

Ιόνιο Πανεπιστήμιο – Τμήμα Πληροφορικής  
Εισαγωγή στην Επιστήμη των Υπολογιστών  
2016-17

# Οργάνωση Υπολογιστών (I)

(η κεντρική μονάδα επεξεργασίας)

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

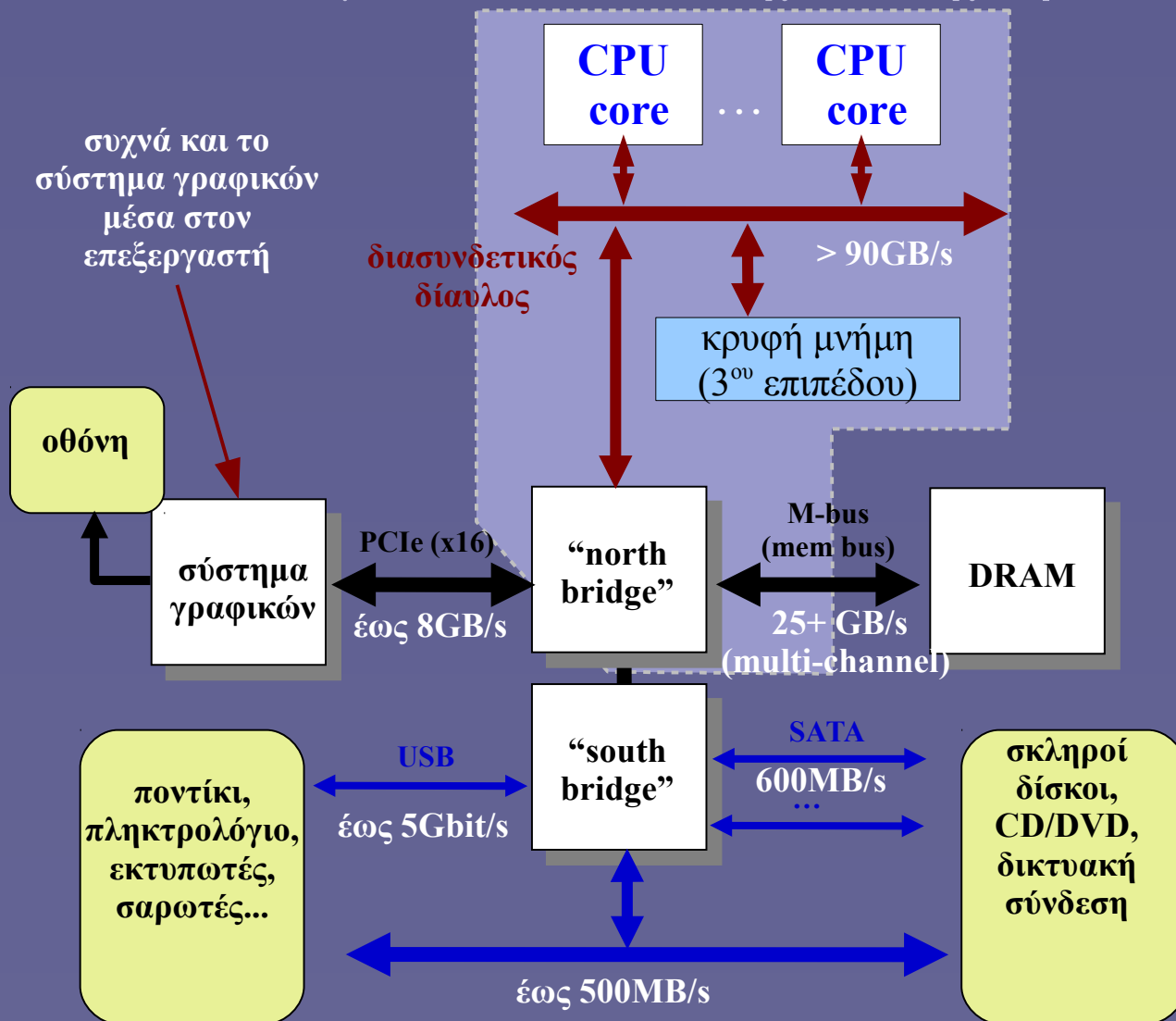
Μ.Στεφανιδάκης



# Ένα τυπικό υπολογιστικό σύστημα σήμερα

- Εισαγωγή

Στα σημερινά συστήματα ο όρος “Κεντρική Μονάδα Επεξεργασίας” (ΚΜΕ) αντιστοιχεί (περίπου) σε έναν επεξεργαστικό “πυρήνα” (CPU core). Οι σύγχρονοι επεξεργαστές διαθέτουν πολλούς επεξεργαστικούς πυρήνες.



