第 6 章　计算机的运算方法

6.1 无符号数和有符号数

6.2 数的定点表示和浮点表示

6.3 定点运算

6.4 浮点四则运算

6.5 算术逻辑单元

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

6.2 数的定点表示和浮点表示

小数点按约定方式标出

一、定点表示

$\boxed{S_f} \; \underbrace{\boxed{S_1S_2\cdots S_n}}_{\text{数值部分}}$ 或 $\boxed{S_f} \; \underbrace{\boxed{S_1S_2\cdots S_n}}_{\text{数值部分}}$

数符　　数符

小数点位置　　小数点位置

定点机	小数定点机	整数定点机
原码	$-(1-2^{-n}) \sim +(1-2^{-n})$	$-(2^n-1) \sim +(2^n-1)$
补码	$-1 \sim +(1-2^{-n})$	$-2^n \sim +(2^n-1)$
反码	$-(1-2^{-n}) \sim +(1-2^{-n})$	$-(2^n-1) \sim +(2^n-1)$

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

$N = S \times r^j$　　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

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

$= 1.10101 \times 2^{1}$

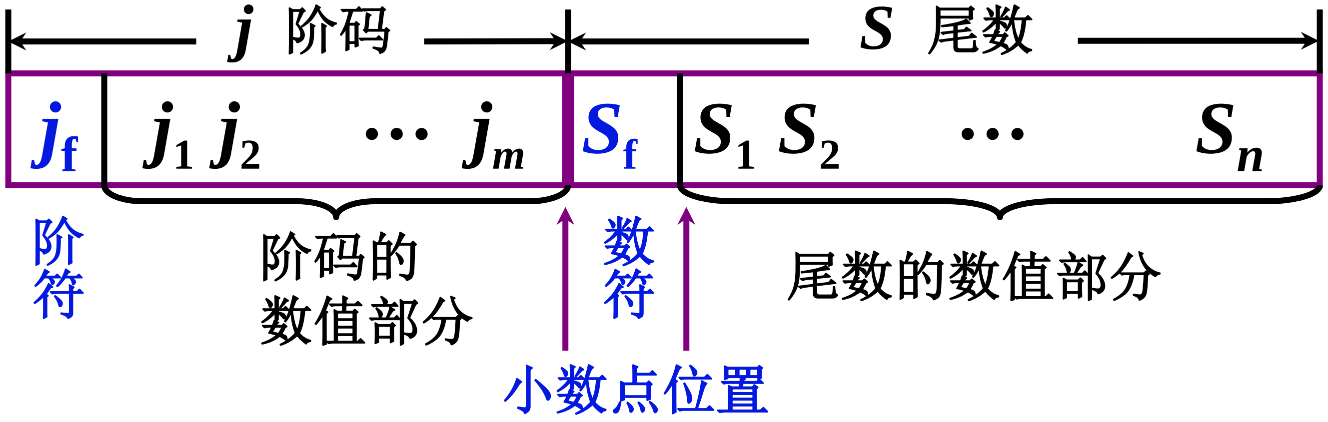
$= 1101.01 \times 2^{-10}$

$\checkmark = 0.00110101 \times 2^{100}$

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

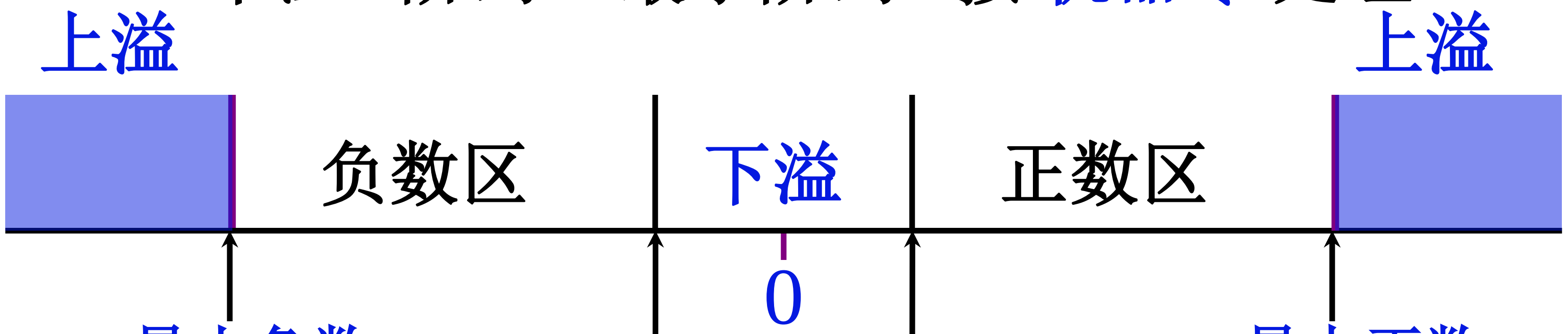
j_f 和 m 共同表示小数点的实际位置

2. 浮点数的表示范围

6.2

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

$-2^{-(2^m-1)}\times2^{-n}$

$-2^{-15}\times2^{-10}$

最小正数

$2^{-(2^m-1)}\times2^{-n}$

$2^{-15}\times2^{-10}$

最大正数

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

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

设 $m = 4$

$n = 10$

练习 6.2

设机器数字长为 24 位，欲表示±3万的十进制数，试问在保证数的最大精度的前提下，除阶符、数符各取1 位外，阶码、尾数各取几位？

解： $\because$ $2^{14} = 16384$ $2^{15} = 32768$

$\therefore$ 如果是定点数15 位二进制数可反映 ±3 万之间的十进制数

$$2^{\boxed{15}} \times 0.\underbrace{\times\times\times \ \cdots \ \times\times\times}_{n\text{位}}$$

$m = 4，5，6，\cdots$

满足 最大精度 可取 $m = 4$， $n = 18$

3. 浮点数的规格化形式

$r = 2$　　尾数最高位为 1

$r = 4$　　尾数最高 2 位不全为 0

$r = 8$　　尾数最高 3 位不全为 0

基数不同，浮点数的规格化形式不同

4. 浮点数的规格化

$r = 2$	左规	尾数左移 1 位，阶码减 1
	右规	尾数右移 1 位，阶码加 1
