

2.4 定点除法运算

2.4.1 原码除法

□ 规则：符号与数值分开计算

- 商的符号：同号为正,异号相除为负。可由异或实现。
- 商的数值部分由两数的绝对值相除获得。可看做是两个正数相除

例19 $x=0.1001, y=0.1011$.求 $x \div y$

$$\begin{array}{r}
0.1 \text{ } 1 \text{ } 0 \text{ } 1 \\
\underline{\hspace{1cm}} \\
0.1 \text{ } 0 \text{ } 1 \text{ } 1 \text{ } 0 \text{ } 1 \text{ } 0 \\
 \underline{0.\color{red}0} \text{ } 1 \text{ } 0 \text{ } 1 \text{ } 1 \\
 0.0 \text{ } 0 \text{ } 1 \text{ } 1 \text{ } 1 \text{ } 0 \\
- \underline{0.\color{red}0 \color{red}0} \text{ } 1 \text{ } 0 \text{ } 1 \text{ } 1 \\
 0.0 \text{ } 0 \text{ } 0 \text{ } 0 \text{ } 1 \text{ } 1 \text{ } 0 \\
- \underline{0.\color{red}0 \color{red}0 \color{red}0} \text{ } 1 \text{ } 0 \text{ } 1 \text{ } 1 \\
 0.0 \text{ } 0 \text{ } 0 \text{ } 0 \text{ } 1 \text{ } 1 \text{ } 0 \text{ } 0 \\
- \underline{0.\color{red}0 \color{red}0 \color{red}0 \color{red}0} \text{ } 1 \text{ } 0 \text{ } 1 \text{ } 1 \\
 0.0 \text{ } 0 \text{ } 0 \text{ } 0 \text{ } 0 \text{ } 0 \text{ } 0 \text{ } 1
\end{array}$$

- 得 $x \div y$ 的商 $q=0.1101$,
- 余数为 $r=0.00000001$

Q1: 被除数-除数, 不够减时怎么办?

Q2: 每一轮除数有什么特点?

Q3: 运算何时结束?

2.4.2 除法器的设计

1、恢复余数除法器

➤通过减法依次比较被除数和除数，判定商。

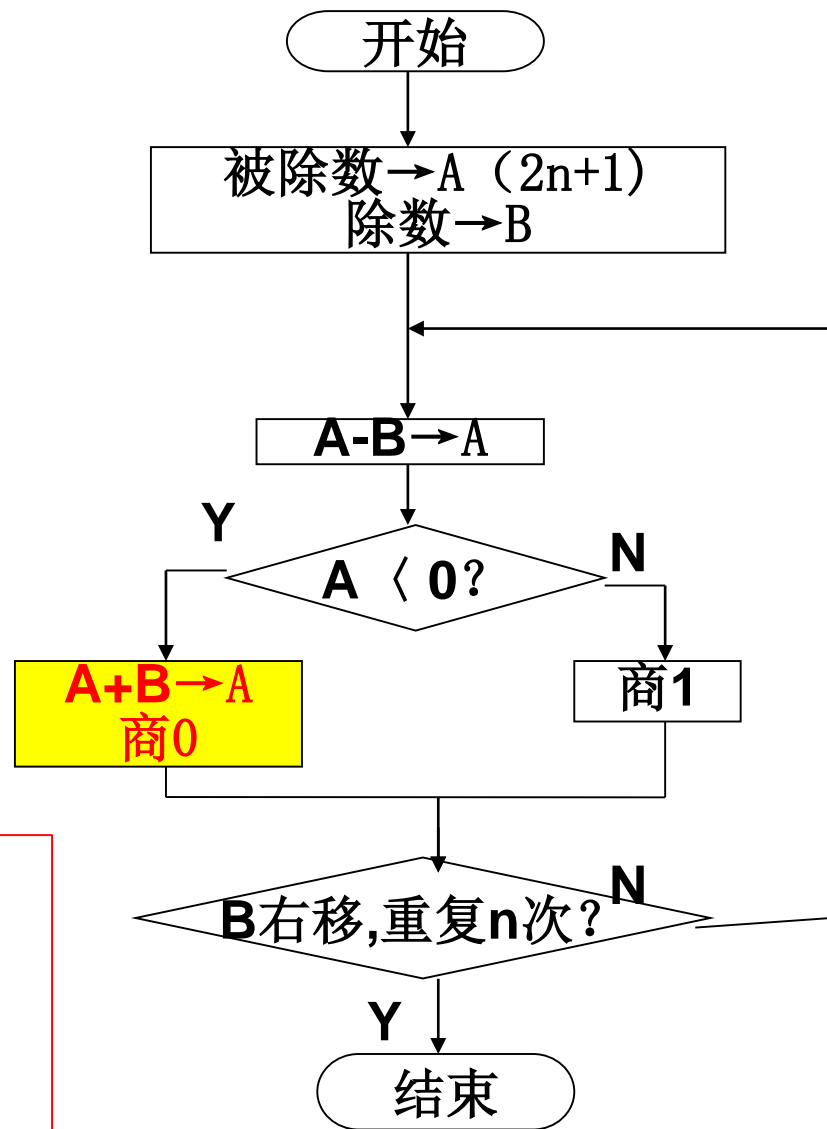
➤若不够减，则通过回加余数恢复A。

缺点：控制复杂

□ 除法运算要求：

1、确保参与运算的均是纯小数，
且被除数小于除数

2、被除数数值位扩充为除数数值位2倍



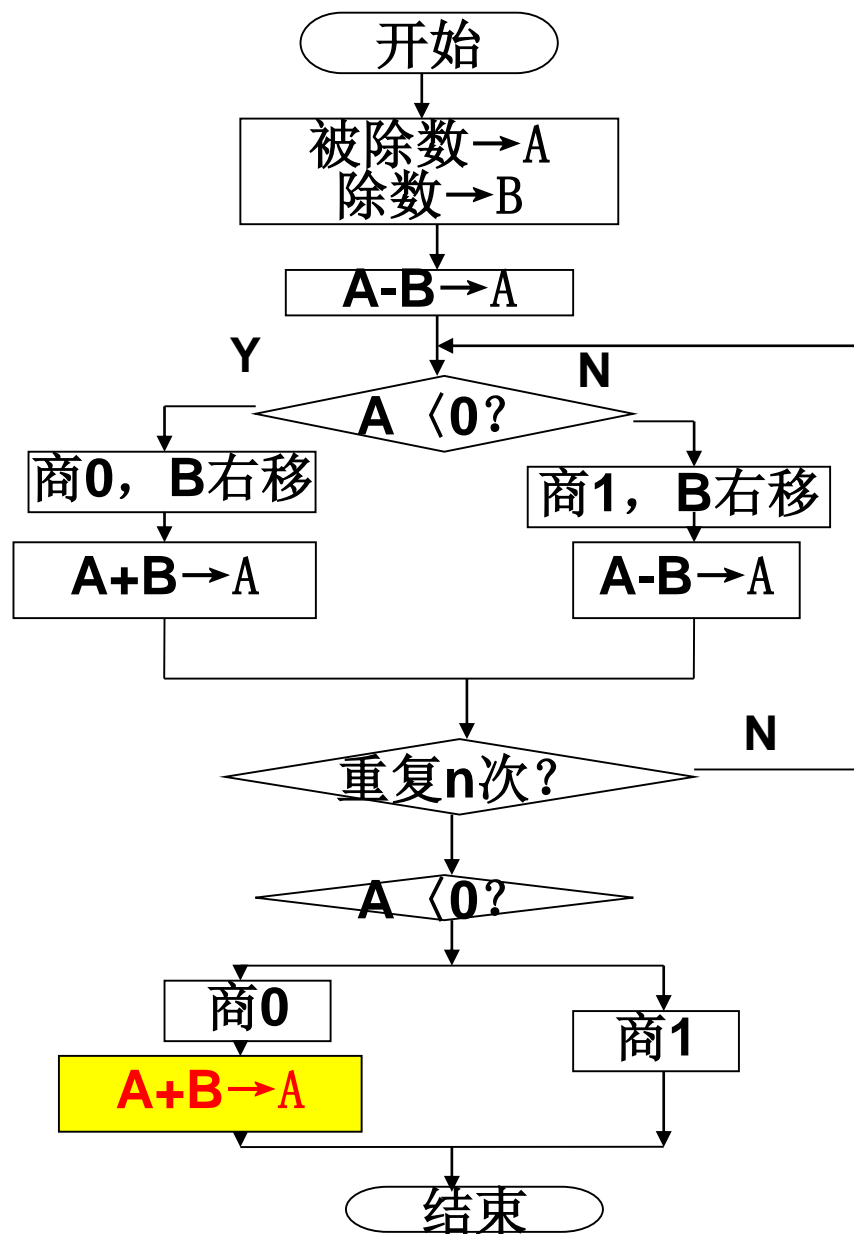
机器运算流程图

2、不恢复余数除法器

不恢复余数法

- 通过减法依次比较被除数和除数，判定商。
- 若不够减，不恢复余数，而根据余数的符号决定下一步操作
- 也称为加减交替法

最后一步的操作特殊处理，若商为0要恢复余数



例20 $x=0.101001$, $y=0.111$, 求 $q = x \div y$ 。

解: $[x]_{\text{补}} = 0.101001$

$[y]_{\text{补}} = 0.111$ $[-y]_{\text{补}} = 1.001$

每一轮余数的符号位进位即为商对应位的数值

$$\begin{array}{r}
 0.101001 \\
 +[-y]_{\text{补}} \quad 1.001 \\
 \hline
 0 \quad 1.110001
 \end{array}$$

$q=0$, 下步做加法, y 右移1位

$$\begin{array}{r}
 +[y]_{\text{补}} \quad 0.0111 \\
 \hline
 1 \quad 0.001101
 \end{array}$$

$q=1$, 下步做减法, y 再右移1位

$$\begin{array}{r}
 +[-y]_{\text{补}} \quad 1.11001 \\
 \hline
 0 \quad 1.111111
 \end{array}$$

$q=0$, 下步做加法, y 再右移1位

$$\begin{array}{r}
 +[y]_{\text{补}} \quad 0.000111 \\
 \hline
 1 \quad 0.000110
 \end{array}$$

$q=1$, 不做任何操作

商 $q = 0.101$

余数 $r = 0.000110$

$$X = 0.101001, \quad Y = 0.111 \quad \text{令 } X' = |X| = 0.101001 \quad Y' = |Y| = 0.111$$

$$\textcircled{1} S_f = S_x \oplus S_y = 0 \oplus 0 = 0$$

$$\textcircled{2} A = [X]_{\text{补}} = 0.101001 \quad B = [Y']_{\text{补}} = 0.111 \quad [-B] = [-Y']_{\text{补}} = 1.001$$

| | |
|----------------|-----|
| 0.101001 | 181 |
| + [-B] 1.001 | |
| 0.110001 | 0 |
| + [B] 0.0111 | |
| 1.001101 | 1 |
| + [-B] 1.11001 | |
| 0.111111 | 0 |
| + [B] 0.000111 | |
| 1.000010 | 1 |

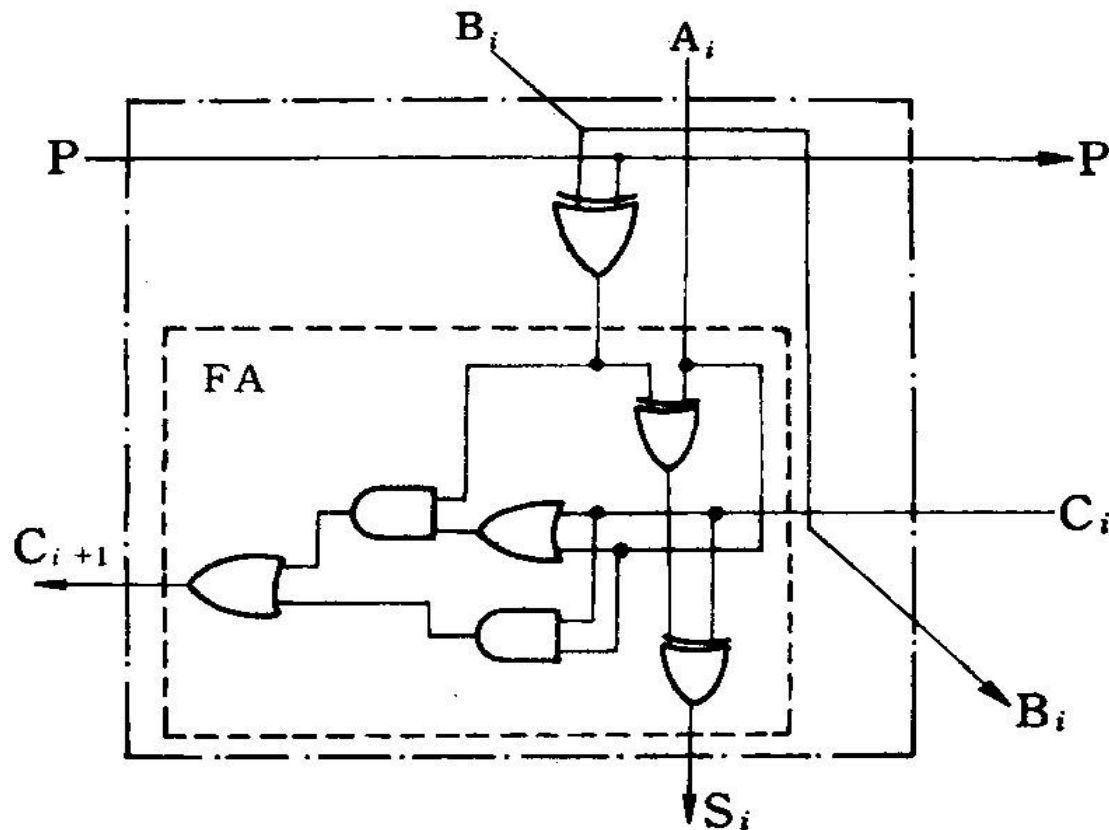
↑ 高
↓ 低

$$\therefore |Q| = 0.101$$

$$\textcircled{3} X \div Y \rightarrow Q = +0.101 \quad r = 0.000110$$

不恢复余数阵列除法器核心部件

——可控加法/减法 (CAS)



