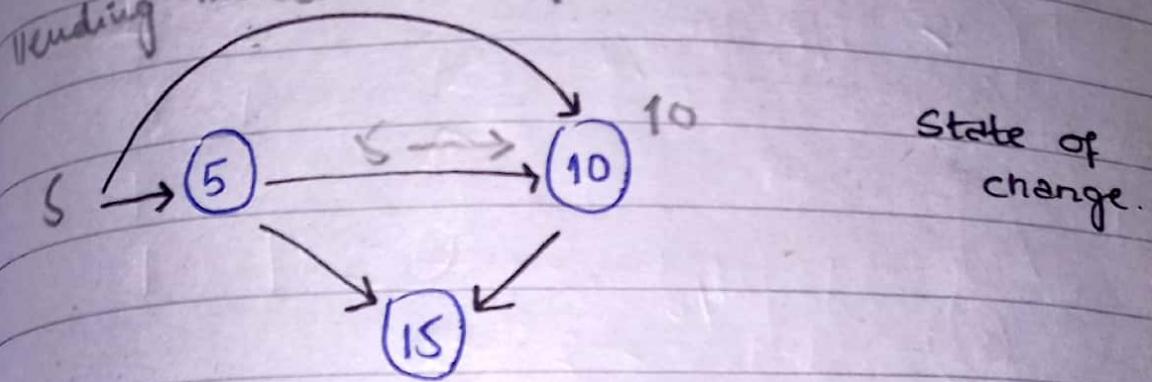


Sequential Circuit Design.

Winding Machine Example.



Random string (pattern matching)

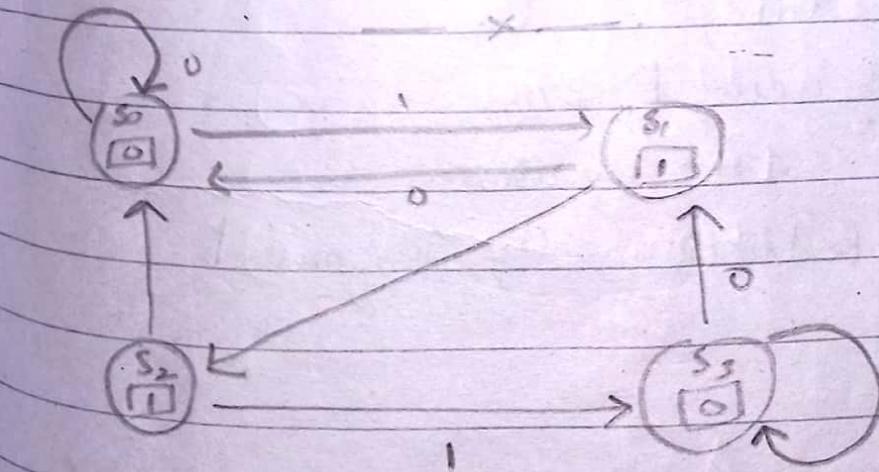
aa bca bcd ddb cab

ab.c.a

ab.dob

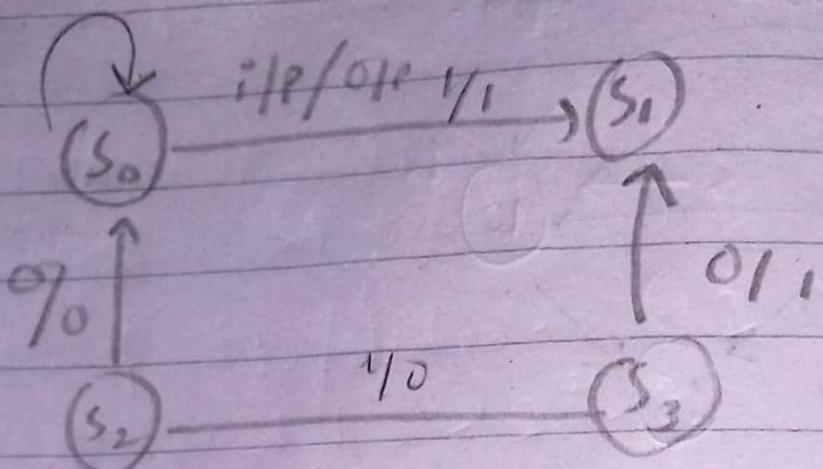
We have to find abcd.

- 1) abc (reset)
- 2) abcd (pattern found)



State is defined in Moore Machine.

Mealy Machine



- if S_0 is zero then it will not move anywhere

- if you are on S_0 & you apply r/p 1 then pt will move to S_1 .

_____ x _____

Q: Design a sequential circuit for the given state diagram. Use SR flip flop.

- 1) with all states.
- 2) with all unused state
- 3) " " state equations
- 4) State Reduction by Assignment.

State diagram is a geographical representation of state table, it conveys some information as state table.

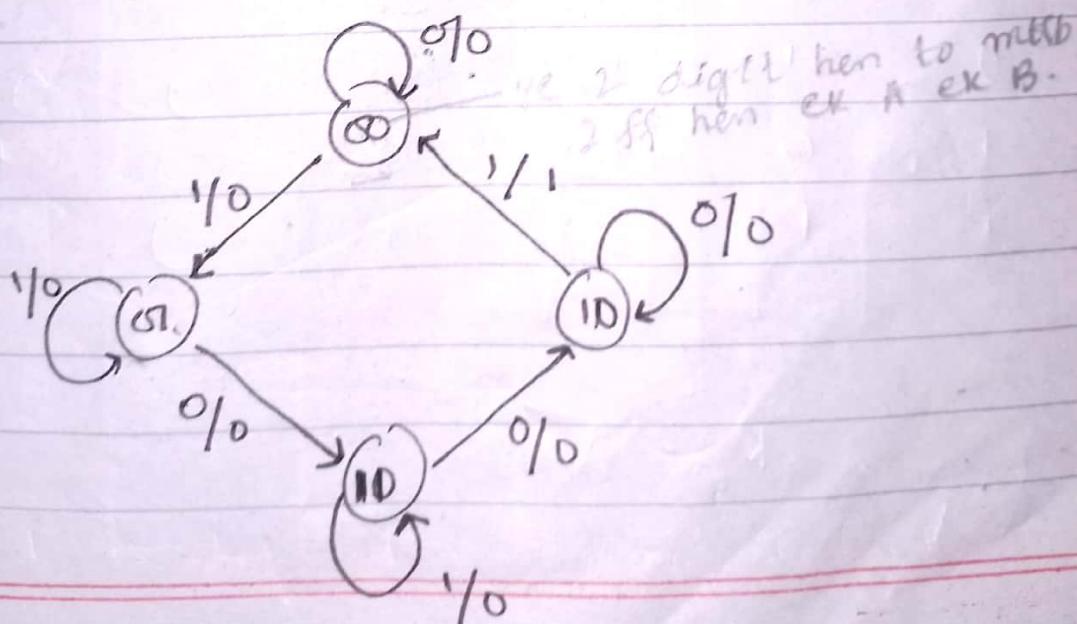
The time sequence for input, output & flip flop states may be enumerated in a table which is known as state table. It consists of three sections present state, next state & output.

A state equation is an algebraic expression that specifies the conditions for a flip-flop

— X —

i) with all states:

State diagram: (given)



Make sure all states are defined and every state must have two arrows.

\Rightarrow Table bnezi ek present state s_i , ek
next state s_o or ek output y_i .

$\Rightarrow n=0$ q $n=1$ input lga.

sol:

No. of bits/states $\times m = 2$

" " states $= 2^2 = 4$

" " flip flop $\times n = m = 2$.

