Αριθμητικά συστήματα - Κώδικες

Ένας οποιοσδήποτε αριθμός *N* μπορεί να εκφραστεί σ' ένα αριθμητικό σύστημα με βάση *r* ως εξής:

$(N)_r = a_n r^n + a_{n-1} r^{n-1} + ... + a_1 r^1 + a_0 r^0$ εάν $N>1$

$(N)_r = a_{-1} r^{-1} + a_{-2} r^{-2} + ... + a_{-n} r^{-n}$ εάν $0<N<1$

όπου *r*: η βάση του συστήματος.
n: 1,2,3,... ανάλογα με την περίπτωση.
α: ακέραιος αριθμός από 0 έως *r*-1.

Παραδείγματα:

$(956)_{10} = 9 \times 10^2 + 5 \times 10^1 + 6 \times 10^0$

$(123,456)_{10} = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2} + 6 \times 10^{-3}$

$(10111)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (23)_{10}$

$(110,101)_2 = 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = (6,625)_{10}$

$(765)_8 = 7 \times 8^2 + 6 \times 8^1 + 5 \times 8^0 = (501)_{10}$

$(A14,B6)_{16} = 10 \times 16^2 + 1 \times 16^1 + 4 \times 16^0 + 11 \times 16^{-1} + 6 \times 16^{-2} = (2580,7)_{10}$

Μερικά ενδιαφέροντα χαρακτηριστικά των συστημάτων αρίθμησης είναι τα ακόλουθα:

1) Ο αριθμός των ψηφίων που χρησιμοποιούνται σ' ένα σύστημα ισούται με την βάση του. Μάλιστα το μεγαλύτερο ψηφίο είναι: βάση-1.

Π.χ. Στο δυαδικό σύστημα τα ψηφία είναι 2. Το 0 και το 1. Επομένως κάθε αριθμός στο δυαδικό σύστημα θα γραφτεί με συνδυασμό των 0 και 1.

Τα ψηφία των σπουδαιότερων αριθμητικών συστημάτων

Δυαδικό	0	1														
Οκταδικό	0	1	2	3	4	5	6	7								
Δεκαδικό	0	1	2	3	4	5	6	7	8	9						
Δεκαεξαδικό	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
											(10)	(11)	(12)	(13)	(14)	(15)

2) Στο άθροισμα που εκφράζεται ένας αριθμός, κάθε ψηφίο του αριθμού πολλαπλασιάζεται με την βάση στην κατάλληλη δύναμη που εξαρτάται από τη θέση του ψηφίου.

Π.χ. $(1\ 1\ 0)_2$

$$1\times2^2 + 1\times2^1 + 0\times2^0$$

3) Στο άθροισμα που παριστάνει έναν αριθμό, το μέρος με τις αρνητικές δυνάμεις της βάσης εκφράζει το κλασματικό μέρος του αριθμού.

Π.χ. $(1230{,}41)_5$

$$1 \underbrace{\times 5^3 + 2 \times 5^2 + 3 \times 5^1 + 0 \times}_{\text{ακέραιο μέρος}} 5^0 + \underbrace{4 \times 5^{-1} + 1 \times}_{\text{κλασματικό μέρος}} 5^{-2}$$

-

Μετατροπή αριθμού από οποιοδήποτε σύστημα στο δεκαδικό

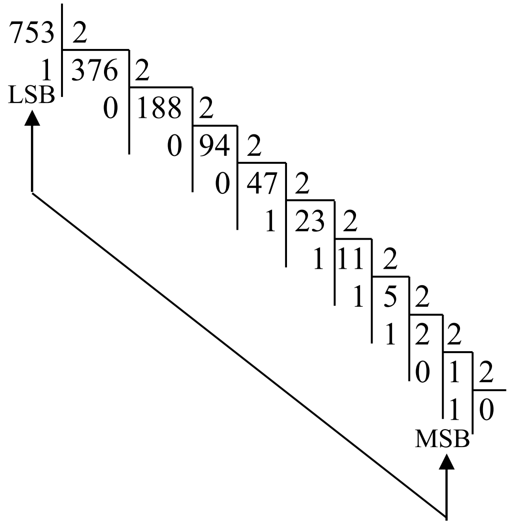
$(1011,1101)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4} = 8 + 0 + 2 + 1 + \frac{1}{2} + \frac{1}{4} + 0 + \frac{1}{16} = (11,8125)_{10}$

$(351,72)_8 = 3 \times 8^2 + 5 \times 8^1 + 1 \times 8^0 + 7 \times 8^{-1} + 2 \times 8^{-2} = (233,90625)_{10}$

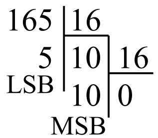
$(3B7C,1A)_{16} = 3 \times 16^3 + 11 \times 16^2 + 7 \times 16^1 + 12 \times 16^0 + 1 \times 16^{-1} + 10 \times 16^{-2} = (15228,1015625)_{10}$

Μετατροπή δεκαδικού σε οποιοδήποτε σύστημα

Παράδειγμα: $(753)_{10} = (1011110001)_2$



$(165)_{10} = (A5)_{16}$



$(320)_{10} = (500)_8$

```
320|8
  0|40|8
LSB|  0| 5|8
        5|0
      MSB
```

Μετατροπή κλασματικού αριθμού από το δεκαδικό σύστημα σε οποιοδήποτε άλλο

Για να μετατρέψουμε ένα κλασματικό αριθμό από το δεκαδικό σύστημα σε ένα άλλο σύστημα, πραγματοποιούμε διαδοχικούς πολλαπλασιασμούς με την βάση του συστήματος.

$0{,}8125 \times 2 = 1{,}6250 = 0{,}6250$ και πλεόνασμα **1** ◄ MSB
$0{,}6250 \times 2 = 1{,}2500 = 0{,}2500$ και πλεόνασμα **1**
$0{,}2500 \times 2 = 0{,}5000 = 0{,}5000$ και πλεόνασμα **0**
$0{,}5000 \times 2 = 1{,}0000 = 0{,}0000$ και πλεόνασμα **1** ◄ LSB
Άρα $(0{,}8125)_{10} = (0{,}1101)_2$

Έτσι λοιπόν $(753{,}8125)_{10} = (1011110001{,}1101)_2$

Από το παραπάνω παράδειγμα παρατηρούμε:
(α) Όταν κάποιο γινόμενο υπερβεί το 1, τότε στην επόμενη γραμμή γράφουμε το κλασματικό του μέρος, ενώ το ακέραιο μέρος 1 το κρατάμε ως πλεόνασμα και αποτελεί ψηφίο του αριθμού του νέου συστήματος.
(β) Οι διαδοχικοί πολλαπλασιασμοί συνεχίζονται είτε μέχρι να καλυφθεί το ίδιο μήκος λέξης, είτε μέχρι να πετύχουμε την ίδια ακρίβεια δεκαδικού και δυαδικού.
(γ) Αν το πλήθος των bits των αριθμών είναι n_2 και n_{10} αντίστοιχα, τότε για να έχουμε την ίδια ακρίβεια πρέπει:

$$2^{n_2} \cong 10^{n_{10}} \Rightarrow n_2 = \frac{n_{10}}{\log_{10} 2} \Rightarrow n_2 = 3{,}3 \times n_{10}$$

Έτσι, στο προηγούμενο παράδειγμα, που ήταν $n_{10} = 4 \Rightarrow n_2 = 13$ για να έχουν οι αριθμοί την ίδια ακρίβεια.

-

Μετατροπή αριθμού από ένα σύστημα σε άλλο
Γενικά ακολουθείται η πορεία:
(αριθμός στο Σ_1) $\rightarrow$ (αριθμός στο δεκαδικό) $\rightarrow$ (αριθμός στο Σ_2)

Μετατροπή οκταδικού αριθμού σε δυαδικό και αντίστροφα
$2^3 = 8$

Π.χ.

$$\begin{array}{ccc} 3 & 6 & 2 \\ \downarrow & \downarrow & \downarrow \\ 011 & 110 & 010 \end{array}$$

$(362)_8 = 011\ \ 110\ \ 010$

Άρα $(362)_8 = (011110010)_2$

Η μετατροπή δυαδικού σε οκταδικό είναι ακριβώς η αντίστροφη διαδικασία.

