中山大學 软件工程学院
SUN YAT-SEN UNIVERSITY SCHOOL OF SOFTWARE ENGINEERING

计算机组成原理

授课老师：吴炜滨

大纲

- 定点运算
  - 除法运算
    - 笔算除法的分析
    - 笔算除法的改进
    - 原码的除法运算

笔算除法的分析

- $x = -0.1011 \quad y = 0.1101$ 求 $x \div y$

$$
\begin{array}{r@{\,}l}
 & \;0.1101 \\
0.1101 \,\big) & 0.10110 \\
 & 0.01101 \\ \hline
 & 0.010010 \\
 & 0.001101 \\ \hline
 & 0.00010100 \\
 & 0.00001101 \\ \hline
 & 0.00000111
\end{array}
$$

$x \div y = -0.1101$ 商符心算求得

余数 0.00000111

- 商符单独处理
- 心算上商
- 余数不动低位补“0”，
  减右移一位的除数
- 上商位置不固定

笔算除法的改进

笔算除法

- 商符单独处理
- 心算上商
  - 比较余数和右移一位的除数的大小
  - 余数<除数：上商0
  - 余数≥除数：上商1
- 余数 不动 低位补“0”，减右移一位 的除数
- 2 倍字长加法器
- 上商位置 不固定

机器除法

- 符号位异或形成
- 根据减法结果上商
  - 左移一位的余数的绝对值减除数的绝对值
  - 差< 0，上商 0
  - 差≥ 0，上商 1
- 余数 左移一位 低位补“0”，减 除数
- 1 倍字长加法器
- 在寄存器 最末位上商，每次上商完左移一位

大纲

- 定点运算
  - 除法运算
    - 原码的除法运算
      - 运算规则
        - 恢复余数法
        - 加减交替法（不恢复余数法）
      - 硬件配置
      - 控制流程

运算规则

- 以小数为例
  - 整数除法过程相同，将小数点改为逗号即可

$$[x]_{原} = x_0.x_1x_2\cdots x_n \qquad [y]_{原} = y_0.y_1y_2\cdots y_n$$

$$[\frac{x}{y}]_{原} = (x_0 \oplus y_0).\frac{x^*}{y^*}$$

- 式中
  - $x^* = 0.x_1x_2\cdots x_n$ 为 x 的绝对值
  - $y^* = 0.y_1y_2\cdots y_n$ 为 y 的绝对值
  - 商的符号位单独处理 $x_0 \oplus y_0$
  - 数值部分为绝对值相除$\frac{x^*}{y^*}$

运算规则

$$[\frac{x}{y}]_{原} = (x_0 \oplus y_0).\frac{x^*}{y^*}$$

■ 约定

- 小数定点除法 $x^* < y^*$
  - 小数定点机，数据绝对值< 1
- 整数定点除法 $x^* \geq y^*$
  - 整数定点机，数据绝对值 ≥ 1
- 被除数不等于 0
  - 结果总为0，无需经过除法运算，直接利用判零电路即可得结果
- 除数不能为 0
  - 结果为无穷大，不能在机器中表示
- 商的位数与操作数的位数相同

恢复余数法

- 设机器字长为5位（含1位符号位），$x = -0.1011$，$y = -0.1101$，求 $[\frac{x}{y}]_原$

解：　$[x^*]_原 = 0.1011$　　$[y^*]_原 = 0.1101$

$[y^*]_补 = 0.1101$　　$[-y^*]_补 = 1.0011$

恢复余数法

- 设机器字长为5位（含1位符号位），$x=-0.1011$，$y=-0.1101$，求 $[\frac{x}{y}]_{原}$

$[x^*]_{原}=0.1011$ $[y^*]_{原}=0.1101$ $[y^*]_{补}=0.1101$ $[-y^*]_{补}=1.0011$

数值部分

被除数（余数）	商	说　明
0.1011	0.0000	
+ 1.0011		$+[-y^*]_{补}$
1.1110	0.0000	余数为负，上商 0
+ 0.1101		恢复余数　$+[y^*]_{补}$
0.1011	0.0000	恢复后的余数
1.0110	0.000	← 1
+ 1.0011		$+[-y^*]_{补}$
0.1001	0.0001	余数为正，上商 1
1.0010	0.001	← 1
+ 1.0011		$+[-y^*]_{补}$