(2 bits bits utne flip flop)

State Table:

Present State		Next State		Output.	
A	B	$x=0$	$x=1$	y	
0	0	00	01	0	0
0	1	11	01	0	0
1	0	10	00	1	
1	1	10	11	0	0

Excitation Table:

Present State $Q(t)$	A	B	X	Next State $Q(t+1)$		Flip flop input				Output Y.
				A	B	S _A	R _A	S _B	R _B	
0 0 0	0	0	0 0	0 0	0	0	X	0	X	0
0 0 1	0	0	1	0 1	0	0	X	1	0	0
0 1 0	0	1	0	1 1	1	1	0	X	0	0
0 1 1	0	1	1	0 1	0	0	X	X	0	0
1 0 0	1	0	0	1 0	1	X	0	0	X	0
1 0 1	1	0	1	0 0	0	0	1	1	0	X
1 1 0	1	1	0	1 0	1	X	0	0	1	0
1 1 1	1	1	1	1 1	1	X	0	X	0	0

Reference Table
(Excitation Table for S-R flip flop)

$Q(t)$	$\alpha(t+1)$	S	R
0 0	0 X		
0 1	1 0		
1 0	0 1		
1 1	X 0		

AB nm equations find Kmap
Kmap we through

S_A	BX	∞	01	11	10
A	
0	X		X	(1)	(X)
1					

$$S_A = B\bar{X}$$

R_A	$BX\infty$	01	11	10	.
A	
0	X	(X)	X		
1		(1)			

$$R_A = \bar{B}X$$

S_B	BX	∞	01	11	10
A	
0		(1)	(X)	X	
1				X	

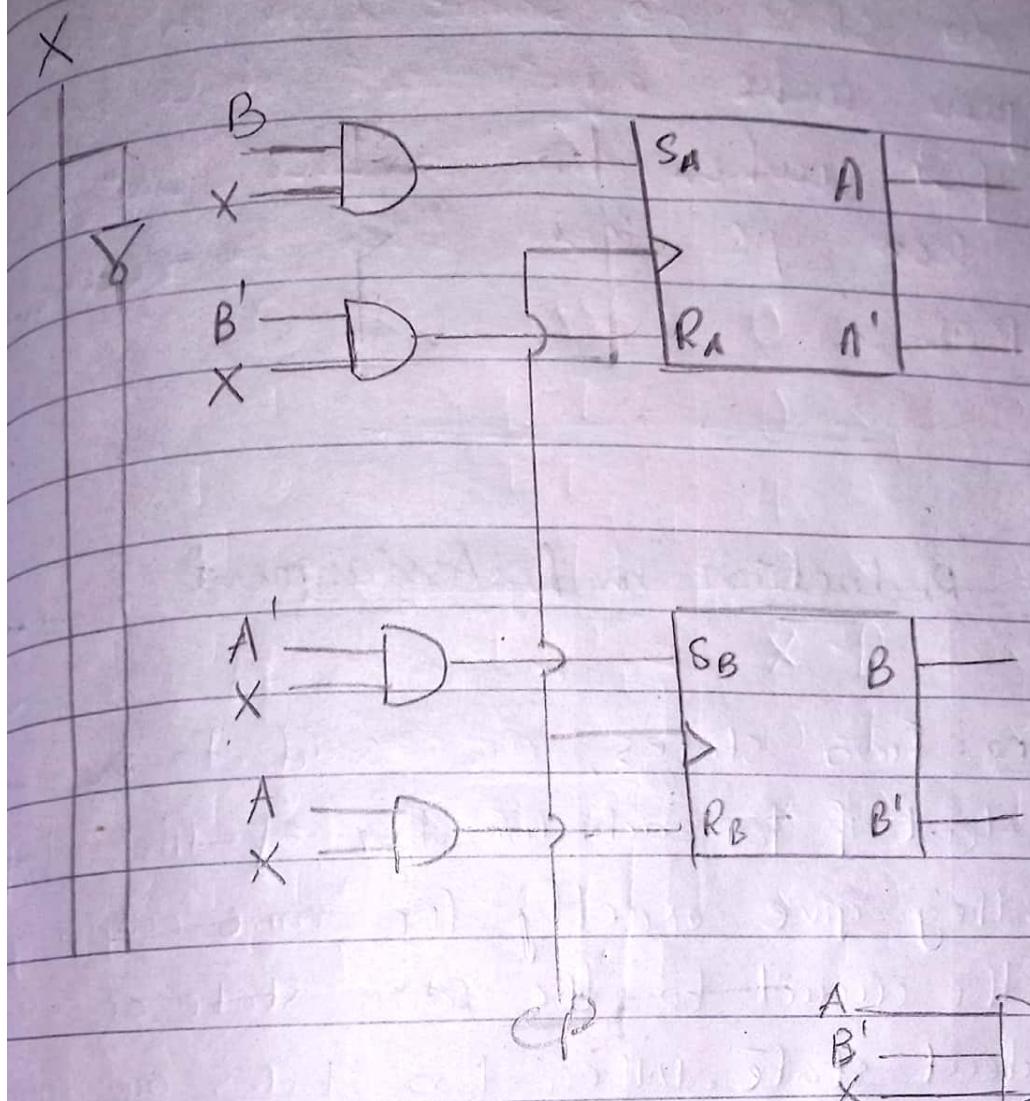
$$S_B = \bar{A}X$$

R_B	BX	∞	01	11	10
A	
0	X				(1)
1	(X)	X		C	

$$R_B = A\bar{X}$$

And

$$Y = AB'X$$



Output (Y) के लिए बिन्दु S_A पर अ. वाला
कम कम होल्ड होना चाहिए ताकि सिर्फ 1
एवं 0 के बीच अवधि ABX देखने
की अवधि लाई जाए। यदि अ. वाला
बहुत छोटी अवधि लाई जाए तो यह
प्रॉफेसर की विवादों में आ जाएगा।

Kot do state jo same ham khele
 hui hain unhe equivalent state kaha
 hain. Or when hm reduce bhi ke
 sick hain ye ham kaha he reduction
 ka.

— → —

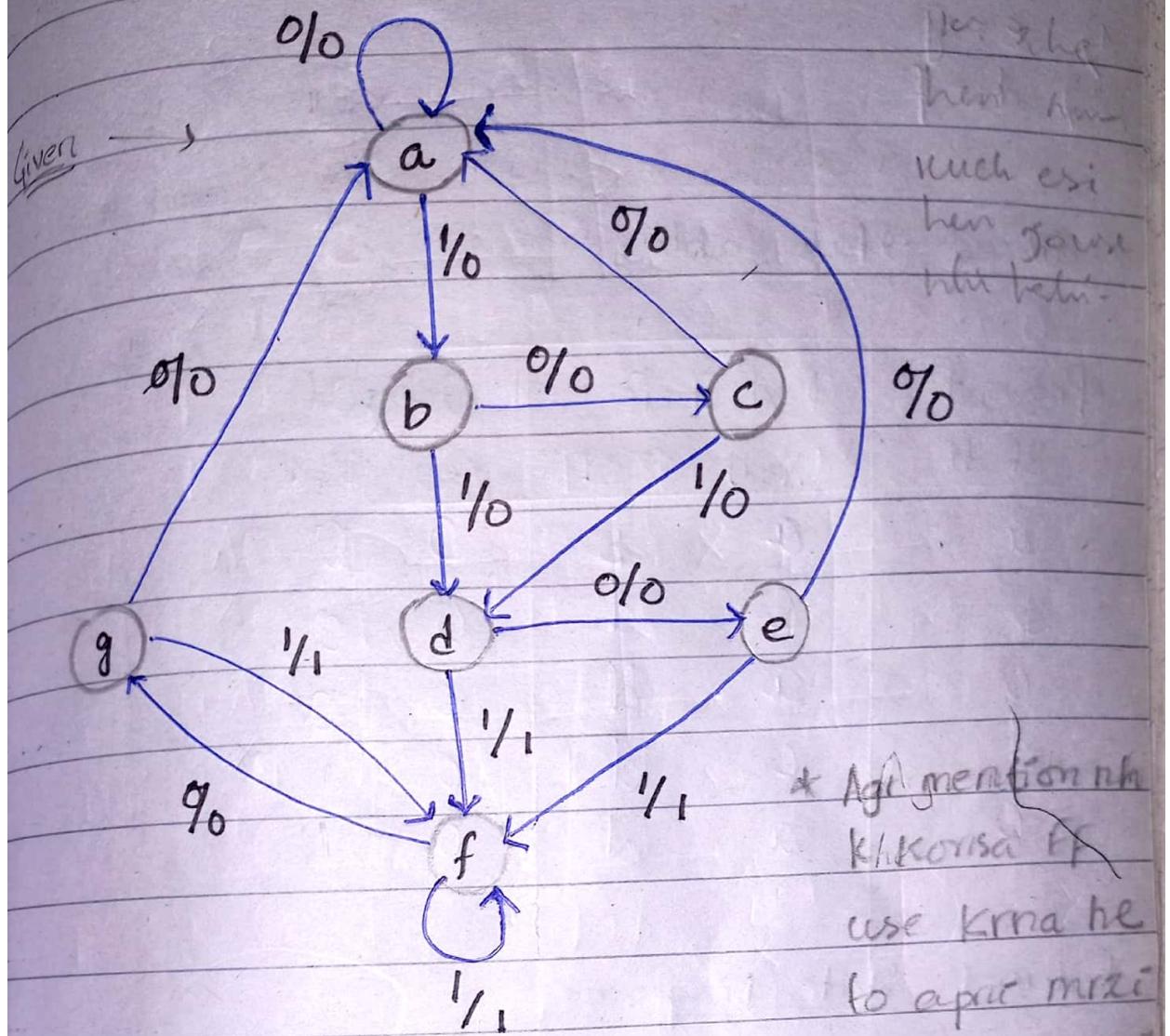
State Reduction and Assignment

Algorithm: Two states are said to be equivalent if for each member of the set of inputs, they give exactly the same output if send the circuit to the same state or an equivalent state. When two states are equivalent, one of them can be removed without altering the input-output relationships.

