Operations on Floating Point Numbers

Prof. Daniele Gorla

Multiplication

$$< s_1 , e_1 , m_1 > \times < s_2 , e_2 , m_2 > = < s , e , m >$$

where

1. $s = \begin{cases} 0 & \text{if } s_1 = s_2 \\ 1 & \text{otherwise} \end{cases}$

2. m and e are the normalized mantissa and exponent of

$$1,m_1 \times 1,m_2 \times b^{e_1+e_2-B}$$

where B is the bias

OBS.: pay attention to the exponent overflow!!

2

Example

Let

A = < 0 , 10000 , 1101000000 >

B = < 1 , 01101 , 1010000000 >

Compute: A × B

Let's first convert A and B in base 10 (to check correctness):

A ➔ $11{,}101_2 = (2 + 1 + ½ + ⅛)_{10} = (3 + 0{,}5 + 0{,}125)_{10} = 3{,}625_{10}$

B ➔ $-0{,}01101_2 = -(2^{-2} + 2^{-3} + 2^{-5})_{10} = -(0{,}25+0{,}125+0{,}03125)_{10} = -0{,}40625_{10}$

3

A = < 0 , 10000 , 1101000000 > ($3{,}625_{10}$)

B = < 1 , 01101 , 1010000000 > ($-0{,}40625_{10}$)

A × B:

Sum of the exponents:

10000 +
01101 =

11101 –
01111 =

01110

Product of the mantisse:

1,1101 ×
1,101 =

11101
00000-
11101-
11101-

10,1111001

Normalized Result:

R = < 1 , 01111 , 0111100100 >

Check:

R ➔ $-1{,}01111001_2$

$= -(1+2^{-2}+2^{-3}+2^{-4}+2^{-5}+2^{-8})_{10}$

$= -1{,}47265625_{10}$

$= (3{,}625 \times -0{,}40625)_{10}$

4

Division

$$< s_1, e_1, m_1 > \div < s_2, e_2, m_2 > = < s, e, m >$$

where

1. $s = \begin{cases} 0 & \text{if } s_1 = s_2 \\ 1 & \text{otherwise} \end{cases}$

2. m and e are the normalized mantissa and exponent of

$$(1,m_1 \div 1,m_2) \times b^{e_1 - e_2 + B}$$

We shall **not** see it in detail because the mantissa division is not easy…

Sum (1)

$$< s_1, e_1, m_1 > + < s_2, e_2, m_2 > = < s, e, m >$$

1. If $e_1 = e_2$
   - $s = \begin{cases} s_1 & \text{if } m_1 \geq m_2 \\ s_2 & \text{otherwise} \end{cases}$
   - m and e are the normalization of m' and e' defined as:

     $e' = e_1 \ (= e_2)$

     $m' = \begin{cases} 1,m_1 + 1,m_2 & \text{if } s_1 = s_2 \\ 1,m_1 - 1,m_2 & \text{if } s_1 \neq s_2 \text{ and } m_1 \geq m_2 \\ 1,m_2 - 1,m_1 & \text{otherwise} \end{cases}$

Sum (2)

$$< s_1, e_1, m_1 > + < s_2, e_2, m_2 > = < s, e, m >$$

2. If $e_1 < e_2$
   - right shift $1, m_1$ of $e_2 – e_1$ positions (by adding 0's at left)
     - OBS.1: after this step, the first operand is no more normalized!
       → we write its exponent as $0 \cdots 0$ (to remember that its I.P. is 0)
     - OBS.2: we can loose digits at the end of m_1
       (potentially, m_1 could become 0!!)
   - in this way, the first operand becomes $< s_1, 0...0, m'_1 >$
   - $s = s_2$
   - m and e are the normalization of m' and e' defined as:

$$e' = e_2 \text{ and } m' = \begin{cases} 1, m_2 + 0, m'_1 & \text{if } s_1 = s_2 \\ 1, m_2 - 0, m'_1 & \text{otherwise} \end{cases}$$

Sum (3)

$$< s_1, e_1, m_1 > + < s_2, e_2, m_2 > = < s, e, m >$$

3. If $e_1 > e_2$
   - like in point (2), but right shift the second operand (to obtain the first exponent)

Subtraction

SAPIENZA Università di Roma Dipartimento di Informatica

Trivially reducible to sum, since:

$$< s_1 , e_1 , m_1 > - < s_2 , e_2 , m_2 > =$$

$$= < s_1 , e_1 , m_1 > + < \overline{s_2} , e_2 , m_2 >$$

where $\overline{s}$ denotes 1 if $s = 0$, and 0, if $s = 1$.

9

Example (continued)

SAPIENZA Università di Roma Dipartimento di Informatica

A = < 0 , 10000 , 1101000000 > ($3,625_{10}$)
B = < 1 , 01101 , 1010000000 > ($-0,40625_{10}$)

Compute A + B

Transform the lower exponent (01101) to the bigger (10000)
To this aim, the mantissa of the second operand must be right shifted by
10000 – 01101 = 11 (i.e., 3)
positions, to obtain
B' = < 1 , 00000 , 0011010000 >

A and B' have different signs:

1,110100000 –
0,001101000 =

1,100111000

Normalized Result: R = < 0 , 10000 , 1001110000 >

Check: R $\rightarrow$ $11,00111_2$ $\rightarrow$ $3,21875_{10}$

10

SAPIENZA Università di Roma Dipartimento di Informatica

A = < 0 , 10000 , 1101000000 > ($3,625_{10}$)
B = < 1 , 01101 , 1010000000 > ($-0,40625_{10}$)

Compute A – B

We still consider the non-normaized B' = < 1 , 00000 , 0011010000 > of the previous slide (also here the numbers must have the same exponent)

Since we now have a subtraction, we compute A + –B'. Now, A and –B' have the same sign:

1,1101000000 +
0,0011010000 =

10,0000010000

The result must be normalized:
R = < 0 , 10001 , 0000001000 >

Check: R $\rightarrow$ $100,00001_2$ $\rightarrow$ $4,03125_{10}$

11

SAPIENZA Università di Roma Dipartimento di Informatica

A = < 0 , 10000 , 1101000000 > ($3,625_{10}$)
B = < 1 , 01101 , 1010000000 > ($-0,40625_{10}$)

Compute B – A

We still consider B'

Also here, since we have a subtraction and operands with different signs, we sum the mantissa

The final result will be negative, since we're summing negative numbers

Result: < 1 , 10001 , 0000001000 >

12

Special cases:

- $(128 + 0{,}0625)_{10} = (2^7)_{10} + (2^{-4})_{10} \rightarrow 10000000_2 + 0{,}0001_2 \rightarrow$

  $\rightarrow < 0 , 10110 , 0000000000 > + < 0 , 01011 , 0000000000 > =$
  $= < 0 , 10110 , 0000000000 > + < 0 , 00000 , 0000000000 > =$
  $= < 0 , 10110 , 0000000000 >$

- $(256 \times 256)_{10} \rightarrow$

  $\rightarrow < 0 , 10111 , 0000000000 > \times < 0 , 10111 , 0000000000 >$

  10111 + 10111 - 01111 = 11111 → INFINITY → *exponent overflow!*