

```
R9 ← R3 + R4  
LOAD R4 ← M[X]  
R10 ← R3 + R7  
STORE M[Y] ← R10
```

■

Κάποιες λύσεις του ανωτέρω προβλήματος έχουν προταθεί που βασίζονται στο μεταγλωττιστή.

Οι επεξεργαστές PIMME που κυκλοφόρησαν εμπορικά είναι πολύ λιγότεροι των υπερβαθμωτών. Ως παράδειγμα επεξεργαστών PIMME αναφέρουμε τους Trimedia, την οικογένεια ST200 και τον Intel Itanium (έχει και πολλά άλλα πρόσθετα χαρακτηριστικά).

Ενότητα 7.3 Πολυνημάτωση σε επίπεδο μίας κεντρικής μονάδας επεξεργασίας

Διαδικασία (process) είναι ένα πρόγραμμα που βρίσκεται σε κάποια φάση της εκτέλεσής του μαζί με τις τρέχουσες τιμές του μετρητή προγράμματος, των καταχωρητών και των μεταβλητών. Κάθε διαδικασία έχει το δικό της χώρο λογικών διευθύνσεων. Όπως αναφέραμε στην εισαγωγή σε υπολογιστές που χρησιμοποιούν την τεχνική του πολυπρογραμματισμού υπάρχουν περισσότερα από ένα προγράμματα ή τμήματα προγραμμάτων στην κυρία μνήμη του υπολογιστή (διαδικασίες). Όταν κατά την εκτέλεση κάποιας διαδικασίας απαιτείται προσπέλαση του δίσκου, η λειτουργία εισόδου/εξόδου υλοποιείται από το σύστημα εισόδου/εξόδου και ο επεξεργαστής αντί να μείνει ανενεργός και να περιμένει να ολοκληρωθεί η λειτουργία εισόδου/εξόδου για να συνεχίσει την εκτέλεση της διαδικασίας, θα ασχοληθεί με την εκτέλεση άλλης διαδικασίας. Για να μπορεί κάποια στιγμή να συνεχιστεί η εκτέλεση της διαδικασίας που διακόπηκε, θα πρέπει πριν αρχίσει η εκτέλεση της νέας διαδικασίας να αποθηκευτεί όλη η πληροφορία που περιγράφει την τρέχουσα κατάσταση της διαδικασίας (context) που διακόπηκε (π.χ. τιμή μετρητή προγράμματος, τιμές καταχωρητών, τιμές σημαιών κατάστασης κλπ.) και να φορτωθούν οι τιμές της νέας διαδικασίας. Γι' αυτό το λόγο η αλλαγή της διαδικασίας που εκτελείται σε μία ΚΜΕ απαιτεί εκατοντάδες έως και χιλιάδες κύκλων ρολογιού του επεξεργαστή. Όμως επειδή η λειτουργία εισόδου/εξόδου είναι πολύ πιο χρονοβόρα διαδικασία συμφέρει αντί η ΚΜΕ να μένει ανενεργή να γίνει η αλλαγή της εκτελούμενης διαδικασίας.

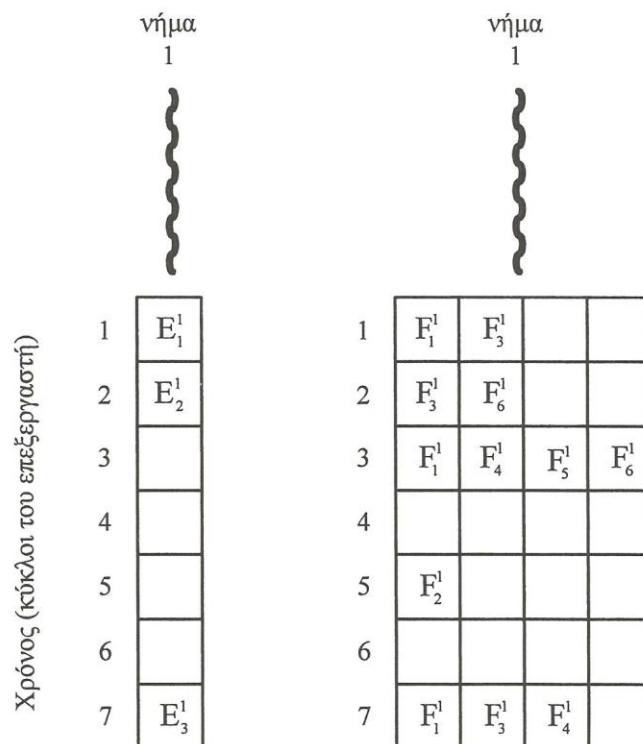
Όταν μιλήσαμε για τον μηχανισμό μερικώς επικαλυπτόμενων λειτουργιών (pipelining), θεωρήσαμε ότι όλες οι εντολές και τα δεδομένα βρίσκονται στην κρυφή μνήμη εντολών και δεδομένων αντίστοιχα πρώτου επιπέδου. Αυτό προφανώς δεν ισχύει στην πραγματικότητα. Κάποια εντολή ή δεδομένα όταν ζητηθούν δεν θα βρεθούν στην κρυφή μνήμη εντολών και δεδομένων πρώτου επιπέδου και θα πρέπει να προσκομιστούν από την κρυφή μνήμη δεύτερου ή τρίτου επιπέδου ή ακόμη χειρότερα από την κύρια μνήμη. Ακόμη και σ' ένα υπερβαθμωτό επεξεργαστή με μέγιστο ρυθμό αποστολής εντολών προς εκτέλεση $k, k > 1$, που υλοποιεί την τακτική της εκτός σειράς αποστολής εντολών προς εκτέλεση, δεν είναι δυνατόν να καλυφτούν όλοι οι κύκλοι αναμονής χωρίς να μείνουν ανενεργές λειτουργικές μονάδες της ΚΜΕ.

Το σχήμα 7.21. α περιγράφει την περίπτωση στην οποία σε ένα βαθμωτό επεξεργαστή μερικώς επικαλυπτόμενων εντολών, η εντολή E_2^1 απαιτεί την προσπέλαση της κρυφής μνήμης δεύτερου επιπέδου λόγω αποτυχίας της κρυφής μνήμης πρώτου επιπέδου. Θεωρήσαμε ότι η προσπέλαση του δεύτερου επιπέδου κρυφής μνήμης κοστίζει 5 κύκλους ρολογιού, γι' αυτό τους επόμενους 4 κύκλους ρολογιού δεν προσκομίζεται εντολή (στο σχήμα έχουμε άδεια κουτιά), παγώνει το περιεχόμενο των προηγούμενων βαθμίδων του μηχανισμού μερικώς επικαλυπτόμενων λειτουργιών. Βαθμωτοί επεξεργαστές μερικώς επικαλυπτόμενων εντολών είναι οι επεξεργαστές μερικώς επικαλυπτόμενων εντολών που σε κάθε κύκλο ρολογιού προσκομίζεται το πολύ μια εντολή. Είναι οι επεξεργαστές με τους οποίους ασχοληθήκαμε στο κεφάλαιο 4.

Το σχήμα 7.21.β περιγράφει υπερβαθμωτό επεξεργαστή με μέγιστο ρυθμό αποστολής τεσσάρων εντολών. Παρατηρούμε ότι λόγω εξαρτήσεων αφενός σε κάθε κύκλο ρολογιού δεν αποστέλλεται προς εκτέλεση στις λειτουργικές μονάδες ο μέγιστος αριθμός εντολών 4, αφετέρου υπάρχουν και κύκλοι ρολογιού που δεν αποστέλλεται στις λειτουργικές μονάδες καμία εντολή προς εκτέλεση.

Στις ανωτέρω περιπτώσεις, το πλήθος των κύκλων ρολογιού για τους οποίους μένουν ανενεργοί πόροι του επεξεργαστή (πχ. λειτουργικές μονάδες) είναι μικρό σε σχέση με το κόστος αλλαγής της εκτελούμενης διαδικασίας (processes). Επομένως δεν συμφέρει να χρησιμοποιηθεί η τεχνική της αλλαγής της εκτελούμενης διαδικασίας.

Για την επίλυση του προβλήματος των ανενεργών πόρων ενός επεξεργαστή, έχει προταθεί η τεχνική της πολυνημάτωσης (multithreading) σε επίπεδο επεξεργαστή. Ο επεξεργαστής μπορεί να διαχειριστεί ως διαφορετικές οντότητες περισσότερες από μία ακολουθίες εντολών, νήματα (threads ή threads of control), και εάν δεν μπορεί να εκτελεστεί κάποια εντολή ενός νήματος ο επεξεργαστής μπορεί να εκτελέσει εντολές άλλου νήματος για να μη μείνουν μονάδες του ανενεργές. Τα νήματα μπορούν να ανήκουν στην ίδια ή σε διαφορετικές διαδικασίες.



