

Cairo University
 Electronics and Communications Engineering
 Computer Arithmetic, Lecture 3:
 The 'fine print' of floating point
 standards
 Hossam A. H. Fahmy

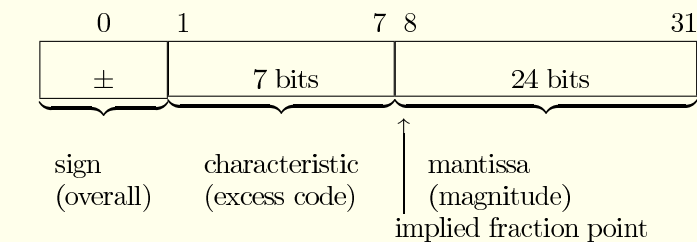


© 2006

2

Exponent range, significand width, and radix

Assume a 32-bit format:



	Largest Number	Smallest Number	Precision	Accuracy
$\beta = 16$	7.2×10^{75}	5.4×10^{-79}	16^{-6}	2^{-21}
$\beta = 2$	9.2×10^{18}	2.7×10^{-20}	2^{-24}	2^{-24}

Accuracy is the guaranteed or minimum number of significant mantissa bits excluding any leading zeros. Base 2 provides a better accuracy but less range.

3

Representation errors

For a number x , $f_x \times \beta^{exp}$ is its exact (normalized) representation. The computer represents x as $f_R \times \beta^{exp}$.

MRRE is the maximum error relative to x ,

$$\begin{aligned}
 MRRE &= \max\left(\frac{|f_x \beta^{exp} - f_R \beta^{exp}|}{f_x \beta^{exp}}\right) \\
 &= \max\left(\frac{1/2 \times 2^{-t}}{f_x}\right) \\
 &= \frac{1/2 \times 2^{-t}}{1/\beta} = 2^{-t-1} \beta
 \end{aligned}$$

To have the same (or better) MRRE for $\beta = 2^k$ and $\beta = 2$, the gaps between two successive numbers in the larger base must be less than or equal to the gaps in the binary-base. $\Rightarrow t_k - t_1 \geq k - 1$.

Shifting speed

Example 1 For a 24-bit mantissa with all bits zero except the least significant bit, what is the maximum number of shifts required for each case of postnormalization.

Binary system: Radix = 2 and 23 shifts are required.

Hexadecimal system: Radix = 16 and 5 shifts are required.

Better accuracy is obtained with small base values and sophisticated round-off algorithms, while computational speed is associated with larger base values and crude round-off procedures such as truncation.

FP does not always obey the law!

A basic law of algebra is $(A + B = A) \Rightarrow B = 0$.

Example 2 In the IEEE single precision, if $A = 1.0 \times 2^{30}$ and $B = 1.0 \times 2^{-40}$ then $A + B = A$ while $B \neq 0$!

Example 3 In a decimal system with five digits after the point, check the associativity with $1.12345 \times 10^1 + 1.00000 \times 10^4 - 1.00000 \times 10^4$.

Solution: Given only five decimal digits, the result of

$$\begin{aligned} (1.12345 \times 10^1 + 1.00000 \times 10^4) - 1.00000 \times 10^4 \\ = 1.00112 \times 10^4 - 1.00000 \times 10^4 \\ = 1.12000 \times 10^1. \end{aligned}$$

However, $1.12345 \times 10^1 + (1.00000 \times 10^4 - 1.00000 \times 10^4) = 1.12345 \times 10^1 + 0 = 1.12345 \times 10^1$.

Associativity fails and the first answer lost three digits of significance.

Yet another loss

Example 4 With $A = 0.100000 \times 16^1$ and $B = 0.FFFFFFF \times 16^0$, what is $A - B$?

Solution:

$$\begin{array}{r} A = 0.1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \times 16^1 \\ B = 0.0 \quad F \quad F \quad F \quad F \quad F \times 16^1 \\ \hline A - B = 0.0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \times 16^1 = 0.1 \times 16^{-4}. \end{array}$$

The real answer is 0.1×16^{-5} .

Thus, the loss of significance (error) is $0.1 \times 16^{-4} - 0.1 \times 16^{-5} = 0.F \times 16^{-5} = 93.75\%$ of the correct result. Quite a large relative error!

We need to *guard* our digits.

Rounding

The rounding is a mapping from the real numbers to the machine representable numbers.

Number	∇	$\triangle$	T	A	RNU	RNE
+38.7	+38	+39	+38	+39	+39	+39
+38.5	+38	+39	+38	+39	+39	+38
+38.2	+38	+39	+38	+39	+38	+38
+38.0	+38	+38	+38	+38	+38	+38
-38.0	-38	-38	-38	-38	-38	-38
-38.2	-39	-38	-38	-39	-38	-38
-38.5	-39	-38	-38	-39	-39	-38
-38.7	-39	-38	-38	-39	-39	-39

Where is the nearest number?

Humans add $1/2$ of the *LSD* position of the desired precision to the *MSD* of the portion to be discarded.

