

Ιόνιο Πανεπιστήμιο – Τμήμα Πληροφορικής  
Αρχιτεκτονική Υπολογιστών  
2021-22

# Αρχιτεκτονικές Συνόλου Εντολών

(Instruction Set Architectures - ISA)

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

Μ.Στεφανιδάκης



# Ο (μικρο)επεξεργαστής

- (Micro)processor
  - Μέρος ενός ευρύτερου υπολογιστικού συστήματος
  - Αρχικά: μόνο η Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)
    - Central Processing Unit (CPU)
  - Περιέχει σήμερα πολλαπλές υπομονάδες επεξεργασίας
    - Cores
    - Με διαφορετικά χαρακτηριστικά ή/και ρόλους
  - Και μέρος της ιεραρχίας μνήμης
  - Καθώς και μέρος των μονάδων ελέγχου συσκευών εισόδου - εξόδου

# Είδη μικροεπεξεργαστών

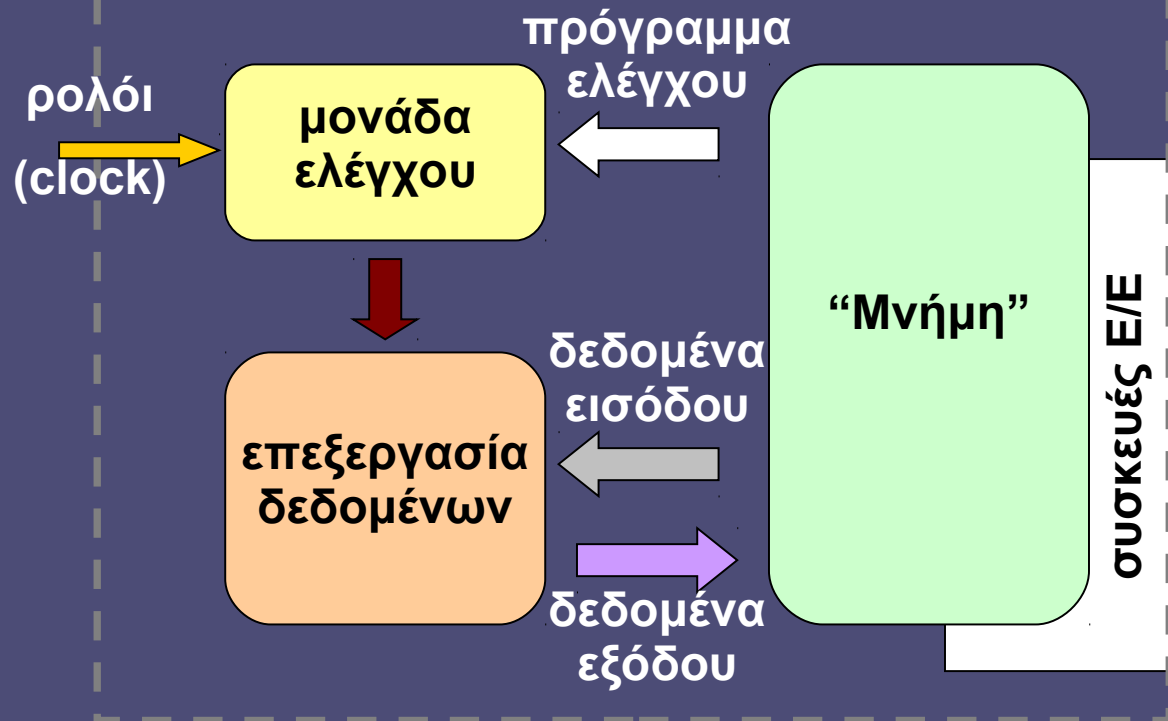
- Γενικού σκοπού
  - Οι περισσότεροι γνωστοί επεξεργαστές
- Συνεπεξεργαστές (co-processors)
  - Ειδικές λειτουργίες
- Μικροελεγκτές (microcontrollers)
  - Συστήματα ελέγχου
- Μέσα σε Systems-on-Chip
  - Μαζί με κυκλώματα σχετικά με τη λειτουργία του συστήματος
    - π.χ. με τον «ασύρματο» ενός κινητού

