ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΑΤΡΩΝ

ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ

ΤΜΗΜΑ ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΥ ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΟ ΔΥΑΔΙΚΟ ΣΥΣΤΗΜΑ ΑΡΙΘΜΗΣΗΣ ΚΑΙ ΟΙ ΕΦΑΡΜΟΓΕΣ ΤΟΥ

ΦΟΙΤΗΤΡΙΕΣ

ΤΣΙΡΩΝΗ ΙΟΥΛΙΑ

ΚΟΥΜΟΥΤΣΙΔΗ ΑΝΑΣΤΑΣΙΑ

ΕΠΙΒΛΕΠΟΥΣΑ ΚΑΘΗΓΗΤΡΙΑ

ΚΑΛΑΠΟΔΗ ΑΛΕΚΑ

ΠΑΤΡΑ 2012

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή εργασία με θέμα το Δυαδικό Σύστημα Αρίθμησης Και Οι Εφαρμογές Του, πραγματοποιήθηκε στο Ανώτατο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Πατρών, στη Σχολή Διοίκησης Και Οικονομίας στο Τμήμα Επιχειρηματικού Σχεδιασμού Και Πληροφοριακών Συστημάτων.

Θέλουμε να ευχαριστήσουμε την επιβλέπουσα καθηγήτριά μας Καλαπόδη Αλέκα η οποία μας βοήθησε ώστε να ολοκληρωθεί αυτή η εργασία. Την ευχαριστούμε για όσα μας δίδαξε, για το επιστημονικό υλικό που μας προσέφερε, τις συμβουλές της, και τις ώρες που μας αφιέρωσε.

Επίσης θα θέλαμε να ευχαριστήσουμε τους γονείς μας για την υπομονή και την συμπαράστασή τους κατά την διάρκεια της εργασία μας.

Το δυαδικό σύστημα και οι εφαρμογές του, αποτέλεσε αντικείμενο μελέτης για την πτυχιακή εργασία που απαιτείται για την ολοκλήρωση των σπουδών μας. Οι γνώσεις που απαιτούνταν για τη δημιουργία της προέρχονταν κυρίως από τον τομέα της πληροφορικής και των μαθηματικών. Και οι δύο τομείς είναι συνυφασμένοι με τη καθημερινή ζωή των ανθρώπων και παίζουν σημαντικό ρόλο σε αυτή. Ακόμα, με την έρευνα και την μελέτη που πραγματοποιήθηκαν προκειμένου να δομηθεί η εργασία, αποκομίσαμε γνώσεις που θα μας βοηθήσουν περισσότερο στην κατανόηση της λειτουργίας του υπολογιστή, μέσα από την γλώσσα των μαθηματικών. Κατά τη διάρκεια της εργασίας μελετήθηκαν οι εφαρμογές του δυαδικού συστήματος και έγινε σύγκριση με άλλα συνηθέστερα συστήματα αρίθμησης κατανοώντας έτσι για ποιο λόγο εφαρμόζεται αυτό και όχι κάποιο από τα υπόλοιπα συστήματα αρίθμησης.

Τέλος κατά το χρονικό διάστημα της εργασίας μας έγινε περισσότερο αντιληπτός ο τρόπος που λειτουργεί ο υπολογιστής με το σύστημα αυτό σήμερα και όχι μόνο.

ΠΕΡΙΛΗΨΗ

Η πτυχιακή αυτή εργασία σκοπό έχει να παρουσιάσει πληροφορίες σχετικά με το δυαδικό σύστημα αρίθμησης και τις εφαρμογές του.

Επειδή οι ηλεκτρονικοί υπολογιστές τα τελευταία χρόνια διαδραματίζουν ολοένα και σημαντικότερο ρόλο στην καθημερινή μας ζωή και παρά το γεγονός ότι η εσωτερική γλώσσα που καθοδηγούσε ορισμένους από τους αρχικούς υπολογιστές ήταν βασισμένη στο δεκαδικό σύστημα, τώρα σχεδόν όλοι οι υπολογιστές χρησιμοποιούν το δυαδικό σύστημα.

Στην εργασία αυτή γίνεται δηλαδή λεπτομερής αναφορά του δυαδικού συστήματος σχετικά με το πως δημιουργήθηκε και ποια η μέχρι τώρα εξέλιξή του (ποια είναι δηλαδή η ιστορική του αναδρομή). Ιστορική αναδρομή γίνεται και σε αριθμητικά συστήματα που χρησιμοποιούσαν στο παρελθόν αλλά και σήμερα. Επίσης γίνεται σύγκριση σχετικά με τα υπόλοιπα συνηθέστερα συστήματα αρίθμησης που χρησιμοποιούνται (οκταδικό, δεκαεξαδικό κ.τ.λ.) και κυρίως με το δεκαδικό σύστημα, κατανοώντας τους λόγους που τελικά χρησιμοποιείται το δυαδικό και όχι τα άλλα.

Εκτός από αυτό, στη πτυχιακή αναφέρονται και οι διάφορες εφαρμογές στις οποίες εφαρμόζεται το δυαδικό σύστημα όπως π.χ. σε ψηφιακό σήμα, δυαδικά δέντρα, δυαδική αναζήτηση κ.ά. Η κύρια εφαρμογή όμως είναι κυρίως στον υπολογιστή συμπεραίνοντας ότι πρόκειται για ένα σύστημα που είναι άμεσα συνδεδεμένο με αυτόν.

Επομένως στην εργασία αυτή θα δοθεί ιδιαίτερη προσοχή στην κατανόηση της σημασίας του, δηλαδή για ποιο λόγο χρησιμοποιείται αυτό το σύστημα αρίθμησης και όχι κάποιο άλλο καθώς επίσης και στην ανάδειξη των εφαρμογών του δυαδικού συστήματος που έχει τη κύρια χρήση του στον τομέα της πληροφορικής.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 4
Βασικές λογικές πράξεις – λογικές πύλες

ΚΕΦΑΛΑΙΟ 5
Δυαδική κωδικοποίηση

ΕΙΣΑΓΩΓΗ

Η ανάγκη του ανθρώπου για μετρήσεις τον οδήγησε αρχικά στην επινόηση των αριθμών, κατόπιν στην επινόηση συμβόλων για την παράστασή τους και τέλος στη δημιουργία των αριθμητικών συστημάτων όπως το οκταδικό, δεκαεξαδικό, δεκαδικό, δυαδικό κ.τ.λ.
Ο προσδιορισμός της πτυχιακής αυτής αφορά το δυαδικό σύστημα και τις διάφορες εφαρμογές που έχει κυρίως στους υπολογιστές.
Σε αντίθεση με τους ανθρώπους οι οποίοι χρησιμοποιούν το δεκαδικό σύστημα αρίθμησης για να κάνουν τις μετρήσεις, οι υπολογιστές χρησιμοποιούν το δυαδικό. Αυτό γίνεται κυρίως λόγω της απλότητας του συστήματος αυτού, επειδή στηρίζεται μόνο σε δύο σύμβολα το 0 και το 1, της ταχύτητας που προσφέρει εξαιτίας αυτών των συμβόλων, του χαμηλού κόστους κατασκευής ηλεκτρονικών κυκλωμάτων και των αποθηκευτικών μέσων κ.τ.λ. Για τους παραπάνω κυρίως λόγους το δυαδικό αποτελεί το πιο διαδεδομένο σύστημα στους υπολογιστές.
Έτσι σκοπός της εργασίας αυτής είναι να παρουσιάσει πληροφορίες σχετικά την έννοια του δυαδικού συστήματος κάνοντας μία ιστορική αναδρομή, ποια η σχέση με τα υπόλοιπα συστήματα αρίθμησης (οκταδικό, δεκαεξαδικό, δεκαδικό κ.α.) στα οποία γίνεται μία σχετική αναφορά, οι μετατροπές που γίνονται μεταξύ αυτών συστημάτων καθώς επίσης και ο λόγος που επιλέγεται το δυαδικό σε σχέση με αυτά.
Ακόμα προβάλλεται ο τρόπος με τον οποίο ο άνθρωπος κατάφερε να εξοικειωθεί με το δυαδικό καθώς επίσης αλλά και ο τρόπος που λειτουργεί σήμερα ο υπολογιστής με βάση αυτό το δυαδικό σύστημα.
Επειδή η αρίθμηση των ανθρώπων είναι διαφορετική από αυτή των υπολογιστών αφού εφαρμόζουν διαφορετικό σύστημα για τις μετρήσεις, παρουσιάζονται αναλυτικά τα βήματα με τα οποία γίνονται οι αριθμητικές πράξεις. Αναφέρεται στο πως γίνεται η ψηφιακή αναπαράσταση εικόνας και ήχου με τη χρήση δυαδικού συστήματος αλλά και στην έννοια των μη προσημασμένων ακεραίων. Δείχνει επίσης τους τρόπους με τους οποίους κωδικοποιούνται οι αρνητικοί προσημασμένοι αριθμοί (παράσταση μέτρου, παράσταση συμπληρώματος ως προς 1 και παράσταση συμπληρώματος ως προς 2), αλλά και τον τρόπο με τον οποίο γίνεται η πρόσθεση προσημασμένων ακεραίων αριθμών ανάλογα με τις περιπτώσεις (Παράσταση πραγματικών αριθμών, παράσταση σταθερής υποδιαστολής και παράσταση κινητής υποδιαστολής).
Εκτός απ’ αυτό στη πτυχιακή προβάλλονται οι βασικές λογικές πράξεις (Άλγεβρα Boole), λογικές πύλες (AND, NOT, OR NAND κ.α.) αλλά και τη μέθοδο δυαδικής κωδικοποίησης χρησιμοποιώντας τους κώδικες με βάρη(Κώδικας BCD) και κώδικες χωρίς βάρη(Κώδικας Gray,Κώδικας υπερβολής κατά 3(excess-3)).
Τέλος αναφέρεται στην εφαρμογή του δυαδικού συστήματος πάνω στα δυαδικά δέντρα αναζήτησης και στην δυαδική αναζήτηση.
Όλα τα παραπάνω χωρίζονται σε κεφάλαια τα οποία είναι τα εξής:

- Κεφάλαιο 1

Το κεφάλαιο αυτό αναφέρει την έννοια των αριθμητικών συστημάτων και χωρίζεται σε δύο τμήματα. Στο πρώτο Παρουσιάζει την ιστορική αναδρομή των αριθμητικών συστημάτων τα οποία χρησιμοποιούνταν στο παρελθόν(λατινικό, ακροφωνικό κ.α.) και στο δεύτερο τα συνηθέστερα αριθμητικά συστήματα που χρησιμοποιούντα στο παρών(οκταδικό ,δεκαδικό κ.α.).

- Κεφάλαιο 2

Έννοια του δυαδικού συστήματος αρίθμησης, ιστορική αναδρομή του, με ποιο τρόπο ο άνθρωπος εξοικειώθηκε με το σύστημα αυτό καθώς επίσης
και οι τρόποι μμετατροπής των δεδομένων από το ένα σύστημα αρίθμησης στο άλλο.

- Κεφάλαιο 3

Επειδή η δεκαδική αρίθμηση που χρησιμοποιούν οι άνθρωποι διαφέρει από την δυαδική αρίθμηση των υπολογιστών, εξετάζονται όλες οι δυαδικές πράξεις όπως αυτές εκτελούνται από τον Η/Υ(πρόσθεση , αφαίρεση, πολλαπλασιασμός, διαίρεση). Δείχνει ακόμα τη μέθοδο με την οποία λειτουργεί ο ηλεκτρονικός υπολογιστής και το δυαδικό σύστημα αρίθμησης και παρατίθενται οι λόγοι για τους οποίους ο Η/Υ χρησιμοποιεί αυτό το σύστημα. Επίσης αναφέρει τον τρόπο που γίνεται η ψηφιακή αναπαράσταση εικόνας και ήχου με τη χρήση δυαδικού συστήματος, τους τρόπους με τους οποίους κωδικοποιούνται οι αρνητικοί προσημασμένοι αριθμοί (παράσταση μέτρου, παράσταση συμπληρώματος ως προς 1 και παράσταση συμπληρώματος ως προς 2), και πως γίνεται η πρόσθεση προσημασμένων ακεραίων αριθμών ανάλογα με τις περιπτώσεις (Παράσταση πραγματικών αριθμών, παράσταση σταθερής υποδιαστολής και παράσταση κινητής υποδιαστολής).

- Κεφάλαιο 4

Στο κεφάλαιο 4 παρουσιάζονται οι βασικές λογικές πύλες(OR, NOT AND, NAND κ.α.), οι βασικές λογικές πράξεις(Άλγεβρα Boole) και η διαδικασία σχεδίασης ψηφιακής λογικής συνάρτησης.

- Κεφάλαιο 5

Αναπαρίστανται οι Δυαδικοί κώδικες με βάρη – Κώδικας BCD και χωρίς βάρη Κώδικας (Gray,Κώδικας υπερβολής κατά 3(excess-3)) που χρησιμοποιούνται για τη δυαδική κωδικοποίηση καθώς και λοιποί άλλοι κώδικες που θα πρέπει να αναφερθούν για τον έλεγχο ή και την διόρθωση λαθών (parity, Hamming).

- Κεφάλαιο 6

Τέλος στο τελευταίο κεφάλαιο αναλύεται η εφαρμογή που έχει το δυαδικό σύστημα πάνω στα δυαδικά δέντρα και δυαδικά δέντρα αναζήτησης.

ΚΕΦΑΛΑΙΟ 1

Συστήματα αρίθμησης

΄΄Για το κεφάλαιο αυτό χρησιμοποιήθηκαν στοιχεία από τις πηγές [11] και [18]΄΄

Κατά τη διάρκεια της ιστορίας, αρκετοί λαοί ανέπτυξαν ένα δικό τους σύστημα αρίθμησης, είτε δανείστηκαν αντίστοιχα άλλων λαών κάνοντας κάποιες τροποποιήσεις σε αυτά. Σ' αυτό το κεφάλαιο θα αναλυθούν τα διάφορα συστήματα αρίθμησης του κάθε λαού από την αρχαιότητα μέχρι το πρόσφατο παρελθόν.

1.1 Εισαγωγή

Ο άνθρωπος από πολύ νωρίς, ασφαλώς από την εποχή που δημιούργησε τα πρώτα ποίμνια από εξημερωμένα ζώα, χρησιμοποίησε σύμβολα για να συμβολίζει αριθμούς: την τελεία ή ένα μικρό ευθύγραμμο τμήμα για το ένα, ένα πρόχειρο σχήμα του χεριού για το πέντε, το σχήμα δύο χεριών για το δέκα (πράγματι το ρωμαϊκό πέντε, V είναι ένα σχηματοποιημένο χέρι, το ρωμαϊκό δέκα, X, δύο σχηματοποιημένα χέρια). Όλα αυτά όμως δεν μπορεί να χαρακτηριστούν ένα κάποιο σύστημα αρίθμησης. Οι αρχαίοι Έλληνες, οι οποίοι, άλλωστε, είναι εκείνοι, που θεμελίωσαν τα μαθηματικά ως πραγματική επιστήμη, χρησιμοποιούσαν για αριθμητικά σύμβολα τα γράμματα του αλφαβήτου τους, αλλά δεν εκπόνησαν ένα κατάλληλο σύστημα για την παράσταση κάθε αριθμού με έναν περιορισμένο αριθμό συμβόλων. Είναι, παρόλα αυτά γεγονός ότι η αρχή της “ θέσης ” ήταν ήδη σε χρήση κατά την κλασσική αρχαιότητα στον άβακα και στα λογιστικά πινάκια, που χρησιμοποιούσαν για τους εμπορικούς λογαριασμούς. Παρά τις τελειοποιήσεις του άβακα στο Μεσαίωνα, έλειπε ένα σύμβολο για το “ μηδέν ”. Το σύμβολο αυτό εισήγαγαν οι Ινδοί και τελειοποίησαν τη χρήση του οι Άραβες, οι οποίοι και μετάδωσαν τη γραφή των αριθμών, που χρησιμοποιείται σήμερα στην Ευρώπη και χαρακτηρίζεται ως αραβική ή ινδική ή αραβοινδική. Παρόλα αυτά χρειάστηκαν 2 αιώνες ($13^{ος}$ – $15^{ος}$) για να επικρατήσει η αραβική αρίθμηση θέσης. Τα αρχαιότερα γραπτά με αριθμούς φέρουν το αποτύπωμα των δέκα δακτύλων (στα αρχαία ιερατικά γραπτά κυρίως των μεσογειακών φυλών οι αριθμοί από το ένα ως το εννιά παριστάνονται μ' αυτόν τον τρόπο).

1.2 Αριθμητικά συστήματα και πολιτισμοί

1.2.1 Οι πρώτες προσπάθειες

Οι Homo Sapiens πριν 300.000 χρόνια είχαν κάνει μια μικρή αρίθμηση με κλαδιά δέντρων και λίγο αργότερα είχαν χρησιμοποιεί κάποιες αριθμητικές εκφράσεις.

Οι κυνηγοί-τροφοσυλλέκτες πριν 70.000-20.000 χρόνια καταλάβαιναν την απλή πρόσθεση , τον πολλαπλασιασμό και την αφαίρεση, ενώ το μοίρασμα της τροφής τους σημαίνει ότι κατανοούσαν και τη διαίρεση. Από ευρήματα της νεολιθικής εποχής, όπως το κόκκαλο "Ισάνγκο" που βρέθηκε στην Αφρική το 20.000 π.Χ. , αποδεικνύεται η χρήση απλών αριθμητικών μεθόδων από τους ανθρώπους της εποχής , κυρίως ως πίνακες θηραμάτων και αποθήκευσης αλλά και για την καταγραφή του χρόνου (φάσεις σελήνης, εποχές).Ο ουρανός, μέσω της κοσμολογίας, έχει ασκήσει ίσως τη μεγαλύτερη επίδραση στην εξέλιξη των μαθηματικών.

1.2.2 Αργότερα

Η χρήση των αριθμητικών συστημάτων από διάφορους πολιτισμούς μπορεί να μας δώσει στοιχεία του τρόπου ζωής , των αναγκών , του πληθυσμού και της εξέλιξής τους. Πρακτικοί είναι και οι λόγοι , εξαιτίας της ανατομίας του ανθρώπινου σώματος, που οδήγησαν στην καθιέρωση του δεκαδικού συστήματος σήμερα ύστερα από πολλές αναζητήσεις. Πεντάδες , δεκάδες και εικοσάδες είναι τα δάχτυλά μας και βοηθούν στη μέτρηση.

1.3 Αριθμητικά συστήματα που έχουν χρησιμοποιηθεί

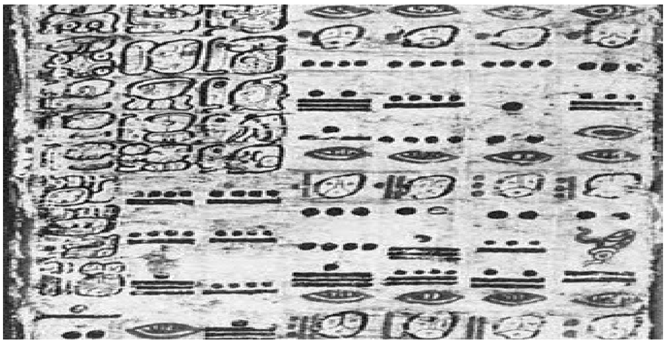
1.3.1 Το εξηνταδικό σύστημα

Στο σύστημα αυτό απαιτούνται 60 απλές μονάδες για να δημιουργήσουν μια μονάδα ανώτερης τάξης , μια εξηντάδα. Με 60 εξηντάδες (δηλαδή 3.600 απλές μονάδες) φτιάχνουμε μια μονάδα επίσης ανώτερης τάξης, μια τρισχιλιοεξακοσάδα κ.ο.κ. Το σύστημα αυτό χρησιμοποιήθηκε από τους Σουμέριους το 2.500 π.Χ. και αργότερα από τους Βαβυλώνιους το 2000 έως το 500 π.Χ. , αλλά και από τους Κινέζους την ίδια περίοδο. Ειδικά οι Βαβυλώνιοι γνώριζαν τις 4 πράξεις , τις ρίζες και έλυναν εξισώσεις πρώτου και δευτέρου βαθμού.
Το αριθμητικό τους σύστημα είχε ως βάση το 60 , ήταν όμως επαναληπτικό και όχι ψηφιακό , αφού οι μονάδες του παριστάνονται με επανάληψη του ίδιου συμβόλου και όχι με διαφορετικά σύμβολα. Χρησιμοποιώντας μόνο δύο σύμβολα: την "σφήνα" και το "καρφί" , αν ήθελαν για παράδειγμα να συμβολίσουν τον αριθμό 5 έγραφαν ισάριθμες σφήνες , ενώ ο αριθμός 19 γραφόταν σαν ένα καρφί και δεξιά του 9 σφήνες.

Επίσης ήταν θεσιακό , που σημαίνει ότι η αξία κάθε αριθμού καθορίζεται από τη θέση του μέσα στον αριθμό.
Τέλος δεν είχαν σύμβολο για το μηδέν , ούτε και υποδιαστολή.
Η μοναδική εφαρμογή του εξηνταδικού συστήματος σήμερα είναι στην καταμέτρηση του χρόνου, όπου η ώρα έχει 60 λεπτά και το λεπτό 60 δευτερόλεπτα.

1.3.2 Το εικοσαδικό σύστημα

Από τα χειρόγραφα της Δρέσδης (σπάνια χειρόγραφα της φυλής των Μάγια) προκύπτει ότι η τελεία και η παύλα, ήταν τα δύο μοναδικά σύμβολα του αριθμητικού συστήματος των Μάγια. Η τελεία αντιπροσώπευε τη μονάδα και η παύλα την πεντάδα. Συνεπώς, το " ._ " για τη φυλή των Μάγια σήμαινε 15.
Πρόκειται για ένα εξελιγμένο προσθετικό και θεσιακό σύστημα, που χρησιμοποιούσαν κατά τον $3^{ο}$ αιώνα μ.Χ. , όπως προκύπτει από τα χειρόγραφά τους. Αν εμείς σήμερα γνωρίζουμε το δεκαδικό σύστημα, οι Μάγια χρησιμοποιούσαν το εικοσαδικό σύστημα, όπου οι τάξεις τους πήγαιναν από τη μονάδα στην εικοσάδα, στοιβάζοντας τους αριθμούς θεσιακά τον έναν πάνω στον άλλον και διαβάζοντας τον αριθμό με κατεύθυνση από πάνω προς τα κάτω. Με αυτό το τρόπο μπορούσαν να πραγματοποιήσουν πολύπλοκους υπολογισμούς.



Ο αριθμός ‘0’ είναι μια άλλη πρωτοπορία της φυλής των Μάγια. Πρόκειται για το σύμβολο που "βασάνισε" τους λαούς που ασχολήθηκαν με την πρώτη αρίθμηση, και που η αξία και σημασία του προέρχεται από την αναγκαιότητα ύπαρξης και εφεύρεσης ενός συμβόλου, για τις τάξεις που είναι μηδενικές. Συμβόλου, που θα μπορεί να εκπροσωπεί το "καθόλου δεκάδες". Στο δικό μας δεκαδικό σύστημα, αν δεν υπήρχε το μηδέν θα ήταν πολύ πιθανό να μπερδεύαμε για παράδειγμα, το 103 με το 13.

Η φυλή των Μάγια, ανεπηρέαστη από τους Βαβυλώνιους και άλλους λαούς που έκαναν τα πρώτα τους βήματα στην ανακάλυψη ενός αριθμητικού συστήματος που γεννούσε η αναγκαιότητα για γραπτούς πρακτικούς υπολογισμούς, υιοθέτησαν εύκολα σύμβολα, για να γεμίσουν το κενό στις αριθμητικές τους πράξεις. Χρησιμοποίησαν μορφές ματιών, κογχυλιών και άλλων μορφών, για ν' αντιμετωπίσουν το πρόβλημα των "άδειων τάξεων" στους υπολογισμούς τους. Και το υιοθέτησαν εύκολα, γιατί δεν ήταν κάτι νέο γι' αυτούς.

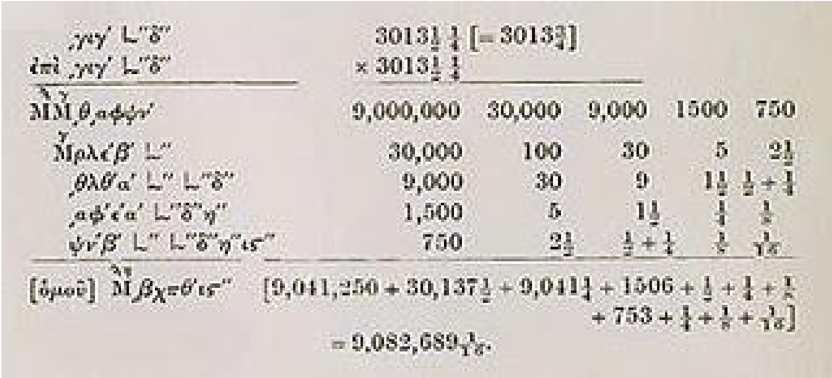
1100 χρόνια πριν από τους Βαβυλώνιους, οι ιερείς-αστρονόμοι πρόγονοι των Μάγια, είχαν αντιμετωπίσει το ίδιο πρόβλημα για πρώτη φορά το 3113 π.Χ. στη προσπάθειά τους ν' αναπτύξουν ένα σύστημα ιερογλυφικών για τη καταγραφή του παρελθόντα χρόνου στις ιερές ημερολογιακές τους στήλες(σύστημα, του οποίου τα σύμβολα αντιπροσώπευαν τα χρονικά διαστήματα, ως πολλαπλάσια των ημερών).

Το ιερογλυφικό αυτό σύστημα δεν είχε σχέση με την αριθμητική, αλλά με την αλληλουχία των χρονικών περιόδων που, σύμφωνα με τη θρησκεία των Μάγια, η κάθε μια προερχόταν από έναν θεό, φορέα και υπεύθυνο για την περίοδο αυτή. Ονόματα όπως "ΚΙΝ", "ΟΥΙΝΤΑΛ", "ΤΟΥΝ" κ.ά. ήταν ονόματα θεών που ήταν υπεύθυνοι για συγκεκριμένη χρονική περίοδο, στους οποίους προσέφεραν θυσίες προκειμένου να τους ευχαριστήσουν και να πάει καλά η χρονιά.

1.3.3 Ελληνικό σύστημα αρίθμησης

600 π.Χ. – 300 μ.Χ

Τα επιτεύγματα των Ελλήνων, για 1000 χρόνια επισκιάζουν όλα τα πνευματικά επιτεύγματα των επόμενων 1500 ετών. Χρησιμοποιούσαν δύο είδη αριθμητικών συστημάτων με βάση το 10: το Ηρωδιανό ή Αττικό και το Ιωνικό ή Αλεξανδρινό. Δε χρησιμοποιούσαν τιμές θέσης όπως έκαναν οι Βαβυλώνιοι και όπως γίνεται σήμερα. Επίσης δε χρησιμοποιούσαν το μηδέν και τα κλάσματα.



͵γιγʹ ∟ʹʹδʹʹ	$3013\frac{1}{2}\,\frac{1}{4}$ [= $3013\frac{3}{4}$]				
ἐπὶ ͵γιγʹ ∟ʹʹδʹʹ	× $3013\frac{1}{2}\,\frac{1}{4}$				
M M ͵θ͵αφψνʹ	9,000,000	30,000	9,000	1500	750
Mρλεʹβʹ ∟ʹʹ	30,000	100	30	5	$2\frac{1}{2}$
͵θλθʹαʹ ∟ʹʹ ∟ʹʹδʹʹ	9,000	30	9	$1\frac{1}{2}$	$\frac{1}{2}+\frac{1}{4}$
͵αφʹεʹαʹ ∟ʹʹδʹʹηʹʹ	1,500	5	$1\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$
ψνʹβʹ ∟ʹʹ ∟ʹʹδʹʹηʹʹιϛʹʹ	750	$2\frac{1}{2}$	$\frac{1}{2}+\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$
[ὁμοῦ] M ͵βχπθʹιϛʹʹ	[$9,041,250 + 30,137\frac{1}{2} + 9,041\frac{1}{4} + 1506 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + 753 + \frac{1}{4} + \frac{1}{8} + \frac{1}{16}$] $= 9,082,689\frac{1}{16}$.				

Πολλαπλασιασμός. Αριστερά αρχαίο Ελληνικό σύστημα , δεξιά σημερινή γραφή.

Οι αρχαίοι Έλληνες χρησιμοποιούσαν γράμματα αντί για αριθμούς, κατά τέτοιο τρόπο ώστε να μπορούν να κάνουνε πολύπλοκους υπολογισμούς με απόλυτη ακρίβεια. Τα ψηφία 1, 2, 3, ... που συνηθίζουμε σήμερα ακόμα δεν είχαν ανακαλυφτεί, αφού πρώτοι τα εφάρμοσαν οι μεταγενέστεροι Άραβες.

Έγραφαν όλους τους αριθμούς από το 1 ως το 999 με γράμματα του αλφαβήτου και με την βοήθεια σημείων στίξεως, τα οποία ήταν:

i. «΄» η κεραία επάνω και μετά από το γράμμα,
ii. «,» η ανάποδη κεραία κάτω και πριν από το γράμμα,
iii. «.» η τελεία μεταξύ των γραμμάτων
iv. «¨» τα διαλυτικά επάνω από το γράμμα.

Χρησιμοποιούνται και κεφαλαία, κυρίως για δυναστικά ονόματα και κεφάλαια βιβλίων. Έτσι έχουμε:

i. α΄ β΄ γ΄ δ΄ ε΄ □ ζ΄ η΄ θ΄ τους αριθμούς 1 2 3 4 5 6 7 8 9 αντίστοιχα.
ii. ι΄ κ΄ λ΄ μ΄ ν΄ ξ΄ ο΄ π΄ □ τούς αριθμούς 10 20 30 ... 90 αντίστοιχα.
iii. ρ΄ σ΄ τ΄ υ΄ φ΄ χ΄ ψ΄ ω΄ □ τούς αριθμούς 100 200 300 ... 900 αντίστοιχα.

Το □ χρησιμοποιείτο ως έξι στην αρχαιότητα. Αντικαταστάθηκε από το στίγμα σταδιακά, αφού είχε πάψει πρώτα να χρησιμοποιείται ως γράμμα. Τις τελευταίες δεκαετίες το στίγμα εξαφανίστηκε από τον γραπτό λόγο για πρακτικούς κυρίως λόγους και τη θέση του πήρε το ΣΤ΄.

Ξεκινώντας από αυτό το σύστημα γραφής, οι πιο σύνθετοι αριθμοί γράφονταν ως σειρά γραμμάτων, έτσι ώστε το άθροισμα να μας δίνει τον συγκεκριμένο αριθμό. Τα γράμματα γράφονταν και διαβάζονταν από τα αριστερά προς τα δεξιά.

Παραδείγματα

i. Ο αριθμός 153 γραφόταν «ρνγ΄» ή «ρνγ».
ii. Ο αριθμός 780 γραφόταν «ψπ΄» ή «ψπ».
iii. Ο αριθμός 306 γραφόταν «τ□ »΄.

Οι χιλιάδες (1000, 2000, κλπ) εκφραζόντουσαν με τα ίδια γράμματα όπως οι εννιά μικροί αριθμοί, είχαν όμως για διακριτικό τον τόνο εμπρός και κάτω του γράμματος.

Παραδείγματα

i. Το «,δ΄» σήμαινε 4.000,
ii. το 1823 γραφόταν «,αωκγ΄»,
iii. το «,αζ΄» σήμαινε 1.007.

Συνοπτικά

Για τους αριθμούς 1-9999 χρησιμοποιούνταν τα εξής γράμματα:

Γράμμα	Αξία	Γράμμα	Αξία	Γράμμα	Αξία	Γράμμα	Αξία
Α΄	1	Ι΄	10	Ρ΄	100	□ Α	1000
Β΄	2	Κ΄	20	Σ΄	200	□ Β	2000
Γ΄	3	Λ΄	30	Τ΄	300	□ Γ	3000
Δ΄	4	Μ΄	40	Υ΄	400	□ Δ	4000
Ε΄	5	Ν΄	50	Φ΄	500	□ Ε	5000
□ ΄	6	Ξ΄	60	Χ΄	600	□ □	6000
Ζ΄	7	Ο΄	70	Ψ΄	700	□Ζ	7000
Η΄	8	Π΄	80	Ω΄	800	□Η	8000
Θ΄	9	□ ΄	90	□ ΄	900	□ Θ	9000

Μυριάδες

Για τους αριθμούς μεγαλύτερους του 9.999 χρησιμοποιούταν ο όρος "μυριάς" ή "μυριάδες", το οποίο υποδηλώνονταν με το γράμμα «Μ» ή την συντόμευση «Μυ», το οποίο προηγείτο του αριθμού, και είχε τα γράμματα από πάνω. Ο Διόφαντος χρησιμοποιούσε για απλούστευση την τελεία, χρησιμοποιώντας τα ίδια γράμματα, και μάλιστα με τρόπο πολύ ανάλογο του σημερινού δεκαδικού συστήματος.

Ο Αρχιμήδης κατόρθωσε με γεωμετρικούς, αλλά και αριθμητικούς υπολογισμούς να εκτιμήσει τον αριθμό των κόκκων άμμου της Γης, πράγμα αφάνταστο για την εποχή του, αφού οι τότε επιστήμονες αρκούνταν να πιστεύουν ότι οι κόκκοι της άμμου είναι «αμέτρητοι». Το έργο του αυτό, με τον τίτλο Ψαμμίτης είναι ορόσημο της μαθηματικής επιστήμης.

Ο Ψαμμίτης ("Άμμου Καταμέτρης") είναι ένα από τα χαρακτηριστικότερα έργα του και αποτελεί υπό μια έννοια την πρώτη επεξηγηματική εργασία.
Προκειμένου να το επιτύχει, έπρεπε πρώτα να επινοήσει ένα σύστημα ονομασίας μεγάλων αριθμών, ώστε να ορίσει ένα άνω όριο· ξεκίνησε λοιπόν από τον μεγαλύτερο αριθμό εκείνης της εποχής, την μυριάδα μυριάδων. Η μυριάς ισούται με 10.000, συνεπώς η μυριάς μυριάδων ισούται με 10.000Χ10.000=100.000.000,εκατό εκατομμύρια. Το σύστημα μέτρησης του Αρχιμήδη φτάνει μέχρι τον αριθμό: " *μυριάδα μυριάδων εις την μυριοστή μυριάδα και όλο εις την μυριοστή μυριάδα*". Ένας άλλος τρόπος να περιγραφεί αυτός ο αριθμός είναι μια μονάδα

ακολουθούμενη από 80 τετράκις εκατομμύρια μηδενικά... Συγκρινόμενο με αυτήν την ποσότητα, το ούτως ή άλλως ασύλληπτα μεγάλο googol (η μονάδα ακολουθούμενη από 100 μηδενικά, 10100) φαντάζει πολύ πενιχρό. Για να έχει ο αναγνώστης μια αίσθηση του μέτρου των μεγεθών, το σύνολο των στοιχειωδών σωματιδίων (πρωτονίων και ηλεκτρονίων) σε όλο το σύμπαν υπολογίζεται κάπου ανάμεσα στο 1070 και 1085, αρκετές τάξεις μεγέθους κάτω από το googol.

1.3.4 Ακροφωνικό σύστημα αρίθμησης

Το πρώτο ελληνικό σύστημα αρίθμησης είναι το ακροφωνικό σύστημα που ήταν εν χρήση στην πρώτη χιλιετία π.Χ. Οι ακροφωνικές αριθμητικές παραστάσεις βρίσκονται πρώτιστα στις αρχαίες ελληνικές επιγραφές από την Αθήνα και άλλες ελληνικές πόλεις - κράτη. Το αττικό ακροφωνικό σύστημα είναι το πιο κοινό και καλά τεκμηριωμένο. Οι ακροφωνικοί χαρακτήρες εμφανίζονται σε έναν μεγάλο αριθμό αρχαίων επιγραφών. Είναι τα πρότυπα σύμβολα που χρησιμοποιούνται για την αντιπροσώπευση του βάρους ή του κόστους.
Στην πρώτη χιλιετία π.Χ δεν υπήρξε ένα ενιαίο ελληνικό εθνικό πρότυπο για το σύστημα αρίθμησης δεδομένου ότι οι διάφορες πόλεις - κράτη ήταν ανεξάρτητα και κάθε πόλης - κράτος είχε το νόμισμα, τα βάρη ,τα μέτρα του κ.λ.π. Αυτό οδήγησε σε μικρές διαφορές στο σύστημα αρίθμησης μεταξύ των διαφορετικών κρατών δεδομένου ότι μια σημαντική λειτουργία ενός συστήματος αρίθμησης στους αρχαίους χρόνους ήταν να αντιμετωπιστούν οι εμπορικές συναλλαγές. Πρέπει να σημειώσουμε ότι οι αρχαίοι Έλληνες είχαν διαφορετικά συστήματα για τους απόλυτους αριθμούς και τους τακτικούς αριθμούς.
«*Ακροφωνικοί*» σημαίνει ότι τα σύμβολα για τους αριθμούς προέρχονται από το πρώτο γράμμα του ονόματος του αριθμού. Εδώ είναι τα σύμβολα για τους αριθμούς 5 ..10 ..100 ..1000, 10000.

Γ	Δ	Η	Χ	Μ
Pente	Deka	Hekaton	Khilioi	Murioi
Πεντε	Δεκα	Ηεκατον	Χιλιοι	Μυριοι
5	10	100	1000	10000

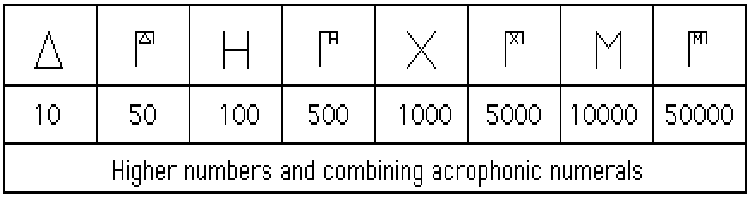
Ακροφωνικοί 5, 10, 100, 1000, 10000.

Το σύμβολο για "το ένα", είναι ένα απλό "|', το οποίο ήταν μια προφανής σημείωση που δεν προέρχεται από το αρχικό γράμμα του αριθμού. Για 5 ..10 ..100 ..1000 ..10000 θα υπάρξει μόνο ένας γρίφος για τον αναγνώστη και αυτό είναι το σύμβολο για 5 που θα έπρεπε να ήταν το π το πρώτο γράμμα του πέντε. Αυτή η διαφοροποίηση είναι απλά μια συνέπεια των αλλαγών στο ελληνικό αλφάβητο αφότου καθορίστηκαν οι αριθμοί που προέρχονται από αυτά τα γράμματα.
Το σύστημα βασίστηκε στην προσθετική αρχή με παρόμοιο τρόπο με τους ρωμαϊκούς αριθμούς. Αυτό σημαίνει ότι 8 είναι απλά V|||, το σύμβολο για πέντε που ακολουθείται από τρία σύμβολα για το ένα.

I	II	III	IIII	Γ	ΓΙ	ΓΙΙ	ΓΙΙΙ	ΓΙΙΙΙ	Δ
1	2	3	4	5	6	7	8	9	10
1 - 10 in Greek acrophonic numbers									

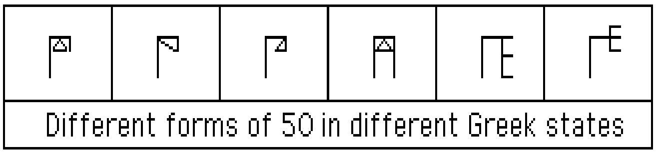
Εδώ οι αριθμοί 1-10 σε ελληνικούς ακροφωνικούς αριθμούς.

Εάν χρησιμοποιείται ένα σύστημα αρίθμησης με βάση το 10 με ένα προσθετικό σύστημα συμβολισμού χωρίς ενδιάμεσα σύμβολα τότε απαιτούνται πολλοί χαρακτήρες για να εκφράσουν ορισμένους αριθμούς π.χ ο αριθμός 9999 θα απαιτούσε 36 σύμβολα σε ένα τέτοιο σύστημα και αυτό είναι πολύ δύσχρηστο. Ήδη έχουμε δει ότι οι ελληνικοί ακροφωνικοί αριθμοί είχαν ένα ειδικό σύμβολο για το 5. Αυτό δεν είναι παράξενο γιατί περικόβει τους χαρακτήρες που απαιτούνται και πιθανώς προκύπτει από τον αριθμό των δαχτύλων. Έχουμε 10 δάχτυλα αλλά υπάρχουν 5 σε κάθε χέρι. Αυτό που είναι πιο παράξενο είναι ότι το σύστημα είχε ενδιάμεσα σύμβολα για τους αριθμούς 50 ..500 ..5000, και 50000 αλλά δεν ήταν νέοι χαρακτήρες, μάλλον σύνθετα σύμβολα που έγιναν από 5 και τα σύμβολα για 10 ..100 ..1000 ..10000 αντίστοιχα. Εδώ σύνθετα διαμορφωμένες μορφές.



Συνδυασμός ακροφωνικών αριθμών.

Εφόσον δεν υπήρξε καμία θεσιακή πτυχή του συστήματος, δεν υπήρξε καμία ανάγκη για μηδέν ως κάτοχος των κενών θέσεων. Το σύμβολο Η αντιπροσώπευσε 100 και κανένα πρόβλημα δεν δημιουργείται στην παράσταση ενός αριθμού που δεν έχει καμία δεκάδα ή μονάδα.
Υπήρξαν διάφορες μορφές συμβόλων για τους αριθμούς. Ήδη έχουν αναφερθεί ότι τα διαφορετικά κράτη χρησιμοποίησαν τις παραλλαγές του συστήματος αρίθμησης. Οι περισσότερες από αυτές τις μορφές είναι παλαιότερες από την κύρια μορφή των αριθμών που έχουμε θεωρήσει ως χαρακτηριστικούς της περιόδου 1500 π.Χ. έως 1000 π.Χ..



Διαφορετικές μορφές του 50 στα διάφορα κράτη πόλεις.

Το επόμενο άξιο λόγου σημείο είναι ότι αυτό το σύστημα αρίθμησης δεν αποτελέστηκε πραγματικά από αφηρημένους αριθμούς με τον τρόπο που σκεφτόμαστε τους αριθμούς σήμερα.
Ξέρουμε ότι οι αρχαίοι Έλληνες είχαν μια κάπως διαφορετική ιδέα επειδή οι αριθμοί χρησιμοποιήθηκαν με ελαφρώς διαφορετικές μορφές που εξαρτώνται από αυτό που ο αριθμός ανάφερε. Η συχνότερη χρήση αυτού του ιδιαίτερου συστήματος αρίθμησης ήταν για τα ποσά των χρημάτων. Η βασική μονάδα των χρημάτων ήταν η δραχμή με μια μεγαλύτερη μονάδα που είναι το τάλαντο αξίας 6000 δραχμών. Η δραχμή υποδιαιρέθηκε στις μικρότερες μονάδες, δηλαδή ο οβολός που ήταν 1 / 6 μιας δραχμής, και ο χαλκός που ήταν 1 / 8 ενός οβολού. Χρησιμοποιήθηκαν επίσης οι μισοί και τα τέταρτα των οβολών . Αυτό το νομισματικό σύστημα δεν βασίστηκε στο δεκαδικό σύστημα αν και το σύστημα αρίθμησης είχε το 10 ως βάση και 5 ως δευτεροβάθμια βάση.

Η μορφή των μονάδων θα έδειχνε τις δραχμές 3807 τα τάλαντα θα γράφονταν όπως:

Οι μονάδες θα εμφανίζονταν τώρα ως Τ (Τ για το τάλαντο).

Ένα ποσό των χρημάτων που περιλαμβάνει και τις δραχμές και τους οβολούς θα γραφόταν όπως: 3807 δραχμές και 3 οβολοί:

Το ακροφωνικό σύστημα χρησιμοποιήθηκε περισσότερο για τα χρήματα. Ένα πολύ παρόμοιο σύστημα χρησιμοποιήθηκε επίσης για τα βάρη και τα μέτρα. Αυτό δεν είναι παράξενο δεδομένου ότι η αξία των χρημάτων θα είχε εξελιχθεί από ένα σύστημα βαρών. Αυτό επιβεβαιώνεται από το γεγονός ότι η δραχμή ήταν επίσης το όνομα της μονάδας του βάρους.

1.3.5 Το Αρμενικό σύστημα αρίθμησης

Το Αρμενικό σύστημα αρίθμησης είναι ένα ιστορικό αριθμητικό σύστημα που δημιουργήθηκε χρησιμοποιώντας τα κεφαλαία γράμματα του Αρμενικού αλφαβήτου. Δεν υπήρχε σύμβολο για το μηδέν στο παλιό σύστημα, και οι αριθμητικές αξίες των γραμμάτων προστίθενται για να εξαχθεί ο τελικός αριθμός.

Οι αρχές που διέπουν αυτό το σύστημα είναι οι ίδιες με του ελληνικού συστήματος αρίθμησης και του Εβραϊκού συστήματος αρίθμησης. Στη σύγχρονη Αρμενία χρησιμοποιούνται οι γνωστοί αραβικοί αριθμοί. Οι Αρμενικοί αριθμοί χρησιμοποιούνται βασικά όπως οι Ρωμαϊκοί αριθμοί. Για παράδειγμα, Գարեգին Բ. σημαίνει Garegin II και Գ. գլուխ σημαίνει Κεφάλαιο III. Επειδή μερικοί browsers δεν υποστηρίζουν χαρακτήρες του Αρμενικού αλφαβήτου, δίνεται και η μεταγραφή σε Λατινικό αλφάβητο.

Σημειώνεται πως τα τελευταία δυο γράμματα του Αρμενικού αλφαβήτου, τα "o" (Օ) και "fe" (Ֆ) προστέθηκαν στο αλφάβητο μετά την έναρξη της χρήσης των αραβικών αριθμών, για να διευκολύνουν τη μεταγραφή λέξεων από άλλες γλώσσες. Έτσι, δεν τους αποδόθηκε κάποια αριθμητική αξία.

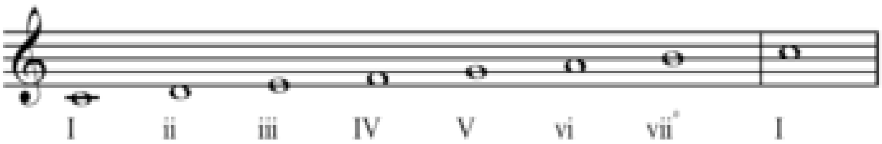
1.3.6 Λατινικό σύστημα αρίθμησης

Units	I	II	III	IV	V	VI	VII	VIII	IX
Tens	X	XX	XXX	XL	L	LX	LXX	LXXX	XC
Hundreds	C	CC	CCC	CD	D	DC	DCC	DCCC	CM
Thousands	M	MM	MMM	$\overline{IV}$	$\overline{V}$	$\overline{VI}$	$\overline{VII}$	$\overline{VIII}$	$\overline{IX}$

Τα λατινικά ψηφία απορρέουν από το σύστημα αρίθμησης της αρχαίας Ρώμης και χρησιμοποιούνται μέχρι και σήμερα (όπως και τα αντίστοιχα ελληνικά). Οι δέκα πρώτοι λατινικοί αριθμοί είναι οι εξής:

I, II, III, IV, V, VI, VII, VIII, IX και X.

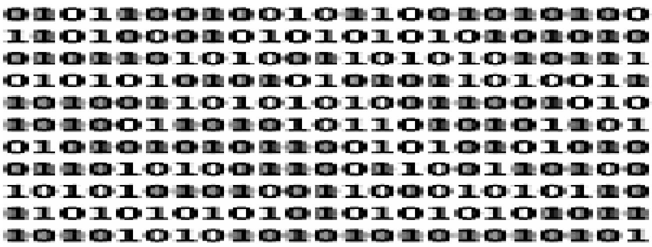
Είναι δεκαδικό σύστημα , μη θεσιακό(επαναληπτικό δηλαδή) και δεν έχει το 0.
Σήμερα τους συναντάμε συνήθως σε αριθμημένες λίστες (όπως είναι το περίγραμμα ενός άρθρου), σε ρολόγια, σε σελίδες πριν το κυρίως σώμα του βιβλίου, στους μήνες του χρόνου, σε διαδοχές αξιωμάτων , σε παιδιά με τα ίδια ονόματα , αλλά και στην μουσική.



Όλα τα παραπάνω συστήματα περιλαμβάνουν την πενταδική, δεκαδική και εικοσαδική αρίθμηση που λόγω πρακτικών ευκολιών στους υπολογισμούς, χρησιμοποιήθηκαν αλλά χρησιμοποιούνται και σήμερα στην καθημερινότητα μας. Υπάρχουν όμως και συστήματα αρίθμησης που έχουν ως βάση το 2 ή και δυνάμεις του.

1.4 Συνηθεστέρα Αριθμητικά Συστήματα

Τα πιο διαδεδομένα είναι το δεκαδικό, το δεκαεξαδικό , το οκταδικό και το δυαδικό.
Τα συστήματα αυτά χρησίμευσαν κυρίως στην μεταφορά και επεξεργασία δεδομένων στους ηλεκτρονικούς υπολογιστές. Ειδικά το δυαδικό σύστημα με μόνο δύο σύμβολα το 0 και το 1 , έχει τεράστια εφαρμογή σε όλα τα ηλεκτρονικά κυκλώματα τα οποία μπορούν να βρίσκονται μόνο σε δύο καταστάσεις: ανοιχτό - κλειστό, αληθές - ψευδές , αγωγή ρεύματος - διακοπή ρεύματος.



Παραπάνω αναφέρονται μερικά παραδείγματα **κωδικοποίησης** των αριθμών από διαφορετικούς πολιτισμούς. Η κωδικοποίηση των αριθμών εξελίχθηκε για τις ανάγκες του κάθε πολιτισμού για να αναπαρασταθούν χρονολογίες, μονάδες μέτρησης αγαθών, χρηματικές συναλλαγές κλπ. Τελικά καθιερώθηκε η χρήση των γνωστών αραβικών αριθμών και το δεκαδικό σύστημα αρίθμησης με 10 ψηφία: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
Για την ευκολότερη και ταχύτερη επεξεργασία των διαφόρων πληροφοριών, οι υπολογιστές χρησιμοποιούν αριθμητικά συστήματα διαφορετικά από το γνωστό μας δεκαδικό (decimal) σύστημα και κυρίως το δυαδικό (binary).
Στον παρακάτω πίνακα φαίνονται τα ψηφία αυτών των αριθμητικών συστημάτων.

Δεκαδικό (βάση 10)	Δυαδικό (βάση 2)	Οκταδικό (βάση 8)	Δεκαεξαδικό (βάση16)
00	0000	00	0
01	0001	01	1
02	0010	02	2
03	0011	03	3
04	0100	04	4
05	0101	05	5
06	0110	06	6
07	0111	07	7
08	1000	10	8
09	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

1.4.1 Δεκαδικό Σύστημα

Ως Δεκαδικό χαρακτηρίζεται το σύστημα αρίθμησης που έχει ως βάση του τον αριθμό δέκα και είναι αυτό το οποίο χρησιμοποιείται σήμερα. Κάθε μονάδα που ανήκει σε μια ορισμένη τάξη, αποτελείται από 10 μονάδες της αμέσως κατώτερης τάξης. Η αρίθμηση είναι θέσης, γιατί ένα ψηφίο ανάλογα με τη θέση που έχει στην παράσταση ενός αριθμού, έχει διαφορετική αξία. Έτσι στον αριθμό 453 ο αριθμός 3 δηλώνει τρεις μονάδες, ο αριθμός 5 δηλώνει πέντε δεκάδες, ενώ ο 4 δηλώνει τέσσερις εκατοντάδες.

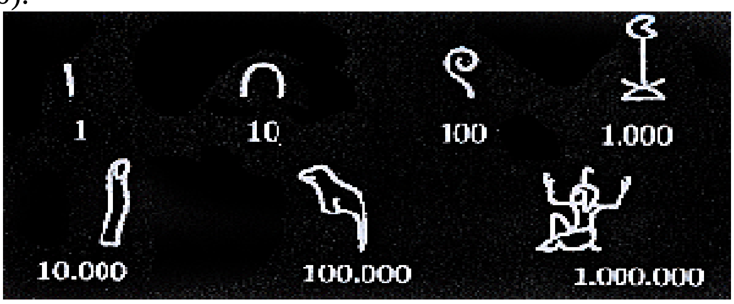
Ιστορία δεκαδικού συστήματος

Επεκτείνοντας την έννοια του δεκαδικού συστήματος, μπορούμε να πούμε ότι έτσι καλείται ένα σύστημα μέτρων και σταθμών, στο οποίο τόσο τα πολλαπλάσια, όσο και τα υποπολλαπλάσια των κυρίων μονάδων είναι δεκαδικές δυνάμεις των βασικών αυτών μονάδων.

Η τάση του ανθρώπου να χρησιμοποιεί το δεκαδικό σύστημα ανάγεται στους προϊστορικούς ακόμα χρόνους· σημαντικό ρόλο σ` αυτό έπαιξε η ύπαρξη δέκα δαχτύλων στα χέρια του ανθρώπου.

i. 5000-332 *π.Χ*

Οι Αιγύπτιοι χρησιμοποιούν σύστημα αριθμών με βάση το 10. Το σύστημά τους ήταν δεκαδικό , επαναληπτικό, μη θεσιακό. Η αιγυπτιακή αρίθμηση διέθετε ένα ειδικό ιερογλυφικό σύμβολο για τη μονάδα και τις δυνάμεις του 10 (10, 100, 1.000, 10.000, 100.000, 1.000.000).



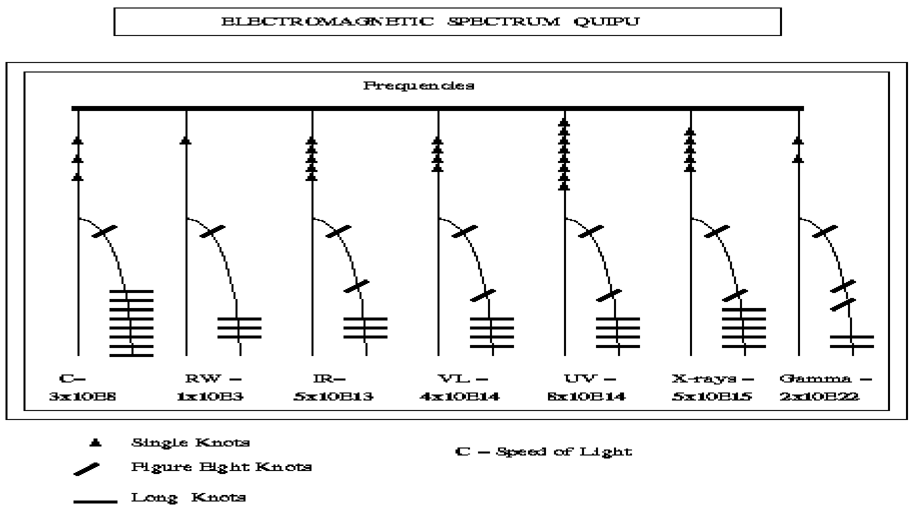
Το ψηφίο της μονάδας ήταν μια μικρή κάθετη γραμμή.
Της δεκάδας έμοιαζε με ένα πέταλο ανάποδα γυρισμένο.
Η χιλιάδα παριστάνεται με ένα λουλούδι λωτού με το κοτσάνι του
Η δεκάδα χιλιάδων με το σχέδιο ενός ανασηκωμένου κυρτού δακτύλου.
Η εκατοντάδα χιλιάδων με ένα βάτραχο ή γυρίνο με ουρά πολύ γερμένη.
Το εκατομμύριο με έναν άνδρα γονατισμένο που σηκώνει τα χέρια στον ουρανό.

ii. 1410-1530 μ.Χ

Οι Ίνκας έφτιαξαν ένα αριθμητικό σύστημα με βάση το 10, για να παρακολουθούν τις καθημερινές δραστηριότητες του μεγάλου πληθυσμού τους (μέσα σε 200 χρόνια είχαν πληθυσμό 6-12.000.000 άτομα, ενώ υπολογίζεται ότι κάποια στιγμή η αυτοκρατορία τους έφτασε και στον αριθμό των 12.000.000 κατοίκων).

Για να υπάρξει σωστή διακυβέρνηση ενός τόσο μεγάλου πληθυσμού είναι σαφές ότι έπρεπε να υπάρχει ένα σύστημα "γραφειοκρατικού" ελέγχου όχι με την σημερινή έννοια του όρου αλλά με την μορφή της καταγραφής των εξουσιών. Καθημερινά οι Ίνκας μετέφεραν και παραλάμβαναν μηνύματα σε μια περιοχή που κάλυπτε σχεδόν όλες τις Άνδεις. Αυτό γινότανε μέσω ενός εκτεταμένου οδικού δικτύου και με την βοήθεια δρομέων.

Το εργαλείο που χρησιμοποιούσαν για την μεταφορά των πληροφοριών πέρα από τον προφορικό λόγο ήταν τα Quipu. Τα Quipu ήταν ένα είδος διακοσμητικού υφαντού φτιαγμένο από χρωματιστά νήματα και επινοήθηκε αρχικά από τους Ινδιάνους. Το μεγαλύτερο μέρος των Quipu καταστράφηκε από Ισπανούς κατακτητές ενώ όσα σώθηκαν βρέθηκαν κλεισμένα σε αεροστεγείς σπηλιές μαζί με μούμιες των Ίνκας.



Σήμερα υπολογίζεται ότι υπάρχουν περίπου 600 Quipu τα οποία βρίσκονται σε διάφορα μουσεία και συλλογές ανά τον κόσμο. Χάρη σε αυτά οι επιστήμονες κατάφεραν να ανακαλύψουν και να μελετήσουν ένα από τα πλέον εντυπωσιακά στοιχεία του πολιτισμού των Ίνκας: το σύστημα μεταφοράς δεδομένων.

Τα Quipu αποτελούνταν από έναν οριζόντιο σπάγκο μήκους περίπου ενός μέτρου, πάνω στον οποίο ήταν δεμένοι κατακόρυφα πολλοί άλλοι σπάγκοι διαφόρων χρωμάτων.

Αρχικά τα Quipu χρησιμοποιούνταν ως αριθμητική μέθοδος με την οποία κατέγραφαν τα διάφορα προϊόντα που μετακινούσαν από περιοχή σε περιοχή.

Ανάλογα με το χρώμα ο κάθε σπάγκος μετρούσε ένα άλλο είδος. Οι λευκοί σπάγκοι αντιστοιχούσαν στο μαλλί, οι κίτρινοι στο χρυσό, οι καφετί στους καρπούς. Στην συνέχεια οι Ίνκας άρχισαν να χρησιμοποιούν την μέθοδο αυτή για να καταγράφουν τους φόρους, τις μετακινήσεις πληθυσμών, ακόμη και τον αριθμό των στρατιωτών που υπήρχε σε κάθε περιοχή στον οποίο αντιστοιχούσαν τα κόκκινα νήματα.

Είχαν ανακαλύψει τον δυαδικό κώδικα, 500 χρόνια πριν εφευρεθεί ο ηλεκτρονικός υπολογιστής!
Μια πιο σύγχρονη έρευνα έγινε από τον ανθρωπολόγο του Harvard, καθηγητή Gary Urton, ο οποίος κατάφερε να μελετήσει 450 από τα διασωθέντα Quipu. Τα ευρήματα του, προκαλούν δέος για τον πολιτισμό των Ίνκας: "*Τα κορδόνια και οι κόμποι των Quipu περιέχουν ένα δυαδικό κώδικα παρόμοιο με αυτόν που χρησιμοποιούν σήμερα οι υπολογιστές, ικανό να μεταβιβάσει περισσότερους από 1.500 χαρακτήρες*".

Ο δημιουργός του Quipu κάθε φορά έπρεπε να πάρει μια απόφαση μεταξύ δύο πιθανοτήτων: παραδείγματος χάρη να κρεμάσει το κορδόνι στο μπροστινό ή στο πίσω μέρος του βασικού οριζόντιου σπάγκου ή να δέσει ένα μάλλινο ή ένα βαμβακερό κορδόνι κλπ. Με δεδομένο ότι οι Ίνκας χρησιμοποιούσαν 24 διαφορετικά χρώματα σπάγκων για την δημιουργία ενός Quipu, οι δυνατοί συνδυασμοί που τελικά είχαν σε έναν κώδικα ήταν 1536.

Με απλά λόγια, εάν δεχτούμε πως όλα αυτά είναι γεγονότα, τότε οι Ίνκας είχαν ανακαλύψει έναν δυαδικό κώδικα που τους επέτρεπε να μεταφέρουν περίπου 1536 μονάδες πληροφοριών και ο οποίος μοιάζει πολύ με αυτόν των σύγχρονων ηλεκτρονικών υπολογιστών. Ο Urton υποστηρίζει ότι με το σύστημα αυτό οι Ίνκας όχι μόνο κατέγραφαν αριθμητικές πληροφορίες αλλά είναι πολύ πιθανό να περνούσαν στα Quipu και την ιστορία του πολιτισμού τους.

iii. 3000 π.Χ.-700 μ.Χ

Οι Ινδοί έχουν το δεκαδικό σύστημα αρίθμησης, το οποίο χρησιμοποιείται παγκοσμίως και το οποίο διέδωσαν οι Άραβες. Η πρώτη προσπάθεια εισαγωγής των Ινδοαραβικών αριθμητικών ψηφίων στην Ευρώπη έγινε από τον Φιμπονάτσι(1180-1250 μ.Χ.). Για να τα υιοθετήσουν όμως οι Ευρωπαίοι χρειάστηκαν ακόμα 400 χρόνια.

Ακόμα και στο τέλος του 16ου αιώνα, η αποδοχή των αρνητικών αριθμών, των ρητών αριθμών και του 0 δεν ήταν πλήρης (πολλοί θεωρούσαν το μηδέν δημιούργημα του Διαβόλου). Το σύστημά τους ήταν δεκαδικό, επαναληπτικό και θεσιακό.

Το δεκαδικό μετρικό σύστημα εφαρμόστηκε στη Γαλλία το 1799 για πρώτη φορά στον κόσμο κι είναι έργο του Ναπολέοντα, όταν αυτός ήταν πρώτος ύπατος.

Σε αυτό οι μονάδες των διάφορων τάξεων του ίδιου είδους ποσοτήτων (μήκος, επιφάνεια, όγκος, βάρος) έχουν μεταξύ τους δεκαδική σχέση. Κι ακόμη οι μονάδες των διάφορων ειδών ποσοτήτων (επιφάνεια, όγκος, βάρος) έχουν σαν πρώτη έννοια πάνω στην οποία βασίζονται το μέτρο, δηλ. τη μονάδα μέτρησης του μήκους, όπως αυτή ορίστηκε απ' τη Γαλλική Ακαδημία την εποχή του Ταλλεϋράνδου.

Ο ορισμός αυτός εξισώνει το μέτρο με το 1 δια 10.000.000 του μήκους του τεταρτημορίου του μεσημβρινού της γης ή το 1 δια 40.000.000 του μήκους ολόκληρου του μεσημβρινού της γης.

Πάνω σ' αυτήν τη μονάδα χτίστηκε ένα ολόκληρο σύστημα μονάδων μέτρησης της επιφάνειας, του όγκου και του βάρους.

Σαν μονάδα μέτρησης της επιφάνειας χρησιμοποιούμε το τετραγωνικό μέτρο, που συμβολίζεται ως τ.μ. ή m^2, και είναι η επιφάνεια ενός τετραγώνου που έχει πλευρά ίση μ' ένα μέτρο. Επίσης σαν μονάδα του όγκου χρησιμοποιούμε το κυβικό μέτρο, κ.μ. ή m^3, που είναι ο όγκος ενός κύβου που έχει ακμή ίση μ' ένα μέτρο. Τέλος το χιλιόγραμμο, KGR, είναι η μονάδα βάρους που υποδιαιρείται σε 1000,gr, γραμμάρια.

Το δεκαδικό μετρικό σύστημα εξαπλώθηκε κατά το $19^{ο}$ αιώνα σ' όλες σχεδόν τις ευρωπαϊκές χώρες, με μόνη εξαίρεση τη Μεγάλη Βρετανία που μόλις πριν 3 χρόνια δέχτηκε εξ ολοκλήρου το δεκαδικό μετρικό σύστημα.

Επίσης στην Ελλάδα μέχρι το 1960 σαν μονάδα βάρους χρησιμοποιούταν η οκά (=400 δράμια) και σαν μονάδα επιφάνειας ο πήχης. Αυτές οι δυο μονάδες προέρχονται από το τουρκικό σύστημα μέτρησης και δεν χρησιμοποιούνται πια. Πρέπει να σημειωθεί πως το δεκαδικό μετρικό σύστημα αποτελεί τη βάση και των συστημάτων των νομισματικών μονάδων των περισσότερων χωρών του κόσμου (π.χ. 1 δρχ=100 λεπτά, αλλά και τα πολλαπλάσια της δραχμής: δεκάδραχμο, εκατοντάδραχμο κλπ.).

Το δεκαδικό σύστημα, που χρησιμοποιεί σήμερα η αριθμητική, προέρχεται προφανώς από το σύστημα των "δέκα δακτύλων" που χρησιμοποιούσε, για να μετρήσει ο πρωτόγονος άνθρωπος. Στους υπολογιστές χρησιμοποιείται το δυαδικό σύστημα (με αριθμούς το μηδέν και το ένα) και το δεκαεξαδικό με αριθμούς 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

1.4.2 Οκταδικό σύστημα αρίθμησης

Στο οκταδικό σύστημα αρίθμησης η βάση είναι το 8 και τα ψηφία που χρησιμοποιούμε για αναπαραστήσουμε έναν αριθμό στο οκταδικό είναι τα 0,1,2,3,4,5,6,7. Κάθε ένα ψηφίο ανάλογα με την θέση του αντιπροσωπεύει αντιστοιχεί σε ένα συντελεστή μιας δύναμης του 8.

Για παράδειγμα ο αριθμός 1821 του δεκαδικού στο οκταδικό σύστημα αρίθμησης γράφεται ως $3435_{(8)}$.

Το οκταδικό σύστημα αρίθμησης χρησιμοποιείται στην ψηφιακή τεχνολογία μερικές φορές αντί του δεκαεξαδικού συστήματος εξαιτίας της εύκολης μετατροπής αριθμών από το δυαδικό στο οκταδικό και αντίστροφα. Προτιμάτε αντί του δεκαεξαδικού όταν δεν θέλουμε να χρησιμοποιήσουμε τα έξτρα σύμβολα A,B,C,D,E,F που χρησιμοποιούνται στο δεκαεξαδικό.

Ποιες είναι οι δυνάμεις του 8;

Δυνάμεις	**Υπολογισμός**	**Αποτέλεσμα**
$\mathbf{8^{-3}}$	$=1/8^3=1/512$	=0,001953125
$\mathbf{8^{-2}}$	$=1/8^2=1/64=0{,}125/8$	=0,015625
$\mathbf{8^{-1}}$	$=1/8^1=1/8$	=0,125
$\mathbf{8^{0}}$	$α^0$ όπου $α \neq 0$ μας κάνει 1	=1
$\mathbf{8^{1}}$	$=8^0 \cdot 8=1 \cdot 8$	=8
$\mathbf{8^{2}}$	$=8^1 \cdot 8=8 \cdot 8$	=64
$\mathbf{8^{3}}$	$=8^2 \cdot 8=64 \cdot 8$	=512

Ο παραπάνω πίνακας δίνει μερικές δυνάμεις του οκτώ από το 8^{-3} μέχρι το 8^{3}. Εδώ και πάλι ξεκινάμε από το $8^{0}=1$ βασιζόμενοι στο ότι οποιοσδήποτε αριθμός υψωμένος στην μηδενική δύναμη μας κάνει 1.

1.4.3 Δεκαεξαδικό σύστημα αρίθμησης

Το δεκαεξαδικό αριθμητικό σύστημα, αναφέρεται στο σύστημα με βάση το δεκαέξι. Σε αυτό το σύστημα, υπάρχουν 16 διαφορετικοί χαρακτήρες. Δηλαδή 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E, και F.

Γιατί χρησιμοποιείται αυτό το σύστημα;

Είναι γνωστό ότι κανένα ηλεκτρονικό σύστημα δε χρησιμοποιεί δεκαέξι διαφορετικές τιμές τάσης, όπως τα δυαδικά ηλεκτρονικά κυκλώματα που χρησιμοποιούν 2 διαφορετικές στάθμες. Το δεκαεξαδικό αριθμητικό σύστημα, δεν απαιτείται από τα μηχανήματα. Είναι ευκολία για τους ανθρώπους. Χρησιμοποιείται στον κόσμο των υπολογιστών, μικροϋπολογιστών και μικροεπεξεργαστών σαν τεχνική συντόμευσης.

Αν σκεφτεί κανείς κάποιες από τις κοινές χρήσεις των δυαδικών αριθμών φαίνεται, ότι υπάρχει σοβαρό πρόβλημα λόγω του μεγέθους τους. Για παράδειγμα, πολλοί μικροεπεξεργαστές χρησιμοποιούν λέξη των 8-bits. Δηλαδή, όταν δουλεύουν με δυαδικό αριθμό αυτός έχει 8-bits. Ένας αριθμός των 8-bits, μπορεί να αντιπροσωπεύει μόνο τις δεκαδικές τιμές 0 έως 255. Αυτός ο αριθμός των 8-bits, έχει τόσους χαρακτήρες όσους οποιοσδήποτε δεκαδικός από την τιμή 0 έως 99.999.999.

Προφανώς, ο δυαδικός αριθμός που αντιπροσωπεύει ένα μεγάλο δεκαδικό αριθμό όπως ο 99.999.999 είναι πολύ μακροσκελής. Στην πραγματικότητα, για να αντιπροσωπευθεί ο δεκαδικός αριθμός 99.999.999 στο δυαδικό σύστημα χρειάζονται 27 ψηφία! Τόσο μακροσκελής αριθμός είναι δύσκολο να διαβαστεί. Το δεκαεξαδικό αριθμητικό σύστημα λοιπόν, χρησιμοποιείται σαν μία μέθοδος συντόμευσης, για να ελαττωθεί το μήκος των δυαδικών αριθμών.

