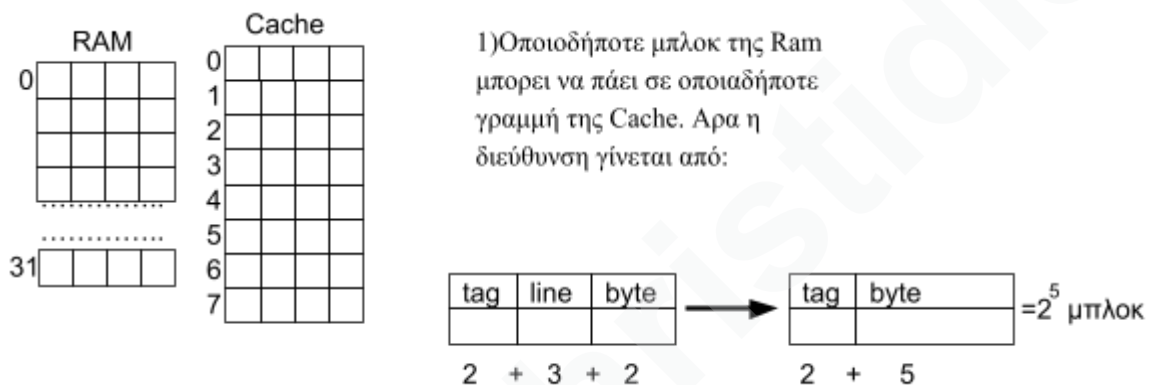


Αρχιτεκτονική Διάλεξη 12

Πλήρης συσχέτιση:

Δίνεται μια κρυφή μνήμη πλήρους συσχέτισης, με RAM 128 bytes, μέγεθος μπλοκ 4 bytes. Η κρυφή μνήμη έχει μέγεθος 32 bytes.

- 1) Ανάλυση διεύθυνσης
- 2) Νά εξηγήσετε τα σήματα του Tag



Ποιό είναι το μέγεθός του Tag Directory ?

Είναι το (πλήθος γραμμών της cache) x (μέγεθος Tag)

$$8 \times 2 = 16 \text{ bit}$$

RAM					Cache				
0	A	B	C	D	0	A	B	C	D
1	1	2	3	4	1	1	2	3	4
2	K	Λ	M	N	2				
3	Z	5	7	12	3				
4	6	10	5	K	4				
5	5	T	5	T	5				
6	A	B	D	B	6				
7	1	1	1	1	7	1	1	1	1
8	10	12	11	13					
9	X	X	X	X					
10	9	8	5	9					
...									
31									

Σε μία χρονική στιγμή, τα Block της RAM 0-7 έχουν φορτωθεί αντίστοιχα στις γραμμές 0-7 της Cache. Για τις διευθύνσεις: 32,33,60,80,81 και 127. Νά εξετάσετε αν υπάρχει HIT η MISS . Η cache χρησιμοποιεί πολιτική FIFO για αντικατάσταση γραμμής

Η πρώτη γραμμή της
Cache

περιέχει τα byte 0-3

00000	00
	01
	10
	11

Tag Directory(16 bit αρα 10000)

0-3	00000
4-7	00001
8-11	00010
12-15	00011
16-19	00100
20-23	00101
24-27	00110
28-31	00111

0	A	B	C	D
1	1	2	3	4
2				
3				
4				
5				
6				
7	1	1	1	1

Η δεύτερη γραμμή της Cache

περιέχει τα byte 4-7

00001	00
00001	01
00001	10
00001	11

32=01000 00 το TAG της ζητούμενης διεύθυνσης συγκρίνεται **TAYTOXPONA** με όλα τα TAG του tag directory.

1) Αν βρεθεί έχουμε HIT

2) Αν όχι έχουμε MISS

ολόκληρο το Block μεταφέρεται στην cache και αν αυτή είναι γεμάτη εφαρμόζεται η πολιτική αντικατάστασης

1) Πλήρη συσχέτιση : τόσοι συγκριτές οσες οι γραμμές της Cache, δηλαδή 8

2) Επειδή $32/4=8$, η ζητούμενη διεύθυνση βρίσκεται στο Block 8

3) το Tag 01000 δεν υπάρχει στο Tag Directory. Αρα Miss

Βγάζουμε τι γραμμή που μπήκε πρώτη και τοποθετούμε το νέο Block:



33=01000 01 αρα είναι hit αφου το tag directory πλέον εχει το 01000 μεσα

60= 0111100 Miss αφου το Tag δεν υπάρχει στα byte
byte 60 και byte/4 = BLOCK αρα 60/4 = 15 tag directory