**οι ρυθμοί μεταφοράς που δίνονται είναι οι θεωρητικά μέγιστοι!**

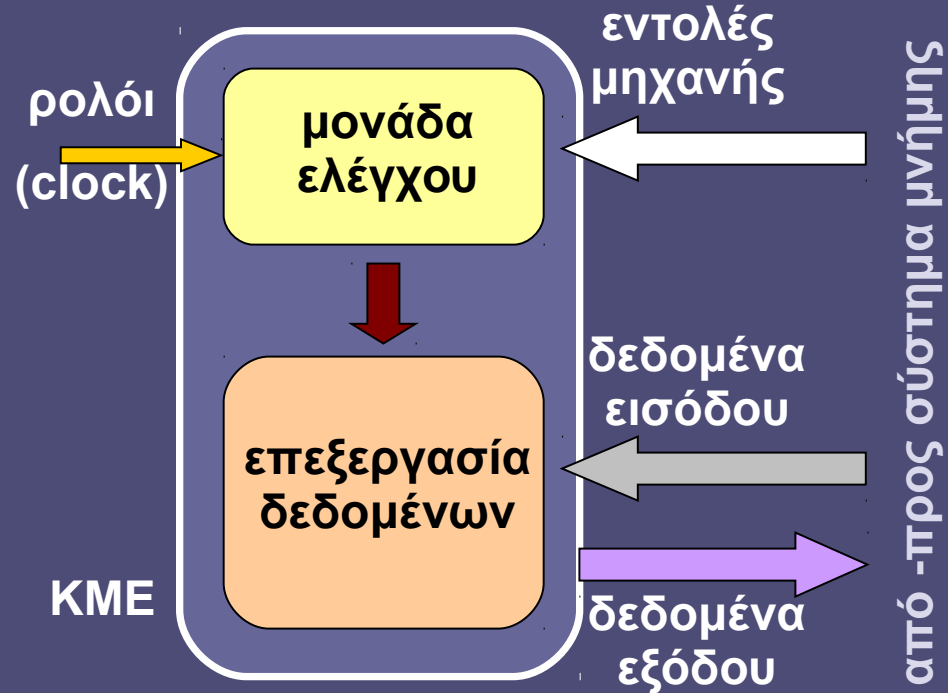
# Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)

- Εισαγωγή
- ΚΜΕ

- **Central Processing Unit (CPU)**
  - Ή απλά “πυρήνας” (“core”)
- **Εκτέλεση πράξεων στα δεδομένα**
- **Κύριες κατηγορίες πράξεων**
  - Αριθμητικές-λογικές πράξεις
  - Μεταφορές δεδομένων από-προς τη μνήμη
  - Συγκρίσεις και διακλάδωση υπό συνθήκη
- **Επιλογή επιθυμητής πράξης**
  - Εντολές μηχανής (σειρές από bits)
  - Πρόγραμμα: ακολουθία εντολών μηχανής

# Τμήματα ΚΜΕ (CPU)

- Εισαγωγή
- ΚΜΕ



- Επεξεργασία δεδομένων
  - Καθοδήγηση από τη μονάδα ελέγχου
    - Βάσει του προγράμματος εκτέλεσης (εντολών)

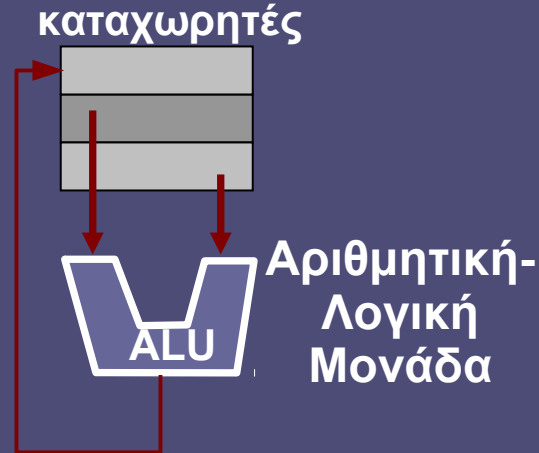
# Επεξεργασία δεδομένων

- Εισαγωγή
- ΚΜΕ

- “Μονοπάτι δεδομένων” (Datapath)
- Ποιος εκτελεί τις πράξεις μεταξύ δεδομένων;
  - Αριθμητικές-λογικές μονάδες (ΑΛΜ)
- Από πού προέρχονται τα δεδομένα εισόδου στις ΑΛΜ και πού αποθηκεύεται το αποτέλεσμα της πράξης;
  - Καταχωρητές
  - Μνήμη

# Καταχωρητές (registers)

- Εισαγωγή
- ΚΜΕ



- Καταχωρητές
  - Αυτόνομες θέσεις αποθήκευσης μέσα στην ΚΜΕ
    - Μία “λέξη” δεδομένων
  - Γρήγορη προσπέλαση
  - Προσωρινή αποθήκευση δεδομένων
    - Δεδομένα εισόδου και εξόδου ΑΛΜ (ALU)
  - Καταχωρητές γενικής χρήσης (general purpose)

# Έλεγχος εκτέλεσης εντολών

- Εισαγωγή
- ΚΜΕ

- Μονάδα ελέγχου (control unit)
  - Εκτέλεση εντολών μηχανής
- Ποια είναι η επόμενη προς εκτέλεση εντολή;
  - Μετρητής προγράμματος (Program Counter – PC ή Instruction Pointer – IP)
- Πώς επιλέγεται η πράξη της και τα δεδομένα εισόδου εξόδου της;
  - Αποκωδικοποίηση εντολών
    - Με βάση τα bits κάθε εντολής δημιουργούνται σήματα ελέγχου λειτουργίας όλης της ΚΜΕ

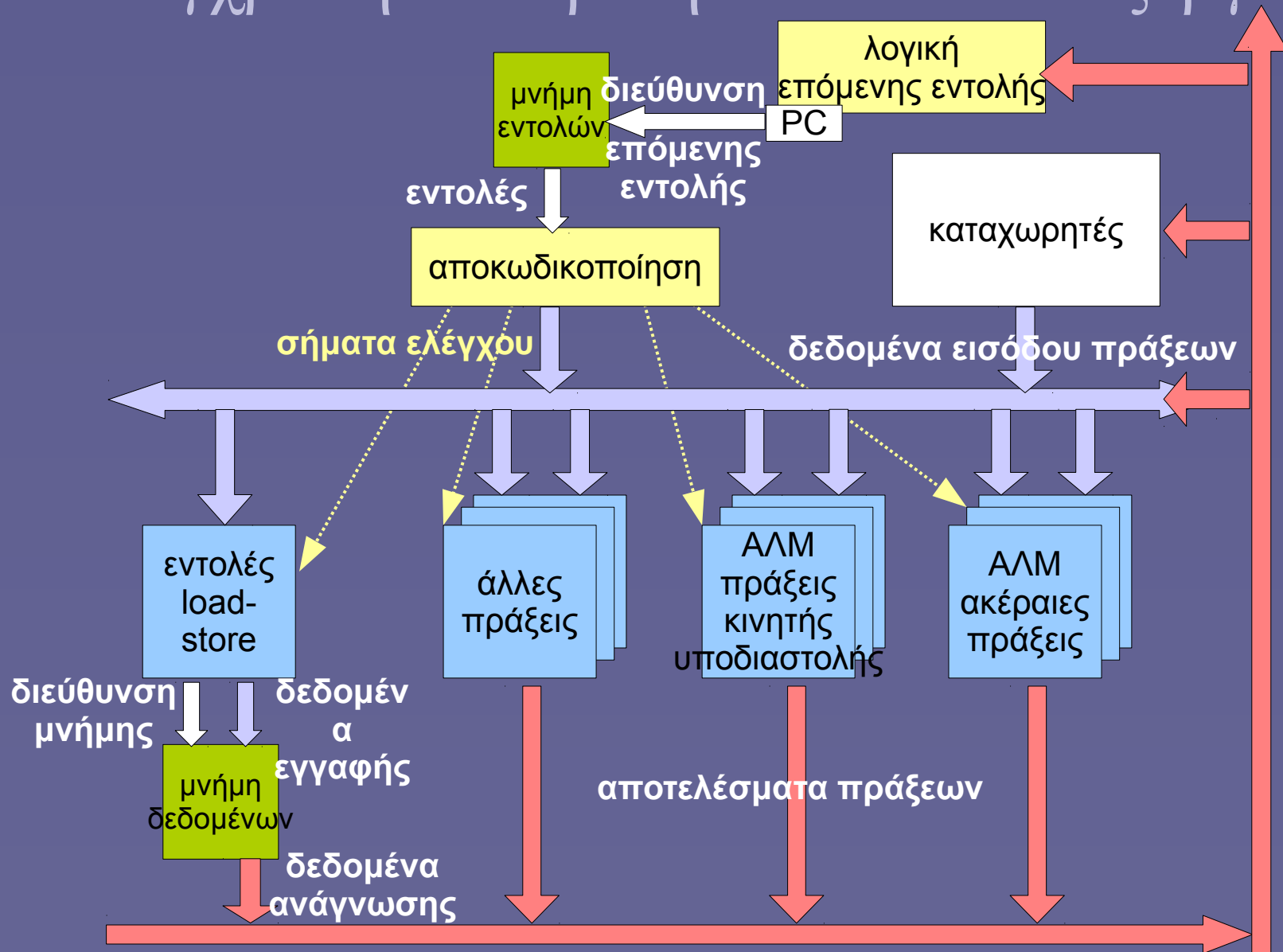
# Program Counter (PC)

