Computer Organization

COMP2120

Qi Zhao

January 31, 2024

Number Representation and arithmetic Part III

Multiplication of integers (2's complement)

The multiplication of unsigned binary integers

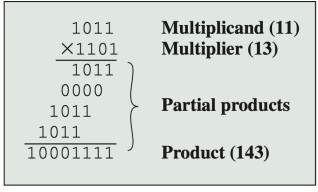

Figure 10.7 Multiplication of Unsigned Binary Integers

two n-bit binary integers, the product: at most $2n$ bits in length.

Multiplication of integers (2's complement)

- Double precision result after multiplication ($2n$ bits)
- Both multiplicand and multiplier are positive numbers: perform unsigned binary integer multiplication
- What about negative multiplicand and multiplier?
  - Calculate the partial product for each bit in the multiplier **except the sign bit**
  - Sign-extend the number to become a double precision number $2n$-bit. See range extension page lecture note Chapter 4.1 P24.
  - **Sign bit of multiplier:** if sign bit=0, do nothing; If sign bit=1, take two's complement of multiplicand and sign extend. Add this to the partial sum
  - Ignore carry out during addition.

Multiplication of integers (2's complement)

```
        10011 (-13)
x)      01011 (11)
--------------------
   1111110011  <- sign extended
   1111100110
   1110011000
--------------------
   1101110001 (-143)
```

```
        01011 (+11)
x)      10011 (-13)
-------------------------
   0000001011
   0000010110
   1101010000
-------------------------
   1101110001 (-143)
```

Floating-point Addition/subtraction

- Align the significands: make two exponents equal.

$$
\begin{aligned}
& 1.231 \times 10^2 + 4.561 \times 10^0 \\
=& 1.231 \times 10^2 + 0.04561 \times 10^2 \\
=& 1.27661 \times 10^2
\end{aligned}
$$

  Choose the number with a smaller exponent and shift its significand right a number of steps equal to the difference in exponents.
  Set the exponent of the result equal to the larger exponent.
- Perform add/sub on the significand and determine the sign of result
- Normalize the result, if necessary, truncate the signficand to the desired length.

Floating-point Addition/subtraction

- Basic phases for Addition/subtraction
  1. Check for zeros (change the sign of subtrahend)
  2. Align the significand
  3. Add (Subtract) the significant
  4. Normalize
  5. Rounding
- If we only have 8 digits to store the final result (we have more digits for the intermediate storage),

$$
\begin{aligned}
& 1.234567 \times 10^5 + 9.876543 \times 10^{-3} \\
=& 1.234567 \times 10^5 + 0.00000009876543 \times 10^5 \ \text{(after shifting)} \\
=& 1.23456709876543 \times 10^5 \ \text{(true sum)} \\
=& 1.234567 \times 10^5 \text{(after rounding and normalization)}
\end{aligned}
$$

- Rounding: Round to nearest, Round toward 0, etc.

Floating-point Addition/subtraction

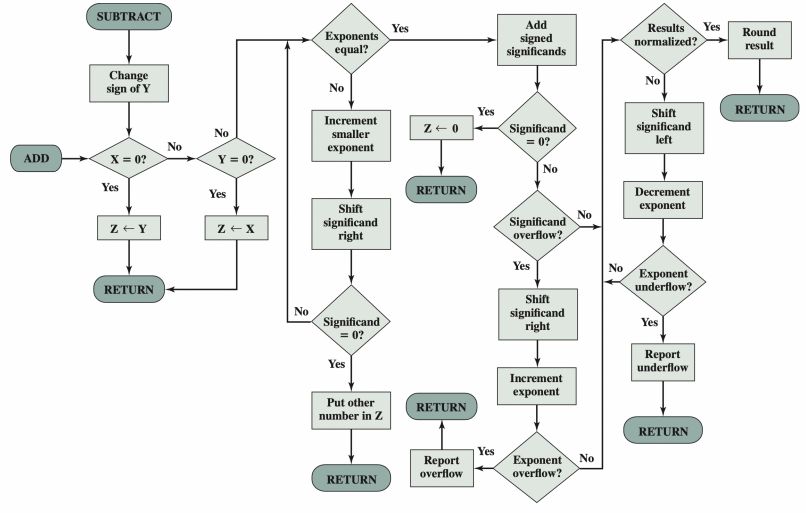

Figure 10.22 Floating-Point Addition and Subtraction ($Z \leftarrow X \pm Y$)

