

# Κρυφές Μνήμες

(οργάνωση, λειτουργία και απόδοση)

<http://mixstef.github.io/courses/comparch/>

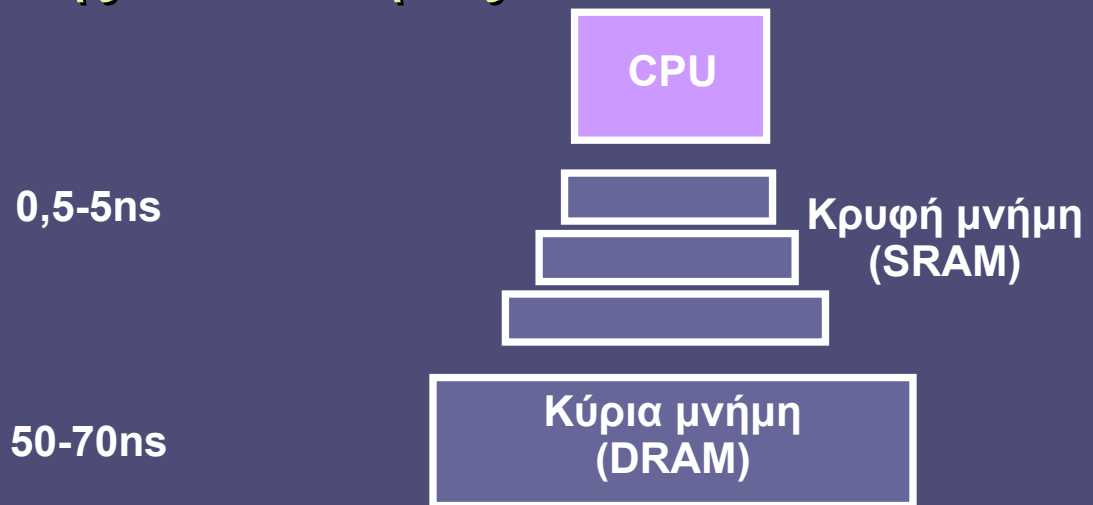
Μ.Στεφανιδάκης



# Σκοπός της Ιεραρχίας Μνήμης

- Ιεραρχία Μνήμης

- Προσέγγιση της ιδανικής μνήμης
  - Ο επεξεργαστής να βλέπει “μνήμη”
  - Με την ταχύτητα του υψηλότερου επιπέδου
  - Και το μέγεθος του χαμηλότερου επιπέδου
- Η ιεραρχία μνήμης εκμεταλλεύεται την αρχή της τοπικότητας



# Ιεραρχία μνήμης και τοπικότητα

- Ιεραρχία Μνήμης

«ένα πρόγραμμα εκτελεί το 90% των εντολών του μέσα στο 10% του κώδικά του»

- **Χρονική Τοπικότητα**
  - Εάν προσπελαστεί μια θέση μνήμης, είναι πολύ πιθανό να προσπελαστεί ξανά στο άμεσο μέλλον
    - Παράδειγμα: οι εντολές ενός βρόχου (loop)
- **Εφαρμογή:**
  - Δεδομένα και εντολές που χρησιμοποιήθηκαν πρόσφατα βρίσκονται ήδη κοντύτερα στον επεξεργαστή (π.χ. στην κρυφή μνήμη)
    - θα προσπελαστούν πολύ γρηγορότερα την επόμενη φορά

# Ιεραρχία μνήμης και τοπικότητα

- Ιεραρχία Μνήμης

- Χωρική Τοπικότητα

- Εάν προσπελαστεί μια θέση μνήμης, είναι πολύ πιθανό να προσπελαστούν και οι γειτονικές θέσεις στο άμεσο μέλλον
  - Εντολές προγραμμάτων, δεδομένα σε πίνακες κλπ

- Εφαρμογή:

- Όταν προσπελαστεί μια θέση μνήμης, μεταφέρονται και οι διπλανές της λέξεις στην κρυφή μνήμη του υψηλότερου επιπέδου
  - Γρηγορότερη προσπέλαση όταν ζητηθούν και αυτές

# Κρυφές μνήμες

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

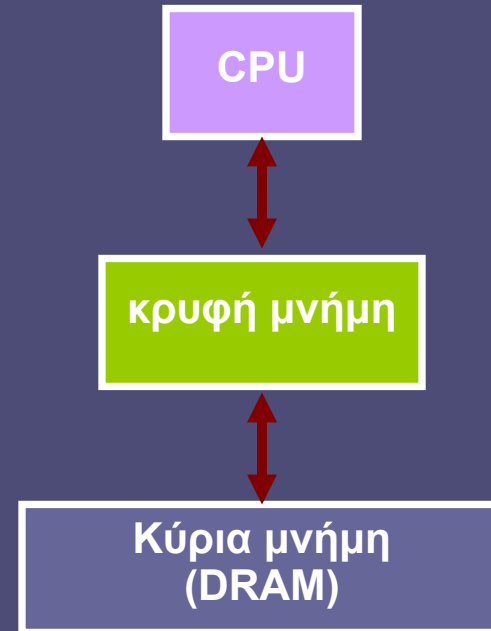
- Σημαντικό τμήμα στην ιεραρχία μνήμης
- Εξέλιξη συστημάτων κρυφής μνήμης
  - 1962: οι πρώτες ιεραρχίες μνήμης (Atlas computer)
    - Όχι όμως κρυφή μνήμη
  - 1965: η πρώτη περιγραφή κρυφής μνήμης (Wilkes)
    - Ο πρώτος υπολογιστής με κρυφή μνήμη (IBM 360/85)
  - 1968: η πρώτη χρησιμοποίηση του όρου “**cache memory**”
  - Στη συνέχεια:
    - Πολλαπλά επίπεδα κρυφής μνήμης (L1, L2, L3...)
    - Βελτιωμένες αρχιτεκτονικές κρυφής μνήμης

# Απλό μοντέλο ιεραρχίας μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

Οι αρχές λειτουργίας της απλής ιεραρχίας μπορούν να επεκταθούν σε πολλαπλά επίπεδα (κρυφή μνήμη L1, L2, L3...)

