

1 ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ

ΣΚΟΠΟΣ ΤΟΥ ΚΕΦΑΛΑΙΟΥ

- Η γνωριμία με τον τομέα της αρχιτεκτονικής υπολογιστών
- Η αναφορά της χρησιμότητας του μαθήματος σε άλλους τομείς
- Η παρουσίαση των μεθόδων προσέγγισης του αντικειμένου
- Η περιγραφή της γενικής μορφής του υπολογιστικού συστήματος

1.1 Εισαγωγή

Στη διεθνή βιβλιογραφία η ανάλυση της δομής και της λειτουργίας των υπολογιστών αναφέρεται ως οργάνωση ή αρχιτεκτονική. Η δημιουργία αυτών των διαφορετικών ορισμών προέκυψε από το γεγονός ότι ένα υπολογιστικό σύστημα μπορεί να μελετηθεί τόσο από τον προγραμματιστή όσο και από το σχεδιαστή. Ο προγραμματιστής ενδιαφέρεται για τις λεπτομέρειες που επηρεάζουν τον τρόπο ανάπτυξης αλλά και την τελική απόδοση των προγραμμάτων που συνθέτει ενώ ο σχεδιαστής ενδιαφέρεται για τον τρόπο λειτουργίας και σύνθεσης των μονάδων που χρησιμοποιούνται για τη λειτουργία του συστήματος (π.χ. εκτέλεση εντολών).

Στην Ελληνική βιβλιογραφία έχει επικρατήσει ο όρος αρχιτεκτονική ο οποίος αναφέρεται στην εσωτερική δομή και λειτουργία του συστήματος. Αυτή η προσέγγιση δίνει και απαντήσεις που σχετίζονται με τον προγραμματισμό. Για παράδειγμα, ο τρόπος αναπαράστασης μη προσημασμένων αριθμών ο οποίος υπαγορεύεται από την αρχιτεκτονική θέτει τις προδιαγραφές και τους περιορισμούς που θα πρέπει ο προγραμματιστής να λάβει υπόψη του προκειμένου να αναπτύξει λειτουργικό κώδικα.

Πολλές φορές θα ακούσετε και θα διαβάσετε ότι **η αρχιτεκτονική υπολογιστών είναι η «μισή» πληροφορική.**

Αυτό είναι αλήθεια αν σκεφτούμε ότι το υπολογιστικό σύστημα ως πλατφόρμα είναι αυτό που στηρίζει και υλοποιεί τις εφαρμογές που αναπτύσσονται. Ο τρόπος μετάβασης από το επίπεδο των εφαρμογών στο φυσικό επίπεδο (ψηφιακά κυκλώματα) και το αντίστροφο καθορίζει στην ουσία τις δυνατότητες της

εκάστοτε εφαρμογής αλλά και του ίδιου του συστήματος. **Μέσω της αρχιτεκτονικής παρουσιάζεται ο τρόπος που το σύστημα πρωτογενώς αναπαριστά, αποθηκεύει και επεξεργάζεται τις πληροφορίες.** Δεδομένου ότι οι εφαρμογές περιλαμβάνουν διαδικασίες διαχείρισης της πληροφορίας, οι γνώσεις που προκύπτουν από την αρχιτεκτονική υπολογιστών εφαρμόζονται άμεσα σε ένα σύνολο από τομείς όπως :

- δίκτυα
- επεξεργασία εικόνας
- προγραμματισμός
- λειτουργικά συστήματα
- ψηφιακές επικοινωνίες
- ασφάλεια
- βιομηχανικές εφαρμογές
- μοντελοποίηση
- μετάδοση δεδομένων, κλπ

Για παράδειγμα, η απαιτούμενη χωρητικότητα σε Mbps για ένα τηλεπικοινωνιακό κανάλι προκειμένου να μεταδώσουμε βίντεο πραγματικού χρόνου εξαρτάται από τη διάσταση του καρέ, το βάθος χρώματος ανά εικονοστοιχείο και το πλήθος των πλαισίων που μεταφέρονται στη μονάδα του χρόνου. Το βάθος χρώματος που δεν είναι τίποτα άλλο από την κωδικοποίηση χρώματος για κάθε εικονοστοιχείο βασίζεται σε αρχές που υπαγορεύει η αρχιτεκτονική των υπολογιστών σε επίπεδο ψηφιακών κυκλωμάτων για την αναπαράσταση της πληροφορίας.

Η αρχιτεκτονική υπολογιστών μπορεί να μελετηθεί σε **επίπεδο προγραμματισμού ή ψηφιακών κυκλωμάτων**. Η πιο δημοφιλής προσέγγιση είναι αυτή του προγραμματισμού σε γλώσσα assembly, μια γλώσσα χαμηλού επιπέδου, ιδιόμορφη αλλά και αποτελεσματική. Οι εφαρμογές της γλώσσας assembly είναι σημαντικές ενώ σε πολλές περιπτώσεις η επιλογή της είναι μονόδρομος. Πιο συγκεκριμένα, η γλώσσα αυτή χρησιμοποιείται σε εφαρμογές όπως:

- έλεγχος του υλικού
- κατασκευή drivers
- ανάκτηση δεδομένων
- λογισμικό κινητών τηλεφώνων και MP3 players
- βιομηχανικοί αυτοματισμοί
- δίκτυα αισθητήρων, κλπ.

Από τα παραπάνω προκύπτει ότι οι γνώσεις της αρχιτεκτονικής υπολογιστών αποτελούν τη βάση σε τομείς που σχετίζονται με τη σχεδίαση και προγραμματισμό μικροεπεξεργαστών και μικροελεγκτών, ψηφιακές διατάξεις, συστήματα πλοήγησης, ρομποτική, κλπ.

1.2 Τρόποι προσέγγισης της αρχιτεκτονικής του συστήματος

Είναι αλήθεια ότι οι περισσότεροι φοιτητές έχουν μεγαλύτερη επαφή με τον προγραμματισμό και όπως είναι αναμενόμενο προτιμούν την ανάλυση της αρχιτεκτονικής υπολογιστών μέσω αυτού του εργαλείου. Η ανάλυση της αρχιτεκτονικής και της λειτουργίας της ΚΜΕ (Κεντρική Μονάδα Επεξεργασίας) θεωρείται η σημαντικότερη γνώση στην αρχιτεκτονική υπολογιστών αφού εκεί εκτελούνται οι εντολές βάσει ειδικών ψηφιακών κυκλωμάτων που υλοποιούν μια μεγάλη ποικιλία διαδικασιών (π.χ. αριθμητικές και λογικές πράξεις, αποκωδικοποίηση, έλεγχος, κλπ).

Ένας κλασικός προγραμματιστής γράφει την εντολή `printf("hello");` η οποία εμφανίζει στην οθόνη το μήνυμα `hello`. Αν αναζητήσουμε το τι κρύβεται πίσω από αυτή την εντολή (αναφορικά με το μηχανισμό εκτέλεση της), θα μπορούμε στη διαδικασία να ανακαλύψουμε τη λειτουργία και την αρχιτεκτονική της ΚΜΕ.

Η γλώσσα χαμηλού επιπέδου `assembly` βρίσκεται στο ενδιάμεσο επίπεδο μεταξύ των γλωσσών υψηλού επιπέδου (π.χ. C, Java) και της ΚΜΕ. Οι εντολές αλλά και η φιλοσοφία της γλώσσας `assembly` βασίζονται αποκλειστικά στην υφιστάμενη αρχιτεκτονική της ΚΜΕ. Αυτό το χαρακτηριστικό (ως προς τη συσχέτιση με την αρχιτεκτονική) ικανοποιεί τους προγραμματιστές αφού δεν αναγκάζονται να κατευθυνθούν σε περιοχές εκτός προγραμματισμού.

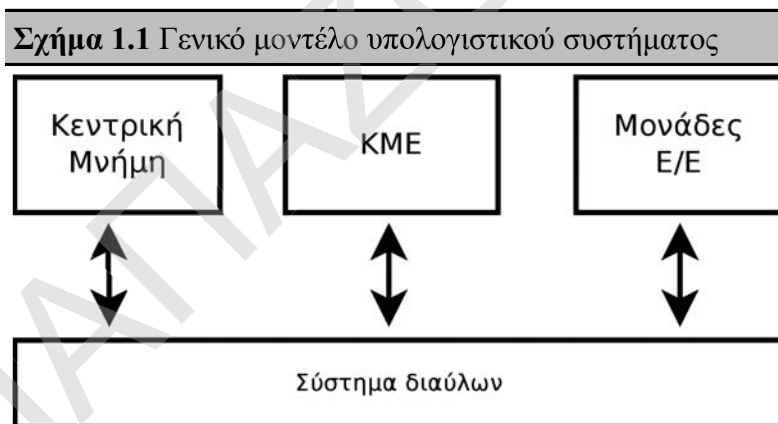
Επιπλέον, η γλώσσα `assembly` διαθέτει αξιοπρόσεκτα χαρακτηριστικά αναφορικά με τον έλεγχο της ΚΜΕ και άρα του υπολογιστικού συστήματος. Έτσι, μπορεί κάποιος ως προγραμματιστής να ελέγξει τη λειτουργία αλλά και να μελετήσει την αρχιτεκτονική του συστήματος.

Από την άλλη πλευρά, η αρχιτεκτονική του συστήματος μπορεί να προσεγγιστεί μελετώντας τα ψηφιακά κυκλώματα που χρησιμοποιούνται στην υλοποίηση των διαφόρων συνιστωσών του συστήματος ή ακόμα και σχεδιάζοντας τα κυκλώματα αυτά μέσω συγκεκριμένων εργαλείων και μεθόδων.

1.3 Γενικό μοντέλο υπολογιστικού συστήματος

Ένα υπολογιστικό σύστημα (π.χ. υπολογιστής) έχει τη δυνατότητα να επεξεργάζεται δεδομένα με τα οποία το τροφοδοτούμε και να παράγει αντίστοιχα αποτελέσματα. Η επεξεργασία καθορίζεται από ένα σύνολο σαφώς καθορισμένων εντολών που ονομάζεται πρόγραμμα. Ανάλογα τον τύπο και το είδος του προγράμματος, η αντίστοιχη επεξεργασία μπορεί να παράγει διαφορετικά αποτελέσματα.

Για παράδειγμα, ένα πρόγραμμα μπορεί να πραγματοποιεί στατιστική επεξεργασία και να αποτυπώνει γραφικά τη συμπεριφορά ενός υπό μελέτη συστήματος ή απλά να εμφανίζει ένα μήνυμα. Επίσης, τα παραγόμενα αποτελέσματα μπορεί να αποτελούν με τη σειρά τους είσοδο σε ένα σύστημα που ανήκει στην επόμενη βαθμίδα ενός μεγαλύτερου συστήματος. Η εκτέλεση των εντολών του προγράμματος και άρα η πραγματοποίηση της ζητούμενης επεξεργασίας γίνεται από την ΚΜΕ. Η ΚΜΕ περιλαμβάνει ένα σύνολο από ψηφιακά κυκλώματα τα οποία με συντονισμένο τρόπο αναλαμβάνουν την εκτέλεση κάθε εντολής. Λόγω αυτού του ρόλου, η ΚΜΕ αποτελεί την καρδιά κάθε υπολογιστικού συστήματος ενώ τα χαρακτηριστικά της (π.χ. συχνότητα ρολογιού) λειτουργούν ως βαρόμετρο για την απόδοσή του. Το σχήμα 1.1 δείχνει το γενικό μοντέλο του υπολογιστικού συστήματος.



Η εισαγωγή δεδομένων (τροφοδοσία δεδομένων) πραγματοποιείται από εξωτερικές συσκευές (ή αλλιώς διατάξεις) εισόδου που αναλαμβάνουν και την απαιτούμενη μετατροπή τους ώστε αυτά να είναι κατανοητά από το σύστημα.

Για παράδειγμα, το ηλεκτρολόγιο αποτελεί μονάδα εισόδου μεταφέροντας τα κατάλληλα ηλεκτρικά σήματα στο σύστημα ανάλογα με τα πλήκτρα που πιέζονται. Αντίστοιχα, η οθόνη (μονάδα εξόδου) αποτυπώνει οπτικά τα αποτελέσματα της επεξεργασίας ώστε να είναι αντιληπτά και κατανοητά από τον άνθρωπο.

Από την άλλη πλευρά, όλα τα δεδομένα προς επεξεργασία αλλά και οι εντολές του προγράμματος φιλοξενούνται στην κεντρική μνήμη.

