

State Reduction:
In two-bit binary addition to convert 3 bits to 2 bits.

① State reduction is the procedure for reducing the number of states in a state table while keeping the external input output requirements unchanged.

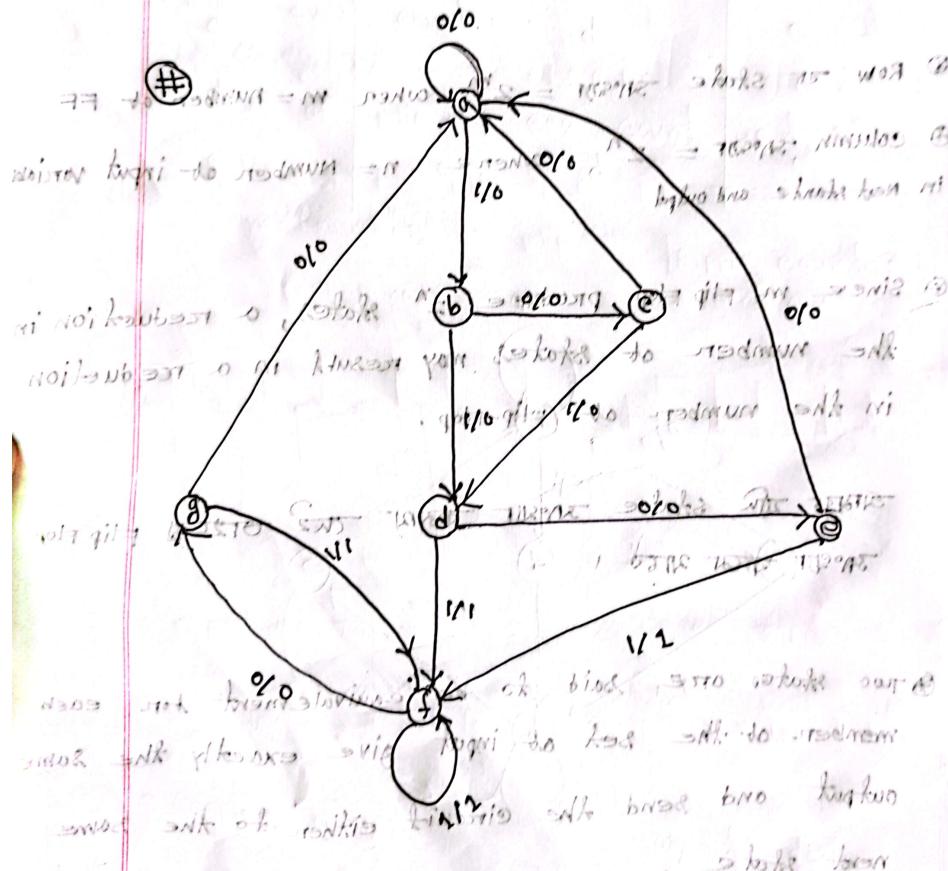
- ② Row for state $m_{\text{state}} = 2^m$, where $m = \text{number of FF}$
- ③ Column power $= 2^n$ where $n = \text{Number of input variable}$
- ④ Since m flip flop produce 2^m states, a reduction in the number of states may result in a reduction in the number of flip-flop.
- ⑤ If state m_{state} is $m_1 m_2 \dots m_m$ then m_i is i^{th} flip flop
- ⑥ Two states are said to be equivalent if each member of the set of input give exactly the same output and send the circuit either to the same next state.

④ When two nodes are equivalent among them,
can be removed without affecting input-output

relationship.

equivalent states are those which do not have

any transitions between them.



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	0
e	a	f	0	1
f	e	f	0	1
g	a	f	0	1

এখন এই Next state & output আমার, যা ২য় ২৪৪০

না, এখন গে কে আমার e দিয়ে replace করতে পারি,

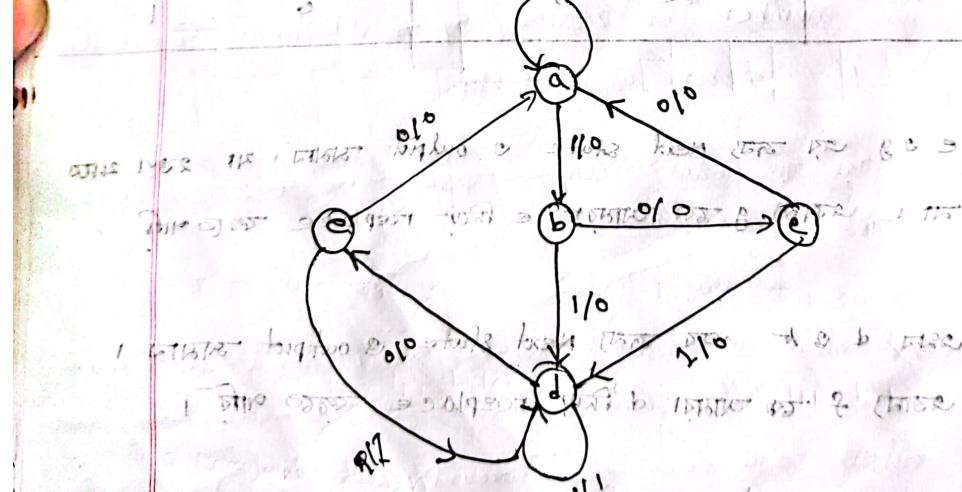
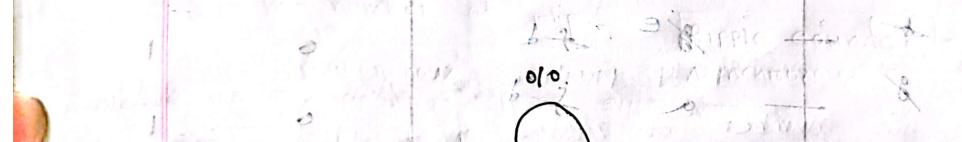
এখন d & f এই, তার Next state & output আমার,
এজন f কে আমার d দিয়ে replace করতে পারি,

Subject :

Date :

Reduced Table

present state input	next state			output	
	$x=0$	$x=1$	$x=1$	$x=0$	$x=1$
a	a	b	b	0	0
b	c	d	d	0	0
c	a	d	d	0	0
d	e	d	d	0	1
e	a	d	d	0	1



State Assignment: is the procedure to assign binary values to the states.

