Data representation: Floating Point Denormalized Numbers and Mastery

Yipeng Huang

Rutgers University

October 21, 2025

Table of contents

Floats: Overview

Floats: Normalized numbers

- Normalized: exp field
- Normalized: frac field
- Normalized: example

Floats: Denormalized numbers

- Denormalized: exp field
- Denormalized: frac field
- Denormalized: examples

Floats: Special cases

Floats: Understanding its design

- Deep understanding 1: Why is exp field encoded using bias?
- Deep understanding 2: Why have denormalized numbers?
- Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?

Floats: Properties

- Floating point multiplication

Floats: Summary

Floating point numbers

Avogadro's number

$+6.02214 \times 10^{23}\, mol^{-1}$

Scientific notation

- sign
- mantissa or significand
- exponent

Floating point numbers

Before 1985

1. Many floating point systems.
2. Specialized machines such as Cray supercomputers.
3. Some machines with specialized floating point have had to be kept alive to support legacy software.

After 1985

1. IEEE Standard 754.
2. A floating point standard designed for good numerical properties.
3. Found in almost every computer today, except for tiniest microcontrollers.

Recent

1. Need for both lower precision and higher range floating point numbers.
2. Machine learning / neural networks. Low-precision tensor network processors.

Floats and doubles

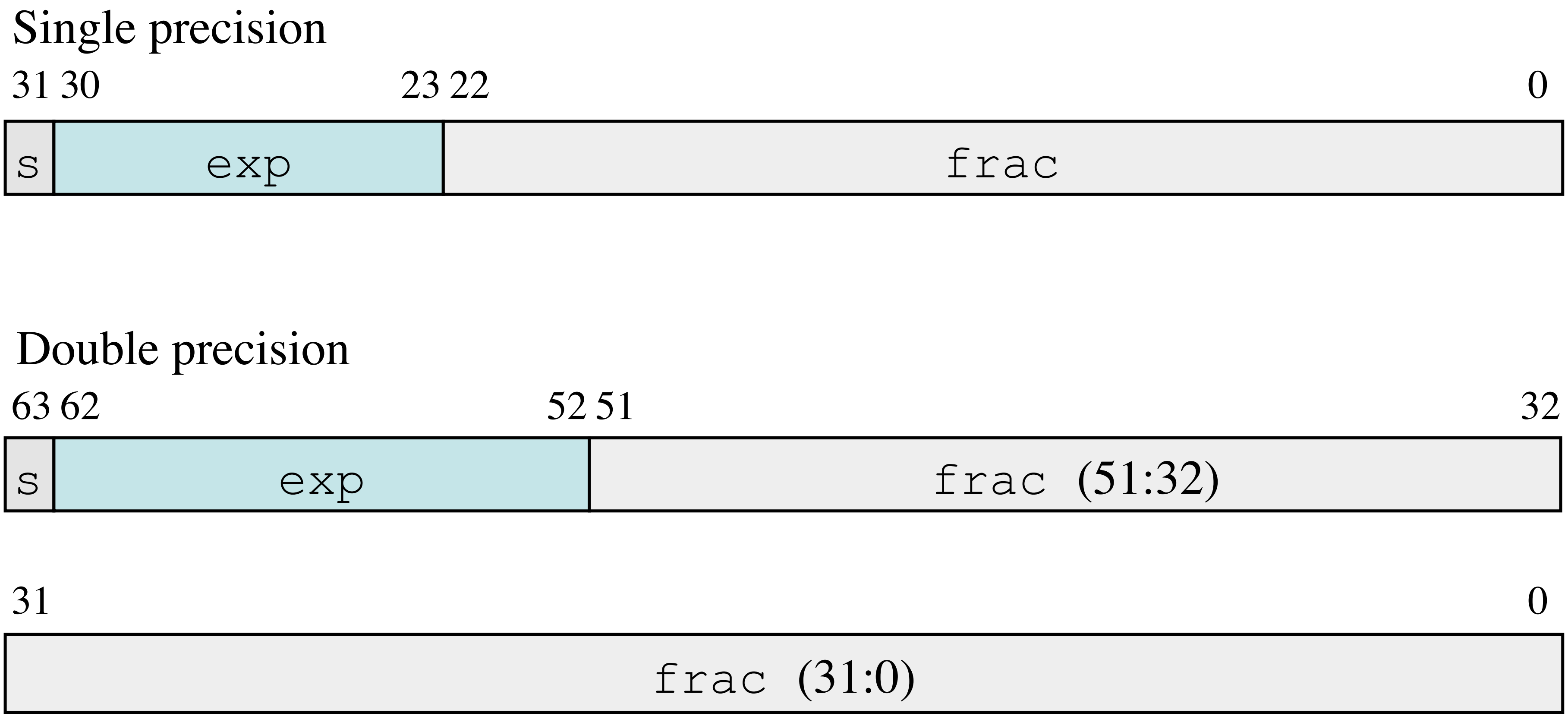


Figure: The two standard formats for floating point data types. Image credit CS:APP

Floats and doubles

property	half*	float	double
total bits	16	32	64
s bit	1	1	1
exp bits	5	8	11
frac bits	10	23	52
C printf() format specifier	None	"%f"	"%lf"

Table: Properties of floats and doubles

The IEEE 754 number line

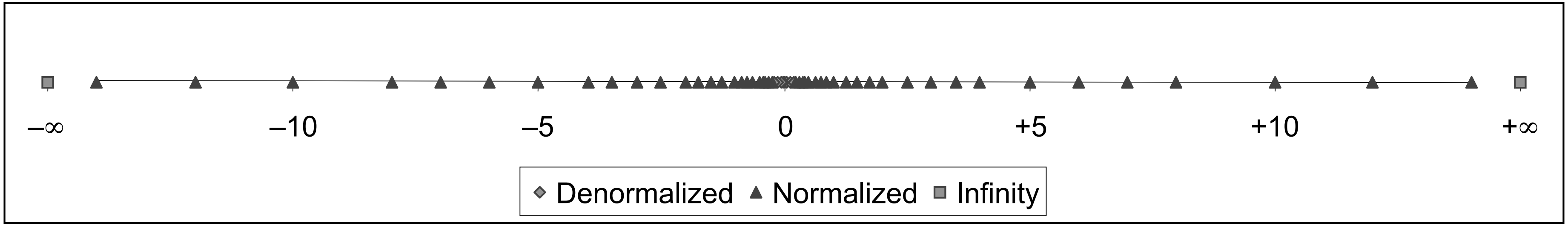


Figure: Full picture of number line for floating point values. Image credit CS:APP

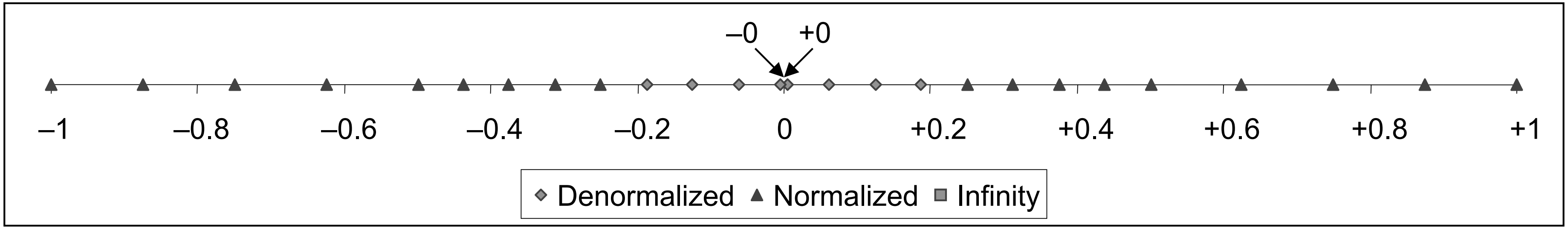


Figure: Zoomed in number line for floating point values. Image credit CS:APP

Different cases for floating point numbers

Value of the floating point number = $(-1)^s \times M \times 2^E$

- E is encoded the exp field
- M is encoded the frac field

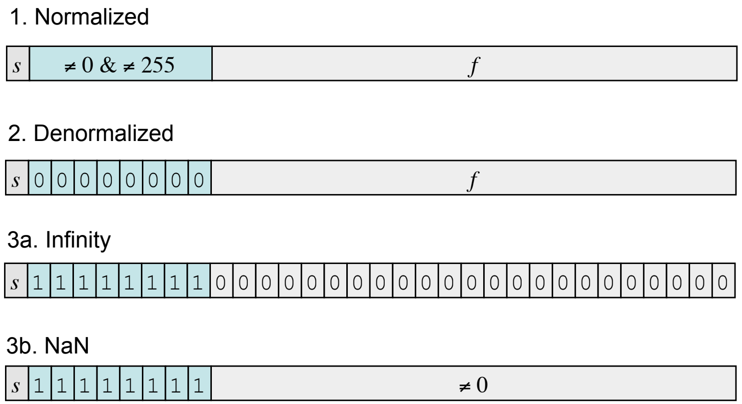


Figure: Different cases within a floating point format. Image credit CS:APP

Normalized and denormalized numbers

Two different cases we need to consider for the encoding of E, M

Table of contents

- Floats: Overview
- Floats: Normalized numbers
  - Normalized: exp field
  - Normalized: frac field
  - Normalized: example
- Floats: Denormalized numbers
  - Denormalized: exp field
  - Denormalized: frac field
  - Denormalized: examples
- Floats: Special cases
- Floats: Understanding its design
  - Deep understanding 1: Why is exp field encoded using bias?
  - Deep understanding 2: Why have denormalized numbers?
  - Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?
- Floats: Properties
  - Floating point multiplication
- Floats: Summary

Normalized: exp field

For normalized numbers,
$0 < \text{exp} < 2^k - 1$

- exp is a k-bit unsigned integer

Bias

- need a bias to represent negative exponents
- bias = $2^{k-1} - 1$
- bias is the k-bit unsigned integer: 011..111

For normalized numbers,
E = exp-bias

In other words, exp = E+bias

property	float	double
k	8	11
bias	127	1023
smallest E (greatest precision)	-126	-1022
largest E (greatest range)	127	1023

Table: Summary of normalized exp field

Normalized: frac field

M = 1.frac

Normalized: example

- 12.375 to single-precision floating point
- sign is positive so s=0
- binary is 1100.011_2
- in other words it is $1.100011_2 \times 2^3$
- $\text{exp} = E + \text{bias} = 3 + 127 = 130 = 1000\_0010_2$
- M = 1.100011_2 = 1.frac
- frac = 100011

positive infinity

largest normalized number

epsilon

0_0111_001

$$\Rightarrow (-1)^s \cdot (1.001_2) \cdot 2^{7-7}$$

$$= (1 + \frac{1}{8}) \cdot 2^0 = \frac{9}{8}$$

epsilon $= \frac{1}{8}$

one

0_0111_000

Smallest normalized number

0_0001_000

$$\Rightarrow (-1)^s \cdot (\text{mantissa}) \cdot 2^E$$

$$= (+1) \cdot (1.000_2) \cdot 2^{(\text{exp}-\text{bias})}$$

$$= (+1) \cdot (1.0) \cdot 2^{(1-7)}$$

$$= \frac{1}{64}$$

$$\text{bias} = (2^{(k-1)} - 1)$$

$$= 2^{(4-1)} - 1$$

$$= 7$$

zero

Table of contents

Floats: Overview

Floats: Normalized numbers

- Normalized: exp field
- Normalized: frac field
- Normalized: example

Floats: Denormalized numbers

- Denormalized: exp field
- Denormalized: frac field
- Denormalized: examples

Floats: Special cases

Floats: Understanding its design

- Deep understanding 1: Why is exp field encoded using bias?
- Deep understanding 2: Why have denormalized numbers?
- Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?

