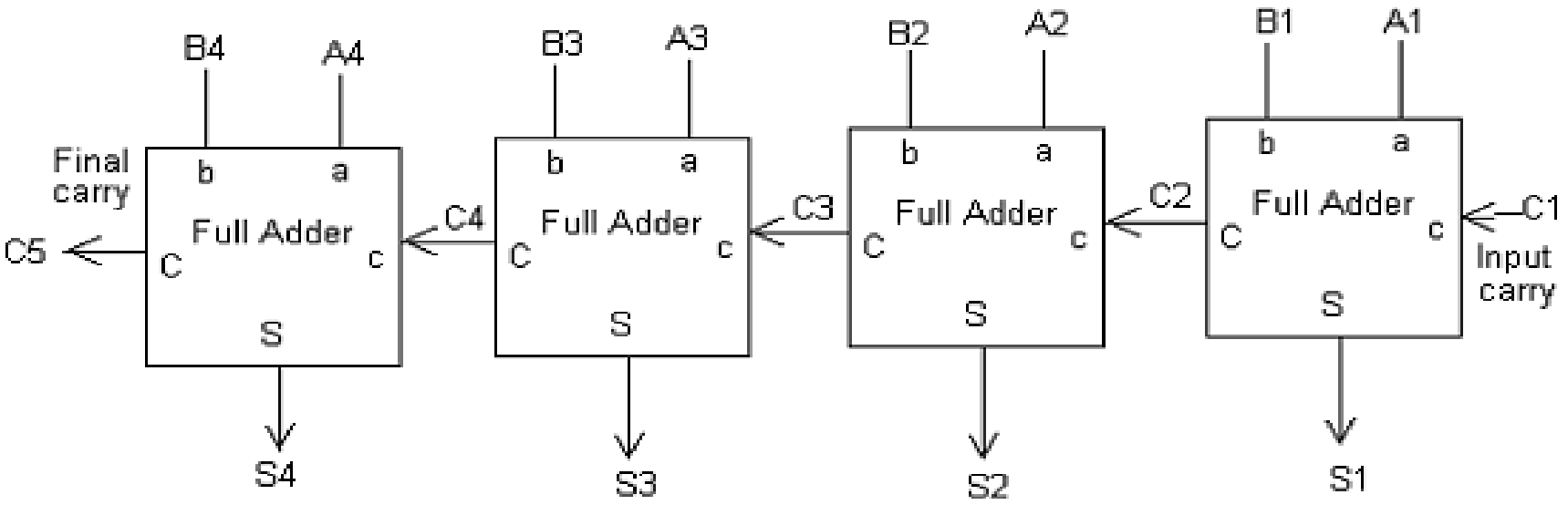
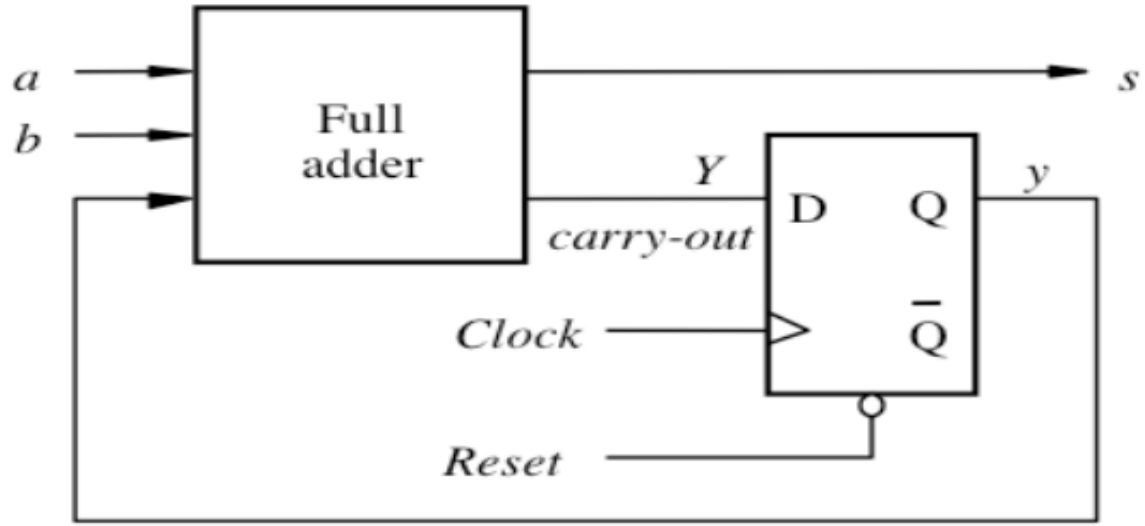
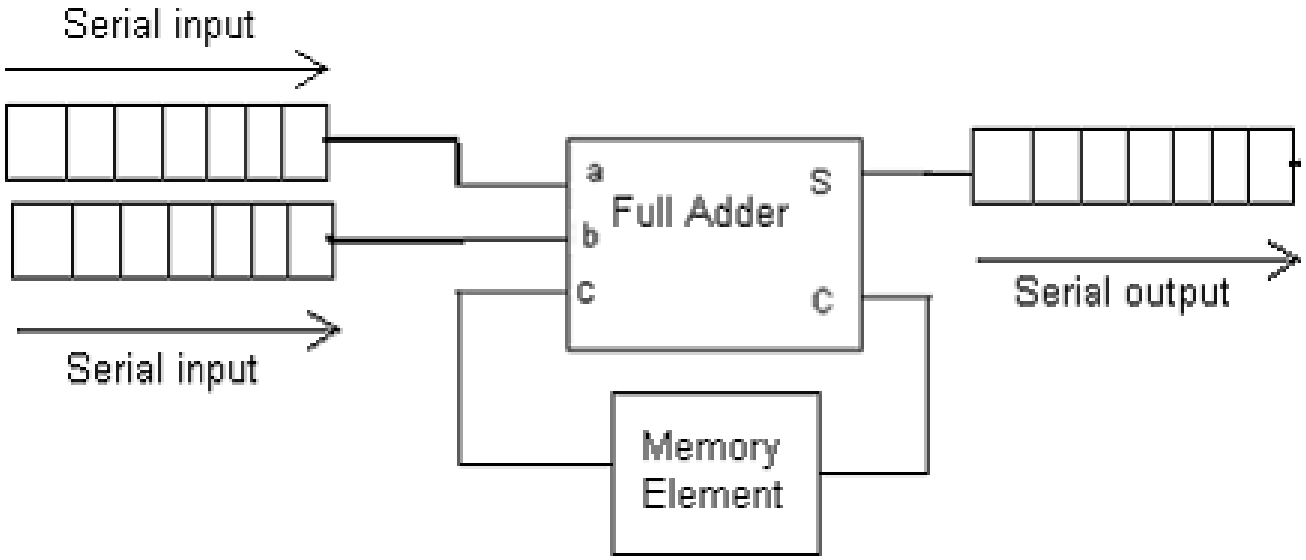
Parallel Adder: Ripple Carry Adder (Carry Look-ahead adders also is parallel adder)



