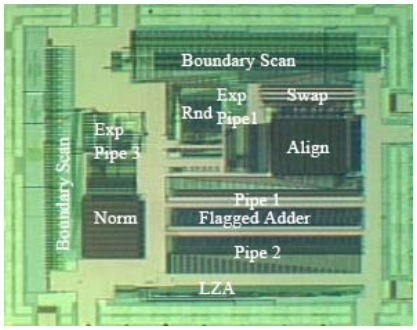
VLSI

REAL ARITHMETIC



•Floating- Point Numbers

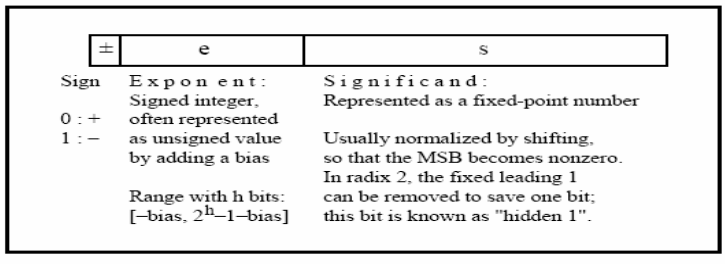
- ΚΑΝΕΝΑ ΑΡΙΘΜΗΤΙΚΟ ΣΥΣΤΗΜΑ ΔΕΝ ΜΠΟΡΕΙ ΝΑ ΑΠΕΙΚΟΝΙΣΕΙ ΟΛΟΥΣ ΤΟΥΣ ΠΡΑΓΜΑΤΙΚΟΥΣ ΑΡΙΘΜΟΥΣ, ΔΙΟΤΙ ΕΧΟΥΜΕ ΠΕΡΙΟΡΙΣΜΕΝΟ ΕΥΡΟΣ ΑΚΡΙΒΕΙΑΣ.
- ΥΠΑΡΧΟΥΝ ΔΙΑΦΟΡΑ ΣΥΣΤΗΜΑΤΑ ΠΟΥ ΜΠΟΡΟΥΝ ΝΑ ΑΠΕΙΚΟΝΙΣΟΥΝ ΣΥΓΚΕΚΡΙΜΕΝΑ ΥΠΟΣΥΝΟΛΑ ΤΩΝ ΠΡΑΓΜΑΤΙΚΩΝ ΑΡΙΘΜΩΝ.

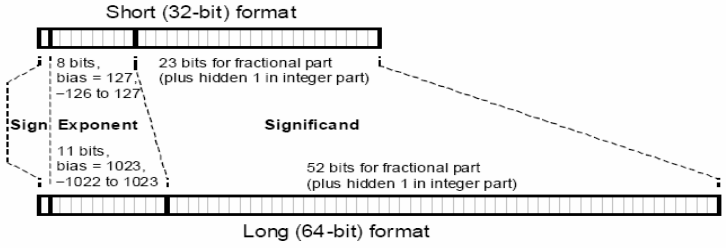
Floating- Point Numbers

- Σταθερής υποδιαστολής (Fixed-point) ± *w* **.** *f*
  low precision and/or range
- Κλασματικοί (Rational) ± *p* / *q* difficult arithmetic
- Κινητής Υποδιαστολής (Floating-point) ± *s* *b^e*
  most common scheme
- Logarithmic ± log*bx* limiting case of floating-point
- Fixed-point numbers
  x = (0000 0000 . 0000 1001) two Small number
  y = (1001 0000 . 0000 0000) two Large number
- Floating-point numbers
  x = ± *s***b*^*e* or ± significand * base^exponent

Floating- Point Numbers

- **Τυπική αναπαράσταση IEEE αριθμών κ.υ.**



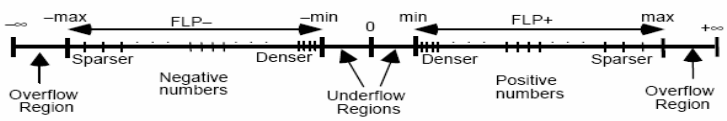


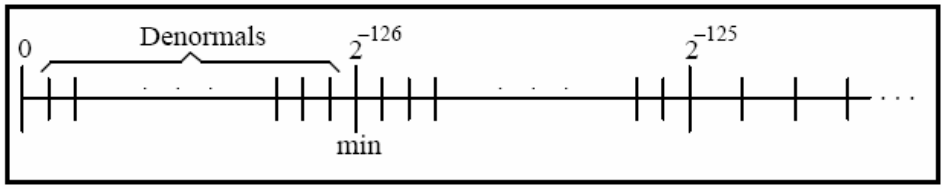
Floating- Point Numbers

Feature	Single/Short	Double/Long
Word width (bits)	32	64
Significand bits	23 + 1 hidden	52 + 1 hidden
Significand range	$[1, 2 - 2^{-23}]$	$[1, 2 - 2^{-52}]$
Exponent bits	8	11
Exponent bias	127	1023
Zero (±0)	$e + bias = 0$, $f = 0$	$e + bias = 0$, $f = 0$
Denormal	$e + bias = 0$, $f \neq 0$ represents $\pm 0.f \times 2^{-126}$	$e + bias = 0$, $f \neq 0$ represents $\pm 0.f \times 2^{-1022}$
Infinity (±∞)	$e + bias = 255$, $f = 0$	$e + bias = 2047$, $f = 0$
Not-a-number (NaN)	$e + bias = 255$, $f \neq 0$	$e + bias = 2047$, $f \neq 0$
Ordinary number	$e + bias \in [1, 254]$ $e \in [-126, 127]$ represents $1.f \times 2^e$	$e + bias \in [1, 2046]$ $e \in [-1022, 1023]$ represents $1.f \times 2^e$
min	$2^{-126} \cong 1.2 \times 10^{-38}$	$2^{-1022} \cong 2.2 \times 10^{-308}$
max	$\cong 2^{128} \cong 3.4 \times 10^{38}$	$\cong 2^{1024} \cong 1.8 \times 10^{308}$

Floating- Point Numbers

- **Υποσύνολα της αναπαράστασης των αριθμών κινητής υποδιαστολής.**





Floating- Point Numbers

- Τι ορίζει το IEEE floating-point standard:
  - Τα αποτελέσματα των τεσσάρων βασικών αριθμητικών πράξεων (+, - *, /) και της τετραγωνικής ρίζας πρέπει να ταιριάζουν με τα αποτελέσματα που θα παίρναμε αν στους υπολογισμούς χρησιμοποιούσαμε άπειρη ακρίβεια.