# Κεντρική Μονάδα Επεξεργασίας

- Central Processing Unit (CPU)
  - Ένας όρος που τείνει προς εξαφάνιση
  - Μετά την εμφάνιση των πολλών/διαφορετικών μονάδων επεξεργασίας στο ίδιο τσιπ
- Ποιος ο ρόλος μιας Μονάδας Επεξεργασίας
  - Μετασχηματίζει (επεξεργάζεται) **δεδομένα** σύμφωνα με ένα **πρόγραμμα ελέγχου**
  - Το πρόγραμμα ελέγχου αποτελείται από **εντολές μηχανής**

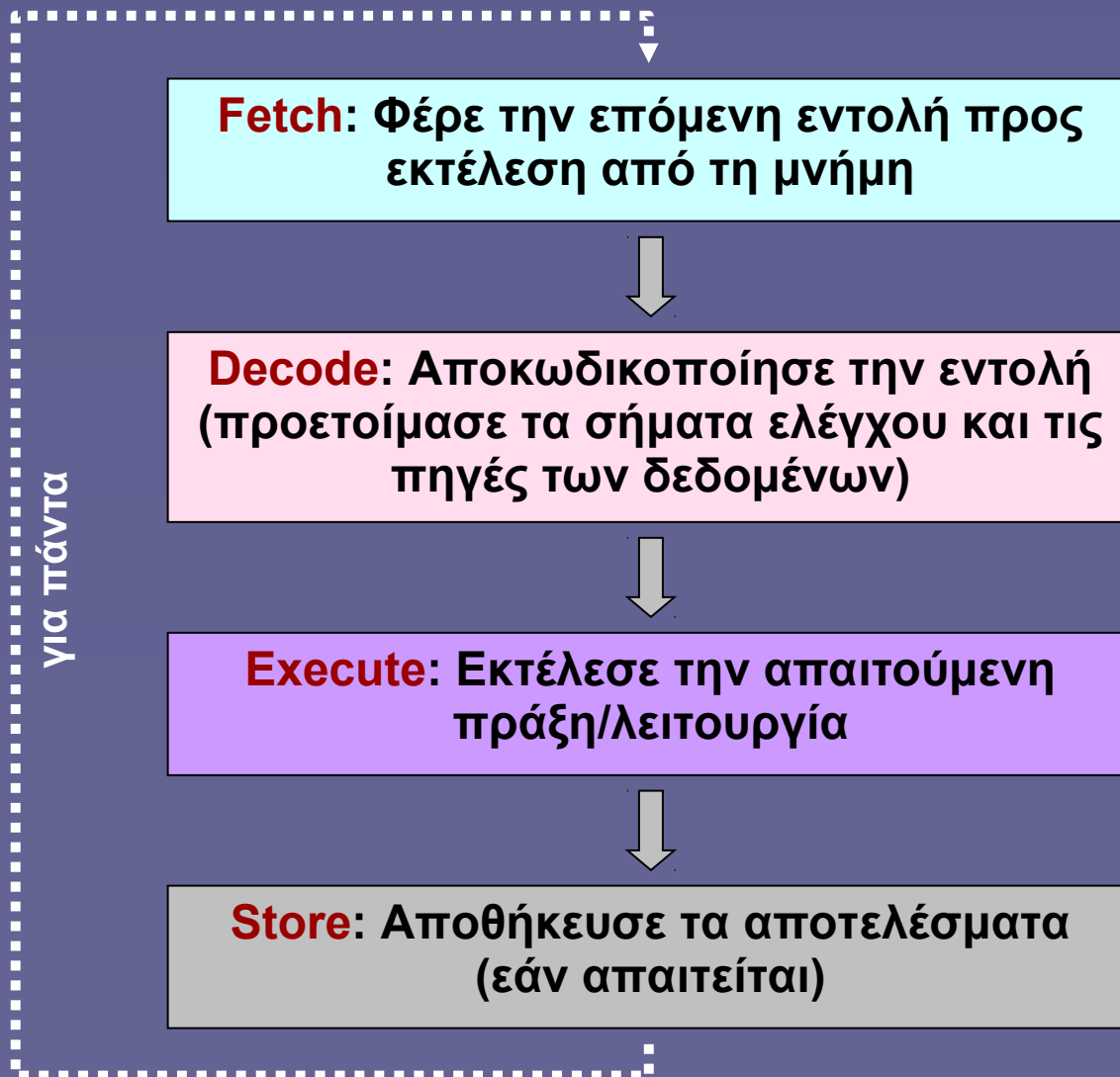
# Το μοντέλο «von Neumann»