Σχήμα 7.21. α. Βαθμωτός επεξεργαστής μερικώς επικαλυπτόμενων εντολών που υποστηρίζει ένα νήμα. E_i^j : δηλώνει την εντολή i του νήματος j . β. Υπερβαθμωτός επεξεργαστής με μέγιστο ρυθμό αποστολής τεσσάρων εντολών προς εκτέλεση. F_i^j : δηλώνει την αποστολή εντολής του νήματος j στη λειτουργική μονάδα i .

Ο μέγιστος αριθμός νημάτων που μπορεί να διαχειριστεί από ένα επεξεργαστή είναι σταθερός και καθορίζεται κατά τη φάση σχεδίασης του επεξεργαστή.

Από πλευράς λειτουργικού συστήματος το νήμα είναι η μικρότερη μονάδα επεξεργασίας που μπορεί να χρονοπρογραμματιστεί από το λειτουργικό σύστημα. Μια διαδικασία αποτελείται από ένα ή περισσότερα νήματα, κάθε ένα των οποίων έχει το δικό του περιβάλλον (context). Ένα μέρος του περιβάλλοντος μίας διαδικασίας είναι κοινό για όλα τα νήματα της διαδικασίας. Ως παράδειγμα αναφέρουμε ότι όλα τα νήματα μιας διαδικασίας μοιράζονται τον ίδιο χώρο λογικών διευθύνσεων. Επομένως η αλλαγή

από ένα νήμα σε άλλο νήμα γίνεται πολύ πιο γρήγορα απ' ό τι η αλλαγή από μία διαδικασία σε άλλη διαδικασία.

Τρεις είναι οι κύριες τεχνικές πολυνημάτωσης σε επίπεδο ενός επεξεργαστή:

- α. Εναλλαγή νημάτων σε επίπεδο εντολής (fine-grain multithreading ή cycle-by-cycle interleaving).
- β. Εναλλαγή νημάτων σε επίπεδο ομάδας εντολών (coarse-grain multithreading ή block interleaving).
- γ. Ταυτόχρονη πολυνημάτωση (simultaneous multithreading).

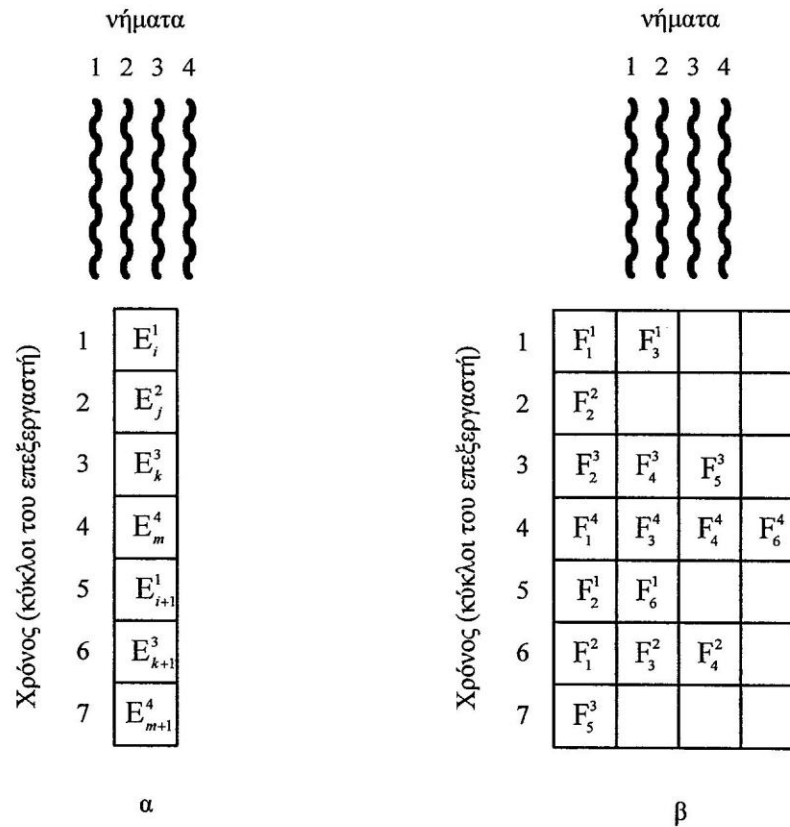
Στη συνέχεια θα μελετήσουμε κάθε μία από τις ανωτέρω αναφερθείσες τεχνικές τόσο για βαθμωτούς επεξεργαστές μερικώς επικαλυπτόμενων εντολών όσο και για υπερβαθμωτούς επεξεργαστές.

7.3.1 Εναλλαγή νημάτων σε επίπεδο εντολής

Σε κάθε κύκλο ρολογιού εκτελείται κυκλικά άλλο νήμα. Επομένως σε κάθε κύκλο ρολογιού έχουμε αλλαγή περιβάλλοντος (context) σε επίπεδο νήματος. Κάθε νήμα έχει το δικό της σύνολο καταχωρητών γενικού σκοπού και μετρητή προγράμματος. Η αλλαγή περιβάλλοντος γίνεται στιγμιαία χωρίς να επιφέρει καμία καθυστέρηση χρησιμοποιώντας κάθε φορά ένα άλλο σύνολο καταχωρητών.

Εφαρμόζοντας αυτή την τεχνική στους βαθμωτούς επεξεργαστές μερικώς επικαλυπτόμενων λειτουργιών σε κάθε κύκλο ρολογιού του επεξεργαστή προσκομίζεται κυκλικά μια εντολή από άλλο νήμα. Ουσιαστικά μια εντολή ενός νήματος τροφοδοτεί το μηχανισμό μερικώς επικαλυπτόμενων λειτουργιών μετά την ολοκλήρωση της προηγούμενης εντολής του ίδιου νήματος. Υπενθυμίζουμε ότι ολοκλήρωση εντολής σημαίνει ότι η εντολή εκτελέστηκε και το αποτέλεσμα αποθηκεύτηκε στον καταχωρητή ή στη θέση μνήμης που δηλώνεται στην εντολή. Ακόμη και αν έρθει η σειρά του νήματος του οποίου εκκρεμεί η ολοκλήρωση εντολής, ο μηχανισμός μερικώς επικαλυπτόμενων λειτουργιών δεν τροφοδοτείται με εντολή αυτού του νήματος, τροφοδοτείται με εντολή του επόμενου στη σειρά νήματος που δεν εκκρεμεί η ολοκλήρωση εντολής. Μ' αυτό τον τρόπο οι καθυστερήσεις λόγω προσπέλασης του συστήματος μνήμης αποκρύπτονται και δεν μειώνεται η απόδοση του επεξεργαστή. Όταν το πλήθος των νημάτων είναι τουλάχιστον ίσο με το πλήθος των βαθμίδων του μηχανισμού μερικώς επικαλυπτόμενων λειτουργιών του επεξεργαστή, η εναλλαγή νημάτων σε επίπεδο εντολών περιορίζει την εμφάνιση διαδικασιακών εξαρτήσεων και εξαρτήσεων δεδομένων. Στο σχήμα 7.22.α δίνεται ένα παράδειγμα εφαρμογής της τεχνικής εναλλαγής νημάτων σε επίπεδο εντολής

σε βαθμωτό επεξεργαστή μερικώς επικαλυπτόμενων εντολών που υποστηρίζει τέσσερα νήματα. Θεωρήσαμε ότι η εντολή E_j^2 απαιτεί 5 κύκλους ρολογιού για να ολοκληρωθεί.



Σχήμα 7.22. α. Εφαρμογή της τεχνικής εναλλαγής νημάτων σε επίπεδο εντολής σε βαθμωτό επεξεργαστή μερικώς επικαλυπτόμενων εντολών που υποστηρίζει τέσσερα νήματα. E_i^j : δηλώνει την εντολή i του νήματος j . Θεωρήσαμε ότι η εντολή E_j^2 απαιτεί 5 κύκλους ρολογιού για να ολοκληρωθεί. **β.** Εφαρμογή της τεχνικής εναλλαγής νημάτων σε επίπεδο εντολής σε υπερβαθμωτό επεξεργαστή που υποστηρίζει 4 νήματα. F_i^j : δηλώνει την αποστολή εντολής του νήματος j στη λειτουργική μονάδα i .