Η διαχείριση της κρυφής μνήμης γίνεται «αυτόματα» από το υλικό



- Τα δεδομένα βρίσκονται αρχικά στην κύρια μνήμη
- Η κρυφή μνήμη περιέχει **υποσύνολο** των δεδομένων
- Μεταφορά μεταξύ επιπέδων μνήμης σε **μπλοκ** λέξεων

# Αποθήκευση δεδομένων στην Ιεραρχία Μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

- **Αποθήκευση δεδομένων**
  - Τα υψηλότερα επίπεδα της ιεραρχίας μνήμης (πιο κοντά στις ΚΜΕ) είναι **υποσύνολα** των χαμηλότερων
  - Όλα τα δεδομένα αποθηκεύονται τελικά στο χαμηλότερο επίπεδο (κύρια μνήμη)
- **Μεταφορά δεδομένων**
  - Αντιγραφή από επίπεδο σε επίπεδο
  - Το ελάχιστο σύνολο δεδομένων που μεταφέρεται μεταξύ δύο επιπέδων ονομάζεται **μπλοκ**
    - Πολλαπλά bytes (πολλές λέξεις μαζί)

# Αναζήτηση δεδομένων στην Ιεραρχία Μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

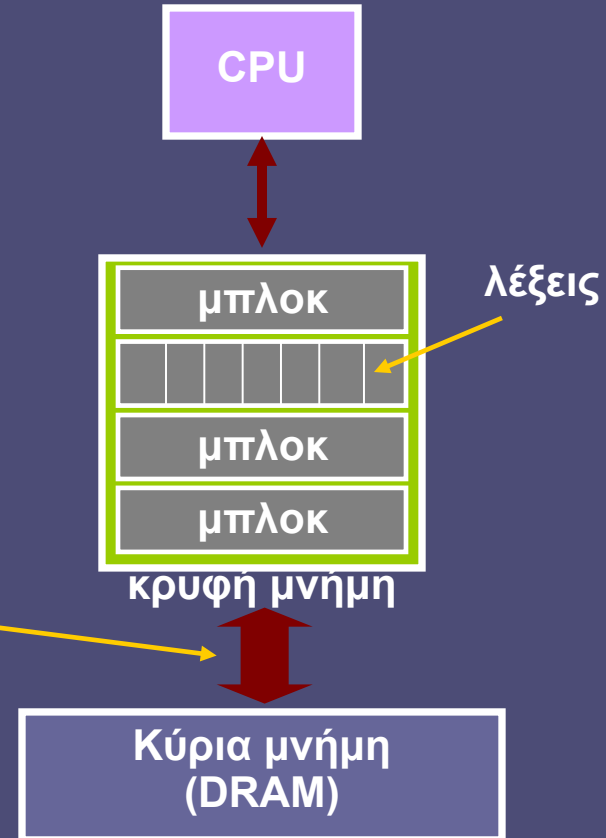
- **Αναζήτηση δεδομένων**
  - Ο επεξεργαστής ζητά **πάντοτε** τα δεδομένα/εντολές από το κοντινότερο σε αυτόν επίπεδο
  - Τα δεδομένα υπάρχουν στο επίπεδο αυτό: **hit**
  - Τα δεδομένα δεν βρίσκονται στο επίπεδο αυτό: **miss**
    - Η αίτηση προωθείται στο επόμενο (χαμηλότερο) επίπεδο
    - Όταν βρεθεί, το **μπλοκ** που περιέχει τα δεδομένα **αντιγράφεται** στο ανώτερο επίπεδο



# Μπλοκ (γραμμές) κρυφής μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

- Για την εκμετάλλευση της χωρικής τοπικότητας
- Όταν πρέπει να μεταφερθεί μια λέξη, μεταφέρεται το μπλοκ που την περιέχει
- Το σύστημα κύριας μνήμης έχει βελτιστοποιηθεί αρχιτεκτονικά για μεταφορές μπλοκ



Οι μοντέρνοι επεξεργαστές διαθέτουν κρυφές μνήμες με μέγεθος μπλοκ ίσο με 64 bytes

# Τοποθέτηση ενός μπλοκ

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

- Η κύρια μνήμη περιέχει πολύ περισσότερα «μπλοκ» από όσα χωρούν στην κρυφή μνήμη
  - Συνεπώς, **στην ίδια θέση** της κρυφής μνήμης πρέπει να τοποθετηθούν **διαφορετικά** μπλοκ (προφανώς όχι ταυτόχρονα!)
    - Σύγκρουση μπλοκ
- Πώς αποφασίζεται η θέση ενός μπλοκ στην κρυφή μνήμη;
  - Η απλή λύση: **άμεση απεικόνιση** (direct mapped caches)
  - Κάθε μπλοκ πηγαίνει σε μία μόνο θέση  
**(αριθμός μπλοκ) mod (θέσεις στην κρυφή μνήμη)**
    - Υπολογίζεται πολύ εύκολα αν οι θέσεις είναι δύναμη του 2

# Μέρη διεύθυνσης στην άμεση απεικόνιση

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

- Η μονάδα επεξεργασίας κάνει αιτήσεις ανάγνωσης/εγγραφής από/σε **διεύθυνση μνήμης**
  - Μέσα σε ένα μπλοκ από bytes