*υπολογιστικό σύστημα*



- Το **πρόγραμμα ελέγχου**, όπως και **τα δεδομένα**, αποθηκεύονται στη **μνήμη** του υπολογιστή
  - “Stored-program computer”

# Εκτέλεση εντολών: ο κύκλος μηχανής



# Εκτέλεση εντολών

- Επόμενη εντολή προς εκτέλεση
  - **Program Counter (PC)**: καταχωρητής που περιέχει τη διεύθυνση της θέσης μνήμης όπου βρίσκεται η επόμενη εντολή
  - **Σειριακή αύξηση διεύθυνσης** μετά την εκτέλεση εντολής
  - Ή μεταπήδηση σε νέα θέση μνήμης (**διακλάδωση**)
  - Η διαδικασία επαναλαμβάνεται συνεχώς
    - Όσο η Μονάδα Επεξεργασίας είναι σε λειτουργία
    - Πολλές μονάδες μπορούν να «παγώσουν» τη λειτουργία τους (κατάσταση HALT)
      - Σε αναμονή εξωτερικού σήματος επανεκκίνησης

# Η πρώτη εντολή που εκτελείται

- Εκκίνηση εκτέλεσης
  - Με την εφαρμογή τάσης ο PC παίρνει μια προκαθορισμένη τιμή
    - Συνήθως στην αρχή ή στο τέλος της υποστηριζόμενης περιοχής μνήμης
    - Ανάλογα με την αρχιτεκτονική της κάθε Μονάδας Επεξεργασίας
    - Εκεί ο κατασκευαστής έχει τοποθετήσει τις πρώτες εντολές αρχικοποίησης του συστήματος



# Αρχιτεκτονική Συνόλου Εντολών

- Instruction Set Architecture (ISA)
  - Το ορατό μέρος ενός υπολογιστικού συστήματος για τον προγραμματιστή (και τον μεταγλωττιστή)
  - Δεκαετία 60-70: συνώνυμο του όρου «αρχιτεκτονική H/Y»
    - « η δομή ενός υπολογιστή, την οποία ο προγραμματιστής πρέπει να γνωρίζει για να γράψει ένα σωστό (χρονικά ανεξάρτητο) πρόγραμμα σε γλώσσα μηχανής για τον υπολογιστή αυτόν» (IBM)

# Η διεπαφή ISA στην ιεραρχία επιπέδων

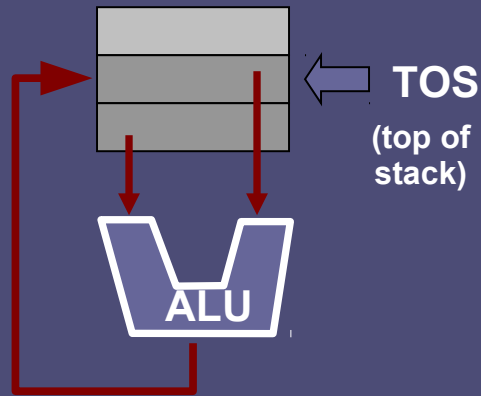


- Αρχιτεκτονική Εντολών (ISA)
  - Η διεπαφή υλικού-λογισμικού

# Αρχιτεκτονική Συνόλου Εντολών (ISA)

- Τι περιγράφει;
  - Διαθέσιμες πράξεις/λειτουργίες
    - Κωδικοποίηση λειτουργιών
  - Μορφή των δεδομένων εισόδου-εξόδου
    - Operands
  - Μέθοδοι προσπέλασης μνήμης
    - Προέλευση των δεδομένων
  - Χώροι προσωρινής αποθήκευσης
    - Καταχωρητές
  - Διακοπές και καταστάσεις σφάλματος
    - Ποια η “αντίδραση” του επεξεργαστή