Q: a) Reduce the number of states for the given state diagram & tabulate the reduced table. Also draw the reduced state diagram.

b) Design the reduced sequential circuit with the given assignment. Use any FF.

State	A #1	A #2	For me koi ek given hga ye practice n lie 2 hain,
a	001		
b	010	000	
c	011	010	
d	100	011	
e	101	101	
		111	



State Table :

Present State	Next State		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1

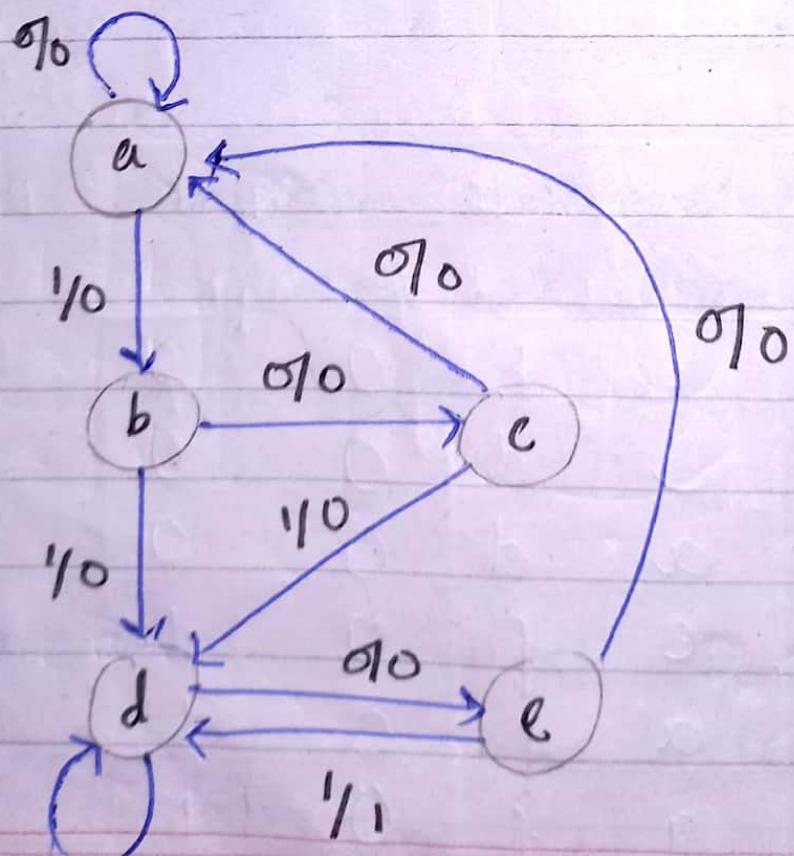
Observations:

- 1) $e = g$
- 2) $d = f$

Reduced State Table:

Present State	Next State		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Reduced State Diagram:



State Table:

Unused state 100

Output care we
~~X=0~~ X=1

Present State A B C	Next State X=0 X=1	Unused state 100	
		ABC	ABC
a 0 0 0	000 010	0	0
b 0 1 0	011 101	0	0
c 0 1 1	000 101	0	0
d 1 0 1	111 101	0	1
e 1 1 1	000 101	0	1

$$\text{nb. of bits } (A+B+C)^n = 3^n$$

$$\text{no. states } 2^3 \cdot 3^3 = 8 \cdot 27$$

↓
in binary
form

& return info which go A#2

me given he or go 0-7 like

me below in jaegi we don't care bn jaegi

Unused states = 001, 100, 110

Excitation Table

Present state A B & X	Next state A B C	Flip flop input D _A D _B D _C	Output Y
0 0 0 0	0 0 0	0 0 0	0
0 0 0 1	0 1 0	0 1 0	0
0 1 0 0	0 1 1	0 1 1	0
0 1 0 1	1 0 1	1 0 1	0
0 1 1 0	0 0 0	0 0 0	0
0 1 1 1	1 0 1	1 0 1	0
1 0 1 0	1 1 1	1 1 1	0
1 0 1 1	1 0 1	1 0 1	1
1 1 1 0	0 1 0	0 0 0	0
1 1 1 1	1 0 1	1 0 1	1

Ref table of D-FF block diagram.

AB	CX	D _A	U	I0
00	X	X	X	X
01	1	1		
11	X	X	1	
10	X	X	1	

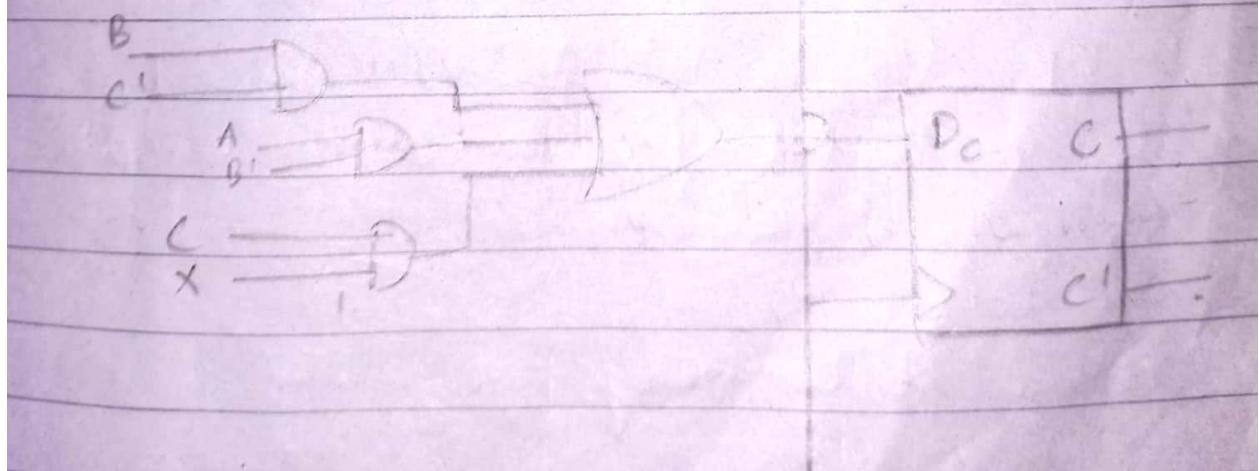
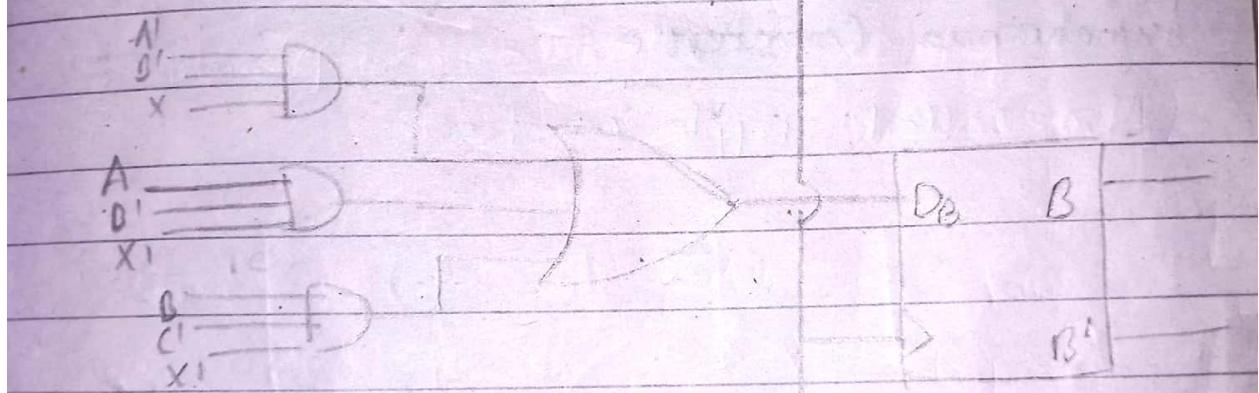
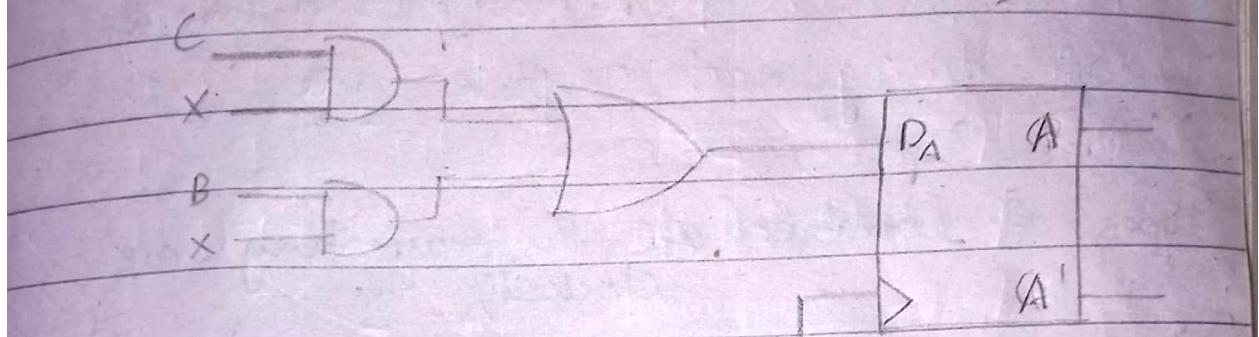
AB	CX	D _B	U	I0
00		1	X	X
01		1		
11	X	X		
10	X	X		1

