



Ελληνική Δημοκρατία
Τεχνολογικό Εκπαιδευτικό
Ίδρυμα Ηπείρου

Αρχιτεκτονική υπολογιστών

Ενότητα 8 : Υποστήριξη από το Λειτουργικό Σύστημα

Φώτης Βαρζιώτης



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Τμήμα Μηχανικών Πληροφορικής Τ.Ε

Αρχιτεκτονική υπολογιστών

Ενότητα 8 : Υποστήριξη από το Λειτουργικό Σύστημα

Φώτης Βαρζιώτης

Καθηγητής Εφαρμογών

Άρτα, 2015





Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.





Χρηματοδότηση

- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «**Εκπαίδευση και Δια Βίου Μάθηση**» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Ηπείρου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



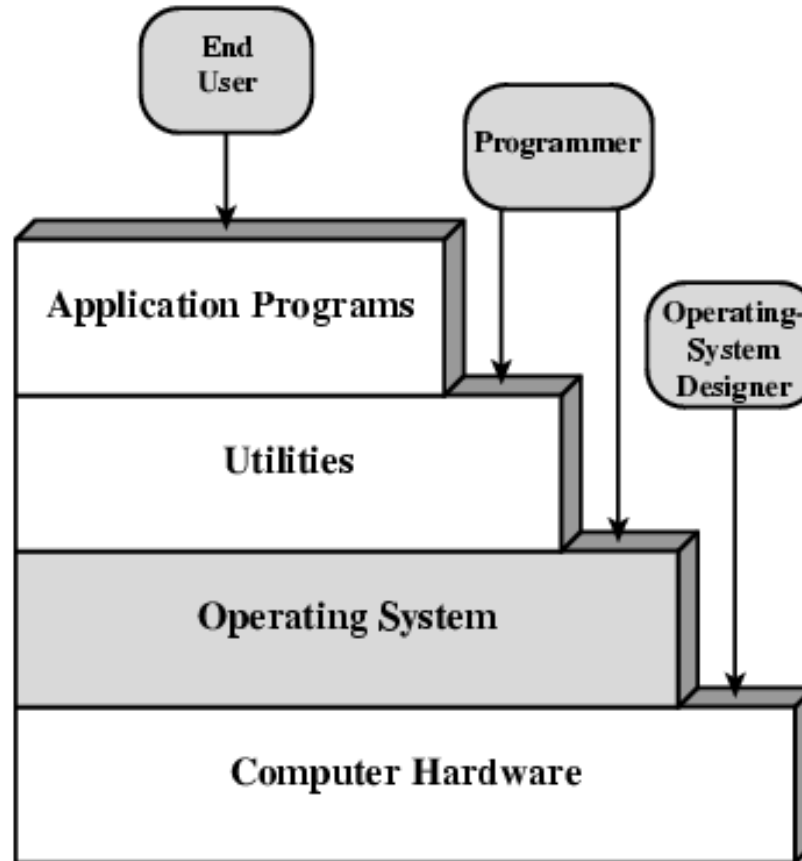


Στόχος και Λειτουργίες

- Ευκολία χρήσης
 - Ένα λειτουργικό σύστημα κάνει πιο ευκολόχρηστο έναν υπολογιστή
- Αποδοτικότητα
 - Βελτιστοποίηση χρήσης πόρων συστήματος



Επίπεδα και Όψεις Υπολογιστικού Συστήματος



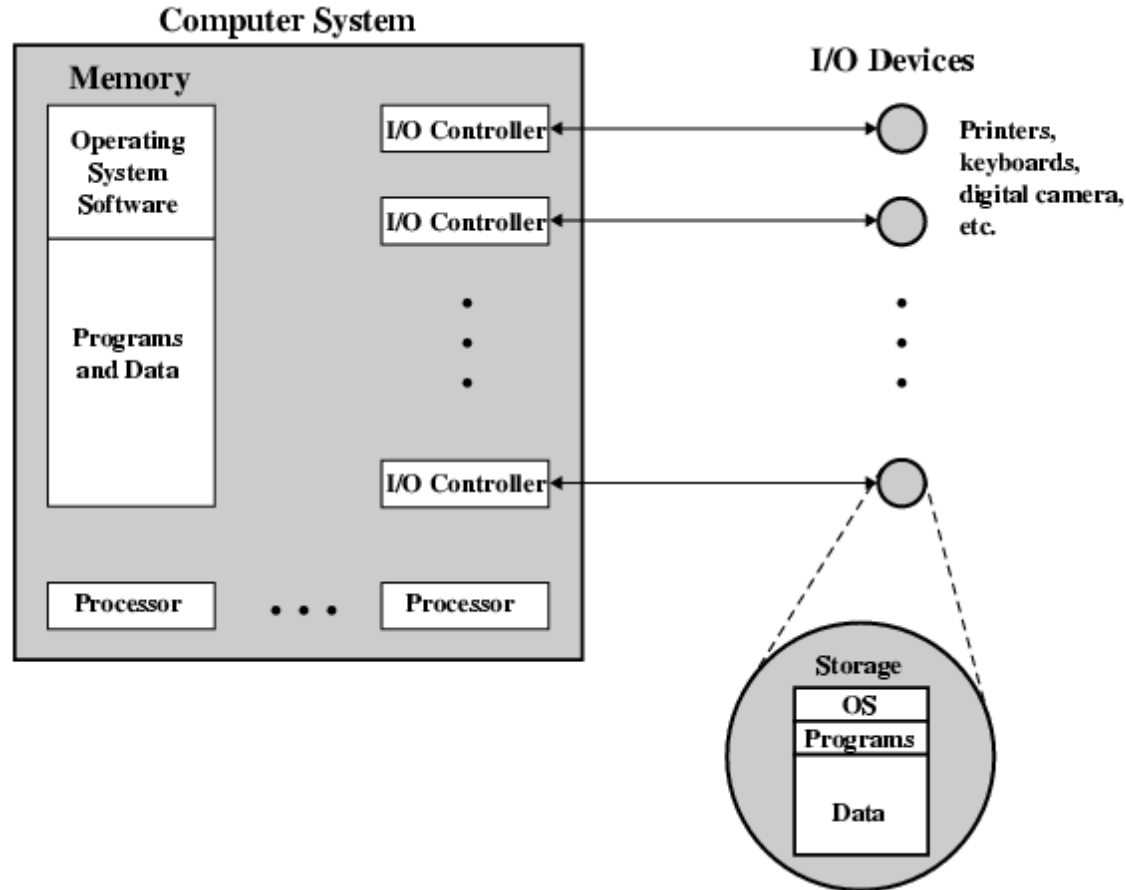


Υπηρεσίες Λειτουργικού Συστήματος

- Δημιουργία προγραμμάτων
- Εκτέλεση προγραμμάτων
- Προσπέλαση σε συσκευές I/O
- Ελεγχόμενη πρόσβαση σε αρχεία
- Πρόσβαση στο σύστημα
- Ανίχνευση και απόκριση σε σφάλμα
- Τεκμηρίωση



Το O/S ως Διαχειριστής Πόρων





Τύποι Λειτουργικών Συστημάτων

- Αλληλεπιδραστικό
- Ομαδοποιητικό
- Μονοπρογραμματιστικό (Uni-programming)
- Πολυπρογραμματιστικό (Multi-tasking)



Τα Πρώτα Συστήματα

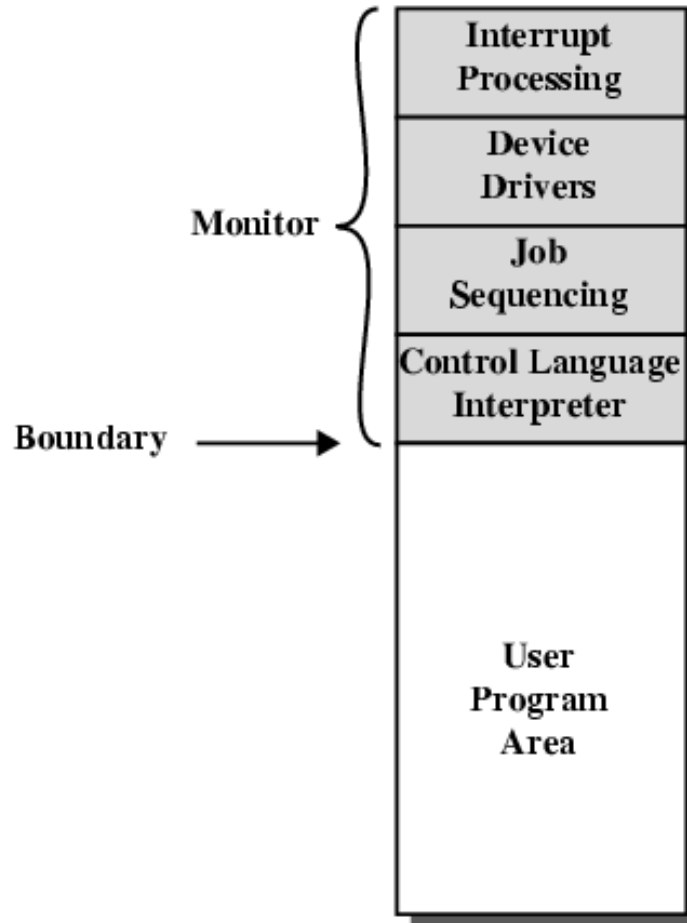
- Αρχές 1940 έως μέσα 1950
- Δεν υπήρχε λειτουργικό σύστημα
- Τα προγράμματα αλληλεπιδρούσαν απευθείας με το υλικό
- Δύο μεγάλα προβλήματα:
 - Προγραμματισμός
 - Χρόνος προετοιμασίας