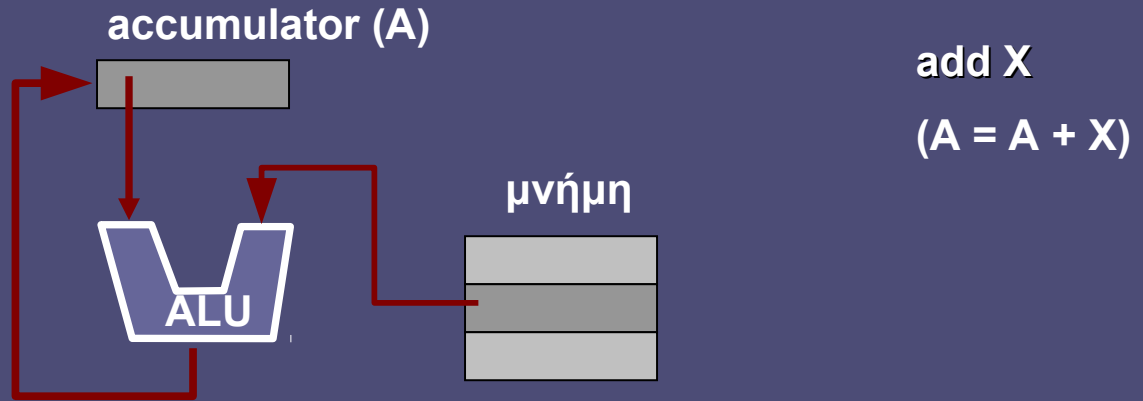
# Αρχιτεκτονική σωρού (stack)



push A  
push B  
add  
pop A

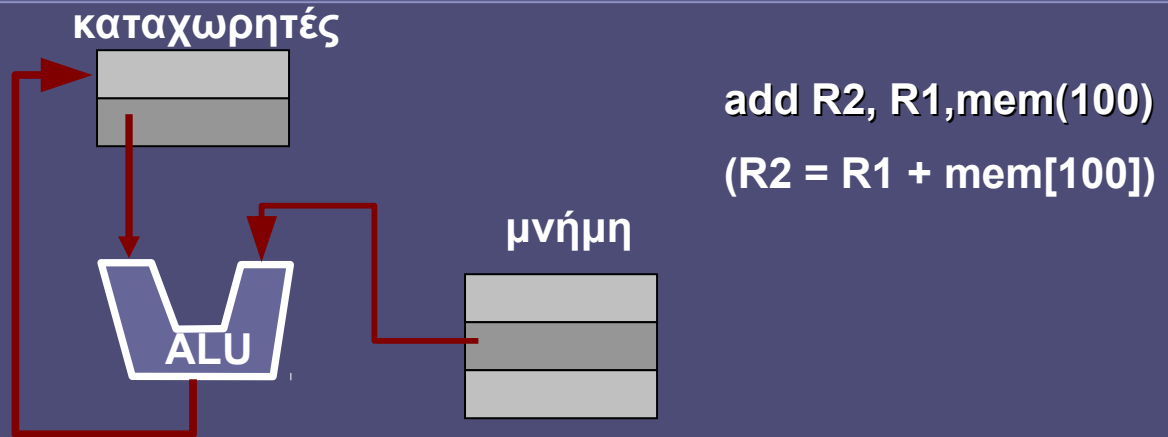
- Οι πηγές προσδιορίζονται έμμεσα
  - Δεν περιγράφονται στην εντολή!
  - 0-address architecture
  - Δημοφιλές σχήμα κατά τη δεκαετία του 60
  - Δύσκολη προσπέλαση σωρού, απαιτούνται πολλαπλές αντιμεταθέσεις και αντιγραφές για να έρθουν τα δεδομένα στη σωστή θέση

