6.2 数的定点表示和浮点表示

- 一、定点表示
- 二、浮点表示
  - 1. 浮点数的表示形式
  - 2. 浮点数的表示范围
  - 3. 浮点数的规格化形式
  - 4. 浮点数的规格化
- 三、举例
- 四、IEEE 754 标准

二、浮点表示

- 为什么在计算机中要引入浮点数表示？
- 浮点表示的格式是什么？
- 尾数和阶码的基值必须是2吗？基值的影响？
- 表数范围与精度和哪些因素有关？
- 为什么要引入规格化表示?
- 目前浮点数表示格式的标准是什么？

二、浮点表示

- 为什么要引入浮点数表示
  - 编程困难，程序员要调节小数点的位置；
  - 数的表示范围小，为了能表示两个大小相差很大的数据，需要很长的机器字长；
    - 例如：太阳的质量是**$0.2*10^{34}$**克，一个电子的质量大约为**$0.9*10^{-27}$**克，两者的差距为**10^{61}**以上，若用定点数据表示：**$2^{x}>10^{61}$**，解的，**x>203**位。
  - 数据存储单元的利用率往往很低。

二、浮点表示 6.2

$N = S \times r^j$　　浮点数的一般形式

S 尾数　j 阶码　r 尾数的基值

计算机中 r 取 2、4、8、16 等

当 $r = 2$　　$N = 11.0101$

二进制表示

✓ $= 0.110101 \times 2^{\boxed{10}}$ 规格化数

$= 1.10101 \times 2^{1}$

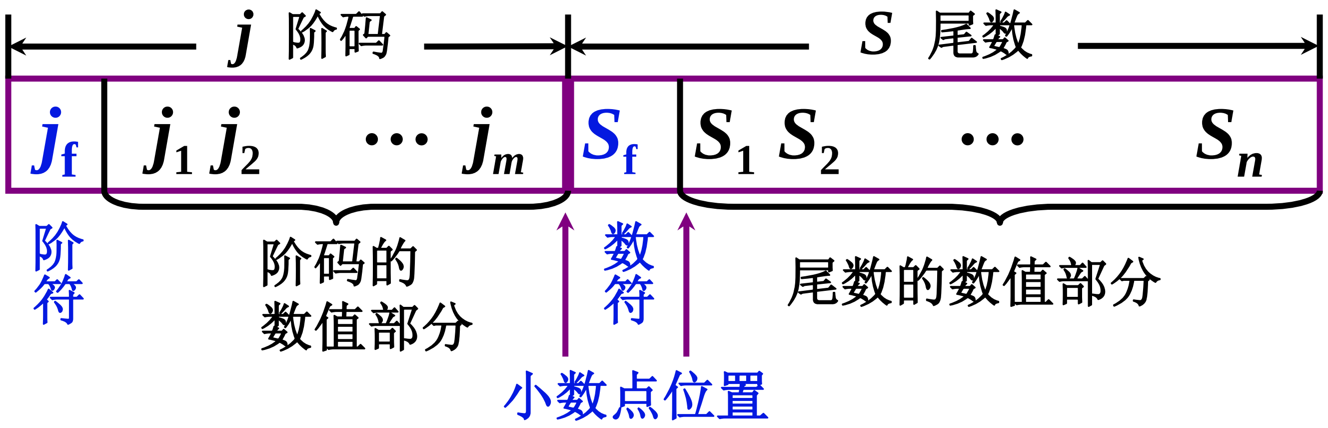
$= 1101.01 \times 2^{-10}$

✓ $= 0.00110101 \times 2^{100}$

计算机中　S　小数、可正可负

j　整数、可正可负

1. 浮点数的表示形式



S_f　代表浮点数的符号

n　其位数反映浮点数的精度

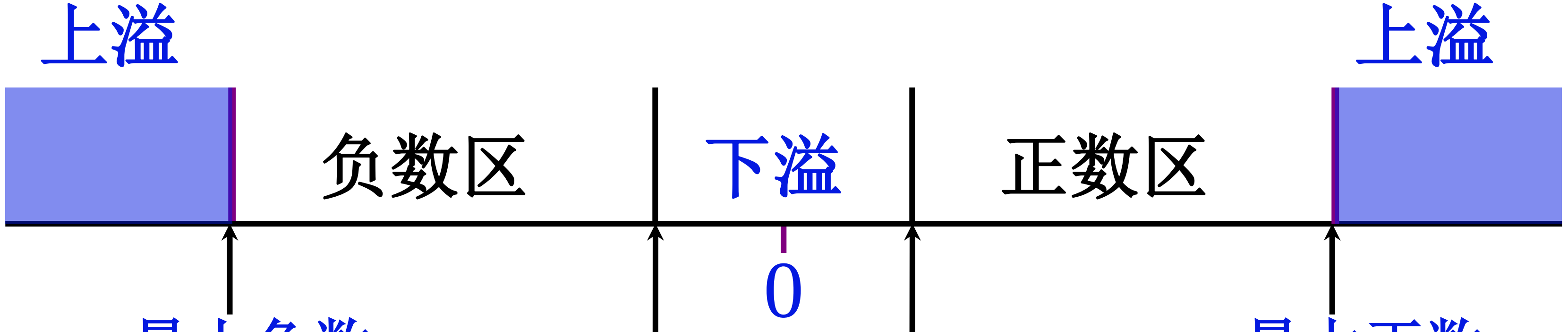
m　其位数反映浮点数的表示范围

j_f 和 m　共同表示小数点的实际位置

2. 浮点数的表示范围

上溢　阶码 > 最大阶码

下溢　阶码 < 最小阶码　按 **机器零** 处理



最小负数

$-2^{(2^m-1)} \times (1-2^{-n})$

$-2^{15} \times (1-2^{-10})$

最大负数

$-2^{-(2^m-1)} \times 2^{-n}$

$-2^{-15} \times 2^{-10}$

最小正数

$2^{-(2^m-1)} \times 2^{-n}$

$2^{-15} \times 2^{-10}$

最大正数

$2^{(2^m-1)} \times (1-2^{-n})$

$2^{15} \times (1-2^{-10})$

设 $m = 4$

$n = 10$