- Εισαγωγή
- ΚΜΕ

- **Καταχωρητής διεύθυνσης**
  - Ονομάζεται επίσης Instruction Pointer (IP)
  - Διεύθυνση της επόμενης εντολής στη μνήμη
  - Ανάκληση εντολής από μνήμη
- **Αυτόματη αύξηση**
  - Δείχνει στην επόμενη θέση μνήμης (επόμενη εντολή) μετά την ανάκληση της τρέχουσας εντολής
- **Ειδική περίπτωση: διακλάδωση**
  - Μετά από σύγκριση
  - Φόρτωση του PC με διαφορετική τιμή
    - Μεταφορά της ροής εκτέλεσης σε άλλο σημείο
  - Υλοποίηση δομών **if ... then ... else**



# Η σύγχρονη Κεντρική Μονάδα Επεξεργασίας



# Αρχιτεκτονική συνόλου εντολών

- Εισαγωγή
- ΚΜΕ

- Τι πρέπει να ξέρουμε για να προγραμματίσουμε έναν επεξεργαστή (instruction set architecture – ISA)
  - Ποιες είναι οι διαθέσιμες εντολές
  - Τύποι και προέλευση δεδομένων
  - Ποιους καταχωρητές χρησιμοποιούμε
  - Ποιο το μοντέλο της διαθέσιμης μνήμης
    - Σήμερα: **επίπεδο μοντέλο**, ενιαίος χώρος μνήμης ανά πρόγραμμα, από τη διεύθυνση 0 έως τη μέγιστη δυνατή
    - Πώς σχηματίζεται μια διεύθυνση μνήμης
  - Πώς αντιδρά ο επεξεργαστής σε εξωτερικές διακοπές και καταστάσεις σφάλματος

# Αρχιτεκτονικές συνόλου εντολών

- Εισαγωγή
- ΚΜΕ

- **Υπολογιστές σύνθετου συνόλου εντολών**
  - Complex Instruction Set Computer (**CISC**)
  - Μεγάλο σετ εντολών (σύνθετες λειτουργίες)
  - Εύκολη συγγραφή προγραμμάτων
  - Πολύπλοκη ΚΜΕ
- **Υπολογιστές περιορισμένου σετ εντολών**
  - Reduced Instruction Set Computer (**RISC**)
  - Μικρό σύνολο απλών εντολών (απλές λειτουργίες)
  - Μεγαλύτερα προγράμματα
  - Απλούστερη ΚΜΕ
    - Μεγαλύτερη χρησιμοποίηση (παραλληλία)

# Η γλώσσα assembly

- Εισαγωγή
- ΚΜΕ

- Μνημονική αναπαράσταση των εντολών μηχανής
  - Αντί να γράφουμε σειρές από 0 και 1...
  - Κάθε εντολή assembly αντιστοιχεί σε μία συγκεκριμένη εντολή μηχανής
- Συγγραφή προγραμμάτων σε χαμηλό επίπεδο
  - Π.χ. κώδικας αρχικοποίησης του υπολογιστή
- Ο κώδικας assembly είναι διαφορετικός ανά επεξεργαστή!
  - Άλλο σετ εντολών, άλλα ονόματα καταχωρητών...
  - Δεν μεταφέρεται το ίδιο πρόγραμμα assembly σε διαφορετικό επεξεργαστή (non-portable)
- Στα παραδείγματά μας: assembly x86-32bit (στα PCs)

# Τι περιγράφουν τα bits μιας εντολής μηχανής;

- Εισαγωγή
- ΚΜΕ



Περιγράφει το είδος της πράξης που θα εκτελεστεί

Περιγράφουν την **προέλευση** των δεδομένων εισόδου (αριθμό καταχωρητή, διεύθυνση μνήμης κλπ) και τον **προορισμό** των δεδομένων εξόδου (αποτελέσματος πράξης)

- Το είδος της πράξης προσδιορίζει
  - τον **τύπο**, των δεδομένων (π.χ. ακέραιοι ή δεκαδικοί)
  - την **προέλευση** των δεδομένων (π.χ. από καταχωρητή ή από τη μνήμη)
  - και τον **αριθμό** των δεδομένων που συμμετέχουν στην πράξη