Floating- Point Numbers

- Τι ορίζει το IEEE floating-point standard:
  - Μεγαλύτερη εσωτερική ακρίβεια:
    - Single-extended: ≥ 11 bits for exponent
      ≥ 32 bits for significand
      bias unspecified, but ≥ exp range [–1022, 1023]
    - Double-extended: ≥ 15 bits for exponent
      ≥ 64 bits for significand
      exp range [–16 382, 16 383]

Floating- Point Numbers

- Τι ορίζει το IEEE floating-point standard:

  4 είδη στρογγύλευσης:

  α) Προς πλησιέστερο

  β) Προς ζυγό

  γ) Προς το άπειρο

  δ) Προς μηδέν (αποκοπή).

Floating- Point Numbers

- Τι ορίζει το IEEE floating-point standard:

Αριθμητικές εξαιρέσεις: Ο παρακάτω πίνακας δείχνει τις εξαιρέσεις και τους ειδικούς αριθμούς της αριθμητικής IEEE

Εξαίρεση	Παράδειγμα	Αποτ. σε IEEE
Invalid op	0/0,0*∞,√(-1)	NaN
overflow		±Inf
Divide by 0	Finite number/0	±Inf
Underflow		Subnormal numbers
inexact	Av fl(x y)≠x y	στρογγύλευση

Basic Floating-Point Algorithms

- **Addition/Subtraction**
- **Multiplication**
- **Division**
- **Square-rooting**

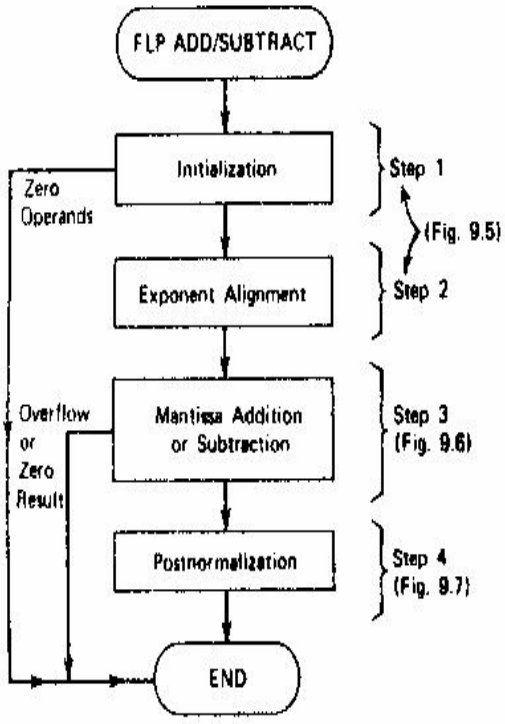
Floating- Point Numbers

Addition / Subtraction

Floating-Point Adders/Subtractors

- Υπάρχουν τέσσερα βασικά βήματα για την εκτέλεση πρόσθεσης/αφαίρεσης floating-point αριθμών:

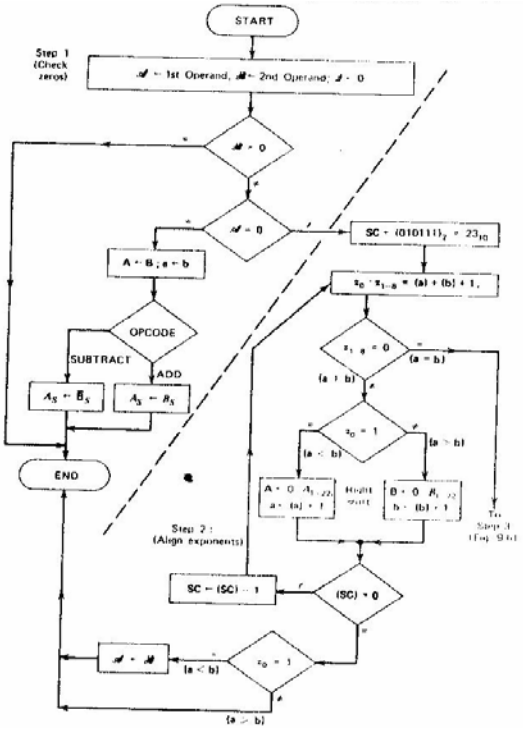
- Έλεγχος για μηδενικούς τελεστές
- Εξισώνει τους εκθέτες στην ίδια τάξη
- Πρόσθεση/αφαίρεση της μάντισσας
- Κανονικοποίηση του αποτελέσματος



Floating-Point Adders/Subtractors

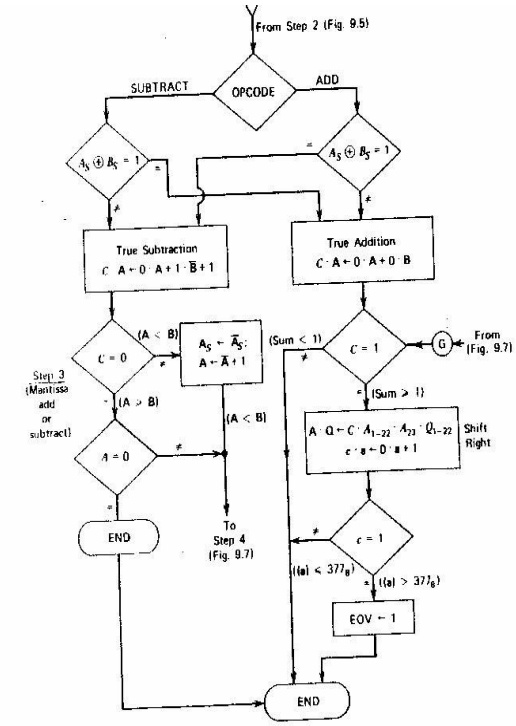
- Περιγραφή των 2 πρώτων βημάτων:

