* Floating Point Representation

The floating point numbers must be represented in floating point representation.

In floating point representation, the computer must be able to represent the numbers and can be operated on them in such a way that the position of the binary point is variable and is automatically adjusted as computation proceeds, for the accommodation of very large integers and very small fractions. In this case, the binary point is said to be the float, and the numbers are called the floating point numbers.

The floating point representation has three fields.

- → sign
- → significant digits
- → exponent

Let us consider the number 111101.1000110 to be represented in floating point format

(i) First Binary point is shifted to the right of the first bit and the number is multiplied by the correct scaling factor to get same value. The number is said to be in normalized form and given as

111101.1000110

→ $1.1110110001 10 \times 2^{5}$ ← Exponent

Significant digit

Scaling factor

→ To represent the number in floating point format, the first binary point is shifted to the right of the first bit and the number is multiplied by the correct scaling factor to get the same value. The number is said to be in the normalized form

→ It is important to note that the base in the scaling factor is fixed 2.

→ The string of the significant digits is commonly known as mantissa.

→ In the above example, we can say that

Sign = 1 (negative)

mantissa = 11101100110

Exponent = 5

→ In floating point numbers, the bias value is added to the true exponent.