Floats: Properties

- Floating point multiplication

Floats: Summary

The IEEE 754 number line

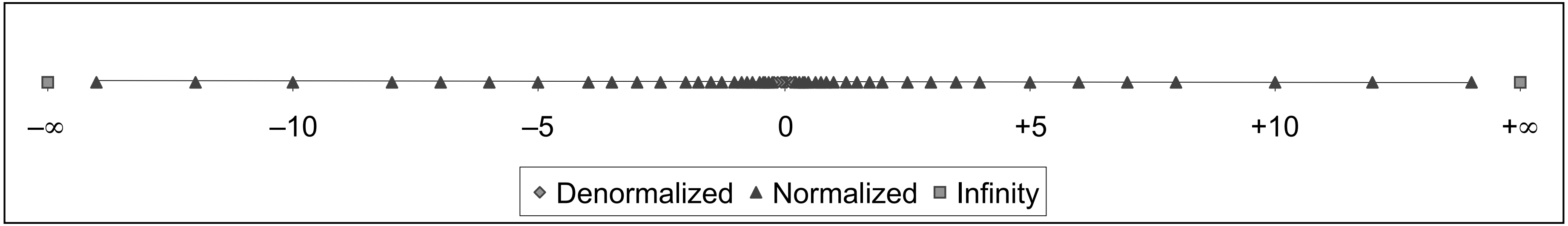


Figure: Full picture of number line for floating point values. Image credit CS:APP

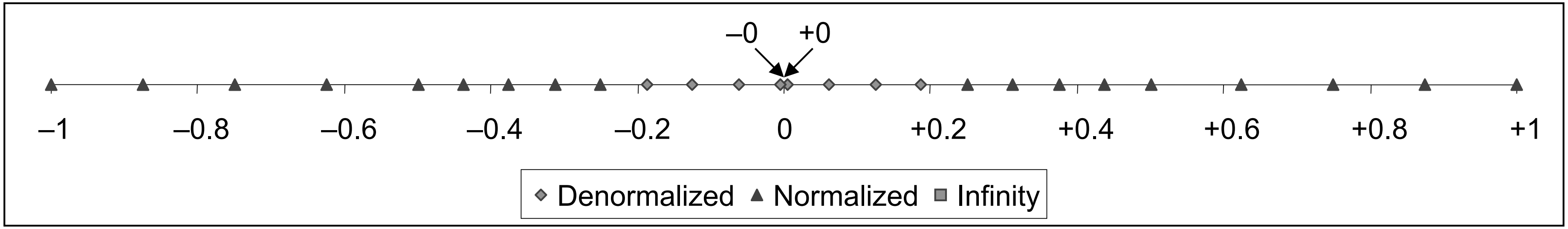


Figure: Zoomed in number line for floating point values. Image credit CS:APP

Denormalized: exp field

For denormalized numbers, exp $= 0$

Bias

- need a bias to represent negative exponents
- bias = $2^{k-1} - 1$
- bias is the k-bit unsigned integer: 011..111

For denormalized numbers,
E = 1-bias

property	float	double
k	8	11
bias	127	1023
E	-126	-1022

Table: Summary of denormalized exp field

Denormalized: frac field

M = 0.frac
value represented leading with 0

Denormalized: examples

largest denormalized number

Without special treatment (not how it works)

0_0000_111

$$\Rightarrow (-1)^0 \cdot (1.111_2) \cdot 2^{0-7}$$

$$= \left(1+\frac{1}{2}+\frac{1}{4}+\frac{1}{8}\right) \cdot \frac{1}{128}$$

$$= \frac{15}{8} \cdot \frac{1}{128} = \frac{15}{2^{10}}$$

$$\frac{1}{64} - \frac{15}{2^{10}} = \frac{1}{2^6} - \frac{15}{2^{10}} = \frac{2^4-15}{2^{10}} = \frac{16-15}{2^{10}}$$

with special treatment

0_0000_111

$$= (-1)^S \cdot (0.111_2) \cdot 2^E$$

$$= (-1)^S \cdot \left(\frac{1}{2}+\frac{1}{4}+\frac{1}{8}\right) \cdot 2^{1-bias}$$

$$= (+1) \cdot \frac{7}{8} \cdot 2^{-6}$$

$$= \frac{7}{2^3} \cdot \frac{1}{2^6} = \frac{7}{2^9} = \frac{7}{512}$$

$$\frac{1}{64} - \frac{7}{512} = \frac{1}{2^6} - \frac{7}{2^9} = \frac{2^3-7}{2^9} = \frac{8-7}{2^9} = \frac{1}{512}$$