Το 16 είναι η τέταρτη δύναμη του 2. Δηλαδή $16=2^{4}$. Κάθε ένας από τους 16 δεκαεξαδικούς χαρακτήρες(0 έως F), μπορεί να παρασταθεί με ένα δυαδικό αριθμό 4-bits. Οι δυαδικοί αριθμοί των 4-bits, είναι 0000 έως 1111. Αυτό σημαίνει, ότι ο κάθε δεκαεξαδικός χαρακτήρας μπορεί να χρησιμοποιηθεί σαν συντόμευση για οποιοδήποτε δυαδικό αριθμό 4-bits.

Επειδή η λέξη δεκαεξαδικός(hexadecimal) είναι μεγάλη, ο όρος hex χρησιμοποιείται συχνά μιλώντας για δεκαεξαδικούς αριθμούς. Η μετατροπή δυαδικού αριθμού σε δεκαεξαδικό είναι απλή. Ο δυαδικός αριθμός, διαχωρίζεται σε ομάδες των 4-bits. Η ομαδοποίηση των δυαδικών αριθμών σε ομάδες των 4-bits τους κάνει ευανάγνωστους, όπως το κόμμα διαχωρίζει πολυψήφιους αριθμούς.

Κεφάλαιο 2

Δυαδικό σύστημα

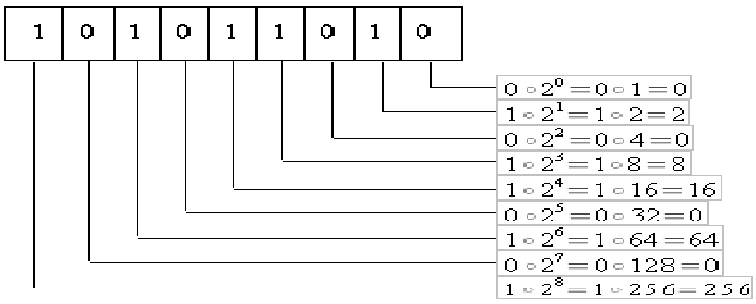
΄΄Για το κεφάλαιο αυτό χρησιμοποιήθηκαν στοιχεία απο τις πηγές [5],[6],[7],[19]΄΄

2.1 Ορισμός δυαδικού συστήματος

*Δυαδικό είναι ένα θεσιακό σύστημα με βάση το δύο(*b=2) και με ψηφία το 0 και το 1. Ένας δυαδικός αριθμός μπορεί να περιέχει και δεκαδικά ψηφία και τότε οι δυνάμεις του 2 είναι αρνητικές, δηλαδή υπάρχουν μισές μονάδες(2^{-1}),τέταρτα μονάδας(2^{-2}), όγδοα μονάδας κλπ. Οι υπολογιστές χρησιμοποιούν το "*δυαδικό*" (binary) σύστημα αρίθμησης σε σχέση με τους ανθρώπους που χρησιμοποιούν το δεκαδικό. Έτσι:

i. χρησιμοποιούνται 2 αριθμοί για συντελεστές αρχίζοντας από το μηδέν, οι οποίοι φυσικά είναι το 0 και το 1. Αυτό είναι και το λεγόμενο ***bit*** που δυνατές τιμές έχει μόνο το μηδέν και το ένα και αποτελεί την μικρότερη ποσότητα πληροφορίας για κάτι.
ii. η θέση κάθε συντελεστή καθορίζει τον εκθέτη της δύναμης (του 2).

*Ένα παράδειγμα ενός δυαδικού αριθμού, του **101011010** είναι:*



Εάν προστεθούν όλα μαζί , το αποτέλεσμα είναι 346. Ουσιαστικά, στον δυαδικό 101011010, το πρώτο (από δεξιά) στοιχείο αναφέρεται στις μονάδες, το δεύτερο στις δυάδες, το τρίτο στις τετράδες κτλ. Στο δυαδικό σύστημα η κάθε τάξη διπλασιάζεται.

Ένα bit αντιπροσωπεύει την ελάχιστη πληροφορία που μπορεί να μεταφέρει ένας αριθμός. Ένας δυαδικός αριθμός των 8 bits ονομάζεται byte (π.χ. 0110 1101), ενώ ένας των 16 bits καλείται word. Το πιο αριστερό ψηφίο του αριθμού ονομάζεται περισσότερο σημαντικό ψηφίο (Most Significant Bit, MSB), γιατί πολλαπλασιάζεται με το μεγαλύτερο συντελεστή, και το πιο δεξιό ψηφίο του αριθμού ονομάζεται λιγότερο σημαντικό ψηφίο (Least Significant Bit, LSB), γιατί πολλαπλασιάζεται με το μικρότερο συντελεστή.

2.1.1 Ομαδοποίηση δυαδικών ψηφίων

i. bit (Binary Digit): η μικρότερη ποσότητα πληροφορίας
ii. byte: Μια ακολουθία 8 δυαδικών ψηφίων:

1 byte= 8 bits

iii. Άλλες μονάδες

- 1 Kilobyte (KB) = 1.024 Bytes
- 1 Megabyte (MB) = 210 KB = 1.048.576 Bytes
- 1 Gigabyte (GB) = 210 MB = 230 Bytes
- 1 Terabyte (TB)= 210 GB

Οι Η/Υ επεξεργάζονται δεδομένα ανά "λέξεις" (words).Εκφράζονται σαν εντολές μηχανής ή μπορεί να παριστάνουν ένα ακέραιο ή πραγματικό αριθμό.

Κάθε λέξη αντιστοιχεί σε ένα συγκεκριμένο αριθμό από bytes.

Κάθε Η/Υ αντιστοιχεί σε ένα χαρακτηριστικό μήκος λέξης (π.χ., λέξη των 32 ή 64 bits)

2.2 Πως ξεκίνησε το δυαδικό σύστημα

Η ιδέα της κωδικοποίησης πληροφοριών με τη χρήση μόνο δύο συμβόλων είναι πανάρχαια. Τα τύμπανα που χρησιμοποιούσαν ορισμένες αφρικάνικες φυλές για να στέλνουν μηνύματα στηρίζονταν σε συνδυασμούς από υψηλές και χαμηλές νότες. Πιο πρόσφατα, ο κώδικας Μορς, στον οποίο τα γράμματα της αλφαβήτου εκπροσωπούνται από ομάδες με τελείες και παύλες, αποτελεί, επίσης κώδικα δύο συμβόλων. Οι ιθαγενείς της Αυστραλίας μετρούσαν κατά δυάδες, ενώ και άλλες ομάδες από κυνηγούς-συλλέκτες της Νέας Γουινέας και της Νότιας Αμερικής αντιμετώπιζαν την αριθμητική με τον ίδιο τρόπο.

Ο κώδικας των δύο συμβόλων δεν είναι η μόνη εναλλακτική λύση απέναντι στο δυαδικό σύστημα. Η αρίθμηση των Βαβυλωνίων στηριζόταν στον αριθμό 60, ενώ η γλώσσα και τα έθιμα των αγγλόφωνων κατακλύζονται από κατάλοιπα του συστήματος με βάση το 12, που κάποτε επικρατούσε στα Βρετανικά Νησιά:12 μήνες σε κάθε χρόνο, 12 ίντσες σε κάθε πόδι, δύο δωδεκάωρα σε κάθε μέρα, η ντουζίνα σαν μέτρο. Τελικά, όμως, το δεκαδικό σύστημα, που δεν είχε άλλη πηγή έμπνευσης από τα δάκτυλα των χεριών ενός ανθρώπου, κατόρθωσε να επισκιάσει κάθε άλλο μέσο αρίθμησης, τουλάχιστον στη Δύση. Ορισμένοι διανοητές της Δύσης, ωστόσο, της περιόδου μετά την Αναγέννηση, γοήτευσαν από την απλότητα του δυαδικού συστήματος αρίθμησης με τις δύο συνθήκες του.

Ένας από τους πρώτους υποστηρικτές του δυαδικού συστήματος ήταν ο μεγαλοφυής Γερμανός Γκότφριντ Βίλχελμ Λάιμπνιτς. Το 1666, όταν βρισκόταν στο τέλος των σπουδών του και πολύ πριν ανακαλύψει τον υπολογιστή με τους οδοντωτούς τροχούς, ο ηλικίας 20 χρόνων Λάιμπνιτς παρουσίασε ένα δοκίμιο που με πολύ μετριοφροσύνη το χαρακτήρισε σχολικό. Με τίτλο "*De Arte Combinatoria*"("Περί της Τέχνης των Συνδυασμών"),η σύντομη εκείνη περιέγραφε μια γενική μέθοδο για την αναγωγή ακρίβειας. Με αυτό τον τρόπο η Λογική(ή, όπως την αποκαλούσε ο ίδιος, οι νόμοι ,της σκέψης) θα μετατοπιζόταν από το βασίλειο λόγου, που βρίθει από αμφιλεγόμενα, στην κυριαρχία των μαθηματικών, όπου οι σχέσεις ανάμεσα στα αντικείμενα ή τις τοποθετήσεις ορίζονται με ακρίβεια.

2.3 Η τελειοποίηση του δυαδικού συστήματος

i. Ο αρχαίος ινδός μαθηματικός Pingala παρουσίασε την πρώτη γνωστή περιγραφή ενός δυαδικού συστήματος αρίθμησης τον 3ο αιώνα π.Χ. , ο οποίος συνέπεσε με την ανακάλυψή της έννοιας μηδενός.

ii. Το σύγχρονο σύστημα δυαδικών αριθμών τεκμηριώθηκε πλήρως από τον Gottfried Leibniz το 17^{o} αιώνα στο άρθρο του ***Explication de l′ Arithemétique Binaire***. Ο Leibniz χρησιμοποίησε το 0 και 1, όπως το σύγχρονο δυαδικό σύστημα αρίθμησης. Οι σύγχρονοι του Gottfried Leibniz, ενοχλημένοι, ίσως, ή και αμήχανοι μπροστά στις ιδέες του, αγνόησαν αυτή την εργασία. Ο ίδιος ο Leibniz από ότι φαίνεται δεν έδωσε συνέχεια στην ιδέα μιας νέας γλώσσας. Δέκα χρόνια αργότερα, όμως, άρχισε να διερευνά τη δύναμη των μαθηματικών κατά ένα νέο τρόπο, όταν συγκέντρωσε την προσοχή του στην τελειοποίηση του δυαδικού συστήματος. Καθώς εργαζόταν σκληρά για τη μετατροπή ατέλειωτων σειρών με αριθμούς από το δεκαδικό στο δυαδικό σύστημα, ο Leibniz συνάντησε ένα αρχαίο χειρόγραφο που του έδωσε ώθηση. Επρόκειτο για ένα σχόλιο πάνω στο σεβάσμιο κινέζικο βιβλίο των Αλλαγών, I Ching, που προσπαθεί να απεικονίσει το σύμπαν στην πολυπλοκότητά του σαν μια σειρά αντιθετικών ζευγών-προτάσεων επιλογής: είτε/είτε-όπως, φως και σκότος, αρσενικό και θηλυκό. Ενθαρρυμένος από αυτή τη φαινομενική επιβεβαίωση των δικών του μαθηματικών ιδεών, ο Leibniz προχώρησε στην τελειοποίηση και τυποποίηση των ατέλειωτων συνδυασμών από μονάδες και μηδενικά που αποτελούν το σύγχρονο δυαδικό σύστημα.

iii. Παρά τη μεγαλοφυΐα του, όμως, ο Leibniz δεν κατόρθωσε να ανακαλύψει οποιαδήποτε άμεση χρησιμότητα για τα προϊόντα των κόπων του. Τον υπολογιστή με τους οδοντωτούς τροχούς τον είχε σχεδιάσει για να χρησιμοποιεί δεκαδικούς αριθμούς και ποτέ δεν τον μετέτρεψε σε δυαδικό σύστημα, ίσως γιατί τον απέτρεψε το μήκος των συνδυασμών ψηφίων που δημιουργεί το δυαδικό. Επειδή χρησιμοποιεί μόνο τα ψηφία 0 και 1, το δυαδικό σύστημα μεταφράζει, παραδείγματος χάρη, τον δεκαδικό αριθμό οκτώ σε 1000,ενώ το ισοδύναμο του δεκαδικό 1000 είναι το δύσχρηστο 1111101000.Αργότερα ο Leibniz ασχολήθηκε με την ιδέα της χρησιμοποίησης των δυαδικών αριθμών σε κάποια υπολογιστική συσκευή, αλλά ποτέ δεν έκανε την προσπάθεια να κατασκευάσει μια τέτοια μηχανή.

iv. Αν η σκέψη ότι ο δυαδικός κώδικας θα μπορούσε να είναι η καθολική γλώσσα του δοκιμίου του, το 1666, πέρασε κάποια στιγμή από το μυαλό του Leibniz, ποτέ δεν διατυπώθηκε ρητά. Ωστόσο, εκατόν είκοσι πέντε χρόνια μετά το θάνατο του Leibniz, το 1716, ένας αυτοδίδακτος Βρετανός μαθηματικός, ο George Boole, έπιασε ορμητικά το νήμα της αναζήτησης μιας καθολικής γλώσσας.

v. Το 1854, ο George Boole δημοσίευσε ένα έγγραφο ορόσημων απαριθμώντας ένα σύστημα της λογικής που θα γινόταν γνωστό ως του Boolean άλγεβρα. Το λογικό σύστημά του αποδείχθηκε οργανικό στην ανάπτυξη του δυαδικού συστήματος, ιδιαίτερα στην εφαρμογή του στα ηλεκτρονικά στοιχεία κυκλώματος.

vi. Το 1937, ο Claude Shannon στη διατριβή του Master του στο MIT εφάρμοσε την του Boolean άλγεβρα και τη δυαδική αριθμητική χρησιμοποιώντας τους ηλεκτρονικούς ηλεκτρονόμους και τις μεταβάσεις για πρώτη φορά στην ιστορία. Με τίτλο «*Μια συμβολική ανάλυση των κυκλωμάτων ηλεκτρονόμων και μετατροπής*», η διατριβή του Shannon έγινε ουσιαστικά το πρακτικό ψηφιακό σχέδιο κυκλωμάτων.

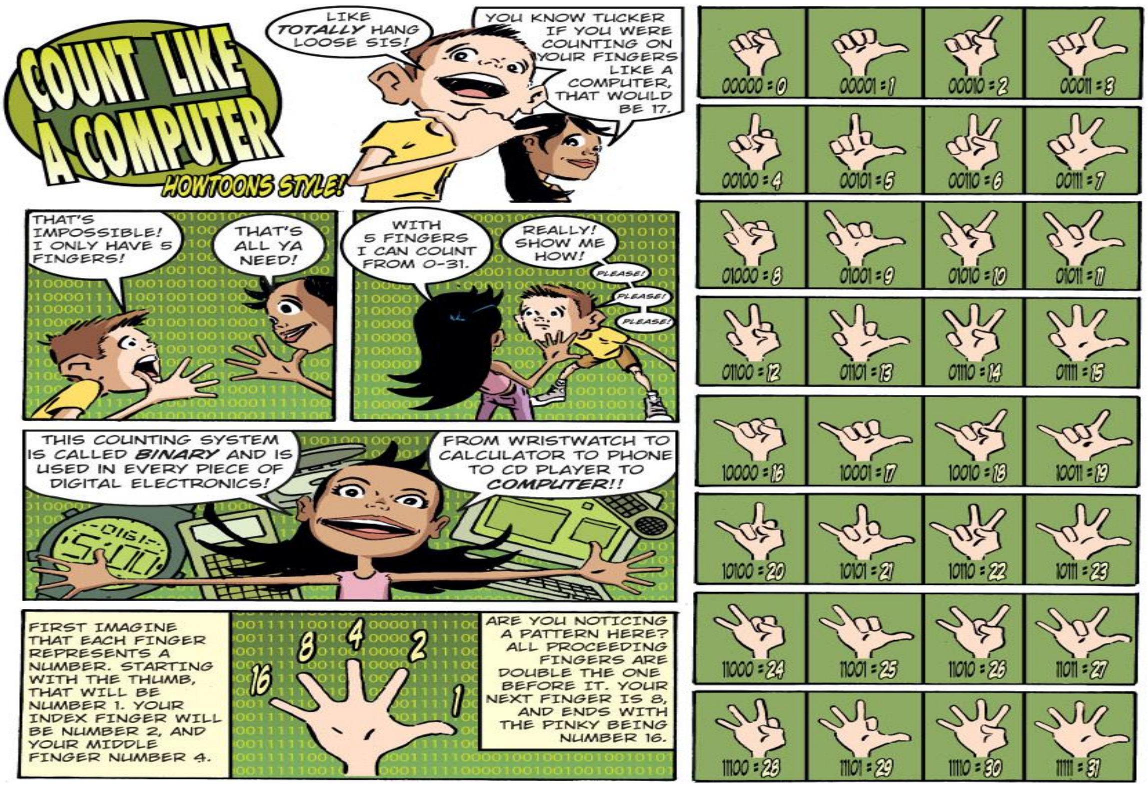
vii. Το Νοέμβριο του 1937, ο George Stibitz, ολοκλήρωσε έναν ηλεκτρονικό υπολογιστή που ονόμασε το «*πρότυπο Κ*» (για «*την κουζίνα*», όπου είχε συγκροτήσει), το οποίο υπολόγιζε χρησιμοποιώντας τη δυαδική προσθήκη.

viii. Σε μια επίδειξη στο American Mathematical Society του κολεγίου Dartmouth στις 11 Σεπτεμβρίου 1940, ο Stibitz ήταν σε θέση να στείλει στο σύνθετο υπολογιστή αριθμού τις μακρινές εντολές μέσω των τηλεφωνικών γραμμών από ένα τηλέτυπο. Ήταν η πρώτη μηχανή υπολογισμού που χρησιμοποιήθηκε από μακριά από μια τηλεφωνική γραμμή. Μερικοί συμμετέχοντες στη διάσκεψη που βεβαίωσαν την επίδειξη ήταν ο John Von Neumann, ο John Mauchly, και ο Norbert Wiener, ο οποίος έγραψε γι' αυτό στα απομνημονεύματά του.

2.4 Εξοικείωση με το δυαδικό σύστημα

Μπορεί να φαίνεται σε πρώτη ματιά το δυαδικό σύστημα αρίθμησης δύσκολο αλλά αυτό οφείλεται απλά στο ότι ο κόσμος δεν έχει εξασκηθεί σε αυτό. Ένας απλός τρόπος εκμάθησης του δυαδικού είναι ο άνθρωπος να χρησιμοποιεί τα δάκτυλα των χεριών του.

Κάθε δάχτυλο αναπαριστά έναν αριθμό. Ο αντίχειρας παίρνει τον αριθμό 1, ο δείκτης τον αριθμό 2, ο μέσος τον αριθμό 4, ο παράμεσος τον αριθμό 8 και το μικρό δάχτυλο τον αριθμό16. Κάθε δάχτυλο έχει διπλάσια αξία από το αμέσως δεξιά του. Έτσι κάνοντας διαδοχικούς συνδυασμούς με τα δάχτυλα των χεριών η μέτρηση φτάνει μέχρι τον αριθμό 32(αφού έχοντας όλα τα δάχτυλα κλειστά εμφανίζεται ο αριθμός μηδέν). Είναι δυνατό να χρησιμοποιηθεί και το δεύτερο χέρι εάν όπως και πριν σε κάθε δάχτυλο αντιστοιχίζονται οι αριθμοί 32,64,128,256,512. Έτσι η μέτρηση φτάνει μέχρι και το 1023 μόνο με τα δάχτυλα του χεριού. Στην εικόνα παρακάτω διακρίνεται μία ιδέα σχετικά με το πως ο άνθρωπος μπορεί να μετρήσει με τα δάχτυλα.



2.5 Μετατροπή αριθμών από ένα αριθμητικό σύστημα σε άλλο

Η μετατροπή ενός αριθμού από ένα αριθμητικό σύστημα με βάση β προς το δεκαδικό σύστημα είναι πολύ απλή:
Υπολογίζεται την τιμή της παράστασης
$\alpha_{m-1}\cdot\beta_{m-1} + \ldots + \alpha^{1}\cdot\beta + \alpha^{0} + \alpha^{-1}\cdot\beta^{-1} + \ldots + \alpha^{-n}\cdot\beta^{-n}$.
Ο δυαδικός αριθμός $\mathbf{10011_{(2)}}$ στο δεκαδικό σύστημα έχει την τιμή $1\cdot2^4 + 0\cdot2^3 + 0\cdot2^2 + 1\cdot2^1+1 = 16+2+1 = 19_{(10)}$.
Ο οκταδικός αριθμός $\mathbf{7123{,}35_{(8)}}$ στο δεκαδικό σύστημα έχει την τιμή $7\cdot8^3 + 1\cdot8^2 + 2\cdot8^1+3+3\cdot8^{-1} + 5\cdot8^{-2} = 3584 + 64 +16 + 3 + 0{,}375 + 0{,}3125 = 3667{,}6875_{(10)}$.
Ο δεκαεξαδικός αριθμός $\mathbf{FC27_{(16)}}$ είναι ισοδύναμος με το δεκαδικό $15\cdot16^3 + 12\cdot16^2+2\cdot16^1+ 7 = 61440 + 3072 + 32 + 7 = 64551_{(10)}$.

Πιο πολύπλοκη είναι η διαδικασία μετατροπής ενός αριθμού από το δεκαδικό σύστημα σε ένα άλλο σύστημα αρίθμησης με βάση β. Η μετατροπή γίνεται χωριστά για το ακέραιο και χωριστά για το κλασματικό μέρος.
Για να μετατραπεί το ακέραιο μέρος του αριθμού Α σε βάση β, γίνονται διαδοχικές διαιρέσεις του ακεραίου μέρους του Α με τον αριθμό β.
Η διαδικασία μετατροπής είναι η εξής:

i. Αρχικά ο νέος αριθμός Χ δεν έχει ψηφία και χρησιμοποιείται μόνο το ακέραιο μέρος του Α.
ii. Διαιρείται ο Α με τη βάση β και παίρνουμε το πηλίκο Π και το υπόλοιπο Υ.
iii. Γράφουμε το Υ στα αριστερά του νέου αριθμού Χ.
iv. Αντικαθιστούμε τον αριθμό Α με το πηλίκο Π.
v. Επαναλαμβάνουμε τα βήματα (2), (3), (4) έως ότου το Α να γίνει 0.

2.5.1 Μετατροπή αριθμού από το οκταδικό στο δεκαδικό σύστημα και αντίστροφα

i. Για να μετατραπεί ένας αριθμός από το οκταδικό σύστημα γραφής στο δεκαδικό προστίθενται τα γινόμενα που προκύπτουν από τον πολλαπλασιασμό κάθε ψηφίου του οκταδικού αριθμού με την αντίστοιχη δύναμη του 8 που αντιστοιχεί σε αυτό το ψηφίο.

Για παράδειγμα ο αριθμός $3435_{(8)}$

$3435_{(8)} =\mathbf{3}\cdot8^3+\mathbf{4}\cdot8^2+\mathbf{3}\cdot8^1+\mathbf{5}\cdot8^0 =\mathbf{3}\cdot512+\mathbf{4}\cdot64+\mathbf{3}\cdot8+\mathbf{5}\cdot1 =1536+256+24+5 =1821$

Παράδειγμα οκταδικού αριθμού με υποδιαστολή

$14{,}53_{(8)}=1\cdot8^1+4\cdot8^0+5\cdot8^{-1}+3\cdot8^{-2}=$

$1\cdot8+4\cdot1+5/8+3/64=8+4+0{,}625+0{,}046875=12{,}671875$

ii. Για να μετατραπεί ένας αριθμός από το δεκαδικό σύστημα αρίθμησης στο οκταδικό η αρχή γίνεται με μια σειρά ακέραιων διαιρέσεων με τον αριθμό 8. Ξεκινώντας με διαιρετέο τον αριθμό που θα μετατραπεί ,το υπόλοιπο που θα προκύψει θα είναι το πρώτο οκταδικό ψηφίο από τα δεξιά. Στην συνέχεια διαιρετέος θα είναι το πηλίκο που βρέθηκε από την προηγούμενη διαίρεση και ξανά διαιρείται με το 8 κ.ο.κ. Η διαδικασία σταματά όταν το πηλίκο γίνει 0 και τα υπόλοιπα που θα προκύψουν θα σχηματίσουν τον αριθμό στο οκταδικό.

Παράδειγμα:

1281:8 δίνει υπόλοιπο 5 και πηλίκο 227

227:8 δίνει υπόλοιπο 3 και πηλίκο 28

28:8 δίνει υπόλοιπο 4 και πηλίκο 3

3:8 δίνει υπόλοιπο 3 και πηλίκο 0 οπότε και σταματά η διαδικασία.

Παρατάσσοντας τα υπόλοιπα , εμφανίζεται ο αριθμός $3435_{(8)}$

2.5.2Μετατροπή οκταδικού αριθμού στο δυαδικό και αντίστροφα

i. Για να μετατραπεί ένας οκταδικός αριθμός στο δυαδικό σύστημα αρίθμησης, κάθε οκταδικό ψηφίο του μετατρέπεται σαν να ήταν του δεκαδικού, σε μια τριάδα δυαδικών ψηφίων. Η μετατροπή γίνεται βάσει του παρακάτω πίνακα:

$0_{(8)}=000_{(2)}$	$4_{(8)}=100_{(2)}$
$1_{(8)}=001_{(2)}$	$5_{(8)}=101_{(2)}$
$2_{(8)}=010_{(2)}$	$6_{(8)}=110_{(2)}$
$3_{(8)}=011_{(2)}$	$7_{(8)}=111_{(2)}$

Με βάση τον προηγούμενο πίνακα η μετατροπή του $3435_{(8)}$= 011 100 011 $101_{(2)}$. Απλώς το 3 γράφεται σαν 011, το 4 σαν 100, το 3 σαν 011 και το 5 σαν 101.

Παρατήρηση: οι οκταδικές τιμές $0_{(8)}$ έως και $7_{(8)}$που έχει ο προηγούμενος πίνακας συμπίπτουν με τα τις αντίστοιχες τιμές 0 έως 7 στο δεκαδικό!

- Σε περίπτωση που ο πίνακας είναι δύσκολο να απομνημονευτεί αρκεί μόνο η μετατροπή των ψηφίων 0 έως 7 από δεκαδικό στο δυαδικό.

  Εναλλακτικά: Γίνεται και χωρίς την μετατροπή με διαιρέσεις αν γίνει η σκέψη το ποιον δεκαδικό αριθμό αντιπροσωπεύει μια τριάδα δυαδικών ψηφίων ... θα είναι ένας ακέραιος αριθμός από το 0 ως το 7, μένει να βρεθεί ποιος.

Έστω Ν ένα ψηφίο του οκταδικού (η δεκαδικού συστήματος που είναι η ίδια περίπτωση) και ψάχνουμε σε ποια τριάδα αντιστοιχεί. Για τον υπολογισμό μπορεί να ακολουθηθεί και η αντίστροφη διαδικασία μιας και οι περιπτώσεις είναι 7 και είναι λίγες. κάθε δυαδικό ψηφίο αντιπροσωπεύει:

$0_{(8)}=000_{(2)}=0+0+0=0$, (το μηδέν είναι μηδέν)

$1_{(8)}=001_{(2)}=0+0+\mathbf{1}=1$, (το 1 είναι δύναμη του 2, μιας και $2^0=1$)

$2_{(8)}=010_{(2)}=0+\mathbf{2}+0=2$, (το 2, είναι δύναμη του 2, μιας και $2^1=2$)

$3_{(8)}=011_{(2)}=0+\mathbf{2}+\mathbf{1}=3$

$4_{(8)}=100_{(2)}=\mathbf{4}+0+0=4$, (το 4, είναι δύναμη του 2, μιας και $2^2=4$)

$5_{(8)}=101_{(2)}=\mathbf{4}+0+\mathbf{1}=5$

$6_{(8)}=110_{(2)}=\mathbf{4}+\mathbf{2}+0=6$

$7_{(8)}=111_{(2)}=\mathbf{4}+\mathbf{2}+\mathbf{1}=7$

Όπως φαίνεται κάθε ψηφίο του δυαδικού αντιστοιχεί σε μια δύναμη του 2. Ο αριθμός 5 γράφεται στο δυαδικό σαν $101_{(2)}$ γιατί ο $101_{(2)}=\mathbf{1}\cdot2^2+\mathbf{0}\cdot2^1+\mathbf{1}\cdot2^0=\mathbf{1}\cdot4+\mathbf{0}\cdot2+\mathbf{1}\cdot1=4+0+1=4+1=5$.

- Δηλαδή αρκεί να βρεθεί ποιες δυνάμεις του 2 πρέπει να προστεθούν έτσι ώστε να εμφανιστεί το 5.
- Άλλος ένας σύμμαχος στο να βρεθεί η τριάδα δυαδικών ψηφίων είναι αν ο αριθμός είναι μονός/περιττός (1,3,5,7), ή ζυγός/άρτιος (0,2,4,6). Αν είναι άρτιος τότε το λιγότερο σημαντικό ψηφίο του θα είναι 0 (το δεξιότερο), αλλιώς αν είναι μονός θα είναι 1.
- Άλλος ένας σύμμαχος στο να βρεθεί η τριάδα είναι πως αν ο οκταδικός αριθμός είναι ένας από τους 1,2,4 τότε είναι δύναμη του 2 οπότε θα είναι μόνο ένα δυαδικό ψηφίο 1 και τα άλλα 0.

Ο τελευταίος τρόπος υπολογισμού είναι λίγο ανάποδος και δεν συνιστάται γενικά. Στην προκειμένη όμως περίπτωση κάποιοι τον χρησιμοποιούν, μιας και αντί να υπολογιστεί η τριάδα με τον κλασικό τρόπο μετατροπής από δεκαδικό στο δυαδικό που θα χρειαζόταν 3 το πολύ διαιρέσεις μπορεί να βρεθεί γρηγορότερα.

ii. Για να γίνει η μετατροπή ένας ενός ακέραιου αριθμού από το δυαδικό στο οκταδικό χωρίζεται σε τριάδες ξεκινώντας *πάντα* από το λιγότερο σημαντικό ψηφίο στα δεξιά. Αν στο τέλος στα αριστερά δεν προκύπτει τριάδα μπορεί να σχηματιστεί προσθέτοντας μηδενικά από αριστερά χωρίς να αλλάξει ο αριθμός. Αφού γίνεται ο διαχωρισμός σε τριάδες, στην συνέχεια μετατρέπεται κάθε τριάδα στο δεκαδικό σύστημα. Κάθε τριάδα θα δίνει ένα ψηφίο του δεκαδικού από το 0 μέχρι το 7 που θα ταυτίζεται με το οκταδικό. Τα ψηφία αυτά που θα προκύψουν στην σειρά θα είναι και ο αριθμός στο οκταδικό.

Παράδειγμα:

Έστω ο αριθμός $11100011101_{(2)}$

Ο δυαδικός αριθμός χωρίζεται σε τριάδες ξεκινώντας πάντα από το λιγότερο σημαντικό ψηφίο στα δεξιά (αυτό των μονάδων) προς τα αριστερά.

$(0)11\ 100\ 011\ 101_{(2)}$

Αν η ομαδοποίηση των ψηφίων γίνει ανά τρία για την τελευταία τριάδα έχουμε μόνο 2 ψηφία. Εκεί θα προστεθεί ένα μηδενικό από αριστερά για να γίνει τριάδα με βάση στο ότι το 11=011 (αντίστοιχα στο δεκαδικό δεν ισχύει 3=03;)

Στην συνέχεια κάθε τριάδα δυαδικών ψηφίων μετατρέπεται στο δεκαδικό σύστημα που θα δώσει ένα ψηφίο από 0 ως το 7.

Το $011_{(2)}$=καμιά τετράδα + 1 δυάδα + 1 μονάδα=2+1=3

Το $100_{(2)}$=μία τετράδα + καμιά δυάδα + καμιά μονάδα=4

Το $011_{(2)}$ έχει υπολογιστεί πιο πάνω και είναι 3

Το $101_{(2)}$=μια τετράδα + καμιά δυάδα + μια μονάδα=5

Αν μπει στην σειρά εμφανίζεται ο αριθμός $3435_{(8)}$

2.5.3 Μετατροπή από οκταδικό σε δεκαεξαδικό και αντίστροφα

i. Ένας γρήγορος τρόπος υπολογισμού είναι η μετατροπή αριθμού από το οκταδικό στο δυαδικό και στην συνέχεια από δυαδικό στο δεκαεξαδικό.

Παράδειγμα: $3435_{(8)}$

$=011\ 100\ 011\ 101_{(2)}$ (Μετατροπή κάθε οκταδικού ψηφίου σε 3 του δυαδικού)

$= 0111\ 0001\ 1101_{(2)}$ (Χωρισμός του αριθμού σε 4αδες από δεξιά)

$=71D_{(16)}$ (Μετατροπή κάθε 4αδας δυαδικών ψηφίων στο δεκαεξαδικό)

ii. Ένας γρήγορος τρόπος υπολογισμού είναι η μετατροπή αριθμού από το δεκαεξαδικό στο δυαδικό και στην συνέχεια από δυαδικό στο οκταδικό.

Παράδειγμα:

71D(16)

$=0111\ 0001\ 1101_{(2)}$ (Μετατροπή κάθε δεκαεξαδικού ψηφίου σε 4 ψηφία του δυαδικού)

$=001\ 100\ 011\ 101_{(2)}$ (Χωρισμός του αριθμού σε 3αδες δυαδικών ψηφίων από δεξιά)

$=3435_{(8)}$ (Μετατροπή κάθε 3αδας δυαδικών ψηφίων στο οκταδικό/δεκαδικό)

2.5.4 Μετατροπή αριθμών από το δεκαδικό στο δυαδικό σύστημα και αντίστροφα

Οι υπολογιστές αυτό που καταλαβαίνουν είναι το 0 και 1.Εφόσον λοιπόν ο άνθρωπος ασχολείται με τους υπολογιστές λίγο παραπάνω είναι και επιβεβλημένη η ανάγκη να γνωρίζει το πως γίνεται η μετατροπή ενός αριθμού από την δεκαδική του μορφή στην δυαδική του και αντίστροφα. Αυτό είναι πολύ σημαντικό, ιδιαίτερα στην υποδικτύωση (subnetting) μίας IP διεύθυνσης.

i. Πιο συγκεκριμένα, έστω ο αριθμός 187.Για να μετατραπεί σε δυαδικό ο αριθμός αυτός θα διαιρείται συνεχώς δια δύο μέχρι να βρεθεί πηλίκο 0.Στην συνέχεια τοποθετούνται τα υπόλοιπα που βρέθηκαν στην σειρά ξεκινώντας από το τελευταίο προς το πρώτο και έτσι σχηματίζεται ο δυαδικός του αριθμός.

Στην προκειμένη περίπτωση δηλαδή είναι:

187/2=93 και υπόλοιπο 1

93/2=46 και υπόλοιπο 1

46/2=23 και υπόλοιπο 0

23/2=11 και υπόλοιπο 1

11/2=5 και υπόλοιπο 1

5/2=2 και υπόλοιπο 1

2/2=1 και υπόλοιπο 0

1/2=0 και υπόλοιπο 1

Ο αριθμός λοιπόν 187 στο δυαδικό σύστημα σύμφωνα με τα υπόλοιπα που υπάρχουν από το τελευταίο προς το πρώτο είναι ο εξής: 10111011

ii. Τώρα για να μετατραπεί το 10111011 πάλι στο δεκαδικό σύστημα αρκεί να ληφθεί υπόψη ο παρακάτω πίνακας και να προστεθούν όλα τα 1.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

Δηλαδή στο παράδειγμα προκύπτει ο δυαδικός αριθμός 10111011. Ξεκινάει από 1 και άρα η αρχή θα γίνει με το 128.Ο δεύτερος αριθμός είναι το 0 όπως φαίνεται και έτσι δεν θα χρησιμοποιηθεί το 64 που είναι ο αμέσως επόμενος αριθμός αλλά το 32 γιατί και πάλι είναι 1 (μόνο τα 1 προστίθενται με βάση τον πίνακα).

