ΤΛ2002 **Ψηφιακά Κυκλώματα Ι**

Μάθημα 1: Δυαδικά συστήματα - Κώδικες

Λευτέρης Καπετανάκης

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ

Άνοιξη 2011

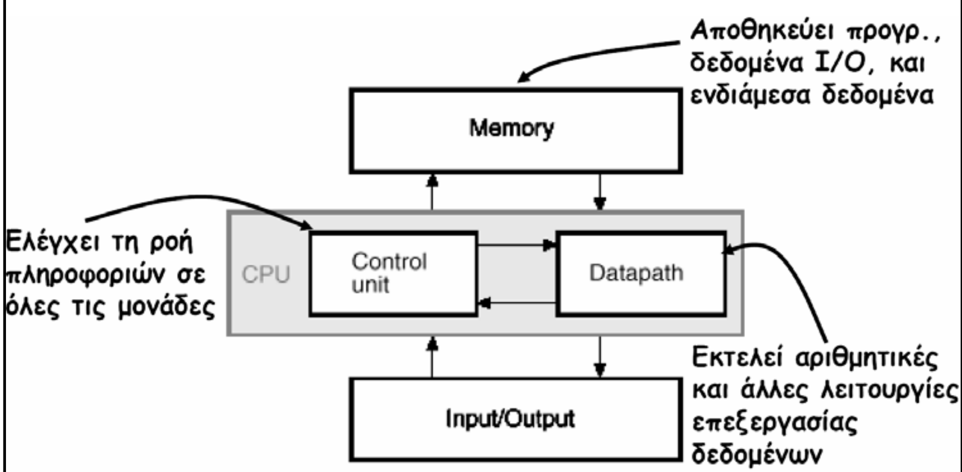
Ψηφιακά Συστήματα

- Κύρια χαρακτηριστικά:
  - επεξεργασία <u>διακριτών</u> στοιχείων πληροφορίας (οποιοδήποτε σύνολο που περιορίζεται σε ένα πεπερασμένο αριθμό στοιχείων)
  - π.χ. 10 δεκαδικά ψηφία, 26 γράμματα, ...
- Διακριτά στοιχεία (σε ψηφιακά συστήματα) μπορούν να αναπαρασταθούν από <u>σήματα</u> (φυσικές οντότητες)
  - Πιο κοινά σήματα: ηλεκτρικά (voltage, current)

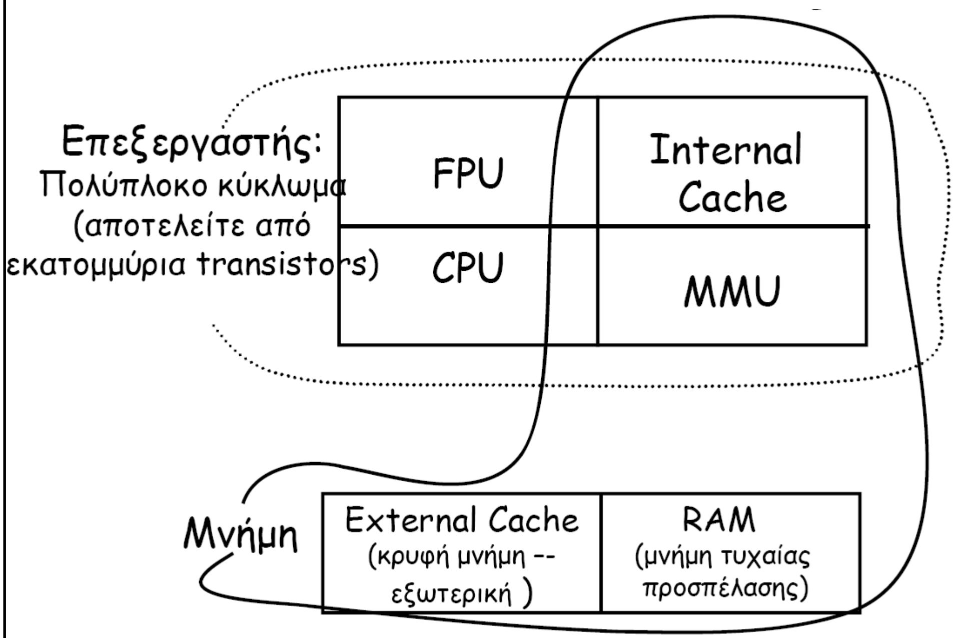
Αναπαράσταση Πληροφοριών

- Δυαδικά σήματα (2 διακριτές τιμές)
  - 0 και 1 (ή LOW και HIGH ή FALSE και TRUE)
  - Δυαδική μονάδα: δυαδικό ψηφίο/μπιτ (digit/bit)
  - Πληροφορία: σύνολο από bits (=words). Τυπικό μέγεθος: 8, 16, 32, 64, …
- Ψηφιακό Υλικό: υπολογίζει <u>δυαδικές συναρτήσεις</u> από <u>διάδικους αριθμούς</u>
  - Συνδυαστικά υλικό (χωρίς μνήμη)
  - Ακολουθιακά υλικό (με μνήμη)

Βασική Δομή Η/Υ



Μια πιο λεπτομερής όψη



Αριθμητικά Συστήματα

- Αναπαράσταση αριθμών
- Radix: η βάση
  - βασική μονάδα μιας ομάδας αριθμών, π.χ. για το δεκαδικό σύστημα το radix = 10 ("βάση" 10)
- Για κάθε σύστημα χρειαζόμαστε αριθμητικές λειτουργίες (πρόσθεση, αφαίρεση, πολλαπλασιασμός)
- Επίσης, μετατροπή από μια βάση σε άλλη

Αριθμητικά Συστήματα (συν.)

