# ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ

ΜΗΧΑΝΙΚΟΙ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

## ΔΥΑΔΙΚΟΙ ΑΡΙΘΜΟΙ

(ΑΚΕΡΑΙΟΙ ΑΡΙΘΜΟΙ)

Γ. Τσιατούχας

Παράρτημα Α

## Διάρθρωση

1. Βάσεις αριθμητικών συστημάτων
2. Μετατροπές μεταξύ βάσεων
3. Αρνητικοί δυαδικοί αριθμοί
4. Αριθμητικές πράξεις δυαδικών αριθμών

# Βάσεις Αριθμητικών Συστημάτων Ι

| | | Θέση Εκατοντάδων (100) | Θέση Δεκάδων (10) | Θέση Μονάδων (1) | | Θέση Δεκάτων (.1) | Θέση Εκατοστών (.01) | Θέση Χιλιοστών (.001) | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $d_n$ | ... | $d_2$ | $d_1$ | $d_0$ | . | $d_{-1}$ | $d_{-2}$ | $d_{-3}$ | ... | $d_{-k}$ |

$$\text{Αριθμός} = \sum_{i=-k}^{n} d_i \times 10^i$$

Γενική μορφή δεκαδικού αριθμού

$d_i \in [0, \ldots, 9]$

Βάση

$$\text{Αριθμός} = \sum_{i=-k}^{n} d_i \times 2^i$$

Δυαδικό σύστημα

$d_i \in [0, 1]$ bit

Οκταδικοί αριθμοί

$d_i \in [0, 1, 2, 3, 4, 5, 6, 7]$

Δεκαεξαδικοί αριθμοί

$d_i \in [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F]$



# Βάσεις Αριθμητικών Συστημάτων ΙΙ

Δυαδικό

| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|
| $1\times2^{10}$ | $+1\times2^9$ | $+1\times2^8$ | $+1\times2^7$ | $+1\times2^6$ | $+0\times2^5$ | $+1\times2^4$ | $+0\times2^3$ | $+0\times2^2$ | $+0\times2^1$ | $+1\times2^0$ |
| 1024 | + 512 | + 256 | + 128 | + 64 | + 0 | +16 | + 0 | + 0 | + 0 | + 1 |

Οκταδικό

| 3 | 7 | 2 | 1 |
|---|---|---|---|
| $3\times8^3$ | $+7\times8^2$ | $+2\times8^1$ | $+1\times8^0$ |
| 1536 | + 448 | + 16 | + 1 |

Δεκαδικό

| 2 | 0 | 0 | 1 |
|---|---|---|---|
| $2\times10^3$ | $+0\times10^2$ | $+0\times10^1$ | $+1\times10^0$ |
| 2000 | + 0 | + 0 | + 1 |

Δεκαεξαδικό

| 7 | D | 1 |
|---|---|---|
| $7\times16^2$ | $+13\times16^1$ | $+1\times16^0$ |
| 1792 | + 208 | + 1 |

Ο αριθμός 2001 στο δυαδικό, οκταδικό, δεκαδικό και δεκαεξαδικό.

# Αντιστοίχηση Αριθμών σε Διαφορετικές Βάσεις Ι

Δεκαδικοί αριθμοί και οι ισοδύναμοί τους δυαδικοί, οκταδικοί και δεκαεξαδικοί αριθμοί.

| Δεκαδικός | Δυαδικός | Οκταδ. | Δεκαεξ |
|---|---|---|---|
| 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 |



# Αντιστοίχηση Αριθμών σε Διαφορετικές Βάσεις ΙΙ

… συνέχεια.

| Δεκαδικός | Δυαδικός | Οκταδ. | Δεκαεξ |
|---|---|---|---|
| 15 | 1111 | 17 | F |
| 16 | 10000 | 20 | 10 |
| 20 | 10100 | 24 | 14 |
| 30 | 11110 | 36 | 1E |
| 40 | 101000 | 50 | 28 |
| 50 | 110010 | 62 | 32 |
| 60 | 111100 | 74 | 3C |
| 70 | 1000110 | 106 | 46 |
| 80 | 1010000 | 120 | 50 |
| 90 | 1011010 | 132 | 5A |
| 100 | 11001000 | 144 | 64 |
| 1000 | 1111101000 | 1750 | 3E8 |
| 2989 | 101110101101 | 5655 | BAD |