- Αν ο 2ος τελεστής είναι μηδέν δεν εκτελείται πράξη
- Αν ο 1ος τελεστής είναι μηδέν το αποτέλεσμα στην πρόσθεση είναι ίσο με 2ο τελεστή, ενώ στην αφαίρεση ισούται με το 2ο τελεστή με αλλαγμένο πρόσημο.
- Το 2ο βήμα εκτελείται εάν και μόνο αν και οι 2 τελεστές είναι μη μηδενικοί
- Όταν οι 2 εκθέτες δεν είναι ίσοι, η μάντισσα του μικρότερου τελεστή θα ολισθήσει τόσο ώστε οι εκθέτες να εξισωθούν.



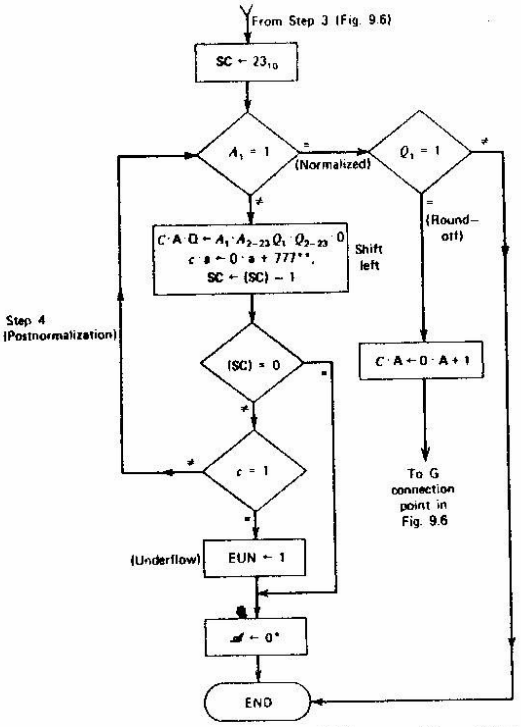
Floating-Point Adders/Subtractors

- Περιγραφή 3ου βήματος:
  - Οι διαδικασίες πρόσθεσης & αφαίρεσης γίνονται δεδομένου ότι οι εκθέτες είναι ίσοι.
  - Αν το άθροισμα υπερβαίνει την μονάδα γίνεται κανονικοποίηση.
  - Κρατούμενο εξόδου από MSB δεν σημαίνει ότι υπάρχει overflow.

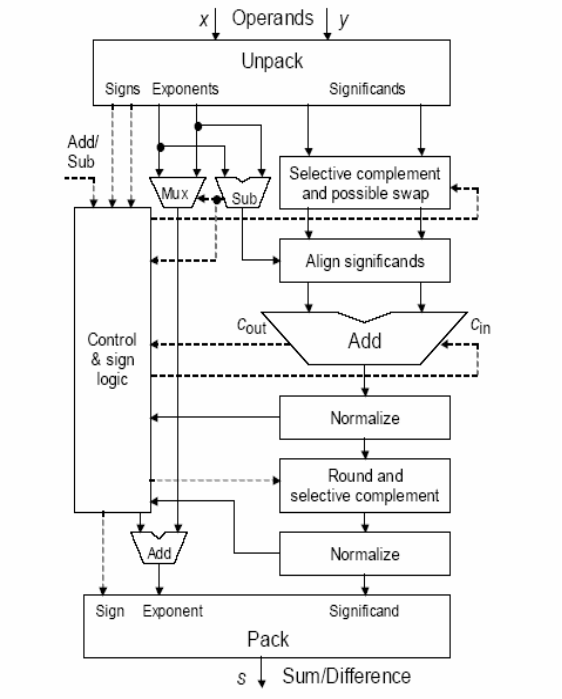


Floating-Point Adders/Subtractors

- Περιγραφή 4ου βήματος:
  - Ο καταχωρητής C.A.Q ολισθαίνει κατά μια θέση τη φορά, μέχρι το MSB να γίνει 1.
  - Το «στρογγυλοποιημένο» 1 προστίθεται στη μάντισσα
  - Μετά απο 23 shift – left αν το αποτέλεσμα δεν είναι κανον/νο, το αποτέλεσμα θεωρείται μηδενικού εύρους.



Floating-Point Adders/Subtractors



- **Αναλυτικό Block διάγραμμα ενός floating-point adder/subtractor.**

Floating-Point Adders/Subtractors

Unpack :

- Ξεχωρίζει το πρόσημο , τον εκθέτη και την μάντισσα.
- Επαναεισαγωγή του «κρυμμένου» 1.
- Μετατροπή των τελεστών σε εσωτερική απεικόνιση.
- Έλεγχος ειδικών τελεστών και σφαλμάτων.

Floating-Point Adders/Subtractors

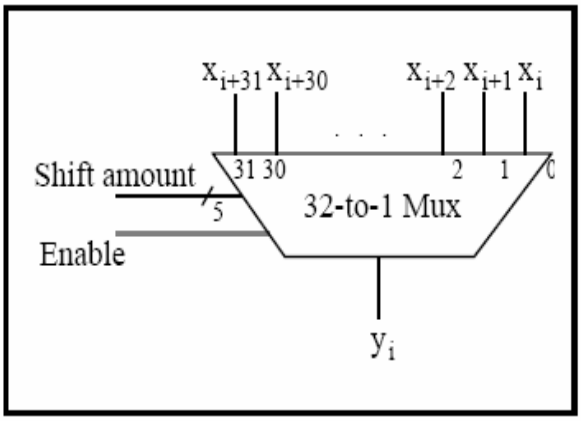
Pack :

- Ενώνει πρόσημο, εκθέτη , μάντισσα.
- Κρύβει το πρώτο 1.
- Έλεγχος για ειδικές περιπτώσεις και σφάλματα.
- Η μετατροπή απο την εσωτερική στην αναπαράσταση εξόδου, αν είναι αναγκαίο, πρέπει να γίνει στο στάδιο της στρογγυλοποίησης.