$r = 4$	左规	尾数左移 2 位，阶码减 1
	右规	尾数右移 2 位，阶码加 1
$r = 8$	左规	尾数左移 3 位，阶码减 1
	右规	尾数右移 3 位，阶码加 1

基数 r 越大，可表示的浮点数的范围越大

基数 r 越大，浮点数的精度降低

例如：设 $m = 4$，$n = 10$，$r = 2$ 　　6.2

尾数规格化后的浮点数表示范围

最大正数 $2^{+1111} \times 0.\underbrace{1111111111}_{10\text{个}1} = 2^{15} \times (1-2^{-10})$

最小正数 $2^{-1111} \times 0.1\underbrace{000000000}_{9\text{个}0} = 2^{-15} \times 2^{-1} = 2^{-16}$

最大负数 $2^{-1111} \times (-0.1\underbrace{000000000}_{9\text{个}0}) = -2^{-15} \times 2^{-1} = -2^{-16}$

最小负数 $2^{+1111} \times (-0.\underbrace{1111111111}_{10\text{个}1}) = -2^{15} \times (1-2^{-10})$

三、举例

例 6.13　将 $+\frac{19}{128}$ 写成二进制定点数、浮点数及在定点机和浮点机中的机器数形式。其中数值部分均取 10 位，数符取 1 位，浮点数阶码取 5 位（含1位阶符），尾数规格化。

解：　设 $x = +\frac{19}{128}$

二进制形式　　$x = 0.0010011$

定点表示　　$x = 0.0010011\,000$

浮点规格化形式　　$x = 0.1001100000 \times 2^{-10}$

定点机中　　$[x]_{原} = [x]_{补} = [x]_{反} = 0.0010011000$

浮点机中　　$[x]_{原} = 1, 0010;\ 0.1001100000$

$[x]_{补} = 1, 1110;\ 0.1001100000$

$[x]_{反} = 1, 1101;\ 0.1001100000$

例 6.14 将 –58 表示成二进制定点数和浮点数，并写出它在定点机和浮点机中的三种机器数及阶码为移码、尾数为补码的形式（其他要求同上例）。

6.2

解： 设 $x = -58$

二进制形式 $x = -111010$

定点表示 $x = -0000\,111010$

浮点规格化形式 $x = -(0.1110100000) \times 2^{110}$

定点机中	浮点机中
$[x]_{原} = 1, 0000111010$	$[x]_{原} = 0, 0110; 1. 1110100000$
$[x]_{补} = 1, 1111000110$	$[x]_{补} = 0, 0110; 1. 0001100000$
$[x]_{反} = 1, 1111000101$	$[x]_{反} = 0, 0110; 1. 0001011111$
	$[x]_{阶移、尾补} = 1, 0110; 1. 0001100000$

机器零 6.2

- 当浮点数 尾数为 0 时，不论其阶码为何值按机器零处理
- 当浮点数 阶码等于或小于它所表示的最小数 时，不论尾数为何值，按机器零处理

如 $m = 4$ $n = 10$

当阶码和尾数都用补码表示时，机器零为

×, × × × ×； 0. 0 0 ⋯ 0

（阶码 = −16） 1, 0 0 0 0； ×.×× ⋯ ×

当阶码用移码，尾数用补码表示时，机器零为

0, 0 0 0 0； 0. 0 0 ⋯ 0

有利于机器中“ 判 0 ” 电路的实现

四、IEEE 754 标准

S	阶码（含阶符）	尾　　数

数符　　小数点位置

尾数为规格化表示

非“0”的有效位最高位为“1”（隐含）

	符号位 *S*	阶码	尾数	总位数
短实数	1	8	23	32
长实数	1	11	52	64
临时实数	1	15	64	80