Απλά συστήματα με ομαδοποίηση προγραμμάτων (batch systems)

- Παραμένον λειτουργικό σύστημα επιτήρησης
- Ο χρήστης υποβάλλει προγράμματα (jobs) στον χειριστή
- Ο χειριστής ομαδοποιεί τα προγράμματα
- Το σύστημα επιτήρησης ελέγχει την ακολουθία των εργασιών για επεξεργασία
- Όταν μια εργασία ολοκληρώνεται, ο έλεγχος επιστρέφει στον επιτηρητή, ο οποίος διαβάζει την επόμενη εργασία
- Ο επιτηρητής ουσιαστικά διαχειρίζεται τους πόρους του συστήματος

Εικόνα της μνήμης για χρήση επιτηρητή





Γλώσσα Ελέγχου Εργασιών

- Εντολές στον επιτηρητή
- Σημειώνονται με \$
- π.χ. - \$JOB
 - \$FTN
 - ... Some Fortran instructions
 - \$LOAD
 - \$RUN
 - ... Some data
 - \$END



Επιθυμητά Χαρακτηριστικά Λογικών Κυκλωμάτων

- Προστασία της μνήμης
 - Ουσιαστικά προστασία του επιτηρητή
- Χρονιστής
 - Όστε να μην μονοπωλείται το σύστημα από μια εργασία
- Προνομιακές εντολές
 - Εκτελέσιμες μόνο από τον επιτηρητή
 - π.χ. I/O
- Διακοπές
 - Επιτρέπουν παραχώρηση και ανάκτηση ελέγχου

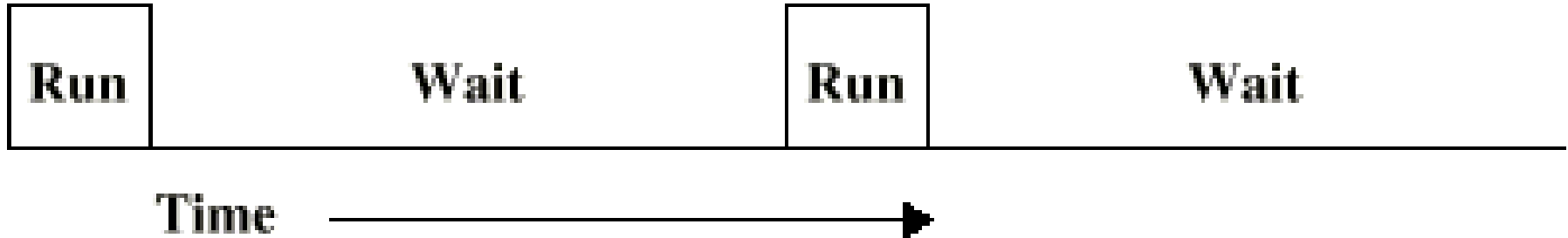


Πολυπρογραμματιζόμενα λειτουργικά συστήματα με ομαδοποίηση προγραμμάτων

- Οι συσκευές I/O είναι πολύ αργές
- Όταν ένα πρόγραμμα εκτελεί λειτουργία I/O, ένα άλλο μπορεί να χρησιμοποιήσει την CPU