- Μη-θεσιακά αριθμιτικά συστήματα
  - Αρχαίο Αιγυπτιακό αριθμητικό σύστημα με βάση-10.
    - 1A1 = 11A = A11
  - Αρχαίο Ελληνικό αριθμιτικό σύστημα.
    - Χρησιμοποιήστε 27 γράμματα για να αντιπροσωπεύσει 1, 2, 3, …, 9, 10, 20, 30, . . ., 90, 100, 200, 300, …, 900

- Θεσιακά αριθμιτικά συστήματα

- Δεκαδικό αριθμητικό σύστημα.
  - $376 = 3x10^2+7x10^1+6x10^0$

βάρος

- Δυαδικό αριθμητικό σύστημα.
  - $(101100101)_2 = 1x2^8+0x2^7+1x2^6+1x2^5+0x2^4+0x2^3+1x2^2+0x2^1+1x2^0 = 256+64+32+4+1=(357)_{10}$

βάρος

Θεσιακά Αριθμητικά Συστήματα -Δεκαδικό

- "βάση" 10 (το radix είναι 10)
- 10 ψηφία: 0...9
- $(251.3)_{10} = \mathbf{2}*10^2 + \mathbf{5}*10^1 + \mathbf{1}*10^0 + \mathbf{3}*10^{-1}$
- Σημείωση: '.' ονομάζεται η υποδιαστολή για το σύστημα *radix* (υποδιαστολή για τη βάση 10)

Θεσιακά Αριθμητικά Συστήματα–Δεκαδικό (συν.)

- Γενικά, ένας δεκαδικός αριθμός με *n* ψηφία αριστερά (πριν) από την υποδιαστολή, και *m* ψηφία στα δεξιά (μετά) γράφεται ως ακολούθως:

$$A_{n-1}\ A_{n-2}\ \ldots\ A_1\ A_0\ .\ A_{-1}\ A_{-2}\ \ldots\ A_{-m+1}\ A_{-m}$$

το A_i λέγεται <u>συντελεστής</u> (coefficient) και παίρνει τιμές μεταξύ 0...9, ενώ το **i** δείχνει το <u>βάρος</u> (την τάξη) ($=10^i$) του A_i.

Θεσιακά Αριθμητικά Συστήματα–Δεκαδικό (συν.)

Η τιμή του

$$A_{n-1}\ A_{n-2}\ \ldots\ A_1\ A_0\ .\ A_{-1}\ A_{-2}\ \ldots\ A_{-m+1}\ A_{-m}$$

υπολογίζεται από

$$\sum_{i=n-1..0} (A_i * 10^i) + \sum_{i=-1..-m} (A_i * 10^i)$$

Θεσιακά Αριθμητικά Συστήματα – Γενικά

- "βάση" *r* (radix *r*)
- *r* ψηφία
- $N_r = A_{n-1} * r^{n-1} + A_{n-2} * r^{n-2} + \ldots + A_1 * r^1 + A_0 + A_{-1} * r^{-1} + A_{-2} * r^{-2} + \ldots + A_{-m} * r^{-m}$

Περισσότερο Σημαντικό Ψηφίο (Most Significant Bit -MSB)

Λιγότερο Σημαντικό Ψηφίο (Least Significant Bit - LSB)

ΤΛ-2002: L1 Slide 11

Θεσιακά Αριθμητικά Συστήματα – Γενικά (συν.)

- π.χ. $r = 6$
  $(312.4)_6 = 3*6^2 + 1*6^1 + 2*6^0 + 4*6^{-1}$
  $= (116.66)_{10}$
- Μετατροπή από *n*-δικό (οποιοδήποτε σύστημα με radix *n*) σε δεκαδικό ακολουθεί παρόμοια διαδικασία όπως την πιο πάνω

ΤΛ-2002: L1 Slide 12

Θεσιακά Αριθμητικά Συστήματα (συν.)

- Τα πιο κοινά αριθμητικά συστήματα για Η/Υ:
  - Δυαδικό ($r = 2$) (Binary)
  - Οκταδικό ($r = 8$) (Octal)
  - Δεκαεξαδικό ($r = 16$) (Hexadecimal)

Δυαδικοί αριθμοί - βάση 2

- Οι Η/Υ αναπαριστούν όλα τα δεδομένα σαν "*συμβολοσειρές bits*", κάθε bit είναι 0 ή 1
- "βάση" 2, με 2 ψηφία: 0 και 1
- π.χ.

$$(101101.10)_2 = \mathbf{1}*2^5 + 0*2^4 + \mathbf{1}*2^3 + \mathbf{1}*2^2 + 0*2^1 + \mathbf{1}*2^0 + \mathbf{1}*2^{-1} + \mathbf{0}*2^{-2}$$

$$\text{(σε δεκαδικό)} = 32 + 0 + 8 + 4 + 0 + 1 + \tfrac{1}{2} + 0$$

$$= (45.5)_{10}$$

Δυαδικοί αριθμοί - βάση 2 (συν.)

- π.χ.

  $(1001.011)_2 = \mathbf{1}*2^3 + \mathbf{0}*2^2 + \mathbf{0}*2^1 + \mathbf{1}*2^0 + \mathbf{0}*2^{-1} + \mathbf{1}*2^{-2} + \mathbf{1}*2^{-3}$

  (σε δεκαδικό) $= 8 + 1 + 0.25 + 0.125$

  $= (9.375)_{10}$

Δυνάμεις του δύο

n	2^n	n	2^n	n	2^n
0	1	8	256	16	65,536
1	2	9	512	17	131,072
2	4	10	1,024	18	262,144
3	8	11	2,048	19	524,288
4	16	12	4,096	20	1,048,576
5	32	13	8,192	21	2,097,152
6	64	14	16,384	22	4,194,304
7	128	15	32,768	23	8,388,608

- **Απομνημονεύστε** τουλάχιστον έως 2^{20}

Kilo, Mega, Giga, Tera, Peta, Exa, Zetta, Yotta

Name	Abbr	Factor	SI size
Kilo	K	2^{10} = 1,024	10^3 = 1,000
Mega	M	2^{20} = 1,048,576	10^6 = 1,000,000
Giga	G	2^{30} = 1,073,741,824	10^9 = 1,000,000,000
Tera	T	2^{40} = 1,099,511,627,776	10^{12} = 1,000,000,000,000
Peta	P	2^{50} = 1,125,899,906,842,624	10^{15} = 1,000,000,000,000,000
Exa	E	2^{60} = 1,152,921,504,606,846,976	10^{18} = 1,000,000,000,000,000,000
Zetta	Z	2^{70} = 1,180,591,620,717,411,303,424	10^{21} = 1,000,000,000,000,000,000,000
Yotta	Y	2^{80} = 1,208,925,819,614,629,174,706,176	10^{24} = 1,000,000,000,000,000,000,000,000

Οκταδικοί αριθμοί (Octal) - βάση 8

- "βάση" 8, με 8 ψηφία: 0..7
- π.χ.

