

---

# Προηγμένα Θέματα Αρχιτεκτονικής Συστημάτων

## Παράλληλη Επεξεργασία (μέρος 1<sup>ο</sup>)

Διδάσκων: Δημόκας Νικόλαος

# Περιεχόμενα

---

- **Παράλληλη Επεξεργασία**
  - Οργάνωση Πολλαπλών Επεξεργαστών
  - Συμμετρικοί Πολυεπεξεργαστές
  - Συνοχή Κρυφής Μνήμης και το Πρωτόκολλο MESI
  - Πολυνημάτωση και Chip Πολυεπεξεργαστών
  - Συστάδες
  - Μη Ομοιόμορφη Προσπέλαση Μνήμης
  - Υπολογισμοί Διανυσμάτων

# Εισαγωγή

---

- Παραδοσιακά ο υπολογιστής θεωρείται μία ακολουθιακή μηχανή
- Οι γλώσσες προγραμματισμού απαιτούν την υλοποίηση των αλγορίθμων με τη μορφή ακολουθιών εντολών
- Οι επεξεργαστές κατά την εκτέλεση ενός προγράμματος εκτελούν τις εντολές μηχανής **με τη σειρά, μία κάθε φορά**
- Κάθε εντολή εκτελείται με μία σειρά λειτουργιών
  - i. Ανάκληση εντολής
  - ii. Ανάκληση παραγόντων
  - iii. Εκτέλεση λειτουργίας
  - iv. Αποθήκευση αποτελεσμάτων

# Εισαγωγή

---

- Η θεώρηση της ακολουθιακής μηχανής δεν είναι εντελώς πραγματική
- Στο επίπεδο μικρολειτουργιών παράγονται ταυτόχρονα πολλαπλά σήματα ελέγχου
- Η διασωλήνωση των εντολών στο επίπεδο επικάλυψης των λειτουργιών ανάκλησης και εκτέλεσης είναι κάτι που συμβαίνει πολλά χρόνια
- Επομένως έχουμε εκτέλεση **Παράλληλων Λειτουργιών**

# Εισαγωγή

---

- Η πολυβαθμωτή οργάνωση εκμεταλλεύεται τον παραλληλισμό σε επίπεδο εντολής
- **Υπερβαθμωτή μηχανή:** Η μηχανή στην οποία υπάρχουν πολλαπλές μονάδες εκτέλεσης εντός ενός επεξεργαστή
- Στην υπερβαθμωτή μηχανή οι πολλαπλές μονάδες εκτέλεσης μπορούν να εκτελούν **παράλληλα** πολλαπλές εντολές του ίδιου προγράμματος

# Εισαγωγή

---

- Η αύξηση της απόδοσης ενός συστήματος επιτυγχάνεται με τη χρήση **πολλαπλών επεξεργαστών** που έχουν τη δυνατότητα να **εκτελούν προγράμματα παράλληλα**

*Κοινές μορφές οργάνωσης πολλαπλών επεξεργαστών*

- 1. Συμμετρικοί Πολυεπεξεργαστές (Symmetric Multiprocessors - SMP)**
- 2. Συστάδες (Clusters)**
- 3. Συστήματα Μη Ομοιόμορφης Προσπέλασης Μνήμης (Nonuniform Memory Access - NUMA)**

# Εισαγωγή

---

- Σύστημα συμμετρικών πολυεπεξεργαστών: Αποτελείται από **πολλούς παρόμοιους επεξεργαστές στον ίδιο Η/Υ** που διασυνδέονται με ένα δίαυλο ή διάταξη μεταγωγέων
- Κάθε *επεξεργαστής* έχει τη *δική του κρυφή μνήμη*
- Μία *γραμμή δεδομένων* μπορεί να βρίσκεται *σε περισσότερες από μία κρυφές μνήμες*.
  - Εάν μία γραμμή τροποποιηθεί τότε η κύρια μνήμη και οι άλλες κρυφές μνήμες έχουν μη έγκυρα δεδομένα
- Το **σημαντικότερο πρόβλημα ενός SMP** είναι η **διατήρηση της συνοχής της κρυφής μνήμης** (Cache Coherence)
  - Πρωτόκολλα συνοχής κρυφής μνήμης για διατήρηση της συνοχής

# Εισαγωγή

---

- **Πολυεπεξεργασία σε επίπεδο Chip (Chip Multiprocessing):** Περισσότεροι του ενός επεξεργαστές υλοποιούνται σε ένα chip
- Τεχνική σχεδίασης: Δημιουργούμε πολλά αντίγραφα ενός απλού επεξεργαστή ώστε ο **επεξεργαστής (multi core)** να μπορεί να εκτελεί **πολλά νήματα ταυτόχρονα**
- Ο επεξεργαστής ονομάζεται και **επεξεργαστής πολλαπλής νημάτωσης (Multithreaded Processor)**



# Εισαγωγή

---

- **Συστάδα:** Ένα σύνολο από **διασυνδεδεμένους αυτόνομους υπολογιστές** που εργάζονται μαζί ως μία μηχανή
  - Αυτόνομος υπολογιστής σημαίνει ότι κάθε υπολογιστής μπορεί να λειτουργεί από μόνος του ανεξάρτητα της συστάδας
- Σύστημα **Μη Ομοιόμορφης Προσπέλασης Μνήμης:**  
Αποτελεί έναν **πολυεπεξεργαστή με κοινόχρηστη μνήμη**
  - Ο **χρόνος προσπέλασης** μιας λέξης της μνήμης από έναν επεξεργαστή **ποικίλει** ανάλογα με το που βρίσκεται η λέξη

# Οργάνωση Πολλαπλών Επεξεργαστών

---

## Κατηγορίες Υπολογιστικών Συστημάτων Παράλληλης Επεξεργασίας

- **Ροή μονής εντολής, μονών δεδομένων** (Single Instruction, Single Data - SISD): Ένας επεξεργαστής εκτελεί μία εντολή πάνω σε δεδομένα που βρίσκονται σε μία μνήμη
  - Παράδειγμα: Μονοεπεξεργαστές
- **Ροή μονής εντολής, πολλαπλών δεδομένων** (Single Instruction, Multiple Data - SIMD): Κάθε εντολή εκτελείται από διαφορετικούς επεξεργαστές πάνω σε ένα διαφορετικό σύνολο δεδομένων
  - Παράδειγμα: Επεξεργαστές Διανυσμάτων και Πινάκων

# Οργάνωση Πολλαπλών Επεξεργαστών

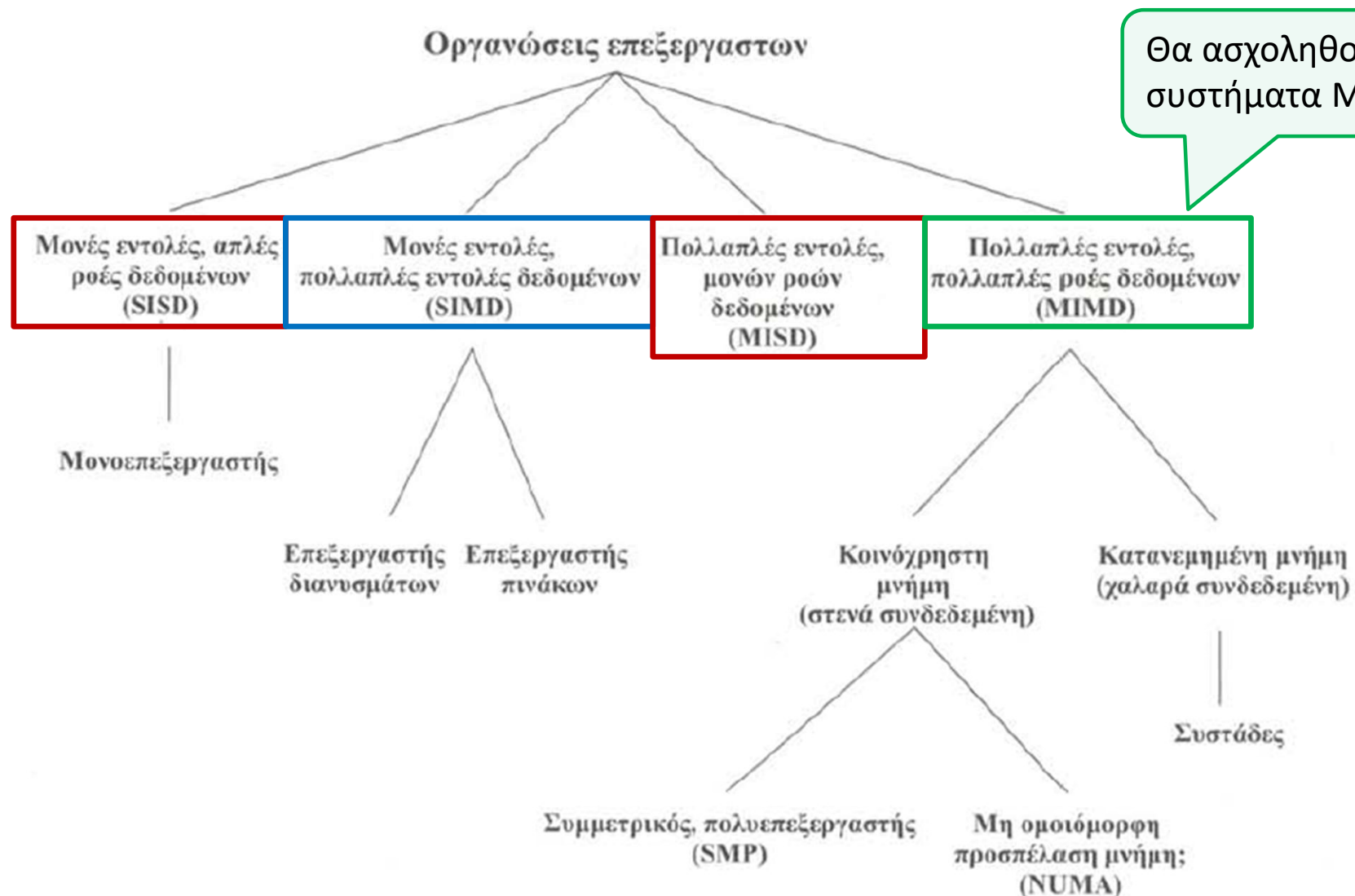
---

## Κατηγορίες Υπολογιστικών Συστημάτων Παράλληλης Επεξεργασίας

- **Ροή πολλαπλής εντολής, μονών δεδομένων** (Multiple Instruction, Single Data - MISD): Μία ακολουθία δεδομένων μεταφέρεται σε ένα σύνολο επεξεργαστών και κάθε επεξεργαστής εκτελεί μία διαφορετική ακολουθία εντολών
  - Δεν υλοποιείται εμπορικά
- **Ροή πολλαπλής εντολής, πολλαπλών δεδομένων** (Multiple Instruction, Multiple Data - SIMD): Πολλαπλοί επεξεργαστές εκτελούν ταυτόχρονα διαφορετικές ακολουθίες εντολών σε διαφορετικά σύνολα δεδομένων
  - Παράδειγμα: Συμμετρικοί Πολυεπεξεργαστές, Συστάδες, NUMA

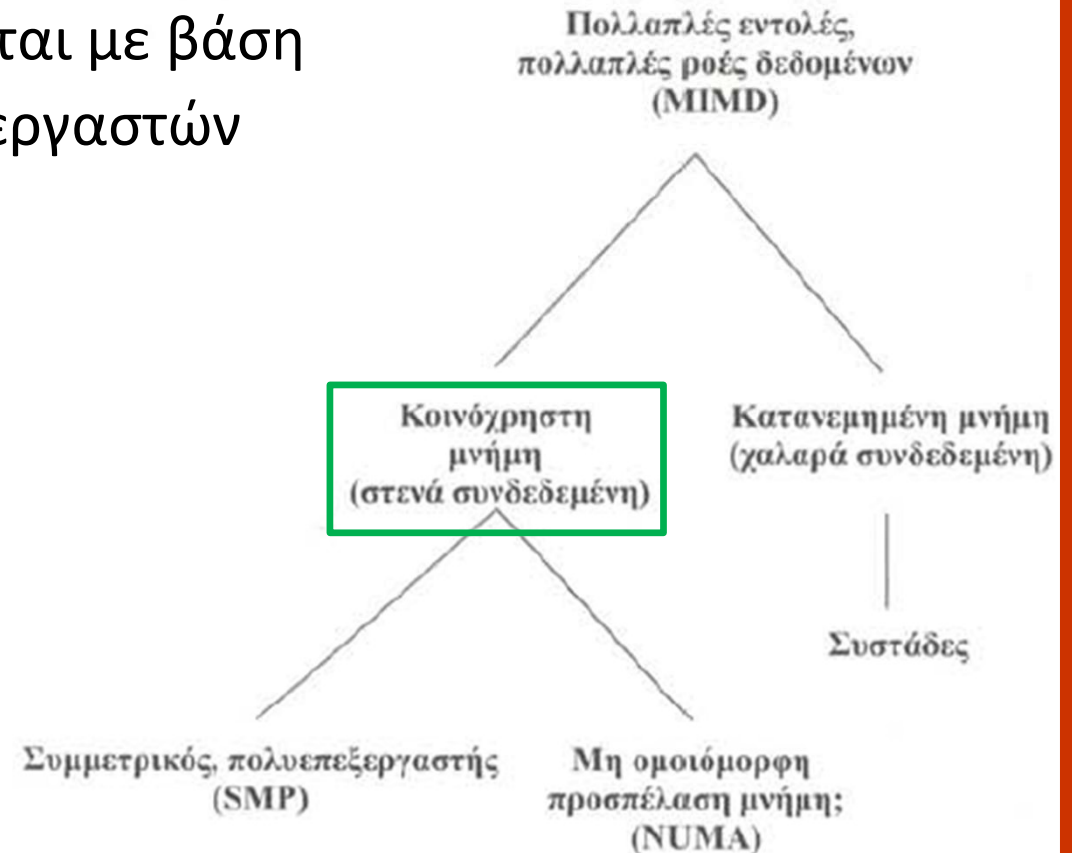
# Οργάνωση Πολλαπλών Επεξεργαστών

## Κατηγορίες Υπολογιστικών Συστημάτων Παράλληλης Επεξεργασίας



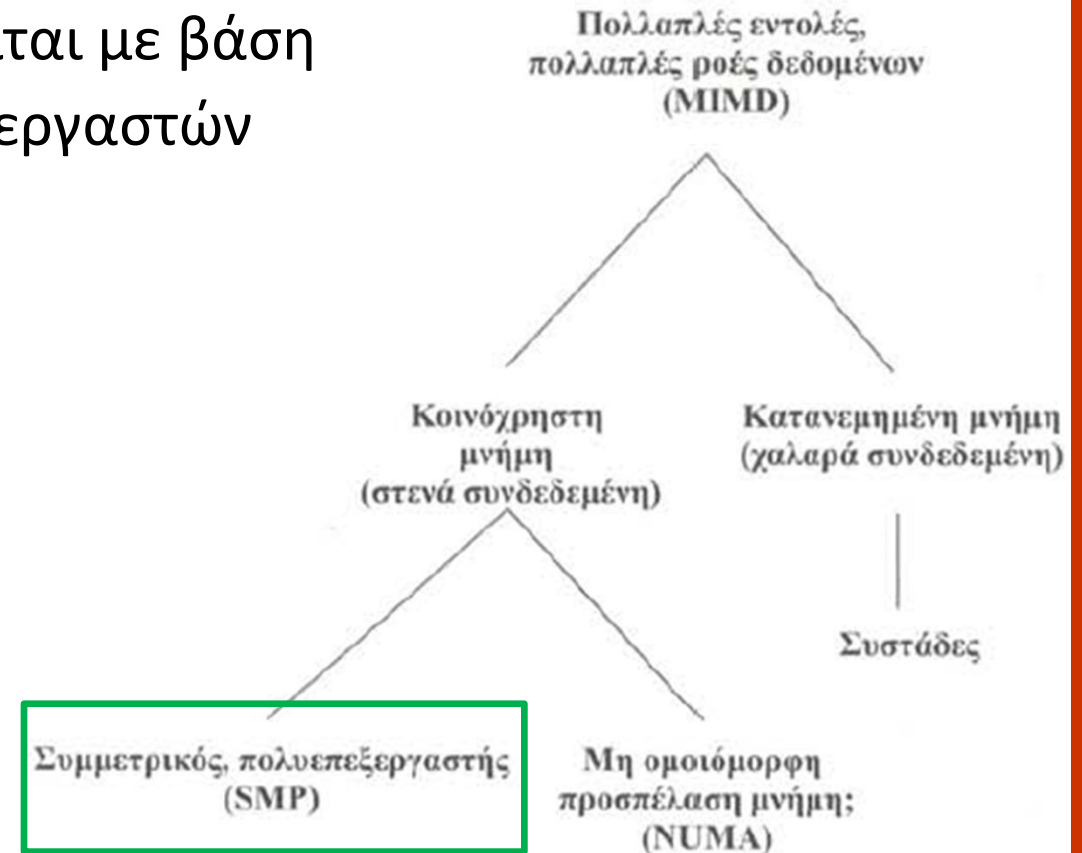
# Οργάνωση Πολλαπλών Επεξεργαστών

- Μία οργάνωση MIMD υποδιαιρείται με βάση τον τρόπο επικοινωνίας των επεξεργαστών
- Αν οι επεξεργαστές μοιράζονται μία κοινή μνήμη τότε προγράμματα και δεδομένα αποθηκεύονται σε αυτή
- Οι επεξεργαστές επικοινωνούν μεταξύ τους μέσω της μνήμης