Σύμφωνα με την τεχνική εναλλαγής νημάτων σε επίπεδο εντολής, σ' ένα υπερβαθμωτό επεξεργαστή σε κάθε κύκλο ρολογιού στέλνονται προς εκτέλεση εντολές άλλου νήματος. Στο σχήμα 7.22.β δίνεται ένα παράδειγμα υπερβαθμωτού επεξεργαστή

με μέγιστο ρυθμό αποστολής τεσσάρων εντολών προς εκτέλεση σε έξι διαφορετικές λειτουργικές μονάδες που υποστηρίζει 4 νήματα. Παρατηρούμε ότι λόγω εξαρτήσεων και πάλι σε κάθε κύκλο ρολογιού δεν αποστέλλεται προς εκτέλεση στις λειτουργικές μονάδες ο μέγιστος αριθμός εντολών.

Το κύριο μειονέκτημα της τεχνικής εναλλαγής νημάτων σε επίπεδο εντολής είναι ότι επιβραδύνεται η ολοκλήρωση της εκτέλεσης ενός νήματος στο οποίο δεν υπάρχουν εμπόδια της συνέχισης της εκτέλεσής του, λόγω της εκτέλεσης εντολών άλλων νημάτων.

Ως παράδειγμα επεξεργαστή που εφαρμόζει την τεχνική εναλλαγής νημάτων σε επίπεδο εντολής μπορούμε να αναφέρουμε τον επεξεργαστή T1 της SUN ο οποίος περιέχει οκτώ πυρήνες (για επεξεργαστές με πολλούς πυρήνες θα μιλήσουμε στην επόμενη ενότητα) που καθένας υποστηρίζει τέσσερα νήματα.

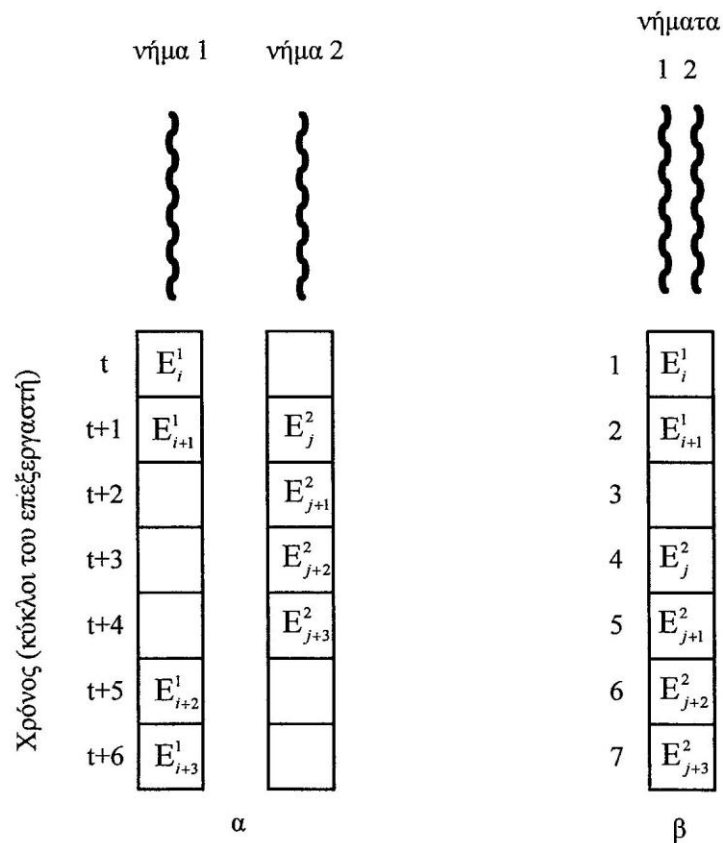
7.3.2 Εναλλαγή νημάτων σε επίπεδο ομάδας εντολών

Εκτελείται ένα νήμα έως ότου να μην μπορεί πλέον να συνεχιστεί η εκτέλεσή του λόγω κάποιου γεγονότος που επιβάλλει μεγάλη καθυστέρηση (πχ. τα ζητούμενα δεδομένα δεν βρίσκονται στην κρυφή μνήμη πρώτου επιπέδου και πρέπει να προσκομιστούν από άλλο επίπεδο του συστήματος μνήμης). Τότε λαμβάνει χώρα αλλαγή περιβάλλοντος (context switch) σε επίπεδο νήματος και αρχίζει η εκτέλεση άλλου νήματος. Ανάλογα του τρόπου υλοποίησης της αλλαγής περιβάλλοντος είναι δυνατόν να μείνουν ανενεργοί πόροι της κεντρικής μονάδας επεξεργασίας για έναν ή περισσότερους κύκλους ρολογιού.

Σε σύγκριση με την τεχνική εναλλαγής νημάτων σε επίπεδο εντολών χρειάζεται ένας μικρότερος αριθμός από νήματα και δεν αυξάνεται σημαντικά ο χρόνος εκτέλεσης ενός νήματος, διότι δεν σταματάει η εκτέλεσή του παρά μόνο όταν δεν μπορεί να συνεχιστεί. Ωστόσο αυτή η τεχνική έχει το μειονέκτημα ότι δεν μπορεί να αποκρύψει τις μικρές καθυστερήσεις που μπορεί να προκύψουν στο μηχανισμό μερικώς επικαλυπτόμενων λειτουργιών (πχ. λόγω εξαρτήσεων από δεδομένα ή διαδικασιακών εξαρτήσεων).

Στα σχήματα 7. 23 και 7. 24 δίνεται αντίστοιχα η εφαρμογή της τεχνικής σε ένα βαθμωτό επεξεργαστή μερικώς επικαλυπτόμενων εντολών που υποστηρίζει δύο νήματα και σε υπερβαθμωτό επεξεργαστή που υποστηρίζει δύο νήματα και έχει μέγιστο ρυθμό αποστολής τεσσάρων εντολών προς εκτέλεση σε έξι διαφορετικές λειτουργικές μονάδες. Και στις δύο περιπτώσεις θεωρήσαμε ότι κατά την αλλαγή του εκτελούμενου νήματος χάνεται ένας κύκλος ρολογιού.

Ο RS64-II ή Northstar της IBM ήταν ο πρώτος επεξεργαστής (εισήχθη το 1998) ο οποίος εφάρμοσε αυτή την τεχνική υποστηρίζοντας δύο νήματα.



Σχήμα 7.23. α. Δίνονται οι καθυστερήσεις που υπεισέρχονται στην περίπτωση στην οποία κάθε ένα από τα δύο νήματα εκτελείται μόνο του σε ένα βαθμωτό επεξεργαστή μερικώς επικαλυπτόμενων εντολών. E_i^j : δηλώνει την εντολή i του νήματος j .

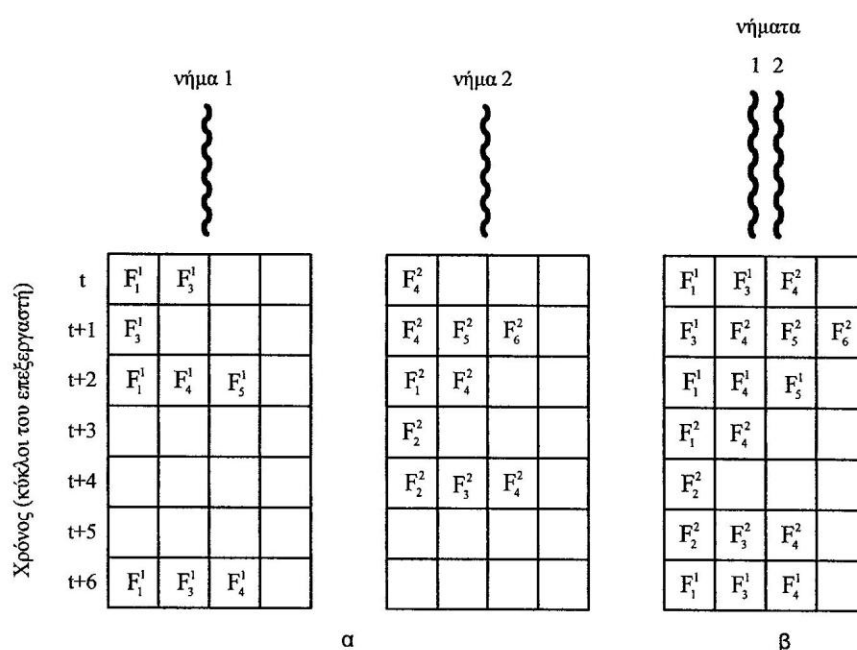
β. Εφαρμογή της τεχνικής εναλλαγής νημάτων σε επίπεδο ομάδας εντολών σε βαθμωτό επεξεργαστή μερικώς επικαλυπτόμενων εντολών που υποστηρίζει δύο νήματα.