Η εκτέλεση κάθε εντολής του προγράμματος προϋποθέτει τη μεταφορά της αντίστοιχης εντολής από την κεντρική μνήμη στην ΚΜΕ. Η ροή των δεδομένων μεταξύ όλων των υπομονάδων του συστήματος (κεντρική μνήμη, ΚΜΕ, μονάδες Ε/Ε-Εισόδου/Εξόδου) πραγματοποιείται μέσω του συστήματος διαύλων το οποίο υποστηρίζει τις αντίστοιχες διαδικασίες.

Περίληψη κεφαλαίου

Σε αυτό το κεφάλαιο παρουσιάστηκαν ορισμένες βασικές έννοιες που αφορούν στο αντικείμενο της αρχιτεκτονικής υπολογιστών ενώ ταυτόχρονα έγινε μια προσπάθεια ώστε να φανεί το πόσο σημαντικό είναι αυτό το αντικείμενο σε όλους τους τομείς της επιστήμης της πληροφορικής.

Η Αρχιτεκτονική των υπολογιστών μπορεί να μελετηθεί από δύο διαφορετικές οπτικές γωνίες ανάλογα με το στόχο του μηχανικού πληροφορικής. Ο σχεδιαστής ενδιαφέρεται για την εσωτερική αρχιτεκτονική και την υλοποίηση των λειτουργιών σε επίπεδο ψηφιακών κυκλωμάτων ενώ ο προγραμματιστής ενδιαφέρεται για τις συνιστώσες εκείνες που επηρεάζουν την ανάπτυξη των προγραμμάτων. Στη βιβλιογραφία γίνεται αναφορά κυρίως στην οργάνωση υπολογιστή που θεωρείται ότι ανταποκρίνεται στα στοιχεία που επηρεάζουν τον προγραμματιστή.

Η γλώσσα Assembly υποστηρίζεται απευθείας από την κεντρική μονάδα επεξεργασίας (ΚΜΕ) και για αυτό το λόγο αποτελεί ιδανικό εργαλείο τόσο για τη μελέτη της αρχιτεκτονικής όσο και για τον απόλυτο έλεγχο του υπολογιστή.

Τέλος, παρουσιάστηκαν οι συνιστώσες του γενικού μοντέλου του υπολογιστικού συστήματος οι οποίες αλληλεπιδρούν μεταξύ τους ώστε να είναι δυνατή η εκτέλεση των προγραμμάτων.

ΑΛΥΤΕΣ ΑΣΚΗΣΕΙΣ

- Άσκηση 1** Από ποιες οπτικές γωνίες μπορεί να μελετηθεί ένα υπολογιστικό σύστημα ;
- Άσκηση 2** Ποια τα πλεονεκτήματα των δύο τρόπων μελέτης ;
- Άσκηση 3** Πως μπορεί η αρχιτεκτονική να επηρεάσει τον κώδικα που αναπτύσσει ο προγραμματιστής ;
- Άσκηση 4** Να αναφέρετε ένα παράδειγμα χρήσης γνώσεων αρχιτεκτονικής σε ένα φαινομενικά διαφορετικό τομέα.
- Άσκηση 5** Ποια είναι τα βασικά πλεονεκτήματα που προσφέρει ο προγραμματισμός σε γλώσσα assembly ;
- Άσκηση 6** Περιγράψτε σε γενικές γραμμές τη διαδικασία υλοποίησης μιας εντολής υψηλού επιπέδου όπως η printf.
- Άσκηση 7** Ποιες είναι οι βασικές συνιστώσες ενός υπολογιστικού συστήματος ;
- Άσκηση 8** Ποια είναι η χρησιμότητα του συστήματος διαύλων ;
- Άσκηση 9** Να αναφέρετε με ένα παράδειγμα τη διαδικασία που λαμβάνει χώρα κατά τη λειτουργία μιας εφαρμογής (χρήση των συνιστωσών του υπολογιστικού συστήματος).
- Άσκηση 10** Να αναφέρετε άλλες ηλεκτρονικές συσκευές ή διατάξεις που διαθέτουν συνιστώσες της ίδιας μορφής με αυτές του κλασικού υπολογιστικού συστήματος.