Ιόνιο Πανεπιστήμιο – Τμήμα Πληροφορικής
Εισαγωγή στην Επιστήμη των Υπολογιστών
2022-23

Πράξεις με δυαδικούς αριθμούς

(αριθμητικές πράξεις)

http://mixstef.github.io/courses/csintro/

Μ.Στεφανιδάκης

Πράξεις με δυαδικούς αριθμούς

- Δυαδικοί Αριθμοί

- Ο υπολογιστής μπορεί να εκτελέσει
  - Λογικές πράξεις
  - Αριθμητικές πράξεις
- Οι πράξεις εκτελούνται
  - Σε ομάδες bits (bytes ή πολλαπλάσιά τους)

Το Byte ως δυαδικός αριθμός

- Δυαδικοί αριθμοί

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

το περισσότερο σημαντικό bit

το λιγότερο σημαντικό bit

1	1	1	1	0	0	1	1
1x128	1x64	1x32	1x16	0x8	0x4	1x2	1x1

128 + 64 + 32 + 16 + 0 + 0 + 2 + 1 =

243 (δεκαδικό)

- Μετατροπή από το δυαδικό στο δεκαδικό σύστημα

!

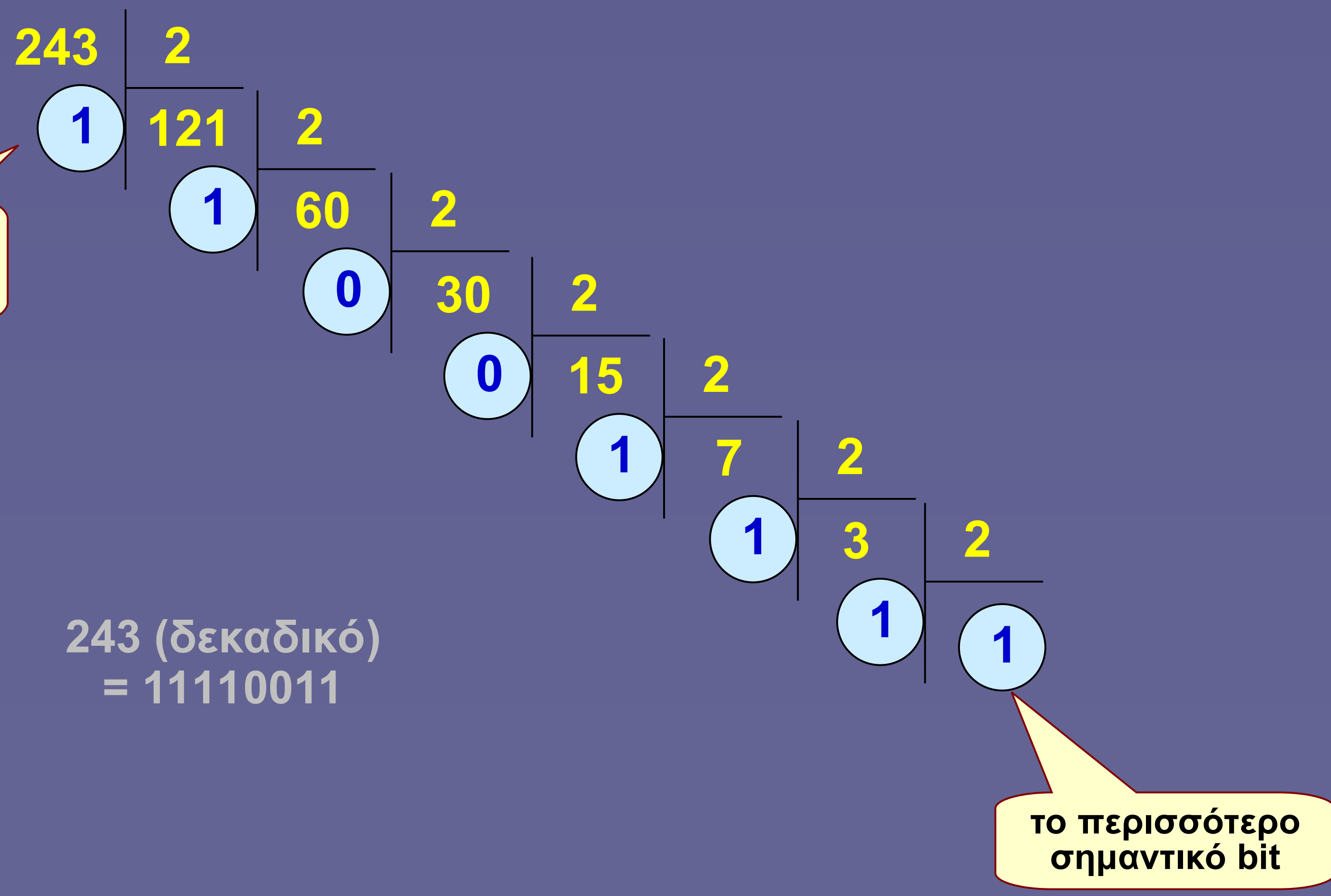
Εάν ο αριθμός διαθέτει περισσότερα bits, χρησιμοποιούμε μεγαλύτερες δυνάμεις του 2