STEPS of

Design sequential circuit from state diagram

1. Determine state diagram
2. From state diagram \rightarrow state table
3. State reduction \rightarrow if possible
4. Binary assignment \rightarrow if possible
5. By finding number of flip-flop assign letter to it
6. Choose type of flip-flop
7. Compose excitation and state table
8. By K-map find the circuit
9. Draw the logic diagram

Counter

top

Subject:

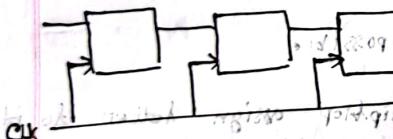
Date:

- ① Counter are sequential circuit to memory state.
- ② Flip Flops are essential part of counters.
- ③ used for counting of pulses.
- ④ If pulses are generated by an event then it is called event counter.
- ⑤ In no of flip flop can count up to 2^n no of pulse.

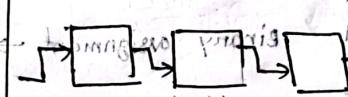
m = number of flip.

Ripple counter

Synchronous

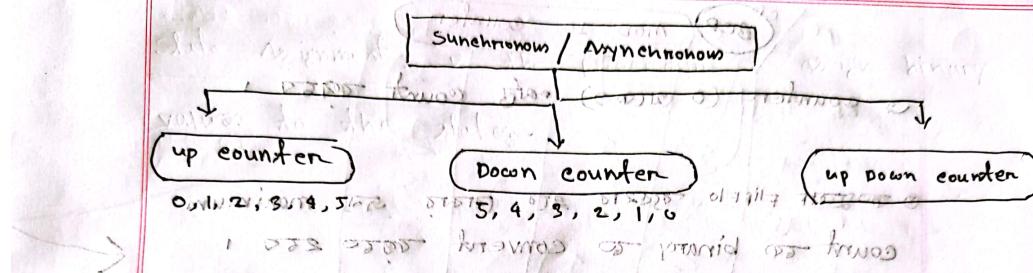


Asynchronous



- ① It's a fast process.
- ② D FF use $\approx 1 \text{ ns}$.
- ③ Time delay $T = T_{FF} + T_{tr}$
- ④ Complex design
- ⑤ Characteristic / excitation table

- ⑥ easy to design
- ⑦ FF (Q_k, T) toggle mode
- ⑧ Q_k \oplus Q_{k-1} = parity pulse
- ⑨ $Q_k Q_{k-1} \dots Q_1 Q_0$ = binary output



④ 2FF flip flop ताकि $n = 3$ रुपे से state $2^3 = 8$ हो

Counting sequence

0	0 0 0	3 bit stored यहाँ सेट करने के लिए 3 bit counter करते हैं। Mod 8 counter 8 तरह states हो 82 counter का क्षमता Mod counter करते हैं।
1	0 0 1	
2	0 1 0	
3	0 1 1	
4	1 0 0	
5	1 0 1	
6	1 1 0	
7	1 1 1	

⑤ 0-7 तक गूंजते रहते हैं।

⑥ 2FF $n = \text{Number of states}$

$n = \text{Number of flip flop}$

$$N = 2^n$$

⑦ Maximum count $2^n - 1$.

$$N = 2^n = 2^3 = 8 \text{ (states)}$$

$$\text{Number of flip flop} = 3$$

गूंजते रहते हैं।

$$\text{maximum count} = 2^n - 1$$

$$= 8 - 1$$

$$= 7$$

Subject :

Date :

BCD

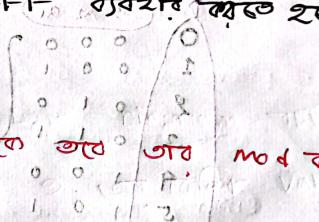
mod 10 counter

↪ counter (0 तक 9) का count रखता है।

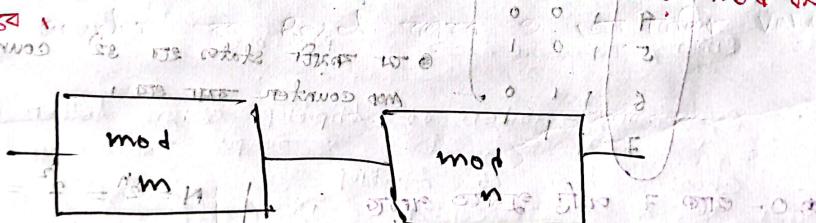
④ काम्प्राया flipflop द्वारा इसे बोर्ड जून्हा maximum count के binary का convert होता है।

Hence max count = 9 (1001)

अये 4 टी की 1 लो 4 टी FF द्वारा होता है।



⑤ यहाँ युटी counter डिजायन करते हैं तो तो mod 4 का



mod (m x n)

1 - S = least significant

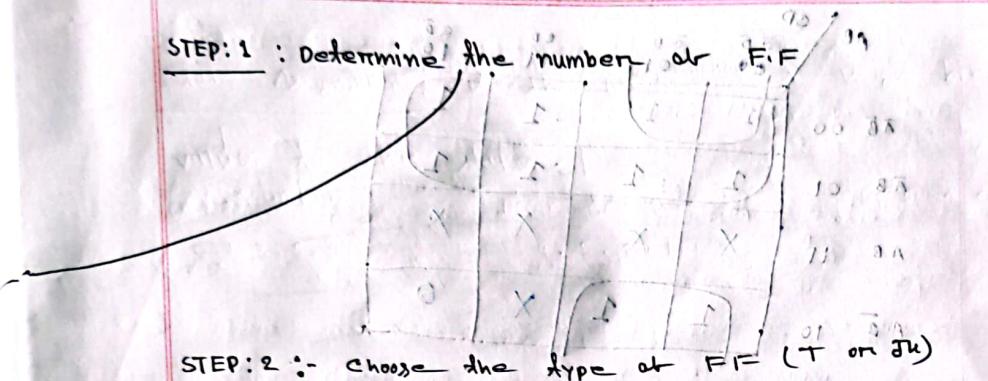
S - P

C

Δ - S = min count

Subject :

Date :

STEP 1 : Determine the number of F.FSTEP 2 :- choose the type of FF (T or J-K)STEP 3 : write the truth table of the counter

STATE 4 bit FF 9(1001) ABCD					output Y
CLK	A	B	C	D	
0	0	0	0	0	0
1	0	0	0	1	1
1	0	0	1	0	1
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	0	1	1
1	0	1	1	0	1
1	0	1	1	1	1
1	1	0	0	0	1
1	1	0	0	1	1
1	1	0	1	0	0
1	1	0	1	0	0
1	1	1	0	0	0
1	1	1	0	1	1
1	1	1	1	0	0
1	1	1	1	1	1

प्राप्ति 9 अंक
हाल हाल 2501

Subject:

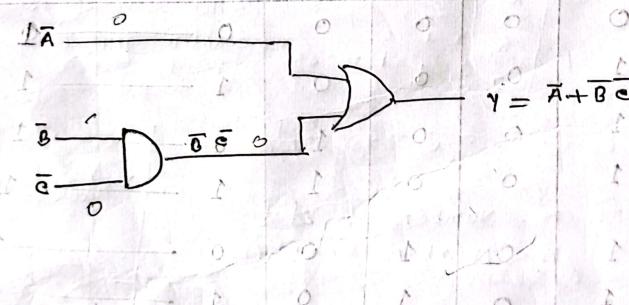
Date:

