

Τ.Ε.Ι. ΠΑΤΡΑΣ  
Τμήμα : ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΘΕΜΑ : " ΕΙΣΑΓΩΓΗ...ΣΤΟΥΣ ΗΛΕΚΤΡΟΝΙΚΟΥΣ ΥΠΟΔΟΧΙΣΤΕΣ...  
ΚΑΙ ΕΦΑΡΜΟΓΗ ΠΡΟΓΡΑΜΜΑΤΟΣ ΑΠΟΘΗΚΗΣ  
ΣΕ ΜΙΑ ΜΙΚΡΟΜΕΣΑΙΑ ΕΠΙΧΕΙΡΗΣΗ"



Σπουδαστής : Γεώργιος Ευαγγέλος  
Εξάμηνο Φοίτησης : Γ' Πτυχίο

Υπεύθυνος Καθηγητής : Κ. Καραούσου Β.

ΑΡΙΘΜΟΣ ΕΙΣΑΓΩΓΗΣ	676
----------------------	-----



## ΠΕΡΙΕΧΟΜΕΝΑ

1. Εισαγωγή
2. Ολοκληρωμένο Κυκλώμα
3. HARDWARE
  - 3.1 Κεντρική μονάδα επεξεργασίας (CPU)
  - 3.2 Σειριακός επεξεργαστής
  - 3.3 Καταχωρητές
  - 3.4 Διακόπτες
  - 3.5 Μικροεπεξεργαστής
  - 3.6 Ο Zilog Z80
  - 3.7 Οι 16-μπιτου και 32-μπιτου μικροεπεξεργαστές
  - 3.8 Η οικογένεια MOTOROLA 68000
4. SOFTWARE
  - 4.1 Επεξεργαστές κειμένου (Scriptum)
  - 4.2 Βάσεις δεδομένων (DBASE IV)
  - 4.3 Ολοκληρωμένα πακέτα (LOTUS 1-2-3)
5. Λειτουργικό Σύστημα
  - 5.1 Το MS-DOS
  - 5.2 Το UNIX
  - 5.3 Το OS/2
6. Η ιστορία του πρώτου P.C (Personal Computer)
7. Νευρωνικά δίκτυα
8. Πρόγραμμα Αποθήκης

## ΕΙΣΑΓΩΓΗ.

Μια θεμελιώδης αλλαγή στις ανθρώπινες υποθέσεις βρίσκεται υπο εξέλιξη, τόσο σημαντική όσο και η βιομηχανική Επανάστασή ή η εφεύρεση της τυπογραφίας από τον Γουτεμβέργιο. Η τεχνική φύση της μικροηλεκτρονικής δεν την άφησε να φανεί σε γενική όψη, μέχρι το τέλος της δεκαετίας του 1970, αλλά καθώς διεισδύει σχεδόν σε κάθε πλευρά της κοινωνίας μας, η σημασία της μικροηλεκτρονικής επανάστασης γίνεται φανερή στον καθένα.

Η ταχύς ενός υπολογιστή μπορεί τώρα να χρησιμοποιηθεί με αμελητέο κόστος πάνω σε ένα πρόβλημα, προσφέροντας μας μια επέκταση των διανοητικών μάλλον παρά των φυσικών μας δυνατοτήτων.

Το κόστος των υπολογιστών έχει πέσει μέσα σε τριάντα χρόνια κατά ένα εκατομμύριο φορές, μεταβάλλοντας τους από μια απόκρυφη πολυτέλεια σε ανάλωσιμό στοιχείο. Το αποτέλεσμα είναι ότι, σχεδόν σε κάθε βιομηχανία, επάγγελμα και προϊόν, η μικροηλεκτρονική αποτελεί την κυρίαρχη δύναμη για αλλαγή. Οι περισσότεροι είμαστε καχύποπτοι για την αλλαγή, αλλά ωθούμαστε αναγκαστικά προς αυτήν, γιατί αν δεν υιοθετήσουμε εμείς με ενθουσιασμό τη νέα τεχνολογία, θα το κάνουν κάποιοι άλλοι. Οι εταιρείες που αντιμετωπίζουν τη μικροηλεκτρονική σαν ένα παραδεδεγμένο στάδιο προορίζονται να αφανιστούν οικονομικά.

Η νέα τεχνολογία μπορεί να εμπλουτίσει τη ζωή μας, απαλλάσσοντάς μας από τα βαρετά καθήκοντα και την αγγαρεία της επαναληπτικής εργασίας, αλλά είναι ένα δίκαιο μαχαίρι. Τα ίδια κυκλώματα μπορούν να ελέγχουν τη θερμοκρασία ενός παιδιού και να οδηγούν ένα βλήμα. Ολόκληρος ο κόσμος θα επωφεληθεί οικονομικά από την τεχνολογία, αλλά ο πλούτος αναμφίβολα θα συγκεντρωθεί ακόμη περισσότερο στις εξελιγμένες χώρες.

Ειδικότερα οι χώρες του Τρίτου Κόσμου απειλούνται σοβαρά από την μικροηλεκτρονική, γιατί η τεχνολογία κυριαρχείται από τις ΗΠΑ, από ορισμένες χώρες της Άπω Ανατολής, και από τη Δυτική Ευρώπη.

Οι βιομηχανίες, τις οποίες οι υποανάπτυκτες χώρες πόλεψαν για να φτιάξουν πρόκειται να χάσουν την αξία τους από τη ρομποτική, τη μαζική παραγωγή και την τεχνολογία της πληροφορικής.

Η εισαγωγή της μικροηλεκτρονικής καταστρέφει πολλές παραδοσιακές βιομηχανίες. Τα μηχανικά ρολόγια και οι αριθμομηχανίες αποτελούν τεκμηριωμένα παραδείγματα προϊόντων, των οποίων οι βιομηχανίες έγιναν απαρχαιωμένες λόγω της μικροηλεκτρονικής.

Η ρομποτική και τα αυτοματοποιημένα μηχανήματα αρχίζουν να μειώνουν τις ανάγκες για εργατικά χέρια στα σύγχρονα εργοστάσια - μια διαδικασία που θα επιταχυνθεί, καθώς σχεδιάζεται η κατασκευή νέων προϊόντων από ρομπότ.

Οι σοβαρότερες μελλοντικές αλλαγές θα μπορούσαν να επέλθουν στην εργασία γραφείου, όπου οι τεχνικές για την εύκολη αποθήκευση μεγάλων ποσοτήτων πληροφοριών και για την ηλεκτρονική μετάδοση του ταχυδρομείου θα βελτιώσουν την παραγωγικότητα, αλλά θα μειώσουν και την απασχόληση. Στις εξελυγμένες χώρες το 50 τοίς εκατό του εργατικού δυναμικού οργανώνει και χειρίζεται πληροφορίες με τη μία ή την άλλη μορφή. Εται ο έξυπνος, βασισμένος στους υπολογιστές εξοπλισμός για την διαχείριση, αποθήκευση και μετάδοση πληροφοριών θα βελτιώσει την αποτελεσματικότητα, αλλά θα καταργήσει θέσεις εργασίας.

Η μικροηλεκτρονική όμως, θα κάνει να ξεκινήσουν και νέες βιομηχανίες. Αποτελεί ήδη την κινούσα δύναμη πίσω από μια ποικιλία συσκευών τόσο διαφορετικών, όσο είναι ο ψηφιακός δίσκος, η ηλεκτρονική ανάφλεξη του αυτοκινητού, οι μουσικοί συνθετητές, και τα βίντεο-παιχνίδια. Μιά καινούργια βιομηχανία είναι η κατασκευή και η εφαρμογή των ίδιων των προσωπικών υπολογιστών, που έχει αναπτυχθεί σε έναν από τους σπουδαιότερους κλάδους μέσα σε λίγα μονάχα χρόνια.

Μέσα σε τρία χρόνια από την εμφάνισή του, ο προσωπικός υπολογιστής IBM πουλούσε σε ρυθμό 200.000 μηχανήματα το μήνα - ένας μοναχά οικιακός υπολογιστής δημιουργεί τόση τόσο μεγάλο, όσος είναι εκείνος της αυτοκινητοβιομηχανίας του Ηνωμένου Βασιλείου. Αν υπάρχει ένας μονάχα λόγος για τον αντίκτυπο της μικροηλεκτρονικής, αυτός είναι η γενική φύση του υπολογιστή. Ως τώρα είμαστε συνηθισμένοι σε μηχανές που εκτελούν μια μονάχα λειτουργία, όπως είναι τα τρυπάνια, οι τέρνοι, τα ποδήλατα και οι γραφομηχανές. Η λειτουργία ενός υπολογιστή καθορίζεται από το πρόγραμμα ελέγχου του, ενώ το ίδιο το μηχάνημα είναι σημαντικό μόνο σαν ένα μέσο που δίνει τη δυνατότητα στο πρόγραμμα, ή software, να τρέξει. Κάθε φορά που ένας υπολογιστής δέχεται ένα ένα καινούργιο πρόγραμμα, γίνεται ένα διαφορετικό μηχάνημα, χωρίς καμιά αλλαγή στη φυσική του μορφή.

Ο ίδιος υπολογιστής μπορεί να προγραμματιστεί, ώστε να εκτελεί αριθμητικές πράξεις, να συνθέτει ομιλία, να ενεργεί σαν ένας ηλεκτρονικός πίνακας σχεδίασης ή να δουλεύει σαν επεξεργαστής

κειμένου. Επειδή τα προγράμματα είναι τόσο ισχυρά, η χρήση τους μπορεί να απλοποιήσει τη μηχανολογική σχεδίαση των μηχανών.

Μια σύγχρονη ηλεκτρονική γραφομηχανή έχει πολύ λιγότερα κινούμενα μέρη, γιατί οι παλιές κασόνιες, οι μοχλοί και τα γρανάζια έχουν αντικατασταθεί από ένα πολύ απλούστερο μηχανήμα το οποίο ελέγχεται από ένα σύνθετο πρόγραμμα.

Παρ' όλη την απλότητα του από μηχανολογική άποψη, το νέο μηχανήμα μπορεί να κάνει περισσότερα από το παλιό. Η πολυπλοκότητα βρίσκεται μέσα στο πρόγραμμα, το οποίο είναι εύκολο να παραχθεί μαζικά και να ελεγχθεί.

Η μηχανολογική πολυπλοκότητα έχει αντικατασταθεί από την πολυπλοκότητα του προγράμματος.

### Τ ο σ χ ο κ λ η ρ ώ μ ε ν ό κ ύ κ λ ω μ α

Για να φτάσουμε όμως στις σημερινές "ευτυχισμένες" μέρες της μικροηλεκτρονικής χρειάστηκε να κατατεθεί το 1959 από τον Jack S. Kilby, η αίτηση ευρεσιτεχνίας του για το ολοκληρωμένο κύκλωμα, όπου ανέφερε τα εξής: "Όπου όλα τα μέρη του ηλεκτρονικού κυκλώματος είναι τελείως ολοκληρωμένα μέσα στο σώμα του υλικού."

Και στην ανακάλυψη του ολοκληρωμένου κυκλώματος, η "μικρο-ταύπη", βρίσκεται το κλειδί της σταθερής ανάπτυξης της μικροηλεκτρονικής. Η διαδικασία που μετασχηματίζει την άμμο σε ολοκληρωμένα κυκλώματα στηρίζεται σε δεκαετίες εντατικής έρευνας, καθώς και τα 25 χρόνια εφαρμοσμένης ανάπτυξης. Η θεωρία που βρίσκεται πίσω από τους ημιαγωγούς διαμορφώθηκε κατά τις δεκαετίες του 1920 και του 1930, ενώ τα πρώτα τρανζίστορ κατασκευάστηκαν στα εργαστήρια της Bell το 1947. Από την εποχή της εφεύρεσης των ολοκληρωμένων κυκλωμάτων το 1958, οι τεχνικές για την κατασκευή πολλών τρανζίστορ πάνω σ' ένα τσιπ ημιαγωγού έχουν αναπτυχθεί τρομακτικά. Τα ολοκληρωμένα κύκλωμα που πουλιούνται για μερικές δραχμές είναι αποτέλεσμα αυτής της ερευνητικής προσπάθειας και της επένδυσης δισεκατομμυρίων δολλαρίων. Σήμερα, φαίνεται να μην υπάρχει τεχνολογικό φράγμα το οποίο να εμποδίζει την κατασκευή ολοκληρωμένων κυκλωμάτων με δεκάδες εκατομμύρια τρανζίστορ. Το γενικό όνομα αυτής της τεχνολογίας είναι ολοκλήρωση πολύ μεγάλης κλίμακας (VLSI).

Η διαδικασία κατασκευής ενός τσιπ έχει τρία στάδια. Στο πρώτο, το ολοκληρωμένο κύκλωμα σχεδιάζεται και τελειοποιείται χρησιμοποιώντας έναν υπολογιστή σαν έναν έξυπνο πίνακα σχεδίασης και σημειωματάριο. Κατόπιν ένα πρόγραμμα προσομοιώνει ολοκληρω το κύκλωμα για να εντοπίσει πιθανά προβλήματα. Όταν ο σχεδιασμός είναι ικανοποιητικός, ο υπολογιστής φτιάχνει μια σειρά από φωτογραφικές μάσκες, οι οποίες ελέγχουν την επεξεργασία των τσιπ

ΠΑΡΑΓΩΓΗ ΟΛΟΚΛΗΡΩΜΕΝΟΥ ΚΥΚΛΩΜΑΤΟΣ

Σχεδιάσε και ελέγξε το ολοκληρωμένο κύκλωμα στον υπολογιστή

Δημιούργησε τις μάσκες που ελέγχουν την επεξεργασία

ΣΧΕΔΙΑΣΗ ΤΟΥ

ΤΣΙΠ

Δημιουργία μεγάλων κρυστάλλων πυριτίου

Κόψε τους κρυστάλλους σε υποστρώματα

Γυάλισε το υποστρώμα

ΠΑΡΑΓΩΓΗ ΥΠΟΣΤΡΩΜΑΤΟΣ

Χημική επεξεργασία του υποστρώματος

Ελέγξε το κόβει τσίπ πάνω στο υποστρώμα

Χώρισε το υποστρώμα σε ξεχωριστά τσίπ

Τοποθέτησε στα περιβλήματα τους τα τσίπ που δουλεύουν

ΚΑΤΑΣΚΕΥΗ ΤΟΥ ΤΣΙΠ

κατά την διάρκεια των σταδίων κατασκευής.

Στο δεύτερο στάδιο, πολύ καθαρό πυρίτιο λιώνεται μέσα σε ένα κλειστό δοχείο, και από το υγρό δημιουργείται ένας μεγάλος κρύσταλλος. Αυτός κόβεται σε λεπτά υποστρώματα, τα οποία στη συνέχεια γυαλίζονται και εξομαλύνονται.

Το τελευταίο στάδιο πραγματοποιείται στο χυτήριο πυρίτιου, όπου τα λεπτά υποστρώματα του πυρίτιου μετατρέπονται σε τσίπ μέσω μιας σειράς χημικών διεργασιών. Εκατομμύρια τρανζίστορ σχηματίζονται ταυτόχρονα στην επιφάνεια του κάθε υποστρώματος, σε σχηματισμούς που ελέγχονται από μάσκες. Τα τσίπ που δημιουργούνται ελέγχονται ενώ βρίσκονται στο υπόστρωμα, το οποίο στη συνέχεια χωρίζεται στα μεμονωμένα τσίπ. Τα δίχως σφάλματα κυκλώματα τοποθετούνται στα περιβλήματα τους και είναι έτοιμα για διάθεση στο εμπόριο.

Ιδιαίτερη μνεία πρέπει να γίνει σε ένα ουσιώδες στοιχείο στη σύγχρονη παραγωγή ολοκληρωμένων κυκλωμάτων και που είναι η δυνατότητα κατασκευής τρανζίστορ όταν υπάρχει πρόσβαση μονάχα στη μία πλευρά του υποστρώματος πυρίτιου -τεχνολογία του ενός επιπέδου.- Η τεχνολογία του ενός επιπέδου επινοήθηκε το 1960 από

μια ομάδα επιστημόνων που δούλευαν στη Fairchild, και εκτόπισε γρήγορα την τεχνολογία ολοκληρωμένων κυκλωμάτων της Texas Instruments. Η τεχνολογία του ενός επιπέδου επιτρέπει να αυτοματοποιηθεί σε μεγάλο βαθμό η κατασκευή των τρανζίστορ. Τα πρώτα ολοκληρωμένα κυκλώματα παράγονταν μέσω μιας κοπιαστικής επεξεργασίας στην οποία περιλαμβάνονταν μια λεπτή τεχνική χάραξης και των δύο πλευρών του ημιαγωγού.

Τώρα το κάθε τρανζίστορ σχηματίζεται με μια σειρά χημικών διεργασιών στη μια μονάχα επιφάνεια του υποστρώματος. Τα ποιές επιφάνειες επηρεάζονται από την κάθε μια απ' αυτές τις διεργασίες, καθορίζεται με τη βοήθεια ενός ομάδας από φωτογραφικές μάσκες, οι οποίες επιτρέπουν την παραγωγή λεπτών προστατευτικών νησίδων πάνω στο υπόστρωμα.

Αυτές οι νησίδες προστατεύουν τμήματα της επιφάνειας του πυρίτιου από τις διάφορες χημικές διεργασίες που σχηματίζουν τα τρανζίστορ. Τα τρανζίστορ μπορούν να γίνουν μικρότερα μικραίνοντας απλώς τις διαστάσεις των νησίδων. Μεγάλου αριθμού τρανζίστορ κατασκευάζονται στην ίδια επιφάνεια, και στη συνέχεια συνδέονται μεταξύ τους με μεταλλικές συνδέσεις για να αποτελέσουν ένα πλήρες κύκλωμα. Τα τρανζίστορ είναι τα δομικά στοιχεία από τα οποία μπορεί να κατασκευαστεί ένας ολοκληρωμένος υπολογιστής.

Μόλις ενωθεί ένας μεγάλος αριθμός από αυτά στο σωστό σχηματισμό, μπορούν να εκτελούν περίπλοκες λογικές και αριθμητικές πράξεις.

Η συγκεκριμένη πράξη που εκτελείται (όπως αφαίρεση ή πρόσθεση) καθορίζεται από το σχηματισμό των συνδέσεων μεταξύ των ομάδων των τρανζίστορ. Ομάδες από τρανζίστορ μπορούν ακόμη να χρησιμοποιούνται σαν διακόπτες για να κατευθύνουν δεδομένα από το ένα τμήμα του τσίπ στα άλλα, ακριβώς όπως τα κλειδιά αλλάζουν την



πορεία του τρένου πάνω στις γραμμές. Μερικά τρανζίστορ μαζί  
μιμούνται μια λογική πράξη ή μεταφέρουν δεδομένα από το ένα  
σημείο του τσίπ στο άλλο.

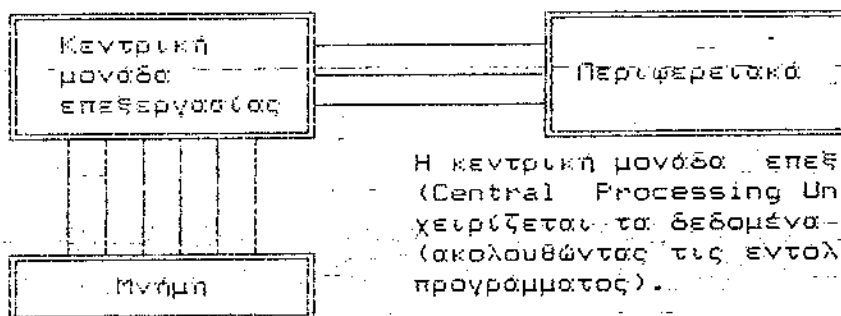
Δύο ή τρεις χιλιάδες τρανζίστορ φτιάχνουν ένα μικρό υπολογιστή,  
ενώ ένα εκατομμύριο φτιάχνει ένα πολύ ισχυρό υπολογιστικό  
σύστημα. Ο πιο κοινός τύπος τρανζίστορ που χρησιμοποιείται στα  
ολοκληρωμένα κυκλώματα είναι το τρανζίστορ MOS (metal oxide  
semiconductor, -- ημιαγωγός μετάλλου-διοξειδίου), το οποίο  
στηρίζεται σε ένα ηλεκτρικό πεδίο για να ελέγχει τη ροή του  
ρεύματος.

Ενας άλλος τύπος τρανζίστορ που χρησιμοποιείται στα ψηφιακά  
ολοκληρωμένα κυκλώματα είναι το CMOS (συμπληρωματικός ημιαγωγός  
μετάλλου-οξειδίου), με ιδιαίτερη εφαρμογή σε τρανζίστορ με  
μικρή ισχύ.

## H A R D W A R E

Υλικό (Hardware) είναι το όνομα που δόθηκε στα φυσικά μέρη του υπολογιστή, σε αντίθεση με το λογισμικό (software), τα προγράμματα που ελέγχουν τη λειτουργία της μηχανής. Η απλούστερη απεικόνιση του υλικού ενός υπολογιστή είναι σαν μια ομάδα συνδεδεμένων συσκευών όπου το καθένα εκτελεί μια ξεχωριστή εργασία. Οι τρεις συσκευές που αποτελούν τον ηλεκτρονικό υπολογιστή είναι:

Ο επεξεργαστής, η μνήμη και τα περιφερειακά



Η κεντρική μονάδα επεξεργασίας (Central Processing Unit, CPU) χειρίζεται τα δεδομένα που έχει (ακολουθώντας τις εντολές ενός προγράμματος).

Η μνήμη αποθηκεύει τις εντολές, καθώς και τις επεξεργαζόμενες πληροφορίες, τόσο πριν, όσο και μετά την επεξεργασία τους.

Τα περιφερειακά, όπως είναι οι οθόνες, οι εκτυπωτές και οι μονάδες δίσκων, είναι συσκευές που συνδέουν τον υπολογιστή και μνήμη με τον εξωτερικό κόσμο.

Η κεντρική μονάδα επεξεργασίας (CPU) χειρίζεται τις πληροφορίες, πάνω στις οποίες δουλεύει ένα πρόγραμμα. Για να το κάνει αυτό, αντιγράφει το πρόγραμμα και τα δεδομένα από την μνήμη. Η CPU μπορεί να είναι ένα μονάχα ολοκληρωμένο κύκλωμα, όπως είναι ο μικροεπεξεργαστής, ή να είναι κατασκευασμένη από πολλά κυκλώματα, όπως στα μεγάλα συστήματα υπολογιστών (mainframes).

Η μνήμη είναι φτιαγμένη από κελλιά, τα οποία αποθηκεύουν τόσο το πρόγραμμα όσο και τις πληροφορίες που επεξεργάζεται το πρόγραμμα.

Το τμήμα εισόδου/εξόδου (input/output section, I/O) του υπολογιστή συνδέει τη CPU με τα περιφερειακά, συσκευές που δεν αποτελούν αναπόσπαστο μέρος του υπολογιστή.

Για παράδειγμα, δίνουν στον υπολογιστή τη δυνατότητα να εκτυπώνει ή να εμφανίζει στην οθόνη πληροφορίες, και να δέχεται δεδομένα από δίσκους ή αισθητήρες. Αυτού οι αισθητήρες θα μπορούσαν να μετράνε το χρόνο, τη θερμοκρασία ή την πίεση, προμηθεύοντας πληροφορίες για να τις χειριστεί ο υπολογιστής.

