Fractional numbers.

Given a number 15.213 in base-10 is

$$1\times 10 + 5\times 1 + 2\times\frac{1}{10} + \frac{1}{100}\times 1 + \frac{3}{1000} \quad \Bigg| \quad \sum_{i=-n}^{m} a_i \cdot 10^i$$

Similarly, we can have representations in binary as well.

$$\underbrace{\sum_{i=-n}^{m} a_i \cdot 2^i}$$

↳ Uses m+n+1 bits.

Qn: How do we represent π ?

↳ How do we represent numbers with repeating parts?

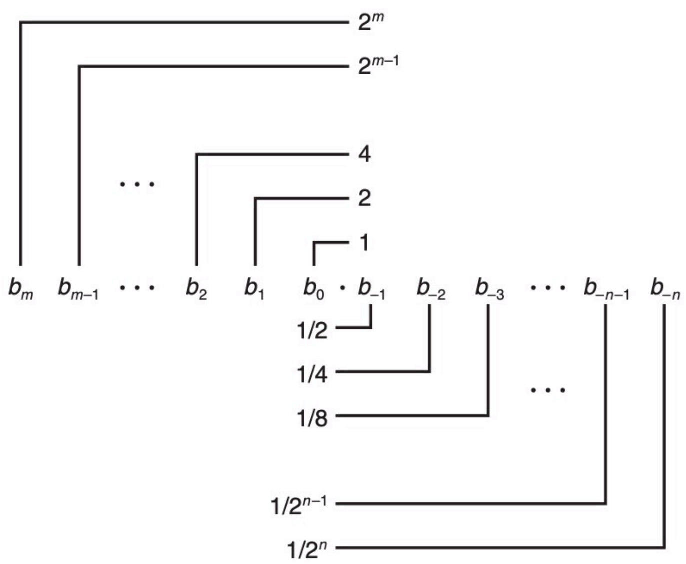


Figure 2.30 **Fractional binary representation.** Digits to the left of the binary point have weights of the form 2^i, while those to the right have weights of the form $1/2^i$.

Qn: Is there a better way of expressing numbers in the form $x \times 2^y$?

Floating point

We can represent numbers in the form $(-1)^s \times M \times 2^E$

s: sign
M: Significand $\in [1,2)$
E: Exponent

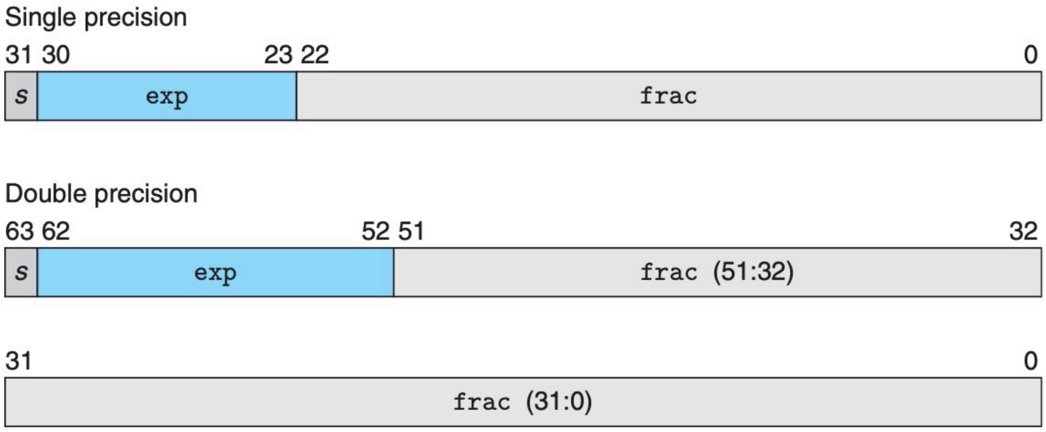


Figure 2.31 **Standard floating-point formats.** Floating-point numbers are represented by three fields. For the two most common formats, these are packed in 32-bit (single precision) or 64-bit (double precision) words.

s : Single bit
exp: $e_{k-1}\, e_{k-2} \cdots e_0$
frac: $f_{n-1}, \dots, f_0$

Normalized values:

Case when exp is not all zeroes nor all 1s.

$E = e - \text{bias}$ ← $2^{k-1}-1$.

e: $\underbrace{e_{k-1} \cdots e_0}_{\text{unsigned repr.}}$

$k=8 \Rightarrow 2^{k-1}-1$
$= 128-1 = 127$.

→ E ranges from -126 to 127.

$M = 1+f$ where $f \in [0,1)$. $\Rightarrow$ $1 \leq M < 2$

Denormalized values: Case when exp is all zeroes.

$E = 1 - \text{bias}$

Note that this is $1-\text{bias}$ instead of $-\text{bias}$.

$M = f$.

Qu: How do we represent zero?

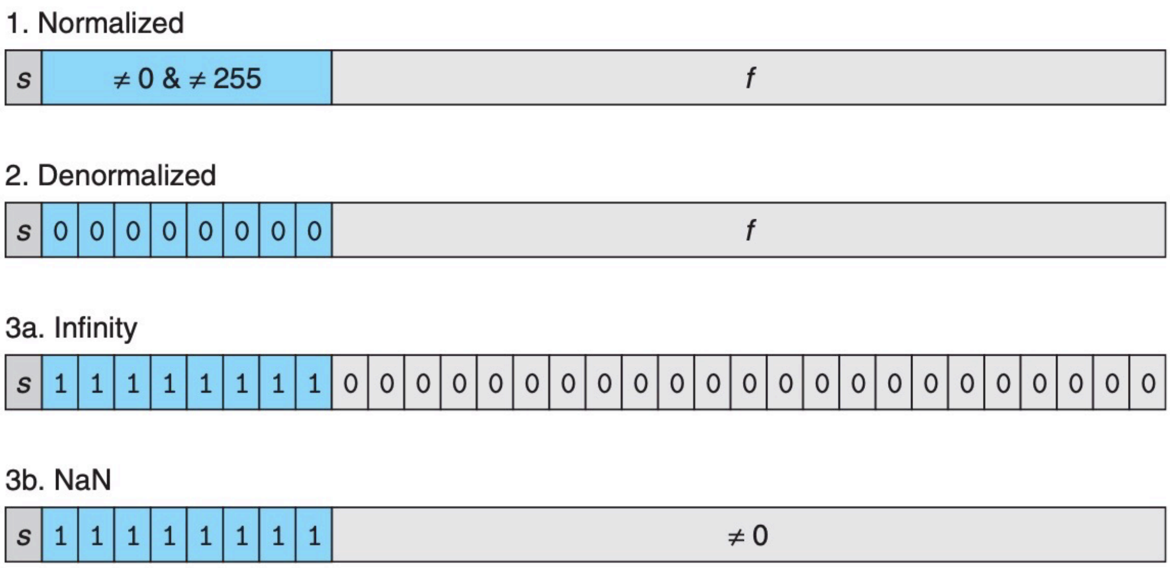


Figure 2.32 **Categories of single-precision, floating-point values.** The value of the exponent determines whether the number is (1) normalized, (2) denormalized, or a (3) special value.

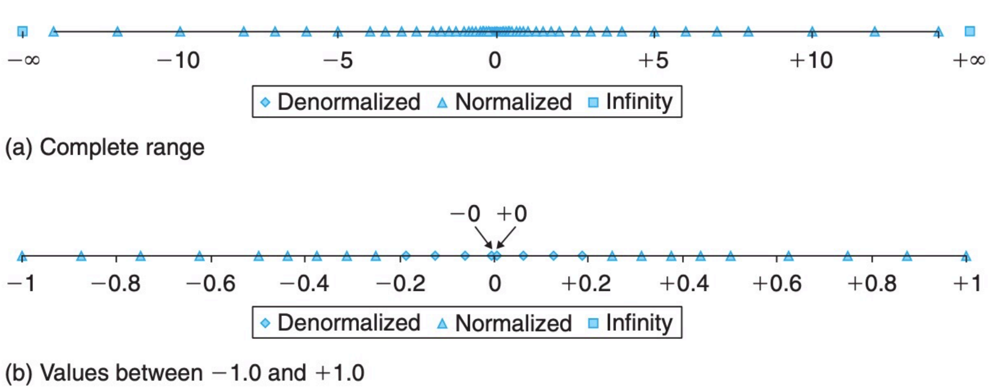


Figure 2.33 **Representable values for 6-bit floating-point format.** There are $k = 3$ exponent bits and $n = 2$ fraction bits. The bias is 3.

Largest denormalized number:

Ex: $k=4$. Bias $=7$, $n=3$

$E = 1-7 = -6$. $\}$ $2^{E} = \frac{1}{64}$.

f can at most be $\frac{1}{2}+\frac{1}{4}+\frac{1}{8} = \frac{7}{8}$.

$$V_{min}^{denorm} = \frac{1}{64}\times\frac{7}{8} = \frac{7}{512}.$$

Smallest normalized number:

$E = 1-7 = -6$

$M \in [1, 1+\frac{7}{8}]$.

Range $= \frac{1}{64}$ to $\frac{15}{512}$ $\Rightarrow$ $\frac{8}{512}$ to $\underbrace{\frac{15}{512}}$

Max value is 240.

More generally,

Denormalized numbers: $\underbrace{M \geq 2^{-n}. \quad E = -2^{k-1}+2.}$ $\Bigg\}$ smallest

$$V = 2^{2-n-2^{k-1}}.$$

$$\left.\begin{aligned} M &= 1-2^{-n} \\ E &= -2^{k-1}+2 \\ V &= (1-2^{-n})\times 2^{-2^{k-1}+2}. \end{aligned}\right\} \text{Largest denorm. value.}$$

Smallest normalized value: $E = -2^{k-1}+2$; $M=1$

$V = 2^{-2^{k-1}+2}$

Qn: How do we represent 1?

Largest normalized value:

$V = (2-2^{-n}) \times 2^{2^{k-1}-1}$

Rounding:

$x^{-} \leq x \leq x^{+}$

FP Multiplication

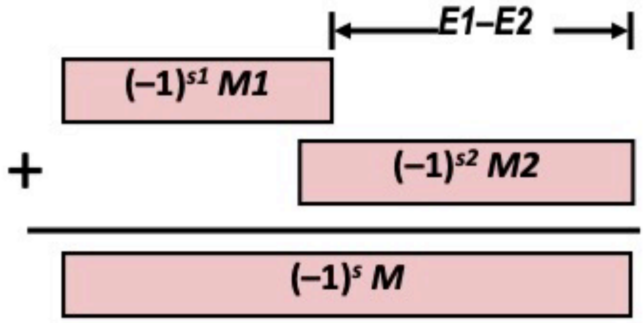
- $(-1)^{s1}\, M1\ 2^{E1}$ x $(-1)^{s2}\, M2\ 2^{E2}$
- **Exact Result:** $(-1)^{s}\, M\ 2^{E}$
  - Sign s: $s1$ ^ $s2$
  - Significand M: $M1 \times M2$
  - Exponent E: $E1 + E2$
- **Fixing**
  - If $M \geq 2$, shift M right, increment E
  - If E out of range, overflow
  - Round M to fit `frac` precision
- **Implementation**
  - Biggest chore is multiplying significands

4 bit significand: $1.010 * 2^2 \times 1.110 * 2^3 = 10.0011 * 2^5$
$= 1.00011 * 2^6 = 1.001 * 2^6$

Floating Point Addition

- $(-1)^{s1}\ M1\ 2^{E1}\ +\ (-1)^{s2}\ M2\ 2^{E2}$
  - Assume $E1 > E2$
- **Exact Result:** $(-1)^{s}\ M\ 2^{E}$
  - Sign s, significand M:
    - Result of signed align & add
  - Exponent E: $E1$
- **Fixing**
  - If $M \geq 2$, shift M right, increment E
  - if $M < 1$, shift M left k positions, decrement E by k
  - Overflow if E out of range
  - Round M to fit `frac` precision

Get binary points lined up



$$1.010*2^2 + 1.110*2^3 = (0.1010 + 1.1100)*2^3$$
$$= 10.0110 * 2^3 = 1.00110 * 2^4 = 1.010 * 2^4$$