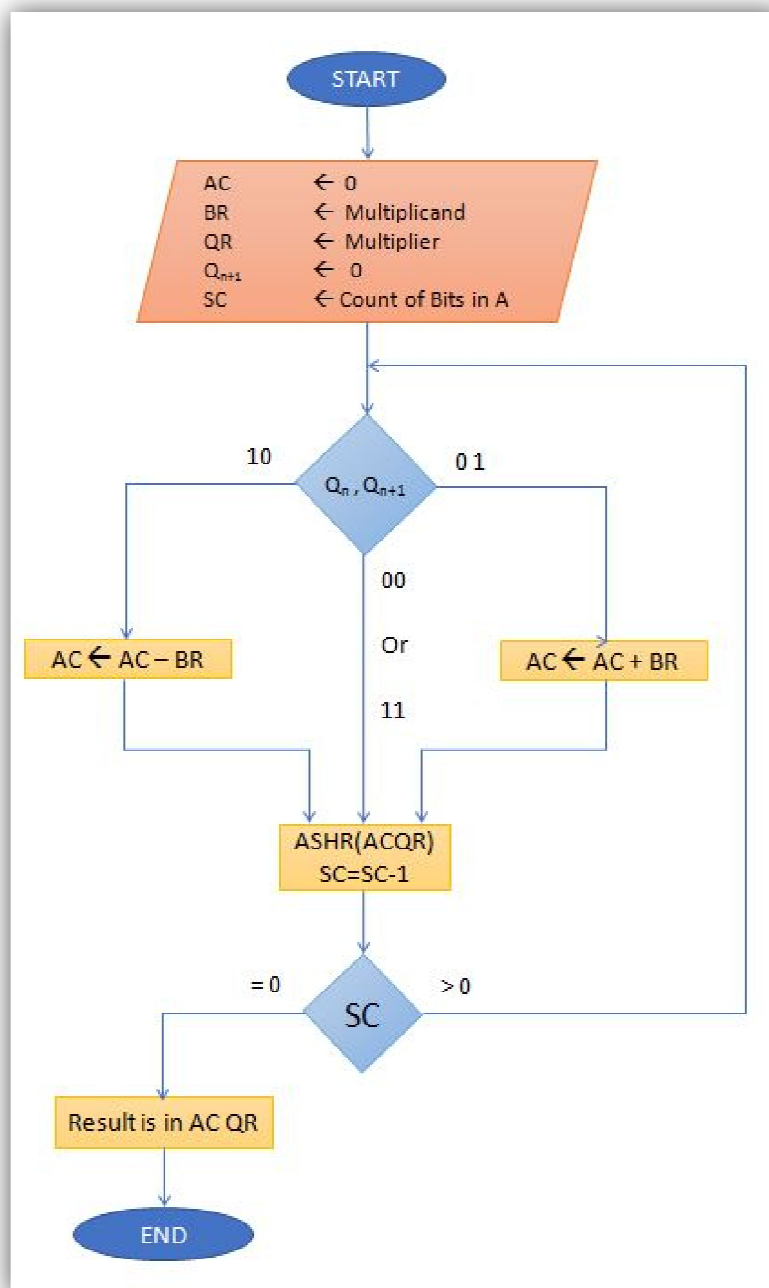


Booth Algorithm: As an advanced method, Booth algorithm is developed for multiplication of signed numbers. The positive numbers are as usual while negative numbers are taken already in 2's complement format. Multiplicand is stored in B and to simplify the A-B operation, 2's complement of B is also computed. Multiplier is again in Q, A works as Accumulator. Another flip flop Q_{n+1} is also taken and initialized by a zero. The flip-flop stores the discarded LSB of Q after shifting AQ. The shift operation is Arithmetic and hence ASHR is followed. SC is decremented after every shift operation.

Consider the following flow-chart:



The following example shows the entire operation with multiplicand (+13) and multiplier (+17). The result that we get at last is (+221):

Multiplicand [B] = (+13)₁₀ = (001101)₂ 2's complement of [B] = (-13)₁₀ = (110011)₂
 Multiplier [Q] = (17)₁₀ = (010001)₂ Accumulator [A] = 000000 SC=6 Q_{n+1} = 0

Q _n	Q _{n+1}	Description	A	Q	Q _{n+1}	SC
			000000	010001	0	110
1	0	A ← A - B	000000	010001	0	110
			110011			
		ASHR(AQ), SC ← SC - 1	110011	010001	0	101
			111001	101000	1	
0	1	A ← A + B	111001	101000	1	100
			001101			
		ASHR(AQ), SC ← SC - 1	000110	010100	0	
0	0	ASHR(AQ), SC ← SC - 1	000011	010100	0	011
0	0	ASHR(AQ), SC ← SC - 1	000001	101010	0	010
1	0	A ← A - B	000000	110101	0	001
			110011			
		ASHR(AQ), SC ← SC - 1	110011	110101	0	001
			111001	111010	1	
0	1	A ← A + B	111001	111010	1	000
			001101			
		ASHR(AQ), SC ← SC - 1	000110	011101	0	
			000011	011101	0	

(000011011101)₂ = (+221)₁₀

Assignment:

1. Draw the flow chart of Booth's multiplication algorithm.
2. Multiply (+24) and (-21) using Booth's algorithm.