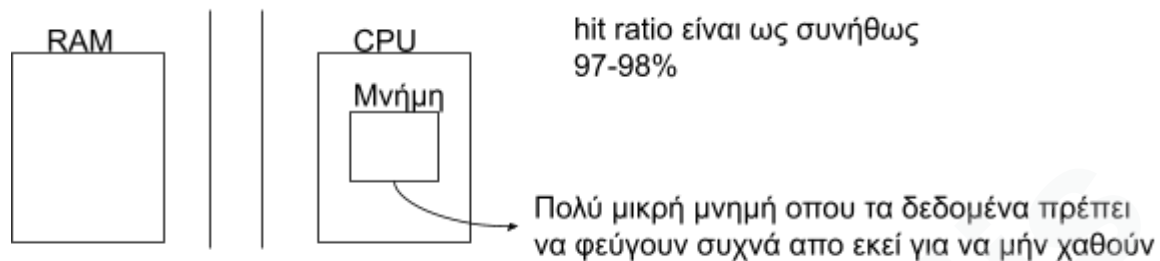
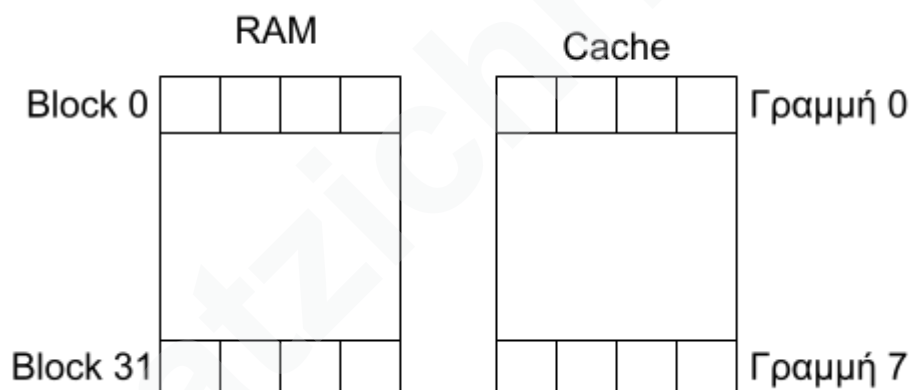


Αρχιτεκτονική Διάλεξη 11

Κρυφή μνήμη



- υπάρχουν τρεις μορφές οργάνωσης
- Το block είναι ουσιαστικά μια γραμμή. Για την Ram την λεμε Block. Για την Cache της λέμε γραμμές.
Δηλαδή γραμμές Ram= Block
γραμμές Cache= γραμμές



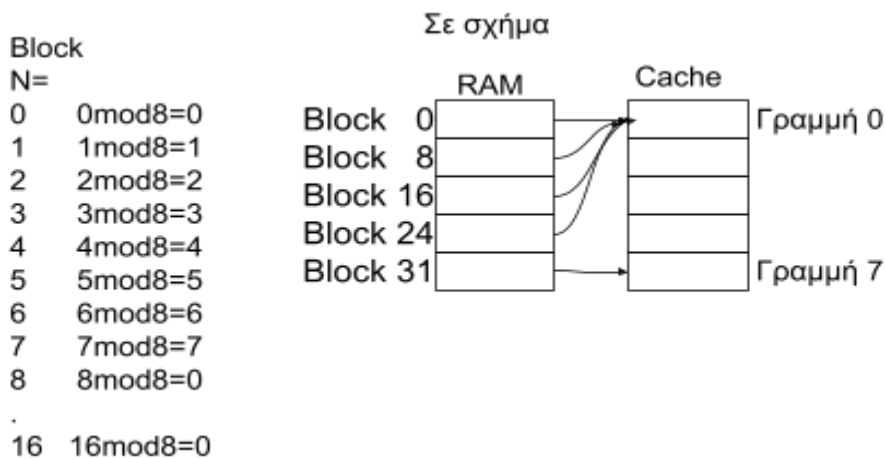
Cache= 8 γραμμές

Δεν μπορούμε να φέρουμε ολη την RAM στην Cache επομένως πρέπει να βρούμε μηχανισμούς αντιστοιχισης block σε γραμμές

Κάθε Block A αντιστοιχίζεται σε μια γραμμή B χρησιμοποιώντας μια πράξη Modulo

Αν οι γραμμές της Cache είναι M τότε κάθε Block N αντιστοιχίζεται στην γραμμή $N \bmod M$

Δηλαδή



Αντιστοιχίες:

Έστω ότι ζητούνται οι διευθύνσεις RAM

5,6,7,8,40,127

5:

