Floating point correction

1-bit	11- bits	52 bits
Sign	exponent	mantissa

Sign bit :- 0 or 1

0 - positive
1 - negative

Exponent :- 11 bits for magnitude and sign are stored using a biased representation, with bias 1023. I.e

True Binary exponent = stored exponent − 1023

Mantissa :- 52 bits

Convert number to binary expansion, with digit next to decimal point to be 1 in the normalised representation. This gives 1-extra bit for free

Example :- 15.8125

$15 = 2 \times 7 + 1$

$7 = 2 \times 3 + 1$

$3 = 2 \times 1 + 1$

$1 = 2 \times 0 + 1$

$15 = 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2 + 1 \cdot 2^0$

$= 1111$

$0.8125 \times 2 = 1. + 0.825$

$0.625 \times 2 = 1 + 0.25$

$0.25 \times 2 = 0 + 0.5$

$0.5 \times 2 = 1 + 0$

$0.8125 = 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 1 \cdot 2^{-4}$

$.1101$

$15.8125 = 1111.1101$

$= 1.1111101 \times 2^3$

normalised

always 1

Sign bit = 0

stored exponent = $3 + 1023 = 1026$

$= 100\ 0000\ 0010$

Normalised mantissa = 111 11 01 45 zeros

15.8125 =

0 100 0000 0010 111 11 010 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000

Other Conventions :-

Exponent $\quad 0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0$ $\quad$ $1\,1\,1\,1\,1\,1\,1\,1\,1\,1\,1$ $\quad$ – reserved

Smallest positive number

$0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1$ (52)

$$= \left(1 + \sum_{j=1}^{52} 0 \cdot 2^{-j}\right) \times 2^{1-1023}$$

$$\simeq 2.2 \times 10^{-308}$$

largest positive number

$0\,1\,1\,1\,1\,1\,1\,1\,1\,1\,0$ (52)

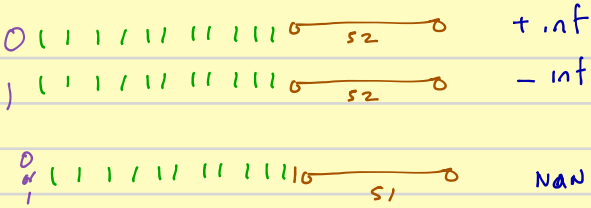
$$= \left(1 + \sum_{j=1}^{52} 2^{-j}\right) \times 2^{(2046-1023)}$$

$$\simeq 1.8 \times 10^{308}$$

Special numbers

$0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0$ (52) $\quad +0$

$1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0$ (52) $\quad -0$



Denormal numbers :- "Smaller than smallest"

- sacrifice on 52 - bit precision and
- enlarge the exponent storage
- thus going below . Machine $\$ double . xmin