$$D_A = CX + BX$$

$$D_B = B + \bar{A}Bx + A\bar{B}\bar{x} \\ + B\bar{C}x$$

D_C	CX_{00}	01	11	10
AB			R	X
CD				
01	1	1		
11	X	X	1	
10	X	X	1	1

$$D_C = B\bar{C} + A\bar{B} + CX.$$

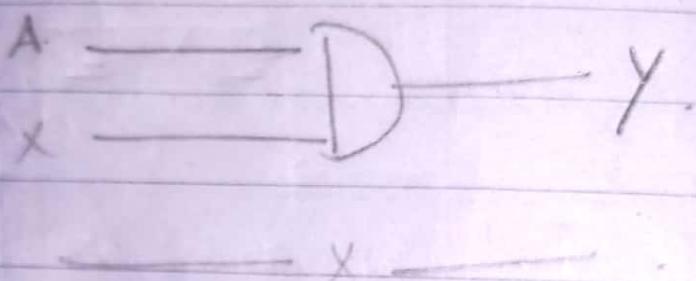


CP

Baki nege →

y	x_00	a	11	10
AB			\times	\times
00				
01				
11	x	x	1	
10	x	x	1	

$$Y = AX$$



CLR ke 2 ways (ek wa. fikrd complete
Ko. & reset kro.) automatic

2nd
ye K. hm use voice apne maza se
 (\rightarrow) forcefully.

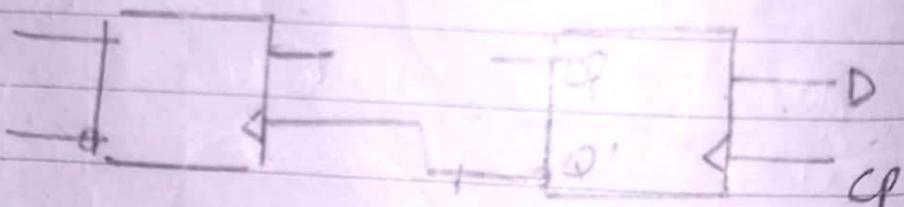
Counters.

Flip flop can be connected together to perform counting operations. Such a group of flip flop is called a counter. The no. of FF used & the way in which they are connected determines the no. of states (also called the modulus/mod) and also specific sequences of states that a counter goes through during each complete cycle.

Types of Counters (all the way they are clocked).

Asynchronous Counters:

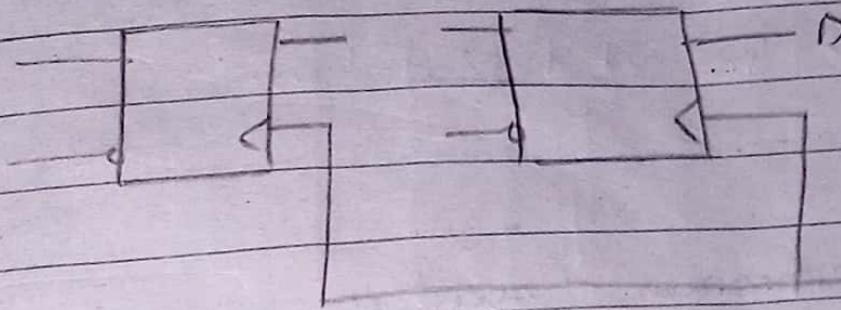
- Also called ripple counters



I CP hogi bs 1st me or uske output ki complementary form baki wali ki CP hogi and so on ...

- Is me state diag optional hoga he.
- JK ya T flip flop use karne ke liye ke toggle karna hoga.

Synchronous Counters:



EK hi CP sb se connect hga simultaneously

We always design a counter for a complete cycle.

0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7, 0, 1, ...
↓ cycle. → Up Counter

1, 6, 5, 4, 3, 2, 1, 0, 7, 6, 5, 4, ...
↓ cycle. → Down Counter

0, 1, 2, 3, 2, 1, 0, 1, 2, 3, 2, 1, 0, ...
↓ cycle. → Twos Complement Counter

- Agr miss ne mention ruku kia to by def.
Up Counter hga

- By default binary counters

- 2 bit counter = mod 4 counter

- No. of states = modulus/mod

- 4 no. of states = mod 16

Counters can be :

- Full modulus (All states)

- Truncated Modulus

→ All States are not
in use

Asynchronous Counters

Q: Design a 2-bit asynchronous counter.

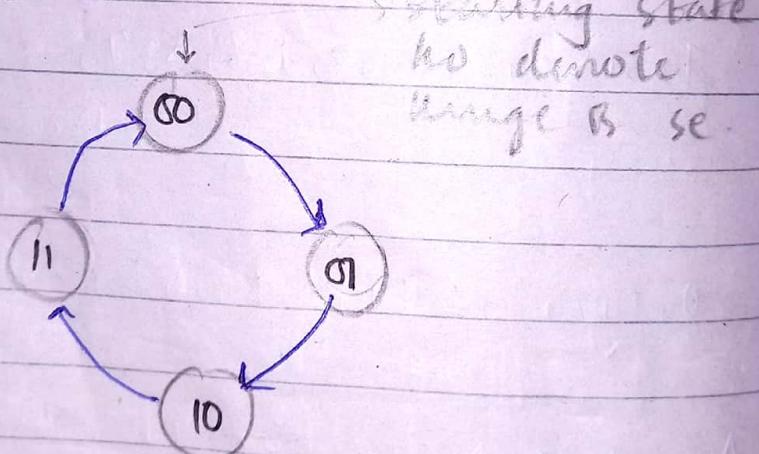
No. of bits = 2

No. of states = Total modulus = $2^n = 4$

No. of FFS required = $m+n = 2$

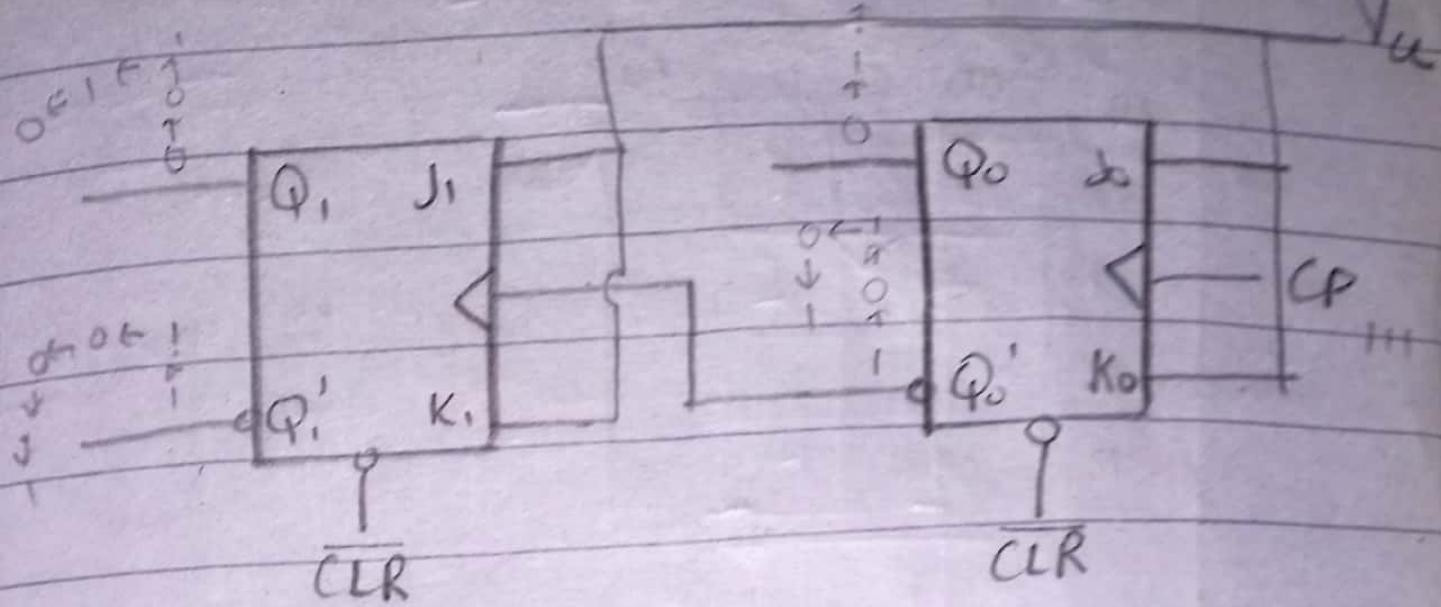
Range = 0 to $2^n - 1 = 0 \text{ to } 3$.

