计算机组成原理与系统结构

第四章

运算方法与运算器

http://jpkc.hdu.edu.cn/computer/zcyl/dzkjdx/

杭州電子科技大学
HANGZHOU DIANZI UNIVERSITY

第 4 章 运算方法与运算器

- 4.1 定点数的加减运算及实现
- 4. 定点数的乘法运算及实现
- 4.3 定点数除法运算及
  实现
- 4.4 定点运算器的组成与结构
- 4. 浮点运算及运算器
- 4. 浮点运算器举例
- 本章小结

BACK

4.1　定点数的加减运算及实现

一　补码加减运算与运算器

二　机器数的移位运算

三　移码加减运算与判溢

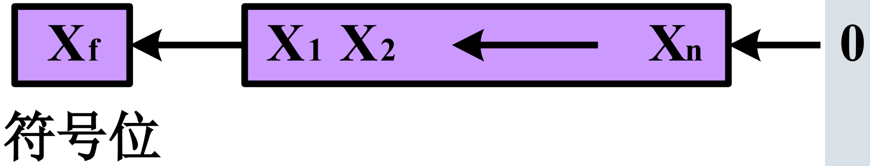
四　十进制加法运算

二、机器数的移位运算

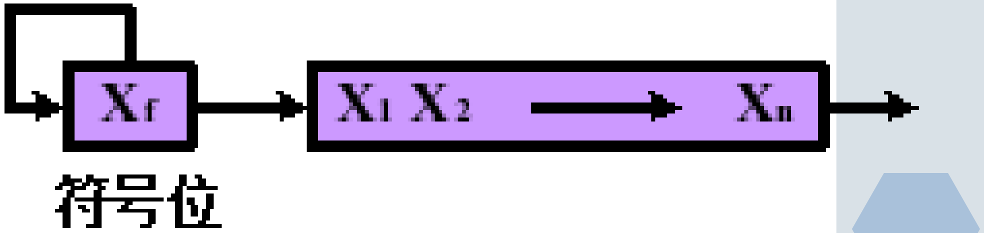
- 二进制数据（真值）每相对于小数点左移一位，相当于乘以 2；每相对于小数点右移一位，相当于除以 2。
- 计算机中的移位运算分为：
  - 1、逻辑移位：将移位的数据视为无符号数据，各数据位在位置上发生了变化，导致无符号数据的数值（无正负）放大或缩小。
  - 2、算术移位：将移位的数据视为带符号数据（机器数）。算术移位的结果，在数值的绝对值上进行放大或缩小，同时，符号位必须要保持不变。
  - 3、循环移位：所有的数据位在自身范围内进行左移或者右移，左移时最高位移入最低位，右移时最低位移入最高位。

补码的算术移位

- **算术左移：**符号位不变，高位移出，低位补 0 。
  - 为保证补码算术左移时不发生溢出，**移位的数据最高有效位必须与符号位相同。**
  - **在不发生溢出的前提下，**用硬件实现补码的算术左移时，直接将数据最高有效位移入符号位，不会改变机器数的符号。



- **算术右移：**符号位不变，低位移出，**高位正数补 0，负数补 1，**即高位补符号位。



补码的算术移位举例

- 例：设 X ＝ 0.1001, Y ＝－ 0.0101 ， 求
  - $[X]_{\text{补}}$＝ 0.1001
  - $[2X]_{\text{补}}$＝ 1.0010 （溢出）
  - $[X/2]_{\text{补}}$＝ 0.0100
  - $[Y]_{\text{补}}$＝ 1.1011
  - $[2Y]_{\text{补}}$＝ 1.0110
  - $[Y/2]_{\text{补}}$＝ 1.1101

三、移码加减运算与判溢

❖ 移码和移码计算

$[X]_移 = 2^n+X \qquad [Y]_移 = 2^n+Y \qquad -2^n \leqslant X \leqslant 2^n-1$

$[X]_移 + [Y]_移 = 2^n+X+2^n+Y = 2^n+(2^n+X+Y)$

$= 2^n + [X+Y]_移$

$[X]_移 \pm [-Y]_移 = 2^n + [X-Y]_移$

❖ 移码和补码混合计算

$[Y]_补 = 2^{n+1}+Y \quad (\mathrm{mod}\ 2^{n+1})$

$[X]_移 + [Y]_补 = [X+Y]_移$

三、移码加减运算与判溢

- 移码运算结果判溢：双符号位
  - [X]$_{移}$的最高符号位恒为 0
  - [Y]$_{补}$采用双符号位

$$\begin{array}{rcccccc} & 0 & X_f & X_1 & X_2 & \ldots\ldots & X_n \\ + & Y_f & Y_f & Y_1 & Y_2 & \ldots\ldots & Y_n \\ \hline & S_{f1} & S_{f2} & S_1 & S_2 & \ldots\ldots & S_n \end{array}$$