Π.χ. $(1010{,}01001)_2 = 001\ \ 010\ \ 010\ \ 010$

$$\begin{array}{cccc} 001 & 010 & 010 & 010 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 2 & 2 & 2 \end{array}$$

Άρα $(1010{,}01001)_2 = (12{,}22)_8$

Μετατροπή δεκαεξαδικού αριθμού σε δυαδικό και αντίστροφα
$2^4 = 16$

Π.χ.

$$\begin{array}{ccccc} A & 2 & C & D & 1 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 1010 & 0010 & 1100 & 1101 & 0001 \end{array}$$

$(A2C{,}D1)_{16} = 1010\ \ 0010\ \ 1100\ \ 1101\ \ 0001$

Άρα $(A2C{,}D1)_{16} = (101000101100{,}11010001)_2$

Π.χ. $(1011101{,}11101)_2 = 0101\ \ 1101\ \ 1110\ \ 1000$

$$\begin{array}{cccc} 0101 & 1101 & 1110 & 1000 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 5 & D & E & 8 \end{array}$$

Άρα $(1011101{,}11101)_2 = (5D{,}E8)_{16}$

Πίνακας των 16 πρώτων αριθμών στα διάφορα συστήματα αρίθμησης			
Δεκαδικός	**Δυαδικός**	**Οκταδικός**	**Δεκαεξαδικός**
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	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

Αριθμητικές πράξεις στο δυαδικό σύστημα

Πρόσθεση: Εκτελείται κατά τον ίδιο τρόπο όπως και η δεκαδική, ισχύουν δε οι ακόλουθοι νόμοι:

A	**B**	**A + B**
0	0	0
0	1	1
1	0	1
1	1	0 και 1 κρατούμενο

Π.χ.

$$\begin{array}{r} 110{,}01 \\ +\ \underline{111{,}01} \\ 1101{,}10 \end{array}$$

-

Αφαίρεση: Εκτελείται κατά τον ίδιο τρόπο όπως και η δεκαδική, ισχύουν δε οι ακόλουθοι νόμοι:

A	**B**	**A - B**
0	0	0
0	1	1 και 1 κρατούμενο
1	0	1
1	1	0

Π.χ.

```
   11000
 -  1111
 -------
    1001
```

Πολλαπλασιασμός: Εκτελείται κατά τον ίδιο τρόπο όπως και ο δεκαδικός, ισχύουν δε οι ακόλουθοι νόμοι:

A	**B**	**A x B**
0	0	0
0	1	0
1	0	0
1	1	1

Π.χ.

```
     110,1
 x    10,1
 ---------
      1101
     0000
    1101
 ---------
  10000,01
```

Διαίρεση: Εκτελείται κατά τον ίδιο τρόπο όπως και η δεκαδική, ισχύουν δε οι ακόλουθοι νόμοι:

A	**B**	**A : B**
0	1	0
1	1	1

Π.χ.

```
101011111 | 1101
 1101     |------
------    | 11011
010001
  1101
 -------
 0010011
     1101
 -------
   001101
     1101
 -------
     0000
```

Προσημασμένοι αριθμοί. Συμπληρώματα αριθμών

Όταν εκτελούνται αριθμητικές πράξεις σε έναν υπολογιστή, είναι πιο βολικό να χρησιμοποιείται για την παράσταση των αρνητικών αριθμών, ένα διαφορετικό σύστημα που ονομάζεται **μέθοδος του συμπληρώματος**. Με τη μέθοδο αυτή κάθε αρνητικός αριθμός δηλώνεται με το συμπλήρωμά του.

Το συμπλήρωμα ως προς 1 (1's complement)

Π.χ. το 1's complement του δυαδικού αριθμού 1011011001 είναι:

(1's) 1011011001 = 0100100110

Το συμπλήρωμα ως προς 2 (2's complement)

Π.χ. το 2's complement του δυαδικού αριθμού 1100 είναι:

(1's) 1100 = 0011

+ 1

0100

Με τη βοήθεια του συμπληρώματος οι πράξεις μεταξύ των δυαδικών αριθμών γίνονται γρηγορότερα και ευκολότερα, η δε αφαίρεση ανάγεται σε πρόσθεση.

-

Παράδειγμα

ΔΕΚΑΔΙΚΟΙ	ΔΥΑΔΙΚΟΙ	ΣΥΜΠΛΗΡΩΜΑΤΑ 1’s	2’s
19	10011	10011	10011
-9	- 01001	+ 10110	+ 10111
10	01010	1 01001	1 01010
		+ 1	αγνοείται
		01010	

- **Στο συμπλήρωμα ως προς 1 (1’s) η υπερχείλιση, εφόσον υπάρχει προστίθεται ενώ στο συμπλήρωμα ως προς 2 (2’s) αγνοείται.**
- **Στις αφαιρέσεις ο μειωτέος και ο αφαιρετέος πρέπει να παριστάνονται με τον ίδιο αριθμό ψηφίων (bits).**
- **Οι υπολογιστές λειτουργούν με τη μέθοδο του συμπληρώματος ως προς 2 (το γεγονός ότι το τελικό κρατούμενο αγνοείται είναι πλεονέκτημα).**

Σημείωση: Σε κάθε αριθμητικό σύστημα με βάση *r* υπάρχουν δύο συμπληρώματα: (α) το συμπλήρωμα ως προς *r* και (β) το συμπλήρωμα ως προς (*r*-1). Αν τώρα αντικατασταθεί η τιμή της βάσης με το 10 τότε έχουμε τα συμπληρώματα ως προς 10 και ως προς 9, για τους δεκαδικούς αριθμούς.

Συμπλήρωμα ως προς Βάση μείον 1

Ένας αριθμός N με ψηφία σε βάση r, έχει συμπλήρωμα ως προς (r-1) που ορίζεται από τη σχέση $\mathbf{(r^n - 1) - N}$.

Για δεκαδικούς αριθμούς, r=10 και r-1=9 το συμπλήρωμα ως προς 9 του N είναι $(10^n - 1) - N$. Το 10^n είναι ένας αριθμός που αποτελείται από ένα μόνο 1 ακολουθούμενο από n μηδέν. Το 10^n-1 είναι ένας αριθμός που αποτελείται από n 9. Αν n=5 έχουμε: $10^5 = 100000$ και 10^5-1=99999.

Το συμπλήρωμα ως προς 9 του 465320 είναι: **999999 – 465320 = 534679.**

Το συμπλήρωμα ως προς 9 του 002179 είναι: **999999 – 002179 = 997820.**

Σύμφωνα με όσα αναφέραμε το συμπλήρωμα ως προς (r-1) των οκταδικών ή δεκαεξαδικών αριθμών προκύπτει με αφαίρεση κάθε ψηφίου τους από το 7 ή το F, αντίστοιχα.

Συμπλήρωμα ως προς Βάση

Ένας αριθμός N με ψηφία σε βάση r, έχει συμπλήρωμα ως προς r που ορίζεται από τη σχέση:

$$r^n - N \text{ για } N \neq 0 \text{ και } 0 \text{ για } N = 0.$$

Συγκρίνοντας το συμπλήρωμα ως προς βάση με το συμπλήρωμα ως προς βάση μείον 1, παρατηρούμε ότι το συμπλήρωμα ως προς βάση προκύπτει αν προσθέσουμε ένα (1) στο συμπλήρωμα ως προς (r-1) αφού ισχύει: $r^n - N = [(r^n - 1) - N] + 1$. Έτσι, το συμπλήρωμα ως προς 10 του δεκαδικού 4328 είναι 5671 + 1 = 5672 που προκύπτει αν προσθέσουμε 1 στο συμπλήρωμα ως προς 9.

Εάν ο αριθμός *N*, του οποίου ζητάμε το συμπλήρωμα περιέχει υποδιαστολή, αυτή αφαιρείται προσωρινά, για να σχηματίσουμε το συμπλήρωμα ως προς *r* ή *r*-1. Μετά η υποδιαστολή ξαναμπαίνει στο νέο αριθμό (συμπλήρωμα) και στην ίδια σχετικά θέση.

$$r^n - (r^n - N) = N \quad \textbf{Συμπλήρωμα συμπληρώματος}$$

-

Αφαίρεση δύο αριθμών με τη βοήθεια συμπληρώματος

Για να αφαιρέσουμε έναν αριθμό (αφαιρετέο) *N*, από έναν άλλο αριθμό (μειωτέο) *M*, είναι γρηγορότερο και ευκολότερο να προσθέσουμε στο μειωτέο το συμπλήρωμα του αφαιρετέου. Έτσι, εξάλλου πραγματοποιούν την πράξη της αφαίρεσης και οι ηλεκτρονικοί υπολογιστές. Η αφαίρεση δύο θετικών αριθμών (M-N) που ανήκουν και οι δύο στην ίδια βάση r, και έχουν n ψηφία, γίνεται ως εξής:

(1) Προσθέτουμε τον μειωτέο M στο συμπλήρωμα ως προς r του αφαιρετέου N. Δηλαδή έχουμε: $M + (r^n - N) = M - N + r^n$ και διακρίνουμε 2 περιπτώσεις:

(2) Εάν $M \geq N$ τότε η παραπάνω σχέση γράφεται: $M + (r^n - N) = r^n + M - N$.
Στην περίπτωση αυτή, ο όρος r^n είναι το τελικό κρατούμενο της πρόσθεσης το οποίο αγνοείται στο τελικό αποτέλεσμα.

(3) Εάν $M < N$ τότε η παραπάνω σχέση γράφεται: $M + (r^n - N) = r^n - (N - M)$.
Δηλαδή, το αποτέλεσμα της πρόσθεσης δεν έχει τελικό κρατούμενο και είναι το συμπλήρωμα ως προς r του (N-M). Άρα, για να βρούμε το τελικό αποτέλεσμα θα πρέπει να πάρουμε το συμπλήρωμα του αποτελέσματος της πρόσθεσης και να βάλουμε το πρόσημο μείον μπροστά.

<u>Παραδείγματα</u>

(α) $(72532)_{10} - (3250)_{10}$

Επειδή οι δύο αριθμοί (*M*, *N*) πρέπει να έχουν τον ίδιο αριθμό ψηφίων, προσθέτουμε ως περισσότερο σημαντικά ψηφία του μικρότερου σε μήκος αριθμού, μηδενικά (ο αριθμός 3250 γράφεται 03250).

Το συμπλήρωμα ως προς 10 του αφαιρετέου *N*=03250 είναι: 100000–03250 = 96750.

$$
\begin{array}{lll}
\text{Άρα:} & M: & 72532 \\
 & N: & \underline{96750} \\
 & & 1)69282
\end{array}
$$

Το αποτέλεσμα είναι 69282 και το τελικό κρατούμενο 1 παραλείπεται.

(β) $(3250)_{10} - (72532)_{10}$

Το συμπλήρωμα ως προς 10 του αφαιρετέου *N*=72532 είναι: 100000–72532 = 27468.

$$
\begin{array}{lll}
\text{Άρα:} & M: & 03250 \\
 & N: & \underline{27468} \\
 & & 30718
\end{array}
$$

Επειδή δεν προκύπτει τελικό κρατούμενο, το αποτέλεσμα της αφαίρεσης είναι αρνητικό (δηλ. $M < N$). Άρα για να βρούμε το τελικό αποτέλεσμα θα πρέπει να πάρουμε το συμπλήρωμα του παραπάνω αποτελέσματος και να βάλουμε το πρόσημο μείον μπροστά.

Δηλαδή:

- (10's) 30718 = -69282.

(γ) $(6324)_8 - (3565)_8$

Το συμπλήρωμα ως προς 8 του αφαιρετέου *N*=3565 προκύπτει αν στο συμπλήρωμα ως προς 7 προσθέσουμε ένα 1. Δηλαδή: (7's) 3565 = 4212.

Έχουμε: (8's) 3565 = 4212 +1 = 4213.

-

Άρα: M: 6324
N: 4213

$$\begin{array}{r} 6324 \\ \underline{4213} \\ 1)2537 \end{array}$$

Το αποτέλεσμα είναι 2537 και το τελικό κρατούμενο 1 παραλείπεται.

(δ) $(3565)_8 - (6324)_8$

Το συμπλήρωμα ως προς 7 του αφαιρετέου N=6324 είναι: 7777 – 6324 = 1453.

Έχουμε ότι το συμπλήρωμα ως προς 8 είναι: 1453 +1 = 1454.

Άρα: M: 3565
N: 1454

$$\begin{array}{r} 3565 \\ \underline{1454} \\ 5241 \end{array}$$

Επειδή δεν προκύπτει τελικό κρατούμενο, το αποτέλεσμα της αφαίρεσης είναι αρνητικό (δηλ. $M < N$). Άρα για να βρούμε το τελικό αποτέλεσμα θα πρέπει να πάρουμε το συμπλήρωμα του παραπάνω αποτελέσματος και να βάλουμε το πρόσημο μείον μπροστά. Δηλαδή:

- (8's) 5241 = -2537.

(ε) $(18B)_{16} - (7A)_{16}$

Το συμπλήρωμα ως προς 16 του αφαιρετέου N=7A προκύπτει αν στο συμπλήρωμα ως προς 15 προσθέσουμε ένα 1. Δηλαδή: (15's) 07A = FFF – 07A = F85.

Έχουμε: (16's) 07A = F85 +1 = F86.

Άρα: M: 18B
N: F86

$$\begin{array}{r} 18B \\ \underline{F86} \\ 1)111 \end{array}$$

Το αποτέλεσμα είναι 111 και το τελικό κρατούμενο 1 παραλείπεται.

(στ) $(7A)_{16} – (18B)_{16}$

Το συμπλήρωμα ως προς 15 του αφαιρετέου *N*=18B είναι: FFF – 18B = E74.

Έχουμε ότι το συμπλήρωμα ως προς 16 είναι: E74 +1 = E75.

$$\begin{array}{lll} \text{Άρα:} & M: & 07A \\ & N: & \underline{E75} \\ & & EEF \end{array}$$

Επειδή δεν προκύπτει τελικό κρατούμενο, το αποτέλεσμα της αφαίρεσης είναι αρνητικό (δηλ. $M < N$). Άρα για να βρούμε το τελικό αποτέλεσμα θα πρέπει να πάρουμε το συμπλήρωμα του παραπάνω αποτελέσματος και να βάλουμε το πρόσημο μείον μπροστά. Δηλαδή:

- (16’s) EEF = -111.

ζ) Εάν X = 1010101 και Y = 1000100, να εκτελεστούν οι αφαιρέσεις X – Y και Y - X.

(i)

$$\begin{array}{rcl} X & = & 1010101 \\ Y & = & \underline{0111100} \\ \text{άθροισμα} & = & 1)0010001 \end{array}$$

Με απόρριψη του τελικού κρατουμένου προκύπτει: X – Y = 00010001.

(ii)

$$\begin{array}{rcl} Y & = & 1000100 \\ X & = & \underline{0101011} \\ \text{άθροισμα} & = & 1101111 \end{array}$$

Επειδή δεν προκύπτει τελικό κρατούμενο το τελικό αποτέλεσμα θα είναι:

Y – X = - (2’s) 1101111 = -0010001.

-

η) Να γίνουν οι παρακάτω αφαιρέσεις, με συμπλήρωμα ως προς 1 (1’s) και συμπλήρωμα ως προς 2 (2’s).