$$(762)_8 = 7*8^2 + 6*8^1 + 2*8^0$$

$$(\text{σε δεκαδικό}) = 448 + 48 + 2$$

$$= (498)_{10}$$

Δεκαεξαδικοί αριθμοί (Hex) - βάση 16

- $r = 16$
- Ψηφία (σύμβαση): 0..9, A, B, C, D, E, F
- A=10, B=11, ... , F = 15
- π.χ.

$$(3FB)_{16} = 3*16^2 + 15*16^1 + \mathbf{11}*16^0$$

(σε δεκαδικό) $= 768 + 240 + 11$

$$= (1019)_{10}$$

Μετατροπή Βάσεων

- Οποιαδήποτε βάση $\boldsymbol{r} \rightarrow$ δεκαδικό
  Εύκολο!
  Υπολογισμός των δυναμοσειρών:
  $103_5 = 1x5^2 + 0x5^1 + 3x5^0 = 25 + 3 = 28_{10}$
  $2EA_{16} = 2 \text{ x } 16^2 + 14 \text{ x } 16^1 + 10x16^0 = 746_{10}$
- Δεκαδικό $\rightarrow$ Δυαδικό
- Οκταδικό $\leftrightarrow$ Δυαδικό
- Δεκαεξαδικό $\leftrightarrow$ Δυαδικό
- Δεκαδικό $\rightarrow$ Όποια βάση $\boldsymbol{r}$

Δεκαδικό σε βάση r

- Για τον ακέραιο μέρος του αριθμού,
  - Διαδοχικές διαιρέσεις με τη «βάση» r μέχρι το πηλίκο = 0.
  - Συλλέξτε τα υπόλοιπα με την αντίστροφη σειρά.
- Για το κλασματικό μέρος του αριθμού,
  - Διαδοχικούς πολλαπλασιασμούς με τη «βάση» r μέχρι το κλασματικό μέρος του αποτελέσματος = 0 .
  - Συλλέξτε τον ακέραιος αριθμό του αποτελέσματος με την αρχική σειρά.

ΤΛ-2002: L1 Slide 21

Μετατροπή Δεκαδικών σε οποιαδήποτε βάση r

- Ακέραιο Μέρος: Αναδρομικά, διαιρέστε το ακέραιο μέρος δια τη βάση, κρατώντας το υπόλοιπο μέχρι το ακέραιο μέρος να γίνει 0
- π.χ. $(153)_{10} = (\ ?\ )_8$, $r = 8$

153 / 8 = 19 + 1/8	υπόλοιπο = 1	LSB	↑
19 / 8 = 2 + 3/8	υπόλοιπο = 3		↑
2 / 8 = 0 + 2/8	υπόλοιπο = 2	MSB	↑
τέλος			

→ $(153)_{10} = (\ 231)_8$

ΤΛ-2002: L1 Slide 22

Μετατροπή Δεκαδικών σε οποιαδήποτε βάση r

- Κλασματικό Μέρος: Αναδρομικά, πολ/στε το κλασματικό μέρος επί τη βάση κρατώντας το ακέραιο μέρος μέχρι το κλασματικό μέρος να γίνει 0

- π.χ. $(0.78125)_{10} = (\ ?\ )_{16}$, $r = 16$

$0.78125 * 16 = 12.5$	ακέραιος = 12 = C	MSB	↓
$0.5 * 16 = 8.0$	ακέραιος = 8 = 8	LSB	
τέλος			

→ $(0.78125)_{10} = (0.C8)_{16}$

Δεκαδικό σε Δυαδικό

- $746_{10} = ?_2$

2	746	
2	373	0
2	186	1
2	93	0
2	46	1
2	23	0
2	11	1
2	5	1
2	2	1
2	1	0
	0	1

$746_{10} = 1011101010_2$

- $46.75_{10} = ?_2$

2	46	
2	23	0
2	11	1
2	5	1
2	2	1
2	1	0
	0	1

0.75	x2	
1 .5	x2	1
1 .0		1

$46.75_{10} = 101110.11_2$

Δεκαδικό σε Οκταδικό

- $746_{10} = ?_8$

8	746	
8	93	2
8	11	5
8	1	3
	0	1

$746_{10}=1352_8$

- $746.75_{10} = ?_8$

8	746	
8	93	2
8	11	5
8	1	3
	0	1

0.75 x8

6 .0 6

$746.75_{10}=1352.6_8$

Δεκαδικό σε Δεκαεξαδικό

- $746_{10} = ?_{16}$

16	746	
16	46	A
16	2	E
	0	2

$746_{10}=2EA_{16}$

- $746.75_{10} = ?_{16}$

16	746	
16	46	A
16	2	E
	0	2

0.75 x16

12 .0 C

$746_{10}=2EA.C_{16}$

Δεκαδικό σε βάση r

- $746_{10} = ?_5$

Διαιρέτης	Πηλίκο	Υπόλοιπο
5	746	
5	149	1
5	29	4
5	5	4
5	1	0
	0	1

$746_{10}=10441_5$

- $746.75_{10} = ?_5$

Διαιρέτης	Πηλίκο	Υπόλοιπο
5	746	
5	149	1
5	29	4
5	5	4
5	1	0
	0	1

	0.75	x5	
3	.75	x5	3
3	.75		3

$746.75_{10}=10441.333..._5$

Δυαδικό σε Οκταδικό και Δεκαεξαδικό

- Οκταδικό:
  $8 = 2^3$
  → κάθε 3 bits μεταφράζονται σε 1 οκταδικό

- Δεκαεξαδικό:
  $16 = 2^4$
  → κάθε 4 bits μεταφράζονται σε 1 δεκαεξαδικό

Δυαδικό ↔ Οκταδικό

$$(011|010|101|000|.|111|101|011|100)_2$$

$$\updownarrow$$

$$(3\ \ 2\ \ 5\ \ 0\ \ .\ \ 7\ \ 5\ \ 3\ \ 4)_8$$

Οκταδικό ↔ Δεκαεξαδικό

$$(0110|1010|1000|.|1111|0101|1100)_2$$

$$\updownarrow$$

$$(6\ \ A\ \ 8\ \ .\ \ F\ \ 5\ \ C)_{16}$$

Οκταδικό ↔ Δεκαεξαδικό

Μέσο δυαδικού!