- Με τη μέθοδο της άμεσης απεικόνισης η διεύθυνση χωρίζεται σε 3 μέρη
  - **byte offset**: σε ποιο byte μέσα στο μπλοκ αρχίζει η ζητούμενη λέξη
    - Για μπλοκ με  $c$  bytes, το byte offset είναι  $\log_2(c)$  bits
  - **index**: σε ποια θέση της κρυφής μνήμης θα πάει το μπλοκ
    - Σε κρυφή μνήμη με  $k$  θέσεις, index είναι  $\log_2(k)$  bits

Τη χρησιμότητα του **tag** (ετικέτας) θα δούμε σε λίγο

# Άμεση απεικόνιση θέσης μπλοκ

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

tag                      byte   offset  
 0x226 = 001 00010 0110  
 θέση μπλοκ (index) = 2



# κρυφή μνήμη

**0x7E9 = 011 11110 1001**

**0x821 = 100 00010 0001**

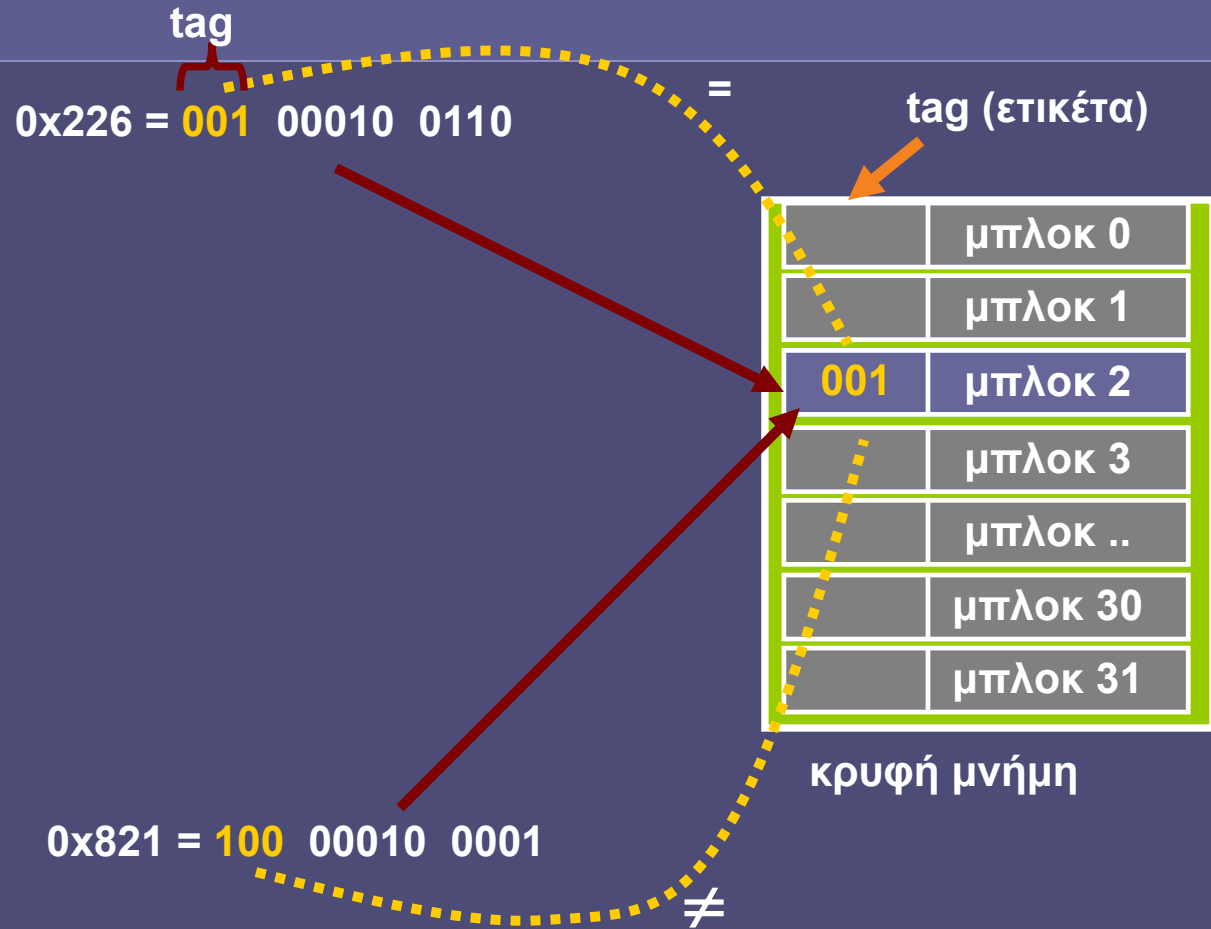
Κάθε μπλοκ μπορεί να βρίσκεται σε μία μόνο θέση στην κρυφή μνήμη

## Παράδειγμα:

Block = 16 bytes  
Cache = 32 blocks

# Ποιο μπλοκ βρίσκεται τώρα σε κάθε θέση;

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη



Σύγκριση με **ετικέτα (tag)** της επιλεγμένης θέσης στην κρυφή μνήμη

Επίσης: είναι η θέση κατειλημμένη από κάποιο μπλοκ;  
**valid bit (V)**

# Ανάγνωση: Cache Hit

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

Σε περίπτωση εύρεσης των δεδομένων στην κρυφή μνήμη, η ΚΜΕ μπορεί να τα λάβει ακόμα και σε 1 κύκλο ρολογιού

read 0x226 = 001 00010 0110

hit



Κύρια μνήμη  
(DRAM)

•Αιτήσεις για ανάγνωση: εντολές και δεδομένα

# Ανάγνωση: Cache Miss

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

read 0x821 = 100 00010 0001

CPU

miss

	μπλοκ 0
	μπλοκ 1
001	μπλοκ 2
	μπλοκ 3
	μπλοκ ..

