

# MPI: Microprocessors and Interfacing

## Academic Year: 2022 - 23

Dr. Praveen Kumar Alapati

[praveenkumar.alapati@mahindrauniversity.edu.in](mailto:praveenkumar.alapati@mahindrauniversity.edu.in)

Department of Computer Science and Engineering

Ecole Centrale School of Engineering



**Mahindra<sup>TM</sup>**  
**University**

Global Thinkers. Engaged Leaders.

Assume that you have a processor (P) and it supports **ADD** (Addition), **SUB** (Subtraction), **INC** (Increment), **DEC**, **CMP** (Compare), **Jcc** (Jump if a condition is met), **JMP**, **XOR**, and **MOV** operations. Assume that A and B are positive integers and  $A > B$ .

1. Write an assembly language program to compute the  $(A^2 - B^2)$  on P.
2. Write an assembly language program to compute the quotient of  $\frac{(A^3 - B^3)}{(A+B)}$  on P.
3. Compute A choose B (i.e.,  $\binom{A}{B}$ ) on P

You are given with  $n$  positive Integers. Write functions to compute the following using **assembly language instructions**.

1. To find the number of even numbers
2. To find the number of odd numbers
3. To find the number of prime numbers
4. To find the GCD of  $n$  numbers
5. To find the LCM of  $n$  numbers

Write assembly language programs to compute the following( use NASM assembler). Assume that all elements are integers.

1. Find the **maximum** of  $n$  numbers.
2. Find the **minimum** of  $n$  numbers.
3. Assume that you are given with a list of  $n$  elements then find the **mode**.
4. Assume that you are given with a list of  $n$  elements then find the **median**.
5. Assume that you are given with a list of  $n$  elements and **key**, search the **key** using **linear search**. If the required **key** is found, return its position. Otherwise, return **-1**.

# Lab1 - Due date: August 31, 2023.

1. Give the configuration of your laptop.
2. Give the configuration of a system in IT-Lab.
3. Develop C-Programs for the following problem statements:
  - 3.1. Print the internal representation of data stored in primary data types: **int, float, and double**.
  - 3.2. Perform addition and multiplication of two 32-bit numbers (Please remember that the input is a 32-bit binary number). The numbers can either be integers or real numbers.

# Submission Guide Lines

**Max. team size is 6.**

Mail-ID: cs3106.mpi@gmail.com

Sub:TEAM\_NUM\_LAB\_NUM

Attach.Name and Type: (Sub.).zip

Late Submission:50%.

Write a readme file to understand your solutions.

# Number Systems

## Representation of Integer Numbers

- Signed Magnitude Representation

- 1's Complement Representation

- 2's Complement Representation

## Representation of Real Numbers

- Fixed Point Representation

- Floating Point Representation

**Resolution is difference between two successive numbers.**

# Representation of Integer Numbers

Let  $A = a_{n-1}a_{n-2}a_{n-3}\dots a_0$  is an n-bit binary number  
if A is an **unsigned integer**, then value of A is :  $\sum_{i=0}^{n-1}(2^i \times a_i)$ .  
if A is a **signed integer**:

Signed Magnitude Representation:

$$A = \sum_{i=0}^{n-2}(2^i \times a_i), \text{ if } a_{n-1} = 0$$

$$A = -\sum_{i=0}^{n-2}(2^i \times a_i), \text{ if } a_{n-1} = 1$$

$$1\text{'s Complement Rep.: } A = -(2^{n-1} - 1) \times a_{n-1} + \sum_{i=0}^{n-2}(2^i \times a_i)$$

$$2\text{'s Complement Rep.: } A = -2^{n-1} \times a_{n-1} + \sum_{i=0}^{n-2}(2^i \times a_i)$$

**Resolution: 1**



# Range of Numbers

Let  $A = a_{n-1}a_{n-2}a_{n-3}\dots a_0$  is an  $n$  bit binary number

if  $A$  is an **unsigned integer**, then range of  $A$  is : 0 to  $(2^n - 1)$ .

if  $A$  is a **signed integer**:

Signed Magnitude Rep., range of  $A$  is :  $-(2^{n-1} - 1)$  to  $(2^{n-1} - 1)$ .

1's Complement Rep., range of  $A$  is :  $-(2^{n-1} - 1)$  to  $(2^{n-1} - 1)$ .

2's Complement Rep., range of  $A$  is :  $-2^{n-1}$  to  $(2^{n-1} - 1)$ .

# Expansion of Bit Length

Add additional bit positions to the left and fill in with value of the sign bit.

