
Προηγμένα Θέματα Αρχιτεκτονικής Συστημάτων

Παράλληλη Επεξεργασία
Φροντιστηριακές Ασκήσεις – Λύσεις

Διδάσκων: Δημόκας Νικόλαος

Άσκηση 1

Έστω ότι a είναι το ποσοστό του κώδικα ενός προγράμματος ο οποίος είναι δυνατόν να εκτελεστεί ταυτόχρονα από n επεξεργαστές ενός συστήματος. Έστω επίσης ότι το υπόλοιπο μέρος του κώδικα θα πρέπει να εκτελεστεί ακολουθιακά, από έναν μόνο επεξεργαστή. Κάθε επεξεργαστής έχει ρυθμό εκτέλεσης ίσο με x MIPS

- Να εξάγετε μία έκφραση για τον ενεργό ρυθμό MIPS ως προς τα n , a , και x , όταν χρησιμοποιείται το σύστημα αποκλειστικά για την εκτέλεση αυτού του προγράμματος.
- Αν $n=16$ και $x=4$ MIPS, να προσδιορίσετε την τιμή του a , η οποία θα δώσει απόδοση συστήματος ίση με 40 MIPS.

Άσκηση 1 – Λύση

- MIPS rate = $(n\alpha + (1 - \alpha)) * x = (n\alpha - \alpha + 1)*x$

- $\alpha = 0.6$

❖ Προσέγγιση 1

1. Διαιρέστε τον αριθμό των εντολών με το χρόνο εκτέλεσης.

- Για παράδειγμα, εάν ένας υπολογιστής ολοκληρώνει 1 million εντολές σε 0.05 seconds, τότε ο υπολογισμός είναι $1 \text{ million} / 0.05 = 20 \text{ million}$.

2. Διαιρέστε τον αριθμό που βρήκατε στο 1 με 1 million

- Για παράδειγμα: $20 \text{ million} / 1 \text{ million} = 20 \text{ MIPS}$.

❖ Προσέγγιση 2

1. Διαιρέστε τον αριθμό των κύκλων ανά second (CPU) με τον αριθμό των κύκλων ανά εντολή (Cycles Per Instruction - CPI)

2. Διαιρέστε τον αριθμό που βρήκατε στο 1 με 1 million

- Για παράδειγμα εάν μία CPU 24 megahertz έχει CPI of 12 τότε:

$$24 * 10^6 / 12 = 2 * 10^6; 2 * 10^6 / 1 \text{ million} = 2 \text{ MIPS}$$

Άσκηση 2

Μπορείτε να προβλέψετε κάποιο πρόβλημα στην προσέγγιση της κρυφής μνήμης με μία μόνον εγγραφή, σε ένα σύστημα πολυεπεξεργαστών βασισμένο σε δίαυλο; Αν ναι, να προτείνετε μία λύση.