Οπότε μέχρι στιγμής είναι 128+32=160

Μετά δύο άσους οπότε είναι 16+8=24

Μετά τα πρώτα 5 bits από τα 8 που έχει ο δυαδικός μας αριθμός προκύπτει ο αριθμός 184 (160+24).Μένουν άλλα 3 bits. Φαίνεται ότι το επόμενο bit είναι το 0 οπότε και δεν προστίθεται τίποτα αλλά μετακινείται πιο δεξιά του πίνακα. Απομένουν το 1 και το 1 τα οποία είναι τα 2 τελευταία bit και σχηματίζουν τον αριθμό 3 (2+1).

Συνοψίζοντας τις εξής προσθέσεις προκύπτει το εξής:

128+32+16+8+2+1=187

Το σημαντικό λοιπόν στην μετατροπή ενός αριθμού από το δυαδικό στο δεκαδικό σύστημα είναι τα παρακάτω:

i. Γνώση του παραπάνω πίνακα.
ii. Προστίθενται μόνο τα 1 που υπάρχουν.
iii. Κάθε φορά που υπάρχει 0 γίνεται μετακίνηση μιας θέσης προς τα δεξιά από τον πίνακα (στον επόμενο αριθμό δηλαδή)

2.5.5 Μετατροπή μεταξύ δυαδικού και δεκαεξαδικού και αντίστροφα

Η μετατροπή από το δυαδικό σύστημα στο δεκαεξαδικό είναι άμεση και δεν απαιτεί οποιεσδήποτε πράξεις. Άλλωστε αυτός είναι ο λόγος για τον οποίο εισήχθη η χρήση του δεκαεξαδικού συστήματος.

i. Μετατροπή δυαδικού σε δεκαεξαδικό

Για την μετατροπή ενός αριθμού από το δυαδικό στο δεκαεξαδικό, χωρίζεται ο δυαδικός από τα δεξιά προς τα αριστερά σε ομάδες των τεσσάρων bits, και σε κάθε μία από αυτές αντιστοιχίζεται το δεκαεξαδικό ψηφίο.

Τα παρακάτω δύο παράδειγμα, απεικονίζουν τη μετατροπή των δυαδικών αριθμών σε δεκαεξαδικούς.

Για να μετατραπεί ο δυαδικός αριθμός 10101011111101 σε δεκαεξαδικό αριθμό, θα ομαδοποιηθεί ανά 4 ψηφία:

0010 1010 1111 1101

Σημείωση!

Μπορούν να προστεθούν ανάλογα μη σημαντικά μηδενικά στην αρχή του αριθμού, προκειμένου να γίνουν όλες οι ομάδες μήκους τεσσάρων bits.

Αντικαθιστάται κάθε ομάδα με τον αντίστοιχο δεκαεξαδικό αριθμό.

2 A F D

Επομένως, ο αριθμός 10101011111101_2 είναι ο $2AFD_{16}$

Για τη μετατροπή του 11000111_2 σε δεκαεξαδικό, είναι :

1100 0111

C 7

Δηλαδή ο $11000111_2 = C7_{16}$

Έτσι ο δεκαεξαδικός αριθμός είναι πιο ευανάγνωστος από τον αντίστοιχο δυαδικό του.

Οπότε προκύπτει ότι:

$10101011111101_2 = 2AFD_{16}$ και ότι $11000111_2 = C7_{16}$

Ο δεκαεξαδικός αριθμός είναι πιο ευανάγνωστος και πιο ευκολομνημόνευτος. Επίσης είναι πιο οικείοι και αποτελούν έναν τρόπο απεικόνισης των δυαδικών. Ο μικροεπεξεργαστής δουλεύει με bits, όχι δεκαεξαδικούς χαρακτήρες.

- Η μετατροπή δυαδικών κλασμάτων σε δεκαεξαδικά, ακολουθεί παρόμοιους κανόνες. Τα bits μετά την υποδιαστολή, ομαδοποιούνται ανά 4, ξεκινώντας από το μέγιστο σημαντικό ψηφίο(MSB) του τμήματος (ο πρώτος αριθμός δεξιά της υποδιαστολής). Κάθε ομάδα 4 bits μετατρέπεται στον αντίστοιχο δεκαεξαδικό αριθμό. Αν είναι αναγκαίο, προστίθενται ασήμαντα 0 (στο τέλος του αριθμού).

Για παράδειγμα, η μετατροπή του δυαδικού κλασματικού τμήματος 0,0101101 σε δεκαεξαδικό γίνεται ως εξής:

0, 0101 1010

Τώρα αντικαθιστάται κάθε ομάδα με τον αντίστοιχο δεκαεξαδικό

0, 5 A

Δηλαδή ο $0{,}01011010_2$ είναι ο $0{,}5A_{16}$

Η μετατροπή του αριθμού $1101{,}0111_2$ σε δεκαεξαδικό γίνεται:

1101 . 0111

D . 7

Σημείωση: $13{,}24_{16} = 13{,}24$

Ένας τρόπος για να αποφευχθεί η σύγχυση μεταξύ των δεκαδικών και των δεκαεξαδικών αριθμών είναι να δίνεται προσοχή στο πως διαβάζονται. Για παράδειγμα, όταν διαβάζεται το δεκαεξαδικό $13{,}24_{16}$ είναι ΄΄ *ένα τρία κόμμα δύο τέσσερα* ΄΄, όχι ΄΄ *δεκατρία κόμμα εικοσιτέσσερα* ΄΄.

Όπως φαίνεται από τα παραδείγματα, οι δυαδικοί αριθμοί συμπτύσσονται σημαντικά όταν μετατραπούν σε δεκαεξαδικούς. Για έναν αριθμό 8-bits που συχνά χρησιμοποιείται στους μικροεπεξεργαστές, η περιοχή αριθμών είναι από 0000 0000_2 έως 1111 1111_2. Αυτό αντιστοιχεί με την περιοχή 00_{16} έως FF_{16} .

ii. Μετατροπή δεκαεξαδικού σε δυαδικό

Η περίπτωση αυτή είναι η αντίστροφη της προηγούμενης, δηλαδή σε κάθε δεκαεξαδικό ψηφίο αντιστοιχίζονται τα 4 δυαδικά ψηφία σύμφωνα με τον πίνακα παραπάνω.

Παράδειγμα: $2D5_{(16)}$ = $???_{(2).}$

2	D	5
0010	1101	0101

Άρα $2D5_{(16)}$ = $001011010101_{(2)}$.

2.5.6 Μετατροπές μεταξύ δεκαδικού και δεκαεξαδικού

Η περίπτωση αυτή παρουσιάζεται σπάνια κατά την ενασχόλησή μας με τους υπολογιστές. Αναφέρεται όμως εδώ, για να δειχτεί η ομοιότητα των διαδικασιών μετατροπής μεταξύ των διαφόρων συστημάτων αρίθμησης.

i. Μετατροπή δεκαεξαδικού σε δεκαδικό

Η μετατροπή δεκαεξαδικού σε δεκαδικό γίνεται ως εξής:

Παράδειγμα: $2D5_{(16)}$ = $????_{(10)}$.

$2D5_{(16)} = 2\times16^2 + 13\times16^1 + 5\times16^0 = 512 + 208 + 5 = 725_{(10)}$.

ii. Μετατροπή δεκαδικού σε δεκαεξαδικό

Η διαδικασία για την μετατροπή ενός DEC σε HEX, είναι αντίστοιχη εκείνης της μετατροπής δεκαδικού σε δυαδικό, και γίνεται με διαδοχικές διαιρέσεις των πηλίκων διά 16.

Παράδειγμα: $725_{(10)}$ = $???_{(16)}$.

725/16=45 υπόλοιπο 5
45/16=2 υπόλοιπο 13
2/16=0 υπόλοιπο 2
Άρα $725_{(10)}$ = $2D5_{(16)}$.

2.6 Δυαδική Αριθμητική

Μια από τις πιο κοινές λειτουργίες που ζητείται από ένα μικροεπεξεργαστή είναι να εκτελέσει απλή αριθμητική-συχνά, πολλές φορές και πολύ γρήγορα. Όμως ο μικροεπεξεργαστής δεν μπορεί να εκτελέσει δεκαδικές πράξεις, διότι πρέπει να χρησιμοποιεί δυαδικούς αριθμούς. Επειδή ο μικροεπεξεργαστής χρησιμοποιεί δυαδική αριθμητική, είναι σημαντικό να κατανοηθούν οι διαδικασίες που χρησιμοποιούνται για την πρόσθεση, αφαίρεση, πολλαπλασιασμός και διαίρεση και επίσης να κατανοηθεί ο τρόπος με τον οποίο χειρίζεται τους μεγάλους αριθμούς και αρνητικούς αριθμούς.

2.6.1 Δυαδική πρόσθεση

Η δεκαδική και η δυαδική πρόσθεση, εκτελούνται με τον ίδιο τρόπο. Και στα δύο αριθμητικά συστήματα, όταν προστεθούν δύο αριθμοί η αρχή γίνεται από την πρώτη από τα δεξιά στήλη. Αυτή η στήλη, αντιπροσωπεύει τον ελάχιστα σημαντικό αριθμό. Όποιο αποτέλεσμα είναι μεγαλύτερο από μονοψήφιο, δημιουργεί κρατούμενο. Προστίθενται το κρατούμενο στους αριθμούς της επόμενης περισσότερο σημαντικής στήλης. Πράξη σε οποιοδήποτε στήλη, μπορεί να προκαλέσει κρατούμενο.

Στο δεκαδικό αριθμητικό σύστημα, αποτέλεσμα μεγαλύτερο του 9 δημιουργεί κρατούμενο. Στο δυαδικό σύστημα, κρατούμενο δημιουργείται όταν το αποτέλεσμα είναι μεγαλύτερο του 1.

Το παρακάτω σχήμα διακρίνονται οι ομοιότητες και οι διαφορές μεταξύ δυαδικής και δεκαδικής πρόσθεσης. Η ίδια πράξη πρόσθεσης, φαίνεται πρώτα στο δεκαδικό και στη συνέχεια στο δυαδικό σύστημα.Στην δεκαδική πράξη, προσθέτοντας το 95 στο 99. Ο προσθετέος είναι το 9 στο 5 στην δεξιά στήλη, με αποτέλεσμα 4 και κρατούμενο 1. Το κρατούμενο φαίνεται στο σχήμα σαν ένα επιπλέον 1 πάνω από τη στήλη των δεκάδων. Σε αυτή την στήλη, το κρατούμενο 1 προστίθεται στα 0 της στήλης των εκατοντάδων. Το αποτέλεσμα στη στήλη αυτή είναι 1, και το σύνολο της πρόσθεσης 194.

Δεκαδική πρόσθεση	Όροι	Δυαδική πρόσθεση
11	Κρατούμενο	1111 1110
099	Προσθετέος	0110 0011
095	Προσθέτης	0101 1111
194	Άθροισμα	1100 0010

Σχήμα-1

Όταν το άθροισμα των δύο αριθμών δεν μπορεί να εκφραστεί με ένα ψηφίο παράγεται κρατούμενο που φαίνεται στην επόμενη στήλη σαν μονάδα (1).Παρακάτω φαίνεται ένας πίνακας που συνοψίζει τους κανόνες της δεκαδικής πρόσθεσης. Ο πίνακας αυτός μπορεί να χρησιμοποιηθεί, για να βρεθεί το σύνολο των δύο παρακάτω αριθμών. Επίσης θα μπορούσε να βρεθεί το αποτέλεσμα της πρόσθεσης οποιονδήποτε ψηφίων, βρίσκοντας που διασταυρώνονται η γραμμή (που αντιπροσωπεύει τον προσθέτη) και η στήλη (που αντιπροσωπεύει τον προσθετέο). Εάν, η γραμμή και η στήλη διασταυρώνονται στην σκιασμένη περιοχή, τότε τα δύο ψηφία παράγουν κρατούμενο 1 στη επόμενη πλέον σημαντική στήλη. Στο δεκαδικό αριθμητικό σύστημα, η διαδικασία αυτή είναι τόσο γνωστή, που μπορεί να ξεχαστούν οι κανόνες που χρησιμοποιούνται.

+		Προσθετέος									
		0	1	2	3	4	5	6	7	8	9
Προσθέτης	0	0	1	2	3	4	5	6	7	8	9
	1	1	2	3	4	5	6	7	8	9	
	2	2	3	4	5	6	7	8	9		
	3	3	4	5	6	7	8	9			
	4	4	5	6	7	8	9				
	5	5	6	7	8	9					
	6	6	7	8	9						
	7	7	8	9							
	8	8	9								
	9	9									

Σχήμα-2(α)

Μπορεί η εκτέλεση της δυαδικής αριθμητικής να φαίνεται δύσχρηστη, όμως είναι πραγματικά απλούστερη από τη δεκαδική αριθμητική.

+		Προσθετέος	
		0	1
Προσθέτης	0	0	1
	1	1	0

Σχήμα-2(β)

Το παραπάνω σχήμα συνοψίζει τους κανόνες της δυαδικής αριθμητικής. Μπορεί να βρεθεί το αποτέλεσμα της πρόσθεσης οποιονδήποτε δύο ψηφίων, βρίσκοντας που διασταυρώνονται η γραμμή και η στήλη (που αντιπροσωπεύει τον προσθέτη) και η στήλη (που αντιπροσωπεύει τον προσθετέο). *Εάν η γραμμή και η στήλη διασταυρώνονται στην σκιασμένη περιοχή, τότε τα δύο ψηφία παράγουν κρατούμενο 1 στη επόμενη πλέον σημαντική στήλη*. Όπως φαίνεται η σκιασμένη περιοχή του παραπάνω σχήματος είναι απλούστερη διότι στο δυαδικό σύστημα έχουμε μόνο δύο χαρακτήρες το και το 1.

Τώρα στο Σχήμα 1 προστίθενται τα δυαδικά αντίστοιχα των αριθμών 99 και 95 που είναι 0110 0011 και 01011111. Ξεκινώντας με το ελάχιστα σημαντικό ψηφίο, φαίνεται ότι και ο προσθέτης και ο προσθετέος είναι 1. Από τον πίνακα δυαδικής πρόσθεσης φαίνεται ότι το αποτέλεσμα είναι 0 με κρατούμενο 1.

Το κρατούμενο φαίνεται στο Σχήμα 1 σαν ένα επιπλέον 1 πάνω από την στήλη των δυάδων. Σε αυτή τη στήλη το κρατούμενο 1 προστίθενται στο 1 συν 1. Προστίθενται τρείς 1.

Πρόσθεση τριών δυαδικών ψηφίων

Η διαδικασία πρόσθεσης τριών ψηφίων φαίνεται στο Σχήμα 3. Εκεί η πρόσθεση των τριών 1, μπορεί να χωριστεί σε δύο ανεξάρτητες προσθέσεις δύο ψηφίων. Οποιαδήποτε δυαδική πρόσθεση συνοψίζεται στον πίνακα πρόσθεσης. Προσθέτοντας 1 συν 1 έχουμε 0 με κρατούμενο 1. Δηλαδή είναι 0 στη στήλη των μονάδων και 1 στη στήλη των δυάδων ή 10.

1		1
1	=	+1
+1		10
11		+1
	11	

Σχήμα-3

Τώρα προστίθενται 1 συν 10. Αυτό δίνει σαν αποτέλεσμα 1 στην στήλη των μονάδων και καθόλου κρατούμενο για πρόσθεση στο 1 της στήλης των δυάδων. Έτσι το δυαδικό άθροισμα 1 συν 1 είναι 11.

Στο πρόβλημα της μεγαλύτερης πρόσθεσης του Σχήματος 1, το άθροισμα στη στήλη των δυάδων είναι 11, ή 1 κρατούμενο 1. Το κρατούμενο, φαίνεται σαν ένα επιπλέον 1 στη στήλη των τετράδων. Αυτό δίνει 0 με κρατούμενο 1 στη στήλη των οκτάδων. Το κρατούμενο φαίνεται σαν επιπλέον 1 στη στήλη των οκτάδων. Αυτό προστίθεται στο 0 και στο 1 που βρίσκονται ήδη στη στήλη των οκτάδων.

Αυτό δίνει 0 με κρατούμενο 1 στη στήλη των δεκαεξάδων. Το κρατούμενο φαίνεται σαν επιπλέον 1 στη στήλη των δεκαεξάδων και προστίθεται στο 0 και στο 1 που βρίσκονται ήδη στη στήλη των δεκαεξάδων. Αυτό δίνει 0 με κρατούμενο 1 που μεταφέρεται στη στήλη των τριανταδυάδων δίνοντας 0 και κρατούμενο 1 που μεταφέρεται στη στήλη των 64δων. Το μεταφερόμενο 1 προστίθεται στις δύο 1 που είναι στη στήλη των 64δων. Το αποτέλεσμα της πρόσθεσης των τριών 1 όπως φαίνεται είναι 1 με κρατούμενο 1. Το μεταφερόμενο 1 προστίθεται στα δύο 0 που είναι στη στήλη των 128δων δίνοντας 1 στη στήλη αυτή χωρίς κρατούμενο για μεταφορά. Έτσι το δυαδικό άθροισμα των 0110 0011 και 0101 1111 είναι 1100 010.

Αν προστεθούν λίγοι δυαδικοί αριθμοί, διαπιστώνεται ότι η δυαδική πρόσθεση είναι εύκολη. Η μόνη δυσκολία είναι ότι οι δυαδικοί αριθμοί χρειάζονται μεγάλο πλήθος ψηφίων για να εκφράσουν μεγάλους αριθμούς. Επομένως, υπάρχουν πολλές στήλες και πολλά κρατούμενα σε μία δυαδική πρόσθεση. Επίτηδες έγινε η επιλογή αντιπροσώπευσης των δυαδικών αριθμών με 8-bits στα παραδείγματά μας, γιατί πολλοί μικροεπεξεργαστές αντιπροσωπεύουν δυαδικούς αριθμούς σαν λέξεις των 8-bits. Παρόλο που και ο προσθέτης και ο προσθετέος σε αυτό το παράδειγμα δεν απαιτούν 8-bits αυτό έγινε διότι δεν μπορεί να μικραίνει το μέγεθος της λέξης ενός μικροεπεξεργαστή επειδή υπάρχουν ασήμαντα 0 μπροστά από τον αριθμό.

Τα ασήμαντα 0 για τις περισσότερες δυαδικές προσθέσεις δεν λαμβάνονται υπόψη. Δηλαδή όταν γράφεται ένα πρόβλημα στο χαρτί γράφεται μόνο τον αριθμό των ψηφίων που είναι απαραίτητος για να εκφράσει τον αριθμό. Πάντως σε αυτό το πρόβλημα πρέπει να μεταφερθεί κρατούμενο στη στήλη των εκατοντάδων. Για αυτό το σχήμα δείχνει τα δύο ασήμαντα 0. Πρέπει πάντα να υπάρχει στο μυαλό του ανθρώπου ότι τα προπορευόμενα 0 υπάρχουν στην πραγματικότητα.

2.6.2 Δυαδική αφαίρεση

Η δυαδική αφαίρεση εκτελείται με τον ίδιο τρόπο όπως η δεκαδική αφαίρεση. Όπως με τη δεκαδική και δυαδική πρόσθεση, απλά υπάρχει διαφορετική ομάδα κανόνων για τη σύνθεση των αριθμών. Τα Σχήματα 3 (α) και (β), δείχνουν τον πίνακα αφαίρεσης για δεκαδική και δυαδική αριθμητική. Στο σχήμα 3(α), φαίνονται οι κανόνες δεκαδικής αφαίρεσης. Σε αυτό το σχήμα, οποιαδήποτε αφαίρεση στην οποία ο αφαιρετέος είναι μεγαλύτερος του αφαιρέτη, φαίνεται στη σκιασμένη περιοχή. Αυτό σημαίνει, ότι η αφαίρεση προκαλεί δανειζόμενο. Δηλαδή γίνεται δανεισμός μίας μονάδας από την επόμενη πλέον σημαντική στήλη του αφαιρέτη.

-		**Αφαιρετέος**									
		0	1	2	3	4	5	6	7	8	9
Αφαιρέτης	0	0	9	8	7	6	5	4	3	2	1
	1	1	0	9	8	7	6	5	4	3	2
	2	2	1	0	9	8	7	6	5	4	3
	3	3	2	1	0	9	8	7	6	5	4
	4	4	3	2	1	0	9	8	7	6	5
	5	5	4	3	2	1	0	9	8	7	6
	6	6	5	4	3	2	1	0	9	8	7
	7	7	6	5	4	3	2	1	0	9	8
	8	8	7	6	5	4	3	2	1	0	9
	9	9	8	7	6	5	4	3	2	1	0

Σχήμα 3(α)

-		**Αφαιρετέος**	
		0	1
Αφαιρέτης	0	0	1
	1	1	0

Σχήμα 3(β)

Αποτελέσματα στις σκιασμένες περιοχές σημαίνει ότι υπάρχει δανειζόμενο

Δεκαδική αφαίρεση	Όροι	Δυαδική αφαίρεση
1	Δανειζόμενο	110 0000
109	Αφαιρέτης	0110 1101
49	Αφαιρετέος	0011 0001
060	Διαφορά	0011 1100

Σχήμα 3(γ)

Στο Σχήμα 3 (β), φαίνεται ότι ίδιο ισχύει για τη δυαδική αφαίρεση. Δηλαδή, όταν ο αφαιρετέος είναι μεγαλύτερος του αφαιρέτη, παράγεται δανειζόμενο. Στη δυαδική αφαίρεση, η μόνη τέτοια περίπτωση είναι όταν αφαιρείται 1 από το 0. Το αποτέλεσμα είναι 1, και η λειτουργία προκαλεί δανειζόμενο 1.

Το Σχήμα 3(γ) δείχνει ένα πρόβλημα δεκαδικής αφαίρεσης. Διακρίνονται τα δανειζόμενα, τον αφαιρέτη, τον αφαιρετέο και τη διαφορά. Στο δεκαδικό πρόβλημα η αρχή γίνεται από τη δεξιά στήλη. Το 9 αφαιρούμενο δίνει 0, χωρίς δανειζόμενο. Στην επόμενη στήλη, το 4 αφαιρείται από το 0 δίνοντας 6, αλλά πρέπει να παρθεί μία μονάδα από τη στήλη των εκατοντάδων. Στη στήλη αυτή, υπάρχει μόνο μία μονάδα, έτσι το αποτέλεσμα στον αφαιρέτη είναι 0. Από το 0 του αφαιρέτη αφαιρείται το 0 του αφαιρετέου στη στήλη αυτή, δίνοντας 0 σαν αποτέλεσμα.

Στη δυαδική αφαίρεση η αρχή γίνεται πάλι με τη δεξιά στήλη. Το 1 αφαιρείται από το 1 και μας δίνει 0. Δεν παράγεται δανειζόμενο. Στη στήλη των δυάδων, 0 αφαιρείται από το 0 και πάλι δεν υπάρχει δανειζόμενο. Στη στήλη των τετράδων, 0 από 1 είναι 1 χωρίς δανειζόμενο. Το ίδιο ισχύει και για την στήλη των οκτάδων. Στη στήλη των 16δων, 1 αφαιρείται από το 0. Το αποτέλεσμα είναι 1, αλλά πρέπει να παρθεί 1 από την στήλη των 32δων. Όταν γίνει δανεισμός 1 από τον αφαιρέτη στη στήλη των 32δων, το αποτέλεσμα στον αφαιρέτη είναι 0. Η αφαίρεση επομένως είναι 1 από 0, δίνει 1 και ξανά δανειζόμενο. Στη στήλη των 64δων πρέπει να γίνει δανεισμός 1 από τον αφαιρέτη και επομένως ο αφαιρέτης γίνεται 0. Η αφαίρεση είναι 0 από 0 και το αποτέλεσμα είναι 0 χωρίς δανειζόμενο. Επειδή γίνεται διαχωρισμός αριθμών των 8-bits, πρέπει να γίνει συμπλήρωση από την αφαίρεση στην στήλη των 128δων εάν και είναι όλα ασήμαντα 0.Φυσικά, το αποτέλεσμα είναι 0.

2.6.3 Δυαδικός πολλαπλασιασμός

Όπως με την πρόσθεση και αφαίρεση, ο δεκαδικός και ο δυαδικός πολλαπλασιασμός είναι παρόμοιοι. Ο καθένας, είναι ένας γρήγορος τρόπος να προστεθεί ένας αριθμός στον εαυτό του πολλές φορές.

Για παράδειγμα, πολλαπλασιάζοντας το 7 με το 5 είναι απλά ένας γρήγορος τρόπος να προστεθεί το 7 στον εαυτό του 5 φορές..

Στον πολλαπλασιασμό, καλούμε τον ένα αριθμό πολλαπλασιαστή και τον άλλο πολλαπλασιαστέο. Στον πολλαπλασιασμό χρησιμοποιείται η μέθοδος ψηφίου προς ψηφίο. Συχνά, παράγεται κρατούμενο στον δεκαδικό πολλαπλασιασμό, αλλά συνήθως αντιμετωπίζεται από μνήμης. Αφού ολόκληρος ο πολλαπλασιαστέος έχει πολλαπλασιαστεί με το ελάχιστα σημαντικό ψηφίο(LSB) του πολλαπλασιαστή, το αποτέλεσμα καλείται πρώτο μερικό γινόμενο. Το δεύτερο μερικό γινόμενο, παράγεται αφού ο πολλαπλασιαστέος πολλαπλασιαστεί με το δεύτερο ελάχιστα σημαντικό ψηφίο του πολλαπλασιαστή. Αυτή η διαδικασία, συνεχίζεται μέχρι να παραχθούν όλα τα απαιτούμενα μερικά γινόμενα. Λόγω του ότι κάθε μερικό γινόμενο έχει δημιουργηθεί από πολλαπλασιαστή, που είναι δέκα φορές μεγαλύτερος από τον προηγούμενο, κάθε μερικό γινόμενο ολισθαίνει προς τα αριστερά κατά μία δεκαδική θέση.

Μετά, όλα τα μερικά γινόμενα προστίθενται για να παράγουν το τελικό γινόμενο. Φυσικά, η διαδικασία πρόσθεσης των μερικών γινομένων μπορεί επίσης να παράγει κρατούμενα, που πρέπει να συμπεριληφθούν στις προσθέσεις.

Ένα παράδειγμα δεκαδικού πολλαπλασιασμού φαίνεται στο σχήμα 4. Σε αυτό το πρόβλημα, πολλαπλασιάζεται το 17 με το 12. Επειδή υπάρχουν δύο ψηφία στον πολλαπλασιαστή, παράγονται δύο μερικά αποτελέσματα. Προσθέτοντας αυτά τα δύο μερικά γινόμενα, παράγεται κρατούμενο στη στήλη των εκατοντάδων. Το γινόμενο είναι 204.

	17	Πολλαπλασιαστέος
Χ	12	Πολλαπλασιαστής
	14	1_0 μερικό κρατούμενο

	17	1_0 μερικό κρατούμενο
Χ	100	Κρατούμενο
	204	Τελικό γινόμενο

Σχήμα 4.1

Στο Σχήμα 4.2 φαίνεται ο πίνακας δυαδικού πολλαπλασιασμού.

Χ		**Πολλαπλασιαστέος**	
		0	1
Πολλαπλασιαστής	0	0	0
	1	0	1

Σχήμα 4.2

Ο πίνακας δυαδικού πολλαπλασιασμού είναι εξαιρετικά απλός διότι υπάρχουν μόνο δύο δυαδικοί χαρακτήρες. Μπορείτε να δείτε ότι ο πολλαπλασιασμός δύο δυαδικών ψηφίων δε θα παράγει ποτέ κρατούμενο.

Έστω ότι πολλαπλασιάζεται το 17 με το 12 χρησιμοποιώντας τον πίνακα αλλά σε δυαδική μορφή. Αυτό φαίνεται στο Σχήμα 4.3. Ο πολλαπλασιαστής και ο πολλαπλασιαστέος είναι απλά 8-bits δυαδικές εκφράσεις των αριθμών 17 και 12.

Το πρώτο πράγμα στο οποίο πρέπει να δοθεί προσοχή, είναι ότι υπάρχουν 8 μερικά γινόμενα. Αυτό είναι επόμενο διότι υπάρχουν 8-bits στον πολλαπλασιαστή. Λόγω του ότι, οι κανόνες του δυαδικού πολλαπλασιασμού είναι τόσο απλοί, ο πολλαπλασιασμός είναι εύκολος.

Το πρώτο μερικό γινόμενο είναι όλο 0. Αυτό διότι το ελάχιστα σημαντικό ψηφίο του πολλαπλασιαστή είναι 0. Από τον πίνακα του δυαδικού πολλαπλασιασμού στο Σχήμα 4.2

φαίνεται ότι 0 φορές είναι 0. Το δεύτερο μερικό γινόμενο είναι ακριβώς το ίδιο : όλα τα ψηφία είναι 0.
Το τρίτο μερικό γινόμενο, είναι ακριβές αντίγραφο του πολλαπλασιαστέου. Η μόνη διαφορά είναι ότι ολίσθησε αριστερά τρείς δυαδικές θέσεις διότι είναι το γινόμενο του πολλαπλασιασμού του τρίτου ψηφίου του πολλαπλασιαστή με τον πολλαπλασιαστέο.
Το τέταρτο μερικό γινόμενο, είναι ξανά αντίγραφο του πολλαπλασιαστέου, αυτή τη φορά με ολίσθηση τεσσάρων θέσεων.
Το πέμπτο, έκτο, έβδομο και όγδοο μερικό γινόμενο είναι επίσης 0, διότι το πέμπτο, έκτο, έβδομο και όγδοο ψηφίο του πολλαπλασιαστή είναι επίσης 0. Προσθέτοντας τα μερικά γινόμενα, έχει παραχθεί ένα τελικό (συνολικό) γινόμενο. Σε αυτές τις προσθέσεις δεν υπήρχε κρατούμενο. Εφόσον θα μπορούσαν να παράγουν κρατούμενο, έχει αφεθεί ένα χώρος για ένα.
Το αποτέλεσμα έχει 16 θέσεις, αλλά τα 8 πλέον σημαντικά ψηφία είναι όλα 0. Επομένως, μπορούν να χρησιμοποιηθούν τα 8 ελάχιστα σημαντικά ψηφία του γινομένου σαν αποτέλεσμα. Είναι γνωστό ότι, οι δυαδικοί αριθμοί 8-bits ή λιγότερο, αντιπροσωπεύουν δεκαδικούς αριθμούς μικρότερους από 255.
Επειδή ο δυαδικός πολλαπλασιασμός είναι τόσο απλός, έχει αναπτυχθεί μία πολύ απλή μέθοδος, που λέγεται ολίσθηση και πρόσθεση.
Η μέθοδος αυτή λειτουργεί ως εξής:

i. Χρησιμοποιείται το 1^0 μερικό γινόμενο. Εάν το ελάχιστα σημαντικό ψηφίο του πολλαπλασιαστή είναι 0, το αποτέλεσμα είναι 0, ενώ αν είναι 1, το αποτέλεσμα είναι ένα ακριβές αντίγραφο του πολλαπλασιαστέου.
ii. Κάθε φορά που χρησιμοποιείται ένα άλλο ψηφίο του πολλαπλασιαστή, ο πολλαπλασιαστέος ολισθαίνει ένα ψηφίο αριστερά.
iii. Κάθε φορά που χρησιμοποιείται ένα ψηφίο του πολλαπλασιαστή που είναι 1, προστίθενται ο πολλαπλασιαστέος στο αποτέλεσμα που ήδη έχουμε, αλλά στη θέση της ολίσθησής του.
iv. Το άθροισμα στο τέλος όλων των ολισθήσεων και των αθροίσεων είναι το γινόμενο.

Στο Σχήμα 4.3(α) φαίνεται ότι έτσι λύνεται το πρόβλημα. Ο πολλαπλασιαστέος δεν ολίσθησε για το πρώτο μερικό γινόμενο, διότι χρησιμοποιήθηκε με το ελάχιστα σημαντικό ψηφίο του πολλαπλασιαστή. Εφόσον αυτό ήταν 0, το μερικό γινόμενο ήταν 0. Για το 2^0 ψηφίο του πολλαπλασιαστή ο πολλαπλασιαστέος ολίσθησε ένα ψηφίο αριστερά. Αλλά και πάλι δεν έγινε πρόσθεση, διότι το 2^0 ψηφίο του πολλαπλασιαστή ήταν επίσης 0. Ο πολλαπλασιαστέος ολίσθησε ένα ακόμα ψηφίο αριστερά. Αυτή τη φορά, ο πολλαπλασιαστέος προστέθηκε στο αποτέλεσμα, γιατί ο πολλαπλασιαστής ήταν 1.
Για το 4^o ψηφίο του πολλαπλασιαστή υπήρχε πάλι ολίσθηση και πρόσθεση γιατί ο πολλαπλασιαστής είναι 1.
Καμία άλλη πρόσθεση δε γίνεται για το 5^0 , 6^0, 7^0 και 8^0 ψηφίο του πολλαπλασιαστή γιατί είναι όλα 0. Η μέθοδος ολίσθησης και πρόσθεσης, φαίνεται σε απλούστερη μορφή στο Σχήμα 4.3(β).
Είναι εύκολο να φανεί πως λειτουργεί η μέθοδος ολίσθησης και πρόσθεσης στον δυαδικό πολλαπλασιασμό.
Αυτή η απλή μέθοδος είναι εφικτή, διότι πολλαπλασιάζοντας δυαδικό αριθμό με το 0 δίνει 0, και πολλαπλασιάζοντας τον αριθμό αυτό με 1 δίνει τον αριθμό 1.

Δυαδική αρίθμηση	Όροι
00010001	Πολλαπλασιαστέος 17_{10}
00001100	Πολλαπλασιαστής 12_{10}
00000000	1^0 μερικό γινόμενο
00000000	2^0 μερικό γινόμενο
00010001	3^0 μερικό γινόμενο
00010001	4^0 μερικό γινόμενο
00000000	5^0 μερικό γινόμενο
00000000	6^0 μερικό γινόμενο
00000000	7^0 μερικό γινόμενο
00000000	8^0 μερικό γινόμενο
000000000000000	Κρατούμενο
0000000011001100	Γινόμενο 204_{10}

Σχήμα 4.3(α)

00010001	
00001100	
10001	Ο πολλαπλασιαστής μετακινήθηκε αριστερά 2 φορές
+ 10001	Ο πολλαπλασιαστής μετακινήθηκε αριστερά 3 φορές
11001100	Άθροισμα των μετατοπίσεων των πολλαπλασιαστέων

Σχήμα 4.3(β)

Σχήμα 4.3
Πολλαπλασιάζοντας το 17 με το 12 με δυαδικό πολλαπλασιασμό. Και οι δύο αριθμοί απεικονίζονται σαν αριθμοί 8-bit και όπως είναι σε έναν 8-bit μικροεπεξεργαστή. Το αποτέλεσμα είναι ένας 16-bit δυαδικός αριθμός.

