Booths Algorithm

Booths recoding scheme

bit i	bit i-1	
0	0	0 × M
0	1	+1 × M
1	0	−1 × M
1	1	0 × M

Eg 1) 1 0 1 1 0 0 [0

−1 +1 0 −1 0 0

2) 0 1 1 0 0 1 [0

+1 0 −1 0 +1 −1

1) Multiplicand +13 → $\boxed{1101}$ ⇒ 01101 (Add sign bit)

Multiplier −6 → 0110 → Take 2's complement

→ $\boxed{1010}$

Add sign bit 11010

2 | 13
2 | 6 1
2 | 3 0
1 1

2 | 6
2 | 3 0
1 1

Take 2's complement

0 1 1 0
1 0 0 1
+ 1
1 0 1 0

1 1 0 1 0 | 0

$\boxed{0\ -1\ +1\ -1\ 0}$

0 1 1 0 1
2's compl 1 0 0 1 0
+ 1
1 0 0 1 1

n = 5
2n = 10

0 1 1 0 1
0 −1 1 −1 0

1 1 1 1 1

0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 0 0 1 1
0 0 0 0 1 1 0 1
1 1 1 0 0 1 1
0 0 0 0 0 0

1 | 1 1 1 0 1 1 0 0 1 0   −78

Carry Ignore

2's complement

0 0 0 1 0 0 1 1 0 1
+ 1
0 0 0 1 0 0 1 1 1 0

$2^6+2^3+2^2+2^1 = 78$

2) Multiplicand = -13
Multiplier = -20

				2's comp	Add sign bit
-13	⇒ 1101	⇒ 01101	⇒	1 0011	⇒ 110011
-20	⇒ 10100	⇒ 10100	⇒	01100	⇒ 101100

1 0 1 1 0 0 [0

-1 1 0 -1 0 0   Recoding

1 1 0 0 1 1
0 0 1 1 0 0 2's complent
           1
0 0 1 1 0 1

```
                        1   1   0   0   1   1
                       -1   1   0  -1   0   0
0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 0 1
0 0 0 0 0 0 0 0 0
1 1 1 1 0 0 1 1
0 0 0 1 1 0 1
1 | 0 0 0 1 0 0 0 0 0 1 0 0      +260
```

carry ignore.

$$\boxed{2^8 + 2^2 = 260}$$

3) Perform Multiplication for -13 and +09 using Booth's algorithm

-13 1101 = 0011 ⟶ 10011

+9 1001 ⟶ 01001

01001[0

+1 -1 0 +1 -1

					1	0	0	1	1
					+1	-1	0	+1	-1
0	0	0	0	0	0	1	1	0	1
1	1	1	1	1	0	0	1	1	
0	0	0	0	0	0	0	0		
0	0	0	1	1	0	1			
1	1	0	0	1	1				

10011

01100

1

01101

1 | 1 1 1 0 0 0 1 0 1 1 = -117

Ignore

⇓

2's complement

0 0 0 1 1 1 0 1 0 0

1

0 0 0 1 1 1 0 1 0 1

$2^6 + 2^5 + 2^4 + 2^2 + 2^0$

$64 + 32 + 16 + 4 + 1 = 117$

7) Multiply the following signed 2's complement numbers using Booths algorithm

multiplicand = $(01\ 0111)_2$

multiplier = $(110110)_2$

1 1 0 1 1 0 $\boxed{0}$

0 −1 +1 0 −1 0

2's
0 1 0 1 1 1
1 0 1 0 0 0
1
1 0 1 0 0 1

0 1 0 1 1 1
0 −1 +1 0 −1 0

1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 1
1 1 1 1 1 1 0 1 0 0 0 1
0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 1 1 1
1 1 1 0 1 0 0 1
0 0 0 0 0 0 0

1] 1 1 1 1 0 0 0 1 1 0 1 0 — 230

2's complement

0 0 0 0 1 1 1 0 0 1 0 1
1
0 0 0 0 1 1 1 0 0 1 1 0

$2^7 + 2^6 + 2^5 + 2^2 + 2^1 = 230$

5.) Multiply 14 X -8 using Booths Algorithm

Multiplicand = 14

Multiplier = -8

14 = 1 1 1 0 ⇒ 2's complement, Add sign bit: 0 1 1 1 0

-8 = 1 0 0 0 ⇒ 1 0 0 0 ⇒ 1 1 0 0 0

1 1 0 0 0 [0

0 -1 0 0 0

2's of 0 1 1 1 0:

```
  0 1 1 1 0
  1 0 0 0 1
          1
  ---------
  1 0 0 1 0
```

```
                    0   1   1   1   0
                    0  -1   0   0   0
---------------------------------------
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
1 1 1 0 0 1 0
0 0 0 0 0 0
---------------------------------------
1 1 1 0 0 1 0 0 0 0        -112
---------------------------------------
```

↓

2's Com

```
0 0 0 1 1 0 1 1 1 1
                  1
-------------------
0 0 0 1 1 1 0 0 0 0
```

$2^6 \quad 2^5 \quad 2^4 = 112$

Scanned by CamScanner

Fast Multiplication

↳ 2 Technique 1) Bit Pair recoding of Multiplier

2) Carry Save addition of Summands

Bit Pair recoding of Multiplier

$n = 6$ bits

recoding $= n/2$ bits $= 6/2 = 3$ bits

Recoding Example

Sign extension bit ← 1 1 1 0 1 0 $\boxed{0}$

0 −1 −2

① Multiply −11 & +27

Multiplicand −11
Multiplier 27

		2's comp		signbit
−11 ⇒ 01011	⇒	10101	⇒	110101
27 ⇒ 11011	⟶			011011

Recoding of Multiplier

$$\overbrace{0\ 1\ 1}^{+2}\ \overbrace{0\ 1\ 1}^{-1}\ \overbrace{1\ [0}^{-1}$$

0 1 1 0 1 1 [0

If −1 2's of Multiplicand

110101

001010
+1

001011

```
          1 1 0 1 0 1
           +2  -1  -1
  ---------------------
  0 0 0 0 0 0 0 0 1 0 1 1
  0 0 0 0 0 0 1 0 1 1
  1 1 1 0 1 0 1 0
  ---------------------
  1 1 1 0 1 1 0 1 0 1 1 1 = -297
```

If +2 → Binary value = 10

Now Multiply multiplicand bits with 10

```
   110101 X 10
   -----------
    000000
   110101
   -----------
   1101010
```

⇓

```
  0 0 0 1 0 0 1 0 1 0 0 0
                        1
  -----------------------
  0 0 0 1 0 0 1 0 1 0 0 1
```

$2^8 + 2^5 + 2^3 + 2^0$

$256 + 32 + 8 + 1 = 297$

Scanned by CamScanner

② Multiply +13 & −6

+13 ⇒ 1101 $\xrightarrow{2^s \text{ complement}}$ Add Sign bit 01101

−6 ⇒ 0110 ⇒ 1010 ⇒ 11010

Recoding 1 1 1 0 1 0 [0

Add sign extended bit

0 −1 −2

If (−1)

Take 2's of Multiplicand

```
 01101
 10010
     1
 -----
 10011
```

```
          0  1  1  0  1
          0     -1    -2
 1 1 1
----------------------------
 1 1 1 1 1 0 0 1 1 0
 1 1 1 1 0 0 1 1
 0 0 0 0 0 0
----------------------------
1 1 1 1 0 1 1 0 0 1 0
                  -78
```

Carry ignore

If (−2)

Take 2's complement of Multiplicand

```
 01101
 10010
     1
 -----
 10011
```

Then Multiply with 10

```
  10011X10
  --------
   00000
  10011.
  --------
  100110
```

↓ 2's complement

```
 0 0 0 1 0 0 1 1 0 1
                   1
---------------------
 0 0 0 1 0 0 1 1 1 0
```

$2^6 + 2^3 + 2^2 + 2^1 = 78$

Scanned by CamScanner

③ Multiply the following pair of signed 2's complement number using bit pair recording of the Multiplier A = 01 0111, B = 101100

```
A = 0 1 0 1 1 1   (+23)
B = 1 0 1 1 0 0   (-20)

Recoding   1 1 0 1 1 0 0 [0
             -1    -1   0
```

If −1
2's complement of Multiplicand

```
  0 1 0 1 1 1
  1 0 1 0 0 0
            1
  -----------
  1 0 1 0 0 1
```

```
                    0 1 0 1 1 1
                     -1  -1   0
  -------------------------------
  0 0 0 0 0 0 0 0 0 0 0 0
  1 1 1 1 1 0 1 0 0 1
  1 1 1 0 1 0 0 1
  -------------------------------
1 1 1 1 0 0 0 1 1 0 1 0 0  => -460
```

Ignore carry

```
  0 0 0 1 1 1 0 0 1 0 1 1
                        1
  -----------------------
  0 0 0 1 1 1 0 0 1 1 0 0
```

2^8 2^7 2^6 2^3 2^2

256 + 128 + 64 + 8 + 4 = 460

Scanned by CamScanner

Integer Division

↳ 2 Method — 1) Restoring Division
2) Non restoring Division.

Restoring Division - Steps

1) $A \leftarrow 0$, $M \leftarrow$ Divisor, $Q \leftarrow$ Dividend
Count $\leftarrow n$
2) Shift A & Q Left one binary position
3) $A \leftarrow A - M$
4) If A is 1
   - ↳ Yes — Assign $Q_0 \leftarrow 0$
     $A \leftarrow A + M$ (Restore)
   - ↳ NO — $Q_0 \leftarrow 1$
5) Count $\leftarrow$ count $- 1$
6) Repeat 2, 3, 4, 5 n times

Example ①

Dividend = 1010 (Q) (10)

Divisor = 0011 (M) (3)

Count = 4/3/2

$A = n+1$ (bits) $= 5$

Q

Initially 0 0 0 0 0 1010

Shift left 0 0 0 0 1 0 1 0 ☐

$A \leftarrow A - M$ 1 1 1 0 1

Add [2's Complement of M & add to A]

1 1 1 1 0 Check Sign bit If (1, then $q_0 = 0$)

then restore

$A \leftarrow A + M$ 0 0 0 1 1 0 1 0 $\boxed{0}$ q_0

Restored A value 1 Carry ignore 0 0 0 0 1 both Same

1 cycle

Add one 0

2's Complement

0 0 0 1 1

1 1 1 0 0

1

1 1 1 0 1

Shift left 0 0 0 1 0 1 0 0 ☐

$A \leftarrow A - M$ 1 1 1 0 1

1 1 1 1 1

② cycle

$A \leftarrow A + M$ (Add) 0 0 0 1 1 1 0 $\boxed{0}$ $\boxed{0}$

1 ignore 0 0 0 1 0

Scanned by CamScanner

0 0 0 1 0

Shift left 0 0 1 0 1

3 cycle 0 0 0 $\square$

$A \leftarrow A-M$ 1 1 1 0 1

$A \leftarrow A+M$ 1 | 0 0 0 1 0

If Signbit = 0 then assign

No restoration is required.

0 0 0 1

Shift left 0 0 1 0 0

0 0 1 $\square$

$A \leftarrow A-M$ 1 1 1 0 1

1 | 0 0 0 0 1

4 cycle

ignore

No Restoration is required.

A = 00001

0 0 0 1

Remainder = 00001

Q = Quotient 0001

Example (2) Perform 1100 ÷ 11 using restoring algorithm

Dividend = 1100 (Q)
Divisor = 11 (M)

	A	Q
	0 0 0 0 0	1 1 0 0
SL	0 0 0 0 1	1 0 0 $\boxed{\ }$
A = A − M	1 1 1 0 1	
	1 1 1 1 0	
A = A + M	0 0 0 1 1	
	ignore 1) 0 0 0 0 1	1 0 0 $\boxed{0}$
SL	0 0 0 1 1	0 0 $\boxed{0}$ $\boxed{\ }$
A = A − M	1 1 1 0 0	
	1 1 1 1 1	
A = A + M	0 0 0 1 1	0 0 $\boxed{0}$ $\boxed{0}$
	ignore) 0 0 0 1 0	
SL	0 0 1 0 0	0 $\boxed{0}$ $\boxed{0}$ $\boxed{\ }$
A = A − M	1 1 1 0 1	
	~~1~~) 0 0 0 0 1	
No restoration		0 $\boxed{0}$ $\boxed{0}$ $\boxed{1}$
SL	0 0 0 1 0	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{\ }$
A = A − M	1 1 1 0 1	
	1 1 1 1 1	
A = A + M	0 0 0 1 1	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{0}$
	~~1~~) 0 0 0 1 0	

M =
00011
11100
1
11101

$\boxed{\text{Remainder} = 00010}$ $\boxed{\text{Quotient} = 0010}$

Example ③ Dividend Q = 1000

Divisor M = 0011

Count = ~~4~~ ~~3~~ ~~2~~ 1 0

	A	Q
Initially	0 0000	1000
Shift	0 0 0 0 1	0 0 0 $\square$
A ← A − M.	1 1 1 0 1	
	1 1 1 1 0	0 0 0 $\boxed{0}$
A ← A + M	0 0 0 1 1	
Restore	1] 0 0 0 0 1 (ignore)	
Shift	0 0 0 1 0	0 0 0 $\square$
A ← A − M	1 1 1 0 1	
	1 1 1 1 1	0 0 0 $\boxed{0}$
A ← A + M	0 0 0 1 1	
	1] 0 0 0 1 0 (ignore)	

$$\begin{array}{r} 00011 \\ 11100 \\ 1 \\ \hline 11101 \end{array}$$

0 0 0 1 0   0 0 0 $\boxed{0}$

Shift   0 0 1 0 0   0 0 0 $\boxed{}$

A ← A - M.   1 1 1 0 1

1 | 0 0 0 0 1

ignore

3 cycle

0 0 0 $\boxed{1}$

Shift   0 0 0 1 0   0 0 1 $\boxed{}$

A ← A - M   1 1 1 0 1

1 1 1 1 1

4 cycle

A ← A + M   0 0 0 1 1   0 0 1 $\boxed{0}$

1 | 0 0 0 1 0

ignore.

$\boxed{\text{Remainder = 00010}}$ $\boxed{\text{Quotient = 0010}}$

Example ①

Dividend = 1010 (Q)

Divisor = 0011 (M)

Count = ~~4~~ ~~3~~ ~~2~~ ~~1~~ 0

M = 00011

11101

	A	Q	
	0 0 0 0 0	1 0 1 0	1st cycle
Shift left	0 0 0 0 1	0 1 0 $\boxed{\ }$	
A ← A − M	1 1 1 0 1		
	1 1 1 1 0	0 1 0 $\boxed{0}$	
SL	1 1 1 0 0	1 0 $\boxed{0}$ $\boxed{\ }$	2nd cycle
A ← A + M	0 0 0 1 1		
	1 1 1 1 1	1 0 $\boxed{0}$ $\boxed{0}$	
SL	1 1 1 1 1	0 $\boxed{0}$ $\boxed{0}$ $\boxed{\ }$	3rd cycle
A ← A + M	0 0 0 1 1		
	0 0 0 1 0	0 $\boxed{0}$ $\boxed{0}$ $\boxed{1}$	
SL	0 0 1 0 0	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{\ }$	4th cycle
A ← A − M	1 1 1 0 1		
	0 0 0 0 1	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{1}$	

Remainder = 0 0 0 0 1

Quotient = 0011

Example ② = 1011 (Q)
Divisor = 0101 (M)

Count = 4

M = 00101
11010
+1
11011

		A	Q
		0 0 0 0 0	1 0 1 1
1st	SL	0 0 0 0 1	0 1 1 □
	A ← A − M	1 1 0 1 1	
		1 1 1 0 0	0 1 1 $\boxed{0}$
2nd	SL	1 1 0 0 0	1 1 $\boxed{0}$ □
	A ← A + M	0 0 1 0 1	
		1 1 1 0 1	1 1 $\boxed{0}$ $\boxed{0}$
3rd	SL	1 1 0 1 1	1 $\boxed{0}$ $\boxed{0}$ □
	A ← A + M	0 0 1 0 1	
		X) 0 0 0 0 0	1 $\boxed{0}$ $\boxed{0}$ $\boxed{1}$
4th	SL	0 0 0 0 1	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ □
	A ← A − M	1 1 0 1 1	
		1 1 1 0 0	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{0}$

If 1, then add divisor with A

1 1 1 0 0 (A)
0 0 1 0 1 (divisor)
X) 0 0 0 0 1

Quotient = 0010

Remainder = 00001

Example ③ – Using non restoring division algorithm, compute 23/5

Dividend → 23 ⇒ 10111 (Q)

Divisor → 5 ⇒ 00101 (M)

Count = ~~5~~ ~~4~~ ~~3~~ ~~2~~ 1

	M	2's comp
	000101	111010
		+ 1
		111011

Step	A	Q
	000000	10111
SL	000001	0111 $\boxed{0}$
A ← A − M	111011	
	111100	
SL	111000	111 $\boxed{0}$ $\boxed{0}$
A = A + M	000101	
	111101	
SL	111011	11 $\boxed{0}$ $\boxed{0}$ $\boxed{1}$
A = A + M	000101	
	✗) 000000	
SL	000001	1 $\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{0}$
A ← A − M	111011	
	111100	
SL	111001	$\boxed{0}$ $\boxed{0}$ $\boxed{1}$ $\boxed{0}$ $\boxed{0}$
A = A + M	000101	
	111110	
+ Divisor	000101	
	✗) 000011	

Quotient = 00100

Remainder = 000011