Let  $A = 1\ 0\ 1\ 0$  is a 4-bit binary number,

Representation of A using 8-bits (i.e. B):  $1\ 1\ 1\ 1\ 1\ 0\ 1\ 0$ .

is  $A=B$  ?

In 2's Complement Rep.: **Yes**.

In 1's Complement Rep.: **Yes**.

In Signed Magnitude Rep.: **No**.

# Expansion of Bit Length

Add additional bit positions to the left and fill in with value of the sign bit.

Let  $A = 1\ 0\ 1\ 0$  is a 4-bit binary number,

Representation of A using 8-bits (i.e. B):  $1\ 1\ 1\ 1\ 1\ 0\ 1\ 0$ .

is  $A=B$  ?

In 2's Complement Rep.: **Yes**.

In 1's Complement Rep.: **Yes**.

In Signed Magnitude Rep.: **No**.

# Expansion of Bit Length

Add additional bit positions to the left and fill in with value of the sign bit.

Let  $A = 1\ 0\ 1\ 0$  is a 4-bit binary number,

Representation of A using 8-bits (i.e. B):  $1\ 1\ 1\ 1\ 1\ 0\ 1\ 0$ .

is  $A=B$  ?

In 2's Complement Rep.: **Yes**.

In 1's Complement Rep.: **Yes**.

In Signed Magnitude Rep.: **No**.

# Expansion of Bit Length

Add additional bit positions to the left and fill in with value of the sign bit.

Let  $A = 1\ 0\ 1\ 0$  is a 4-bit binary number,

Representation of A using 8-bits (i.e. B):  $1\ 1\ 1\ 1\ 1\ 0\ 1\ 0$ .

is  $A=B$  ?

In 2's Complement Rep.: **Yes**.

In 1's Complement Rep.: **Yes**.

In Signed Magnitude Rep.: **No**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.



# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

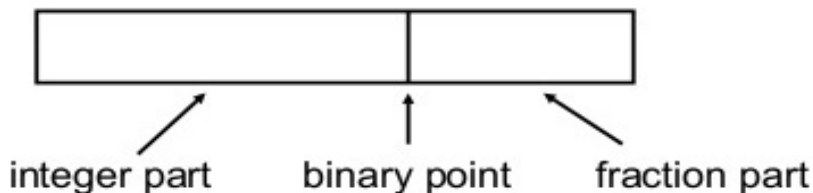
# Real Numbers

1.  $(4.5)_{10} = (100.1)_2$
2.  $(8.25)_{10} = (1000.01)_2$
3.  $(16.125)_{10} = (10000.001)_2$
4.  $(0.875)_{10} = (0.111)_2$
5.  $(4.5)_{10} = (1.001)_2 \times 2^2$
6.  $(8.25)_{10} = (1.00001)_2 \times 2^3$
7.  $(16.125)_{10} = (1.0000001)_2 \times 2^4$
8.  $(0.875)_{10} = (1.11)_2 \times 2^{-1}$

**Representations 5 to 8 are called Normalized Representations.**

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**,  
'xxxxxx' is a **Fraction/Mantissa**.

# Fixed Point (FP) Representation



Find the decimal equivalent of the following binary numbers. Assume that the binary numbers are represented using FP representation (6,2), i.e., 6 bits for integer part and 2 bits for fractional part.

$$(00101011)_2 = ?$$

$$(11111011)_2 = ?$$

Smallest +ve number that can be represented using FP (6,2) rep.: ?

Biggest +ve number that can be represented using FP (6,2) rep.: ?

# Fixed Point Arithmetic

Consider a 16-bit binary representation, in which least significant 8 bits are used for precision then give the range of values that can be represented using the 16-bit representation.

+ve Values:  $2^{-8}$  to  $2^7 - 2^{-8}$

-ve Values:  $-2^7$  to  $-2^{-8}$

Zero

Resolution is ?

Advantages of FP Arithmetic:

Easy to implement and occupies less space.

If performance is important than precision.

Once can choose a trade off between range and precision.



# Fixed Point Arithmetic

Consider a 16-bit binary representation, in which least significant 8 bits are used for precision then give the range of values that can be represented using the 16-bit representation.

+ve Values:  $2^{-8}$  to  $2^7 - 2^{-8}$

-ve Values:  $-2^7$  to  $-2^{-8}$

Zero

Resolution is ?

Advantages of FP Arithmetic:

Easy to implement and occupies less space.

If performance is important than precision.

Once can choose a trade off between range and precision.