Κύρια μνήμη  
(DRAM)

• Αιτήσεις για ανάγνωση: εντολές και δεδομένα

## Miss penalty:

ο χρόνος για την μεταφορά του μπλοκ από κύρια μνήμη και επιστροφή δεδομένων στον επεξεργαστή

# Ανάγνωση: Cache Miss

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

## Miss penalty:

ο χρόνος για την μεταφορά του μπλοκ από κύρια μνήμη και επιστροφή δεδομένων στον επεξεργαστή

read 0x821 = 100 00010 0001

CPU

miss

	μπλοκ 0
	μπλοκ 1
001	μπλοκ 2
	μπλοκ 3
	μπλοκ ..

read 0x820...0x82F

Κύρια μνήμη  
(DRAM)

• Αιτήσεις για ανάγνωση: εντολές και δεδομένα



# Ανάγνωση: Cache Miss

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

## Miss penalty:

ο χρόνος για την μεταφορά του μπλοκ από κύρια μνήμη και επιστροφή δεδομένων στον επεξεργαστή

• Αιτήσεις για ανάγνωση: εντολές και δεδομένα

read 0x821 = 100 00010 0001

miss



# Ανάγνωση: Cache Miss

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

## Miss penalty:

ο χρόνος για την μεταφορά του μπλοκ από κύρια μνήμη και επιστροφή δεδομένων στον επεξεργαστή

read 0x821 = 100 00010 0001

miss

read 0x820...0x82F

CPU

μπλοκ 0

μπλοκ 1

100

μπλοκ 3

μπλοκ ..

Κύρια μνήμη  
(DRAM)

• Αιτήσεις για ανάγνωση: εντολές και δεδομένα

# Εγγραφή στην κρυφή μνήμη

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

## Συνοχή δεδομένων:

Πώς επηρεάζουν οι κρυφές μνήμες τη σχεδίαση πολυεπεξεργαστικών συστημάτων;

- Μόνο για δεδομένα
- Write Hit – Ενημέρωση κρυφής μνήμης
  - Η νέα τιμή βρίσκεται **μόνο** στην κρυφή μνήμη
  - Η τιμή στην κύρια μνήμη (ή γενικότερα, στο χαμηλότερο επίπεδο) ενημερώνεται **όταν το μπλοκ εκτοπίζεται από την κρυφή μνήμη (victim)**
    - Απαιτείται επιπλέον λογική (hardware) για τον έλεγχο της **συνοχής** των δεδομένων
    - Όλοι οι πυρήνες πρέπει να βλέπουν τα ίδια δεδομένα
- Write Miss
  - Πρέπει το μπλοκ να έρθει (**ανάγνωση!**) πρώτα στην κρυφή μνήμη από την κύρια μνήμη

# Τι δημιουργεί cache misses;

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- Η πρώτη φορά προσπέλασης ενός μπλοκ
  - Μπλοκ που δεν βρέθηκαν **ποτέ μέχρι τώρα** στην κρυφή μνήμη
- Λόγω χωρητικότητας της κρυφής μνήμης
  - Η κρυφή μνήμη **δεν χωράει** όλα τα μπλοκ (ταυτόχρονα)
  - Μπλοκ που τοποθετούνται στην **ίδια θέση** στην κρυφή μνήμη, συναγωνίζονται για τη θέση αυτή
    - ακόμα κι αν άλλο μέρος της κρυφής μνήμης είναι ελεύθερο...

# Χαρακτηριστικά απόδοσης κρυφής μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- **Hit Rate**

- Ποσοστό προσπελάσεων μνήμης, όπου τα δεδομένα βρίσκονται στην κρυφή μνήμη

- **Miss Rate**

- Ποσοστό προσπελάσεων μνήμης, όπου τα δεδομένα δεν βρίσκονται στην κρυφή μνήμη
    - (1-hit rate)

- **Hit Time**

- Ο χρόνος για την προσπέλαση δεδομένων σε hit

- **Miss Penalty**

- Ο χρόνος για την προσπέλαση, μεταφορά και τοποθέτηση των δεδομένων miss από την κύρια στην κρυφή μνήμη και στον επεξεργαστή

# Το κόστος των cache misses

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- **Χαμένοι κύκλοι ρολογιού**
  - Σε αναμονή για προσπέλαση κύριας μνήμης
- **Κύκλοι Αναμονής =**  
***Προσπελάσεις μνήμης \* Miss Rate \* Miss Penalty***
- **Είναι απλουστευμένο μοντέλο γιατί:**
  - Διαφορετικό Miss Rate ανά κατηγορίες εντολών
  - Διαφορετικό Miss Rate για ανάγνωση-εγγραφή
    - Δυσκολεύει τον υπολογισμό ενός ακριβούς miss rate
  - Σύνθετη ανάλυση για **εκτέλεση εκτός σειράς**
    - Ο επεξεργαστής “κρύβει” την καθυστέρηση εκτελώντας κάτι άλλο
    - Δυσκολεύει τον υπολογισμό ενός ακριβούς miss penalty

# Παράδειγμα υπολογισμού

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- Σύστημα έχει ιδανικό  $CPI = 1$ 
  - Όταν έχουμε cache hits
- 40% των εντολών διαβάζουν ή γράφουν δεδομένα από/στη μνήμη
- Miss rate = 2%
- Miss penalty = 20 κύκλοι ρολογιού
  - Πόσες προσπελάσεις μνήμης ανά εντολή;
  - Πόσα misses ανά εντολή;
  - Ποιο το πραγματικό CPI αν λάβουμε υπόψη και τα misses;