Άσκηση 2 – Λύση

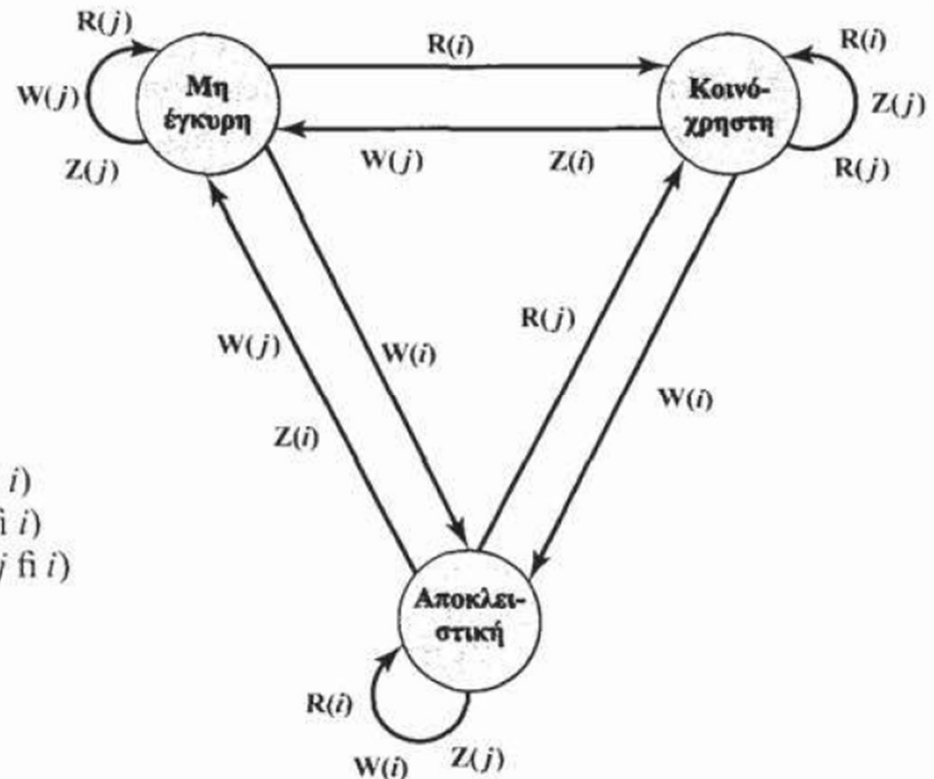
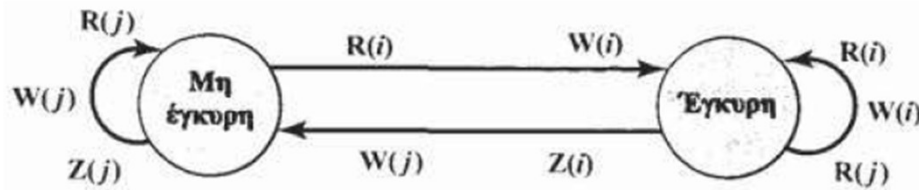
Ο επεξεργαστής A έχει ένα block μνήμης στη cache. Όταν ο A γράψει στο block την πρώτη φορά, τότε η μνήμη ενημερώνεται. Αυτό είναι ένα σήμα στους άλλους επεξεργαστές να ακυρώσουν τα δικά τους αντίγραφα του block.

Οι επόμενες εγγραφές από τον A θα επηρεάσουν μόνο τη cache του A. Εάν ένας άλλος επεξεργαστής προσπαθήσει να διαβάσει το block από τη κύρια μνήμη, το block θα είναι μη έγκυρο.

Λύση: Εάν ο A κάνει μια δεύτερη ενημέρωση, τότε θα πρέπει να επισημάνει αυτό το block στη κύρια μνήμη ως μη έγκυρο. Εάν κάποιος άλλος επεξεργαστής θέλει το block, τότε θα πρέπει να αιτηθεί στον A να εγγράψει την τελευταία έκδοση από τη cache στη κύρια μνήμη.

Άσκηση 3

Το σχήμα απεικονίζει τα διαγράμματα κατάστασης δύο πιθανών πρωτοκόλλων συνοχής της κρυφής μνήμης. Να εξάγετε και να εξηγήσετε κάθε πρωτόκολλο, και να το συγκρίνετε με το πρωτόκολλο MESI.



$W(i)$ = Εγγραφή γραμμής από τον επεξεργαστή i
 $R(i)$ = Ανάγνωση γραμμής από τον επεξεργαστή i
 $Z(i)$ = Μετατόπιση γραμμής από τον επεξεργαστή i
 $W(j)$ = Εγγραφή γραμμής από τον επεξεργαστή j ($j \neq i$)
 $R(j)$ = Ανάγνωση γραμμής από τον επεξεργαστή j ($j \neq i$)
 $Z(j)$ = Μετατόπιση γραμμής από τον επεξεργαστή j ($j \neq i$)

Παρατήρηση: Τα διαγράμματα καταστάσεων αντιστοιχούν σε δεδομένη γραμμή της λανθάνουσας μνήμης