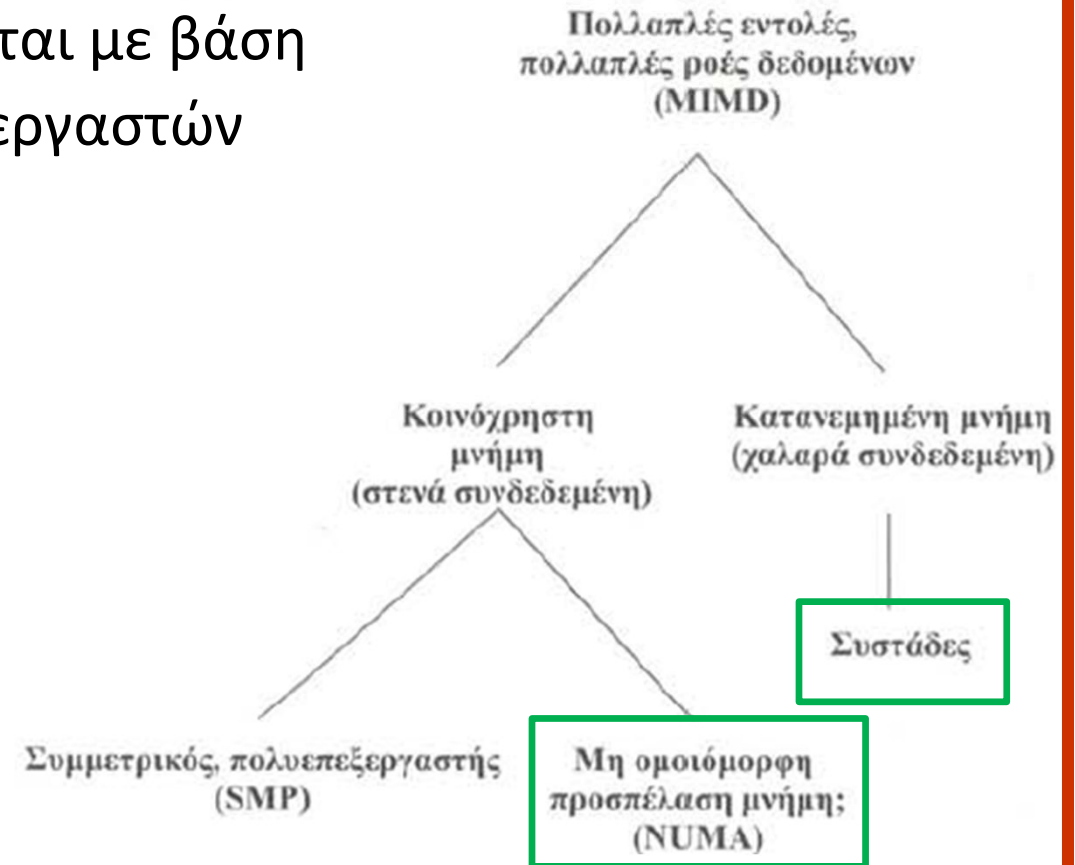
# Οργάνωση Πολλαπλών Επεξεργαστών

- Μία οργάνωση MIMD υποδιαιρείται με βάση τον τρόπο επικοινωνίας των επεξεργαστών
- Η οργάνωση SMP περιλαμβάνει επεξεργαστές που μοιράζονται μνήμη/ες με τη χρήση ενός κοινόχρηστου διαύλου
- Ο χρόνος προσπέλασης κάθε περιοχής της μνήμης είναι ίδιος για όλους τους επεξεργαστές



# Οργάνωση Πολλαπλών Επεξεργαστών

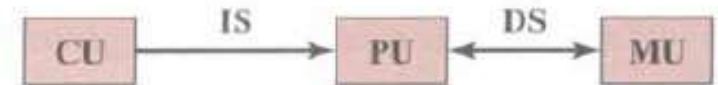
- Μία οργάνωση MIMD υποδιαιρείται με βάση τον τρόπο επικοινωνίας των επεξεργαστών
- Στην οργάνωση NUMA ο χρόνος προσπέλασης σε διαφορετικές περιοχές της μνήμης ποικίλει μεταξύ των επεξεργαστών
- Στην οργάνωση Cluster ανεξάρτητοι μονοεπεξεργαστές ή πολλά SMP διασυνδέονται
  - Επικοινωνία μέσω σταθερών μονοπατιών ή δικτύου



# Οργάνωση Πολλαπλών Επεξεργαστών

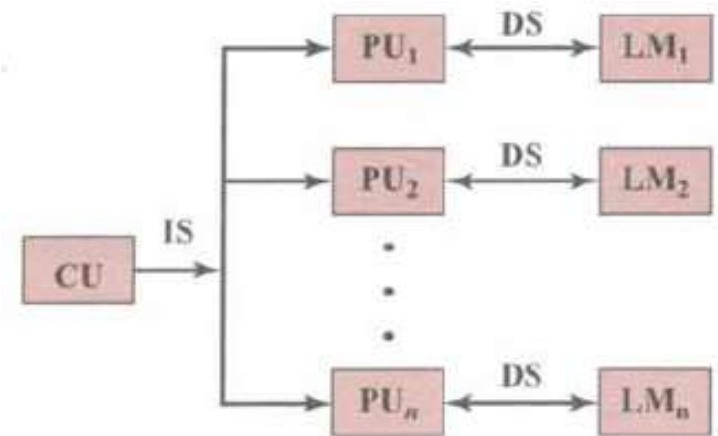
- SISD

1. Μία μονάδα ελέγχου (CU) παρέχει μία ροή εντολών (IS) σε μία μονάδα επεξεργασίας (PU)
2. Η μονάδα επεξεργασίας (PU) χρησιμοποιεί μία ροή δεδομένων (DS) που βρίσκεται στη μονάδα μνήμης (MU)



- SIMD

1. Μία μονάδα ελέγχου (CU) παρέχει μία ροή εντολών (IS) σε πολλές PU
2. Κάθε μονάδα επεξεργασίας (PU) έχει τη δική της μνήμη ή μπορεί να υπάρχει κοινή μνήμη





# Οργάνωση Πολλαπλών Επεξεργαστών

- MIMD

- Υπάρχουν πολλαπλές μονάδες ελέγχου (CU) και κάθε μία παρέχει μία ροή εντολών (IS) στη δική της μονάδα επεξεργασίας (PU)
- Η μνήμη μπορεί να είναι:
  - Κοινόχρηστη
  - Κατανεμημένη

CU = Μονάδα ελέγχου

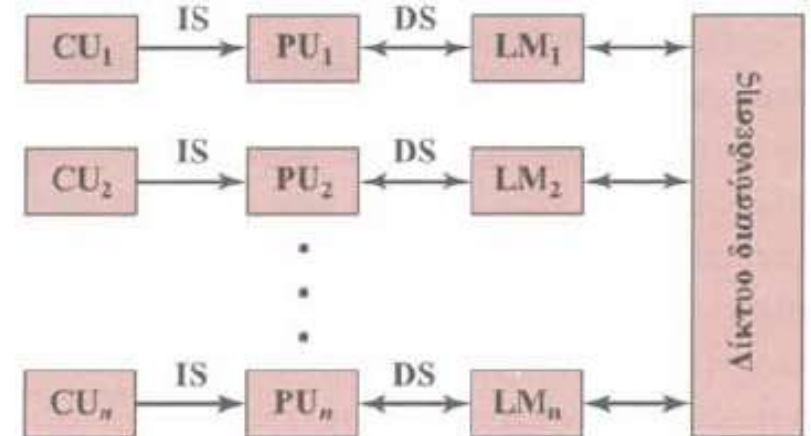
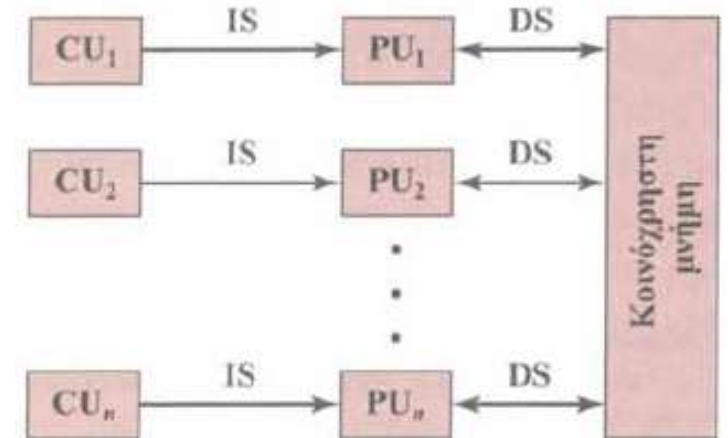
IS = Ροή εντολών

PU = Μονάδα επεξεργασίας

DS = Ροή δεδομένων

MU = Μονάδα μνήμης

LM = Τοπική μνήμη



# Συμμετρικοί Πολυεπεξεργαστές

---

- Χαρακτηριστικά Συμμετρικού Πολυεπεξεργαστή
  - Δύο ή περισσότεροι παρόμοιοι επεξεργαστές
  - Οι επεξεργαστές
    - Μοιράζονται την ίδια κοινή μνήμη
    - Μοιράζονται τις μονάδες E/E
    - Συνδέονται με ένα δίαυλο
    - Ο χρόνος προσπέλασης της μνήμης είναι περίπου ο ίδιος
  - Οι επεξεργαστές μοιράζονται την προσπέλαση σε μονάδες E/E είτε μέσα από τα ίδια κανάλια είτε από διαφορετικά
  - Όλοι οι επεξεργαστές μπορούν να εκτελέσουν τις ίδιες λειτουργίες
  - Το σύστημα ελέγχεται από ένα λειτουργικό σύστημα που παρέχει διασύνδεση μεταξύ των επεξεργαστών και των προγραμμάτων τους σε επίπεδο εργασίας, αρχείων και δεδομένων

# Συμμετρικοί Πολυεπεξεργαστές

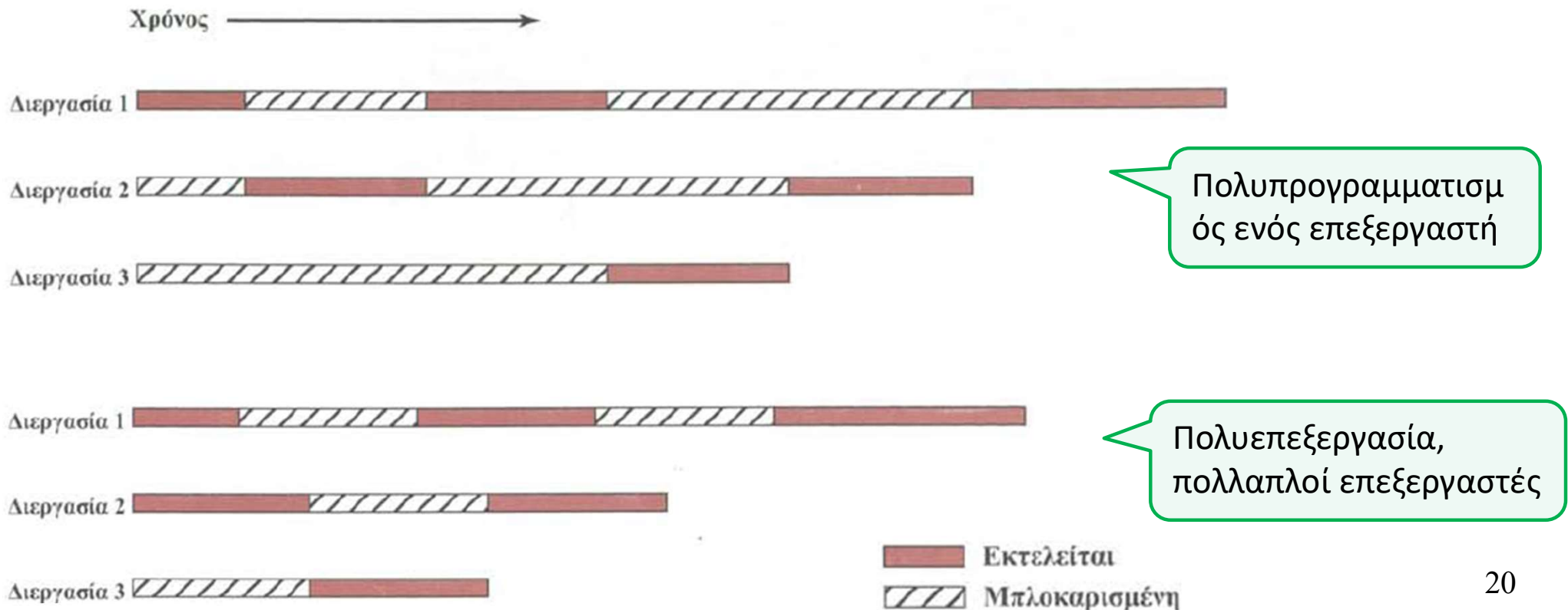
---

- **Συστάδες:** η φυσική μονάδα συναλλαγής είναι συνήθως ένα *μήνυμα*, ή ένα *ολόκληρο αρχείο*.
- **Συμμετρικοί πολυεπεξεργαστές:** η μονάδα συναλλαγής μπορεί να είναι διαφορετικά *στοιχεία δεδομένων*
  - Μπορεί να υπάρχει σημαντική συνεργασία ανάμεσα στις διεργασίες.
- Το **λειτουργικό σύστημα** ενός συστήματος συμμετρικών πολυεπεξεργαστών *χρονοδρομολογεί* τις *διεργασίες* ή *νήματα* (threads) και τις *μοιράζει* στους *επεξεργαστές*.