State Diagram:



State Sequence Table:

CP	Q_1	Q_0	
Initial	0	0	When $(CR=0)$
1	0	1	4th pulse starts
2	1	0	recycling
3	1	1	



→ hm isko 0 pass
 kenge ye 1 hoga
 to reset hoga.
 (mtlb 1 de on hoga
 GT clear hoga)

- Vcc mtlb 1. Hun ne sare JK ko 1 ko dia
- 0 CP pe FF work nahi heta, to mtlb same values return hoga

Q1 Design a decade asynchronous counter
 $\text{mod } 10$
 $(0-9)$

$$\text{No. of states} = 2^n = 10$$

$$\text{" " bit} \approx 4$$

$$\text{" " FFs required} \Rightarrow m = n \approx 4$$

$$\text{Range} = 0 \text{ to } 9.$$

binaculated
counter

State Sequence Table:

CP	\dot{Q}_3	\dot{Q}_2	\dot{Q}_1	\dot{Q}_0
Initially	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
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	1	0	1	0

→ Is me microseconds

to the aega zamez

let's display who has

how we want big dage

4 bits ke 60

15 in game

ke koshish

karya ye.

Part from TSE

rokege.

change

jhana pe time counter nahi jana

dena ke (10) wo state or jhn

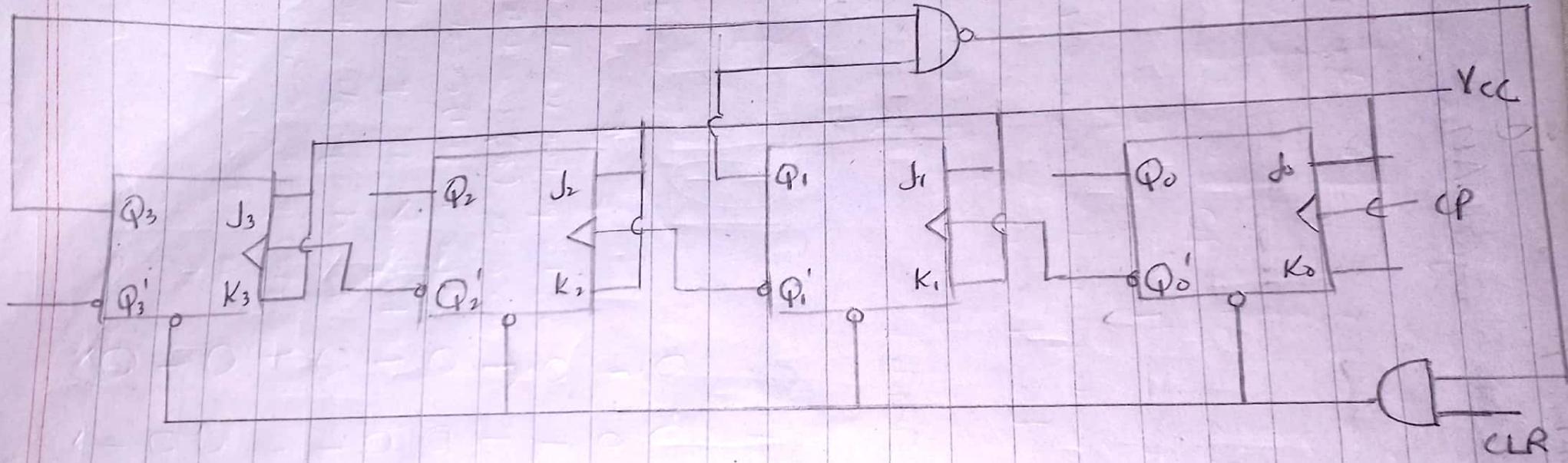
jhana ke (10) wo state comp

hogni. ✓ change and X

no change. jhne no. of

change hoga nand gate

one value inputs aage



Q_3 and Q_0 pe change
ke to use nand gate
& connect kya.

CLR hmare hath me he
hme use 1 dage or !
Nand J krdega to 1010
reset hjaega.

Synchronous Counters:

→ State diagram
 Q: Design a 2-bit synchronous → Koc kbu FF we
 counter. Use any FF. Kravite.

$$\text{No. of bits} = n = 2$$

$$\text{No. of states} = \text{Total modulus} = 2^n = 4$$

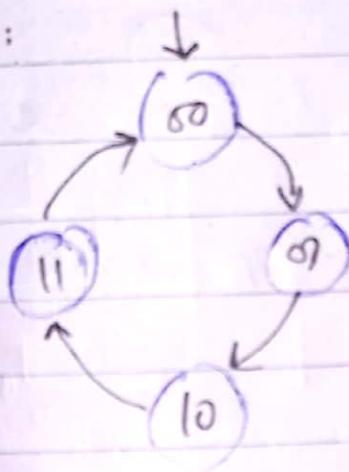
$$\text{No. of FFs required} = m = n = 2$$

$$\text{Range} = 0 \text{ to } 2^n - 1 = 0 \text{ to } 3$$

By default

Up Counter.

State Diagram:



- Asynchronous mode
 so stops the
 taking sync
 proper state

State Table:

Present State	Next State	to train solve
AB	AB	begin
00	01	1. State Diagram
01	10	2. State Table
10	11	3. Excitation
11	00	4. MUX
		5. Eq F
		6. Logic Circ

Excitation, Excitation Table

P.S	N.S	FF Inputs	
AB	AB	T _A	T _B
00	01	0	1
01	10	1	1
10	11	0	1
11	00	1	1

Minimization:

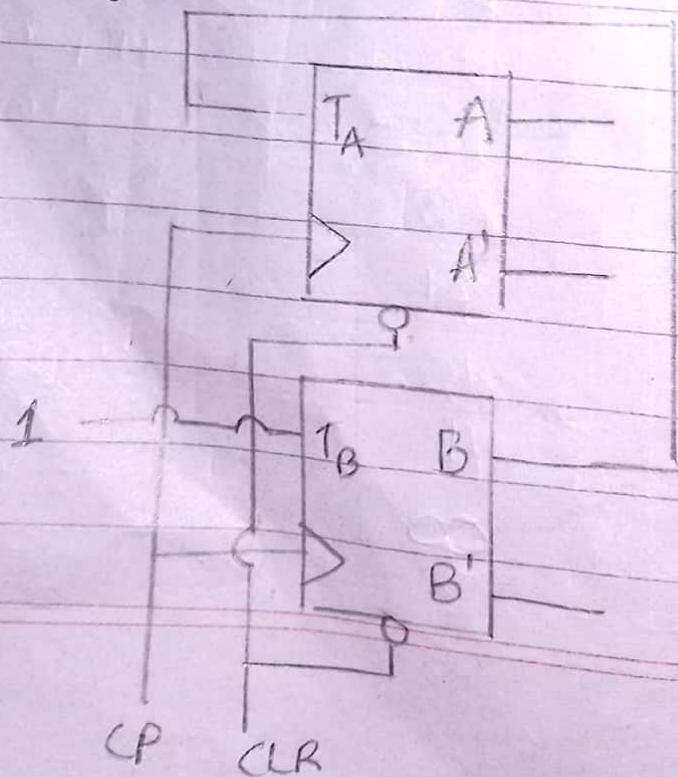
T _A	B	O	I
A			
0		1	
1		1	

T _B	B	O	I
A			
0		1	1
1		1	1

$$T_A = B$$

$$T_B = 1$$

Logic Diagram:



Q: Design a 2-bit synchronous down counter

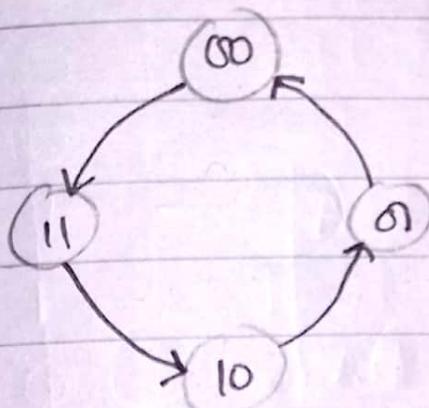
No. of bits = $n = 2$

No. of states = $2^n = 4$

No. of FFs required = $m = n = 2$

Range = 0 to $2^n - 1 = 0$ to 3.

