Note Title 07-09-2012

Sept-7

Floating Point Addition, Subtraction
Multiplication, Division

Addition:

$x_1 + x_2 \quad (x_1 \geq x_2)$

Ex:1 $1.11 + 1.00$

$$\begin{array}{r} 1.11 \\ +\ 1.00 \\ \hline 10.11 \end{array}$$

$= (1.011 \times 2)$

Ex:2 $1.11 + 1.00 \times 2^{-2}$

Ex:3 $1.11 + 1.10 \times 2^{-2}$

$$\begin{array}{r} 1.11 \\ +\ 0.011 \\ \hline 10.001 \end{array}$$

x_1 1.11
x_2 0.01
10.00 $= 1.000 \times 2^1$

1.0001×2 (x)
Rounding.

Steps $x_1 + x_2$ $(x_1 \geq x_2)$
(e_1) (e_2)

1). Right shift x_2 $(e_1 - e_2)$ times

2) Perform the addition

3) It is possible that there is a carry-out
If there is a carry out
a) Right shift result
b) Increment exponent

4) Round the result

5) Possible : Step 4 might have a carry out

If it is so,

a) Shift (right), increment exponent

Example (3) once again

Intermediate result : $1.\overline{00}\overset{\times}{0}1$

IEEE 754 format has four rounding modes.

- $+\infty$
- $-\infty$
- 0
- even

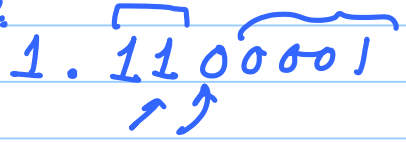
decimal example.

	∞	$-\infty$	0	even
3.4	4	3	3	3
3.5	4	3	3	4

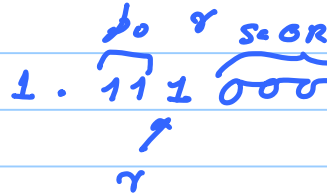
[precision = 2]

Example

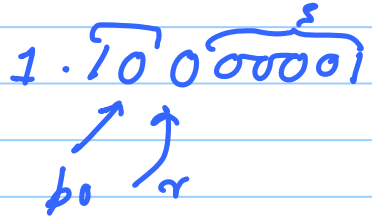
round sticky



$p_0 = 1$ $r = 0$ $s = 1$



$p_0 = 1$ $r = 1$ $s = 0$



$p_0 = 0$ $r = 0$ $s = 1$

Given the intermediate result:
compute $[p_0, r, s]$

	$x \geq 0$	$x < 0$
$+\infty$	$(r \vee s)$ [Add +1] to LSB	Truncate
$-\infty$	Truncate	$(r \vee s)$ Add(+1) to LSB
0	Truncate	Truncate
even (nearest)	[$(r \wedge p_0) \vee (r \wedge s)$ add 1 to LSB]	[$(r \wedge p_0) \vee (r \wedge s)$ add 1 to LSB]

1.11000 →

↑ r

$(r = 0, s = 0)$

1.11 1 . . . . .

x

1.11
+.01
10.00

[−3]

(−3.5) ↑ +∞

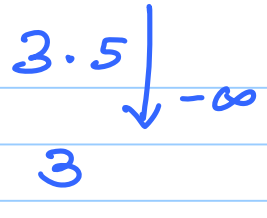
(−) 1.11 1 - -X- -

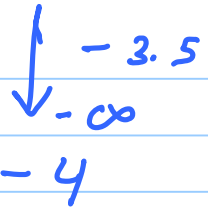
normal | discarded

[1. — - - . .] [ ]