Ο δίαυλος του υπολογιστή (computer bus) είναι το σύνολο των γραμμών ελέγχου που συνδέονται διάφορα τμήματα του υπολογιστή, δίνοντας τους τη δυνατότητα να μεταφέρουν πληροφορίες και να συγχρονίζονται μεταξύ τους. Οι κάρτες Interface (interface boards) είναι κάρτες κυκλωμάτων που μεταφέρουν τα ηλεκτρικά σήματα πάνω στον δίαυλο, και να συνδέουν τον υπολογιστή με τα περιφερειακά.

## Η ΚΕΝΤΡΙΚΗ ΜΟΝΑΔΑ ΕΠΕΞΕΡΓΑΣΙΑΣ (CPU)

Η κεντρική μονάδα επεξεργασίας είναι η καρδιά του υπολογιστή. Στοιχείο προς στοιχείο, το πρόγραμμα και οι πληροφορίες που θα επεξεργαστεί μεταφέρονται από τη μνήμη για να δεχτούν επεξεργασία από την CPU. Από φυσική άποψη η CPU είναι κατασκευασμένη από πολλές χιλιάδες τρανζίστορ τα οποία συνδέονται μεταξύ τους, το καθένα από τα οποία μπορεί να βρίσκεται είτε στην κατάσταση "εγτός" ("on") είτε στην κατάσταση "εκτός" ("off", δεν επιτρέπει τη διέλευση ηλεκτρικού ρεύματος).

Οι διάφορες εντολές της CPU αντιστοιχούν σε μικρές ομάδες από τρανζίστορ που βρίσκονται σε διαφορετικές θέσεις μέσα στο τσίπ. Η κάθε ομάδα μπορεί να εκτελεί μια σταθερή εντολή πάνω σε δεδομένα που δίνονται στην είσοδο. Η αριθμητική και λογική μονάδα (Arithmetic and Logic Unit, ALU) της CPU εκτελεί τις αριθμητικές ή λογικές πράξεις πάνω στις πληροφορίες. Όλες οι εντολές μπορούν να σχηματίζονται από μικρότερες υποεντολές οι οποίες συνδέονται μεταξύ τους μέσα στον υπολογιστή. Η καθένα από τις εντολές της CPU διαρκεί τουλάχιστον ένα, και συνήθως περισσότερους, κύκλους του ρολογιού του υπολογιστή, και ένας μικροεπεξεργαστής μπορεί να εκτελέσει περίπου ένα εκατομμύριο παρόμοιες πράξεις ανά δευτερόλεπτο.

Τόσο οι εντολές όσο και τα δεδομένα φυλάγονται μέσα στη μνήμη, και για να εκτελέσει μια εντολή η CPU πρώτα αντιγράφει το πρώτο κελλάκι της μνήμης του προγράμματος, και να καθορίζει τι είδους πράξη θα εκτελεστεί.

Αν πρόκειται να χρησιμοποιηθούν δεδομένα από αυτή την εντολή, η CPU τα αντιγράφει κι αυτά από τη μνήμη, και στη συνέχεια συνδέει τα δεδομένα με την εντολή αλλάζοντας την κατάσταση των τρανζίστορ κατά κάποιο τρόπο όπως γίνεται στις διασταυρώσεις, τρένων έτσι ώστε να υπάρχει μια συνεχής ηλεκτρική ροή ανάμεσα στα δεδομένα και στην ομάδα των εντολών που θα τα επεξεργαστεί.

Η πράξη εκτελείται πάνω στα δεδομένα, και στη συνέχεια το αποτέλεσμα στέλνεται στον προορισμό του, χρησιμοποιώντας άλλη μια συνεχή διαδρομή που δημιουργείται από τις ίδιες τεχνικές αλλαγής καταστάσεων. Ο προορισμός των επεξεργασμένων δεδομένων επιλέγεται από τον ίδιο το πρόγραμμα του υπολογιστή. Το σύνολο των εντολών εκτελεί μια απλή εργασία, και έτσι σχηματίζεται ένα πρόγραμμα.

Η ακολουθία των εντολών μηχανής που σχηματίζουν ένα πρόγραμμα έχουν επιλεγεί απ' ευθείας είτε από τον προγραμματιστή είτε από τον μεταγλωττιστή (compiler), ο οποίος βρίσκεται μέσα στον υπολογιστή, και μεταφράζει ένα πρόγραμμα γλώσσας υψηλού επιπέδου σε εντολές μηχανής.

## Ο ΣΕΙΡΙΑΚΟΣ ΕΠΕΞΕΡΓΑΣΤΗΣ

Ο υπολογιστής ονομάζεται σειριακός επεξεργαστής γιατί υπακούει σε μια εντολή κάθε φορά. Το μόνο που γνωρίζει το μηχάνημα είναι να εκτελεί την τρέχουσα εντολή που έχει πάρει από τη μνήμη του, μια απλή λειτουργία σε ένα πρόγραμμα που περιέχει πολλές χιλιάδες παρόμοιες.

Πριν τρέξει σε ένα μηχάνημα κάποιο πρόγραμμα γραμμένο σε γλώσσα υψηλού επιπέδου, σαν την BASIC, πρέπει να μεταφραστεί στις εντολές που διαθέτει ο συγκεκριμένος υπολογιστής.

Ενας τυπικός υπολογιστής έχει ένα ρεπερτόριο από 50 έως 500 διαφορετικές βασικές λειτουργίες, οι οποίες αποτελούν όλες μαζί το σετ εντολών. Το σετ εντολών που διαθέτει μια CPU θα ποικίλλει ανάλογα με τον κατασκευαστή, εκτός κι αν μια σειρά CPU έχουν σχεδιαστεί ειδικά ώστε να είναι συμβατές.

Η γλώσσα assembly είναι μια συμβολική αναπαράσταση των εντολών μηχανής ενός υπολογιστή, σχεδιασμένη για να κάνει ευκολότερο το γράψιμο των προγραμμάτων. Η κάθε εντολή παριστάνεται με ένα συμβολικό όνομα (όπως είναι το ADD για την πρόσθεση ή το SUB για την αφαίρεση) που σαν σκοπό του έχει να δείχνει την δράση αυτής της εντολής. Αυτές οι συμβολικές εντολές μεταφράζονται από ένα πρόγραμμα στις αντίστοιχες εντολές μηχανής.

Κανονικά τα προγράμματα σε γλώσσα assembly δεν μπορούν να χρησιμοποιηθούν σε διαφορετικό μηχάνημα.

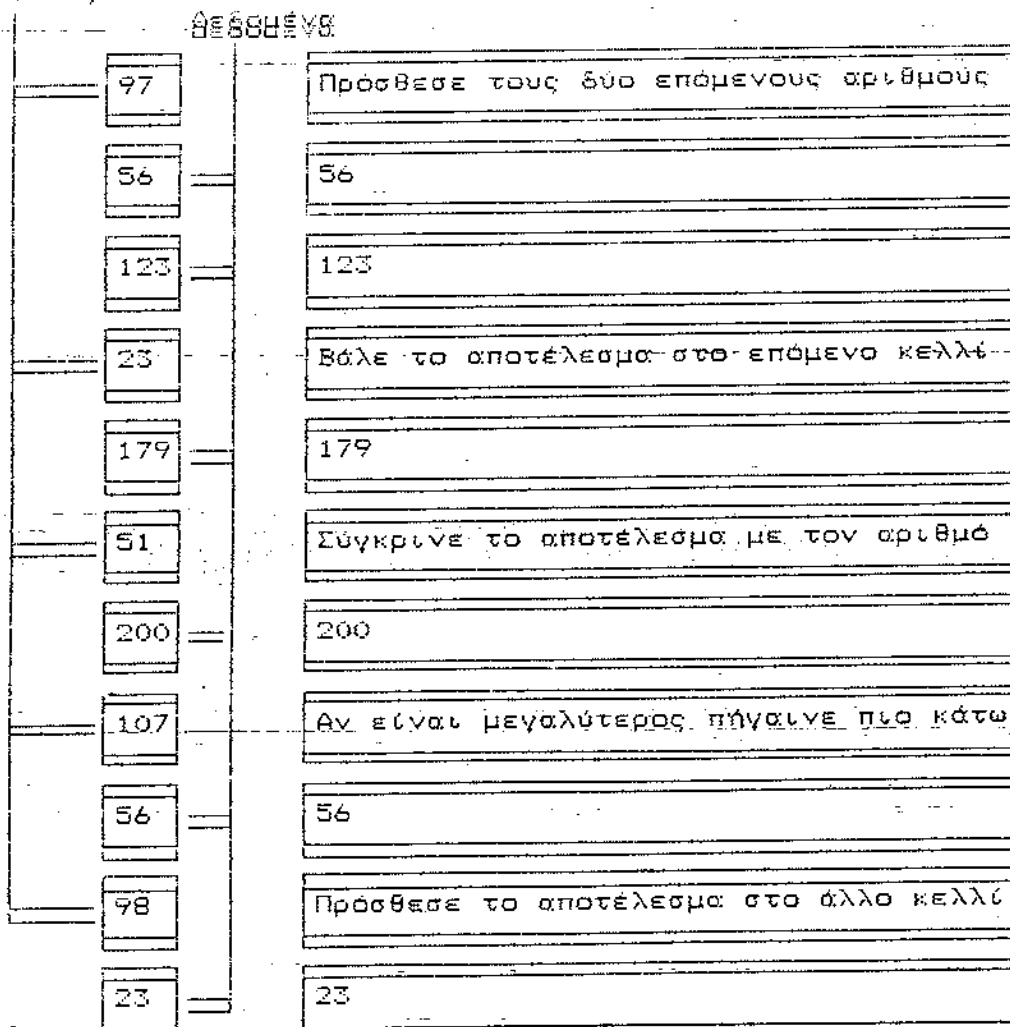
Μικρό τμήμα ενός προγράμματος  
στη μνήμη του υπολογιστή

Το καθένα από τα κελιά της μνήμης ενός υπολογιστή περιέχει έναν αριθμό, ο οποίος μπορεί να περιστήνει είτε μία εντολή για εκτέλεση είτε την τιμή ενός δεδομένου. Στο αριστερό μέρος υπάρχει μια στήλη με έντακα κελιά της μνήμης, ενώ η κεντρική στήλη περιγράφει τι σημαίνει ο αριθμός σε κάθε κελλί. Οι αριθμοί αυτοί μαζί σχηματίζουν ένα μικρό τμήμα κάποιου προγράμματος. Στο δεξιό μέρος, η CPU φαίνεται να διαβάζει ένα κελλί της μνήμης που περιέχει τον αριθμό 107, ο οποίος περιστήνει μία εντολή. Η CPU έχει ήδη επεξεργαστεί τις τρεις πρώτες εντολές του υποπρογράμματος. Επίσης έχει διαβάσει τρεις τιμές δεδομένων, και έχει γράψει στη μνήμη μια καινούργια.

Κανονικά ο υπολογιστής επεξεργάζεται τα κελιά της μνήμης το ένα μετά το άλλο, αλλά η εντολή σύγκρισης θα μπορούσε να υποχρεώσει τη CPU να πηδήξει σε κάποιο άλλο τμήμα της μνήμης για εντολές.

Εντολές

ΔΕΔΟΜΕΝΟ



Οι εντολές χαμηλού επιπέδου που εκτελούνται από την CPU είναι πάντοτε αθέατες στον προγραμματιστή που χρησιμοποιεί γλώσσες υψηλού επιπέδου σαν την BASIC ή την FORTRAN, ο οποίος μπορεί να κάνει χρήση περισσότερο αφηρημένων λειτουργιών που προσφέρονται από αυτές τις γλώσσες.

Χρησιμοποιώντας αυτές τις γλώσσες, ο προγραμματιστής φτιάχνει μια προδύσγραφη του τι θέλει να κάνει το πρόγραμμα, και αυτή στη συνέχεια μεταφράζεται από ένα πρόγραμμα μεταγροφής στις ειδικές εντολές που μπορεί να εκτελέσει ο συγκεκριμένος υπολογιστής.

Αν μια γλώσσα υπολογιστών χρησιμοποιείται από όλους και είναι τυποποιημένη, τότε πολλοί κατασκευαστές θα κυκλοφορήσουν μεταγροφείς για αυτή τη γλώσσα, οπότε το ίδιο πρόγραμμα μπορεί να μεταφραστεί για πολλά μηχανήματα.

Το πρόγραμμα σε γλώσσα υψηλού επιπέδου θα παραγάγει σε κάθε μηχανήμα τα ίδια αποτελέσματα, ακόμη κι αν οι εντολές μηχανής που θα χρησιμοποιούνται είναι διαφορετικές.

## Κ Α Τ Α Χ Ω Ρ Η Τ Ε Σ

Η CPU διαθέτει πολλούς τύπους καταχωρητών, στους οποίους αποθηκεύει ενδιάμεσα αποτελέσματα ή δεδομένα που μπορεί να χρειάζεται συχνά. Οι εντολές που χρησιμοποιούν αυτή την εσωτερική μνήμη είναι ταχύτερες από αυτές που χρειάζεται να πηγαίνουν για πληροφορίες στην εξωτερική μνήμη ημιαγωγών, γιατί δεν υπάρχει η καθυστέρηση για την ανάγνωση ή το γράψιμο πληροφοριών μεταξύ της CPU και της μνήμης.

Ο περισσότεροι καταχωρητές μπορούν να χρησιμοποιούνται και από τον προγραμματιστή, όπως συμβαίνει στην Turbo Pascal, αλλά οι καταχωρητές "μετρητής προγράμματος" (program counter) και οι "δείκτης στοίβας" (stack pointer) συνήθως αλλάζουν περιεχόμενα αυτόματα από τον υπολογιστή ή το λειτουργικό σύστημα.

Ο προγραμματιστής μπορεί να εξετάζει τις τιμές σε αυτούς τους καταχωρητές για να κάνει διάγνωση προβλημάτων, αλλά δεν μπορεί να τις αλλάξει.

## ΔΙΑΚΟΠΕΣ

Οι διακοπές (interrupts) δίνουν στον υπολογιστή τη δυνατότητα να ελέγχει πολλά προγράμματα ταυτόχρονα. Ένα καλό παράδειγμα ανάλογο της διακοπής είναι η τηλεφωνική κλήση. Δεν περιμένουμε εμείς να χτυπήσει το τηλέφωνο. Το τηλέφωνο κουδουνίζει, και εμείς σταματάμε αυτό που κάνουμε. Αν βρισκόμαστε στη μέση κάποιας πραγματικής σημαντικής εργασίας, αγνοούμε το τηλέφωνο.

Όμοια ο υπολογιστής θα ασχολείται με εργασίες μεγάλου μήκους, ενώ εξακολουθεί να είναι σε θέση να ανταποκρίνεται σε διακοπές, είτε είναι επείγουσες είτε όχι. Η CPU μπορεί να δουλεύει σε μια δουλειά της μιας ώρας, όταν ξαφνικά μια άλλη διάταξη της ζητάει να την προσέξει. Η διάταξη που διακόπτεται στέλνει σήμα στη CPU, και ο υπολογιστής ελέγχει για να διαπιστώσει αν η αίτηση έχει υψηλότερη προτεραιότητα από τη δουλειά που κάνει τώρα. Η "προτεραιότητα διακοπής" καθορίζει αν θα ενοχληθεί ή όχι η παρούσα εργασία. Αν η καινούργια δουλειά είναι αρκετά σημαντική ώστε να διακόψει την παρούσα εργασία, οι πληροφορίες που βρίσκονται στους καταχωρητές και στη μνήμη μεταφέρονται και διατηρούνται αλλού, είτε με αποθήκευση σε δίσκο είτε στη στοίβα.

Στη συνέχεια εκτελείται η καινούργια δουλειά, και κατόπιν η CPU επιστρέφει στην προηγούμενη, ξανατοποθετώντας τις αρχικές τιμές στους καταχωρητές και στη μνήμη, αφού τις βρεί προηγουμένως στο δίσκο ή στη στοίβα.

Υπάρχουν δύο τύποι διακοπής: χωρίς μάσκα (non maskable) και με μάσκα (maskable), οι οποίοι ελέγχονται από το hardware και το software αντίστοιχα. Οι διακοπές είναι ουσιαστικής σημασίας για την ομαλή λειτουργία υπολογιστικών συστημάτων με πολλούς χρήστες. Όταν έχουμε πολλούς χρήστες που πληκτρολογούν προγράμματα, ο καθένας χρήστης που κάθεται σε ένα τερματικό μπορεί να εισάγει πιθανώς ένα γράμμα ανά δευτερόλεπτο περίπου. Αν ήταν να αφιερωθεί ολόκληρος ο υπολογιστής σε ένα μονάχα άτομο, θα χρησιμοποιούταν ένα μικρό τμήμα μονάχα της συνολικής ισχύος. Αντί γι' αυτό, ο υπολογιστής ασχολείται με άλλες δουλειές, και μαθαίνει ότι τον χρειάζονται από μια διακοπή, ακόμη κι αν πρόκειται να ασχοληθεί μονάχα με την εισαγωγή ενός γράμματος. Οι υπολογιστές ποικίλλουν πάρα πολύ ως προς τον τρόπο οργάνωσης των διακοπών τους. Είναι δύσκολο να τοποθετηθούν οι διακοπές μέσα σε μια γλώσσα υψηλού επιπέδου, και έτσι συνήθως αντιμετωπίζονται σε επίπεδο λειτουργικού συστήματος. Παρ' όλο που αυτό σημαίνει ότι μένουν κρυμμένες από τους προγραμματιστές εκείνους που χρησιμοποιούν γλώσσες υψηλού επιπέδου μονάχα, εξακολουθούν να αποτελούν ένα ζωτικό στοιχείο στην αποτελεσματικότητα του υπολογιστή, ιδίως στους υπολογιστές ή τα συστήματα πολλών χρηστών που πρέπει να αντιδρούν σε γεγονότα του κόσμου που βρίσκεται έξω από τον υπολογιστή.

## Ο μικροεπεξεργαστής

Ο μικροεπεξεργαστής είναι ένας υπολογιστής που έχει φτιαχτεί πάνω σε ένα μοναχά τσίπ από πυρίτιο τετραγωνικού σχήματος πλευράς 4 έως 6 χιλιοστών, και πάχους 0,5 χιλιοστών. Ο μικρότερος υπολογιστής χρειάζεται τουλάχιστον μερικές χιλιάδες τρανζίστορ, και τσα είχαν τοποθετηθεί για πρώτη φορά σε ένα μοναχά τσίπ το 1971.

Ο πρώτος μικροεπεξεργαστής (ο Intel 4004) ήταν ένα 4-μπιτο τσίπ (δηλαδή χειρίζεται ταυτόχρονα 4 bit δεδομένων), με ταχύτητα ρολογιού 100.000 hertz. Στη δεκαετία που ακολούθησε την εμφάνιση του 4004, η επεξεργαστική ισχύς ενός τσίπ αυξήθηκε πολλές εκατοντάδες φορές. Οι μικροεπεξεργαστές (συνήθως αναφέρονται σαν "micro") έχουν γίνει τώρα εξαιρετικά παλύπλοκοι, και ο σχεδιασμός τους απαιτεί πολλά χρόνια προσπάθειών από ομάδες μηχανικών.

Τα περισσότερα από τα τσίπ που ονομάζονται μικροεπεξεργαστές, στην πραγματικότητα δεν είναι πλήρεις υπολογιστές, αλλά αποτελούν τις κεντρικές μονάδες επεξεργασίας των υπολογιστών. Τα τσίπ χρειάζονται πρόσθετη μνήμη, ένα ρολόι, και ευκολίες εισόδου/εξόδου για να έρχονται σε επαφή με τον εξωτερικό κόσμο.

Παράγονται και μικροεπεξεργαστές που είναι αυτόρκειοι μέσα σε ένα μόνο τσίπ - ο πρώτος, ο 8048, κατασκευάστηκε από την Intel το 1976 - αλλά τσίπ του είδους συνήθως προορίζονται να ελέγχουν απλές μηχανές με μια λειτουργία, όπως είναι τα πλυντήρια, ή οι μηχανές λήψης. Οι προσωπικοί υπολογιστές συνήθως κατασκευάζονται από τσίπ που είναι μοναχά CPU, γιατί είναι ευκολότερα να προστεθούν και άλλα τσίπ ή κάρτες σ' αυτά παρά σε τελείως αυτόρκειο τσίπ.

Επομένως ο όρος μικροεπεξεργαστής εφαρμόζεται σε κάθε τσίπ που περιέχει το σύνολο της CPU. Οι πρώτοι μικροεπεξεργαστές ήταν αρκετά απλοί ώστε οι εντολές τους να σχεδιάζονται απ'ευθείας μέσα στο τσίπ σαν συνδέσεις μεταξύ των τρανζίστορ του τσίπ. Η εξέταση των τσίπ αυτού του είδους δείχνει ένα σύνθετο δίκτυο συρμάτωσης, που φαίνεται να συδέει στην τύχη διαφορετικά τμήματά του τσίπ.

Σήμερα οι εντολές μηχανής των περισσότερων μικροεπεξεργαστών ελέγχονται από μια ενσωματωμένη "μνήμη μοναχά για διάβασμα" (ROM).

Αν υπάρχει κάποιο λάθος στη σχεδίαση, η τροποποίηση της ROM είναι μια σχετική απλή εργασία και είναι απίθανο να επηρεάσει άλλα τμήματα του μικροεπεξεργαστή.



Η οργάνωση των λειτουργιών ενός μικροεπεξεργαστή με τη ROM απαιτεί να χρησιμοποιηθεί μια περιοχή από σιλικόνη κατά 20 τοις εκατό περίπου μεγαλύτερη απ'όσο θα χρειαζόταν αν χρησιμοποιόνταν συνδέσεις μέσα στο τσίπ. Η διαδικασία σχεδίασης όμως, είναι πολύ πιο απλή, ενώ αργότερα μπορούν να προστεθούν στο μικροεπεξεργαστή επιπρόσθετες εντολές.

Εξαιτίας της γενικής εφαρμογής του, ο μικροεπεξεργαστής είναι πολύ πιο αργός, και το τσίπ μεγαλύτερο, από όσο θα ήταν ένα ολοκληρωμένο κύκλωμα σχεδιασμένο για μια συγκεκριμένη δουλειά.

Πάντως, είναι αρκετά γρήγορος για τις περισσότερες περιπτώσεις, και η ευελιξία του συνήθως μας αποζημιώνει για την απώλεια ταχύτητας. Ένα τσίπ μπορεί να προγραμματίζεται ώστε να εκτελεί πολλές και διαφορετικές εργασίες, αντί να χρειάζεται ένας καινούργιος σχεδιασμός κυκλώματος για το κάθε πρόβλημα.

Οι μικροεπεξεργαστές έχουν αλλάξει το ρόλο των ηλεκτρονικών μηχανικών, οι οποίοι προηγουμένως έδωσαν τον καιρό τους σχεδιάζοντας και συνδέοντας κάρτες χρησιμοποιώντας μερικές δεκάδες ολοκληρωμένα κυκλώματα. Κάθε φορά που χρειαζόταν ένα καινούργιο κύκλωμα, σχεδιάζόταν και κατασκευαζόταν από το μηδέν.

Τώρα, αντί να σχεδιάζει ένα κύκλωμα για κάποιο συγκεκριμένο πρόβλημα, ο μηχανικός γράφει ένα πρόγραμμα που θα υποχρεώνει ένα τυποποιημένο κύκλωμα να εκτελεί τη συγκεκριμένη εργασία - μια απόδειξη του γεγονότος ότι το Hardware και το Software είναι εναλλάξιμα σε μεγάλο βαθμό. Βέβαια, οι μικροεπεξεργαστές πρέπει να είναι σε θέση να επικοινωνούν με τον έξω κόσμο. Για παράδειγμα, αν πρόκειται να ελέγχουν την θερμοκρασία, πρέπει να μπορούν να διαβάσουν θερμομέτρα, και να ανοιγοκλείνουν βάνες.

Συνεπώς, τη περίπτωση μπορεί να χρειαζόταν να κατασκευαστεί μια μικρή ποσότητα από καινούργια ηλεκτρονικά κυκλώματα, αλλά η μεγάλη ποικιλία των τυποποιημένων κυκλωμάτων που είναι διαθέσιμα, μειώνει δραστικά την ανάγκη για σχεδιασμό και κατασκευή ειδικών κυκλωμάτων.

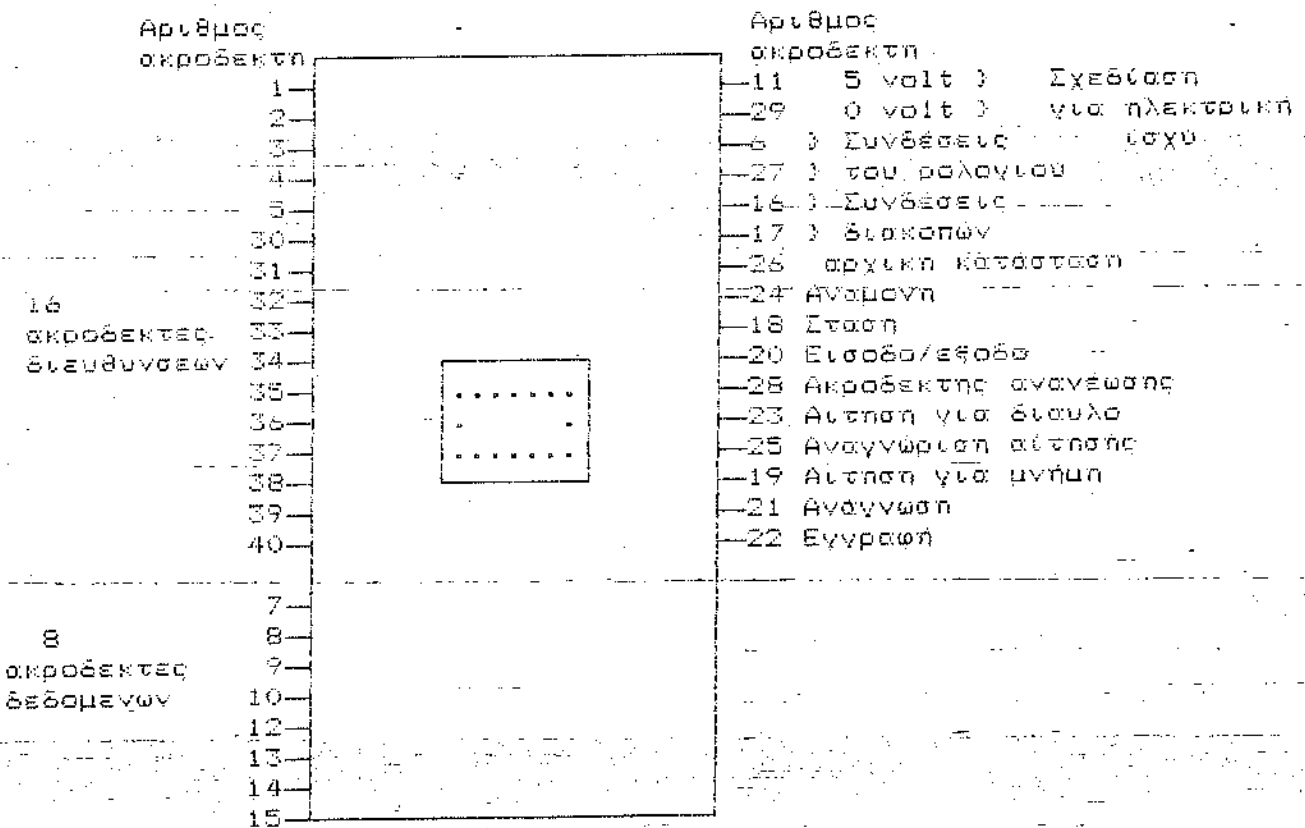
## Ο Zilog Z80

Ο Z80 είναι ένας από τους πιο δημοφιλείς μικροεπεξεργαστές που κατασκευάστηκαν ποτέ, και βρίσκεται στην καρδιά πολλών προώπικων υπολογιστών.

Ο Z80 αναπτύχθηκε αρχικά από την Zilog Corporation of Cupertino, της Καλιφόρνια, μια εταιρία που ιδρύθηκε από μια μικρή ομάδα μηχανικών οι οποίοι έφυγαν από την Intel το 1975. Τώρα το τσίπ κατασκευάζεται, όχι μόνο από τη Zilog που εξακολουθεί να το

Οι συνδέσεις των ακροδεκτών  
στον Z80

Ο Z80 είναι ένα τσίπ που χρησιμοποιεί 8 -μπιτα δεδομένα και 16 -μπιτες διευθύνσεις. Αυτό σημαίνει ότι εκτελεί όλες τις βασικές του εντολές σε 8 bit δεδομένων ταυτόχρονα, ενώ διαθέτει 16-bit για να απευθύνεται στη μνήμη τυχαίας προσπέλασης (RAM).



Ο κάθε συνδυασμός απο 1 και 0 στις γραμμές των διευθύνσεων επιτρέπει την προσπέλαση σε κάποιο κελλί της μνήμης του υπολογιστή και έτσι δίνει στο τσίπ τη δυνατότητα να έχει προσπέλαση σε 65536 bytes της μνήμης.

παράγει, αλλά και από μια δεκάδα άλλων εταιριών κατόπιν άδειας, καθώς και από αρκετές άλλες παρόνομα.

Η Zilog πήρε την έξυπνη απόφαση να κατασκευάσει ένα μικροεπεξεργαστή που θα μπορούσε να κάνει όλα όσα έκανε και ο ήδη δημοφιλής μικροεπεξεργαστής Intel 8080, κι ακόμη περισσότερα. Ο Z80 είναι συμβατό ως προς το Software με τον 8080 και 158 εντολές του Z80 περιλαμβάνουν το σύνολο των 78 εντολών του 8080.

Αυτό σημαίνει ότι Z80 μπορεί να τρέξει όλο το Software που έχει γραφτεί για τον 8080, αλλά το τσίπ του Z80 δεν μπορεί να τοποθετηθεί στην ίδια κάρτα με του 8080, γιατί οι συνδέσεις των ακροδεκτών είναι διαφορετικές. Η Zilog αντιλήφθηκε ότι η συμβατότητα στο Software είναι σημαντικότερη από τη συμβατότητα στο Hardware των προσωπικών υπολογιστών που χρησιμοποιούσαν το τσίπ 8080.

Αρχικά, η υψηλότερη ταχύτητα ρολογιού του Z80 ήταν παραπάνω από διπλάσια εκείνης του 8080, 2.5 MHz (2 1/2 εκατομμύρια κύκλοι ανά δευτερόλεπτο) έναντι 1 MHz. Εκτός από αυτή την αύξηση στην ταχύτητα, οι υπάρχουσες εφαρμογές μπορούσαν να επιταχυνθούν εκμεταλλευόμενες τις επιπλέον εντολές και τις βελτιωμένες δυνατότητες διακοπών του Z80. Αντίθετα με τον 8080, ο Z80 μπορεί να προγραμματιστεί ώστε να μετακινεί μεγάλες ομάδες δεδομένων με μια μονάχα εντολή. Ένα άλλο από τα πλεονεκτήματα του είναι ότι χρειαζόταν μονάχα μια παροχή τάσης (5 volt) ενώ το τσίπ της Intel χρειαζόταν τρεις (5, -5 και 12 volt). Το καινούργιο Software που γράφτηκε για να εκμεταλλευθεί αυτά βελτιωμένα χαρακτηριστικά δεν μπορεί χρησιμοποιηθεί με το τσίπ της Intel. Παρ'όλα που ο Z80 είναι τώρα από τεχνική άποψη απαρχαιωμένος, οι ταχύτερες εκδόσεις του (Z80A και Z80B), καθώς και η επένδυση στο υπάρχον Software, θα εξασφαλίσουν ότι ο βασικός του σχεδιασμός θα βρίσκεται μαζί μας για πολλά χρόνια ακόμα.

## 16-μπιτοι και 32-μπιτοι μικροεπεξεργαστές

Οι μεγαλύτερες εταιρίες μικροηλεκτρονικής, όπως είναι η Motorola, η National Semiconductor και η Fairchild, αντιδυσάστηκαν σε μεγάλο βαθμό από την εξέλιξη των 8-μπιτων μικροεπεξεργαστών.

Πολλοί πήραν σκληρά μαθήματα κατά την διάρκεια των πρώτων χρόνων του αγώνα για κυριαρχία στους μικροεπεξεργαστές. Η Motorola ήταν η πρώτη μεγάλη εταιρία που απάντησε στον 8080 της Intel και το 1974 κυκλοφόρησε τον μικροεπεξεργαστή 6800. Το τσίπ αποδείχθηκε παρόμοιο σε αποτελεσματικότητα με το τσίπ Intel 8080, αλλά δεν μπόρεσε ποτέ να αποσπασει το μεγάλο μέρος της αγοράς από τα συμβατά με την Intel τσίπ.

Η Motorola ήταν υπερβολικά καθυστερημένη, γιατί οι περισσότεροι πελάτες είχαν ήδη αγοράσει από την Intel ή την Zilog, και η αλλαγή θα τους κοστίζε υπερβολικά. Η δημιουργία ενός προϊόντος που θα κάνει χρήση ενός συγκεκριμένου μικροεπεξεργαστή χρειάζεται μεγάλες επενδύσεις τόσο σε εξοπλισμό όσο και σε πείρα, και μόλις επιλεγεί κάποια συγκεκριμένη οικογένεια μικροεπεξεργαστών, η εταιρεία θα προσπαθήσει να μείνει πιστή σ' αυτήν. Παρόλη τη καταπληκτική εμπορική επιτυχία των 8-μπιτων τσίπ, σύντομα φάνηκαν οι στελαιές τους. Συγκεκριμένα, ο περιορισμός της προσπέλασης της στη μνήμη μόνο με 16 Bit αποδείχθηκε ένα σοβαρό μειονέκτημα καθώς οι τιμές στις μνήμες έπεφταν και τα προγράμματα γίνονταν μεγαλύτερα. Το 1984, τα 64 kbytes μνήμης, τα οποία κόστιζαν χιλιάδες δολάρια στα μέσα της δεκαετίας του 1970, είχαν πέσει στα 25 δολάρια. Επιπρόσθετα το κόστος του ίδιου του επεξεργαστή έπεσε στα λίγα δολάρια.

Η Intel ήταν η πρώτη εταιρία που κατασκεύασε ένα εμπορικά επιτυχημένο 16-μπιτο τσίπ, το 8086, το οποίο τοποθετήθηκε σε πολλά καινούργια μηχανήματα. Ο 8086 και μία ελαφρά διαφορετική έκδοση, ο 8088 έγιναν εμπορικές επιτυχίες. Η Zilog μάλλον δεν τα πήγε καλά με το αντίστοιχό τους τσίπ, το Z8000, το οποίο δεν υπήρξε ποτέ πολύ δημοφιλές.

Τόσο ο 8086 όσο και ο Z8000 υποφέρουν από το ότι οι περιοχές της μνήμης στις οποίες μπορούν να απευθυνθούν είναι χωρισμένες σε ομάδες των 64 kbytes. Παρόλο που μπορούν να έχουν προσπέλαση σε πολλές τέτοιες ομάδες, το Software πρέπει να είναι δομημένο έτσι ώστε να μπορεί να αντιμετωπίζει αυτή τη διαμερισματοποίηση της μνήμης.

Η Motorola κατασκεύασε 16-μπιτο μικροεπεξεργαστή αργότερα από την Intel και την Zilog, ίσως και πόλι υπερβολικά αργά για τη μαζική αγορά, αλλά κατασκεύασε ένα προϊόν πραγματικά ανώτερο από τις πρώτες 16-μπιτες προσφορές των άλλων εταιριών την οικογένεια Motorola 68000.

## Η οικογένεια Motorola 68000

Ο μικροεπεξεργαστής Motorola 68000 είναι πολύ πιο ισχυρός από τον Z80. Αυτό δεν σημαίνει κατά κανένα τρόπο ότι το τσίπ της Zilog δεν είναι καλό, αλλά δείχνει τη γρήγορη πρόοδο της μικροηλεκτρονικής μέσα στα πάντα περίπου χρόνια που χωρίζουν τα σχεδιασμά των δύο τσίπ. Ο 68000 διαθέτει μια (λίγο-πολύ) 32-μπιτη εξωτερική κατασκευή, έχει 24 ακροδέκτες για την προσπέλαση στις διευθύνσεις της μνήμης, αλλά εξωτερικά χειρίζεται μονάχα 16 bit δεδομένων. Έτσι τα 32 bit δεδομένων χωρίζονται στα δύο για να μεταφερθούν σε εξωτερικές συσκευές. Το μεγάλο πλεονέκτημα του περιορισμού της εξωτερικής προσπέλασης σε διευθύνσεις και σε εύρη δεδομένων βρίσκεται στο ότι μειώνεται ο αριθμός των συνδέσεων των ακροδεκτών στο τσίπ. Η 24-μπιτη προσπέλαση στη μνήμη σημαίνει ότι το τσίπ μπορεί να έχει απευθείας προσπέλαση σε 16 megabytes μνήμης ημιαγωγών, που είναι περισσότερο από αρκετά για την πλειοψηφία των προγραμμάτων υπολογιστών. Πάντως, η οικογένεια 68000 έχει σχεδιαστεί κατά τέτοιο τρόπο ώστε να μπορούν να αναπτυχθούν μικροεπεξεργαστές με "προς τα πάνω συμβατότητα" οι οποίοι θα διαθέτουν 32-μπιτη εξωτερική προσπέλαση σε διευθύνσεις και δεδομένα.

Ο 68000 είναι ένα τρομακτικά πολύπλοκο τσίπ, κατασκευασμένο από 68000 τρανζίστορ. Παρουσιάζει πολλά πλεονεκτήματα απέναντι στον Z80. Κατά πρώτο, το αυξημένο εύρος δεδομένων (από 8 σε 32 bit) και το εύρος διευθύνσεων (από 16 σε 24 bit) κάνουν το τσίπ και ισχυρότερο και ευκολότερο στον προγραμματισμό. Περιέχει ενσωματωμένα χαρακτηριστικά για την αναγνώριση ορισμένων τυπικών λαθών στα προγράμματα. Για παράδειγμα, μια προσπάθεια διαίρεσης με το μηδέν σε κάποιο πρόγραμμα προξενεί μια αυτόματη προειδοποίηση από το τσίπ. Αυτό αυτομετωπίζεται από τον επεξεργαστή σαν ένας τύπος διακοπής. Ο 68000 περιέχει ακροδέκτες που επιτρέπουν σε άλλες συσκευές (όπως είναι οι μονάδες δίσκων) να παίρνουν προσωρινά τον έλεγχο του δίκου του υπολογιστή χρησιμοποιώντας τη μέθοδο της απευθείας προσπέλασης στη μνήμη.

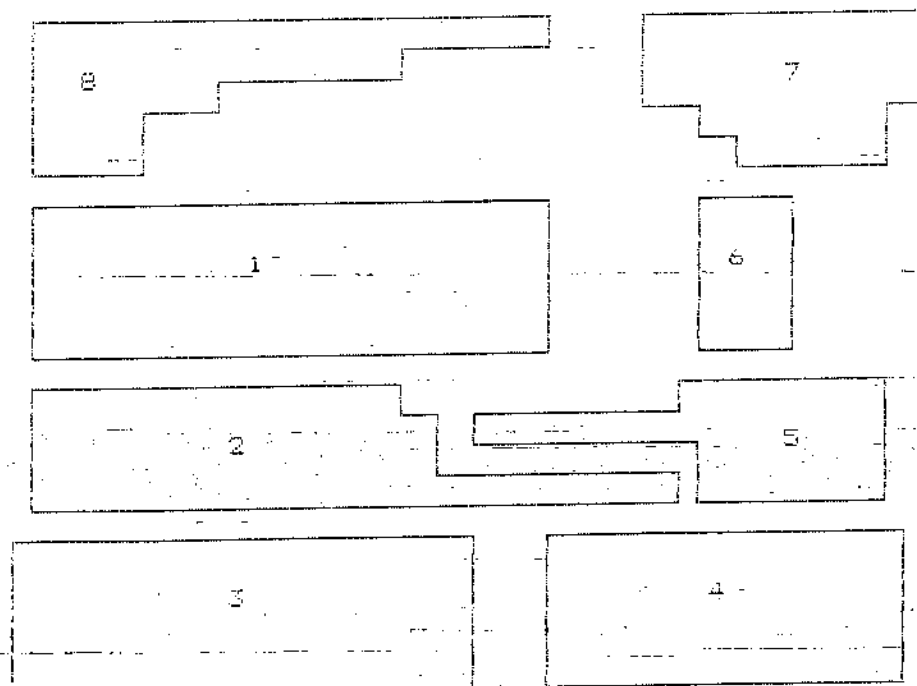
Ο 68000 διαθέτει μεγάλο φάσμα εντολών (γύρω στις 1000), στις οποίες συμπεριλαμβάνονται ορισμένες εντολές διαίρεσης και πολλαπλασιασμού. Παρά τα 32 bit, οι εντολές του τσίπ μπορούν να ενεργούν σε 16-μπιτα και 8-μπιτα δεδομένα, σε δεκαδικούς και σε απλά bit δεδομένων, παρακολουθώντας τη γκάμα των τύπων δεδομένων που τυπικά χρειάζονται στα προγράμματα. Η ταχύτερη εντολή χρειάζεται μονάχα 4 ωρολογιακούς κύκλους, δηλαδή το τσίπ μπορεί να εκτελεί γύρω στα δύο εκατομμύρια παρόμοιες εντολές το δευτερόλεπτο. Η πιο αργή εντολή (η διαίρεση των 32 bit) χρειάζεται 170 ωρολογιακούς κύκλους και επομένως το τσίπ μπορεί να εκτελεί γύρω στις 50.000 παρόμοιες το δευτερόλεπτο.

Επίσης ο 68000 "φέρνει προκαταβολικά" εντολές, δηλαδή μεταφέρει στη CPU το επόμενο τμήμα του προγράμματος, πριν χρειαστεί να εκτελεστεί.

Αυτό εξοικονομεί το χρόνο μεταφοράς από τη μνήμη και επιτρέπει τη γρηγορότερη εκτέλεση των εντολών.

Ο 68000 διαθέτει οχτώ καταχωρητές για δεδομένα, και εφτά για την προσπέλαση σε πληροφορίες. Από την άποψη του προγραμματιστή, οι καταχωρητές είναι πολύ πιο ευέλικτοι από εκείνους του Z80. Αντί να είμαστε υποχρεωμένοι συνεχώς να μεταφέρουμε δεδομένα προς και από μονάχα δύο καταχωρητές (τον Α και τον HL), όλοι οι καταχωρητές δεδομένων λειτουργούν ταυτόσημα, κάνοντας πολύ πιο εύκολο τον προγραμματισμό σε γλώσσα assembly.

Γραφική παράσταση του τσιπ Motorola 68000.



- 1 Μνήμη μονάχα-για-διάβασμα (ROM) για τις εντολές
- 2 Μονάδα ελέγχου του μικροεπεξεργαστή
- 3 Περιοχή εκτέλεσης διευθύνσεων
- 4 Περιοχή εκτέλεσης δεδομένων
- 5 Αριθμητική και λογική μονάδα
- 6 Παγίδευση και παράνομες εντολές
- 7 Λογική ελέγχου του βύθου
- 8 Βολαι και χρονομέτρηση

## S O F T W A R E

"Software" είναι το γενικό όνομα για τα προγράμματα των υπολογιστών, το καθένα από τα οποία εκτελεί μια συγκεκριμένη εργασία.

Το hardware του υπολογιστή έχει σημασία μονάχα σαν όχημα για το software και άσχετα με το πόσο ισχυρός είναι ένας υπολογιστής, είναι τόσο χρήσιμος όσο χρήσιμα είναι και το software που τρέχει σ' αυτόν. Το software δεν έχει μια συγκεκριμένη μορφή. Είναι πληροφορίες, προδιαγραφή για μια εργασία και από φυσική άποψη υπάρχει μόνο με την έννοια ότι πρέπει να αποθηκευτεί φυσικά στο hardware του υπολογιστή. Επιπλέον δεν υπάρχει συστήση διαχωριστική γραμμή ανάμεσα στο software και στο hardware, εφόσον μια εργασία που εκτελείται από hardware μπορεί να τη μιμηθεί το software και αντίστροφα.

Όταν οι προσωπικοί υπολογιστές εμφανίστηκαν στην αγορά στα τέλη της δεκαετίας του 1970, όλες οι ιδέες για software που είχαν αναπτυχθεί στα προηγούμενα είκοσι χρόνια πέρασαν μαζί σ' αυτούς, δημιουργώντας έως την εντύπωση ότι η παραγωγή νεωτεριστικών προγραμματιστικών ιδεών είναι μια απλή και φθηνή δουλειά.

Αυτό βέβαια δεν είναι αλήθεια και τώρα που η πλημμύρα του ξαναγραμμένου software από παλαιά προγράμματα κόπασε, μπορούμε να περιμένουμε ότι το βήμα του νεωτερισμού στο software θα επιβραδυνθεί, εκτός κι αν αναπτυχθούν νέες μέθοδοι προγραμματισμού.

Το software μπορεί να χωριστεί χονδρικά σε δύο περιοχές: στα προγράμματα εφαρμογών και στα προγράμματα του συστήματος.

Τα προγράμματα εφαρμογών εκτελούν τις εργασίες για τις οποίες αγοράζεται ένας υπολογιστής, ενώ τα προγράμματα του συστήματος είναι προγράμματα υποστήριξης που επιτρέπουν την πλήρη χρήση του υπολογιστή.

Στα προγράμματα εφαρμογών τα πιο γνωστά "πακέτα" είναι οι επεξεργαστές κειμένου, τα λογιστικά φύλλα και οι βάσεις δεδομένων. Στην συνέχεια θα δούμε έναν αντιπρόσωπο για το κάθε είδος και πιο συγκεκριμένα το LOTUS 1-2-3 για τα λογιστικά φύλλα, το dBASE IV για τις βάσεις δεδομένων και το Scriptum για τους επεξεργαστές κειμένου.

## ΕΠΕΞΕΡΓΑΣΤΕΣ ΚΕΙΜΕΝΟΥ

### SCRIPTUM

Με στόχο να εξυπηρετήσει τον Έλληνα χρήστη η εταιρία Memotek δημιούργησε ένα πολύ ισχυρό επεξεργαστή κειμένου, τον Scriptum.

