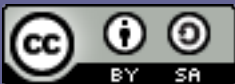


Ιεραρχίες Μνήμης (I)

(τεχνολογίες κύριας μνήμης και εισαγωγή στις κρυφές μνήμες)

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

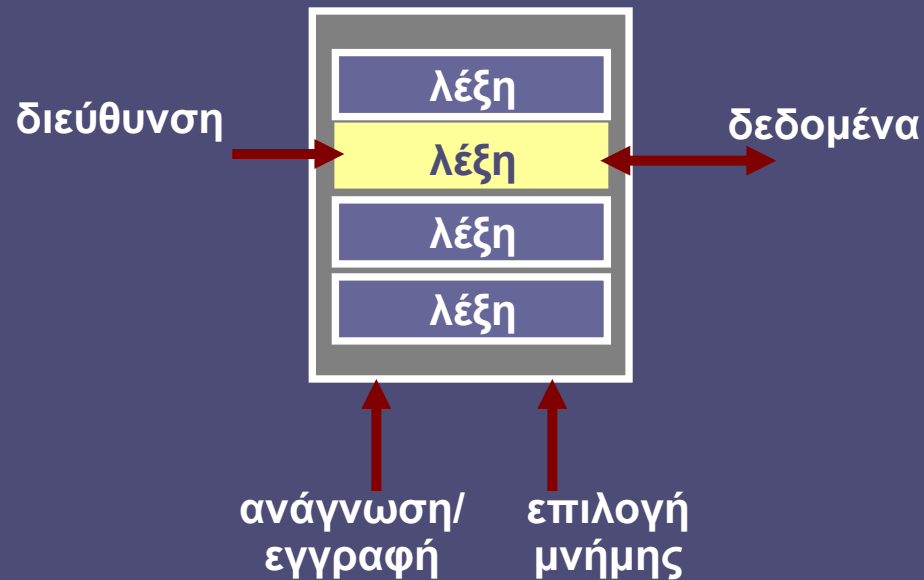
Μ.Στεφανιδάκης



Τεχνολογίες Κύριας Μνήμης

- **Στους πρώτους υπολογιστές**
 - Ιστορικά, η κατασκευή κύριας μνήμης ήταν **πολύ πιο δύσκολη** από την κατασκευή των πρώτων υπολογιστικών κυκλωμάτων
- **Αρχικές τεχνολογίες**
 - Flip-flop με λυχνίες κενού
 - Γραμμές καθυστέρησης υδραργύρου κ.ο.κ
- **Μαγνητικές μνήμες (core memories - 1950)**
 - Η πρώτη αξιόπιστη και σχετικά φθηνή τεχνολογία RAM
 - Κυριάρχησε για 20 περίπου χρόνια
- **Ημιαγωγικές μνήμες (Intel – 1970)**
 - Η αρχή: 1Kbit DRAM (“core killer”)

Το μοντέλο της Μνήμης Τυχαίας Προσπέλασης

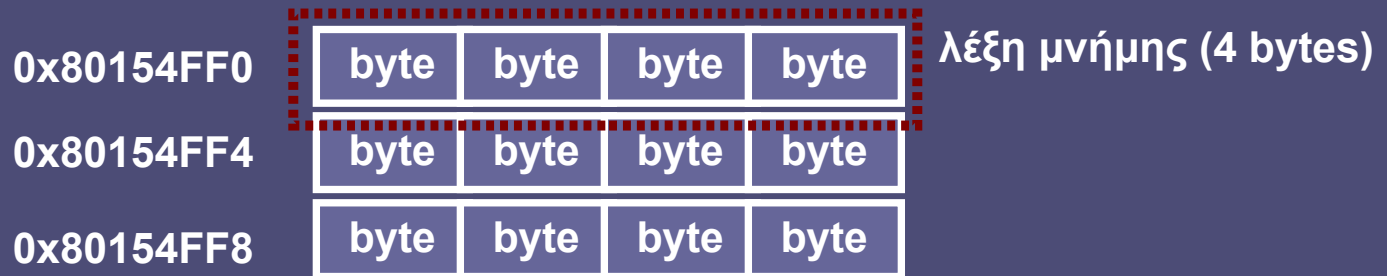


- **Random Access Memory (RAM)**

- Λέξη μνήμης (**word**) με εύρος **M** bits
- Διεύθυνση (**address**) επιλογής λέξης, **N** bits
- Μέγεθος (χωρητικότητα) μνήμης **$2^N \times M$** bits