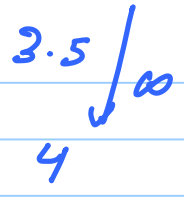
+1

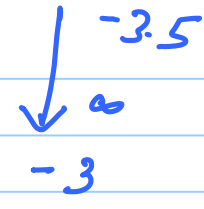
Rounding towards $[-\infty]$



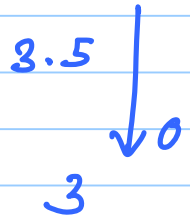


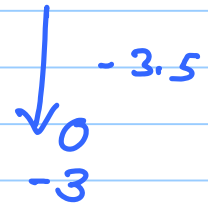
Rounding towards $[+\infty]$

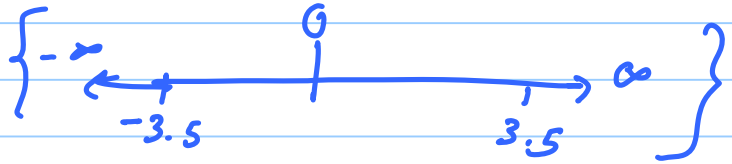




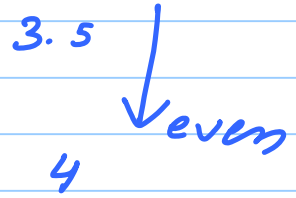
Round towards 0

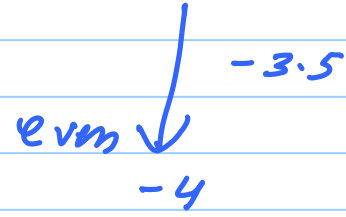






Round towards even





Binary number

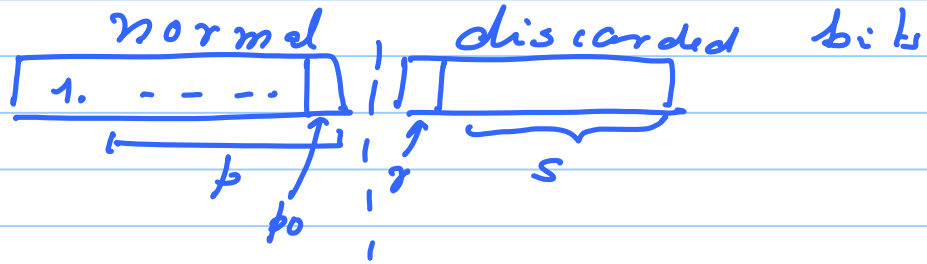
normal

1.11 0001

If the LSB [$(f_0) = 1$] consider number to be odd

If LSB [$(f_0) = 0$] consider number to be even

When are you going to add 1 to the LSB



(r AS) : discarded bits (D)

$D > 0.5 \times 2^{-p}$

Add 1 to LSB

50-50 case.

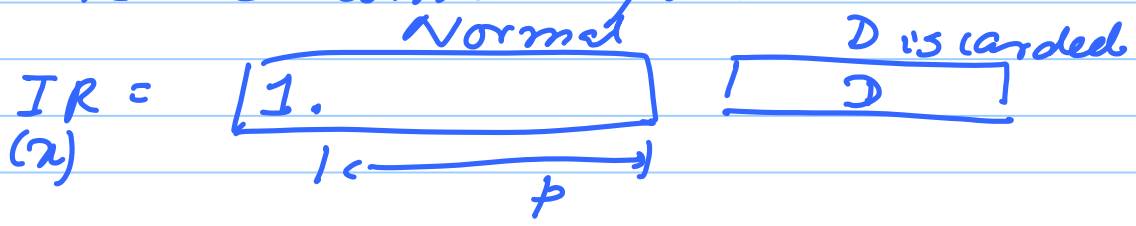
$D = 0.5 \times 2^{-t}$

$r = 1$

$(r \wedge p_0)$ add 1 to the LSB

Big Picture

It is possible that the intermediate result has more than p bits after the decimal point.



Option 1:

→ Truncate discarded bits,

Option 2:

" " ", add 1 to the

LSB of IR

	$x > 0$	$x < 0$
$+\infty$	$[D > 0]$ add 1	–
$-\infty$	–	$[D > 0]$ add 1
0	–	–

default

even

- $(D > 0.5 \times 2^{-1})$ add 1
- or
- $(D = 0.5 \times 2^{-1})$ & $(b_0 = 1)$ add 1

} Same

$$\begin{array}{r} 1.111 \\ +0.01 \\ \hline 10.000 \end{array}$$

Subtraction.

x_1
$-\ x_2$

1) Align the numbers.
2) Take a 2s complement of x_2
:
:
!
:

(difference: right shift of a (-)ve number (shift in 1s))

Multiplication.

$$1.x \times 2^{e_1} \times 1.y \times 2^{e_2}$$
$$(1.x \times 1.y) \times 2^{e_1+e_2}$$

1) Compute the sign of the result
2) Add the exponents.

3) Perform multiplication.

4) Adjust and round.

5) Adjust again