# Συμμετρικοί Πολυεπεξεργαστές

*Πλεονεκτήματα σε σχέση με σύστημα ενός επεξεργαστή*

- **Απόδοση:** Αν η εργασία οργανωθεί κατά τέτοιο τρόπο ώστε η υλοποίηση της να είναι παράλληλη, τότε ένα σύστημα παράλληλων επεξεργαστών θα είναι πιο αποτελεσματικό



# Συμμετρικοί Πολυεπεξεργαστές

---

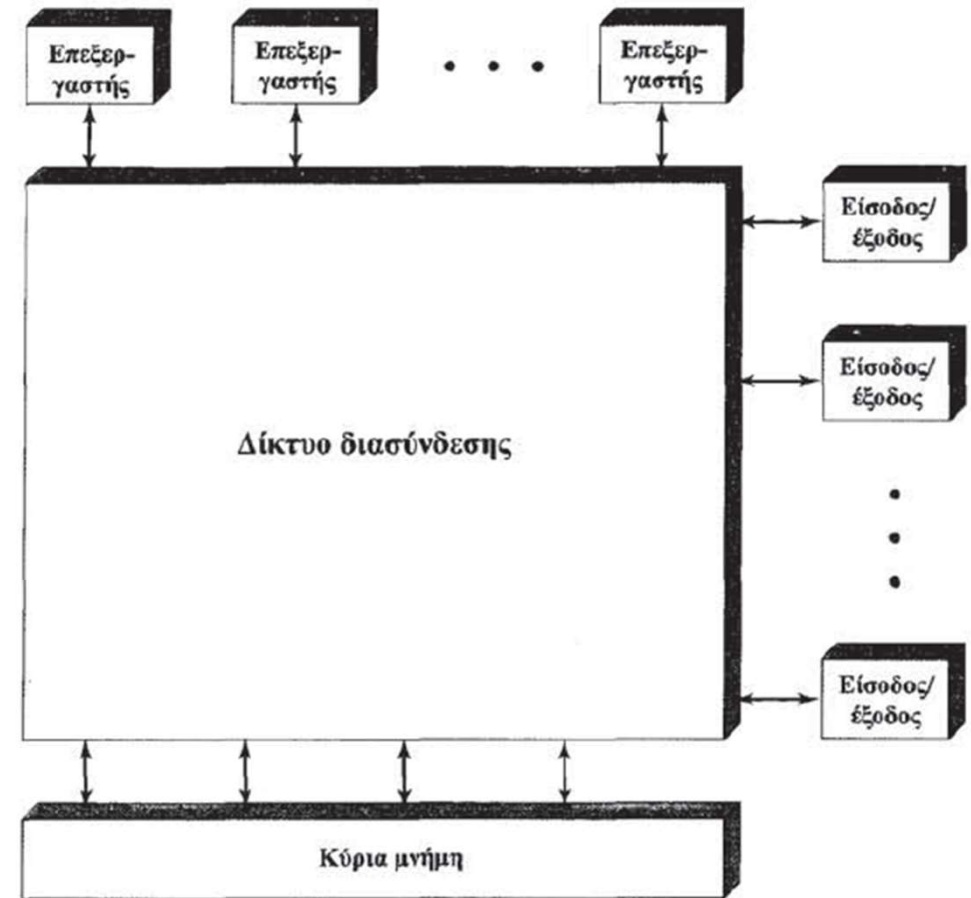
*Πλεονεκτήματα σε σχέση με σύστημα ενός επεξεργαστή*

- **Διαθεσιμότητα:** Η βλάβη ενός επεξεργαστή δε σταματά τη μηχανή. Το σύστημα συνεχίζει να λειτουργεί με μειωμένη απόδοση
- **Αυξημένη Απόδοση:** Η απόδοση του συστήματος μπορεί να βελτιωθεί με τη προσθήκη ενός επεξεργαστή
- **Κλιμάκωση:** Η αύξηση των υπολογιστικών δυνατοτήτων (επεξεργαστών) συνεπάγεται αύξηση της απόδοσης

# Συμμετρικοί Πολυεπεξεργαστές

## Οργάνωση Συστήματος Πολυεπεξεργασίας

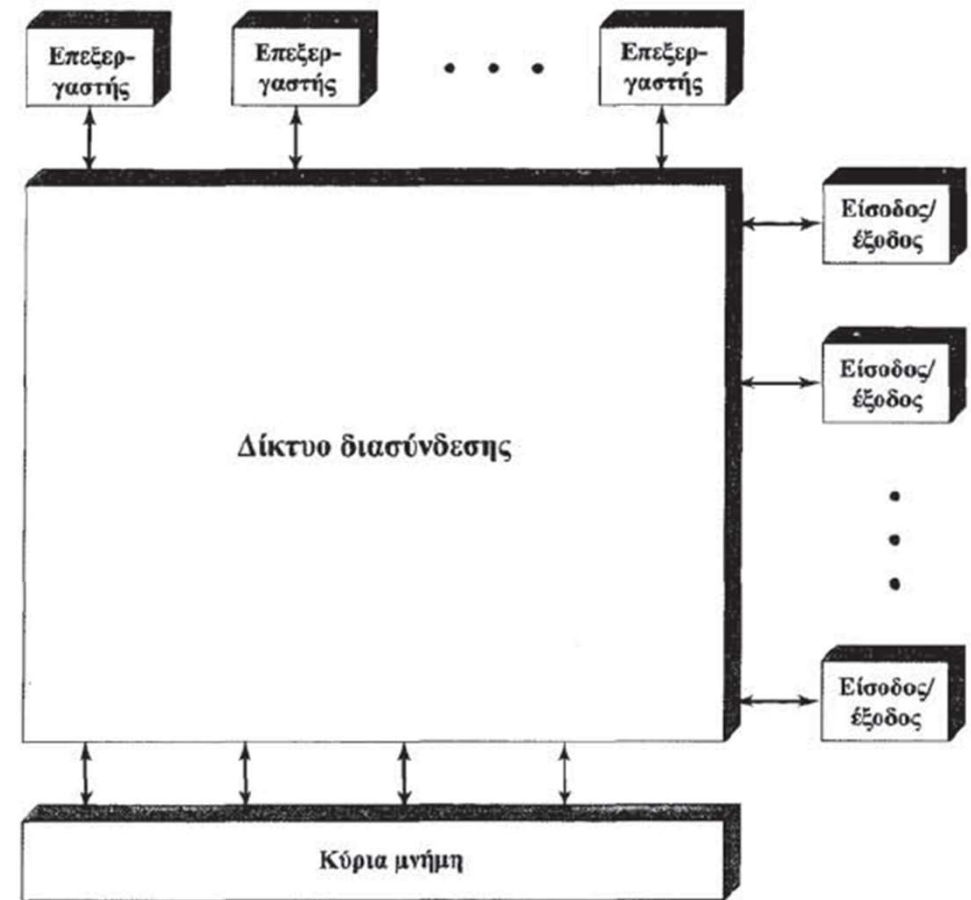
- $\geq 2$  επεξεργαστές
- Κάθε επεξεργαστής έχει
  - Μονάδα Ελέγχου (**Control Unit**)
  - **ALU**
  - Καταχωρητές (**Registers**)
  - Μία ή περισσότερες **κρυφές μνήμες**
- Κάθε επεξεργαστής μέσα από ένα μηχανισμό διασύνδεσης προσπελάνει:
  - **Κοινόχρηστη μνήμη**
  - **Μονάδες Ε/Ε**



# Συμμετρικοί Πολυεπεξεργαστές

## Οργάνωση Συστήματος Πολυεπεξεργασίας

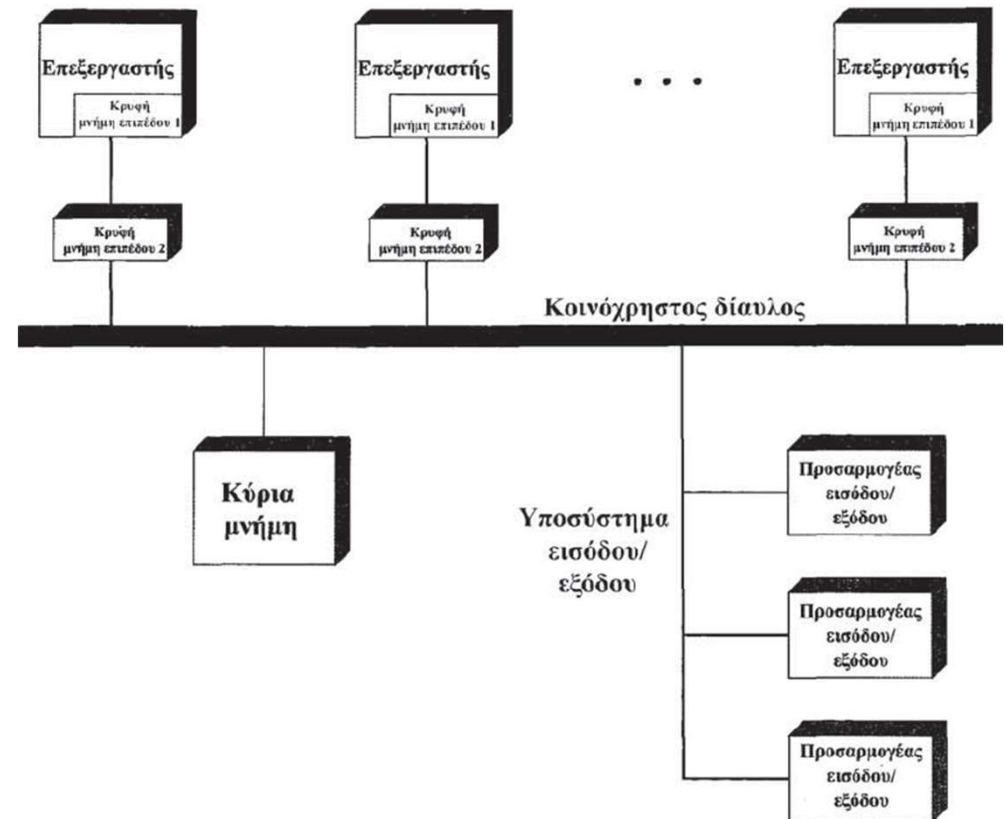
- Υπάρχει η δυνατότητα **άμεσης ανταλλαγής σημάτων** από τους επεξεργαστές
- Η **μνήμη** είναι οργανωμένη με τέτοιο τρόπο ώστε να **επιτρέπει πολλαπλές ταυτόχρονες προσπελάσεις** σε ξεχωριστά μπλοκ
- Ορισμένες φορές κάθε επεξεργαστής διαθέτει:
  - Δική του μνήμη και
  - Κανάλια Ε/Ε



# Συμμετρικοί Πολυεπεξεργαστές

## Οργάνωση Συμμετρικού Πολυεπεξεργαστή

- Ο δίαυλος διαμοιραζόμενου χρόνου αποτελεί τον πιο απλό μηχανισμό κατασκευής ενός **συστήματος πολυεπεξεργασίας**
- Η δομή και οι διεπαφές είναι **ίδιες με ένα σύστημα ενός επεξεργαστή**
- Ο δίαυλος αποτελείται από:
  - γραμμές ελέγχου,
  - διευθύνσεων και
  - δεδομένων





# Συμμετρικοί Πολυεπεξεργαστές

---

- Για να διευκολυνθούν οι μεταφορές άμεσης προσπέλασης μνήμης από τους επεξεργαστές E/E, παρέχονται τα ακόλουθα:
  - **Διευθυνσιοδότηση:** Παρέχεται η δυνατότητα **διάκρισης των μονάδων πάνω στο δίαυλο**, ώστε να ορίζεται η αφετηρία και ο προορισμός των δεδομένων
  - **Διαιτησία:** Παρέχεται ένας **μηχανισμός διαιτησίας** ώστε να εξυπηρετούνται οι αιτήσεις για τον έλεγχο του διαύλου,
    - **τεχνική βασισμένη σε προτεραιότητες**
  - **Διαμοιρασμός χρόνου:** Όταν μία μονάδα ελέγχει το δίαυλο, οι άλλες μονάδες είναι κλειδωμένες και **αναστέλλουν τη λειτουργία τους έως ότου λάβουν ξανά πρόσβαση στο δίαυλο**

# Συμμετρικοί Πολυεπεξεργαστές

---

- Σε ένα σύστημα συμμετρικών πολυεπεξεργαστών υπάρχουν πολλαπλοί επεξεργαστές και πολλαπλές μονάδες E/E που προσπαθούν να αποκτήσουν πρόσβαση σε μία ή περισσότερες μονάδες μνήμης μέσα από το δίαυλο.
- ❖ **Χαρακτηριστικά της οργάνωσης του διαύλου**
  - **Απλότητα:** Η πιο απλή προσέγγιση της οργάνωσης πολλαπλών επεξεργαστών.
    - Η λογική της διευθυνσιοδότησης, της διαιτησίας και του διαμοιρασμού του χρόνου κάθε επεξεργαστή είναι όπως σε ένα σύστημα απλού επεξεργαστή.
  - **Ευελιξία:** Εύκολο να επεκτείνουμε το σύστημα συνδέοντας περισσότερους επεξεργαστές στο δίαυλο.
  - **Αξιοπιστία:** Η βλάβη οποιασδήποτε συνδεδεμένης συσκευής δεν θα πρέπει να προκαλεί βλάβη σε ολόκληρο το σύστημα.

# Συμμετρικοί Πολυεπεξεργαστές

---

- Βασικό μειονέκτημα της οργάνωσης του διαύλου, είναι η **απόδοση**. Όλες οι αναφορές στη μνήμη περνούν μέσα από τον κοινό δίαυλο.
  - Ο χρόνος του κύκλου διαύλου περιορίζει την ταχύτητα του συστήματος
- Βελτίωση της απόδοσης όταν ο κάθε επεξεργαστής να έχει μία κρυφή μνήμη
  - Μεγάλη μείωση του πλήθους των προσπελάσεων του διαύλου
- ❖ *Οι συμμετρικοί πολυεπεξεργαστές έχουν δύο επίπεδα κρυφής μνήμης*

# Συμμετρικοί Πολυεπεξεργαστές

---

❖ *Κρυφή μνήμη εισάγει νέα σχεδιαστικά ζητήματα*

## **Πρόβλημα Συνοχής Κρυφής Μνήμης**

- Κάθε τοπική κρυφή μνήμη περιέχει μία απεικόνιση ενός τμήματος της κύριας μνήμης
- Αν αλλάξει μία λέξη σε μία κρυφή μνήμη, αυτομάτως ακυρώνεται μία λέξη σε μία άλλη κρυφή μνήμη
- Οι επεξεργαστές θα πρέπει να ειδοποιούνται όταν συμβαίνει μία ενημέρωση λέξης

# Συμμετρικοί Πολυεπεξεργαστές

---

## Ζητήματα Σχεδίασης Λειτουργικού Συστήματος σε SMP

- Το λειτουργικό σύστημα ενός συστήματος συμμετρικών πολυεπεξεργαστών διαχειρίζεται τους επεξεργαστές και τους υπόλοιπους πόρους του υπολογιστή
  - Θα εμφανίζεται ως σύστημα πολυπρογραμματισμού μονού επεξεργαστή
- Το λειτουργικό σύστημα πρέπει να χρονοδρομολογήσει την εκτέλεση των διεργασιών και να καταναείμει τους πόρους γιατί:
  - Υπάρχουν ενεργές πολλαπλές εργασίες ή διεργασίες
  - Υπάρχουν εφαρμογές οι οποίες χρησιμοποιούν πολλαπλές διεργασίες ή πολλαπλά νήματα εντός διεργασιών

# Συμμετρικοί Πολυεπεξεργαστές

---

## Ζητήματα Σχεδίασης Λειτουργικού Συστήματος σε SMP

- Βασικά ζητήματα σχεδίασης του Λειτουργικού Συστήματος:
  - Ταυτόχρονες συντρέχουσες διεργασίες: Οι **ρουτίνες του ΛΣ** θα πρέπει να **επανακαλούνται**, ώστε διαφορετικοί επεξεργαστές να εκτελούν ταυτόχρονα τον κώδικα της ίδιας ροής εντολών
  - Χρονοδρομολόγηση: Κάθε επεξεργαστής έχει τη δυνατότητα να **χρονοδρομολογεί διεργασίες**, για να **αποφεύγονται οι διαμάχες**
  - Συγχρονισμός: Είναι μία υπηρεσία η οποία **ενισχύει τον αμοιβαίο αποκλεισμό** και την οργάνωση των γεγονότων σε μία σειρά
    - Με πολλαπλές ενεργές διεργασίες, οι οποίες έχουν τη δυνατότητα προσπέλασης σε κοινόχρηστους χώρους διευθύνσεων ή κοινόχρηστες μονάδες Ε/Ε, θα πρέπει να παρέχεται αποτελεσματικός συγχρονισμός

# Συμμετρικοί Πολυεπεξεργαστές

---

## Ζητήματα Σχεδίασης Λειτουργικού Συστήματος σε SMP

- Βασικά ζητήματα σχεδίασης του Λειτουργικού Συστήματος:
  - **Διαχείριση μνήμης:** Η διαχείριση της μνήμης σε ένα σύστημα πολλαπλών επεξεργαστών θα πρέπει:
    - να **επιλύει** όλα τα **ζητήματα** που υπάρχουν σε **μηχανές μονοεπεξεργαστών**
    - να **εκμεταλλευτεί** το διαθέσιμο **παραλληλισμό** σε **επίπεδο υλικού**, όπως οι μνήμες πολλαπλών θυρών, για να βελτιώσει την απόδοση
    - να **συντονιστούν** οι **μηχανισμοί σελιδοποίησης σε διαφορετικούς επεξεργαστές**, ώστε να ενισχύουν τη σταθερότητα, σε περιπτώσεις όπου πολλοί επεξεργαστές μοιράζονται μια σελίδα ή τμήμα και αποφασίσουν ότι πρέπει να γίνει αντικατάσταση σελίδας

# Συμμετρικοί Πολυεπεξεργαστές

---

## Ζητήματα Σχεδίασης Λειτουργικού Συστήματος σε SMP

- Βασικά ζητήματα σχεδίασης του Λειτουργικού Συστήματος:
  - **Αξιοπιστία και ανοχή σε σφάλματα:** Το ΛΣ πρέπει να παρέχει μία **ανεκτή μείωση της απόδοσης** σε περίπτωση **βλάβης ενός επεξεργαστή**. Ο χρονοδρομολογητής και τα άλλα μέρη του λειτουργικού συστήματος θα πρέπει να *ανακατασκευάζουν με κατάλληλο τρόπο τους πίνακες διαχείρισης*.