# Αρχιτεκτονική συσσωρευτή (accumulator)



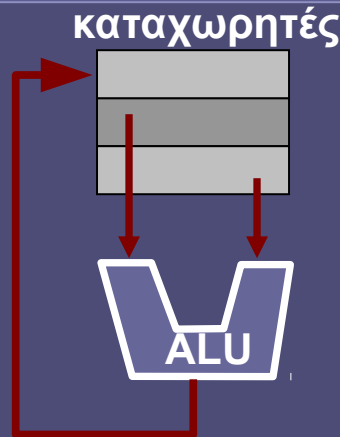
- Μια πηγή δεδομένων **και ταυτόχρονα** θέση αποθήκευσης του αποτελέσματος είναι πάντα ο συσσωρευτής
  - 1-address architecture
  - **Αρχιτεκτονική των πρώτων υπολογιστών**

# Αρχιτεκτονικές με καταχωρητές (registers)



- **Memory-register**
  - Οποιαδήποτε εντολή μπορεί να προσπελάσει τη μνήμη
- Όμως:
  - Πολλαπλές προσπελάσεις μνήμης
  - Λήψη εντολής – Λήψη δεδομένων εντολής
  - Πολύπλοκη εκτέλεση εντολής σε στάδια
  - Συνωστισμός στον δίαυλο επικοινωνίας με μνήμη

# Αρχιτεκτονικές με καταχωρητές (registers)



**add R1, R2,R3**

**(R1 = R2 + R3)**

- **Register-register (load-store)**
  - Μόνο εντολές **load-store** μπορούν να προσπελάσουν τη μνήμη
- **Η αρχιτεκτονική των σύγχρονων επεξεργαστών**
  - Οι καταχωρητές προσπελούνται πολύ γρήγορα
  - Χρειάζονται λιγότερα bits για να επιλεγούν
  - Οι μεταγλωττιστές αναθέτουν μεταβλητές σε καταχωρητές

# Η εξέλιξη της αρχιτεκτονικής εντολών

- Οι πρώτοι υπολογιστές (.. - '60)
  - Αρχιτεκτονική συσσωρευτή και αργότερα σωρού
    - Ικανοποιητική λύση λόγω της απλής τεχνολογίας των μεταγλωττιστών
- Πολύπλοκες αρχιτεκτονικές ('70 - ..)
  - Ενσωμάτωση σύνθετων μορφών εντολών και μεθόδων προσπέλασης μνήμης
    - Προσπάθεια υποστήριξης υψηλών γλωσσών προγραμματισμού – μείωσης κόστους λογισμικού
    - Πολλά χαρακτηριστικά μένουν αχρησιμοποίητα...
  - **Complex Instruction Set Computers (CISC)**



# Η εξέλιξη της αρχιτεκτονικής εντολών

- **Reduced Instruction Set Computers (RISC) ('80 - ...)**
  - Απλούστερες και φθηνότερες load-store αρχιτεκτονικές με σταθερό μήκος εντολών
  - Μεγαλύτερη απόδοση – ταχύτερη εκτέλεση εντολών
  - Ευνοείται από την αφθονία υλικού χαμηλού κόστους και την προηγμένη τεχνολογία των μεταγλωττιστών
  - Οι σημερινοί επεξεργαστές με εντολές CISC (π.χ. η αρχιτεκτονική x86), μεταφράζουν εσωτερικά σε εντολές RISC

# Εντολές: κατηγορίες λειτουργιών

- Βασικές κατηγορίες
  - Αριθμητικές και λογικές πράξεις
  - Μεταφορά δεδομένων
    - Από-πρός Καταχωρητές και Μνήμη
  - Έλεγχος ροής εκτέλεσης
    - Διακλαδώσεις και κλήσεις συναρτήσεων
- Άλλες κατηγορίες
  - Ειδικές εντολές συστήματος
    - ΛΣ, εικονική μνήμη
    - Απαιτούνται αυξημένα προνόμια εκτέλεσης
  - Επεξεργασία πολλαπλών δεδομένων ταυτόχρονα («παράλληλα»)
    - Χρήσιμο για γραφικά, σειρές χαρακτήρων, multimedia