Άσκηση 3 – Λύση

- ❖ Το (a) είναι ένα απλό πρωτόκολλο συνέπειας της cache. Απαιτεί από όλους τους επεξεργαστές να χρησιμοποιούν μία *δια μέσου εγγραφής* πολιτική. Εάν πραγματοποιηθεί μία εγγραφή σε μία κρυφή μνήμη τότε τα αντίγραφα στις απομακρυσμένες κρυφές μνήμες ακυρώνονται. Η προσέγγιση είναι απλή στην υλοποίηση αλλά απαιτεί περισσότερη κυκλοφορία στο δίαυλο και στη κύρια μνήμη εξαιτίας της πολιτικής *δια μέσου εγγραφής*.
- ❖ Το (b) πραγματοποιεί μία διάκριση μεταξύ κοινόχρηστης και αποκλειστικής κατάστασης. Όταν μία cache λαμβάνει πρώτη φορά μία γραμμή τότε την τοποθετεί σε κοινόχρηστη κατάσταση. Εάν η γραμμή είναι σε τροποποιημένη κατάσταση σε μία άλλη cache, τότε η αρχική cache μπλοκάρει την ανάγνωση μέχρι η γραμμή να ενημερωθεί στην κύρια μνήμη (όπως συμβαίνει και στο πρωτόκολλο MESI). Η διάκριση σε κοινόχρηστη και αποκλειστική κατάσταση μειώνει το πλήθος των ακυρωτικών μηνυμάτων.

Δια μέσου εγγραφή: Όλες οι λειτουργίες εγγραφών γίνονται στην κύρια μνήμη και στην κρυφή μνήμη, εξασφαλίζοντας την εγκυρότητα των δεδομένων της κύριας μνήμης.

Άσκηση 4

Θεωρείστε ένα συμμετρικό πολυεπεξεργαστή, ο οποίος διαθέτει κρυφές μνήμες Επιπέδου 1 και 2 και χρησιμοποιεί το πρωτόκολλο MESI. Μία από τις τέσσερις καταστάσεις σχετίζεται με κάθε γραμμή της κρυφής μνήμης Επιπέδου 2. Είναι αναγκαίες και οι τέσσερις καταστάσεις για κάθε γραμμή, στην κρυφή μνήμη Επιπέδου 1;

- Αν ναι, γιατί;
- Αν όχι, εξηγήστε ποια ή ποιες καταστάσεις είναι δυνατόν να εξαλειφθούν.

Άσκηση 4 – Λύση

Εάν η κρυφή μνήμη Επιπέδου 1 χρησιμοποιεί μία πολιτική *δια μέσου εγγραφής* τότε δε χρειάζεται να γνωρίζει τη κατάσταση M (τροποποιημένη). Εάν η κρυφή μνήμη Επιπέδου 1 χρησιμοποιεί την πολιτική *εγγραφή προς τα πίσω*, τότε το πλήρες πρωτόκολλο MESI χρειάζεται να εφαρμοστεί μεταξύ των κρυφών μνημών Επιπέδου 1 και Επιπέδου 2.

Άσκηση 5

Ο παρακάτω πίνακας δείχνει την απόδοση μίας διάταξης δύο επιπέδων κρυφής μνήμης για ένα υπολογιστικό σύστημα.

Υποσύστημα Μνήμης	Επιβάρυνση Προσπέλασης (Κύκλοι της PU)	Μέγεθος	Ρυθμός Ευστοχίας (%)
Κρυφή μνήμη Επιπέδου 1	1	32 KB	90
Κρυφή μνήμη Επιπέδου 2	5	256 KB	7
Μνήμη	32	8 GB	3

- Να προσδιορίσετε την επιβάρυνση προσπέλασης (μέσο πλήθος κύκλων της PU) για ένα σύστημα το οποίο διαθέτει μόνο κρυφή μνήμη Επιπέδου 1 και να κανονικοποιήσετε αυτή τη τιμή στην τιμή 1.0.
- Να προσδιορίσετε την επιβάρυνση προσπέλασης (μέσο πλήθος κύκλων της PU) για ένα σύστημα το οποίο διαθέτει κρυφή μνήμη Επιπέδου 1 και κρυφή μνήμη Επιπέδου 2 και να κανονικοποιήσετε αυτή τη τιμή στην τιμή 1.0.