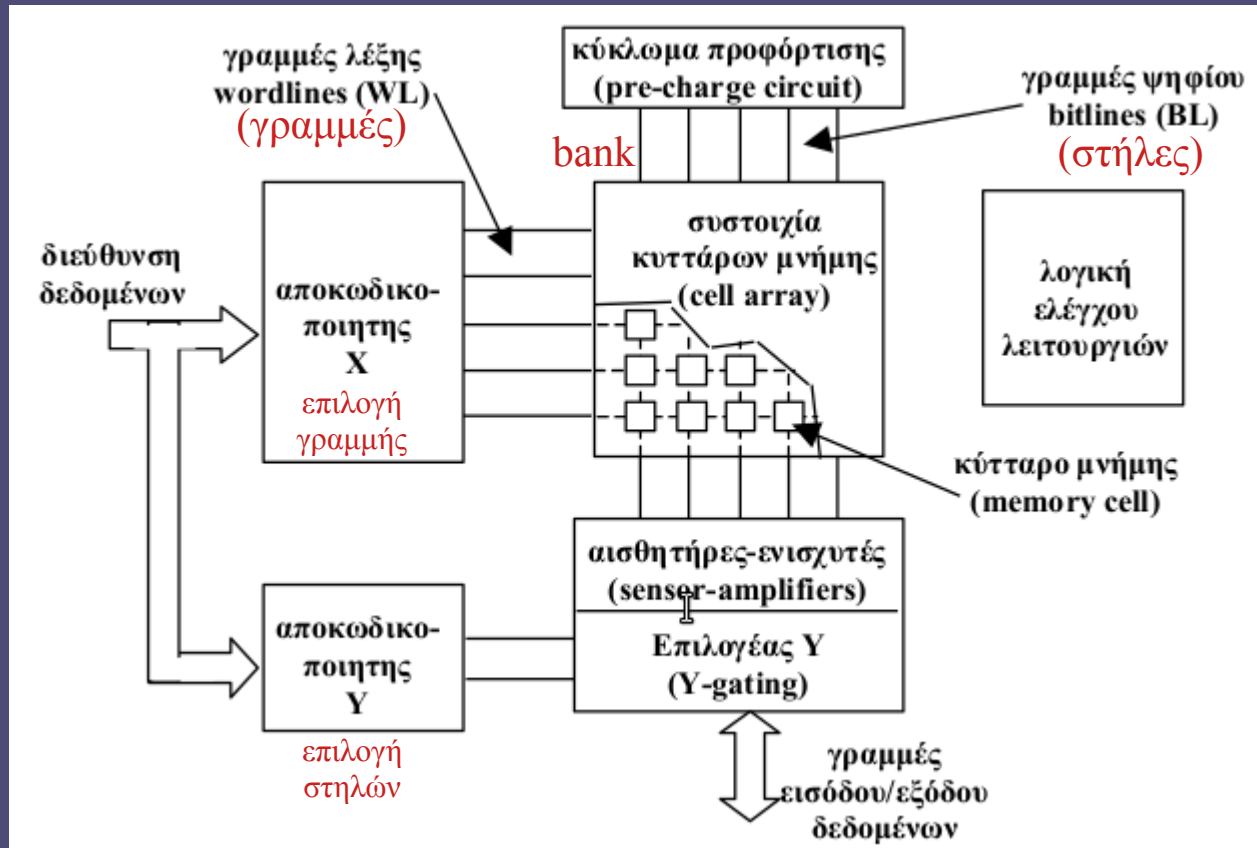
Η λέξη είναι η μικρότερη προσπελάσιμη ομάδα bits (π.χ. ένα byte ή πολλαπλάσιά του).