80= 1010000 Miss αφου το Tag δεν υπάρχει στα byte
byte 80 και byte/4 = BLOCK αρα 80/4 = 20 block



127= 1111111 Miss αφου το Tag δεν υπάρχει στα byte
byte 127 και byte/4 = BLOCK αρα 127/4 = 31 block

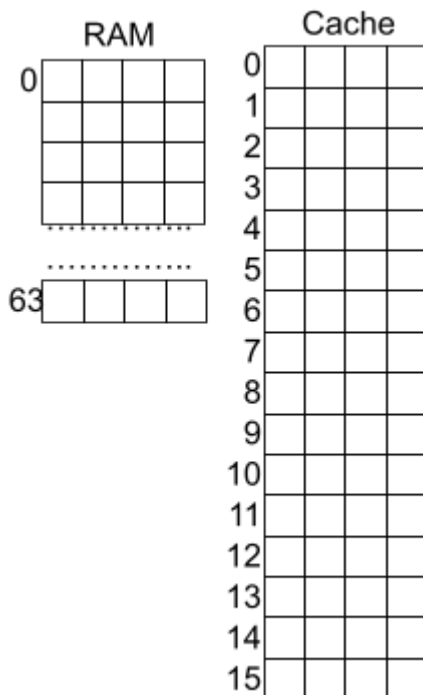


1) Το πλήθος συγκριτών στην άμεση συσχέτιση είναι 1 και στην πλήρη M, οπού M το πλήθος γραμμών της cache

2) Στην πλήρη συσχέτιση δεν υπάρχει LINE

Συσχέτιση συνόλων RAM 256 byte, 4 byte / block, CACHE 64 byte,
οργανωμένη σε συνολα τεσσάρων δρόμων (4 way)

- 1) Ανάλυση διεύθυνσης
- 2) Εξήγηση του tag(τι δείχνει)
- 3) Να δώσετε όλες τις διευθύνσεις που αντιστοιχίζονται στο σύνολο 2



Σύνολο: Ένα πλήθος γραμμών της Cache
 K-WAY: Κάθε σύνολο περιέχει K Γραμμές
 Κάθε σύνολο έχει δηλαδή 4 γραμμές

Έχουμε 16 γραμμές / 4 δρόμους = 4 σύνολα

tag	set	byte
-----	-----	------

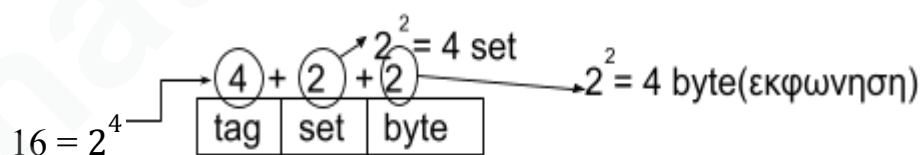
Για 256 byte Ram είναι 8 γιατί $2^8=256$

4	+	2	+	2
tag		set		byte

256 byte RAM : $2^8 \rightarrow 8$ bit διευθυνσιοδότηση

256/4 byte ανα Block δηλαδή $256/4=64$ Block

Έχουμε 64 block και έχουμε 4 σύνολα. Αν διαιρέσουμε $64/4=16$ σε κάθε σύνολο αντιστοιχίζονται 16 Block



- **Άμεση αντιστοίχιση:** Κάθε Block αντιστοιχεί σε 1 γραμμή
- **Αντιστοίχιση Συνολων:** Κάθε Block αντιστοιχίζεται σε A σύνολο.
 Αρα αντιστοιχίζεται σε K Γραμμές όπου K οι δρόμοι
- **Πλήρη Αντιστοίχιση:** αντιστοίχιση σε όλες