Approximation of floating-point arithmetic

- Not all numbers can be represented precisely, e.g. 0.2

$$0.2 = 0.0011\ldots_2$$

- Round-off error.
- Different order of operation may yield different results. Associative law may no longer holds. Decimal for illustration: use 10 digits in significand

$$\begin{aligned}
&(0.123 \times 10^{-10} + 0.123 \times 10^{-20}) - 0.123 \times 10^{-14} \\
=&(0.123 \times 10^{-10} + 0.000000000 \times 10^{-10)} - 0.123 \times 10^{-14} \\
=&0.123 \times 10^{-10} - 0.0000123 \times 10^{-10}
\end{aligned}$$

$$\begin{aligned}
&0.123 \times 10^{-10} + (0.123 \times 10^{-20}) - 0.123 \times 10^{-14}) \\
=&0.123 \times 10^{-10} + (0.000000123 \times 10^{-14} - 0.123 \times 10^{-14}) \\
=&0.123 \times 10^{-10} - 0.122999877 \times 10^{-14} \\
=&0.123 \times 10^{-10} - 0.000012299 \times 10^{-10}
\end{aligned}$$

Floating-point Multiplication

We want to calculate $(\pm)m_1 \times 2^{exp_1} \times (\pm)m_2 \times 2^{exp_2} = (\pm)m_1 \times m_2 \times 2^{exp_1+exp_2}$

- Exponent: excess-K representation, $e - K = exp$,
  exp is the exponent, e is the bit pattern (the value of bit pattern) in this representation.
- Add exponent and subtract bias k,

$$exp_1 = e_1 - K, exp_2 = e_2 - K$$

  We want: $exp_1 + exp_2$, bit pattern add: $e_1 + e_2 = exp_1 + exp_2 + 2K$ this represents $exp_1 + exp_2 + K$. Thus, we need to subtract K
- Multiply significand and determine sign of result
- Normalize and round

Floating-point Multiplication

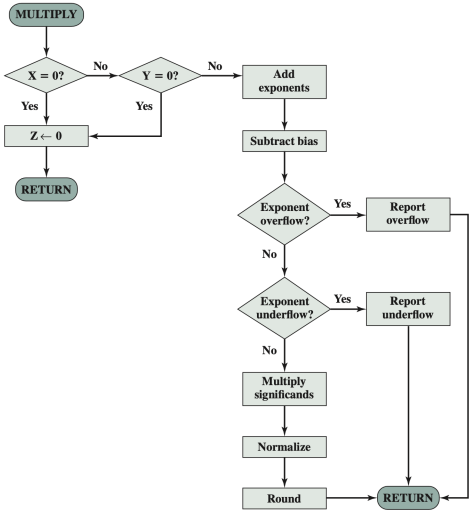

Figure 10.23 Floating-Point Multiplication ($Z \leftarrow X \pm Y$)

Floating-point Division

We want to calculate $(\pm)m_1 \times 2^{exp_1} \div (\pm)m_2 \times 2^{exp_2} = (\pm)(m_1/m_2) \times 2^{exp_1 - exp_2}$

- Subtract exponent and add bias (similar argument as multiplication)
- Divide significand and determine sign of result
- Normalize and round

Floating-point Division

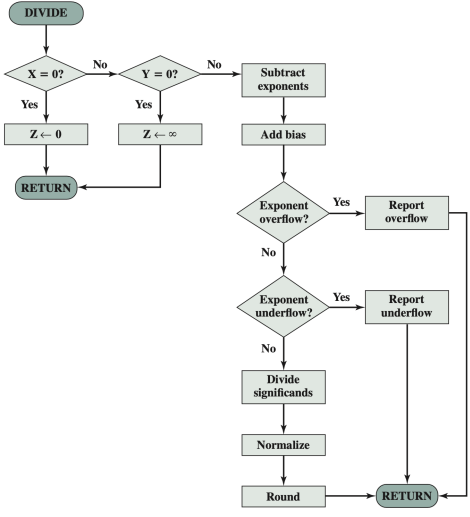

Figure 10.24 Floating-Point Division (Z ← X/Y)