

Computer Systems and Architecture

Data Representation

Bart Meyers

University of Antwerp

October 29, 2012

Outline

Positive Numbers

Negative Numbers

Floating point Numbers

Exercises

Base 10 to base 2

Example. Convert $(23.375)_{10}$ to base 2.

Base 10 to base 2 (II)

Step 1. Remainder method: convert $(23)_{10}$ to base 2

$23 / 2 = 11$	Remainder	1	LSB
$11 / 2 = 5$		1	
$5 / 2 = 2$		1	
$2 / 2 = 1$		0	
$1 / 2 = 0$		1	MSB

$$(23)_{10} = (10111)_2$$

Base 10 to base 2 (III)

Step 2. Multiplication method: convert $(.375)_{10}$ to base 2

$$.375 \times 2 = \mathbf{0} \ .75$$

$$.75 \times 2 = \mathbf{1} \ .5$$

$$.5 \times 2 = \mathbf{1} \ .0$$

$$(.375)_{10} = (.011)_2$$

Step 3. Total: $(23.375)_{10} = (10111.011)_2$

Base 2 to base 10

- ▶ Number in base 2: $b_n b_{n-1} \dots b_1 b_0 . b_{-1} b_{-2} \dots b_{-m}$
- ▶ Value in base 10 = $\sum_{i=-m}^n b_i \cdot 2^i$
- ▶ **Weighted position code**
- ▶ Polynomial method

Base 2 to base 10 (II)

Example. Convert $(1010.01)_2$ to base 10.

$$\begin{aligned}(1010.01)_2 &= (1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2})_{10} \\ &= (8 + 0 + 2 + 0 + 0 + .25)_{10} \\ &= (10.25)_{10}\end{aligned}$$

Extensible to other bases

Conversion between base k and base 10:

- ▶ **Remainder method.** Divide by k .
- ▶ **Multiplication method.** Multiply by k .

- ▶ **Polynomial method.** $\sum_{i=-m}^n b_i \cdot k^i$

Bases 2, 4, 8, 16

Example. Convert $(10110110)_2$ to base 4, 8 and 16.

$$(10110110)_2$$

$$= (10)_2(11)_2(01)_2(10)_2 = (2)_4(3)_4(1)_4(2)_4 = (2312)_4$$

$$= (010)_2(110)_2(110)_2 = (2)_8(6)_8(6)_8 = (266)_8$$

$$= (1011)_2(0110)_2 = (B)_{16}(6)_{16} = (B6)_{16}$$

Negative numbers

- ▶ Signed magnitude
- ▶ One's complement
- ▶ Two's complement
- ▶ Excess (biased)

Signed magnitude

- ▶ First bit as sign
- ▶ **Example.**
 - ▶ $(+12)_{10} = (00001100)_2$
 - ▶ $(-12)_{10} = (10001100)_2$

One's complement

- ▶ Convert all zero's in 1's and 1's in zero's.
- ▶ **Example.**
 - ▶ $(+12)_{10} = (00001100)_2$
 - ▶ $(-12)_{10} = (11110011)_2$

Two's complement

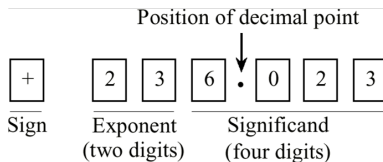
- ▶ Take the one's complement, and add 1.
- ▶ Only 1 representation for 0 !
- ▶ **Example.**
 - ▶ $(+12)_{10} = (00001100)_2$
 - ▶ $(-12)_{10} = (11110100)_2$

Excess (biased)

- ▶ Add *bias* to every number and convert it as a positive number.
- ▶ **Example:** Excess 128 (*bias* = 128)
 - ▶ $12 + 128 = 140$
 $(+12)_{10} = (10001100)_2$
 - ▶ $-12 + 128 = 116$
 $(-12)_{10} = (01110100)_2$

Floating point

- ▶ Wide range of numbers in a limited number of bits.
- ▶ **Example.** $+6.023 \cdot 10^{23}$



Normalization

- ▶ Different representations for the same number.
- ▶ **Example.** $+254 \cdot 10^0$
 - ▶ $25.4 \cdot 10^1$
 - ▶ $0.0254 \cdot 10^4$
 - ▶ $254000 \cdot 10^{-3}$
- ▶ Normalization: the dot is placed right of the leftmost digit.
 - ▶ $2.54 \cdot 10^2$
- ▶ Hidden bit: in base 2 the leftmost bit can be discarded
 - ▶ $1.1010 \rightarrow 1010$

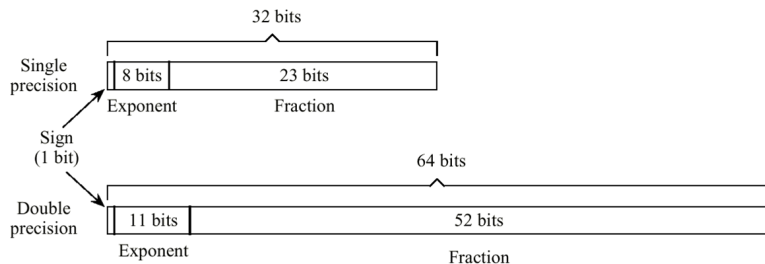
Conversion

Example. Convert $(9.375 \cdot 10^{-2})_{10}$ to base 2 (scientific notation).

- ▶ Fixed point: $(.09375)_{10}$
- ▶ Multiplication method: $(.00011)_2$
- ▶ Normalize: $.00011 = .00011 \cdot 2^0 = 1.1 \cdot 2^{-4}$

IEEE-754

- ▶ Sign bit
- ▶ Exponent: Excess 127 - 1023
- ▶ Number representation with hidden bit



IEEE-754 (II)

Example. Represent $(-12.625)_{10}$ in IEEE-754 Single precision format.

Step 1. Convert to base 2: $(-12.625)_{10} = (-1100.101)_2$

Step 2. Normalize: $(-1100.101)_2 = (-1.100101)_2 \cdot 2^3$

Step 3. Calculate exponent in excess 127:

$$3 + 127 = 130 \rightarrow (10000010)_2$$

1 1000 0010 100 1010 0000 0000 0000 0000

IEEE-754 (III)

Special cases (Single precision):

▶ +0 :

0 0000 0000 000 0000 0000 0000 0000 0000

▶ $+\infty$:

0 1111 1111 000 0000 0000 0000 0000 0000

▶ + NaN :

0 1111 1111 001 0010 0000 0100 1000 0000

▶ 2^{-128} (Denormalized) :

0 0000 0000 010 0000 0000 0000 0000 0000

Error

- ▶ Distance between 2 consecutive representable numbers
- ▶ Notation:
 - ▶ b = base
 - ▶ s = # significante digits
 - ▶ M = grootste exponent
 - ▶ m = kleinste exponent

Error (II)

- ▶ Number of representable bits (hidden bit):
 $2 \cdot ((M - m) + 1) \cdot b^s + 1$
- ▶ Biggest representable number:
 $b^M \cdot (1 - b^{-s})$
- ▶ Smallest representable number:
 $b^m \cdot b^{-1}$
- ▶ Biggest gap:
 $b^M \cdot b^{-s}$
- ▶ Smallest gap:
 $b^m \cdot b^{-s}$

Exercises

- ▶ <http://msdl.cs.mcgill.ca/people/hv/teaching/ComputerSystemsArchitecture/#CS4>