2	RAM 256 byte : 8 bit tag set δείξη						
9								
10								
11								
12	set 3	<table border="1"> <thead> <tr> <th>tag</th> <th>set</th> <th>δείξη</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>2</td> <td>2</td> </tr> </tbody> </table>	tag	set	δείξη	4	2	2
tag			set	δείξη				
4			2	2				
13								
14								
15								

set 2 byte αφού έχουμε 4 set (00, 01, 10, 11)

Tag	Set	byte	Tag	Set	byte	
0000	10	00 = 8	0001	10	00 = 24	Πόσα block της RAM αντιστοιχίζονται στο set 2?
0000	10	01 = 9	0001	10	01 = 25	Εφόσον έχουμε 4 bit σε tag $2^4 = 16$
0000	10	10 = 10	0001	10	10 = 26	
0000	10	11 = 11	0001	10	11 = 27	

$$2) \frac{64 \text{ block}}{4 \text{ set}} = 16$$

1) Ζητείται η διεύθυνση 8

0000	10	00
------	----	----

Η διεύθυνση 8 βρίσκεται (αντιστοιχίζεται) στο set 2. Συνεχώς πηγαίνουμε στο set 2 και συγκρίνουμε το tag 0000 με τα 4 tag που υπάρχουν

Εκεί απά MISS

tag directory	DATA	
1000	ACD9	set 0
1001	853A	
1010	C5DA	
1100	852A	
1000	253A	set 1
1001	382A	
1010	451A	
1100	755A	set 2
1100	253A	
1101	6574	
1110	3281	
1000	6000	set 3

LRU:

tagdir	Data
0000	DATA

2) Ζητείται η διεύθυνση 129.

1000	00	01
------	----	----

set=0 απά Hit

Cache 2 δρομικών γραμμών 32 byte RAM 64 kbyte ενώ η cache 256 bytes
1) Ανάλυση διεύθυνσης
2) Αρχικά η cache είναι αδιά. Για αντικατάσταση FIFO. Η CPU παράγει τις διευθύνσεις 23AB, 4B4F, FFFF, 2727, 5555, 3894 α) miss/hit β) να δώσει τη θέση που θα μπει η Cache

64 Kb = 2^{16} , 16 bit Κάθε γραμμή 32 byte, άρα το πεδίο λέξης = 5 bit

cache $\frac{256 \text{ byte}}{32 \text{ byte/γραμμή}} = 8 \text{ γραμμές}$

cache 2 way $\Rightarrow \frac{8}{2} = 4$ σύνολα Πλήθος συνόλων = $\frac{\text{Γραμμές cache}}{\text{Πλήθος δρόμων}}$

tag	set	word
9	2	5

Tag	Data	Μεθοδολογία:
0 00110001	set 0	23AB
1		tag set word
2 00100011	32 byte set 1	00100011 001 01011 = 2 tag
3 00100010		Η cache είναι κενή
4 01010110	set 2	Η εισερχόμενη διεύθυνση δεν υπάρχει έχουμε MISS
5 010101010		Τοποθετούμε το μπλοκ στο οποίο ανήκει
6 11111111	set 3	η διεύθυνση στην γραμμή 8 του set 1
7		Σημειώθηκε το byte 11 01001 01001

4B4F

0100 10110 10 01111 tag 4

2121

001000010 01 00001 tag 3

FFFF

11111111 11 11111 tag 6

5555

010101010 10 10101 tag 5

3894

001110001 00 10100 tag 0

Σημειώσεις Δεκαεξαδικό

Δεκαδικό	Δυαδικό	Δεκαεξαδικό
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Μεθοδολογία:

- 1) Η μετατροπή θα είναι από το δεκαεξαδικό στο δυαδικό
- 2) Κάθε ψηφίο/γράμμα ισούται με μια τετραδάδα αριθμών
- 3) Ξεκινάω από τα αριστερά και μετατρέπω το πρώτο γράμμα σε δυαδικό αριθμό
- 4) Είναι συνέχεια συντάζω και στη συνέχεια

Παράδειγμα 23AB

↓ ↓ ↓ ↓
0010 0011 1010 1011 άρα 23AB = 0010 0011 1010 1011

Step by step παράδειγμα. Δεκαεξαδικός ACAB

1. ACAB A = 1010

2. ACAB C = 1100

3. ACAB A = 1010

4. ACAB B = 1011

→ 1010 1100 1010 1011