2.6.4 Δυαδική διαίρεση

Η διαίρεση είναι το αντίθετο του πολλαπλασιασμού. Δηλαδή αφαιρείται ένας αριθμός από έναν άλλον έως ότου δεν μπορεί να αφαιρεθεί άλλο. Ο αριθμός των αφαιρέσεων που μπορεί να εκτελεστεί δείχνει πόσες φορές διαιρείται ο ένας αριθμός με τον άλλο. Έτσι είναι το αντίθετο της επαναληπτικής πρόσθεσης που εκτελείται στον πολλαπλασιασμό.
Ο πολλαπλασιασμός με επαναλαμβανόμενη πρόσθεση και η διαίρεση με επαναλαμβανόμενη αφαίρεση, φαίνονται στο Σχήμα 5 όπου πέντε διαδοχικές προσθέσεις του 7 δίνουν αποτέλεσμα 35. Στη στήλη της διαίρεσης, το αποτέλεσμα είναι 0 αν αφαιρεθεί το 7 από το 35 πέντε φορές.

Η διαδικασία της διαίρεσης, είναι δυσκολότερη από αυτήν του πολλαπλασιασμού. Σε ένα πρόβλημα διαίρεσης μεγαλύτερου δεκαδικού, φαίνεται ότι η διαίρεση απαιτεί μεγαλύτερη κατανόηση της διαδικασίας που εκτελείται.
Για παράδειγμα το πρόβλημα που φαίνεται στο Σχήμα 5(α) είναι αντίθετο του Σχήματος 4.3 που δείχνει δυαδικό πολλαπλασιασμό. Δηλαδή το γινόμενο (204) είναι τώρα ο διαιρετέος, ο πολλαπλασιαστής (12) είναι τώρα ο διαιρέτης και ο πολλαπλασιαστέος του Σχήματος 4.3 (17) γίνεται τώρα το πηλίκο.
Η διαίρεση ξεκινάει με τη διαδικασία της εξέτασης. Εξετάζεται το πρόβλημα καταλήγοντας ότι το 12 χωράει στο 20 μόνο μία φορά. Μετά δοκιμάζοντάς το αφαιρείται το 12 από το 20 και πράγματι το υπόλοιπο (8) είναι μικρότερο του διαιρέτη. Συνθέτοντας το υπόλοιπο από την προηγούμενη διαφορά με το επόμενο ψηφίο του διαιρετέου και ξανά μαντεύοντας το αποτέλεσμα.
Στο Σχήμα 5(β) αντιμετωπίζεται το ίδιο πρόβλημα εκφρασμένο με δυαδικούς αριθμούς. Η πρώτη εξέταση είναι εύκολη : το 1100 χωράει στο 1100 ακριβώς μία φορά. Τοποθετώντας το δοκιμαστικό 1 στο πηλίκο, εκτελείται ο πολλαπλασιασμός και στη συνέχεια αφαιρείται για να ολοκληρωθεί ο έλεγχος. Η διαφορά (σε αυτή τη περίπτωση 0) είναι μικρότερη του διαιρέτη και έτσι συνεχίζεται η διαδικασία κατεβάζοντας το επόμενο ψηφίο του διαιρετέου. Το 1100 χωράει στο 1 μηδέν φορές. Η διαδικασία συνεχίζεται μέχρι να ολοκληρωθεί η διαίρεση.
Αυτός ο τρόπος διαίρεσης δεν είναι εύκολο να μηχανογραφηθεί όπως ο πολλαπλασιασμός. Υπάρχουν πολλοί έλεγχοι και δοκιμές. Πάντως, έχει αναπτυχτεί ένας έξυπνος τρόπος εκτέλεσης *δυαδικής διαίρεσης*, στους μικροεπεξεργαστές.
Η διαδικασία της δυαδικής διαίρεσης είναι απλή, διότι κάθε ψηφίο του πηλίκου μπορεί να είναι 1 ή 0. Για άλλη μία φορά, θα χρησιμοποιηθεί η διαδικασία ολίσθησης από δυαδική διαίρεση.
Πριν ξεκινήσει η διαίρεση του 204 με το 12 πρέπει να παρθεί το συμπλήρωμα του 2 του αριθμού 12. Αυτό γίνεται, για να χρησιμοποιηθεί δυαδικός αθροιστής και για την αφαίρεση και για την πρόσθεση. Το συμπλήρωμα του 2 του 12 είναι:

0	1100 Δυαδικός 12_2
1	0011 συμπλήρωμα του 1 του 12
0	0001 προστίθενται 1
1	0100 συμπλήρωμα του 2 του 12
ψηφίο προσήμου	ψηφία μεγέθους

Όπως και στη μακριά διαίρεση, θα εξεταστεί αν ο διαιρέτης θα χωρέσει στον ίδιο αριθμό των περισσότερο σημαντικών ψηφίων του διαιρετέου. Φυσικά ο μικροεπεξεργαστής δεν μπορεί να μαντέψει και επομένως η αρχή γίνεται αφαιρώντας στη πραγματικότητα το διαιρέτη από τα σημαντικότερα ψηφία του διαιρετέου. Αν ο διαιρέτης δε χωράει στα ψηφία αυτά του διαιρετέου, προστίθενται τα αφαιρούμενα ψηφία πίσω. Αυτό θα γίνεται αντιληπτό από το αποτέλεσμα της αφαίρεσης που θα είναι αρνητικό, δηλαδή το ψηφίο προσήμου θα είναι 1.
Κάνοντας τη πρώτη αφαίρεση θα είναι:

0	11001100	διαιρετέος
1	01000000	Αφαιρούμε 12*
0	00001100	Πρώτο αποτέλεσμα

Το 0 εδώ σημαίνει ότι το πρώτο ψηφίο του πηλίκου είναι 1 Πηλίκο = 1XXXX

Αν ο διαιρέτης χωράει στα ανάλογα ψηφία του διαιρετέου, τότε το ψηφίο προσήμου θα είναι 0. Αυτό σημαίνει ότι, το αποτέλεσμα της διαίρεσης είναι θετικός αριθμός. Όπως φαίνεται το πρώτο βήμα έχει ολοκληρωθεί.

```
   17
 12)204 διαιρέτης
   12
 84
   84
    0
 (α)

        10001
 1100)11001100
      1100
         01
          0
         011
          0
        1100
        1100
           0
 (β)
```

Σχήμα 5.Μακριά διαίρεση. (α) Αριθμητικό παράδειγμα με δεκαδική αριθμητική. (β) Αριθμητικό παράδειγμα με χρήση δυαδικής αριθμητικής για την ίδια διαίρεση με το (α).
Είναι γνωστό ότι το πρώτο ψηφίο του πηλίκου είναι 1

Σημείωση!
Σε αυτό και στα επόμενα παραδείγματα, οι αφαιρέσεις γίνονται προσθέτοντας το συμπληρωματικό του 12 , όπως βρέθηκε νωρίτερα στην διαδικασία.
Το επόμενο βήμα θα είναι να αφαιρεθεί πάλι ο διαιρέτης αλλά πρέπει να μετακινηθεί πάλι το πρώτο αποτέλεσμα έτσι ώστε να παραχθεί το δεύτερο ψηφίο του πηλίκου. Έτσι είναι:

```
0 | 00001100   Πρώτο αποτέλεσμα
0 | 0001100    Μετακινούμενο πρώτο αποτέλεσμα
```

Επόμενη αφαίρεση θα είναι:

```
0 | 0001100    Μετακινούμενο πρώτο αποτέλεσμα

1 | 01000000   Αφαιρούμε 12
1 | 0101100    Δεύτερο αποτέλεσμα
```

Το 1 εδώ σημαίνει ότι το δεύτερο ψηφίο του πηλίκου είναι ο   Πηλίκο = 10XXX

Εξετάζοντας αυτή την αφαίρεση φαίνεται ένα 1 στο ψηφίο προσήμου, που σημαίνει ότι έχουμε αρνητικό αποτέλεσμα. Αρνητικός αριθμός σημαίνει ότι η δοκιμή για 1 στο πηλίκο απέτυχε. Αυτό σημαίνει ότι, πρέπει να γίνουν δύο πράγματα. Πρώτα, πρέπει να γίνει αυτό το ψηφίο στον διαιρετέο 0. Σε αυτή την περίπτωση, το δεύτερο ψηφίο του διαιρετέου γίνεται 0.

0 | 0101100 Δεύτερο αποτέλεσμα

1 | 1100000 Πρόσθεση του 12
1 | 0001100 Μετακινούμενο πρώτο αποτέλεσμα(ξανά)

Μετά επαναλαμβάνεται πάλι η μετακίνηση

0 | 0001100 Μετακινούμενο πρώτο αποτέλεσμα(ξανά)

1 | 001100 Πρώτο αποτέλεσμα μετακινούμενο δύο φορές.

Έτσι φαίνεται ότι καταλήγει πάλι στο πρώτο αποτέλεσμα, αλλά έχει γίνει μετακίνηση δύο φορές. Δηλαδή κάθε φορά γίνεται έλεγχος και μετακίνηση.
Για άλλη μία φορά θα εξεταστεί αν ο διαιρέτης χωράει στον διαιρετέο, αφαιρώντας το δυαδικό 12 από το μετακινούμενο αποτέλεσμα.

0 | 001100 Μετακινούμενο πρώτο αποτέλεσμα(ξανά)

1 | 010000 Αφαιρούμε 12
1 | 011100 Τρίτο αποτέλεσμα

Το 1 εδώ σημαίνει
ότι το τρίτο ψηφίο
του πηλίκου είναι ο Πηλίκο = 100XX

Κοιτώντας τρίτο αποτέλεσμα φαίνεται ξανά αρνητικό αριθμό. Ξανά αυτό σημαίνει ότι το αποτέλεσμα δεν ήταν ίσο ή μεγαλύτερο του διαιρέτη, δηλαδή το τρίτο ψηφίο του πηλίκου θα είναι θα είναι 0. Αυτό σημαίνει επίσης ότι πρέπει να προστεθεί πίσω το 12 στο αποτέλεσμα αφού μετακινηθεί.

1 | 011100 Τρίτο αποτέλεσμα

0 | 110000 Πρόσθεση ξανά του 12
0 | 001100 Πρώτο αποτέλεσμα
μετακινούμενο δύο φορές.

Τώρα που διορθώθηκε το λάθος, πρέπει να μετακινηθεί ξανά το αποτέλεσμα.
Αυτό δίνει:

0 | 001100 Πρώτο αποτέλεσμα μετακινούμενο δύο φορές.

0 | 01100 Πρώτο αποτέλεσμα μετακινούμενο τρεις φορές

Και πάλι θα αφαιρεθεί ο διαιρέτης

0	01100	Πρώτο αποτέλεσμα μετακινούμενο τρεις φορές
1	01000	Αφαιρείται το 12
1	10100	Τέταρτο αποτέλεσμα

Το 1 εδώ σημαίνει
ότι το τέταρτο ψηφίο
του πηλίκου είναι ο Πηλίκο = 1000X

Το αποτέλεσμα της αφαίρεσης είναι επίσης αρνητικός αριθμός. Αυτό σημαίνει ότι το τέταρτο ψηφίο του πηλίκου είναι επίσης 0. Ξανά, πρέπει να προστεθεί το 12 πίσω στο αποτέλεσμα.

0	10100	Τέταρτο αποτέλεσμα
0	11000	Πρόσθεση ξανά του 12
0	01100	Πρώτο αποτέλεσμα μετακινούμενο τρεις φορές.

Τώρα το πρώτο αποτέλεσμα είναι μετακινούμενο τρεις φορές και ξαναμετακινείται το αποτέλεσμα και δίνει ολίσθηση 4 ψηφίων.

0	01100	Πρώτο αποτέλεσμα μετακινούμενο τρεις φορές
0	1010	Πρώτο αποτέλεσμα μετακινούμενο τέσσερις φορές

Ξαναφαιρείται ο διαιρέτης από το αποτέλεσμα. και θα αφαιρεθεί έτσι το 12 με το 12.

0	1100	Πρώτο αποτέλεσμα μετακινούμενο τέσσερις φορές
1	0100	Αφαιρούμε το 12
0	0000	Πέμπτο αποτέλεσμα

Το 0 εδώ σημαίνει ότι το πέμπτο ψηφίο του πηλίκου είναι 1 Πηλίκο = 10001

Το πέμπτο αποτέλεσμα είναι θετικός αριθμός επομένως το πηλίκο του είναι 1.
Η διαδικασία μπορεί να σταματήσει εδώ, διότι η απάντηση έχει βρεθεί και έτσι 1100 1100 διαιρούμενο με το 1100 δίνει 10001.
Η διαδικασία της διαίρεσης είναι δυσκολότερη του πολλαπλασιασμού αλλά αυτό το είδος της διαίρεσης μπορεί να γίνει ακολουθώντας τα βήματα ένα κάθε φορά και αυτό είναι πολύ σημαντικό στους μικροεπεξεργαστές.

Κεφάλαιο 3
Ο ηλεκτρονικός υπολογιστής και το δυαδικό σύστημα αρίθμησης

΄΄Για το κεφάλαιο αυτό χρησιμοποιήθηκαν στοιχεία από τις πηγές [9] ,[8] ,[12] ΄΄

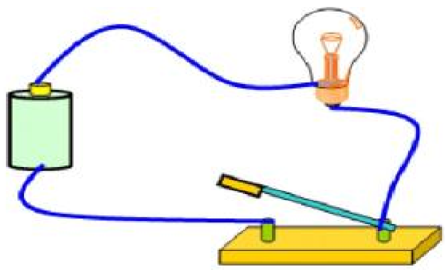
3.1 Γενικά

Για να γίνει αντιληπτός ο τρόπος με τον οποίο λειτουργεί ένας ηλεκτρονικός υπολογιστής και κατορθώνει να καταλαβαίνει και να κάνει τόσο γρήγορα πράγματα απαιτούνται γνώσεις Μαθηματικών, Φυσικής και Ηλεκτρονικής.

Ο Η/Υ λειτουργεί με ηλεκτρικό ρεύμα. Δεν είναι δηλαδή παρά μία ηλεκτρική συσκευή όπως τόσες άλλες π.χ. η κουζίνα, το πλυντήριο, το ψυγείο και άλλα.

Κάθε ηλεκτρική συσκευή αποτελείται από ένα ή περισσότερα ηλεκτρικά κυκλώματα. Ένα τέτοιο απλό ηλεκτρικό κύκλωμα, δημιουργείται εάν συνδεθούν οι πόλοι μιας μπαταρίας με δύο καλώδια που να καταλήγουν σε ένα λαμπτήρα. Πρέπει απαραίτητα να υπάρχει ακόμα συνδεδεμένος ένας διακόπτης.

Όταν αυτός είναι κλειστός όπως φαίνεται στη παρακάτω εικόνα ο λαμπτήρας φωτοβολεί γιατί ο διακόπτης ενώνει τα δύο καλώδια και έτσι κυκλοφορεί ηλεκτρικό ρεύμα σε όλο το κύκλωμα. Ενώ όταν ο διακόπτης είναι ανοικτός δεν έρχονται σε επαφή τα δύο καλώδια και ο λαμπτήρας δεν ανάβει επειδή δεν κυκλοφορεί το ηλεκτρικό ρεύμα στο κύκλωμα.



Ο λαμπτήρας λοιπόν μπορεί να είναι είτε σβηστός είτε αναμμένος ανάλογα αν περνά ηλεκτρικό ρεύμα ή όχι από αυτόν. Τις δύο αυτές καταστάσεις στις οποίες μπορεί να βρεθεί ο λαμπτήρας πρέπει να τις εκμεταλλευτεί κανείς. Για το σκοπό αυτό πρέπει να χρησιμοποιηθεί μία κωδικοποιημένη γλώσσα. Ένας κώδικας δηλαδή που στη θέση του αναμμένου ή σβηστού λαμπτήρα να βάζει γνωστά σύμβολα, όχι βέβαια οποιαδήποτε σύμβολα αλλά ψηφία τέτοια που να επιτρέπουν να σχηματιστούν με αυτά αριθμούς αφού για αυτό το σκοπό χρησιμοποιείται το ηλεκτρικό ρεύμα. Σαν τέτοια ψηφία έχουν επιλεγεί το 0 και το 1 και ο κώδικας περιέχει τις εξής οδηγίες:

i. Όταν στο κύκλωμα κυκλοφορεί το ηλεκτρικό ρεύμα (διακόπτης κλειστός) η κατάσταση που δημιουργείται (αναμμένος λαμπτήρας) αναπαριστάται με το ψηφίο 1.
ii. Όταν στο κύκλωμα δεν κυκλοφορεί το ηλεκτρικό ρεύμα (διακόπτης ανοικτός) η κατάσταση που δημιουργείται (σβηστός λαμπτήρας) αναπαρίσταται με το ψηφίο 0.

Τα ψηφία 0 και 1 δεν είναι τυχαία διαλεγμένα αλλά είναι εκείνα τα οποία χρησιμοποιούνται στο δυαδικό σύστημα αρίθμησης, προκειμένου να παρασταθεί το πλήθος των μονάδων κάθε τάξης οποιουδήποτε αριθμού. Επομένως ο Η/Υ μπορεί να καταλάβει μόνο αριθμούς που είναι γραμμένοι στο δυαδικό σύστημα αρίθμησης. Όμως αυτό το σύστημα δεν είναι ιδιαίτερα γνωστό επειδή για την αρίθμηση χρησιμοποιείται σε καθημερινή βάση το δεκαδικό σύστημα. Έτσι ο Η/Υ ρυθμίστηκε ώστε να παίρνει αριθμούς του δεκαδικού και στη συνέχεια να τους μετατρέπει στους αντίστοιχους αριθμούς του δυαδικού συστήματος,τους οποίους καταλαβαίνει.

Για τον λόγο αυτό δεν γίνεται αντιληπτή η χρησιμοποίηση του δυαδικού συστήματος από οποιοδήποτε χειριστή ενός Η/Υ.
Ας σημειωθεί ακόμα ότι αφού ο Η/Υ κάνει τις πράξεις και βρει το αποτέλεσμα σε δυαδικό αριθμό, φυσικά, τον μετατρέπει στην συνέχεια στον αντίστοιχο του δεκαδικού συστήματος. Και αυτός ο αριθμός εμφανίζεται στην οθόνη για να γίνεται κατανοητός από τους ανθρώπους.
Παρακάτω φαίνεται πως ο Η/Υ "καταλαβαίνει" τον φυσικό αριθμό 13 του δεκαδικού συστήματος (ή 13_{10}). Ο αριθμός αυτός γράφεται στο δυαδικό σύστημα ως εξής:
$13_{10} = 1*8+1*4+0*2+1*1$
ή 13_{10}= 1 οκτάδα+1 τετράδα+0 δυάδα+1 μονάδα

Άρα τα ψηφία 0 και 1 μπορούν να χρησιμοποιηθούν ώστε να ανοίγουν και να κλείνουν διακόπτες σε διάφορα ηλεκτρικά κυκλώματα. Με άλλα λόγια να διακόπτουν ή να επιτρέπουν την κυκλοφορία του ηλεκτρικού ρεύματος οπότε θα σχηματίζεται με λαμπτήρες η εικόνα του οποιουδήποτε φυσικού αριθμού. Έτσι λοιπόν ο Η/Υ διαθέτοντας ηλεκτρικά κυκλώματα με πολλούς λαμπτήρες το καθένα μπορεί να καταλαβαίνει, ανάλογα με τη σειρά που αυτοί ανάβουν οποιονδήποτε δυαδικό αριθμό.
Με τον τρόπο αυτό φαίνεται πως ο Η/Υ καταλαβαίνει τους αριθμούς. Τι γίνεται όμως με τα γράμματα, γιατί και αυτά πρέπει να τα χρησιμοποιηθούν όταν γράφονται σε έναν Η/Υ ονόματα, διευθύνσεις αλλά και ολόκληρες φράσεις.
Ο Η/Υ αντικαθιστά στη θέση κάθε γράμματος έναν αριθμό του δεκαδικού συστήματος ως εξής:

$A \rightarrow 65_{10}$
$B \rightarrow 66_{10}$
.
.
.
.
$N \rightarrow 78_{10}$

Όταν λοιπόν δοθεί στον Η/Υ η λέξη ΑΝΝΑ αυτός "καταλαβαίνει" στην θέση των γραμμάτων Α, Ν, Ν, Α τους αριθμούς 65, 78, 78, 65 του δεκαδικού συστήματος. Στην συνέχεια τους μετατρέπει στους αντίστοιχους του δυαδικού συστήματος 1000001, 1001100, 1001100, 1000001. Και αυτοί με τη σειρά τους επιτρέπουν ή απαγορεύουν την διέλευση του ρεύματος, οπότε σχηματίζονται στο εσωτερικό της μηχανής οι αντίστοιχες εικόνες των αριθμών αυτών με λαμπτήρες.
Το ίδιο γίνεται και για τα διάφορα άλλα σύμβολα π.χ. +, -, *, : κ.λπ.

Αν προστεθεί το 0 και το 1 τότε γίνεται αντιληπτό ότι ο Η/Υ αποτελείται από πολλά ηλεκτρονικά κυκλώματα προσεκτικά σχεδιασμένα από το ανθρώπινο μυαλό. Η ταχύτητα με την οποία εκτελούνται οι πράξεις οφείλεται στο γεγονός ότι το ηλεκτρικό ρεύμα διαδίδεται στα δίκτυα των κυκλωμάτων με ταχύτητα σχεδόν ίση με την ταχύτητα του φωτός. Εξαιτίας αυτής της ιλιγγιώδους ταχύτητας, συνηθίζεται να αποδίδεται στον Η/Υ υπεράνθρωπη δύναμη. Στην πραγματικότητα όμως εκτελεί πράξεις που θα μπορούσε να τις κάνει και ο άνθρωπος αν άξιζε να διαθέσει για τον σκοπό αυτό τον απαιτούμενο χρόνο. Όμως ο υπολογιστής και για το πιο απλό πρόβλημα που λύνει χρειάζεται προσεκτική και αρκετά χρονοβόρα προετοιμασία, που μόνο το ανθρώπινο μυαλό μπορεί να κάνει για να είναι δυνατόν να λυθεί στη συνέχεια το πρόβλημα από τη μηχανή.

Σημείωση! Ο πρώτος υπολογιστής λειτουργούσε με άξονες και γρανάζια. Αργότερα αντικαταστάθηκαν από ηλεκτρονικά κυκλώματα με διακόπτες. Οι παλαιότεροι Η/Υ χρησιμοποιούσαν λυχνίες κενού (λάμπες) στις διάφορες μονάδες του. Οι συσκευές αυτές ήταν πολύ ογκώδεις και χρειάζονταν μεγάλα δωμάτια για να εγκατασταθούν. Αργότερα χρησιμοποιήθηκαν ταχύτερα κυκλώματα με τρανζίστορ. Σήμερα οι πιο σύγχρονοι λειτουργούν με τσιπς (ομάδες τρανζίστορ). Όμως η βασική αρχή παραμένει η ίδια.

3.1.1Τρανζίστορ (Transistor)

Ο ηλεκτρονικός υπολογιστής βασίζεται σε Ψηφιακά Ηλεκτρονικά Κυκλώματα που αποτελούνται από τρανζίστορ**.**

Το τρανζίστορ αντικατέστησε με επιτυχία τις λυχνίες που χρησιμοποιούσαν οι ηλεκτρονικοί υπολογιστές λόγω:

i. μικρότερου όγκου και χώρου,
ii. μικρότερης κατανάλωσης ηλεκτρικής ενέργειας,
iii. μεγαλύτερης διάρκειας ζωής.

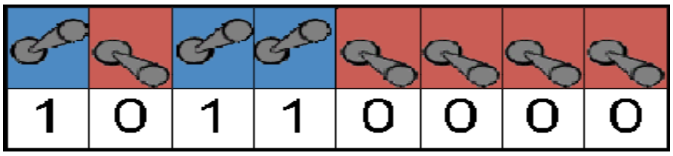
Το τρανζίστορ είναι μια ηλεκτρονική μικρή διάταξη που:

i. αποτελείται από ημιαγωγούς κρυστάλλους
ii. περιβάλλεται από μία προστατευτική θήκη (κάλυμμα),
iii. έχει τρεις επαφές(ακροδέκτες) και χρησιμοποιείται σαν διακόπτης ή ενισχυτής.

Χρησιμοποιείται σε διάφορες ηλεκτρονικές συσκευές σαν:

i. ανορθωτής
ii. ενισχυτής του ηλεκτρικού ρεύματος
iii. ταλαντωτής

3.2 Λόγοι χρήσης δυαδικού συστήματος



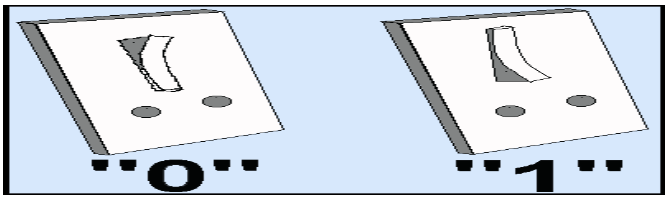
Παρά το γεγονός ότι η εσωτερική γλώσσα που καθοδηγούσε ορισμένους από τους αρχικούς υπολογιστές ήταν βασισμένη στο δεκαδικό σύστημα, από την εποχή της δεκαετίας του ‘50 σχεδόν όλοι οι υπολογιστές χρησιμοποιούν το δυαδικό.

Οι λόγοι που χρησιμοποιείται το δυαδικό σύστημα είναι:

i. **Απλότητα**: Οι μικροσκοπικοί ηλεκτρονικοί διακόπτες μέσα στον κεντρικό επεξεργαστή ενός σύγχρονου υπολογιστή δεν είναι ανάγκη να διαθέτουν παρά μόνο δύο επιτρεπόμενες καταστάσεις-ανοιχτοί ή κλειστοί, αντιπροσωπεύοντας αν αντίστοιχα το μηδέν και τη μονάδα-αντί για τις 10 που απαιτούνται για ένα δεκαδικό κύκλωμα. Αυτό το δυϊκό χαρακτηριστικό του δυαδικού συστήματος βρίσκεται, επίσης, σε αντιστοιχία με το αλγεβρικό σύστημα λογικής που επινόησε ο Βρετανός μαθηματικός George Boole τον 19ο αιώνα: μια πρόταση είναι είτε αληθής, είτε ψευδής, ακριβώς όπως ένας διακόπτης είναι είτε ανοιχτός, είτε κλειστός ή ένα δυαδικό ψηφίο είναι είτε μονάδα, είτε μηδέν.
ii. **Σαφήνεια**: Τα λάθη μειώνονται όταν μία τιμή μπορεί μόνο να είναι 0 ή 1. Ο υπολογιστής ξέρει ότι δεν υπάρχουν τιμές μεταξύ του 0 και του1, κάτι που είναι πολύ χρήσιμο όταν κάποια ηλεκτρικά σήματα γίνονται "βρώμικα", δηλαδή αλλοιώνονται. Αν μία τιμή 0.95 φανεί στην γραμμή του modem, ο υπολογιστής ξέρει ότι είναι πολύ πιθανό αυτή να είναι το 1, αφού το 0.95 δεν είναι μία έγκυρη τιμή. Έτσι θα μεταφράσει το 0.95 σε 1 και δεν θα υπάρξει απώλεια δεδομένων.

iii. **Ταχύτητα**: Οι υπολογιστές παίρνουν εκατομμύρια αποφάσεις σε ένα δευτερόλεπτο και αυτές οι αποφάσεις είναι ευκολότερο να παίρνονται όταν ο αριθμός των τιμών είναι μικρός.

iv. **Χαμηλότερο Κόστος**: Επειδή στηρίζεται μόνο σε δύο σύμβολα, το δυαδικό σύστημα προσφέρει χαμηλό κόστος κατασκευής ηλεκτρονικών κυκλωμάτων αλλά και χαμηλό κόστος κατασκευής αποθηκευτικών μέσων.

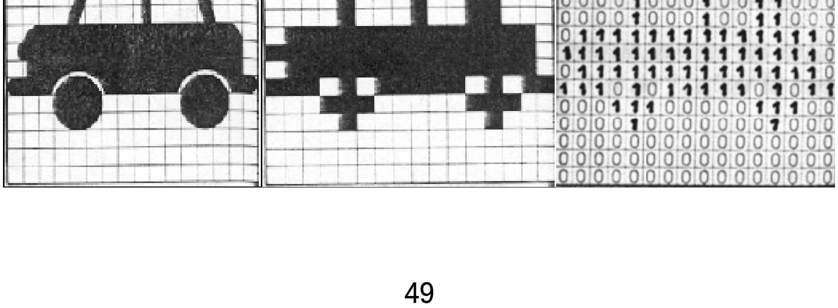


3.3 Ψηφιακή αναπαράσταση εικόνας και ήχου με την χρήση δυαδικού συστήματος

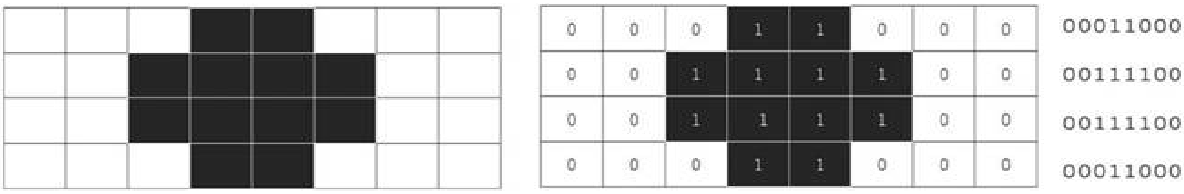
Οι υπολογιστές καλούνται, επίσης, να αντιμετωπίσουν μορφές πληροφοριών που δεν έχουν, από πρώτη άποψη, τίποτε να κάνουν με τους αριθμούς ή τη λογική. Μπορούν, παραδείγματος χάρη, να επεξεργαστούν ήχους που τους μεταβιβάζονται διαμέσου ενός μικροφώνου και να τους αναπαράγουν διαμέσου ηχείων ή να τους καταγράψουν σε ειδικούς δίσκους. Μπορούν, ακόμα να χειρίζονται εικόνες στην τηλεόραση. Σ΄ αυτές τις περιπτώσεις, ο υπολογιστής πρέπει, αρχικά, να δώσει “ψηφιακή” μορφή στις πληροφορίες, δηλαδή να τις μεταφράσει σε δυαδικά ψηφία.

3.3.1 Αναπαράσταση εικόνων

Υπάρχουν διάφορα είδη εικόνας κατάλληλο το καθένα για συγκεκριμένα είδη εφαρμογών. Το πιο απλό σε σχέση με τη πολυπλοκότητα αναπαράστασής του, είναι οι *διτονικές* *(duotone)* εικόνες. Στις διτονικές εικόνες υπάρχουν μόνο δύο χρώματα (συνήθως μαύρο και άσπρο).
Μεγαλύτερη πολυπλοκότητα αναπαράστασης εμφανίζουν οι εικόνες *συνεχούς τόνου* (continuous tone) στις οποίες κάθε σημείο της εικόνας μπορεί να έχει σε αντίθεση με τις διτονικές, πολλές τονικές διαβαθμίσεις. Οι περισσότερες διαβαθμίσεις δίνουν και μια μεγαλύτερη ομαλότητα στην εικόνα. Στην κατηγορία των εικόνων συνεχούς τόνου υπάρχουν εικόνες κλίμακας του γκρίζου(gray scale) και έγχρωμες (color).
Η αναπαράσταση μιας εικόνας σε ένα υπολογιστικό σύστημα ξεκινά με την
κατάτμησή της σε πολύ μικρές κουκίδες (εικονοστοιχεία, picture elements, pixels),με οριζόντιες και κάθετες νοητές γραμμές. Στις εικόνες παρακάτω απεικονίζεται αυτή η διαδικασία για μια ασπρόμαυρη εικόνα ενός αυτοκινήτου.
Παρατηρείστε ότι η διαδικασία αυτή επιφέρει παραμόρφωση της αρχικής εικόνας. Όσο μεγαλύτερος είναι ο αριθμός των κουκίδων στις οποίες διαιρείται μια εικόνα, τόσο πιο πιστή θα είναι η αναπαράστασή της στον υπολογιστή (τόσο μεγαλύτερη ανάλυση υπάρχει).



Ένα εικονοστοιχείο σε μια ασπρόμαυρη εικόνα αποτελείται από μία ορθογώνια περιοχή λευκού ή μαύρου χρώματος. Αν οι λευκές περιοχές αναπαρασταθούν με 0 και οι μαύρες με 1, τότε μία αντιστοίχηση φαίνεται παρακάτω.



Μετά έπεται η αντιστοίχηση της πληροφορίας του κάθε εικονοστοιχείου σε κάποια από τις στάθμες τόνου. Για την περίπτωση μονόχρωμων εικόνων (μαύρο και άσπρο) απαιτείται ένα μόνο δυαδικό ψηφίο για αυτή την αντιστοίχηση, αφού κάθε εικονοστοιχείο μπορεί να είναι άσπρο ή μαύρο.

Όσο περισσότεροι είναι οι επιθυμητοί τόνοι για την αναπαράσταση μιας εικόνας, τόσο περισσότερα είναι τα δυαδικά ψηφία που απαιτούνται για τον καθορισμό του τόνου κάθε κουκίδας με αντίστοιχες επιπτώσεις στο μέγεθος της αναπαράστασης. Για παράδειγμα, για 256 τόνους απαιτούνται 8 δυαδικά ψηφία.

Ως γνωστόν όλα τα χρώματα πλην του κόκκινου, του πράσινου και του μπλε, μπορούν να παραχθούν από τα παραπάνω. Για παράδειγμα το λευκό είναι η παρουσία στον ίδιο βαθμό των τριών βασικών χρωμάτων ενώ το μαύρο η απουσία και των τριών. Συνεπώς για μια έγχρωμη εικόνα, αρκεί να καθοριστεί για κάθε εικονοστοιχείο της το ποσοστό συμβολής στο χρώμα του, κάθε ενός εκ των τριών βασικών χρωμάτων, ή με άλλα λόγια το τόνο κάθε βασικού χρώματος.

Έτσι, η αναπαράσταση μιας εικόνας που σε κάθε βασικό χρώμα επιτρέπει 256 τόνους, απαιτεί για κάθε εικονοστοιχείο της 24 δυαδικά ψηφία. Κάθε εικονοστοιχείο μπορεί να πάρει ένα από περίπου 16 εκατομμύρια χρώματα (2563). Ο αριθμός των δυαδικών ψηφίων που αφιερώνεται για την αναπαράσταση μιας εικόνας ονομάζεται και βάθος.

3.3.2 Αναπαράσταση ήχου

Για να δώσει ψηφιακή μορφή στη μουσική, λόγου χάρη, ο υπολογιστής εκτελεί περιοδικές μετρήσεις των ηχητικών κυμάτων και καταγράφει κάθε μέτρηση σαν ένα δυαδικό αριθμό. Εκτελώντας αυτές τις μετρήσεις σε συγκεκριμένα τακτά και πολύ σύντομα διαστήματα, ο υπολογιστής είναι σε θέση να καταγράψει-ηχογραφήσει το ηχητικό αποτέλεσμα μιας ολόκληρης συμφωνικής ορχήστρας και στη συνέχεια να αναπαράγει τη μουσική με εκπληκτική πιστότητα, απλά και μόνο αντιστρέφοντας τη διαδικασία μετατροπής σε ψηφιακή μορφή.