共有四个输入和四个输出。
其中 $P=0$ ，做加法；
 $P=1$ ，做减法；

考虑最大的信号时延，则对于 C_i 的输入 C_{i+1} 的时延为： $3T$

4位不恢复余数阵列除法器逻辑原理图

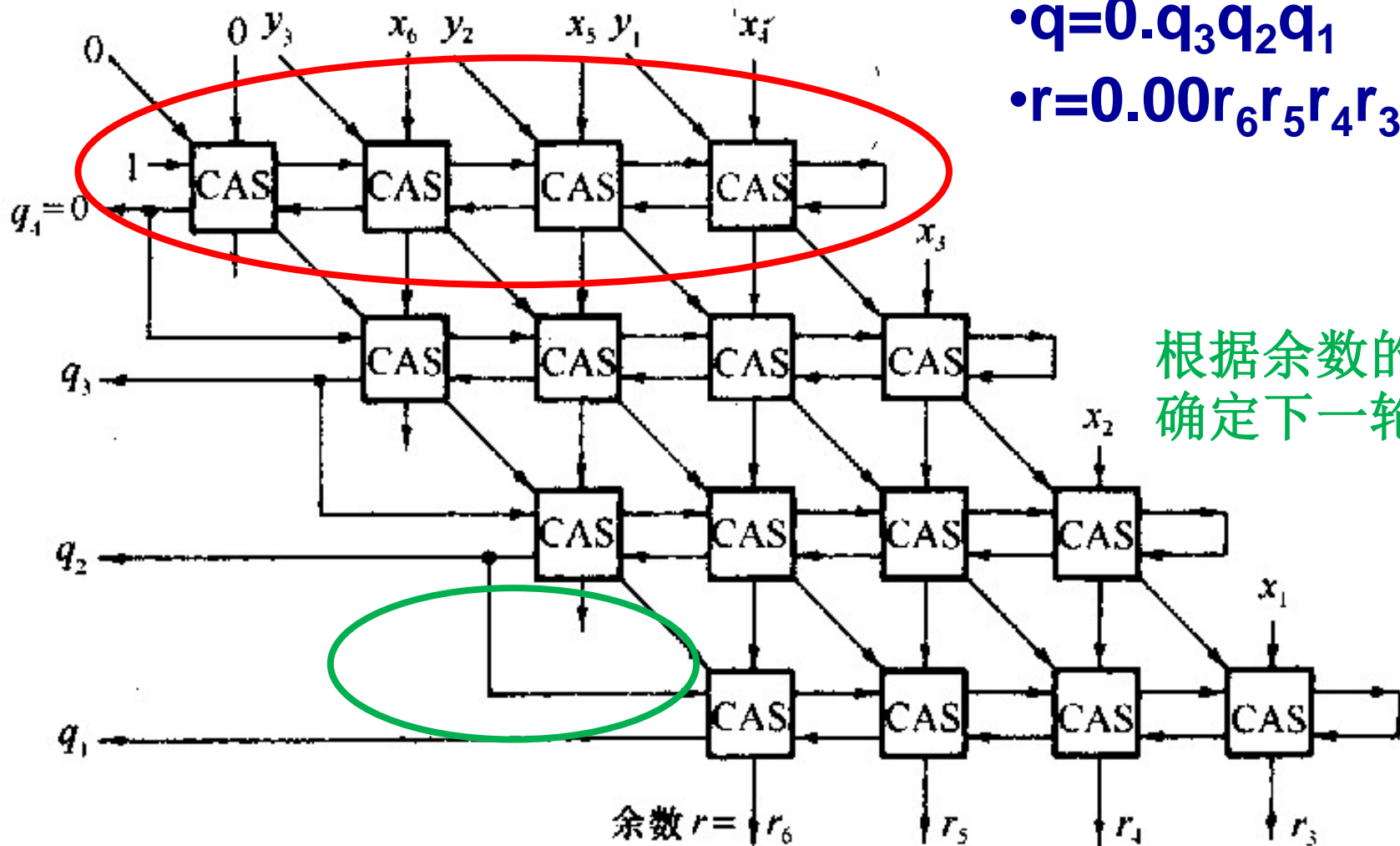
流程图第1步: $X-Y \rightarrow X$

$$\bullet x = 0.x_6x_5x_4x_3x_2x_1$$

$$\bullet y = 0.y_3y_2y_1$$

$$\bullet q = 0.q_3q_2q_1$$

$$\bullet r = 0.00r_6r_5r_4r_3$$



注意：

- ◆ 被除数（数值部分，即尾数）长度
= $2 * \text{除数长度}$
- ◆ 送入不恢复余数阵列除法器计算时，必有被除数 $<$ 除数。若原始数据不满足，则通过比例因子进行处理。
- ◆ $2n$ 位除以 n 位的不恢复余数阵列除法器，含有 $(n+1)^2$ 个CAS单元。考虑最大信号延迟，除法执行时间为： $3 * (n+1)^2 T$