计算机组成原理 Homework5 (9.28)

中国人民大学　信息学院　崔冠宇　2018202147

1. 设机器数的字长为 8 位 (含 1 位符号位), 分别写出下列各个二进制数真值的原码补码.

$$0, -0, 1111, -1111, 1101, -1101$$

解:

(1) 原码: 对于无负号的数, 符号位为 0; 对于有负号的数, 符号位为 1. 结果如下,

$[0]_{原} = 00000000$, $[-0]_{原} = 10000000$,

$[1111]_{原} = 00001111$, $[-1111]_{原} = 10001111$,

$[1101]_{原} = 00001101$, $[-1101]_{原} = 10001101$.

(2) 补码: 对于非负数, 直接补零; 对于负数, 将其绝对值的原码取反再加一. 结果如下,

$[0]_{补} = [-0]_{补} = 00000000$,

$[1111]_{补} = 00001111$, $[-1111]_{补} = 11110001$,

$[1101]_{补} = 00001101$, $[-1101]_{补} = 11110011$.

2. 请问 8 位二进制整数 01000111 和 10011010 的编码分别是原码、补码和无符号数时的真值是多少? (结果用十进制数表示)

解:

(1) $[01000111]_{原} = (1000111)_2 = (71)_{10}$,

$[01000111]_{补} = (1000111)_2 = (71)_{10}$,

$[01000111]_{无符号} = (1000111)_2 = (71)_{10}$.

(2) $[10011010]_{原} = (-11010)_2 = (-26)_{10}$,

$[10011010]_{补} = (-(\sim(10011010 - 1)))_2 = (-1100110)_2 = (-102)_{10}$,

$[10011010]_{无符号} = (10011010)_2 = (154)_{10}$.

3. 用补码运算计算下列各组数的和 (X+Y), 以及差 (X-Y), 结果用真值表示, 并判断是否溢出.

(1) X=-0.011010, Y=-0.010111　(2) X=0.110101, Y=-0.101011

(3) X=-0.011111, Y=0.001011　(4) X=0.1101101, Y=-0.100100

解:

(1) $[X]_补 = (1.100110)_补, [Y]_补 = (1.101001)_补, [-Y]_补 = (0.010111)_补.$

$[X+Y]_补 = [X]_补 + [Y]_补 = (1.001111)_补 = (-0.110001)_2$, 两操作数符号位相同, 和的符号位没变, 没有溢出;

$[X-Y]_补 = [X]_补 + [-Y]_补 = (1.111101)_补 = (-0.000011)_2$, 两操作数符号位不同, 没有溢出.

(2) $[X]_补 = (0.110101)_补, [Y]_补 = (1.010101)_补, [-Y]_补 = (0.101011)_补.$

$[X+Y]_补 = [X]_补 + [Y]_补 = (0.001010)_补 = (0.001010)_2$, 两操作数符号位不同, 没有溢出;

$[X-Y]_补 = [X]_补 + [-Y]_补 = (1.100000)_补 = (-0.100000)_2$, 两操作数符号位相同, 和的符号位变了, 发生溢出 (正溢).

(3) $[X]_补 = (1.100001)_补, [Y]_补 = (0.001011)_补, [-Y]_补 = (1.110101)_补.$

$[X+Y]_补 = [X]_补 + [Y]_补 = (1.101100)_补 = (-0.010100)_2$, 两操作数符号位不同, 没有溢出;

$[X-Y]_补 = [X]_补 + [-Y]_补 = (1.010110)_补 = (-0.101010)_2$, 两操作数符号位相同, 和的符号位没变, 没有溢出.

(4) $[X]_补 = (0.1101101)_补, [Y]_补 = (1.0111000)_补, [-Y]_补 = (0.1001000)_补.$

$[X+Y]_补 = [X]_补 + [Y]_补 = (0.0100101)_补 = (0.0100101)_2$, 两操作数符号位不同, 没有溢出;

$[X-Y]_补 = [X]_补 + [-Y]_补 = (1.0110101)_补 = (-0.1001011)_2$, 两操作数符号位相同, 和的符号位变了, 发生溢出 (正溢).