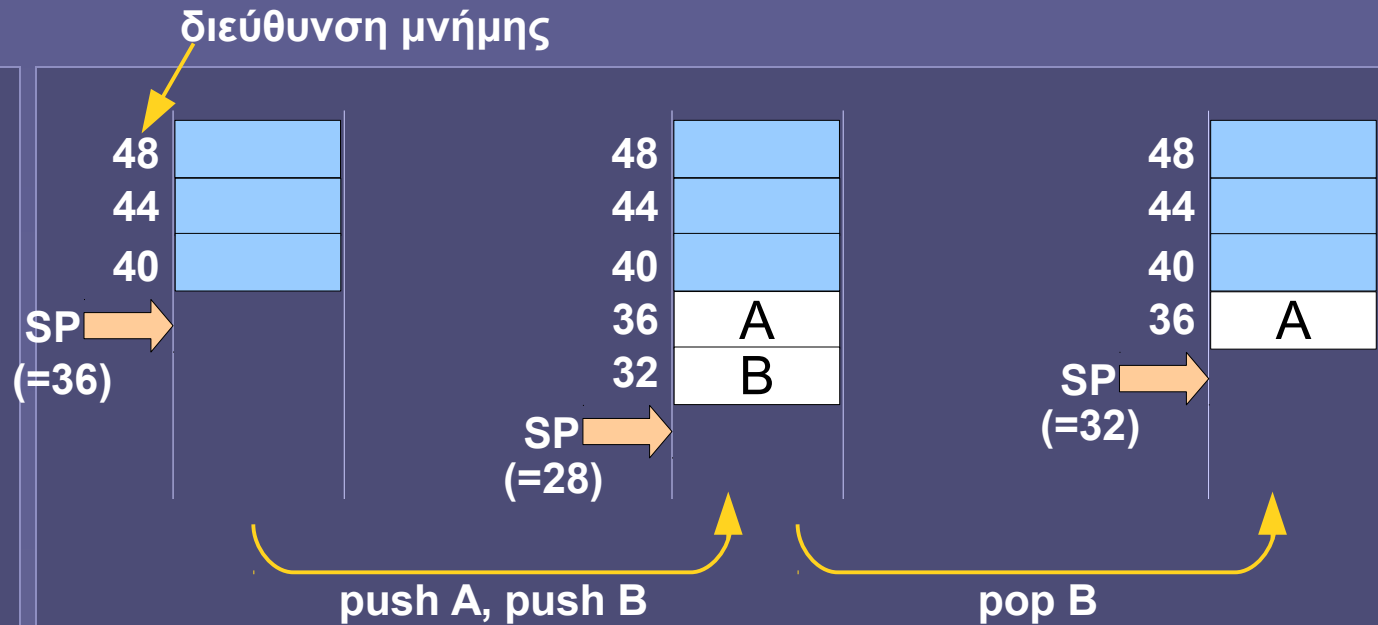
# Πηγές και προορισμός δεδομένων εντολών

- Εισαγωγή
- ΚΜΕ

- Από-προς καταχωρητές  
`mov eax,ebx` ; μετακίνησε το περιεχόμενο του  
; καταχωρητή ebx στον eax
- Από-προς θέσεις μνήμης  
`cmp eax,[intlist+edi*4]` ; σύγκρινε το περιεχόμενο  
; του eax με θέση μνήμης
  - διεύθυνση = σταθερά + τιμή καταχωρητή\*σταθερά
  - εκτός από το προηγούμενο παράδειγμα, η διεύθυνση μνήμης μπορεί να προκύψει και με άλλους συνδυασμούς!
- Από σταθερές μέσα στην εντολή  
`add edi,33` ; αύξησε το περιεχόμενο του  
; καταχωρητή edi κατά 33
  - ο αριθμός 33 βρίσκεται μέσα στα bits της εντολής

