里昂学长

补码一位乘法

理解了原码一位乘法后，我们继续从数学原理出发，推导补码乘法

为了得到补码一位乘法的规律，我们先从补码和真值的转换公式开始讨论

补码与真值的转换公式

设[x]补=x0.x1x2...xn,当x≥0时，

当x<0时

所以

故得出 x=

补码的右移

正数补码右移一位，相当于乘1/2，
负数用补码表示时，右移一位还相当于乘1/2么

设[x]补=x0.x1x2...xn,因为

x=

所以

写成补码形式，即得

补码乘法规则

设被乘数[x]补=x0.x1x2...xn,乘数[y]补=y0.y1y2...yn均为任意符号

则有补码乘法算式

[x*y]补=

证明如下

（1）被乘数x符号任意，乘数y符号为正

根据补码定义

[x]补=

[y]补=

所以

[x]补*[y]补=

其中（y1y2...yn)是大于0的正整数，根据模运算性质有

2(y1y2..yn)=

所以

[x]补*[y]补=

即

[x*y]补=

（2）被乘数x符号任意，乘数y符号为负

[x]补=

[y]补=

由此

y=

所以

x*y=

[x*y]补=

所以

[x*y]补=

（3）被乘数x符号任意，乘数y符号为负

将以上（1）（2）两种情况综合起来，即得补码乘法的统一算式

[x*y]补=

为了推出串行逻辑实现的分步算法，将上式展开加以变换

[x*y]补=

写成递推公式如下：

[z0]补=

因此，我们不难得到补码乘法，本质上要进行如下n+1次加法操作和n次移位操作

对比 原码是yi*x

1.加法（y(n+1)-yn）*[x]补

- 1.y(n+1)-yn=0,则加0
- 2.y(n+1)-yn=1,则加[x]补
- 3.y(n+1)-yn=-1,则减[x]补，等价于加[-x]补

这样的操作总共n+1次！

2.乘2^(-1) → 等价于右移一位

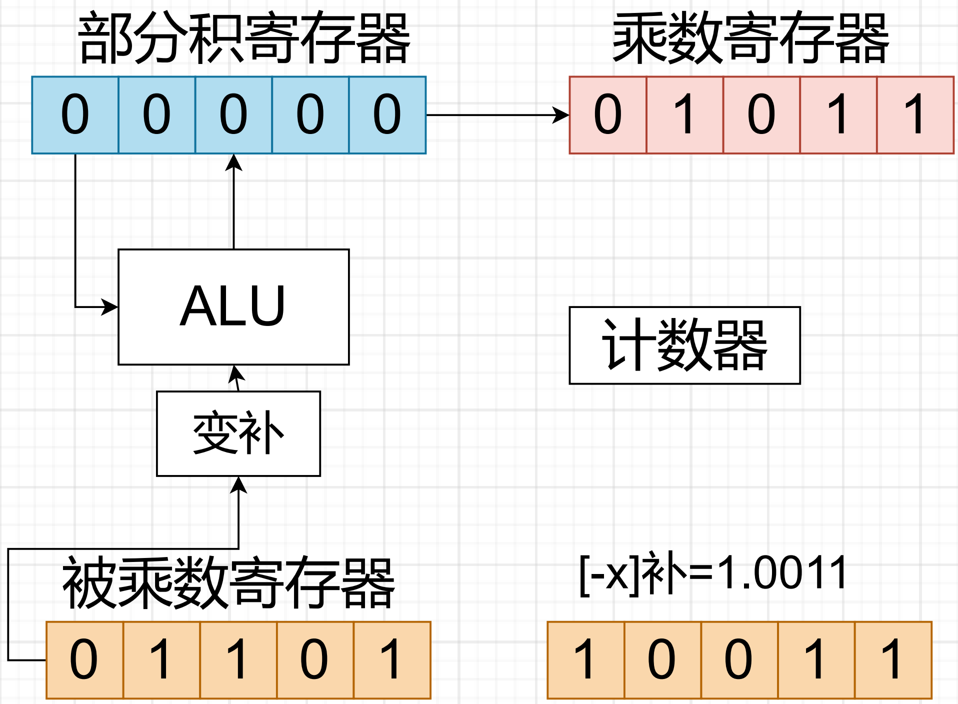
右移操作总共n次！

注意，补码乘法中，包含符号位，因此每次移位一定是***算术右移***

懂了原理之后，我们来看看计算机是如何进行补码乘法运算的

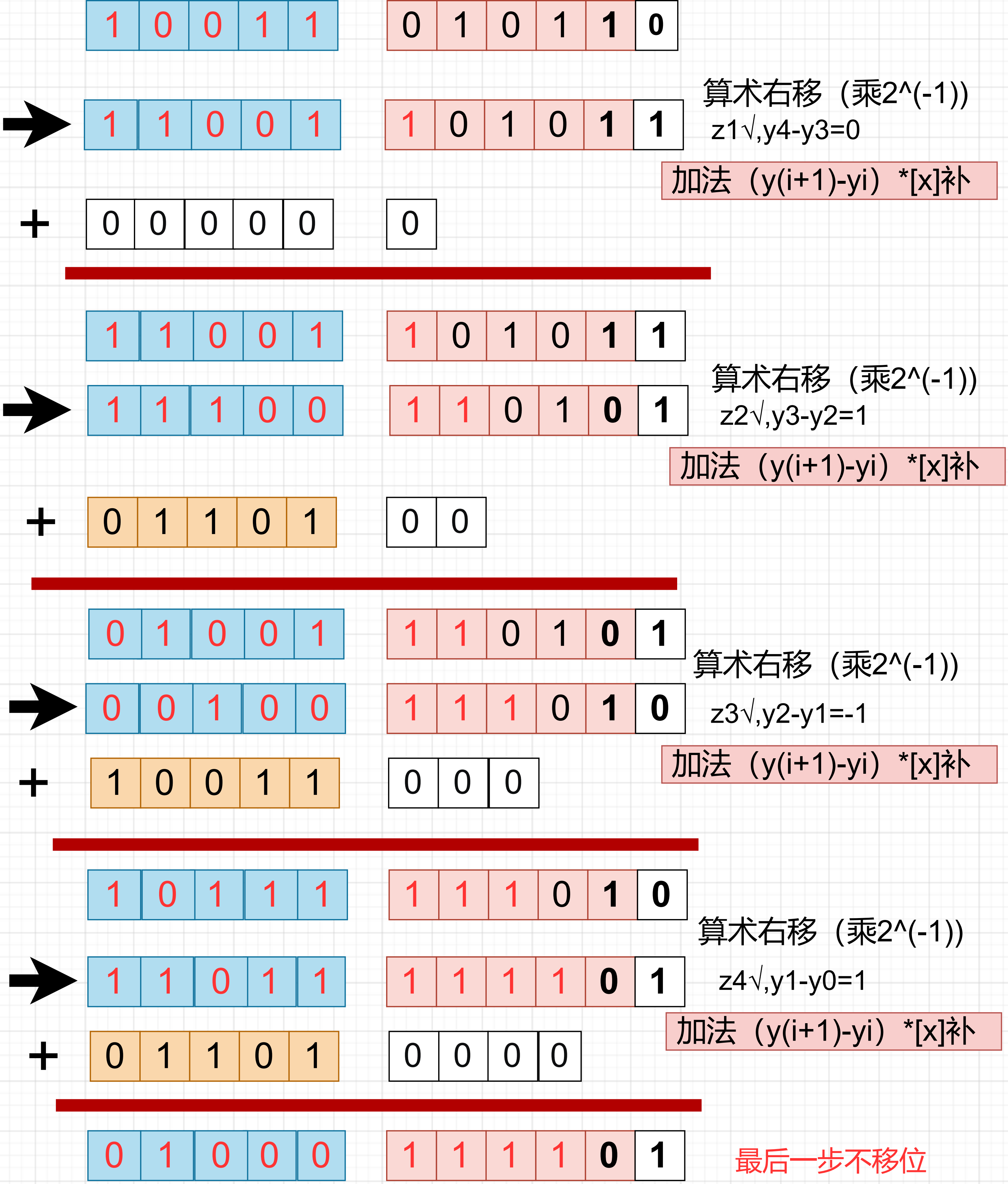
x=0.1101   y=0.1011

机器字长为5



手算模拟

部分积	乘数	说明
0 0 0 0 0	0 1 0 1 1 0	z0=0,y5-y4=-1 加法（y(i+1)-yi）*[x]补
+ 1 0 0 1 1		



总结，定点小数n+1位补码乘法（包括符号位），进行n+1次加法，n次右移，得到2n+1位结果

口诀：蒜头补肾454或者蒜头补乘454

蒜头（ou）：算术右(ou)移，符号位不变，正数补0，负数补1

454表示，4位尾数，5次加法，4次右移

里昂学长

接下来我们再进一步改进手算模拟，使用双符号位

x=0.1101

y=0.1011