Μετατροπή δεκαδικού σε δυαδικό

- Δυαδικοί αριθμοί

!

Προσοχή στη θέση του περισσότερο σημαντικού bit!



243 (δεκαδικό) = 11110011

Δεκαεξαδικό Σύστημα

- Δυαδικοί αριθμοί

- 16 ψηφία
  - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
  - Αντιστοιχία με τους δεκαδικούς 0 έως 15
- Σε δυνάμεις του 16
  - $16^n \ldots 16^4\ 16^3\ 16^2\ 16^1\ 16^0$
  - Π.χ. 16F(hex) = $1 \text{x} 16^2 + 6 \text{x} 16^1 + 15 \text{x} 16^0$
  - = 256 + 96 + 15 = 367 (δεκαδικό)
- Χρήσιμο μόνο ως «συντομογραφία» δυαδικών αριθμών

Δεκαεξαδικό Σύστημα

- Δυαδικοί αριθμοί

- Κάθε 4 δυαδικά ψηφία αντιστοιχούν σε ένα δεκαεξαδικό ψηφίο

0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

Παράδειγμα στο δεκαεξαδικό σύστημα

- Δυαδικοί αριθμοί

- Παράδειγμα: 1100100110010100

1100 1001 1001 0100

C 9 9 4 = C994(hex)

- Παράδειγμα: 10000101011110

0010 0001 0101 1110

2 1 5 E = 215E (hex)

  - Συμπλήρωση με 0 στα αριστερά
  - Δεν αλλάζει τον αριθμό, όπως ακριβώς και στο δεκαδικό σύστημα

Δεκαεξαδικό Σύστημα

- Δυαδικοί αριθμοί

- Κάθε 4 δυαδικά ψηφία αντιστοιχούν σε ένα δεκαεξαδικό ψηφίο

0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

Φυσικοί αριθμοί (χωρίς πρόσημο)