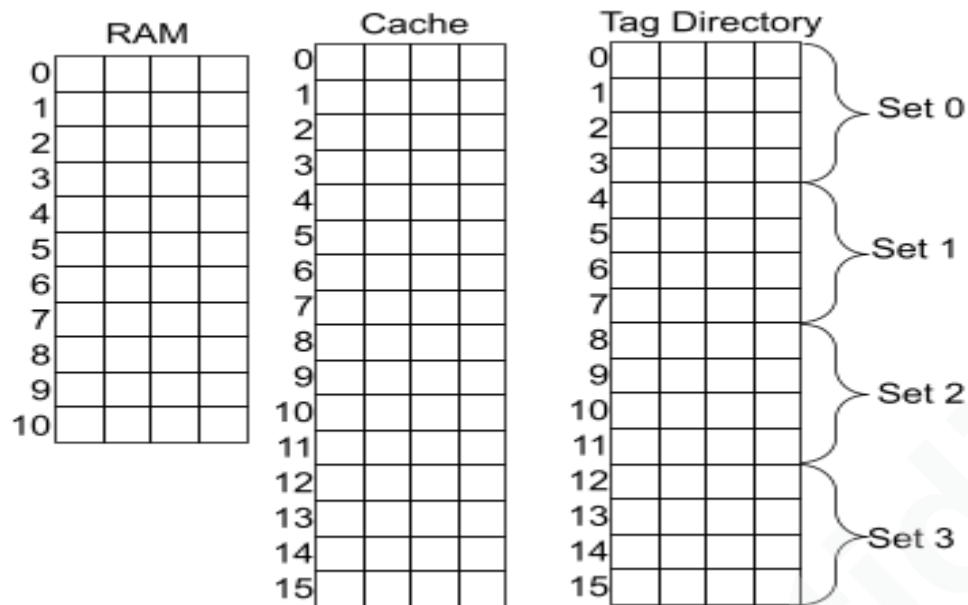
Μία cache με K γραμμές όπου $K=16$

Μια υλοποίηση σε M σύνολα, L δρόμων

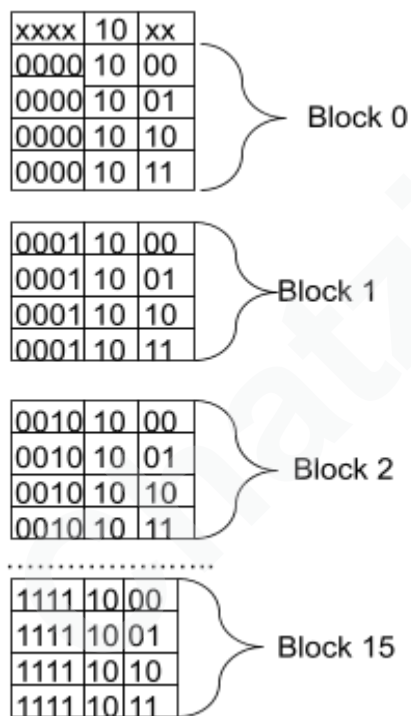
$K=M \times L$ αρα $M=4$, $L=4$

Αμεσή Συνολων Πλήρη

1 < L < K



Αρχικά η Cache είναι αδεια.



Άσκηση

Ζητούνται η διευθύνσεις:

0,1,2,3,4,16,17,18,19,32,33,34,35,48,49,50,51,64,65,66,67

-0=0000 00 00

Διαβάζουμε οτι set=0

Αυτό σημαίνει ότι θα αναζητήσουμε το tag 0000 στις 4 γραμμές του set 0 (ταυτόχρονα)

4 συγκριτές (οσο είναι το K)

(γενικά θελω συγκριτές= δρόμοι)

1=0000 00 01

set=0 πάμε στο set και ψάχνουμε το tag 0000 → *HIT*

2,3: είναι hit γιατί στις 2,3 το tag=0000 → *HIT*

16: 0001 00 00 16/4= 4 γραμμή

set=0 Αρα πανε στο set 0 και ψάχνουμε το tag 0001

ταυτόχρονα σε σχέση τις γραμμής set → *MISS*

17: 0001 00 01 17/4= γραμμή 4 → *HIT*

Cache	Tag Directory								
17 <table><tr><td>A</td><td>B</td><td>C</td><td>D</td></tr></table>	A	B	C	D	<table><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr></table>	0	0	0	1
A	B	C	D						
0	0	0	1						

18,19: → *HIT*

32=0010 00 00 32/4 = 8 γραμμή

Cache	Tag Directory								
32 <table><tr><td>X</td><td>Y</td><td>Z</td><td>B</td></tr></table>	X	Y	Z	B	<table><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr></table>	0	0	1	0
X	Y	Z	B						
0	0	1	0						

48=0011 00 00 48/4 = 12 γραμμή

Cache	Tag Directory								
48 <table><tr><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	<table><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>	0	0	1	1
1	1	1	1						
0	0	1	1						

64=0100 00 00 64/4 = 16 γραμμή

Cache	Tag Directory								
64 <table><tr><td>K</td><td>K</td><td>K</td><td>K</td></tr></table>	K	K	K	K	<table><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr></table>	0	1	0	0
K	K	K	K						
0	1	0	0						

αλλάζει από

Cache	Tag Directory								
0 <table><tr><td>0</td><td>1</td><td>2</td><td>3</td></tr></table>	0	1	2	3	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0
0	1	2	3						
0	0	0	0						

 →

Cache	Tag Directory								
64 <table><tr><td>K</td><td>K</td><td>K</td><td>K</td></tr></table>	K	K	K	K	<table><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr></table>	0	1	0	0
K	K	K	K						
0	1	0	0						