B0911006Y-01: Computer Organization and Design **2023 Spring**

Homework 5 — April 3

Lecturer: Ke Zhang *Completed by:* 吉骏雄

习题：6.20(原码一位乘、原码两位乘、补码一位乘 Booth 算法必做, 补码两位乘选做), 6.23, 6.21(原码加减交替除法，补码加减交替除法)

6.20 用原码一位乘、两位乘和补码一位乘 (Booth 算法) 、两位乘计算 xy.

1. $x = 0.110111$, $y = -0.101110$;
2. $x = -0.010111$, $y = -0.010101$;
3. $x = 19$, $y = 35$;
4. $x = 0.11011$, $y = -0.11101$.

解

1. $x = 0.110111$, $y = -0.101110$;

原码一位乘:

	部分积	乘数	
	0.000000	101110	
$+$	0.000000		$+0$
	0.000000		
	0.000000	0 10111	$\to 1$
$+$	0.110111		$+x$
	0.110111		
	0.011011	10 1011	$\to 1$
$+$	0.110111		$+x$
	1.010010		
	0.101001	010 101	$\to 1$
$+$	0.110111		$+x$
	1.100000		
	0.110000	0010 10	$\to 1$
$+$	0.000000		$+0$
	0.110000		
	0.011000	00010 1	$\to 1$
$+$	0.110111		$+x$
	1.001111		
	0.100111	100010	$\to 1$

符号位异或为 1, 结果的原码为: 1.100111 100010

原码两位乘:

	部分积	乘数	C_j	
	000.000000	00.101110	0	
+	001.101110			$+2x$
	001.101110			
	000.011011	10 001011	0	$\to 2$
+	111.001001			$-x$
	111.100100			
	111.111001	0010 0010	1	$\to 2$
+	111.001001			$-x$
	111.000010			
	111.110000	100010 00	1	$\to 2$
+	000.110111			$+x$
	000.100111	100010		

符号位异或为 1, 结果的原码为: 1.100111 100010

$[x]_{补} = 0.110111$, $[y]_{补} = 1.010010$;

Booth 算法:

	部分积	乘数	附加位	
	00.000000	1.010010	0	$+0, \to 1$
	00.000000	0 101001	0	$-x$
+	11.001001			
	11.001001			$\to 1$
	11.100100	10 10100	1	$+x$
+	00.110111			
	00.011011			$\to 1$
	00.001101	110 1010	0	$+0, \to 1$
	00.000110	1110 101	0	$-x$
+	11.001001			
	11.001111			$\to 1$
	11.100111	11110 10	1	$+x$
+	00.110111			
	00.011110			$\to 1$
	00.001111	011110 1	0	$-x$
+	11.001001			
	11.011000	011110 x		

结果的补码为: 1.011000 011110

2. $x = -0.010111$, $y = -0.010101$;

原码一位乘:

	部分积	乘数	
	0.000000	010101	
+	0.010111		$+x$
	0.010111		
	0.001011	1 01010	$\rightarrow 1$
+	0.000000		+0
	0.001011		
	0.000101	11 0101	$\rightarrow 1$
+	0.010111		$+x$
	0.011100		
	0.001110	011 010	$\rightarrow 1$
+	0.000000		+0
	0.001110		
	0.000111	0011 01	$\rightarrow 1$
+	0.010111		$+x$
	0.011110		
	0.001111	00011 0	$\rightarrow 1$
+	0.000000		+0
	0.001111		
	0.000111	100011	$\rightarrow 1$

符号位异或为 0, 结果的原码为: 0.000111 100011

原码两位乘:

	部分积	乘数	C_j	
	000.000000	00.010101	0	
+	000.010111			$+x$
	000.010111			
	000.000101	11 000101	0	$\rightarrow 2$
+	000.010111			$+x$
	000.011100			
	000.000111	0011 0001	0	$\rightarrow 2$
+	000.010111			$+x$
	000.011110			
	000.000111	100011 00	0	$\rightarrow 2$
+	000.000000			+0
	000.000111	100011		

符号位异或为 0, 结果的原码为: 0.000111 100011

$[x]_{补} = 1.101001$, $[y]_{补} = 1.101011$;

Booth 算法:

	部分积	乘数	附加位	
	00.000000	1.101011	0	$-x$
$+$	00.010111			
	00.010111			$\rightarrow 1$
	00.001011	1 110101	1	$+0, \rightarrow 1$
	00.000101	11 11010	1	$+x$
$+$	11.101001			
	11.101110			$\rightarrow 1$
	11.110111	011 1101	0	$-x$
$+$	00.010111			
	00.001110			$\rightarrow 1$
	00.000111	0011 110	1	$+x$
$+$	11.101001			
	11.110000			$\rightarrow 1$
	11.111000	00011 11	0	$-x$
$+$	00.010111			
	00.001111			$\rightarrow 1$
	00.000111	100011 1	1	$+0$
	00.000111	100011		

结果的补码为: 0.000111 100011

3. $x = 19 = 0,010011$, $y = 35 = 0,100011$;

原码一位乘:

	部分积	乘数	
	0,000000	100011	
+	0,010011		$+x$
	0,010011		
	0,001001	1 10001	$\rightarrow 1$
+	0,010011		$+x$
	0,011100		
	0.001110	01 1000	$\rightarrow 1$
+	0,000000		$+0$
	0,001110		
	0,000111	001 100	$\rightarrow 1$
+	0,000000		$+0$
	0,000111		
	0,000011	1001 10	$\rightarrow 1$
+	0,000000		$+0$
	0,000011		
	0,000001	11001 1	$\rightarrow 1$
+	0,010011		$+x$
	0,010100		
	0,001010	011001	$\rightarrow 1$

符号位异或为 0, 结果的原码为: 0,001010 011001

原码两位乘:

	部分积	乘数	C_j	
	000.000000	00.100011	0	
+	111.101101			$-x$
	111.101101			
	111.111011	01 001000	1	$\rightarrow 2$
+	000.010011			$+x$
	000.001110			
	000.000011	1001 0010	0	$\rightarrow 2$
+	000.100110			$+2x$
	000.101001			
	000.001010	011001 00	0	$\rightarrow 2$
+	000.000000			$+0$
	000.001010	011001		

符号位异或为 0, 结果的原码为: 0,001010 011001

$[x]_{补} = 0,010011$, $[y]_{补} = 0,100011$;

Booth 算法:

	部分积	乘数	附加位	
	00.000000	0.100011	0	$-x$
+	11.101101			
	11.101101			$\rightarrow 1$
	11,110110	1 010001	1	$+0, \rightarrow 1$
	11,111011	01 01000	1	$+x$
+	00,010011			
	00,001110			$\rightarrow 1$
	00,000111	001 0100	0	$+0, \rightarrow 1$
	00,000011	1001 010	0	$+0, \rightarrow 1$
	00,000001	11001 01	0	$-x$
+	11.101101			
	11,101110			$\rightarrow 1$
	11,110111	011001 0	1	$+x$
+	00,010011			
	00,001010	011001		

结果的补码为: 0,001010 011001

4. $x = 0.11011$, $y = -0.11101$.

原码一位乘:

	部分积	乘数	
	0.00000	11101	
+	0.11011		$+x$
	0.11011		
	0.01101	1 1110	$\rightarrow 1$
+	0.00000		$+0$
	0.01101		
	0.00110	11 111	$\rightarrow 1$
+	0.11011		$+x$
	1.00001		
	0.10000	111 11	$\rightarrow 1$
+	0.11011		$+x$
	1.01011		
	0.10101	1111 1	$\rightarrow 1$
+	0.11011		$+x$
	1.10000		
	0.11000	01111	$\rightarrow 1$

符号位异或为 1, 结果的原码为: 1.11000 01111

原码两位乘:

	部分积	乘数	C_j	
	000.00000	0.11101	0	
+	000.11011			$+x$
	000.11011			
	000.00110	11 0111	0	$\rightarrow 2$
+	111.00101			$-x$
	111.01011			
	111.11010	1111 01	1	$\rightarrow 2$
+	001.10110			$+2x$
	001.10000			
	000.11000	01111		$\rightarrow 1$ (!)

符号位异或为 1, 结果的原码为: 1.11000 01111

$[x]_{补} = 0.11011$, $[y]_{补} = 1.00011$;

Booth 算法:

	部分积	乘数	附加位	
	00.00000	1.00011	0	$-x$
+	11.00101			
	11.00101			$\rightarrow 1$
	11.10010	1 10001	1	$+0, \rightarrow 1$
	11.11001	01 1000	1	$+x$
+	00.11011			
	00.10100			$\rightarrow 1$
	00.01010	001 100	0	$+0, \rightarrow 1$
	00.00101	0001 10	0	$+0, \rightarrow 1$
	00.00010	10001 1	0	$-x$
+	11.00101			
	11.00111	10001		

结果的补码为: 1.00111 10001

6.23 画出实现 Booth 算法的运算器框图, 要求如下:

1. 寄存器和全加器均用方框表示, 指出寄存器和全加器的位数.
2. 说明加和移位的次数.
3. 详细画出最低位全加器的输入电路.
4. 描述 Booth 算法重复加和移位的过程.

解 如图 5.1
寄存器和全加器都是 $n+2$ 位. 加法和移位运算的次数都是 n 次.

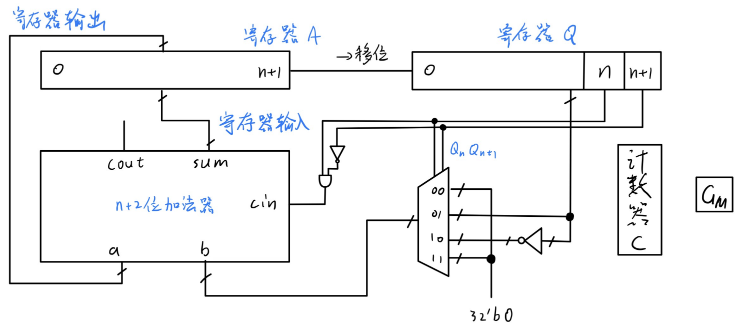

图 5.1. 6.23 Booth 算法运算器框图

每个周期 (不一定是单个时钟周期), 取出两个寄存器中的数据, 输入选择器和加法器得到结果, 然后将结果输入回寄存器 A. 之后, 将整个寄存器 A 与 Q 中的所有数字视为一个整体, 进行算术右移. 这样就结束了一个周期.

6.21 用原码加减交替法和补码加减交替法计算 $x \div y$。

1. x = 0.100111, y = 0.101011;
2. x = -0.10101 , y = 0.11011 ;
3. x = 0.10100 , y = -0.10001 ;
4. x = 13/32 , y = -27/32 .

解

1. x = 0.100111, y = 0.101011;

原码加减交替法:

	被除数 (余数)	商	
	0.100111	0.000000	
+	1.010101		$-y$
	1.111100	0	余数为负, 上商为 0
	1.111000	0	$\leftarrow 1$
+	0.101011		$+y$
	0.100011	01	余数为正, 上商为 1
	1.000110	01	$\leftarrow 1$
+	1.010101		$-y$
	0.011011	011	余数为正, 上商为 1
	0.110110	011	$\leftarrow 1$
+	1.010101		$-y$
	0.001011	0111	余数为正, 上商为 1
	0.010110	0111	$\leftarrow 1$
+	1.010101		$-y$
	1.101011	01110	余数为负, 上商为 0
	1.010110	01110	$\leftarrow 1$
+	0.101011		$+y$
	0.000001	011101	余数为正, 上商为 1
	0.000010	011101	$\leftarrow 1$
+	1.010101		$-y$
	1.010111	0111010	余数为负, 上商为 0

符号位异或为 0, 所以结果为 0.111010.

补码加减交替法:

	被除数 (余数)	商	
	0.100111	0.000000	
+	1.010101		x, y 符号位同号, $-y$
	1.111100	0	R, y 符号位异号, 上商为 0
	1.111000	0	$\leftarrow 1$
+	0.101011		$+y$
	0.100011	01	R, y 符号位同号, 上商为 1
	1.000110	01	$\leftarrow 1$
+	1.010101		$-y$
	0.011011	011	R, y 符号位同号, 上商为 1
	0.110110	011	$\leftarrow 1$
+	1.010101		$-y$
	0.001011	0111	R, y 符号位同号, 上商为 1
	0.010110	0111	$\leftarrow 1$
+	1.010101		$-y$
	1.101011	01110	R, y 符号位异号, 上商为 0
	1.010110	01110	$\leftarrow 1$
+	0.101011		$+y$
	0.000001	011101	R, y 符号位同号, 上商为 1
	0.000010	0111011	$\leftarrow 1$, 末位置 1

答案商的补码为 0.111011.

2. x = -0.10101 , y = 0.11011 ;

原码加减交替法:

	被除数 (余数)	商	
	0.10101	0.00000	
+	1.00101		$-y$
	1.11010	0	余数为负, 上商为 0
	1.10100	0	$\leftarrow 1$
+	0.11011		$+y$
	0.01111	01	余数为正, 上商为 1
	0.11110	01	$\leftarrow 1$
+	1.00101		$-y$
	0.00011	011	余数为正, 上商为 1
	0.00110	011	$\leftarrow 1$
+	1.00101		$-y$
	1.01011	0110	余数为负, 上商为 0
	0.10110	0111	$\leftarrow 1$
+	0.11011		$+y$
	1.10001	01100	余数为负, 上商为 0
	1.00010	01100	$\leftarrow 1$
+	0.11011		$+y$
	1.11101	011000	余数为负, 上商为 0
+	0.11011		恢复余数 $+y$
	0.11000	011000	

符号位异或为 1, 所以商结果原码为 1.11000, 补码为 1.01000, 余数为 0.11000

补码加减交替法:

	被除数 (余数)	商	
	1.01011	0.00000	
+	0.11011		x, y 符号位异号, $+y$
	0.00110	1	R, y 符号位同号, 上商为 1
	0.01100	1	$\leftarrow 1$
+	1.00101		$-y$
	1.10001	10	R, y 符号位异号, 上商为 0
	1.00010	10	$\leftarrow 1$
+	0.11011		$+y$
	1.11101	100	R, y 符号位异号, 上商为 0
	1.11010	100	$\leftarrow 1$
+	0.11011		$+y$
	0.10101	1001	R, y 符号位同号, 上商为 1
	1.01010	1001	$\leftarrow 1$
+	1.00101		$-y$
	0.01111	10011	R, y 符号位同号, 上商为 1
	0.11110	100111	$\leftarrow 1$, 末位置 1

商的补码为 1.00111.

3. x = 0.10100 , y = -0.10001 ;

原码加减交替法:

	被除数 (余数)	商	
	0.10100	0.00000	
+	1.01111		$-y$
	0.00011	1	余数为正, 上商为 1, 产生溢出

补码加减交替法:

	被除数 (余数)	商	
	0.10100	0.00000	
+	1.01111		x, y 符号位异号, $+y$
	0.00011	0	R, y 符号位异号, 上商为 0, 产生溢出

4. $x = \dfrac{13}{32} = [0.01101]_{原} = [0.01101]_{补}$, $y = -\dfrac{27}{32} = [1.11011]_{原} = [1.00101]_{补}$.

原码加减交替法:

	被除数 (余数)	商	
	0.01101	0.00000	
+	1.00101		$-y$
	1.10010	0	余数为负, 上商为 0
	1.00100	0	$\leftarrow 1$
+	0.11011		$+y$
	1.11111	00	余数为负, 上商为 0
	1.11110	00	$\leftarrow 1$
+	0.11011		$+y$
	0.11001	001	余数为正, 上商为 1
	1.10010	001	$\leftarrow 1$
+	1.00101		$-y$
	0.10111	0011	余数为正, 上商为 1
	1.01110	0011	$\leftarrow 1$
+	1.00101		$-y$
	0.10011	00111	余数为正, 上商为 1
	1.00110	00111	$\leftarrow 1$
+	1.00101		$+y$
	0.01011	001111	余数为正, 上商为 1

符号位异或为 1, 所以商结果原码为 1.01111, 补码为 1.10001, 余数为 0.01011

补码加减交替法:

	被除数 (余数)	商	
	0.01101	0.00000	
+	1.00101		x, y 符号位异号, $+y$
	1.10010	1	R, y 符号位同号, 上商为 1
	1.00100	1	$\leftarrow 1$
+	0.11011		$-y$
	1.11111	11	R, y 符号位同号, 上商为 1
	1.11110	10	$\leftarrow 1$
+	0.11011		$+y$
	0.11001	110	R, y 符号位异号, 上商为 0
	1.10010	110	$\leftarrow 1$
+	1.00101		$+y$
	0.10111	1100	R, y 符号位异号, 上商为 0
	1.01110	1100	$\leftarrow 1$
+	1.00101		$-y$
	0.10011	11000	R, y 符号位异号, 上商为 0
	1.00110	110001	$\leftarrow 1$, 末位置 1

商的补码为 1.10001.