# Το κόστος των cache misses

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- **Βελτίωση της απόδοσης**
  - Μείωση του miss rate
  - Μείωση του miss penalty



# Τεχνικές μείωσης miss rate

- Ιεραρχία Μνήμης
  - Κρυφή Μνήμη
  - Απόδοση κρυφής μνήμης
- Αντιμετώπιση αιτιών που προκαλούν misses
  - Αύξηση χωρητικότητας κρυφής μνήμης
    - **Αλλά:** μια μεγάλη κρυφή μνήμη μπορεί να είναι πιο αργή (αύξηση hit time)
  - Αύξηση του μεγέθους του μπλοκ
    - Προσπάθεια εκμετάλλευσης της χωρικής τοπικότητας
    - **Αλλά:** αυξάνει το miss penalty
    - Πιθανόν να αυξάνει τελικά το miss rate, λόγω των **λιγότερων** μπλοκ στην κρυφή μνήμη
  - Ευέλικτες τεχνικές τοποθέτησης των μπλοκ
    - Όστε να παραμένουν περισσότερο στην κρυφή μνήμη

# Το πρόβλημα με την άμεση απεικόνιση

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

```
for (i..) {  
    a[i] = b[i]+c[i]  
}
```

Τι θα γίνει αν οι πίνακες a,b,c, τοποθετηθούν στην ίδια θέση της κρυφής μνήμης;

- Η τοποθέτηση των μπλοκ στις θέσεις της κρυφής μνήμης με τη μέθοδο της άμεσης απεικόνισης
  - Είναι γρήγορη και απαιτεί απλούστερο κύκλωμα
  - Κατάλληλη για τις κρυφές μνήμες **κοντά** στη μονάδα επεξεργασίας (1<sup>ο</sup> επιπέδου, L1)
- Επειδή όμως κάθε μπλοκ τοποθετείται **ανελαστικά** σε μια και μόνο θέση
  - Μπορεί να προκαλέσει αυξημένες συγκρούσεις μπλοκ μέσα στο ίδιο εκτελούμενο πρόγραμμα
  - Με αποτέλεσμα τη συνεχή αντικατάσταση μπλοκ που έτυχε να απεικονιστούν στην ίδια θέση της κρυφής μνήμης
    - Ακόμα κι αν υπάρχουν άλλες θέσεις που δεν χρησιμοποιούνται τη στιγμή εκείνη

# Ευέλικτες τεχνικές τοποθέτησης μπλοκ

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

Πιθανή η αύξηση του hit time λόγω πιο πολύπλοκου κυκλώματος



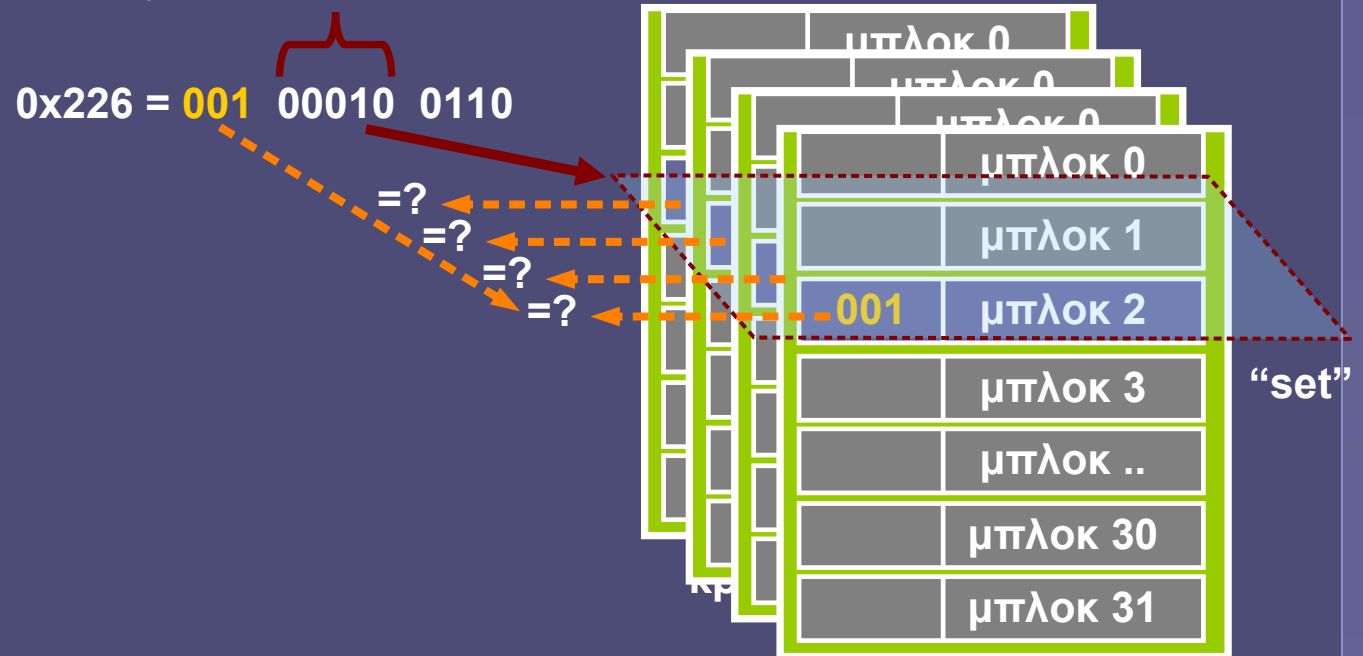
Πιθανή τοποθέτηση  
Παράλληλη αναζήτηση

• Αντικατάσταση: least recently used (LRU)

