

本节内容

定点数
原码除法运算

王道考研/CSKAOYAN.COM

1

本节总览

除法运算

除法运算的思想

原码除法：恢复余数法

原码除法：加减交替法（不恢复余数法）

补码除法：加减交替法

王道考研/CSKAOYAN.COM

2


手算除法（十进制）

r 进制: $K_n K_{n-1} \dots K_2 K_1 K_0 \overset{.}{K_{-1} K_{-2} \dots K_{-m}}$

$$= K_n \times r^n + K_{n-1} \times r^{n-1} + \dots + K_2 \times r^2 + K_1 \times r^1 + K_0 \times r^0 + \overset{.}{K_{-1} \times r^{-1} + K_{-2} \times r^{-2} + \dots + K_{-m} \times r^{-m}}$$

0.211 ÷ 0.985 = ?

你怎么这个亚子



985

0.214

211

000

2110

1970

1400

985

4150

3940

210

→

0.985

0.214

0.211

0.000

0.2110

0.1970

0.01400


0.00985

0.004150

0.003940

0.000210

心情复杂



0.214

0.985


$0.985 \times 0.214 = (985 \times 2 \times 10^{-4}) + (985 \times 1 \times 10^{-5}) + (985 \times 4 \times 10^{-6})$
 $= 0.1970 + 0.00985 + 0.00394$

$x \div y = a \text{ (余数 } b) \rightarrow x = ay + b$

$0.211 = 0.985 \times 0.214 + 0.000210$

王道考研/CSKAOYAN.COM

3



举个栗子

手算除法（二进制）

符号位	绝对值
-----	-----

两个正数相除

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，求 x/y

$(0.1011 \times 2^4) \div (0.1101 \times 2^4)$

01101

0.1101

01011

00000

10110

01101

10010

01101

01010

00000

10100

01101

0111

→

0.1101

0.1011

0.0000

0.10110

0.01101

0.010010

0.001101

0.0001010

0.0000000

0.00010100

0.00001101

0.00000111

规律：忽略小数点，每确定一位商，进行一次减法，得到4位余数，在余数末尾补0，再确定下一位商。确定5位商即可停止（机器字长为5位）

x/y 结果为0.1101，余数为0.00000111

王道考研/CSKAOYAN.COM

4

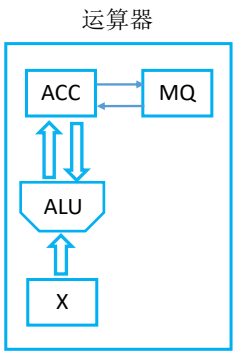
王道考 研,cskaoyan.com

2

穿越：运算器的基本组成



缓缓地回忆过去



运算器：用于实现算术运算（如：加减乘除）、逻辑运算（如：与或非）

- ACC: 累加器，用于存放操作数，或运算结果。
- MQ: 乘商寄存器，在乘、除运算时，用于存放操作数或运算结果。
- X: 通用的操作数寄存器，用于存放操作数
- ALU: 算术逻辑单元，通过内部复杂的电路实现算数运算、逻辑运算

	加	减	乘	除
Accumulator	ACC	被加数、和	乘积高位	被除数、余数
Multiple-Quotient Register	MQ		乘数、乘积低位	商
Arithmetic and Logic Unit	X	加数	减数	除数

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0

数值位取绝对值进行除法计算

01101 / 01011
00000

10110
01101

10010
01101

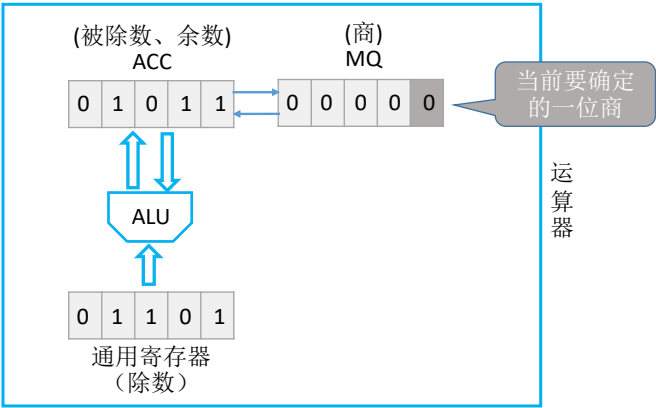
01010
00000

10100
01101

0111

手算时，每一位商取0/1 是通过判断当前余数和除数的大小确定的

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”



原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-|y|]_{补}\rightarrow$ ACC
 $01011+10011=11110$

(被除数、余数)
ACC
0 1 0 1 1

(商)
MQ
0 0 0 0 1

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

通用寄存器
(除数)
0 1 1 0 1

ALU

运算器

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

7

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-|y|]_{补}\rightarrow$ ACC
 $01011+10011=11110$

(被除数、余数)
ACC
1 1 1 1 0

(商)
MQ
0 0 0 0 1

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

通用寄存器
(除数)
0 1 1 0 1

ALU

运算器

相减结果是个负数，说明应该上商0

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

我知道错了

王道考研/CSKAOYAN.COM

8

王道考 研,cskaoyan.com

4

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

恢复余数：
(ACC)+(除数) \rightarrow ACC

(ACC)+ $[-|y|]_{补} \rightarrow$ ACC
 $01011+10011 = 11110$

(ACC)+ $[|y|]_{补} \rightarrow$ ACC
 $11110+01101 = 01011$

(被除数、余数)
ACC
1 1 1 1 0

(商)
MQ
0 0 0 0 0

↑↓
ALU
↑
0 1 1 0 1
通用寄存器
(除数)

应该商0

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

整改，必须整改！

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

9

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

恢复余数：
(ACC)+(除数) \rightarrow ACC

(ACC)+ $[-|y|]_{补} \rightarrow$ ACC
 $01011+10011 = 11110$

(ACC)+ $[|y|]_{补} \rightarrow$ ACC
 $11110+01101 = 01011$

(被除数、余数)
ACC
0 1 0 1 1

(商)
MQ
0 0 0 0 0

↑↓
ALU
↑
0 1 1 0 1
通用寄存器
(除数)

应该商0

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

整改，必须整改！

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

10

王道考 研,cskaoyan.com

5

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

ACC、MQ整体“逻辑左移”。ACC高位丢弃，MQ低位补0

(被除数、余数)
ACC
1 0 1 1 0

(商)
MQ
0 0 0 0 0

低位补0

运算器

ALU

通用寄存器
(除数)
0 1 1 0 1

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

11

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC
(ACC)+ $[-y]_{补}\rightarrow$ ACC
10110+10011 = 01001

(被除数、余数)
ACC
1 0 1 1 0

(商)
MQ
0 0 0 0 1

当前要确定的一位商

运算器

ALU

通用寄存器
(除数)
0 1 1 0 1

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

12

王道考 研,cskaoyan.com

6

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-y]_{补}\rightarrow$ ACC
 $10110+10011=01001$

相减结果
是个正数，
上商1是
没错滴~

(被除数、余数)
ACC
0 1 0 0 1

(商)
MQ
0 0 0 0 1

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

王道考研/CSKAOYAN.COM

13

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

ACC、MQ整体
“逻辑左移”

低位补0

(被除数、余数)
ACC
0 1 0 0 1 0

(商)
MQ
0 0 0 1 0

低位补0

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

王道考研/CSKAOYAN.COM

14

王道考 研,cskaoyan.com

7

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行除法计算

实现方法：上商0/1，得到余数，余数末尾补0

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

ACC、MQ整体
“逻辑左移”

(被除数、余数)
ACC
1 0 0 1 0

(商)
MQ
0 0 0 1 1

↑ ↓
ALU
↑
0 1 1 0 1
通用寄存器
(除数)

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

15

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[|y|]_{补}=0.1101$ ， $[-|y|]_{补}=1.0011$

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行除法计算

实现方法：上商0/1，得到余数，余数末尾补0

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数)→ACC
(ACC)+ $[-|y|]_{补}$ →ACC
10010+10011 = 00101

(被除数、余数)
ACC
1 0 0 1 0

(商)
MQ
0 0 0 1 1

↑ ↓
ALU
↑
0 1 1 0 1
通用寄存器
(除数)

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

16

王道考 研,cskaoyan.com

8

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-y]_{补}\rightarrow$ ACC
10010+10011 = 00101

相减结果
是个正数，
上商1是
没错滴~

(被除数、余数)
ACC
0 0 1 0 1

(商)
MQ
0 0 0 1 1

当前要确定
的一位商

计算机很傻，会先
默认上商1，如果
搞错了再改上商0。
并“恢复余数”

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

17

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

ACC、MQ整体
“逻辑左移”

低位补0

(被除数、余数)
ACC
0 0 1 0 1 0

(商)
MQ
0 0 1 1 0

低位补0

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

18

王道考 研,cskaoyan.com

9

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-y]_{补}\rightarrow$ ACC
 $01010+10011=11101$

(被除数、余数)
ACC
0 1 0 1 0

(商)
MQ
0 0 1 1 1

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

19

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-y]_{补}\rightarrow$ ACC
 $01010+10011=11101$

(被除数、余数)
ACC
1 1 1 0 1

(商)
MQ
0 0 1 1 1

当前要确定的一位商

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

相减结果是个负数，不该上商1

当事人 现在就是后悔，非常后悔

王道考研/CSKAOYAN.COM

20

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行除法计算

实现方法：上商0/1，得到余数，余数末尾补0

01101

01011

00000

10110

01101

10010

01101

01010

00000

10100

01101

0111

恢复余数：

$(ACC)+(除数) \rightarrow ACC$

$(ACC)+[-y]_{补} \rightarrow ACC$

$01010+10011 = 11101$

$(ACC)+[y]_{补} \rightarrow ACC$

$11101+01101 = 01010$

(被除数、余数)

ACC

01010

(商)

MQ

00110

应该商0

运算器

ALU

通用寄存器

(除数)

01101

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

21

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行除法计算

实现方法：上商0/1，得到余数，余数末尾补0

01101

01011

00000

10110

01101

10010

01101

01010

00000

10100

01101

0111

ACC、MQ整体

“逻辑左移”

(被除数、余数)

ACC

01010

(商)

MQ

00110

低位补0

运算器

ALU

通用寄存器

(除数)

01101

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

22

王道考 研,cskaoyan.com

11

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

求余数：
(ACC)-(除数) \rightarrow ACC

(ACC)+ $[-y]_{补}\rightarrow$ ACC
 $10010+10011=00111$

相减结果
是个正数，
应上商 1

(被除数、余数)
ACC
0 0 1 1 1

(商)
MQ
0 1 1 0 1

当前要确定的一位商

注：若最后一步商
余数为负，也需要
恢复余数并商0

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

23

原码除法：恢复余数法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$ 符号单独处理：符号位 = $x_s \oplus y_s$

实现方法：上商0/1，得到余数，余数末尾补0 数值位取绝对值进行除法计算

01101 $\overline{)01011}$
00000
10110
01101
10010
01101
01010
00000
10100
01101
0111

余数= 0.0111×2^{-n}

(被除数、余数)
ACC
0 0 1 1 1

(商)
MQ
0 1 1 0 1

商=0.1101

小数点隐含位置

运算器

通用寄存器
(除数)
0 1 1 0 1

ALU

计算机很傻，会先默认上商1，如果搞错了再改上商0。并“恢复余数”

王道考研/CSKAOYAN.COM

24

原码除法：恢复余数法（手算）

我有一个新思路

能否不恢复余数？

符号位

绝对值

符号位与数值位分开处理

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y
 $|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

被除数/余数

商

0.1011

$+[-y]_{补}$

1.0011

1.1110

$+ [y]_{补}$

0.1101

0.1011

左移

1.0110

$+ [-y]_{补}$

1.0011

0.1001

左移

1.0010

$+ [-y]_{补}$

1.0011

0.0101

...

余数为负，就要商0，并恢复余数

余数为正，就要商1，不用恢复余数

0

01

011

...

老余数 - |除数| = 新余数

新余数为负？

N

商1

Y

商0，+|除数|恢复为老余数

余数逻辑左移

左移n次，上商n+1次
最后一次上商余数不左移

王道考研/CSKAOYAN.COM

25

原码除法：恢复余数法（手算）

我有一个新思路

能否不恢复余数？

符号位

绝对值

符号位与数值位分开处理

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码恢复余数法求 x/y
 $|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

被除数/余数

商

0.1011

$+[-y]_{补}$

1.0011

1.1110

$+ [y]_{补}$

0.1101

0.1011

左移

1.0110

$+ [-y]_{补}$

1.0011

0.1001

左移

1.0010

$+ [-y]_{补}$

1.0011

0.0101

...

余数a为负

若余数为负，则可直接商0，并让余数左移1位再加上|除数|

0

01

011

...

a

b

$a+b$

$(a+b) \times 2 = 2a + 2b$

$(a+b) \times 2 - b = 2a + 2b - b = 2a + b$

...

王道考研/CSKAOYAN.COM

26

王道考 研,cskaoyan.com

13

原码除法：加减交替法

又名：不恢复余数法

符号位与数值位分开处理

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码加减交替除法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

若余数为负，则可直接商0，让余数左移1位再加上 $|除数|$ ，得到下一个新余数

若余数为正，则商1，让余数左移1位再减去 $|除数|$ ，得到下一个新余数

被除数/余数	商	ACC	MQ
0.1011		01011	00000
$+[-y]_{补}$ 1.0011			
1.1110	0	11110	00000
左移 1.1100			
$+ [y]_{补}$ 0.1101		11100	00000
0.1001	01	01001	00001
左移 1.0010			
$+ [-y]_{补}$ 1.0011		10010	00010
0.0101	011	00101	00011
左移 0.1010			
$+ [-y]_{补}$ 1.0011		01010	00110
1.1101	0110	11101	00110
左移 1.1010			
$+ [y]_{补}$ 0.1101		11010	01100
0.0111	01101	00111	01101

若余数为负，需商0，并 $+ [y]_{补}$ 得到正确余数

$Q_s = x_s \oplus y_s = 0 \oplus 0 = 0$
得 $x/y = +0.1101$
余 0.0111×2^{-4}

注：余数的正负性与商相同

恢复余数法：当余数为负时商0，并 $+ |除数|$ ，再左移，再 $- |除数|$

加减交替法：当余数为负时商0，并左移，再 $+ |除数|$

王道考研/CSKAOYAN.COM

27

原码除法：加减交替法

又名：不恢复余数法

符号位与数值位分开处理

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码加减交替除法求 x/y

$|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

若余数为负，需商0，并 $+ [y]_{补}$ 得到正确余数

被除数/余数	商
0.1011	
$+ [-y]_{补}$ 1.0011	
1.1110	0
左移 1.1100	
$+ [y]_{补}$ 0.1101	
0.1001	01
左移 1.0010	
$+ [-y]_{补}$ 1.0011	
0.0101	011
左移 0.1010	
$+ [-y]_{补}$ 1.0011	
1.1101	0110
左移 1.1010	
$+ [y]_{补}$ 0.1101	
0.0111	01101

$Q_s = x_s \oplus y_s = 0 \oplus 0 = 0$
得 $x/y = +0.1101$
余 0.0111×2^{-4}

被除数-|除数|=新余数

新余数为负？

Y: 商0，余数左移并 $+ |除数|$

N: 商1，余数左移并 $- |除数|$

加/减 $n+1$ 次，每次加减确定一位商；左移 n 次（最后一次加减完不移位）最终可能还要再多一次加

王道考研/CSKAOYAN.COM

28

本节内容

定点数
补码除法运算

王道考研/CSKAOYAN.COM

29

原码除法：加减交替法

符号位与数值位分开处理

设机器字长为5位（含1位符号位， $n=4$ ）， $x=0.1011$ ， $y=0.1101$ ，采用原码加减交替除法求 x/y
 $|x|=0.1011$ ， $|y|=0.1101$ ， $[y]_{补}=0.1101$ ， $[-y]_{补}=1.0011$

被除数/余数	商
0.1011	
$+[-y]_{补}$ 1.0011	
1.1110	0
左移 1.1100	
$+ [y]_{补}$ 0.1101	
0.1001	01
左移 1.0010	
$+ [-y]_{补}$ 1.0011	
0.0101	011
左移 0.1010	
$+ [-y]_{补}$ 1.0011	
1.1101	0110
左移 1.1010	
$+ [y]_{补}$ 0.1101	
0.0111	01101

若余数为负，需商0，并 $+ [y]_{补}$ 得到正确余数

第一次必然是 $-|除数|$

$Q_s = x_s \oplus y_s = 0 \oplus 0 = 0$
得 $x/y = +0.1101$
余 0.0111×2^{-4}

符号位单独确定

被除数 - $|除数|$ = 新余数

新余数为负？
Y 商0，余数左移并 $+ |除数|$
N 商1，余数左移并 $- |除数|$

之后每次根据余数的正负性来确定加/减

加/减 $n+1$ 次，每次加减确定一位商；
左移 n 次（最后一次加减完不移位）
最终可能还要再多一次加

王道考研/CSKAOYAN.COM

30

补码除法：加减交替法

设机器字长为5位（含1位符号位， $n=4$ ）， $x=+0.1000$ ， $y=-0.1011$ ，采用补码加减交替除法求 x/y
[x]_补=00.1000，[y]_补=11.0101，[-y]_补=00.1011 [x/y]_补=1.0101，余0.0111×2⁻⁴

被除数/余数		ACC	MQ
	00.1000	001000	00000
+[y] _补	11.0101		
	11.1101	111101	00001
逻辑左移	11.1010	111010	00010
左移	00.1011		
+[y] _补	00.0101	000101	00010
	00.0101	001010	00100
左移	11.1111		
左移	11.1110	111111	00101
+[y] _补	00.1011	111110	01010
	00.1001	001001	01010
左移	01.0010	010010	10100
+[y] _补	11.0111	000111	10101
	00.0111		

补码除法：
• 符号位参与运算
• 被除数/余数、除数
采用双符号位

被除数和除数同号，则被除数减去除数；
异号则被除数加上除数。

余数和除数同号，商1，余数左移一位减去除数；
余数和除数异号，商0，余数左移一位加上除数。
重复n次

精度误差
不超过 2⁻ⁿ

末位商恒置1

王道考研/CSKAOYAN.COM

31

除法运算总结回顾

除法类型	符号位参与运算	加减次数	移位		上商、加减原则	说明
			方向	次数		
原码加减交替法	否	N+1或N+2	左	N	余数的正负	若最终余数为负，需恢复余数
补码加减交替法	是	N+1	左	N	余数和除数是否同号	商末位恒置1

王道考研/CSKAOYAN.COM

32