Διευθυνσιοδότηση μνήμης RAM



- **Byte addressing**
 - Οι διαδοχικές διευθύνσεις μνήμης αυξάνονται **ανά byte**
 - Ακόμα κι όταν η λέξη μνήμης έχει πολλαπλάσιο εύρος
 - Σε επεξεργαστές γενικού σκοπού
- **Εναλλακτικά: word addressing**
 - Οι διευθύνσεις αυξάνονται ανά **λέξη**
 - Σε υπερυπολογιστές ή ειδικούς επεξεργαστές ψηφιακών σημάτων – εδώ η προσπέλαση ανά byte είναι σπάνια

Οργάνωση Μνήμης Τυχαίας Προσπέλασης (RAM)



- Οι τρέχουσες μνήμες RAM διαθέτουν πολλαπλές (π.χ. 8) συστοιχίες κυττάρων μνήμης (banks)

Ταχύτητα Προσπέλασης RAM

- **Access Time (χρόνος προσπέλασης)**
 - Ο απαιτούμενος χρόνος για την ολοκλήρωση μιας αίτησης προς τη μνήμη RAM
 - Συχνά διαφορετικός για Ανάγνωση - Εγγραφή
- **Cycle Time (χρόνος κύκλου προσπέλασης)**
 - Ο ελάχιστος απαιτούμενος χρόνος μεταξύ διαδοχικών αιτήσεων προς τη μνήμη RAM
 - Προσθήκη χρόνου για ενδιάμεσες λειτουργίες (προετοιμασία για την επόμενη προσπέλαση)

Τύποι Μνήμης Τυχαίας Προσπέλασης

- **Στατική Μνήμη RAM (SRAM)**
 - Κάθε bit αποθηκεύεται σε κύτταρο (“cell”) 6 τρανζίστορ
 - Διατήρηση bit όσο υπάρχει τροφοδοσία της μνήμης
- **Η προσπέλαση είναι γρήγορη**
 - Ο χρόνος προσπέλασης μιας μνήμης SRAM βρίσκεται μεταξύ 0,5 και 5 ns
- **Αλλά:**
 - Πολυπλοκότερο κύκλωμα
 - Δεν επιτρέπει μεγάλη ολοκλήρωση
 - Μεγαλύτερη κατανάλωση ενέργειας
- **Χρησιμοποιείται στις κρυφές μνήμες (caches)**

Τύποι Μνήμης Τυχαίας Προσπέλασης

- **Δυναμική Μνήμη RAM (DRAM)**
 - Κάθε bit αποθηκεύεται ως φορτίο
 - Διατήρηση μόνο με συχνή **ανανέωση** του φορτίου (κάθε 16 έως 128 ms)
- **Απλούστερο κύκλωμα – μεγάλη ολοκλήρωση**
 - Πολύ μεγάλες χωρητικότητες (1Gbit/chip και μεγαλύτερες)
 - Η προσπέλαση είναι αργή
 - Ο χρόνος προσπέλασης μιας μνήμης DRAM βρίσκεται μεταξύ 50 και 70 ns
 - Αρχιτεκτονικές βελτιώσεις για την **αύξηση του ρυθμού μεταφοράς δεδομένων**
- **Χρησιμοποιείται για τη συγκρότηση της κύριας μνήμης**