# Μεγάλο Υπολογιστικό Σύστημα SMP

---

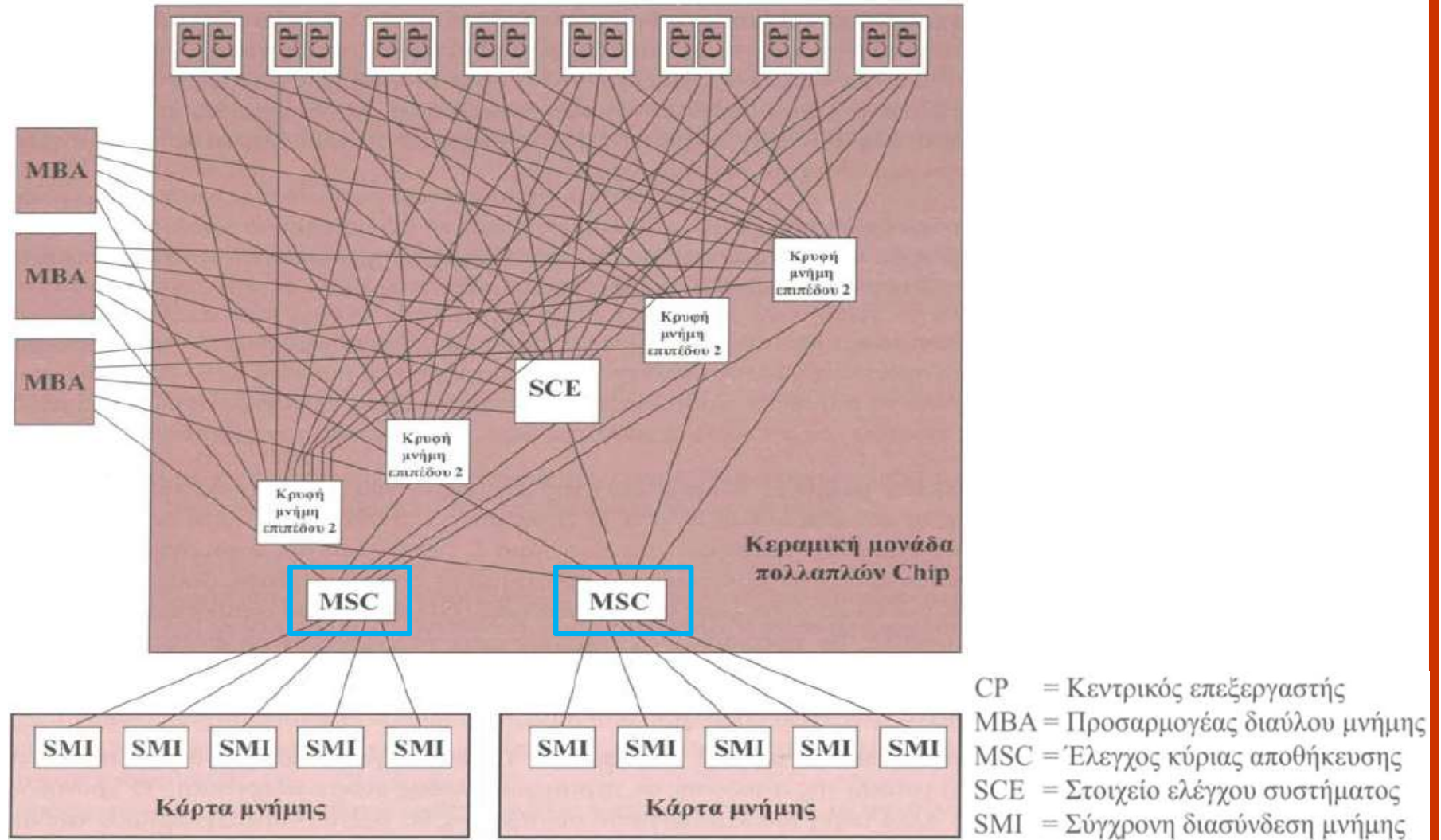
- Οι περισσότεροι συμμετρικοί πολυεπεξεργαστές χρησιμοποιούν μία στρατηγική διασύνδεσης διαύλου
- Μία εναλλακτική προσέγγιση, η οποία χρησιμοποιείται στην οικογένεια μεγάλων υπολογιστικών συστημάτων **zSeries** της **IBM**, με την ονομασία **z990** έχει τα εξής χαρακτηριστικά:
  1. **Chip επεξεργαστή με δύο πυρήνες:** Κάθε chip επεξεργαστή συμπεριλαμβάνει δύο παρόμοιους κεντρικούς επεξεργαστές
    - a. Ο κάθε **επεξεργαστής είναι υπερβαθμωτός με τεχνολογία CISC** και οι περισσότερες εντολές του είναι προκατασκευασμένες.
    - b. Κάθε **επεξεργαστής** συμπεριλαμβάνει μία **κρυφή μνήμη εντολών** και μία **κρυφή μνήμη δεδομένων Επιπέδου 1** με χωρητικότητα 256 KB.

# Μεγάλο Υπολογιστικό Σύστημα SMP

---

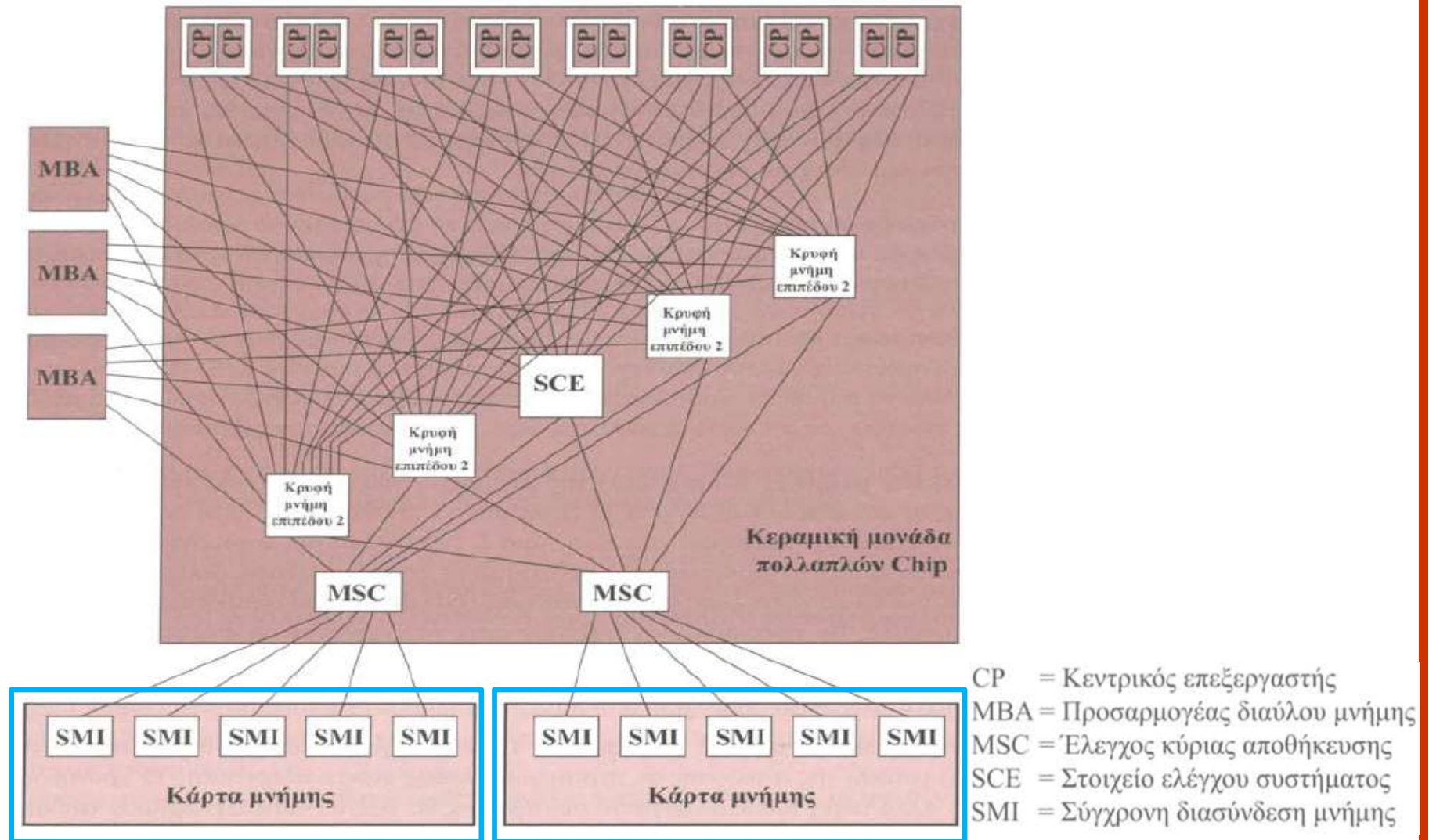
- Η οικογένεια μεγάλων υπολογιστικών συστημάτων zSeries της IBM, με την ονομασία z990 έχει τα εξής χαρακτηριστικά:
  - 2. Κρυφή μνήμη Επιπέδου 2:** Κάθε κρυφή μνήμη Επιπέδου 2 έχει χωρητικότητα 32 MB
    - a. Οι κρυφές μνήμες Επιπέδου 2 διατάσσονται σε συστάδες αποτελούμενες από 5 μνήμες
    - b. Κάθε συστάδα υποστηρίζει 8 chips επεξεργαστών και παρέχει προσπέλαση σε ολόκληρο το χώρο της κύριας μνήμης.
  - 3. Στοιχείο ελέγχου του συστήματος (SCE):** Διαιτητεύει την επικοινωνία μέσα στο σύστημα και έχει ένα βασικό ρόλο στη διατήρηση της συνοχής της κρυφής μνήμης.
    - Παρέχει συνδέσμους σε άλλα βιβλία της διάταξης

# Μεγάλο Υπολογιστικό Σύστημα SMP



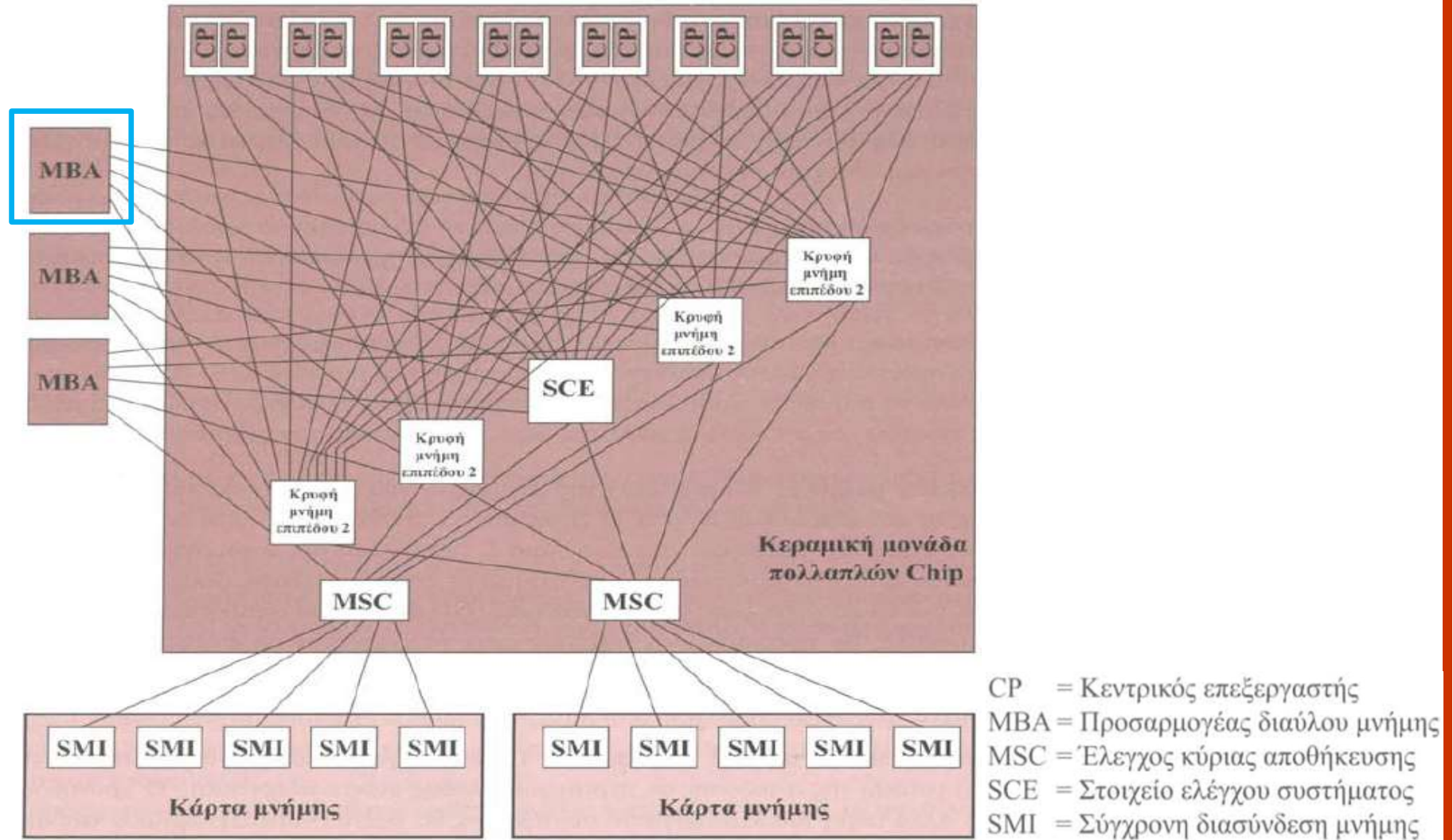
4. Έλεγχος κύριας μνήμης (MSC): Ο έλεγχος κύριας μνήμης διασυνδέει τις κρυφές μνήμες Επιπέδου 2 με την κύρια μνήμη

# Μεγάλο Υπολογιστικό Σύστημα SMP



5. **Κάρτα μνήμης:** Κάθε κάρτα περιέχει **32 MB** μνήμης. Η μέγιστη διάταξη μνήμης αποτελείται από 8 κάρτες. Διασυνδέονται με τον έλεγχο της κύριας μνήμης μέσω σύγχρονων διεπαφών μνήμης (SMI)

# Μεγάλο Υπολογιστικό Σύστημα SMP



6. Προσαρμογέας διαύλου μνήμης (MBA): Παρέχει μία διεπαφή σε διάφορους τύπους καναλιών E/E. Τα δεδομένα από/προς τα κανάλια μεταφέρονται άμεσα στη κρυφή μνήμη επιπέδου 2

# Μεγάλο Υπολογιστικό Σύστημα SMP

---

- Το σύστημα **z990** αποτελείται από **1 ως 4 βιβλία**.
- Κάθε **βιβλίο** περιέχει:
  - 12 επεξεργαστές
  - έως 64 GB μνήμης
  - προσαρμογείς E/E και
  - ένα στοιχείο ελέγχου του συστήματος, το οποίο συνδέει τα άλλα στοιχεία.
- Το **στοιχείο ελέγχου συστήματος** περιέχει μία **κρυφή μνήμη Επιπέδου 2** με μέγεθος 32 MB. η οποία εξυπηρετεί ως κεντρικό σημείο συνοχής για το συγκεκριμένο βιβλίο.