For RNE, this gives the wrong result:

$$\begin{array}{r}
 38.5 \text{ X X X X} \leftarrow \text{Number to be rounded} \\
 0.5 \text{ 0 0 0 0} \leftarrow \text{Add 0.5} \\
 \hline
 39.0 \text{ X X X X} \leftarrow \text{Result} \\
 39 \qquad \qquad \leftarrow \text{Truncate}
 \end{array}$$

The *sticky bit* is the *OR* function of all the bits we want to check.

The *round bit* is the *MSB* of the discarded part.

1.		L	G	R	S
$\leftarrow$ desired precision $\rightarrow$					

Normalize then round

Left shift: *S* does not participate but *G* is shifted into the number and *R* into the old position of *G*.

Right shift: *S* and *R* guard bits are ORed into *S* (i.e., $L \rightarrow G$ and $G + R + S \rightarrow S$).

1.		L	G	S
$\leftarrow$ desired precision $\rightarrow$				
			a	

The proper action to obtain unbiased rounding-to-even (RNE) is:

L	G	S	Action	a
X	0	0	Exact result, no action.	0
X	0	1	Inexact result, but no action needed.	0
0	1	0	Tie with even significand, no action.	0
1	1	0	Tie with odd significand, round to nearest even.	1
X	1	1	Round to nearest by adding 1.	1

The sticky is important

Example 5 Let us see the importance of the sticky bit to Directed Upward Rounding when we round to the integer in the following two cases.

Case 1: No sticky bit is used;

$38.00001 \rightarrow 38$

$38.00000 \rightarrow 38$

Case 2: Sticky bit is used:

$38.00001 \rightarrow 39$ (sticky bit = 1)

$38.00000 \rightarrow 38$ (sticky bit = 0,
exact number).

When the sticky bit is one and we neglect using it, the result is incorrect.

Exceptions

The IEEE standard specifies five exceptional conditions that may arise during an arithmetic operation:

1. invalid operation, $(\infty - \infty, \infty \times 0, \sqrt{-3}, \dots)$
2. division by zero,
3. overflow,
4. underflow, and
5. inexact result.

Overflow and infinities

The overflow flag is raised whenever the magnitude of what would be the result exceeds **max** in the destination format.

When *traps* are disabled, the rounding mode and the sign of the intermediate result determine the final result:

	RNE	RZ	RP	RM
+ve	$+\infty$	+max	$+\infty$	+max
-ve	$-\infty$	-max	-max	$-\infty$

If the overflow trap is enabled when an overflow occurs, a value is delivered to the trap handler that allows the handler to determine the correct result.

Gradual underflow

The gradual underflow preserves an important mathematical property: if M is the set of representable numbers according to the standard then

$$\forall x, y \in M, \quad x - y = 0 \iff x = y.$$

Example 6 Assume that a system uses the single precision format of IEEE but without denormalized numbers. In such a system, what is the result of $1.0 \times 2^{-120} - 1.1111 \dots 1 \times 2^{-121}$?

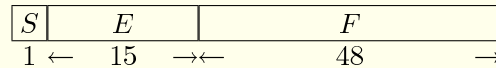
Solution: The exact result is obviously

$$\begin{array}{r} 1.000 \dots 0 \quad \times 2^{-120} \\ - \quad 0.111 \dots 1|1 \quad \times 2^{-120} \\ \hline 0.000 \dots 0|1 \quad \times 2^{-120} = 2^{-144} \end{array}$$

which is not representable in this system. Hence the returned result is zero although the two numbers are not equal.

Going for the speed: Cray

As before, the format ($\beta = 2$) consists of sign bit, biased exponent and fraction (mantissa):



where

S = sign bit of fraction
 E = biased exponent
 F = fraction

then

e = true exponent = E -bias
 f = true mantissa = $0.F$

A normalized nonzero number X is

$$X = (-1)^S \times 2^{E-bias} \times (0.F)$$

with a bias = $2^{14} = 16384$.

Cray: overflow and underflow

- **max** = $2^{2^{13}-1}(1 - 2^{-48}) = 2^{8191}(1 - 2^{-48})$
- Any result with an exponent containing two leading ones indicates overflow.
- **min** = $2^{-(2^{13})} \cdot 2^{-1} = 2^{-8193}$
- Any result with an exponent containing two leading zeros indicates underflow. (Flush to zero)
- Testing for over and underflow is done *before* normalization.
- Inputs are *not* tested.

Penalty for speeding

A number $s < \mathbf{min}$ can participate in computations:

- $(\mathbf{min} + s) - \mathbf{min} = s$, where s is 2^{-2} to 2^{-48} times $\mathbf{min}$, since $\mathbf{min} + s > \mathbf{min}$ *before postnormalization*.

The machine normalizes such results producing a number up to 2^{-48} smaller than $\mathbf{min}$. This number is not set to zero.

- $s \times Y = 0$ if the exponent of Y is not positive enough to bring $\exp(s) + \exp(Y)$ into range.
- $s \times Y = s \times Y$ if $\exp(s) + \exp(Y) \geq \exp(\mathbf{min})$.

Does it really matter?

- In 3D graphics animation, an error in a few pixels in a frame that flashes on the screen is tolerable.
- In general, audio and video signal processing tolerates a number of errors.
- However, if fast and inaccurate results are delivered in scientific or financial computations catastrophes might occur.