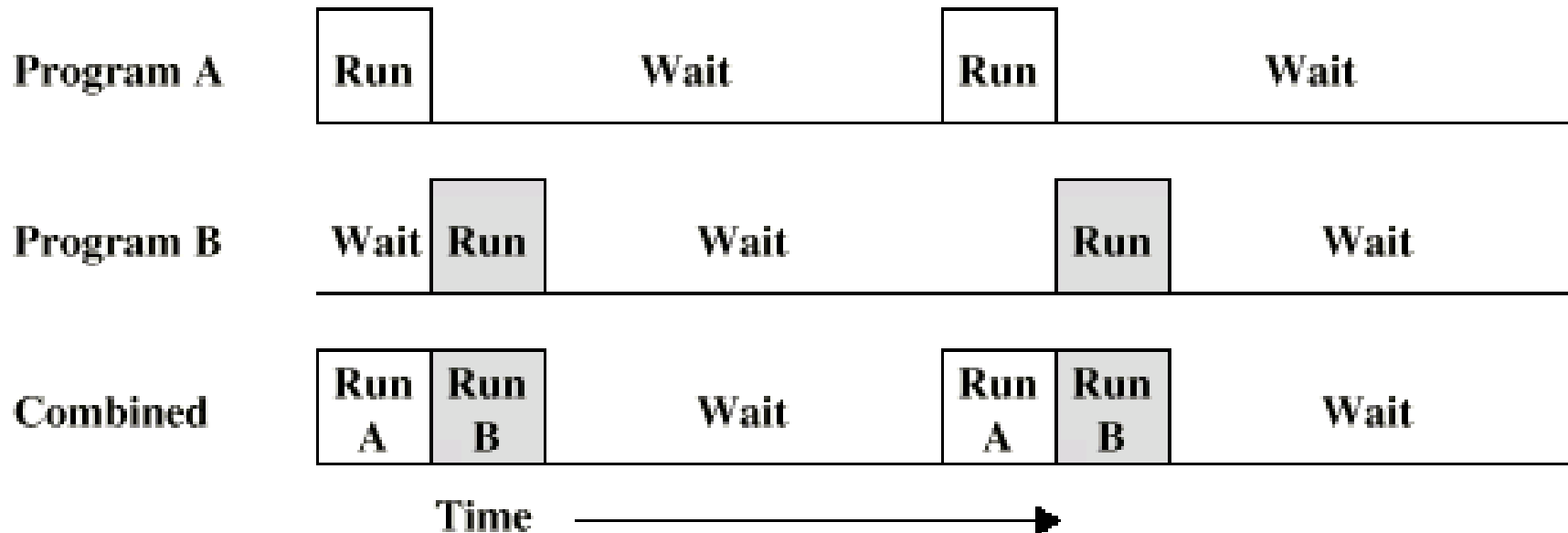


Μονοπρογραμματισμός



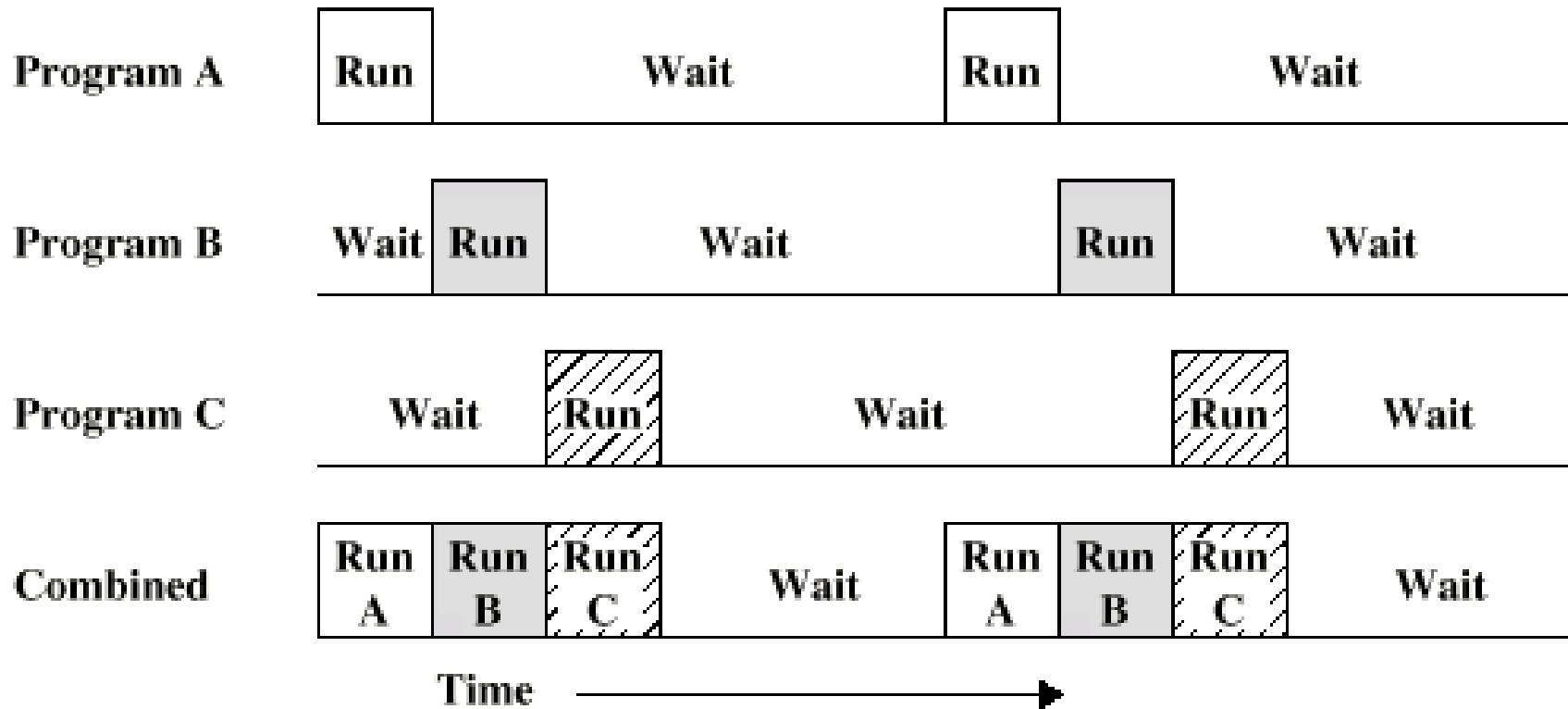


Πολυπρογραμματισμός με δύο προγράμματα



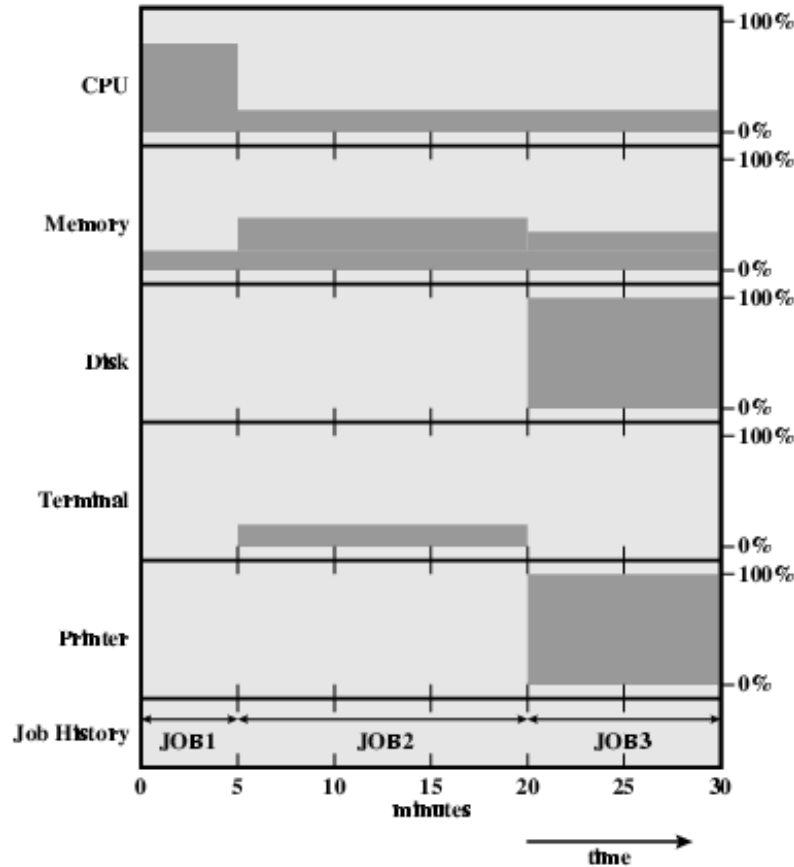


Πολυπρογραμματισμός με τρία προγράμματα

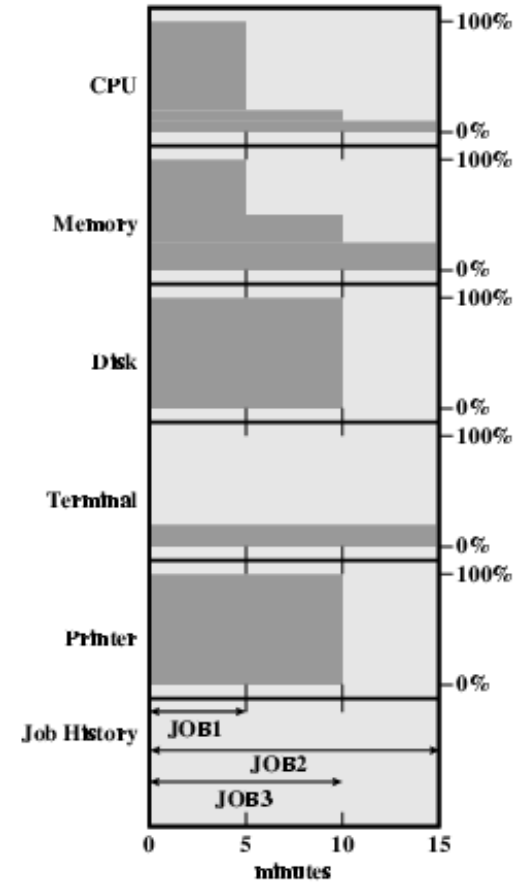




Χρησιτικότητα



(a) Unprogramming



(b) Multiprogramming



Συστήματα με κατανομή χρόνου

- Επιτρέπει στους χρήστες να αλληλεπιδρούν απευθείας με τον υπολογιστή.
- Ο πολυπρογραμματισμός επιτρέπει σε έναν αριθμό χρηστών να αλληλεπιδρούν με το σύστημα.



Προγραμματισμός πόρων – Χρονοδιαγράμματα εργασιών

- Είναι το κλειδί του πολυπρογραμματισμού
- Μακροπρόθεσμο Χρονοδιάγραμμα
- Μεσοπρόθεσμο Χρονοδιάγραμμα
- Βραχυπρόθεσμο Χρονοδιάγραμμα
- Χρονοδιάγραμμα I/O



Μακροπρόθεσμο Χρονοδιάγραμμα

- Αποφασίζει ποια προγράμματα θα υποβληθούν στο σύστημα για επεξεργασία
- π.χ. Ελέγχει τον βαθμό του πολυπρογραμματισμού
- Με την υποβολή της, η εργασία γίνεται μια διεργασία του βραχυπρόθεσμου χρονοδιαγράμματος
- (ή μεταβαίνει στο μεσοπρόθεσμο χρονοδιάγραμμα - swapped out job- σε μια κατάσταση εκτός κύριας μνήμης)

Μεσοπρόθεσμο Χρονοδιάγραμμα

- Μέρος της λειτουργίας αντιμετάθεσης (παρακάτω...)
- Συνήθως βασίζεται στην ανάγκη για καλύτερη διαχείριση του πολυπρογραμματισμού
- Αν δεν χρησιμοποιείται εικονική μνήμη, η διαχείριση της μνήμης είναι ένα σημαντικό θέμα

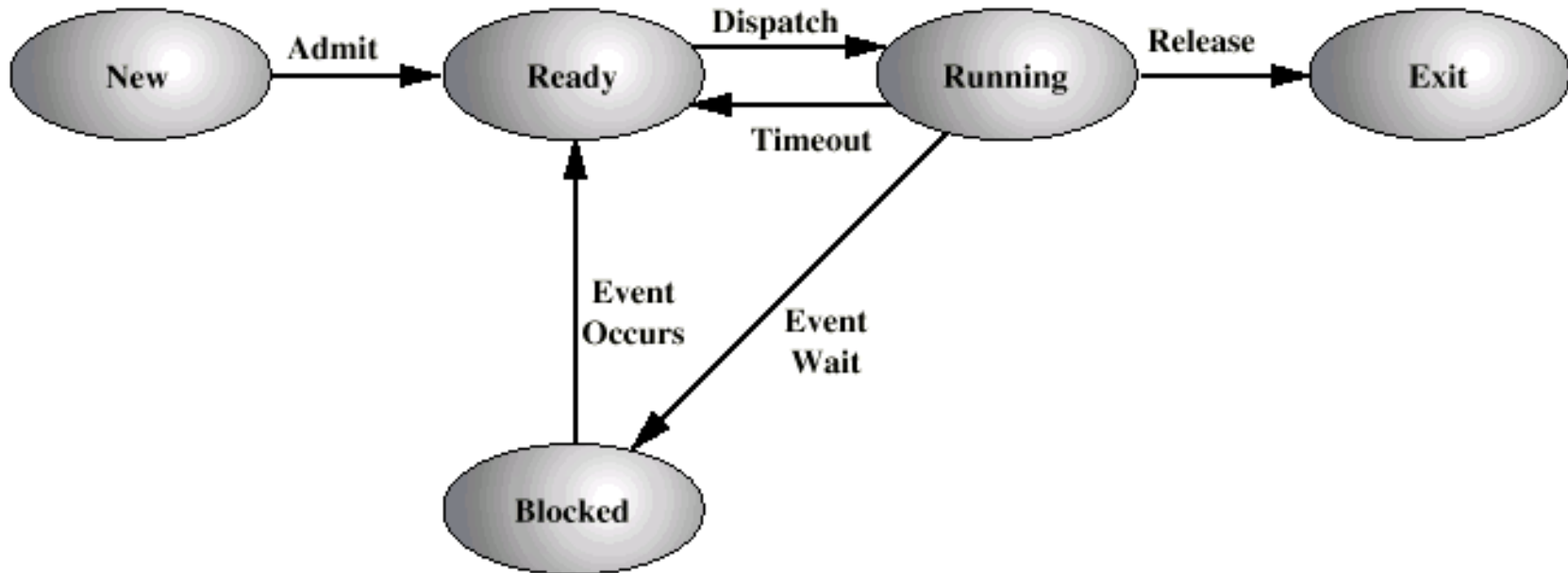


Βραχυπρόθεσμο Χρονοδιάγραμμα

- Γνωστό ως Dispatcher (Αποστολέας)
- Αποφασίζει ποια διεργασία θα εκτελεστεί μετά
- π.χ. ποια εργασία θα χρησιμοποιήσει τον επεξεργαστή στην επόμενη χρονική θυρίδα



Το μοντέλο διεργασίας πέντε καταστάσεων



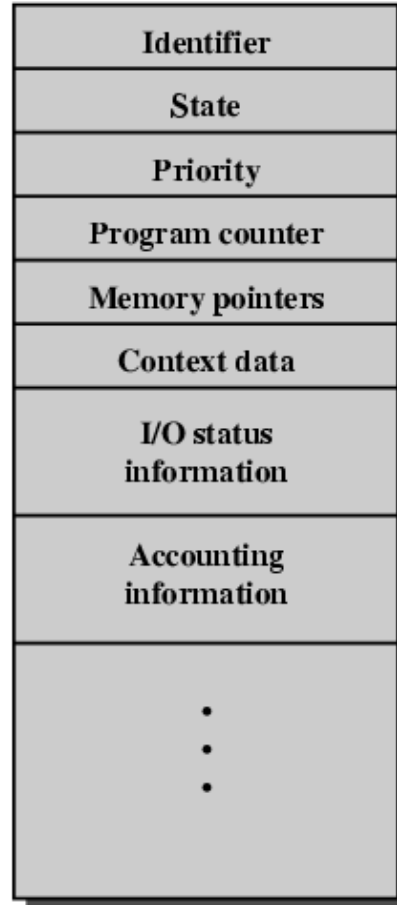


Τμήμα (Μνήμης) Ελέγχου Διεργασίας

- Πληροφορίες ταυτοποίησης
- Πληροφορίες κατάστασης
- Πληροφορίες προτεραιότητας
- Πληροφορίες απεριθμητή προγράμματος
- Δείκτες μνήμης
- Συναφή δεδομένα
- Πληροφορίες κατάστασης I/O
- Λογιστικές πληροφορίες