4. 设某计算机的字长为 16 位. 定点表示时, 数值 15 位, 符号位 1 位. 试求下列几种情况所能表示的数值的范围:

(1) 无符号数; (2) 用原码表示定点小数; (3) 用补码表示定点小数;

(4) 用原码表示定点整数; (5) 用补码表示定点整数.

解:

(1) 无符号数: 16 位均为有效数值位, 故为 $0 \sim 2^{16}-1$.

(2) 原码表示定点小数: 最高位为符号位, 其余 15 位为有效小数位, 故为 $-(1-2^{-15}) \sim (1-2^{-15})$.

(3) 补码表示定点小数: 补码相对原码多了一个数 -1, 故为 $-1 \sim (1-2^{-15})$.

(4) 原码表示定点整数: 最高位为符号位, 其余 15 位为有效整数位, 故为 $-(2^{15}-1) \sim (2^{15}-1)$.

(5) 补码表示定点整数: 补码相对原码多了一个数 -2^{15}, 故为 $-2^{15} \sim (2^{15}-1)$.

5. 某浮点数字长为 12 位; 阶码 4 位, 其中阶符 1 位; 尾数 8 位, 其中数符 1 位; 阶码的基数为 2; 阶码和尾数均用补码表示. 试求它能表示的:

(1) 最大正数及最小正数; (2) 绝对值最大的负数及绝对值最小的负数;

(3) 规格化的最小正数; (4) 规格化的绝对值最小的负数.

解: 根据浮点数的表示 $N = M \times R^E$, $R = 2$, 其中 M 是 8 位尾数 (补码表示), E 是 4 位阶码 (补码表示). 此时 M 的范围是 $-1 \sim (1 - 2^{-7})$, E 的范围是 $-2^3 \sim (2^3 - 1)$.

(1) 最大正数: 使尾数和阶码为最大正数, $N_{\text{最大正数}} = (1 - 2^{-7}) \times 2^7 = 2^7 - 1$.

最小正数: 使尾数为最小正数, 阶码为绝对值最大负数, $N_{\text{最小正数}} = 2^{-7} \times 2^{-8} = 2^{-15}$.

(2) 绝对值最大的负数: 使尾数为绝对值最大的负数, 阶码为最大正数, $N_{\text{绝对值最大的负数}} = -1 \times 2^7 = -2^7$.

绝对值最小的负数: 使尾数为绝对值最小的负数, 阶码为绝对值最大负数, $N_{\text{绝对值最小的负数}} = -2^{-7} \times 2^{-8} = -2^{-15}$.

(3) 规格化的最小正数: 使尾数为规格化的最小正数 ($(0.10\cdots0)_{\text{补}}$), 阶码为绝对值最大的负数,

$N_{\text{规格化的最小正数}} = 2^{-1} \times 2^{-8} = 2^{-9}$.

(4) 规格化的绝对值最小的负数: 使尾数为规格化的绝对值最小的负数 (1.01 $\cdots$ 1), 阶码为绝对值最大的负数,

$N_{\text{规格化的绝对值最小的负数}} = (-2^{-1} - 2^{-7}) \times 2^{-8} = -2^{-9} - 2^{-15}$.

6. (思考题, 上网百度答案) 最少几位二进制数据可以表示任意 5 位长的十进制正整数? IEEE754 单精度 (32 位), 其中阶码占多少位? 尾数多少位? 可以表示多少位 10 进制数? 双精度 (64 位), 其中阶码多少位? 尾数多少位? 可以表示多少位 10 进制数?

解:

(1) 5 位长十进制正整数的范围是 $1 \sim 99999$, $2^{16} = 65536$, $2^{17} = 131072$, 所以需要 17 位二进制数.

(2) IEEE754 单精度浮点数有 1 位符号位 S, 8 位阶码 E, 23 位尾数 M. 表示范围约为 $-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$, 有效位数一般是 7 位 (最少 6 位).

(3) IEEE754 双精度浮点数有 1 位符号位 S, 11 位阶码 E, 52 位尾数 M. 表示范围约为 $-1.798^{308} \sim 1.798 \times 10^{308}$, 有效位数一般是 16 位 (最少 15 位).