# Κωδικοποίηση Εντολών



- Σειρά δυαδικών ψηφίων
  - Μεταβλητού μήκους
    - Συμπαγή προγράμματα (μικρότερο μέγεθος)
    - **Σημαντικά πολυπλοκότερο υλικό**
  - Σταθερού μήκους
    - **Απλούστερη και ταχύτερη λήψη-αποκωδικοποίηση**
    - Μεγαλύτερα προγράμματα
    - Μέθοδοι συμπίεσης

# Κωδικοποίηση Εντολών



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, την προέλευση και τον αριθμό των δεδομένων που συμμετέχουν στην πράξη !*

# Αριθμητικές/λογικές εντολές

- Αριθμητικές-λογικές πράξεις

- Είδος πράξης
- Είδος δεδομένων
- Πηγές δεδομένων και προορισμός
- Παράδειγμα (θεωρητικό):

- $Rd = Rs1 + Rs2$

add	Rd	Rs1	Rs2
-----	----	-----	-----

- Rd: προορισμός (καταχωρητής αποθήκευσης αποτελέσματος)
  - Rs1, Rs2: πηγές (καταχωρητές δεδομένων εισόδου)

# Εντολές μεταφοράς δεδομένων

- Μεταφορά δεδομένων
  - Κατεύθυνση: από (load) ή προς (store) τη μνήμη,
  - Πηγή δεδομένων και προορισμός
  - Μήκος μεταφερόμενης λέξης
  - Παράδειγμα (θεωρητικό):
    - $Rd = mem[addr]$
    - |      |    |      |
|------|----|------|
| load | Rd | addr |
|------|----|------|
    - Στο παράδειγμα η διεύθυνση είναι **απόλυτη** (προσδιορίζεται μέσα στην εντολή)
      - Δεν είναι η χρησιμότερη μορφή διεύθυνσης!

# Εντολές διακλάδωσης

- Ο επεξεργαστής
- ISA
- Κατηγορίες εντολών

- Διακλάδωση
  - Με ή χωρίς συνθήκη
    - `bne R1, R2, +8` // branch if not R1==R2
  - Σε απόλυτη διεύθυνση
    - `jump 0xFF97DE00`
  - Σχετικά ως προς την τρέχουσα θέση (**offset**)
    - `jump +130` // offset = +130
    - Ο παραγόμενος κώδικας μπορεί να τοποθετηθεί οπουδήποτε στη μνήμη

bne	R1	R2	+8
-----	----	----	----

# Μέθοδοι προσπέλασης μνήμης

- Τουλάχιστον κάποιες εντολές προσπελαύνουν τη μνήμη
  - για ανάγνωση ή εγγραφή δεδομένων
  - Πώς σχηματίζεται η διεύθυνση προσπέλασης;
  - Η γενική ιδέα: υποβοήθηση του λογισμικού
  - Διαφορετικός σχηματισμός διεύθυνσης για
    - Τοπικές μεταβλητές
    - Δείκτες (έμμεση προσπέλαση)
    - Στατικά δεδομένα
    - Διάσχιση πινάκων
    - (Σταθερές τιμές)
- Υποστήριξη ανάλογα με αρχιτεκτονική



# Μέθοδοι προσπέλασης μνήμης

- Στο σχηματισμό της διεύθυνσης μνήμης μπορούν να συμμετέχουν:
  - Απόλυτες τιμές διεύθυνσης
  - Καταχωρητές
  - Σταθερές τιμές μετατόπισης (offsets)

<i>displacement</i>	mem[offs+reg]	πιθανή χρήση ↓ τοπικές
<i>register indirect</i>	mem[reg]	δείκτες
<i>indexed</i>	mem[reg1+reg2]	πίνακες
<i>direct</i>	mem[addr]	στατικές
<i>memory indirect</i>	mem[mem[reg]]	*δείκτες
<i>auto-increment</i>	mem[reg++]	πίνακες
<i>scaled</i>	mem[offs+reg1+reg2*d]	πίνακες