Floating-Point Adders/Subtractors

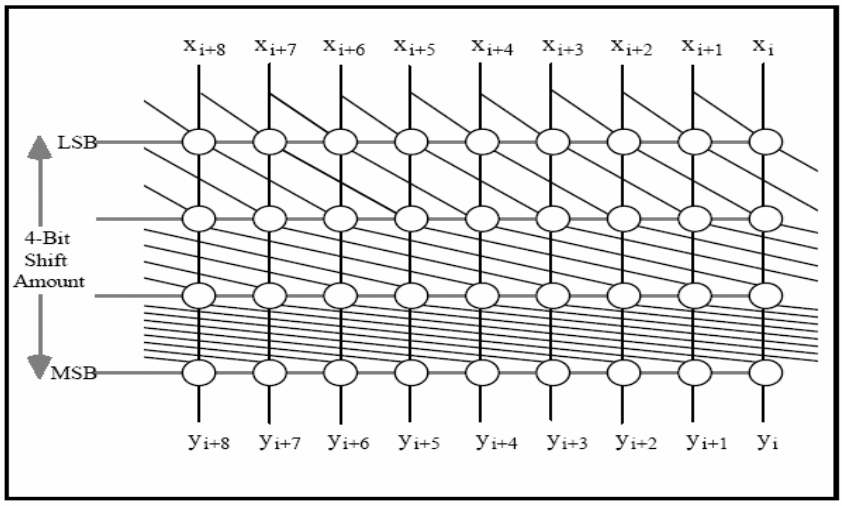
Μερικά άλλα μέρη του κυκλώματος:

- Preshifter



One bit-slice of a single-stage pre-shifter.

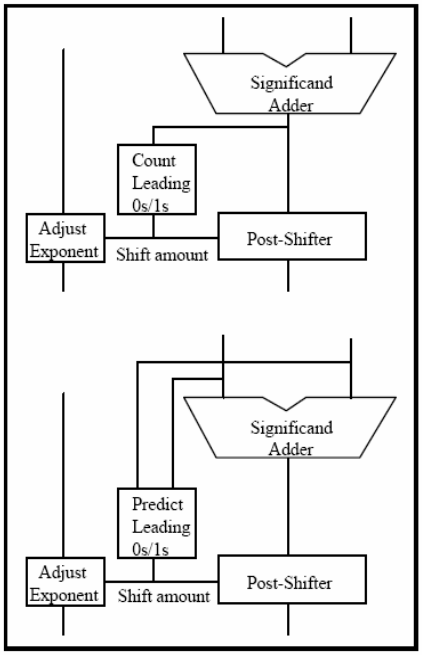
Floating-Point Adders/Subtractors



Floating-Point Adders/Subtractors

Μερικά άλλα μέρη του κυκλώματος:

- Κανονικοποιητής αποτελέσματος ή postshifter, (περιέχει ανιχνευτή /προβλεπτή για αρχικά μηδενικά).



Leading zeros/ones counting versus prediction.

Floating-Point Adders/Subtractors

Μερικά άλλα μέρη του κυκλώματος:

- Μονάδα στρογγυλοποίησης:

  Adder result = $(c_{out} z_1 z_0 . z_{-1} z_{-2} \ldots z_{-l}\ G\ R\ S)_{2's\text{-}compl}$