$\bar{A}B$	CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$	$A\bar{B}$
00	00	1	1	1	1	1
01	01	1	1	1	1	1
11	X	X	X	X	X	0
10	1	1	X	0		

$$F = \bar{A} + \bar{B}$$

5. Now we will design logic diagram for this

It counter has Reset value 9 and Output value 2

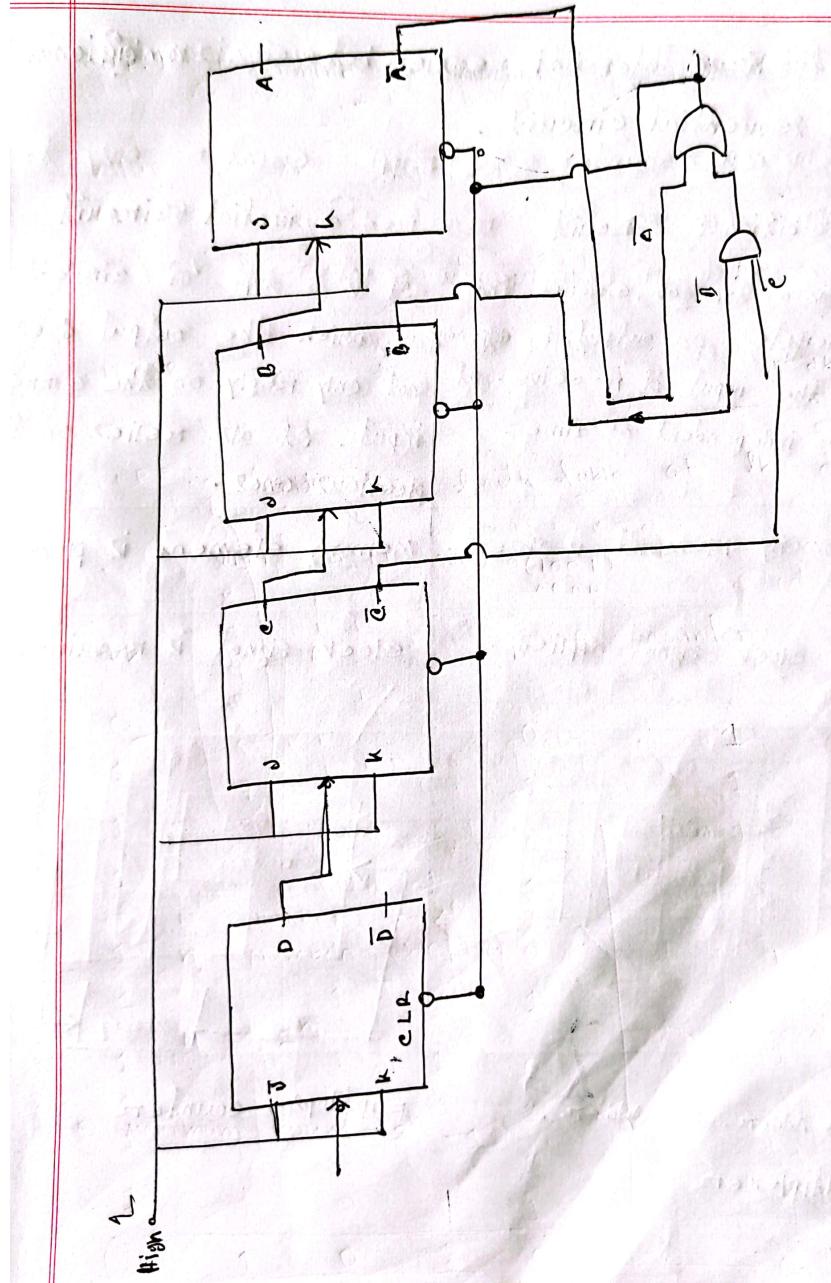


Now we will design the counter:



Subject :

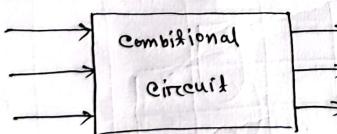
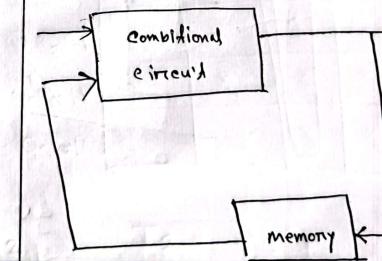
Date :



Subject :

Date :

(b) write down the difference between combinational and sequential circuit.

parameters	combinational Circuit	Sequential Circuit
Meaning and Definition	It is a type of circuit that generates an output by depending on the input it receives. It stays independent of time.	It is a type of circuit in which the output does not only rely on the current input. It also relies on the previous ones.
Memory Element	Memory Element is absent.	Memory element is present.
clock signal	No clock signal applied.	Clock signal is required.
circuit diagram		
Example	Half Adder, Full Adder, Multiplexers	Flipflops, counter

Differences between synchronous and asynchronous counters.

parameters	Synchronous Counter	Asynchronous Counter
Definition	On the synchronous counter there are continuous clock input signals with flip-flop used to produce the output.	On Asynchronous counter there are different clock signals used to produce output.
process	process is fast.	process is slow.
Error	Synchronous counter produce less error than asynchronous.	produce more error
Delay (T)	$T = T_{FF} + T_{ec}$	$T = n \times T_{FF} + T_{ec}$
Circuit		
Made at	Made at D flip flop	Made at T or J-K flip flop

Subject : _____

top[®]
PEN & STATIONERY MANUFACTURERS

Date : _____

Q) Which Counter is easier to design?

Asynchronous counter or ripple counter easy to design. Because the required number of logic gates to design asynchronous counter is very less.