0000101

 Άρα η διεύθυνση 5 βρίσκεται στο
tag γραμμή byte
byte 01=1 και στην γραμμή 001=1

Δεύτερος τρόπος: παίρνω byte 5. Διαιρώ με 4 και βρίσκω $5/4=1$ (Πλήθος byte/block)
(κρατάω το ακέραιο κομμάτι) $1 = \text{δείκτης Block της Ram δηλαδή το byte βρίσκεται στο Block 1}$

6:

0000110

 Άρα η διεύθυνση 6 βρίσκεται στο byte 10=2 και στην
tag γραμμή byte γραμμή 001=1
και $6/4 = 1 \text{ block}$ δηλαδή αντιστοιχίζεται στην γραμμή
 $1 \bmod 8 = 1$

7:

0000111

 $7/4 = 1 \rightarrow 1 \bmod 8 = 1$ γραμμή
tag γραμμή byte

8:

0001000

 $8/4 = 2 \rightarrow 2 \bmod 8 = 2$ γραμμή
tag γραμμή byte

40:

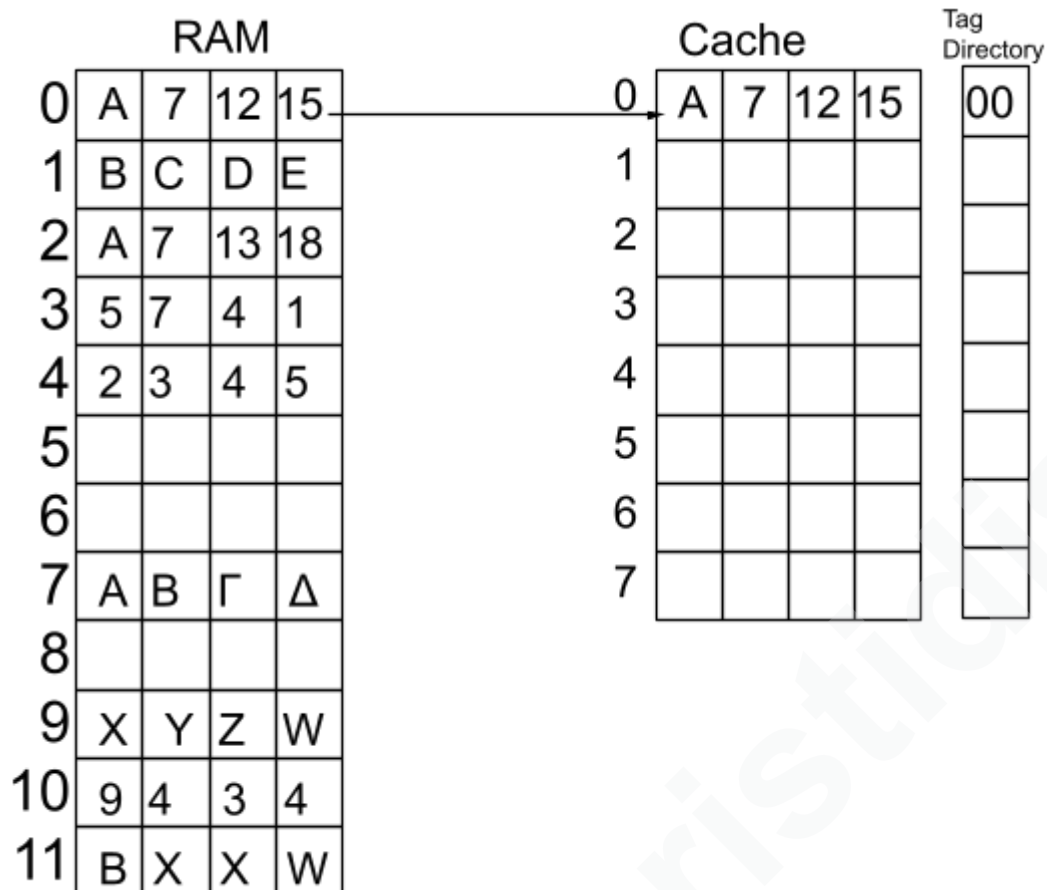
0101000

 $40/4 = 10$ $10 \bmod 8 = 2$ γραμμή
tag γραμμή byte

127:

1111111

 $127/4 = 31$ $31 \bmod 8 = 7$ γραμμή
tag γραμμή byte
3 7 3



Αρχικά η Cache είναι άδεια .

Ένα πρόγραμμα ζητάει διαδοχικά τις διευθύνσεις 0,1,2,3,4,5,6,7,32,33,34,35,36

Να βρείτε το ποσοστό ευστοχίας και αστοχίας και να δείξετε την τελική κατάσταση της Cache. Αρχικά η cache είναι αδεια

Byte : Πόσα byte/block

Line: Πλήθος γραμμών cache

Tag: Οτι περισσεύει $7-5=2$

$M = 32/4 = 4$ block

αντιστοιχίζονται / γραμμή

$tag = \log_2(4) = 2$

Διαιρούμε το πλήθος Block/πλήθος γραμμών για να βρούμε ποσα block

αντιστοιχίζονται σε κάθε γραμμή $M = \text{αριθμος Block} / \text{αριθμός γραμμών}$

$tag = \log_2(M)$

Αν η κάθε ζητούμενη διεύθυνση αναζητείται στην Cache. Αν βρεθεί → hit . Αν

δεν βρεθεί → miss και φέρνουμε ολο το Block της ζητούμενης διεύθυνσης απο την RAM

0 miss: 00 00000

Πως γίνεται ο ελεγχος?

1) Διαβάζεται το πεδίο LINE=000

Επειδή έχουμε άμεση αντιστοιχισή το σύστημα ελεγχει την γραμμή
00000(00-11)→ Block 0 (0-3)

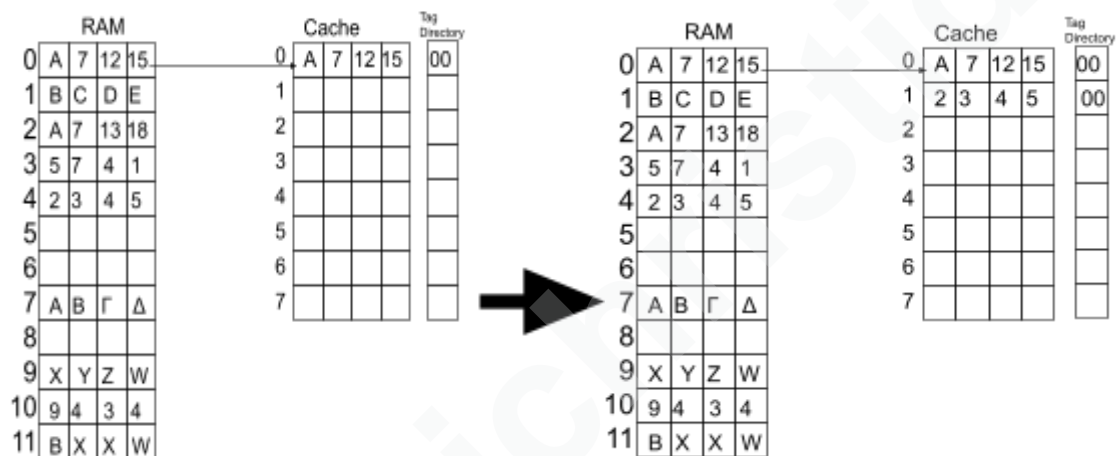
01000(00-11)→ Block 8

10000(00-11)→ Block 16

11000(00-11)→ Block 24

4→ 00 001 00

Αναζητούμε το tag 00 στη γραμμή 001 της Cache αρα Miss αρα μπαίνουν
BCDE στην Cache όπως δείχνει το σχήμα :



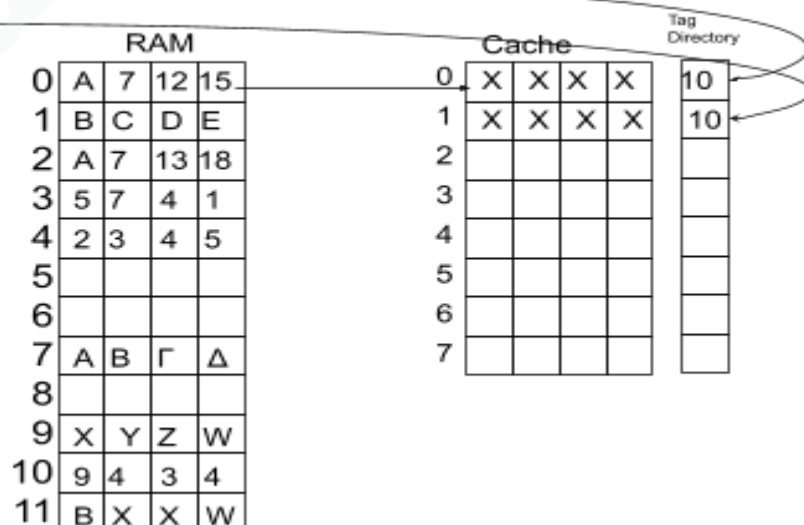
5 → 00 001 01 hit Αφου εχουμε γεμίσει την γραμμή 001 και εχουν ιδιο Tag=00

65→ 10|000|01 miss

70→ 10|001|10 miss

Το 65 κάνει Miss
γιατί είναι στην
γραμμή μηδέν
αλλα είχε
διαφορετικο Tag.
Το αρχικο ηταν
01 και της 65
είναι 10. Οπότε
θα γίνει 10

Στο 70 στην
γραμμη 1 είχαμε
00 και θα γίνει 10



Άσκηση:

Έστω μία μνήμη αποτελείται από λέξεις 1 byte κάθε λέξη .