# Μεγάλο Υπολογιστικό Σύστημα SMP

---

- Η κρυφή μνήμη Επιπέδου 2 και η κύρια μνήμη, είναι **προσπελάσιμες** από έναν **επεξεργαστή** ή έναν **προσαρμογέα E/E** εντός **οποιοδήποτε βιβλίου του συστήματος**.
- Ο z990 παρουσιάζει ορισμένα ενδιαφέροντα γνωρίσματα:
  - Διασύνδεση με στοιχεία μεταγωγής
  - Κοινές κρυφές μνήμες Επιπέδου 2

# Διασύνδεση με στοιχεία Μεταγωγής

---

- Ο κοινόχρηστος δίαυλος αποτελεί ένα στοιχείο πρόκλησης συμφορήσεων (**bottleneck**)
  - Επηρεάζει αρνητικά τη κλιμάκωση
- Η αρχιτεκτονική z990 αντιμετωπίζει αυτό το ζήτημα με δύο τρόπους
  1. Η κύρια μνήμη διαχωρίζεται σε πολλαπλές κάρτες, κάθε μία από τις οποίες διαθέτει το δικό της ελεγκτή (MSC)
    - Το μέσο φορτίο κυκλοφορίας προς την κύρια μνήμη περικόπτεται, λόγω της ύπαρξης ανεξάρτητων μονοπατιών προς ξεχωριστά μέρη της μνήμης.



# Διασύνδεση με στοιχεία Μεταγωγής

---

- Η αρχιτεκτονική z990 αντιμετωπίζει αυτό το ζήτημα με δύο τρόπους
  2. Η **σύνδεση** από τους **επεξεργαστές** (στην πραγματικότητα από τις κρυφές μνήμες Επιπέδου 2) **προς μία κάρτα**, δεν έχει τη μορφή ενός κοινόχρηστου διαύλου, αλλά είναι **σύνδεσμοι από σημείο σε σημείο**.
    - Κάθε **επεξεργαστής** έχει ένα σύνδεσμο προς κάθε μία από τις **κρυφές μνήμες Επιπέδου 2** που βρίσκονται στο ίδιο βιβλίο
    - Κάθε **κρυφή μνήμη Επιπέδου 2** έχει ένα σύνδεσμο, μέσα από τον έλεγχο της κύριας μνήμης (MSC), προς κάθε μία από τις δύο κάρτες μνήμης που βρίσκονται στο ίδιο βιβλίο.

# Κοινές Κρυφές Μνήμες

---

- Κάθε επεξεργαστής διαθέτει μία δεσμευμένη κρυφή μνήμη Επιπέδου 1 και Επιπέδου 2 (σε ένα τυπικό σύστημα συμμετρικών πολυεπεξεργαστών)
- Στις μηχανές G4, G5 και z990 η IBM χρησιμοποιεί **διαμοιραζόμενη κρυφή μνήμη Επιπέδου 2** έναντι της μηχανής G3 όπου χρησιμοποίησε δεσμευμένες κρυφές μνήμες γιατί:
  1. Υπάρχει ένας **σημαντικός βαθμός κοινόχρηστων εντολών και δεδομένων** μεταξύ των επεξεργαστών.
  2. Χωρίς αναβάθμιση, ο **δίαυλος θα αποτελούσε σημείο συμφορήσεων** (bottleneck) επειδή θα υπήρχε σημαντική αύξηση της κυκλοφορίας στο δίαυλο

# Κοινές Κρυφές Μνήμες

---

- Η κοινή χρήση της κρυφής μνήμης Επιπέδου 2 φαίνεται **κακή ιδέα αρχικά**.
  - Η προσπέλαση της μνήμης από τους επεξεργαστές θα γίνει πιο αργή λόγω ανταγωνισμού για την προσπέλαση μιας κοινής κρυφής μνήμης Επιπέδου 2.
- Αν όμως **οι επεξεργαστές μοιράζονται μία επαρκή ποσότητα δεδομένων**, τότε η **κοινή κρυφή μνήμη αυξάνει την απόδοση** και δεν την επιβραδύνει.
  - Τα δεδομένα τα οποία είναι κοινά και βρίσκονται στην κοινή κρυφή μνήμη

# Συνοχή Κρυφής Μνήμης

---

## Πρόβλημα Συνοχής της Κρυφής Μνήμης

- Είναι δυνατόν να υπάρχουν **πολλαπλά αντίγραφα** των ίδιων δεδομένων, **ταυτόχρονα** σε **πολλές κρυφές μνήμες**
- Αν οι **επεξεργαστές** επιτρέπεται να **ενημερώνουν ελεύθερα τα δικά τους αντίγραφα**, τότε είναι δυνατόν να προκύψει μία **μη συνεπή εικόνα της κύριας μνήμης**

# Συνοχή Κρυφής Μνήμης

---

- Πολιτικές εγγραφής της κύριας μνήμης:
  1. **Εγγραφή προς τα πίσω:** Συνήθως, οι λειτουργίες εγγραφής εκτελούνται **μόνο** προς την **κρυφή μνήμη**. Η κύρια μνήμη ενημερώνεται **μόνο** όταν διαγράφεται η αντίστοιχη γραμμή της κρυφής μνήμης
  2. **Δια μέσου εγγραφή:** Όλες οι λειτουργίες εγγραφών γίνονται στην **κύρια μνήμη** καθώς και στην **κρυφή μνήμη**, εξασφαλίζοντας την εγκυρότητα των δεδομένων της κύριας μνήμης.

# Συνοχή Κρυφής Μνήμης

---

- Η πολιτική **εγγραφής προς τα πίσω**, είναι δυνατόν να οδηγήσει σε **ασυμφωνίες**.
  - Αν δύο κρυφές μνήμες περιέχουν την ίδια γραμμή και η γραμμή ενημερωθεί μόνο στη μία, η άλλη θα περιέχει μη έγκυρη τιμή
- Η πολιτική της **δια μέσου εγγραφής**, είναι δυνατόν να εμφανιστούν **ασυμφωνίες**, εάν **μια κρυφή μνήμη δεν παρακολουθεί την κυκλοφορία της κύριας μνήμης**

# Συνοχή Κρυφής Μνήμης

---

## Επίλυση του Προβλήματος Συνοχής Κρυφής Μνήμης

- Πρωτόκολλο **MESI** (Modified Exclusive Shared Invalid)
  - Χρησιμοποιείται στο Pentium 4 και PowerPC
- Ο στόχος κάθε πρωτοκόλλου είναι να επιτραπεί σε πρόσφατα χρησιμοποιημένες τοπικές μεταβλητές να εισέλθουν στην κατάλληλη κρυφή μνήμη και να παραμείνουν εκεί για ένα πλήθος από λειτουργίες ανάγνωσης και εγγραφής
  - Οι κοινές μεταβλητές μπορεί να βρίσκονται σε πολλές κρυφές μνήμες
- Οι προσεγγίσεις σχετικά με τη συνοχή της κρυφής μνήμης διαιρούνται γενικά σε **προσεγγίσεις λογισμικού και υλικού**.

# Συνοχή Κρυφής Μνήμης – Λύσεις Λογισμικού

---

- Οι λύσεις που βασίζονται στο λογισμικό, **αποφεύγουν την ανάγκη χρησιμοποίησης επιπλέον κυκλωμάτων, βασιζόμενες στο ΛΣ και στο μεταγλωττιστή.**
- Είναι ελκυστικές, επειδή η **επιβάρυνση** που προκύπτει λόγω της ανίχνευσης προβλημάτων, **μεταφέρεται από το χρόνο εκτέλεσης στο χρόνο μεταγλώττισης,**
  - η πολυπλοκότητα σχεδίασης μεταφέρεται από το υλικό στο λογισμικό
- Οι προσεγγίσεις λογισμικού που σχετίζονται με το χρόνο μεταγλώττισης, θα πρέπει να λαμβάνουν **συντηρητικές αποφάσεις → μη αποτελεσματική χρήση της κρυφής μνήμης**



# Συνοχή Κρυφής Μνήμης – Λύσεις Λογισμικού

---

- Οι μηχανισμοί συνοχής που βασίζονται στο μεταγλωττιστή, αναλύουν τον κώδικα για να προσδιορίσουν **ποια αντικείμενα δεδομένων είναι πιθανό να καταστούν ανασφαλή** και τα **μαρκάρουν κατάλληλα**
- Στη συνέχεια, το **λειτουργικό σύστημα** ή το υλικό **αποτρέπει την τοποθέτηση** αυτών των **αντικειμένων δεδομένων στην κρυφή μνήμη**

# Συνοχή Κρυφής Μνήμης – Λύσεις Λογισμικού

---

- **Απλή προσέγγιση:** *Απαγορεύεται η τοποθέτηση στην κρυφή μνήμη κάθε κοινόχρηστης μεταβλητής δεδομένων*
  - Όμως, μία κοινόχρηστη δομή δεδομένων είναι πιθανό να χρησιμοποιείται αποκλειστικά για ένα συγκεκριμένο χρονικό διάστημα και μόνον για λειτουργίες ανάγνωσης
- **Αποτελεσματική προσέγγιση:** Ο κώδικας αναλύεται για να προσδιοριστούν *ασφαλείς περίοδοι για τις κοινόχρηστες μεταβλητές.*
  - Στη συνέχεια, ο μεταγλωττιστής εισάγει εντολές στον κώδικα που παράγεται, για να ενισχύσει τη συνοχή της κρυφής μνήμης κατά τη διάρκεια κρίσιμων περιόδων.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

- Οι λύσεις οι οποίες βασίζονται στο υλικό, αναφέρονται γενικά ως **πρωτόκολλα συνοχής της κρυφής μνήμης**
- Οι λύσεις υλικού παρέχουν δυναμική αναγνώριση, κατά τη διάρκεια εκτέλεσης, των συνθηκών οι οποίες είναι πιθανό να προκαλέσουν ασυμφωνίες
  - Επομένως, πιο αποτελεσματική χρήση των κρυφών μνημών
- Οι λύσεις υλικού δεν είναι ορατές στον προγραμματιστή και στο μεταγλωττιστή → μειώνει την επιβάρυνση της ανάπτυξης λογισμικού

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

- Οι λύσεις Υλικού χωρίζονται σε δύο κατηγορίες
  - Πρωτόκολλα Καταλόγου
  - Αδιάκριτα Πρωτόκολλα

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Πρωτόκολλα Καταλόγου

- **Συλλέγουν και διατηρούν** τις πληροφορίες σχετικά με τη θέση στην οποία αποθηκεύονται αντίγραφα των γραμμών
- Υπάρχει ένας **κεντρικός ελεγκτής** (τμήμα του ελεγκτή της κύριας μνήμης) και ένας **κατάλογος** ο οποίος αποθηκεύεται στην κύρια μνήμη.
- Ο **κατάλογος** περιέχει γενικές πληροφορίες κατάστασης για τα **περιεχόμενα** των τοπικών κρυφών μνημών.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Πρωτόκολλα Καταλόγου

- Όταν ένας ελεγκτής κρυφής μνήμης κάνει μία αίτηση, τότε ο κεντρικός ελεγκτής ελέγχει και εκδίδει τις κατάλληλες εντολές μεταφοράς δεδομένων ανάμεσα:
  - στη μνήμη και τις κρυφές μνήμες ή
  - ανάμεσα στις κρυφές μνήμες
- Κάθε ενέργεια σε τοπικό επίπεδο η οποία έχει τη δυνατότητα να επηρεάσει τη γενικότερη κατάσταση μίας γραμμής θα πρέπει να αναφέρεται στην κεντρικό ελεγκτή

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Πρωτόκολλα Καταλόγου

- ο ελεγκτής διατηρεί πληροφορίες σχετικά με το ποιοι επεξεργαστές έχουν αντίγραφα και από ποιες γραμμές
  1. Ένας επεξεργαστής θέλει να γράψει σε ένα τοπικό αντίγραφο μιας γραμμής
  2. Ο επεξεργαστής πρέπει να αιτηθεί από τον ελεγκτή άδεια αποκλειστικής χρήσης της γραμμής
  3. Ο ελεγκτής στέλνει ένα μήνυμα προς όλους τους επεξεργαστές αναγκάζοντας κάθε επεξεργαστή να ακυρώσει αυτό το αντίγραφο

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Πρωτόκολλα Καταλόγου

4. Μετά τη λήψη της επιβεβαίωσης από κάθε επεξεργαστή, ο ελεγκτής αποδέχεται την αποκλειστική προσπέλαση την οποία αιτήθηκε ο επεξεργαστής
5. Όταν ένας επεξεργαστής επιδιώξει να διαβάσει μία γραμμή η οποία έχει δοθεί για αποκλειστική προσπέλαση σε έναν άλλο επεξεργαστή, θα στείλει μία ειδοποίηση αστοχίας (cache miss) προς τον ελεγκτή.
6. Στη συνέχεια, ο ελεγκτής εκδίδει μία εντολή στον επεξεργαστή που προσπελαύνει αυτή τη γραμμή με την οποία απαιτεί να εκτελέσει μία λειτουργία εγγραφής προς τα πίσω, στην κύρια μνήμη



# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Πρωτόκολλα Καταλόγου

7. Η γραμμή είναι δυνατόν να μοιραστεί για ανάγνωση τόσο από τον αρχικό επεξεργαστή, όσο και από τον επεξεργαστή που έκανε την αίτηση

### ❖ **Μειονεκτήματα:**

- ❖ κεντρικής συμφόρησης
- ❖ της επιβάρυνσης λόγω της επικοινωνίας ανάμεσα σε διάφορους ελεγκτές κρυφών μνημών και τον κεντρικό ελεγκτή

### ❖ **Πλεονεκτήματα:**

- ❖ Αποτελεσματικά σε συστήματα υψηλής κλιμάκωσης τα οποία χρησιμοποιούν, πολλούς διαύλους ή κάποιο πολύπλοκο σύστημα διασύνδεσης.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα

- **Κατανέμουν** την **ευθύνη** για τη διατήρηση της συνοχής της κρυφής μνήμης ανάμεσα **σε όλους τους ελεγκτές κρυφών μνημών** ενός πολυεπεξεργαστή
- Μία κρυφή μνήμη θα πρέπει να αναγνωρίζει πότε μία γραμμή την οποία αποθηκεύει, είναι κοινή με άλλες κρυφές μνήμες
- Όταν γίνει μία λειτουργία ενημέρωσης θα πρέπει να ανακοινωθεί σε όλες τις κρυφές μνήμες με κάποιο μηχανισμό
- Κάθε ελεγκτής κρυφής μνήμης έχει τη δυνατότητα να **εξετάσει αδιάκριτα** το δίκτυο, προκειμένου να παρατηρήσει αυτές τις ενημερώσεις

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα

- Τα αδιάκριτα πρωτόκολλα είναι **ιδανικά** για **πολυεπεξεργαστές** που βασίζουν την **επικοινωνία τους σε διαύλους**
  - επειδή ο κοινός δίαυλος παρέχει έναν απλό τρόπο εκπομπής και εξέτασης.
- Θα πρέπει να προσεχθεί ότι η αυξημένη κυκλοφορία στο δίαυλο η οποία χρειάζεται για την εκπομπή αλλά και την αδιάκριτη εξέταση
  - δεν ακυρώνει τα κέρδη που υπάρχουν από τη χρήση των τοπικών κρυφών μνημών.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα

- Δύο βασικές προσεγγίσεις έχουμε στα αδιάκριτα πρωτόκολλα:
  - **Ακύρωση εγγραφής**: Μπορούν να υπάρχουν αρκετές λειτουργίες ανάγνωσης αλλά μία μόνον λειτουργία εγγραφής
  - **Ενημέρωση εγγραφής** (ή εκπομπή εγγραφής): Μπορούν να υπάρχουν αρκετές λειτουργίες εγγραφής αλλά και ανάγνωσης.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα – Ακύρωση Εγγραφής

### Λειτουργία Πρωτοκόλλου

1. Μία γραμμή **μοιράζεται σε αρκετές κρυφές μνήμες** για ανάγνωση (διάβασμα).
2. Όταν μία από τις κρυφές μνήμες επιθυμεί να **εκτελέσει μία λειτουργία εγγραφής** σε αυτή τη γραμμή, τότε **εκδίδει μία ειδοποίηση** με την οποία **ακυρώνει τη γραμμή αυτή στις άλλες μνήμες**, με αποτέλεσμα η γραμμή να διατίθεται αποκλειστικά στη συγκεκριμένη κρυφή μνήμη.
3. Ο επεξεργαστής στον οποίο ανήκει η κρυφή μνήμη μπορεί να **εκτελέσει τοπικά διάφορες εγγραφές**, μέχρι να ζητηθεί η γραμμή από έναν άλλο επεξεργαστή.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα – Ενημέρωση Εγγραφής

### Λειτουργία Πρωτοκόλλου

1. Όταν ένας επεξεργαστής επιθυμεί να ενημερώσει μία κοινή γραμμή, η λέξη η οποία θα ενημερωθεί διανέμεται σε όλους τους άλλους επεξεργαστές
2. Οι κρυφές μνήμες οι οποίες περιέχουν αυτή τη γραμμή μπορούν να την ενημερώσουν.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα

- *Καμία από τις δύο αυτές προσεγγίσεις δεν είναι ανώτερη της άλλης κάτω από κάθε συνθήκη.*
- Η απόδοση εξαρτάται:
  - από το πλήθος των τοπικών κρυφών μνημών και
  - από το υπόδειγμα των λειτουργιών εγγραφής και ανάγνωσης της μνήμης.

