

# Αρχιτεκτονική Υπολογιστών



Γιώργος Δημητρίου

Ενότητα 11<sup>η</sup>:

Εισαγωγή σε

Παράλληλες Αρχιτεκτονικές

# Παράλληλη Επεξεργασία

- Επίπεδο Παραλληλισμού
  - Από εντολές έως ανεξάρτητες διεργασίες
- Οργανώσεις Παράλληλων Αρχιτεκτονικών
  - Συμμετρικοί, κατανεμημένης μνήμης
- Μέθοδοι Επικοινωνίας και Δίκτυα
- Μέθοδοι Συγχρονισμού
- Διανυσματικοί Πολυεπεξεργαστές
- Πολλαπλές Ροές Ελέγχου

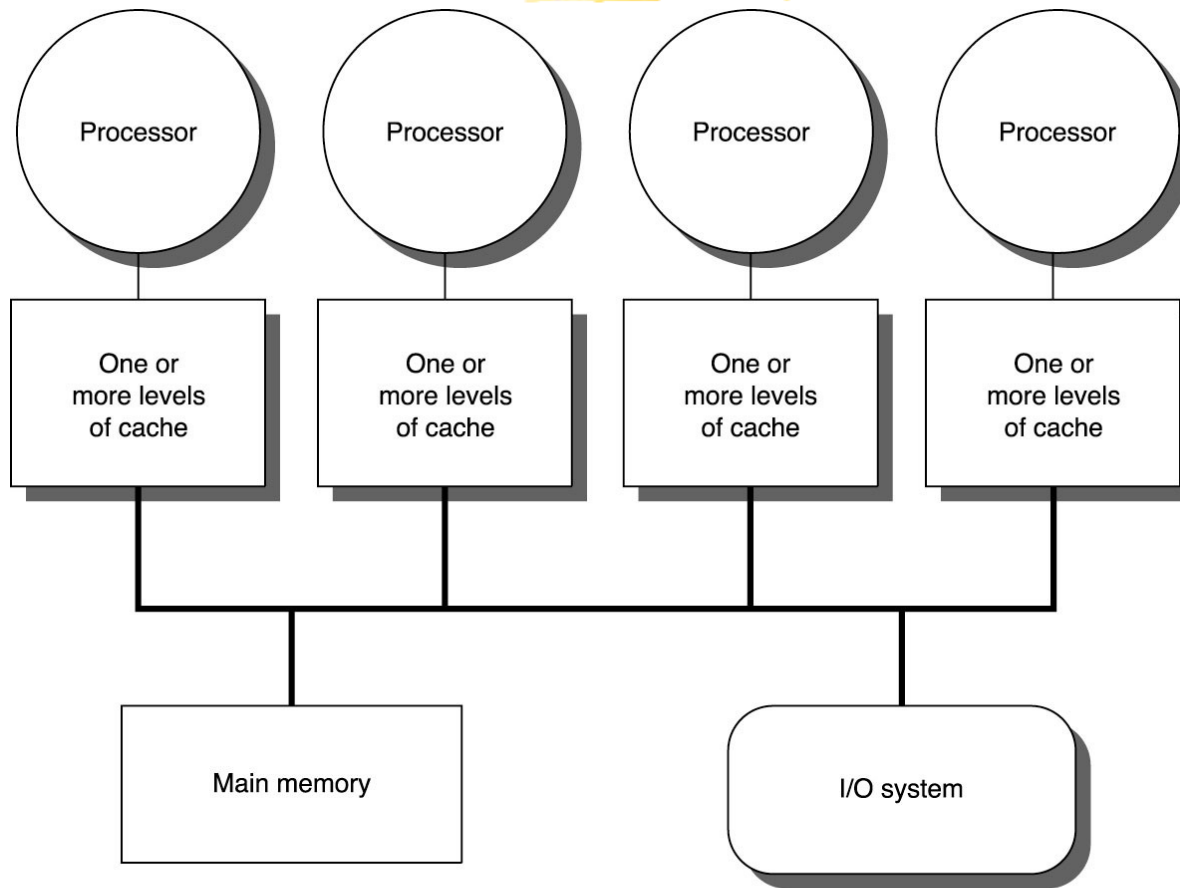
# Επίπεδο Παραλληλισμού

- ILP
  - Μερική επικάλυψη, εκτέλεση εκτός σειράς, υπερβαθμωτή εκτέλεση και εκτέλεση VLIW
- Πολλαπλές ροές ελέγχου
  - Από την ίδια ή διαφορετικές διεργασίες
  - Εκμετάλλευση παραλληλισμού βρόχων και εντολών υψηλού επιπέδου
  - Διανυσματικοί υπολογιστές
- Πολλαπλές διεργασίες

