

# Μικροεπεξεργαστές και Εφαρμογές 2016 - 2017

## 3<sup>η</sup> Εργαστηριακή Άσκηση.

Δημοκρίτειο Πανεπιστήμιο Θράκης  
Πολυτεχνική Σχολή  
Τμήμα Ηλεκτρολόγων Μηχανικών &  
Μηχανικών Υπολογιστών  
Τομέας Ηλεκτρονικής & Τεχνολογίας  
Συστημάτων Πληροφορικής  
Εργαστήριο Ηλεκτρονικής

Αναπληρωτής Καθηγητής  
ΓΕΩΡΓΙΟΣ ΣΥΡΑΚΟΥΛΗΣ

Δούρβας Νίκος  
Λιώλης Ορέστης  
Πιστόλα Θεοδώρα

# Αναπαράσταση Floating-point αριθμών με Βάση το IEEE 754 Standard

- Αρχικά μετατρέπουμε των αριθμό σε αυτή τη μορφή:

$$FP_{10} = (-1)^s * 2^{(b)} * 1, f$$

- Π.χ.  $5,34 = (-1)^0 * 2^2 * 1,335$



Single-precision format

- $e = 127 + b$  (exponent – 8 bits)
- $f = \frac{f_{22}}{2^1} + \frac{f_{21}}{2^2} + \frac{f_{20}}{2^3} + \dots + \frac{f_0}{2^{23}}$  (fraction – 23 bits)

# Πρόσθεση floating-point δεδομένων - 1

$$\begin{array}{r}
 5,34 \\
 + 7,53 \\
 \hline
 12,87
 \end{array}
 \quad \longrightarrow \quad
 \begin{array}{r}
 \text{+1} \\
 0 | 100\ 0000\ 1 | 010\ 1010\ 1110\ 0001\ 0100\ 1000 \\
 0 | 100\ 0000\ 1 | 111\ 0000\ 1111\ 0101\ 1100\ 0011 \\
 \hline
 0 | 100\ 0001\ 0 | 100\ 1101\ 1110\ 1011\ 1000\ 0101
 \end{array}$$

$$(2^2 * 1,335) + (2^2 * 1,8825) = 2^2 * 3,2175 = 2^3 * \frac{3,2175}{2} = 2^3 * 1,60875 = 12,87$$

$$\begin{array}{r}
 1,010\ 1010\ 1110\ 0001\ 0100\ 1000 \\
 1,111\ 0000\ 1111\ 0101\ 1100\ 0011 \\
 \hline
 11,001\ 1011\ 1101\ 0111\ 0000\ 1011
 \end{array}
 \xrightarrow{\text{LSRS } 1}
 1,100\ 1101\ 1110\ 1011\ 1000\ 0101\ 1$$

## Πρόσθεση floating-point δεδομένων - 2

$$\begin{array}{r} 5,34 \\ + 8,53 \\ \hline 13,87 \end{array}$$



$$\begin{array}{r} 0|100\ 0000\ 1|010\ 1010\ 1110\ 0001\ 0100\ 1000 \\ 0|100\ 0001\ 0|000\ 1000\ 0111\ 1010\ 1110\ 0001 \\ \hline 0|100\ 0001\ 0|101\ 1101\ 1110\ 1011\ 1000\ 0101 \end{array}$$

$$(2^2 * 1,335) + (2^3 * 1,06625) = (2^3 * \frac{1,335}{2}) + (2^3 * 1,06625) = 2^3 * 1,73375 = 13,87$$

$$\begin{array}{r} 1,010\ 1010\ 1110\ 0001\ 0100\ 1000 \\ 1,000\ 1000\ 0111\ 1010\ 1110\ 0001 \\ \hline \end{array} \xrightarrow{\text{LSRS } 1} \begin{array}{r} 0,101\ 0101\ 0111\ 0000\ 1010\ 0100 \\ 1,000\ 1000\ 0111\ 1010\ 1110\ 0001 \\ \hline 1,101\ 1101\ 1110\ 1011\ 1000\ 0101 \end{array}$$

# Αλγόριθμος πρόσθεσης floating-point δεδομένων - 1

1. Χωρίζουμε για κάθε αριθμό το exponent και το fraction
2. Συγκρίνουμε τα  $e_1$  και  $e_2$ 
  - a) Αν είναι ίσα, προσθέτουμε τα  $f_1$  και  $f_2$  και δημιουργούμε το  $f'$ .
  - b) Αν  $e_1 > e_2$ , αφαιρώ το  $e_2$  από το  $e_1$ . Κάνουμε δεξιά ολίσθηση στο  $f_2$  τόσες θέσεις όσες υποδεικνύει η παραπάνω αφαίρεση και στην συνέχεια προσθέτουμε τα  $f_1$  και  $f_2$  και δημιουργούμε το  $f'$ .
  - c) Αν  $e_1 < e_2$ , αφαιρώ το  $e_1$  από το  $e_2$ . Κάνουμε δεξιά ολίσθηση στο  $f_1$  τόσες θέσεις όσες υποδεικνύει η παραπάνω αφαίρεση και στην συνέχεια προσθέτουμε τα  $f_1$  και  $f_2$  και δημιουργούμε το  $f'$ .

## Πράξεις μεταξύ floating-point δεδομένων - 2

3. Δημιουργούμε το τελικό exponent, παίρνοντας το μεγαλύτερο από τα  $e_1$  και  $e_2$ .
4. Ελέγχουμε αν το  $f'$  ξεπερνά τα 24 bits
  - a) Αν δεν τα ξεπερνά, αφαιρούμε το πιο σημαντικό bit και φτιάχνουμε το  $f$ .
  - b) Αν τα ξεπερνά, κάνουμε δεξιά ολίσθηση κατά μια θέση (επειδή προσθέτουμε 2 αριθμούς) και μετά αφαιρούμε το πιο σημαντικό bit για να φτιάξουμε το  $f$ .
5. Τέλος ενώνουμε τα  $e$  και  $f$  και φτιάχνουμε τον τελικό αριθμό