Το SCRIPTUM τρέχει στο MS-DOS (έκδοση 3.0 ή μεταγενέστερη) και υποστηρίζει οθόνες Hercules, CGA, και EGA.

Προστατεύεται από αντιγραφή μέσω ενός παρεμβλήματος που τοποθετείται στην παράλληλη θύρα του υπολογιστή και επί του οποίου συνδέεται (μετά) το βύσμα του εκτυπωτή. Με αυτό τον τρόπο εξασφαλίζεται η νόμιμη χρήση του προγράμματος μια και η δισκέτα είναι άχρηστη χωρίς το παρέμβλημα.

Η εξυπηρέτηση που προσφέρει στον Έλληνα χρήστη φαίνεται από το σωστό "χωρισμό λέξεων" σύμφωνα με τους κανόνες της γραμματικής.

-Δεδομένου δε ότι και όλες οι εντολές και οδηγίες είναι Ελληνικές, μας απαλλάσσει από την ανάγκη "αλλαγών" μεταξύ των δύο αλφαβήτων και μας προσφέρει ένα καθαρά Ελληνικό περιβάλλον στη δουλειά μας.

### ΕΓΚΑΤΑΣΤΑΣΗ

Με το πρόγραμμα SETUPSCR ορίζουμε ορισμένες μόνιμες παραμέτρους για το πρόγραμμα, όπως το είδος του εκτυπωτή με τον οποίο θα συνεργαστεί, τα χρώματα της οθόνης εργασίας, τα περιθώρια και άλλα χαρακτηριστικά της σελίδας κειμένου.

Μπορούμε επίσης να επιλέξουμε την εργασία μετατροπής αρχείων, που είτε μετατρέπει ένα αρχείο από την format του SCRIPTUM σε ASCII, είτε μετατρέπει αρχεία άλλων γνωστών επεξεργαστών κειμένου σε μορφή αναγνώσιμη από το SCRIPTUM.

Η λειτουργία αυτή θα αποδειχθεί ιδιαίτερα χρήσιμη σε όσους θα θελήσουν να αρχίσουν να χρησιμοποιούν το SCRIPTUM, χωρίς όμως να χάνουν κείμενα που έχουν ήδη έτοιμα.



ΠΡΟΓΡΑΜΜΑ ΕΓΚΑΤΑΣΤΑΣΗΣ

1. Επιλογές εκτύπωσης
2. Χρώματα θραύσης
3. Παραμετροί
4. Μετατροπή αρχείων
5. Αποθήκευση & εξόδος
6. Εξόδος σε DOS

ΧΡΗΣΙΜΑ ΠΛΗΚΤΡΑ

- <-> - Μετακίνηση κατά γραμμή
- Pg... - Μετακίνηση κατά σελίδα
- Enter - Συνέχεια ή Αποθήκευση
- Esc - Εξόδος μενού
- F1 - Οδηγίες

SCRIPTUM

Το μενού του προγράμματος εγκατάστασης του SCRIPTUM.

## ΓΕΝΙΚΑ

Το SCRIPTUM διαθέτει 3 επιλογές στην οθόνη που εμφανίζει αμέσως μετά το "φόρτωμα" του, τις εξής:

1. Επεξεργασία κειμένου
2. Εκτύπωση κειμένου
3. Εξοδος σε DOS

Επιλέγοντας την πρώτη έχουμε στην διαθεσή μας όλες τις λειτουργίες του προγράμματος. Η δυνατότητα εκτύπωσης έχει απλώς τεθεί ΚΑΙ σαν ξεχωριστή επιλογή για την άνεση του χρήστη στην περίπτωση που θέλει να δώσει κείμενα (από τα υπάρχοντα) για εκτύπωση, χωρίς να προηγηθούν επεμβάσεις.

Πρώτη ερώτηση προς τον χρήστη είναι η κλασική για το όνομα του αρχείου. Εδώ είναι και η μοναδική στιγμή που απαιτείται μια κάποια ελάχιστη γνώση των κανόνων του DOS, γιατί πρέπει να δοθεί το πλήρες όνομα του αρχείου (drive, path, filename) όταν θέλουμε να δουλέψουμε σε άλλο εκτός από το current directory. Το πρόγραμμα επιτρέπει τη δημιουργία αρχείου ακόμα και χωρίς όνομα, μεταθέτοντας την επιλογή για άλλη στιγμή, πριν την αποθήκευση βέβαια.

Η οθόνη εργασίας του SCRIPTUM είναι εμπλουτισμένη με δύο πληροφοριακές γραμμές, μια στο πάνω μέρος και μια στο κάτω. Επανε βλέπουμε το όνομα του αρχείου την ώρα και τους αριθμούς σελίδας, γραμμής και στήλης που αντιστοιχούν στην τρέχουσα θέση του cursor. Ακριβώς από κάτω έχουμε μια γραμμή με σημειωμένα τα περίθωρα (αριστερά και δεξιά) και τα TABS (τις θέσεις στις οποίες "πηδά"ο cursor όταν πιέσουμε το ειδικό για αυτό το σκοπό πλήκτρο).

Η τελευταία γραμμή της οθόνης περιέχει την ένδειξη "F1. Μενού" για υπενθύμιση του τρόπου ενεργοποίησης των διαφόρων μενού με τις ειδικές λειτουργίες του SCRIPTUM, τον ελεύθερο χώρο σε ποσοστό και μια σειρά από ενδείξεις. Σε αυτές περιλαμβάνονται, εκτός από τα τρία κλασικά πλήκτρα (NUM, CAPINS), και ενδείξεις για την ενεργοποίηση ή όχι των περιθωρίων (PEP στην οθόνη) της στοίχισης (ΣΤΧ), του χωρισμού λέξεων (ΧΩΡ), του σχεδιασμού πλαισίων (ΣΧΔ) και της αριθμομηχανής (ΑΡΘ).

Η επιλογή των μενού του προγράμματος γίνεται με κίνηση του cursor (αφού βέβαια τα ενεργοποιήσουμε με το πλήκτρο F1). Οι συνήθεις λειτουργίες των μενού είναι δυνατόν να ενεργοποιηθούν και μέσα από το κείμενο με ορισμένους συνδυασμούς πλήκτρων, αυξάνοντας την ταχύτητα εργασίας μας. Γι αυτό το λόγο μάλιστα έχει περιληφθεί στο manual και μια ξεχωριστή σελίδα με κατάλογο των "συντομογραφιών" των εντολών.

Αρχαίο - Αναμόρφωση Παράμετροι - Μαρκόσιμα - Γενικά - Εκτύπωση

In	1. Ασφάλεια	F9
ba	2. Εξόδος	F10
ti	3. Παράθυρα	
DOS ve	4.	
files	1. Άνοιγμα	F2
consid	2. Κλείσιμο	F3
disk (	3. Επιλογή	F4
insert	4. Προηγούμενο	F5
will t	5. Μετάφωρα	F6
	Επιλογή :	

tion allows you to make an incremental backup (marked to be added). You will be asked to verify if you wish to make an incremental backup or not. This is the same as the "verify" option in the manual and it means that all files are checked for correctness. It slows things down considerably but it is a very popular option. You may be asked to insert a backup disk (of course). Don't worry about it. FLEXIBAK automatically checks this and if a new backup disk is required, then you will be asked to insert a blank disk. This disk should be formatted and completely blank although it may contain bad sectors and can be of any capacity. Should you not have any blank formatted disks, you can return to the main menu by pressing <ESCAPE> and either Exit or Shell Out in order to format another disk (see DOS menu section).

FLEXIBAK will not allow you to do an incremental backup while there are still any outstanding queries. So these should be answered using the Answer Queries option of the Queries menu prior to taking a backup.

Full Backup Option :

Δυνατότητα διαχείρισης παραθύρων με ανεξάρτητα κείμενα από το σχετικό μενού

Ας δούμε τώρα πιο αναλυτικά τις εντολές χειρισμού του SCRIPTUM οι οποίες ομαδοποιούνται στα παρακάτω 6 μενού :

Αρχεία  
Διαμόρφωση  
Παράμετροι  
Μαρκάρισμα  
Γενικά  
Εκτύπωση

---

### ΑΡΧΕΙΑ

---

Το μενού αυτό περιλαμβάνει τις εντολές:

1. Διαφύλοξη
2. Εξόδος
3. Παράθυρα
4. Εντολές DOS

Η πρώτη αποθηκεύει το κείμενο στο δίσκο, η δεύτερη τελειώνει το πρόγραμμα και επιστρέφει στο λειτουργικό (ζητώντας όνομα για το αρχείο, αν δεν το έχουμε δώσει από την αρχή) και η τέταρτη υποκαθιστά 5 εντολές του DOS (Αντιγραφή, Μετανομασία, Διαγραφή, Ευρετήριο και Αλλαγή δίσκου/καταλόγου) που καλύπτουν τις συνήθεις ανάγκες σε ό,τι τουλάχιστον αφορά στην επεξεργασία κειμένου.

Αφήσαμε για το τέλος την πιο χρήσιμη λειτουργία, τα παράθυρα, που οδηγεί σε ένα υπομενού με τις εξής επιλογές:

1. Άνοιγμα
2. Κλείσιμο
3. Επιλογή
4. Προηγούμενο
5. Μεταφορά

Με την πρώτη μας δίνεται η δυνατότητα να φέρουμε στην οθόνη μας ένα δεύτερο κείμενο, εκτός από αυτό που δουλεύουμε. Στην οθόνη διατηρείται η πρώτη γραμμή του αρχικού κειμένου υπενθυμίζοντάς μας ότι έχουμε και άλλο κείμενο για επεξεργασία. Μπορούμε να έχουμε μέχρι 6 αρχεία ταυτόχρονα με αυτόν τον τρόπο. Προφανώς με τις επιλογές 1-4 του παραπάνω μενού μας δίνεται η δυνατότητα επιλογής κάποιου κειμένου που βρίσκεται στην οθόνη, κλεισίματός του και μεταπήδησης στο προηγούμενο. Με την 5η λειτουργία μπορούμε να μεταφέρουμε ένα αρχείο στο τρέχον κείμενο της οθόνης.

---

### ΓΕΝΙΚΑ

---

Στις γενικές λειτουργίες του SCRIPTUM περιλαμβάνονται :

Αρχεία - Διαμόρφωση - Παράμετροι - Μάρκασημα - Γενικά - Εκτύπωση

1. Διαφυλάξη ~Q
2. Εξοδος F10
3. Παραθύρα
4. Εντολές DOS

1. Αντιγραφή αρχείων
2. Μεταφορά αρχείων
3. Διαγραφή αρχείων
4. Ξυστερηρία

Μασκα ευρετηριου : F:\EGGHI\Jimie\\*.\*

XEROX.TXT	MODEL70.TXT	H6.DIR
VX.BAT	INDEX.BAS	CS46.TXT

Πατήστε οποιοδήποτε πλήκτρο για συνέχεια

Απόψη των pull down menus του προγράμματος, Εδώ ευρετηρία αρχείων σε ένα directory.

1. Αναζήτηση λέξης
2. Σελιδοδείκτες..
3. Σελίδα
4. Σχεδίαση
5. Αρύθμομηχανή.

Η πρώτη λειτουργία περιλαμβάνει την απλή αναζήτηση με ταυτόχρονη αντικατάσταση και τη συνέχιση της αναζήτησης για την επομένη λέξη που ταιριάζει στην αναζητούμενη. Εκτός από τη λέξη που ψάχνουμε, το SCRIPTUM δέχεται και τα "κριτήρια" αναζήτησης που κάνουν τη λειτουργία αυτή πολύ πιο ευέλικτη.

Μπορούμε λοιπόν να καθορίσουμε:

- α) αν ζητάμε τη λέξη αυτούσια (π.χ. δώσαμε τη λέξη "όλο" και ζητάμε την ίδια ακριβώς ή και άλλες όπως "στόλο", "καθόλου" που την περιέχουν)
- β) αν έχει σημασία για την αναζήτηση η γραφή με κεφαλαία ή μικρά γράμματα
- γ) έναν αριθμό που καθορίζει αν θέλουμε να βρούμε τον 1ο, 2ο, 3ο, ... επανάληψη της ζητούμενης λέξης.

---

#### ΣΥΜΠΕΡΑΣΜΑ

---

Όπως από την αρχή δηλώθηκε, το SCRIPTUM δαύχνει με την πρώτη ματιά τις ανάλογες των διόσημων επεξεργαστών κειμένου δυνατότητες του. Προσφέρει τη δυνατότητα μετατροπής των αρχείων από αρκετούς εξ αυτών στο δικό του format, επιτρέποντας την χωρίς απώλειες αλλαγή προγράμματος. Διαθέτει την ικανότητα ταυτόχρονης επεξεργασίας 6 αρχείων και τέλος κοστίζει το λογικό ποσό των 16.000 δραχμών.

## ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

### dBASE IV

Πολλά άλλαξαν από την εμφάνιση της dBASE II πρὸν από 9 χρόνια περίπου. Η εξέλιξη αυτής της βάσης δεδομένων ακολουθεί κατά γράμμα την εξέλιξη της τεχνολογίας λογισμικού στις βάσεις δεδομένων. Έτσι μετά την dBASE II που εισήγαγε μια δικιά της γλώσσα και αποδέσμευσε τὸν προγραμματιστή από τη διαχείριση των αρχείων, ήρθε η dBASE III με μεγαλύτερες δυνατότητες. Η dBASE III αποτέλεσε ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών με μια καλή διαχείριση της οθόνης, εύκολη γεννήτρια εκτυπώσεων και μια πρώτη επαφή στις σχέσεις μεταξύ δύο αρχείων. Η dBASE III μπήκε στα νερά των σχεσιακών βάσεων με την υποστήριξη της SQL.

Η dBASE IV στοχεύει στην αγορά του DOS, μια οπωσδήποτε μεγάλη αγορά που δεν πάει όμως να περιορίζεται στους προσωπικούς υπολογιστές ή στα τυπικά δίκτυα. Δεν τρέχει σε κανένα λειτουργικό σύστημα και ενδεικνύεται στον προγραμματιστή που σκοπεύει να αναπτύξει εφαρμογές μόνο για προσωπικούς υπολογιστές.

Πολλές είναι οι βελτιώσεις ως πρὸς την προηγούμενη έκδοση, την dBASE III. Μερικά από τα πιο ενδιαφέροντα σύνολα εντολών περιλαμβάνουν τα εξής: Ένα πλήρες σύνολο εντολών για τον ορισμό, χρήση και αποθήκευση παραβήρων, ένα πλήρες σετ εντολών για τον ορισμό και χρήση μενού pull-down, μια εντολή BROWSE, εντολές για την επεξεργασία δυσδιάστατων πινάκων, είκοσι οκτώ νέες εντολές SET, νέο άμεσο χειρισμό της μορφής της ημερομηνίας και αρκετές άλλες.

Η μεγαλύτερη ίσως καινοτομία της dBASE IV είναι το κέντρο ελέγχου, κάτι ανάλογο με τον τρόπο επικοινωνίας με το χρήστη των μεγάλων συστημάτων DBMS. Το κέντρο ελέγχου (control centre) είναι για την dBASE IV ότι ήταν το Assist στην dBASE III-Plus.

Ενώ όμως το Assist δεν ήταν αρκετό, το κέντρο ελέγχου είναι ένα ισχυρό κέλυφος που επιτρέπει στον προγραμματιστή να δημιουργήσει μια εφαρμογή χωρίς να γράψει πρόγραμμα. Ένας non-procedural τρόπος ανάπτυξης εφαρμογών.

Αρκετά εύκολος στη χρήση είναι και ο σχεδιασμός οθονών. Το σύστημα που ζωγραφίζει την οθόνη, ο screen painter, δείχνει μόνο 19 γραμμές στην οθόνη. Υποστηρίζει όμως ολόκληρη την οθόνη και ακόμα οθόνες πολλών σελιδών. Μπορεί κανείς να προσθέσει εικόνες, εύρη (ranges), μηνύματα λάθους και συνθήκες που να ταυριάζουν με κάθε πεδίο. Μπορεί κανείς να ορίσει αναζητήσεις (query) σε μια οθόνη τύπου QBE, η οποία μπορεί να χρησιμοποιεί έναν ή

Στοιμιαστος απο την εγγραση σε μια κενη φορμα δεδομενων  
στην dBASE IV

CURSOR- <-- -->	UP DOWN	DELETE	Insert Mode : Ins
Char :	Field :	Char : Del	Exit/Save : ^End
Word : Home End	Page : PgUp PgDn	Field : ^Y	Abort : Esc
	Help : F1	Record: ^U	Memo : ^Home

LNAME  
FNAME  
ADDRESS  
CITY  
STATE  
ZIP

Κενή φόρμα Δεδομένων

CURSOR- <-- -->	UP DOWN	DELETE	Insert Mode : Ins
Char :	Field :	Char : Del	Exit/Save : ^End
Word : Home End	Page : PgUp PgDn	Field : ^Y	Abort : Esc
	Help : F1	Record: ^U	Memo : ^Home

LNAME  
FNAME  
ADDRESS  
CITY  
STATE  
ZIP

Ρυζος  
Νικος  
Κορινθου 123  
Πατρα  
GR  
91234

Πρώτη εγγραφή του MAIL



περισσότερους πίνακες μαζί. Τα query μπορούν να κατασκευαστούν χωρίς να γράψει ο προγραμματιστής ούτε μια γραμμή κώδικα. Αυτή είναι βέβαια η ομορφιά της μεθόδου QBE.

Η dBASE IV υποστηρίζει τη σχεσιακή γλώσσα SQL. Στην πραγματικότητα, αυτό που κάνει είναι να προσομοιώνει μια SQL χρησιμοποιώντας πίνακες δεδομένων της μορφής dBASE. Μπορεί να χρησιμοποιήσει ο προγραμματιστής ένα σύνολο εντολών SQL μέσα στην dBASE πληκτρολόγητας SET SQL ON. Αυτό απενεργοποιεί τις εντολές της dBASE, που αλληλεπιδρούν με τη γλώσσα SQL.

Η dBASE IV υποστηρίζει πολυχρηστικό περιβάλλον για τυπικά δίκτυα. Το κλείδωμα των αρχείων και των εγγράφων είναι καλύτερα σχεδιασμένα από την dBASE III και οι εντολές BEGIN TRANSACTION/ROLLBACK είναι απαραίτητες για ένα πολυχρηστικό περιβάλλον. Έτσι δεν χρειάζεται να δημιουργεί ο προγραμματιστής μικρούς πίνακες για κάθε συναλλαγή και μετά να πτόυς προσθέσει στους κανονικούς πίνακες όταν θα τελειώσει επιτυχώς μια συναλλαγή.

## ΛΟΓΙΣΤΙΚΑ ΦΥΛΛΑ

### LOTUS 1-2-3 RELEASE 3

Ένας ενθουσιώδης χρήστης του περίφημου spreadsheet Lotus 1-2-3, χαρακτήρισε τη νέα έκδοση, Release 3, σαν "μια ουσία που έχει γεύση σοκολάτας, θεραπεύει τον καρκίνο και κοστίζει μόνο ένα δολλάριο".

Το πρωί της 17ης Ιουνίου του 1989, περίπου 200 στελέχη της Lotus Development Corp. γιόρτασαν μαζί με τον πρόεδρο Jim Manzi, στα γραφεία της εταιρείας στο Cambridge της Μασσαχουσέτης, την ολοκλήρωση και διάθεση στην αγορά της νεότερης και πολυπόθητης Release 3 του πασίγνωστου προγράμματος spreadsheet Lotus 1-2-3.

Μετά από τρία χρόνια ανάπτυξης και μια σειρά από αδικαιολόγητες καθυστερήσεις, το νέο προϊόν άρχισε τελικά να διατίθεται στην αγορά, έχοντας τις δυνατότητες να υπερασπισθεί την κυριαρχία της Lotus στην αγορά των spreadsheet.

Το Lotus 1-2-3 Release 3 παρουσιάζει αλλαγές και προσθήκες σχεδόν σε όλες τις λειτουργίες του, σε σύγκριση με την Release 2, συμπεριλαμβανομένων των spreadsheet, των γραφικών, της database, των εκτυπώσεων και των macros. Θα μπορούσαμε όμως να πούμε ότι, παρά τις νέες εξελιγμένες δυνατότητες και παρά την μεγαλύτερη ευχρησία και φιλικότητα προς τον χρήστη, η Release 3 παραμένει κατά βάθος "Lotus 1-2-3".

### Η ΤΡΙΣΔΙΑΣΤΑΤΗ ΔΥΝΑΤΟΤΗΤΑ

Η σημαντικότερη ίσως δυνατότητα του Release 3 είναι η τρισδιάστατη λειτουργία των λογιστικών φύλλων εργασίας. Με αυτό τον τρόπο μπορούμε να έχουμε πολλά worksheets τα οποία να είναι

	A	B	C	D	E
1		Ισοτιμία Δολαρίου και Δράχμης το 1987			
2		1ο Δεκαήμερο	2ο Δεκαήμερο	3ο Δεκαήμερο	
3	Αυγούστος	108.6	110.0	111.6	
4	Σεπτέμβριος	110.5	110.8	110.9	
5	Οκτώβριος	110.3	112.0	112.8	
6	Νοεμβριος	111.0	112.0	113.0	
7					
8	Μεσος Όρος	110.1	111.2	112.7	
9					

	A	B	C	D	E
1		Ισοτιμία Δολαρίου και Δράχμης το 1987			
2		1ο Δεκαήμερο	2ο Δεκαήμερο	3ο Δεκαήμερο	
3	Αυγούστος	108.6	110.0	111.6	
4	Σεπτέμβριος	110.5	110.8	110.9	
5	Οκτώβριος	110.3	112.0	112.8	
6	Νοεμβριος	111.0	112.0	113.0	
7					
8	Μεσος Όρος	110.1	111.2	112.7	
9					

Το Release 3 εμφανίζει δυο συνεχόμενα worksheet τη φορά.  
 Το κλειδί window επιτρέπει τη μετακίνηση από το ένα worksheet  
 στο άλλο. Το κλειδί Zoom μαγεύθεται ένα window μέχρι να καταλαβεί  
 ολοκληρωτή την οθόνη.

ενεργά και εγκατεστημένα, συγχρόνως στη μνήμη.

Όταν θέλουμε να εργαστούμε σε ένα νέο worksheet, απλά "γυρίζουμε τη σελίδα" σε ένα κενό worksheet, ένα worksheet το οποίο μπορεί να έχει μέγιστο εύρος 256 στήλες και μέγιστο πλάτος 8192 σειρές, ενώ ένα και μόνο αρχείο μπορεί να περιέχει μέχρι 256 ενεργά worksheets.

### DATABASE

Οι μεγαλύτερες αλλαγές στη διαχείριση της database περιλαμβάνουν εξωτερική προσπέλαση της database, περισσότερα κλειδιά ταξινόμησης και καλύτερα database-reports. Για να συνδεθούμε με εξωτερικά database αρχεία θα χρειαστούμε την εντολή Data-External και τον καθορισμό ενός ονόματος στην on-disk-database που θέλουμε να χρησιμοποιήσουμε. Οι στατιστικές λειτουργίες της database μπορούν να αναφέρονται σε εξωτερικά database αρχεία και να επεξεργάζονται εξωτερικά στοιχεία σύμφωνα με διάφορα κριτήρια.