7.3.3 Ταυτόχρονη πολυνημάτωση

Όταν σ' ένα υπερβαθμωτό επεξεργαστή οι εξαρτήσεις μέσα σε ένα νήμα δεν επιτρέπουν την αποστολή σ' ένα κύκλο ρολογιού του μέγιστου πλήθους εντολών προς εκτέλεση στις λειτουργικές μονάδες, τότε μπορούν να σταλούν προς εκτέλεση εντολές και από άλλο νήμα. Σ' αυτή την περίπτωση δεν έχουμε αλλαγή περιβάλλοντος από νήμα

πολυνημάτωση μπορεί να στείλει εντολές από πολλά νήματα προς εκτέλεση στις λειτουργικές μονάδες, άρα εκμεταλλεύεται τόσο την παραλληλία μεταξύ εντολών του ίδιου νήματος όσο και την παραλληλία μεταξύ νημάτων.

Στο σχήμα 7. 25 δίνεται υπερβαθμωτός επεξεργαστής που υποστηρίζει δύο νήματα και έχει μέγιστο ρυθμό αποστολής τεσσάρων εντολών προς εκτέλεση σε έξι διαφορετικές λειτουργικές μονάδες. Παρατηρούμε ότι και σε αυτή την περίπτωση υπάρχουν κύκλοι ρολογιού κατά τους οποίους δεν αποστέλλονται εντολές προς εκτέλεση σε λειτουργικές μονάδες (κενά κουτιά στον πίνακα). Ωστόσο παρατηρούμε ότι τα κενά κουτιά στον πίνακα β του σχήματος 7.25.β είναι πολύ λιγότερα από αυτά στον πίνακα β του σχήματος 7. 24. Τα κενά κουτιά στον πίνακα β του σχήματος 7.25.β θα μειώνονταν εάν ο υπερβαθμωτός επεξεργαστής υποστήριζε μεγαλύτερο αριθμό νημάτων.



Σχήμα 7.25. α. Δίνονται οι καθυστερήσεις που υπεισέρχονται στην περίπτωση στην οποία κάθε ένα από τα δύο νήματα εκτελούνταν μόνο του σε ένα υπερβαθμωτό επεξεργαστή με μέγιστο ρυθμό αποστολής τεσσάρων εντολών προς εκτέλεση σε έξι διαφορετικές λειτουργικές μονάδες. F_i^j : δηλώνει την αποστολή εντολής του νήματος j στη λειτουργική μονάδα i . **β.** Εφαρμογή της τεχνικής ταυτόχρονης πολυνημάτωσης σε ένα υπερβαθμωτό επεξεργαστή με μέγιστο ρυθμό αποστολής τεσσάρων εντολών προς εκτέλεση σε έξι διαφορετικές λειτουργικές μονάδες.

Ο επεξεργαστής της Intel Pentium 4 υποστηρίζει δύο νήματα υλοποιώντας μία τεχνική πολυνημάτωσης, την οποία η Intel καλεί hyperthreading και ιδεατά είναι όμοια με την τεχνική της ταυτόχρονης πολυνημάτωσης. Ως άλλα παραδείγματα επεξεργαστών που εφαρμόζουν την τεχνική ταυτόχρονης πολυνημάτωσης μπορούμε να αναφέρουμε τον επεξεργαστή Intel i7 και IBM Power7 οι οποίοι υποστηρίζουν 2 και 4 νήματα αντίστοιχα ανά πυρήνα.

Ενότητα 7.4 Πολυεπεξεργαστές και πολυπύρρηνοι επεξεργαστές

Η απαίτηση για σχεδίαση και κατασκευή υπολογιστών μεγαλύτερης απόδοσης είναι συνεχής. Η αύξηση της απόδοσης επετεύχθη σε μεγάλο βαθμό με την εξέλιξη της τεχνολογίας (μικροηλεκτρονική). Η συνεχής μείωση των διαστάσεων του τρανζίστορ οδήγησε αφενός στη συνεχή αύξηση της συχνότητας λειτουργίας των κυκλωμάτων και αφετέρου έδωσε τη δυνατότητα στην ίδια επιφάνεια πυριτίου να υλοποιηθούν περισσότερες μονάδες του επεξεργαστή. Αρχικά η κεντρική μονάδα επεξεργασίας, KME, υλοποιούνταν με περισσότερα από ένα ολοκληρωμένα κυκλώματα. Με την εξέλιξη της τεχνολογίας κατέστη δυνατόν η KME να υλοποιείται σε ένα ολοκληρωμένο κύκλωμα το οποίο καλούμε μικροεπεξεργαστή. Με την περαιτέρω βελτίωση της τεχνολογίας στην ίδια επιφάνεια πυριτίου μαζί με τα κυκλώματα της κεντρικής μονάδας επεξεργασίας υλοποιούμε και άλλες μονάδες του υπολογιστή όπως ένα, δύο ή και τρία επίπεδα κρυφής μνήμης καθώς και τη μονάδα διαχείρισης της ιδεατής μνήμης, με συνέπεια περαιτέρω αύξηση της απόδοσης του επεξεργαστή (η επικοινωνία μεταξύ μονάδων που βρίσκονται στην ίδια επιφάνεια πυριτίου βάζει μικρότερες καθυστερήσεις λόγω της μικρότερης χωρητικότητας των γραμμών διασύνδεσης). Η τεχνική των μερικώς επικαλυπτόμενων εντολών (pipelining) μείωσε το χρόνο εκτέλεσης του προγράμματος, δηλαδή αύξησε την απόδοση, επιτρέποντας να ξεκινά η προσκόμιση και εκτέλεση μιας εντολής πριν ολοκληρωθεί η προηγούμενη. Οι υπερβαθμωτοί (superscalar) επεξεργαστές και οι επεξεργαστές πολύ μεγάλου μήκους εντολών (VLIW) μειώνουν το χρόνο εκτέλεσης ενός προγράμματος εκμεταλλευόμενοι την παραλληλία σε επίπεδο εντολών (Instruction Level Parallelism, ILP) του προγράμματος, δηλαδή εντολές του προγράμματος που είναι ανεξάρτητες μεταξύ τους εκτελούνται παράλληλα σε διαφορετικές λειτουργικές μονάδες. Ένα κύριο χαρακτηριστικό των ανωτέρω τεχνικών είναι ότι ο προγραμματιστής δεν χρειάζεται να γράψει το πρόγραμμά του με συγκεκριμένο τρόπο για να αξιοποιηθούν οι ανωτέρω τεχνικές. Οι επεξεργαστές που υποστηρίζουν την τεχνική της πολυνημάτωσης (multithreading) μειώνουν το χρόνο εκτέλεσης ενός προγράμματος, το οποίο έχει γραφεί κατάλληλα για την υποστήριξη

αυτής της τεχνικής, εκμεταλλευόμενοι την παραλληλία που υπάρχει μεταξύ των νημάτων.

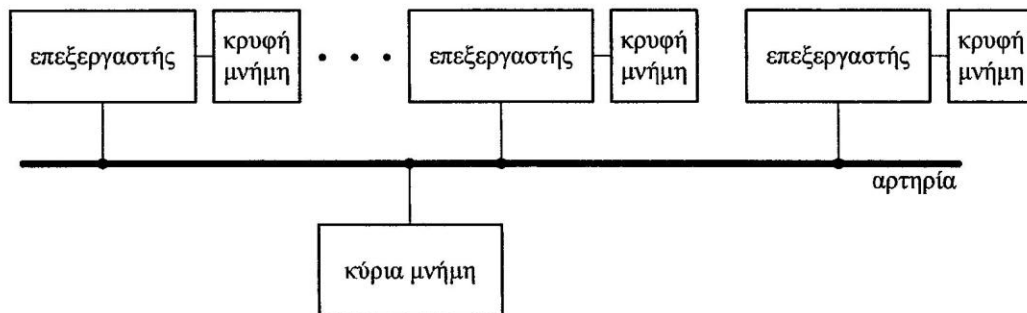
Παράλληλα με τις τεχνικές που αναφέρθηκαν ανωτέρω και είχαν ως στόχο την αύξηση της απόδοσης των επεξεργαστών, οι σχεδιαστές υπολογιστικών συστημάτων ανέπτυξαν τεχνικές διασύνδεσης πολλών από τους διαθέσιμους επεξεργαστές για την κατασκευή πιο ισχυρών υπολογιστών. Τα συστήματα αυτά καλούνται πολυεπεξεργαστές (multiprocessors). Στο παρελθόν κάθε επεξεργαστής υλοποιούνταν σε διαφορετικό ολοκληρωμένο κύκλωμα. Ο χρήστης, ανάλογα με τις ανάγκες του, μπορεί να αγοράσει ένα πολυεπεξεργαστή με περισσότερους ή λιγότερους επεξεργαστές. Στην περίπτωση που εκτελούμε ένα σύνολο από ανεξάρτητα προγράμματα, ένα πρόγραμμα σε κάθε επεξεργαστή, είναι προφανές ότι αυξάνεται το υπολογιστικό έργο, δηλαδή το πλήθος των προγραμμάτων που εκτελείται σε ένα συγκεκριμένο χρονικό διάστημα. Εάν θέλουμε όμως να μειώσουμε το χρόνο εκτέλεσης ενός προγράμματος θα πρέπει τμήματα του προγράμματος να εκτελούνται σε διαφορετικούς επεξεργαστές. Επομένως το πρόγραμμα θα πρέπει να έχει γραφεί ως πολυνηματικό (multithreaded), παράλληλο, πρόγραμμα. Ένα ακόμη χρήσιμο χαρακτηριστικό των πολυεπεξεργαστικών συστημάτων είναι ότι μπορούν να σχεδιαστούν έτσι ώστε να συνεχίζουν να λειτουργούν, με μειωμένη απόδοση, όταν κάποιος επεξεργαστής τεθεί εκτός λειτουργίας λόγω βλάβης, δηλαδή έχουν βελτιωμένη διαθεσιμότητα (availability).