Άσκηση 5 – Λύση

Εάν χρησιμοποιείται μόνο η κρυφή μνήμη Επιπέδου 1 τότε 90% των προσπελάσεων είναι προς τη κρυφή μνήμη Επιπέδου 1 και το υπόλοιπο 10% είναι προς τη κύρια μνήμη.

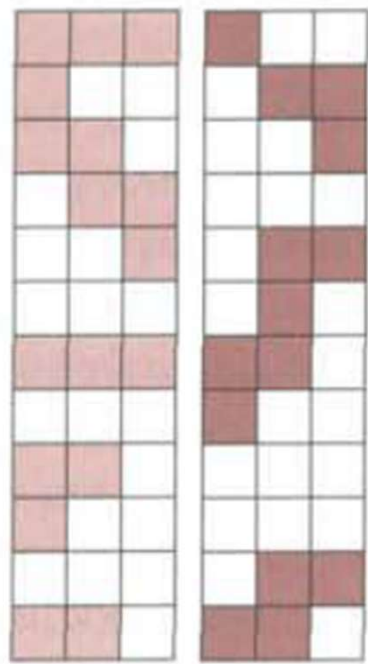
- Επιβάρυνση Προσπέλασης = $1*0.9 + 32*0.1 = 4.1$

Εάν χρησιμοποιούνται η κρυφή μνήμη Επιπέδου 1 και 2 τότε:

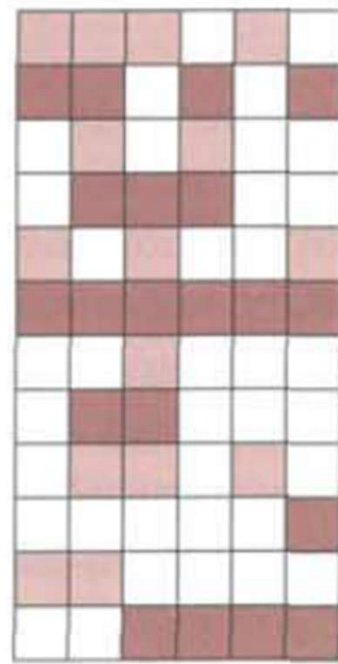
- Επιβάρυνση Προσπέλασης = $1*0.9 + 5*0.07 + 32*0.03 = 2.21$
- Κανονικοποιημένη Επιβάρυνση = $2.21 / 4.1 = 0.53$
- Η προσθήκη της κρυφής μνήμης Επιπέδου 2 μειώνει την επιβάρυνση στο 53% σε σχέση με τη χρήση μόνο κρυφής μνήμης Επιπέδου 1.

Άσκηση 6

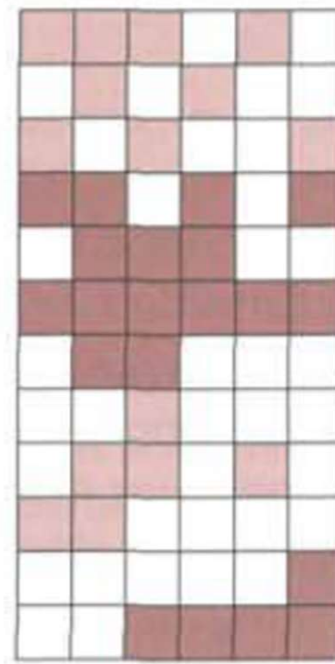
Ποια εναλλακτική μορφή οργάνωσης προτείνει κάθε τμήμα του Σχήματος