# Συνοχή Κρυφής Μνήμης – Λύσεις Υλικού

---

## Αδιάκριτα Πρωτόκολλα

- Η **ακύρωση εγγραφής** είναι η πιο **διαδεδομένη** στους εμπορικούς υπολογιστές πολλαπλών επεξεργαστών όπως στον Pentium 4
- Η κατάσταση κάθε γραμμής της κρυφής μνήμης μαρκάρεται (χρησιμοποιώντας δύο επιπλέον bits), ως:
  - τροποποιημένη **Modified**
  - αποκλειστική **Exclusive**
  - κοινόχρηστη **Shared**
  - μη έγκυρη **Invalid**



# Πρωτόκολλο MESI

---

- Είναι ένα πρωτόκολλο ακύρωσης εγγραφής
- Κάθε γραμμή βρίσκεται σε μία από τις ακόλουθες καταστάσεις:
  - **Τροποποιημένη:** Η γραμμή της κρυφής μνήμης έχει τροποποιηθεί (είναι διαφορετική σε σχέση με την κύρια μνήμη) και είναι διαθέσιμη μόνον σε αυτή την κρυφή μνήμη.
  - **Αποκλειστική:** Η γραμμή της κρυφής μνήμης είναι η ίδια με εκείνη της κύριας μνήμης και δεν υπάρχει σε άλλη κρυφή μνήμη.
  - **Κοινόχρηστη:** Η γραμμή της κρυφής μνήμης είναι ίδια με εκείνη της κύριας μνήμης και μπορεί να υπάρχει σε άλλη κρυφή μνήμη
  - **Μη έγκυρη:** Η γραμμή της κρυφής μνήμης περιέχει μη έγκυρα δεδομένα.

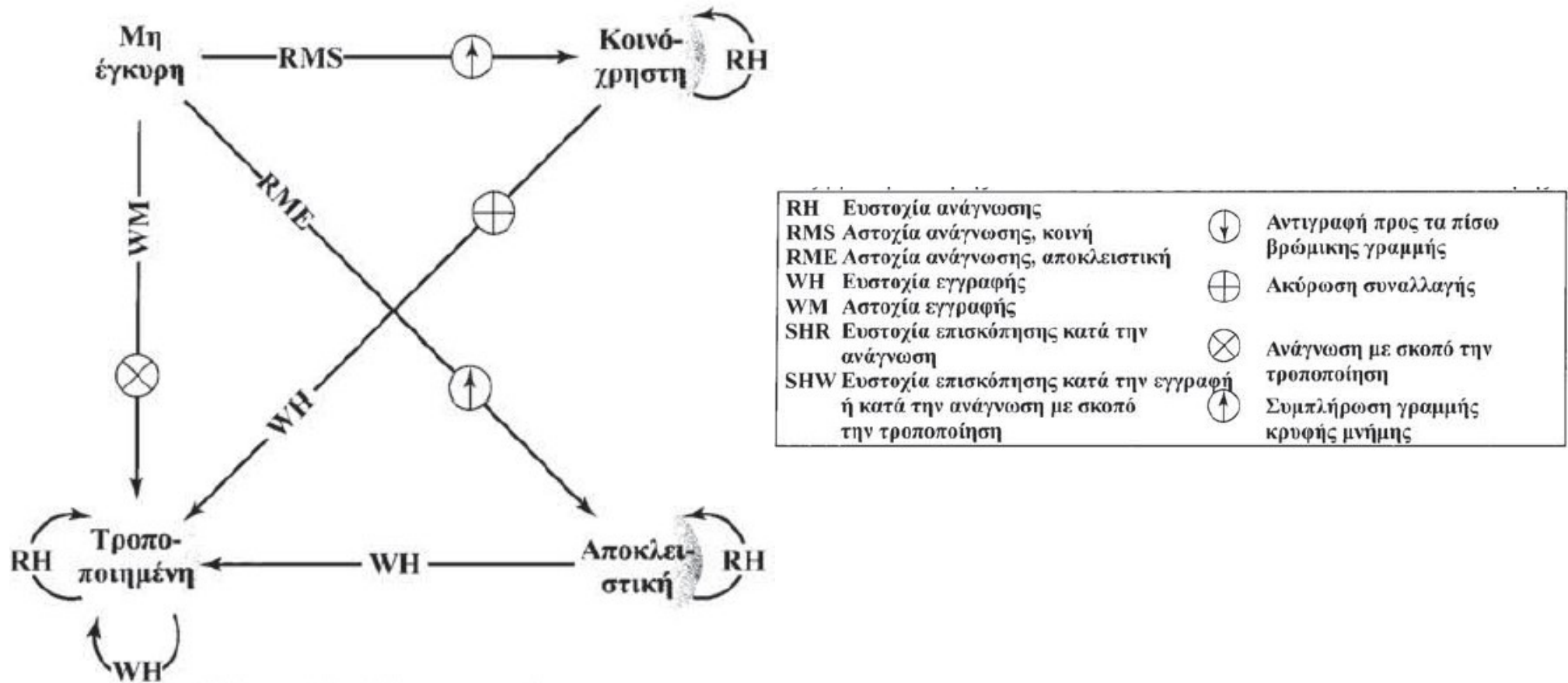
# Πρωτόκολλο MESI

- Καταστάσεις γραμμών της κρυφής μνήμης στο πρωτόκολλο MESI

	<b>M</b> <b>Modified</b> (Τροποποιημένη)	<b>E</b> <b>Exclusive</b> (Αποκλειστική)	<b>S</b> <b>Shared</b> (Κοινόχρηστη)	<b>I</b> <b>Invalid</b> (Μη έγκυρη)
Είναι έγκυρη αυτή η γραμμή της κρυφής μνήμης;	Ναι	Ναι	Ναι	Όχι
Το αντίγραφο της μνήμης είναι	μη ενημερωμένο	έγκυρο	έγκυρο	-
Υπάρχουν αντίγραφα σε άλλες κρυφές μνήμες;	Όχι	Όχι	Ίσως	Ίσως
Μία εγγραφή σε αυτή τη γραμμή	δεν πηγαίνει στο δίαυλο	δεν πηγαίνει στο δίαυλο	πηγαίνει στο δίαυλο και ενημερώνει την κρυφή μνήμη	πηγαίνει απ' ευθείας στο δίαυλο

# Πρωτόκολλο MESI

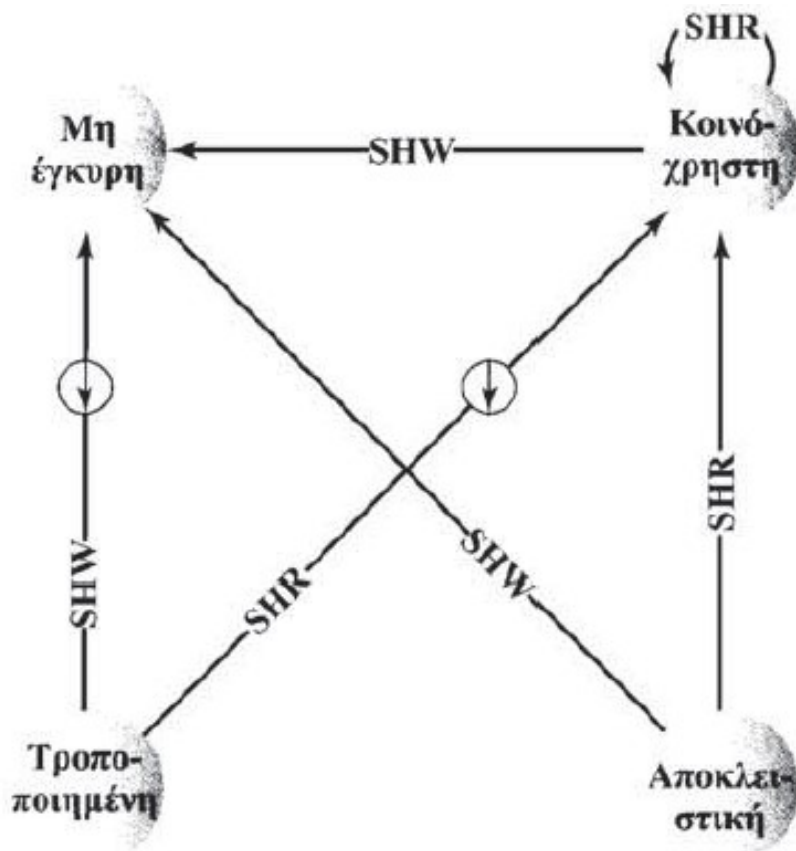
- **Μεταβάσεις** που λαμβάνουν χώρα εξαιτίας ενεργειών του επεξεργαστή που συνδέεται με τη κρυφή μνήμη



- Αν το επόμενο γεγονός προέρχεται από το συνδεδεμένο επεξεργαστή τότε η μετάβαση προκύπτει από το παραπάνω διάγραμμα

# Πρωτόκολλο MESI

- **Μεταβάσεις** που λαμβάνουν χώρα εξαιτίας **γεγονότων που ανιχνεύονται** μετά από **αδιάκριτη εξέταση πάνω στο δίαυλο**



RH	Ευστοχία ανάγνωσης	⊖	Αντιγραφή προς τα πίσω βρώμικης γραμμής
RMS	Αστοχία ανάγνωσης, κοινή	⊕	Ακύρωση συναλλαγής
RME	Αστοχία ανάγνωσης, αποκλειστική	⊗	Ανάγνωση με σκοπό την τροποποίηση
WH	Ευστοχία εγγραφής	⊙	Συμπλήρωση γραμμής κρυφής μνήμης
WM	Αστοχία εγγραφής		
SHR	Ευστοχία επισκόπησης κατά την ανάγνωση		
SHW	Ευστοχία επισκόπησης κατά την εγγραφή ή κατά την ανάγνωση με σκοπό την τροποποίηση		

- Αν το επόμενο γεγονός προέρχεται από το δίαυλο τότε η μετάβαση προκύπτει από το παραπάνω διάγραμμα

# Πρωτόκολλο MESI

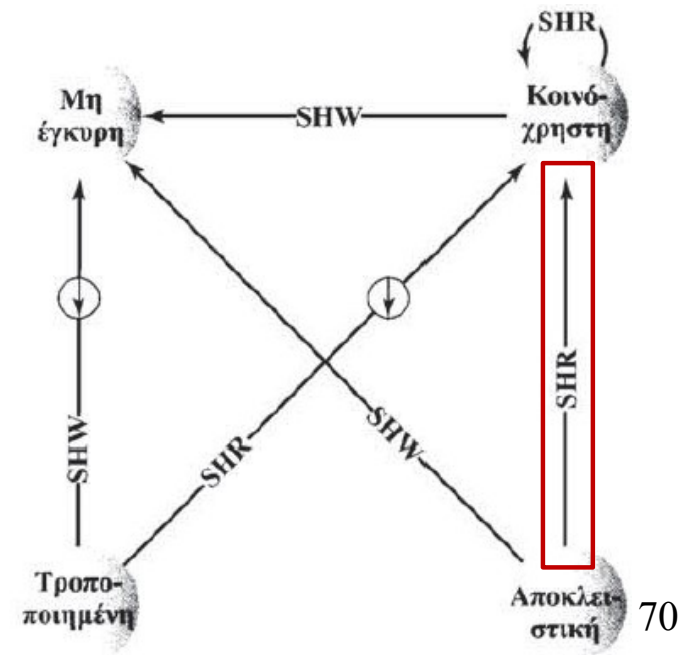
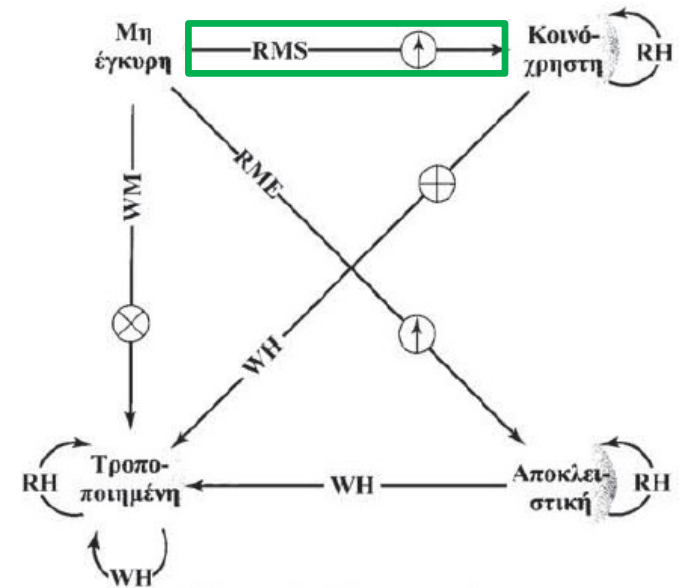
---

- **Αστοχία Ανάγνωσης:** Όταν συμβεί μία αστοχία ανάγνωσης στην τοπική κρυφή μνήμη ο επεξεργαστής αρχικοποιεί μία λειτουργία ανάγνωσης από την κύρια μνήμη
- Ο επεξεργαστής εισάγει ένα σήμα στο δίαυλο με το οποίο ειδοποιεί όλες τις άλλες μονάδες επεξεργαστών/κρυφών μνημών να εξετάσουν αδιάκριτα τη μετάβαση.
- Υπάρχει ένα πλήθος από πιθανά αποτελέσματα.

