IEEE 754

IEEE 754 is the currently-used convention of storing the floating-point numbers (simply floats) in a computer register of a certain bit size. Each register has one sign bit (sbit), few exponent bits (ebits) and some mantissa bits (mbits). For a 64 bit machine, there are 11 ebits and 52 mbits. Each bit stores either 0 or 1 (duh!). There are certain patterns which are used for storing the special numbers.

sbit	**ebits**	**mbits**	**type**
0 or 1	all 1s	all 0s	$\pm\infty$
0 or 1	all 0s	all 0s	± 0
0 or 1	00. . .01 to 11. . .10	anything	normal numbers
0 or 1	all 0s	not all 0s	subnormal or denormalized numbers
0 or 1	all 11s	not all 0s	NaN

Let N_e = number of ebits = 11. Let e_i denote the exponent bits (counted from *r.h.s.*). Let m_i denote the mantissa bits (counted from *l.h.s.*).

$$\boxed{s} \quad \boxed{e_{10}}\boxed{e_9} \cdots \boxed{e_1}\boxed{e_0} \quad \boxed{m_1}\boxed{m_2} \cdots \boxed{m_{51}}\boxed{m_{52}}$$

Steps to convert the bits to a float

1. Check if the bits match with any of the special patterns other than normal or subnormal numbers. If not, do the following.
2. Calculate the *bias*. $bias = (2^{N_\mathrm{e}} - 2)/2 = 1023$.
3. Calculate the decimal value of the number stored in the exponent. This is the old-fashioned binary to decimal conversion, i.e.

$$\exp = \sum_{i=0}^{10} e_i \times 2^i,$$

   where the counting is from the *rhs* (see the figure above).
4. If it is a normal number, then, the value is

$$(-1)^s \times 2^{\exp-\mathrm{bias}} \times \left(1.0 + \sum_{i=1}^{52} \frac{m_i}{2^i}\right).$$

   The 1 in the bracket is the *leading one.*
5. If it is a subnormal number, then, the value is

$$(-1)^s \times 2^{\exp-\mathrm{bias}+1} \times \sum_{i=1}^{52} \frac{m_i}{2^i}.$$

   Note: there is no *leading one.*