Serial Adder





Multiplication

- 1. **For unsigned/+tive integers**

Sequential Circuit Binary Multiplier

(based on multiplication by hand)

1101---M (Multiplicand) (+13) 4bit

1011---Q (Multiplier) (+11) x 4bit

1101 1101x2^0 x1 8 bits at most

1101 1101x2^1 x1

0000 1101x2^2 x 0

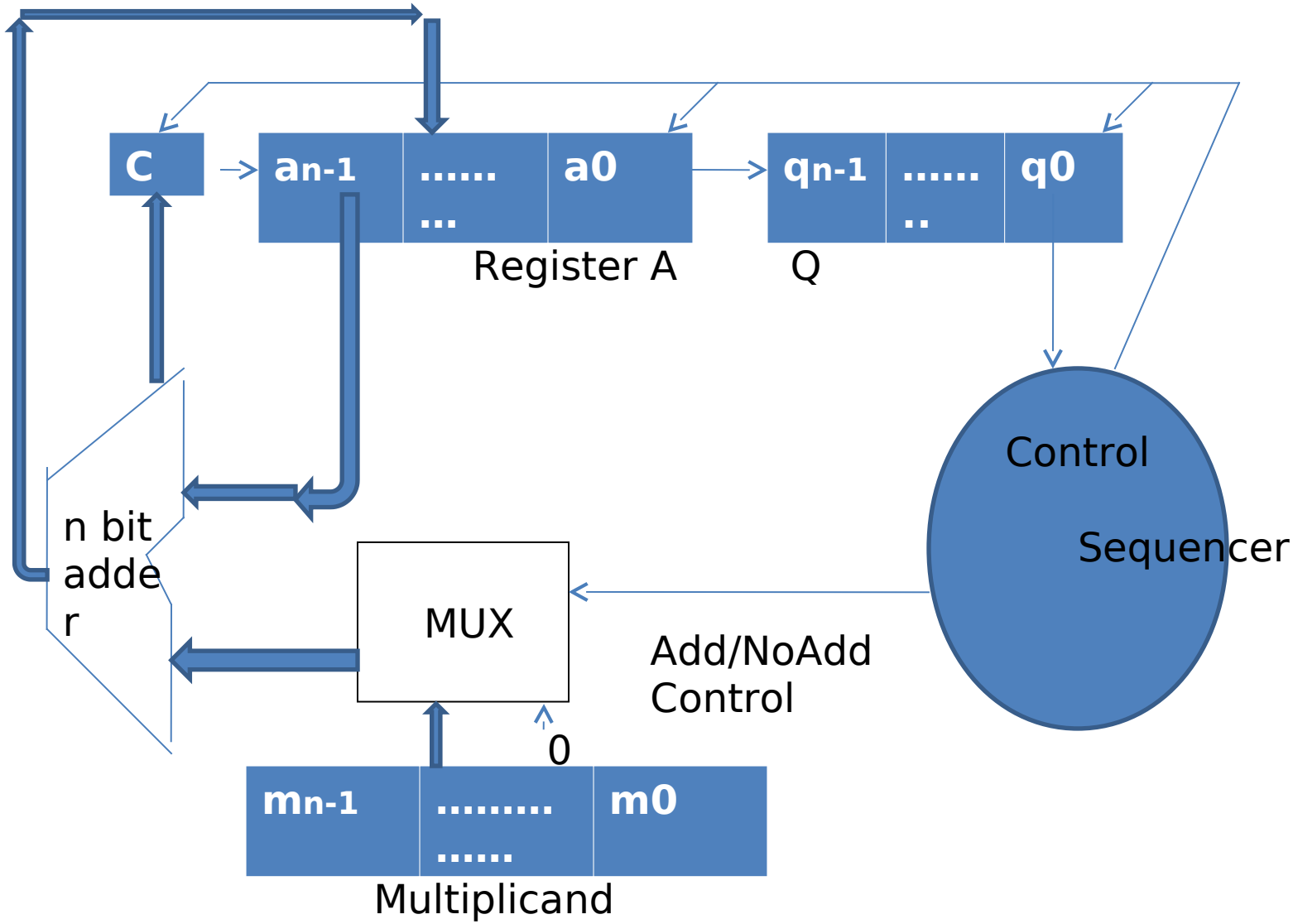
11 01 1101x2^3 x 1

10001111 (143)

4bit x4bit multiplication need 4bit adder repeat addshift /shift 4 times

C		1101	M	Q		Initial State	13
0		0000	A	1011			11
0		1101			Add M	Cycle 1	
0		1101		1011			
0		0110		1101	Shift		
0		1101		1101	Add	Cycle 2	
1		0011		1101			
0		1001		1110	Shift		
0		0100		1111	Shift	Cycle 3	
0		1101		1111	Add M	Cycle 4	
1		0001		1111			
0		1000		1111	Shift		143

Register Configuration Sequential Circuit



Multiplication

- 2. **For signed number in 2's complement**

Booth Algorithm

Facts used in this algorithm

$Q=001110 \quad = 2^3 + 2^2 + 2^1 = +14 = 2^4 - 2^1$

$M \times Q = M \times 2^4 - M \times 2$

M (Multiplicand) 4bit

Q (Multiplier) x 4bit

8 bits product

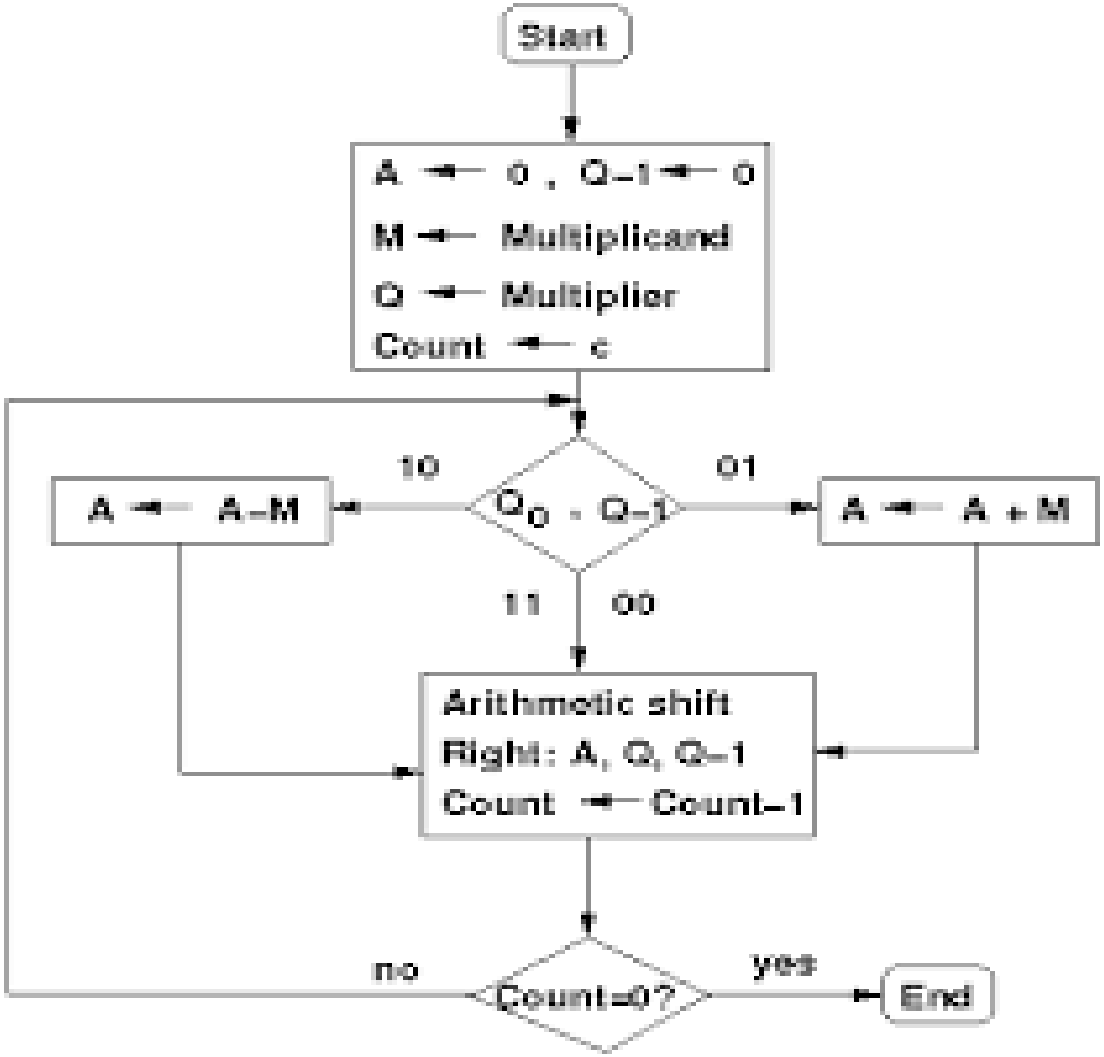
4 cycles to finish calculation

Sign extension needed to keep track of sign

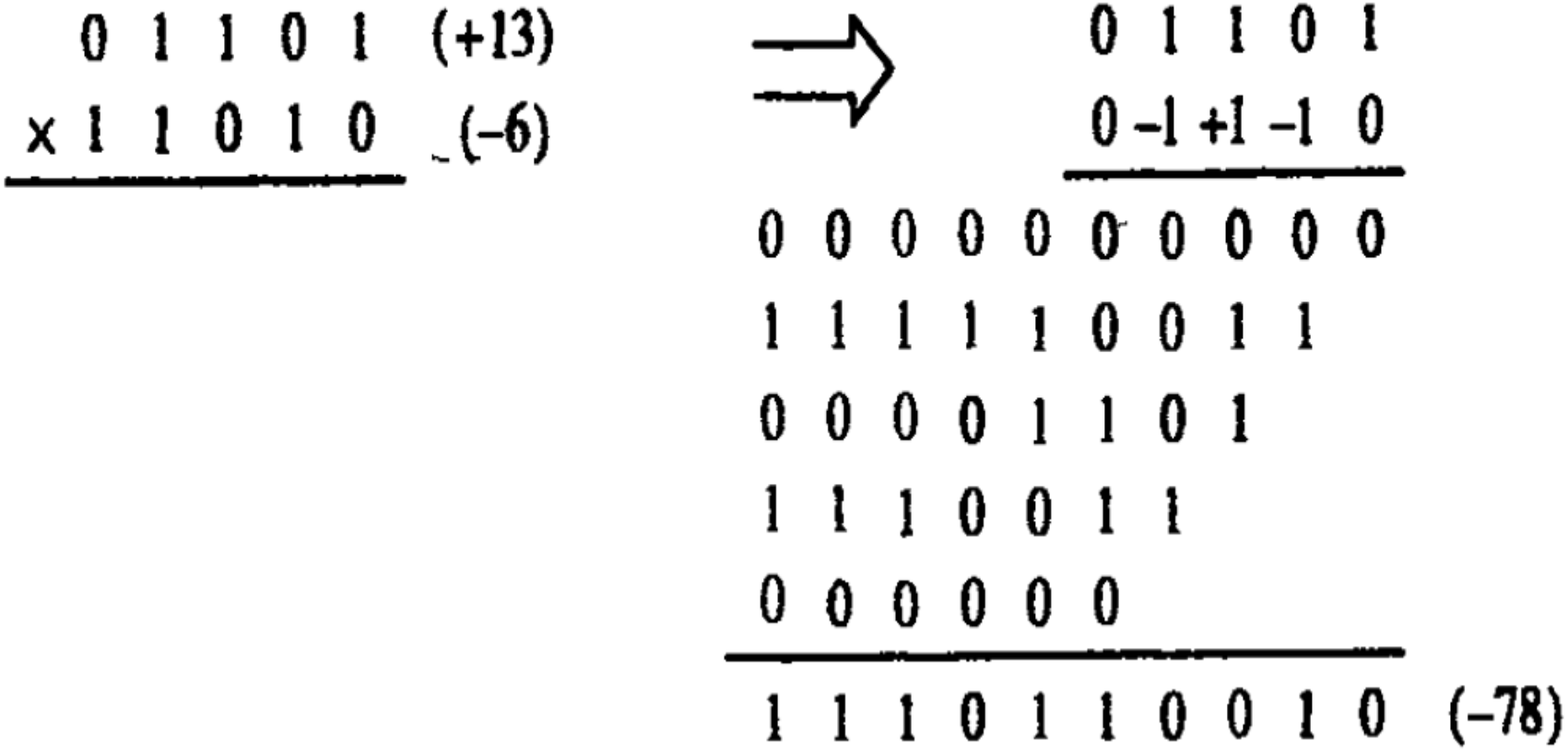
Booth's Algorithm M x Q = 7 x 3

A	Q	Q_{-1}	M	-M	Initial State	7
0000	0011	0	0111	1001		3
1001	0011	0	A-M		Cycle 1	
1001	0011	0				
1100	1001	1	Arithmetic RightShift			
1110	0100	1	ARS		Cycle 2	
0111	0100	1	A+M		Cycle 3	
0101 0010	0100 1010	1 0	ARS			
0001 Prod	0101 uct	0	ARS		Cycle 4	21

Flowchart of Multiplication using Booth's Algorithm



Multiplication (13 x -6) using Booth's Algorithm

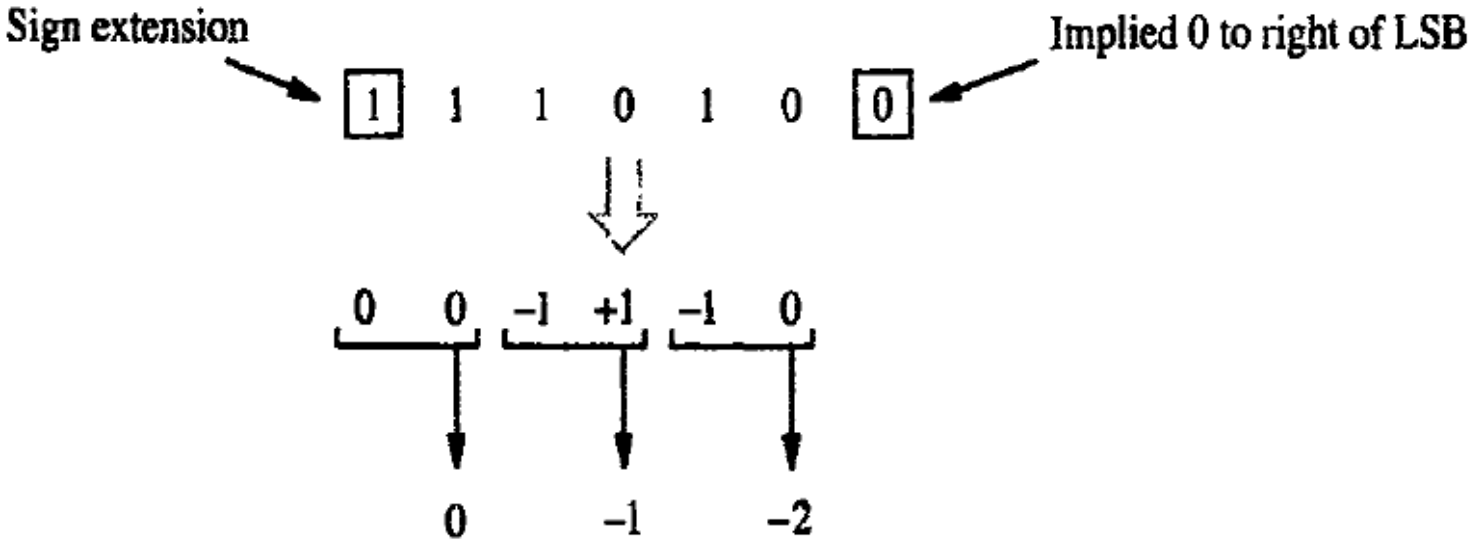


Multiplier bit-pair		Multiplier bit on the right	Multiplicand selected at position i
$i+1$	i	$i-1$	
0	0	0	$0 \times M$
0	0	1	$+1 \times M$
0	1	0	$+1 \times M$
0	1	1	$+2 \times M$
1	0	0	$-2 \times M$
1	0	1	$-1 \times M$
1	1	0	$-1 \times M$
1	1	1	$0 \times M$

(b) Table of multiplicand selection decisions

Figure 6.14 Multiplier bit-pair recoding.

Bit-pair Recoding of (-6) using Booth Recoding



(a) Example of bit-pair recoding derived from Booth recoding

Multiplication (13 x -6) using bit-pair recoding of Multipliers using Booth Recoding

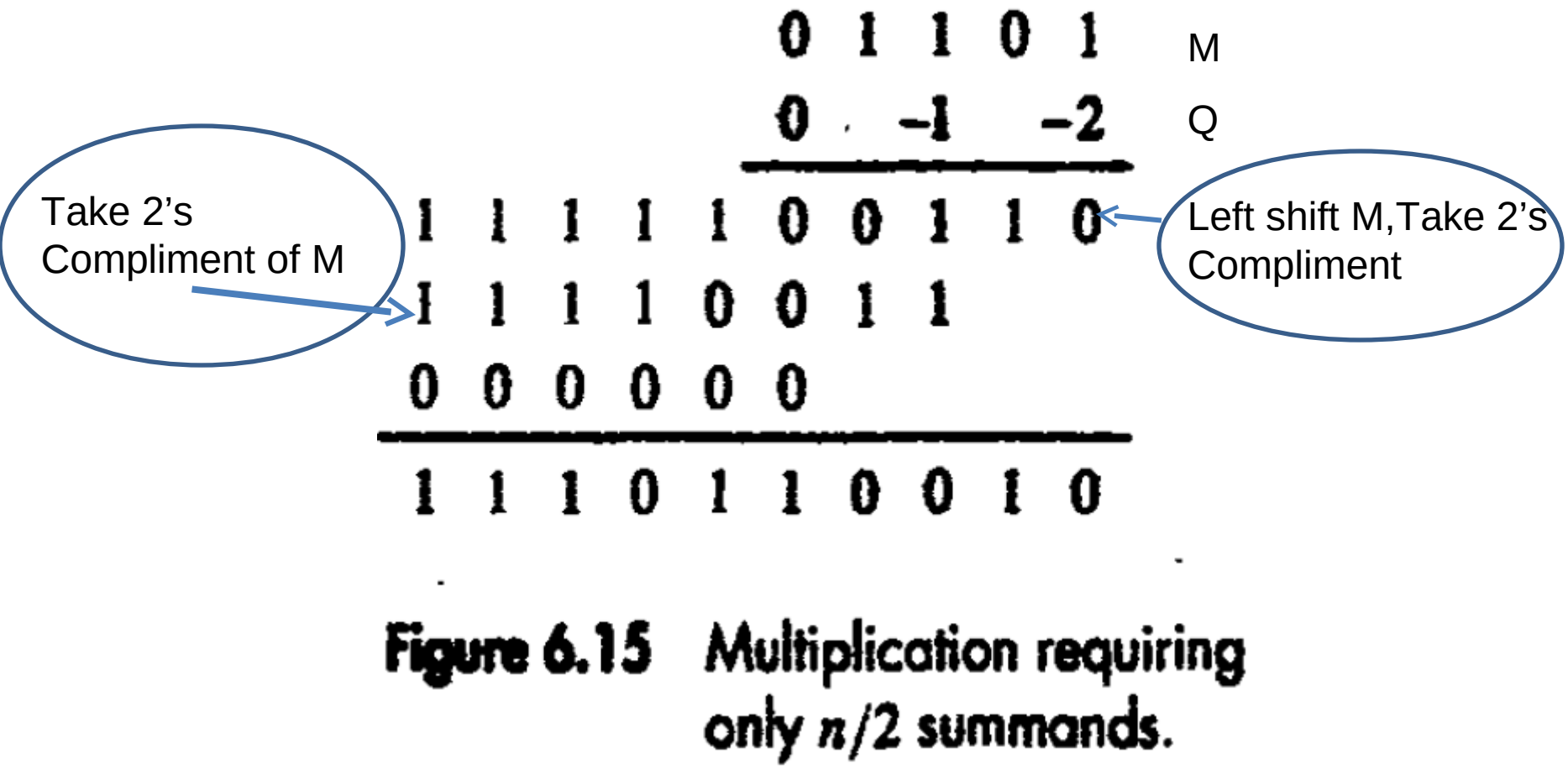


Figure 6.15 Multiplication requiring only $n/2$ summands.