Ενώ η μείωση των διαστάσεων του τρανζίστορ συνεχίζεται, δεν είναι δυνατόν να αυξήσουμε τη συχνότητα λειτουργίας του επεξεργαστή, διότι η καταναλισκόμενη ισχύς ανά μονάδα επιφάνειας (πυκνότητα καταναλισκώμενης ισχύος) θα ξεπεράσει τα επιτρεπτά όρια. Επίσης μελέτες έχουν δείξει ότι η περαιτέρω εκμετάλλευση της παραλληλίας σε επίπεδο εντολών ενός προγράμματος θα οδηγήσει σε μικρή αύξηση της απόδοσης σε σχέση με την κατανάλωση ισχύος. Επομένως φαίνεται ότι οι πολυεπεξεργαστές αποτελούν τη λύση για περαιτέρω αύξηση της απόδοσης. Λαμβάνοντας υπ' όψιν ότι το πλήθος των κυκλωμάτων που μπορεί να υλοποιηθεί στην ίδια επιφάνεια πυριτίου συνεχίζει να αυξάνεται, κατασκευάζονται στην ίδια επιφάνεια πυριτίου περισσότερες από μία κεντρικές μονάδες επεξεργασίας, τις οποίες καλούμε πυρήνες (cores), μαζί με κρυφές μνήμες και το σύστημα διασύνδεσης όλων αυτών των μονάδων ώστε να λειτουργούν ως ένα σύστημα το οποίο καλούμε πολυπύρρηνο επεξεργαστή (multicore processor). Παρατηρούμε λοιπόν ότι ο πολυπύρρινος επεξεργαστής είναι ένας πολυεπεξεργαστής που υλοποιείται σε ένα ολοκληρωμένο κύκλωμα. Ο αριθμός των πυρήνων αναμένεται να διπλασιάζεται κάθε δύο χρόνια. Στη συνέχεια όταν αναφερόμαστε σε επεξεργαστή θα εννοούμε επεξεργαστή που περιέχει μια κεντρική μονάδα επεξεργασίας. Όταν θέλουμε να αναφερθούμε σε πολυπύρρινους επεξεργαστές θα το δηλώνουμε ρητά.

7.4.1 Πολυεπεξεργαστές μοιραζόμενης μνήμης (shared memory multiprocessors)

Ένας πολυεπεξεργαστής στον οποίο οι κεντρικές μονάδες επεξεργασίας μοιράζονται τον ίδιο χώρο φυσικών διευθύνσεων καλείται πολυεπεξεργαστής μοιραζόμενης μνήμης. Οι επεξεργαστές επικοινωνούν μέσω κοινών μεταβλητών που αποθηκεύονται στη μνήμη, και όλοι οι επεξεργαστές είναι ικανοί να προσπελάσουν οποιαδήποτε θέση μνήμης με εντολές LOAD και STORE. Οι πολυεπεξεργαστές μοιραζόμενης μνήμης ταξινομούνται σε δύο κατηγορίες:

α. Πολυεπεξεργαστές στους οποίους ο χρόνος προσπέλασης της κυρίας μνήμης δεν εξαρτάται από τον επεξεργαστή που κάνει την προσπέλαση, ούτε από τη διεύθυνση της θέσης μνήμης που προσπελαίνεται. Αυτοί οι πολυεπεξεργαστές καλούνται πολυεπεξεργαστές με ομοιόμορφο χρόνο προσπέλασης μνήμης (Uniform Memory Access, UMA, multiprocessors). Όταν το πλήθος των επεξεργαστών είναι μικρό, τότε η επικοινωνία μεταξύ των επεξεργαστών και της κύριας μνήμης είναι δυνατόν να γίνεται μέσω μιας αρτηρίας (βλέπε σχήμα 7.26). Ωστόσο, επειδή όλοι οι επεξεργαστές επικοινωνούν με την κύρια μνήμη μέσω της αρτηρίας, η αρτηρία υπερφορτώνεται και οδηγεί σε καθυστερήσεις.

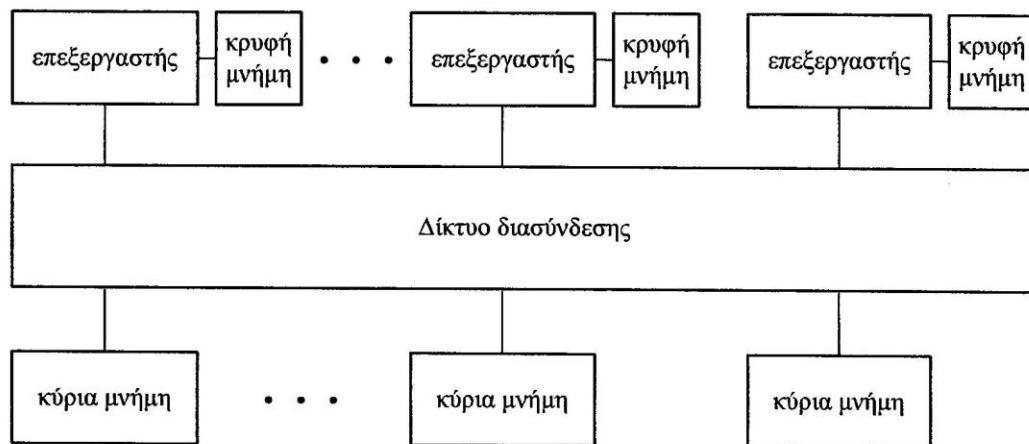


Σχήμα 7. 26 Πολυεπεξεργαστές με ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης που βασίζονται σε χρήση κοινής αρτηρίας.

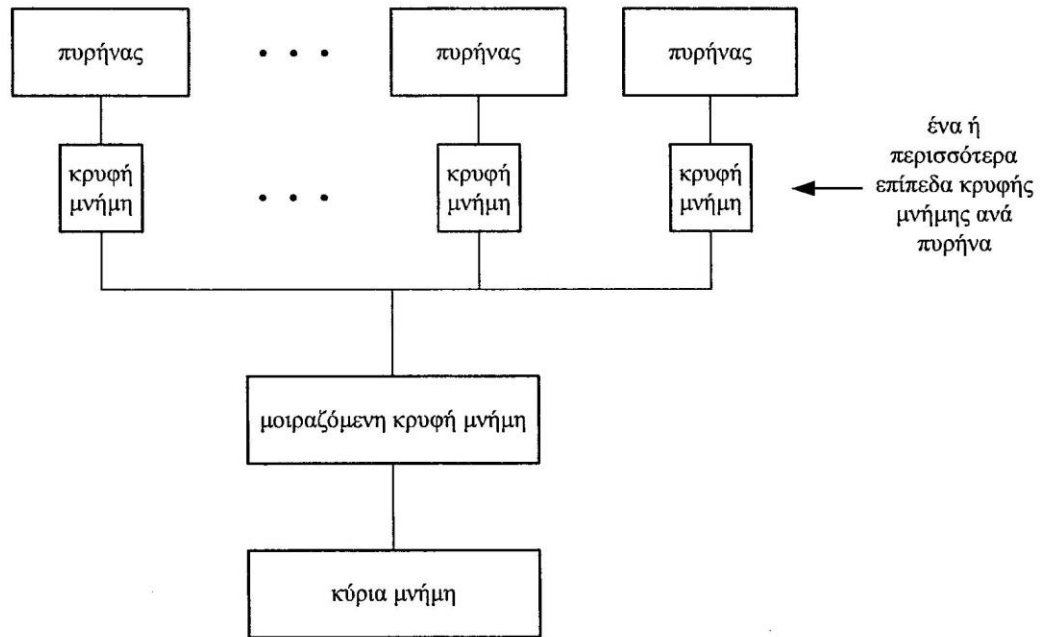
Όταν το πλήθος των επεξεργαστών είναι μεγάλο, η μνήμη αποτελείται από τμήματα (banks) τα οποία συνδέονται με τους επεξεργαστές μέσω ενός δικτύου διασύνδεσης όπως φαίνεται στο σχήμα 7.27. Το δίκτυο διασύνδεσης μπορεί να είναι ένας πίνακας διακοπών (crossbar switch) ή ένα δίκτυο διακοπών πολλών βαθμίδων (ως παράδειγμα