(i) 110111 – 101110 (ii) 1111001 – 0110110 (iii) 11100011 – 11110100

ΣΥΜΠΛΗΡΩΜΑ ΩΣ ΠΡΟΣ 1		ΣΥΜΠΛΗΡΩΜΑ ΩΣ ΠΡΟΣ 2
(i)	110111 010001 1 001000 + 1 001001	110111 010010 1 001001 αγνοείται
(ii)	1111001 1001001 1 1000010 + 1 1000011	1111001 1001010 1 1000011 αγνοείται
(iii)	11100011 00001011 11101110	11100011 00001100 11101111

- (1’s) 11101110 = -00010001 = $(-17)_{10}$ - (2’s) 11101111 = -00010001 = $(-17)_{10}$

Κώδικες

Δυαδικοί Κώδικες

Δυαδικός κώδικας είναι ένας συστηματικός τρόπος παράστασης πληροφοριών σε δυαδική μορφή.
Για παράδειγμα, οι τέσσερις εποχές του χρόνου θα μπορούσαν να παρασταθούν ως εξής:

Άνοιξη	↔	00
Καλοκαίρι	↔	01
Φθινόπωρο	↔	10
Χειμώνας	↔	11

Η παραπάνω αντιστοιχία των εποχών με δυαδικούς αριθμούς είναι ένας δυαδικός κώδικας. Η αντιστοιχία αυτή δεν είναι μοναδική και επιλέγεται ανάλογα με την εφαρμογή.

Αν το πλήθος των στοιχείων που πρόκειται να κωδικοποιηθούν δεν είναι δύναμη του 2, τότε μερικοί από τους συνδυασμούς των bits δεν χρησιμοποιούνται.

Για παράδειγμα, τα 10 ψηφία του δεκαδικού συστήματος μπορούν να παρασταθούν με έναν δυαδικό κώδικα των 4 bits. Με 4 bits, όμως, μπορούν να αναπτυχθούν 16 συνδυασμοί. Επομένως, δεν χρησιμοποιούνται 6 συνδυασμοί.
Οι δυαδικοί κώδικες ανήκουν στις δύο ακόλουθες κατηγορίες ανάλογα με τον τρόπο κατασκευής τους:

- **δυαδικοί κώδικες με βάρη**
- **δυαδικοί κώδικες χωρίς βάρη**

Δυαδικοί κώδικες με βάρη

Οι δυαδικοί κώδικες με βάρη κατασκευάζονται με τέτοιον τρόπο ώστε στη θέση κάθε bit του κώδικα να αντιστοιχεί ένα βάρος (κάθε θέση έχει μία αξία).

Οι ακόλουθοι δυαδικοί κώδικες με βάρη στα bits ανάλογα με τη θέση τους, χρησιμοποιούνται για την κωδικοποίηση των 10 ψηφίων του δεκαδικού συστήματος:

- **ο BCD κώδικας που χρησιμοποιεί 4 bits με βάρη 8 4 2 1.**
- **ο κώδικας με βάρη 2 4 2 1 που χρησιμοποιεί 4 bits με βάρη 2 4 2 1.**
- **ο Biquinary κώδικας που χρησιμοποιεί 7 bits με βάρη 5 0 4 3 2 1 0.**

Ο κώδικας BCD

Ο κώδικας BCD είναι δυαδικός κώδικας με βάρη, που χρησιμοποιείται για την κωδικοποίηση των 10 ψηφίων του δεκαδικού συστήματος, όπως δηλώνει άλλωστε το όνομά του: Binary Coded Decimal (δυαδικά κωδικοποιημένο δεκαδικό).

Κώδικες

Δεκαδικό ψηφίο	**BCD 8421**	**2421**	**5211**	**Excess-3**	**Biquinary 5043210**
0	0000	0000	0000	0011	0100001
1	0001	0001	0001	0100	0100010
2	0010	0010	0011	0101	0100100
3	0011	0011	0101	0110	0101000
4	0100	0100	0111	0111	0110000
5	0101	1011	1000	1000	1000001
6	0110	1100	1010	1001	1000010
7	0111	1101	1100	1010	1000100
8	1000	1110	1110	1011	1001000
9	1001	1111	1111	1100	1010000

Ο κώδικας BCD είναι ένας τρόπος παράστασης των 10 ψηφίων του δεκαδικού συστήματος, το κάθε ένα από τα οποία αντιστοιχεί σε μία τετράδα bits.

Για παράδειγμα, ο δεκαδικός αριθμός 5 αντιστοιχεί στην τετράδα 0101 ($0\times8+1\times4+0\times2+1\times1=5$).

Μετατροπή από BCD σε δεκαδικό

Για τη μετατροπή ενός BCD αριθμού σε δεκαδικό αριθμό χωρίζεται ο BCD αριθμός σε ομάδες **τεσσάρων** (4) bits και κάθε ομάδα μετατρέπεται στο ισοδύναμο δεκαδικό ψηφίο, σύμφωνα με τον Πίνακα 2.8.1.

Για παράδειγμα, ο BCD αριθμός 1000011000101001 αντιστοιχεί στο δεκαδικό αριθμό 8629 αφού:

1000	0110	0010	1001
8	6	2	9

Παρατήρηση. Ο κώδικας BCD χρησιμοποιεί τους 10 από τους 16 δυνατούς συνδυασμούς των 4 bits. Οι 6 συνδυασμοί 1010, 1011, 1100, 1101, 1110 και 1111 δεν χρησιμοποιούνται.

Μετατροπή από δεκαδικό σε BCD

Για τη μετατροπή ενός δεκαδικού αριθμού σε BCD αριθμό, μετατρέπεται κάθε ψηφίο του δεκαδικού αριθμού σε μία ομάδα **τεσσάρων** (4) bits που αποτελούν τον ισοδύναμο BCD αριθμό του κάθε δεκαδικού ψηφίου, σύμφωνα με τον Πίνακα 2.8.1.

Για παράδειγμα, ο δεκαδικός αριθμός 4738 αντιστοιχεί στον BCD αριθμό 0100011000111000 αφού:

4	7	3	8
0100	0110	0011	1000

Αριθμοί του κώδικα BCD και δυαδικοί αριθμοί

Ο κώδικας BCD δεν είναι ένα άλλο αριθμητικό σύστημα (όπως το δεκαδικό, το δυαδικό, το οκταδικό, το δεκαεξαδικό), αλλά είναι ένας τρόπος παράστασης των 10 ψηφίων του δεκαδικού συστήματος, το κάθε ένα από τα οποία αντιστοιχεί σε μία τετράδα bits.

Ο κώδικας BCD είναι ένας άμεσος δυαδικός μετατροπέας μόνο για τους δεκαδικούς αριθμούς 0-9. Για τους δεκαδικούς αριθμούς που είναι μεγαλύτεροι από 9, η κωδικοποίηση και η μετατροπή είναι διαφορετικές.

Για παράδειγμα, ο δεκαδικός αριθμός 253 αντιστοιχεί:
- στον 12-bits BCD αριθμό 001001010011
- στον 8-bits δυαδικό αριθμό 11111101

Δυαδικοί κώδικες χωρίς βάρη

Στους δυαδικούς κώδικες χωρίς βάρη η θέση κάθε bit του κώδικα δεν αντιστοιχεί σε κάποιο βάρος, όπως γίνεται στους δυαδικούς κώδικες με βάρη. **Αυτοί οι κώδικες προκύπτουν από κάποιον κανόνα.**

Τέτοιοι δυαδικοί κώδικες χωρίς βάρη είναι οι ακόλουθοι:

- **ο κώδικας Gray**
- **ο κώδικας υπερβολής κατά 3 (excess-3)**

Ο κώδικας Excess-3 χρησιμοποιεί επίσης 4 bits για να παραστήσει τους δεκαδικούς αριθμούς 0 έως 9, όπως φαίνεται στον Πίνακα 2.8.1. Το όνομά του προκύπτει από το ότι η παράσταση κάθε δεκαδικού ψηφίου 0 έως 9 στον κώδικα αυτόν, είναι μεγαλύτερη κατά 3 από την αντίστοιχη του κώδικα BCD, **δηλαδή ο κώδικας Excess-3 προκύπτει από τον κώδικα BCD προσθέτοντας 3.** Το πλεονέκτημά του σε σχέση με τον κώδικα BCD είναι ότι είναι αυτό-συμπληρούμενος κώδικας όπως θα δούμε ευθύς αμέσως.