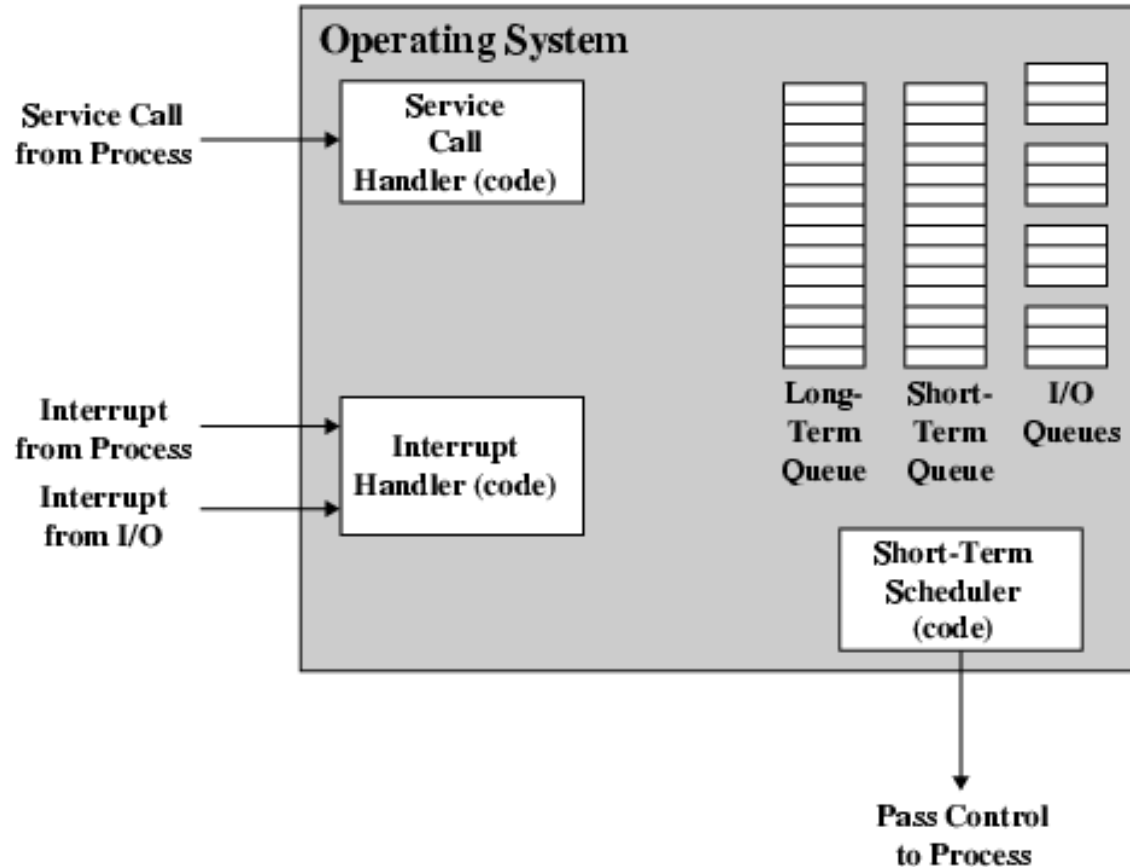


Διάγραμμα τμήματος ελέγχου διεργασιών



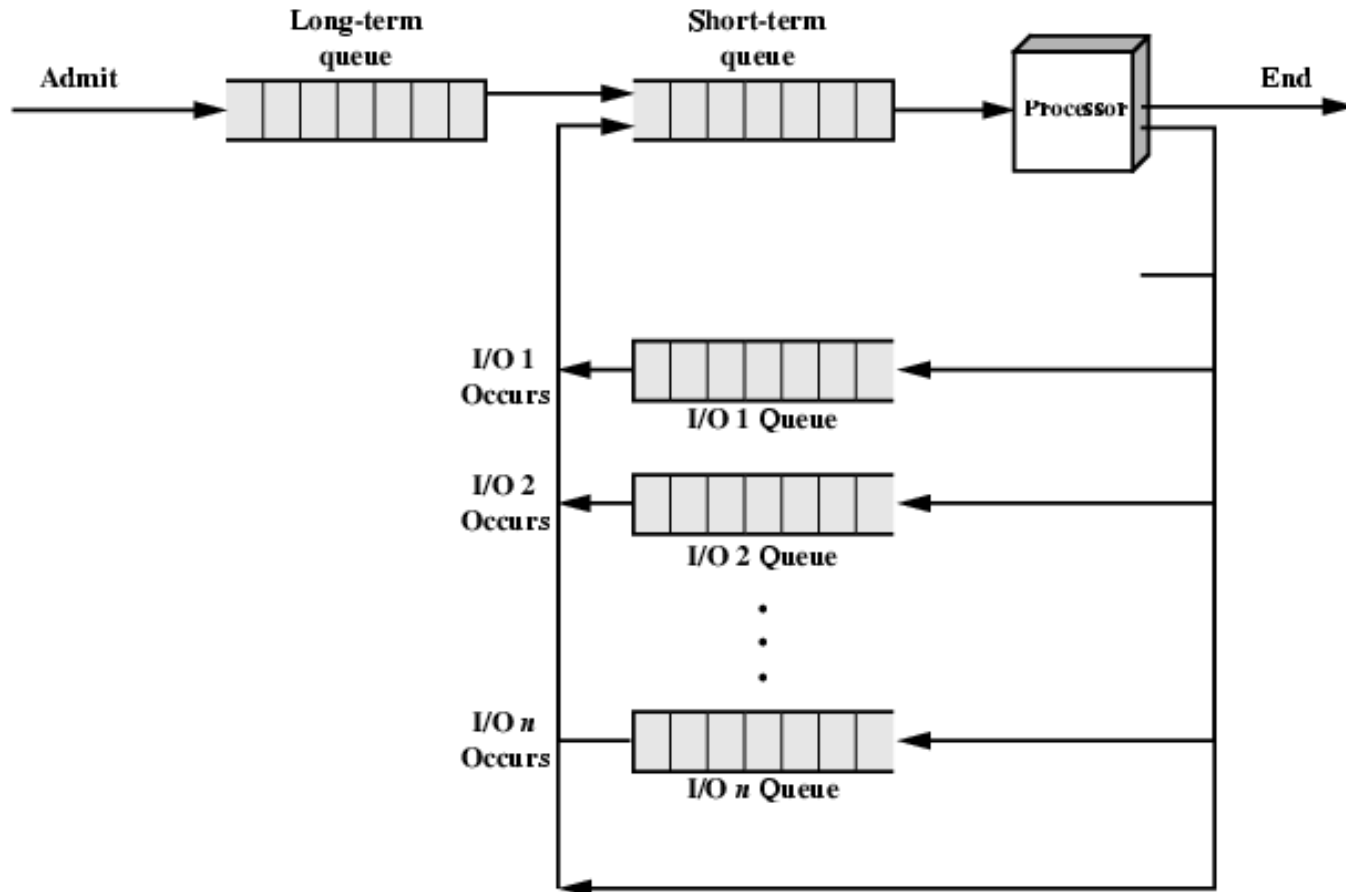


Βασικά στοιχεία O/S για πολυπρογραμματισμό





Προγραμματισμός διεργασιών



Διαχείριση Μνήμης

- Μονοπρογραμματισμός
 - Η μνήμη χωρίζεται σε δύο μέρη
 - Ένα για το λειτουργικό σύστημα (monitor)
 - Ένα για το εκτελούμενο πρόγραμμα χρήστη
- Πολυπρογραμματισμός
 - Το κομμάτι του χρήστη υποδιαιρείται και διαμοιράζεται στις ενεργές διεργασίες



Αντιμετάθεση

- **Πρόβλημα:** Η λειτουργία I/O είναι τόσο αργή σε σύγκριση με τη CPU ώστε ακόμη και σε ένα πολυπρογραμματιζόμενο σύστημα , η CPU μπορεί να παραμένει αδρανής το περισσότερο χρόνο
- **Λύση:**
 - Αύξηση της κύριας μνήμης
 - Ακριβό
 - Οδηγεί σε μεγαλύτερα προγράμματα
 - Αντιμετάθεση (Swapping)

Τι είναι αντιμετάθεση (Swapping)

- Το μακροπρόθεσμο χρονοδιάγραμμα διεργασιών αποθηκεύεται στον δίσκο
- Οι διεργασίες αντιμετατίθενται (“swapped”) μόλις κάποιος χώρος είναι διαθέσιμος
- Όταν ολοκληρωθεί μια διεργασία απομακρύνεται από την κύρια μνήμη
- Αν καμιά διεργασία δεν είναι σε κατάσταση ετοιμότητας, (όλες μπλοκαρισμένες σε λειτουργία I/O)
 - Αντιμετάθεση μπλοκαρισμένης διεργασίας σε ενδιάμεση ουρά
 - Εισαγωγή διεργασίας σε κατάσταση ετοιμότητας ή νέας διεργασίας
 - Ωστόσο το «swapping» είναι μια λειτουργία I/O ...

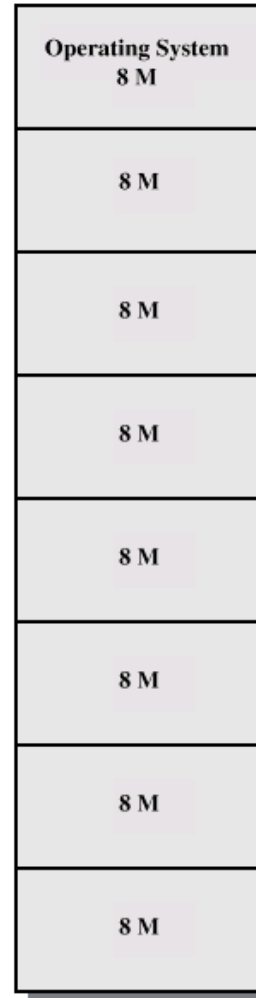


Κατάτμηση_{1/2}

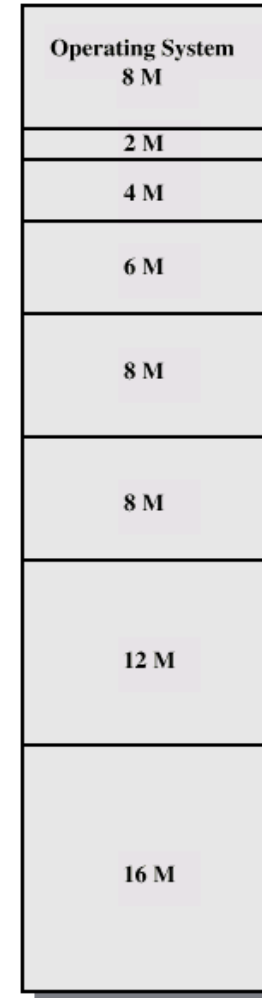
- Κατάτμηση της μνήμης σε τομείς και απόδοση διεργασιών σε κάθε τομέα (περιλαμβάνοντας και το λειτουργικό σύστημα)
- Τομείς σταθερού μεγέθους
 - Μπορεί να μην έχουν όλοι το ίδιο μήκος
 - Η διεργασία τοποθετείται στον τομέα που ταιριάζει καλύτερα (best fit)
 - Σπατάλη μέρους της μνήμης
 - Οδηγεί σε τομείς μεταβλητού μεγέθους



Κατάτμηση_{2/2}



(a) Equal-size partitions



(b) Unequal-size partitions

Τομείς μεταβλητού μεγέθους_{1/2}

- Απόδοση ακριβούς μεγέθους μνήμης σε κάθε διεργασία
- Οδηγεί σε ένα μικρό κενό στο τέλος της μνήμης
 - Μόνο ένα μικρό κενό – μικρότερη σπατάλη μνήμης
- Όταν όλες οι διεργασίες είναι μπλοκαρισμένες, γίνεται αντιμετάθεση
- Οι νέες διεργασίες μπορεί να είναι μικρότερες από τις προηγούμενες
- Άρα δημιουργείται νέο κενό

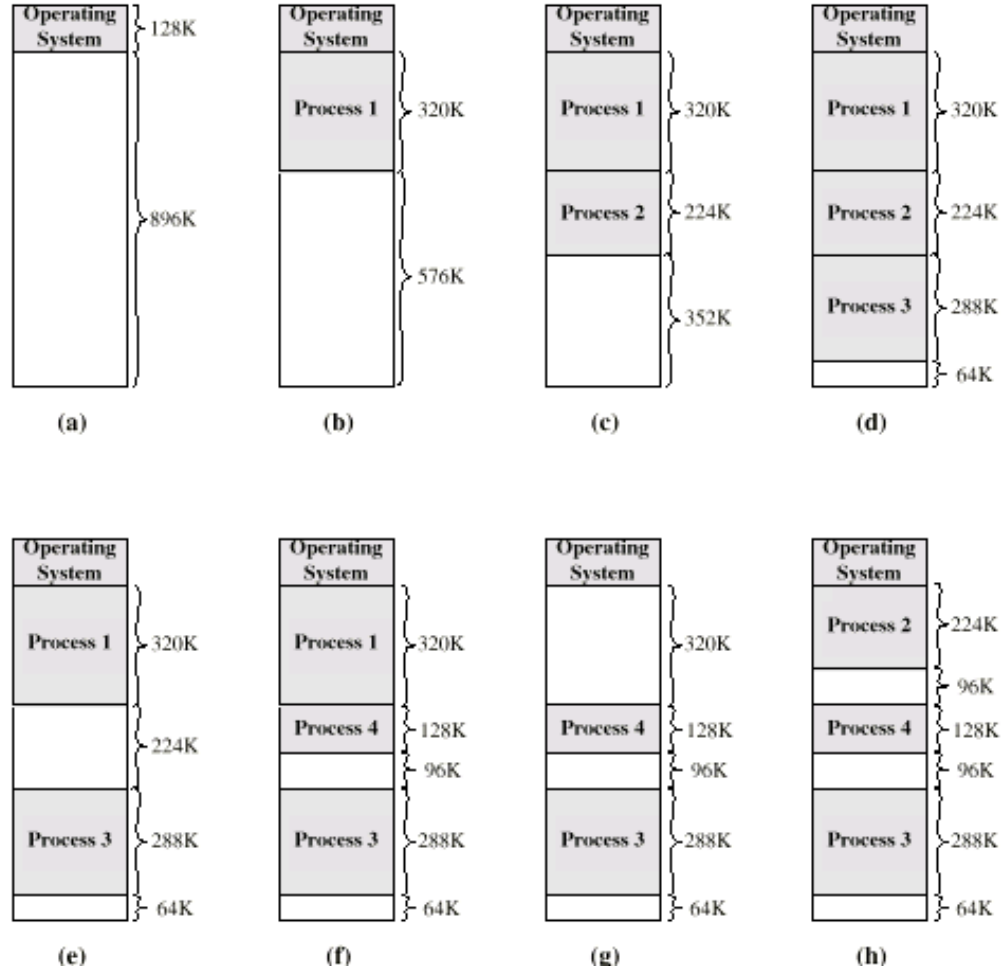


Τομείς μεταβλητού μεγέθους_{2/2}

- Τελικά δημιουργούνται πολλά κενά (fragmentation)
- **Λύση:**
 - Coalesce - Σύνδεση παραπλήσιων κενών για δημιουργία μεγαλύτερων κενών
 - Συμπύκνωση- Ανά τακτά χρονικά διαστήματα μετακίνησε όλα τα κενά σε έναν ελεύθερο τομέα (disk defragmentation)



Δυναμική κατάτμηση





Επανατοποθέτηση Διεργασιών

- Δεν υπάρχει εγγύηση για φόρτωση μιας διεργασίας στην ίδια θέση μνήμης
- Ωστόσο οι εντολές περιέχουν διευθύνσεις
 - Θέση δεδομένων
 - Διευθύνσεις εντολών(branching)
- Λογική διεύθυνση – σχετική με την έναρξη του προγράμματος
- Φυσική διεύθυνση – πραγματική θέση στη μνήμη
- Αυτόματη μετατροπή με χρήση της διεύθυνσης βάσης

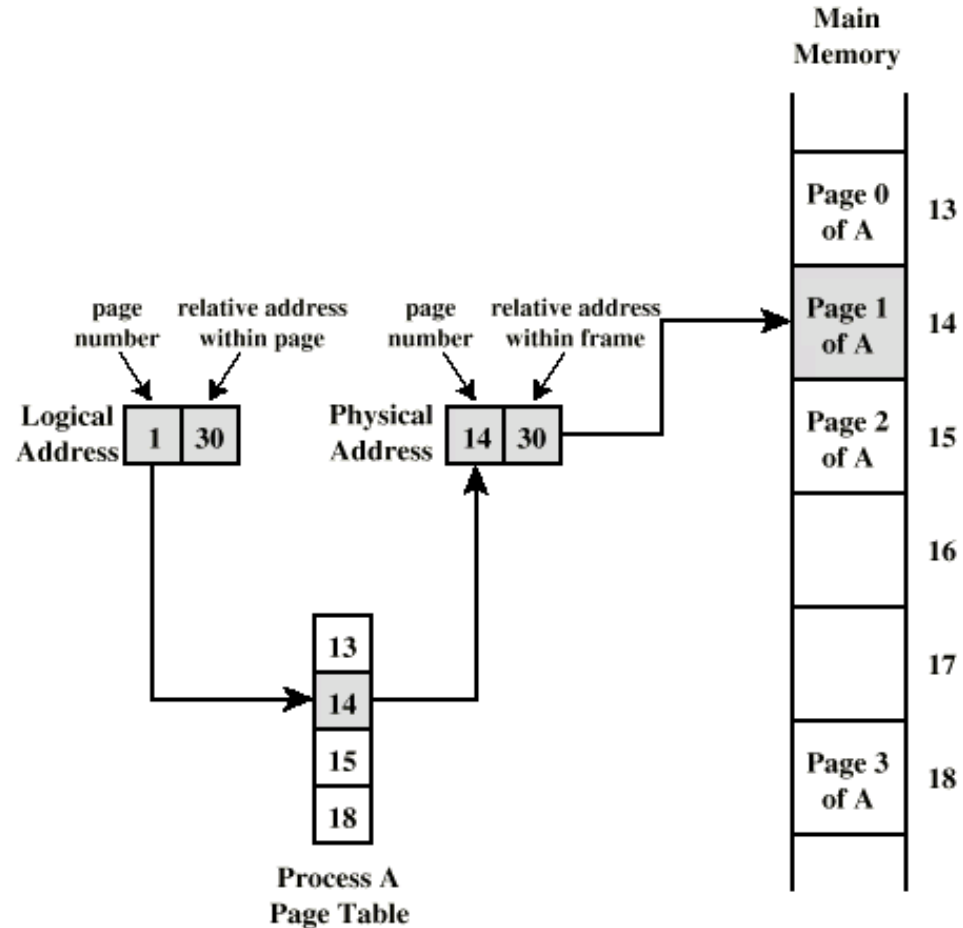


Σελιδοποίηση (Paging)