αναφέρουμε το δίκτυο Ωμέγα). Ένα πολύ ενδιαφέρον χαρακτηριστικό του πίνακα διακοπών είναι ότι κάθε επεξεργαστής μπορεί να επικοινωνήσει με οποιοδήποτε τμήμα της κύριας μνήμης αρκεί αυτό να είναι διαθέσιμο, άσχετα από τους άλλους επεξεργαστές που επικοινωνούν με άλλα τμήματα της κύριας μνήμης (ένα δίκτυο που έχει αυτό το χαρακτηριστικό καλείται nonblocking network. Το δίκτυο Ωμέγα δεν έχει αυτή την ιδιότητα, είναι ένα blocking network). Το μειονέκτημα του πίνακα διακοπών είναι ότι το πλήθος των διακοπών αυξάνεται ανάλογα με το γινόμενο του πλήθους των επεξεργαστών και των τμημάτων της κύριας μνήμης. Ένα μειονέκτημα όλων των δικτύων διασύνδεσης είναι ότι κάθε αναφορά στην κύρια μνήμη πρέπει να περάσει διαμέσου του δικτύου διασύνδεσης.

Τα συστήματα ενός πολυπύρηνου επεξεργαστή ανήκουν σ' αυτή την κατηγορία. Στο σχήμα 7.28 δίνεται η βασική δομή ενός πολυεπεξεργαστή μοιραζόμενης μνήμης με ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης που υλοποιείται με ένα πολυπύρηνου επεξεργαστή.



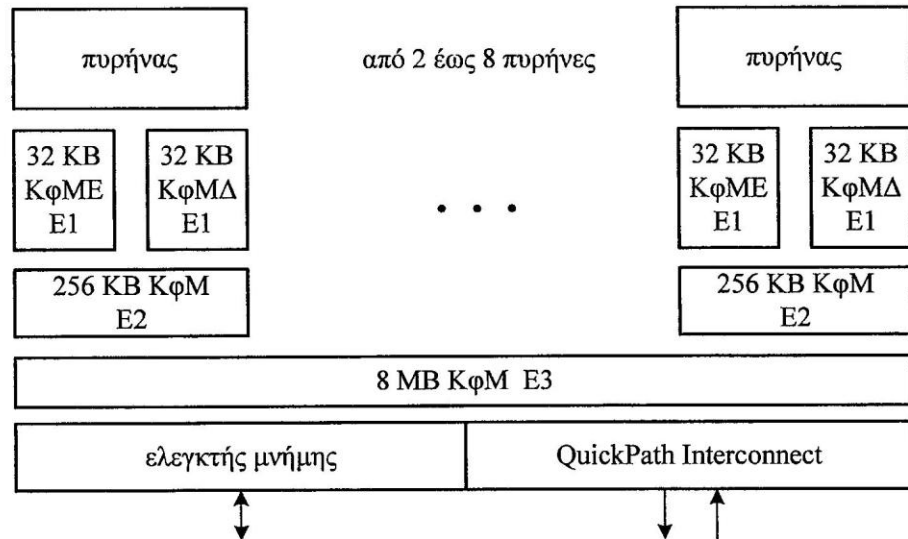
Σχήμα 7. 27 Πολυεπεξεργαστές με ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης που βασίζονται σε δίκτυο διασύνδεσης.



Σχήμα 7. 28 Πολυεπεξεργαστής με ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης που υλοποιείται με ένα πολυπύρηνο επεξεργαστή.

Στο σχήμα 7. 29 δίνεται η δομή ενός πολυπύρηνου επεξεργαστή Intel Core i7. Ο Intel Core i7 είναι ένας υψηλής απόδοσης γενικού σκοπού επεξεργαστής, ο οποίος υλοποιείται με από δύο έως οκτώ πυρήνες. Κάθε πυρήνας υλοποιεί μια υπερβαθμωτή κεντρική μονάδα επεξεργασίας με μέγιστο αριθμό αποστολής εντολών προς εκτέλεση 4 και αποστολή εντολών εκτός σειράς (four issue out-of-order) και υποστηρίζει την τεχνική της ταυτόχρονης πολυνημάτωσης για δύο νήματα (two-way simultaneous multithreading). Υποστηρίζει το σύνολο εντολών των x86 επεξεργαστών. Κάθε πυρήνας έχει ξεχωριστή κρυφή μνήμη δεδομένων και κρυφή μνήμη εντολών πρώτου επιπέδου με χωρητικότητα 32 KB η καθεμία και ενοποιημένη κρυφή μνήμη δεύτερου επιπέδου, με χωρητικότητα 256 KB. Όλοι οι πυρήνες του επεξεργαστή μοιράζονται μια κρυφή μνήμη τρίτου επιπέδου, με χωρητικότητα 8 MB. Ο επεξεργαστής επίσης περιέχει έναν ελεγκτή μνήμης και ένα σύστημα διασύνδεσης αποτελούμενο από αρτηρίες αποκλειστικής χρήσης (point-to-point links), το οποίο η Intel ονομάζει QuickPath Interconnect (QPI). Το σύστημα αυτό διασύνδεσης χρησιμοποιείται για να διασυνδεθούν μεταξύ τους περισσότεροι πολυπύρηνου επεξεργαστές Intel Core i7 και να σχηματίσουν ένα μεγαλύτερο σύστημα. Ο χρόνος προσπέλασης της κύριας μνήμης από οποιοδήποτε

πυρήνα του πολυπύρηνου επεξεργαστή Intel Core i7 είναι ο ίδιος, επομένως ο πολυπύρηνος επεξεργαστής Intel Core i7 είναι ένα παράδειγμα πολυεπεξεργαστή με ομοιόμορφο χρόνο προσπέλασης της κύριας μνήμης.

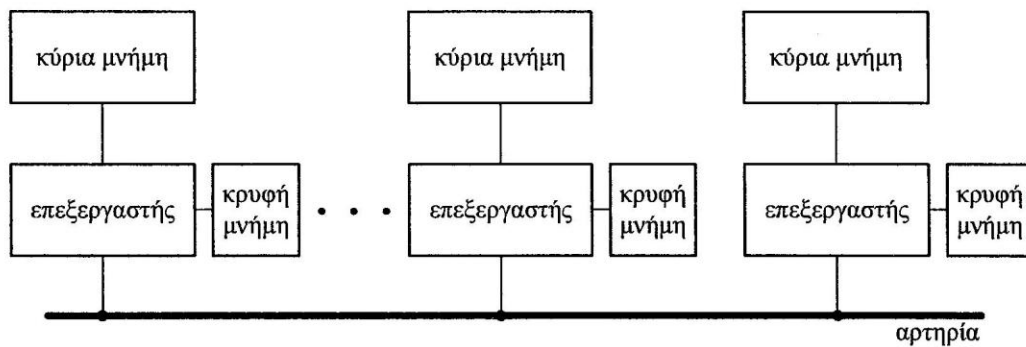


Στο σχήμα 7. 29 Η δομή ενός πολυπύρηνου επεξεργαστή Intel Core i7.