# Η στοίβα (stack)

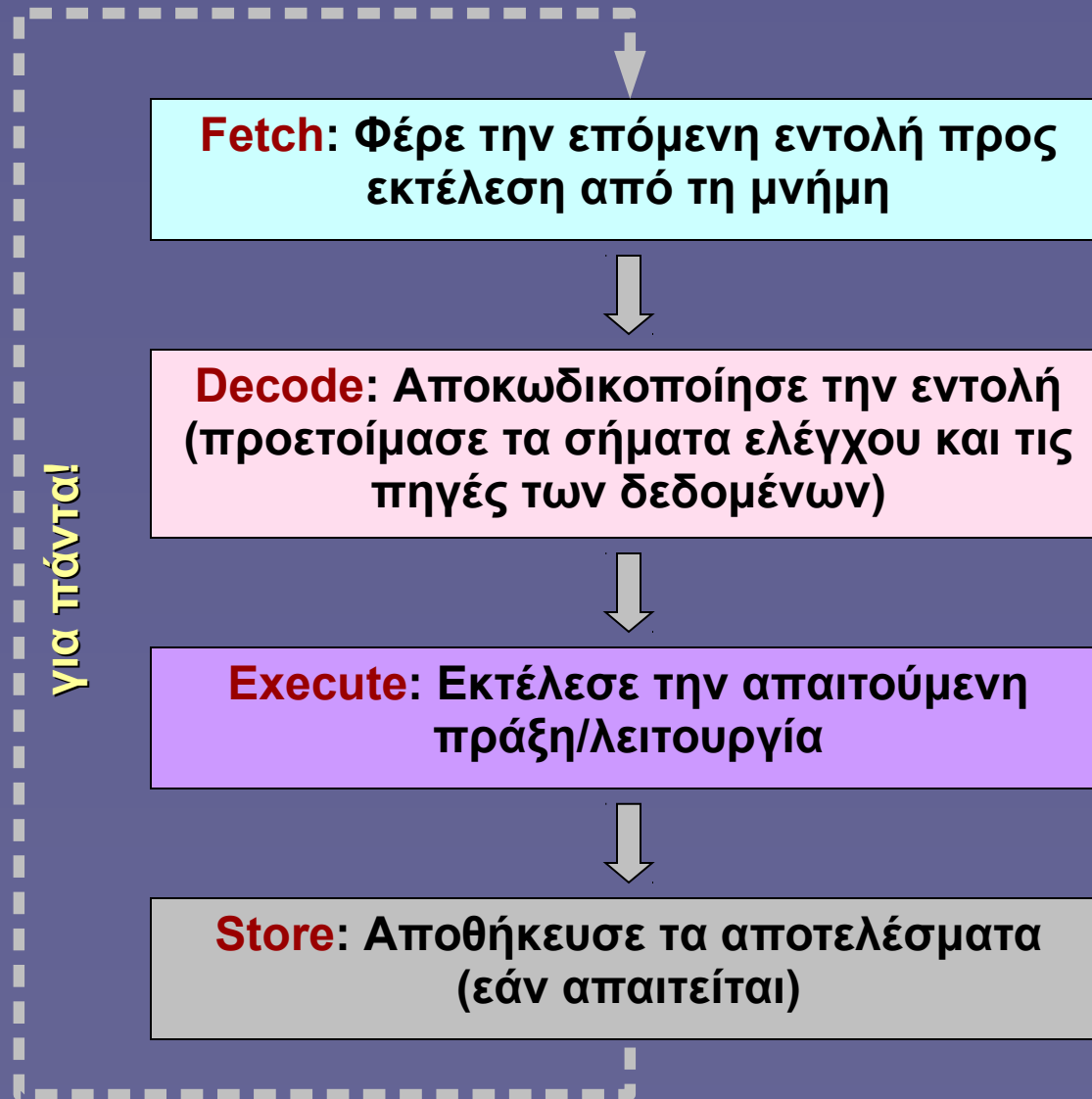
- Εισαγωγή
- ΚΜΕ



- Η ΚΜΕ παρέχει ειδικό καταχωρητή (stack pointer – SP) και εντολές **push** και **pop**
- Η στοίβα δεν είναι ειδική μνήμη αλλά ένα εναλλακτικό μοντέλο προσπέλασης της κανονικής μνήμης
  - Αποθήκευση διεύθυνσης επιστροφής από συναρτήσεις
  - Αποθήκευση τοπικών μεταβλητών δομημένων γλωσσών