So they are simple to design

• Asynchronous counter
• Ripple counter

• word counter

• clock in register

• timer

• sequence detector

• 2-to-4 decoder

• 4-to-10 decoder

• 4-to-16 decoder

• 8-to-16 decoder

• 16-to-32 decoder

• 32-to-64 decoder

• 64-to-128 decoder

• 128-to-256 decoder

• 256-to-512 decoder

• 512-to-1024 decoder

• 1024-to-2048 decoder

• 2048-to-4096 decoder

• 4096-to-8192 decoder

• 8192-to-16384 decoder

• 16384-to-32768 decoder

• 32768-to-65536 decoder

• 65536-to-131072 decoder

• 131072-to-262144 decoder

• 262144-to-524288 decoder

• 524288-to-1048576 decoder

• 1048576-to-2097152 decoder

• 2097152-to-4194304 decoder

• 4194304-to-8388608 decoder

• 8388608-to-16777216 decoder

• 16777216-to-33554432 decoder

• 33554432-to-67108864 decoder

• 67108864-to-134217728 decoder

• 134217728-to-268435456 decoder

• 268435456-to-536870912 decoder

• 536870912-to-1073741824 decoder

• 1073741824-to-2147483648 decoder

• 2147483648-to-4294967296 decoder

• 4294967296-to-8589934592 decoder

• 8589934592-to-17179869184 decoder

• 17179869184-to-34359738368 decoder

• 34359738368-to-68719476736 decoder

• 68719476736-to-137438953472 decoder

• 137438953472-to-274877906944 decoder

• 274877906944-to-549755813888 decoder

• 549755813888-to-1099511627776 decoder

• 1099511627776-to-2199023255552 decoder

• 2199023255552-to-4398046511104 decoder

• 4398046511104-to-8796093022208 decoder

• 8796093022208-to-17592186044416 decoder

• 17592186044416-to-35184372088832 decoder

• 35184372088832-to-70368744177664 decoder

• 70368744177664-to-140737488355328 decoder

• 140737488355328-to-281474976710656 decoder

• 281474976710656-to-562949953421312 decoder

• 562949953421312-to-1125899906842624 decoder

• 1125899906842624-to-2251799813685248 decoder

• 2251799813685248-to-4503599627370496 decoder

• 4503599627370496-to-9007199254740992 decoder

• 9007199254740992-to-18014398509481984 decoder

• 18014398509481984-to-36028797018963968 decoder

• 36028797018963968-to-72057594037927936 decoder

• 72057594037927936-to-144115188075855872 decoder

• 144115188075855872-to-288230376151711744 decoder

• 288230376151711744-to-576460752303423488 decoder

• 576460752303423488-to-1152921504606846976 decoder

• 1152921504606846976-to-2305843009213693952 decoder

• 2305843009213693952-to-4611686018427387904 decoder

• 4611686018427387904-to-9223372036854775808 decoder

• 9223372036854775808-to-18446744073709551616 decoder

• 18446744073709551616-to-36893488147419103232 decoder

• 36893488147419103232-to-73786976294838206464 decoder

• 73786976294838206464-to-147573952589676412928 decoder

• 147573952589676412928-to-295147905179352825856 decoder

• 295147905179352825856-to-590295810358705651712 decoder

• 590295810358705651712-to-118059162071741130344 decoder

• 118059162071741130344-to-236118324143482260688 decoder

• 236118324143482260688-to-472236648286964521376 decoder

• 472236648286964521376-to-944473296573929042752 decoder

• 944473296573929042752-to-1888946593147858085504 decoder

• 1888946593147858085504-to-3777893186295716171008 decoder

• 3777893186295716171008-to-7555786372591432342016 decoder

• 7555786372591432342016-to-15111572745182864684032 decoder

• 15111572745182864684032-to-30223145490365729368064 decoder

• 30223145490365729368064-to-60446290980731458736128 decoder

• 60446290980731458736128-to-120892581961462917472256 decoder

• 120892581961462917472256-to-241785163922925834944512 decoder

• 241785163922925834944512-to-483570327845851669889024 decoder

• 483570327845851669889024-to-967140655691703339778048 decoder

• 967140655691703339778048-to-1934281311383406679556096 decoder

• 1934281311383406679556096-to-3868562622766813359112192 decoder

• 3868562622766813359112192-to-7737125245533626718224384 decoder

• 7737125245533626718224384-to-15474250491067253436448768 decoder

• 15474250491067253436448768-to-30948500982134506872897536 decoder

• 30948500982134506872897536-to-61897001964269013745795072 decoder

• 61897001964269013745795072-to-123794003928538027491590144 decoder

• 123794003928538027491590144-to-247588007857076054983180288 decoder

• 247588007857076054983180288-to-495176015714152109966360576 decoder

• 495176015714152109966360576-to-990352031428304219932721152 decoder

• 990352031428304219932721152-to-1980704062856608439865442304 decoder

• 1980704062856608439865442304-to-3961408125713216879730884608 decoder

• 3961408125713216879730884608-to-7922816251426433759461769216 decoder

• 7922816251426433759461769216-to-15845632522852867518923538432 decoder

• 15845632522852867518923538432-to-31691265045705735037847076864 decoder

• 31691265045705735037847076864-to-63382530091411470075694153728 decoder

• 63382530091411470075694153728-to-126765060182822940151388307556 decoder

• 126765060182822940151388307556-to-253530120365645880302776615112 decoder

• 253530120365645880302776615112-to-507060240731291760605553230224 decoder

• 507060240731291760605553230224-to-1014120481462583521211106460448 decoder

• 1014120481462583521211106460448-to-202824096292516704242221292096 decoder

• 202824096292516704242221292096-to-405648192585033408484442584192 decoder

• 405648192585033408484442584192-to-811296385170066816968885168384 decoder

• 811296385170066816968885168384-to-1622592770340133633937770336768 decoder

• 1622592770340133633937770336768-to-3245185540680267267875540673536 decoder

• 3245185540680267267875540673536-to-6490371081360534535751080346712 decoder

• 6490371081360534535751080346712-to-12980742162721069071502160693424 decoder

• 12980742162721069071502160693424-to-25961484325442138143004320386848 decoder

• 25961484325442138143004320386848-to-51922968650884276286008640773696 decoder

• 51922968650884276286008640773696-to-103845937301769532520017281547392 decoder

• 103845937301769532520017281547392-to-207691874603539065040034563094784 decoder

• 207691874603539065040034563094784-to-415383749207078130080069126189568 decoder

• 415383749207078130080069126189568-to-830767498414156260160013825218916 decoder

• 830767498414156260160013825218916-to-166153499682831252032002765043792 decoder

• 166153499682831252032002765043792-to-332306999365662504064005530087888 decoder

• 332306999365662504064005530087888-to-664613998731325008128001106565776 decoder

• 664613998731325008128001106565776-to-132922799746265001625600221313552 decoder

• 132922799746265001625600221313552-to-26584559949253000325120044262704 decoder

• 26584559949253000325120044262704-to-53169119898506000650240088525408 decoder

• 53169119898506000650240088525408-to-106338239797012001300480177050816 decoder

• 106338239797012001300480177050816-to-212676479594024002600960354101632 decoder

• 212676479594024002600960354101632-to-425352959188048005201920708203264 decoder

• 425352959188048005201920708203264-to-850705918376096001040384141640628 decoder

• 850705918376096001040384141640628-to-1701411836752192002080768283281256 decoder

• 1701411836752192002080768283281256-to-3402823673504384004161536566562536 decoder

• 3402823673504384004161536566562536-to-680564734700876800832307313312512 decoder