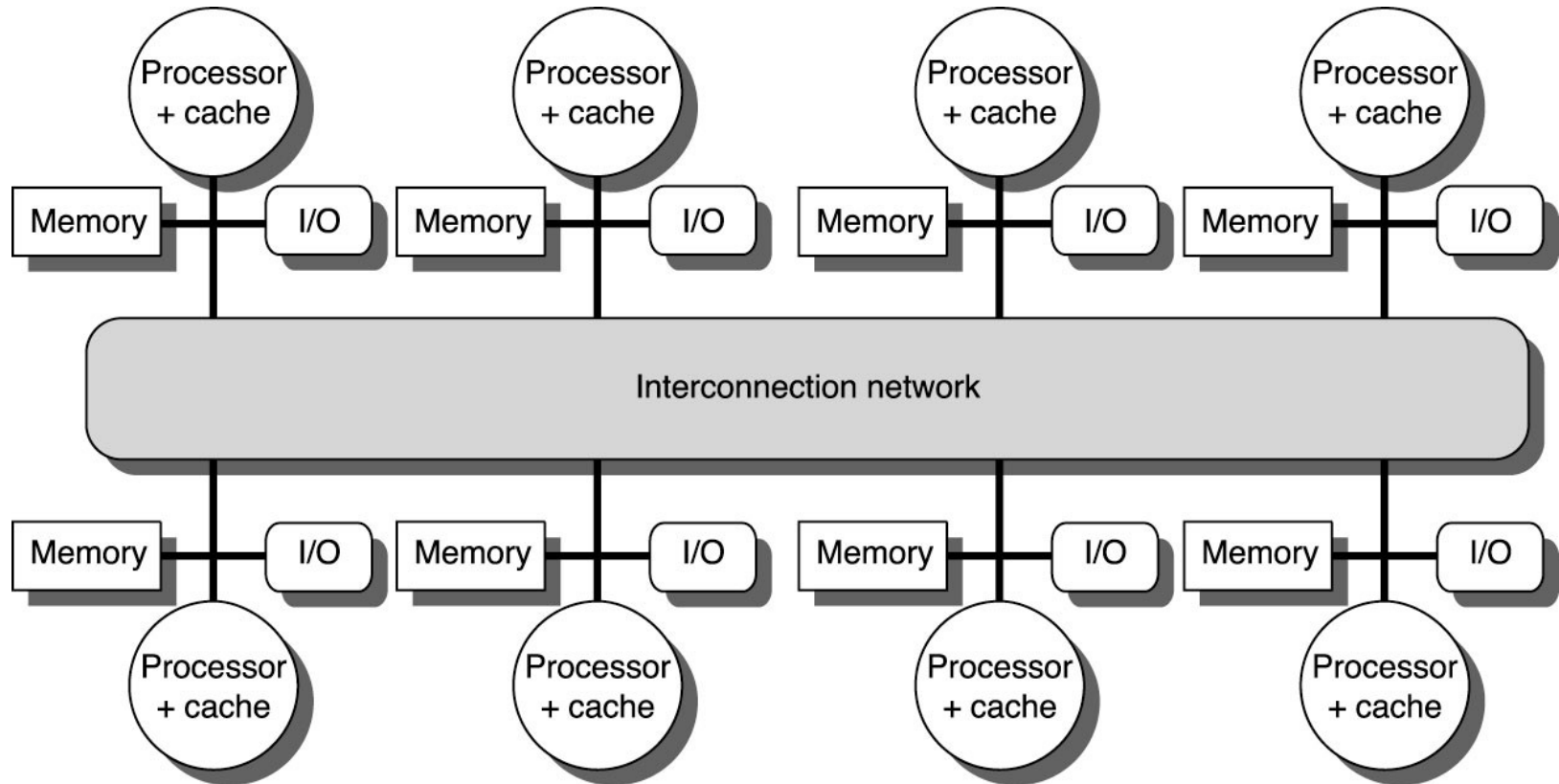
# Πολυεπεξεργαστές & Πολυυπολογιστές

- Πολυεπεξεργαστές: Πολλαπλοί επεξεργαστές που συνδέονται στον ίδιο χώρο διευθύνσεων μνήμης
  - Κοινή μνήμη
- Πολυυπολογιστές: Ανεξάρτητοι υπολογιστές συνδεδεμένοι μεταξύ τους
  - Κατανεμημένη μνήμη
  - Κατανεμημένη κοινή μνήμη: πολυυπολογιστές που εμφανίζονται ως πολυεπεξεργαστές!

# Πολυεπεξεργαστές



# Πολυπολογιστές



# Τρόποι Επικοινωνίας & Δίκτυα

- Πολυπολογιστές: Επικοινωνία με μετάδοση μηνυμάτων
- Πολυεπεξεργαστές: Επικοινωνία με εντολές φόρτωσης/αποθήκευσης
  - Συμμετρικοί πολυεπεξεργαστές (SMP ή UMA)
  - NUMA πολυεπεξεργαστές
- Διασύνδεση με διάφορες τοπολογίες:
  - Αρτηρία, πλέγμα, υπερκύβος, Ω-δίκτυα

# Ταξινόμηση Flynn

- SISD (Single Instruction - Single Data)
  - Οι μονοεπεξεργαστές
- SIMD (Single Instruction - Multiple Data)
  - Οι διανυσματικοί επεξεργαστές
- MISD (Multiple Instruction - Single Data)
  - Μηχανές ροής δεδομένων
- MIMD (Multiple Instruction - Multiple Data)
  - Οι συνηθισμένοι παράλληλοι επεξεργαστές



# Προβλήματα Μνήμης

- Συνοχή κρυφών μνημών
  - Το πρόβλημα των εγγραφών και πώς αυτές γίνονται ορατές σε άλλους επεξεργαστές
- Συνέπεια μνήμης
  - Το πρόβλημα των εγγραφών και πότε αυτές γίνονται ορατές σε άλλους επεξεργαστές

# Συνοχή Κρυφών Μνημών

- Το πρόβλημα:
  - Η διασύνδεση πολλαπλών κρυφών μνημών σε κοινή μνήμη (κρυφή ή κύρια) επόμενου επίπεδου ιεραρχίας μνήμης
  - Οι εγγραφές έναντι των αναγνώσεων και οι τιμές που βλέπουν οι τελευταίες
- Οι λύσεις:
  - Έλεγχος συνοχής με βάση πρωτόκολλα
  - Υλοποίηση σε υλικό ή λογισμικό

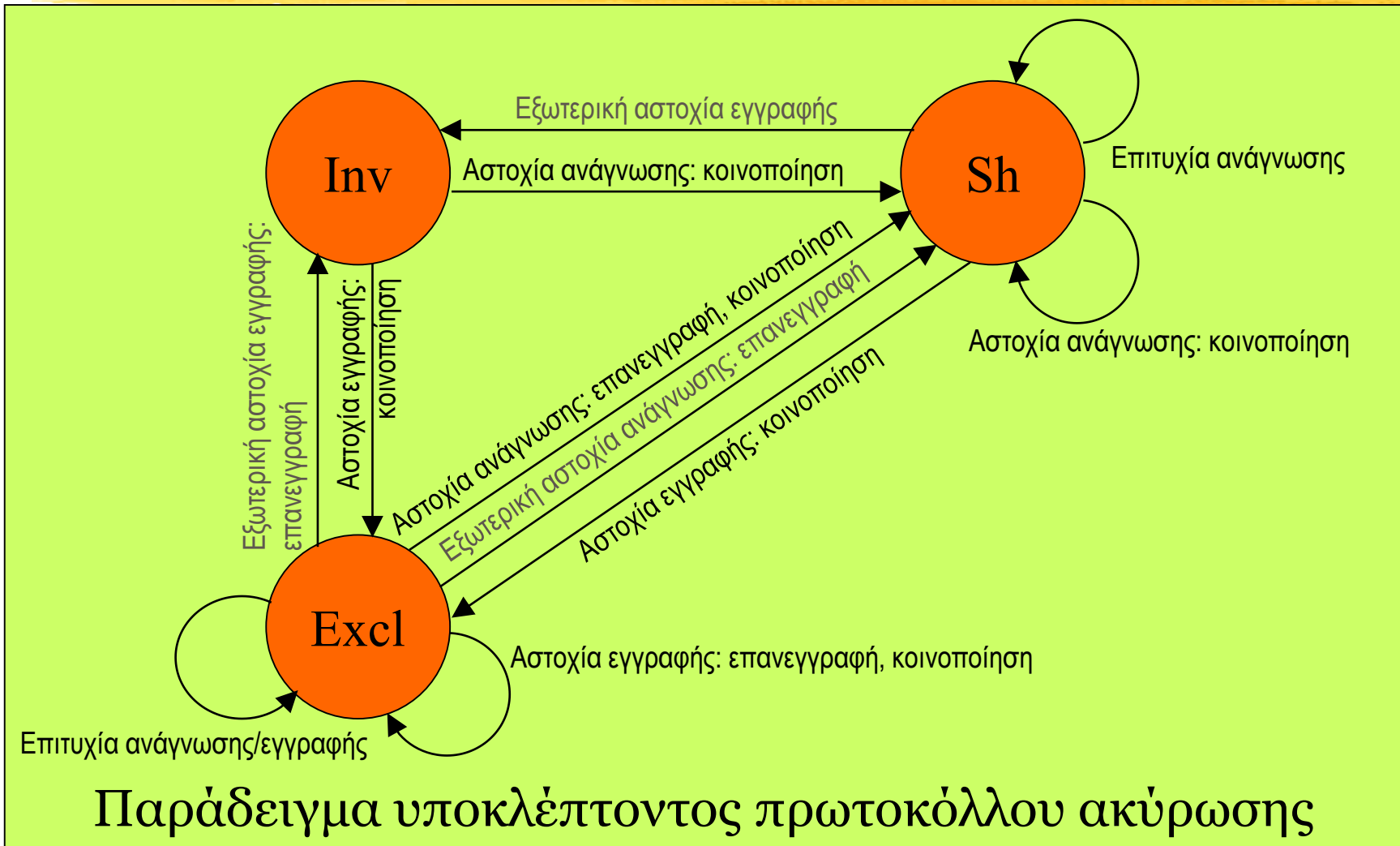
# Συνοχή Κρυφών Μνημών

- Ορισμός: Ένα σύστημα είναι συνεκτικό αν
  - Οι προσπελάσεις μνήμης σε έναν επεξεργαστή συμβαίνουν πάντα με τη σειρά προγράμματος.
  - Μια εγγραφή σε κάποιον επεξεργαστή εμφανίζεται (κάποια στιγμή) σε ανάγνωση άλλου επεξεργαστή.
  - Οι εγγραφές στην ίδια διεύθυνση εμφανίζονται σε αναγνώσεις με τη σειρά που γίνονται.

# Υποκλέπτοντα Πρωτόκολλα

- Για μικρά συστήματα SMP με διασύνδεση μέσω αρτηρίας
- Ακύρωση κατά την εγγραφή (write-invalidate)
  - Κοινοποίηση της διεύθυνσης μόνο
  - Λιγότερη επιβάρυνση στην αρτηρία
- Ενημέρωση κατά την εγγραφή (write-update)
  - Κοινοποίηση διεύθυνσης και μπλοκ
  - Περισσότερη επιβάρυνση στην αρτηρία

# Πρωτόκολλο ESI



Παράδειγμα υποκλέπτοντος πρωτοκόλλου ακύρωσης

# Υλοποίηση Πρωτοκόλλων

- Μη ατομικές λειτουργίες
  - Ανίχνευση αστοχίας → Κοινοποίηση
  - Κίνδυνος αδιεξόδου!
- Αρτηρίες διαχωρισμένης προσπέλασης
- Περισσότερες καταστάσεις για μείωση μετακινήσεων
  - Αστοχία εγγραφής όταν το μπλοκ είναι παρόν στην κρυφή μνήμη!?

# Ψευδής Κοινή Χρήση

- Το πρόβλημα κοινής χρήσης του ίδιου μπλοκ μνήμης, αλλά για διαφορετικά δεδομένα (false sharing):

cc	P1	P2
1	Write x1	
2		Read x2
3	Write x1	
4		Write x2
5	Read x1	

# Συνέπεια Μνήμης

- Πρόβλημα συνέπειας:

P1: A = 0;	P2: B = 0;
...	...
A = 1;	B = 1;
L1: if (B == 0) ...	L2: if (A == 0) ...

- Αυστηρό μοντέλο: σειριακή συνέπεια
- Χαλαρά μοντέλα συνέπειας:
  - συνέπεια επεξεργαστή
  - συνέπεια μερικής σειράς εγγραφών
  - άλλα μοντέλα



# Συγχρονισμός Παράλληλης Επεξεργασίας

- Η ανάγκη συγχρονισμού
  - Βρόχος doacross
- Μέθοδοι συγχρονισμού
  - Συγχρονισμός μεταξύ δύο παράλληλων επεξεργασιών
  - Φράγμα
- Θέματα υλοποίησης
  - Ατομικές μικρολειτουργίες
  - Υποστήριξη από το σύνολο εντολών

# Ατομικές Μικρολειτουργίες

- Εκτέλεση σε μία εντολή:
  - Test-and-Set
  - Fetch-and-Increment
  - Compare-and-Swap
- Εκτέλεση σε δύο εντολές (ψευδοατομική μικρολειτουργία):
  - LL (Load Linked)
  - SC (Store Conditional)

# Spin Locks

- Διεύθυνση κλειδαριάς: 0(\$8)
- Χρησιμοποιώντας test-and-set:

```
Repeat:  t&set $4, 0 ($8)
         bne $4, $0, Repeat
```

- Χρησιμοποιώντας ll/sc:

```
Repeat:  ori $3, $0, 1
Read:    ll $2, 0 ($8)
         bne $2, $0, Read
         sc $3, 0 ($8)
         beq $3, $0, Repeat
```

# Διανυσματικοί Υπολογιστές

- Παράλληλη επεξεργασία τύπου SIMD:
- Υπολογιστές που εκτελούν διανυσματικές πράξεις
  - Από τον Cray και τον IBM 3090
  - Μέχρι τους πιο πρόσφατους μικροεπεξεργαστές που εκτελούν διανυσματικές εντολές για media processing

# Επεξεργασία Πολλαπλών Ροών Ελέγχου

- Ροές ελέγχου (ή νήματα) από την ίδια ή διαφορετικές διεργασίες
- Διάφορα επίπεδα παραλληλισμού
- Blocked και interleaved
- Συνδυασμός υπερβαθμωτής εκτέλεσης και εκτέλεσης πολλαπλών ροών ελέγχου
- Παραγωγή κώδικα πολλαπλών ροών ελέγχου!