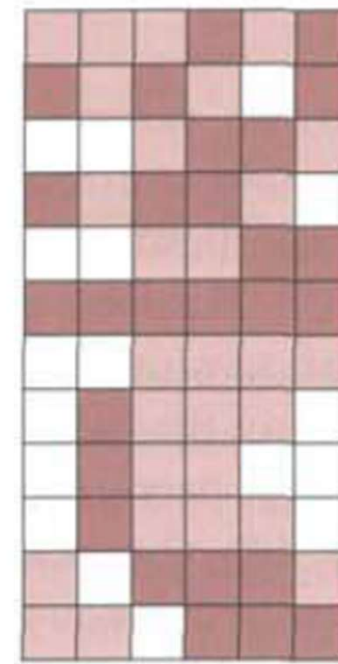
(α)



(β)



(γ)



(δ)

Άσκηση 6 – Λύση

- a. Chip Πολυεπεξεργαστών
- b. Υπερβαθμωτή φυλλωμένη πολυνημάτωση
- c. Υπερβαθμωτή παρεμποδισμένη πολυνημάτωση
- d. Ταυτόχρονη πολυνημάτωση

Άσκηση 7

Θεωρείστε την απεικόνιση μίας λειτουργίας διασωλήνωσης στο Σχήμα α. Σε αυτή την απεικόνιση, αγνοούνται τα στάδια ανάκλησης και αποκωδικοποίησης και παρουσιάζεται η εκτέλεση ενός νήματος A. Το Σχήμα β απεικονίζει την εκτέλεση ενός διαφορετικού νήματος B. Στις δύο αυτές περιπτώσεις, χρησιμοποιείται ένας απλός επεξεργαστής με διασωλήνωση.

- Να δείξετε ένα διάγραμμα έκδοσης εντολών, παρόμοιο με εκείνο του Σχήματος που βρίσκεται στη Διάλεξη 2 slide 16, για καθένα από τα δύο νήματα.
- Υποθέστε ότι τα δύο νήματα πρόκειται να εκτελεστούν παράλληλα σε ένα chip πολυεπεξεργαστών, όπου καθένας από τους δύο επεξεργαστές του chip χρησιμοποιεί μία απλή διασωλήνωση. Να δείξετε ένα διάγραμμα έκδοσης εντολών.

	CO	FO	EI	WO
1	A1			
2	A2	A1		
3	A3	A2	A1	
4	A4	A3	A2	A1
5	A5	A4	A3	A2
6				A3
7				
8	A15			
9	A16	A15		
10		A16	A15	
11			A16	A15
12				A16

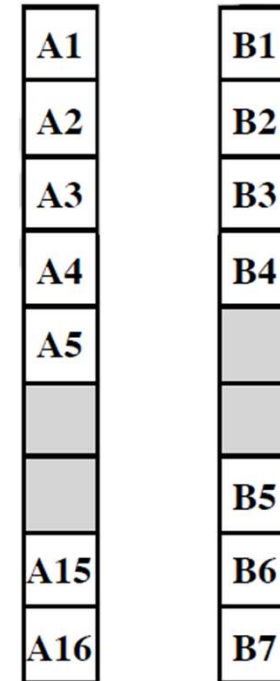
(α)

	CO	FO	EI	WO
1	B1			
2	B2	B1		
3	B3	B2	B1	
4	B4	B3	B2	B1
5			B3	B2
6				B3
7	B5	B4		
8	B6	B5	B4	
9	B7	B6	B5	B4
10		B7	B6	B5
11			B7	B6
12				B7

(β)

Άσκηση 7 – Λύση

a. Η λύση του ερωτήματος a παρουσιάζεται στο σχήμα



b. Οι δύο διασωληνώσεις λειτουργούν ανεξάρτητα σε δύο χωριστούς επεξεργαστές στο ίδιο chip. Επομένως τα διαγράμματα της εκφώνησης και η λύση του ερωτήματος a μπορούν να εφαρμοστούν.