3.4 Παράσταση ακεραίων αριθμών

Η μνήμη κάθε υπολογιστή είναι οργανωμένη σε λέξεις (words), δηλαδή ομάδες των n bits (το n είναι συνήθως ένα πολλαπλάσιο του 8). Το n ονομάζεται μήκος λέξης (word length) του υπολογιστή. Κάθε αριθμός θα καταλαμβάνει χώρο όσο μία λέξη της μνήμης του υπολογιστή. Όπως είναι γνωστό, με n δυαδικά ψηφία μπορούν να παρασταθούν 2^n ,το πλήθος διαφορετικών αριθμών, τους 0 … 2^{n-1}. Στην περίπτωση που χρειάζεται με n δυαδικά ψηφία να παρασταθούν προσημασμένοι ακέραιοι αριθμοί, τότε χρησιμοποιείται το αριστερότερο bit (δηλαδή το MSB) του αριθμού, όπου γίνεται κωδικοποίηση του πρόσημου του. Αν το πρόσημο έχει την τιμή 0, τότε ο αριθμός είναι θετικός, ενώ αν έχει την τιμή 1 είναι αρνητικός. Με τα υπόλοιπα n-1 δυαδικά ψηφία κωδικοποιείται η απόλυτη τιμή του αριθμού, δηλαδή το

μέτρο του. Για παράδειγμα οι αριθμοί $01110010_{(2)}$ και $00001_{(2)}$ είναι θετικοί, ενώ οι αριθμοί $1110010_{(2)}$ και $100001_{(2)}$ είναι αρνητικοί.
Ένας θετικός αριθμός αναπαριστάται θέτοντας στο πιο σημαντικό bit (δηλαδή το πρόσημο) την τιμή 0, και στα υπόλοιπα n-1 bits την τιμή του μέτρου του, δηλαδή στην τιμή του αριθμού. Επειδή ο μεγαλύτερος ακέραιος αριθμός που παριστάνεται με n-1 bits είναι ο $2^{n-1}-1$, η τιμή του αριθμού δεν μπορεί να ξεπερνά το όριο αυτό.
Σε έναν υπολογιστή όπου το μήκος λέξης είναι 8, ο αριθμός 23 θα παρασταθεί ως 00010111. Το αριστερότερο bit δηλώνει ότι ο αριθμός είναι θετικός, και τα υπόλοιπα bits περιέχουν τον αριθμό 23. Ο μεγαλύτερος θετικός αριθμός που μπορεί να αποθηκευθεί με 8 bits είναι ο 01111111, δηλαδή ο 127 ($2^{8-1}-1 = 2^7-1 = 128-1$).
Υπάρχουν τρεις διαφορετικοί τρόποι για να κωδικοποιηθούν οι αρνητικοί προσημασμένοι αριθμοί για τα υπόλοιπα n-1 bits:

i. Η παράσταση μέτρου
ii. Η παράσταση συμπληρώματος ως προς 1
iii. Η παράσταση συμπληρώματος ως προς 2

3.4.1 Παράσταση μέτρου

Το Most Significant Bit (MSB) δηλώνει αν ο αριθμός είναι θετικός ή αρνητικός, ενώ τα υπόλοιπα n-1 ψηφία παριστάνουν το μέτρο του αριθμού σε δυαδική μορφή.
Σε έναν υπολογιστή με λέξη 6 bits, ο αριθμός $+12_{(10)}$ παριστάνεται ως εξής: $001100_{(2)}$,
ενώ ο αριθμός $-12_{(10)}$ παριστάνεται ως εξής: $101100_{(2)}$.
Ο μέγιστος θετικός αριθμός που μπορεί να παρασταθεί στο σύστημα αυτό με λέξη μήκους n bits έχει την παράσταση:

0	**1**	**1**	**1**	**1**	**1**	**…**	**1**	**1**	**1**

Πρόσημο **n-1 bits**

και είναι ο $2^{n-1}-1$.
Ο ελάχιστος αρνητικός που μπορεί να παρασταθεί με μήκος λέξης n bits έχει την παράσταση:

1	**1**	**1**	**1**	**1**	**1**	**…**	**1**	**1**	**1**

Πρόσημο **n-1 bits**

και είναι ο $-(2^{n-1}-1)$
Ο αριθμός 0 μπορεί να παρασταθεί με δύο τρόπους: σαν 00…00 και σαν 10…00.

3.4.2 Παράσταση συμπληρώματος ως προς 1

Στην παράσταση αυτή, αν το MSB του αριθμού είναι 0, ο αριθμός είναι θετικός και το μέτρο του δίδεται από τα υπόλοιπα n-1 bits. Αν το MSB είναι 1, τότε ο αριθμός είναι αρνητικός και το συμπλήρωμα ως προς 1 των υπολοίπων n-1 bits δίνει το μέτρο του.
Το συμπλήρωμα ως προς 1 ενός δυαδικού αριθμού βρίσκεται εύκολα αν αντικατασταθούν όλα τα 1 του αριθμού με 0 και όλα τα 0 με 1.
Για παράδειγμα, το συμπλήρωμα ως προς 1 του αριθμού 11010110 είναι 00101001.
Επομένως, με βάση τα παραπάνω, ο αριθμός $+12_{(10)}$ παριστάνεται σε υπολογιστή με λέξη μήκους n=6 bits σαν $001100_{(2)}$, ενώ ο αριθμός $-12_{(10)}$ παριστάνεται σαν $110011_{(2)}$.

Ο μέγιστος θετικός αριθμός που μπορεί να παρασταθεί στο σύστημα αυτό με λέξη μήκους n bits έχει την παράσταση:

0	**1**	**1**	**1**	**1**	**1**	…	**1**	**1**	**1**

Πρόσημο **n-1 bits**

και είναι ο 2^{n-1}-1

Ο ελάχιστος αρνητικός που μπορεί να παρασταθεί με μήκος λέξης n bits έχει την παράσταση:

1	**0**	**0**	**0**	**0**	**0**	…	**0**	**0**	**0**

Πρόσημο **n-1 bits**

και είναι ο $-(2^{n-1}-1)$

Ο αριθμός 0 μπορεί να παρασταθεί με δύο τρόπους: σαν 00…00 και σαν 11…11.

Το κύριο πλεονέκτημα της παράστασης συμπληρώματος ως προς 1 είναι η συμμετρία του και η ευκολία της εύρεσης του συμπληρώματος ως προς 1. Χρησιμοποιήθηκε στο παρελθόν από μερικούς κατασκευαστές αλλά σήμερα σπάνια περιλαμβάνεται στους νέους σχεδιασμούς.

3.4.3 Παράσταση συμπληρώματος ως προς 2

Η παράσταση συμπληρώματος ως προς 2 είναι αυτή που χρησιμοποιείται περισσότερο, γιατί διευκολύνει και απλοποιεί πολύ την εκτέλεση των αριθμητικών πράξεων, τόσο για τους θετικούς, όσο και για τους αρνητικούς αριθμούς.

Όπως και στις προηγούμενες παραστάσεις, αν το MSB του αριθμού είναι 0, ο αριθμός είναι θετικός και το μέτρο του δίδεται από τα υπόλοιπα n-1 bits. Εάν το MSB του αριθμού είναι 1, τότε ο αριθμός είναι αρνητικός. Για να βρεθεί το μέτρο του αριθμού, πρέπει να υπολογιστεί το συμπλήρωμα ως προς 2 και των n ψηφίων του (δηλαδή λαμβάνεται υπόψη και το πρόσημο). Το συμπλήρωμα ως προς 2 ενός δυαδικού αριθμού βρίσκεται, εάν αντικατασταθεί το 0 με 1 και το 1 με 0 και στη συνέχεια προστεθεί 1. Για να μετατραπεί ένας αρνητικός στην παράσταση συμπληρώματος του 2, ακολουθείται παρόμοια διαδικασία: γράφεται το μέτρο του σε δυαδική μορφή, αντικαθίσταται το 0 με 1 και το 1 με 0 και στη συνέχεια προστίθεται 1. Αν δεν υπάρχει ήδη ως κρατούμενο, τοποθετείται στα αριστερά του αριθμού το ψηφίο 1 του προσήμου.

i. Για να βρεθεί η παράσταση συμπληρώματος ως προς 2 του αριθμού -17 σε ένα υπολογιστή με μήκος λέξης 16 bits, αρχικά θα γραφεί ο αντίστοιχος θετικός (17) σε δυαδική μορφή, δηλαδή 0000000000010001. Στη συνέχεια θα αντικατασταθεί το 0 με 1 και το 1 με 0 στον αριθμό αυτό, και θα πάρουμε 1111111111101110. Στον αριθμό αυτό θα προστεθεί το 1. Η τελική του παράσταση θα είναι λοιπόν 1111111111101111.

ii. Για να βρεθεί η τιμή που παριστάνει ο αριθμός 11100110 θα υπολογιστεί το συμπλήρωμά του ως προς 2 μαζί με το πρόσημο. Αρχικά αντιστρέφονται όλα τα ψηφία του δίνοντας το 00011001. Μετά προστίθεται ο αριθμός 1, και δίνει 00011001+1=00011010. Άρα το μέτρο του αριθμού είναι το $00011010_{(2)} = 26_{(10)}$ και ο αριθμός είναι ο -26.

Ο μέγιστος θετικός αριθμός που μπορεί να παρασταθεί στο σύστημα αυτό με λέξη μήκους n bits έχει την παράσταση:

0	**1**	**1**	**1**	**1**	**1**	…	**1**	**1**	**1**

Πρόσημο **n-1 bits**

και είναι ο 2^{n-1}-1.

Ο ελάχιστος αρνητικός που μπορεί να παρασταθεί με μήκος λέξης n bits έχει την παράσταση:

1	**0**	**0**	**0**	**0**	**0**	**…**	**0**	**0**	**0**

Πρόσημο **n-1 bits**

Για να βρεθεί η τιμή του αριθμού αυτού υπολογίζεται το συμπλήρωμά του ως προς 2. Αντιστρέφοντας τα ψηφία του δίνεται ο αριθμός $01111…1111_{(2)}$ και μετά προστίθεται 1, παίρνοντας τον $10000…000_{(2)} = 2^{n-1}$.Άρα ο ελάχιστος αριθμός είναι ο -2^{n-1}.
Η παράσταση συμπληρώματος του 2 έχει, από ότι φαίνεται, μία ιδιαιτερότητα: ο μικρότερος αρνητικός αριθμός που μπορεί να παρασταθεί έχει μεγαλύτερη απόλυτη τιμή (2^{n-1}) από το μεγαλύτερο θετικό (2^{n-1}-1). Αυτό συμβαίνει γιατί ο αντίστοιχός του θετικός, ο 2^{n-1}, χρειάζεται και το ψηφίο του προσήμου για να παρασταθεί.

Σημείωση! Οι έννοιες «συμπλήρωμα ως προς 2» και «παράσταση συμπληρώματος ως προς 2» είναι διαφορετικές. Το συμπλήρωμα ως προς 2 ενός αριθμού είναι το αποτέλεσμα της αντιστροφής των ψηφίων του αριθμού από 0 σε 1 και από 1 σε 0, και της πρόσθεσης σε αυτόν του 1. Η παράσταση συμπληρώματος ως προς δύο χρησιμοποιεί το συμπλήρωμα ως προς 2 για να παραστήσει τους αρνητικούς αριθμούς. Ανάλογα ισχύουν και για τις έννοιες «συμπλήρωμα ως προς 1» και «παράσταση συμπληρώματος ως προς 1».

3.5 Πρόσθεση προσημασμένων ακεραίων αριθμών

Η παραστάσεις του συμπληρώματος ως προς 1 και ως προς 2 έχουν το εξής πλεονέκτημα: η πρόσθεση μεταξύ των αριθμών γίνεται απευθείας, χωρίς να χρειάζεται μετατροπή τους, ανεξάρτητα από το πρόσημό τους. Έτσι η διαδικασία της πρόσθεσης εφαρμόζεται αυτούσια σχεδόν και σε προσημασμένους αριθμούς.
Εάν, όμως, προκύψει κρατούμενο από την πρόσθεση των MSB, αγνοείται στην περίπτωση συμπληρώματος ως προς δύο, ή προστίθεται στο αποτέλεσμα στην περίπτωση συμπληρώματος ως προς 1.
Στη συνέχεια δίδονται κάποια παραδείγματα προσθέσεων σε παράσταση συμπληρώματος ως προς 2:

00010101(21)	11001010(-54)	00010010(18)	11111101(-3)
+	+	+	+
00110011(51)	01000100(68)	10001101(-115)	11110010(-14)
01001000(72)	100001110 (14)	10011111 (-97)	111101111(-17)

Στο δεύτερο άθροισμα, το αποτέλεσμα της πρόσθεσης έχει και ένα επιπλέον bit, γιατί $11001010_{(2)}$ + $01000100_{(2)}$ = $100001110_{(2)}$. Αυτό το επιπλέον bit το αγνοείται και παίρνοντας το σωστό αποτέλεσμα. Το ίδιο ισχύει και για το τέταρτο άθροισμα.

Αντίθετα, στην περίπτωση πράξεων με παράσταση συμπληρώματος ως προς 1, το επιπλέον bit προστίθεται στο αποτέλεσμα:

00010101(21)	11001001(-54)	00010010(18)	11111100(-3)
+	+	+	+
00110011(51)	01000100(68)	10001100(-115)	11110001(-14)
01001000(72)	100001101	10011110(-97)	111101101
	+		+
	1		1
	00001110(14)		11101110(-17)

Η μόνη περίπτωση στην εκτέλεση των πράξεων που χρειάζεται προσοχή είναι όταν το αποτέλεσμα μίας πράξης είναι πολύ μεγάλο ή πολύ μικρό και δεν μπορεί να παρασταθεί με το πλήθος των bits που έχουμε στη διάθεσή μας. Τότε λέγεται ότι η πράξη προκάλεσε *υπερχείλιση* (overflow) του αποτελέσματος.
Για παράδειγμα, σε περίπτωση που χρησιμοποιείται η παράσταση συμπληρώματος ως προς 2:

```
   10010010(-110)
  +11011111(-33)
 1 01110001(113)

Αγνοείται
```

Το άθροισμα των αριθμών -110 και -33, που είναι -143, δεν μπορεί να παρασταθεί με 8 bits, γιατί ο μικρότερος αριθμός που μπορεί να παρασταθεί με 8 bits είναι ο -128. Αν προστεθούν αυτοί οι δύο αριθμοί, όπως φαίνεται και πιο πάνω, το αποτέλεσμα είναι λανθασμένο.
Γενικότερα:

i. ΔΕΝ υπάρχει υπερχείλιση όταν προστεθεί ένας θετικός με έναν αρνητικό αριθμό
ii. ΔΕΝ υπάρχει υπερχείλιση όταν τα πρόσημα είναι τα ίδια στην αφαίρεση

Υπερχείλιση ΜΠΟΡΕΙ ΝΑ ΥΠΑΡΞΕΙ όταν το μέγεθος (magnitude)
του αριθμού επηρεάζει την τιμή του πρόσημου:

i. Πρόσθεση 2 θετικών μπορεί να δώσει αρνητικό αποτέλεσμα
ii. ή, πρόσθεση 2 αρνητικών μπορεί να δώσει θετικό αποτέλεσμα
iii. ή, αφαίρεση αρνητικού από θετικό δίνει αρνητικό
iv. ή, αφαίρεση θετικού από αρνητικό δίνει θετικό

3.6 Παράσταση πραγματικών αριθμών.

Εκτός από τους ακέραιους αριθμούς, στον υπολογιστή αναπαρίστανται και «πραγματικοί» αριθμοί, δηλαδή αριθμοί με ακέραιο και κλασματικό μέρος.

3.6.1 Παράσταση σταθερής υποδιαστολής

Στην παράσταση σταθερής υποδιαστολής, ο διαχωρισμός μεταξύ του ακέραιου και του δεκαδικού μέρους, η υποδιαστολή, βρίσκεται πάντα σε μια συγκεκριμένη, σταθερή, θέση στην ακολουθία ψηφίων του αριθμού.

Το πρώτο ψηφίο, το ψηφίο προσήμου, κρατείται για να δείχνει το πρόσημο του αριθμού, συνήθως 0 όταν είναι θετικός ή 1 όταν είναι αρνητικός. Ο αριθμός στην συνέχεια μετατρέπεται σε δύναμη του 2, με την υποδιαστολή να διαχωρίζει τους θετικούς εκθέτες συμπεριλαμβανόμενου και του 0, και τους αρνητικούς εκθέτες.
Για παράδειγμα, ο δεκαδικός αριθμός που αντιστοιχεί στον δυαδικό αριθμό $01.101_{(2)}$ καθορίζεται με τον ακόλουθο τρόπο:
$01.101_{(2)} =(+)(1\times2^0)+(1\times2^{-1})+(0\times2^{-2})+(1\times2^{-3})=1.625_{(10)}$
Η ακρίβεια του αριθμητικού συστήματος ορίζεται ως η αύξηση μεταξύ των δύο διαδοχικών αριθμών και καθορίζεται από την τιμή του πλέον δεξιού ψηφίου του αριθμού που ονομάζεται ελάχιστα σημαντικό ψηφίο(LSB). Στο αριθμητικό σύστημα που είδαμε στο προηγούμενο παράδειγμα, ένας αριθμός μπορεί να παρασταθεί στο πιο κοντινό 10^{-3} ή 0.125. Το εύρος του αριθμητικού συστήματος ορίζεται σαν το διάστημα μεταξύ του πιο θετικού και του πιο αρνητικού αριθμού οι οποίοι μπορούν να παρασταθούν στο σύστημα. Η υποδιαστολή καθορίζει την αναλογία της ακρίβειας και του εύρους. Όταν η υποδιαστολή βρίσκεται στα δεξιά δεν υπάρχει δεκαδικό μέρος και μόνο ακέραιες τιμές επιτρέπονται. Αυτό αντιστοιχεί στην μεταβλητή INTEGER της FORTRAN. Στα ψηφιακά φίλτρα φαίνεται ότι είναι πιο βολικό η υποδιαστολή να βρίσκεται μία ή δύο θέσεις πιο δεξιά του ψηφίου πρόσημου, επιτρέποντας την παράσταση δεκαδικών αριθμών.

3.6.2 Παράσταση κινητής υποδιαστολής

Συνήθως στους υπολογιστές χρησιμοποιείται η παράσταση κινητής υποδιαστολής (floating point representation). Σ’ αυτήν την παράσταση ένας αριθμός x έχει την εξής μορφή:

$$x = m\,2^e$$

όπου m είναι η βάση και e είναι ο εκθέτης. Η παράσταση ορίζεται ώστε $1/2 \leq |m| < 1$. Για παράδειγμα, το νούμερο 1.5 εκφράζεται σαν $(0.75)2^1$. Και ο m και ο e εκφράζονται σαν αριθμοί σταθερής υποδιαστολής. Αν η λέξη αποτελείται από b ψηφία τα b_m αναφέρονται στην βάση και τα b_e αναφέρονται στον εκθέτη. Μια και $b = b_m + b_e$, σε ένα ψηφιακό σύστημα πρέπει να αποφασιστεί από πριν για το πόσα ψηφία θα χρησιμοποιηθούν για τον εκθέτη και πόσα για τη βάση. Ο αριθμός των ψηφίων που περιέχει ο b_e καθορίζουν το εύρος του αριθμητικού συστήματος, ενώ ο αριθμός των ψηφίων του b_m καθορίζουν την ακρίβεια. Μια συνηθισμένη μορφή έχει b_m=3b/4. Σε πολλούς μικροϋπολογιστές, 2 λέξεις 16 ψηφίων χρησιμοποιούνται για να παραστήσουν έναν αριθμό κινητής υποδιαστολής σαν μια λέξη 32 ψηφίων, από τα οποία b_m=24 και b_e= 8.

3.7 Μη Προσημασμένοι Ακέραιοι

Ένας *μη προσημασμένος ακέραιος* είναι ένας ακέραιος χωρίς πρόσημο που μπορεί να πάρει τιμές από το 0 μέχρι το θετικό άπειρο.
Επειδή δεν υπάρχει υπολογιστής που να μπορεί να αναπαραστήσει όλους τους ακέραιους σε αυτό το διάστημα τιμών, ορίζεται μια σταθερά που ονομάζεται *μέγιστος μη προσημασμένος ακέραιος* και έτσι ένας μη προσημασμένος ακέραιος μπορεί να πάρει τιμές από το 0 μέχρι αυτή τη σταθερά
Ο μέγιστος μη προσημασμένος ακέραιος εξαρτάται από τον αριθμό των μπιτ N που χρησιμοποιεί ο υπολογιστής για την αναπαράσταση ενός μη προσημασμένου ακέραιου

Διάστημα τιμών**:** $0\ldots(2^{N-1})$

Αριθμός μπιτ	**Διάστημα τιμών**
8	0…255
16	0…65.535

Η αποθήκευση μη προσημασμένων ακέραιων είναι μια απλή διαδικασία
η οποία περιγράφεται με τα επόμενα βήματα:
Ο αριθμός μετατρέπεται στο δυαδικό σύστημα. Αν το πλήθος των μπιτ είναι μικρότερο από N, τότε προστίθενται μηδενικά στα αριστερά του δυαδικού αριθμού ώστε να υπάρχουν συνολικά N μπιτ.

Έστω ότι ζητείται να αποθηκευτεί ο αριθμός 7 σε μια θέση μνήμης 8 μπιτ
Λύση
Πρώτα μετατρέπεται ο αριθμός στο δυαδικό σύστημα(111).
Προστίθενται πέντε μηδενικά ώστε να υπάρξει ένα σύνολο από N(8) μπιτ(00000111).
Ο αριθμός κατόπιν αποθηκεύεται στη θέση μνήμης
Αν ο ακέραιος προς αποθήκευση είναι μεγαλύτερος από το μέγιστο μη προσημασμένο τότε παρουσιάζεται μια κατάσταση που ονομάζεται *υπερχείλιση*.

Δεκαδικός	**Δέσμευση 8 μπιτ**	**Δέσμευση 16 μπιτ**
7	00000111	0000000000000111
234	11101010	0000000011101010
258	Υπερχείλιση	0000000100000010
24.760	Υπερχείλιση	0110000010111000
1.245.678	Υπερχείλιση	Υπερχείλιση

ΚΕΦΑΛΑΙΟ 4
Βασικές λογικές πράξεις – λογικές πύλες

´´Για το κεφάλαιο αυτό χρησιμοποιήθηκαν στοιχεία από τη πηγή [10]´´

4.1 Λογικές πύλες

Μία λογική πράξη μεταξύ μεταβλητών είναι μία συνάρτηση που ορίζεται από έναν πίνακα αληθείας (truth table). Το ηλεκτρικό κύκλωμα που εκτελεί μία λογική πράξη ονομάζεται *λογική ή ψηφιακή πύλη* και παριστάνεται από ένα σύμβολο. Τα δυαδικά ψηφία 1 και 0, που ουσιαστικά παριστάνουν τις δύο καταστάσεις αληθής (true), ψευδής (false), στη φυσική τους υπόσταση είναι δυο διακριτά επίπεδα ηλεκτρικής τάσης (συνήθως στην ιδανική περίπτωση 5V και 0V). Οι λογικές πύλες είναι λογικά κυκλώματα με πολλές εισόδους αλλά μια έξοδο.

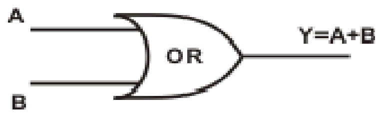
4.1.1 Ψηφιακή πύλη OR

Η πύλη OR ('Η) πραγματοποιεί την λογική πρόσθεση και εκφράζεται από την λογική συνάρτηση: Y = A + B
Ονομάζεται πύλη OR ('Η) γιατί η έξοδος Y είναι "1" (High), όταν η είσοδος A είναι "1" (High) ή η είσοδος B είναι "1" (High).
Ο πίνακας αληθείας και το σύμβολο της πύλης OR δύο εισόδων φαίνεται στο παρακάτω σχήμα:

Πίνακας αληθείας

A	B	Y=A+B
0	0	0
0	1	1
1	0	1
1	1	1

Σύμβολο



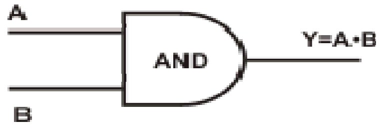
4.1.2 Ψηφιακή πύλη AND

Η πύλη AND (ΚΑΙ) πραγματοποιεί τη λογική πράξη του πολλαπλασιασμού δύο ή περισσότερων μεταβλητών. Ονομάσθηκε AND διότι πρέπει και οι δυο μεταβλητές εισόδου της λογικής πύλης να έχουν την τιμή 1, για να έχει η έξοδος τιμή 1. Όταν η μια ή και οι δυο μεταβλητές εισόδου είναι 0, τότε η έξοδος είναι 0.
Αν οι είσοδοι της λογικής πύλης είναι δυο, συμβολιζόμενες με τα γράμματα A και B και η έξοδος με το γράμμα Y, τότε η πύλη AND συμβολίζεται με τη λογική συνάρτηση:
Y=A·B
Ο πίνακας αληθείας και το σύμβολο της πύλης AND δυο εισόδων φαίνονται στο παρακάτω σχήμα:

Πίνακας αληθείας

A	B	Y=A•B
0	0	0
0	1	0
1	0	0
1	1	1

Σύμβολο



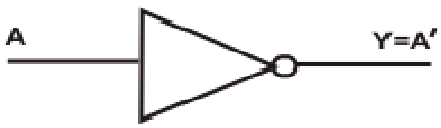
Η πύλη AND μπορεί να έχει περισσότερες από δυο εισόδους, οπότε η έξοδος της πύλης είναι μηδέν αν μία τουλάχιστον από τις εισόδους είναι 0 και έχει έξοδο 1 μόνο όταν όλες οι είσοδοι είναι 1.

4.1.3 Ψηφιακή πύλη NOT

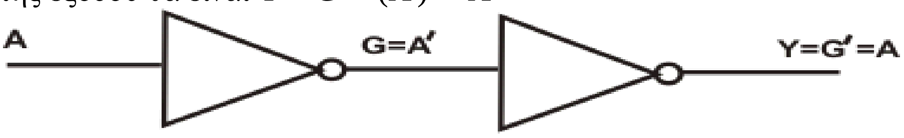
Η λογική πύλη NOT (ΟΧΙ) πραγματοποιεί την ομώνυμη πράξη που έχει λογική συνάρτηση: Y = A' και δηλώνει ότι η έξοδος είναι αντίθετη ή συμπλήρωμα της εισόδου.
Ο πίνακας αληθείας και το σύμβολο της πύλης NOT φαίνεται στο παρακάτω σχήμα:

Πίνακας αληθείας **Σύμβολο**

A	Y = A'
0	1
1	0



Η πύλη NOT (ΟΧΙ) αντίθετα με τις άλλες λογικές πύλες, έχει μόνο μια είσοδο και μια έξοδο.
Αν συνδεθούν σε σειρά 2 αναστροφείς, τότε η έξοδος θα είναι η ίδια με την είσοδο. Η λογική συνάρτηση της εξόδου θα είναι Y = G' = (A')' = A

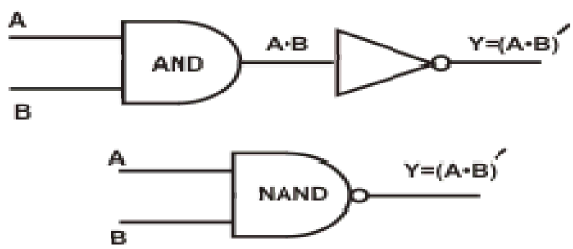


4.1.4 Ψηφιακή πύλη NAND (NOT AND)

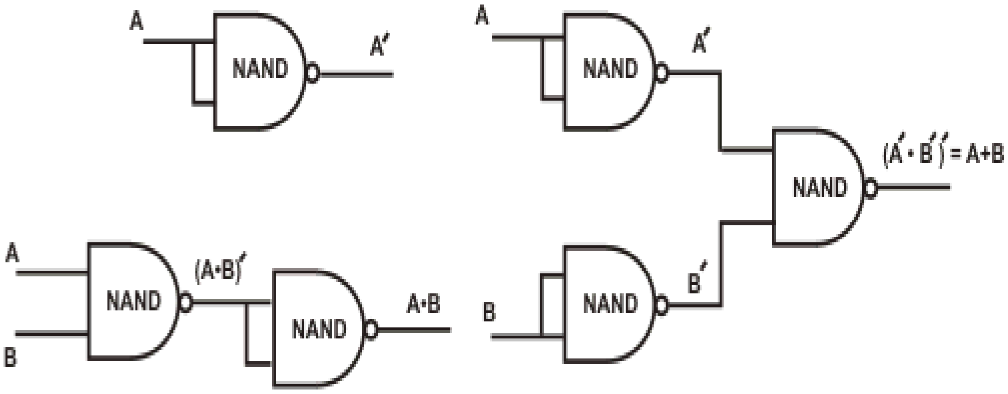
Μια τέταρτη χρήσιμη πύλη είναι η πύλη NAND η οποία είναι το συμπλήρωμα της λογικής πύλης AND.
Αυτή δημιουργείται εάν συνδεθούν σε σειρά μια πύλη AND και μια πύλη NOT και συμβολίζεται όπως η πύλη AND με ένα κύκλο στο άκρο της που δηλώνει την άρνηση.
Η λογική συνάρτηση της πύλης NAND είναι: Y = (A · B)' Ο πίνακας αληθείας και το σύμβολο της πύλης NAND φαίνονται στο παρακάτω σχήμα:

Πίνακας αληθείας **Σύμβολο**

A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0



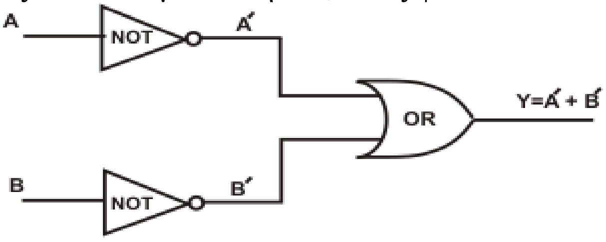
Η έξοδος της πύλης NAND είναι 1 μόνο όταν μια από τις εισόδους είναι 0.
Η πύλη NAND είναι πολύ εύχρηστη και μπορούν να κατασκευαστούν οι πύλες AND, OR, NOT χρησιμοποιώντας μόνο πύλες NAND. Για τον λόγο αυτό η πύλη NAND λέγεται πύλη γενικής χρήσης ή παγκόσμια πύλη.
Μπορούν να αντικατασταθούν οι πύλες των πυλών NOT, AND, OR με συνδυασμούς πυλών NAND όπως φαίνεται στο παρακάτω σχήμα:



Κατά τον αντίστροφο τρόπο μπορεί να υλοποιηθεί η λογική πύλη NAND με διάφορους συνδυασμούς των πυλών AND, OR, NOT.

- ***Κατασκευή πύλης NAND από πύλες NOT και OR***

Η λογική συνάρτηση της πύλης NAND μπορεί να αναλυθεί με τον ακόλουθο τρόπο, αν χρησιμοποιηθεί ο νόμος του De Morgan:
Y = (A · B)' = A' + B' . Επομένως η λογική πύλη NAND μπορεί να κατασκευαστεί χρησιμοποιώντας 2 πύλες NOT και μια πύλη OR, όπως φαίνεται στο παρακάτω σχήμα:

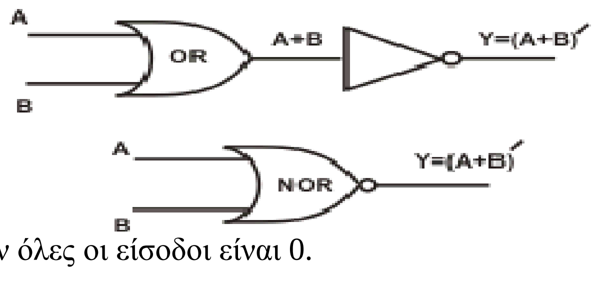


4.1.5 Ψηφιακή πύλη NOR (NOT OR)

Η λογική πύλη NOR είναι συνδυασμός της πύλης OR και της πύλης NOT. Η λογική συνάρτηση που περιγράφει την πύλη NOR είναι: Y = (A + B)'
Το σύμβολο της είναι το ίδιο με αυτό της πύλης OR αλλά προστίθεται ένας μικρός κύκλος που δηλώνει άρνηση.
Ο πίνακας αληθείας και το σύμβολο της φαίνονται στο παρακάτω σχήμα:

Πίνακας αληθείας **Σύμβολο**

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0



Η έξοδος της πύλης NOR είναι 1 μόνο όταν όλες οι είσοδοι είναι 0.

4.1.6 Ψηφιακή πύλη XOR

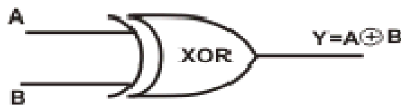
Η πύλη XOR (exclusive OR) έχει δυο εισόδους και μία έξοδο που είναι "1", αν οι δυο είσοδοι είναι διαφορετικές μεταξύ τους (γι' αυτό ονομάζεται και πύλη διαφωνίας ή σύγκρισης).
Η λογική συνάρτηση της πύλης XOR είναι:

$$Y = A \oplus B = A \cdot B' = A' \cdot B$$

Ο πίνακας αληθείας και το σύμβολο της πύλης XOR φαίνονται στο παρακάτω σχήμα:

Πίνακας αληθείας **Σύμβολο**

A	B	Y=A⊕B
0	0	0
0	1	1
1	0	1
1	1	0



4.1.7Ψηφιακή πύλη XNOR (NOT XOR)

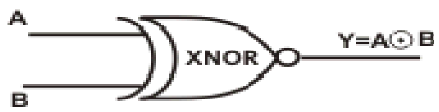
Η πύλη XNOR (exclusive NOR) έχει δύο εισόδους και μια έξοδο που είναι "1", αν οι δυο είσοδοι είναι ίσες (γι' αυτό ονομάζεται και πύλη σύμπτωσης). Η λογική συνάρτηση της πύλης XNOR είναι:

$Y = A \odot B = A \cdot B = A' \cdot B'$

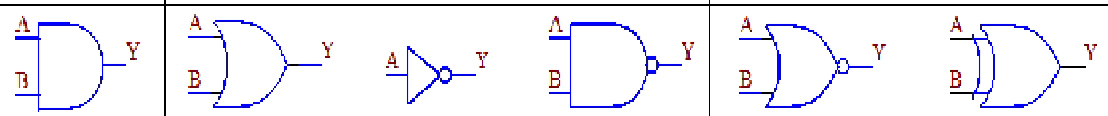
Ο πίνακας αληθείας και το σύμβολο της πύλης XNOR φαίνονται στο παρακάτω σχήμα:

Πίνακας αληθείας **Σύμβολο**

A	B	Y=A⊙B
0	0	1
0	1	0
1	0	0
1	1	1



Συνοπτικός πίνακας λογικών πυλών



AND			OR			NOT		NAND			NOR			XOR		
Y=AB			Y=A+B			Y = !A ή Y – A		Y = !(AB) ή Y – A·B			Y=!(A+B) ή Y – A+B			Y = A ⊕ B		
A	B	Y	A	B	Y	A	Y	A	B	Y	A	B	Y	A	B	Y
0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0
0	1	0	0	1	1	1	0	0	1	1	0	1	0	0	1	1
1	0	0	1	0	1			1	0	1	1	0	0	1	0	1
1	1	1	1	1	1			1	1	0	1	1	0	1	1	0

4.1.8 Δυνατοί πίνακες αληθείας στο δυαδικό σύστημα

Ένας πίνακας αληθείας παριστάνει τη συνάρτηση μεταξύ των εισόδων και της εξόδου ενός λογικού συστήματος. Για δυο εισόδους (**F,T**) υπάρχουν τέσσερις πιθανοί συνδυασμοί πραγματικών τιμών:

FF, FT, TF, TT

Επειδή κάθε δυνατή είσοδος μπορεί να δώσει δύο διαφορετικές εξόδους (0,1) συνεπάγεται ότι οι δυνατοί πίνακες αληθείας για ένα λογικό σύστημα δύο εισόδων είναι: $\mathbf{2^4=16}$

Όλοι οι πίνακες αληθείας για δύο εισόδους A, B και μία έξοδο Z

A	Τιμές εισόδου F	F	T	T		
B	F	T	F	T	Συνάρτηση (έξοδος Z)	Σύμβολο
0	F	F	F	F	πάντοτε 0	0
1	F	F	F	T	AND	$A \cdot B$
2	F	F	T	F	–	–
3	F	F	T	T	είσοδος A	A
4	F	T	F	F	–	–
5	F	T	F	T	είσοδος B	B
6	F	T	T	F	XOR	$A \oplus B$
7	F	T	T	T	OR	$A+B$
8	T	F	F	F	NOR	$\overline{A+B}$
9	T	F	F	T	XNOR	$\overline{A \oplus B}$
10	T	F	T	F	Not B	$\overline{B}$
11	T	F	T	T	–	–
12	T	T	F	F	Not A	$\overline{A}$
13	T	T	F	T	–	–
14	T	T	T	F	NAND	$\overline{A \cdot B}$
15	T	T	T	T	πάντοτε 1	1

4.2 Άλγεβρα Boole-Λογικές πράξεις

Οι αρχές της λογικής αναπτύχθηκαν από τον George Boole (1815-1884) και τον Augustus De Morgan. Εκατό χρόνια αργότερα ο Claude Shannon (ως μεταπτυχιακός φοιτητής στο MIT) έδειξε ότι η άλγεβρα Boole ήταν σχετική με την ανάλυση διακοπτικών (switching) κυκλωμάτων.