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί

Με κίτρινο φαίνεται ο ελάχιστος αριθμός bits που απαιτείται

0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
…..	…

- Με n bits περιγράφονται
  - Οι φυσικοί αριθμοί από 0 έως και 2^n-1

Χρήση των φυσικών αριθμών

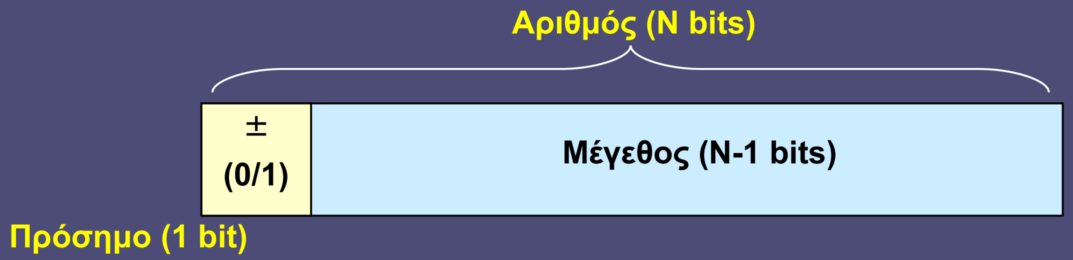
- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί

- Για αναπαράσταση
  - Διαφορετικών «πραγμάτων»
  - Συνήθως χωρίς αριθμητική έννοια
- Απαρίθμηση
  - Παρέχοντας μοναδικούς αναγνωριστικούς αριθμούς
  - Παραδείγματα
    - Οι ξεχωριστές διευθύνσεις μνήμης
    - Οι χαρακτήρες σε ένα αλφάβητο
- Ξανά: με *n* bits απαριθμούνται έως και 2^n *διαφορετικά «πράγματα»*

Ακέραιοι αριθμοί (με πρόσημο - signed)

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι

- Πώς θα αναπαρασταθούν οι αρνητικοί;
  - Για να γίνονται εύκολα οι πράξεις
- Όχι καλή ιδέα:
  - Ξεχωριστό bit πρόσημου



  - Διάστημα τιμών για αριθμούς με *n* bits

$-(2^{n-1}-1)$ *έως* $+(2^{n-1}-1)$ (για n=8, –127 ... +127)

    - ένα χρήσιμο bit λιγότερο
    - δυσκολία στις πράξεις
    - 2 αναπαραστάσεις του 0;

Ακέραιοι αριθμοί (με πρόσημο - signed)

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι

- Επίσης όχι καλή ιδέα:
  - Συμπλήρωμα ως προς 1
    - Αντιστροφή όλων των bits του αριθμού
    - Πιο σημαντικό bit: 0 για θετικούς, 1 για αρνητικούς
  - Διάστημα τιμών για αριθμούς με n bits

  $-(2^{n-1}-1)$ *έως* $+(2^{n-1}-1)$ (γιατί;)
  - Τα ίδια προβλήματα με την χρήση ξεχωριστού bit πρόσημου

Ακέραιοι αριθμοί (με πρόσημο - signed)

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι

- Καλή ιδέα!
  - Οι αρνητικοί αριθμοί είναι οι «συμπληρωμένοι ως προς 2» θετικοί
- Συμπλήρωμα ως προς 2
  - Τι σημαίνει «συμπλήρωμα ως προς 2»;
  - Πώς υπολογίζεται;

Συμπλήρωμα ως προς 2

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι

- Ίσο με το «συμπλήρωμα ως προς 1» + 1
- Εμπειρικός κανόνας:
  - Αντιστροφή όλων των bits εκτός από τα δεξιότερα συνεχόμενα 0 και το πρώτο 1 αριστερά από αυτά
- Συμπλήρωμα ως προς 2: παραδείγματα

001011100 ⇒ 110100100

011111111 ⇒ 100000001

- Προσοχή στο 0000...00 και στο 1000...00

Ακέραιοι σε συμπλήρωμα ως προς 2

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι

- Διάστημα τιμών για αριθμούς με *n* bits
  $-(2^{n-1})$ *έως* $+(2^{n-1}-1)$ (για n=8, –128 ... +127)
  - Μόνο το $+(2^{n-1})$ δεν μπορεί να αναπαρασταθεί
- Ευκολία στις πράξεις
  - αφαίρεση = πρόσθεση του συμπληρώματος ως προς 2
  - Μία και μοναδική αναπαράσταση του 0
- Πιο σημαντικό bit: 0 για θετικούς, 1 για αρνητικούς
  - Δεν είναι όμως bit προσήμου!

Αριθμητικές πράξεις

- Αριθμητικές πράξεις

- Οι βασικές πράξεις
  - Πρόσθεση
  - Αφαίρεση
- Άλλες πράξεις
  - Πολλαπλασιασμός
  - Διαίρεση
  - Επίσης:
    - Τετραγωνική ρίζα, τριγωνομετρικές συναρτήσεις, εκθετικά, λογάριθμοι κλπ..
    - Υλοποίηση σε υλικό με διάφορες τεχνικές
      - Π.χ με πολυώνυμα

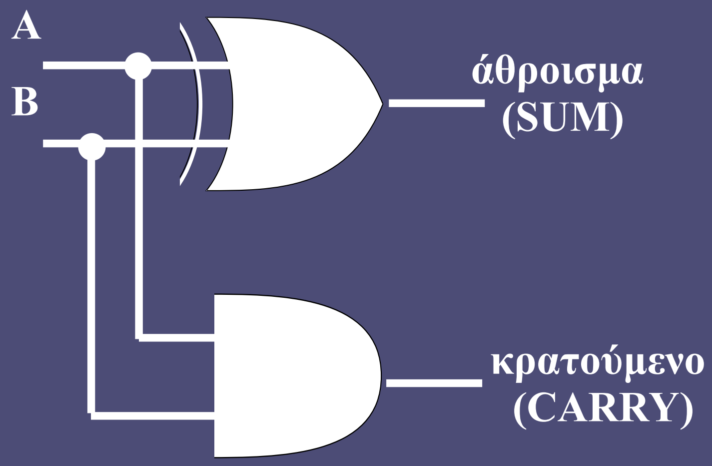
Προσθέτοντας 2 bits

- Αριθμητικές πράξεις

bits	άθροισμα	κρατούμενο
0 + 0	0	0
0 + 1	1	0
1 + 0	1	0
1 + 1	0	1

Ημιαθροιστής (half-adder)

- Αριθμητικές πράξεις



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

;

Αν απαιτείται πρόσθεση αριθμών με περισσότερα bits;

Προσθέτοντας δυαδικούς αριθμούς
(μη προσημασμένους)

Κρατούμενο		**1**	**1**	**1**				
Α’ Αριθμός (119)	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**1**
Β’ Αριθμός (88)	**0**	**1**	**0**	**1**	**1**	**0**	**0**	**0**
Άθροισμα (207)	**1**	**1**	**0**	**0**	**1**	**1**	**1**	**1**

1. Αριθμοί με ίδιο μήκος (ίσος αριθμός bits)
2. Αρχίζοντας από το λιγότερο σημαντικό bit (το δεξιότερο)
3. Προσθέτουμε ζεύγη bits και μεταφέρουμε το κρατούμενο (αν υπάρχει) προς τα αριστερά
   - Το προσθέτουμε στο επόμενο ζεύγος bits

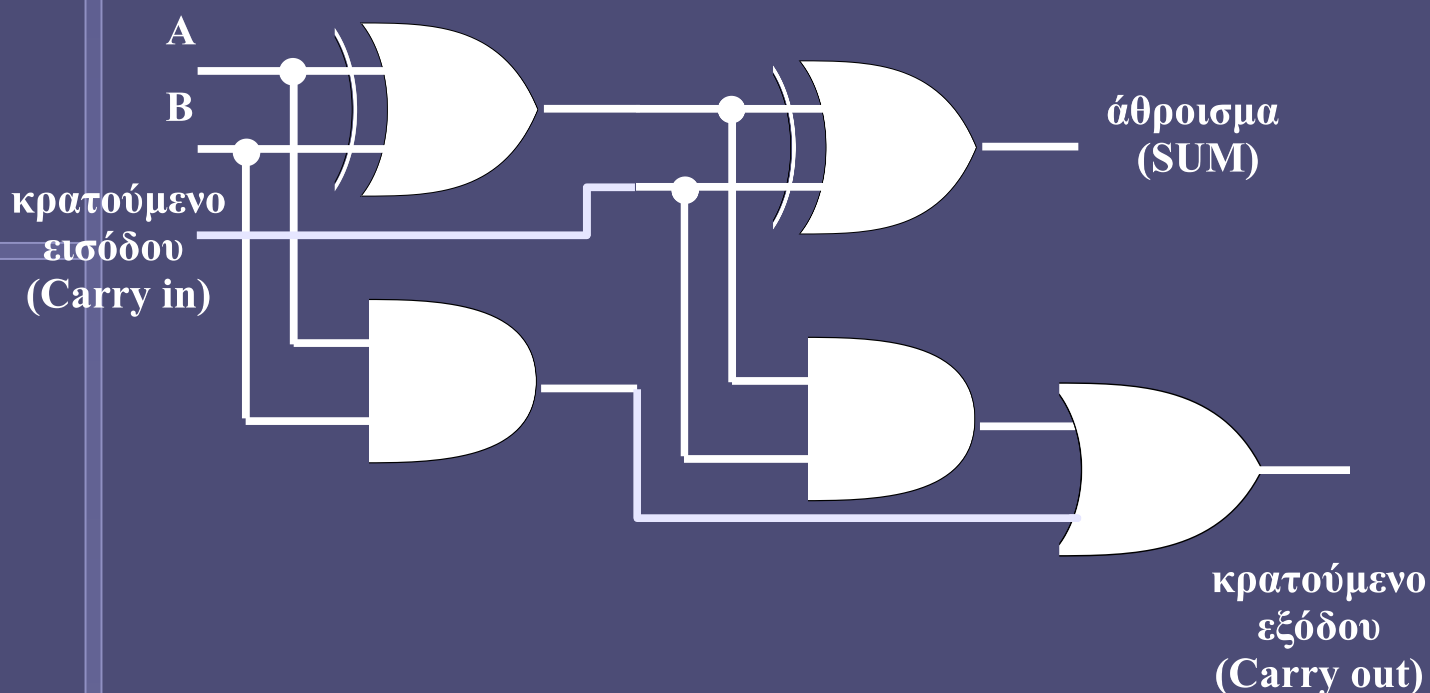
Πλήρης αθροιστής (full-adder)

- Αριθμητικές πράξεις

;

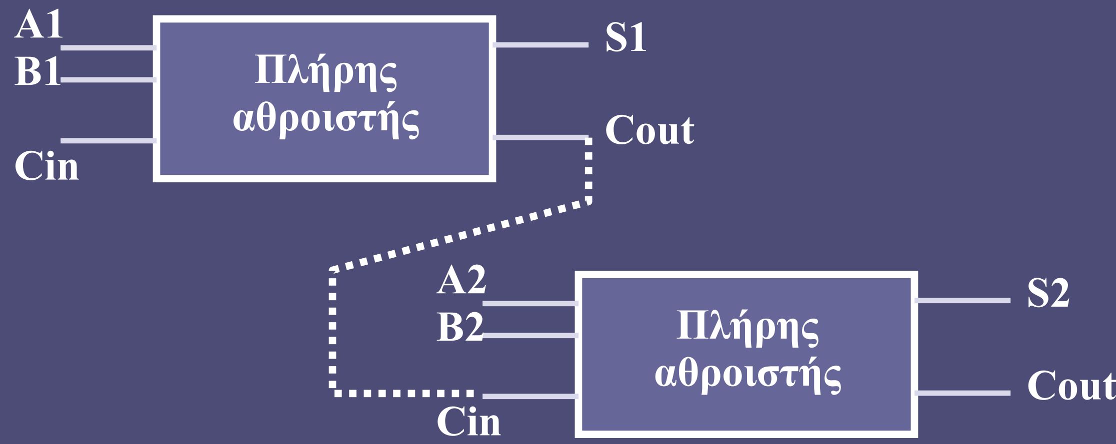
Ποιος πίνακας αλήθειας υλοποιείται;

- Μία από τις πιθανές υλοποιήσεις
  - με δύο ημιαθροιστές



Πρόσθεση αριθμών με πλήρεις αθροιστές

- Αριθμητικές πράξεις



- Πολλαπλά τμήματα πλήρη αθροιστή
  - Όμως: πόσο γρήγορα διαδίδεται το κρατούμενο; (ripple carry)
  - Τεχνικές πρόβλεψης κρατουμένου (carry look-ahead)

Προσθέτοντας δυαδικούς αριθμούς
(μη προσημασμένους)

- Υπερχείλιση
  - Στον υπολογιστή το πλήθος των bits ανά αριθμό είναι προκαθορισμένο
  - Το αποτέλεσμα της πρόσθεσης θα πρέπει να χωρά στα διαθέσιμα bits ενός καταχωρητή
  - Μη προσημασμένοι αριθμοί:
    - αριθμός με N bits ⇒ πεδίο τιμών [0 ... 2^N - 1]
    - π.χ. για αριθμούς με 8 bits, από 0 έως 255

Κρατούμενο		**1**	**1**	**1**	**1**	**1**	**1**		
Α' Αριθμός (180)		**1**	**0**	**1**	**1**	**0**	**1**	**0**	**0**
Β' Αριθμός (78)		**0**	**1**	**0**	**0**	**1**	**1**	**1**	**0**
Άθροισμα (258)	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

διαθέσιμος χώρος

ύπαρξη τελικού κρατουμένου = υπερχείλιση

Προσθέτοντας δυαδικούς αριθμούς
(προσημασμένους)

- Προσημασμένοι ακέραιοι
  - Συμπλήρωμα ως προς 2
    - Το περισσότερο σημαντικό bit υποδηλώνει το πρόσημο
    - 0=θετικός, 1=αρνητικός
  - αριθμός με N bits ⇒ πεδίο τιμών [-2^{N-1} ...0... $+2^{N-1} - 1$]
    - π.χ. για αριθμούς με 8 bits,   από -128 έως +127
- Πρόσθεση
  - Όπως σε μη προσημασμένους
  - Τελικό κρατούμενο αγνοείται
    - Πώς γίνεται τώρα ο έλεγχος υπερχείλισης;
  - Αφαίρεση = πρόσθεση του συμπληρώματος ως προς 2 του αφαιρετέου
    - A – B = A + (-B)
    - χωρίς πρόσθετα κυκλώματα για την αφαίρεση!

Προσθέτοντας δυαδικούς αριθμούς
(προσημασμένους)

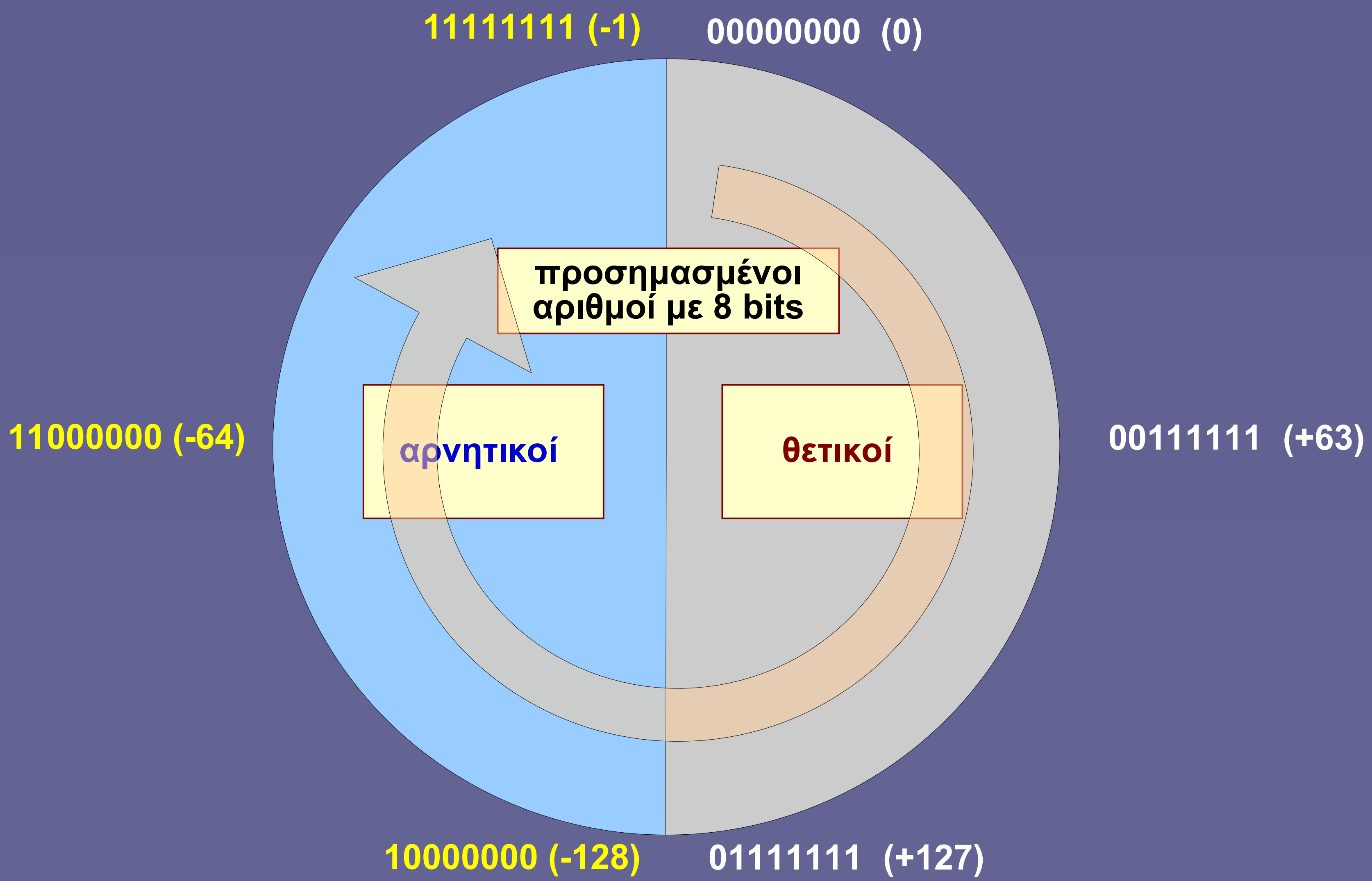
Κρατούμενο				**1**				
Α’ Αριθμός (+17)	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**1**
Β’ Αριθμός (+22)	**0**	**0**	**0**	**1**	**0**	**1**	**1**	**0**
Άθροισμα (+39)	**0**	**0**	**1**	**0**	**0**	**1**	**1**	**1**

Προσθέτοντας δυαδικούς αριθμούς
(προσημασμένους)

Κρατούμενο	~~1~~	1	1	1	1			
Α' Αριθμός (+24)	0	0	0	1	1	0	0	0
Β' Αριθμός (-17)	1	1	1	0	1	1	1	1
Άθροισμα (+7)	0	0	0	0	0	1	1	1

- το κρατούμενο αγνοείται

Υπερχείλιση σε προσημασμένους αριθμούς



Υπερχείλιση σε προσημασμένους αριθμούς

Κρατούμενο		**1**	**1**	**1**	**1**	**1**	**1**	**1**
Α’ Αριθμός (+127)	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Β’ Αριθμός (+3)	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**1**
Άθροισμα (-126;)	**1**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

- Το άθροισμα αριθμών με ίδιο πρόσημο θα πρέπει να έχει επίσης το ίδιο πρόσημο
  - στην αντίθετη περίπτωση: υπερχείλιση

Υπερχείλιση σε προσημασμένους αριθμούς

Κρατούμενο	~~**1**~~						**1**	
Α' Αριθμός (-126)	**1**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
Β' Αριθμός (-5)	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**0**
Άθροισμα (+124;)	**0**	**1**	**1**	**1**	**1**	**1**	**0**	**0**

- Το άθροισμα αριθμών με ίδιο πρόσημο θα πρέπει να έχει επίσης το ίδιο πρόσημο
  - στην αντίθετη περίπτωση: υπερχείλιση

Κλασματικοί αριθμοί

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι
- Κλασματικοί

- Θεωρητικά
  - Θα μπορούσαμε να επεξεργαζόμαστε ξεχωριστά το ακέραιο και το κλασματικό μέρος
- Αλλά
  - Αδυναμία αναπαράστασης πολύ μεγάλων και πολύ μικρών αριθμών
- Η λύση
  - Αριθμοί κινητής υποδιαστολής (floating point)
  - Εύκολη αναπαράσταση τόσο του 1.000.000.000.000 όσο και του 0,0000000000000001

Αριθμοί κινητής υποδιαστολής

- Δυαδικοί αριθμοί
- Φυσικοί αριθμοί
- Ακέραιοι
- Κλασματικοί

i

Το πρότυπο που περιγράφεται (IEEE 754) δεν είναι το μόνο. Στις εφαρμογές AI χρησιμοποιούνται και μορφές με λιγότερα bits

- 3 μέρη
  - Πρόσημο (Π) (1 bit)
    - 0 = +   1 = -
  - Εκθέτης (Ε) (8 ή 11 bits)
    - Η βάση είναι το 2 (εννοείται)
    - Θετικοί και αρνητικοί εκθέτες με πλεόνασμα 127 ή 1023 (π.χ. αντί -55, Ε= -55+127 = 72!)
  - Σημαινόμενο τμήμα (Σ) (23 ή 52 bits)
    - Κανονικοποίηση: μορφή 1,xxxxxxxxxxxx…
    - Το ‘1,’ εννοείται και δεν αποθηκεύεται
- Τελικός αριθμός: $-1^{Π} \times 1.Σ \times 2^{Ε-127}$ (ή $2^{Ε-1023}$)
  - Ειδικοί αριθμοί: 0, ∞, NaN (Not a Number)

Πράξεις με αριθμούς κινητής υποδιαστολής

- Αριθμητικές πράξεις

- Σύνθετη διαδικασία
- Η γενική μορφή της πρόσθεσης:
  1. Σύγκριση προσήμων
     - αν είναι ίδια ⇒ πρόσθεση
     - αλλιώς ⇒ αφαίρεση
  2. Εξίσωση εκθετών
     - μετακίνηση υποδιαστολής
  3. Πρόσθεση ή αφαίρεση σημαινόμενων τμημάτων
     - ακέραιο και κλασματικό μέρος
  4. Κανονικοποίηση αποτελέσματος
  5. Έλεγχος για υπερχείλιση

Πράξεις με αριθμούς κινητής υποδιαστολής

132

Α' αριθμός: 0 10000100 10110000000000000000000

$+ \quad 2^{132-127} \times 1{,}1011 \qquad (+2^{5} \times 1{,}1011)$

130

Β' αριθμός: 0 10000010 01100000000000000000000

$+ \quad 2^{130-127} \times 1{,}011 \quad (+2^{3} \times 1{,}011)$

A	$+2^{5}$	x	1,10110
+ B	$+2^{5}$	x	0,01011
=	$+2^{5}$	x	10,00001
κανονικοποίηση	$+2^{6}$	x	1,000001

αποτέλεσμα: 0 10000101 00000100000000000000000

133