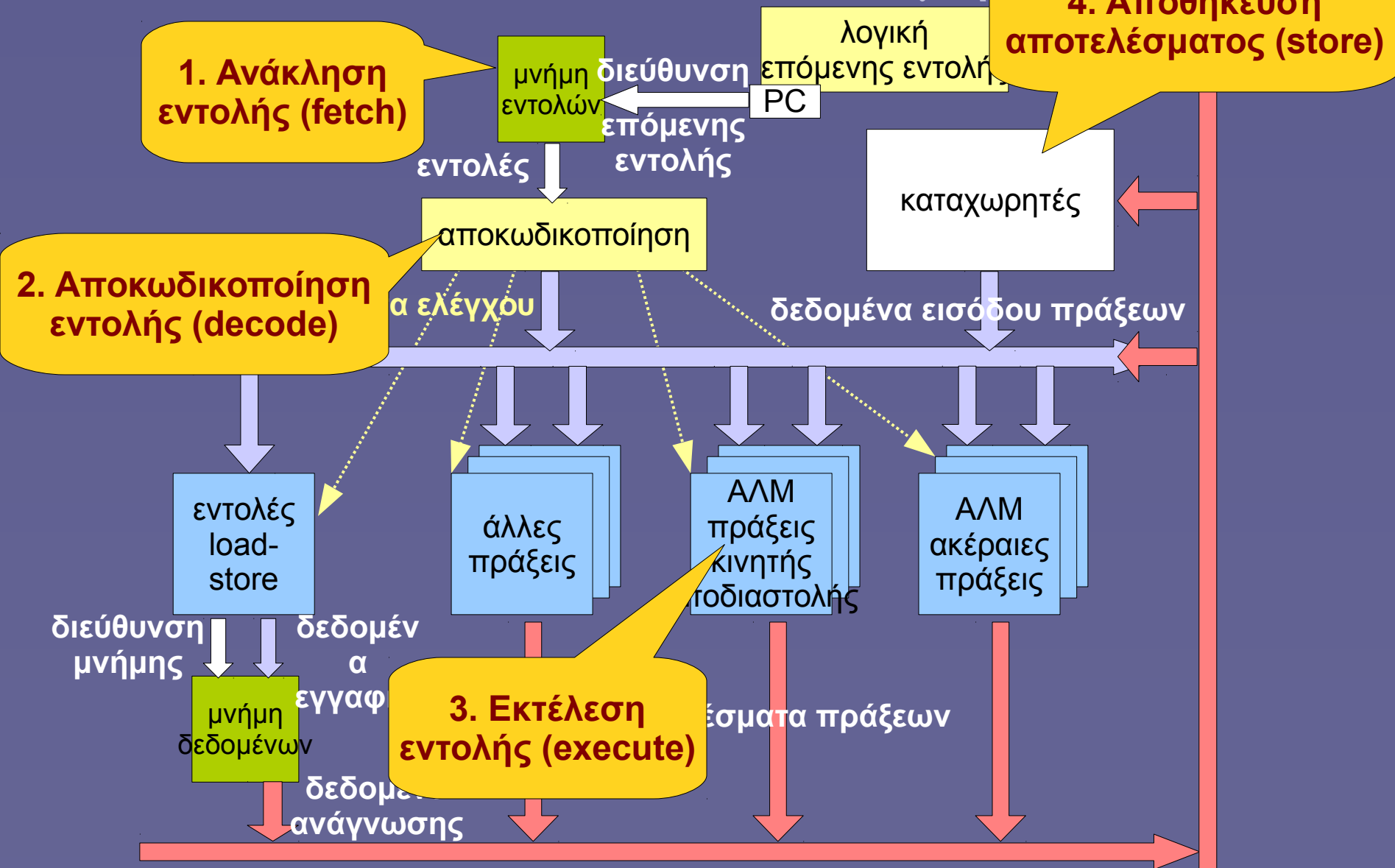
# Εκτέλεση εντολών: ο κύκλος μηχανής

- Εισαγωγή
- ΚΜΕ





# Τα στάδια του κύκλου μη



# Διακοπές και Σφάλματα

- Εισαγωγή
- ΚΜΕ

- Ο κύκλος μηχανής επαναλαμβάνεται ντετερμινιστικά και συνεχώς
  - Είτε με την επόμενη εντολή είτε με την εντολή μετά από διακλάδωση
- Εκτός αν συμβεί **μια διακοπή ή ένα σφάλμα**
  - **Διακοπή (interrupt)**: προκαλείται από εξωτερική συσκευή, οδηγώντας έναν ακροδέκτη εισόδου του επεξεργαστή από 0 σε 1 (ή ανάποδα).
  - **Σφάλμα (exception)**: κατά την εκτέλεση, λόγω π.χ. διαίρεσης δια 0, προσπέλασης άκυρης διεύθυνσης μνήμης κ.ά.

# Αντίδραση σε διακοπές

- Εισαγωγή
- ΚΜΕ

- Διακόπτεται το εκτελούμενο πρόγραμμα
  - Αφού ολοκληρωθεί η τρέχουσα εντολή
- Αποθηκεύονται στη στοίβα οι τιμές των καταχωρητών και ο Program Counter
  - Δηλαδή, η κατάσταση του προγράμματος
- Η εκτέλεση μεταβαίνει σε προκαθορισμένη για κάθε είδους διακοπή θέση μνήμης
  - Όπου το λειτουργικό σύστημα έχει τοποθετήσει μια διακλάδωση στη ρουτίνα εξυπηρέτησης της διακοπής (interrupt service routine – ISR)
- Με την ολοκλήρωση της ρουτίνας εξυπηρέτησης, η εκτέλεση επιστρέφει στο πρόγραμμα που διακόπηκε
  - Αφού αποκατασταθούν οι καταχωρητές

# Σημαίες κατάστασης (flags)

- Εισαγωγή
- ΚΜΕ

- Μια ομάδα bits που αναφέρουν την κατάσταση της ΚΜΕ μετά την εκτέλεση μιας εντολής
  - Κάθε εντολή επηρεάζει ορισμένα μόνο flags
- Τα πιο κοινά flags:
  - (Z)ero flag = μηδενικό αποτέλεσμα (της προηγούμενης πράξης)
  - (S)ign flag = αρνητικό αποτέλεσμα
  - (C)arry flag = ύπαρξη τελικού κρατουμένου
  - o(V)erflow flag = ένδειξη υπερχείλισης
- Άλλα flags τίθενται από το πρόγραμμα για να ειδοποιήσουν την ΚΜΕ για μια επιλογή
  - Π.χ. το (I)nterrupt flag δηλώνει αν επιτρέπουμε διακοπές ή όχι

# Διακλάδωση υπό συνθήκη και flags

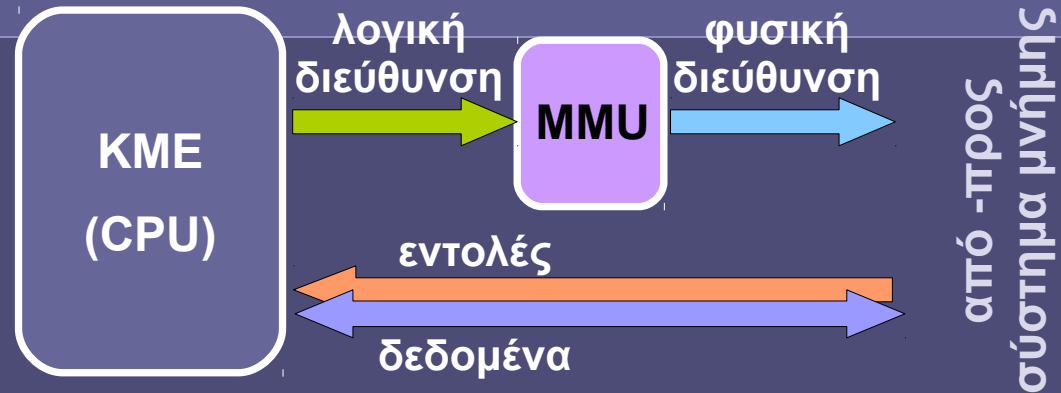
- Εισαγωγή
- ΚΜΕ

- Κάθε εντολή διακλάδωσης υπό συνθήκη εξετάζει ορισμένα flags για να αποφασίσει αν θα εκτελεστεί η διακλάδωση ή όχι
- Τα flags έχουν τεθεί από την **αμέσως προηγούμενη** εντολή
- Παράδειγμα:

**cmp edi,10** ; σύγκριση του περιεχομένου του  
; καταχωρητή edi με το 10  
; η εντολή cmp θέτει ανάλογα τα Z, C, S και V flags  
**jne again** ; διακλάδωση εάν Z flag = 0

# Εικονική μνήμη

- Εισαγωγή
- ΚΜΕ



- Ένα πρόγραμμα “βλέπει” **λογικές** διευθύνσεις
  - Εικονική Μνήμη (virtual memory)
  - Μετάφραση σε **φυσικές** διευθύνσεις μνήμης
  - Από το σύστημα διαχείρισης μνήμης (memory management unit – MMU) που βρίσκεται επίσης μέσα στον επεξεργαστή
    - Ευκολία στη μεταγλώττιση – εκτέλεση
    - Προστασία δεδομένων διεργασιών
    - Εικονική μνήμη μεγαλύτερη από τη φυσική

# Παράδειγμα εντολών: εύρεση μεγίστου

intlist+0	+4	+8	+12	+16	+20	+24	+28	+32	+36
0A 00 00 00	FD FF FF FF	1B 00 00 00	23 00 00 00	BD FF FF FF	E4 FF FF FF	07 00 00 00	FD FF FF FF	2B 00 00 00	FF FF FF FF
(10)	(-3)	(27)	(35)	(-67)	(-28)	(7)	(-3)	(43)	(-1)

\_start:

```
mov eax,[intlist]      ; maxint = intlist[0]
mov edi,1              ; i = 1 (2nd item of list)
```

again:

```
cmp eax,[intlist+edi*4] ; compare maxint with intlist[i]
jge skip                ; if maxint >= intlist[i] continue
mov eax,[intlist+edi*4] ; else, maxint = intlist[i]
```

skip:

```
add edi,1              ; i += 1
cmp edi,10             ; compare i with length(intlist)
jne again              ; if i != length(intlist), loop again
```

# Επεξηγήσεις

- Εισαγωγή
- ΚΜΕ

- Στην προηγούμενη διαφάνεια:
  - Λίστα 10 ακεραίων (32 bits)
  - Κάθε στοιχείο απέχει 4 bytes από το προηγούμενο
  - Η γλώσσα assembly ανήκει στην αρχιτεκτονική x86 (των PCs μας)
  - eax και edi είναι ονόματα καταχωρητών 32-bit
  - Οι αγκύλες [...] υποδηλώνουν προσπέλαση μνήμης
  - Το πρόγραμμα μπορεί να γραφεί και με πιο αποδοτικό τρόπο