# Παράδειγμα: 4-way set associativity

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη

προσδιορίζει το set των 4 θέσεων



Η τοποθέτηση γίνεται σε μία από 4 πιθανές θέσεις

# Παράδειγμα οργάνωσης κρυφής μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- Επεξεργαστής έχει 48 bits διεύθυνσης
- 32KB κρυφή μνήμη 1<sup>ου</sup> επιπέδου (L1)
  - 8-way set associative
  - Μέγεθος μπλοκ = 64 bytes
- Πόσες θέσεις για μπλοκ συνολικά;
- Πόσα sets;
- Ποια τα μέρη της διεύθυνσης και το εύρος τους σε bits;
  - Πώς θα ήταν τα παραπάνω μεγέθη αν είχαμε απλή άμεση απεικόνιση;

# Παράδειγμα οργάνωσης κρυφής μνήμης

- Ιεραρχία Μνήμης
  - Κρυφή Μνήμη
  - Απόδοση κρυφής μνήμης
- Επεξεργαστής έχει 32 bits διεύθυνσης
  - 4KB κρυφή μνήμη δεδομένων 1<sup>ου</sup> επιπέδου
    - 64-way set associative
    - Μέγεθος μπλοκ = 16 bytes
  - Πόσες θέσεις για μπλοκ συνολικά;
  - Πόσα sets;
  - Ποια τα μέρη της διεύθυνσης και το εύρος τους σε bits;
    - Πώς θα ήταν τα παραπάνω μεγέθη αν είχαμε απλή άμεση απεικόνιση;

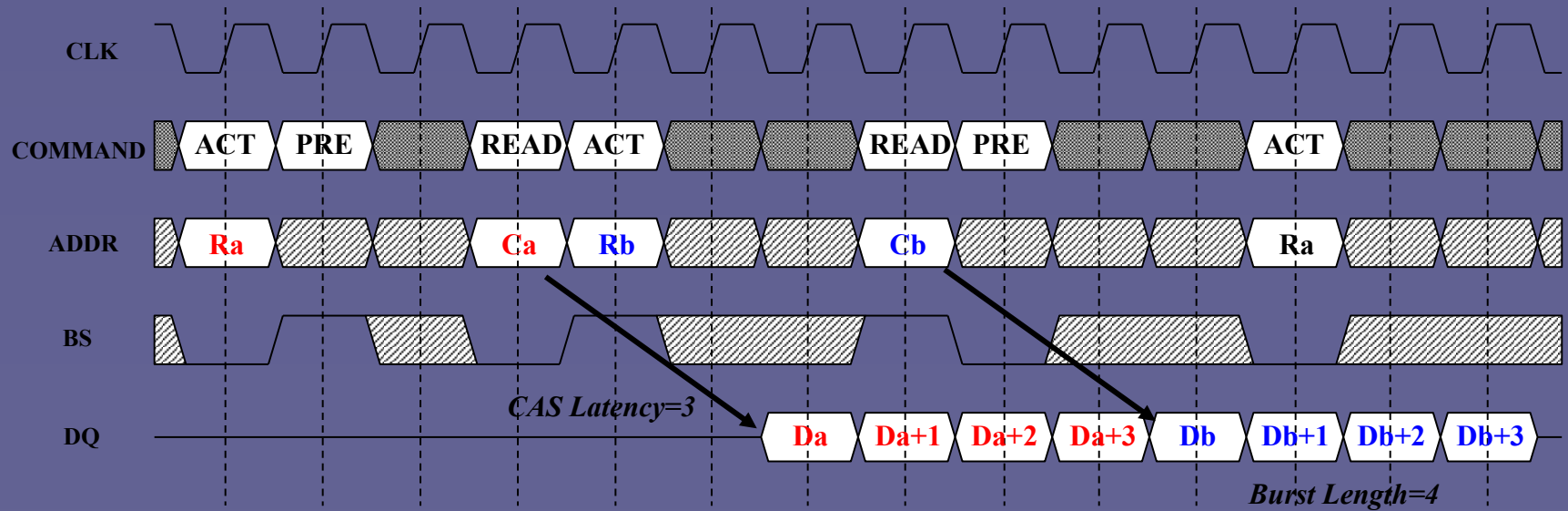
# Τεχνικές μείωσης miss penalty

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

Οι σύγχρονοι επεξεργαστές έχουν L1, L2 και L3 cache μέσα στο ίδιο το chip τους (ίσως και 4ο επίπεδο, ως cache «τελευταίας ευκαιρίας»)

- Μείωση των χρόνων μεταφοράς μπλοκ
- Βελτιστοποιήσεις στην επικοινωνία με την κύρια μνήμη
  - Έτσι ώστε ένα ολόκληρο μπλοκ να μεταφέρεται με τη μικρότερη δυνατή καθυστέρηση (bursts)
- Πολυεπίπεδες ιεραρχίες κρυφής μνήμης
  - Μείωση miss penalty πρώτου επιπέδου (L1)
  - L1: μικρότερο μέγεθος, μεγαλύτερη ταχύτητα
    - Μεγαλύτερο miss rate αλλά miss penalty μικρότερο
  - L2: μεγαλύτερο μέγεθος, μικρότερη ταχύτητα
    - Αργότερη αλλά δεν επηρεάζει hit time επεξεργαστή
  - L3: κοινή για ομάδες πυρήνων

