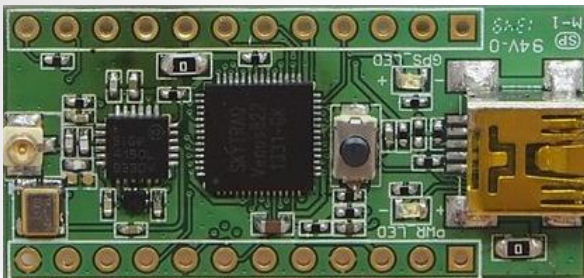


计算机组成原理

第三章 运算方法与运算器

3.7 浮点数加减运算



1

规格化浮点数的概念

- 由于浮点数是将数据的表示范围与精确度分别表示的数据表示方法，若不对浮点数的表示作出明确规定，同一个浮点数的表示就不唯一，
- 规格化浮点数是指把一个浮点数按指定的格式进行转换，
- 由于浮点数是将数据的表示范围与精确度分别表示的数据表示方法，若不对浮点数的表示作出明确规定，同一个浮点数的表示就不唯一，
- 以浮点数一般格式为例，规格化浮点数的尾数形式为：

00.1 Φ ... Φ 或 11.0 Φ ... Φ 。

2

浮点数规格化方法

- 当尾数结果为 $00.0\Phi\ldots\Phi$ 或 $11.1\Phi\Phi\Phi\Phi$ ，需要左规格化即将尾数向左移动，每移动一次，阶码减1，直到尾数形式为 $00.1\Phi\ldots\Phi$ 或 $11.0\Phi\ldots\Phi$ 。
- 当尾数的结果为 $01.\Phi\ldots\Phi$ 或 $10.\Phi\Phi\Phi\Phi$ ，表明尾数求和的结果 > 1 ，此时仅需要执行一次右移规格化，阶码加1，尾数形式即为 $00.1\Phi\ldots\Phi$ 或 $11.0\Phi\ldots\Phi$ 。

3

浮点数加减运算方法及步骤

设 $x = 2^{E_x} \cdot M_x$ $y = 2^{E_y} \cdot M_y$

则： $x + y = (2^{E_x - E_y} \cdot M_x + M_y) \times 2^{E_y}$ ($E_y \geq E_x$)

1)对阶

- 求阶差；
- 右移阶码小的浮点数的尾数并同步增加其阶码，直至两数阶码相等。

2)尾数加/减

尾数加/减运算 （用对阶后的尾数）

3)结果规格化

3

浮点数加减运算方法及步骤

4)舍入

右移规格化时可能丢失一些低位的数值位, 为提高精度, 可采取舍入的方法:

- 0 舍 1 入 : 若右移出的是1则在最低位加1;
- 恒置 1 : 只要数字位1被移掉,就将最后一位恒置成1。

5)溢出处理

浮点数的溢出标志: 阶码溢出

- 阶码上溢 : 阶码的符号位为 01
- 阶码下溢 : 阶码的符号位为 10

4

浮点数加减运算举例

例 设 $x = 2^{010} \times 0.11011011$ $y = 2^{100} \times (-0.10101100)$ 求 $x+y$

解：先用补码形式表示 x 和 y

$$[X]_{\text{补}} = 00\ 010 \quad , \quad 00.11011011$$

$$[Y]_{\text{补}} = 00\ 100 \quad , \quad 11.01010100$$

(1) 对阶

$$[\Delta E]_{\text{补}} = [Ex]_{\text{补}} + [-Ey]_{\text{补}} = 00010 + 11100 = 11\ 110$$

$\therefore \Delta E = -2$ x 的阶码 小于 y 的阶码

将 x 的尾数向右移动2位，同时阶码加 2，对阶后的 x 为：

$$[X]_{\text{补}} = 00\ 100 \quad , \quad 00.0011011011$$

4

浮点数加减运算举例

例 设 $x = 2^{010} \times 0.11011011$ $y = 2^{100} \times (-0.10101100)$ 求 $x+y$

2) 尾数运算

$$\begin{array}{r}
 00.00110110 \text{ } 11 \\
 + \underline{11.01010100} \\
 11.10001010 \text{ } 11
 \end{array}$$

3) 尾数规格化处理

分析发现，只左移一次即可达到规格化要求。规格化后的结果为：

$$[X+Y]_{\text{补}} = 00 \text{ } 011, 11.000101011$$

4) 舍入（0舍1入）

在结果尾数的最低位加1，最后的结果为：

$$[X+Y]_{\text{补}} = 00 \text{ } 011, 11.00010110 \quad X+Y = -0.11101010 \times 2^{011}$$

4

浮点数加减运算举例

例2 浮点数加减运算过程一般包括对阶、尾数运算、规格化、舍入和判溢出等步骤。设浮点数的阶码和尾数均采用补码表示，且位数分别为5位和7位（均含2位符号位）。若有两个数 $X=2^7 \times 29/32$ ， $Y=2^5 \times 5/8$ ，则用浮点加法计算 $X+Y$ 的最终结果是：

- A . 00111 1100010 B. 00111 0100010
C . 01000 0010001 D. 发生溢出

解题思路：

- $X = 2^{00111} \times 0.11101$ ； $Y = 2^{00101} \times 0.101$ ；对阶后大的阶码为00111
- 位数相加后的结果为：01.00010，
- 尾数需右移规格化，同时阶码加1后变成 01 000