# Πρωτόκολλο MESI

- **Αστοχία Ανάγνωσης**

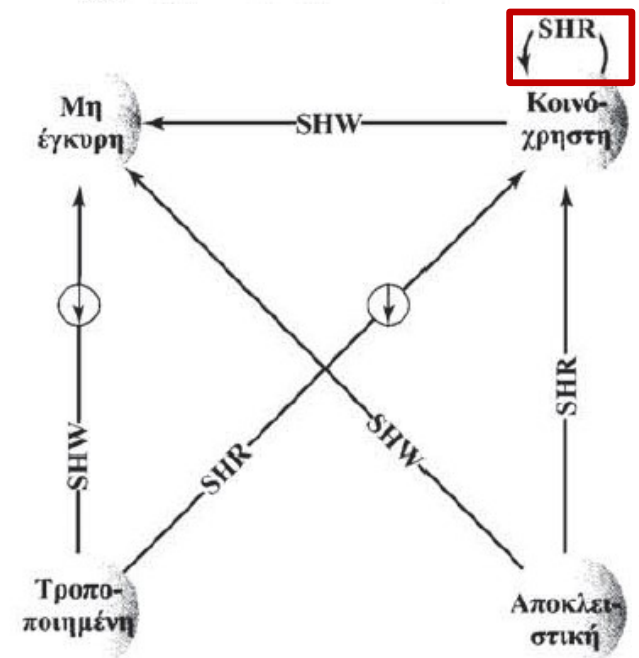
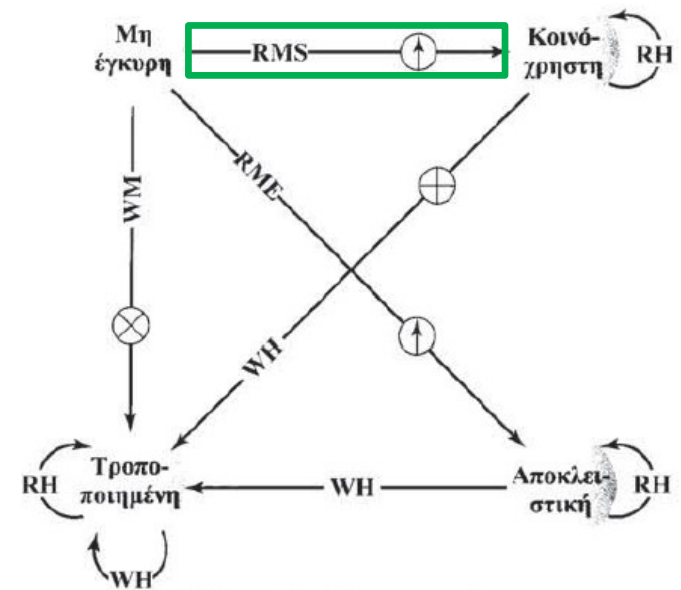
1. Αν μία άλλη κρυφή μνήμη διαθέτει ένα μη τροποποιημένο αντίγραφο της γραμμής σε **αποκλειστική κατάσταση** στέλνει ένα σήμα με το οποίο δείχνει ότι μοιράζεται τη γραμμή
2. Στη συνέχεια, **ο αποκρινόμενος επεξεργαστής αλλάζει την κατάσταση του αντιγράφου του από αποκλειστική σε κοινόχρηστη**
3. Ο επεξεργαστής που ξεκίνησε τη διαδικασία διαβάζει τη γραμμή από την κύρια μνήμη και αλλάζει τη γραμμή της δικής του κρυφής μνήμης από μη έγκυρη σε κοινόχρηστη



# Πρωτόκολλο MESI

- **Αστοχία Ανάγνωσης**

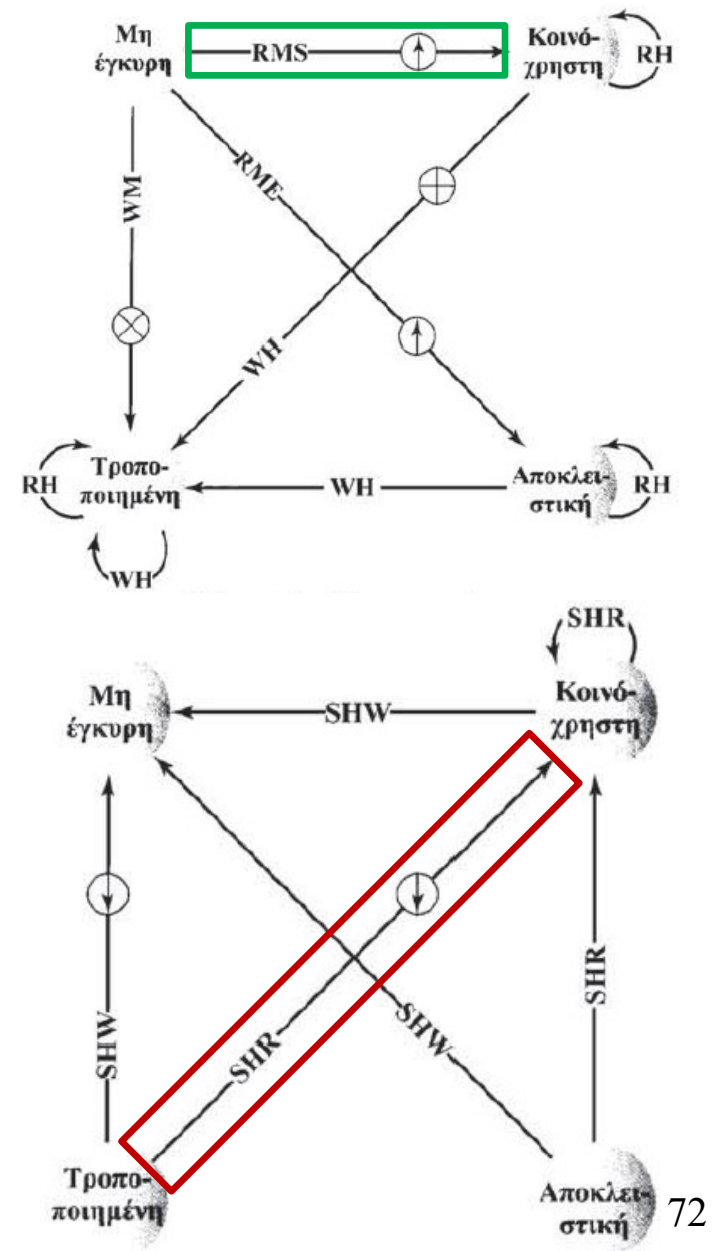
1. Αν μία ή περισσότερες κρυφές μνήμες διαθέτουν ένα μη τροποποιημένο αντίγραφο αυτής της γραμμής σε κατάσταση **κοινόχρηστη**, κάθε μία από αυτές στέλνει ένα σήμα ότι μοιράζεται τη γραμμή
2. Ο επεξεργαστής που ξεκίνησε τη διαδικασία **διαβάζει τη γραμμή** και **αλλάζει τη γραμμή της δικής του κρυφής μνήμης από μη έγκυρη σε κοινόχρηστη**



# Πρωτόκολλο MESI

- **Αστοχία Ανάγνωσης**

1. Αν μία άλλη κρυφή μνήμη διαθέτει τροποποιημένο αντίγραφο της γραμμής τότε αυτή η κρυφή μνήμη εμποδίζει την ανάγνωση της μνήμης και **παρέχει τη γραμμή στη κρυφή μνήμη** που τη ζήτησε μέσω του **δίαυλου**
2. Στη συνέχεια, η αποκριθείσα κρυφή μνήμη **αλλάζει τη γραμμή της από τροποποιημένη σε κοινόχρηστη**
3. Η γραμμή η οποία αποστέλλεται στην κρυφή μνήμη η οποία έκανε την αίτηση, λαμβάνεται επίσης και υφίσταται επεξεργασία από τον **ελεγκτή της κύριας μνήμης**, ο οποίος αποθηκεύει το μπλοκ σε αυτή

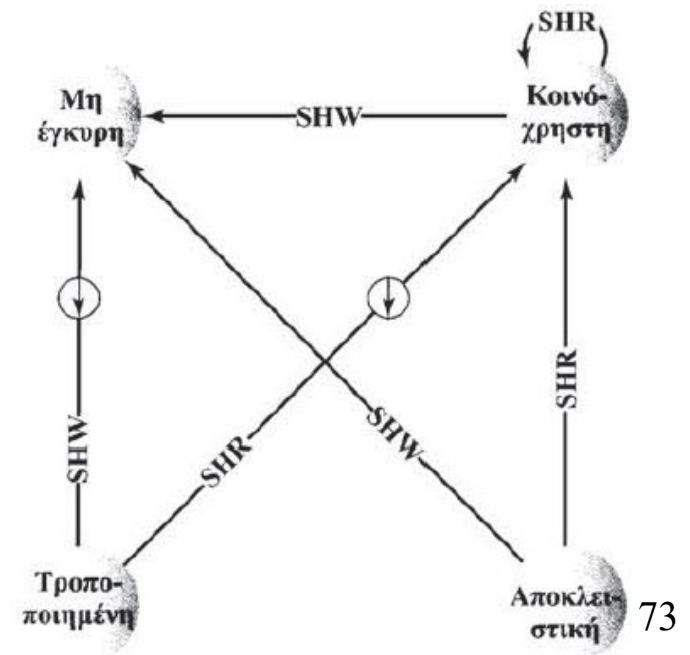
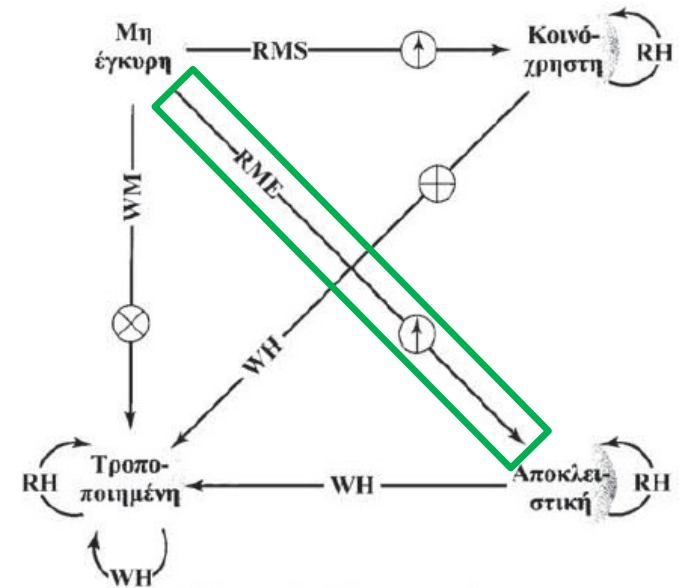




# Πρωτόκολλο MESI

- **Αστοχία Ανάγνωσης**

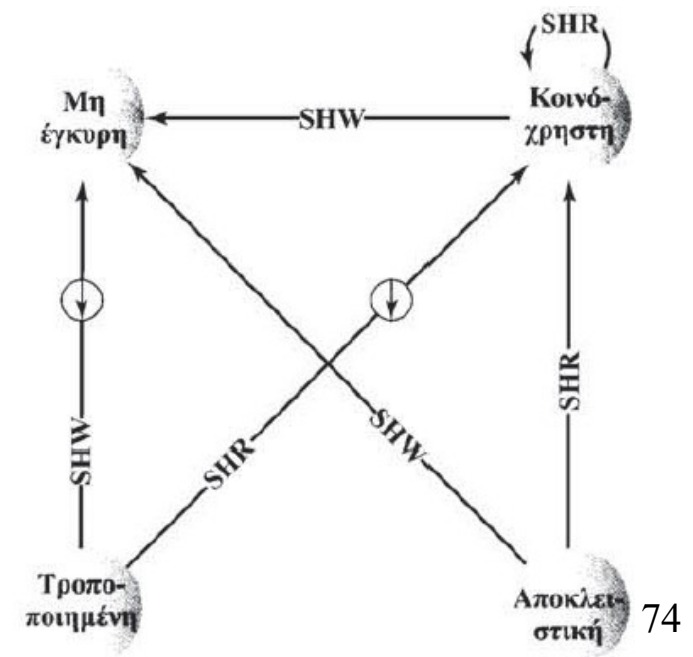
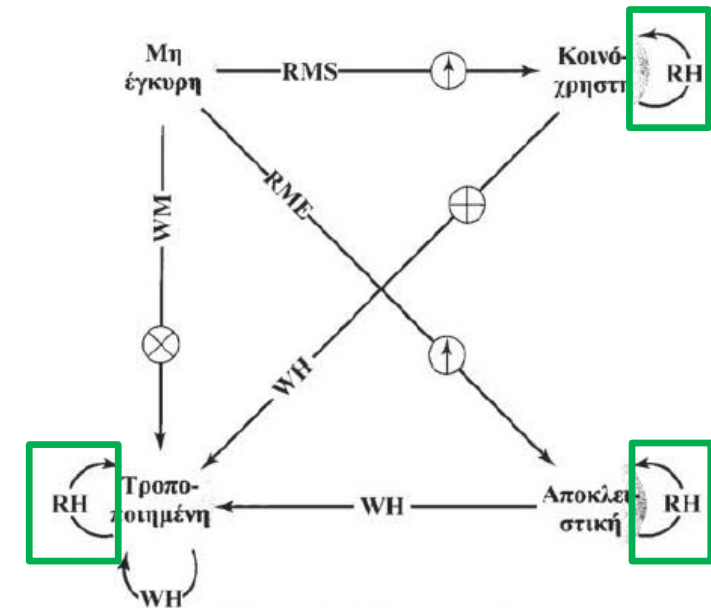
1. Αν καμία άλλη κρυφή μνήμη δεν διαθέτει **αντίγραφο** της γραμμής (καθαρό ή τροποποιημένο), τότε **δεν επιστρέφονται σήματα**.
2. Ο επεξεργαστής που ξεκίνησε τη διαδικασία **διαβάζει τη γραμμή και αλλάζει τη γραμμή της δικής του κρυφής μνήμης από μη έγκυρη σε αποκλειστική**



# Πρωτόκολλο MESI

- **Ευστοχία Ανάγνωσης**

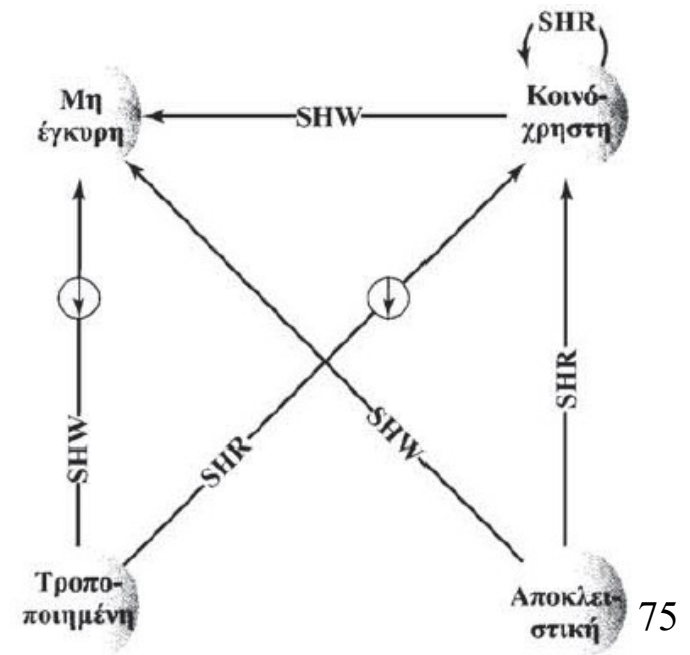
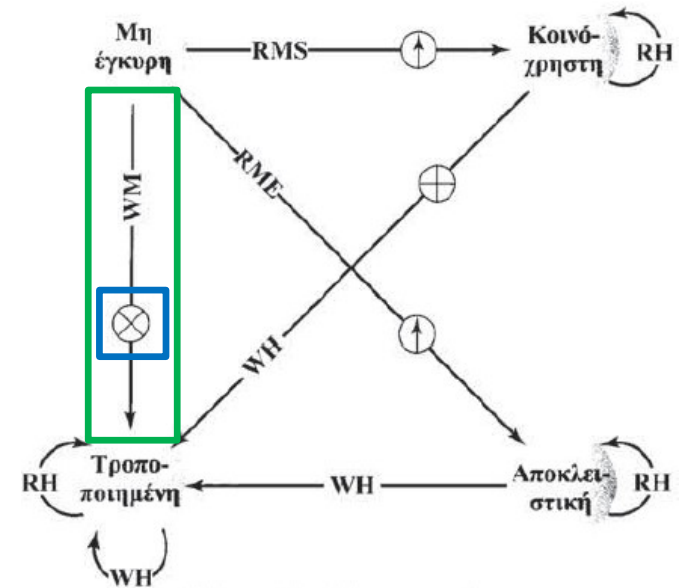
1. Όταν υπάρξει **ευστοχία ανάγνωσης** σε μία γραμμή η οποία βρίσκεται στην τοπική κρυφή μνήμη, ο επεξεργαστής απλώς διαβάζει το **ζητούμενο αντικείμενο**. **Δεν υπάρχει αλλαγή κατάστασης**.



# Πρωτόκολλο MESI

- **Αστοχία Εγγραφής**

1. Όταν συμβεί **αστοχία εγγραφής στην κρυφή μνήμη**, ο επεξεργαστής ξεκινά μία λειτουργία **ανάγνωσης από την κύρια μνήμη**, με σκοπό να διαβάσει τη γραμμή της κύριας μνήμης
2. Ο επεξεργαστής **εκδίδει ένα σήμα στο δίαυλο** το οποίο έχει τη σημασία **ανάγνωση με σκοπό τη τροποποίηση**. Όταν φορτωθεί αυτή η γραμμή, μαρκάρεται αμέσως ως **τροποποιημένη**.

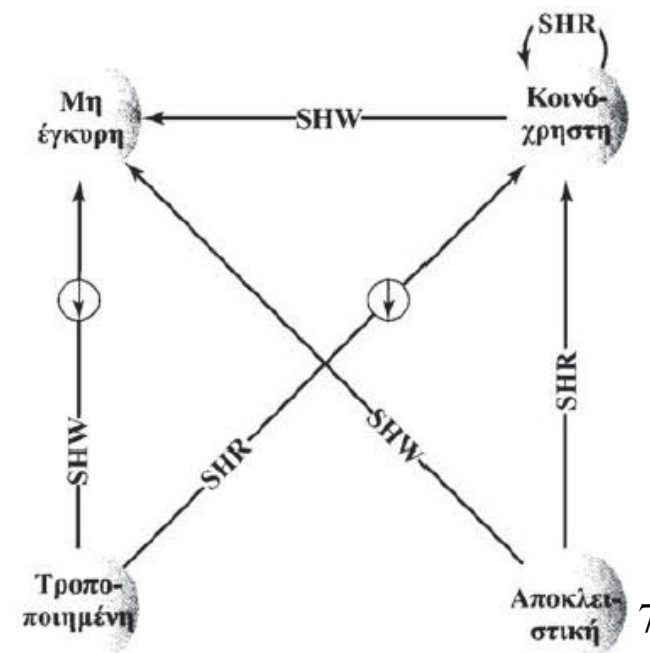
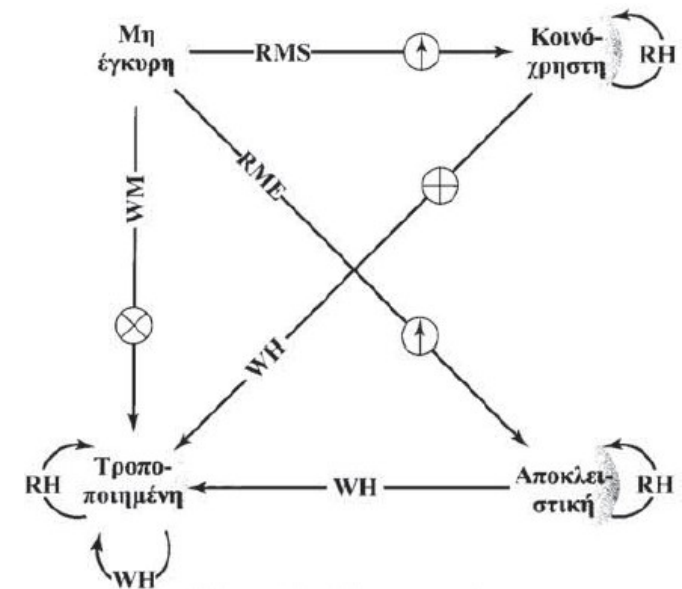


# Πρωτόκολλο MESI

- **Ευστοχία Εγγραφής**

- Όταν συμβεί μία ευστοχία εγγραφής για μία γραμμή η οποία βρίσκεται στην τοπική κρυφή μνήμη, το **αποτέλεσμα εξαρτάται** από την τρέχουσα **κατάσταση** αυτής της γραμμής στην **τοπική κρυφή μνήμη**:

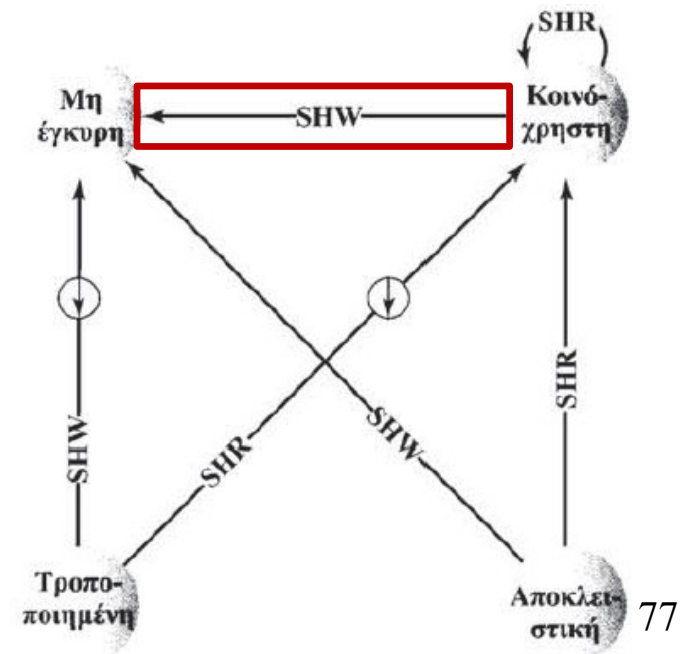
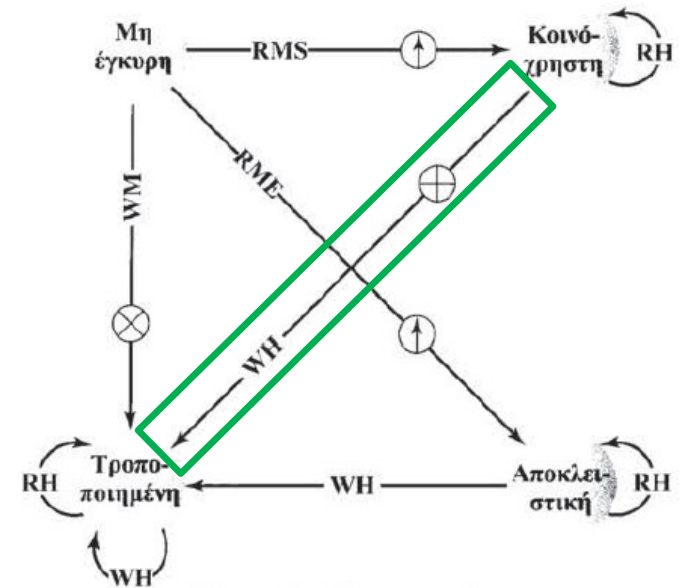
1. **Κοινόχρηστη**
2. **Αποκλειστική**
3. **Τροποποιημένη**



# Πρωτόκολλο MESI

- **Ευστοχία Εγγραφής - Κοινόχρηστη**

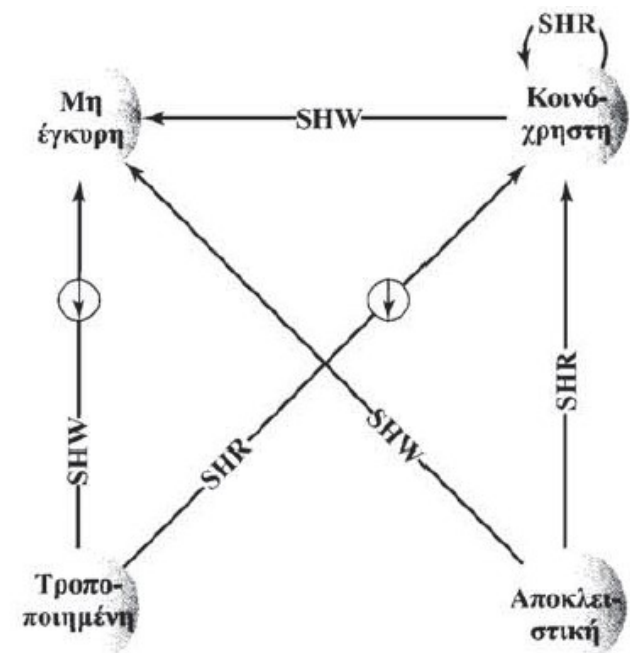
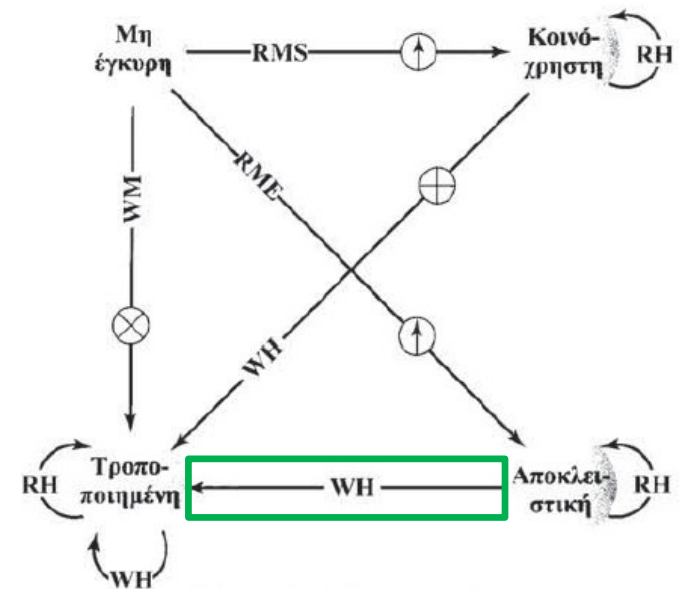
1. Πριν κάνει την ενημέρωση, ο επεξεργαστής θα πρέπει να **αποκτήσει αποκλειστική ιδιοκτησία της γραμμής**. Ο επεξεργαστής **στέλνει ένα σήμα στο δίαυλο**
2. Κάθε επεξεργαστής ο οποίος διαθέτει ένα κοινόχρηστο αντίγραφο της γραμμής μέσα στη δική του κρυφή μνήμη **αλλάζει τη γραμμή από κοινόχρηστη σε μη έγκυρη**.
3. Στη συνέχεια, ο επεξεργαστής που ξεκίνησε τη διαδικασία **εκτελεί την ενημέρωση και αλλάζει την κατάσταση της γραμμής του από κοινόχρηστη σε τροποποιημένη**



# Πρωτόκολλο MESI

- **Ευστοχία Εγγραφής - Αποκλειστική**

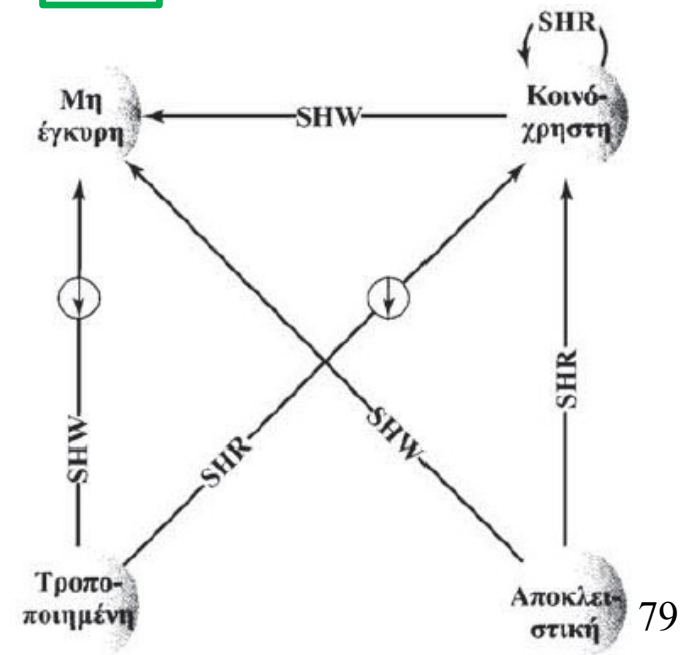
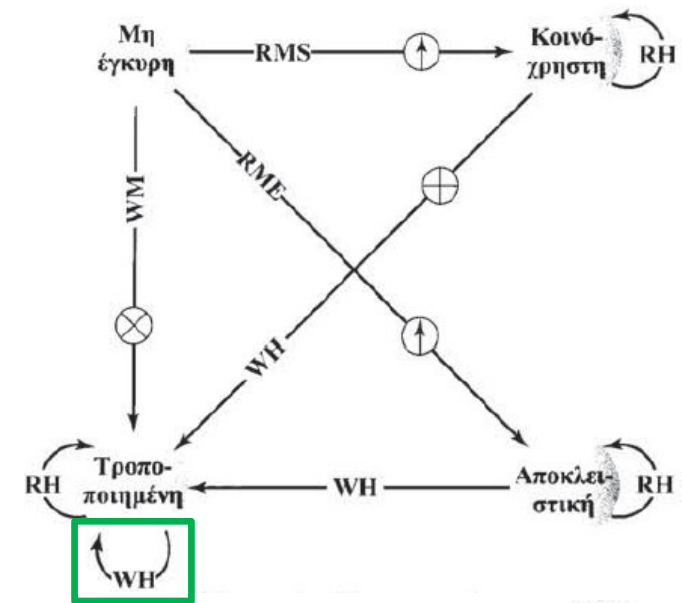
1. Ο επεξεργαστής έχει ήδη αποκτήσει αποκλειστική χρήση της γραμμής και απλώς εκτελεί την ενημέρωση και αλλάζει την κατάσταση από **αποκλειστική** σε **τροποποιημένη**



# Πρωτόκολλο MESI

- **Ευστοχία Εγγραφής - Τροποποιημένη**

1. Ο επεξεργαστής έχει ήδη αποκτήσει αποκλειστική χρήση της γραμμής και έχει μαρκάρει τη γραμμή ως τροποποιημένη, επομένως, **εκτελεί απλώς την ενημέρωση**



# Πρωτόκολλο MESI

---

- Μέχρι τώρα μιλήσαμε μόνο για τη διατήρηση της συνοχής στη κρυφή μνήμη Επιπέδου 2
- Όμως, κάθε επεξεργαστής διαθέτει και μία κρυφή μνήμη Επιπέδου 1, η οποία δεν συνδέεται άμεσα με το δίαυλο

*Τι γίνεται με αυτή;*

- ❖ Πρέπει να υπάρχει μία **τεχνική διατήρησης της ακεραιότητας των δεδομένων** και στα **δύο επίπεδα της κρυφής μνήμης**



# Πρωτόκολλο MESI

---

- Η στρατηγική είναι να επεκταθεί το πρωτόκολλο MESI, ώστε να υλοποιείται και σε κρυφές μνήμες Επιπέδου 1.
- Κάθε γραμμή της κρυφής μνήμης Επιπέδου 1 συμπεριλαμβάνει bits, ώστε να δείχνει την κατάσταση
- **Προσέγγιση 1:** Υιοθέτηση στη κρυφή μνήμη Επιπέδου 1 μιας **πολιτικής δια μέσου εγγραφής**.
- **Προσέγγιση 2:** Υιοθέτηση στη κρυφή μνήμη Επιπέδου 1 μιας **πολιτικής εγγραφής προς τα πίσω**.
  - Είναι πιο πολύπλοκη

# Πρωτόκολλο MESI

---

## Προσέγγιση 1: Υιοθέτηση πολιτικής δια μέσου εγγραφής

- Η δια μέσου εγγραφή γίνεται προς τη κρυφή μνήμη **Επιπέδου 2** και όχι προς την κύρια μνήμη
- Η δια μέσου εγγραφή **επιβάλλει τη μεταφορά κάθε αλλαγής** που λαμβάνει χώρα σε μία γραμμή της, προς την κρυφή μνήμη **Επιπέδου 2** → **γίνεται ορατή** και στις άλλες **κρυφές μνήμες Επιπέδου 2**
- Η χρήση της δια μέσου εγγραφής απαιτεί **τα περιεχόμενα της κρυφής μνήμης Επιπέδου 1 να είναι υποσύνολο της κρυφής μνήμης Επιπέδου 2**