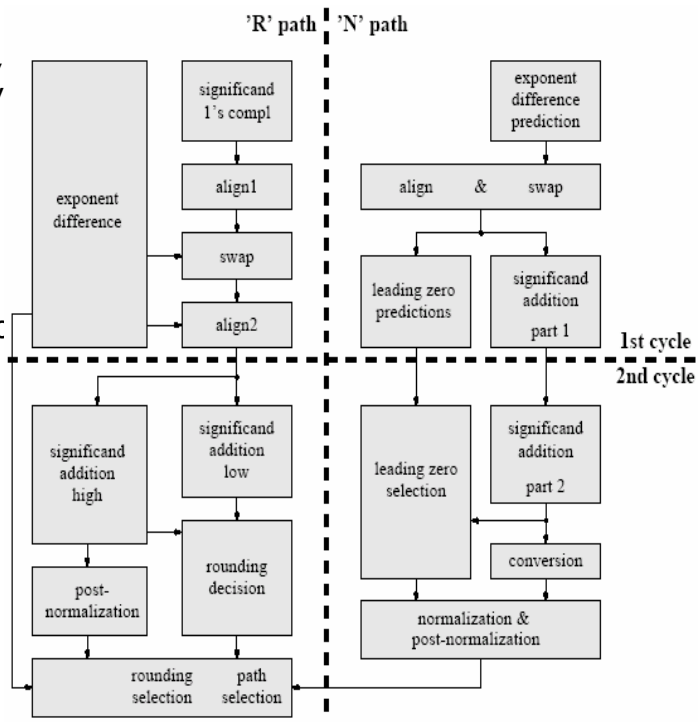
  G : Guard bit

  R : Round bit

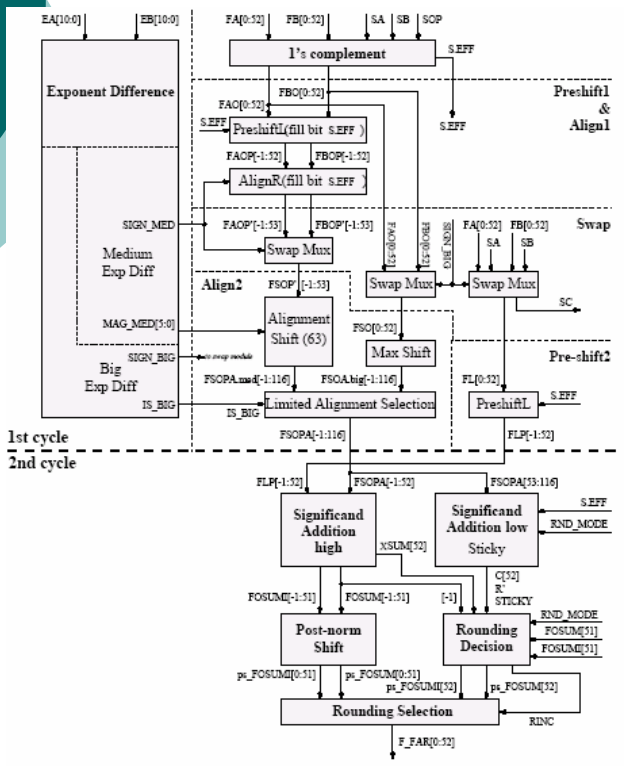
  S : Sticky bit

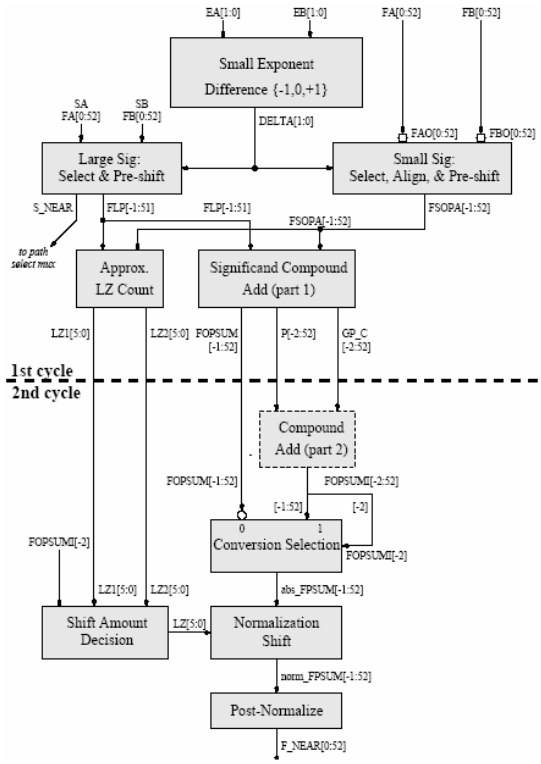
Floating-Point Adders/Subtractors

- Εναλλακτικός αλγόριθμος πρόσθεσης, με χρήση 2 μονοπατιών και 2 επιπέδων pipelining.
- Το αποτέλεσμα επιλέγεται ανάμεσα στα αποτελέσματα των 2 μονοπατιών, με βάση το σήμα IS_R.