Από τον πίνακα 2.8.1 βλέπουμε ότι

$$(1)_{10} = (0100)_{\text{Excess-3}}$$

$$(8)_{10} = (1011)_{\text{Excess-3}}$$

Δηλαδή, τα αριστερά μέλη των ισοτήτων είναι το συμπλήρωμα ως προς 9 το ένα του άλλου, ενώ τα δεξιά μέλη είναι το συμπλήρωμα ως προς 1 το ένα του άλλου. Ομοίως,

$$(4)_{10} = (0111)_{\text{Excess-3}}$$

$$(5)_{10} = (1000)_{\text{Excess-3}}$$

Άρα, ο κώδικας Excess-3 μπορεί να χρησιμοποιηθεί για να εκτελέσουμε την αφαίρεση δεκαδικών αριθμών με τη μέθοδο συμπληρώματος ως προς 9. Όμως, ο κώδικας Excess-3 δεν είναι κώδικας με βάρη όπως ο BCD. Για παράδειγμα,

$$\left(\frac{0111}{4} + \frac{0101}{2} \neq \frac{1001}{6}\right)_{\text{Excess}-3}$$

Ενώ

$$\left(\frac{0100}{4} + \frac{0101}{5} = \frac{1001}{9}\right)_{\text{BCD}}$$

Ο κώδικας 2421 συνδυάζει τα πλεονεκτήματα τόσο του BCD όσο και του Excess-3, δηλαδή, είναι αυτό-συμπληρούμενος και κώδικας με βάρη. Για παράδειγμα,

$$(2)_{10} = (0010)_{2421}$$

$$(7)_{10} = (1101)_{2421}$$

Δηλαδή, τα αριστερά μέλη των ισοτήτων είναι το συμπλήρωμα ως προς 9 το ένα του άλλου, ενώ τα δεξιά μέλη είναι το συμπλήρωμα ως προς 1 το ένα του άλλου. Επίσης,

$$\left(\frac{0100}{4} + \frac{1011}{5} = \frac{1111}{9}\right)_{2421}$$

-

Ο Κώδικας GRAY

Ο κώδικας Gray είναι δυαδικός κώδικας χωρίς βάρη που χρησιμοποιείται για την κωδικοποίηση των δεκαδικών αριθμών (όχι μόνο των 10 ψηφίων του δεκαδικού συστήματος, όπως γίνεται στον κώδικα BCD).

Ο κώδικας Gray που χρησιμοποιεί 4 bits (κωδικοποίηση των 16 πρώτων δεκαδικών αριθμών 0-15) παρουσιάζεται στον Πίνακα 2.8.2.

Ο κώδικας Gray ονομάζεται κατοπτρικός κώδικας, λόγω του τρόπου κατασκευής του.

Στον Πίνακα φαίνεται ότι:

Η πρώτη στήλη από δεξιά (LSB) ξεκινάει πρώτα με ένα "0" και μετά με ένα "1". Αυτά είναι τα 2 πρώτα κατακόρυφα bits. Τα επόμενα 2 κατακόρυφα bits είναι κατοπτρικά των 2 πρώτων bits (υπάρχει συμμετρία ως προς τη μέση τους). Έτσι, δημιουργούνται 4 bits. Τα επόμενα 4 κατακόρυφα bits είναι κατοπτρικά των 4 πρώτων bits. Έτσι, δημιουργούνται 8 bits. Τα επόμενα 8 bits είναι κατοπτρικά των 8 πρώτων bits.

Η δεύτερη στήλη από δεξιά ξεκινάει πρώτα με δύο "0" και μετά με δύο "1". Τα επόμενα 4 bits είναι κατοπτρικά των 4 πρώτων bits. Έτσι, δημιουργούνται 8 bits. Τα επόμενα 8 bits είναι κατοπτρικά των 8 πρώτων bits.

Αυτή η διαδικασία επαναλαμβάνεται και στις επόμενες στήλες. Η τρίτη στήλη από δεξιά ξεκινάει πρώτα με τέσσερα "0" και μετά με τέσσερα "1" και είναι κατοπτρική ως προς το μέσον της. Η τέταρτη στήλη από δεξιά ξεκινάει πρώτα με οκτώ "0" και μετά με οκτώ "1".

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

Στον κώδικα Gray αλλάζει ένα μόνο bit μεταξύ δύο διαδοχικών αριθμών

Αν χρησιμοποιούνται δυαδικοί αριθμοί για τη μετάβαση από έναν αριθμό στον επόμενο, τότε υπάρχει η πιθανότητα σφάλματος: η μετάβαση από το 0111 (7) στο 1000 (8) μπορεί να οδηγήσει (για μικρό χρονικό διάστημα) στο 0110 (4) αν το LSB αλλάζει γρηγορότερα από τα άλλα bits, με αποτέλεσμα να γίνει λάθος στη μετατροπή. Αν χρησιμοποιείται ο κώδικας Gray για τη μετάβαση από έναν αριθμό στον επόμενο, τότε η πιθανότητα σφάλματος εξαλείφεται: η μετάβαση από το 0100 (7) στο 1100 (8) επιτυγχάνεται με την αλλαγή ενός (1) μόνο bit.

Ο κώδικας excess-3 προκύπτει από τον κώδικα BCD προσθέτοντας 3.

Αλφαριθμητικοί κώδικες

Πολλές εφαρμογές των ηλεκτρονικών υπολογιστών απαιτούν τη χρήση δεδομένων που αποτελούνται από αριθμούς αλλά και από γράμματα και από ειδικούς χαρακτήρες. Για παράδειγμα, το λογιστήριο μιας εταιρείας χρησιμοποιεί ηλεκτρονικό υπολογιστή για να επεξεργάζεται τα αρχεία της μισθοδοσίας της εταιρείας. Για να παρασταθούν τα ονόματα των εργαζομένων σε δυαδική μορφή, πρέπει να υπάρχει ένας δυαδικός κώδικας για το αλφάβητο. Για να παρασταθούν οι μισθοί των εργαζομένων σε δυαδική μορφή πρέπει να υπάρχει ένας δυαδικός κώδικας για τους δεκαδικούς αριθμούς και για κάποιους ειδικούς χαρακτήρες, όπως είναι ο χαρακτήρας ”$”.
Οι **αλφαριθμητικοί χαρακτήρες** περιλαμβάνουν:

- τα 26 κεφαλαία γράμματα του αγγλικού αλφαβήτου A-Z
- τα 26 μικρά γράμματα του αγγλικού αλφαβήτου a-z
- τα 10 δεκαδικά ψηφία 0-9
- τους ειδικούς χαρακτήρες (τα σημεία στίξης όπως ! . , ? και άλλοι χαρακτήρες όπως @ # $ % & * + /).

Ένας αλφαριθμητικός κώδικας είναι ένας συστηματικός τρόπος παράστασης των αλφαριθμητικών χαρακτήρων σε δυαδική μορφή. Κάθε αλφαριθμητικός χαρακτήρας παριστάνεται με μία ομάδα bits, το μέγεθος της οποίας εξαρτάται από το πλήθος των αλφαριθμητικών χαρακτήρων που παριστάνει ο κώδικας.

Τέτοιοι δυαδικοί αλφαριθμητικοί κώδικες είναι οι ακόλουθοι:

- ο κώδικας ASCII που χρησιμοποιεί 7 bits
- ο κώδικας Baudot που χρησιμοποιεί 5 bits

-

Ο Κώδικας ASCII

Ο πλέον συχνά χρησιμοποιούμενος δυαδικός αλφαριθμητικός κώδικας είναι ο κώδικας ASCII (American Standard Code for Information Interchange) ο οποίος χρησιμοποιεί 7 bits για την κωδικοποίηση 128 χαρακτήρων.

Ο κώδικας ASCII περιλαμβάνει 94 εκτυπώσιμους γραφικούς χαρακτήρες και 34 μη εκτυπώσιμους χαρακτήρες ελέγχου (control characters), δηλαδή συνολικά 128 χαρακτήρες που παρουσιάζονται στον Πίνακα 2.8.3.

Οι εκτυπώσιμοι χαρακτήρες είναι:

- τα 26 κεφαλαία γράμματα του αγγλικού αλφαβήτου A-Z
- τα 26 μικρά γράμματα του αγγλικού αλφαβήτου a-z
- οι 10 αριθμοί 0-9
- οι 32 ειδικοί χαρακτήρες.

Οι χαρακτήρες ελέγχου χωρίζονται σε:

- διαμορφωτές μορφής
- διαχωριστές πληροφορίας
- χαρακτήρες ελέγχου-επικοινωνίας.

