IEEE 754 32-Bit method: Following are the steps that we follow to represent a Floating Point number in 32 bit register:

1. Binary Conversion
2. Normalization
3. Representation

Ex. 1- Represent the number (+362.625) in 32 bit register using IEEE 754 method.

Here the number is (+362.625). Its integer part is 362 and fractional part is .625

Step1- We convert the binary of both parts.

\[(362)_{10} = (101101010)_{2}\]

\[(.625)_{10} = (101)_{2}\]

Hence the entire number becomes \[(+362.625)_{10} = (101101010.101)_{2}\]

Step2- We normalize the converted binary in to the \([m \times r^e]\) format. For this, we move the decimal point to the extreme left leaving one single 1 omitted.

Thus it becomes 1.01101010101 \(x 2^8\) \{since the decimal point has been moved 8 places left\}

Step3- We prepare our parts for representation.

The sign = 0 \{for positive\}

Mantissa = 01101010101 \{the fractional part leaving the MSB 1 omitted\}

Exponent = (8 + 127) = 135 \{See the NOTE\}

NOTE: The fact behind adding 127 to the exponent is a bit more interesting. Since IEEE hasn’t defined the exact side where the fractional point should be moved while normalization. A left movement will produce a +ve exponent while a right movement will produce a -ve exponent. Now to make sure that the exponent is always +ve while storing in the register, we do add \(127 \times (2^8 / 2)\) to our produced exponent.
Now the representation

**NOTE:** Another example of this representation will be discussed tomorrow for -ve numbers. Till then observe this example.

**Assignment:**

1. Represent the number (+589.1250) in 32 bit register using IEEE 754 method.