逻辑左移

逻辑左移

恢复余数法

	被除数（余数）	商	说　明
	0.0101	0.0011	余数为正，上商 1
逻辑左移	$\boxed{0.1010}$	0.011	← 1
	+ 1.0011		$+[-y^*]_{补}$
	1.1101	0.0110	余数为负，上商 0
	+ 0.1101		恢复余数 $+[y^*]_{补}$
	$\boxed{0.1010}$	0.0110	恢复后的余数
逻辑左移	1.0100	0.110	← 1
	+ 1.0011		$+[-y^*]_{补}$
	0.0111	0.1101	余数为正，上商 1

$$\frac{x^*}{y^*} = 0.1101$$

真正余数由最终余数乘上2^{-n}所得：0.00000111

恢复余数法

- 设机器字长为5位（含1位符号位），$x = -0.1011$，$y = -0.1101$，求 $[\frac{x}{y}]_原$

$$[x]_原 = 1.1011 \quad [y]_原 = 1.1101$$

- 商符

$$x_0 \oplus y_0 = 1 \oplus 1 = 0$$

- 商值由两数绝对值相除而得

$$\frac{x^*}{y^*} = 0.1101$$

- 真正余数由最终余数乘上2^{-n}所得：0.00000111

$$\therefore \ [\frac{x}{y}]_原 = 0.1101 \qquad 余数：0.00000111$$

恢复余数法

■ 特点

- 每次上商时，减除数
  - 余数为正：上商 1
  - 余数为负：上商 0，恢复余数
  - 余数逻辑左移1位，与除数进行比较，准备下次上商
- 第一次上商判溢出
  - 小数定点机，第一次上商为1，发生溢出
- 上商$n+1$次
- 移位n次

加减交替法（不恢复余数法）

- 恢复余数法运算规则
  - 每次上商时，减除数
    - 余数 $R_i > 0$，上商 “1”
      - 余数逻辑左移1位，减除数，进行下次上商判断：$2R_i - y^*$
    - 余数 $R_i < 0$，上商 “0”
      - 恢复余数：$R_i + y^*$，再逻辑左移1位，减除数，进行下次上商判断：$2(R_i + y^*) - y^* = 2R_i + y^*$
- 不恢复余数法运算规则
  - 第一次上商时，减除数
    - 余数 $R_i > 0$，上商 “1”
      - 下次判断上商时余数：$2R_i - y^*$
    - 余数 $R_i < 0$，上商 “0”
      - 下次判断上商时余数：$2R_i + y^*$
  - 加减交替

加减交替法（不恢复余数法）

- 已知机器字长为5位（含1位符号位），$x = -0.1011$，$y = -0.1101$，求$[\frac{x}{y}]_{原}$

解：　$[x^*]_{原} = 0.1011$　　$[y^*]_{原} = 0.1101$

$[y^*]_{补} = 0.1101$　　$[-y^*]_{补} = 1.0011$

解:

$[x^*]_{原} = 0.1011$

$[y^*]_{原} = 0.1101$

$[y^*]_{补} = 0.1101$

$[-y^*]_{补} = 1.0011$

逻辑左移

逻辑左移

逻辑左移

逻辑左移

被除数（余数）	商	说　　明
0.1011	0.0000	
+1.0011		$+[-y^*]_{补}$
1.1110	0.0000	余数为负，上商 0
1.1100	0.000	← 1
+0.1101		$+[y^*]_{补}$
0.1001	0.0001	余数为正，上商 1
1.0010	0.001	← 1
+1.0011		$+[-y^*]_{补}$
0.0101	0.0011	余数为正，上商 1
0.1010	0.011	← 1
+1.0011		$+[-y^*]_{补}$
1.1101	0.0110	余数为负，上商 0
1.1010	0.110	← 1
+0.1101		$+[y^*]_{补}$
0.0111	0.1101	余数为正，上商 1

加减交替法（不恢复余数法）

■ 设机器字长为5位（含1位符号位），$x = -0.1011$，$y = -0.1101$，求 $[\frac{x}{y}]_{原}$

$$[x]_{原} = 1.1011 \quad [y]_{原} = 1.1101$$

- 商符

$$x_0 \oplus y_0 = 1 \oplus 1 = 0$$

- 商值由两数绝对值相除而得

$$\frac{x^*}{y^*} = 0.1101$$

- 真正余数由最终余数乘上2^{-n}所得：0.00000111

$\therefore \ [\frac{x}{y}]_{原} = 0.1101$ 余数：0.00000111

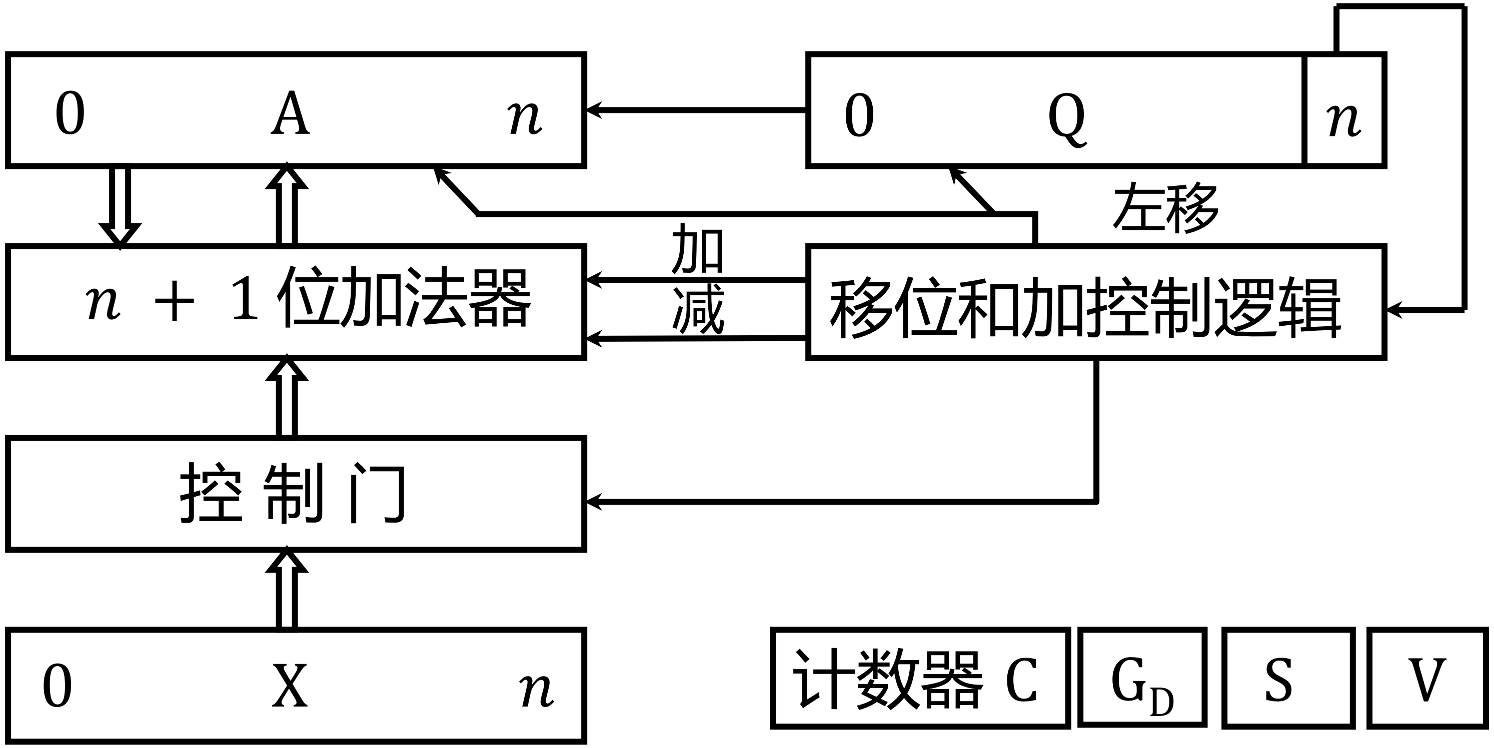
加减交替法（不恢复余数法）

特点

- 第一次上商时，减除数
  - 余数 $R_i > 0$，上商 “1”
    - 下次判断上商时余数：$2R_i - y^*$
  - 余数 $R_i < 0$，上商 “0”
    - 下次判断上商时余数：$2R_i + y^*$
- 第一次上商判溢出
  - 小数定点机，第一次上商为1，发生溢出
- 上商 $n+1$ 次
- 移位 n 次，加 $n+1$ 次
- 用移位的次数判断除法是否结束

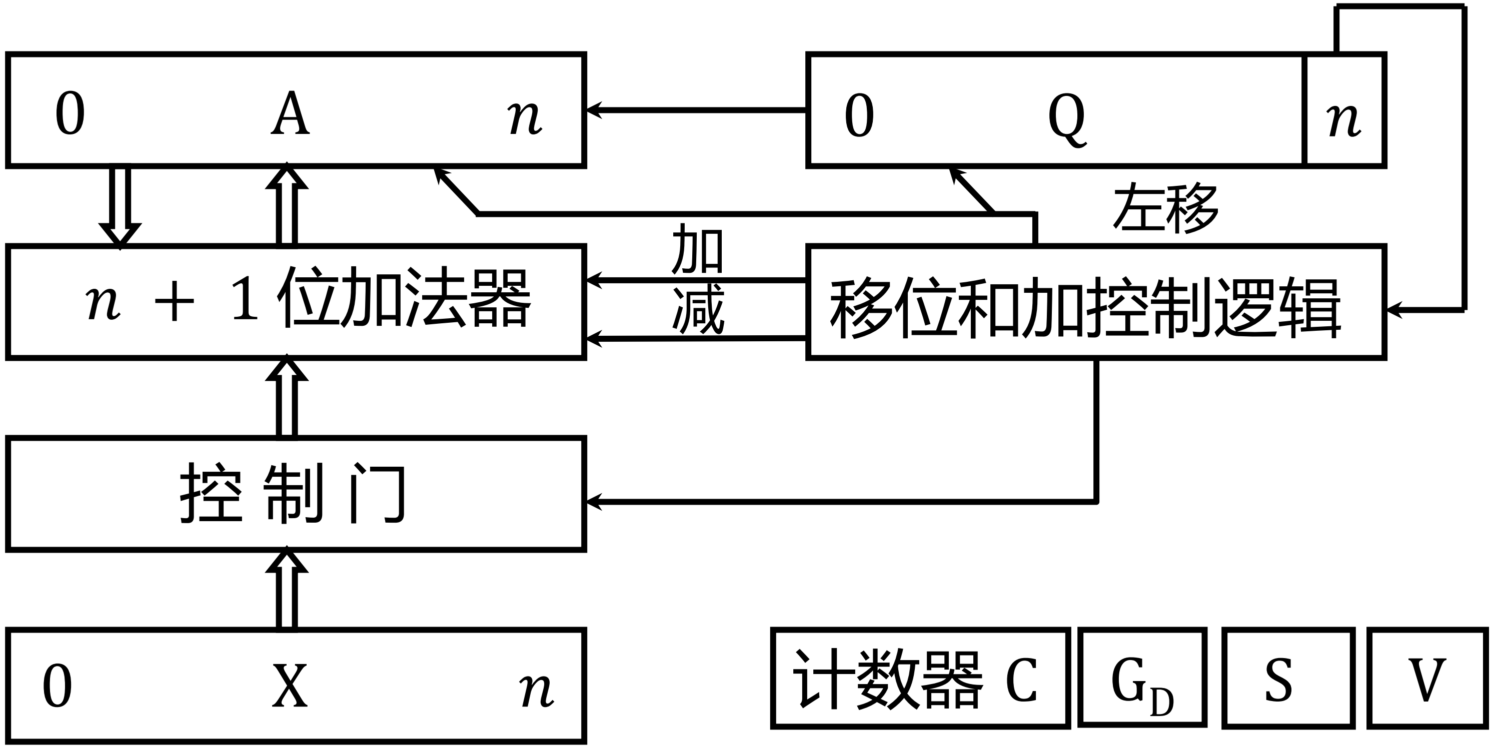
原码加减交替除法硬件配置

- 寄存器A、X、Q 、加法器均 $n+1$ 位
  - A：被除数的原码、余数
  - X：除数的原码
  - Q（MQ）：商的原码
- 用 Q_n 控制加减交替
  - 左移一位
  - Q_n=1做减法
  - Q_n=0做加法



原码加减交替除法硬件配置

- 计数器C
  - 计数器值=移位次数=数值部分位数=n
  - 每移位一次，计数器值减1
- S：商符
  - 值=被除数和除数的符号位进行异或
- G_D：除法标志
- V：溢出标志



原码加减交替除法控制流程

- 准备
  - Q清零准备接收商，被除数原码 → A，除数原码 → X，除数位数n → C
- 求商符
  - $\mathrm{A_0} \oplus \mathrm{X_0} \rightarrow \mathrm{S}$
- 变被除数/除数为绝对值
  - $0 \rightarrow \mathrm{A_0}, 0 \rightarrow \mathrm{X_0}$
- 第一次上商判断溢出
  - $[\mathrm{A}] - [\mathrm{X}] \rightarrow \mathrm{A}$
  - $\mathrm{A} < 0?$
    - Y：$0 \rightarrow \mathrm{Q_n}$
    - N：溢出$1 \rightarrow \mathrm{V}$，停止运算，进行中断处理，重新选择比例因子
  - A、Q 同时左移一位
  - $[\mathrm{C}] - 1 \rightarrow \mathrm{C}$
  - $[\mathrm{A}] + [\mathrm{X}] \rightarrow \mathrm{A}$

原码加减交替除法控制流程

- 逐位上商
  - $A < 0?$
    - Y:
      - $0 \rightarrow Q_n$，A、Q 左移一位，$[A] + [X] \rightarrow A$
    - N:
      - $1 \rightarrow Q_n$，A、Q 左移一位，$[A] - [X] \rightarrow A$
  - $[C] - 1 \rightarrow C$
  - $C = 0?$
    - N：回到判断$A < 0?$
    - Y：最后一次上商
      - $A < 0?$
        - Y：$0 \rightarrow Q_n$
        - N：$1 \rightarrow Q_n$

谢谢！