Η άλγεβρα Boole αποτελεί τη μαθηματική βάση για την ηλεκτρονική επεξεργασία της δυαδικής πληροφορίας.

4.2.1 Ιδιότητες και κανόνες της άλγεβρας Boole

Οι ιδιότητες και οι κανόνες της Άλγεβρας Boole εφαρμόζονται και ισχύουν σε τρεις κύριες ομάδες πράξεων.

i. Λογικές πράξεις με σταθερές.
ii. Λογικές πράξεις με μια μεταβλητή.
iii. Λογικές πράξεις με δυο ή περισσότερες μεταβλητές.

Λογικές πράξεις με σταθερές

AND	OR	NOT
$0 \bullet 0 = 0$	$0+0=0$	
$0 \bullet 1 = 0$	$0+1=1$	$\bar{0}=1$
$1 \bullet 0 = 0$	$1+0=1$	$\bar{1}=0$
$1 \bullet 1 = 1$	$1+1=1$	

Λογικές πράξεις με μια μεταβλητή.

AND	OR	NOT
$A \bullet 0 = 0$	$A + 0 = A$	
$A \bullet 1 = A$	$A + 1 = 1$	$\overline{A} = A$
$A \bullet A = A$	$A + A = A$	
$A \bullet \overline{A} = 0$	$A + \overline{A} = 1$	

Λογικές πράξεις με δυο ή περισσότερες μεταβλητές.

Αντιμεταθετική ιδιότητα

$A + B = B + A$

$A \bullet B = B \bullet A$

Προσεταιριστική ιδιότητα

$A + (B + C) = (A + B) + C$

$A \bullet (B \bullet C) = (A \bullet B) \bullet C$

Απορροφητική ιδιότητα

$A + (A \bullet B) = A$

$A \bullet (A + B) = A$

Επιμεριστική ιδιότητα

$A \bullet (B + C) = (A \bullet B) + (A \bullet C)$

$A + (B \bullet C) = (A + B) \bullet (A + C)$

4.2.2 Κανόνας De Morgan

$$\overline{A+B} = \overline{A} \bullet \overline{B}$$

$$\overline{A \bullet B} = \overline{A} + \overline{B}$$

Εδώ θα πρέπει να επισημανθεί πως:

$\overline{A+B}$ διαβάζεται A NOR B

$\overline{A \cdot B}$ διαβάζεται A NAND B

Τα θεωρήματα De Morgan είναι πιο σημαντικά στην λογική σχεδίαση όπου συσχετίζονται AND και NOR πύλες, ή OR και NAND πύλες. Για παράδειγμα χρησιμοποιούνται τα θεωρήματα De Morgan για να σχεδιαστεί ένα συνδυασμός πυλών NAND που είναι ισοδύναμος με μια πύλη OR δύο εισόδων. Για μία πύλη OR ισχύει:

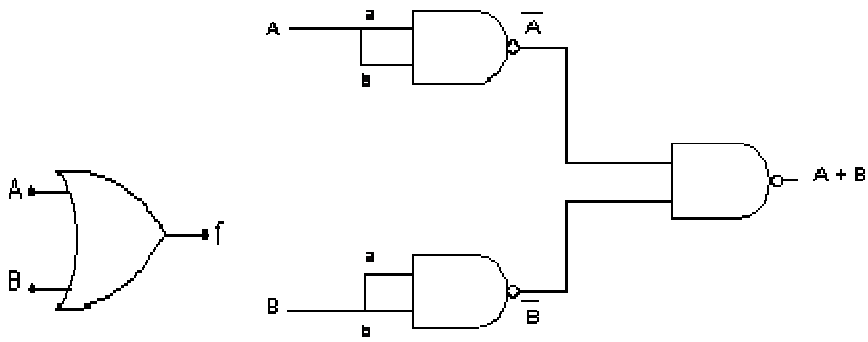
$$f = A + B$$

χρησιμοποιώντας τα θεωρήματα De Morgan:

$$f = A + B = \overline{\overline{A} \bullet \overline{B}}$$

Η παραπάνω σχέση υποδηλώνει μια πύλη NAND με NOT εισόδους. Επίσης, επειδή $\overline{A \cdot A} = \overline{A}$, μια πύλη NAND με ίδιες εισόδους παρουσιάζει την λειτουργία της πύλης NOT.

Συνεπώς προκύπτει το παρακάτω κύκλωμα:



4.3 Διαδικασία σχεδίασης ψηφιακής λογικής συνάρτησης

Με τον όρο σχεδιασμός ψηφιακής λογικής συνάρτησης, εννοείται ένας συνδυασμός λογικών πυλών για την πραγματοποίηση της επιθυμητής συνάρτησης, η συμπεριφοράς.

Η διαδικασία σχεδίασης περιλαμβάνει τα παρακάτω βήματα:

i. Σαφής διατύπωση της επιθυμητής συνάρτησης-συμπεριφοράς
ii. Πίνακας αληθείας
iii. Έκφραση της συνάρτησης υπό μορφή μεταβλητών (άλγεβρα Boole)
iv. Κατάλληλη επεξεργασία της συνάρτησης για την εξαγωγή μιας απλούστερης μορφής
v. Υλοποίηση του ψηφιακού κυκλώματος με πύλες AND, OR και NOT. Σε πολλές περιπτώσεις η υλοποίηση του κυκλώματος μπορεί να γίνει μόνο με πύλες NAND, η μόνο με πύλες NOR.

4.4 Κανονικές μορφές λογικών συναρτήσεων

Υπάρχουν δύο κανονικές μορφές λογικών συναρτήσεων, η κανονική μορφή αθροίσματος και η κανονική μορφή γινομένου

4.4.1 Κανονική μορφή αθροίσματος

Δημιουργείται από τον πίνακα αληθείας και είναι το λογικό άθροισμα (δηλαδή συνδυάζονται υπό μορφή OR) όρων που είναι εκφράσεις AND των μεταβλητών εισόδου στην κανονική, ή συμπληρωματική τους μορφή ανάλογα με την τιμή που έχουν(1 ή 0).Οι όροι που συμπεριλαμβάνονται στο λογικό άθροισμα είναι οι όροι για τους οποίους η τελική συνάρτηση έχει τιμή 1.

Παράδειγμα:

Πίνακας αλήθειας

A	B	C	F	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\bar{A}BC$
1	0	0	1	$A\bar{B}\bar{C}$
1	0	1	1	$A\bar{B}C$
1	1	0	0	
1	1	1	0	

F=1
A=0, B=1 & C=1
A=1, B=0 & C=0
A=1, B=0 & C=1

Δηλαδή οι δεκαδικοί αριθμοί 3, 4 & 5

Για κάθε έξοδο F=1 δημιουργούνται οι όροι AND όπου οι μεταβλητές είναι στη κανονική τους μορφή εάν είναι 1 και στην αντιστροφή τους μορφή εάν είναι 0. Κατόπιν οι παραπάνω όροι αθροίζονται λογικά, οπότε η κανονική μορφή αθροίσματος που προκύπτει είναι:

$$\boldsymbol{F} = \bar{A}BC + A\bar{B}\bar{C} + A\bar{B}C$$

❖ *Ημιαθροιστής*

Ο ημιαθροιστής είναι ένα ψηφιακό κύκλωμα που πραγματοποιεί την αλγεβρική άθροιση δύο δυαδικών ψηφίων:

Αλγεβρική άθροιση

A	1	1	0	0
B	1	0	1	0
A+B	10	1	1	0

ο αντίστοιχος πίνακας αληθείας είναι:

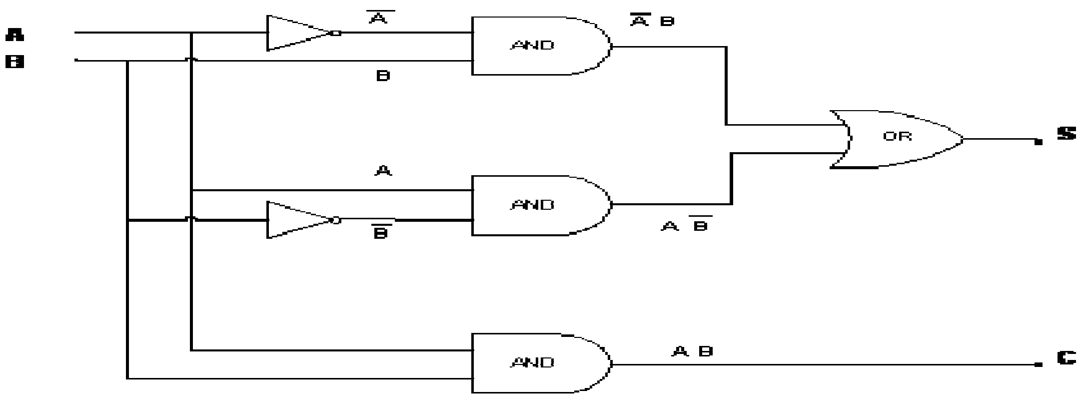
Πίνακας Αληθείας

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Συνεπώς

$S = \overline{A}B + A\overline{B}$ *και* $C = AB$

Το ψηφιακό του κύκλωμα είναι:



4.4.2 Κανονική μορφή γινομένου

Αυτή είναι μια εναλλακτική μορφή υλοποίησης της πρώτης μορφής. Οι όροι είναι αθροίσματα (δηλαδή τύπου OR) και πολλαπλασιάζονται μεταξύ τους προκειμένου να σχηματίσουν την έξοδο. Η κατανόηση της διατύπωσης του κανόνα που θα χρησιμοποιείται στο σχηματισμό της κανονικής μορφής γινομένου γίνεται με το παρακάτω παράδειγμα:

Έστω ότι δίνεται ο πίνακας αληθείας:

A	B	C	F
0	0	0	0 $\overline{A}\ \overline{B}\ \overline{C}$
0	0	1	0 $\overline{A}\ \overline{B}\ C$
0	1	0	0 $\overline{A}\ B\ \overline{C}$
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0 $A\ B\ \overline{C}$
1	1	1	0 $A\ B\ C$

οπότε προκύπτει:

$$\overline{F} = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + ABC$$

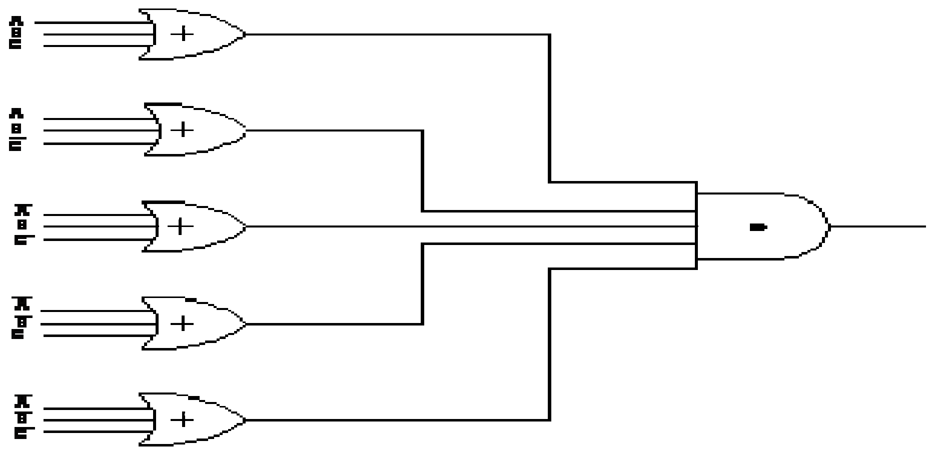
$$F = \overline{\overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + ABC}$$

$$F = \overline{\overline{A}\,\overline{B}\,\overline{C}} \cdot \overline{\overline{A}\,\overline{B}C} \cdot \overline{\overline{A}B\overline{C}} \cdot \overline{AB\overline{C}} \cdot \overline{ABC}$$

και τελικά:

$$F = (A + B + C)(A + B + \overline{C})(A + \overline{B} + \overline{C})(\overline{A} + \overline{B} + C)(\overline{A} + \overline{B} + \overline{C})$$

Η υλοποίηση της τελικής συνάρτησης γίνεται με το κύκλωμα:



Τελικά η κανονική μορφή γινομένου μπορεί να αποκτηθεί κατευθείαν από τον πίνακα αληθείας χωρίς τη χρήση κάποιων πράξεων ως εξής:

i. Εντοπίζονται οι όροι που δίνουν F=0.
ii. Δημιουργούνται τα αθροίσματα των μεταβλητών, όπου εάν η μεταβλητή έχει τιμή 0 γράφεται στην κανονική της μορφή, ενώ εάν έχει τιμή 1, γράφεται στην αντίστροφη μορφή της, παίρνοντας το γινόμενο των παραπάνω αθροισμάτων.

ΚΕΦΑΛΑΙΟ 5

Δυαδική κωδικοποίηση

΄΄Για το κεφάλαιο αυτό χρησιμοποιήθηκαν στοιχεία απο τη πηγή [17]΄΄

Η κωδικοποίηση των κάθε είδους δεδομένων σε δυαδική μορφή είτε αυτά είναι αριθμητικής φύσεως είτε άλλης προέλευσης είναι απαραίτητη για τα ψηφιακά συστήματα και τους ψηφιακούς ηλεκτρονικούς υπολογιστές, γιατί τα συστήματα αυτά λειτουργούν εσωτερικά με δυαδικά σήματα. Σκοπός του κεφαλαίου αυτού είναι να μας παρουσιαστούν οι δυαδικές μέθοδοι κωδικοποίησης(κώδικες με και χωρίς βάρη) καθώς και οι μη αριθμητικές εφαρμογές(Κώδικας ASCII, Κώδικας Baudot). Αναλύεται , επίσης, το σημαντικότατο θέμα της προστασίας των δεδομένων από τις κακές συνέπειες του θορύβου και παρουσιάζονται στοιχειώδεις μέθοδοι κωδικοποίησης για την ανίχνευση (ψηφίο ισοτιμίας) και τη διόρθωση (κώδικας Hamming) αλλοιώσεων εξ αιτίας του θορύβου.

5.1 Δυαδικοί κώδικες

Μέχρι τώρα εξετάστηκε ο τρόπος με τον οποίο αναπαρίστανται τα διάφορα είδη των αριθμών σε δυαδική μορφή, δηλαδή σε μορφή επεξεργάσιμη από τον υπολογιστή. Όμως ο υπολογιστής θα πρέπει να είναι σε θέση να επεξεργαστεί και άλλου είδους δεδομένων εκτός από αριθμητικά. Για παράδειγμα αλφαβητικά δεδομένα, όπως ένα κατάλογο ονομάτων τα οποία πρέπει να ταξινομηθούν. Φαίνεται λοιπόν πως υπάρχει ανάγκη αντιστοίχισης/κωδικοποίησης των γραμμάτων με ειδικά σύμβολα.

Οι δυαδικοί κώδικες ανήκουν στις εξής 2 κατηγορίες ανάλογα με τον τρόπο κατασκευής τους:

i. Δυαδικοί κώδικες με βάρη
ii. Δυαδικοί κώδικες χωρίς βάρη

5.1.1 Δυαδικοί κώδικες με βάρη – Κώδικας BCD

Οι δυαδικοί κώδικες με βάρη σχεδιάζονται με τέτοιο τρόπο, ώστε τα βάρη να καθορίζουν την *αξία* κάθε ψηφίου ανάλογα με τη θέση του. Στους πίνακες 5.1.1α και 5.1.1β φαίνονται δύο δυαδικοί κώδικες με βάρη 8-4-2-1 και 7-4-2-1, αντίστοιχα. Οι στήλες που αντιστοιχούν στα βάρη 8 και 7 των δύο πινάκων, αποτελούν τις στήλες στις οποίες καταχωρούνται τα περισσότερο σημαντικά ψηφία (MSB) των δύο κωδίκων, τις στήλες δηλαδή με τη μεγαλύτερη αξία (*το μεγαλύτερο βάρος*). Οι στήλες που αντιστοιχούν στα βάρη 1, αποτελούν τις στήλες που καταχωρούνται τα λιγότερο σημαντικά τους ψηφία (LSB) και αποτελούν τις στήλες με το *μικρότερο βάρος*. Ένας τέτοιος, ιδιαίτερα σημαντικός και πάρα πολύ χρήσιμος κώδικας, είναι ο ***BCD*** (***Binary Coded Decimal*** - *Δυαδικά κωδικοποιημένο δεκαδικό*) με βάρη 8-4-2-1 (. Ο BCD κώδικας με βάρη 8-4-2-1 κωδικοποιεί τα δέκα ψηφία, από το 0 μέχρι και το 9, του δεκαδικού συστήματος. Κύριο πλεονέκτημά του η άμεση αντιστοιχία κάθε κωδικοποιημένου δεκαδικού ψηφίου με το δυαδικό του ισοδύναμο. Ο BCD είναι ένας τετραψήφιος (4-bit) κώδικας, που σημαίνει ότι, κάθε κωδικοποιημένο δεκαδικό ψηφίο, παριστάνεται στο κώδικα με τέσσερα δυαδικά ψηφία. Έτσι το (5)10 είναι ο 0101 BCD, ίδιος δηλαδή με τον ισοδύναμο δυαδικό του. Ο (12)10 όμως σε BCD κώδικα, είναι ο 0001 0010, που αντιστοιχεί στο 4–bit BCD κώδικα των δεκαδικών ψηφίων 1 και 2 του αριθμού (12)10 και όχι ο ισοδύναμος δυαδικός του 1100. Για αυτό χρειάζεται προσοχή όταν πρόκειται για κωδικοποίηση μη μονοψήφιων δεκαδικών αριθμών.

Πίνακας 5.1.1 Κώδικας BCD με βάρη 8-4-2-1 και κώδικας BCD με βάρη 7-4-2-1. Ανάλογα τέλος με τα βάρη που δίνονται κάθε φορά, μπορούν να δημιουργηθούν διαφορετικοί τέτοιοι κώδικες. Έτσι, εκτός του κώδικα με βάρη 8-4-2-1, έχουμε τη δυνατότητα σχεδίασης και άλλων τέτοιων κωδίκων, όπως ο κώδικας με βάρη 7-4-2-1, που φαίνεται στο πίνακα 5.1.1β. Σ′ αυτόν ο δυαδικός συνδυασμός 1001 θα αντιστοιχεί στο δεκαδικό ψηφίο 8 (7.1+0.4+0.2+1.1=8) και όχι στο 9, όπως συνέβαινε όταν τα βάρη του κώδικα ήταν 8-4-2-1. Ο 0101 όμως αντιστοιχεί και στους δύο κώδικες στο ίδιο δεκαδικό ψηφίο, το 5.

Δεκαδικό ψηφίο	**BCD 7 4 2 1**
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 1 0

(α)

Δεκαδικό ψηφίο	**BCD 8 4 2 1**
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1

(β)

Σε έναν δυαδικό κώδικα μπορούν να δοθούν και αρνητικά βάρη.

i. Μετατροπή από BCD σε δεκαδικό

Ο τρόπος μετατροπής μιας δυαδικής ακολουθίας, κωδικοποιημένης σε BCD, στον ισοδύναμο δεκαδικό της αριθμό είναι ιδιαίτερα απλός και έχει ως εξής. Χωρίζεται η κωδικοποιημένη ακολουθία σε ομάδες τεσσάρων ψηφίων ξεκινώντας από το λιγότερο σημαντικό ψηφίο και αντικαθιστώντας στη συνέχεια κάθε τέτοια ομάδα με το ισοδύναμό της
δεκαδικό ψηφίο. Για παράδειγμα ο BCD : 0100010110001001 είναι ο (4589)10, αφού αποτελείται από τις τετραψήφιες δυαδικές ομάδες : 0100, 0101, 1000, 1001 οι οποίες αντιστοιχούν στα δεκαδικά ψηφία : 4, 5, 8 και 9.
Σημείωση! Ο κώδικας BCD χρησιμοποιεί τους 10 από τους 16 δυνατούς συνδυασμούς των 4 bits. Οι συνδυασμοί 1010,1011,1100,1101,1110 και 1111 δεν χρησιμοποιούνται.

ii. Μετατροπή από δεκαδικό σε BCD

Το ίδιο εύκολη είναι και η μετατροπή από το δεκαδικό στο BCD. Εδώ χρειάζεται η μετατροπή κάθε δεκαδικού ψηφίου σε μια ακολουθία τεσσάρων δυαδικών ψηφίων, η οποία θα αντιστοιχεί στον ισοδύναμο BCD του κάθε ψηφίου. Η διαδικασία εύρεσης του BCD κώδικα για τον δεκαδικό αριθμό 7639 είναι : 7 = 0111, 6 = 0110, 3 = 0011 και 9 =1001 και τελικά ο BCD του δεκαδικού 7639 θα είναι ο : 0111011000110001.

Επομένως, είναι σημαντική η διαφορά ανάμεσα στη δυαδική κωδικοποίηση ενός δεκαδικού αριθμού και στη μετατροπή ενός δεκαδικού αριθμού στο δυαδικό σύστημα.

5.1.2 Δυαδικοί κώδικες χωρίς βάρη

Στους κώδικες χωρίς βάρη η θέση κάθε bit του κώδικα δεν αντιστοιχεί σε κάποιο βάρος, όπως γίνεται στους δυαδικού κώδικες με βάρη. Οι κώδικες αυτοί προκύπτουν από κάποιον κανόνα. Τέτοιοι κώδικες είναι οι:

i. Κώδικας Gray
ii. Κώδικας υπερβολής κατά 3(excess-3)

Γκρίζος κώδικας(Gray Code)

Είναι δυαδικός κώδικας χωρίς βάρη που χρησιμοποιείται για την κωδικοποίηση των δεκαδικών αριθμών(όχι μόνο των 10 ψηφίων του δεκαδικού συστήματος όπως γίνεται στον BCD.
Χρησιμοποιεί 4 bits(κωδικοποίηση των 16 πρώτων δεκαδικών αριθμών 0-15).

Δεκαδικός Αριθμός	**Gray**	**Δεκαδικός Αριθμός**	**Gray**
0	0000	8	1100
1	0001	9	1101
2	0011	10	1111
3	0010	11	1110
4	0110	12	1010
5	0111	13	1011
6	0101	14	1001
7	0100	15	1000

Ο κώδικας Gray ονομάζεται και κατοπτρικός κώδικας, λόγω του τρόπου κατασκευής του.
Έχει το εξής σημαντικό χαρακτηριστικό: *αλλάζει ένα μόνο bit μεταξύ δύο διαδοχικών αριθμών*. Για παράδειγμα οι διαδοχικοί αριθμοί 5 και 6 είναι οι αριθμοί 0111 και 0101 αντίστοιχα, δηλαδή αλλάζει μόνο το δεύτερο bit από δεξιά. Επίσης, οι διαδοχικοί αριθμοί 7 και 8 στον Gray είναι οι 0100 και 1100, αντίστοιχα, δηλαδή αλλάζει μόνο το τέταρτο bit από αριστερά. Αυτό δεν συμβαίνει στο δυαδικό σύστημα. Οι διαδοχικοί αριθμοί 5 και 6 στο δυαδικό σύστημα είναι 0101 και 0110 αντίστοιχα, δηλαδή αλλάζουν τα 2 bits από δεξιά. Ακόμη οι αριθμοί 7 και 8 στο δυαδικό σύστημα είναι οι 0111 και 1000 αντίστοιχα, δηλαδή αλλάζουν και τα 4 bits.
Όταν χρησιμοποιείται ο κώδικας Gray για την μετάβαση από έναν αριθμό στον επόμενο, τότε η πιθανότητα σφαλμάτων εξαλείφεται.

Κώδικας υπερβολής κατά 3(excess-3)

Ένας άλλος κώδικας που φαίνεται στον παρακάτω πίνακα είναι ο κώδικας της υπερβολής κατά 3 (Excess 3 code). Αυτός παράγεται από τον κώδικα
BCD 8421, αν παραλείψουμε τους τρείς πρώτους συνδυασμούς και
αντιστοιχίσουμε το 0 στο συνδυασμό 0011.

ΔΕΚΑΔΙΚΟ ΨΗΦΙΟ	**ΚΩΔΙΚΑΣ 8 4 2 1**	**ΚΩΔΙΚΑΣ Excess-3**	**ΚΩΔΙΚΑΣ GRAY**
0	0000	0011	0000
1	0001	0001	0001
2	0010	0101	0011
3	0011	0110	0010
4	0100	0111	0110
5	0101	1000	0111
6	0110	1001	0101
7	0111	1010	0100
8	1000	1011	1100
9	1001	1100	1101

Σε σχέση με τον κώδικα BCD, ο κώδικας υπερβολής κατά 3, δίνει τη δυαδική παράσταση κάθε δεκαδικού ψηφίου Κ, με πρόσθεση του δυαδικού αριθμού 0011, στην αντίστοιχη παράσταση του αριθμού Κ στον κώδικα BCD.

5.2 Αλφαριθμητικοί κώδικες

Πολλές εφαρμογές στους ηλεκτρονικούς υπολογιστές απαιτούν την χρήση δεδομένων που αποτελούνται από αριθμούς αλλά και από γράμματα και ειδικούς χαρακτήρες(π.χ χρήση φύλλων excel για καταχώρηση μισθοδοσίας εταιρίας όπου απαιτούνται και μισθοί και ονόματα υπαλλήλων).

Οι αλφαριθμητικοί χαρακτήρες περιλαμβάνουν:

i. Τα 10 δεκαδικά ψηφία 0-9
ii. 26 κεφαλαία γράμματα του αγγλικού αλφαβήτου A-Z
iii. 26 μικρά γράμματα του αγγλικού αλφαβήτου a-z
iv. Ειδικούς χαρακτήρες όπως τα σημεία στίξης(. , ? !) και άλλοι χαρακτήρες όπως &#$*+=

Ένας *αλφαριθμητικός κώδικας* είναι ένας συστηματικός τρόπος παράστασης των αλφαριθμητικών χαρακτήρων σε δυαδική μορφή.

Τέτοιοι κώδικες είναι οι ακόλουθοι:

i. Ο κώδικας **ASCII** που χρησιμοποιεί 7 bits
ii. Ο κώδικας **Baudot** που χρησιμοποιεί 5 bits
iii. Ο κώδικας **Unicode** που χρησιμοποιεί 16 bits

5.2.1 Κώδικας ASCII

Ο κώδικας ASCII (American Standard Code for Information Interchange, Αμερικανικός Πρότυπος Κώδικας για Ανταλλαγή Πληροφοριών), είναι ένα κωδικοποιημένο σύνολο χαρακτήρων του λατινικού αλφάβητου όπως αυτό χρησιμοποιείται σήμερα στην Αγγλική γλώσσα και σε άλλες δυτικοευρωπαϊκές γλώσσες. Χρησιμοποιείται για αναπαράσταση κειμένου στους υπολογιστές. σε συσκευές τηλεπικοινωνίας, καθώς και σε άλλες συσκευές που δουλεύουν με κείμενο. Οι περισσότερες σύγχρονες κωδικοποιήσεις χαρακτήρων βασίζονται στον ASCII, αν και υποστηρίζουν πολύ περισσότερους χαρακτήρες.

Ιστορικά, ο ASCII αναπτύχθηκε από τηλεγραφικούς κώδικες. Η πρώτη εμπορική χρήση του ήταν ως κώδικας ενός τηλέτυπου επτά bit της Bell. Σε σύγκριση με τους παλαιότερους τηλεγραφικούς κώδικες, ο προτεινόμενος κώδικας της Bell και ο ASCII ήταν διατεταγμένοι για πιο άνετη ταξινόμηση (π.χ. αλφαβητική σειρά) καταλόγων ενώ είχαν χαρακτηριστικά και για άλλες συσκευές εκτός από τηλέτυπα.

Ο ASCII περιλαμβάνει ορισμούς για 128 χαρακτήρες: 33 είναι μη εκτυπώσιμοι χαρακτήρες ελέγχου (πλέον κατά κύριο λόγο παρωχημένοι) που επηρεάζουν το πως γίνεται η επεξεργασία του κειμένου και των κενών, 94 είναι εκτυπώσιμοι χαρακτήρες, και το κενό που θεωρείται αόρατο γραφικό. Η πλέον κοινώς χρησιμοποιούμενη κωδικοποίηση χαρακτήρων στο διαδίκτυο ήταν η US-ASCII μέχρι τον Δεκέμβριο του 2007, οπότε ξεπεράστηκε από την κωδικοποίηση UTF-8.

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

Η κωδικοποίηση διατάχθηκε έτσι ώστε οι περισσότεροι κωδικοί ελέγχου να είναι μαζί, και όλοι οι γραφικοί κωδικοί μαζί. Οι πρώτες δύο στήλες (32 θέσεις) δεσμεύθηκαν για χαρακτήρες ελέγχου. Ο χαρακτήρας κενού (space) τοποθετήθηκε πριν από τους γραφικούς χαρακτήρες έτσι ώστε να γίνουν ευκολότεροι οι αλγόριθμοι ταξινόμησης, έτσι κατέλαβε την θέση 0x20. Η επιτροπή αποφάσισε ότι ήταν σημαντικό να υποστηρίζονται κεφαλαιογράμματα αλφάβητα 64 χαρακτήρων, και έτσι επέλεξε να δομήσει έτσι τον ASCII ώστε να μπορεί εύκολα να μετατραπεί σε σύνολο 64 γραφικών χαρακτήρων. Τα μικρά γράμματα έτσι δεν ανακατεύτηκαν με τα κεφαλαία. Οι ειδικοί και αριθμητικοί κωδικοί τοποθετήθηκαν πριν από τα γράμματα ώστε να υπάρχει ευελιξία, ενώ το γράμμα 'A' τοποθετήθηκε στη θέση 0x41 ώστε να ταιριάζει με το προσχέδιο του αντίστοιχου Βρετανικού προτύπου. Τα ψηφία 0-9 διατάχθηκαν έτσι ώστε να αντιστοιχούν σε τιμές με ψηφιακό πρόθεμα 011, κάνοντας έτσι εύκολη την αποκωδικοποίηση στο δεκαδικό.

Πολλοί από τους μη αλφαριθμητικούς χαρακτήρες τοποθετήθηκαν έτσι ώστε να αντιστοιχούν με την αλλαγμένη (shifted) θέση της γραφομηχανής. Έτσι τα #, $ και % τοποθετήθηκαν ώστε να αντιστοιχούν στα 3, 4, και 5 στη διπλανή στήλη. Οι παρενθέσεις ωστόσο, δεν ήταν δυνατόν να αντιστοιχούν στο 9 και 0, καθώς η αντίστοιχη θέση του 0 είχε καταληφθεί από τον χαρακτήρα κενού. Τελικώς επιλέχθηκαν οι θέσεις 8 και 9, καθώς πολλές ευρωπαϊκές γραφομηχανές είχαν εκεί τις παρενθέσεις. Το σύμβολο @ δεν χρησιμοποιούνταν στην ηπειρωτική Ευρώπη και έτσι η επιτροπή περίμενε ότι στη γαλλική εκδοχή θα αντικαθιστούνταν με το À, έτσι το @ τοποθετήθηκε στη θέση 0x40 δίπλα στο γράμμα A.

ΛΕΞΗ	B	O	O	K
Κωδικοί ASCII στο δεκαδικό	66	111	111	107
Κωδικοί ASCII στο δυαδικό	01000010	01101111	01101111	01101011

Η Αγγλική λέξη “Book” με χρήση του κώδικα ASCII μετατράπηκε σε μια σειρά 0 και 1 ώστε να είναι δυνατός ο χειρισμός της (π.χ. αποθήκευση στη μνήμη ή το σκληρό δίσκο) από τον υπολογιστή.

5.2.2 Κώδικας Baudot

Ο κώδικας Baudot βοήθησε στην αυτοματοποίηση της αποστολής και παραλαβής μηνυμάτων μέσω τηλέγραφου και πήρε και πολλά χαρακτηριστικά και από τον ακόμα παλιότερο και πασίγνωστο κώδικα Morse. Όμως σε αντίθεση με αυτόν ο Baudot κώδικας είχε σταθερό μήκος κωδικοποίησης. Σε σύγκριση με άλλους παλιότερους κώδικες τηλέγραφου, ο προτεινόμενος κώδικας από την Bell και ο ASCII ήταν και οι δύο αναδιαταγμένοι για πιο εύκολη ταξινόμηση λιστών, και πρόσθεταν και επιπλέον χαρακτηριστικά και για άλλες συσκευές εκτός από τους τελεκτυπωτές.

5.2.3 Ο κώδικας UNICODE

Ο κώδικας UNICODE έχει αναγνωριστεί από το διεθνή Οργανισμό Τυποποιήσεως ISO, από το 1993, ως παγκόσμιο πρότυπο. Παρέχει τη δυνατότητα κωδικοποίησης όλων των χαρακτήρων που χρησιμοποιούνται από ένα μεγάλο αριθμό γλωσσών του κόσμου και, έτσι, ξεπέρασε τον κώδικα ASCII (που καλύπτει μόνο το λατινικό αλφάβητο) στον οποίο, κυρίως, βασίστηκε.

Για την κωδικοποίηση του μεγάλου πλήθους των διαφορετικών χαρακτήρων που χρησιμοποιούνται στα αλφάβητα των διαφόρων γλωσσών ο κώδικας UNICODE χρησιμοποιεί 16 bits. Τα 16 bits παρέχουν τη δυνατότητα αξιοποιήσεως 65536 (2^{16}) διαφορετικών συνδυασμών που υπερκαλύπτουν το σύνολο των χαρακτήρων όλων των γνωστών γλωσσών του πλανήτη μας. Έτσι, ο κώδικας UNICODE, με το πλήθος των συνδυασμών του, επιτρέπει την αναπαράσταση των Λατινικών, των Ελληνικών, των Κυριλλικών, των Αρμενικών, των Εβραϊκών, των Αραβικών αλλά και πολλών άλλων χαρακτήρων λιγότερο διαδεδομένων γλωσσών. Επίσης, καλύπτει και το ενοποιημένο σύνολο των Κινέζικων, Ιαπωνικών και Κορεατικών (CJK set) ιδεογραμμάτων. Συμπεριλαμβάνει τα σημεία στίξης, διάφορα διακριτικά, μαθηματικά και τεχνικά σύμβολα, βέλη, τυπογραφικά σημεία κ.λπ. Με τον τρόπο αυτό διευκολύνονται οι συναλλαγές και η ανταλλαγή αρχείων κειμένου ανάμεσα στις χώρες με διαφορετικές γλώσσες. Ο κώδικας UNICODE, στη 2η έκδοσή του, παρέχει σχεδόν 39.000 κωδικοποιημένους χαρακτήρες από τα παγκόσμια αλφάβητα, τα σύνολα ιδεογραμμάτων και τις συλλογές ειδικών συμβόλων. Επίσης, περίπου 6.000 κωδικοί είναι δεσμευμένοι για ιδιωτική χρήση από τους δημιουργούς υλικού και λογισμικού. Οι υπόλοιποι συνδυασμοί(πάνω από 20.000) παραμένουν, προς το παρόν, αχρησιμοποίητοι για μελλοντική κωδικοποίηση.

5.3 Λοιποί Κώδικες

Άλλοι κώδικες που θα πρέπει να αναφερθούν είναι εκείνοι οι κώδικες για τον έλεγχο ή και την διόρθωση λαθών (parity, Hamming).

5.3.1 Έλεγχος ισοτιμίας(parity checking)

Τέλος, για την ανίχνευση λάθους κατά την μετάδοση ή και την αποθήκευση δυαδικών δεδομένων, χρησιμοποιείται ευρύτατα ο έλεγχος ισοτιμίας (parity checking). Σύμφωνα με αυτόν, για κάθε δυαδική λέξη μήκους n που αποστέλλεται ή αποθηκεύεται, δημιουργείται ένα επιπλέον bit, το bit ισοτιμίας, με τέτοιο τρόπο ώστε το συνολικό πλήθος των μονάδων να είναι άρτιο ή περιττό, ανάλογα με το αν έχει επιλεγεί άρτια ή περιττή ισοτιμία. Το επιπλέον αυτό bit γίνεται μέρος της νέας λέξης που δημιουργείται, αυξάνοντας έτσι το συνολικό μήκος της κάθε λέξης σε n+1 bits. Κατά την λήψη ή την ανάγνωση της λέξης ελέγχεται αν το συνολικό πλήθος των μονάδων είναι άρτιο ή περιττό. Με τον τρόπο αυτό διαπιστώνεται αν υπήρξε λάθος σε κάποιο bit (δηλαδή από 0 να έγινε 1 ή το αντίστροφο). Το πιθανό λάθος ανιχνεύεται μόνο, χωρίς να είναι δυνατόν να εντοπισθεί η θέση του bit που άλλαξε. Είναι φανερό πως με τον τρόπο αυτό μπορούμε να ελέγξουμε μόνο περιττό πλήθος λαθών.

i. Κώδικας περιττής ισοτιμίας

Το ψηφίο ισοτιμίας είναι 0 αν το σύνολο των ψηφίων, 1, είναι περιττό. Το ψηφίο ισοτιμίας είναι 1 αν το σύνολο των ψηφίων, 1, είναι άρτιο.

Για παράδειγμα η δυαδική λέξη 010001 έχει αριθμό ψηφίων ‘1’ άρτιο, συνεπώς θα μεταδοθεί με ψηφίο ισοτιμίας ‘1’, είτε: 1 | 010001

ii. Κώδικας άρτιας ισοτιμίας

Αντίστροφος της περιττής ισοτιμίας. Το ψηφίο ισοτιμίας είναι 1 αν το σύνολο των ‘1’ είναι περιττό. Το ψηφίο ισοτιμίας είναι 0 αν το σύνολο των ‘1’ είναι άρτιο.

Για παράδειγμα η δυαδική λέξη 10110 έχει αριθμό ψηφίων ‘1’ περιττό, συνεπώς θα μεταδοθεί με ψηφίο ισοτιμίας ‘1’, είτε: 1 | 10110

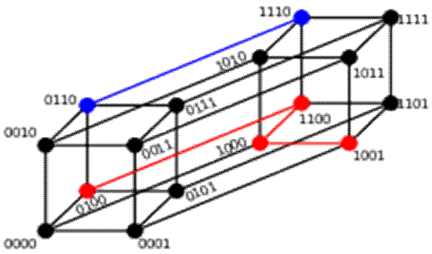
5.3.2 Απόσταση Hamming

Η απόσταση Hamming πήρε το όνομά της από τον Richard Hamming, ο οποίος την εισήγαγε στην θεμελιώδη εργασία του για τους κώδικες Hamming, Error detecting and error correcting codes (Κώδικες εντοπισμού και διόρθωσης σφαλμάτων, 1950).
Χρησιμοποιείται στην τηλεπικοινωνία για την μέτρηση των ανεστραμμένων bit σε μία σταθερού μήκους ψηφιακή λέξη ως εκτίμηση σφάλματος, και συνεπώς μερικές φορές αποκαλείται και απόσταση σήματος. Η ανάλυση βάρους Hamming χρησιμοποιείται σε αρκετούς τομείς, όπως η θεωρία πληροφορίας, θεωρία κωδικοποίησης και κρυπτογραφία. Ωστόσο για την σύγκριση συμβολοσειρών διαφορετικού μήκους, ή συμβολοσειρών όπου δεν έχουν γίνει μόνο αντικαταστάσεις αλλά αναμένονται και εισαγωγές ή διαγραφές είναι καταλληλότερες πιο πολύπλοκες μέθοδοι όπως η απόσταση Levenshtein. Για q-δικές συμβολοσειρές επί ενός αλφάβητου μεγέθους $q \geq 2$ η απόσταση Hamming εφαρμόζεται στην περίπτωση της ορθογωνικής διαμόρφωσης, ενώ η απόσταση Lee χρησιμοποιείται για την φασική διαμόρφωση. Αν $q = 2$ ή $q = 3$ οι αποστάσεις συμπίπτουν.
Η απόσταση Hamming χρησιμοποιείται επίσης στη συστηματική ως μέτρο της γενετικής απόστασης.
Στην θεωρία πληροφορίας, ως απόσταση Hamming μεταξύ δύο συμβολοσειρών ίσου μήκους ορίζεται ο αριθμός θέσεων στις οποίες τα αντίστοιχα σύμβολα είναι διαφορετικά. Η απόσταση Hamming, μετρά τον ελάχιστο αριθμό αντικαταστάσεων που χρειάζονται ώστε να μετατραπεί η μία συμβολοσειρά στην άλλη, ή αλλιώς, τον αριθμό των λαθών που μετέτρεψαν την μία συμβολοσειρά στην άλλη. Για παράδειγμα η απόσταση Hamming μεταξύ:

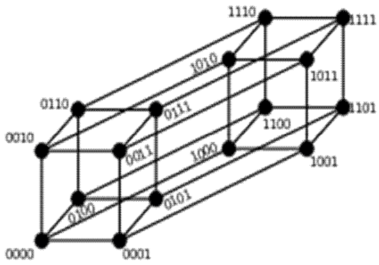
i. "**τ**ό**ν**ο**ς**" και "**π**ό**θ**ο**ι**" είναι 3.
ii. 10**11**1**1**01 και 10**01**0**0**01 είναι 2.
iii. 2**173**896 και 2**233**796 είναι 3.

Για δεδομένο μήκος **n**, η απόσταση Hamming είναι μετρική στον διανυσματικό χώρο των λέξεων αυτού του μήκους, καθώς εμφανώς ικανοποιεί τις προϋποθέσεις της μη αρνητικότητας, της ταυτότητας των μη διακρίσιμων και της συμμετρίας, και μπορεί να δειχθεί εύκολα με πλήρη επαγωγή ότι ικανοποιεί και την τριγωνική ανισότητα. Η απόσταση Hamming μεταξύ δύο λέξεων a και b μπορεί επίσης να ειδωθεί ως το βάρος Hamming του a−b για κατάλληλη του τελεστή −.
Για δυαδικές συμβολοσειρές a και b η απόσταση Hamming είναι ίση με τον αριθμό των μονάδων στο a XOR b. Ο μετρικός χώρος δυαδικών συμβολοσειρών μήκους n, με την απόσταση Hamming είναι γνωστός ως κύβος Hamming. Είναι ισοδύναμος ως μετρικός χώρος με το σύνολο των αποστάσεων μεταξύ κορυφών σε ένα γράφημα υπερκύβου. Αν η δυαδική συμβολοσειρά μήκους n θεωρηθεί ως διάνυσμα στο Rn μεταχειρίζοντας κάθε σύμβολο της ως πραγματική συντεταγμένη, οι συμβολοσειρές σχηματίζουν τις κορυφές ενός n-διάστατου υπερκύβου, και η απόσταση Hamming των συμβολοσειρών είναι ισοδύναμη με την απόσταση Manhattan των κορυφών.

Δύο παραδείγματα αποστάσεων:



0100->1001 έχει απόσταση (κόκκινο μονοπάτι),
0110->1110έχει απόσταση 1 (μπλε μονοπάτι)



Δυαδικός υπερκύβος 4-bitγια την εύρεση της απόστασης Hamming

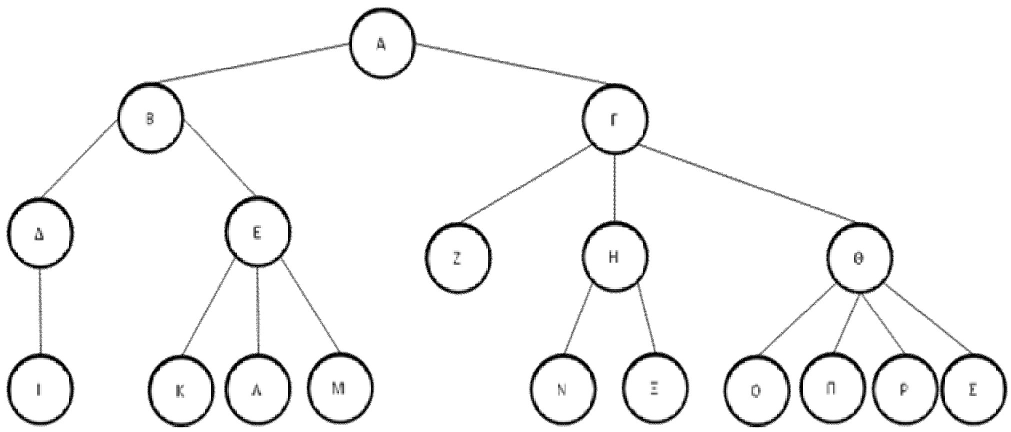
ΚΕΦΑΛΑΙΟ 6

ΔΥΑΔΙΚΑ ΔΕΝΤΡΑ ΑΝΑΖΗΤΗΣΗΣ

΄΄Για το κεφάλαιο αυτό χρησιμοποιήθηκαν στοιχεία απο τη πηγή [20] [21]΄΄

6.1 Δέντρα Αναζήτησης

Δένδρο είναι ένα σύνολο κόμβων που συνδέονται με ακμές. Ένας από τους κόμβους λέγεται *ρίζα* ***(root)***. Σε κάθε κόμβο καταλήγει μια μόνο ακμή, εκτός από τη ρίζα, στην οποία δεν καταλήγουν ακμές, αλλά μόνο ξεκινούν. Από όλους τους κόμβους μπορούν να ξεκινούν καμία, μια ή περισσότερες ακμές. Στο σχ. 6.1 φαίνεται ο συνηθέστερος τρόπος σχηματικής αναπαράστασης ενός δένδρου:



Σχήμα 6.1

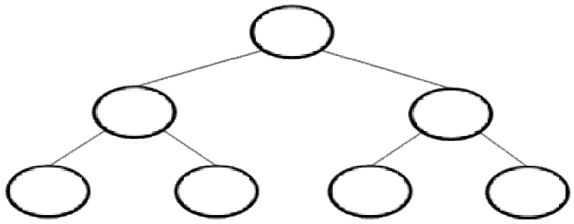
Για να είναι εύκολη η αναφορά στους κόμβους του δένδρου και στις σχέσεις μεταξύ τους, είναι απαραίτητη η υιοθέτηση κάποιας ορολογίας. Έτσι:

i. Ο κόμβος Α είναι ο *πατέρας **(father)*** των κόμβων Β και Γ. Ομοίως ο Ε είναι ο πατέρας των Κ, Λ και Μ, ενώ οι Ν και Ξ είναι *παιδιά* (***children***) του κόμβου Η. Δηλαδή από τον πατέρα ξεκινούν ακμές, οι οποίες καταλήγουν στα παιδιά του. Με την ίδια λογική, ο Π είναι *εγγονός* κόμβος του Γ, ενώ ο Β είναι *παππούς* του Κ.
ii. Ο κόμβος Ε είναι κόμβος βαθμού (degree) 3, ενώ ο Η είναι βαθμού 2. *Δηλαδή βαθμός ενός κόμβου είναι ο αριθμός των παιδιών του.* Βαθμός ορίζεται και για ολόκληρο το δένδρο. Αυτός ισούται με τον μέγιστο από τους βαθμούς των κόμβων του. Το δένδρο του πιο πάνω σχήματος είναι ένα τετραδικό δένδρο. Πολύ συνηθισμένα δένδρα είναι τα *δυαδικά **(binary)***, τα *τριαδικά **(ternary)*** και τα *τετραδικά **(quadtree)***.
iii. Οι κόμβοι Ζ, Ν, Σ είναι τερματικοί κόμβοι ή φύλλα (terminal nodes ή leaves) του δένδρου, γιατί δεν έχουν παιδιά. Οι μη τερματικοί λέγονται επίσης και *εσωτερικοί **(internal)**ή κλαδιά **(branches)***.
iv. Το επίπεδο (level) του κόμβου Ε είναι 3, δηλαδή ισούται με τον αριθμό των προγόνων του μέχρι τη ρίζα συν 1.
v. Το βάθος (depth) ή ύψος (height) του δένδρου του σχ.6.1 είναι 4, δηλαδή ισούται με το μέγιστο επίπεδο των κόμβων του δένδρου.
vi. Ένα δένδρο έχει πολλά υποδέντρα (subtrees) τα οποία προκύπτουν αν θεωρηθεί κάθε φορά ως ρίζα ένας άλλος κόμβος πλην της πραγματικής αρχικής ρίζας.
vii. Ένα δένδρο λέγεται *πλήρες **(complete)*** όταν περιέχει το μέγιστο δυνατό αριθμό κόμβων. Ο μέγιστος αυτός αριθμός εξαρτάται από το βαθμό και το βάθος του

δένδρου. Ένα πλήρες δυαδικό δένδρο βάθους k έχει 2^{k-1} κόμβους. Το πλήθος m των κόμβων στη γενική περίπτωση δένδρου βαθμού p και βάθους k δίνεται από τον τύπο:

$$m = \frac{p^k - 1}{p - 1}$$

Το σχ. 6.2 δείχνει ένα πλήρες δυαδικό δένδρο.



Σχήμα 6.2

viii. *Μήκος διαδρομής* ***(path length)*** για ένα κόμβο λέγεται ο αριθμός των ακμών από τη ρίζα ως τον κόμβο. Το μήκος διαδρομής ισούται με το επίπεδο του κόμβου μείον 1. Το μήκος διαδρομής για τον κόμβο Μ του σχ. 6.2 είναι 3.

Ένα δένδρο είναι μια ειδική μορφή συνδεδεμένης λίστας, σε αυτό δε εφαρμόζονται όλα όσα είναι γνωστά για τις λίστες, όπως παρεμβολή, διαγραφή κλπ. Μια από τις δυσκολίες τις οποίες καλείται κανείς να αντιμετωπίσει στα δένδρα είναι η αναδρομικότητα των περισσοτέρων συναρτήσεων που χρησιμοποιούνται. Αυτό είναι δικαιολογημένο, αν σκεφτεί κανείς ότι το δένδρο είναι από τη φύση του αναδρομική δομή δεδομένων: *κάθε υποδένδρο είναι επίσης δένδρο*.

6.2. Δυαδικά Δένδρα

Μια ειδική μορφή δένδρων είναι τα *δυαδικά δένδρα*, αυτά δηλαδή που έχουν βαθμό 2. Συνεπώς, κανένας κόμβος τους δεν έχει περισσότερα από δύο παιδιά. Τα δυαδικά δένδρα είναι πολύ χρήσιμα για τη δημιουργία μοντέλων συνδυασμού δύο στοιχείων, τα οποία παράγουν ένα νέο στοιχείο, το οποίο χρησιμοποιείται μαζί με κάποιο άλλο για τη δημιουργία ενός νέου στοιχείου κ.ο.κ.

Ένα δένδρο χρησιμοποιείται λοιπόν για την αναπαράσταση των δεδομένων, τα οποία βρίσκονται στους κόμβους, αλλά και των συσχετίσεων μεταξύ τους, δηλαδή των ακμών, ενώ η ολοκλήρωση των ανώτερων στοιχείων εξαρτάται από τα κατώτερα.

6.3. Διάσχιση Δέντρων

Έστω το δυαδικό δένδρο του Σχ.6.3. *Διάσχιση(traverse)* εννοείται η λειτουργία κατά την οποία πρέπει να επισκέπτεται κάθε κόμβος ακριβώς μια φορά. Υπάρχουν τρεις διαφορετικοί τρόποι για τη διάσχιση ενός δένδρου, σε καθένα από τους οποίους γίνονται οι παρακάτω ενέργειες, με διαφορετική όμως σε κάθε περίπτωση σειρά:

i. Επίσκεψη της ρίζας.

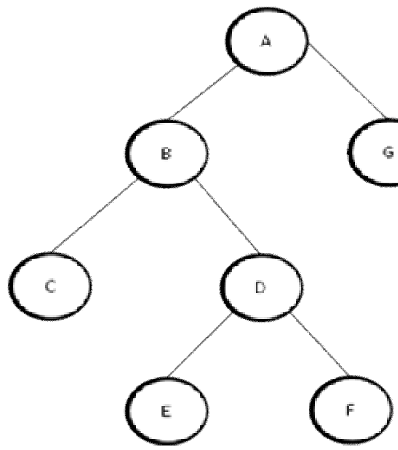
ii. Επίσκεψη του αριστερού υποδένδρου.

iii. Επίσκεψη του δεξιού υποδένδρου.

Ανάλογα με τη σειρά που εκτελούνται οι ενέργειες αυτές υπάρχουν οι εξής τρόποι διάσχισης:

i. *Προδιατεταγμένος(Preorder):* σε αυτόν οι ενέργειες εκτελούνται με τη σειρά (α)–(β)–(γ). Η διάσχιση του δένδρου του σχ.6.3 θα έδινε:
ABCDEFG

ii. *Ενδοδιατεταγμένος(Inorder):*οι ενέργειες εκτελούνται με τη σειρά (β)–(α)–(γ). Η διάσχιση του δένδρου του σχ.6.3 θα έδινε:
CBEDFAG



Σχήμα 6.3

iii. *Μεταδιατεταγμένος(Postorder)*: η σειρά των ενεργειών είναι (β)–(γ)–(α). Η διάσχιση του δένδρου του σχ. 6.3 θα έδινε:
CEFDBGA

6.4 Υλοποίηση δυαδικών δένδρων με πίνακα

Μέσω των δένδρων αναπαρίστανται δεδομένα, αλλά και συσχετίσεις μεταξύ τους. Στη μνήμη του υπολογιστή χρησιμοποιούνται διάφοροι τρόποι για την αποθήκευσή τους.

i. Ένας τρόπος που εφαρμόζεται στα πλήρη κυρίως δυαδικά δένδρα κάνει *χρήση ενός μονοδιάστατου πίνακα.*

Σε ένα πλήρες δυαδικό δένδρο ύψους h, ο αριθμός των κόμβων του είναι $m=2^h–1$. Οι κόμβοι αριθμούνται ανά επίπεδο από αριστερά προς τα δεξιά και τοποθετούνται σε ένα πίνακα m θέσεων(τον pinax),όπως δείχνει το παράδειγμα του σχ. 6.4α.

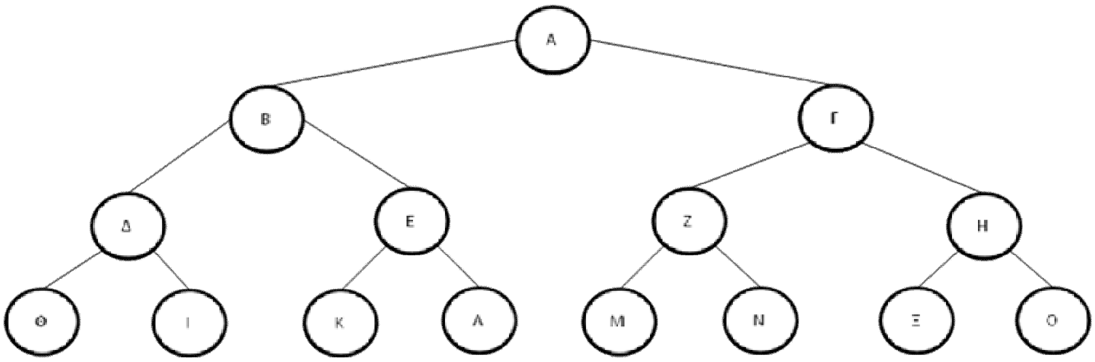
Έτσι, τα παιδιά του κόμβου με αύξοντα αριθμό 5 είναι τα 10 και 11, ενώ ο κόμβος πατέρας είναι ο 2.

Γενικά, για τον κόμβο x ισχύουν τα εξής:

i. Ο πατέρας του βρίσκεται στη θέση [x/2], αν $x\neq1$(ακέραια διαίρεση). Αν το x είναι 1, τότε ο κόμβος είναι η ρίζα.

ii. Το αριστερό παιδί βρίσκεται στη θέση 2x, εάν $2x\leq m$. Εάν $2x>m$, τότε ο κόμβος δεν έχει αριστερό παιδί.

iii. Το δεξί παιδί βρίσκεται στη θέση 2x+1,εάν 2x+1≤ m. Εάν 2x+1>m, τότε ο κόμβος δεν έχει δεξί παιδί.



(α)

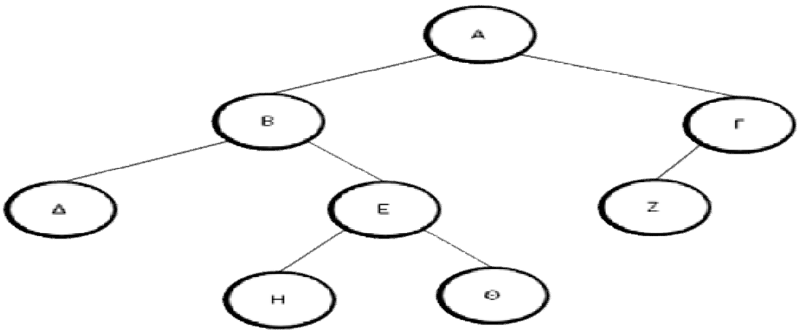
Pinax

Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(β)

Σχήμα 6.4

Με τον ίδιο τρόπο όπως και παραπάνω αποθηκεύονται *και μη πλήρη δένδρα*, όπως αυτό του σχ. 6.5α. Η αποθήκευση γίνεται στον πίνακα του σχ. 6.5β. Στην περίπτωση αυτή δεν χρησιμοποιούνται όλες οι θέσεις του πίνακα (αν και φυσικά έχουν δεσμευτεί), άρα υπάρχει σπατάλη μνήμης. Η ποσότητα της μνήμης που σπαταλείται εξαρτάται από τη μορφή του δένδρου. Για την αρίθμηση των κόμβων έχουν ληφθεί υπόψη και όσοι από τους κόμβους λείπουν, προκειμένου να υπάρχει πλήρες δένδρο.



(α)

Pinax

Α	Β	Γ	Δ	Ε	Ζ				Η	Θ				
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

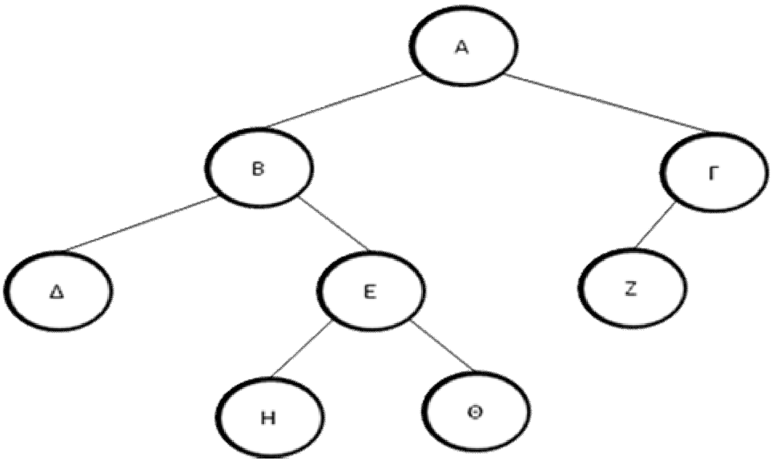
(β)

Σχήμα 6.5

ii. Για την αποφυγή του πιο πάνω προβλήματος οι κόμβοι του δένδρου αποθηκεύονται και πάλι σε ένα πίνακα, αλλά εκτός από τα δεδομένα *αποθηκεύονται και οι θέσεις του αριστερού και του δεξιού παιδιού*. Κάθε κόμβος περιγράφεται από μια δομή komvos. Αν υποθέσουμε ότι τα δεδομένα είναι ένας πίνακας χαρακτήρων 30 θέσεων, τότε η δομή περιγράφεται από το παρακάτω:

```
Struct komvos
{
        Char dedomena [30];
        Unsigned a_paidi;
        Unsigned d_paidi;
} TREE;
```

Ένα δυαδικό δένδρο μπορεί να παρασταθεί σαν ένας πίνακας τέτοιων δομών. Η αρίθμηση των κόμβων και η αποθήκευσή τους στη μνήμη με τον τρόπο αυτό για το δένδρο του σχ. 6.5α παρουσιάζεται στο σχ. 6.6



(α)

Dedomena	a_paidi	d_paidi	Θέση κόμβου
Α	2	3	1
Β	4	5	2
Γ	6	0	3
Δ	0	0	4
Ε	7	8	5
Ζ	0	0	6
Η	0	0	7
Θ	0	0	8

(β)

Σχήμα 6.6

Το πρόβλημα είναι ότι και σε αυτό τον τρόπο αποθήκευσης του δένδρου, ο πίνακας είναι σταθερών διαστάσεων. Στην επόμενη παράγραφο παρουσιάζεται ο τρόπος υλοποίησης των δυαδικών δένδρων με τη χρήση δεικτών.

6.5 Υλοποίηση δυαδικών δένδρων με δείκτες

Η χρήση των δεικτών δεν περιορίζει το πλήθος των κόμβων του δένδρου, όπως κάνει ο πίνακας της προηγούμενης παραγράφου. Για ένα δένδρο με μια ακέραια τιμή σε κάθε κόμβο, η δήλωση της δομής που περιγράφει κάθε κόμβο θα είναι η πιο κάτω:

```
Struct komvos

{

        Int rec;

        Struct komvos *lp;

        struct komvos *rp;

};
```

Το δένδρο οργανώνεται έτσι ώστε ο δείκτης lp κάθε κόμβου να δείχνει στο αριστερό παιδί του και ο δείκτης rp να δείχνει στο δεξί παιδί του. Αν δεν υπάρχει αριστερό ή δεξί παιδί, οι δείκτες αυτοί έχουν αντίστοιχα τιμή NULL. Ο τρόπος δημιουργίας, γεμίσματος και διάσχισης ενός δυαδικού δένδρου παρουσιάζεται στην επόμενη παράγραφο για μια ειδική κατηγορία δένδρων, τα δυαδικά δένδρα αναζήτησης.

6.6 Δυαδικά δένδρα αναζήτησης

Τα δένδρα αυτά αποτελούν ειδική κατηγορία. Είναι διατεταγμένα, δηλαδή δένδρα στα οποία η *διάταξη των παιδιών κάθε κόμβου έχει σημασία*. Στα δένδρα αναζήτησης *κάθε κόμβος έχει μεγαλύτερη τιμή από όλους τους κόμβους που βρίσκονται στα αριστερά του και μικρότερη από την τιμή όλων των κόμβων δεξιά του.*

Τα δυαδικά δένδρα χρησιμοποιούνται όταν απαιτείται γρήγορη προσπέλαση δεδομένων, για τα οποία υπάρχει κατάταξη.

i) Αλγόριθμος αναζήτησης:

Ο αλγόριθμος με βάση τον οποίο επισημαίνεται μια τιμή σε δυαδικό δένδρο αναζήτησης είναι ο εξής:

i. Συγκρίνεται η τιμή της ρίζας του δένδρου με την τιμή που αναζητείται.

ii. Αν η ζητούμενη τιμή είναι μικρότερη από τη ρίζα, τότε η αναζήτηση συνεχίζεται στο αριστερό υποδένδρο.

iii. Αν η ζητούμενη τιμή είναι μεγαλύτερη από τη ρίζα, τότε η αναζήτηση συνεχίζεται στο δεξί υποδένδρο.

iv. Η αναζήτηση συνεχίζεται μέχρι να βρεθεί ο κόμβος με την τιμή που αναζητείται ή να καταλήξει σε δείκτη με τιμή NULL.

ii) Αλγόριθμος εισαγωγής:

Ο αλγόριθμος με βάση τον οποίο εισάγεται μια τιμή σε δυαδικό δένδρο αναζήτησης είναι ο εξής:

i. Αρχικά ακολουθείται η διαδικασία αναζήτησης, αναζητώντας ένα κόμβο με τιμή ίση με αυτή του κόμβου που πρόκειται να προστεθεί. Τέτοια τιμή δεν υπάρχει και καταλήγουμε σε δείκτη NULL.

ii. Δεσμεύεται μνήμη για την τοποθέτηση της νέας τιμής και δημιουργείται ένας νέος κόμβος.

iii. Ο μηδενικός δείκτης στον οποίο καταλήγει η διαδικασία πιο πάνω τοποθετείται έτσι ώστε να δείχνει στο νέο κόμβο.

iii) Αλγόριθμος διαγραφής:

Ο αλγόριθμος με βάση τον οποίο διαγράφεται ένας κόμβος από δυαδικό δένδρο αναζήτησης είναι πιο περίπλοκος. Συγκεκριμένα:

i. Αν ο υπό διαγραφή κόμβος είναι φύλλο του δένδρου, τότε απλά διαγράφεται, μηδενίζοντας τον αντίστοιχο δείκτη ο οποίος ξεκινά από τον πατέρα του.

ii. Αν ο υπό διαγραφή κόμβος έχει ένα μόνο παιδί, τότε αντικαθίσταται από το παιδί του.

iii. Αν ο υπό διαγραφή κόμβος έχει δύο παιδιά, τότε αντικαθίσταται είτε από τον πιο δεξιό κόμβο του αριστερού υποδένδρου, είτε από τον πιο αριστερό κόμβο του δεξιού υποδένδρου.

ΒΙΒΛΙΟΓΡΑΦΙΑ

ΒΙΒΛΙΑ

1. ΧΡΗΣΤΟΣ ΚΟΙΛΙΑΣ *΄΄ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΟΡΓΑΝΩΣΗ ΑΡΧΕΙΩΝ΄΄*
2. ΠΑΝΑΓΙΩΤΗΣ ΛΙΝΑΡΔΗΣ*΄΄ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ΄΄*
3. ΓΕΩΡΓΙΟΣ ΔΗΜ. ΠΕΚΟΣ *΄΄ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΓΙΑ ΟΙΚΟΝΟΜΙΚΕΣ ΕΠΙΣΤΗΜΕΣ΄΄*
4. ΑΘ. Γ. ΤΣΟΥΡΟΠΛΗΣ και ΚΩΝ.Σ. ΚΛΗΜΟΠΟΥΛΟΣ *΄΄ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ΄΄*
5. GILMORE *΄΄ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΕΣ ΘΕΩΡΙΑ ΚΑΙ* ΕΦΑΡΜΟΓΕΣ΄΄ ΔΕΥΤΕΡΗ ΕΚΔΟΣΗ ΤΖΙΟΛΑ

ΗΛΕΚΤΡΟΝΙΚΕΣ ΠΗΓΕΣ

6. http://www.computinghistory.org.uk/det/5913/Gottfried%20Wilhelm%20Leibniz%20invents%20the%20Binary%20System
7. http://dtps.unipi.gr/files/notes/2007-2008/eksamino_1/arxitektonikes_ypologistwn/arithmitika_sistimata.pdf
8. http://www.ceid.upatras.gr/faculty/alexiou/eis_sys/material/SYS_YPOL.pdf
9. http://digitalschool.minedu.gov.gr/modules/ebook/show.php/DSB101/4/28,70/
10. http://anamorfosi.teiser.gr/ekp_yliko/e-notes/Data/circuits/main.htm
11. http://el.wikipedia.org/wiki/%CE%95%CE%BB%CE%BB%CE%B7%CE%BD%CE%B9%CE%BA%CF%8C_%CF%83%CF%8D%CF%83%CF%84%CE%B7%CE%BC%CE%B1_%CE%B1%CF%81%CE%AF%CE%B8%CE%BC%CE%B7%CF%83%CE%B7%CF%82
12. http://www.it.uom.gr/project/mycomputer/intro/calculat.html
13. http://www.kerryr.net/pioneers/binary.htm
14. http://www.slideshare.net/basheerahmad/week-09-numbering-system
15. http://users.teicrete.gr/taxd/02/notes/arithmetics/01.bits%20and%20bytes.htm#bmk3
16. http://www.cpusers.gr/showthread.php?t=210
17. http://el.wikipedia.org/wiki/ASCII
18. http://el.wikipedia.org/wiki/%CE%91%CF%81%CE%BC%CE%B5%CE%BD%CE%B9%CE%BA%CF%8C_%CF%83%CF%8D%CF%83%CF%84%CE%B7%CE%BC%CE%B1_%CE%B1%CF%81%CE%AF%CE%B8%CE%BC%CE%B7%CF%83%CE%B7%CF%82
19. http://www.cslab.ntua.gr/courses/csintro/files/binary-2005.pdf
20. http://www.cs.uoi.gr/~faturu/courses/2005/ds-section4.pdf
21. http://www.softlab.ntua.gr/~fotakis/data_structures/trees4.pdf