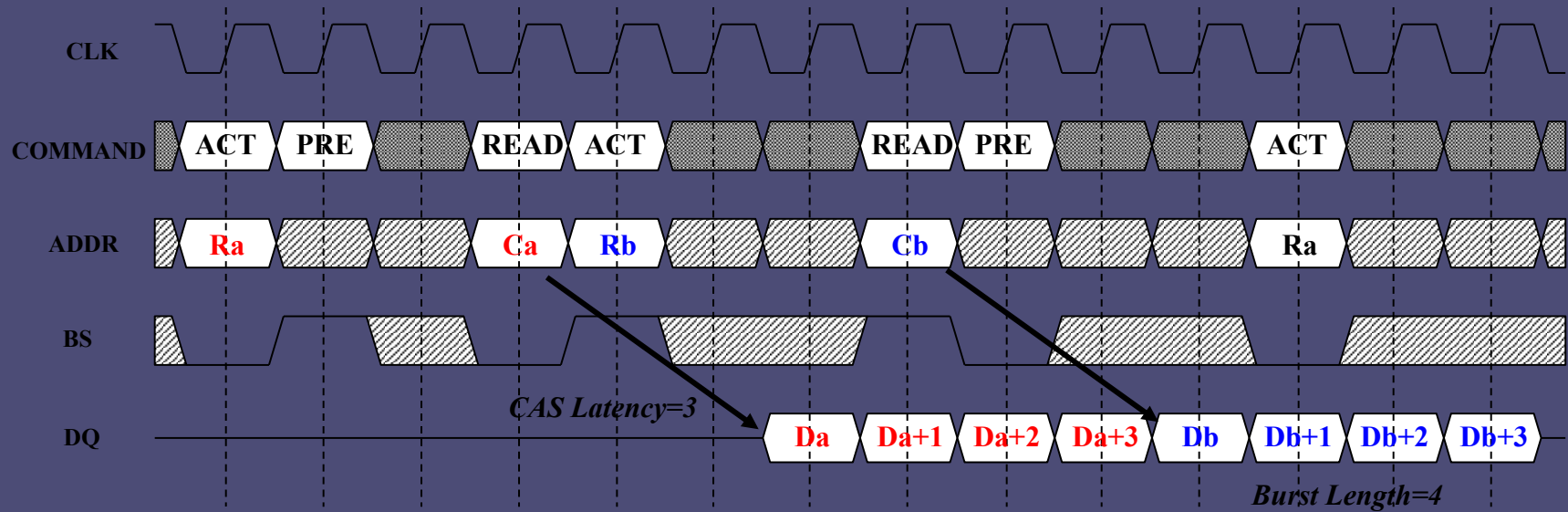
Βασικές λειτουργίες DRAM

- **ACTIVATE**
 - Επιλογή **γραμμής** (row) για ανάγνωση ή εγγραφή μέσω μέρους της διεύθυνσης
- **READ/WRITE**
 - Επιλογή **στηλών** (column) για ανάγνωση ή εγγραφή μέσω της υπόλοιπης διεύθυνσης
- **PRECHARGE**
 - Επιλογή **συστοιχίας** (bank) για προφόρτιση πριν την επόμενη ανάγνωση ή εγγραφή
- Λοιπές λειτουργίες
 - Refresh, ρυθμίσεις (μέγεθος μεταφοράς, αρχικοποίηση σημάτων κλπ)

Επικοινωνία με τη μνήμη DRAM

- Η βασική λειτουργία της μνήμης είναι **ασύγχρονη**
 - Η ανάγνωση και εγγραφή ολοκληρώνεται μετά από **συγκεκριμένο χρόνο** ανάλογα με την **τεχνολογία της μνήμης**
- Προσθήκη σημάτων **ρολογιού** για συγχρονισμό μεταφοράς δεδομένων
 - CLK : συγχρονίζει τα σήματα ελέγχου και διεύθυνσης (από ελεγκτή μνήμης)
 - Ξεχωριστό σήμα (strobe) DQS συγχρονίζει τη μεταφορά των δεδομένων (DQ)
 - Οδηγείται από τον ελεγκτή μνήμης (εγγραφή) ή τη μνήμη (ανάγνωση)
 - Μνήμες DDR: μεταφορά και στις δύο ακμές DQS (double-data rate)
- Πρότυπα
 - DDRx (x = 3,4,5...) για επικοινωνία με ξεχωριστά modules μνήμης
 - HBM για μνήμες που βρίσκονται μέσα στο τσιπ του επεξεργαστή