# Ανάγνωση από κύρια μνήμη





# Πολυεπίπεδη οργάνωση κρυφής μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

Τύπος	Μέγεθος	Χρόνος προσπέλασης	Ρυθμός μεταφοράς
L1	έως 64KB	4ns	50GB/s
L2	έως 8MB	10ns	25GB/s
L3	έως 64MB	20ns	10GB/s

Οι σύγχρονοι επεξεργαστές έχουν ξεχωριστή κρυφή μνήμη L1 για εντολές και δεδομένα. Ποια τα πλεονεκτήματα-μειονεκτήματα;

- Παράδειγμα: **Pentium4**
  - L1 cache: 4 κύκλοι ρολογιού (pipelined: 1)
  - L2 cache: 20 κύκλοι ρολογιού
  - Προσπέλαση στη μνήμη: >100 κύκλοι ρολογιού

# Στην εποχή των multicore συστημάτων

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

Τύπος	Μέγεθος	Χρόνος προσπέλασης	Ρυθμός μεταφοράς
L1	32KB 8-way	4-6 cycles	192b/cycle
L2 (MLC)	1MB 16-way	14 cycles	64b/cycle
L3 (LLC)	1.375MB /core	50-70 cycles	32b/cycle

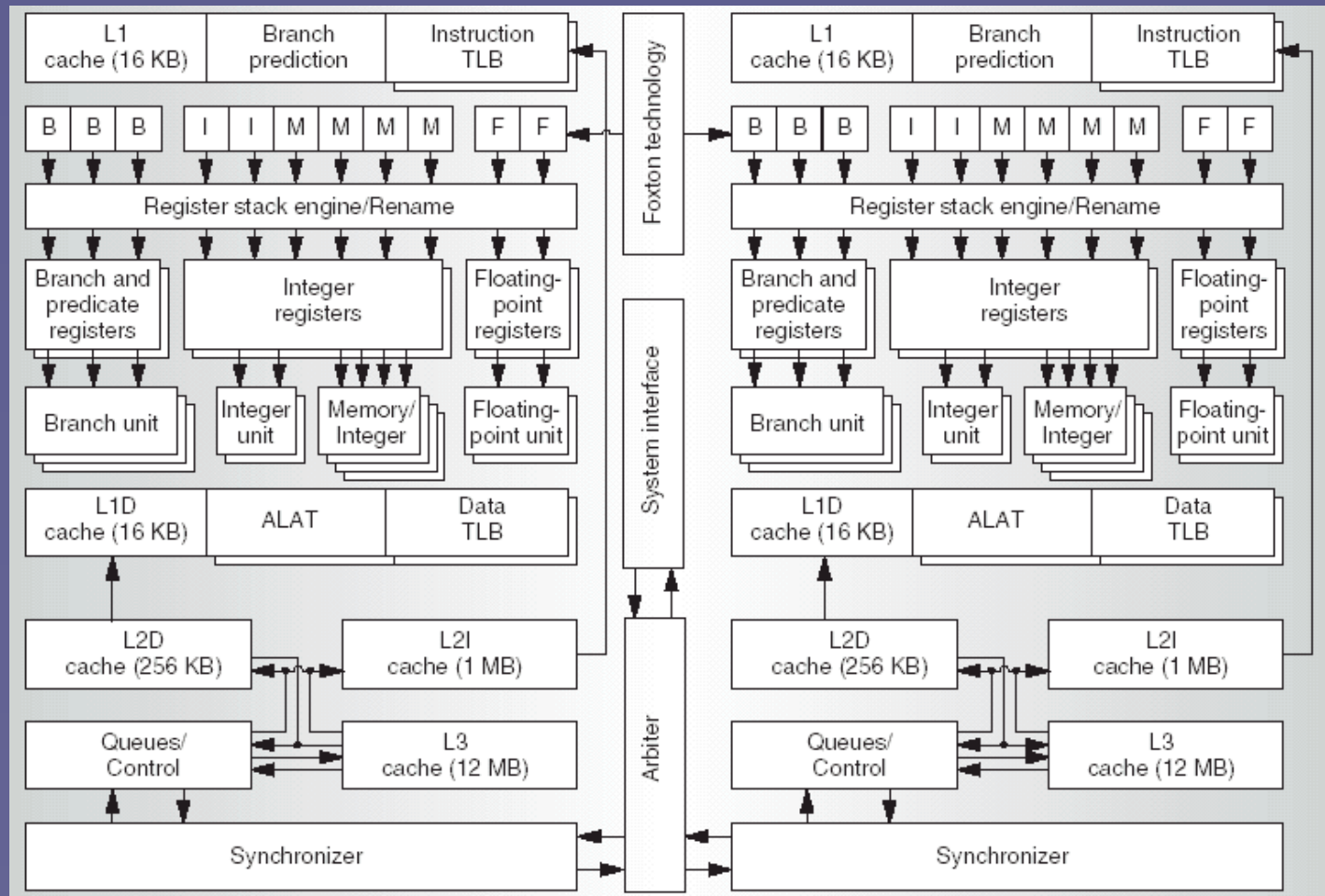
- Παράδειγμα: Intel Xeon Scalable Processors
  - max 28 cores
  - L1 και L2 caches: κάθε πυρήνας έχει τις δικές του
  - L3: κοινή για όλους τους πυρήνες
    - δεν περιέχει υποχρεωτικά ότι υπάρχει σε L1, L2