• 680564734700876800832307313312512-to-136112946940175360166461466625248 decoder

• 136112946940175360166461466625248-to-272225893880350720332928293312516 decoder

• 272225893880350720332928293312516-to-544451787760701440665856586625312 decoder

• 544451787760701440665856586625312-to-108890357532140280133111113325064 decoder

• 108890357532140280133111113325064-to-217780715064280560266222226650128 decoder

• 217780715064280560266222226650128-to-435561430128561120532444446300256 decoder

• 435561430128561120532444446300256-to-871122860256112240106488889200512 decoder

• 871122860256112240106488889200512-to-1742245720522244802129777784

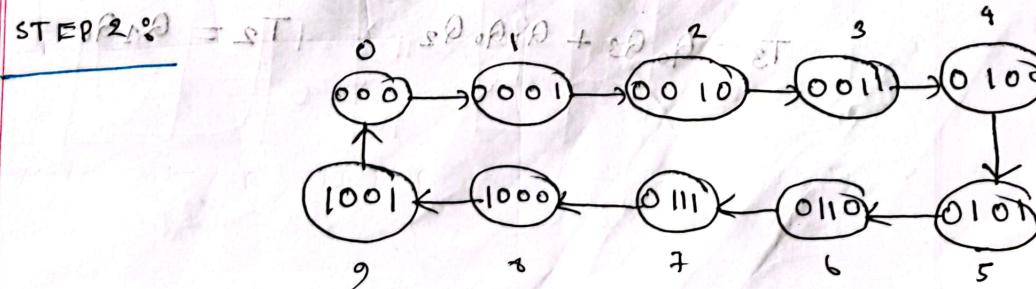
c) Explain different types of clocked triggered circuit.

Clocked Circuits are sequential circuits that needs a clock pulse to function. The output and functionalities of the circuits are controlled based on clock.

Example:	Flip Flop	1	0	1	0	0	0	1	0	2
	Counter	0	1	1	0	1	0	1	0	3
	Register	1	1	1	0	0	1	1	0	4
		1	1	0	0	0	1	1	1	5
		0	0	1	0	0	1	0	0	6

Draw the table and block diagram of 4 bit

Binary parallel mode - 10				Counter using JK flip flop			
01 11 10 20				01 11 10 20			
Using T				01 11 10 20			
STEP 1 :- Number of flip flop needed				01 11 10 20			



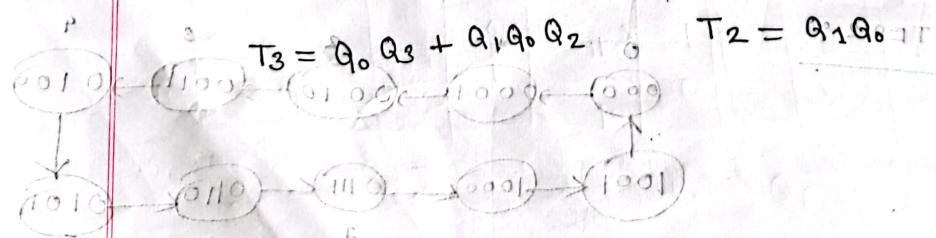
STEP: 3 : Truth Table

to implement Mealy type

CLk	P.S.				N.S.			Required Excitation				
	Q_3	\bar{Q}_2	Q_2	$Q_1 Q_0$	Q_3	\bar{Q}_2	Q_2	Q_0	T_3	T_2	T_1	T_0
1	0	0	0	0	0	0	0	1	0	0	0	1
2	0	0	0	1	0	1	0	0	0	0	1	1
3	0	0	1	0	0	0	1	1	0	0	0	1
4	0	0	1	1	0	1	0	0	0	1	1	1
5	0	1	0	0	0	1	0	1	0	0	0	1
6	0	1	0	1	0	1	1	0	0	1	1	1
7	0	1	1	0	0	1	1	1	0	0	0	1
8	0	1	1	1	1	0	0	0	1	1	1	1
9	1	0	0	0	1	0	0	1	0	0	0	1
10	1	0	0	1	0	0	1	0	1	0	0	1

$\bar{Q}_3 \bar{Q}_2$	$Q_3 \bar{Q}_2$	$Q_3 Q_2$	$\bar{Q}_3 Q_2$	
00	0°	0¹	0³	0²
01	0⁴	0⁵	1⁷	0⁶
11	X	X¹	X³	X²
10	0⁸	1?	X⁹	X¹⁰
				T_3

$\bar{Q}_3 \bar{Q}_2$	$Q_3 \bar{Q}_2$	$Q_3 Q_2$	$\bar{Q}_3 Q_2$	
00	0°	0¹	1	0
01	0⁴	0⁵	0	1
11	X	X	X	X
10	0⁸	0	X	X
				T_2



Subject :

(9017917463) 30/3/2024

top®
PEN & STATIONERY INDIA LTD.

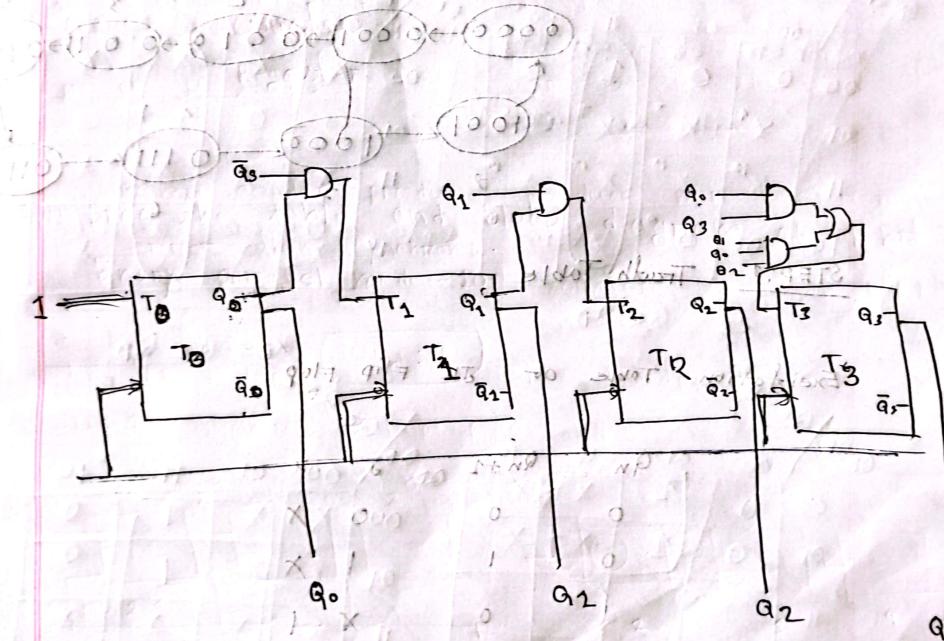
Date :

$Q_3 Q_2$	Q_1	Q_0	
00	0	1	1
01	1	1	1
11	X	X	X
10	X	X	X

$Q_3 Q_2$	Q_1	Q_0	
00	1	1	1
01	1	1	1
11	X	X	X
10	1	1	X

$$T_1 = Q_0 \bar{Q}_3$$

$$T_0 = 1$$



Subject :

Using JK Flip Flop

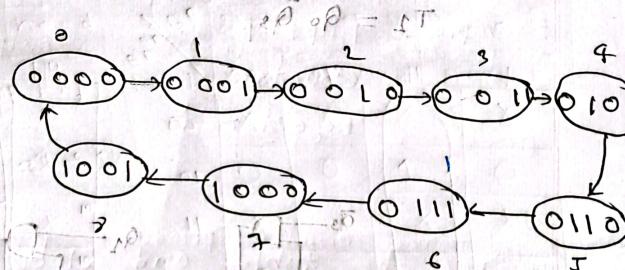
Date :

STEP: 1

$$N \leq 2^n, 10 \leq 2^n, n = 4$$

STEP: 2

00	11	10	01	10	00
X	X	X	X	X	X
11	00	11	00	11	00
X	X	X	X	X	X
01	10	01	10	01	10

STEP: 3 Truth Table

Excitation Table or

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

Subject :

Date :

Num	D.S				N.S				In	KA	J _D	K _B	J _E	K _E	
	G _A	G _B	G _C	G _D	G _{AII}	G _{BII}	G _{CII}	G _{DII}							
0	0	0	0	0	0	0	0	1	0	X	0	X	0	X	1
1	0	0	0	1	0	0	1	0	0	X	0	X	1	X	1
2	0	0	1	0	0	0	1	1	0	X	0	X	X	0	1
3	0	0	1	1	0	1	0	0	0	X	1	X	X	1	1
4	0	1	0	0	0	X	0	1	0	X	X	0	X	X	
5	0	1	0	1	0	1	0	0	0	X	X	0	1	X	1
6	X	0	1	0	0	1	0	1	0	X	X	0	X	0	1
7	0	0	1	0	1	0	0	0	X	X	X	1	X	0	1
8	1	0	0	X	0	0	1	X	X	0	X	0	X	1	X
9	1	0	0	1	0	0	0	0	X	X	0	X	0	X	1
10	1	0	1	0	X	X	X	X	X	X	X	X	X	X	X
11	1	0	1	0	X	X	X	X	X	X	X	X	X	X	X
12	1	1	0	0	X	X	X	X	X	X	X	X	X	X	X
13	1	1	0	1	X	X	X	X	X	X	X	X	X	X	X
14	1	1	1	0	X	X	X	X	X	X	X	X	X	X	X
15	1	1	1	1	X	X	X	X	X	X	X	X	X	X	X
16					X	X	X	X	X	X	X	X	X	X	X

Here

$$J_D = K_D = 1$$

$$V_D = 1$$

$$A_D = 2 \times 6$$

Subject :

Date :

$Q_A Q_B$	$Q_C Q_D$	$\bar{Q}_A \bar{Q}_B$	$\bar{Q}_C \bar{Q}_D$
00	0	1	3
01	1	2	6
10	X ¹²	X ¹³ X ¹⁵	X ¹⁴
11	X ⁹ X ¹⁰	X ¹¹	X ¹⁴

$Q_A Q_B$	$Q_C Q_D$	$\bar{Q}_A \bar{Q}_B$	$\bar{Q}_C \bar{Q}_D$
00	X ⁰	X ¹ X ³	X ²
01	X ⁴	X ⁵ X ⁷	X ⁶
10	X ¹²	X ¹³ X ¹⁵	X ¹⁴
11	0 ⁸	1 ⁹ X ¹⁰	X ¹¹

$$J_A = Q_C Q_D \bar{Q}_B$$

$Q_A Q_B$	$Q_C Q_D$	$\bar{Q}_A \bar{Q}_B$	$\bar{Q}_C \bar{Q}_D$
00	0	0'	1 ³
01	X ⁴	X ⁵	X ⁷
10	X ¹²	X ¹³ X ¹⁵	X ¹⁴
11	0 ⁸	0 ⁹ X ¹⁰	X ¹¹

$Q_A Q_B$	$Q_C Q_D$	$\bar{Q}_A \bar{Q}_B$	$\bar{Q}_C \bar{Q}_D$
00	X ⁰	X ¹	X ³
01	0 ¹⁰	1 ¹¹	0 ¹²
10	X ¹³	X ¹⁴ X ¹⁵	X ¹⁶
11	X ¹⁷	X ¹⁸	X ¹⁹

$$J_B = Q_C Q_D X$$

$$K_B = Q_C Q_D$$

$Q_A Q_B$	$Q_C Q_D$	$\bar{Q}_A \bar{Q}_B$	$\bar{Q}_C \bar{Q}_D$
00	0	1	X
01	0	1	X
10	X	X X	X X
11	0	0	X X

$$J_C = Q_D \bar{Q}_A$$

$Q_A Q_B$	$Q_C Q_D$	$\bar{Q}_A \bar{Q}_B$	$\bar{Q}_C \bar{Q}_D$
00	X	X	1
01	X	X	1
10	X	X X	X
11	X	X X	X

$$J_D = Q_D \bar{Q}_A$$

$$K_C = Q_D$$

Subject :

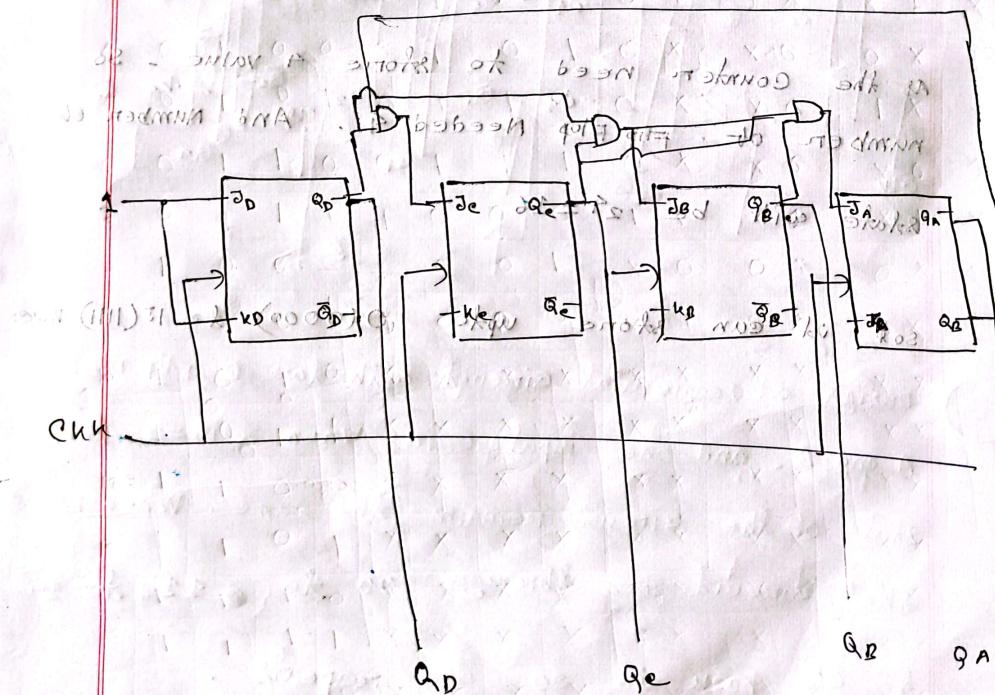
Date :

$$J_A = Q_e Q_D Q_B \quad K_A = Q_D k_D \quad \text{both} \Rightarrow \text{state diagram}$$

$$J_C = Q_e Q_D \quad K_B = Q_e Q_D \quad \text{NG: Bi-directional transition}$$

$$J_C = Q_D \bar{Q}_A \quad K_e = Q_D$$

$$J_D = k_D = 1$$



Here $J_D = K_D = 1$

Subject :

Date :

(c) Draw truth table and block diagram of 4-bit binary ripple counter using JK flip flop

$$Q_0 = S_0$$

$$Q_1 = Q_0 \oplus S_1$$

$$Q_2 = Q_1 \oplus S_2$$

$$Q_3 = Q_2 \oplus S_3$$

STEP: 1 :-

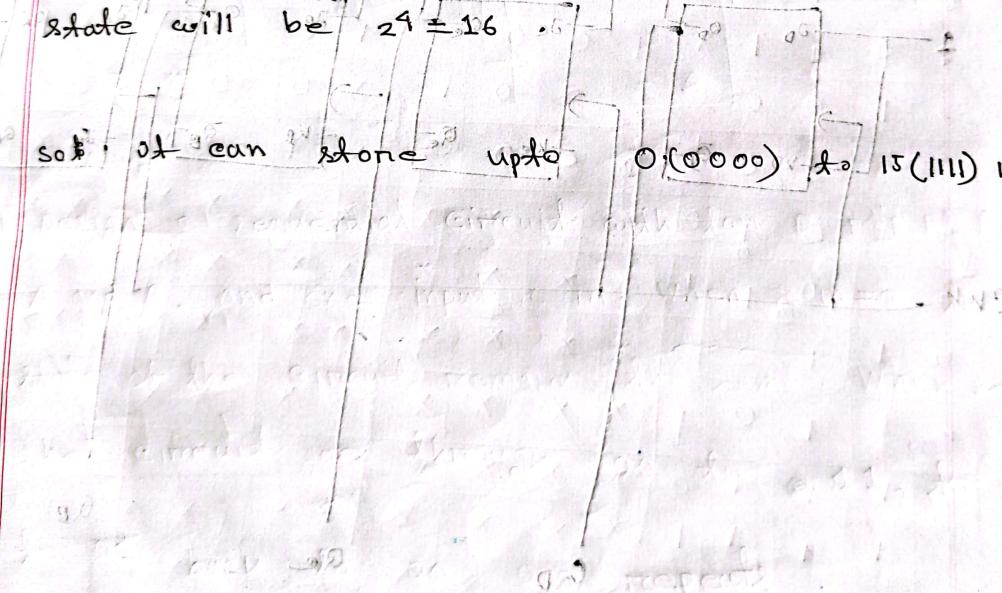
$$N \leq 2^n, 16 \leq 2^n, n = 4$$

$$L = 2^4 = 16$$

As the Counter need to store 4 value - so
number of Flip Flop Needed 4. And Number of

state will be $2^4 = 16$

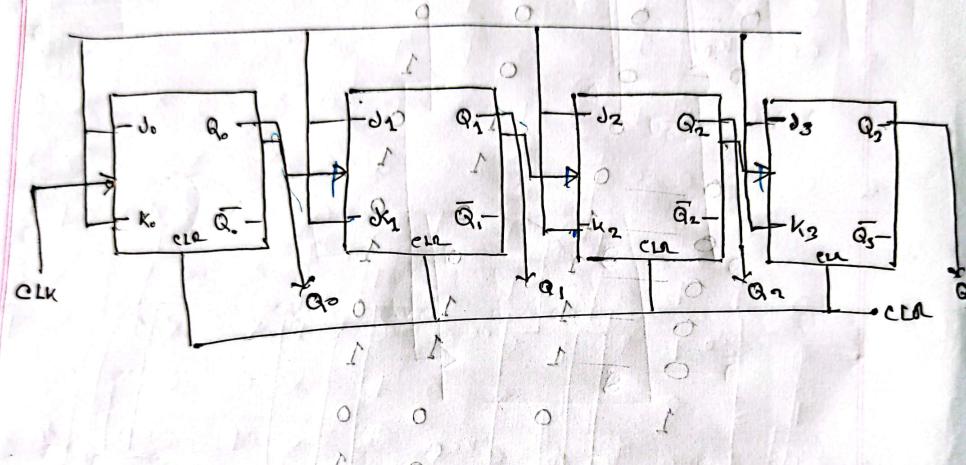
so it can store upto 0(0000) to 15(1111) value.



STEP: 2 Truth Table

Q_3	Q_2	Q_1	Q_0	Y
0	0	0	0	0
0	0	0	1	1
0	0	1	1	0
0	1	0	1	1
0	1	1	0	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	1	1	1	1

Subject : Date :

Diagram :

)) Design a sequential circuit with two D flip flop A and B, and one input x in. When x in = 0 the state of the circuit remains the same. When x in = 1 the circuit goes through 00 to 011, to 10 to 01 back to 00 and repeats.

Subject:
P. S

top

Date:

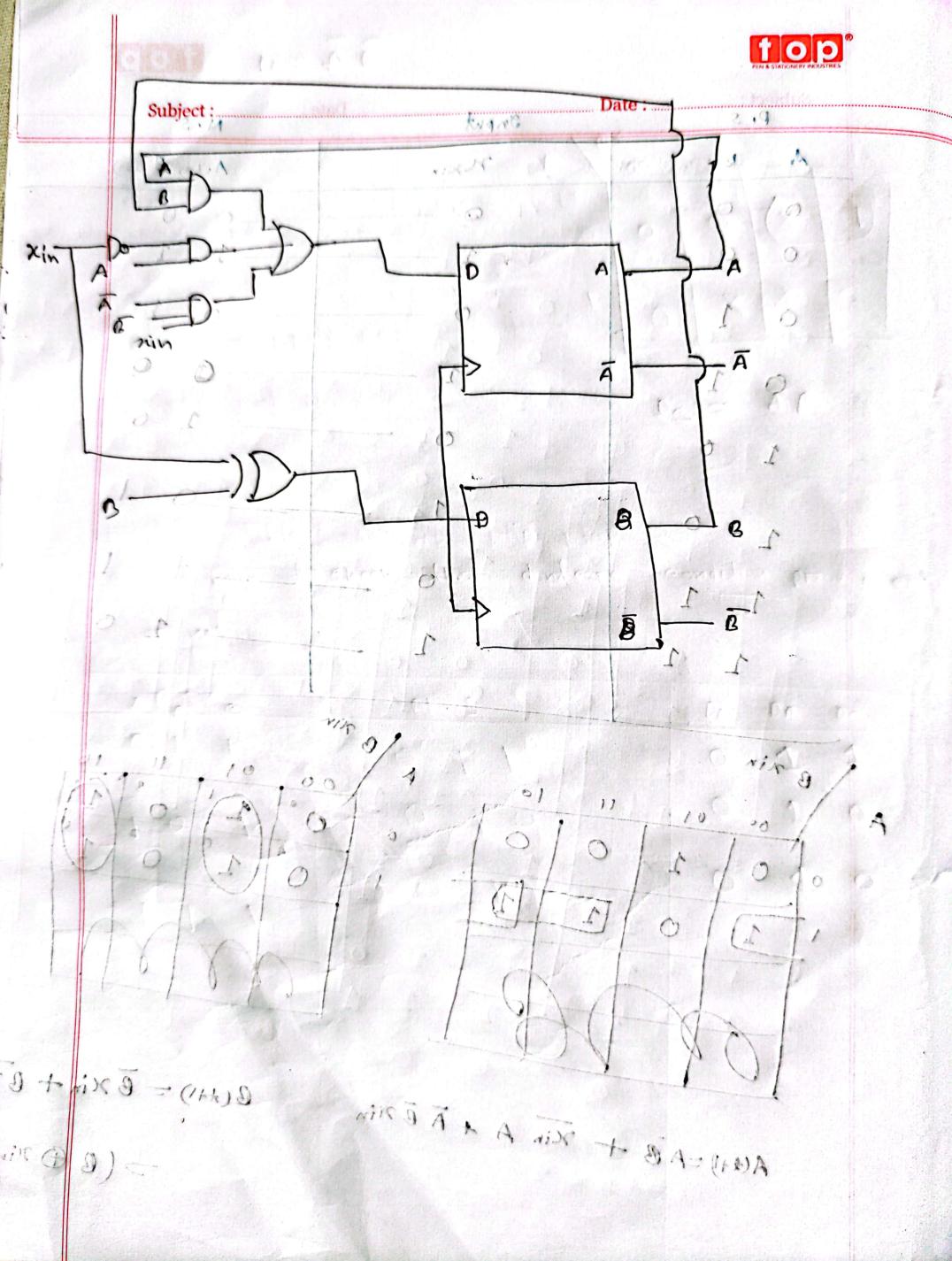
N-3

A	B	x_{in}	$A \oplus B \oplus x_{in}$
0	0	0	0 0
0	0	1	1 1
0	1	0	0 1
0	1	1	0 0
1	0	0	1 0
1	0	1	0 1
1	1	0	1 1
1	1	1	2 0

A	$B \oplus x_{in}$			
	00	01	11	10
0	0 1	0 0	1 1	1 0
1	1 0	1 1	0 0	0 1

$$A(x_1) = A \bar{B} + \bar{x}_{in} A + \bar{A} \bar{B} x_{in}$$

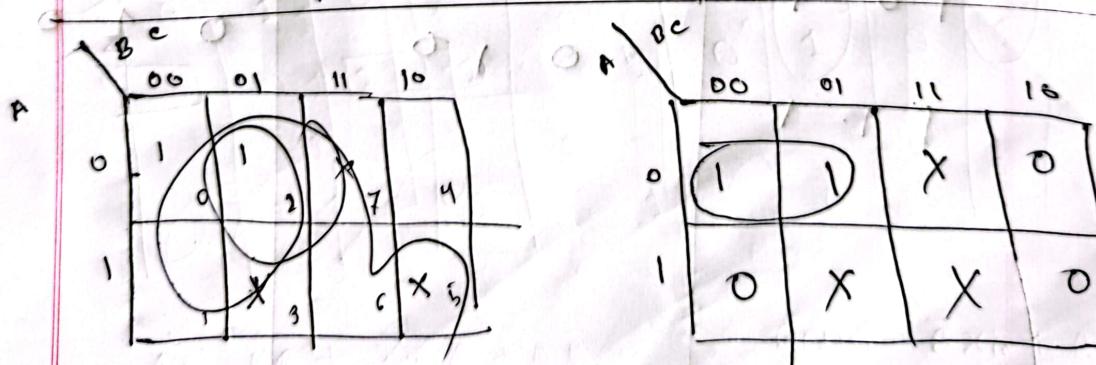
$$\begin{aligned} B(x_1) &= \bar{B} x_{in} + B \bar{x}_{in} \\ &\Rightarrow (B \oplus x_{in}) \end{aligned}$$



(d) Design a counter with the following sequence
0 9 2 1 6, and repeat. Use D FF to build it.

STATE TABLE

P.S	N.S			D	I	FF input		
	A	B	C			A(t+1)	B(t+1)	C(t+1)
0	0	0	0	1	0	0	1	0
1	0	0	1	1	1	0	1	0
2	0	1	0	0	0	1	0	1
3	0	1	1	X	X	X	X	X
4	1	0	0	0	1	0	0	1
5	1	0	1	X	X	X	X	X
6	1	1	0	0	0	0	0	0
7	1	1	1	X	X	X	X	X



$$D_A = A(t+1) = \bar{A} \bar{B} \bar{C}$$

$$D_n = \bar{A} \bar{B}$$

Subject :

Date :

		BC		AB	
		00	01	11	10
A	B	00	0	X	0
		01	X	X	0
10		11			

		BC		AB	
		00	01	11	10
A	B	00	0	(X)	1
		01	X	X	0
10		11			

$$D_B = \bar{A} \bar{B} + A \bar{B}$$

$$D_C = \bar{A} B$$

④ we have to check that don't care condition provide error

A	P	O	S	C	M	S	D_A	D_B	D_C
0	0	0	1	0	0	1	0	1	1
1	X	X	X	X	X	X	1	1	0

A	P	O	S	C	M	S	D_A	D_B	D_C
0	1	1	0	0	1	1	0	1	1
1	X	X	0	1	X	X	1	1	0
0	0	1	0	1	0	0	0	1	0

$$\bar{A} \bar{B} = (\bar{A} \bar{B})' = AB$$