- Κατάτμηση μνήμης σε πολύ μικρά, ίσου μεγέθους πλαίσια
- Κατάτμηση προγραμμάτων (διεργασιών) σε πολύ μικρές, ίσου μεγέθους σελίδες
- Απόδοση του απαιτούμενου αριθμού πλαισίων σε μια διεργασία
- Το λειτουργικό σύστημα διατηρεί λίστα ελεύθερων πλαισίων
- Μια διεργασία δεν απαιτεί συνεχόμενα πλαίσια
- Χρήση πίνακα σελίδων για ανίχνευση



Λογικές και φυσικές διευθύνσεις - Paging



Εικονική μνήμη

- Απαιτεί σελιδοποίηση
 - Δεν απαιτεί όλες οι σελίδες της διεργασίας να είναι στη κύρια μνήμη
 - Εισαγωγή σελίδων μετά από απαίτηση
- Σφάλμα σελίδας
 - Η επιθυμητή σελίδα δεν είναι στην μνήμη
 - Το λειτουργικό σύστημα πρέπει να εισάγει την σελίδα
 - Ίσως χρειαστεί αντιμετάθεση
 - Επιλογή σελίδας για απομάκρυνση μέσω πρόσφατου ιστορικού



Thrashing

- Πάρα πολλές διεργασίες σε πολύ μικρή μνήμη
- Το λειτουργικό σύστημα ξοδεύει όλο το χρόνο σε αντιμεταθέσεις
- Δεν παράγεται έργο ή είναι ελάχιστο
- Η ένδειξη του δίσκου είναι μόνιμα αναμμένη
- **Λύση**
 - Καλός αλγόριθμος αντιμετάθεσης σελίδων
 - Μείωση εκτελούμενων διεργασιών
 - Τοποθέτηση πρόσθετης μνήμης

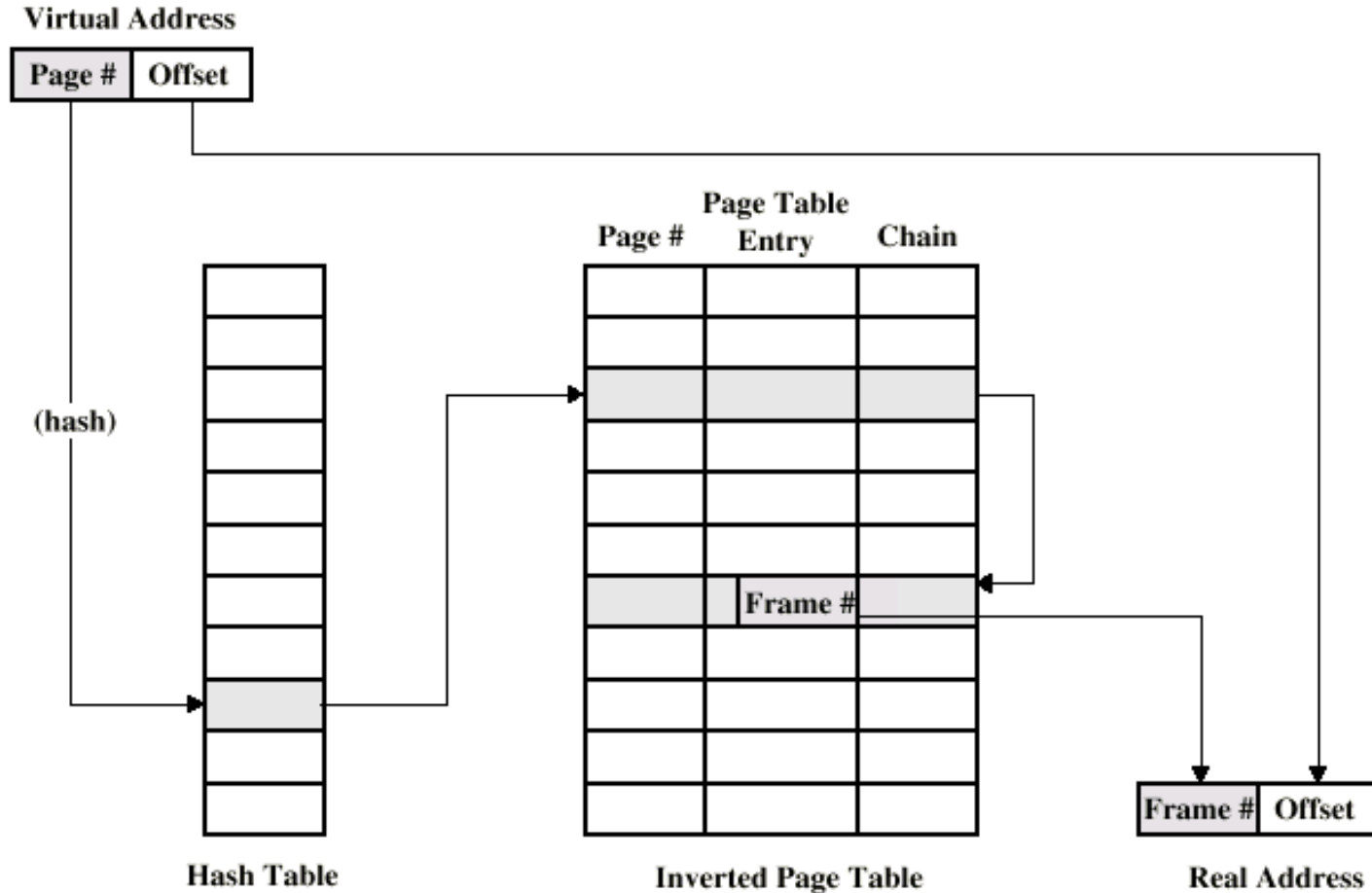


Bonus

- Η εκτέλεση μιας διεργασίας σε ένα συγκεκριμένο χρονικό διάστημα δεν απαιτεί την φόρτωση του συνόλου της διεργασίας
- Μπορούμε να εισάγουμε τις απαιτούμενες σελίδες μετά από αίτημα
- Άρα μπορούμε να εκτελούμε διεργασίες με απαιτήσεις μεγαλύτερες της μνήμης!
- Η κυρίως μνήμη καλείται πραγματική μνήμη
- Ο χρήστης/προγραμματιστής βλέπει πολύ μεγαλύτερη μνήμη – εικονική μνήμη



Δομή Πίνακα Σελίδων



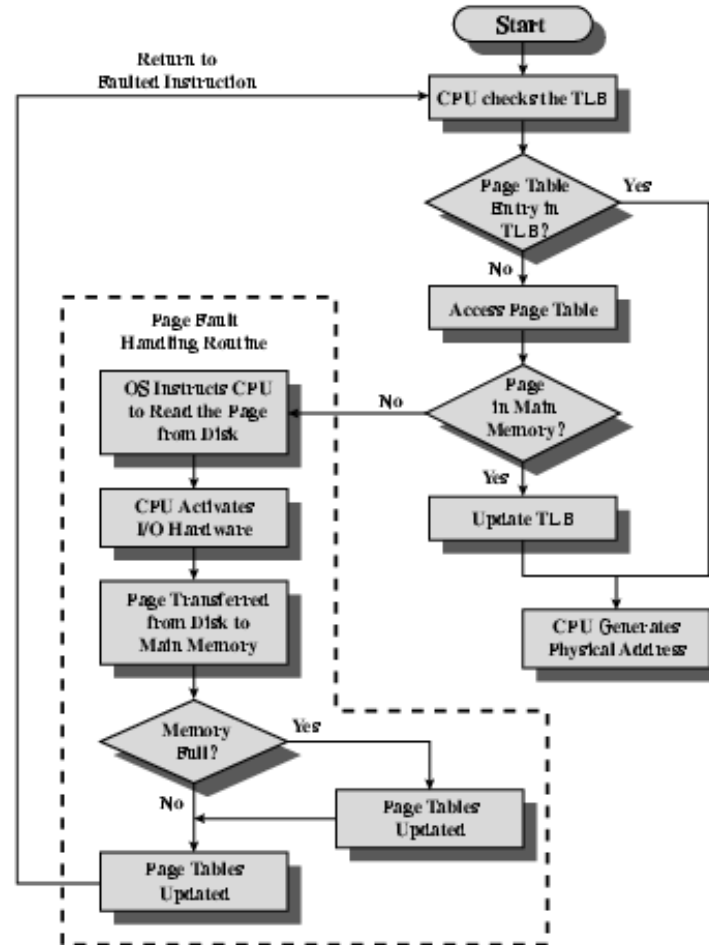


Η ενδιάμεση μνήμη μετατροπής - Translation Lookaside Buffer

- Κάθε αναφορά στην εικονική μνήμη προκαλεί δύο προσπελάσεις της πραγματικής
 - Προσπέλαση πίνακα σελίδων
 - Εξαγωγή δεδομένων
- Χρήση εξειδικευμένης cache για τον πίνακα σελίδων
 - TLB