Το μέγεθος RAM=128 bytes. Το μέγεθος block=4 bytes.

Η cache έχει μέγεθος 32 bytes και μέγεθος γραμμής 4 bytes.

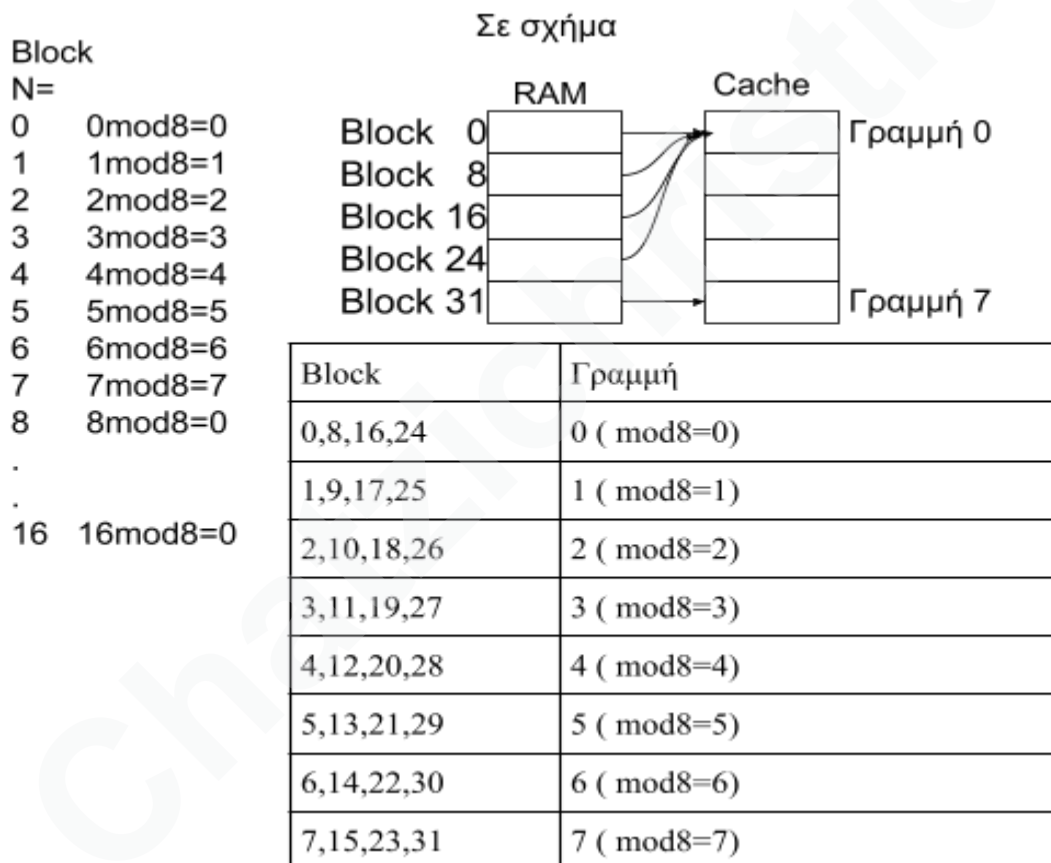
- 1) Να αναλύσετε τη διεύθυνση της RAM
- 2) Να δώσετε τις διευθύνσεις των λέξεων που σχετίζονται με την γραμμή 2 της Cache αν έχουμε τεχνική άμεσης συσχέτισης

Λύση κομμάτι 1 :

Μέγεθος διεύθυνσης καθορίζεται από την RAM 128 bytes= $2^7=7$ bit

Τύπος: $2^N=N$ bit

128 bytes Ram/4 block= 32 block αρα για να μπουν τα 32 block της RAM στις 8 γραμμές της Cache θέλουμε $32/8=4$ tag αρα το Tag έχει 2 δυαδικά ψηφία



Λύση κομμάτι 2 :

Για να πάει μια διεύθυνση στη γραμμή 2, πρέπει το πεδίο γραμμής να δείχνει 010

tag	Line	byte
xx	010	xx
4		4

	Byte	
00 010 00 →	8	Block 00 που αντιστοιχίζεται στη γραμμή 2
00 010 01 →	9	
00 010 10 →	10	
00 010 11 →	11	
01 010 00 →	40	Block 01 που αντιστοιχίζεται στη γραμμή 2
01 010 01 →	41	
01 010 10 →	42	
01 010 11 →	43	
10 010 00 →	72	Block 10 που αντιστοιχίζεται στη γραμμή 2
10 010 01 →	73	
10 010 10 →	74	
01 010 11 →	75	
11 010 00 →	104	Block 11 που αντιστοιχίζεται στη γραμμή 2
11 010 01 →	105	
11 010 10 →	106	
11 010 11 →	107	