Οι ηλεκτρονικοί υπολογιστές συνήθως χρησιμοποιούν δυαδικές λέξεις των 8 bits (1 byte), ενώ ο κώδικας ASCII χρησιμοποιεί 7 bits. Έτσι, κάθε χαρακτήρας του κώδικα ASCII συνήθως αναπαρίσταται με 1 byte των 8 bits, οπότε μπορεί να γίνει κωδικοποίηση 256 χαρακτήρων. Για την κωδικοποίηση των 128 χαρακτήρων του κώδικα ASCII χρησιμοποιείται το MSB με τιμή “0” (και τα υπόλοιπα 7 bits είναι τα 7 bits του κώδικα ASCII).

Κώδικας ASCII

$b_7b_6b_5$ $b_4b_3b_2b_1$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	'	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	-	o	DEL

Παράδειγμα: Η λέξη **bit** στον κώδικα ASCII είναι:

b	i	t
1100010	1101001	1110100

Για την κωδικοποίηση άλλων χαρακτήρων (για παράδειγμα τα γράμματα του ελληνικού αλφαβήτου) χρησιμοποιείται το MSB με τιμή “1”. Με τον τρόπο αυτόν έχει προκύψει το Πρότυπο ΕΛΟΤ-928 του Ελληνικού Οργανισμού Τυποποίησης που είναι εγκεκριμένο από την ISO (International Standards Organisation).

-

Άλγεβρα Boole

Η δυαδική λογική αποτελείται από δυαδικές μεταβλητές και λογικές πράξεις. Ορίζονται τρεις βασικές λογικές πράξεις: “ΚΑΙ” (AND), “Η” (OR) και “ΟΧΙ” (NOT).

1. ΚΑΙ (AND): $\mathbf{x \cdot y = z}$ ή $\mathbf{xy = z}$ διαβάζεται “x ΚΑΙ y ίσον z”.

Η λογική πράξη ΚΑΙ (AND) σημαίνει ότι το $z = 1$ όταν και μόνον όταν το $x = 1$ και το $y = 1$, διαφορετικά το $z = 0$.

2. Ή (OR): $\mathbf{x + y = z}$ διαβάζεται “x Ή y ίσον z”, που σημαίνει ότι το $z = 1$ αν το $x = 1$ ή αν το $y = 1$ ή αν και τα δύο είναι $x = 1$ και $y = 1$. Αν και τα δύο είναι $x = 0$ και $y = 0$, τότε $z = 0$.

3. ΟΧΙ (NOT): $\mathbf{\overline{x} = z}$ διαβάζεται “όχι x ίσον z”, που σημαίνει ότι το z είναι το “αντίθετο” ή “αντίστροφο” του x: αν το $x = 1$, τότε το $z = 0$ και αν το $x = 0$, τότε το $z = 1$.

ΚΑΙ (AND)		
x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

‘Η (OR)		
x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

ΟΧΙ (NOT)	
x	$\overline{x}$
0	1
1	0

Η άλγεβρα Boole είναι μια αλγεβρική δομή ορισμένη πάνω σ’ ένα σύνολο στοιχείων B, μαζί με τους δύο δυαδικούς τελεστές (πράξεις) + και · , αρκεί να ικανοποιούνται τα παρακάτω αξιώματα:

1. (α) Κλειστή ως προς τον τελεστή + .
(β) Κλειστή ως προς τον τελεστή · .

2. (α) Ένα **ουδέτερο στοιχείο ως προς +,** που συμβολίζεται με 0: $\mathbf{x + 0 = 0 + x = x}$.
(β) Ένα **ουδέτερο στοιχείο ως προς ·,** που συμβολίζεται με 1: $\mathbf{x \cdot 1 = 1 \cdot x = x}$.

3. (α) Αντιμεταθετική ως προς +: $x + y = y + x$.
   (β) Αντιμεταθετική ως προς · : $x \cdot y = y \cdot x$.

4. (α) Ο · είναι επιμεριστικός ως προς + : $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$.
   (β) Ο + είναι επιμεριστικός ως προς · : $x + (y \cdot z) = (x + y) \cdot (x + z)$.

5. Για κάθε στοιχείο $x \in B$, υπάρχει ένα στοιχείο $\overline{x} \in B$ (που ονομάζεται **"συμπλήρωμα"** του x) τέτοιο ώστε:
(α) $x + \overline{x} = 1$ και (β) $x \cdot \overline{x} = 0$.

6. Υπάρχουν τουλάχιστον δύο στοιχεία $x, y \in B$ που να είναι **$x \neq y$.**

Βασικά Θεωρήματα της άλγεβρας Boole

<u>ΘΕΩΡΗΜΑ 1</u> (α): **$x + x = x$.**

$x + x = (x + x) \cdot 1$	από το αξίωμα : 2(β)		
$= (x + x)(x + \overline{x})$	5(α)		
$= x + x\overline{x}$	4(β)		
$= x + 0$	5(β)	$= x$	2(α)

<u>ΘΕΩΡΗΜΑ 1</u> (β): **$x \cdot x = x$.**

$x \cdot x = xx + 0$	από το αξίωμα : 2(α)		
$= xx + x\overline{x}$	5(β)		
$= x(x + \overline{x})$	4(α)		
$= x \cdot 1$	5(α)	$= x$	2(β)

<u>ΘΕΩΡΗΜΑ 2</u> (α): **$x + 1 = 1$.**

$x + 1 = 1 \cdot (x + 1)$	από το αξίωμα : 2(β)		
$= (x + \overline{x})(x + 1)$	5(α)		
$= x + \overline{x} \cdot 1$	4(β)		
$= x + \overline{x}$	2(β)	$= 1$	5(α)

-

<u>ΘΕΩΡΗΜΑ 2</u> (β): $x \cdot 0 = 0$ από τον δυϊσμό.

<u>ΘΕΩΡΗΜΑ 3</u> : $\bar{\bar{x}} = x$. Από το αξίωμα 5, έχουμε ότι $x + \bar{x} = 1$ και $x\bar{x} = 0$, που ορίζουν το συμπλήρωμα του x. Άρα το συμπλήρωμα του $\bar{x}$ είναι το x και επίσης το $\bar{\bar{x}}$. Επομένως, αφού το συμπλήρωμα είναι μοναδικό, έχουμε ότι $\bar{\bar{x}} = x$.

<u>ΘΕΩΡΗΜΑ 4</u> (α): $x + xy = x$.

$x + xy = x \cdot 1 + xy$	από το αξίωμα : 2(β)	
$= x(1 + y)$	5(α)	
$= x(y + 1)$	4(α)	
$= x \cdot 1$	5(α)	$= x$ 2(β)

<u>ΘΕΩΡΗΜΑ 4</u> (β): $x(x + y) = x$ από τον δυϊσμό.

Τα θεωρήματα της άλγεβρας Boole μπορούν να αποδειχτούν και με τη βοήθεια των πινάκων αληθείας.

<u>ΘΕΩΡΗΜΑ 5</u> (α): $x + (y + z) = (x + y) + z$.

<u>ΘΕΩΡΗΜΑ 5</u> (β): $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ από τον δυϊσμό.

<u>ΘΕΩΡΗΜΑ 6</u> (α): $\overline{(x + y)} = \bar{x}\,\bar{y}$ (De Morgan).

<u>ΘΕΩΡΗΜΑ 6</u> (β): $\overline{(xy)} = \bar{x} + \bar{y}$ από τον δυϊσμό (De Morgan).

x	y	$x + y$	$\overline{(x + y)}$	$\bar{x}$	$\bar{y}$	$\bar{x}\,\bar{y}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Η γενικευμένη μορφή του ανωτέρω θεωρήματος De Morgan λέει ότι το συμπλήρωμα μιας συνάρτησης μπορεί να βρεθεί εναλλάσσοντας τους τελεστές AND με OR και συμπληρώνοντας κάθε όρο. Π.χ. το συμπλήρωμα της λογικής συνάρτησης $F = \bar{x}yz + \bar{x}\bar{y}z$ θα δίνεται από

$$\bar{F} = \left(\overline{\bar{x}y\bar{z} + \bar{x}\bar{y}z}\right) = \left(\overline{\bar{x}y\bar{z}}\right)\left(\overline{\bar{x}\bar{y}z}\right) = \left(x + \bar{y} + z\right)\left(x + y + \bar{z}\right)$$

Στον πιο κάτω πίνακα δίνονται συνοπτικά τα αξιώματα και τα κυριότερα θεωρήματα της άλγεβρας Boole.

Αξίωμα 2	(α) $x + 0 = x$	(β) $x \cdot 1 = x$
Αξίωμα 5	(α) $x + \bar{x} = 1$	(β) $x \cdot \bar{x} = 0$
Θεώρημα 1	(α) $x + x = x$	(β) $x \cdot x = x$
Θεώρημα 2	(α) $x + 1 = 1$	(β) $x \cdot 0 = 0$
Θεώρημα 3, (δύο αρνήσεις)	$\bar{\bar{x}} = x$	
Αξίωμα 3, αντιμεταθετική	(α) $x + y = y + x$	(β) $x \cdot y = y \cdot x$
Θεώρημα 4, απορρόφηση	(α) $x + xy = x$	(β) $x\,(x + y) = x$
Αξίωμα 4, επιμεριστική	(α) $x\,(y + z) = xy + xz$	(β) $x + yz = (x + y)\,(x + z)$
Θεώρημα 5, προσεταιριστική	(α) $x + (y + z) = (x + y) + z$	(β) $x\,(yz) = (xy)\,z$
Θεώρημα 6, De Morgan	(α) $\overline{(x + y)} = \bar{x}\,\bar{y}$	(β) $\overline{(xy)} = \bar{x} + \bar{y}$

-

Λογικές συναρτήσεις

Μία λογική συνάρτηση ή συνάρτηση Boole είναι μία έκφραση που σχηματίζεται από δυαδικές μεταβλητές, τους δύο δυαδικούς τελεστές OR και AND, τον τελεστή NOT, παρενθέσεις και ένα "ίσον". Για μία δεδομένη τιμή των μεταβλητών, η συνάρτηση μπορεί να είναι είτε 0 είτε 1.

Οι λογικές συναρτήσεις εκφράζονται συνήθως είτε ως άθροισμα γινομένων είτε ως γινόμενο αθροισμάτων

F = AB + BC + AD (άθροισμα γινομένων)

F = (A+B) (C+D) (γινόμενο αθροισμάτων)

Οι αλγεβρικές εκφράσεις Boole δεν είναι μοναδικές. Άρα, μπορεί να υπάρχουν δύο διαφορετικές αλγεβρικές εκφράσεις που να περιγράφουν την ίδια συνάρτηση.

Απλοποίηση λογικών συναρτήσεων

Παράδειγμα 2.1:

Δίνεται η λογική συνάρτηση 4 λογικών μεταβλητών A, B, C, και D

$$F = \overline{A}B\overline{C}D + \overline{A}BCD + AB\overline{C}D + ABCD$$

Να απλοποιηθεί.

Λύση:

$$\begin{aligned} F &= \overline{A}BD\left(C + \overline{C}\right) + ABD\left(\overline{C} + C\right) \\ &= \overline{A}BD \cdot 1 + ABD \cdot 1 \\ &= \overline{A}BD + ABD \\ &= BD\left(\overline{A} + A\right) = BD \cdot 1 = BD \end{aligned}$$

Παράδειγμα 2.2:

Δίνεται η λογική συνάρτηση 3 λογικών μεταβλητών A, B, και C

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

Να απλοποιηθεί.

Λύση:

$$\begin{aligned}
F &= \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC \\
&= \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC + ABC && \text{[Θεώρημα 1(α)]} \\
&= \left(\overline{A} + A\right)BC + A\overline{B}C + AB\overline{C} + ABC && \text{[Αξίωμα 5(α)]} \\
&= BC + A\overline{B}C + AB\overline{C} + ABC \\
&= BC + A\overline{B}C + AB\overline{C} + ABC + ABC && \text{[Θεώρημα 1(α)]} \\
&= BC + AC\left(\overline{B} + B\right) + AB\left(C + \overline{C}\right) \\
&= BC + AC + AB
\end{aligned}$$

Μία λογική συνάρτηση μπορεί επίσης να παριστάνεται με τη βοήθεια του πίνακα αληθείας της. Έστω λ.χ. για παράδειγμα η συνάρτηση

$$F = \overline{A}\overline{B}C + \overline{A}BC + A\overline{B}$$

Οι δυνατοί συνδυασμοί τριών μεταβλητών είναι 8. Ο πίνακας αληθείας λοιπόν της συνάρτησης F θα είναι

x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Παράδειγμα 2.3:

Να απλοποιηθούν οι πιο κάτω λογικές συναρτήσεις

$$F_1 = \overline{A}\,\overline{B} + A\overline{B} + \overline{A}B \quad , \quad F_2 = A\overline{B} + ABC\overline{C}... $$

Λύση:

$$F_1 = \overline{A}\,\overline{B} + A\overline{B} + \overline{A}B = \overline{B}\left(A + \overline{A}\right) + \overline{A}B = \overline{B}\cdot 1 + \overline{A}B = \overline{B} + \overline{A}B\,.$$

$$F_2 = A\overline{B} + AB\overline{C} + ABCD + ABC\overline{D} = A\overline{B} + AB\overline{C} + ABC\left(D + \overline{D}\right) =$$
$$= A\left(\overline{B} + B\overline{C} + BC\right) = A\left[\overline{B} + B\left(\overline{C} + C\right)\right] = A\left(B + \overline{B} = A\right).$$

Παράδειγμα 2.4:

Με χρήση του θεωρήματος De Morgan να απλοποιηθεί η συνάρτηση

$$F = \left(\overline{AC + \overline{A}BC + \overline{B}C}\right) + AB\overline{C}$$

Λύση :

$$F = \left(\overline{AC + \overline{A}BC + \overline{B}C}\right) + AB\overline{C} = \overline{AC}\cdot\overline{\overline{A}BC}\cdot\overline{\overline{B}C} + AB\overline{C} = \left(\overline{A} + \overline{C}\right)\left(\overline{\overline{A}} + \overline{B} + \overline{C}\right)\left(\overline{\overline{B}} + \overline{C}\right) + AB\overline{C} = \left[\overline{A}\left(A + \overline{B} + \overline{C}\right) + \overline{C}\left(A + \overline{B} + \overline{C}\right)\right]\cdot\left(B + \overline{C}\right) + AB\overline{C}$$
$$= \left(A\overline{A} + \overline{A}\,\overline{B} + \overline{A}\,\overline{C} + A\overline{C} + \overline{B}\,\overline{C} + \overline{C}\,\overline{C}\right)\left(B + \overline{C}\right) + AB\overline{C}$$
$$= \left(\overline{A}\,\overline{B} + \overline{C}\right)\left(B + \overline{C}\right) + AB\overline{C} = \overline{A}\,\overline{B}B + B\overline{C} + \overline{A}\,\overline{B}\overline{C} + C\overline{C} + AB\overline{C}$$
$$= \overline{C}\left(B + \overline{A}\,\overline{B} + 1 + AB\right) = \overline{C}\,.$$

Λογικές συναρτήσεις ως άθροισμα γινομένων

Θα δούμε στη συνέχεια, πώς μπορούμε να βρούμε τον πίνακα αληθείας μιας συνάρτησης αν γνωρίζουμε τη μορφή της.

Έστω η συνάρτηση

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

Παρατηρούμε τα εξής:

α) Αν κάποιος όρος της συνάρτησης γίνει 1, όλοι οι υπόλοιποι όροι θα είναι μηδενικοί (διότι διαφέρουν τουλάχιστον κατά την τιμή μιας μεταβλητής) και τελικά η τιμή της συνάρτησης θα είναι 1.
β) Αν όλοι οι όροι είναι μηδενικοί τότε η συνάρτηση θα πάρει τιμή 0.

A	B	C	Γινόμενα
0	0	0	$\overline{A}\,\overline{B}\,\overline{C}$
0	0	1	$\overline{A}\,\overline{B}C$
0	1	0	$\overline{A}B\overline{C}$
0	1	1	$\overline{A}BC$
1	0	0	$A\overline{B}\,\overline{C}$
1	0	1	$A\overline{B}C$
1	1	0	$AB\overline{C}$
1	1	1	ABC

Αυτές οι δύο παρατηρήσεις μας οδηγούν στην κατάστρωση του πιο πάνω πίνακα. Φτιάχνουμε κατ' αρχήν όλους τους δυνατούς συνδυασμούς των τιμών που μπορούν να πάρουν οι μεταβλητές A, B, C (είναι $2^3 = 8$. Για *n* μεταβλητές θα έχουμε 2^n δυνατούς συνδυασμούς). Σε κάθε συνδυασμό αντιστοιχεί ένα γινόμενο. Σε κάθε γινόμενο οι μεταβλητές εμφανίζονται κανονικά αν οι τιμές τους είναι 1, ή συμπληρωμένες (δηλ. με αντιστροφή) αν οι τιμές τους στο συνδυασμό είναι 0. Τα γινόμενα αυτά ονομάζονται ***βασικά γινόμενα*** **(fundamental products) ή *ελαχιστόροι* (minterms).**

-

Αν θέλουμε να βρούμε τον πίνακα αληθείας μιας γνωστής συνάρτησης π.χ. της $F = \overline{A}BC + \overline{A}\,\overline{B}C + ABC$, τότε θέτουμε 1 για κάθε συνδυασμό των A, B, C που το αντίστοιχο βασικό γινόμενο υπάρχει σαν όρος στη δοθείσα συνάρτηση και 0 στις υπόλοιπες περιπτώσεις. Τα αποτελέσματα αυτής της διαδικασίας φαίνονται στον πιο κάτω πίνακα.

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Πολλές φορές μπορεί σε μια λογική συνάρτηση F να λείπουν από ένα ή και περισσότερους όρους μία ή και περισσότερες μεταβλητές. Π.χ.

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

Σ' αυτήν την περίπτωση με χρήση των θεωρημάτων της άλγεβρας Boole καταλήγουμε σε μορφή όπου θα υπάρχουν όλες οι μεταβλητές. Π.χ.

$$\begin{aligned}
F &= A \cdot 1 + BC \cdot 1 + \overline{A}\,\overline{B}C \\
&= A\left(B + \overline{B}\right) + BC\left(A + \overline{A}\right) + \overline{A}\,\overline{B}C \\
&= AB + A\overline{B} + BCA + BC\overline{A} + \overline{A}\,\overline{B}C \\
&= AB \cdot 1 + A\overline{B} \cdot 1 + ABC + \overline{A}BC + \overline{A}\,\overline{B}C \\
&= AB\left(C + \overline{C}\right) + A\overline{B}\left(C + \overline{C}\right) + ABC + \overline{A}BC + \overline{A}\,\overline{B}C \\
&= ABC + AB\overline{C} + A\overline{B}C + A\overline{B}\,\overline{C} + ABC + \overline{A}BC + \overline{A}\,\overline{B}C \\
&= ABC + AB\overline{C} + A\overline{B}C + A\overline{B}\,\overline{C} + \overline{A}BC + \overline{A}\,\overline{B}C
\end{aligned}$$

οπότε ο πίνακας αληθείας κατασκευάζεται κατά τα γνωστά.

A	B	C	F	Γινόμενα
0	0	0	0	$\overline{A}\,\overline{B}\,\overline{C}$
0	0	1	1	$\overline{A}\,\overline{B}C$
0	1	0	1	$\overline{A}B\overline{C}$
0	1	1	0	$\overline{A}BC$
1	0	0	0	$A\overline{B}\,\overline{C}$
1	0	1	0	$A\overline{B}C$
1	1	0	1	$AB\overline{C}$
1	1	1	1	ABC

Η λογική συνάρτηση είναι: $F = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + ABC$.

-

Λογικές συναρτήσεις ως γινόμενο αθροισμάτων

Ας δούμε κατ' αρχήν πώς μπορούμε να βρούμε τον πίνακα αληθείας αν μας δίνεται η μορφή της συνάρτησης
Έστω π.χ. η συνάρτηση

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

Παρατηρούμε τα εξής:

α) Για να είναι η F μηδενική θα πρέπει κάποιο από τα αθροίσματα να είναι μηδέν (ένα ή και περισσότερα).
β) Αν κάποιο άθροισμα είναι μηδέν τότε όλοι οι προσθετέοι είναι μηδενικοί, πράγμα που αντιστοιχεί σε ένα συνδυασμό τιμών των A, B, C στον πίνακα αληθείας.

Με βάση τα παραπάνω καταστρώνουμε τον παρακάτω πίνακα. Στα *βασικά αθροίσματα* (fundamental sums) ή *μεγιστόρους* (maxterms) οι μεταβλητές βρίσκονται στην κανονική μορφή τους αν οι τιμές στους αντίστοιχους συνδυασμούς είναι 0. Ο πίνακας αληθείας της F έχει μηδενικές τιμές αν κάποιο βασικό άθροισμα εμφανίζεται στη συνάρτηση.

A	B	C	Αθροίσματα	F
0	0	0	$A + B + C \longrightarrow$	0
0	0	1	$A + B + \overline{C}$	1
0	1	0	$A + \overline{B} + C \longrightarrow$	0
0	1	1	$A + \overline{B} + \overline{C}$	1
1	0	0	$\overline{A} + B + C \longrightarrow$	0
1	0	1	$\overline{A} + B + \overline{C}$	1
1	1	0	$\overline{A} + \overline{B} + C$	1
1	1	1	$\overline{A} + \overline{B} + \overline{C}$	1

Στο αντίστροφο πρόβλημα, δηλαδή στην εύρεση της συνάρτησης αν είναι δεδομένος ο πίνακας αληθείας, σχηματίζουμε τη συνάρτηση από τα βασικά αθροίσματα που αντιστοιχούν σε 0 τιμές της F. Π.χ.

A	B	C	F	Αθροίσματα
0	0	0	1	$A+B+C$
0	0	1	0 →	$A+B+\overline{C}$
0	1	0	1	$A+\overline{B}+C$
0	1	1	1	$A+\overline{B}+\overline{C}$
1	0	0	1	$\overline{A}+B+C$
1	0	1	0 →	$\overline{A}+B+\overline{C}$
1	1	0	0 →	$\overline{A}+\overline{B}+C$
1	1	1	1	$\overline{A}+\overline{B}+\overline{C}$

Η λογική συνάρτηση είναι: $F=\left(A+B+\overline{C}\right)\left(\overline{A}+B+\overline{C}\right)\left(\overline{A}+\overline{B}+C\right)$.

Πολλές φορές μπορεί σε μια λογική συνάρτηση F να λείπουν από ένα ή και περισσότερους όρους μία ή και περισσότερες μεταβλητές. Π.χ.

$$F = AB + \overline{A}C$$

Σ' αυτήν την περίπτωση με χρήση των θεωρημάτων της άλγεβρας Boole καταλήγουμε σε μορφή όπου θα υπάρχουν όλες οι μεταβλητές. Π.χ.

$$\begin{aligned}F &= AB+\overline{A}C\\ &= \left(AB+\overline{A}\right)\left(AB+C\right) && \text{από το αξίωμα : 4(β)}\\ &= \left(A+\overline{A}\right)\left(B+\overline{A}\right)\left(A+C\right)\left(B+C\right)\\ &= \left(\overline{A}+B\right)\left(A+C\right)\left(B+C\right)\end{aligned}$$

Στη συνέχεια, εισάγουμε σε κάθε όρο την τρίτη μεταβλητή

$$\begin{aligned}\overline{A}+B &= \overline{A}+B+C\overline{C} = \left(\overline{A}+B+C\right)\left(\overline{A}+B+\overline{C}\right)\\ A+C &= A+C+B\overline{B} = \left(A+B+C\right)\left(A+\overline{B}+C\right)\\ B+C &= B+C+A\overline{A} = \left(A+B+C\right)\left(\overline{A}+B+C\right)\end{aligned}$$

και τελικά

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

οπότε ο πίνακας αληθείας κατασκευάζεται κατά τα γνωστά.

-