Παράδειγμα ανάγνωσης από DRAM

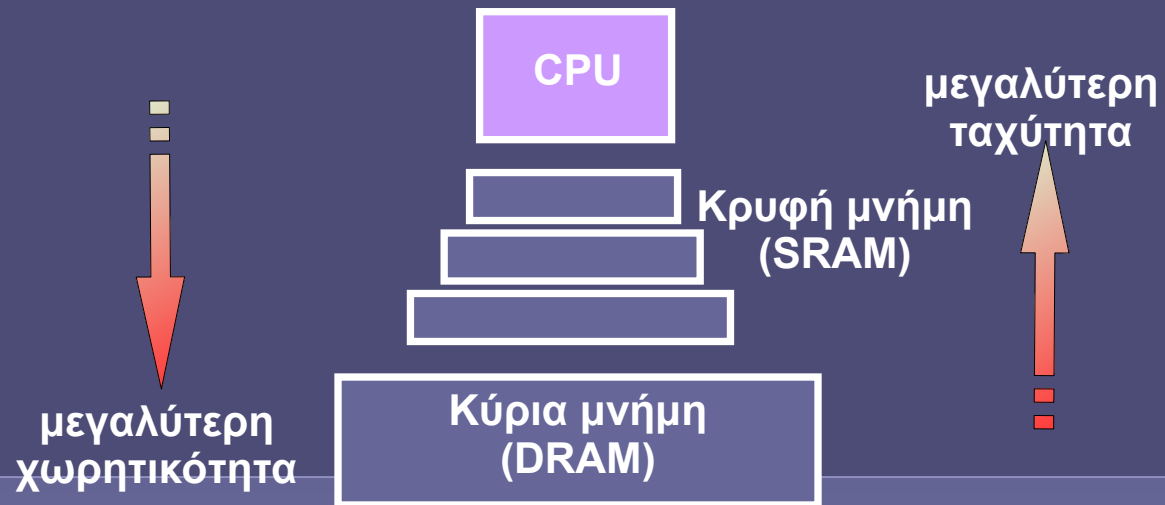


Απαιτήσεις από το σύστημα μνήμης

- Παράδειγμα: ένας επεξεργαστικός πυρήνας
 - με ρολόι 3 GHz
 - και έναρξη εκτέλεσης έως και 8 εντολών ανά κύκλο
 - απαιτεί από τη μνήμη 24G εντολές/sec
 - Τι συμβαίνει σε συστήματα με πολλούς πυρήνες;
- Η «ιδανική μνήμη» θα έπρεπε να είναι
 - Πολύ γρήγορη
 - Πολύ φθηνή
 - Με πολύ μεγάλη χωρητικότητα
 - Ιδιαίτερα χρήσιμη στις σημερινές εφαρμογές AI

Ιεραρχίες 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
 - Για να γεφυρώσει το χάσμα απόδοσης μεταξύ επεξεργαστών και μνημών



Ιεραρχία μνήμης και τοπικότητα

- **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

- **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:**

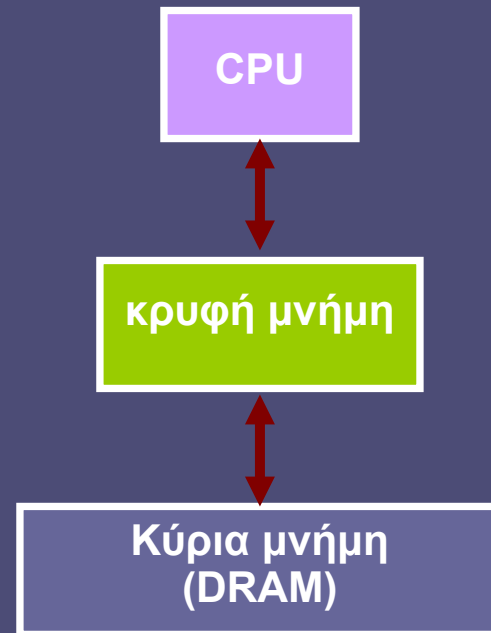
- Όταν προσπελαστεί μια θέση μνήμης, μεταφέρονται **και οι διπλανές της λέξεις** στην κρυφή μνήμη
 - Γρηγορότερη προσπέλαση όταν θα ζητηθούν και αυτές