State Diagram:



State Table:

Excitation Table:

PS	N.S	P.S	N.S	FF Inputs	
AB	AB	A·B	A·B	TA	T _B
00	11	00	11	1	1
01	00	01	00	0	1
10	01	10	01	1	1
11	10	11	10	0	1

Minimization:

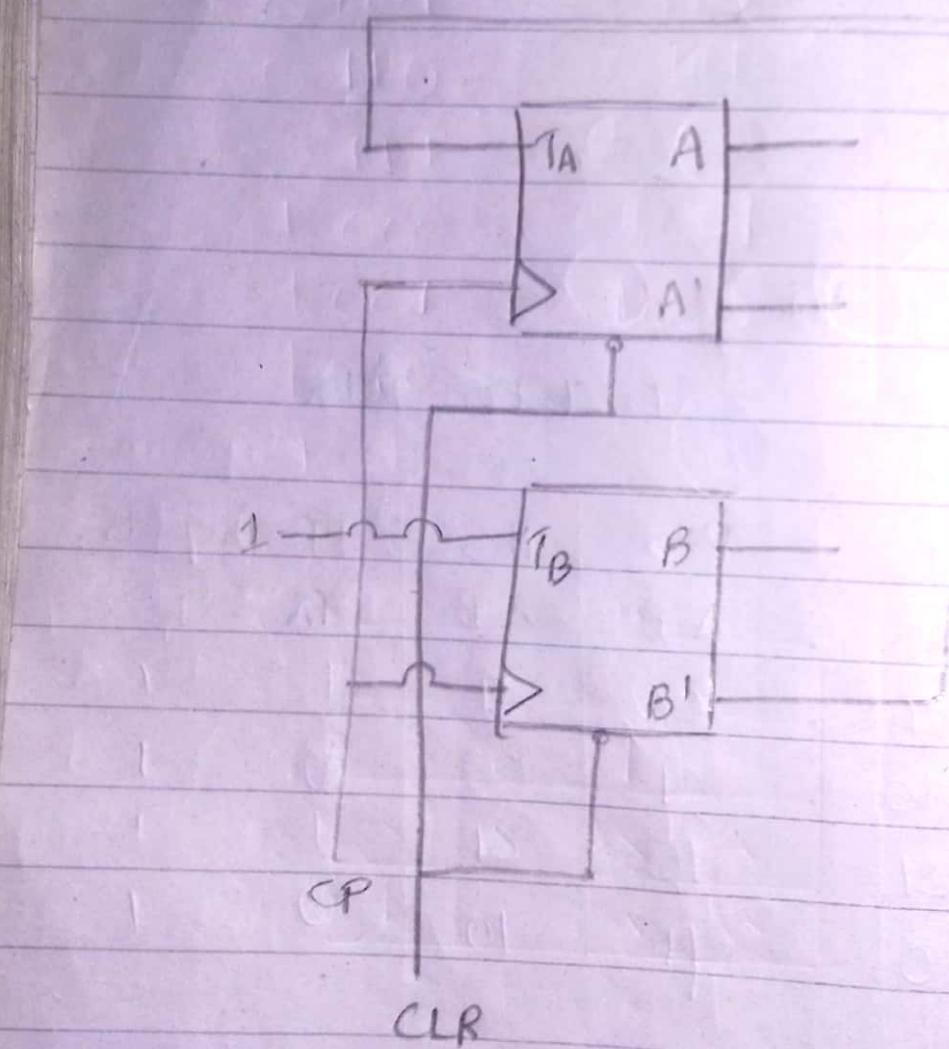
T_A	B	O	I
A	0	1	
O	1	1	
I			

$$T_A = B'$$

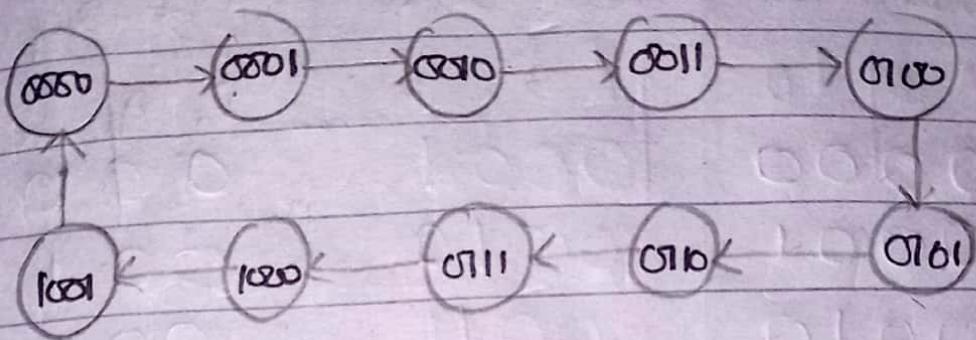
T_B	B	O	I
A	0	1	
O	1	1	
I	1	1	

$$T_B = 1$$

Logic Diagram:



Q) Design a decade synchronous counter.



$$\text{No. of States} = 2^n = 10$$

$$\text{No. of bits} = n = 4$$

$$\text{No. of FFs required} = m = n = 4$$

$$\text{Range} = 0 \text{ to } 9$$

State Table:

Present State A B C D	Next State A B C D
0 0 0 0	0 0 0 1
0 0 0 1	0 0 1 0
0 0 1 0	0 0 1 1
0 0 1 1	0 1 0 0
0 1 0 0	0 1 0 1
0 1 0 1	0 1 1 0
0 1 1 0	0 1 1 1
0 1 1 1	1 0 0 0
1 0 0 0	1 0 0 1
1 0 0 1	0 0 0 0

Nand ye sb
kam aye
a re the

Excitation Table:

Present State A B C D	Next State A B C D	FF Inputs D_A D_B D_C D_D
0000	0001	0 0 0 1
0001	0010	0010
0010	0011	0011
0011	0100	0100
0100	0101	0101
0101	0110	0110
0110	0111	0111
0111	1000	1000
1000	1001	1001
1001	0000	0000
1010		

Minimization:

D_A	CD	00	01	11	10
AB	00				
00					
01				1	
11	x	x		x	x
10	1		x	x	

$$D_A = AD' + BCD.$$

D_B	CD	00	01	11	10
AB					
CB					
01		1	1	1	1
11	X	X	X	X	
10			X	X	

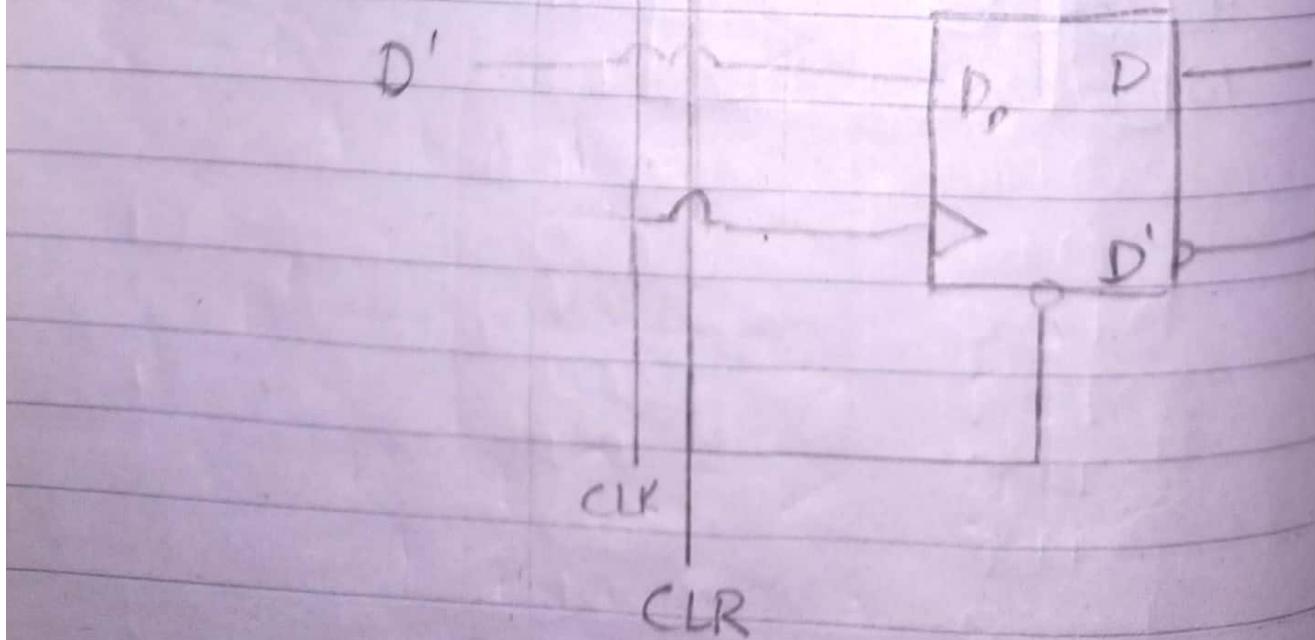
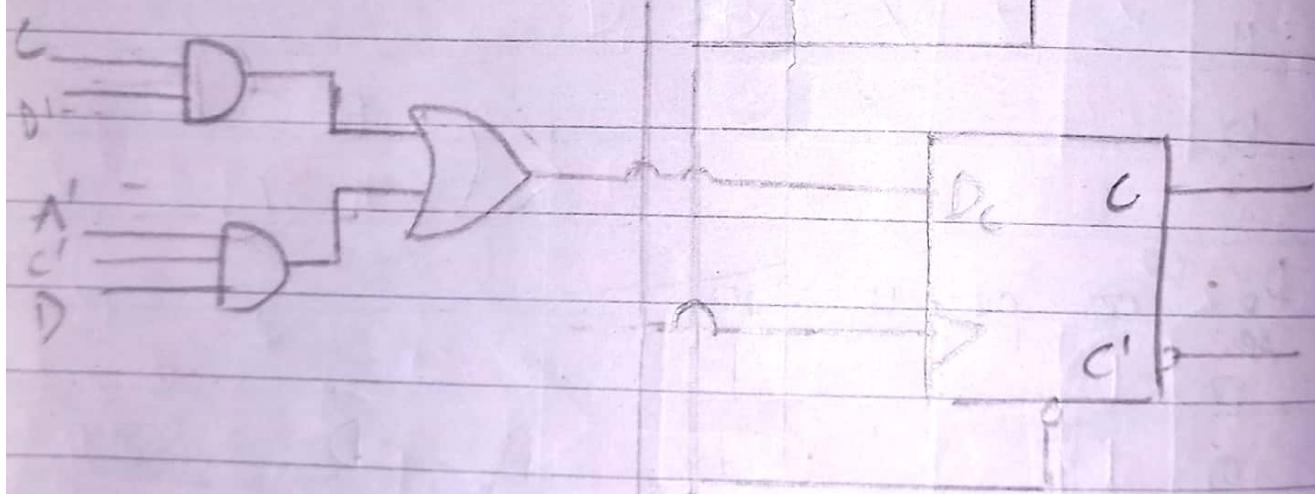
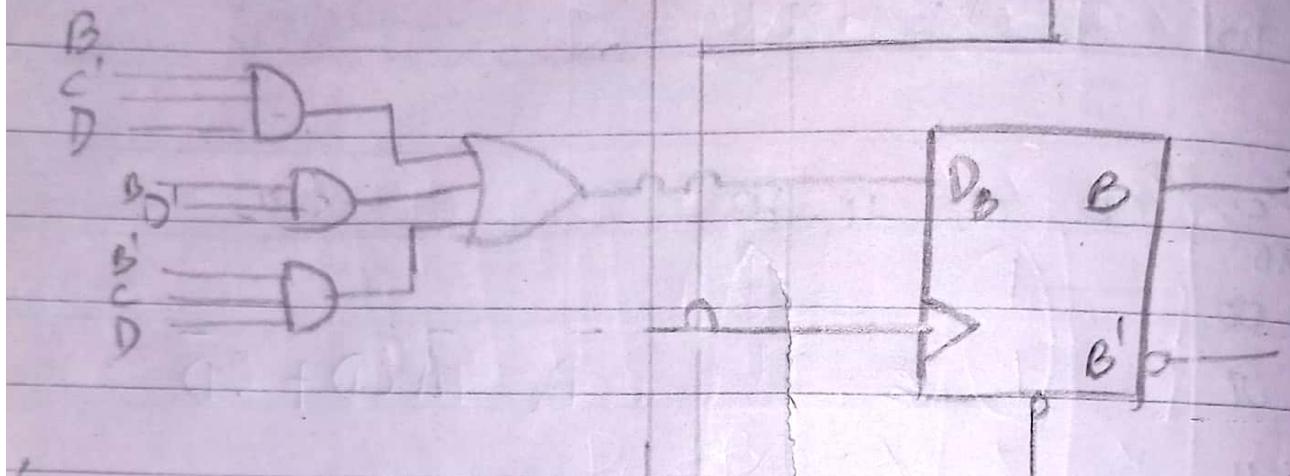
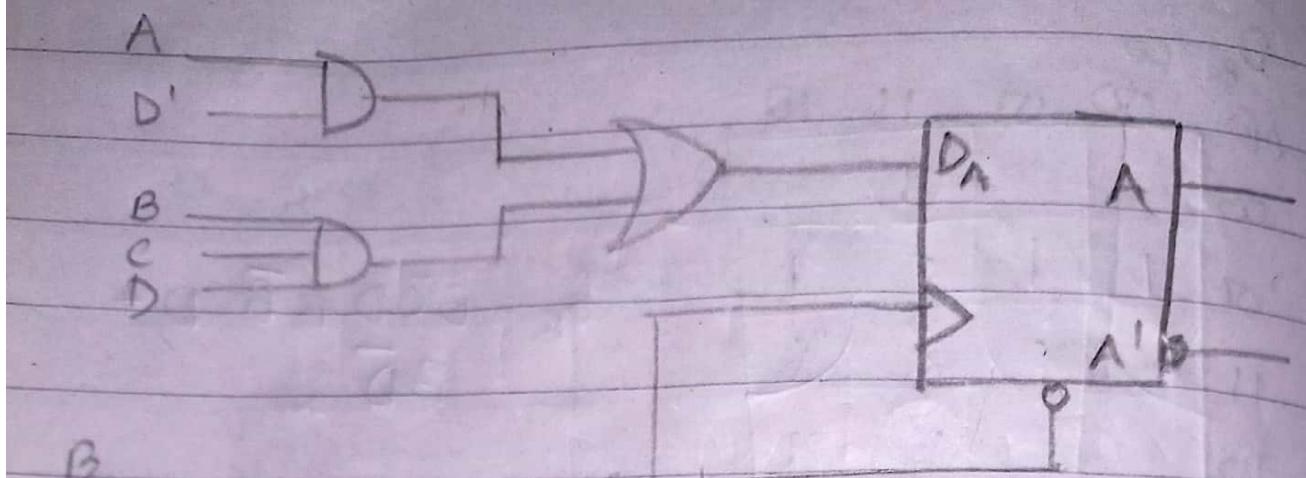
$$D_B = B\bar{C}D + \bar{B}CD + B\bar{D}$$

D_C	CD	00	01	11	10
AB					
CB					
01		1			1
11	X	X	X	X	
10			X	X	

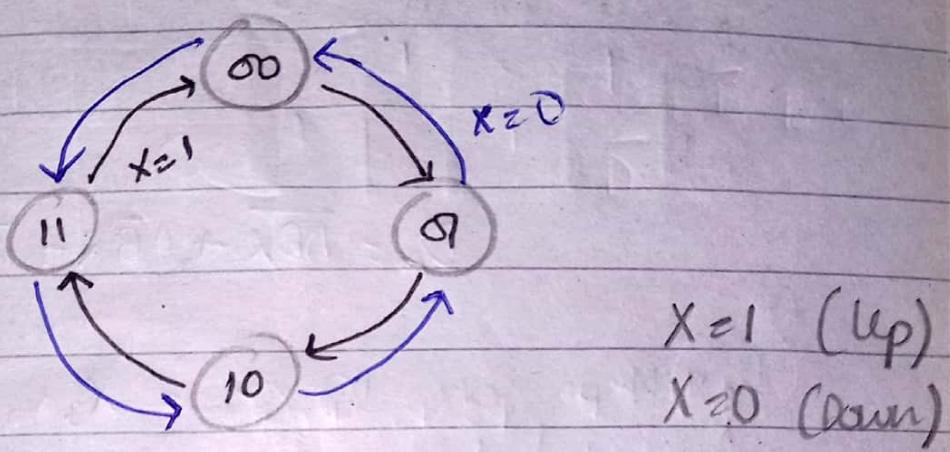
$$D_C = \bar{A}\bar{C}D + \bar{C}\bar{D}$$

D_D	CD	00	01	11	10
AB					
CB					
01		1			1
11	X	X	X	X	
10	1			X	X

$$D_D = \bar{D}$$



Q: Design a 2-bit synchronous up/down counter.



State Table:

Present State A B	Next State $X=0$ $X=1$	
	A B	A B
00	11	01
01	00	10
10	01	11
11	10	00

Excitation Table:

P.S A B X	N.S A B	FF Inputs	
		D _A	D _B
000	11	1	1
001	01	0	1
010	00	0	0
011	10	1	0
100	01	0	1
101	11	1	1
110	10	1	0
111	00	0	0

Minimization:

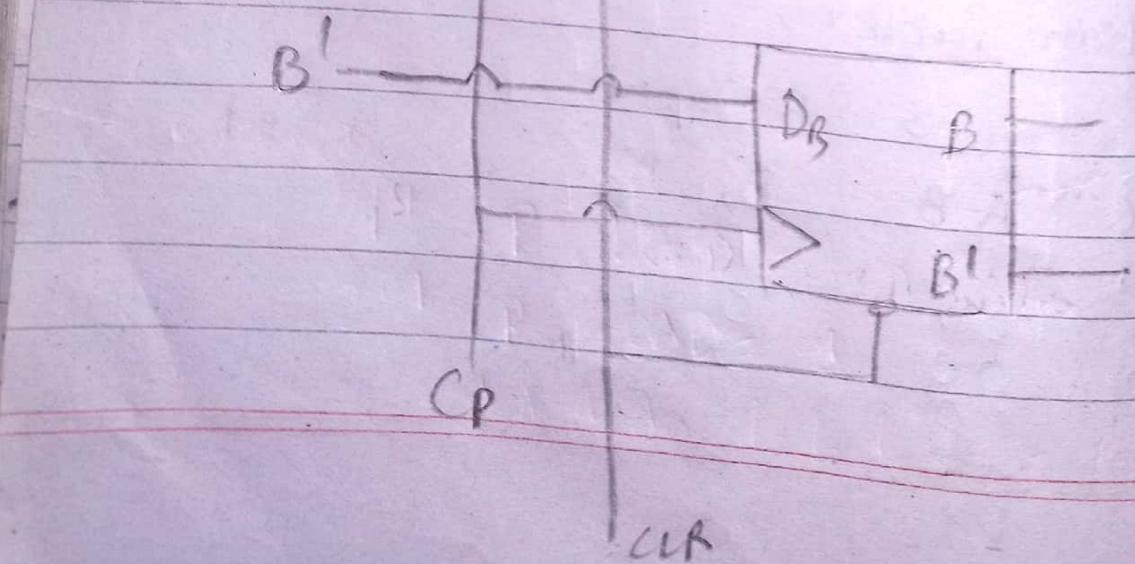
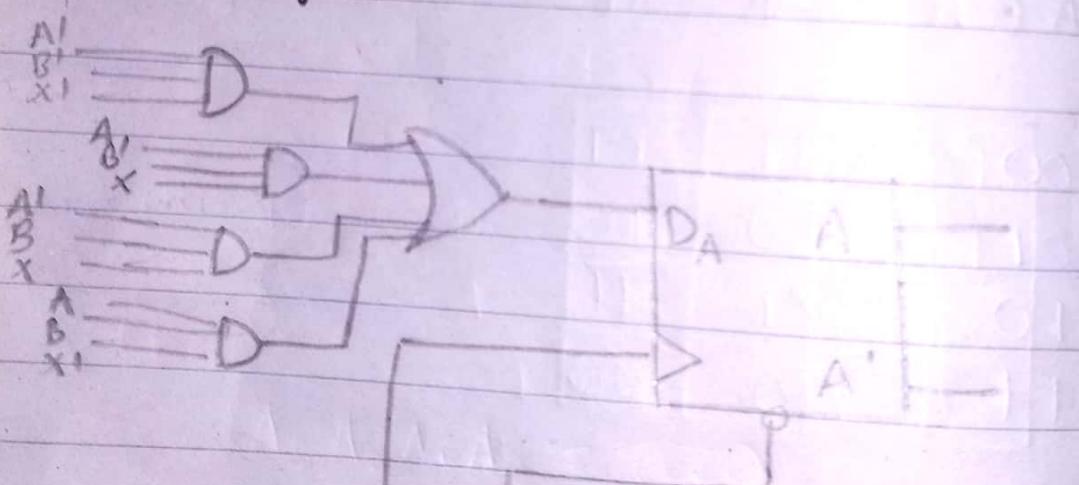
D_A	$B'X$	00	01	11	10
A	0	1	1	1	1
	1	1	1	1	1

$$D_A = \overline{ABX} + A\bar{B}X + \bar{A}BX + AB\bar{X}$$

D_B	$B'X$	00	01	11	10
A	0	1	1	1	1
	1	1	1	1	1

$$D_B = \bar{B}$$

Logic Diagrams:



Registers

A register is a group of [elements
(here we use flip-flops) that work together
as a unit.] memory

two basic functions of registers:

- Data Storage
- Data Movement.

Two types

① Buffer Register

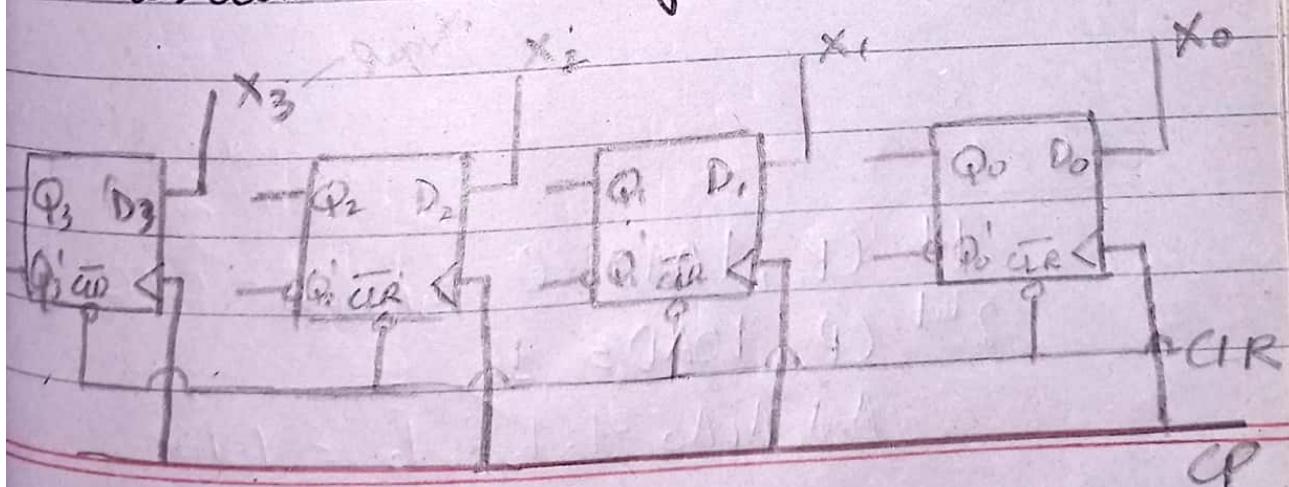
② Shift "

store暂时
the data
or load to
control the data

moves data left or
right, or load to
control the data.

Buffer Register: (store digital word)

Q: Design a 4-bit buffer register and
discuss its working.



Working:

CLR = 0,

BuR always
use parallel
loading.

CLR = 0:

$$Q_3 Q_2 Q_1 Q_0 = 0000$$

CLR = 1:

$$\text{1st CP, } Q_3 Q_2 Q_1 Q_0 = X_3 X_2 X_1 X_0 = 1010$$

1st CP me
no load to
data to R

just on
example
input

Controlled Buffer Register:

Q: Design a 4-bit controlled buffer register & discuss its working.

O (No Load)

1. (Load)

Working:

CLR = 0:

$$A_3 A_2 A_1 A_0 = 0000$$

CLR = 1:

1st CP, LOAD = 0, $A_3 A_2 A_1 A_0 = 0000$

2nd CP, LOAD = 1

$$A_3 A_2 A_1 A_0 = 1, 1, 1, 0 = 1010$$

previous state
maintain