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.

Sign	Biased Exponent	Mantissa/Fraction
------	-----------------	-------------------

Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

**Biased Exponent = True Exponent + Bias Value**

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

$$\text{Mantissa} = 001 = 001\text{0000}\ 0000\ 0000\ 0000\ 0000$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\text{00000000000000000000}$$

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.

Sign	Biased Exponent	Mantissa/Fraction
------	-----------------	-------------------

Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

$$\text{Mantissa} = 001 = 001\text{0000}\ 0000\ 0000\ 0000\ 0000$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\text{00000000000000000000}$$

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.

Sign	Biased Exponent	Mantissa/Fraction
------	-----------------	-------------------

Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

$$\text{Mantissa} = 001 = 001\text{0000}\ 0000\ 0000\ 0000\ 0000$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\text{00000000000000000000}$$

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.

Sign	Biased Exponent	Mantissa/Fraction
------	-----------------	-------------------

Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

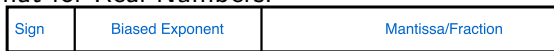
$$\text{Mantissa} = 001 = 001\text{0000}\ 0000\ 0000\ 0000\ 0000$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\text{00000000000000000000}$$

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.



Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

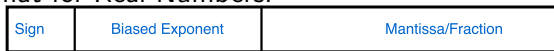
$$\text{Mantissa} = 001 = 001\ 0000\ 0000\ 0000\ 0000\ 0000$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\ 00000000000000000000$$

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.



Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

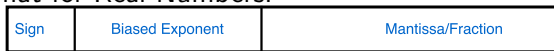
$$\text{Mantissa} = 001 = 001\textcolor{red}{0000\ 0000\ 0000\ 0000\ 0000}$$

$$\text{Sign} = +ve = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\textcolor{red}{00000000000000000000}$$

# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.



Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

$$\text{Mantissa} = 001 = 001\textcolor{red}{0000\ 0000\ 0000\ 0000\ 0000}$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\textcolor{red}{00000000000000000000}$$



# IEEE 754 Representation of Real Numbers

IEEE 754 format for Real Numbers.

Sign	Biased Exponent	Mantissa/Fraction
------	-----------------	-------------------

Single Precision N=32	1 bit	8 bits	23 bits	Bias Value: +127
Double Precision N=64	1 bit	11 bits	52 bits	Bias Value : +1023

Biased Exponent = True Exponent + Bias Value

Rep. of  $(4.5)_{10}$  using Single Precision

$$(4.5)_{10} = (100.1)_2 = (1.001)_2 \times 2^2$$

Normalized Rep.:  $(\pm 1.xxxxxx)_2 \times 2^E$ , Where 'E' is a **True Exponent**, 'xxxxxx' is a **Fraction/Mantissa**.

$$\text{Biased Exponent} = 2 + 127 = 129 = 1000\ 0001$$

$$\text{Mantissa} = 001 = 001\textcolor{red}{0000\ 0000\ 0000\ 0000\ 0000}$$

$$\text{Sign} = + \text{ve} = 0$$

$$(4.5)_{10} = (1.001)_2 \times 2^2 = 0\ 10000001\ 001\textcolor{red}{00000000000000000000}$$

## IEEE 754

Biased Exponent = True Exponent + Bias Value,  
where  $1 \leq \text{Biased Exponent} \leq (2^{\text{Length of Biased Exponent}} - 2)$ .

Single Precision (N=32),  $1 \leq \text{Biased Exponent} \leq 254$ .

Biased Exponent = 0,

Mantissa =  $\pm 0$ , then Value is  $\pm 0$ .

Mantissa  $\neq 0$ , then Value is **not a normalized number**.

Biased Exponent = 255,

Mantissa =  $\pm 0$ , then Value is  $\pm \infty$ .

Mantissa  $\neq 0$ , then Value is **NAN**.

Range of positive values:  $[1.0 \times 2^{-126}, (2 - 2^{-23}) \times 2^{127}]$

Range of negative values:  $[-(2 - 2^{-23}) \times 2^{127}, -1.0 \times 2^{-126}]$

Single Precision Number Resolution:  $2^{-23} \times 2^{\text{True Exponent}}$

# BCD Representation

BCD: Binary Coded Decimal

It uses a 4-bit binary number to represent each decimal digit.

Decimal Number	BCD Rep.
0	0000
1	0001
2	0010
3	0011
9	1001
10	0001 0000
25	0010 0101
99	1001 1001

Table 1: BCD equivalent of a decimal number.

All the best 😊