# Intel “Montecito”: Επίπεδα κρυφής μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

## Intel Montecito:

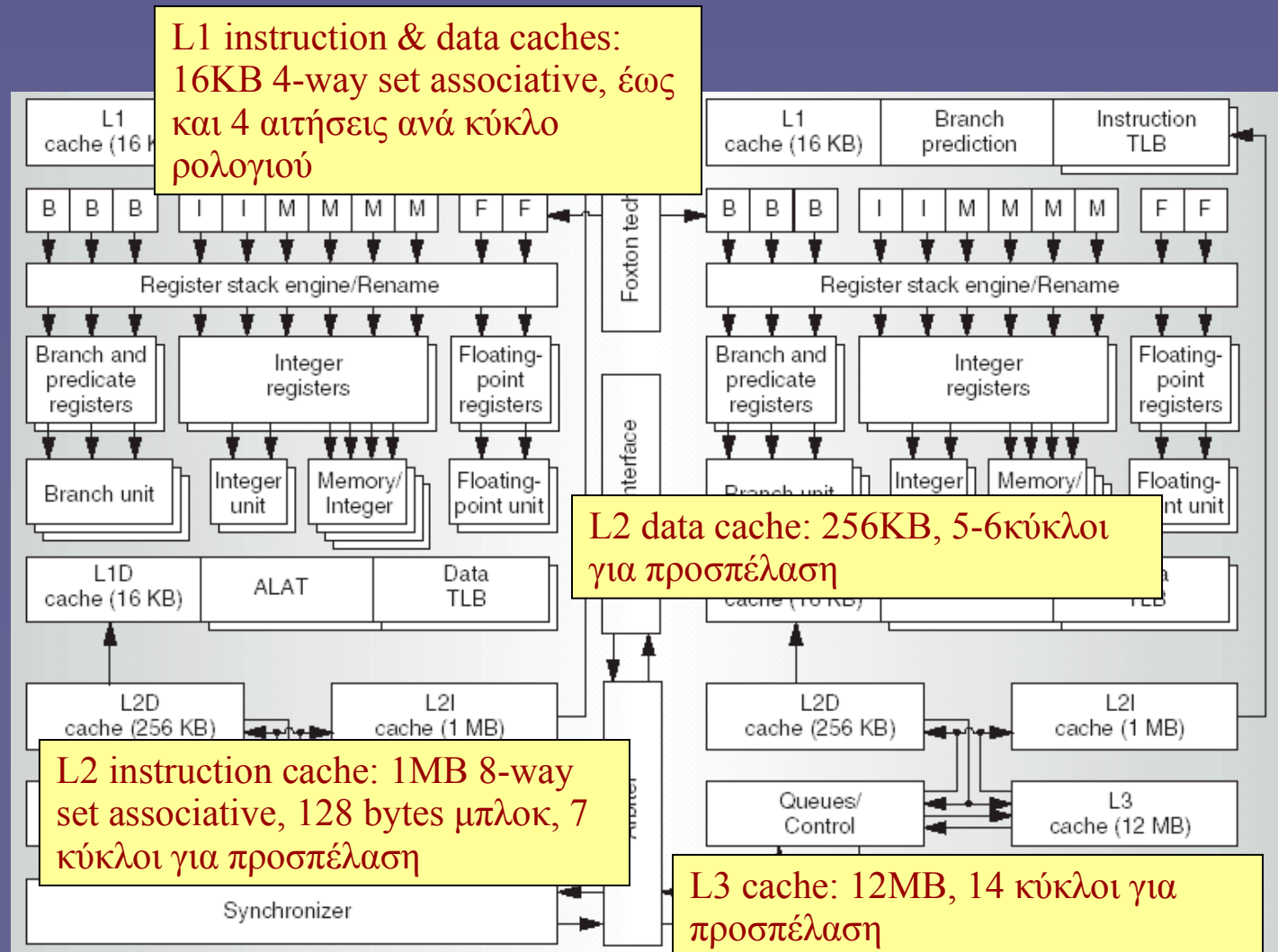
1,72 δις  
τρανζίστορ  
2 επεξεργαστές  
Itanium2  
1.8GHz @ 100W



# Intel “Montecito”: Επίπεδα κρυφής μνήμης

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

**Intel Montecito:**  
συνολικά 27MB  
κρυφή μνήμη μέσα  
στο chip



# Βελτιστοποίηση απόδοσης κρυφής μνήμης

- Ιεραρχία Μνήμης
  - Κρυφή Μνήμη
  - Απόδοση κρυφής μνήμης
- Αρχιτεκτονικές βελτιώσεις
    - Pipelining
    - Non-blocking – εξυπηρέτηση πολλαπλών αιτήσεων
    - Πολλαπλά επίπεδα κρυφής μνήμης στο chip του επεξεργαστή
  - Ο ρόλος του λογισμικού (μεταγλωττιστές)
    - Αναδιοργάνωση προγραμμάτων για αύξηση της τοπικότητας (κυρίως στους βρόχους επανάληψης)
    - Prefetching: μετακίνηση δεδομένων στην κρυφή μνήμη πριν αυτά χρειαστούν στον επεξεργαστή

# Η απόδοση της κρυφής μνήμης συνοπτικά

- Ιεραρχία Μνήμης
- Κρυφή Μνήμη
- Απόδοση κρυφής μνήμης

- Καθοριστική για τα σύγχρονα υπολογιστικά συστήματα
- Μείωση του miss rate ή του miss penalty
  - Όμως: η συμπεριφορά της ιεραρχίας μνήμης επηρεάζεται από πολλούς παράγοντες
- Η πραγματική συμπεριφορά
  - Είναι σύνθετη – απαιτούνται εξομοιώσεις πριν τη σχεδίαση νέων συστημάτων
  - Είναι διαφορετική ανά εφαρμογή – δεν υπάρχει ένα μόνο αντιπροσωπευτικό πρόγραμμα
  - Είναι διαφορετική ανά υπολογιστικό σύστημα – desktop, server ή embedded