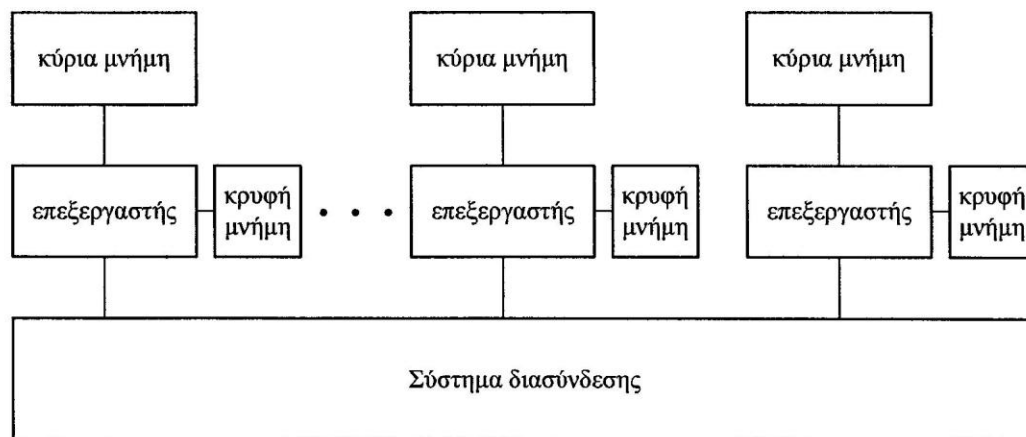
β. Πολυεπεξεργαστές στους οποίους ο χρόνος προσπέλασης της κύριας μνήμης εξαρτάται από τον επεξεργαστή που κάνει την προσπέλαση και τη διεύθυνση της θέσης μνήμης που προσπελάσσεται. Αυτοί οι πολυεπεξεργαστές καλούνται πολυεπεξεργαστές με μη ομοιόμορφο χρόνο προσπέλασης μνήμης (Non-Uniform Memory Access, NUMA, multiprocessors). Κάποιες προσπελάσεις είναι πολύ πιο γρήγορες από κάποιες άλλες. Αυτό οφείλεται στο γεγονός ότι σε κάθε επεξεργαστή έχει συνδεθεί ένα τμήμα της κύριας μνήμης (βλέπε σχήματα 7. 30 και 7. 31). Πάλι ο χώρος φυσικών διευθύνσεων είναι ενιαίος και κάθε επεξεργαστής μπορεί να προσπελάσει οποιαδήποτε θέση μνήμης χρησιμοποιώντας εντολές LOAD και STORE. Ωστόσο ο χρόνος προσπέλασης είναι πολύ μικρότερος όταν η προσπέλαση γίνεται στο τμήμα της κύριας μνήμης που είναι συνδεδεμένο στον επεξεργαστή που κάνει την προσπέλαση.

Στους πολυεπεξεργαστές με μη ομοιόμορφο χρόνο προσπέλασης της κύριας μνήμης μπορεί να χρησιμοποιηθεί κοινή αρτηρία για τη διασύνδεση περισσότερων μονάδων απ' ότι στην περίπτωση πολυεπεξεργαστών με ομοιόμορφο χρόνο προσπέλασης της κύριας

μνήμης διότι όλες οι προσπελάσεις της κύριας μνήμης δεν γίνονται μέσω της αρτηρίας και επομένως δεν υπερφορτώνεται.



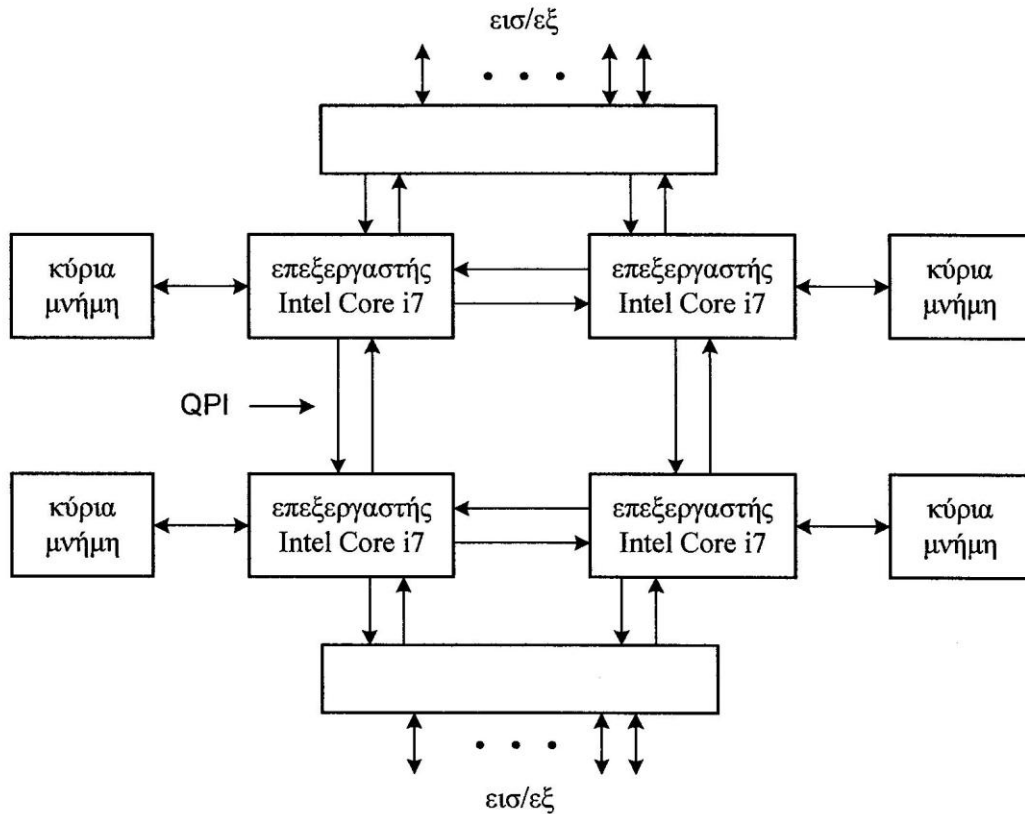
Σχήμα 7. 30 Πολυεπεξεργαστές με μη ομοιόμορφο χρόνο προσπέλασης της κύριας μνήμης που βασίζονται σε χρήση κοινής αρτηρίας.



Σχήμα 7. 31 Πολυεπεξεργαστές με μη ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης που βασίζονται σε δίκτυο διασύνδεσης.

Περιγράφοντας τη δομή του πολυπύρηνου επεξεργαστή Intel Core i7 είπαμε ότι περιέχει ένα σύστημα διασύνδεσης, το οποίο η Intel ονομάζει QuickPath Interconnect (QPI). Το σύστημα αυτό διασύνδεσης χρησιμοποιείται για να διασυνδεθούν μεταξύ τους περισσότεροι πολυπύρηνου επεξεργαστές Intel Core i7 και να σχηματίσουν ένα μεγαλύτερο σύστημα, όπως φαίνεται στο σχήμα 7.32. Κάθε επεξεργαστής έχει τη δική

του κύρια μνήμη την οποία κάθε πυρήνας του την προσπελαίνει μέσω του ολοκληρωμένου ελεγκτή μνήμης. Όταν ένας πυρήνας ενός επεξεργαστή χρειάζεται να προσπελάσει την κύρια μνήμη ενός άλλου επεξεργαστή, η προσπέλαση γίνεται μέσω συστήματος διασύνδεσης QPI. Ο χώρος φυσικών διευθύνσεων είναι κοινός για όλους τους επεξεργαστές, ο χρόνος προσπέλασης εξαρτάται από τον πυρήνα και τη φυσική διεύθυνση της θέσης μνήμης. Επομένως ένας πολυεπεξεργαστής αποτελούμενος από περισσότερους από ένα επεξεργαστές Intel Core i7 δεν θα έχει ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης.



Σχήμα 7.32 Πολυεπεξεργαστής με μη ομοιόμορφο χρόνο προσπέλασης κύριας μνήμης αποτελούμενος από 4 επεξεργαστές Intel Core i7.

Στη συνέχεια θα δούμε διάφορα προβλήματα που μπορούν να προκύψουν κατά την εκτέλεση ενός πολυνηματικού προγράμματος σε ένα πολυεπεξεργαστή μοιραζόμενη μνήμη.

7.4.1.1 Συγχρονισμός

Για να κατανοήσουμε μερικά προβλήματα συγχρονισμού που προκύπτουν κατά την εκτέλεση πολυνηματικών προγραμμάτων θεωρούμε τον ακόλουθο βρόχο:

```
for (i=1; i≤1000000; i++)
    w(i)=y(i) × z(i);
```

Οι επαναλήψεις του ανωτέρου βρόχου είναι ανεξάρτητες, επομένως μπορούν να εκτελεστούν παράλληλα. Η ανάθεση του υπολογισμού των βρόχων σε περισσότερους από έναν επεξεργαστές μπορεί να γίνει δυναμικά χρησιμοποιώντας μια κοινή (global) μεταβλητή i και αυξάνοντας την τιμή της έτσι ώστε ο επόμενος επεξεργαστής που ζητάει δουλειά θα αναλαμβάνει την επόμενη επανάληψη του βρόχου. Εάν η διαχείριση της μεταβλητής i γίνεται με τις κανονικές εντολές LOAD και STORE, είναι δυνατόν να προκύψει η ακόλουθη κατάσταση:

Κύκλος ρολογιού	Επεξεργαστής 1	Επεξεργαστής 2
1	LOAD r1, i	
2		LOAD r1, i
3		ADI r1, #1
4	ADI r1, #1	

Και οι δύο επεξεργαστές διαβάζουν την ίδια τιμή του i και εκτελούν την ίδια επανάληψη του βρόχου. Ακόμη χειρότερα, η επόμενη επανάληψη του βρόχου δεν εκτελείται καθόλου, διότι κάθε ένας από τους δύο επεξεργαστές αυξάνει την τιμή της μεταβλητής i κατά ένα. Το παραπάνω πρόβλημα δεν θα μπορούσε να προκύψει εάν το διάβασμα της μεταβλητής i και η αύξηση της τιμής της κατά ένα γίνονταν ως μια αδιαίρετη λειτουργία, ατομική λειτουργία (atomic operation). Ένας εναλλακτικός τρόπος είναι να έχουμε ένα ζεύγος από εντολές (load linked/store conditional) εκ των οποίων η δεύτερη εντολή να επιστρέφει μια τιμή από την οποία μπορεί να βγει το συμπέρασμα εάν το ζεύγος των εντολών εκτελέστηκε ως μια ενιαία αδιαίρετη λειτουργία ή όχι. Τα σύνολα εντολών σύγχρονων επεξεργαστών όπως οι MIPS, PowerPC και IA-64 προσφέρουν αυτή τη λύση.

Ο κατάλληλος συγχρονισμός είναι απαραίτητος όχι μόνον στην περίπτωση που ένα πολυνηματικό πρόγραμμα εκτελείται σε ένα πολυεπεξεργαστή, αλλά και στην περίπτωση που εκτελείται σε ένα πολυνηματικό επεξεργαστή ή σε ένα πολυεπεξεργαστή στον οποίο κάθε επεξεργαστής υποστηρίζει την τεχνική του πολυνηματισμού.

7.4.1.2 Συνέπεια μνήμης

Στην ενότητα των υπερβαθμωτών επεξεργαστών μιλήσαμε για το μοντέλο ισχυρής συνέπειας και το μοντέλο ασθενούς συνέπειας της μνήμης. Υπενθυμίζουμε ότι όταν υιοθετείται το μοντέλο ισχυρής συνέπειας η προσπέλαση της μνήμης γίνεται με την ίδια σειρά που θα γίνονταν και κατά τη σειριακή εκτέλεση του προγράμματος. Στην περίπτωση της ασθενούς συνέπειας η σειρά με την οποία εκτελούνται εντολές που προσπελαίνουν τη μνήμη μπορεί να είναι διαφορετική από αυτή που θα είχαμε στη σειριακή εκτέλεση, υπό την προϋπόθεση ότι δεν παραβιάζονται εξαρτήσεις. Είναι προφανές ότι στην περίπτωση που έχουμε πολλούς επεξεργαστές που εκτελούν νήματα της ίδιας διαδικασίας (process) η επιβολή του μοντέλου ισχυρής συνέπειας θα επιφέρει μεγάλες καθυστερήσεις (μείωση της απόδοσης). Επιπλέον, το μοντέλο ισχυρής συνέπειας δεν είναι αναγκαίο για την ορθότητα της εκτέλεσης του προγράμματος. Θεωρήστε ότι η εκτέλεση του επόμενου προγράμματος γίνεται από περισσότερους από έναν επεξεργαστές. Οι προσπελάσεις των y , και z του πρώτου βρόχου είναι ανεξάρτητες, επομένως δεν χρειάζεται να γίνουν με συγκεκριμένη σειρά.

```
for (i=1; i≤1000000; i++)
    w(i)=y(i) × z(i);
for (j=1000000; j>0; j--)
    w(j)=w(j) × f(j);
```

Ωστόσο δεν θα πρέπει να ξεκινήσουν οι προσπελάσεις του δεύτερου βρόχου πριν ολοκληρωθεί η αποθήκευση της μεταβλητής w σε όλες τις επαναλήψεις του πρώτου βρόχου. Επομένως, συγχρονισμός απαιτείται σε συγκεκριμένα σημεία. Για το λόγο αυτό έχουν προταθεί διάφορα μοντέλα ασθενούς συνέπειας (relaxed consistency models) για να επιβάλλεται η χρονική διάταξη των προσπελάσεων της μνήμης μόνον όταν απαιτείται. Η υιοθέτηση αυτών των μοντέλων απλοποιεί τα απαραίτητα κυκλώματα απαιτώντας από τον προγραμματιστή να αναγνωρίσει και να δηλώσει στο πρόγραμμα τις αναφορές στη μνήμη (αναγνώσεις και εγγραφές) οι οποίες πρέπει να γίνουν με συγκεκριμένη σειρά, ενώ επιτρέπει στο υλικό να προχωράει χωρίς κάποια συγκεκριμένη σειρά στην εκτέλεση των υπόλοιπων προσπελάσεων της μνήμης.

7.4.1.3 Συνάφεια του περιεχομένου των κρυφών μνημών (cache coherence)

Στη συνέχεια θα δούμε ένα πρόβλημα που μπορεί να προκύψει σε πολυεπεξεργαστές που κάθε επεξεργαστής έχει ένα ή περισσότερα επίπεδα κρυφής μνήμης. Έστω ότι

έχουμε ένα επίπεδο κρυφής μνήμης και ότι η τιμή της μεταβλητής A δεν υπάρχει στην κρυφή μνήμη κανενός επεξεργαστή. Θεωρούμε την ακόλουθη σειρά γεγονότων.

1. Ο επεξεργαστής E1 διαβάζει τη μεταβλητή A. Τότε η τιμή της μεταβλητής A αποθηκεύεται στην κρυφή μνήμη του E1.
2. Ο επεξεργαστής E2 διαβάζει τη μεταβλητή A. Τότε η τιμή της μεταβλητής A αποθηκεύεται στην κρυφή μνήμη του E2.
3. Ο επεξεργαστής E1 γράφει νέα τιμή στη μεταβλητή A.
4. Ο επεξεργαστής E2 διαβάζει τη μεταβλητή A (την τιμή την οποία βρίσκει στην κρυφή του μνήμη).

Είναι προφανές ότι στο τέταρτο βήμα της ανωτέρω διαδικασίας ο επεξεργαστής E2 διαβάζει την αρχική τιμή της μεταβλητής A, και όχι την τιμή που έγραψε ο επεξεργαστής E1. Πρέπει να σημειώσουμε ότι ο επεξεργαστής E2 διαβάζει λάθος τιμή είτε ο επεξεργαστής E1 ακολουθεί την τακτική της τελικής ενημέρωσης (write back) είτε την τακτική της άμεσης ενημέρωσης (write through) της κυρίας μνήμης. Στη δεύτερη περίπτωση η τιμή της μεταβλητής A στην κύρια μνήμη είναι η σωστή, ωστόσο ο επεξεργαστής E2 διαβάζει τη λάθος τιμή από την κρυφή του μνήμη.

Από τα ανωτέρω είναι φανερό ότι στους πολυεπεξεργαστές στους οποίους κάθε επεξεργαστής έχει ένα ή περισσότερα επίπεδα κρυφής μνήμης θα πρέπει να εξασφαλίσουμε ότι όταν κάποιος επεξεργαστής διαβάζει κάποια μεταβλητή θα διαβάζει την τελευταία τιμή που δόθηκε σ' αυτή τη μεταβλητή (cache coherence). Αυτό μπορεί να γίνει εάν όταν γίνεται εγγραφή σε μια μεταβλητή από ένα επεξεργαστή, ενημερώνονται με τη νέα τιμή όλες οι κρυφές μνήμες που περιέχουν αντίγραφο της μεταβλητής (πρωτόκολλο ενημέρωσης, update protocol). Λόγω του υπερβολικού ρυθμού μεταφοράς δεδομένων που απαιτεί ένα τέτοιο πρωτόκολλο, δεν χρησιμοποιείται στους σύγχρονους πολυεπεξεργαστές για την εξασφάλιση συνέπειας του περιεχομένου των κρυφών μνημών. Ένας άλλος τρόπος εξασφάλισης της συνέπειας του περιεχομένου των κρυφών μνημών είναι όταν γίνεται εγγραφή μιας μεταβλητής από ένα επεξεργαστή να ακυρώνονται οι τιμές των αντιγράφων της μεταβλητής που βρίσκονται στις κρυφές μνήμες άλλων επεξεργαστών (πρωτόκολλο ακύρωσης τιμών, invalidate protocol). Τα πρωτόκολλα ενημέρωσης και ακύρωσης τιμών διακρίνονται σ' αυτά που βασίζονται στη χρήση πίνακα ή πινάκων (directory based) και σ' αυτά που βασίζονται στην παρακολούθηση της αρτηρίας διευθύνσεων (snooping). Τα πρωτόκολλα εξασφάλισης συνέπειας του περιεχομένου των κρυφών μνημών υλοποιούνται με κυκλώματα.