Κρυφές μνήμες

- 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

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



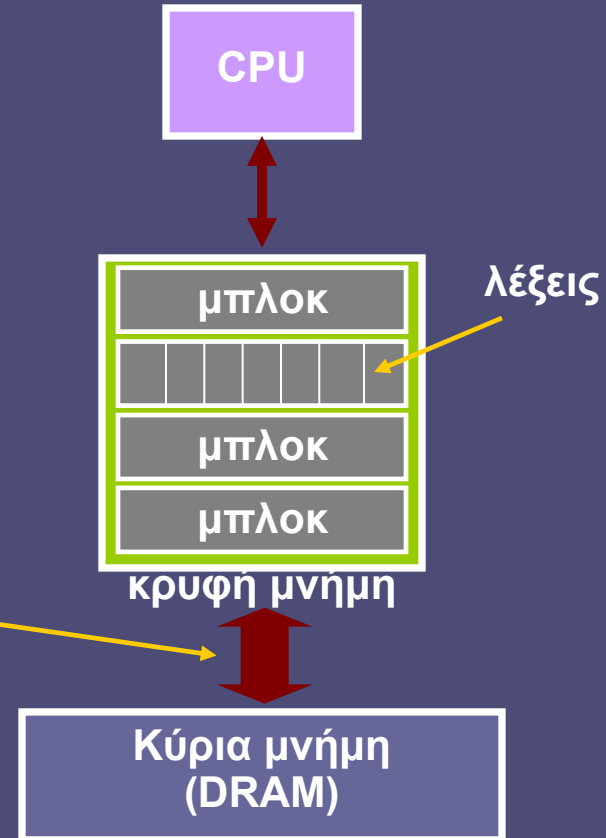
- Οι αρχές λειτουργίας της απλής ιεραρχίας μπορούν να επεκταθούν σε πολλαπλά επίπεδα (κρυφή μνήμη L1, L2, L3...)

Αποθήκευση δεδομένων στην Ιεραρχία Μνήμης

- **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)

Μπλοκ (γραμμές) κρυφής μνήμης

- 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 σημερινοί επεξεργαστές διαθέτουν κρυφές μνήμες με μέγεθος μπλοκ ίσο με 64 bytes



Αναζήτηση δεδομένων στην Ιεραρχία Μνήμης

- **Αναζήτηση δεδομένων**

- Η μονάδα επεξεργασίας ζητά **πάντοτε** τα δεδομένα/εντολές από το κοντινότερο σε αυτήν επίπεδο (κρυφή μνήμη)
- Τα δεδομένα υπάρχουν στην κρυφή μνήμη: **hit**
 - Τα δεδομένα επιστρέφονται γρήγορα στη μονάδα επεξεργασίας
- Τα δεδομένα δεν βρίσκονται στην κρυφή μνήμη: **miss**
 - Η αίτηση προωθείται στο επόμενο (χαμηλότερο) επίπεδο (κύρια μνήμη)
 - Το **μπλοκ** που περιέχει τα δεδομένα **αντιγράφεται** στην κρυφή μνήμη
 - Και τα δεδομένα που ζητήθηκαν επιστρέφονται στη μονάδα επεξεργασίας

Τι δημιουργεί cache misses;

- Η πρώτη φορά προσπέλασης ενός μπλοκ
 - Όταν ζητούνται από τη μονάδα επεξεργασίας μπλοκ που δεν βρέθηκαν ποτέ μέχρι τώρα στην κρυφή μνήμη
- Λόγω της πεπερασμένης 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
 - Ένα νέο μπλοκ όταν τοποθετηθεί στην κρυφή μνήμη εκτοπίζει ένα προηγούμενο διαφορετικό μπλοκ που βρισκόταν στην ίδια θέση

Θέματα κρυφών μνημών

- Πού αποθηκεύεται ένα μπλοκ στην κρυφή μνήμη;
- Πώς εντοπίζεται ένα μπλοκ στην κρυφή μνήμη;
- Ποιο μπλοκ θα αντικατασταθεί όταν χρειαστεί;
- Τι συμβαίνει στην εγγραφή νέων δεδομένων;
- Πώς υπολογίζεται η απόδοση της ιεραρχίας μνήμης;

(στο επόμενο μάθημα..)