9 Sept

TUTORIAL - 1

(U19CS012) (COMPUTER ORGANISATION) (CO)

[comp. = complement]

1.) (d) 00101
11010 $\xrightarrow{1's\ comp.}$ 00101

2.) (a) True (2's comp. = 1's comp. + 1)

3.) (b) Binary Numbers (binary equivalent's of decimal numbers

4.) (b) Booth's Algorithm

5.) (d) Multiplier (a (Multiplicand) * b (multiplier))

6.) (A) 6
M Q
(-2) * (-3) Register size = 4 bits
= + (Product)

M 0010 $M \rightarrow (-2)_{10} \xrightarrow{2's\ comp.} (1110)_2$

1's 1101
2's + 1 = 1110

$-M \rightarrow (2)_{10} \rightarrow (0010)_2$

Q 0011 (-8 4 2 1)

1's 1100
2's + 1 = 1101

$Q \rightarrow (-3)_{10} \rightarrow (1101)_2$

	AC	Q	Q_{-1}	operation
1st	0000	1101 (Q_0)	0	i) AC = AC - M 0000 + 0010 (0010)$_2$
	0010 → 0001	1101 → 0110	0	
			1	ii) ASR

(Preserved)

	AC	Q	Q_{-1}	Operation
2nd →	0001	0110	1	Cold I
	1111	0110	1	(i) AC = AC + M
	1111	1011	0	0001 + 1110 = $(1111)_2$
				(ii) ASR
3rd	0001	1011	0	(i) AC = AC - M
	0000	1101	1	①① 1111 + 0010 *0001
				(Discard carry)
				(ii) ASR
4th step	0000	0110	1	(i) ASR

Ans is $(0000\ 0110)_2 = (6)_{10}$

7.) (c) -21

$(-7)_{10} \times (3)_{10}$ register bits = ⑤

$M \rightarrow (-7)_{10} \rightarrow (11001)_2$

$-M \rightarrow (7)_{10} \rightarrow (00111)_2$

$Q \rightarrow (3)_{10} \rightarrow (00011)_2$

(7) 00111
1's 11000
+ 1
2's (11001)

	AC	Q Q_0	Q_{-1}	Operation
Step①	00000	0001①	0	
	00111	0011	0	(i) AC = AC - M 00000 + 00111 $(00111)_2$
	00011	1000①	1	

	AC	Q	Q_{-1}	Operation
Step 2:				(i) ASR
	0 0001	1 1000	1	
Step 3:				
	1 1010	1 1000	1	(i) AC = AC + M
				0 0001
				+ 1 1001
	1 1101	0 1100	0	1 1010
				(ii) A.S.R
Step 4:				
	1 1110	1 0110	0	(i) ASR
Step 5:				(i) ASR
	1 1111	0 1011		

$-7 \times 3 = (-ve)$ → 2's comp.

2's $(1\,1111\;0\,1011)_2$

1's 0 0000 1 0100

\+ 1

2's $(0\,0000\;1\,0101)_2$ ← Final Ans

(16 8 4 2 1 2)

$-7 \times 3 = \boxed{-21} = (0\,0000\;1\,0101)_2$

8.) Difference between Shift Right (Logical) and Arithmetic shift right (Most Significant Bit)

(A) Logical Shift Right: Shifting the bits to the right and MSB becomes 0.

Example: Logical shift right (n) 1011 0101

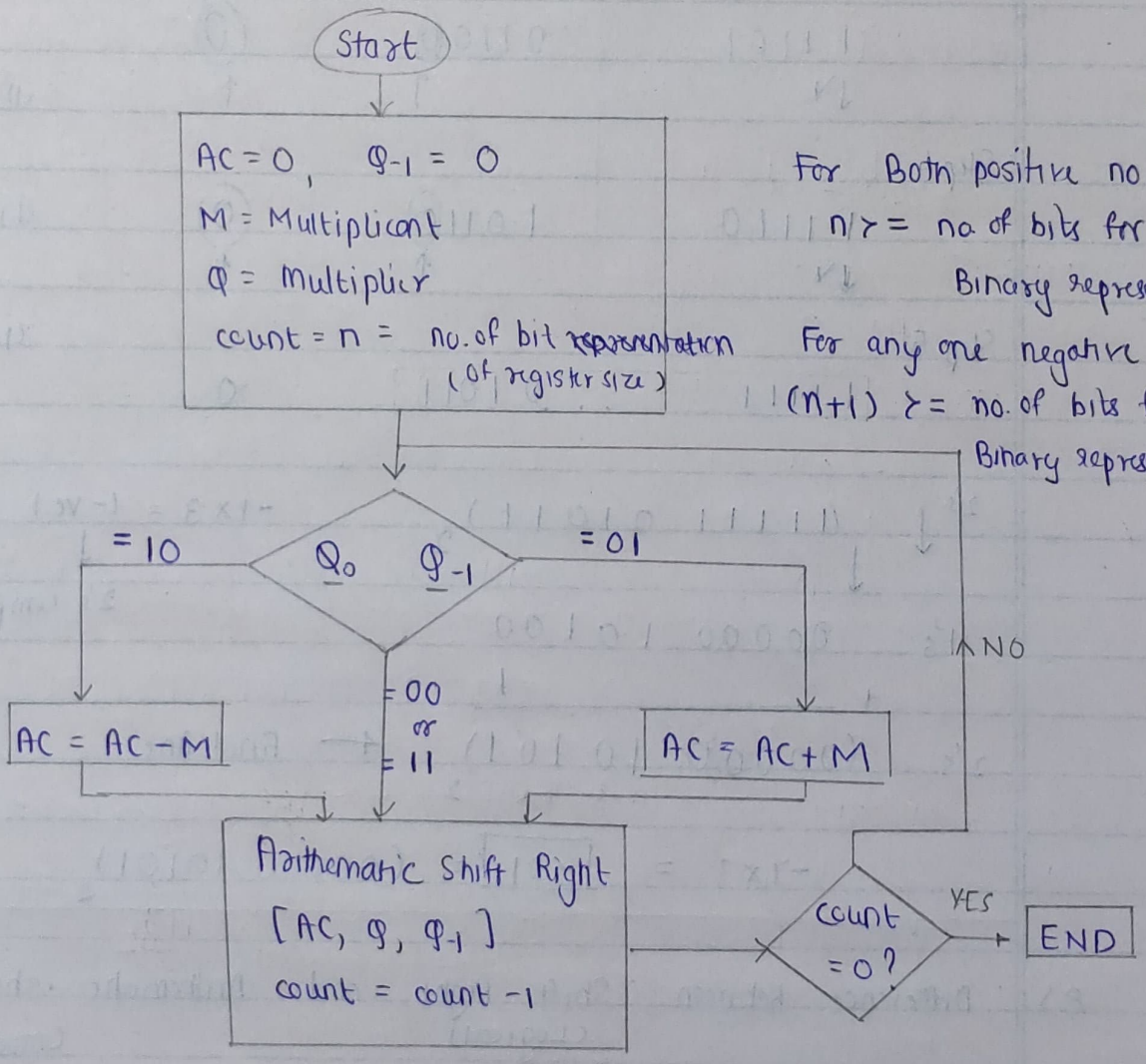
(n >> 1) 0 1011 010

sion

(B) Arithematic Right Shift : means shifting the bits to right and MSB(most significant Bit) is same as MSB of original number.

Eg: 10110101 $\xrightarrow{MSB}$ 1 1011010

9.) Booth's Flowchart



For Both positive no.'s
n/x = no of bits for Binary representation
For any one negative
(n+1) x = no. of bits for Binary representation

Q_0 = last bit of Q
AC = Accumulator

① Intialise cond^n for AC, Q_{-1}, find, M, -M, Q
② Check cond^n Q_0 Q_{-1} → 10 → AC = AC-M
3 cases acc. to flowchart → 00/11
→ 10 → AC = AC+M
③ A.S.R of AC, Q, Q_{-1} & decrement count
④ if count == 0 → End otherwise continue

10.) (A) 7×5 $n = 4$ = size of register

$= (35)_{10}$ M = binary of $(7)_{10}$ = $(0111)_2$

$Q = (5)_{10} = (0101)_2$

$-M = (-7)_{10} = (1001)_2$

0111
1's 1000
+ 1
2's (1001)

AC	Q	Q-1	operation
(step1)			
0000	0101	0	(i) AC = AC - M 0000 + 1001 $(1001)_2$
1001 ↓	0101	0	
1100	1010	1	(ii) A.S.R.
Step2:			(i) AC = AC + M 1100 + 0111 X0011
0011 ↓	1010	1	
0001	1101	0	(ii) A.S.R
Step3:			
1010 ↓	1101	0	(i) AC = AC - M 0001 + 1001 1010
1101	0110	1	(ii) A.S.R
Step4:			(i) AC = AC + M 1101 + 0111 X0100
0100 ↓	0110 ↓	1	
0010	0011	0	(neglect carry)
			(iii) A.S.R

ANS: $(00100011)_2 = (35)_{10}$

10.) (B) (3) * (-6) n = 4 bits

$= (-18)_{10}$

$M = (3)_{10} = (0011)_2$

$-M = (-3)_{10} = (1101)_2$

$(Q) = (-6)_{10} = (1010)_2$

8 4 2 1

6 0 1 1 0

1's 1 0 0 1

+ 1

(2's) 1 0 1 0

0011

1's 1100

+ 1

1101

	AC	Q	Q_{-1}	Operation
(Step 1)	0000	1010	0	
	↓↓			(i) ASR
	0000	0101	0	
		↑	↑	
(Step 2)				(i) AC = AC - M
	1101	0101	0	0000
	↓↓			+ 1101
	1110	1010	1	$(1101)_2$
		↑	↑	(ii) A.S.R.
(Step 3)	0001	1010	1	(i) AC = AC + M
	↓↓			1110
				+ 0011
				X0001
				(neglect carry)
	0000	1101	0	(ii) A.S.R
		↑	↑	
(Step 4)				(i) AC = AC - M
	1101	1101	0	0000
	↓↓			+ 1101
	1110	1110	1	$(1101)_2$
				(ii) A.S.R

(+) * (-ve) = (-ve) (1110 1110) (-ve number)

1's 0001 0001

+ 1

$(0001\ 0010)_2 = (-ve)(18)_{10}$

Ans: $(3)_{10} \times (-6)_{10} = (-18)_{10}$

10.X (c) $(-2) * (4)$ (M, Q)

n = 4 bits

0010

1's 1101

\+ 1

2's 1110

$M = (-2)_2 = (1110)_2$

$-M = (2)_2 = (0010)_2$

$Q = (4)_2 = (0100)_2$

	AC	Q	Q-1	Operations
(Step 1)	0000	0100	0	
	↓	↑	↑	(i) A.S.R.
	0000	0010	0	
	↓	↑	↑	
(Step 2)	0000	0001	0	(i) A.S.R
		↑	↑	
				(i) AC = AC - M
				0000
	0010	0001	0	+ 0010
	↓			(0010)
	0001	0000	1	(ii) A.S.R
		↑	↑	
(Step 3)				(i) AC = AC + M
				0001
	1111	0000	1	+ 1110
	↓			(1111)
	1111	1000	0	(ii) A.S.R
	↓	↑	↑	
Step 4	1111	1100	0	(i) A.S.R

(+ve) × (-ve) = (-ve)

$(1111\ 1100)_2$ (-ve)

1's $(0000\ 0011)_2$

\+ 1

$(0000\ 0100)_2 = (-8)_{10}$

Ans: $(-2)_{10} \times (4)_{10} = (-8)_{10}$

(U19CS012)

(D) $(-5) \times (-3)$ (M = −5, Q = −3)

$\Rightarrow$ n = 4 bits : no. of registers

$M = (-5)_{10} = (1011)_2$

$-M = (5)_{10} = (0101)_2$

$Q = (-3)_{10} = (1101)_2$

$(0101)_2$

1's 1010

\+ 1

(1011)

0011

1's 1100

\+ 1

2's 1101

	AC	Q	Q_{-1}	Operation
(Step 1)	0000	1101	0	(i) AC = AC − M 0000 + 0101 $(0101)_2$
	0101	1101	0	(ii) A.S.R
	0010	1110	1	
(Step 2)				(i) AC = AC + M 1 0010 + 1011 1101
	1101	1110	1	
	1110	1111	0	(iii) A.S.R
(Step 3)	0011	1111	0	(i) AC = AC − M 1110 + 0101 X0011 (neglect carry)
	0001	1111	1	(ii) A.S.R
(Step 4)	0000	1111	1	(i) A.S.R

(−ve) × (−ve) = (+ve)

$(0000\ 1111)_2 = (15)_{10}$

Ans: $(-5)_{10} \times (-3)_{10} = (15)_{10}$

Submitted By:
U19CS012 (D-12)