Smallest denormalized number

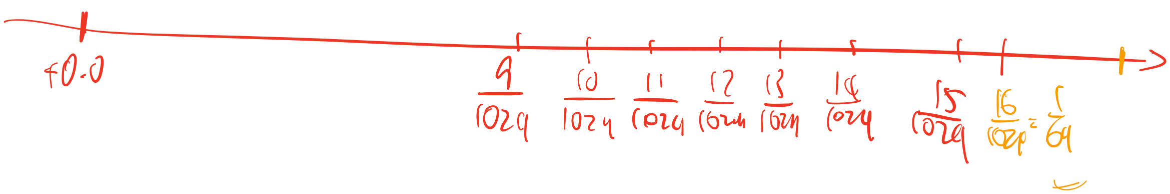
Without special treatment (not how it works)

0_0000_001

$$\Rightarrow (-1)^s \cdot (1.001) \cdot 2^{0-7}$$

$$= \left(1+\frac{1}{8}\right) \cdot 2^{-7}$$

$$= \frac{9}{8} \cdot 2^{-7} = \frac{9}{2^3} \cdot 2^{-7} = \frac{9}{1024}$$



With special treatment

0_0000_001

$$\Rightarrow (-1)^s \cdot (0.001_2) \cdot 2^{1-\text{bias}}$$

$$= (+1) \cdot \frac{1}{8} \cdot 2^{-6}$$

$$= \frac{1}{2^9} = \frac{1}{512}$$

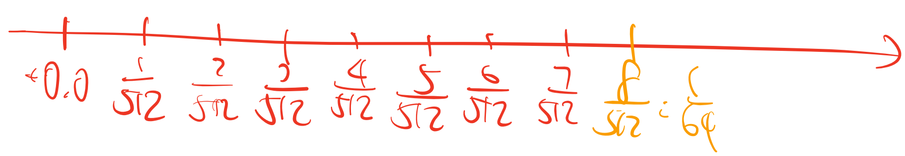


Table of contents

- Floats: Overview
- Floats: Normalized numbers
  - Normalized: exp field
  - Normalized: frac field
  - Normalized: example
- Floats: Denormalized numbers
  - Denormalized: exp field
  - Denormalized: frac field
  - Denormalized: examples
- Floats: Special cases
- Floats: Understanding its design
  - Deep understanding 1: Why is exp field encoded using bias?
  - Deep understanding 2: Why have denormalized numbers?
  - Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?
- Floats: Properties
  - Floating point multiplication
- Floats: Summary

Floats: Special cases

number class	when it arises	exp field	frac field
+0 / -0		0	0
+infinity / -infinity	overflow or division by 0	$2^k - 1$	0
NaN not-a-number	illegal ops. such as $\sqrt{-1}$, inf-inf, inf*0	$2^k - 1$	non-0

Table: Summary of special cases

Why have both +0.0 and -0.0?

1_0000_000

$\Rightarrow (-1)^1 \cdot (0.000) \cdot 2^{1-7}$

$= -0.0$

$\frac{1}{-\infty} = -0.0$

$\frac{1}{\frac{1}{-\infty}} = -\infty$

$\frac{1}{\infty} = 0.0$

$\frac{1}{\frac{1}{\infty}} = +\infty$

Table of contents

- Floats: Overview
- Floats: Normalized numbers
  - Normalized: exp field
  - Normalized: frac field
  - Normalized: example
- Floats: Denormalized numbers
  - Denormalized: exp field
  - Denormalized: frac field
  - Denormalized: examples
- Floats: Special cases
- Floats: Understanding its design
  - Deep understanding 1: Why is exp field encoded using bias?
  - Deep understanding 2: Why have denormalized numbers?
  - Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?
- Floats: Properties
  - Floating point multiplication
- Floats: Summary

Deep understanding 1: Why is exp field encoded using bias?

exp field needs to encode both positive and negative exponents.

Why not just use one of the signed integer formats? 2's complement, 1s' complement, signed magnitude?