# Μετατροπή Μεταξύ Αριθμητικών Βάσεων Ι

**Παράδειγμα 1**

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Δεκαεξαδικό | 1 | 9 | 4 | 8 | . | B | 6 | | | |
| Δυαδικό | 0001 | 1001 | 0100 | 1000 | . | 1011 | 0110 | 0 | | |
| Οκταδικό | 1 | 4 | 5 | 1 | 0 | . | 5 | 5 | 4 | |

Δυαδικό: 0001100101001000.101101100

**Παράδειγμα 2**

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Δεκαεξαδικό | 7 | B | A | 3 | . | B | C | 4 | | |
| Δυαδικό | 0111 | 1011 | 1010 | 0011 | . | 1011 | 1100 | 0100 | | |
| Οκταδικό | 7 | 5 | 6 | 4 | 3 | . | 5 | 7 | 0 | 4 |

Δυαδικό: 0111101110100011.101111000100

Παραδείγματα μετατροπής από οκταδικό σε δυαδικό και από δεκαεξαδικό σε δυαδικό.



# Μετατροπή Δεκαδικού σε Δυαδικό

Μετατροπή του δεκαδικού αριθμού 1492 σε δυαδικό με διαδοχικές διαιρέσεις με το 2.

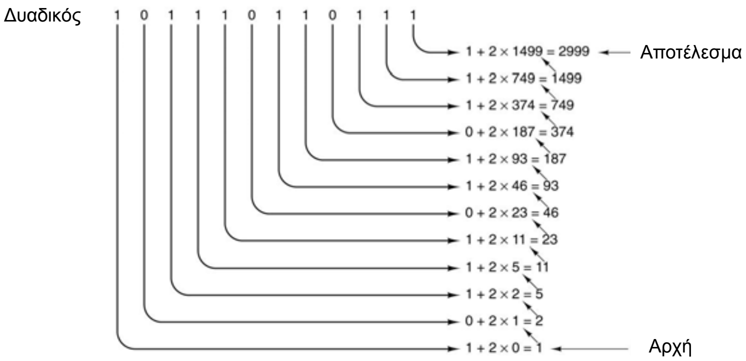
Η διαδικασία γίνεται κατακόρυφα ξεκινώντας από την κορυφή.

Π.χ. το 93 διαιρούμενο με το 2 δίνει πηλίκο 46 και υπόλοιπο 1, τα οποία γράφονται στην επόμενη γραμμή.

| Πηλίκα | Υπόλοιπα |
|---|---|
| 1492 | |
| 746 | 0 |
| 373 | 0 |
| 186 | 1 |
| 93 | 0 |
| 46 | 1 |
| 23 | 0 |
| 11 | 1 |
| 5 | 1 |
| 2 | 1 |
| 1 | 0 |
| 0 | 1 |

1 0 1 1 1 0 1 0 1 0 0 $= 1492_{10}$

# Μετατροπή Δυαδικού σε Δεκαδικό




Μετατροπή του δυαδικού αριθμού 101110110111 σε δεκαδικό με διαδοχικούς διπλασιασμούς, αρχίζοντας από χαμηλά.

Κάθε γραμμή σχηματίζεται διπλασιάζοντας την γραμμή που βρίσκεται από κάτω και προσθέτοντας το αντίστοιχο bit του δυαδικού αριθμού.

Π.χ. το 749 είναι το διπλάσιο του 374 συν το bit 1 του δυαδικού αριθμού που βρίσκεται στην ίδια γραμμή με το 749.



# Αρνητικοί Δυαδικοί Αριθμοί

Με m bits μπορούμε να αναπαραστήσουμε τους ακέραιους αριθμούς από 0 έως $2^m-1$ (δηλ. από $\underbrace{00\ldots0}_{\text{m-bits}}$ έως $\underbrace{11\ldots1}_{\text{m-bits}}$).

Πως θα μπορούσαμε όμως να αναπαραστήσουμε αρνητικούς αριθμούς;

Μια απλή λύση είναι η δέσμευση ενός bit (π.χ. το αριστερότερο bit) που θα αναπαριστά το πρόσημο (+ ή −) και συνεπώς θα παραμείνουν m−1 bits για την αναπαράσταση του μέτρου του αριθμού. Αυτό το σύστημα ονομάζεται αναπαράσταση *προσημασμένου μέτρου* (*signed magnitude*). Υπάρχουν όμως και άλλα συστήματα, όπως:

- Συμπλήρωμα ως προς 1 (one's complement)
- Συμπλήρωμα ως προς 2 (two's complement)
- Πλεόνασμα κατά $2^{m-1}$ (excess $2^{m-1}$)

# Προσημασμένο Μέτρο

- Το αριστερότερο (πιο σημαντικό) bit χρησιμοποιείται για το πρόσημο:
  (0 → + και 1 → −)
- Τα υπόλοιπα bit περιέχουν την απόλυτη τιμή του αριθμού (μέτρο).
- Συνεπώς με m bits συμβολίζουμε τους αριθμούς:
  $[-(2^{m-1}-1)$ έως $+(2^{m-1}-1)]_{10}$ δηλαδή $[\,\underbrace{\underline{1}11...1}_{m}$ έως $\underbrace{\underline{0}11...1}_{m}\,]_2$

Π.χ. η αναπαράσταση των αριθμών +37 και −37 με 8 bits θα είναι:

+37 = <u>0</u>0100101

−37 = <u>1</u>0100101

δηλ. ένα bit για το πρόσημο και επτά bits για το μέτρο.

- Προβλήματα:
  - Υπάρχουν δύο αναπαραστάσεις του μηδέν (000...0 & 100...0)
  - Υπάρχει δυσκολία στην υλοποίηση των πράξεων



# Συμπλήρωμα ως προς 1

- Το αριστερότερο (πιο σημαντικό) bit χρησιμοποιείται για το πρόσημο:
  (0 → + και 1 → −)
- Τα υπόλοιπα bit περιέχουν την απόλυτη τιμή του αριθμού (μέτρο).
- Για να βρούμε τον αρνητικό αριθμό αντιστρέφουμε όλα τα bits του αντίστοιχου θετικού αριθμού.
- Συνεπώς με m bits συμβολίζουμε τους αριθμούς:
  $[-(2^{m-1}-1)$ έως $+(2^{m-1}-1)]_{10}$ δηλαδή $[\,\underline{1}00...0$ έως $\underline{0}11...1\,]_2$

Π.χ. η αναπαράσταση των αριθμών +37 και −37 με 8 bits θα είναι:

+37 = <u>0</u>0100101

−37 = <u>1</u>1011010

δηλ. ένα bit για το πρόσημο και επτά bits για το μέτρο.

- Προβλήματα:
  - Υπάρχουν δύο αναπαραστάσεις του μηδέν (000...0 & 111...1)
  - Υπάρχει δυσκολία στην υλοποίηση των πράξεων

# Συμπλήρωμα ως προς 2

- Το αριστερότερο (πιο σημαντικό) bit χρησιμοποιείται για το πρόσημο:
  ($0 \rightarrow +$ και $1 \rightarrow -$)
- Τα υπόλοιπα bit περιέχουν την απόλυτη τιμή του αριθμού (μέτρο).
- Για να βρούμε τον αρνητικό αριθμό υπολογίζουμε το συμπλήρωμα ως προς 1 του θετικού αριθμού και προσθέτουμε στο αποτέλεσμα το 1. *Τυχόν τελικό κρατούμενο αγνοείται !*
- Συνεπώς με m bits συμβολίζουμε τους αριθμούς:
  $[-(2^{m-1}) \text{ έως } +(2^{m-1}-1)]_{10}$ δηλαδή $[\,\underline{1}00...0 \text{ έως } \underline{0}11...1\,]_2$

  Π.χ. η αναπαράσταση των αριθμών +37 και −37 με 8 bits θα είναι:

  $+37 = \underline{0}0100101$

  $-37 = \underline{1}1011010 + 1 = \underline{1}1011011$

  δηλ. ένα bit για το πρόσημο και επτά bits για το μέτρο.
- Παρατηρήσεις:
  - Μοναδική αναπαράσταση του μηδέν (000...0)
  - Ευκολότερη υλοποίηση των πράξεων
  - Υπάρχει ένας αρνητικός περισσότερος σε σχέση με τους θετικούς



# Πλεόνασμα κατά $2^{m-1}$

- Ο αριθμός αναπαρίσταται αθροίζοντάς τον με το $2^{m-1}$.
- Συνεπώς με m bits έχουμε πλεόνασμα κατά $2^{m-1}$ και συμβολίζουμε τους αριθμούς:
  $[-(2^{m-1}) \text{ έως } +(2^{m-1}-1)]_{10}$ δηλαδή $[\,\underline{0}00...0 \text{ έως } \underline{1}11...1\,]_2$

  Π.χ. η αναπαράσταση των αριθμών +37 και −37 με 8 bits θα είναι:

  $+37 = +37 + 128 = 165 = \underline{1}0100101$

  $-37 = -37 + 128 = 91 = \underline{0}1011011$

  δηλ. ένα bit για το πρόσημο και επτά bits για το μέτρο.

- Παρατηρήσεις:
  - Μοναδική αναπαράσταση του μηδέν (100...0)
  - Ευκολότερη υλοποίηση των πράξεων
  - Υπάρχει ένας αρνητικός περισσότερος σε σχέση με τους θετικούς
  - Ταυτόσημο με το συμπλήρωμα ως προς 2 αλλά με αντεστραμμένο πρόσημο

# Παραδείγματα Αναπαράστασης Αρνητικών Αριθμών I

| N decimal | N binary | –N signed mag. | –N 1's compl. | –N 2's compl. | –N excess 128 |
|---|---|---|---|---|---|
| 1 | 00000001 | 10000001 | 11111110 | 11111111 | 01111111 |
| 2 | 00000010 | 10000010 | 11111101 | 11111110 | 01111110 |
| 3 | 00000011 | 10000011 | 11111100 | 11111101 | 01111101 |
| 4 | 00000100 | 10000100 | 11111011 | 11111100 | 01111100 |
| 5 | 00000101 | 10000101 | 11111010 | 11111011 | 01111011 |
| 6 | 00000110 | 10000110 | 11111001 | 11111010 | 01111010 |
| 7 | 00000111 | 10000111 | 11111000 | 11111001 | 01111001 |
| 8 | 00001000 | 10001000 | 11110111 | 11111000 | 01111000 |
| 9 | 00001001 | 10001001 | 11110110 | 11110111 | 01110111 |
| 10 | 00001010 | 10001010 | 11110101 | 11110110 | 01110110 |

Αρνητικοί αριθμοί των 8-bit στα 4 συστήματα αναπαράστασης.



# Παραδείγματα Αναπαράστασης Αρνητικών Αριθμών II

| N decimal | N binary | –N signed mag. | –N 1's compl. | –N 2's compl. | –N excess 128 |
|---|---|---|---|---|---|
| 20 | 00010100 | 10010100 | 11101011 | 11101100 | 01101100 |
| 30 | 00011110 | 10011110 | 11100001 | 11100010 | 01100010 |
| 40 | 00101000 | 10101000 | 11010111 | 11011000 | 01011000 |
| 50 | 00110010 | 10110010 | 11001101 | 11001110 | 01001110 |
| 60 | 00111100 | 10111100 | 11000011 | 11000100 | 01000100 |
| 70 | 01000110 | 11000110 | 10111001 | 10111010 | 00111010 |
| 80 | 01010000 | 11010000 | 10101111 | 10110000 | 00110000 |
| 90 | 01011010 | 11011010 | 10100101 | 10100110 | 00100110 |
| 100 | 01100100 | 11100100 | 10011011 | 10011100 | 00011100 |
| 127 | 01111111 | 11111111 | 10000000 | 10000001 | 00000001 |
| 128 | Nonexistent | Nonexistent | Nonexistent | 10000000 | 00000000 |

# Αριθμητικές Πράξεις Δυαδικών

| | | | | |
|---|---|---|---|---|
| 1ος Προσθετέος | 0 | 0 | 1 | 1 |
| 2ος Προσθετέος | +0 | +1 | +0 | +1 |
| Άθροισμα | 0 | 1 | 1 | 0 |
| Κρατούμενο | 0 | 0 | 0 | 1 |

Άθροισμα bits

## Πρόσθεση με αναπαράσταση στα συμπληρώματα ως προς 1 και 2

| Δεκαδικό | Συμπλήρωμα ως προς 1 | Συμπλήρωμα ως προς 2 |
|---|---|---|
| 10 | 00001010 | 00001010 |
| + (–3) | 11111100 | 11111101 |
| +7 | 1 00000110 | 1 00000111 |
| | κρατούμενο 1 | αποβάλλεται |
| | 00000111 | |

Απλούστερη υλοποίηση !
Κάνουμε πρόσθεση όλων των bits κατά θέση
και αγνοούμε τυχόν τελικό κρατούμενο



# Πράξεις στο Συμπλήρωμα ως προς 2

| Δεκαδικό | Δυαδικό | Δεκαδικό | Δυαδικό |
|---|---|---|---|
| + 6 | 00000110 | − 6 | 11111010 |
| +13 | 00001101 | +13 | 00001101 |
| +19 | 00010011 | + 7 | 00000111 |
| + 6 | 00000110 | − 6 | 11111010 |
| −13 | 11110011 | −13 | 11110011 |
| − 7 | 11111001 | − 19 | ~~1~~ 11101101 |

αγνοείται !

Βλ. ενότητα 1-6, Ψηφιακή Σχεδίαση, Μ. Mano
Προσοχή ∃ λάθη !

Εύρεση αριθμού

Αντιστρέφουμε τον αριθμό:
11101101 → 00010010
και προσθέτουμε τη μονάδα:
00010010 + 1 = 00010011 ≡ $19_{10}$
Συνεπώς ο αρχικός αριθμός είναι ο $-19_{10}$

- Υπερχείλιση (overflow):

| | |
|---|---|
| − 113 | 10001111 |
| − 20 | 11101100 |
| − 133 | ~~1~~ 01111011 = + $61_{10}$ |

Η πεπερασμένη μνήμη των υπολογιστών μας αναγκάζει να χρησιμοποιούμε μόνο αριθμούς που μπορούν να αναπαρασταθούν με καθορισμένο αριθμό ψηφίων – *αριθμοί πεπερασμένης ακρίβειας* !

Ανιχνεύεται υπερχείλιση όταν οι προσθετέοι έχουν το ίδιο πρόσημο και το αποτέλεσμα έχει διαφορετικό.

# Αφαίρεση στο Συμπλήρωμα ως προς 2

Παίρνουμε το συμπλήρωμα ως προς 2 του αφαιρετέου (συμπεριλαμβανομένου και του bit προσήμου) και το προσθέτουμε στο μειωτέο (συμπεριλαμβανομένου και του bit προσήμου). Τυχόν κρατούμενο αγνοείται.

αγνοείται !

Π.χ. η αφαίρεση –6 – (−13) = –6 + (+13) = +7 θα είναι:

$$\underbrace{(11111010)}_{\text{μειωτέος}} - \underbrace{(11110011)}_{\text{αφαιρετέος}} = (11111010) + \underbrace{(00001101)}_{\text{συμπλήρωμα ως προς 2 αφαιρετέου}} = \cancel{1}(00000111)$$

Συμβολικά η αφαίρεση παρουσιάζεται ως ακολούθως:

$$x - y = x + (-y) = x + \overline{y} + 1$$

αντιστροφή

δηλ. συντελείται μία αντιστροφή και ολοκληρώνεται η πράξη αποκλειστικά με προσθέσεις !