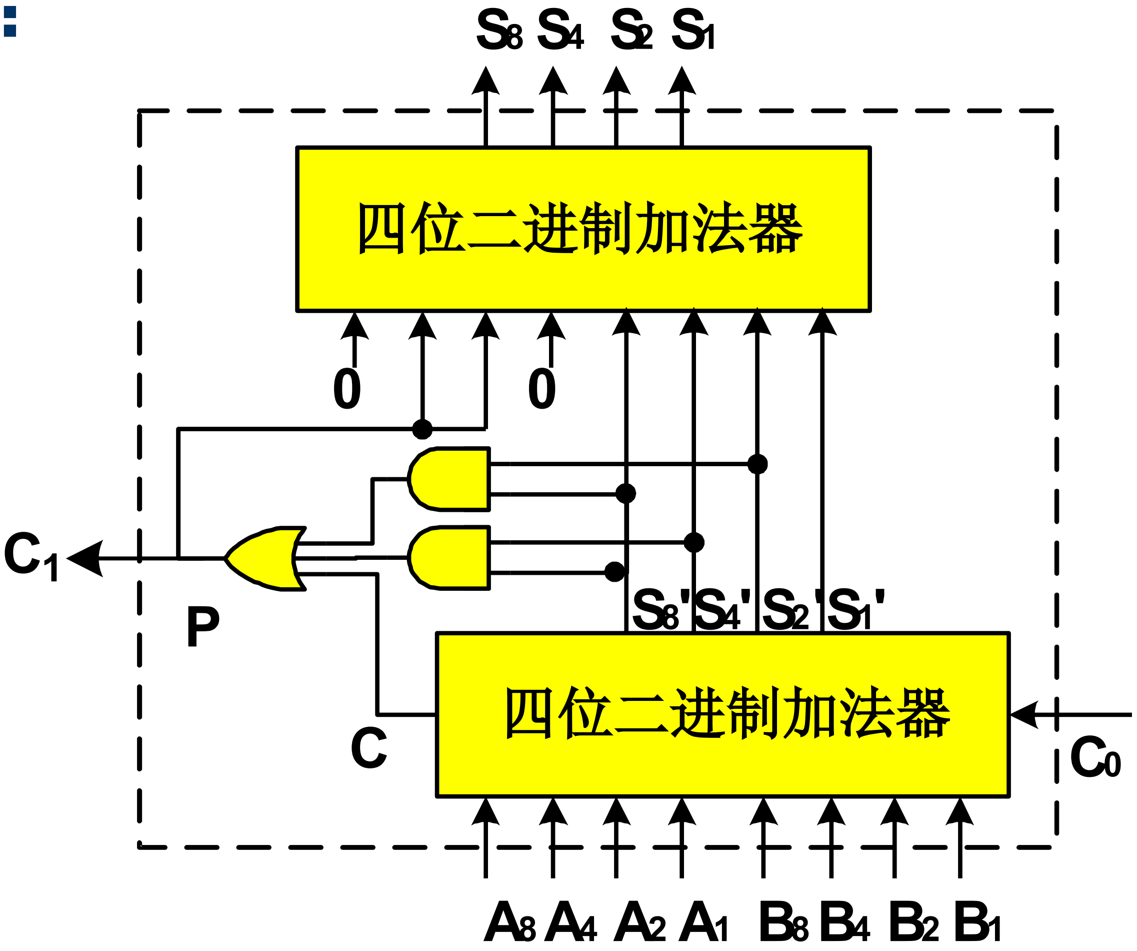
- 移码运算结果溢出的判断条件是：
  - 结果的最高符号位 $S_{f1}=1$ 时溢出， $S_{f1}=0$ 时结果正确。
    - $S_{f1}\ S_{f2}=10$ 时，结果正溢出；
    - $S_{f1}\ S_{f2}=11$ 时，结果负溢出。
  - 由于移码运算用于浮点数的阶码，当阶码运算结果正溢出时，浮点数上溢；当运算结果负溢出时，浮点数下溢，当作机器零处理。

四、十进制加法运算

- 十进制加法器实现：二进制加法器＋校正电路
- 当相加的两数之和 S>9 时，加 6 校正；当 S≤9 时，结果正确，不需校正。
- 对于 8421 BCD 码，校正条件为：
  - $S_8' S_4' S_2' S_1' \geq 1010$
  - 或 C=1

四、十进制加法运算

- 校正条件的逻辑式： $P= S_8'S_4' + S_8'S_2' + C$



The End !