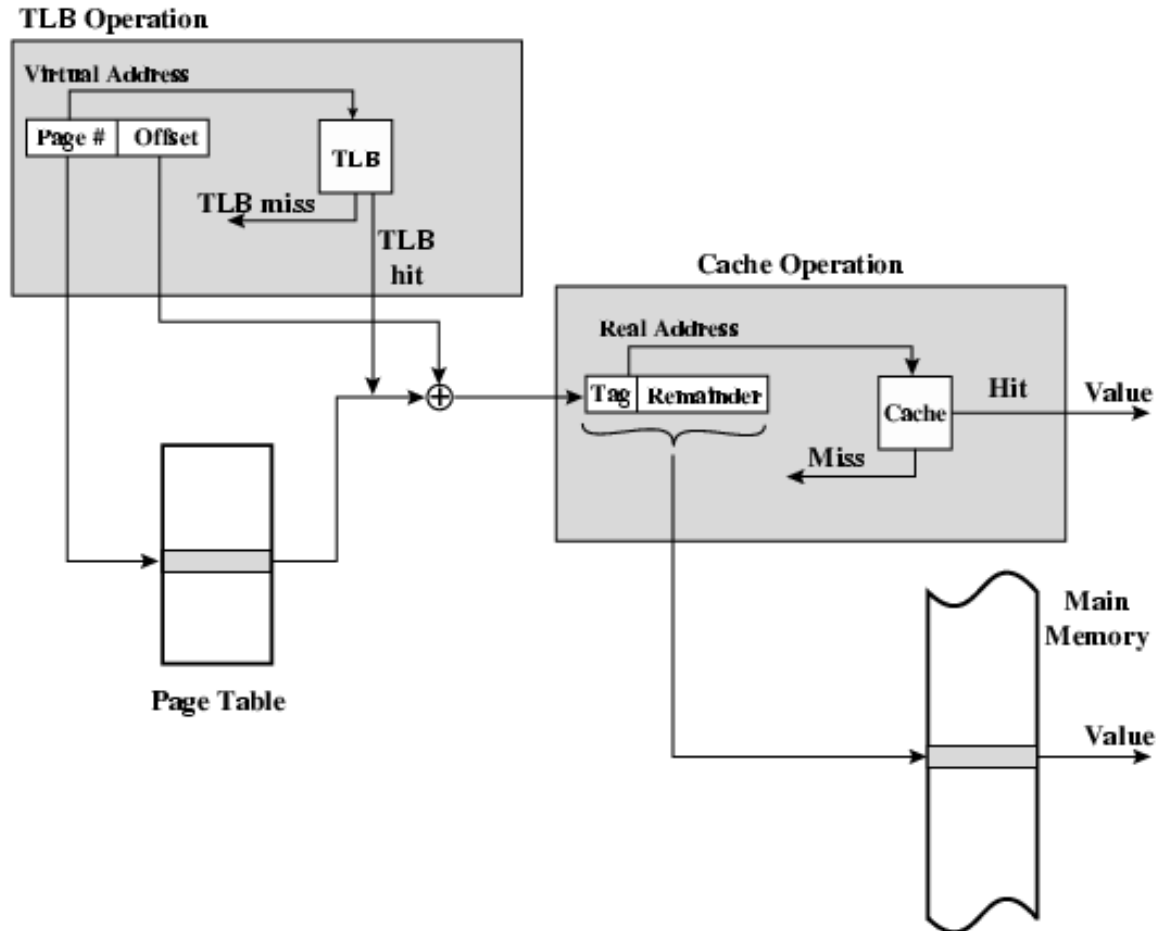


Λειτουργία TLB





Λειτουργία TLB και Cache



Τμηματοποίηση

- Η σελιδοποίηση συνήθως δεν είναι ορατή στον προγραμματιστή
- Η τμηματοποίηση είναι
- Συνήθως διαφορετικά τμήματα αποδίδονται σε διεργασίες και δεδομένα
- Μπορεί να υπάρχει ένας αριθμός τμημάτων προγράμματος όπως και ένας αριθμός τμημάτων δεδομένων



Πλεονεκτήματα Τμηματοποίησης

- Απλοποιεί τον χειρισμό αυξανόμενου μεγέθους δομών δεδομένων
- Επιτρέπει σε προγράμματα να αλλαχθούν και να μεταγλωτιστούν ανεξάρτητα, χωρίς επανασύνδεση – επαναφόρτωση ενός συνόλου προγραμμάτων
- Εύκολη χρήση πόρων από πολλές διεργασίες
- Διευκολύνεται η προστασία των δεδομένων
- Μερικά συστήματα συνδυάζουν την τμηματοποίηση με την σελιδοποίηση



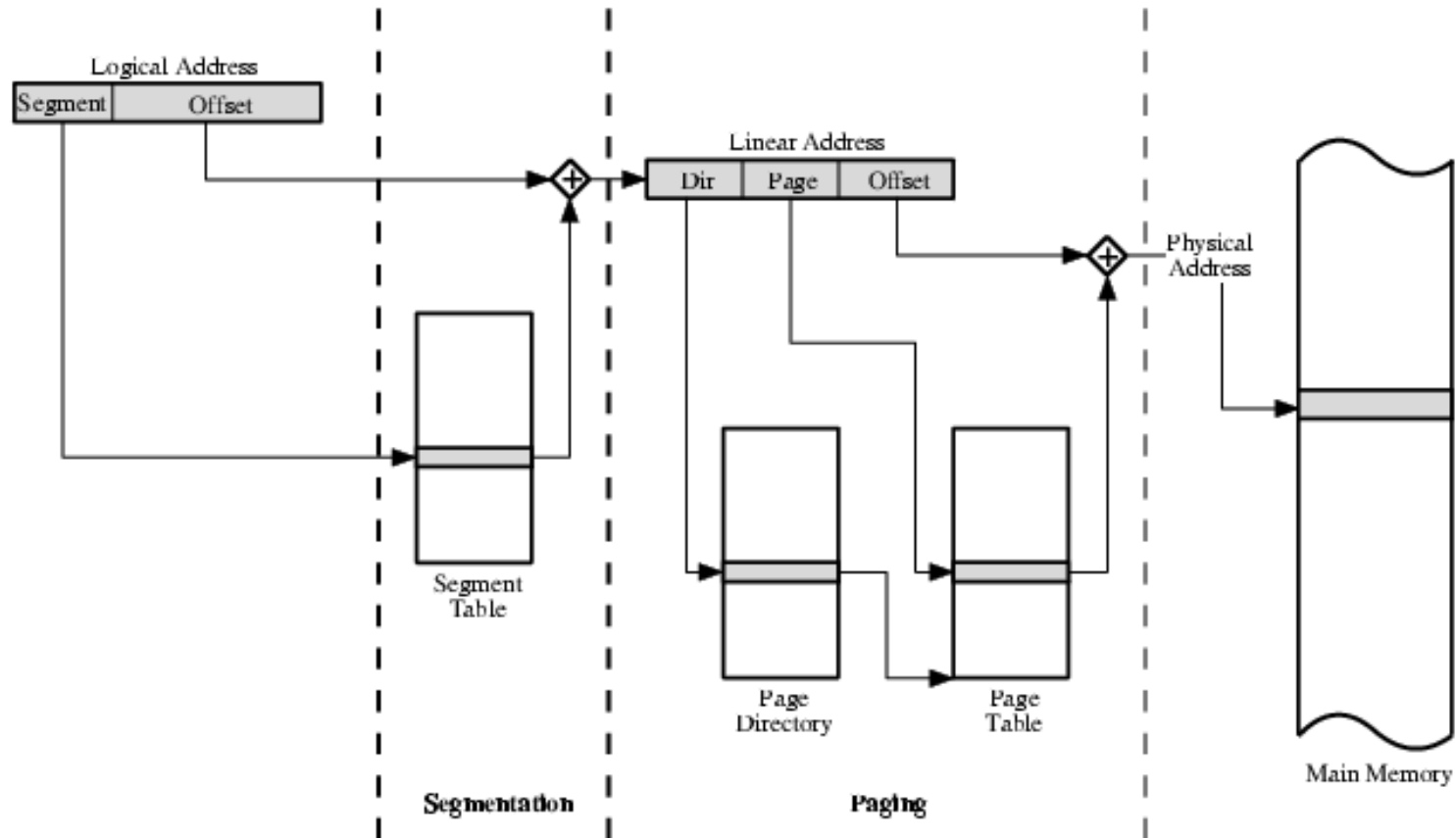
Pentium II

- Hardware για τμηματοποίηση και σελιδοποίηση
- Μη τμηματοποιημένη, μη σελιδοποιημένη
 - Εικονική διεύθυνση = πραγματική διεύθυνση
 - Απλό σχήμα
 - Υψηλή απόδοση
- Μη τμηματοποιημένη, σελιδοποιημένη
 - Η μνήμη βλέπεται σαν ένας σελιδοποιημένος γραμμικός χώρος διευθύνσεων
 - Προστασία και διαχείριση μέσω σελιδοποίησης
 - Berkeley UNIX
- Τμηματοποιημένη, μη σελιδοποιημένη
 - Συλλογή χώρων τοπικών διευθύνσεων
 - Προστασία σε επίπεδο byte
 - Ο απαιτούμενος πίνακας μετατροπής είναι on chip για τμήματα που είναι στην κύρια μνήμη
- Τμηματοποιημένη, σελιδοποιημένη
 - Η τμηματοποίηση χρησιμοποιείται για να ορίσει λογικά τμήματα μνήμης που υπόκεινται σε έλεγχο πρόσβασης
 - Η σελιδοποίηση διαχειρίζεται την κατανομή της μνήμης εντός των τμημάτων
 - Unix System V



