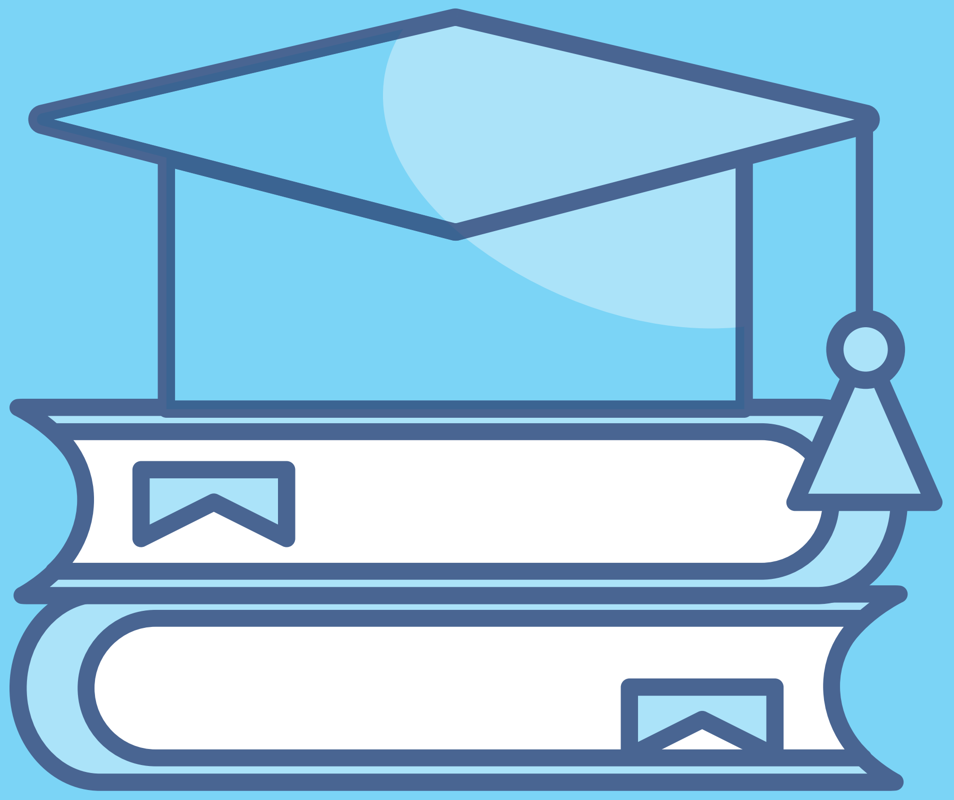
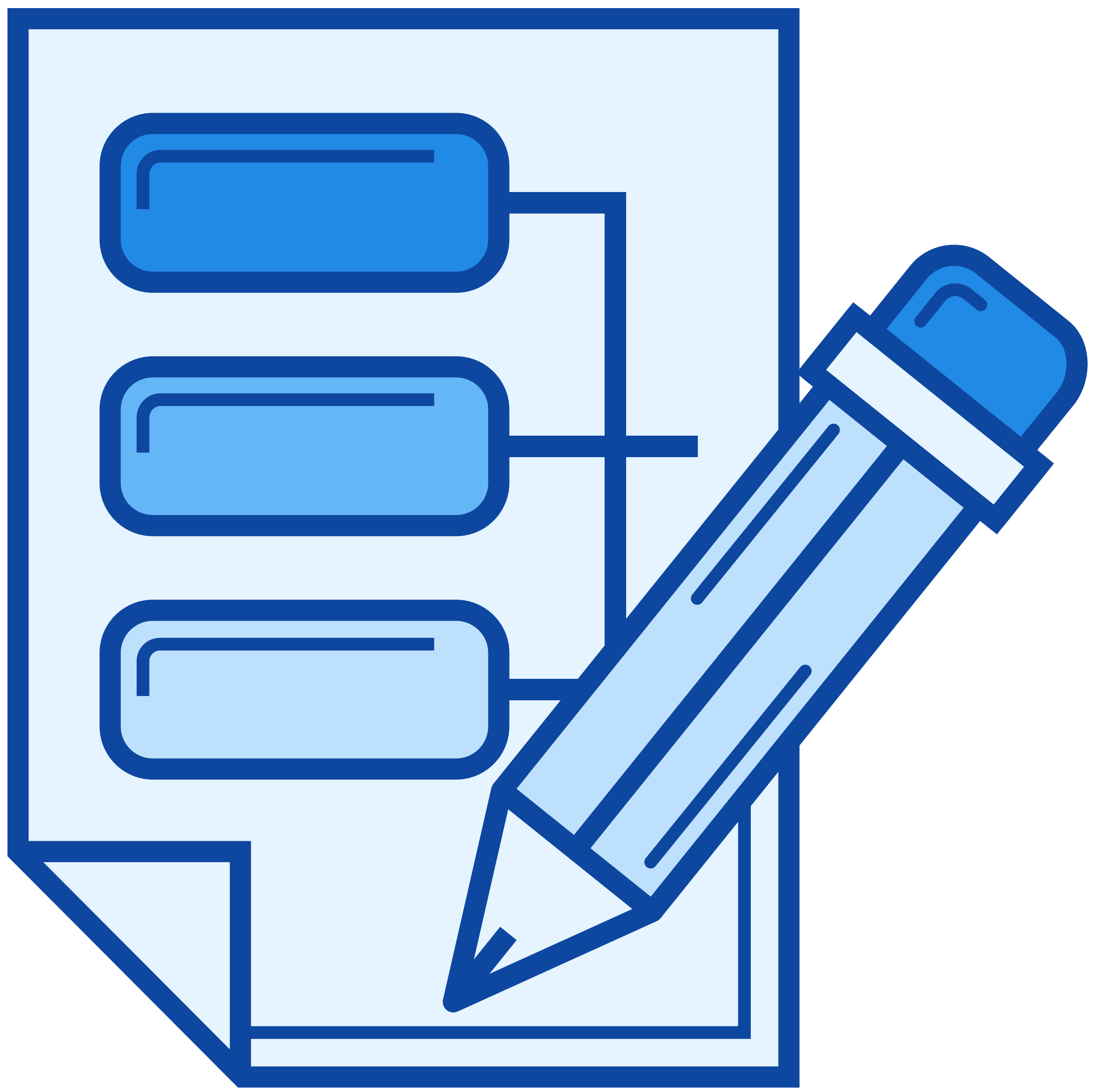
IEEE 754 NUMBER SYSTEM

COMPUTER ARCHITECTURE (EC502)
SOUVIK GHOSH 13000320025
ELECTRONICS & COMMUNICATION ENGINEERING



Contents

- Inroduction
- Basic Components
- Types
- Example
- Special Values
- Special Operations



Introduction

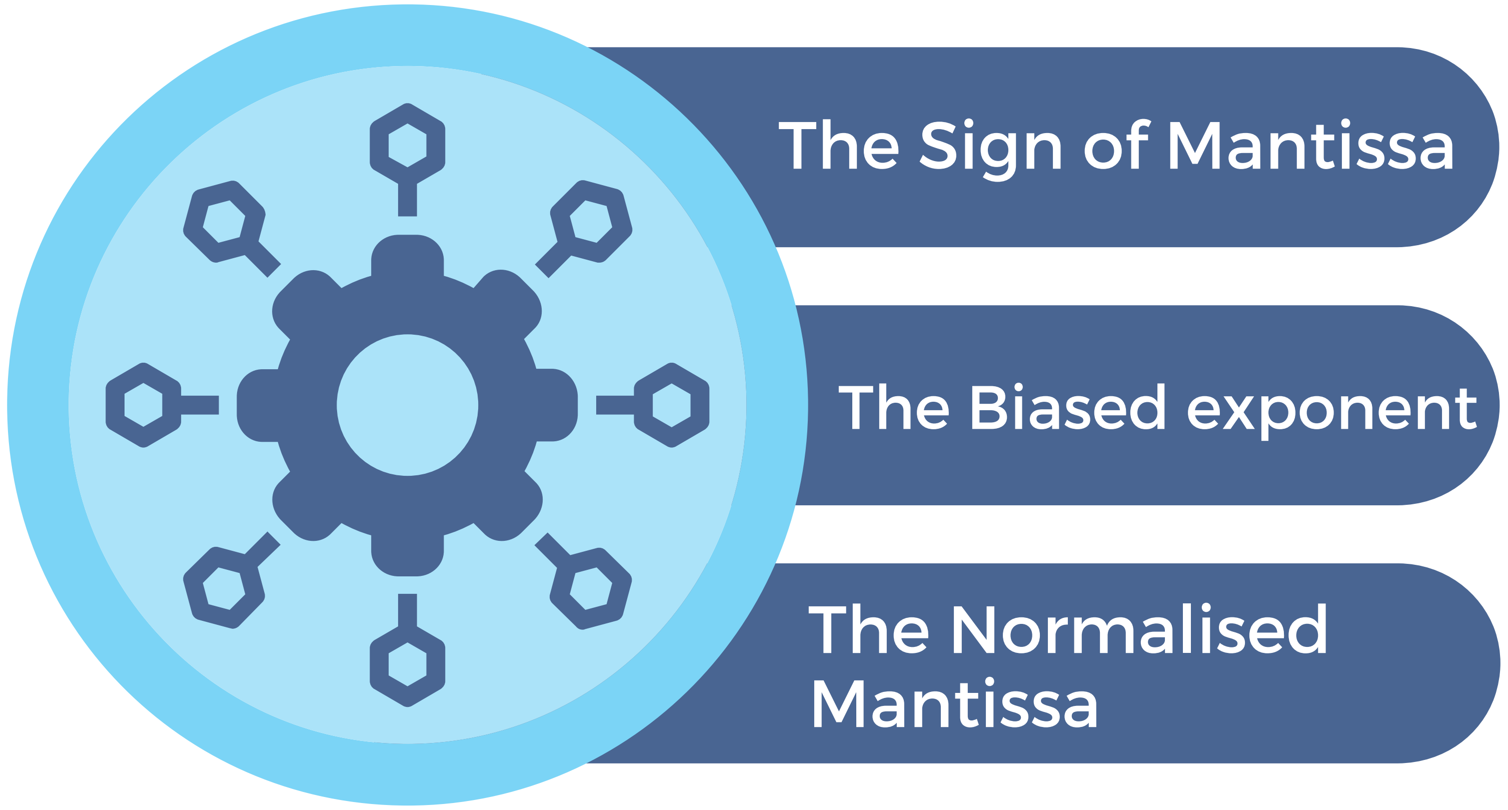
- There were many problems in the conventional representation of floating-point notation like we could not express 0(zero), infinity number. To solve this, scientists have given a standard representation and named it as **IEEE Floating point representation**.
- The IEEE Standard for Floating-Point Arithmetic **(IEEE 754)** is a technical standard for floating-point computation which was established in 1985 by the **Institute of Electrical and Electronics Engineers (IEEE).** The standard addressed many problems found in the diverse floating point implementations that made them difficult to use reliably and reduced their portability. IEEE Standard 754 floating point is the most common representation today for real numbers on computers, including Intel-based PCs, Macs, and most Unix platforms.

→

Basic Components

There are several ways to represent floating point numbers but IEEE 754 is the most efficient in most cases. IEEE 754 has **3 basic components**:

1. **The Sign of Mantissa** – This is as simple as the name. 0 represents a positive number while 1 represents a negative number.
2. **The Biased exponent** – The exponent field needs to represent both positive and negative exponents. A bias is added to the actual exponent in order to get the stored exponent.
3. **The Normalised Mantissa** – The mantissa is part of a number in scientific notation or a floating-point number, consisting of its significant digits. Here we have only 2 digits, i.e. O and 1. So a normalised mantissa is one with only one 1 to the left of the decimal.



Types

IEEE 754 numbers are divided into **two types** based on the three components:

1. Single Precision (32-bit)
2. Double Precision (64-bit)

TYPES	SIGN	BIASED EXPONENT	NORMALISED MANTISA	BIAS
Single Precision	1 bit (31^{st} bit)	8 bits (30-23)	23 bits (22-0)	127
Double Precision	1 bit (63^{rd} bit)	11 bits (62-52)	52 bits (51-0)	1023

Example

85.125
85 = 1010101
0.125 = 001
85.125 = 1010101.001
 =1.010101001 x 2^6
sign = 0

1. **Single precision:**
biased exponent 127+6=133
133 = 10000101
Normalised mantissa = 010101001
we will add 0's to complete the 23 bits

The IEEE 754 Single precision is:
= 0 10000101 01010100100000000000000
This can be written in hexadecimal form 42AA4000

2. **Double precision:**
biased exponent 1023+6=1029
1029 = 10000000101
Normalised mantissa = 010101001
we will add 0's to complete the 52 bits

The IEEE 754 Double precision is:
=0 10000000101 0101010010000000000000000000000000000000000
000000000
This can be written in hexadecimal form 4055480000000000

→

Special Values

IEEE has reserved some values that can ambiguity.

EXPONENT	MANTISA	VALUE
0	0	exact 0
255	0	infinity
0	not 0	demoralized
255	not 0	NAN (not a number)

*Similar for Double precision (just replacing 255 by 2049)

Special Operations

Operation	Result
n ÷ ±Infinity	0
±Infinity × ±Infinity	±Infinity
±nonZero ÷ ±0	±Infinity
±finite × ±Infinity	±Infinity
Infinity + Infinity Infinity – -Infinity	+Infinity
-Infinity – Infinity -Infinity + – Infinity	-Infinity
±0 ÷ ±0	NaN
±Infinity ÷ ±Infinity	NaN
±Infinity × 0	NaN
NaN == NaN	False

Thank you!