Deep understanding 1: Why is exp field encoded using bias?

exp field needs to encode both positive and negative exponents.

Why not just use one of the signed integer formats? 2's complement, 1s' complement, signed magnitude?

Answer: allows easy comparison of magnitudes by simply comparing bits.

Deep understanding 1: Why is exp field encoded using bias?

exp field needs to encode both positive and negative exponents.

Why not just use one of the signed integer formats? 2's complement, 1s' complement, signed magnitude?

Answer: allows easy comparison of magnitudes by simply comparing bits.

Consider hypothetical 8-bit floating point format (from the textbook)

1-bit sign, $k = 4$-bit exp, 3-bit frac.

What is the decimal value of 0b1_0110_111?

What is the decimal value of 0b1_0111_000?

Deep understanding 1: Why is exp field encoded using bias?

exp field needs to encode both positive and negative exponents.

Why not just use one of the signed integer formats? 2's complement, 1s' complement, signed magnitude?

Answer: allows easy comparison of magnitudes by simply comparing bits.

Consider hypothetical 8-bit floating point format (from the textbook)

1-bit sign, $k = 4$-bit exp, 3-bit frac.

What is the decimal value of 0b1_0110_111?

-1.875×2^{-1}

What is the decimal value of 0b1_0111_000?

-2.000×2^{-1}

Deep understanding 2: Why have denormalized numbers?

Why not just continue normalized number scheme down to smallest numbers around zero?

Answer: makes sure that smallest increments available are maintained around zero.

Suppose denormalized numbers NOT used.

What is the decimal value of 0b0_0000_001?

1.125×2^{-7}

What is the decimal value of 0b0_0000_111?

1.875×2^{-7}

What is the decimal value of 0b0_0001_000?

2.000×2^{-7}

Deep understanding 2: Why have denormalized numbers?

Why not just continue normalized number scheme down to smallest numbers around zero?

Answer: makes sure that smallest increments available are maintained around zero.

Suppose denormalized numbers ARE used.

What is the decimal value of 0b0_0000_001?

0.125×2^{-6}

What is the decimal value of 0b0_0000_111?

0.875×2^{-6}

What is the decimal value of 0b0_0001_000?

1.000×2^{-6}

Table of contents

Floats: Overview

Floats: Normalized numbers

- Normalized: exp field
- Normalized: frac field
- Normalized: example

Floats: Denormalized numbers

- Denormalized: exp field
- Denormalized: frac field
- Denormalized: examples

Floats: Special cases

Floats: Understanding its design

- Deep understanding 1: Why is exp field encoded using bias?
- Deep understanding 2: Why have denormalized numbers?
- Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?

Floats: Properties

- Floating point multiplication

Floats: Summary

How to multiply scientific notation?

Recall: $log(x \times y) = log(x) + log(y)$

Floating point multiplication

FP Multiplication

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

Table of contents

- Floats: Overview
- Floats: Normalized numbers
  - Normalized: exp field
  - Normalized: frac field
  - Normalized: example
- Floats: Denormalized numbers
  - Denormalized: exp field
  - Denormalized: frac field
  - Denormalized: examples
- Floats: Special cases
- Floats: Understanding its design
  - Deep understanding 1: Why is exp field encoded using bias?
  - Deep understanding 2: Why have denormalized numbers?
  - Deep understanding 3: Why is bias chosen to be $2^{k-1} - 1$?
- Floats: Properties
  - Floating point multiplication
- Floats: Summary

Floats: Summary

	normalized	denormalized
value of number	$(-1)^s \times M \times 2^E$	$(-1)^s \times M \times 2^E$
E	E = exp-bias	E = -bias + 1
bias	$2^{k-1} - 1$	$2^{k-1} - 1$
exp	$0 < exp < (2^k - 1)$	$exp = 0$
M	M = 1.frac	M = 0.frac
	M has implied leading 1	M has leading 0
	greater range	greater precision
	large magnitude numbers	small magnitude numbers
	denser near origin	evenly spaced

Table: Summary of normalized and denormalized numbers

Connecting to actual number ranges for 32-bit float and 64-bit double: