CPE 431/531

Chapter 3 – Arithmetic for Computers

Dr. Rhonda Kay Gaede

3.1 Introduction

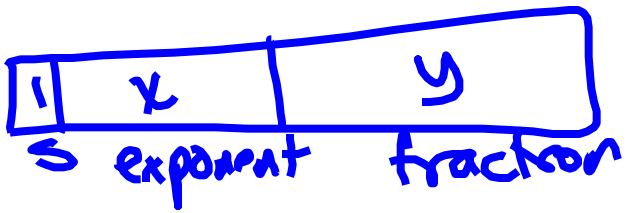
Bits are bits, what is important is how they are ___interpreted___.

- You may have an ___instruction___.
- You may have a ___signed number___(integer).
- You may have an ___unsigned number___(integer).
- You may have a ___floating-point numbers___.

3.5 Floating Point - Basics

- Floating-point numbers are represented in scientific notation
  1.38 E10
- Floating-point numbers use normalized representation.
- In general, floating-point numbers are of the form
  $(-1)^S \times 1.F \times 2^E$
  F fraction
  E exponent
- There is a tradeoff between range and precision
  - More y bits gives you more precision
  - More x bits gives you more range
- IEEE defines two types of floating-point numbers
  - Single Precision
  - Double Precision

Quad Precision
Half



3.5 Floating Point – More of the Story

IEEE 754 Floating Point Standard

Adding a bias to the exponent simplifies sorting

The leading one is implicit

Representation expanded

8 bits of exponent

-128 to +127

127 0111 1111

1 only choice -128 1000 0000

-1 1111 1111

0 0000 0000

+127 1111 1111

decimal

1.0

2.

3.

Example: Represent -0.75 in single and double precision

167

S = 1

$0.75_{10} = 0.11_2$

1.1×2^{-1}

8 exponent $-1 + 128 =$ 0111 1111

bias

23 fraction 10 . . . 0

1 0111 1111 10 . . . 0

0xBFC0 0000

3.5 Floating Point – More Examples

Example: What decimal number is represented by this single precision float? 0x4493 AB00

0 100 0100 | 1001 0011 1010 1011 0000 0000

S | EXP | FRACTION

S=0 number is positive

EXP = 137

- BIAS - 128

9

FRACTION = $2^{-3} + 2^{-6} + 2^{-7} + 2^{-8} + 2^{-10} + 2^{-12} + 2^{-14} + 2^{-15}$

Number = $(-1)^S (1.\text{FRACTION}) \times 2^{EXP}$

590.

single & double: 2^{n-1}

11

3.6 Subword Parallelism

Graphics systems originally used 8 bits to represent color and 8 bits to represent position.

Support for sound led to 16 bits of information.

Subword items have been supported for a long time in data transfer.

Graphics processing called for arithmetic on subword items.

Often the same operation is performed on groups (vectors) of data.

128 bit adders can handle (Data Level Parallelism)

1	128 bit operands
2	64 bit operands
4	32 bit operands
8	16 bit operands
16	8 bit operands

2008 Standard FP adds

half precision 16 bits

quad precision 128 bits

3.9 Fallacies and Pitfalls

$x + (y + z) = (x + y) + z$

Pitfall: Floating-point addition is not _associative_.

Because floating-point numbers are _approximations_ of real numbers and because computer arithmetic has _limited precision_, associativity does not hold for floating-point numbers.

$x = -1.5_{10} \times 10^{38}$, $y = 1.5_{10} \times 10^{38}$, $z = 1.0$

$x + y + z = 1.0$

$(x + y) + z = 0 + 1 = 1$

$x + (y + z) = 0 + 0 = 0$

Fallacy: Parallel execution strategies that work for _integer_ data types also work for _floating point_ data types.

Results may be _credible_ but not _identical_.