

FLOATING POINT NUMBERS

Introduction

Richard Hayden (with thanks to **Naranker Dulay**)
rh@doc.ic.ac.uk

<http://www.doc.ic.ac.uk/~rh/teaching>

or

<https://www.doc.ic.ac.uk/~wl/teachlocal/arch1>

or

CATE

Large integers

Example: How can we represent integers up to 30 decimal digits long?

Large integers

Example: How can we represent integers up to 30 decimal digits long?

- **Binary:** $\log_2(10^{30}) \approx 100$ bits (1 decimal digit ≈ 3.32 bits)
- **BCD:** $30 \times 4 = 120$ bits
- **ASCII:** $30 \times 8 = 240$ bits

IA-32 includes instructions which allow the construction of multi-precision integer arithmetic routines using any of these formats

Floating point numbers

Recall scientific notation:

$$M \times 10^E$$

Decimal

$$M \times 2^E$$

Binary

This is the basis for most floating point representation schemes

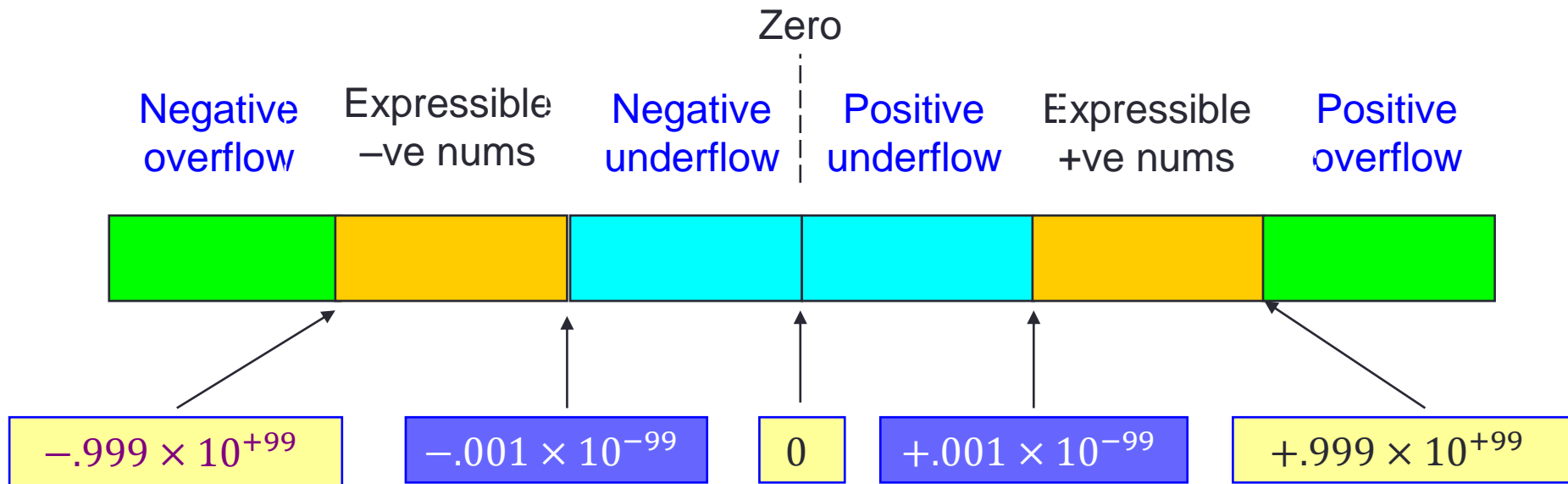
- M is the **mantissa** (aka. **significand**, **fraction** or **argument**)
- E is the **exponent** (aka. **characteristic**)
- 10 (or for binary, 2) is the **radix** (aka. **base**)

- No. bits in **exponent** determines the **range** (**bigness/smallness**)
- No. bits in **mantissa** determines the **precision** (**exactness**)

Zones of expressibility

- **Example:** assume numbers are formed with a **signed 3-digit mantissa** and a **signed 2-digit exponent**

Zones of expressibility:



Real vs. floating point numbers

	Mathematical real	Floating pointer number
Range	$-\infty \dots + \infty$	Finite
No. of values	(Uncountably) infinite	Finite
Spacing	?	Gap between numbers varies
Errors	?	Incorrect results are possible

Some questions (**assume signed 3-digit mantissa** and a **signed 2-digit exponent as before**):

- What are the **closest** floating point numbers to $.001 \times 10^{-99}$? What is the **gap** between this number and them?
- What about $.001 \times 10^{-50}$?

Normalised floating point numbers

- Depending on how you interpret the mantissa, floating point numbers **can have multiple forms**, e.g.:

$$\begin{aligned}0.023 \times 10^4 &= 0.230 \times 10^3 \\ &= 2.3 \times 10^2 \\ &= 0.0023 \times 10^5\end{aligned}$$

- For hardware implementations it is desirable for each number to have a unique floating point representation, a **normalised form**
- We'll normalise mantissas in the **range** $[1, \dots R)$ where R is the base, e.g.:

$[1, \dots, 10)$ for decimal

$[1, \dots, 2)$ for binary

Normalised forms (base 10)

Number	Normalised form
23.2×10^4	

Normalised forms (base 10)

Number	Normalised form
23.2×10^4	2.32×10^5
-4.01×10^{-3}	

Normalised forms (base 10)

Number	Normalised form
23.2×10^4	2.32×10^5
-4.01×10^{-3}	-4.01×10^{-3}
$343\,000 \times 10^0$	

Normalised forms (base 10)

Number	Normalised form
23.2×10^4	2.32×10^5
-4.01×10^{-3}	-4.01×10^{-3}
$343\,000 \times 10^0$	3.43×10^5
$0.000\,000\,098\,9 \times 10^0$	

Normalised forms (base 10)

Number	Normalised form
23.2×10^4	2.32×10^5
-4.01×10^{-3}	-4.01×10^{-3}
$343\,000 \times 10^0$	3.43×10^5
$0.000\,000\,098\,9 \times 10^0$	9.89×10^{-8}

Binary fractions

Binary	Decimal
0.1	0.5
0.01	0.25
0.001	

Binary fractions

Binary	Decimal
0.1	0.5
0.01	0.25
0.001	0.125
0.11	

Binary fractions

Binary	Decimal
0.1	0.5
0.01	0.25
0.001	0.125
0.11	0.75
0.111	

Binary fractions

Binary	Decimal
0.1	0.5
0.01	0.25
0.001	0.125
0.11	0.75
0.111	0.875
0.011	

Binary fractions

Binary	Decimal
0.1	0.5
0.01	0.25
0.001	0.125
0.11	0.75
0.111	0.875
0.011	0.375
0.101	

Binary fractions

Binary	Decimal
0.1	0.5
0.01	0.25
0.001	0.125
0.11	0.75
0.111	0.875
0.011	0.375
0.101	0.625

Binary fraction to decimal fraction

- What is the binary value **0.011010** in decimal?

	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
.	0	1	1	0	1

- $\frac{1}{4} + \frac{1}{8} + \frac{1}{32} = \frac{13}{32} = 0.40625$

	32	16	8	4	2	1
.	0	1	1	0	0	1

- $\frac{8+4+1}{32} = \frac{13}{32}$

- What about 0.000 110 011?

Binary fraction to decimal fraction

- What is the binary value **0.011 01** in decimal?

	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
.	0	1	1	0	1

- $\frac{1}{4} + \frac{1}{8} + \frac{1}{32} = \frac{13}{32} = 0.40625$

	32	16	8	4	2	1
.	0	1	1	0	0	1

- $\frac{8+4+1}{32} = \frac{13}{32}$

- What about 0.000 110 011?

- Answer: $\frac{32+16+2+1}{512} = \frac{51}{512} = 0.099609375$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$\begin{aligned} 0.6875 &= \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8} \\ &= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} \end{aligned}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$$

So the answer is **0.1011**

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$$

So the answer is **0.1011**

- What is the decimal value **0.1** in binary?

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$$

So the answer is **0.1011**

- What is the decimal value **0.1** in binary?

$$0.1 = \frac{1.6}{16}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$$

So the answer is **0.1011**

- What is the decimal value **0.1** in binary?

$$0.1 = \frac{1.6}{16} = \frac{1}{16} + \frac{0.6}{16}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$0.6875 = \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8}$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$$

So the answer is **0.1011**

- What is the decimal value **0.1** in binary?

$$0.1 = \frac{1.6}{16} = \frac{1}{16} + \frac{0.6}{16} = \frac{1}{16} + \frac{1.2}{32}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$\begin{aligned}0.6875 &= \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8} \\ &= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}\end{aligned}$$

So the answer is **0.1011**

- What is the decimal value **0.1** in binary?

$$0.1 = \frac{1.6}{16} = \frac{1}{16} + \frac{0.6}{16} = \frac{1}{16} + \frac{1.2}{32} = \frac{1}{16} + \frac{1}{32} + \frac{0.2}{32}$$

Decimal fraction to binary fraction

- What is the decimal value **0.6875** in binary?

$$\begin{aligned}0.6875 &= \frac{1.375}{2} = \frac{1}{2} + \frac{0.375}{2} = \frac{1}{2} + \frac{0.75}{4} = \frac{1}{2} + \frac{1.5}{8} \\ &= \frac{1}{2} + \frac{1}{8} + \frac{0.5}{8} = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}\end{aligned}$$

So the answer is **0.1011**

- What is the decimal value **0.1** in binary?

$$0.1 = \frac{1.6}{16} = \frac{1}{16} + \frac{0.6}{16} = \frac{1}{16} + \frac{1.2}{32} = \frac{1}{16} + \frac{1}{32} + \frac{0.2}{32} = \frac{1}{16} + \frac{1}{32} + \frac{1.6}{256}$$

...

Normalised forms (base 2)

Number	Normalised form
100.01×2^1	

Normalised forms (base 2)

Number	Normalised form
100.01×2^1	1.0001×2^3
1010.11×2^2	

Normalised forms (base 2)

Number	Normalised form
100.01×2^1	1.0001×2^3
1010.11×2^2	1.01011×2^5
0.00101×2^{-2}	

Normalised forms (base 2)

Number	Normalised form
100.01×2^1	1.0001×2^3
1010.11×2^2	1.01011×2^5
0.00101×2^{-2}	1.01×2^{-5}
1100101×2^{-2}	1.100101×2^4

Floating point multiplication

$$\begin{aligned}N_1 \times N_2 &= (M_1 \times 10^{E_1}) \times (M_2 \times 10^{E_2}) \\ &= (M_1 \times M_2) \times (10^{E_1} \times 10^{E_2}) \\ &= (M_1 \times M_2) \times (10^{E_1+E_2})\end{aligned}$$

- That is, we **multiply the mantissas** and **add the exponents**
- Example:

$$\begin{aligned}(2.6 \times 10^6) \times (5.4 \times 10^{-3}) &= (2.6 \times 5.4) \times (10^3) \\ &= 14.04 \times 10^3\end{aligned}$$

- We must also **normalise the result**, so final answer is 1.404×10^4

Truncation and rounding

- For many computations, the result of a floating point operation is **too large to store in the mantissa**
- Example (with a **2-digit mantissa**):

$$(2.3 \times 10^1) \times (2.3 \times 10^1) = 5.29 \times 10^2$$

- **Truncation** $\rightarrow 5.2 \times 10^2$ (biased error)
- **Rounding** $\rightarrow 5.3 \times 10^2$ (unbiased error)

Floating point addition

- A floating point addition such as $4.5 \times 10^3 + 6.7 \times 10^2$ is not a simple mantissa addition, unless the exponents are the same. Otherwise, we need to align them first

$$\begin{aligned} N_1 + N_2 &= (M_1 \times 10^{E_1}) + (M_2 \times 10^{E_2}) \\ &= (M_1 + M_2 \times 10^{E_2 - E_1}) \times 10^{E_1} \end{aligned}$$

- To align, **choose the number with the smaller exponent** and **shift its mantissa** the corresponding number of digits to the right

$$\begin{aligned} 4.5 \times 10^3 + 6.7 \times 10^2 &= 4.5 \times 10^3 + 0.67 \times 10^3 \\ &= 5.17 \times 10^3 = 5.2 \times 10^3 \end{aligned}$$

(rounded)

Exponent overflow and underflow

- **Exponent overflow** occurs when the result is **too large** i.e. when the **result's exponent > maximum exponent**

Example: if max exponent is 99 then $10^{99} \times 10^{99} = 10^{198}$ (**overflow**)

To handle **overflow**, set value as infinity or raise an exception

Exponent underflow occurs when the result is **too small** i.e. when the **result's exponent < smallest exponent**

Example: if min exponent is -99 then $10^{-99} \times 10^{-99} = 10^{-198}$
(**underflow**)

To handle **underflow**, set value as zero or raise an exception

Comparing floating point values

- Because of the potential for producing **inexact results**, comparing floating point values should **account for close results**
- If we know the **desired magnitude and precision** of results, we can adjust for closeness (**epsilon**). For example:

$$a = b \qquad a > (b - \epsilon) \quad \text{and} \quad a < (b + \epsilon)$$

$$a = 1 \qquad a > 1 - 0.000005 \qquad \text{and} \qquad a < 1 + 0.000005$$
$$\qquad a > 0.999995 \qquad \text{and} \qquad a < 1.000005$$

- A more general approach is to calculate closeness of two numbers based on the **relative size** of the two numbers being compared