Δεκαεξαδικό → Δυαδικό → Οκταδικό
Οκταδικό → Δυαδικό → Δεκαεξαδικό

Δυαδικές Αριθμητικές Πράξεις: Πρόσθεση

- Ακολουθεί τους ίδιους κανόνες με τη δεκαδική πρόσθεση, με την διαφορά ότι όταν το άθροισμα είναι 2 (και όχι 10) έχουμε κρατούμενο
- Νέοι κανόνες κρατουμένου (carry)
  - 0+0 = 0c0 (άθροισμα 0 με carry 0)
  - 0+1 = 1+0 = 1c0
  - 1+1 = 0c1
  - 1+1+1 = 1c1

Κρατούμενο	1	1	1	1	1	0	
Προσθετέος 1	0	0	1	0	0	1	
Προσθετέος 2	0	1	1	1	1	1	+
Αποτέλεσμα	1	0	1	0	0	0	

Δυαδικές Αριθμητικές Πράξεις: Πρόσθεση (συν.)

- "Ημιάθροισμα" (δεξιότερο bit, π.χ LSB): μόνο 2 bits προσθέτονται, με αποτέλεσμα ένα ψηφίο αθροίσματος και ένα κρατουμένου
- "Πλήρες Άθροισμα" (υπόλοιπες θέσεις): 3 bits προσθέτονται με αποτέλεσμα ένα άθροισμα (3 ψηφίων) και ένα κρατούμενο

Υπερχείλιση

- Εάν το μέγεθος της λέξης (word) είναι *n* bits και το αποτέλεσμα του αθροίσματος είναι *(n+1)* bits, έχουμε υπερχείλιση (overflow)
  → το αποτέλεσμα δεν μπορεί να αναπαρασταθεί ορθά (πλήρως) με *n* bits
- Υπερχείλιση δεν συμβαίνει ποτέ στην αφαίρεση. Γιατί;

Δυαδικές Αριθμητικές Πράξεις: Αφαίρεση

- Νέοι κανόνες δανεικού (borrow)
  - 0-0 = 1-1 = 0b0 (αποτέλεσμα 0 με δανεικό 0)
  - 1-0 = 1b0
  - 0-1 = 1b1
  - …

Δανεικό	1	1	0	0		
Αφαιρετέος	1	1	0	1	1	
Αφαιρέτης	0	1	1	0	1	-
Αποτέλεσμα	0	1	1	1	0	

Κλειδιά για επιτυχία

- Χρήση των ίδιων αλγορίθμων που χρησιμοποιούνται για την εκτέλεση δεκαδικών αριθμητικών λειτουργιών
- Γενίκευση για τη καινούργια βάση (οι κανόνες carry, borrow αλλάζουν)
- Διατήρηση της βάσης! Στο δυαδικό, 1+1=10

Δυαδικές Αριθμητικές Πράξεις: Πολ/σμός

- Αλγόριθμος Ολίσθησης-και-πρόσθεσης (Shift-and-add), όπως για τη βάση 10

Πολλ/στής	0	0	0	1	1	0	1
Πολλ/στέος	0	0	0	0	1	1	0
(1)	0	0	0	0	0	0	0
(2)	0	0	1	1	0	1	0
(3)	0	1	1	0	1	0	0
Άθροισμα	1	0	0	1	1	1	0

- Επαλήθευση: 13 * 6 = 78

Κώδικες

- Αναπαράσταση ενός συνόλου από στοιχεία (π.χ. αριθμούς) αντιστοιχώντας ένα κώδικα (codeword) για κάθε στοιχείο του συνόλου.
- Ο κώδικας είναι μια συμβολοσειρά

  Δυαδικός κώδικας με *n* bits: μια ομάδα από *n* bits που κωδικοποιούν 2^n διακριτά στοιχεία

  π.χ. Ένα σύνολο από 4 διακριτούς αριθμούς μπορεί να αναπαρασταθεί με κώδικα 2-bit έτσι ώστε κάθε αριθμός του συνόλου να αντιστοιχεί ακριβώς σε ένα συνδυασμό στο σύνολο {00,01,10,11}.

- Επομένως, για να κωδικοποιηθεί ένα σύνολο από Μ πληροφορίες απαιτούνται μ bits, όπου μ είναι ο μικρότερος ακέραιος αριθμός για τον οποίο: $M \leq 2^{\mu}$.

Μέθοδοι κωδικοποίησης

- Η κωδικοποίηση είναι μία αυθαίρετη αντιστοίχιση συμβόλων, αλλά ανάλογα με την εφαρμογή που ένας κώδικας προορίζεται να εξυπηρετήσει ορισμένες μορφές κωδίκων είναι καλύτερες από άλλες.
- Οι γενικές κατηγορίες κωδικοποίησης είναι: με **πίνακες αντιστοιχίας** και με **Κανόνες**.
- **Κανόνες** υπάρχουν πολλών ειδών, κυριότεροι είναι **με βάρη**, **χωρίς βάρη** (Gray) και **μικτοί** (BCD).

Δεκαδικοί με Δυαδική Κωδικοποίηση (Binary Coded Decimals - BCD)

- Ένας δεκαδικός κώδικας: Δεκαδικοί αριθμοί (0..9) κωδικοποιούνται χρησιμοποιώντας διακριτές δυαδικές λέξεις 4ων bit
- 1010 .. 1111 (δεκαδικοί 10..15) δεν αναπαρίστανται (άκυρες λέξεις για BCD)

TABLE 1-3
Binary-Coded Decimal (BCD)

Decimal Symbol	BCD Digit
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Table 1-3 Binary-Coded Decimal (BCD)

Δεκαδικοί με Δυαδική Κωδικοποίηση (συν.)

- Για την κωδικοποίηση αριθμών με *n* δεκαδικά ψηφία, χρειαζόμαστε *4n* bits στο BCD
  π.χ. $(365)_{10}$ = $(0011\ 0110\ 0101)_{BCD}$
- Αυτό είναι διαφορετικό από την μετατροπή σε δυαδικό όπου $(365)_{10}$ = $(101101101)_2$
- Ο κώδικας BCD χρειάζεται περισσότερα bits. Όμως, παρέχει μεγαλύτερη ευκολία στην ανάγνωση/ερμηνεία.

Πρόσθεση με BCD

- Όταν 2 κώδικες BCD προστίθενται:
  - Εάν το δυαδικό άθροισμα είναι μικρότερο από 1010_2 ($=10_{10}$), το αποτέλεσμα της πρόσθεσης δίνει έγκυρο και ορθό κώδικα για BCD
  - Εάν το δυαδικό άθροισμα είναι ίσο ή μεγαλύτερο από 1010_2, τότε το αποτέλεσμα της πρόσθεσης δίνει άκυρο ή/και λανθασμένο κώδικα για BCD. Διορθώνεται με την πρόσθεση του 0110_2 ($=6_{10}$) στο αποτέλεσμα της πρόσθεσης, έτσι ώστε να παραχθεί το σωστό κρατούμενο στο αριστερό ψηφίο.
    Γιατί 6_{10};

Πρόσθεση με BCD (συν.)

- Παράδειγμα: Πρόσθεση 448 & 489 σε BCD.

```
0100   0100   1000   (448 σε BCD)
0100   1000   1001   (489 σε BCD)
             10001   ( > 9, πρόσθεσε 6)
            1 0111   (κρατούμενο 1 στο μεσαίο ψηφίο)
       1101          ( > 9, πρόσθεσε 6)
1001 1 0011          (κρατούμενο 1 στο αριστερότερο ψηφίο)
1001   0011   0111   (κώδικας BCD για 937₁₀)
```


Κώδικας Gray

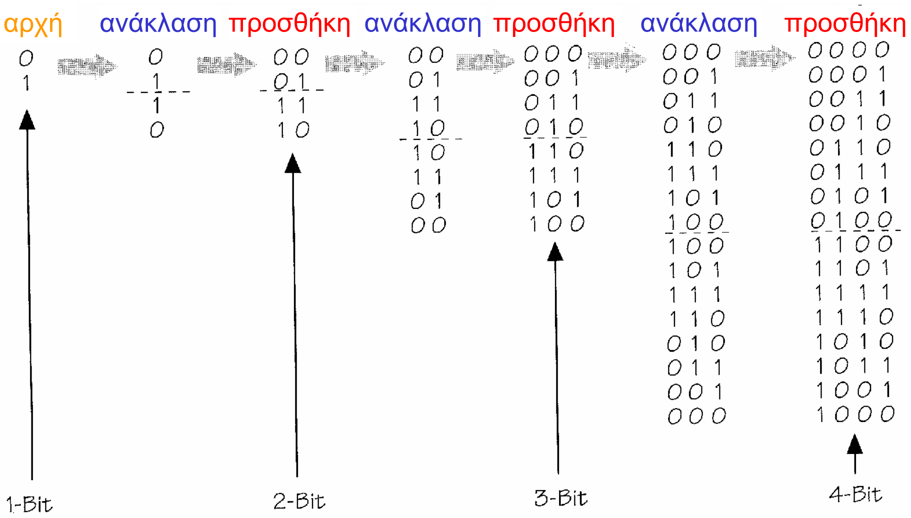
- Απόσταση Hamming: Ο # των αλλαγών στις τιμές των bit μεταξύ δύο δυαδικών τιμών/κωδίκων
- Στον κώδικα Gray, η απόσταση Hamming πρέπει να είναι 1 μεταξύ κάθε δύο συνεχόμενων κωδίκων
- Σε ένα κώδικα Gray των *n* *κωδίκων* (*n* άρτιος):
  - Οι πρώτοι *n/2* κώδικες έχουν 0 για MSB και άρτια ισοτιμία μεταξύ συνεχόμενων bit
  - Οι υπόλοιποι n/2 παράγονται παίρνοντας την πρώτη λίστα ανάποδα, με 1 για MSB

Για οκταδικό

Δυαδικός κώδικας	# αλλαγών	Κώδικας Gray	# αλλαγών
000		000	
001	1	001	1
010	2	011	1
011	1	010	1
100	3	110	1
101	1	111	1
110	2	101	1
111	1	100	1
000	3	000	1

Κώδικας Gray

Παραγωγή των λέξεων του κώδικα Gray με ανάκλαση



Κωδικοποίησης με πίνακες αντιστοιχίας

- Ο πίνακας αντιστοιχίας είναι η απλούστερη μέθοδος κωδικοποίησης. Παράδειγμα ο Τηλεφωνικός Κατάλογος, όπου αντιστοιχείται το όνομα του συνδρομητή με τον αριθμό του τηλεφώνου του.
- Στους υπολογιστές χρησιμοποιείται ευρύτατα για τη δυαδική παράσταση αλφαριθμητικών πληροφοριών ένας διεθνώς καθιερωμένος πίνακας αντιστοιχίας, ο ονομαζόμενος **πίνακας** ASCII. Στην τυπική του μορφή, χρησιμοποιεί 7 bits, ήτοι 128 λέξεις, για να κωδικοποιήσει σε δυαδική μορφή:
- **α)** *αλφαριθμητικούς* χαρακτήρες, δηλ.:
  τα γράμματα του λατινικού αλφαβήτου μικρά και κεφαλαία,
  τα σημεία στίξεως,
  τα αριθμητικά ψηφία 0 … 9, και
- **β)** χαρακτήρες *ελέγχου,* δηλ. λέξεις που χρησιμοποιούνται για το συντονισμό της επικοινωνίας δύο συσκευών (αρχή μηνύματος, τέλος, stop κ.λπ.).

Κώδικας ASCII (American Standard Code for Information Interchange)

Bits	$b_7\,b_6\,b_5$	000	001	010	011	100	101	110	111
$b_4\,b_3\,b_2\,b_1$	HEX	0	1	2	3	4	5	6	7
0 0 0 0	0	NUL	DLE	SP	0	@	P	`	p
0 0 0 1	1	SOH	DC1	!	1	A	Q	a	q
0 0 1 0	2	STX	DC2	"	2	B	R	b	r
0 0 1 1	3	ETX	DC3	#	3	C	S	c	s
0 1 0 0	4	EOT	DC4	$	4	D	T	d	t
0 1 0 1	5	ENQ	NAK	%	5	E	U	e	u
0 1 1 0	6	ACK	SYN	&	6	F	V	f	v
0 1 1 1	7	BEL	ETB	'	7	G	W	g	w
1 0 0 0	8	BS	CAN	(	8	H	X	h	x
1 0 0 1	9	HT	EM	)	9	I	Y	i	y
1 0 1 0	A	LF	SUB	*	:	J	Z	j	z
1 0 1 1	B	VT	ESC	+	;	K	[	k	{
1 1 0 0	C	FF	FS	,	<	L	\	l	|
1 1 0 1	D	CR	GS	-	=	M	]	m	}
1 1 1 0	E	SO	RS	.	>	N	^	n	~
1 1 1 1	F	SI	US	/	?	O	-	o	DEL

Ο Κώδικας ΕΛΟΤ 928

Ο Ελληνικός Οργανισμός Τυποποίησης, ΕΛΟΤ, έχει αναπτύξει τον πρότυπο κώδικα ΕΛΟΤ 928 για την ενιαία παράσταση Ελληνικών και Λατινικών χαρακτήρων.

Bits	$b_8\,b_7\,b_6\,b_5$	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
$b_4\,b_3\,b_2\,b_1$	HEX	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0 0 0 0	0			SP	0	@	P	`	p			NBSP	°	ΐ	Π	ΰ	π
0 0 0 1	1			!	1	A	Q	a	q			‘	±	Α	Ρ	α	ρ
0 0 1 0	2			"	2	B	R	b	r			’	2	Β		β	ς
0 0 1 1	3			#	3	C	S	c	s			£	3	Γ	Σ	γ	σ
0 1 0 0	4			$	4	D	T	d	t				΄	Δ	Τ	δ	τ
0 1 0 1	5			%	5	E	U	e	u				΅	Ε	Υ	ε	υ
0 1 1 0	6			&	6	F	V	f	v			¦	Ά	Ζ	Φ	ζ	φ
0 1 1 1	7			'	7	G	W	g	w			§	·	Η	Χ	η	χ
1 0 0 0	8			(	8	H	X	h	x			¨	Έ	Θ	Ψ	θ	ψ
1 0 0 1	9			)	9	I	Y	i	y			©	Ή	Ι	Ω	ι	ω
1 0 1 0	A			*	:	J	Z	j	z				Ί	Κ	Ϊ	κ	ϊ
1 0 1 1	B			+	;	K	[	k	{			«	»	Λ	Ϋ	λ	ϋ
1 1 0 0	C			,	<	L	\	l	|			¬	Ό	Μ	ά		ό
1 1 0 1	D			-	=	M	]	m	}			SHY	½	Ν	έ	ν	ύ
1 1 1 0	E			.	>	N	^	n	~				Ύ	Ξ	ή	ξ	ώ
1 1 1 1	F			/	?	O	-	o	DEL			-	Ώ	Ο	ί	ο	

Κώδικας UNICODE

- Υποστηρίζεται από την Java, Windows NT
- Κάθε χαρακτήρας → 16 bits
- $2^{16} \rightarrow 65536$
- Οι πρώτοι 128 χαρακτήρες είναι ίδιοι με τον ASCII
- Ομάδες χαρακτήρων. Λατινικά: 336 σημεία, Ελληνικά 144, εβραϊκά 112 κ.ο.κ.
- Ούτε ο UNICODE είναι επαρκής: Στον πλανήτη χρησιμοποιούνται γύρω στους 200.000 χαρακτήρες !!!

Δυαδικές αναπαραστάσεις (1/2)

- Κώδικες που αναπαριστούν Θετικούς Ακεραίους:
  - Δυαδικός (συνηθισμένος)
  - Οκταδικός
  - Δεκαεξαδικός
  - Gray
  - BCD
- Κώδικες που αναπαριστούν Αρνητικούς Ακεραίους:
  - Προσημασμένου μεγέθους
  - Συμπληρώματος ως προς 1
  - Συμπληρώματος ως προς 2

Δυαδικές αναπαραστάσεις (2/2)

- Κώδικες που αναπαριστούν πραγματικούς Αριθμούς:
  - Απλής- και διπλής-ακρίβειας κινητής υποδιαστολής
  (single- and double-precision floating point)

- Κώδικες που αναπαριστούν χαρακτήρες:
  - ASCII
  - Unicode

Προσημασμένοι αριθμοί

Για την παράσταση θετικών και αρνητικών αριθμών σε μορφή σταθερής υποδιαστολής υπάρχουν τρεις συμβολισμοί. Και στους τρεις συμβολισμούς το ψηφίο που βρίσκεται στην αριστερότερη θέση της παράστασης δηλώνει εάν ο αριθμός είναι θετικός ή αρνητικός. Εάν το ψηφίο αυτό έχει την τιμή 0, ο αριθμός είναι θετικός, ενώ εάν έχει την τιμή β–1, ο αριθμός είναι αρνητικός. Οι τρείς συμβολισμοί έχουν ως εξής:

Προσημασμένοι αριθμοί βάσης β

α. Παράσταση προσημασμένου μεγέθους (sign–magnitude representation).

$N(\beta) = 0a_{\nu-2}a_{\nu-3} \ldots a_1a_0,$

$-N(\beta) = (\beta-1)a_{\nu-2}a_{\nu-3} \ldots a_1a_0,$

β. Παράσταση συμπληρώματος ως προς ελαττωμένη βάση (diminished – radix complement representation).

$N(\beta) = 0a_{\nu-2}a_{\nu-3} \ldots a_1a_0,$

$-N(\beta) = (\beta-1)\, \bar{a}_{\nu-2}\bar{a}_{\nu-3} \ldots \bar{a}_1\bar{a}_0,$

όπου $\bar{a}_\lambda = (\beta-1) - a_\lambda$ για $0 \ldots \lambda \ldots \nu-2$.

Αυτή η παράσταση καλείται επίσης παράσταση συμπληρώματος ως προς β–1.

γ. Παράσταση συμπληρώματος ως προς βάση (radix complement representation).

$N_{(\beta)} = 0\, a_{\nu-2}a_{\nu-3} \ldots a_1a_0,$

$-N_{(\beta)} = (\beta-1)\, \bar{a}_{\nu-2}\bar{a}_{\nu-3} \ldots \bar{a}_1\bar{a}_0 + 1$

Προσημασμένοι δυαδικοί αριθμοί

Στο δυαδικό σύστημα, για να διακρίνουμε τους αριθμούς σε θετικούς και αρνητικούς, χρησιμοποιούμε σαν πρόσημα τα ψηφία 0 και 1. Κατά σύμβαση ένας αριθμός είναι **θετικός** όταν το πρώτο bit από αριστερά είναι **0** και **αρνητικός** όταν το bit αυτό είναι 1, π.χ. ο αριθμός 01011 είναι θετικός και ο 11010 είναι αρνητικός.

Επειδή η χρησιμοποίηση των 0 και 1 ταυτόχρονα σαν προσήμων και σαν ψηφίων του αριθμού δημιουργεί σύγχυση, είναι απαραίτητο, όταν αναφερόμαστε σε προσημασμένους αριθμούς, να αναφέρουμε και το πλήθος των ψηφίων τους. Για παράδειγμα ο αριθμός 11010 είναι αρνητικός όταν θεωρηθεί ότι έχει 5 ψηφία και θετικός, όταν έχει 6 ψηφία δηλ. γραφεί ως 011010.

Παράδειγμα

Στο δυαδικό σύστημα αρίθμησης χρησιμοποιώντας 8 δυαδικά ψηφία οι αριθμοί $X_{(10)} = 11$ και $X_{(10)} = -11$ έχουν αντίστοιχα την ακόλουθη μορφή:

α. Παράσταση προσημασμένου μεγέθους

$X_{(10)} = 11 \rightarrow X_{(2)} = 00001011$

$X_{(10)} = -11 \rightarrow X_{(2)} = 10001011$

β. Παράσταση συμπληρώματος ως προς ελαττωμένη βάση

$X_{(10)} = 11 \rightarrow X_{(2)} = 00001011$

$X_{(10)} = -11 \rightarrow X_{(2)} = 11110100$

γ. Παράσταση συμπληρώματος ως προς βάση

$X_{(10)} = 11 \rightarrow X_{(2)} = 00001011$

$X_{(10)} = -11 \rightarrow X_{(2)} = 11110101$

Προσημασμένοι δυαδικοί αριθμοί

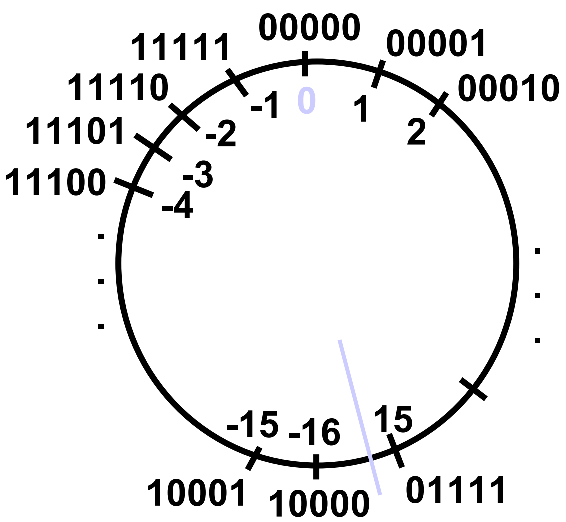
Περιοχές ακεραίων και παραστάσεις του μηδενός για δυαδική αριθμητική σταθερής υποδιαστολής.

Σύστημα Αναπαράστασης Προσημασμένων Αριθμών	Περιοχή Ακεραίων	Παραστάσεις Μηδενός
Προσημασμένου μεγέθους	$-(2^{\nu-1}-1) \leq A \leq (2^{\nu-1}-1)$ $(111\ldots1) \leq A \leq (011\ldots1)$	00…0 και 100…0
Συμπληρώματος ως προς 1	$-(2^{\nu-1}-1) \leq A \leq (2^{\nu-1}-1)$ $(100\ldots0) \leq A \leq (011\ldots1)$	00…0 και 11…1
Συμπληρώματος ως προς 2	$-2^{\nu-1} \leq A \leq (2^{\nu-1}-1)$ $(100\ldots0) \leq A \leq (011\ldots1)$	00…0

Προσημασμένοι δυαδικοί αριθμοί

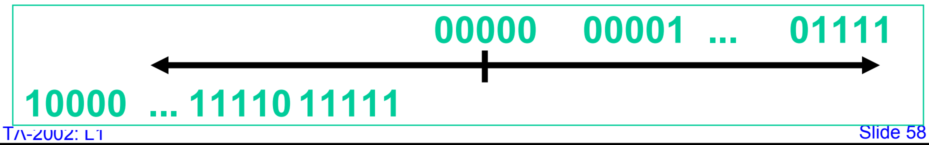
Δεκαδικός	Προσημασμένο συμπλήρωμα ως προς 2	Προσημασμένο συμπλήρωμα ως προς 1	Προσημασμένο μέγεθος
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
+4	0100	0100	0100
+3	0011	0011	0011
+2	0010	0010	0010
+1	0001	0001	0001
+0	0000	0000	0000
−0	–	1111	1000
−1	1111	1110	1001
−2	1110	1101	1010
−3	1101	1100	1011
−4	1100	1011	1100
−5	1011	1010	1101
−6	1010	1001	1110
−7	1001	1000	1111
−8	1000	–	–

Σχηματική διάταξη αριθμών στο Συμπλ. ως προς 2



N = 5

- 2^{N-1} μη-αρνητικοί
- 2^{N-1} αρνητικοί
- Ένα μηδέν
- Πόσοι θετικοί;



Συμπλήρωμα ως προς 2 για N=32

$0000 \ldots 0000\ 0000\ 0000\ 0000_{10}$ =	0_{10}
$0000 \ldots 0000\ 0000\ 0000\ 0001_{10}$ =	1_{10}
$0000 \ldots 0000\ 0000\ 0000\ 0010_{10}$ =	2_{10}
. . .	
$0111 \ldots 1111\ 1111\ 1111\ 1101_{10}$ =	$2.147.483.645_{10}$
$0111 \ldots 1111\ 1111\ 1111\ 1110_{10}$ =	$2.147.483.646_{10}$
$0111 \ldots 1111\ 1111\ 1111\ 1111_{10}$ =	$2.147.483.647_{10}$
$1000 \ldots 0000\ 0000\ 0000\ 0000_{10}$ =	$-2.147.483.648_{10}$
$1000 \ldots 0000\ 0000\ 0000\ 0001_{10}$ =	$-2.147.483.647_{10}$
$1000 \ldots 0000\ 0000\ 0000\ 0010_{10}$ =	$-2.147.483.646_{10}$
. . .	
$1111 \ldots 1111\ 1111\ 1111\ 1101_{10}$ =	-3_{10}
$1111 \ldots 1111\ 1111\ 1111\ 1110_{10}$ =	-2_{10}
$1111 \ldots 1111\ 1111\ 1111\ 1111_{10}$ =	-1_{10}

- Ένα μηδέν; Το 1^{o} ψηφίο καλείται ψηφίο προσήμου (sign bit)
- ένας “επιπλέον” αρνητικός: δεν υπάρχει θετικός $2,147,483,648_{10}$

Τύπος του συμπληρώματος ως προς 2

- Μπορεί να αναπαριστά θετικούς και αρνητικούς αριθμούς ως συνάρτηση της τιμής του δυαδικού ψηφίου επί μία δύναμη του 2:

  $d_{31} \times -(2^{31}) + d_{30} \times 2^{30} + \ldots + d_2 \times 2^2 + d_1 \times 2^1 + d_0 \times 2^0$

- Παρ’αδειγμα: 1101_2

  $= 1\text{x}-(2^3) + 1\text{x}2^2 + 0\text{x}2^1 + 1\text{x}2^0$

  $= -2^3 + 2^2 + 0 + 2^0$

  $= -8 + 4 + 0 + 1$

  $= -8 + 5$

  $= -3_{10}$

Πρόσθεση προσημασμένων δυαδικών αριθμών

Το αποτέλεσμα της πρόσθεσης δύο προσημασμένων δυαδικών αριθμών που παριστώνται σε μορφή συμπληρώματος ως προς 2 προκύπτει από την πρόσθεση των δύο αριθμών, συμπεριλαμβανομένων και των μπιτ προσήμου. Το κρατούμενο εξόδου της θέσης του μπιτ του πρόσημου αγνοείται.

+ 6	00000110	− 6	11111010
+13	00001101	+13	00001101
+19	00010011	+ 7	00000111
+ 6	00000110	− 6	11111010
− 13	11110011	−13	11110011
− 7	11111001	−19	11101101

Αφαίρεση προσημασμένων δυαδικών αριθμών

Η αφαίρεση δύο δυαδικών αριθμών με πρόσημο, όταν οι αρνητικοί αριθμοί είναι σε μορφή συμπληρώματος ως προς 2, είναι πολύ απλή και μπορεί να περιγραφεί ως εξής:

Παίρνουμε το συμπλήρωμα ως προς δύο του αφαιρετέου (στον οποίο συμπεριλαμβάνεται και το μπιτ προσήμου) και το προσθέτουμε στο μειωτέο (στον οποίο συμπεριλαμβάνεται και το μπιτ προσήμου). Το κρατούμενο εξόδου στη θέση του μπιτ προσήμου αγνοείται.

Αυτή η μέθοδος είναι εφαρμόσιμη, επειδή μια πράξη αφαίρεσης μπορεί να μετατραπεί σε πράξη πρόσθεσης εάν το πρόσημο του αφαιρετέου αλλάξει. Αυτό φαίνεται από την παρακάτω σχέση:

$$(\pm A) - (+B) = (\pm A) + (-B)$$
$$(\pm A) - (-B) = (\pm A) + (+B)$$

Αφαίρεση προσημασμένων δυαδικών αριθμών

Το πρόβλημα, λοιπόν, της αφαίρεσης μετατοπίζεται στην εύρεση του αντι-θέτου του αριθμού. Η *εύρεση του αντιθέτου* ενός αριθμού στο συμπλήρωμα ως πρός 2 γίνεται πολύ εύκολα στο δυαδικό σύστημα με τον εξής τρόπο:

α. αντιστρέφουμε όλα τα bit του αριθμού, δηλ. όπου 0 → 1 και 1 → 0, και

β. στο αποτέλεσμα προσθέτουμε τη μονάδα.

Παράδειγμα

Αφαίρεση με υπολογισμό του αντιθέτου:

Έστω η πράξη (+1)–(+2). Ο αντίθετος του (+2) σε δυαδική μορφή είναι:

(+2) => 010	αντιστροφή bit	101	
	πρόσθεση 1	+ 1	
		110	=> (–2)

Η όλη διαδικασία της αφαίρεσης συντομεύεται, εάν γίνει κατευθείαν η πρό-σθεση των αριθμών:

(+1)	=>	001	
(– 2)	=>	101	αντιστροφή bit
		+ 1	πρόσθεση 1
		111	=> (–1)

Παράδειγμα

Αφαίρεση με υπολογισμό του αντιθέτου:

Έστω η αφαίρεση 46 – 17 στο συμπλήρωμα ως προς 2 με δυαδικούς αριθμούς των 8 bit. Είναι 46_{10} => 00101110_2 και 17_{10} => 00010001_2:

$$
\begin{array}{lcrcl}
(+46) & => & 00101110 & & \\
(-17) & => & 11101110 & & \\
 & & \underline{+ \qquad\quad 1} & & \\
 & & 00011101 & => & (+29)
\end{array}
$$

Υπερχείλιση συμπληρώματος ως προς 2

- Η πρόσθεση δύο θετικών αριθμών δίνει αρνητικό αποτέλεσμα
- Η πρόσθεση δύο αρνητικών αριθμών δίνει θετικό αποτέλεσμα

- **Σωστό αποτέλεσμα**

$$
\begin{array}{rr}
1111 & -1 \\
+\ 1010 & -6 \\
\hline
\cancel{1}\ 1001 & -7
\end{array}
\qquad
\begin{array}{rr}
0011 & +3 \\
+\ 0010 & +2 \\
\hline
0101 & +5
\end{array}
$$

- **Λάθος αποτέλεσμα**

$$
\begin{array}{rr}
0110 & +6 \\
+\ 0100 & +4 \\
\hline
1010 & -6
\end{array}
\qquad
\begin{array}{rr}
1001 & -7 \\
+\ 1010 & -6 \\
\hline
\cancel{1}\ 0011 & +3
\end{array}
$$

Διάφορα περί συμπληρώματος ως προς 2

- **Συμπλήρωμα ως προς 2 για μη-ακεραίους**
  - $1.6875_{10} = 01.1011_2$
  - $-1.6875_{10} = 10.0101_2$
- **Επέκταση πρόσημου**
  - Γράψε +6 και –6 ως συμπλήρωμα του 2
    - 0110 και 1010
  - Επέκταση στα 8-bit bytes
    - 00000110 και 11111010
- **Δεν μπορούμε να συμπεράνουμε την αναπαράσταση από τον αριθμό**
  - 11001 = 25 (unsigned)
  - 11001 = –9 (sign magnitude)
  - 11001 = –6 (ones complement)
  - 11001 = –7 (twos complement)