### ΠΡΟΓΡΑΜΜΑΤΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

Τα προγράμματα του συστήματος ενός υπολογιστή δίνουν βοήθεια για ένα πλήθος εργασιών που εκτελούνται στο σύστημα του υπολογιστή. Ένα από τα πιο σημαντικά βοηθήματα του υπολογιστή είναι το λειτουργικό σύστημα το οποίο θα εξετάσουμε αναλυτικότερα στην συνέχεια.

## Τ Ο Λ Ε Ι Τ Ο Υ Ρ Γ Ι Κ Ο Σ Υ Σ Τ Η Μ Α

Τι είναι το λειτουργικό σύστημα

Όταν μιλάμε για Η/Υ πρέπει πρωταρχικά να ξεκαθαρίσουμε κάτι. Αυτό το κάτι είναι ότι ο Η/Υ αποτελείται από δύο συνεργαζόμενα μέρη. Το Hardware και το Software. Το κύκλωμα και το πρόγραμμα. Το μόν Hardware (κύκλωμα) περιλαμβάνει ότι έχει σχέση με κύκλωμα και καλωδιώσεις.

Το Software (πρόγραμμα) περιλαμβάνει την λογική βάση της οποίας θα λειτουργήσει το Hardware. Είναι όμως κάπως δύσκολο να ξεχωρίσουμε αυτό που κάνει το Hardware από αυτό που κάνει Software.

Ο λόγος είναι ότι, κάποια λειτουργία μπορεί να υλοποιηθεί μόνο με Hardware ή μόνο με Software. Υπάρχει όμως και η λύση της συνεργασίας των δύο για την υλοποίηση της λειτουργίας.

Ετσι ποτέ δεν μπορούμε να πούμε ότι αυτή η λειτουργία είναι μόνο Hardware ή μόνο Software. Υπάρχει όμως το εξής. Χρειαζόμαστε ένα ελάχιστο (minimum) από Hardware για να υλοποιήσουμε με το Software κάποια απαίτηση. Ετσι οι δυνατότητες ενός Η/Υ είναι η συνισταμένη των δύο οντοτήτων, Hardware-Software.

Ετσι το Hardware επιδέχεται μέχρι ένα σημείο ρυθμίσεις, όπως επίσης και το Software. Όταν όμως αναφερόμαστε στο Software, πρέπει να ξεχωρίσουμε σε ποιο Software αναφερόμαστε. Αναφερόμαστε στο Standard Software του συστήματος (System software) ή στο Software των εφαρμογών.

Ο Η/Υ για να τεθεί σε λειτουργία χρειάζεται το minimum Hardware και το minimum Software του συστήματος. Από εκεί και πέρα οι παραπάνω δυνατότητες, κυρίως σε Software, αποτελούν δουλειά του προγραμματιστή εφαρμογών και αυτό αποτελεί όπως λέμε το Application. Μέσα σε αυτό το System-Software εμπεριέχονται όλες οι κανόνες βάσει των οποίων θα συμπεριφερθεί ο Η/Υ σε οποιαδήποτε απαίτησή μας. Δηλαδή αποτελεί το σύνταγμα βάσει του οποίου κινούνται οι πληροφορίες μεταξύ των επιμέρους μέρων, αλλά και σε σχέση με το περιβάλλον.

Τα λειτουργικά συστήματα που χρησιμοποιούνται σήμερα στο σύνολο των εμπορικών εφαρμογών είναι τρία :

- 1.) Το MS-DOS (χρησιμοποιείται από έναν χρήστη).
- 2.) Το UNIX (χρησιμοποιείται από πολλούς χρήστες).
- 3.) Το OS/2 (χρησιμοποιείται από πολλούς χρήστες).

## Τ Ο Λ Ε Ι Τ Ο Υ Ρ Γ Ι Κ Ο Σ Υ Σ Τ Η Μ Α MS-DOS

### 1. Γενικά

Το λειτουργικό σύστημα MS-DOS είναι σύστημα που αναπτύχθηκε από την εταιρεία Microsoft ειδικά για το PC της IBM.

Σήμερα όμως το λειτουργικό αυτό σύστημα έχει επικρατήσει στον χώρο των PC's, (υπολογίζεται ότι χρησιμοποιείται πάνω από 3.000.000 χρήστες σε όλο το κόσμο, τώρα μάλιστα με την εμφάνιση της Version 3.0 των Windows το MS-DOS θα επικρατήσει και αυτή την δεκαετία αυξάνοντας και τον αριθμό των ατόμων που το χρησιμοποιούν).

Πρίν γεννηθεί το MS-DOS υπήρχε στα PC το CP/M. Σήμερα δεν υπάρχει σχεδόν καθόλου το CP/M. Το λειτουργικό MS-DOS φτιάχθηκε, όπως ήδη είπαμε, για PC (Personal Computer = Προσωπικός υπολογιστής).

Ετσι εξυπηρετεί τις ανάγκες ενός και μόνο χρήστη. Στο λειτουργικό αυτό έχουμε αρκετή εξέλιξη από την αρχή μέχρι σήμερα (Version).

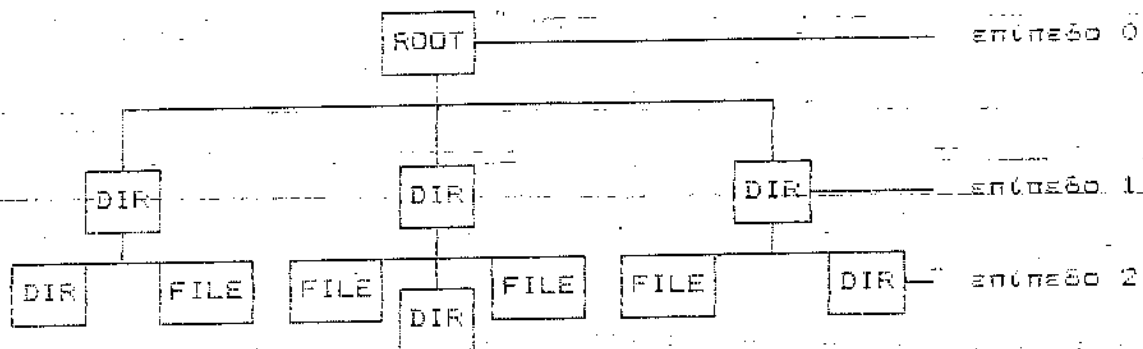
Εδώ περιγράφουμε την Version 3.2.

### 2. Διαχείριση των αρχείων

Στην διαχείριση των αρχείων του λειτουργικού αυτού παρατηρούμε ότι σε αρκετά σημεία μοιάζει με λειτουργικά μεγαλύτερων συστημάτων και συστημάτων Multiuser.

Ετσι η τεχνική του χωρισμού του όλου συστήματος αρχείων σε καταλόγους και υποκαταλόγους είναι κάτι βασικό εδώ.

Στο σχήμα φαίνεται η οργάνωση αυτή του συστήματος :



Στο σχήμα αυτό δείχνεται η οργάνωση των αρχείων. Η οργάνωση είναι μορφής δέντρου (tree structure). Ξεκινάμε από το ROOT που είναι η ρίζα του δέντρου και συνεχίζουμε προς τα κάτω στα κλαδιά του δέντρου. Τα κλαδιά του δέντρου μπορεί να είναι είτε Directories (κατάλογος) είτε Files. Εάν είναι Files, τότε αποτελούν τερματικό κλαδί του δέντρου. Εάν είναι Directories, τότε είναι κομβικό κλαδί που επιδέχεται επιπλέον ανάλυση. Στο σχήμα επίσης έχουμε χωρίσει τα επίπεδα του δέντρου. Το επίπεδο 0 είναι αυτό που βρίσκεται η ρίζα του δέντρου, Από εκεί και κάτω έχουμε όλα τα άλλα επίπεδα. Στην οργάνωση που φαίνεται έχουμε μέχρι το επίπεδο 2.

Πως δημιουργούμε ένα directory (mkdir-md)

Για να φτιάξουμε ένα directory χρησιμοποιούμε την εντολή mkdir (make directory). Η σύνταξη της εντολής είναι :

C > mkdir όνομα του directory

Το "C >" είναι η ένδειξη του συστήματος (prompt).

Πως διαγράφουμε ένα directory (rmdir-rd)

Η σύνταξη της εντολής είναι : C > rmdir όνομα του directory

Για να διαγράψουμε το directory π.χ. TEST το οποίο βρίσκεται κάτω από το directory JOE και αυτό κάτω από το directory USER, μπορούμε να το κάνουμε με δύο τρόπους.

Ο πρώτος τρόπος είναι δώσουμε ολόκληρο το pathname από όπου και εάν βρισκόμαστε :

C > rmdir \USER\JOE\TEST

η

rd

Με τον δεύτερο τρόπο θα πάμε πρώτα στο directory JOE και μετά θα δώσουμε : C > rmdir TEST.

Πως μπορούμε να αλλάξουμε ένα directory (chdir-cd)  
 Για να αλλάξουμε directory και να πάμε εκεί που θέλουμε θα  
 χρησιμοποιήσουμε την εντολή chdir. Η σύνταξη της εντολής είναι :

C > chdir pathname ή C > cd pathname

Όνόματα των αρχείων που χρησιμοποιούμε  
 Τα ονόματα των αρχείων που χρησιμοποιούμε, είναι της μορφής :

A B

Το Α μέρος πρέπει να είναι μέχρι 8 χαρακτήρες συνήθως γράμματα  
 και αριθμοί.

Μπορούμε να χρησιμοποιήσουμε και άλλα σύμβολα, αλλά δεν υπάρχει  
 ποτέ ο λόγος. Το Β μέρος είναι μέχρι 3 χαρακτήρες και δηλώνει την  
 μορφή του αρχείου. Εάν το αρχείο είναι πρόγραμμα cobol θα είναι  
 CBL, εάν είναι πρόγραμμα basic θα είναι BAS κ.ο.κ. Από τα πιο  
 συνηθισμένα extensions είναι :

.CBL	Πρόγραμμα	Cobol
.BAS	//	Basic
.FOR	//	Fortran
.PAS	//	Pascal
.C	//	C
.DAT	DATA	
.DOC	Κείμενο	
.EXE	Αρχείο εκτελέσιμο (Executable)	
.COM	Αρχείο του συστήματος	

### 3.) Οι εντολές του MS-DOS

Στις επόμενες σειρές θα γίνει παρουσίαση μερικών από των πιο  
 γνωστών εντολών του MS-DOS.

#### Break

Με την εντολή αυτή γίνεται επέκταση της ισχύος του  
 Control-C. Γνωρίζουμε ότι με Control-C μπορούμε να σταματήσουμε  
 κάποια διαδικασία που τρέχει εκείνη την στιγμή. Εάν όμως δώσουμε  
 πριν :

C > BREAK ON

τότε με Control-C μπορούμε να διακόψουμε οποιαδήποτε διαδικασία  
 τρέχει. Εάν όμως πούμε :BREAK OFF

Τότε μπορούμε να σταματήσουμε μόνο διαδικασίες που έχουν σχέση  
 με την οθόνη, το πληκτρολόγιο και τον εκτυπωτή. Η εντολή είναι  
 εσωτερική εντολή του MS-DOS και δουλεύει κάτω από δίκτυο.



#### Cls (Clear screen)

Με την εντολή αυτή καθαρίζει η οθόνη ότι και αν είχε και ο cursor πάει επάνω και αριστερά.

#### Copy

Με την εντολή αυτή γίνεται μεταφορά αρχείων. Η σύνταξη της εντολής είναι :

```
copy [drive:] pathname [drive:] pathname [/v] [/a] [/b]
```

#### Del (Delete)

Με την εντολή αυτή μπορούμε να διαγράψουμε αρχείο. Η σύνταξη της εντολής είναι :del [drive:] pathname

#### Dir (Directory)

Με την εντολή αυτή γίνεται εμφάνιση του directory. Η σύνταξη της είναι :dir [drive:] [pathname] [/p] [/w]

#### Format

Με την εντολή αυτή γίνεται προετοιμασία του φυσικού μέσου για εγγραφή στοιχείων. Η σύνταξη της εντολής είναι :

```
Format drive: [/1] [/4] [/8] [/b] [/v] [/s]
```

#### Print

Με την εντολή αυτή γίνεται εκτύπωση ενός αρχείου. Η σύνταξη της εντολής είναι :

```
print [drive:] [pathname] [/b:device] [/b:size] [/q:value] [/t] [/c] [/p]
```

#### Type

Με την εντολή αυτή γίνεται εμφάνιση του περιεχομένου ενός αρχείου. Η σύνταξη της εντολής είναι : type [drive:] filename

#### Backup

Με την εντολή αυτή γίνεται backup από σκληρό δίσκο σε δισκέτες. Η σύνταξη της εντολής είναι :

```
backup [drive:] [path] [filename] [drive:] [/s] [/m] [/a] [/d: mm-dd-yy]
```

#### Restore

Το command αυτό κάνει το αντίθετο από το backup. Κατεβάζει το backup. Η σύνταξη είναι :

```
Restore [drive:] [drive:] [path] [filename] [/s] [/p]
```

#### 4.) Batch Αρχεία

Τα batch αρχεία είναι αρχεία με εντολές του MS-DOS. Επιπλέον χρησιμοποιούνται και κάποιες εντολές των batch αρχείων για να φτιαχθεί ένα απλό πρόγραμμα με έλεγχο της ροής του. Ένα batch

αρχείο που έχει ιδιαίτερη σημασία είναι το AUTOEXEC.BAT. Το αρχείο αυτό είναι που ψάχνει να βρει το σύστημα όταν φορτώνεται. Έτσι αφού φορτωθεί το MS-DOS εκτελείται αμέσως μετά το AUTOEXEC.BAT. Το αρχείο αυτό είναι βασικό, γιατί μέσα από αυτό γίνεται η ενεργοποίηση των ελληνικών χαρακτήρων και διαφόρων άλλων λειτουργιών που πρέπει να τεθούν εξ' αρχής. Οι εντολές που χρησιμοποιούμε για τον έλεγχο της ροής του batch αρχείου είναι:

Echo, For, Goto, If, Pause, Rem, Shift.

Η εντολή Echo

Με την εντολή αυτή εμφανίζουμε στην οθόνη κάποιο μήνυμα.

Η σύνταξη της εντολής είναι:

echo [ON] ή echo [OFF] ή echo [message]

Η εντολή For

Με την εντολή αυτή γίνεται έλεγχος ροής του batch αρχείου.

Η σύνταξη της εντολής είναι:

for %% <c> in <set> do <command>

Η εντολή Goto

Με την εντολή αυτή γίνεται μεταφορά ελέγχου χωρίς συνθήκη.

Η σύνταξη της εντολής είναι:

goto label

Η εντολή If

Με την εντολή αυτή γίνεται εκτέλεση κάποιων εντολών όταν η συνθήκη συμβαίνει.

Η σύνταξη της εντολής είναι:

if condition command

Το condition μπορεί να είναι :

- 1.ERRORLEVEL αριθμός. Η συνθήκη αυτή είναι ο βαθμός λάθους που μπορεί να συμβεί σε προηγούμενη εντολή.
- 2.string1 == string2. Εδώ η συνθήκη συμβαίνει όταν το string 1 είναι ίδιο με το string2.
- 3.EXIST filename. Η συνθήκη συμβαίνει όταν το αρχείο που δίνουμε στο filename υπάρχει.
- 4.NOT condition. Η συνθήκη συμβαίνει όταν το condition δεν συμβαίνει.

Η εντολή Pause

Με την αυτή σταματά η εκτέλεση του batch αρχείου.

Η εντολή Rem

Με την εντολή αυτή βάζουμε σχόλια στο batch αρχείο.

Η εντολή Shift

Με την εντολή αυτή υπάρχει η δυνατότητα για χρήση παραμέτρων περισσότερων των 10.

## 5.) Φίλτρα

Φίλτρο είναι μια εντολή που παίρνει κάτι στην είσοδο εκτελεί κάποιον μετασχηματισμό πάνω σε αυτό και μετά το βγάζει σε κάποιο έξοδο. Τα φίλτρα στο MS-DOS είναι τρία. Η εντολή Find, η More και η Sort. Η εντολή Find αυτό που κάνει είναι να ψάχνει για κάποιο string μέσα σε ένα αρχείο. Η εντολή More εμφανίζει στην οθόνη κάποιο αρχείο σε μορφή μια οθόνη κάθε φορά. Η εντολή Sort ταξινομεί κάποιο αρχείο. Τα φίλτρα χρησιμοποιούνται σε συνδυασμό με άλλες εντολές.

### 6.) Piping (δυνατότητα μέσω σωλήνα)

Πολλές φορές το αποτέλεσμα κάποιας εντολής αντί να το στείλουμε στο standard output μπορούμε μέσω σωλήνα (pipe) να το πούμε σε επόμενη εντολή. Το σύμβολο του pipe είναι <|>.

Π.χ. dir | sort

Με την εντολή αυτή εκτελείται η εντολή dir δεν εμφανίζει όμως τίποτα στην οθόνη. Το αποτέλεσμα περνά στο φίλτρο sort. Έτσι η εμφάνιση θα γίνει μετά το sort. Επομένως θα εμφανισθεί το directory, αλλά ταξινομημένο.

## Το Λειτουργικό Σύστημα U N I X

Στο τέλος της δεκαετίας του 1960 στο M.I.T., έκανε την εμφάνισή του το λειτουργικό σύστημα Multics. Σαν ένα από τα νεότερα συστήματα χρονικού καταμέρισμού (time-sharing-system), το Multics είχε πολλές από τις ιδέες που εφαρμόζονται στα σημερινά multitasking συστήματα, ενώ υπέφερε από τους νεωτερισμούς του και ήταν περισσότερο σύνθετο και δύσκολο από όσο πραγματικά χρειαζόταν.

Την ίδια εποχή, η AT&T απέσυρε την συμμετοχή της από αυτό το project αφήνοντας μια ομάδα ταλαντούχων ανθρώπων να ανακαλύψουν μόνη της το πως θα έπρεπε να είναι ένα time sharing σύστημα. Η ομάδα αυτή, μη έχοντας τη δυνατότητα της πρόσβασης σε ένα μοντέρνο λειτουργικό σύστημα, αναγκάστηκε να δημιουργήσει ένα καινούργιο.

Με αρχικούς σχεδιαστές τους Ken Thompson και Dennis Ritchie, η ομάδα ενισχύεται με άλλους ικανούς επιστήμονες, όπως τους R.Morris και J.F.Ossana, ενώ παράλληλα αποκτά και ένα παρωπλισμένο DEC PDP-7. Σύντομα, η εργασία αυτών των ανθρώπων είχε σαν αποτέλεσμα τη γέννηση ενός νέου λειτουργικού συστήματος που ονομάστηκε UNIX και έτρεξε για πρώτη φορά στο PDP-7, στις αρχές του 1970.

Σήμερα, είκοσι χρόνια αργότερα, η ιστορία συνεχίζεται. Η εμφάνιση—μικροεπεξεργαστών—όπως οι Motorola 68020, Intel 80286, Intel 60386 στο χώρο των μικρών μηχανών, επεκτείνει την αγορά του UNIX. Με την υποστήριξη της AT&T, που έχει χαρακτηριστεί λόγω της πειροσύνης και της δύναμής της ως η μεγαλύτερη εταιρεία που έχει υπάρξει ποτέ στη γή, το UNIX κτυπά σήμερα όσο ποτέ άλλοτε την πόρτα κάθε επιχείρησης, κάθε γραφείου και ίσως, κάθε σπιτιού!

Το UNIX χαρακτηρίζεται μοναδικό στον κόσμο της πληροφορικής από το γεγονός ότι κατάφερε να προσελκύσει μαζί, πιστούς υπερασπιστές και δριμύτατους επικριτές.

Οι φίλοι του τείνουν να προάγουν την κομψότητα, τη δύναμη και την ευκαμψία του, ενώ οι εχθροί διαμορτώνονται για τη συντομογραφική γραφή των εντολών του, τη δυσκολία διαχείρισης και την απαιτησία του επιβελιστή.

Η AT&T με την τελευταία της έκδοση, System V Release 3 (SVR3 ή 5.3), προσπαθεί να κερδίσει τους αμύητους, (συνήθως αυτοί είναι οι εχθροί), των οποίων οι ανάγκες είχαν παραμεληθεί στις πρώτες εκδόσεις, μια και το UNIX είναι ένα σύστημα που αναπτύχθηκε από ειδικούς—για ειδικούς—. Στις επόμενες γραμμές θα γίνει μια παρουσίαση του UNIX και των λόγων της αποδοχής του και της μακροζωίας του.

## Τ Α Τ Μ Η Μ Α Τ Α Τ Ο Υ U N I X

Το λειτουργικό σύστημα UNIX αποτελείται από πολλά κομμάτια software. Μέσα σε αυτήν τη συλλογή των προγραμμάτων, το Shell και το Kernell κατέχουν μια ιδιαίτερη προνομιούχα θέση.

Το Kernell είναι το Supervisor του συστήματος που θα χειριστεί όλες τις επεξεργασίες και θα τις εκτελέσει με τον κατάλληλο τρόπο στον κατάλληλο χρόνο. Ποιά προγράμματα θα εκτελεστούν, πως θα γίνει αυτό, που θα πάνε τα outputs, είναι θέματα που απασχολούν το χρήστη και πρέπει να γνωστοποιηθούν στο σύστημα. Το Shell είναι αυτό που θα μας φέρει σε επαφή με το σύστημα.

Θα μπορούσαμε να παρομοιάσουμε το Shell με τη μεμβράνη που περιβάλλει ένα ζωντανό κύτταρο. Στο κύτταρο, η εξωτερική μεμβράνη είναι μια μάλλον "έξυπνη" οντότητα που ελέγχει τη ροή της τροφής και των κατάλληλων πληροφοριών (DNA) από και προς το κύτταρο, προστατεύοντας το και από την είσοδο βλαβερών ουσιών. Μέσα στο κύτταρο υπάρχει ο πυρήνας που φέρει τις οδηγίες για οτε χρειάζεται για να αναπτύξει τις δραστηριότητες του, καθώς και άλλα κυτταρικά οργανίδια που εκτελούν συγκεκριμένες εργασίες.

Στο UNIX, το Shell είναι η μεμβράνη που περιβάλλει utilities και kernell. Ο πυρήνας (kernell) περιέχει τα προγράμματα που δραστηριοποιούν το Hardware. Το Shell παίρνει τις πληροφορίες που δίνει ο χρήστης και αποφασίζει αν πρέπει να μπουν στο σύστημα, πη θα πάνε και τι να κάνουν.

## Τ Ο S H E L L

Το Shell είναι μια γλώσσα εντολών (Command Language) που ερμηνεύει τις εντολές και ενεργεί παρέχοντας επικοινωνία ανάμεσα στο χρήστη και τα ατομικά προγράμματα του συστήματος. Είναι αυτό που δίνει το prompt (\$ ή %) στην οθόνη, πληροφορώντας το χρήστη οτι το σύστημα είναι να δεχτεί εντολές. Ο χρήστης δίνει την εντολή και το shell ψάχνει να βρεί ένα πρόγραμμα -command- με αυτό το όνομα. Αν υπάρχει, το shell προτρέπει το kernell να το εκτελέσει και να επιστρέψει το αποτέλεσμα στο χρήστη, ενώ παράλληλα εμφανίζει στην οθόνη ένα νέο prompt.