Pentium II

Μηχανισμός μετατροπής διεύθυνσης





Τμηματοποίηση στον Pentium II

- Κάθε εικονική διεύθυνση αποτελείται από 16-bit τομέα και 32-bit μετατόπιση
- 2 bits τομέα αποτελούν μηχανισμό προστασίας
- 14 bits καθορίζουν τον τομέα
- Μη τμηματοποιημένη εικονική μνήμη $2^{32} = 4\text{Gbytes}$
- Τμηματοποιημένη $2^{46}=64\text{ terabytes}$
 - Μπορεί να είναι και μεγαλύτερη – Εξαρτάται από το ποια διεργασία είναι ενεργή
 - Μισή (8K segments of 4Gbytes) είναι γενικής χρήσης
 - Μισή διαμοιράζεται τοπικά και ξεχωριστά σε κάθε διεργασία



Προστασία στον Pentium II

- Τα bits προστασίας δίνουν 4 επίπεδα δικαιωμάτων
 - 0 μεγαλύτερη προστασία, 3 ελάχιστη
 - Η χρήση των επιπέδων εξαρτάται από το software
 - Συνήθως το επίπεδο 3 για εφαρμογές, το επίπεδο 1 για O/S και το επίπεδο 0 για kernel (το επίπεδο 2 δεν χρησιμοποιείται)
 - Το επίπεδο 2 μπορεί να χρησιμοποιηθεί από εφαρμογές που διαθέτουν εσωτερική ασφάλεια π.χ. database
 - Κάποιες εντολές τρέχουν μόνο στο επίπεδο 0



Σελιδοποίηση στον Pentium II

- Η τμηματοποίηση μπορεί να απενεργοποιηθεί
 - Σε αυτή την περίπτωση χρησιμοποιείται γραμμικός χώρος διευθύνσεων
- Δύο επίπεδα αναφοράς
 - Πρώτα, κατάλογος σελίδων
 - 1024 καταχωρήσεις max
 - Διαχωρίζει 4G γραμμική μνήμη σε 1024 ομάδες σελίδων 4Mbyte
 - Κάθε πίνακας σελίδων έχει 1024 καταχωρήσεις που αντιστοιχούν σε σελίδες 4Kbyte
 - Μπορεί να χρησιμοποιεί ένα κατάλογο σελίδων για όλες τις διεργασίες ή ένα για κάθε διεργασία ή μεικτά
 - Ο κατάλογος σελίδων για την εκτελούμενη διεργασία είναι πάντα στη μνήμη
 - Χρήση TLB για 32 καταχωρήσεις πίνακα σελίδων
 - Δύο μεγέθη σελίδων διαθέσιμα 4k ή 4M



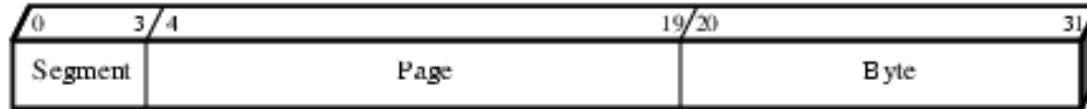
Υλικό διαχείρισης μνήμης για powerPC

- 32 bit – Σελιδοποίηση με απλή τμηματοποίηση
 - 64 bit Σελιδοποίηση με ισχυρότερη τμηματοποίηση
- Ή, και οι δύο κάνουν μετατροπή διευθύνσεων block
 - Χαρτογράφηση 4 μεγάλων blocks εντολών & και 4 μνήμης για αποφυγή σελιδοποίησης
 - π.χ. OS πίνακες ή γραφικά
- 32 bit ενεργή διεύθυνση
 - 12 bit επιλογή byte
 - =4kbyte σελίδες
 - 16 bit ταυτότητα σελίδας
 - 64k σελίδες ανά τομέα
 - 4 bits δεικνύουν 16 καταχωρητές τμήματος
 - Οι τελευταίοι υπό τον έλεγχο του OS

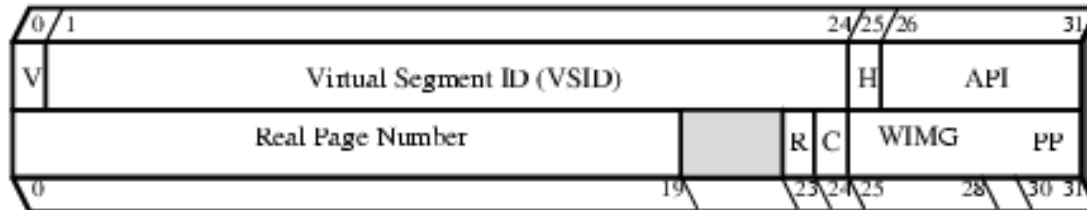


PowerPC 32-bit

Μορφοποίηση για Διαχείριση Μνήμης

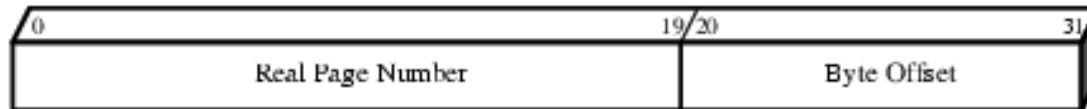


(a) Effective address



V = Entry valid bit
 H = Hash function identifier
 API = Abbreviated page index
 R = Referenced bit
 C = Changed bit
 WIMG = Cache and storage access control bits
 PP = Page protection bits

(b) Page Table Entry

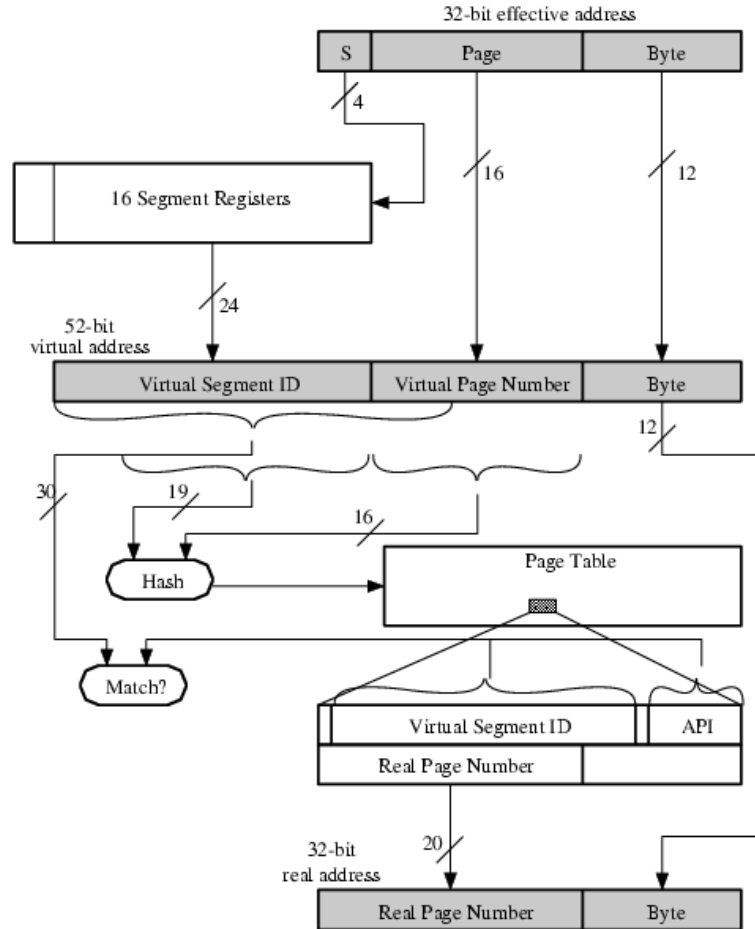


(c) Real address



PowerPC 32-bit

Μετατροπή διεύθυνσης





Βιβλιογραφία

William Stallings. (2011). Αρχιτεκτονική & Οργάνωση Υπολογιστών. Εκδόσεις Τζιόλα.

Δημοσθένης Ε. Μπολανάκης. (2011). Αρχιτεκτονική Μικροϋπολογιστών: αρχές προγραμματισμού χαμηλού επιπέδου και εφαρμογές με το μικροελεγκτή M68HC908GP32, Εκδόσεις Σύγχρονη Παιδεία.

Tanenbaum Andrew S. (1995). Η Αρχιτεκτονική των Υπολογιστών μια δομημένη προσέγγιση Συγγραφέας Tanenbaum Andrew S. Εκδόσεις Κλειδάριθμος.

Luce T. (1991). Αρχιτεκτονική των Υπολογιστών. Εκδόσεις Τζιόλα.

Gilmore. (1999). Μικροεπεξεργαστές θεωρία και εφαρμογές. Εκδόσεις Τζιόλα.

Predko M. (2000). Προγραμματίζοντας τον Μικροελεγκτή PIC, Εκδόσεις Τζιόλα.

Μπεκάκος Μ.Π. (1994). Αρχιτεκτονική υπολογιστών & τεχνολογία παράλληλης επεξεργασίας, Εκδόσεις Σταμούλης.



Σημείωμα Αναφοράς

Copyright Τεχνολογικό Ίδρυμα Ηπείρου. Φώτης Βαρζιώτης.
Αρχιτεκτονική υπολογιστών.

Έκδοση: 1.0 Άρτα, 2015. Διαθέσιμο από τη δικτυακή
διεύθυνση:

<http://eclass.teiep.gr/OpenClass/courses/COMP115/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού-Μη Εμπορική Χρήση-Όχι Παράγωγα Έργα 4.0 Διεθνές [1] ή μεταγενέστερη. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, Διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/deed.el>



Τέλος Ενότητας

Επεξεργασία: Ευάγγελος Καρβούνης
Άρτα, 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Τέλος Ενότητας

Υποστήριξη από το Λειτουργικό Σύστημα



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης