SOLUTION (Digital Logic Design)

ST-2 odd REC-301
17-18

Section A

Ans 1.)

Combinational circuit

*) Output is a function of present inputs

*) Logic Gates are the elementary building blocks.

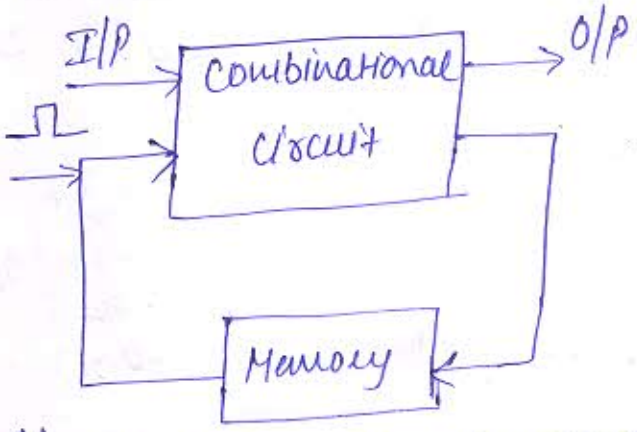
*) Used mainly for arithmetic and boolean operations.

*) Independent of clock and hence does not Require triggering to operate.

*) Example → Adder

Sequential circuit

*) Output is a function of clock, present inputs and previous state of the system.



*) Flip-Flops are elementary building blocks.

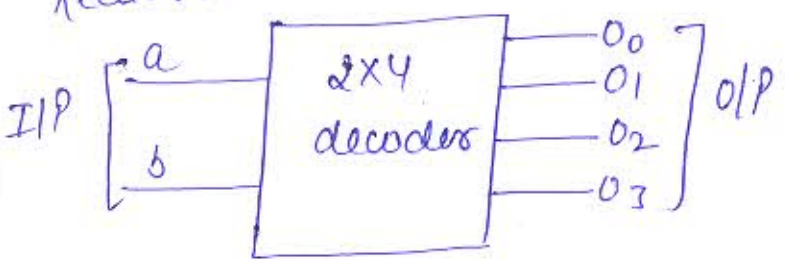
*) Used for storing data.

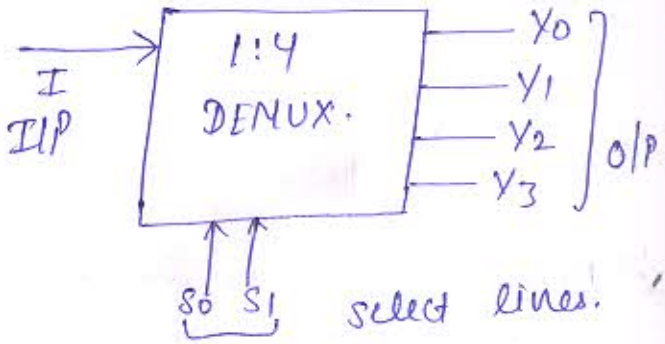
*) Clocked (Triggered for operation with electronic pulses)

*) Example → Counter.

(Ans 2) Demultiplexer → On the basis of Inputs that are given to the select lines, it fetches the data provided at the input line and provides it to the one of the output lines. The selection of output line is determined on the basis of Inputs provided to the select lines.

*) Decoder → A decoder does not have any select lines. It has n input lines and m output lines and one enable signal. Decoder performs the function of decoding i.e it translates the input in some other form so that the output can be understandable by the Receiver.





(Ans 4) No. of Flip Flops Required in MOD 16 synchronous counter ⇒ (counts from 0 to 15) 4 Flip Flops.

*) No. of Flip Flops Required in MOD-16 Johnson counter ⇒ 5 Flip Flops.

(Ans 5) Race-Round condition → In JK Flip Flop when J=1, K=1 and Clk=1, output will toggle b/w 0 and 1. Thus the output will be unstable creating a race-round problem with the basic J-K circuit.

condition to avoid Race Round Problem →

*) Pulse width <= Flip Flop delay <= Time period of clock.

*) By using Master Slave JK Flip Flop.

Ans 3) Difference between Latches and Flip Flop.

Latch	Flip Flop.
*) Latches continuously checks its input and changes its output correspondingly.	*) They continously checks its inputs and changes its output correspondingly only at times determined by clocking signal
*) It is based on enable function i/p.	*) It works on basis of clock pulses.
*) These can be build from logic gates.	*) These can be build from latches.
*) It is a level triggered. It means that the O/P of the present state and input of the next state depends on level i.e 1 and 0	*) It is an edge triggered. It means that the output and next state i/p changes when there is change in clock pulse whether it may a +ve or -ve clock pulse.

Section B.

(Qn 6) BCD to Excess-3 Code Converter

Decimal	BCD A	B	C	D	Excess-3 W	X	Y	Z
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0

For W.

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	X	X	X	X
10	1	1	X	X

$W \Rightarrow A + BD + BC$

For X.

AB \ CD	00	01	11	10
00	0	1	1	1
01	1	0	0	0
11	X	X	X	X
10	0	1	X	X

$X \Rightarrow B\bar{C}\bar{D} + \bar{B}D + \bar{B}C$

For Y →

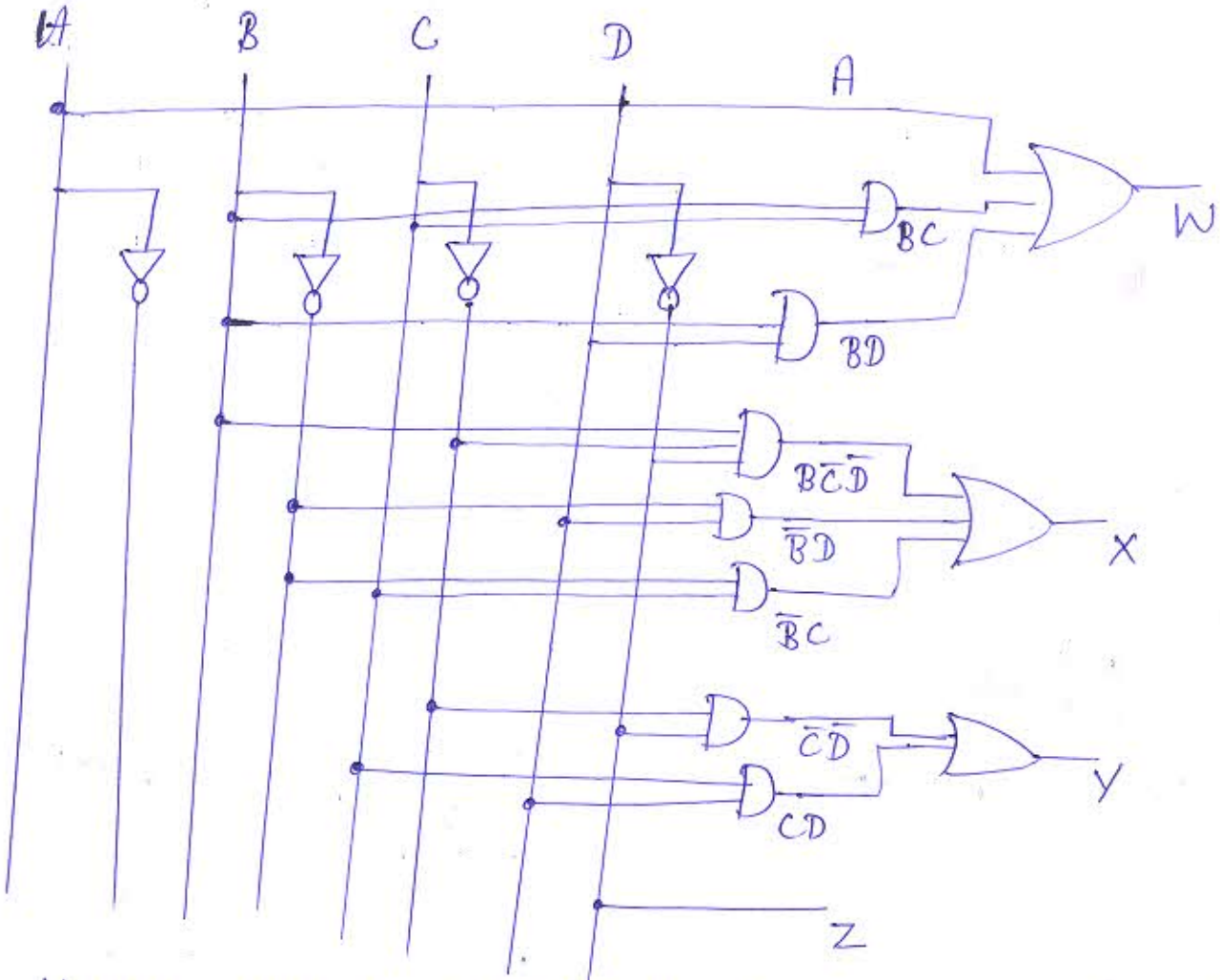
AB \ CD	00	01	11	10
00	1	0	1	0
01	1	0	X	0
11	X	X	X	X
10	1	0	X	X

$Y \Rightarrow \bar{C}\bar{D} + CD.$

For Z →

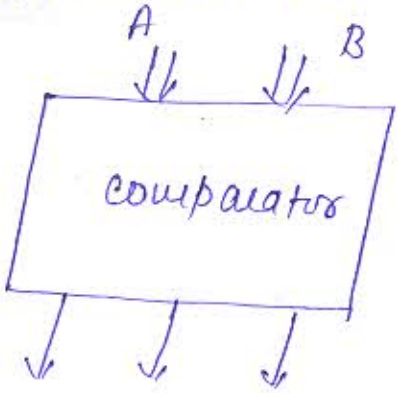
AB \ CD	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	X	X	X	X
10	1	0	X	X

$Z \Rightarrow \bar{D}$



*) Logic circuit of BCD to Excess-3 code converter.

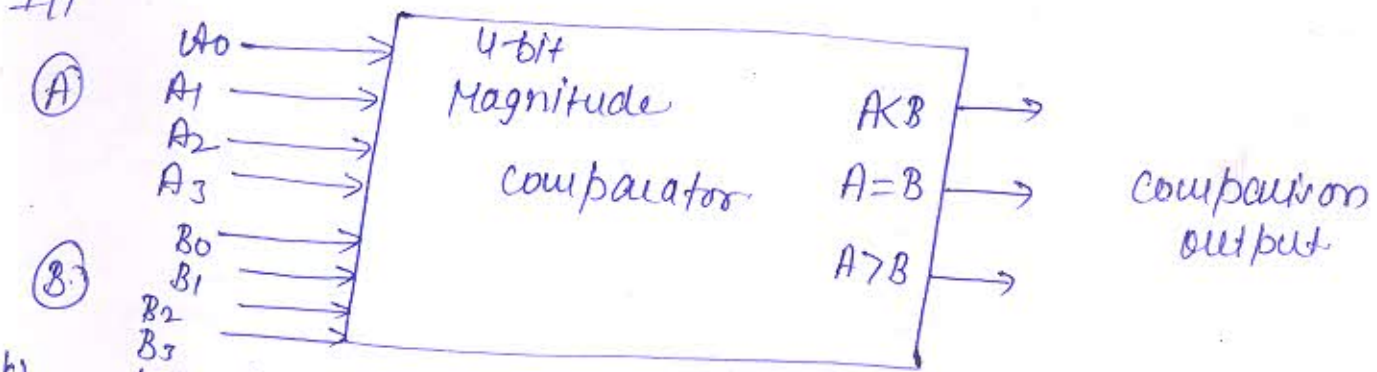
Ans 7) Magnitude comparator → It is a digital circuit design to compare two n bit binary words.



I/P (multiple bit inputs)

O/P (single bit outputs)

4-bit Magnitude comparator →



*) let $A \Rightarrow A_3 A_2 A_1 A_0$ $B \Rightarrow B_3 B_2 B_1 B_0$

*) two numbers are equal if all pairs of significant digits are equal

i.e $A_3 = B_3$, $A_2 = B_2$, $A_1 = B_1$, $A_0 = B_0$

*) When the numbers are binary then digits are either 1 or 0. and the equality of each pair of bits can be expressed logically with EX-NOR function.

So $\boxed{X_i = A_i B_i + \bar{A_i}\,\bar{B_i}}$

$X_1 \Rightarrow A_1 B_1 + \bar{A_1}\,\bar{B_1}$ $X_3 = A_3 B_3 + \bar{A_3}\,\bar{B_3}$

$X_2 = A_2 B_2 + \bar{A_2}\,\bar{B_2}$ $X_0 = A_0 B_0 + \bar{A_0}\,\bar{B_0}$

So, $(A = B) = X_3 X_2 X_1 X_0$

*) Two determine whether A is greater or less than B, inspect the Relative magnitude of pair of significant digits, starting from the MSB.

*) If the two digits of a pair are equal, we compare next lower significant pair of digits. The comparison continues until a pair of unequal digits is reached.

*) If the corresponding digit of A is 1 and that of B is 0, ~~(or)~~ then A > B.

*) If the corresponding digit of A is 0 and that of B is 1 then A < B.

example →

*) A ⇒ 1001 , B ⇒ 0111 , (A > B)

$A_3 \Rightarrow 1$ $B_3 \Rightarrow 0$ $\Rightarrow A_3 > B_3$ i.e $A_3 \& B_3 = 1$

*) A = 1101 , B ⇒ 1011

$A_3 = 1$, $B_3 = 1$ $\Rightarrow A_3 = B_3$

$A_2 = 1$, $B_2 = 0$ $\Rightarrow A_2 > B_2$

So $\boxed{x_3 \Rightarrow 1}$ and $\boxed{A_2 \& B_2 \Rightarrow 1}$ Hence (A > B)

*) A ⇒ 1010 , B ⇒ 1001

$A_3 \Rightarrow 1$, $B_3 = 1$ So $x_3 = 1$

$A_2 = 0$, $B_2 = 0$ so $x_2 = 1$

$A_1 = 1$, $B_1 = 0$ so $A_1 > B_1 \Rightarrow \boxed{A_1 \& B_1 \Rightarrow 1}$

Hence (A > B)

*) $A \Rightarrow 1011$, $B = 1010$

$A_3 = 1$, $B_3 = 1$ $\Rightarrow x_3 = 1$

$A_2 = 0$, $B_2 = 0$ $\Rightarrow x_2 = 1$

$A_1 = 1$, $B_1 = 1$ $\Rightarrow x_1 = 1$

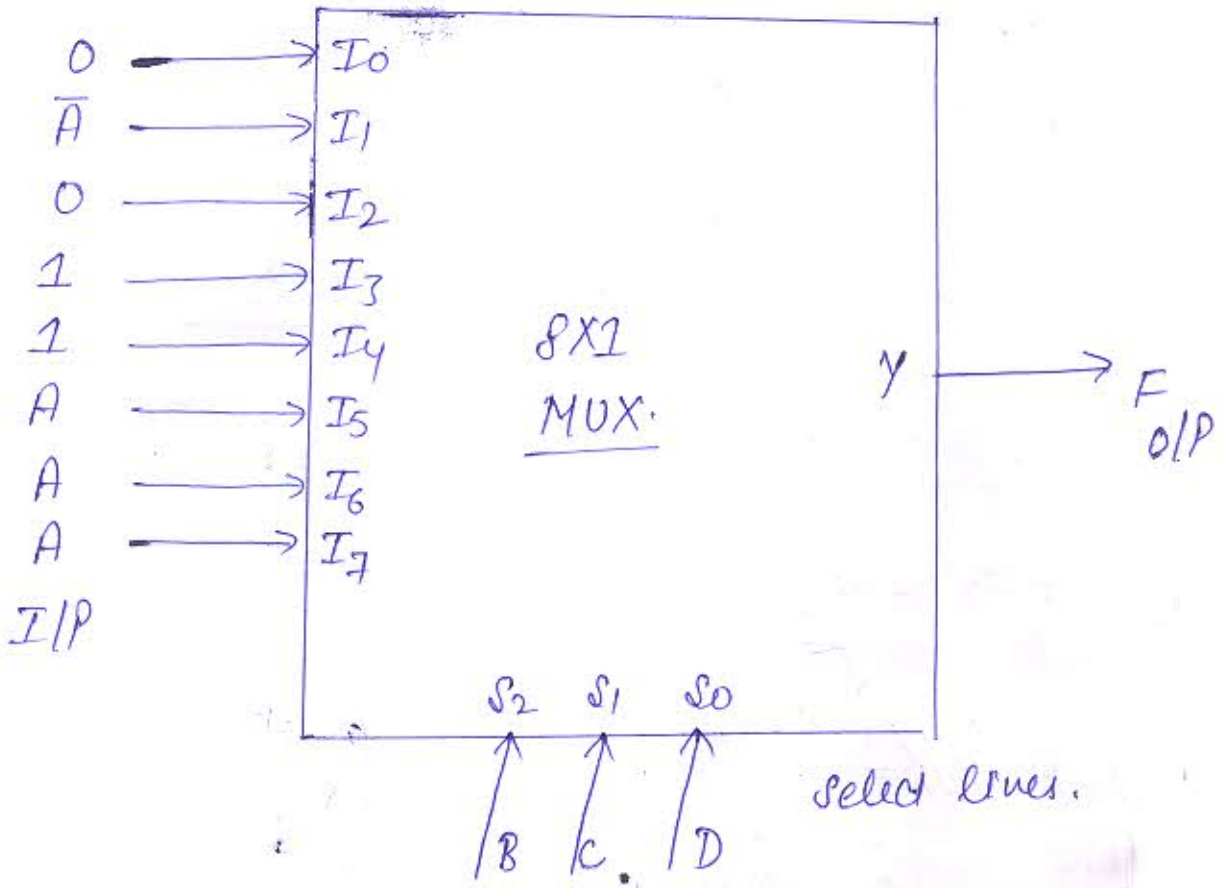
$A_0 = 1$, $B_0 = 0$ $\Rightarrow (A_0 > B_0) \Rightarrow A_0 \& \overline{B_0} = 1$

So $A > B \Rightarrow A_3 \overline{B_3} + x_3 A_2 \overline{B_2} + x_3 x_2 A_1 \overline{B_1} + x_3 x_2 x_1 A_0 \overline{B_0}$

$A < B \Rightarrow \overline{A_3} B_3 + x_3 \overline{A_2} B_2 + x_3 x_2 \overline{A_1} B_1 + x_3 x_2 x_1 \overline{A_0} B_0$

Ans 8) $F(A, B, C, D) = \Sigma(1, 3, 4, 11, 12, 13, 14, 15)$

	A	B	C	D	F
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1



*) Here divide the I/p into two parts from 0 to 7 and 8 to 15

*) I/P to select lines will be B, C, D.

*) case 1.) When compare 0 and 8 then O/P is 0 so $I_0 = 0$

*) case 2) When compare 1 and 9 O/P is opposite of input value of A so $I_1 = \bar{A}$

*) case 3) When compare 2 and 10 O/P is 0 so $I_2 = 0$

*) case 4) When compare 3 and 11 O/P is 1 so $I_3 = 1$

*) case 5) When compare 4 and 12 O/P is 1 so $I_4 = 1$

*) case 6) When compare 5 and 13 O/P is equal to the value of A so $I_5 = A$

*) Case 7) When compare 6 and 14 and 7 and 15 O/P in both case is equal to the value of A so $I_6 = A$ and $I_7 = \bar{A}$

Q.No 9.) Conversion of JK Flip Flop using SR Flip Flop.

Step1.) available FF ⇒ SR.

Required FF ⇒ JK.

Step2) Characteristic table of JK Flip Flop.

Qn	J	K	Qn+1
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Step3) Excitation table of SR Flip Flop.

Qn	Qn+1	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

Step 3) Combine both tables.

Q_n	J	K	Q_{n+1}	S	R
0	0	0	0	0	X
0	0	1	0	0	X
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	1	X	0
1	0	1	0	0	1
1	1	0	1	X	0
1	1	1	0	0	1

Step 4) for S

Q_n \ JK	00	01	11	10
0	0	0	1	1
1	X	0	0	X

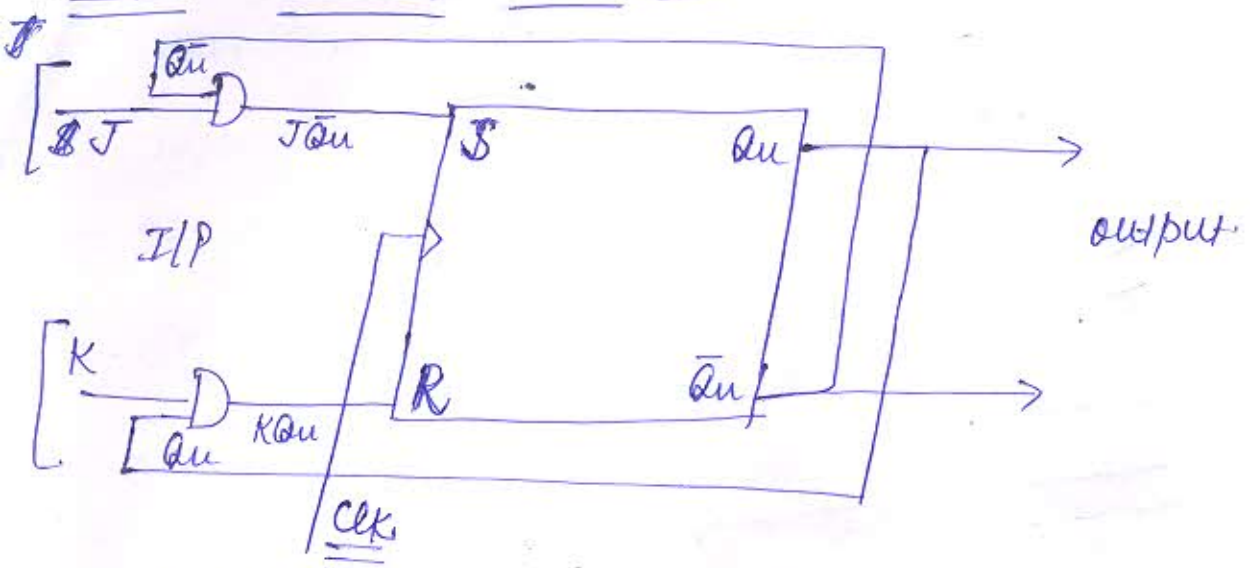
$S \Rightarrow \bar{Q}_n J$

for R.

Q_n \ JK	00	01	11	10
0	X	X	0	0
1	0	1	1	0

$R \Rightarrow Q_n K$

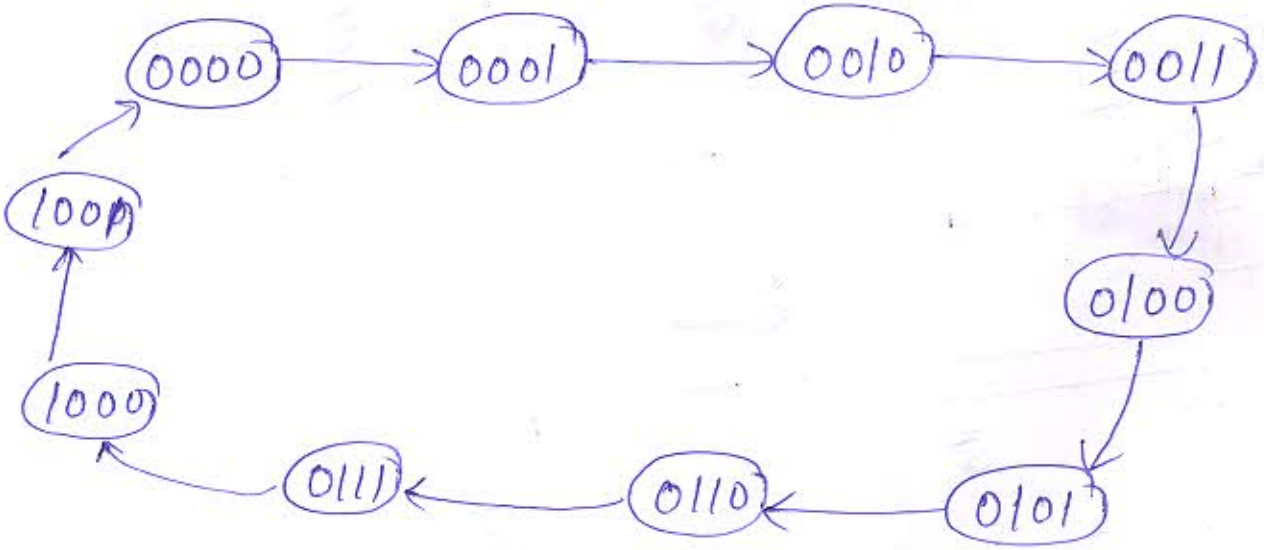
Step 5.) Circuit diagram →



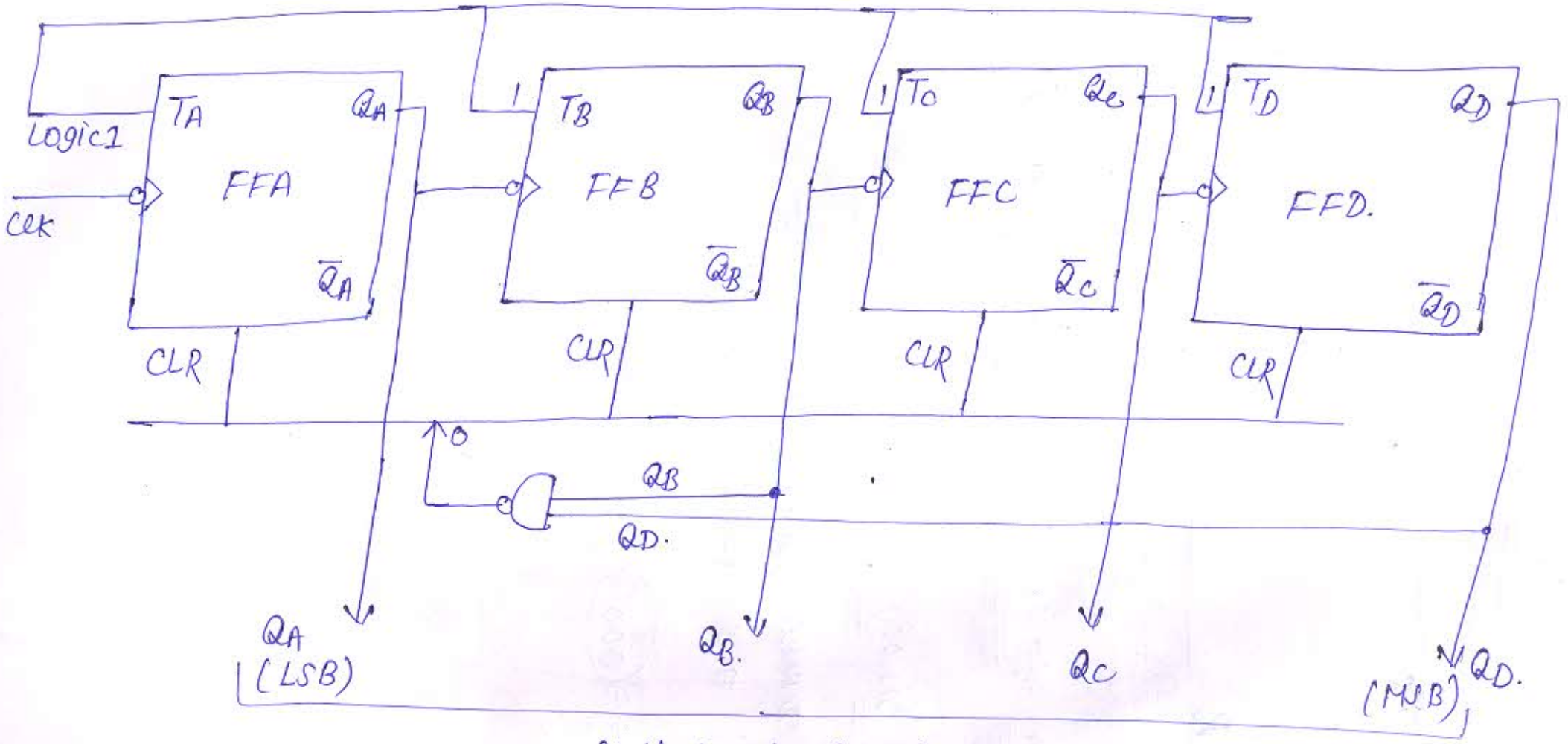
Realization of JK Flip Flop using SR Flip Flop.

Ans 10) MOD-10 binary asynchronous Counter. →

A decimal counter follows a sequence of 10 states (from 0 to 9) and Returns to again 0.



State diagram



Output of counter

*) When CLR=0 (Clear) then output Q=0 (Reset) Regardless of the value of I/P and CLR.

*) This counter is also known as BCD Ripple counter.

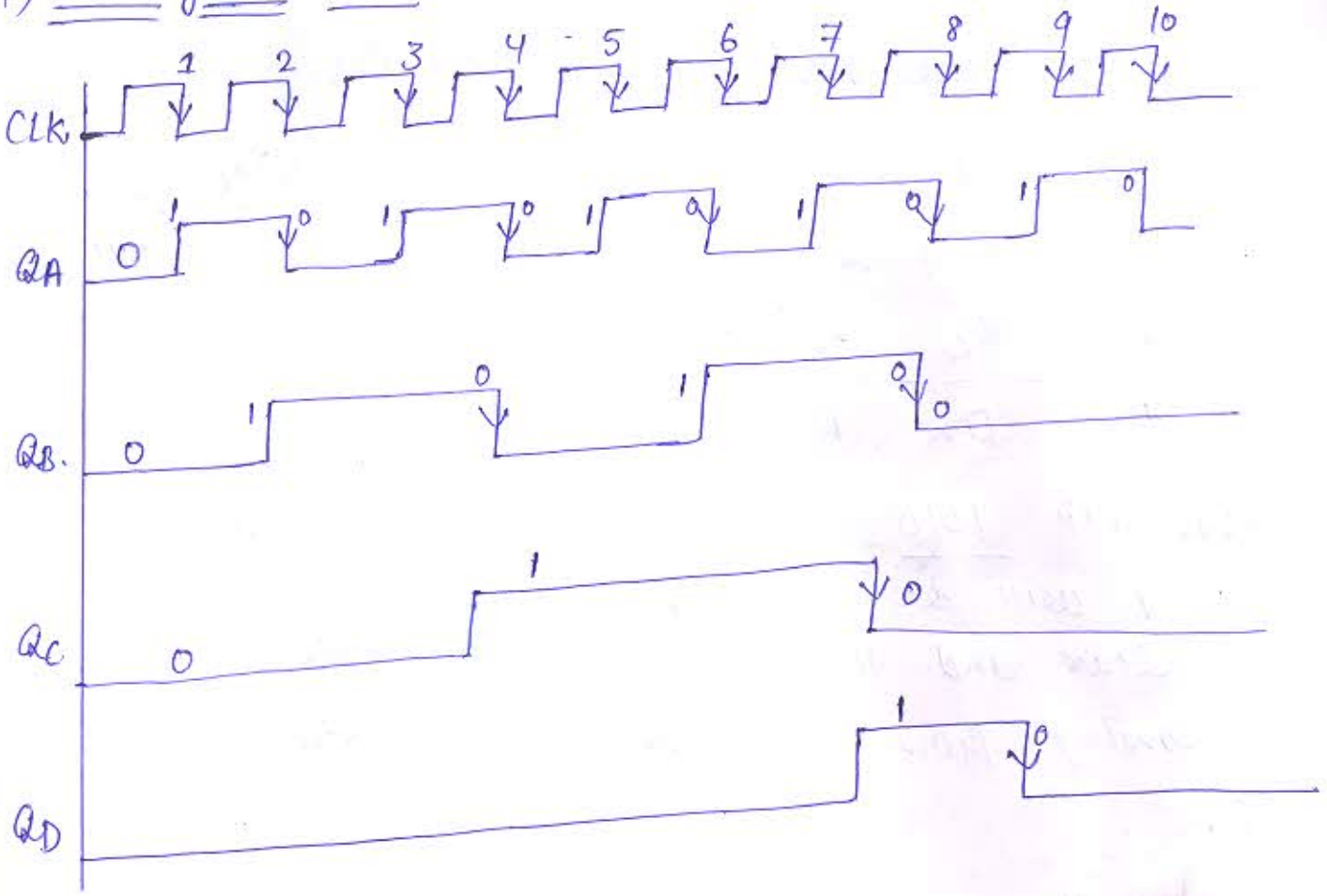
*) It will count from 0 to 9., when 10 occurs then

Q_D	Q_C	Q_B	Q_A
1	0	1	0

then this I/P 1010 is passed to NAND Gate and output will be 0. and this output will pass to clear and then it will Reset all the outputs coming from flip flop and counter will count 0.

CLR	Q_D	Q_C	Q_B	Q_A	
initially (0)	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	↓ (UP counter) counts from 0 to 9.
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	0	0	0	0	

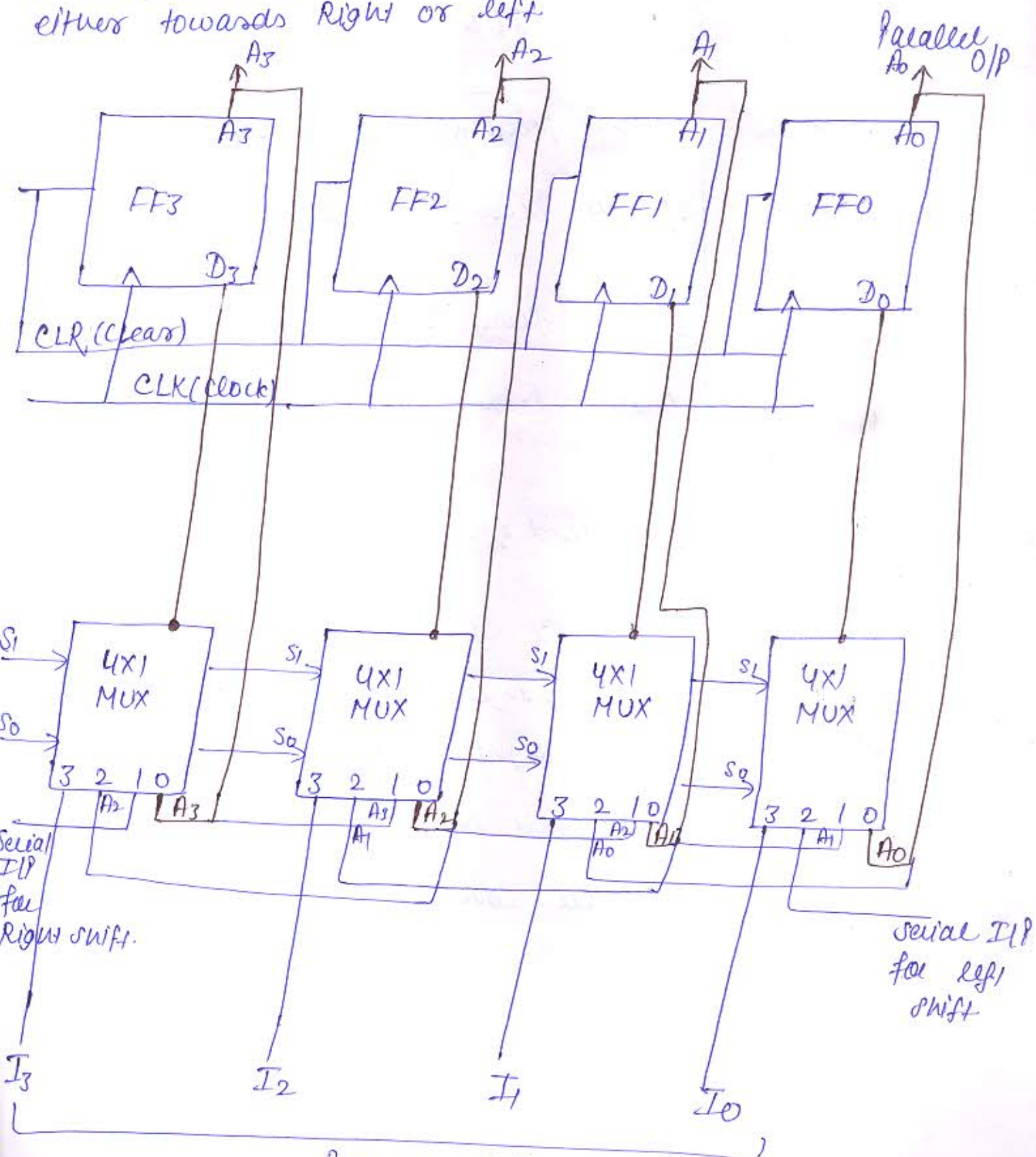
*) Timing diagram →



Section C

(Ans 11) **4-bit Universal Shift Register →**

It is a Register which can be configured to load / Retrieve data in any mode (either either serial or parallel) by shifting it either towards Right or left.



*) All types SISO, SIPO, PISO, PIPO are covered in this type of Register.

*) In this type of Register Select lines will decide which value pass to flip flop.

S_1	S_0	I/P to Flip Flop.
0	0	output from line 0 is selected for D.
0	1	output from line 1 is selected for D
1	0	output from line 2 is selected for D.
1	1	output from line 3 is selected for D.

*) Selector variable are used for Mode control (i.e L/R) (Left/ Right Shift)

S_1	S_0	Register operation.
0	0	No change in State
0	1	Shift Right Mode
1	0	Shift left Mode
1	1	Parallel Loading.

Ans (2.)

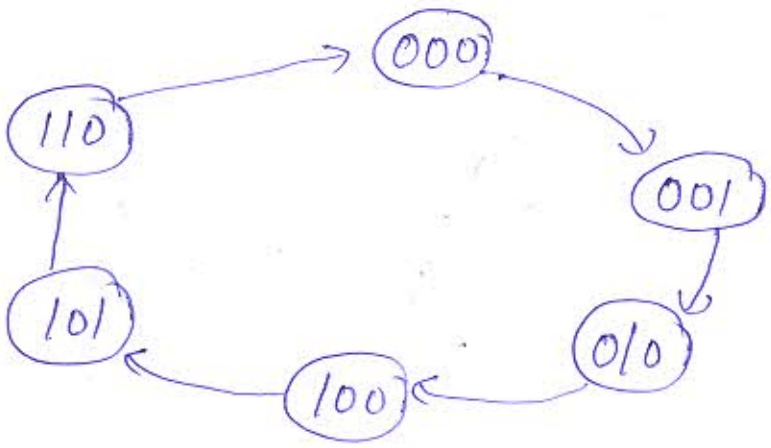
Step 1.) Type of Flip Flop ⇒ JK

Number of Flip Flop ⇒ 3.

Step 2.) Excitation table of JK Flip Flop.

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Step 3.) State diagram →

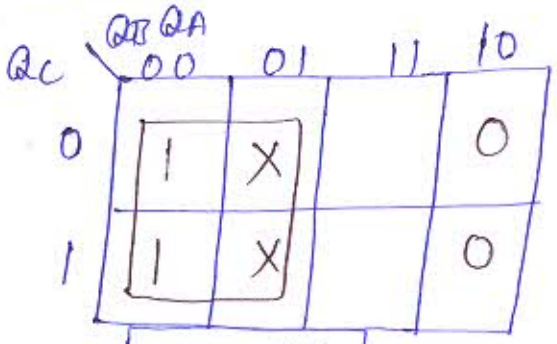


*) Circuit Excitation table →

Present state			Next state			I/P from FLIP FLOP.					
						J_C	K_C	J_B	K_B	J_A	K_A
Q_C	Q_B	Q_A	Q_C^*	Q_B^*	Q_A^*						
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	1	0	0	1	X	X	1	0	X
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	0	0	0	X	1	X	1	0	X

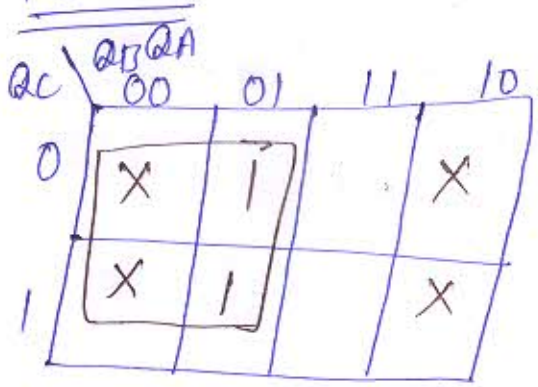
Step 4.) I/P of FLIP FLOP using K-map.

For J_A →



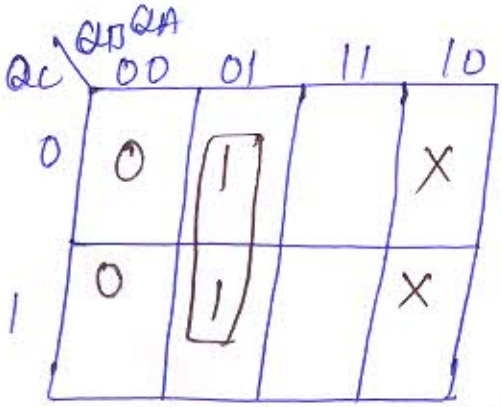
$J_A = \bar{Q}_B$

For K_A



$K_A = \bar{Q}_B$

For J_B →



$J_B = \bar{Q}_B Q_A$

For $K_B \rightarrow$

Q_C \ $Q_B Q_A$	00	01	11	10
0	X	X		1
1	X	X		1

$K_B \Rightarrow \bar{Q}_A$

For $J_C \rightarrow$

Q_C \ $Q_B Q_A$	00	01	11	10
0	0	0		1
1	X	X		X

$J_C = Q_B \bar{Q}_A$

For $K_C \rightarrow$

Q_C \ $Q_B Q_A$	00	01	11	10
0	X	X		X
1	0	0		1

$K_C = Q_B \bar{Q}_A$

Step 5.) Circuit diagram →