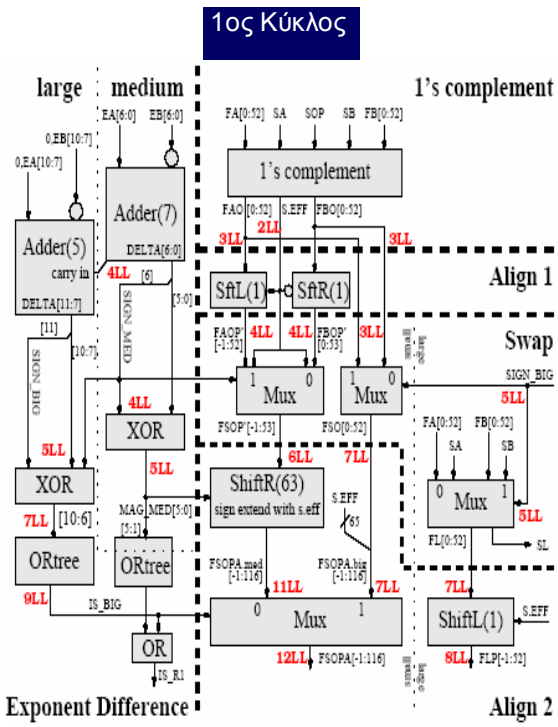
Floating-Point Adders/Subtractors

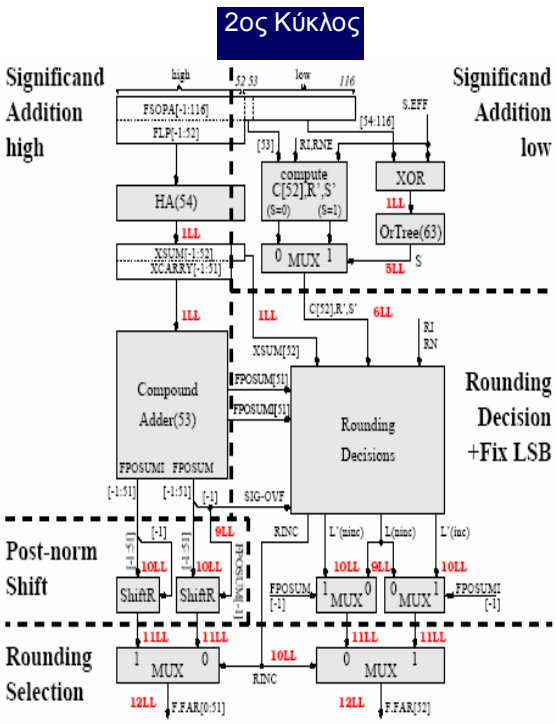




Floating-Point Adders/Subtractors

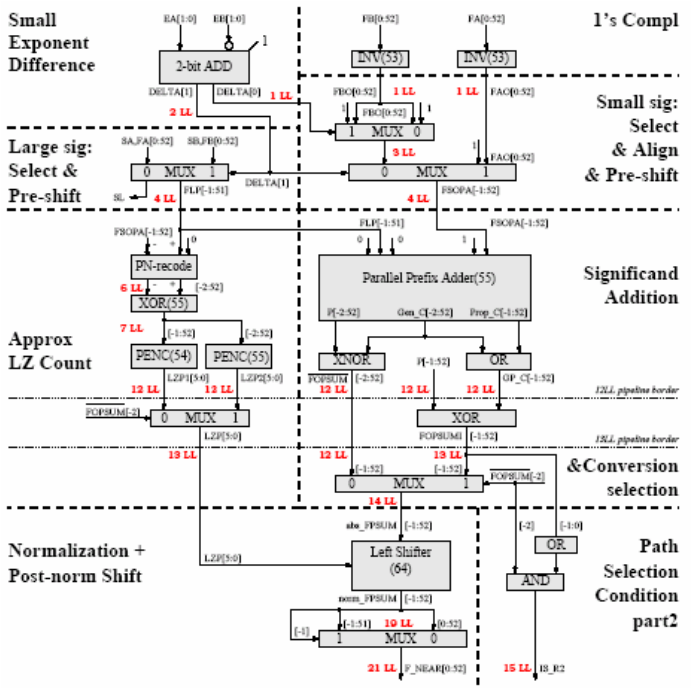
Μονοπάτι R





Floating-Point Adders/Subtractors

Μονοπάτι N

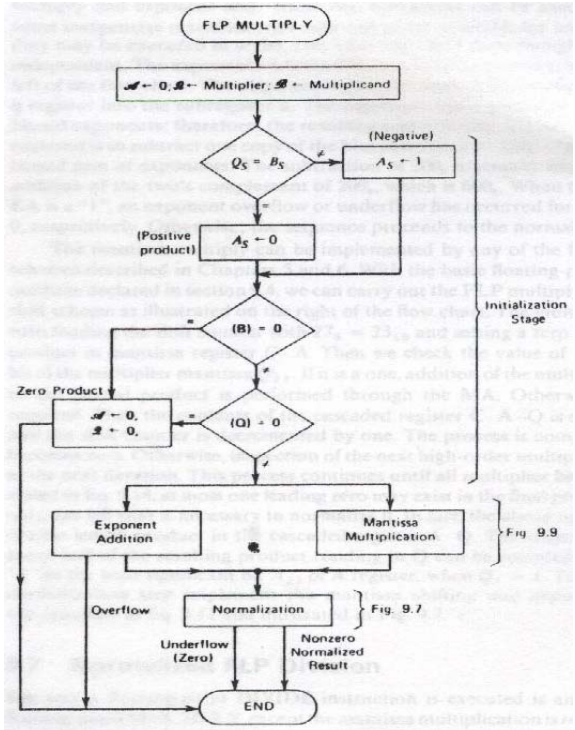


Floating- Point Numbers

Multiplication

Floating-Point Multipliers

- Ο πολ/σμός εκτελείται πολ/ντας την μάντισσα των τελεστών ενώ ταυτόχρονα προσθέτονται οι εκθέτες
- Είναι πιθανόν να έχουμε overflow/underflow κατά την πρόσθεση εκθετών με το ίδιο πρόσημο
- Υπάρχουν 4 βασικοί υπολογισμοί που έχουν σχέση με το πολ/σμό. Αυτοί οι 4 υπολογισμοί μπορούν να εκτελεστούν σε 3 στάδια όπως φαίνεται στο διπλανό σχήμα:

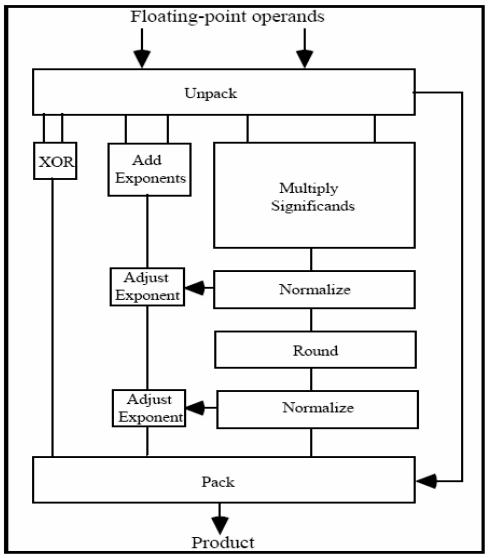


Floating-Point Multipliers

- Στο αρχικό στάδιο ελέγχουμε για μηδενικούς τελεστές και ορίζουμε το πρόσημο του αποτελέσματος.
- Στη συνέχεια η πρόσθεση των εκθετών και ο πολ/σμός της μάντισσα μπορούν να εκτελεστούν παράλληλα αρκεί να έχουν ήδη συγχρονιστεί.
- Ο πολ/σμός μπορεί να εκτελεστεί με οποιοδήποτε σύστημα σταθερής υποδιαστολής επιθυμούμε.

Floating-Point Multipliers

- Αλγόριθμος:
  (± s1*b^e1) * (±s2 * b^e2) = ± (s1 * s2) * b^(e1+e2)
- Block διάγραμμα ενός Floating-Point Πολ/στη.



Floating-Point Multipliers

- Πολλοί πολλαπλασιαστές παράγουν πρώτα το λιγότερο σημαντικό μέρος του αποτελέσματος.
- Η ανάγκη για κανονικοποίηση γίνεται γνωστή στο τέλος ή κοντά σε αυτό.
- Οπότε, η στρογγυλοποίηση μπορεί να περιληφθεί στη δημιουργία του πάνω μισού, παράγοντας δυο εκδόσεις από αυτά τα bits.

Floating - Point Numbers

Division

Floating-Point Dividers

- Η διαίρεση λειτουργεί παρόμοια με το πολ/σμό ενώ διαφέρουν στα εξής σημεία:

  - Η διαίρεση εκτελείται διαιρώντας την μάντισσα του διαιρετέου με την μάντισσα του διαιρέτη

  - Οι εκθέτες αφαιρούνται μεταξύ τους .

  - Είναι πιθανόν να έχουμε overflow ή underflow κατά την αφαίρεση εκθετών με διαφορετικό πρόσημο.

Floating-Point Dividers

- Υπάρχουν 4 βασικοί υπολογισμοί σχετικοί με την διαίρεση floating-point αριθμών:
  - Initialization
  - Mantissa Alignment
  - Exponent Subtraction
  - Mantissa Division

Floating-Point Dividers

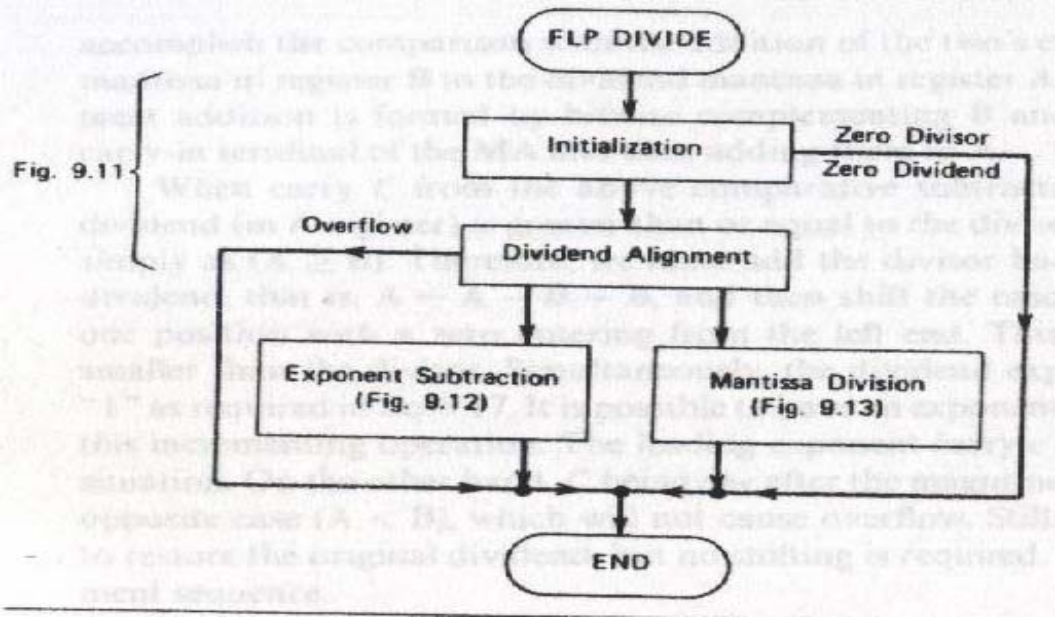
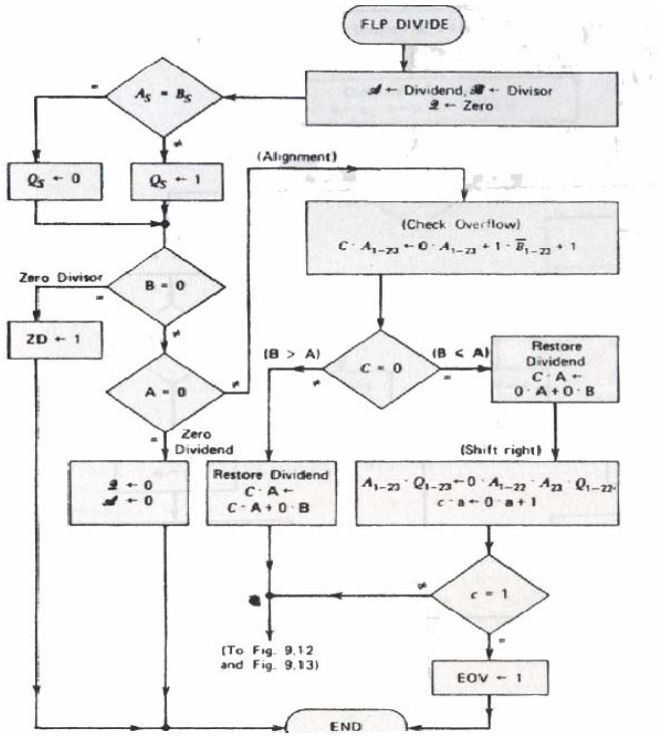
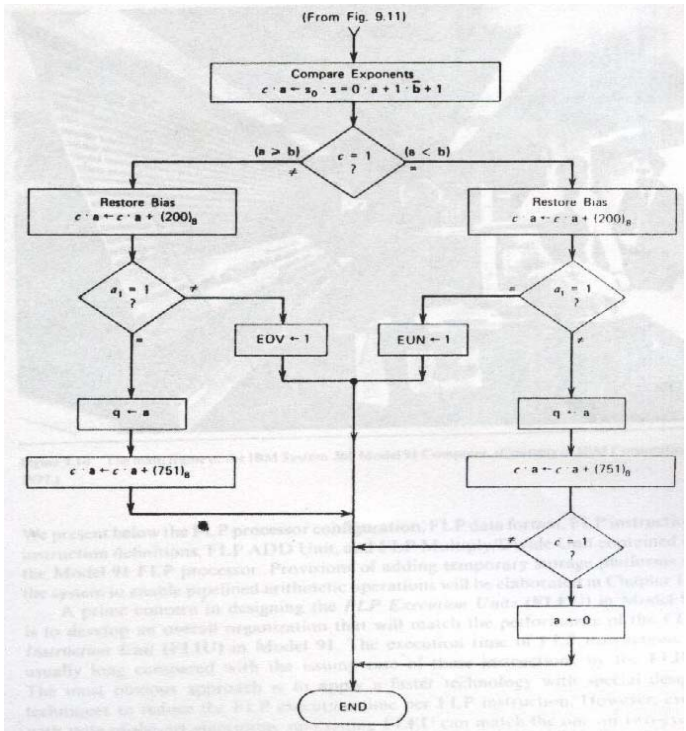


Fig. 9.11

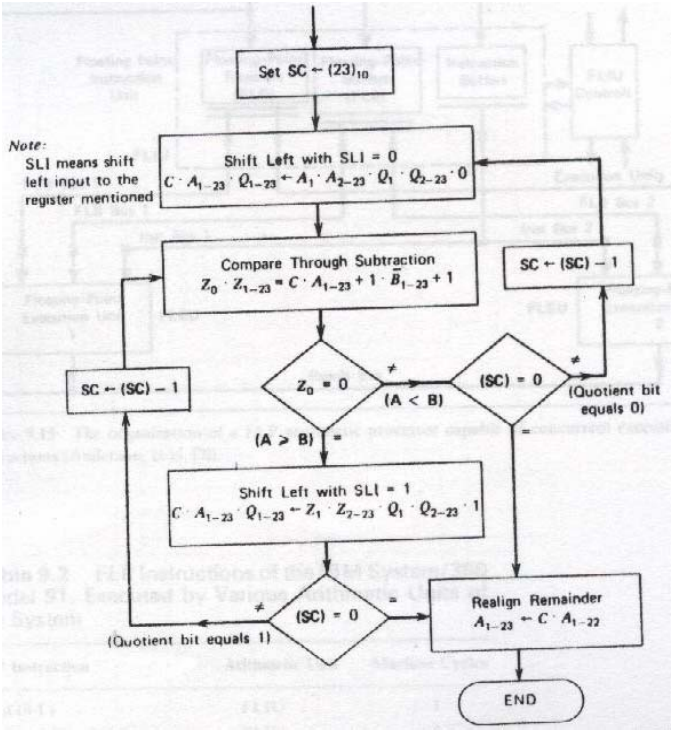
Floating-Point Dividers



Floating-Point Dividers

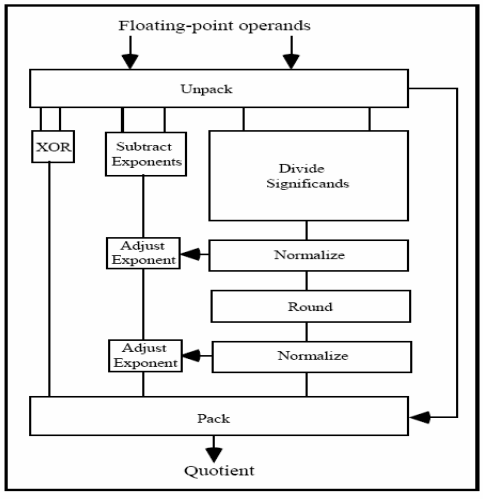


Floating-Point Dividers



Floating-Point Dividers

- Αλγόριθμος:
  (± *s*1 * *b*^*e*1) / (± *s*2 * *b*^*e*2) = ± (*s*1/*s*2) * *b*^(*e*1–*e*2)
- Block διάγραμμα ενός Floating-Point Διαιρέτη.

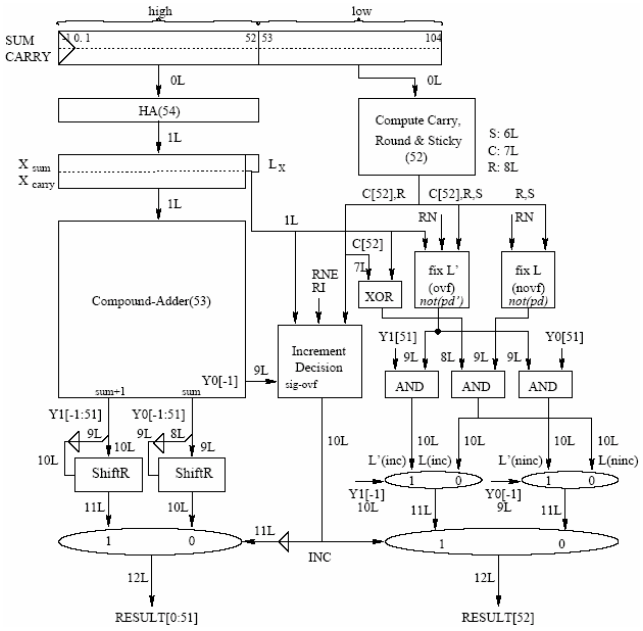


Floating - Point Numbers

Rounding

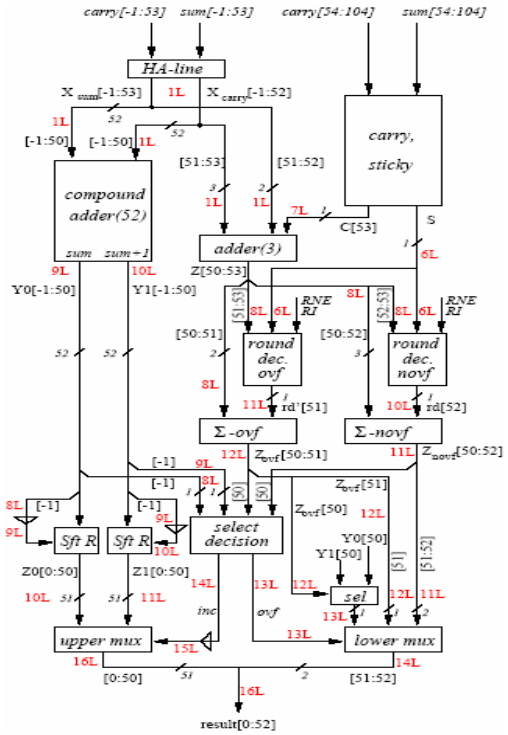
Floating-Point Rounding

Αλγόριθμος στρογγυλοποίησης ES



Floating-Point Rounding

Αλγόριθμος Στρογγυλοποίησης YZ



Floating-Point Rounding

Αλγόριθμος στρογγυλοποίησης