Υπάρχουν δύο είδη shell εντολών, interactive και programmed. Όταν χρησιμοποιείτε το shell, είναι σαν να γράφετε προγράμματα μιας γραμμής. Με τη χρήση ενός interactive command, όπως pr, cd, mkdir, δίνετε την εντολή και παίρνετε αμέσως την απάντηση. Απλές interactive εντολές σε συνδυασμό με programmed εντολές του shell αποτελούν μια shell procedure που μπορεί να αποθηκευτεί στο δίσκο

σαν αρχείο και να εκτελεστεί οπότε χρειάζεται, χωρίς τη παρεμβολή του χρήστη.

Αφού το shell είναι ένα πρόγραμμα, το UNIX μπορεί να το τρέξει όπως θα έτρεχε κάθε άλλο πρόγραμμα. Κάθε φορά που κάνετε Login, το UNIX αντιγράφει το Shell τοποθετώντας το στο δικό σας directory. Αυτό το Shell που έρχεται με το Login ονομάζεται Login Shell είναι υπο-shell του κύριου shell.

Είναι δυνατόν να έχετε στο UNIX σας περισσότερα από ένα shell. Το "Bourne" shell και το "C" shell είναι τα πιο διαδεδομένα. Το "Bourne" shell γράφτηκε από τον Steve Bourne (Bell Labs, AT&T) το 1978 και θεωρείται σύνθετο, δανειζόμενο πολλά στοιχεία από την γλώσσα "Algol 68". Το "C" shell ή "UC Berkeley" shell γράφτηκε από τον Bill Joy στο Πανεπιστήμιο της Καλιφόρνιας στο Berkeley για την έκδοση BSD (Berkeley Software Distribution) και προσαρμόζεται προς τη γλώσσα "C". Συνήθως το βρίσκουμε σε νεότερες εκδόσεις του UNIX ενώ πολλές φορές έχουμε κάποιες προσθήκες από τον Jim Kulp (IIASA, Viena).

Μια νεότερη έκδοση του Shell είναι το "Korn" shell από τον David Korn (Bell Labs, AT&T).

## Τ Ο Κ Ε Ρ Ν Ε Λ Λ

Το Kernell είναι το κομμάτι του UNIX που ελέγχει θέματα που αφορούν το hardware, όπως : δίσκους, εκτυπωτές, γραμμές επικοινωνίας κ.λ.π. Αποτελείται από ένα κεντρικό supervisory πρόγραμμα και έναν αριθμό από ρουτίνες χαμηλού επιπέδου.

Η κάθε μια από αυτές είναι επιφορτισμένη με την εκτέλεση κάποιας συγκεκριμένης εργασίας όπως η ανάγνωση των χαρακτήρων από το πληκτρολόγιο, το γράψιμο στη μνήμη, η πρόσβαση στους δίσκους κ.λ.π.

Το kernell, παρόλο που θεωρείται ένα βασικό κομμάτι του UNIX, είναι πολύ μικρό σε σχέση με το ολόκληρο σύστημα. Ο χρήστης εισάγει εντολές που απευθύνονται στο shell το οποίο θα ενεργοποιήσει τις κατάλληλες ρουτίνες υπό την επίβλεψη του kernell. Η "εξωτερική" εικόνα που δίνει το UNIX στο χρήστη του φαίνεται να αποτελείται από δύο μέρη : ένα μεγάλο αριθμό προγραμμάτων (το καθένα αντιστοιχεί σε μια εντολή) και το shell, που τα υποστηρίζει και ελέγχει την εκτέλεσή τους. Με ανάλογο τρόπο η "εσωτερική" εικόνα που μπορεί να δώσει το UNIX χωρίζεται σε δύο μέρη : τις ρουτίνες που απευθύνονται χειρίζονται το hardware και το kernell που τις υποστηρίζει μετά από πρότροπή του shell.

Οι ρουτίνες αυτές (του kernell) καλούνται όταν χρειάζονται, ενώ το supervisory το kernell παραμένει μόνιμα στη μνήμη και είναι έτσι σχεδιασμένο ώστε να μην πιάνει χώρο που θα μπορούσε να διαθέτει για άλλες λειτουργίες.

Οι περισσότεροι χρήστες δεν ενδιαφέρονται για το τι πραγματικά συμβαίνει μέσα σε έναν Η/Υ. Το UNIX, υποστηρίζει αυτήν την παρόρμηση των χρηστών, με το να καλύπτει το kernel μέσω του shell.

Η "εσωτερική" εικόνα που προσφέρει το UNIX καλύπτει τη φυσική υπόσταση ενός Η/Υ από τους χρήστες που επιθυμούν πρόσβαση στο χαμηλό επίπεδο του συστήματος. Υπεύθυνα γι' αυτό είναι το kernel και το πετυχαίνει δημιουργώντας μια ιδεατή μηχανή (virtual machine) με χαρακτηριστικά που ταυρίζουν σε όλους σχεδόν τους Η/Υ. Η ιδεατή μηχανή που δημιουργεί το kernel χωρίζεται σε 3 κομμάτια που το καθένα προσφέρει ορισμένες βασικές λειτουργίες:

- Το πρώτο κομμάτι απευθύνεται άμεσα στην ιδεατή μηχανή και είναι το μεγαλύτερο. Αποτελείται από 7.000-8.000 γραμμές κώδικα, γραμμένες σε "C" και είναι φορητό για κάθε Η/Υ που διαθέτει ένα compiler της "C". Είναι υπεύθυνο για τον καθορισμό της αποδοτικότερης σειράς εκτέλεσης ενός αριθμού εργασιών (scheduling), το συντονισμό αυτών των εργασιών (coordinate), τη διαχείριση περιχών μνήμης (stacks) που προορίζονται για προσωρινή και αυτόματη αποθήκευση των στοιχείων ορισμένων καταχωρητών, τη διαχείριση της μνήμης κ.λ.π. Οι ρουτίνες αυτές απαιτούν άμεση απόκριση, γι' αυτό παραμένουν μόνιμα στη μνήμη υποστηρίζοντας το κεντρικό supervisory πρόγραμμα του kernel.

- Στο δεύτερο μέρος έχουμε ρουτίνες (Device drivers) που υποστηρίζουν τα περιφερειακά. Παρόλο που εξαρτώνται σχεδόν ολοκληρωτικά από το hardware, πολλές φορές τις βρίσκουμε γραμμένες σε "C". Αποτελούνται από 1.000 γραμμές κώδικα ή και περισσότερες, ανάλογα με το πόσο σύνθετες και πολλές είναι οι περιφερειακές μονάδες.

- Το τελευταίο βασικό κομμάτι του kernel, το μόνο γραμμένο αποκλειστικά σε Assembly, αναφέρεται στο χαμηλότερο επίπεδο του hardware. Είναι αυτό που πραγματικά ευθύνεται για τη δημιουργία της ιδεατής μηχανής και αποτελείται από 1.000 γραμμές κώδικα, αριθμός που καθορίζεται από την πολυπλοκότητα κάθε φυσικής μηχανής. Οι 10.000 γραμμές κώδικα του kernel δίνουν μια αναλογία σε σχέση με το ολό σύστημα που φτάνει το 5% με 10%. Κάθε Η/Υ μπορεί να προσαρμοστεί στα όρια της ιδεατής μηχανής που δημιουργεί το kernel ξαναγράφοντας μερικές εκατοντάδες γραμμές κώδικα από αυτό το ποσοστό του 10%.

Αυτός ο ευκολός μετασχηματισμός είναι το κλειδί για τη λεγόμενη μεταφερσιμότητα (portability) του UNIX, που αποδέσμευει το χρήστη από το hardware και το software κάθε εταιρείας.

## Ο Ρ Γ Α Ν Ω Σ Η Α Ρ Χ Ε Ι Ω Ν

Η διαχείριση των αρχείων σε ένα λειτουργικό σύστημα μπορεί να γίνει με οριζόντια ή την ιεραρχική οργάνωση. Στην οριζόντια

οργάνωση ή αναζήτηση κάθε αρχείου γίνεται άμεσα και κανένα αρχείο δεν εμπεριέχεται σε κάποιο άλλο. Σε ένα ιεραρχικό σύστημα αρχείων χρησιμοποιούνται κάποια ειδικά αρχεία -directories- που, αντί να περιέχουν δεδομένα, κρατούν κατάλογους με ονόματα άλλων αρχείων και στοιχεία για το που μπορούν να βρεθούν τα αρχεία που θέλετε μέσα στο σύστημα.

Κάθε directory μπορεί να έχει υπο-directories που με τη σειρά τους έχουν άλλα υπο-directories. Όταν ο χρήστης αναζητά κάποιο αρχείο, το σύστημα ψάχνει στο directory τα στοιχεία που καθορίζουν τη θέση αυτού του αρχείου στο δίσκο και έπειτα το μεταφέρει στη μνήμη.

Το σύστημα αρχείων του UNIX είναι ένα πολύπλοκο κατασκευάσμα. Η έκδοση SVR3 στην πλήρη της μορφή περιλαμβάνει περισσότερα από 2.200 αρχεία (περίπου 450 από αυτά είναι απλές εντολές) και 400 directories. Η οργάνωση τους είναι ιεραρχική και έχει τη δομή δέντρου.

## Τ Ο Σ Υ Σ Τ Η Μ Α Α Ρ Χ Ε Ι Ω Ν Τ Ο Υ U N I X

Εδώ το σύστημα αρχείων διαιρεί το δίσκο σε κομμάτια των 1.024 bytes (για το σύστημα V) που ονομάζονται blocks και αριθμούνται από το 0 μέχρι τον αριθμό που μπορεί να φτάσει κάθε δίσκος.

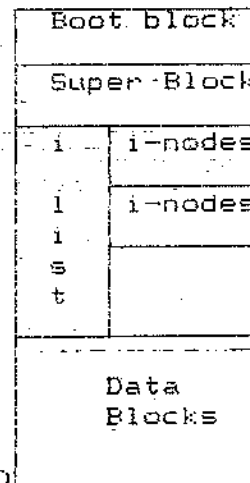
Το μήκος του block μπορεί να διαφέρει από σύστημα σε σύστημα. Το μικρότερο που χρησιμοποιείται είναι 512 bytes και το μεγαλύτερο 8.192 bytes. Για παράδειγμα, ένας δίσκος των 40 Mbytes έχει blocks που αριθμούνται από το 0 μέχρι το 39.000 περίπου.

Τα blocks αυτά χωρίζονται σε 4 περιοχές.

1. Block 0, (boot block): Στην Block 0 ουσία είναι αμεταχείριστο από το σύστημα αρχείων και είναι υπεύθυνο για το boot-up του UNIX. Σε μερικά συστήματα, λόγω περιορισμών στο hardware, δεν επιτρέπεται η χρήση του block 0 και η αρίθμηση ξεκινά από το 1.

2. Block 1, (super block): Περιέχει πληροφορίες για το μήκος του δίσκου καθώς και για τα μήκη των άλλων δυο περιοχών. Είναι το βασικότερο κομμάτι του συστήματος και πρέπει να διατηρείται ασφαλές.

Block 39000





3. Blocks 2 έως χ, (i-List): Είναι ένας μεταβλητός αριθμός blocks που περιέχουν i-nodes για κάθε αρχείο ή directory. Πολλές φορές το κομμάτι αυτό ονομάζεται και inode-table.

4. Blocks χ έως 39.000, (Data blocks): Αφιερώνονται στα περιεχόμενα αρχείων και directories.

Το πως πραγματικά ένα αρχείο γράφεται πάνω στο δίσκο είναι κάτι εντελώς διαφορετικό από αυτό που βλέπουμε στην οθόνη μας. Ένα αρχείο που ξεπερνά σε μήκος ένα block αποθηκεύεται τα περιεχόμενα του σε διασκορπισμένα blocks πάνω στο δίσκο. Το σύστημα αρχείων ψάχνει και βρίσκει αυτά τα blocks και στη συνέχεια τα εμφανίζει με τον κατάλληλο τρόπο στην οθόνη. Φυσικά πρέπει να υπάρχει κάποιο είδος πίνακα στο οποίο το UNIX θα μπορεί να κάνει αυτή τη μεταφορά.

Εδώ είναι που γίνεται η χρήση του i-node. Ένα i-node είναι ένας πίνακας 64 bytes που περιέχει πληροφορίες για κάθε αρχείο όπως: το μήκος του αρχείου, τον ιδιοκτήτη του, το είδος του αρχείου (απλό, special, directory), την ημερομηνία δημιουργίας, την ημερομηνία τελευταίας ενημέρωσης. Το σημαντικότερο όμως στοιχείο κάθε i-nodes είναι η λίστα με τις πραγματικές διευθύνσεις των blocks που θα καταλάβει το αρχείο πάνω στο δίσκο. Η λίστα αυτή (disk address list), περιέχει τους αριθμούς και τις αντίστοιχες διευθύνσεις 13 blocks.

Τα πρώτα 10 blocks παρέχονται άμεσα στο αρχείο που δημιουργείται. Για να γίνει λοιπόν η αναζήτηση ενός αρχείου, το σύστημα αρχείων του UNIX διαβάζει το disk address list που δίνει τη θέση των 10 αυτών blocks πάνω στο δίσκο. Το σύστημα αρχείων του UNIX επιτρέπει την εύκολη την εύκολη δημιουργία και την άμεση ανάκτηση δισκωλάσσοντας τον ωφέλιμο χώρο του δίσκου. Πολλά λειτουργικά, για να το πετύχουν αυτά, διαθέτουν δυσανάλογα πολύ χώρο για να καθορίσουν τη θέση των αρχείων στο δίσκο.

Το UNIX στο θέμα αυτό συμπεριφέρεται οικονομικά. Για παράδειγμα, σε ένα σύστημα 25.000 αρχείων που καταλαμβάνουν 130 Mbytes, τα i-nodes και indirect-blocks απαιτούν 11,5 Mbytes, ενώ η δομή βέντρου που υποστηρίζει αυτό το σύστημα χρησιμοποιεί 1500 directories που χρειάζονται και αυτά χώρο 1,1 Mbytes. Τα 13 αυτά Mbytes στο σύνολο των 130 δισκωθούν ένα ποσοστό χώρου 10% που είναι ικανοποιητικό και φανερώνει την επιτυχία του όλου μοντέλου.

#### T O M E Λ Λ O N

Καθώς σήμερα έχουμε στη διαθεσή μας ολο και πιο ισχυρούς υπολογιστές, οι περιορισμοί που επιβάλλονται από το ίδιο το μηχάνημα εξαφανίζονται, αφήνοντας μια νέα πρόκληση για τη δημιουργία προγραμμάτων που θα έχουν τη δυνατότητα να

εκμεταλλεύονται το πλήρες δυναμικό της τεχνολογίας. Το UNIX με το σύστημα SVR3 βοηθάει τον αρχάριο χρήστη να γνωρίσει αυτό το όντως πολύπλοκο λειτουργικό, δίνοντας έτσι την δυνατότητα και με την εμφάνιση των μηχανών 80486 να τρέχει και το MS-DOS κάτω από το UNIX, δίνοντάς του την αίσθηση ότι δουλεύει στο δικό του γνώριμο κόσμο. Η αληθινή προσπάθεια σίγουρα κάνει το UNIX να φιγουράρει σαν το λειτουργικό σύστημα του μέλλοντος.

## Το Λειτουργικό Σύστημα OS/2

Κάθε λειτουργικό σύστημα έχει για κύριο σκοπό του την εκμετάλλευση του δυναμικού που προσφέρει το Hardware, συντονίζοντας και ελέγχοντας ταυτόχρονα όλες τις εφαρμογές που εκτελούνται στο Σύστημα. Το DOS είναι ένα λειτουργικό Σύστημα που κυριόρχησε στο χώρο των μικροϋπολογιστών την παρασμένη δεκαετία. Τα κύρια μειονεκτήματα του είναι το φρόγμα 640 KB, η προστασία της ακεραιότητας του συστήματος και η δομημένη καθώς και η αρχική του χρήση αποκλειστικά σαν σύστημα προγραμματισμού, αποτέλεσαν τα τελευταία κυρίως χρόνια ένα σοβαρό εμπόδιο για τη σωστή εκμετάλλευση των δυνατοτήτων καινούργιων προϊόντων Hardware.

Το OS/2 είναι το πιο φιλόδοξο έργο της Microsoft, που μαζί με την IBM, δημιούργησαν ένα λειτουργικό Σύστημα που αν και θυμίζει στοιχεία άλλων Η.Σ. όπως VMS, MULTICS, παραμένει συμβατό με το DOS. Ο βασικός σκοπός της σημερινής μορφής του OS/2 είναι η καλύτερη εξυπηρέτηση ενός χρήστη και η παροχή των κατάλληλων μέσων για την ανάπτυξη εφαρμογών. Η μέγιστη εκμετάλλευση του Hardware, εκφραζόμενη σε μεγάλους αριθμούς MIPS και η εξυπηρέτηση των τελικών χρηστών περνούν προς το παρόν σε δεύτερη μοίρα.

Η δομημένη σε επίπεδα κατασκευή II OS/2 είναι ένα από τα βασικά του πλεονεκτήματα. Τρία είναι τα κύρια επίπεδα του : Οι οδηγίες των μονάδων (Device Drivers), ο πυρήνας (Kernel) και οι βιβλιοθήκες ή υποσύστημα (Dynamic Link Libraries/api, Subsystem).

Η κατασκευή του αυτή το κάνει εύκολο σε μεταβολές καθώς το υποσύστημα και οι οδηγίες μονάδων μπορούν εύκολα να αναπροσαρμοστούν σε νέες ανάγκες (OS/2 1.1, OS/2 Extended Edition). Τα τρία αυτά επίπεδα επικοινωνούν και αλληλεπιδρούν μεταξύ τους, εκτελώντας βέβαια την προκαθορισμένη τους εργασία μέσα στο σύστημα. Το OS/2 task model αναφέρεται σε τρεις καινούργιες έννοιες, που περιλαμβάνουν τα χαρακτηριστικά κάθε εφαρμογής. Το thread είναι η βασική μονάδα εκτέλεσης του OS/2.

Αυτό δίνει σε κάθε πρόγραμμα εφαρμογής ένα περιβάλλον λειτουργίας, που αποτελείται από τιμές registers, stack και τον τρόπο λειτουργίας της CPU. κάθε πρόγραμμα εφαρμογής είναι ένα στατικό στοιχείο, που αποκτά οντότητα μόνο μέσα από ένα thread.

Υπάρχει βέβαια η δυνατότητα εκτέλεσης πολλών threads στο σύστημα.

(ένα κάθε χρονική στιγμή) που το καθένα έχει τα ίδια στοιχεία (κώδικα προγράμματος).

Μια process είναι μια συλλογή από ένα ή περισσότερα threads καθώς επίσης και μέσω του συστήματος, όπως μνήμη, ανοικτά αρχεία κ.ά. Μπορούμε να πούμε ότι μια process είναι μια στιγμή από την εκτέλεση ενός προγράμματος, μαζί με τα μέσα που χρησιμοποιούνται από το πρόγραμμα.

Ο πυρήνας (kernel) είναι η καρδιά του συστήματος και εκτελεί τη διαχείριση της μνήμης, τον προγραμματισμό των εργασιών, επικοινωνία των processes, διαχείριση των αρχείων κ.ά.

### Π ο λ υ π ρ ο γ ρ α μ μ α τ ι σ μ ο ς

Το πλεονέκτημα του OS/2 έναντι του DOS είναι η ικανότητα του πολυπρογραμματισμού. Σε ένα τέτοιο περιβάλλον το Λ.Σ. διαχειρίζεται το μικροεπεξεργαστή έτσι ώστε, όλα τα ανεξάρτητα τμήματα του hardware να φαίνεται ότι λειτουργούν ταυτόχρονα.

Για το DOS, γράφτηκαν πολλά προγράμματα που δημιουργούν περιβάλλον πολυπρογραμματισμού (TopView, Desqview, MS-Windows) αλλά σε πραγματικό (real) τρόπο λειτουργίας. Το OS/2 προσφέρει προστασία της μνήμης και ακεραιότητα του συστήματος, με τον προστατευμένο (protected) τρόπο λειτουργίας. Με αυτό τον τρόπο ο 80286 επεκτείνει την ποσότητα προσπέλασης της Κεντρικής Μνήμης στα 16 MB. Το OS/2 χρησιμοποιεί επίσης εικονική (virtual) μνήμη που είναι η χρήση μιας περιφερειακής μονάδας αποθήκευσης, για να υποστηρίξει ανάγκες εφαρμογών που ξεπερνούν τις πραγματικές δυνατότητες της Κεντρικής Μνήμης.

Το κύριο στοιχείο ενός πραγματικού συστήματος πολυπρογραμματισμού είναι η ικανότητα του να διαχειρίζεται τα μέσα που διαθέτει (Resources). Το κυριότερο μέσο ενός συστήματος είναι ο επεξεργαστής, και το OS/2 διαθέτει την CPU στα προγράμματα που τη χρειάζονται, δίνοντας σε καθένα ένα συγκεκριμένο χρονικό τμήμα (time slice).

### Δ ι α χ ε ι ρ η σ η μ ν η μ η ς

Το φράγμα των 640 KB, για το DOS προκύπτει από τη δυνατότητα προσπέλασης του μικροεπεξεργαστή 8086. Όταν οι επεξεργαστές 80286 και 80386 εργάζονται κάτω από DOS, μιμούνται τον 8086 (emulation) και έχουν τους ίδιους περιορισμούς. Από 1 MB, τα 640 KB χρησιμοποιούνται από προγράμματα εφαρμογών και το DOS.

Τα υπόλοιπα 384 KB κρατούνται από το DOS για τις βασικές λειτουργίες Εισόδου - Εξόδου, καθώς και για το χειρισμό της οθόνης. Με τη βοήθεια κάποιων τεχνικών, το φράγμα των 640 KB μπορεί να ξεπεραστεί (LIM-EMS, CEMM), δημιουργώντας όμως προβλήματα μετακίνησης τμημάτων από και προς το χώρο της μνήμης.

Το OS/2 με τον προστατευμένο τρόπο λειτουργίας, αυξάνει τη δυνατότητα διαχείρισης πραγματικής μνήμης σε 16 MB και της virtual σε 1 GB. Αυτό το τελευταίο νούμερο είναι βέβαια θεωρητικό γιατί το OS/2 υποφέρει από τον περιορισμό των partitions μεγέθους 32 MB (όπως το DOS) και έτσι το μέγεθος της Virtual μνήμης περιορίζεται σημαντικά, στην πράξη.

Για τη σωστή λειτουργία των εφαρμογών πολυπρογραμματισμού πρέπει να υπάρχουν τουλάχιστον 2 MB κεντρικής μνήμης, και αν υπάρχουν εφαρμογές DOS, τότε πρέπει η μνήμη αυτή να αυξηθεί σε 3 MB. Το OS/2 είναι ένα σύστημα που έχει αρκετή ανάγκη από RAM και με την προσθήκη του Manager η ανάγκη αυτή θα μεγαλώσει.

Σε τελική ανάλυση το OS/2 στην παρούσα μορφή του είναι ένα σύστημα που είναι προσαρμοσμένο για την εξυπηρέτηση του χρήστη. Περισσότερο όμως και από το χρήστη, το OS/2 εξυπηρετεί τον προγραμματιστή που θέλει να αναπτύξει εφαρμογές σε περιβάλλον πολυπρογραμματισμού. Βέβαια τα εργαλεία για την ανάπτυξη εφαρμογών, που λειτουργούν σε περιβάλλον OS/2 είναι γλώσσες υψηλού επιπέδου (Fortran, Pascal, C, κ.ά.) και μόνο ορισμένα πακέτα. Είναι απαραίτητο για την καθιέρωση του συστήματος να κυριαρχούν σήμερα ανάλογες απαιτήσεις και το ενδιαφέρον για την ανάπτυξη του να στρέφεται προς τη δημιουργία μεγάλης βάσης από προγράμματα εφαρμογών.

Αυτό το ενδιαφέρον μορφοποιεί το OS/2 στη σημερινή φάση ανάπτυξης του και θα σταθεί καθοριστικό για μερικά βασικά μελλοντικά χαρακτηριστικά του.

## Η ιστορία του πρώτου PC (personal computer)

Η IBM παρουσίασε τον προσωπικό της υπολογιστή στο τέλος της δεκαετίας του 1970, για τον απλό λόγο ότι τα στελέχη των επιχειρήσεων, τις εταιρείες των οποίων προμήθευε μέχρι τότε η IBM με mainframes και παρελκόμενα, είχαν αρχίσει να χρησιμοποιούν υπολογιστές όπως ο PET της Commodore, ο TRS-80 της Tandy και ο Apple II, κύρια στις δουλειές γραφείου. Αν η IBM ήθελε πραγματικά να κρατήσει τις εταιρείες αυτές πιστούς πελάτες της, έπρεπε σύντομα να διαθέτει και λύση επιτραπέζιου υπολογιστή.

Η απόφαση δεν άργησε καθόλου να παρθεί, αλλά σε τέτοιες περιπτώσεις και όταν η πίεση αυξάνεται, δημιουργούνται σειρές από ανυπέρβλητα προβλήματα. Η IBM δεν είχε καμιά προηγούμενη εμπειρία σε PCs, έπρεπε να τον κατασκευάσει όσο το δυνατόν πιο γρήγορα και απασφάλιστο, το μηχάνημα έπρεπε να ακολουθεί την τελευταία λέξη της τεχνολογίας. Εχοντας κατά νου, τέτοιες καινοτομίες που σίγουρα θα είχαν μεγάλη εμπορική απήχηση, ήταν φυσικό να επηρεαστεί τόσο η σχεδίαση του μηχανήματος όσο και η γκάμα του software και του hardware που θα μπορούσαν οι χρήστες να προσθέσουν στα PCs τους.

Αποτέλεσμα της μεγάλης βιασύνης και της απαίτησης για όσο το δυνατόν πιο σύγχρονη τεχνολογία, ήταν να οδηγηθούν σε τρίτους, ανεξάρτητους κατασκευαστές. Η αλλαγή αυτή στη πολιτική της IBM ήταν σημαδιακή μια και σε αυτό σφείλεται κατά κύριο λόγο η οργάνωση ανοιχτής αρχιτεκτονικής που διαθέτει ο PC, από εκαντοτάδες ανεξάρτητους κατασκευαστές.

Η αρχική κατασκευή, βάση της ανοιχτής αρχιτεκτονικής, σήμαινε ότι ο καταναλωτής μπορούσε να διαλέξει από μια μεγάλη γκάμα συμβατών εξαρτημάτων. Πολυ σύντομα, μετά την εμφάνισή του μηχανήματος, εμφανίστηκαν εναλλακτικές προτάσεις για οθόνες, πληκτρολόγια και κάρτες επέκτασης. Πολλά από τα προϊόντα αυτά ξεπερνούν κατά πολύ τις προδιαγραφές της IBM, ενώ κοστίζουν πολύ πιο φθηνά.

### Ο επεξεργαστής

Δύο είναι τα βασικά στοιχεία που πρέπει να επιλεγούν, και πάνω σε αυτά μπορεί να στηθεί ένας μικρουπολογιστής: ο μικροεπεξεργαστής και το λειτουργικό σύστημα. Για άλλη μια φορά, οι αποφάσεις που πάρθηκαν σχετικά με το τι θα χρησιμοποιηθεί είχαν άμεση επίδραση στην εξέλιξη του PC. Στα τέλη της δεκαετίας του '70 οι προσωπικοί υπολογιστές χρησιμοποιούσαν μικροεπεξεργαστές των 8 bit, ενώ το πιο δημοφιλές λειτουργικό σύστημα ήταν το CP/M (Control Program for Microcomputers, πρόγραμμα ελέγχου για μικρουπολογιστές) της Digital Research.

Όσον αφορούσε το λειτουργικό σύστημα λοιπόν, η εκλογή ήταν προφανής. Πρόβλημα όμως παρέμεινε η εκλογή μικροεπεξεργαστή. Η IBM θα μπορούσε κάλλιστα να ενσωματώσει έναν υπάρχον 8 bit μικροεπεξεργαστή, πράγμα που θα σήμαινε ότι δεν θα είχε κανένα τεχνολογικό προβάδισμα απέναντι στον ανταγωνισμό... Απο την άλλη πλευρά, η χρησιμοποίηση ενός 16 bit επεξεργαστή θα σήμαινε ολοκληρωμένα κυκλώματα υποστήριξης για το υπόλοιπο κύκλωμα επίσης 16 bit. Η τεχνολογία αυτή, όμως, το 1980 δεν ήταν ούτε φτηνή, ούτε όμως εφαρμόσιμη.

Η βολική λύση που βρέθηκε τότε η χρησιμοποίηση από την IBM, του 8088 της Intel, μέλος της οικογένειας των μικροεπεξεργαστών 8086. Το πιο ενδιαφέρον σημείο του 8088 είναι ότι εσωτερικά είναι ένας 16 bit επεξεργαστής, αλλά επικοινωνεί με τα υπόλοιπα κυκλώματά χρησιμοποιώντας μόνο 8 bit τη φορά. Αυτό επέτρεψε στην IBM να σχεδιάσει και να χτίσει ένα 16μπιτο υπολογιστή, χρησιμοποιώντας ταυτόχρονα τα πιο φθηνά και πιο διαδοδεμένα, εκείνη την εποχή, θμπιτα ολοκληρωμένα κυκλώματα.

Η απόφαση για τη χρησιμοποίηση του μικροεπεξεργαστή της Intel, καθόρισε τη μελλοντική κατεύθυνση της σειράς των PC. Η σειρά δέθηκε με την οικογένεια των ολοκληρωμένων επεξεργαστών 8086 και αποτέλεσε την βάση για την παραγωγή του 80286 του 80386 και μόλις πρόσφατα του 80486.

#### Το λειτουργικό σύστημα

Το CP/M κυριαρχούσε κυριολεκτικά στο στερέωμα όταν η IBM ξεκίνησε να σχεδιάζει τον προσωπικό της υπολογιστή. Το σύστημα βέβαια είχε γραφτεί για μηχανές 8 bit και δε θα μπορούσε να δουλέψει στον PC με τον εσωτερικά 16 μπιτο 8088 χωρίς μετατροπές. Όμως, όπως συμβαίνει σήμερα με το MS-DOS, μεγάλος όγκος software υπήρχε μέχρι εκείνη τη στιγμή έτοιμος και έτρεχε κάτω από το CP/M, έτσι η IBM δεν ήθελε να χάσει την ευκαιρία που της δινόταν να εκμεταλλευθεί αυτό το software.

Η IBM πλησίασε την Microsoft, μια εταιρεία γνωστή μέχρι τότε για την κατασκευή δυνατόν γλωσσών προγραμματισμού. Αγόρασαν και διαμόρφωσαν ένα υπάρχον λειτουργικό σύστημα (το QDos) με αρκετές ομοιότητες στο CP/M και έτσι γεννήθηκε το MS-DOS. Το λειτουργικό σύστημα της Microsoft σχεδιάστηκε έτσι ώστε να έχει έντονες ομοιότητες με το CP/M, γιατί έτσι γινόταν πιο εύκολο στους χρήστες άλλων μικρουπολογιστών να μάθουν το νέο σύστημα και να εξοικιωθούν με αυτό ώστε να μεταφέρουν το υπάρχον software στα PC. Το MS-DOS δεν ήταν το μόνο λειτουργικό σύστημα που συγκόδευε τους πρώτους PC. Μαζί υπήρχε και το UCSD p-system, το οποίο ήταν σχεδιασμένο περισσότερο σαν σύστημα ανάπτυξης προγραμμάτων παρα, ως μέσο επικοινωνίας υπολογιστή - τελικού χρήστη. Το λειτουργικό σύστημα της microsoft ραγδαία αναδείχθηκε σε νικητή, με αποτέλεσμα τη δημιουργία πληθώρας έτοιμου software για το PC.

Ενα άλλο αποτέλεσμα της υιοθέτησης του MS-DOS ήταν κοινός περιορισμός που κληρονομήθηκαν μαζί με το σύστημα, όπως η δυσκολία για multi-tasking και multi-user εφαρμογές και ο περιορισμός στην μνήμη.

#### Η σχεδίαση του PC

Οι τρευματικές ομάδες της IBM που φημίζονται για την ταχύτητά και την απόδοση τους, παράγγααν διάφορα πρωτότυπα του PC, ένα από τα οποία επιλέχθηκε τελικά και ανακοινώθηκε τον Αυγούστο του 1981. Οι βασικές προδιαγραφές της πρώτης μηχανής φαντάζουν, εννέα μόλις χρόνια αργότερα, απελπιστικά χαμηλές :16 KB RAM, κόνενας οδηγός δισκέτας, ένα μόνο κασσετόφωνο.

Ενα βασικό χαρακτηριστικό του PC, εκτός από την ανοιχτή αρχιτεκτονική του, ήταν η κατασκευή του σε τρία χωριστά μέρη : την κεντρική μονάδα το monitor και το πληκτρολόγιο.

Η ιδέα της ανοιχτής αρχιτεκτονικής, με την οποία το σύστημα μπορούσε να επεκταθεί προσθέτοντας συσκευές -αχι-υποχρεωτικά κατασκευασμένες από τον προμηθευτή, έπαιρνε σάρκα και οστά με τις θύρες επέκτασης που βρίσκονταν μέσα στην κεντρική μονάδα του PC. Στο αρχικό μοντέλο οι θύρες ήταν πέντε, αλλά πολύ σύντομα έγιναν οκτώ αν και οι δύο ήταν κατηλειμμένες από τον controller των δίσκων και την κάρτα οθόνης.

#### Οι κάρτες οθόνης και γραφικών

Η απεικόνιση είναι ένα βασικό δομικό στοιχείο στο σύστημα του PC. Μια σειρά από διαφορετικά standards πέρασαν μέσα από τα χρόνια και συνδέθηκαν με μια τεράστια γκάμα από διατεθεμένα monitor (οθόνες). Το PC αποτελεί μοναδικό φαινόμενο ανάμεσα στους προσωπικούς υπολογιστές, αφού το hardware απεικόνισης βρίσκεται πάνω σε μια κάρτα επέκτασης. Όταν θελήσουμε να περάσουμε σε υψηλότερο standard, βγάζουμε την παλιά κάρτα και τοποθετούμε την νέα, αλλάζοντας και την οθόνη.



ΤΟ ΜΕΛΛΟΝ ΕΙΝΑΙ .....

ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

## ΝΕΥΡΩΝΙΚΑ

### ΔΙΚΤΥΑ

Οι σημερινοί υπολογιστές, λίγο πολύ έχουν ξεφύγει από την αρχική ιδέα της μηχανής του John von Neumann. Τα καταφέρνουν δηλαδή πραγματικά πολύ καλά, σε προβλήματα των οποίων η διαδικασία επίλυσης αποτελείται από συγκεκριμένη σειρά βημάτων. Για να γίνει όμως αυτό, θα πρέπει πρώτα κάποιος να κατανοήσει το πρόβλημα αρκετά καλά, ώστε να μπορέσει να κατασκευάσει το αλγόριθμο επίλυσής του.

Πολλές φορές όμως, η σχεδίαση του αλγόριθμου αυτού είναι εξαιρετικά πολύπλοκη (π.χ. επεξεργασία αξονικής τομογραφίας) ή ακόμη και αδύνατη, οπότε και προσεγγίζονται τα όρια δυνατοτήτων των σημερινών υπολογιστών. Αντίθετα, ο ανθρώπινος εγκέφαλος έχει τη δυνατότητα να επεξεργάζεται πολύ μεγαλύτερη γκάμα προβλημάτων, παρόλο μάλιστα που οι μονάδες επεξεργασίας του εγκέφαλου (νευρώνες) είναι σημαντικά μικρότερης ταχύτητας από τις μονάδες από τις μονάδες επεξεργασίας ενός σύγχρονου υπολογιστή (τόση μεγέθους αντίδρασης milliseconds για τους νεύρωνες και nanoseconds για επεξεργαστές).

Προβλήματα όπως η αναγνώριση χειρογράφων χαρακτήρων, η κατανόηση φυσικής ομιλίας και η αναγνώριση ενός αντικειμένου σε μια εικόνα είναι σχετικά απλά για τον άνθρωπο, αλλά σχεδόν άλυτα για έναν υπολογιστή, αφού δεν έχει βρεθεί ακόμη ικανοποιητική αλγοριθμική προσέγγιση, σε αυτά τα προβλήματα. Οι ιδιαίτερες αυτές δυνατότητες του ανθρώπινου εγκέφαλου είναι πολύ πιθανό να βασίζονται στον τρόπο και στη μορφή οργάνωσής του.

Ο εγκέφαλος -σύμφωνα με την εικόνα που έχουμε γι' αυτόν σήμερα - αποτελείται από ένα δίκτυο νευρώνων της τάξης -σε αριθμό- του 10 στην εντεκάτη. Οι νευρώνες αυτοί αλληλοσυνδέονται μεταξύ τους, κάθε ένας με μερικές χιλιάδες άλλους. Δεν έχει βρεθεί ακόμη με ποιόν ακριβώς τρόπο αυτό το βιολογικό νευρωνικό δίκτυο επιτρέπει τις λειτουργίες της σκέψης, της μνήμης κ.λ.π., αλλά είναι προφανές ότι το καταφέρει αρκετά καλά! Ένα τεχνητό νευρωνικό δίκτυο βασίζεται στην παραπάνω ιδέα, αλλά και στο αντίστοιχο θεωρητικό υπόβαθρο. Ας δούμε όμως τα πράγματα πιο αναλυτικά.

### ΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΑΠΟ ΠΙΟ ΚΟΝΤΑ

Νευρωνικό δίκτυο είναι ένα σύστημα επεξεργασίας πληροφοριών, αποτελούμενο από - μεγάλο - αριθμό αλληλοσυνδεδεμένων στοιχείων επεξεργασίας (νευρώνων).

Η δραστηριότητα κάθε νευρώνα υπολογίζεται τοπικά, με βάση τις συνδέσεις με άλλους νευρώνες και τη βαρύτητα (weight) των συνδέσεων αυτών. Ο υπολογισμός της δραστηριότητας αυτής γίνεται από τη συνάρτηση μεταφοράς (transfer function). Ο κάθε νευρώνας μπορεί να έχει πολλά σήματα εισόδου, αλλά παράγει ένα μόνο σήμα εξόδου.

Τα βασικά χαρακτηριστικά και συστατικά μέρη ενός Νευρωνικού Δικτύου είναι :

1. Ο νευρώνας : Πρόκειται για μια μονάδα επεξεργασίας, με περιορισμένη μνήμη και υπολογιστική ικανότητα, που αποτελεί το δομικό-λίθο ενός νευρωνικού δικτύου.

2. Αρχιτεκτονική : Η Αρχιτεκτονική του δικτύου χαρακτηρίζεται κυρίως από τον αριθμό των μονάδων επεξεργασίας (νευρώνων), από τον τρόπο με τον οποίο γίνονται οι συνδέσεις, αλλά και από το σύστημα των εξισώσεων και αλγορίθμων που χρησιμοποιούνται για τις λειτουργίες της μεταφοράς (transfer), μάθησης (learning) κ.λ.π. Η Αρχιτεκτονική counter propagation, για παράδειγμα, βασίστηκε στη δουλειά των Kohonen και Crossberg πήρε μορφή από τον R.H.Nielsen, ιδρυτή της Hecht Nielsen Neurocomputer Corp.

Το δίκτυο με την αρχιτεκτονική αυτή αποτελείται από 3 επίπεδα. Το πρώτο (input layer) χρησιμεύει για να δέχεται το διάνυσμα εισαγωγής στοιχείων, το ενδιάμεσο (hidden layer) δέχεται τα σταθμισμένα (weighted) σήματα του πρώτου και το επίπεδο εξόδου (output layer) δέχεται -με τη σειρά του- τα σταθμισμένα σήματα του ενδιάμεσου και παράγει το διάνυσμα εξόδου. Άλλος τύπος Αρχιτεκτονικής είναι η back propagation, που βασίζεται σ'έναν αλγόριθμο, για εκπαίδευση (training) δικτύων με πολλά επίπεδα.

3. Νόμος εκμάθησης (learning law) : Πρόκειται για μια εξίσωση, που μεταβάλλει όλους ή ορισμένους από τους συντελεστές βαρύτητας  $w(i)$  των σημάτων σε κάθε νευρώνα, με σκοπό να προσεγγίσει καλύτερο έναν επιθυμητό στόχο, για τη στάθμη εξόδου του νευρώνα. Ένας τέτοιος νόμος, που χρησιμοποιείται σε ευρεία κλίμακα, είναι ο νόμος του D.D.Hebb (1949). Σύμφωνα με αυτόν, η αλλαγή στο συντελεστή βαρύτητας,  $w(i, j)$  ενός νευρώνα  $j$ , είναι ανάλογη με το γινόμενο των σταθμών εξόδου του νευρώνα που εκπέμπει και του νευρώνα αποδέκτη.

Η εκπαίδευση ενός νευρωνικού δικτύου μπορεί να γίνει με δύο κυρίως τρόπους : Επιβλεπόμενη (supervised) και μη επιβλεπόμενη.

Στην πρώτη περίπτωση, τροφοδοτούμε το δίκτυο με ένα ζευγάρι διανυσμάτων που θέλουμε να συσχετίσουμε. Τα διανύσματα αυτά (με τη μαθηματική τους έννοια) μπορεί να είναι, για παράδειγμα, ζευγάρι χειρογράφων και τυπογραφικών χαρακτήρων. Τροφοδοτώντας το δίκτυο με το διάνυσμα εισόδου (χειρόγραφος χαρακτήρας), αυτό παράγει ένα διάνυσμα εξόδου, που συγκρίνεται με το διάνυσμα στόχο (τυπογραφικός χαρακτήρας). Στη συνέχεια, υπολογίζονται οι διαφορές μεταξύ διανύσματος εξόδου και διανύσματος στόχου.

Οι διαφορές αυτές συνιστούν ένα "λάθος". Το δίκτυο τότε μεταβάλλει τους συντελεστές βαρύτητας  $w(i, j)$ , με στόχο να μειώσει το λάθος αυτό, έτσι ώστε, στο δεύτερο και στους επόμενους κύκλους εκμάθησης, να προσεγγίσει καλύτερα το διάνυσμα στόχο και προοδευτικά να μειώσει το λάθος στο ελάχιστο.

Αντίθετα, στη μη επιβλεπόμενη εκμάθηση, δεν υπάρχει διάνυσμα στόχος. Το σύστημα προσαρμόζεται μόνο του, έτσι ώστε να παράγει μια συγκεκριμένη έξοδο, κάθε φορά που δέχεται την αντίστοιχη είσοδο.

4. Συνάρτηση μεταφοράς (transfer function) : Πρόκειται πάλι για εξίσωση, από την οποία υπολογίζεται η στάθμη εξόδου του νευρώνα, με βάση τις πιο πρόσφατες στάθμες εισόδου και τους συντελεστές βαρύτητας. Στη συνάρτηση μεταφοράς, ενσωματώνεται και ο νόμος εκμάθησης.

5. Scheduling function : Είναι μια συνάρτηση ή αλγόριθμος, που καθορίζει το αν, το πότε και το πόσο συχνά ένας νευρώνας θα λειτουργήσει, θα εφαρμόσει δηλαδή τη συνάρτηση μεταφοράς του. Στη συνέχεια θα δούμε πως λειτουργούν τα παραπάνω σε ένα συγκεκριμένο πρόβλημα.

#### ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΤΥΠΟΥ

Το πρόβλημα της αναγνώρισης προτύπου (pattern recognition) συνίσταται ουσιαστικά στην εύρεση μιας αντιστοίχισης ενός αντικειμένου και μιας κλάσης αντικειμένων. Δεδομένου δηλαδή κάποιου αντικειμένου, κάνουμε αναγνώριση προτύπου, όταν το κατατάσσουμε σε μια κλάση αντικειμένων.

Ετσι, βλέποντας ένα χειρόγραφο χαρακτήρα, τον αναγνωρίζουμε, όταν λέμε ότι αυτός ο χαρακτήρας είναι π.χ. το γράμμα "Α". Με τον ίδιο τρόπο, όταν βρισκόμαστε σε ένα δάσος, αναγνωρίζουμε ότι αυτό που βλέπουμε είναι δένδρο και μάλιστα έλατο. Αλλά ποιά είναι αυτά τα χαρακτηριστικά, που μας κάνουν, όταν βλέπουμε π.χ. το συγκεκριμένο χαρακτήρα, να λέμε ότι πρόκειται για το "Α", ή όταν βλέπουμε το δέντρο, να αναγνωρίζουμε ένα έλατο, τη στιγμή που όλα τα έλατα διαφέρουν μεταξύ τους; Ο Πλάτων, στην "Πολιτεία", υποστήριζε ότι πέρα από τον κόσμο που μας περιβάλλει, υπάρχει ο κόσμος των ιδεών. Ετσι, πίσω από τις φαινομενικές διαφορές των πραγμάτων, υπάρχει η ίδια ουσία, που τα συνδέει και τα κάνει να ανήκουν σε μεγαλύτερες κλάσεις.

Η "ιδέα" που έχουμε για το τι είναι ένα έλατο κυριαρχεί, σε σχέση με τις διαφορές που βλέπουμε από έλατο σε έλατο, παρόλο μάλιστα που είναι πολύ πιο δύσκολο να περιγράψουμε την "ιδέα" αυτή, πάνω στην οποία βασίζουμε την κρίση μας, από τις διαφορές τις οποίες αγνοούμε.

Κατά ένα παράξενο τρόπο, τα νευρωνικά δίκτυα φαίνονται ικανά να εντοπίζουν αυτή την ουσία των πραγμάτων, που κρύβεται πίσω από τις διαφορές τους, μέσω της δυνατότητάς τους για μάθηση.

Χωρίς αυτό να αποτελεί διαφανή διαδικασία η - πολύ περισσότερο - διαδικασία που να μπορεί να επεξηγηθεί από το ίδιο το σύστημα (όπως γίνεται με τα Εμπειρικά Συστήματα), η ικανότητα των νευρωνικών δικτύων, στην επίλυση του προβλήματος της αναγνώρισης προτύπου, είναι σημαντική. Σε πιο πρακτικούς όρους, ένα σύστημα αναγνώρισης προτύπου δέχεται σαν είσοδο ένα σύνολο στοιχείων, που αναπαριστούν η περιγράφουν ένα αντικείμενο και αποτελούν το διάνυσμα εισόδου. Το σύστημα τότε πρέπει να αποφασίσει, σε ποιά κλάση αντικειμένων ανήκει αυτό το διάνυσμα. Αυτό έχει εφαρμογές σε πολλά προβλήματα που μοιάζουν διαφορετικά μεταξύ τους.

Μια τέτοια κατηγορία προβλημάτων είναι ο έλεγχος (control) και η προσομοίωση (simulation). Στην προσομοίωση π.χ., το σύστημα τροφοδοτείται με ιστορικά στοιχεία και με τα αποτελέσματά τους, οπότε μαθαίνει να αναγνωρίζει τη σχέση μεταξύ τους.

Όταν λοιπόν του δώσουμε ένα σύνολο στοιχείων που δεν έχει αντιμετωπίσει ξανά, μας δίνει τα αποτελέσματα που αυτά θα είχαν (simulation), με βάση αυτά που έχει μάθει. Η λήψη αποφάσεων είναι επίσης πρόβλημα αναγνώρισης προτύπου. Όταν ένας τραπεζίτης συγκεντρώνει στοιχεία για να αποφασίσει εάν θα δώσει ένα δάνειο η όχι, προσπαθεί ουσιαστικά να κατατάξει ένα σύνολο στοιχείων (διάνυσμα εισόδου) σε μια από τις κλάσεις "Δεκτό" η "Μη Δεκτό".

Ένας συγκεκριμένος τύπος νευρωνικού δικτύου, το Nestor Learning System, λειτουργεί - σε γενικές γραμμές - ως εξής: Κάθε νευρώνας στο επίπεδο εισόδου (input layer), αντιστοιχεί σε ένα στοιχείο περιγραφής (συνιστώσα) του διανύσματος εισόδου. Για την περίπτωση π.χ. του δανείου, τέτοια στοιχεία θα μπορούσαν να είναι το ύψος του δανείου, ο χρόνος εξόφλησης κ.λ.π.

Για την περίπτωση των χειρογράφων χαρακτήρων, θα μπορούσαμε να μετράμε τον αριθμό των τομών μιας ευθείας που σαρώνει χαρακτήρα οριζόντια και κατακόρυφα, με τον ίδιο-το χαρακτήρα.

Κάθε νευρώνας, στο επίπεδο εξόδου, αντιστοιχεί σε μια κατηγορία προτύπου. Έτσι, για το δάνειο, θα είχαμε δύο τέτοιες κατηγορίες ("Δεκτό" και "Μη Δεκτό"), ενώ για τους χαρακτήρες θα είχαμε 24.

Όταν το διάνυσμα εισόδου καταλήγει να ενεργοποιήσει ένα μόνο νευρώνα του επιπέδου εξόδου, η αναγνώριση είναι 100% επιτυχής (unambiguous).

Το κυρίως πρόβλημα που αντιμετωπίζει το νευρωνικό δίκτυο είναι η κατασκευή της περιοχής που αντιστοιχεί σε κάθε κατηγορία προτύπου. Αν θεωρήσουμε το διανυσματικό χώρο  $N$  διάστασης, που έχει σαν βάση τα  $N$  διανύσματα περιγραφής (π.χ. αριθμός τομών σε οριζόντια σάρωση και αριθμός τομών σε κατακόρυφη σάρωση), τότε η κατηγορία προτύπου (π.χ. το γράμμα "Α") ορίζεται σαν μια περιοχή του χώρου αυτού, που σημαίνει ότι κάθε συνδυασμός τομών, που βρίσκεται μέσα στην περιοχή του "Α", αντιστοιχεί στο γράμμα αυτό.

Η διαδικασία εκμάθησης του νευρωνικού δικτύου αντιστοιχεί, λοιπόν, ακριβώς στην κατασκευή αυτών των περιοχών, για κάθε κατηγορία πρότυπου.

Η μαθηματική αναπαράσταση της διαδικασίας είναι η κατασκευή αλληλοκαλυπτόμενων δίσκων, που προσδευτικά δημιουργούν την κάθε περιοχή. Με τον τρόπο αυτό, το νευρωνικό δίκτυο χρειάζεται - για την εκπαίδευσή του - περιορισμένο αριθμό ζευγαριών στοιχείων, που αντιστοιχούν στα κέντρα των δίσκων, ενώ - με τη δημιουργία των δίσκων και ταλικά της περιοχής - επεκτείνει τη δυνατότητα αναγνώρισης, σε απείρως συνδυασμών.

### Η ΑΓΟΡΑ ΓΙΑ ΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Κάνοντας χρήση μιας εντελώς διαφορετικής προσέγγισης στην επίλυση προβλημάτων, τα νευρωνικά δίκτυα δεν είναι - τουλάχιστον όσον αφορά - ανταγωνιστικά με άλλα προϊόντα hardware, η software. Σύμφωνα και με τον R.H.Nielsen, η τεχνολογία τους έρχεται να καλύψει ένα κενό, που δεν μπορούσε - εκ των πραγμάτων - να καλυφθεί από άλλες τεχνολογίες.

Ετσι, η αλγοριθμική προσέγγιση και οι νευρωνολογιστές είναι συμπληρωματικές και όχι ανταγωνιστικές τεχνολογίες.

Κανείς δεν θα χρησιμοποιήσει νευρωνικά δίκτυα για εφαρμογές μισθοδοσίας, αλλά είναι πολύ πιθανό να τα χρησιμοποιήσει π.χ. για ένα σύστημα αναγνώρισης φωνής, όπου η αλγοριθμική προσέγγιση - όπως αναφέρθηκε - είναι εξαιρετικά δύσκολη. Από το πρώτο νευρωνικό δίκτυο (Perceptron), που αναπτύχθηκε στο πανεπιστήμιο Cornell των ΗΠΑ - το 1957 - και αναγνώριζε τοπογραφικούς χαρακτήρες, μέχρι σήμερα, σημαντικός αριθμός επιχειρήσεων έχει επενδύσει σε αυτή την τεχνολογία. Ωστόσο, δεν είναι παρά τα τελευταία χρόνια, που η έρευνα και οι εφαρμογές για τα νευρωνικά δίκτυα έχουν έρθει στην επιφάνεια.

Η Behavioiristics Inc., για παράδειγμα, διαθέτει στην αγορά ένα νευρωνικό δίκτυο, για προγραμματισμό πτήσεων με βάση τον αριθμό και την αξία των κρατήσεων, έτσι ώστε να βελτιστοποιούνται τα κέρδη της εταιρείας. Ακόμη, στο Ελσίνκι, ένας πολύ γνωστός ερευνητής στο χώρο αυτό, ο Teuvo Kohonen, παρουσίασε το ακριβέστερο - μέχρι σήμερα - σύστημα αναγνώρισης φωνής, βασισμένο σε νευρωνικό δίκτυο.

THE FIFTH

THE FIFTH

## ΓΕΝΙΚΑ

Το πρόγραμμα που ακολουθεί ανήκει στην κατηγορία των προγραμμάτων αποθήκης, δηλαδή η φιλοσοφία του προγράμματος βρίσκεται μέσα στο τετραπύλο :

ΑΠΟΘΗΚΗ (Stock)  
ΠΕΛΑΤΕΣ ΠΡΟΜΗΘΕΥΤΕΣ  
ΤΑΜΕΙΟ

Όποια λογιστική κίνηση και να γίνει έχει ως αποτέλεσμα την αλλαγή και των υπόλοιπων συντελεστών. Δηλαδή μια νέα παραγγελία θα επιφέρει αλλαγή στο stock της αποθήκης αλλά και αλλαγή στην καρτέλα του πελάτη όπως επίσης και στα έσοδα του ταμείου. Εδώ βρίσκεται και η συμβολή του προγράμματος το οποίο με το να το ενημερώνουμε το είδος της κίνησης που έγινε (αγορά ή πώληση προϊόντων), αναλαμβάνει αυτό την ενημέρωση των υπόλοιπων μερών του τετραπύλου. Στην συνέχεια θα γίνει μια αναλυτική παρουσίαση του προγράμματος και των δυνατοτήτων που παρέχει σε μια επιχείρηση.

Η επικοινωνία μεταξύ προγράμματος και χρήστη γίνεται μέσω των διαφόρων μενού (μενού = σύνολο από επιλογές), το πρώτο μενού που εμφανίζεται στην οθόνη μας είναι το εξής :

### ΚΕΝΤΡΙΚΟ ΜΕΝΟΥ ΕΠΙΛΟΓΩΝ

- |                   |
|-------------------|
| 1. ΑΠΟΘΗΚΗ        |
| 2. ΕΚΤΥΠΩΣΕΙΣ     |
| 3. ΠΩΛΗΣΕΙΣ       |
| 4. ΒΟΗΘ.ΕΡΓΑΣΙΕΣ  |
| 0. ΕΞΟΔΟΣ / ΤΕΛΟΣ |

Η πρώτη επιλογή που πρέπει να διαλέξουμε είναι η Αποθήκη και αυτό γιατί πρέπει να εισάγουμε τα είδη με τα οποία θα γίνονται οι διάφορες παραγγελίες.

Μετακινώντας την φάσεινή μπάρα στην λέξη Αποθήκη και πατώντας Enter θα εμφανιστεί στην οθόνη μας το εξής μενού :



1. ΕΙΣΑΓΩΓΗ ΕΙΔΩΝ
2. ΔΙΟΡΘΩΣΗ ΕΙΔΟΥΣ
3. ΔΙΑΓΡΑΦΗ ΕΙΔΟΥΣ
4. ΕΜΦΑΝΙΣΗ ΕΙΔΟΥΣ
5. ΑΠΟΓΡΑΦΗ
6. ΕΞΟΔΟΣ ΑΠΟ ΑΠΟΘΗΚΗ

Ανάλογα με το τι θέλουμε να κάνουμε μετακινούμε την φωτεινή μπάρα στην ανάλογη επιλογή. Εάν θέλουμε να εισαγάγουμε στην αποθήκη ένα νέο είδος επιλέγουμε την εισαγωγή ειδών και ενημερώνουμε το αρχείο του είδους με τα στοιχεία που θέλουμε.

Η οθόνη που εμφανίζεται όταν επιλέξουμε την εισαγωγή ειδών είναι :

Κωδικός :	Περιγραφή :	
Προμηθευτής :	Ελάχιστη ποσότητα :	
Τιμή πώλησης :	Κατηγορία Φ.Π.Α :	Μονάδα μέτρησης :
Τιμή αγοράς :	Ημερ/νια τελευτ. αγοράς :	Ποσότης :
Συντελεστής κέρδους :	Πωλήσεις μέχρι σήμερα :	

Οι υπόλοιπες επιλογές μας δίνουν την δυνατότητα είτε να διορθώσουμε κάποιο στοιχείο ενός είδους είτε να διαγράψουμε ένα προϊόν, είτε απλά να το εμφανίσουμε στην οθόνη.

Μία χρήσιμη επιλογή είναι η Απογραφή η οποία μας δίνει την δυνατότητα να δούμε είτε στην οθόνη του υπολογιστή είτε σε εκτύπωση την κατάσταση που βρίσκεται η Αποθήκη μας και αυτό γίνεται με το να μας εμφανίζει τα αποθέματα που έχουμε για κάθε προϊόν, μία ανάλογη εκτύπωση σε printer μας δίνει τις εξής πληροφορίες :

ΑΠΟΓΡΑΦΗ ΑΠΟΘΕΜΑΤΩΝ

ΚΩΔΙΚΟΣ	ΠΕΡΙΓΡΑΦΗ	ΣΕ STOCK	ΕΛΑΧΙΣΤΗ Μ.Μ.	ΚΟΣΤΟΣ ΠΟΣΟΤΗΤΑ
100	MARTINI	100	55 1	333
1000	ΟΥΙΣΚΙ	80	20 1	2000
13	ΚΟΛΑ-ΚΟΚΑ	15	6 1	123
2	ΒΟΤΚΑ	29	10 10	1480
233	SPRITE	40	35 1	110
3	ΤΖΙΝ	16	10 1	400
*** Total ***		282	136	4446

Μετα την περάτωση των εργασιών μας στην Αποθήκη γυρίζουμε στο Κεντρικό μενού και επιλέγουμε τις Βοηθ.Εργασίες με σκοπό να ειδοώσουμε τα στοιχεία των προμηθευτών και των πελατών. Επιλέγοντάς λοιπόν με την φτερυγή μπάρα τις Βοηθ.Εργασίες εμφανίζεται στην οθόνη μας το παρακάτω μενού :

#### ΒΟΗΘΗΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

1. Ανακατάταξη αρχείων
2. Αντιγραφή αρχείων
3. Επανάφορα αρχείων
  
4. Προεκτύπωση αποδείξεων
5. Τομείο
6. Διαγραφή τομείου
7. Πελάτες
8. Προμηθευτες
9. Επωνυμία - Στοιχεία
  - Α. Εκκαθάρ. Φ.Π.Α. (18%)
  - Β. Ετικετογράφος
  - Γ. Επιστροφή ειδους
  - Δ. Αρίθμηση αποδείξεων
  
0. ΕΞΟΔΟΣ στο κ.μενου

Βλέποντας αυτό το μενού διακρίνουμε ένα διαχωρισμό μεταξύ των τριών πρώτων επιλογών και των υπόλοιπων, αυτό γίνεται γιατί οι τρεις πρώτες επιλογές αναφέρονται σε εσωτερικές καθαρά λειτουργίες του συστήματος. Έτσι αν διαλέξουμε την πρώτη επιλογή θα γίνει μέσα στο πρόγραμμα μία εκκαθάριση των διαγραμμενων στοιχείων (το είδαμε πριν με την επιλογή Αποθήκη -> Διαγραφή Ειδους) και αυτό για να μειωθεί το μέγεθος των αρχείων και να γίνει πιο γρήγορο το πρόγραμμα. Εμείς παρακάμπτοντας τις τρεις πρώτες επιλογές πηγαίνουμε στην επιλογή Προεκτύπωση αποδείξεων - πάταμε Επτέρ και εμφανίζεται στην οθόνη μας το εξής μενού :

#### ΠΡΟΕΚΤΥΠΩΣΗ ΑΠΟΔΕΙΞΕΩΝ ΓΙΑ ΤΗΝ ΕΦΟΡΙΑ

Αριθμός έναρξης :

Αριθμός περάτωσης :

Σειρά :

Με την προεκτύπωση αποδείξεων ορίθμουμε σύμφωνα με τους αριθμούς που μας έχει δώσει η εφορία τα τιμολόγια πώλησης. Είτον εκτυπώνη θα παρουμε ως τιμολόγιο πώλησης αλλά και ως απόδειξη λιανικής πώλησης το εφεξής παραστέτοκό :

egrated Systems & Solutions

ΑΡΜΟΓΕΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

γής 66 Περιστέρη Αθήνα 12157

Τ: 5718113 5733592

Μ: 12345678

ΣΕΙΡΑ 017

ΑΡΙΘ.ΣΕΡ.:

ΑΡΙΘ.ΠΑΡΑ:

ΗΜΕΡ/ΜΙΑ ΔΡΑ

ΔΙΚΟΣ	ΕΙΔΟΣ	ΕΚΠ/ΠΟΣΟΤ.	ΜΗ	ΤΙΜΗ	ΦΠΑ	ΑΞΙΑ
οση:Απόδειξη λιανικής πώλησης Τιμολόγιο πώλησης - Δελτίο αποστολής Δελτίο Επιστροφής - Πιστωτικό						ΣΥΝΟΛΟ ΕΚΠΩΣΗ Φ.Π.Α. ΠΑΡΩΤΕΟ ΠΟΣΟ

ΣΥΝΟΛΟ ΦΠΑ

Μηχανογράφηση: I.S.S., Τηλ. 5718113

μετά την εισαγωγή των αποδείξεων-πρωτοκόμια στην επιλογή πελάτες, για να εισαγούμε τα στοιχεία του κάθε πελάτη.  
 Στην οθόνη μας θα έχουμε την εξής εικόνα:

ΟΝΟΜΑ : _____	ΚΩΔΙΚΟΣ : 00
ΔΙΕΥΘΥΝΣΗ : _____	ΑΡΧΗ ΣΥΝΕΡΓΑΣΙΑΣ : / /
ΠΟΛΗ : _____	ΕΠΑΓΓΕΛ. : _____
ΤΚ : _____	ΤΗΛΕΦΩΝΟ : _____

ΛΟΓΙΣΤΙΚΗ ΚΑΡΤΕΛΑ	ΤΕΛΕΥΤΑΙΑ ΣΥΝΑΛΛΑΓΗ
ΜΕΧΡΙ ΣΗΜΕΡΑ : _____	ΔΡΥ
ΥΠΟΛΟΙΠΟ : _____	/ /

Νέος πελάτης Διόρθωση Αναζήτηση Σβήσιμο/Διαγραφή Πληρωμή/Αγορά Καρτέλα Διαγραφ. Καρτέλ. Εξόδος
---

Ανάλογα με την επιλογή που θα διαλέξουμε γίνονται ορισμένες εργασίες, έτσι έχουμε :

- Νέος πελάτης : Εισαγωγή στοιχείων πελάτη
- Διόρθωση : Διόρθωση " "
- Αναζήτηση : Δίνοντας τον κωδικό κάποιου πελάτη μας βγάζει την προσωπικά του καρτέλα.
- Σβήσιμο : Σβήνει κάποιον πελάτη από το αρχείο.
- Πληρωμή/Αγορά : Εισαγουμε στοιχεία στην Λογιστική Καρτέλα
- Διαγραφ. Καρτέλ. : Σβήνουμε στοιχεία μόνο από την Λογιστική Καρτέλα
- Εξόδος : Επιστροφή στο προηγούμενο μενού.

Την ίδια διαδικασία πραγματοποιούμε και με την επόμενη επιλογή από το μενού που είναι οι ΠΡΟΜΗΘΕΥΤΕΣ.

Με την επιλογή ΕΠΙΡΩΝΥΜΙΑ - ΣΤΟΙΧΕΙΑ εισαγουμε τα στοιχεία που θα θέλαμε να τυπώνονται με κάθε δελτίο λιανικής πώλησης δηλ. το όνομα της εταιρείας, το τηλέφωνο και ότι άλλο κρίνουμε εμείς απαραίτητο.

Με την επιλογή Έκκαθαρ. Φ.Π.Α. (18%) εκτυπώνεται μια κατάσταση με τα είδη που κινήθηκαν και έχουν Φ.Π.Α. (18%).  
 Γυρίζοντας στο κεντρικό μενού και επιλεγώντας τις ΠΩΛΗΣΕΙΣ εισαγούμε τον κωδικό του είδους που θα πωληθεί, αμέσως εμφανίζεται το όνομα του και η τιμή που έχει κατά μονάδα προϊόντος, εμείς πληκτρολογούμε την ποσότητα που θα πωληθεί και ο Η/Υ μας εμφανίζει το Φ.Π.Α. που έχει το είδος καθώς και την συνολική τιμή, στον πούλαμε περισσότερα από ένα κομμάτια και είδη. Αμέσως μετά μας ζητά το πρόγραμμα να πούμε αν προκειται για δελτίο λιανικής πώλησης ή Τιμολόγιο πώλησης.  
 Ανάλογα με το τι θα επιλέξουμε θα μας εκτυπώσει για μεν το δελτίο πώλησης :

ΑΠΟΔΕΙΞΗ ΛΙΑΝΙΚΗΣ ΠΩΛΗΣΗΣ

Αριθμός Παραστατικού : 124  
 04/12/90 06:33:53

Κωδικός	Είδος	Εκπτ.	Ποσot.	ΜΜ	ΤΙΜΗ	ΦΠΑ	ΑΞΙΑ
100	MARTINH	0.0	1	1	1000	18	1180
Συνολο							1180

Ενώ για ΤΙΜΟΛΟΓΙΟ ΠΩΛΗΣΗΣ εκτυπώνεται η παρακάτω απόδειξη :

ΤΙΜΟΛΟΓΙΟ ΠΩΛΗΣΗΣ - ΔΕΛΤΙΟ ΑΠΟΣΤΟΛΗΣ  
 Γιαννοπούλας & ΣΙΑ Πατρα  
 ΑΦΜ: 128797

Κωδικός	Είδος	Εκπτ.	ΦΠΑ	ΑΞΙΑ
100	MARTINH	0.0	18	1180
Συνολο Υπολογισο				1000 180

Η τελευταία επιλογή που μένει από το Κεντρικό μενού είναι οι ΕΚΤΥΠΩΣΕΙΣ, με αυτήν την επιλογή μας δίνεται η δυνατότητα να εκτυπώσουμε είτε με αλφαβητική σειρά είτε κατά κωδικό είτε τέλος μεμονωμένα τα στοιχεία των ειδών που έχουμε στην Αποθήκη μας. Στις επόμενη σελίδα υπάρχει τα αποτελέσματα αυτών των εκτυπώσεων.

Page No. 1  
23/03/88

ΚΑΤΑΛΟΓΟΣ ΕΜΠΟΡΕΥΜΑΤΩΝ ΜΕ ΚΩΔΙΚΟΥΣ  
ΑΡΙΘΜΟΥΣ ΕΙΔΟΥΣ ΚΑΙ ΚΑΤΗΓΟΡΙΕΣ ΦΠΑ

ΚΩΔΙΚΟΣ	ΠΕΡΙΓΡΑΦΗ	ΚΑΤ. ΦΠΑ	ΜΙΑΝΙΚΗ	ΠΡΟΜΗΘΕΥΤΗΣ
100	MARTINI	2	434	3
13	ΚΟΛΑ-ΚΟΚΑ	12	130	1
3	TZIN	35	450	3
2	ΒΟΥΤΑ	12	1600	2
233	SPRITE	8	130	12
1000	ΟΥΙΣΚΙ	4	1500	1



## ΒΙΒΛΙΟΓΡΑΦΙΑ

Μικροπολογιστές

του Adam Osborne

εκδόσεις : ΝΙΚΚΑΝ 1980.

Ο οδηγός της Borland για την TURBO PASCAL 5,0 & 5,5

του Steve Wood

εκδόσεις : Κλειδάριθμος 1989.

Όλα όσα θέλετε να μάθετε για τους Υπολογιστές

του Richard Stevens

εκδόσεις : Κλειδάριθμος 1986.

Το λειτουργικό σύστημα

του Παν. Τασουλα

Θεσσαλονίκη 1987.

Το λειτουργικό σύστημα MS-DOS

του Γ. Ιωάννου

Αθήνα 1988

Εγχειρίδιο Εισαγωγής Στα Εμπειρικά Συστήματα

των Β. Κρικετου-Κ. Παστρα

εκδόσεις : Εταιρεία Ανάπτυξης της  
Ναυτικής Τεχνολογίας.

Το λειτουργικό σύστημα OS/2.

του Gordon Letwin

εκδόσεις : Κλειδάριθμος 1988.

Α Ρ Θ Ρ Α από περιοδικά.

Η ιστορία των PC.

του Θεόδωρη Καλικά

Από το περιοδικό Computer software  
τεύχος 30, ΑΥΓ.-ΣΕΠΤ. 1989.

Τηλεματική και Κοινωνία.

του Γιάννη Ηλιαδη

Από το περιοδικό Computer & Software  
τεύχος 33, ΔΕΚ. 1989.

