BUBT
Committed to Academic Excellence

BANGLADESH UNIVERSITY OF BUSINESS AND TECHNOLOGY

Assignment on

Floating Point Representation, Addition and Multiplication Algorithm

Course Code: CSE 215

Course Title: Computer Architecture

Assignment no: 3

Submitted to:

Name: Md. Ashiqur Rahman

Lecturer,

Dept. of CSE

at Bangladesh University of Business and Technology.

Submitted by:

Name: Syeda Nowshin Ibnat

ID: 17183103020

Intake: 39

Section: 01

Program: B.Sc. in CSE

Date of Submission: 17.05.2020

Assignment - 3

Q1: Show the IEEE 754 binary representation of the number -0.625_{ten} in single and double precision.

Soln: The number -0.625_{ten} is also $-\frac{5}{8}_{ten}$ or $-\frac{5}{2^3}_{ten}$

It is also represented by the binary fraction:

$$-0.101_{two}$$

In scientific notation, the value is:

$$-0.101_{two} \times 2^{0}$$

and in normalized scientific notation, it is:

$$-1.01_{two} \times 2^{-1}$$

The general representation for a single precision number is

$$(-1)^{S} \times (1 + \text{Significand}) \times 2^{(\text{Exponent} - 127)}$$

and so when we add the bias 127 to the exponent of $-1.01_{two} \times 2^{-1}$, the result

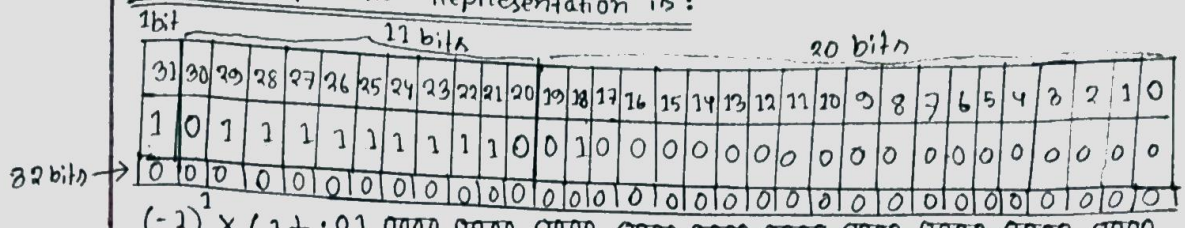
$$(-1)^{1} \times (1 + .01\ 0000\ 0000\ 0000\ 0000\ 00000)_{two} \times 2^{(126-127)}$$

The single precision representation is:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1 bit | 8 bits | 23 bits

The double precision representation is:



$(-1)^1 \times (1 + .01\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 00)_{two} \times 2^{(1022-1023)}$

Ans:

Q2: Add two numbers 0.5_{ten} and 0.25_{ten} in binary using floating point addition algorithm.

Soln: $0.5 = 0.10 = 0.10 \times 2^0 = 1.0 \times 2^{-1}$

$0.25 = 0.010 = 0.010 \times 2^0 = 0.10 \times 2^{-1} = 1.00 \times 2^{-2}$

Step-1: (change the smaller number)

$1.00 \times 2^{-2} = 0.10 \times 2^{-1}$

Step-2: (Addition of the significands)

$$\begin{array}{r} 1.00 \\ +\ 0.10 \\ \hline 1.10 \end{array} \longrightarrow 1.10 \times 2^{-1}$$

Step-3: (Find the normalized form & check if overflow and underflow)

The sum is already normalized. And there is no overflow or underflow.

Step-4: (Round the number)

1.10×2^{-1}

converting the result to decimal:

$1.10 \times 2^{-1} = 0.110$

$= 0.75_{ten}$

This sum is what we would expect from adding 0.5_{ten} and 0.25_{ten}.

Ans:

Q3: Multiply two numbers. 1.5_{ten} and 1.25_{ten} in binary using floating point multiplication algorithm.

Soln:

$1.5 = 01.10 = 01.10 \times 2^0 = 1.10 \times 2^0$

$1.25 = 01.010 = 01.010 \times 2^0 = 1.010 \times 2^0$

Step-1: (Find the exponent of the product)

New exponent $= 0+0 = 0$

considering bias,

New exponent $= 0+127 = 127$

Step-2: (Multiplication of the significands)

```
   1.10
 x 1.01
 ------
   110
  000
 110
 ------
 11110
```

∴ The product is $= 1.1110 \times 2^0$

Step-3: (Normalize the product)

The product is already normalized. And here is no overflow or underflow.

Step-4: (Round the number)

$$11.110 \times 2^{-1}$$

Step-5: (Put the sign of the product)

$$+11.110 \times 2^{-1}$$

converting to decimal to check our results:

$$11.110 \times 2^{-1} = 1.1110$$
$$= 1.875_{ten}.$$

Ans: