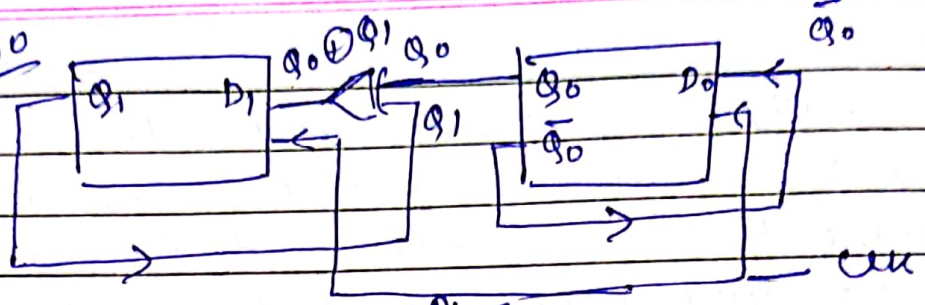
21.11.2022 Monday

Date: / / Page no:

Q) Initial state 00

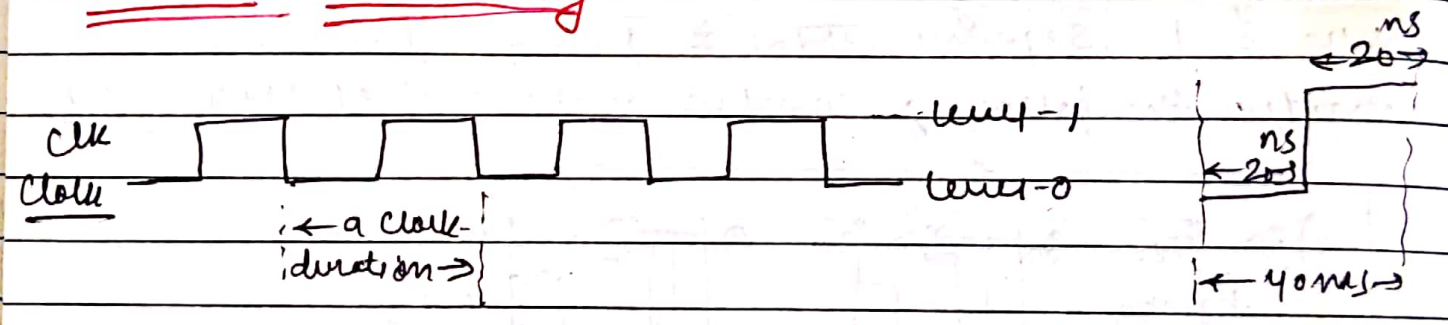


$D_1 = Q_0 \oplus Q_1$, $D_0 = \bar{Q}_0$

Q_1	Q_0	clk	D_1	D_0	Q_1	Q_0
0	0	0	–	–	0	0
		1	0	1	0	1
		2	1	0	1	0
		3	1	1	1	1
		4	0	0	0	0

Same sequence stop

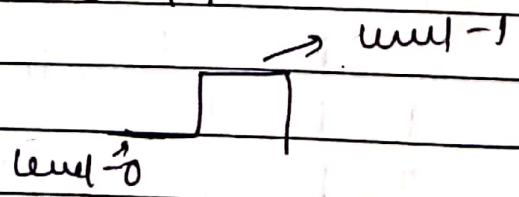
CLOCK Presuming



Every clock has 50% duty cycle

Based on the processing of clock, ffs can be classified as

(i) Level Triggered ffs : ffs will work on either at level 0 or at level 1

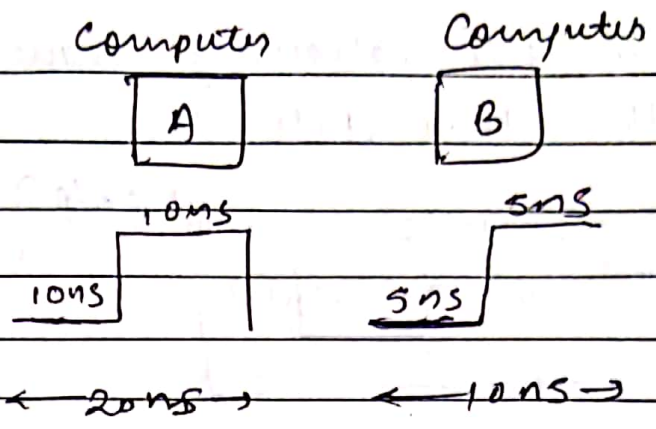


→ either at level 0 or at level 1 we can process as 50% duty cycle

computer has speed = 2 GHz → cycles or clocks
1 s = 2×10^9 cycles / clocks
A clock duration = 0.5

Date: / / Page no:

Communication



All computers don't have same freq. then there will be no problem.

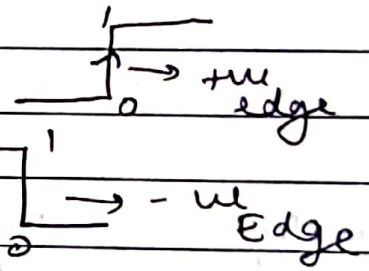
Let 1 bit can be processed in 5ns
'A' can process 12 bits and 'B' can receive 1 bit per clock

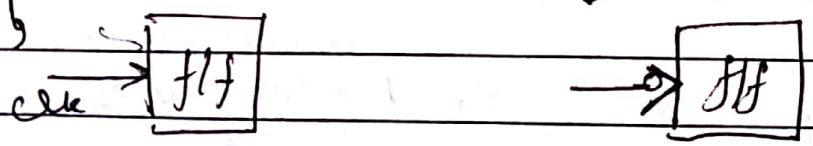
If we go for clk. synchronization then it means every 2nd bit transferred will be lost.

(ii) Edge Triggered flf : It can work either on +ve edge or -ve edge of a clk.

(a) +ve ~~Egde~~ Edge ~~trigg~~ triggered flf :

(b) -ve Edge triggered flf :





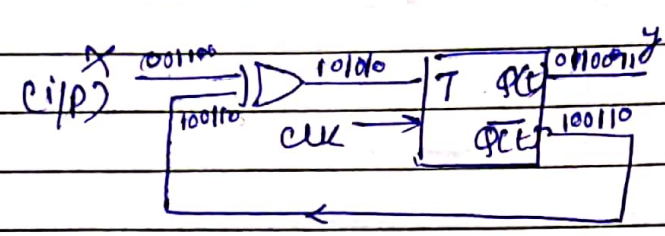
Race around condition → due to not synchronising clk (data is lost).

We generally used Edge Triggered.
No dependency of on Time Duration
1 clk → same work since only once edge changes
∴ Communication possible.

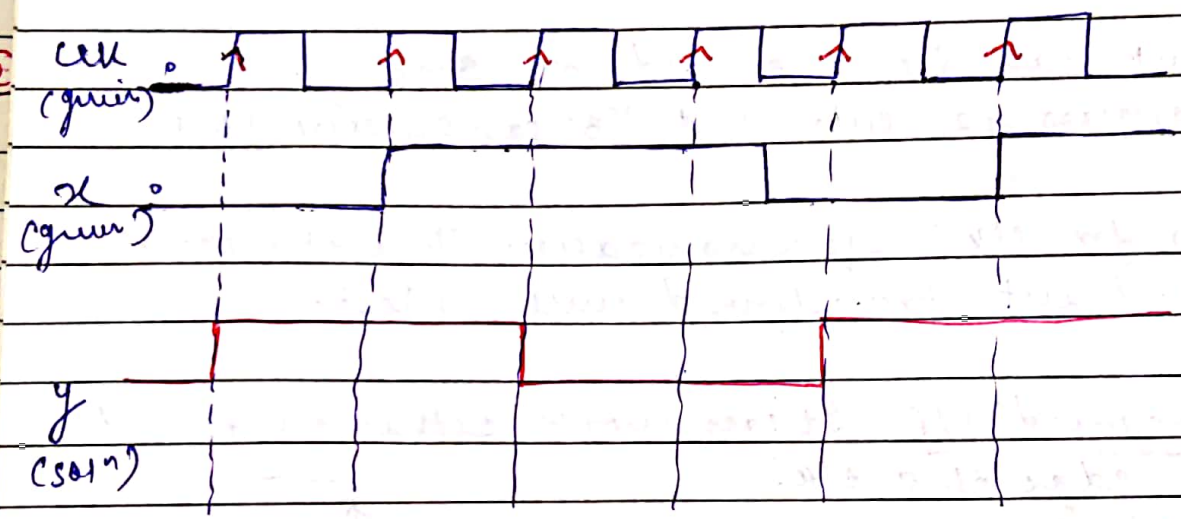
Scanned with CamScanner

Date: / / Page no:

Q) consider the following sequential ckt with T flip flop with initial state '0'.



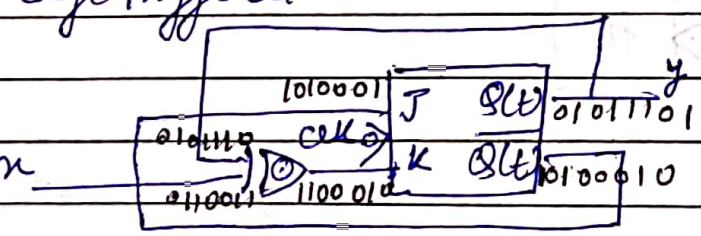
Find the waveform of o/p y.



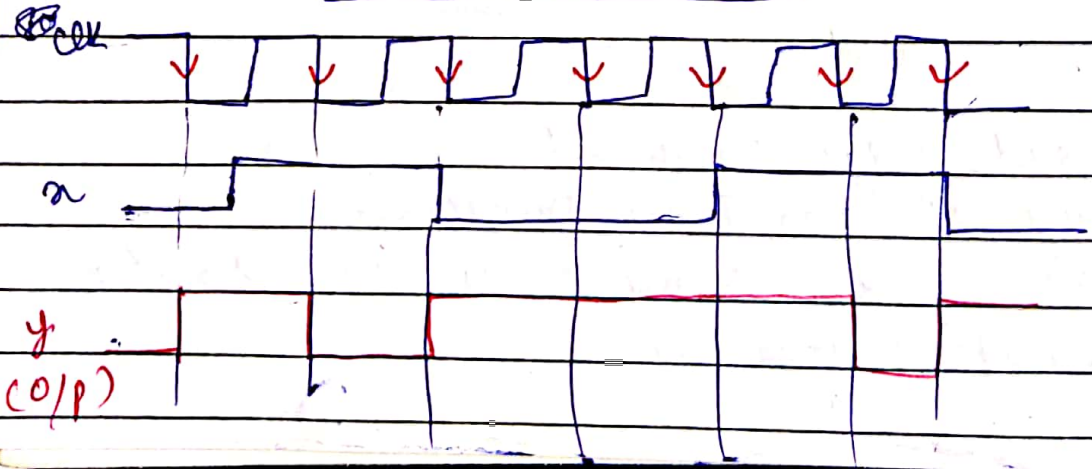
mark the +ve edge

Whenever there is the falling pulse one

Q) consider JK flip flop with initial state '0' -ve edge triggered



Find o/p waveform of y



mark -ve edge for clk.

Given

By default initial state = 000.

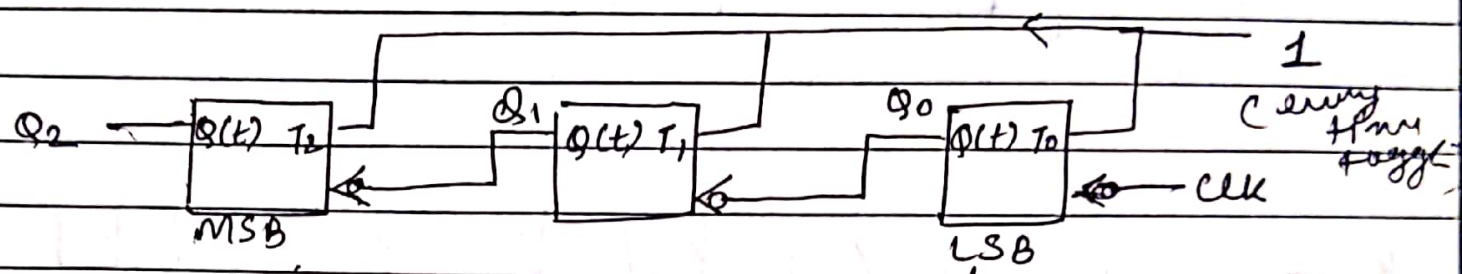
Date: / / Page no:

ASYNCHRONOUS COUNTER

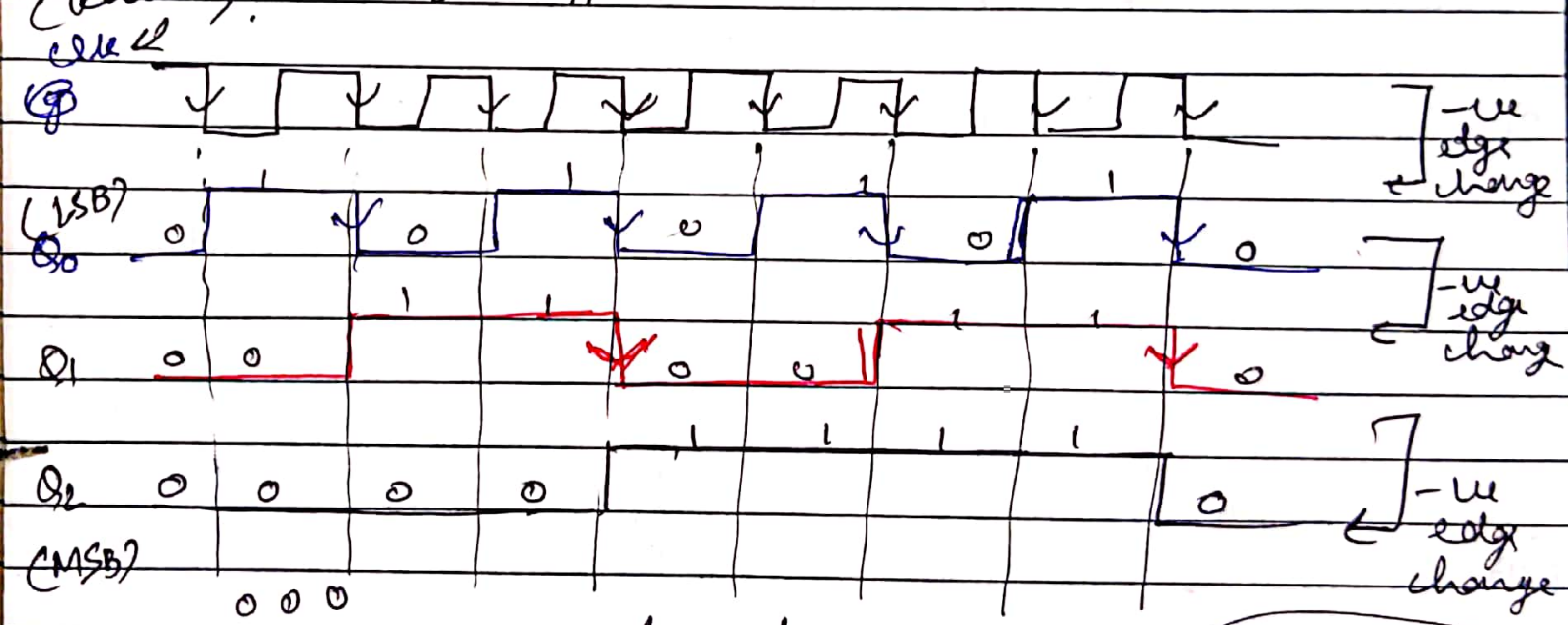
O/p of ffs may work as clk for other ffs. (Ripple Counter) It means all the ffs will not work parallelly.

There are 2 types of Ripple Counter:

(i) **Binary Ripple Counter**: It counts sequences either in forward ~~di~~ inorder or backward order



Q_2 will work on carry - ve edge triggered of Q_1 (known) similarly Q_1



0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1
0	0	0

depends on previous o/p as clk. ∴ LSB T_0 will work first thus in a cascading way all.

T=1 ∴ easy

Always change previous output

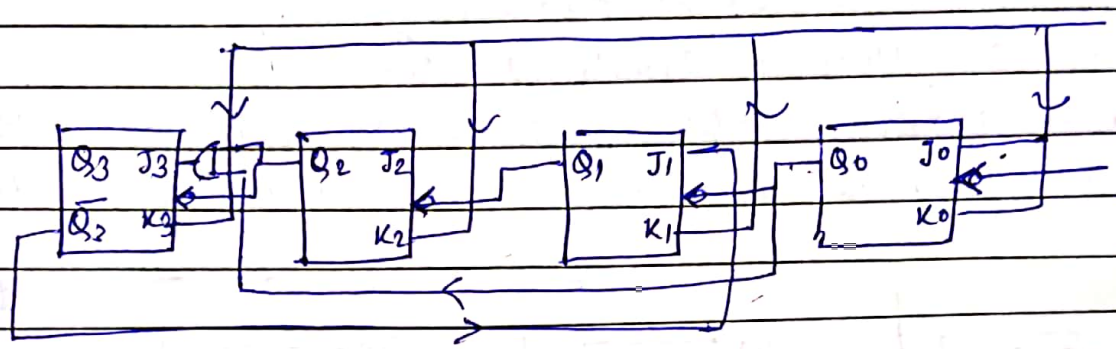
24-11-2022
Thursday

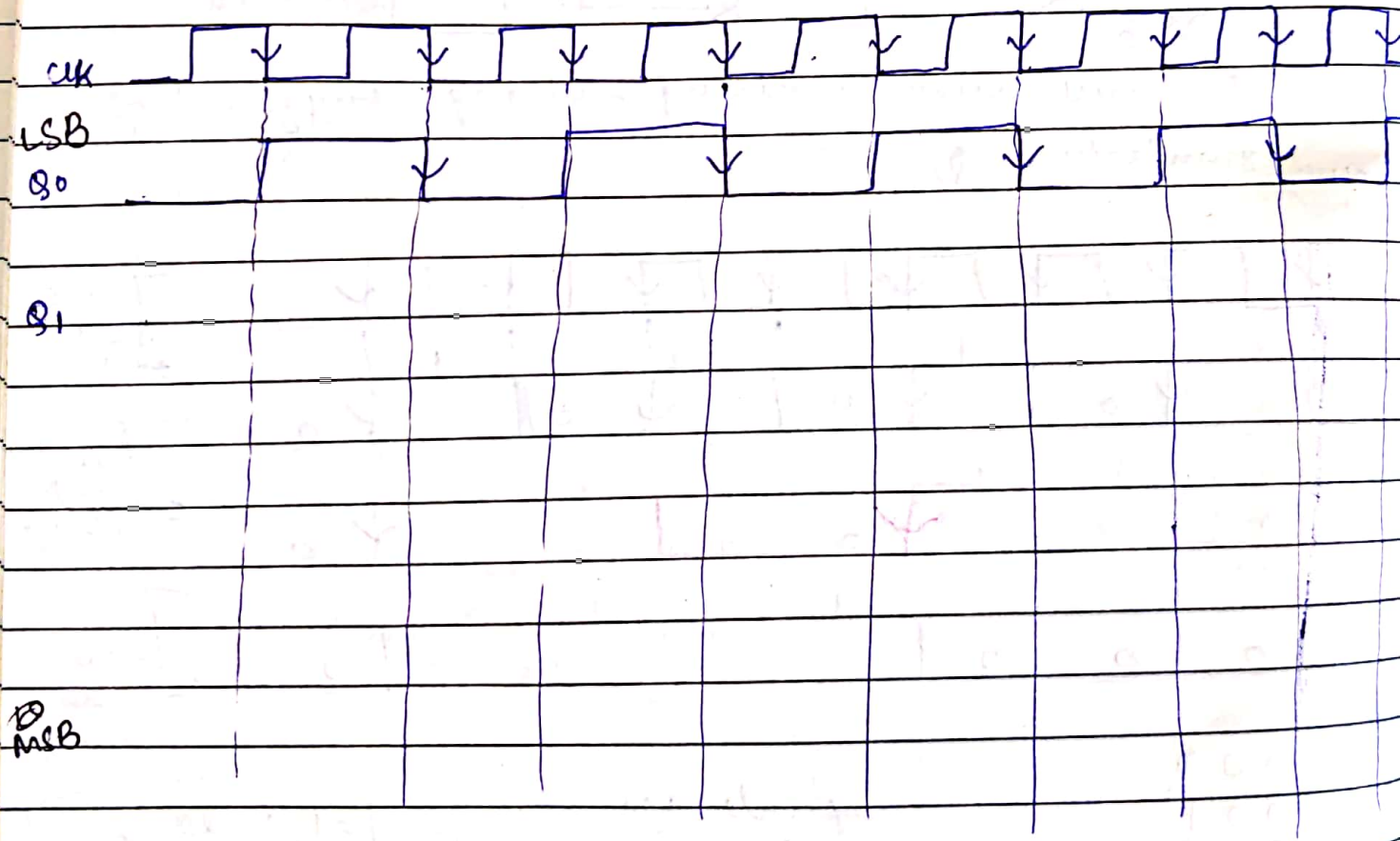
No designing
Only processing

Date: / / Page no:

(ii) BCD Ripple Counter :- It counts the sequence from 0 to 9

4 f/f s are needed.





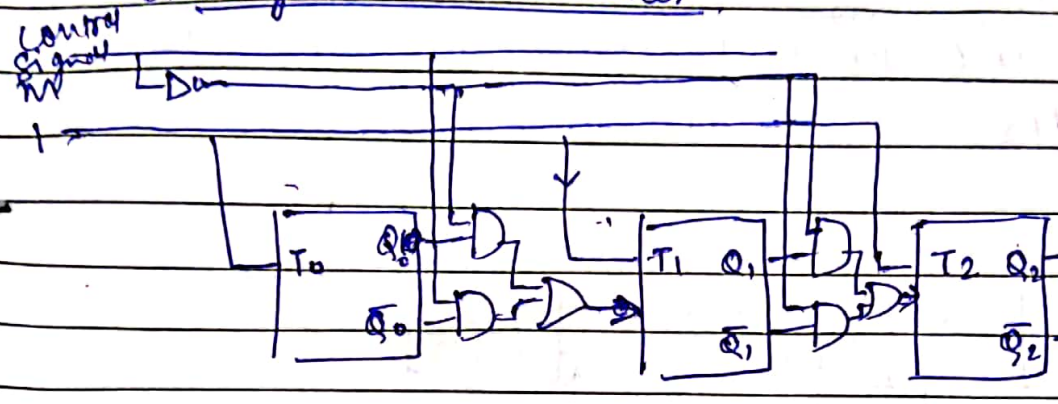
Scanned with CamScanner

Date: / / Page no:

M=0 up counting
M=1 down counting
Q clk
$\bar{Q}$ clk

IMP

Binary Down counter



M	Q	$\bar{Q}$	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

M \ $Q\bar{Q}$	00	01	11	10
0	0	0	1	1
1	0	1	1	0

$Y = \bar{M}Q + M\bar{Q} = M \oplus Q$

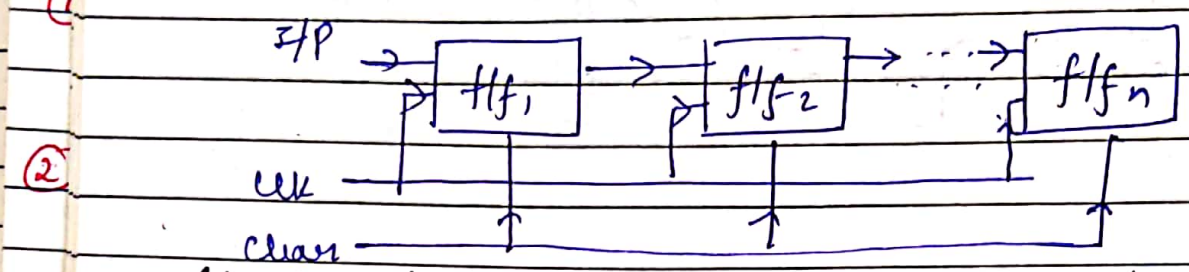
Scanned with CamScanner

No need to learn ckt.

Date: / / Page no:

REGISTER

It is a synchronous sequential ckt that stores n bit data when it has n flfs

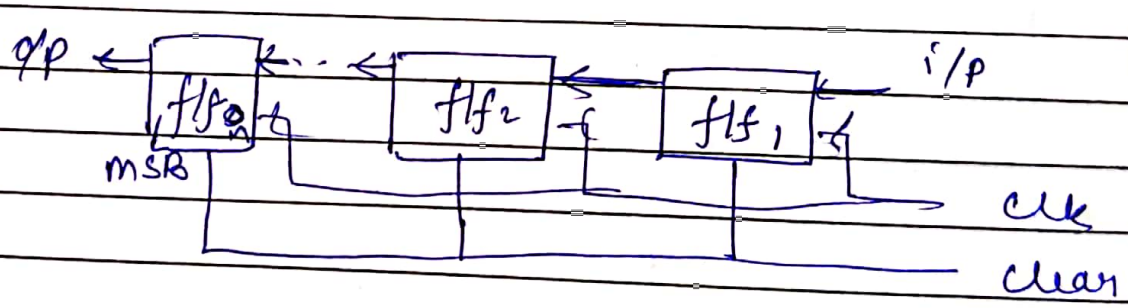


(to clear the content of register at any time)

There are 4 types of registers:

(i) Serial In serial out
(ii) Serial In Parallel out
(iii) Parallel In ~~Parallel~~ Serial Out
(iv) Parallel In Parallel out

(i) Serial In Serial Out : We can supply 1 bit i/p either from LSB or MSB and depending upon i/p we get o/p from other side MSB or LSB



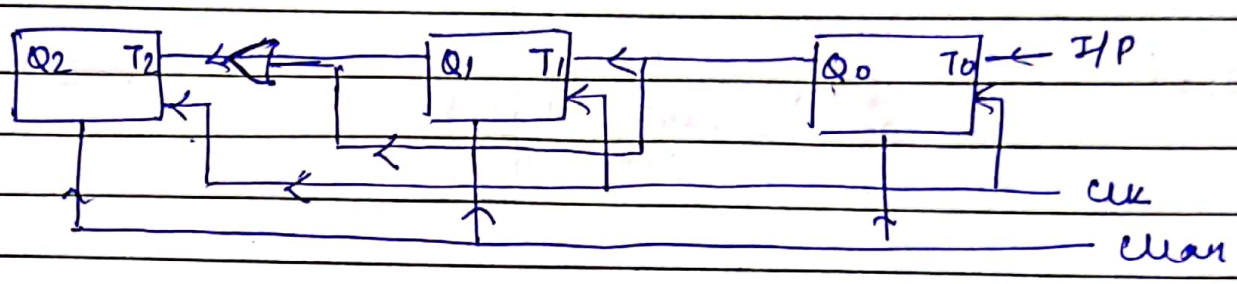
(ii) Serial In Parallel Out: We can supply 1 bit i/p either from LSB or MSB and can get o/p from all flfs parallelly.

Scanned with CamScanner

Date: / / Page no:

Q) Consider a the following SISO Register with T f/f with initial state 000

Sol: 3 ffs (initial state = 000).



(No ~~Time~~ need to process clk in case of synchronous)

The i/p terminal is connected to LSB and it supplies 1 bit per clock data 10101100 from MSB. Find the content after 8th clock

1 2 3 4 5 6 7 8
1 0 1 0 1 1 0 0
LSB MSB

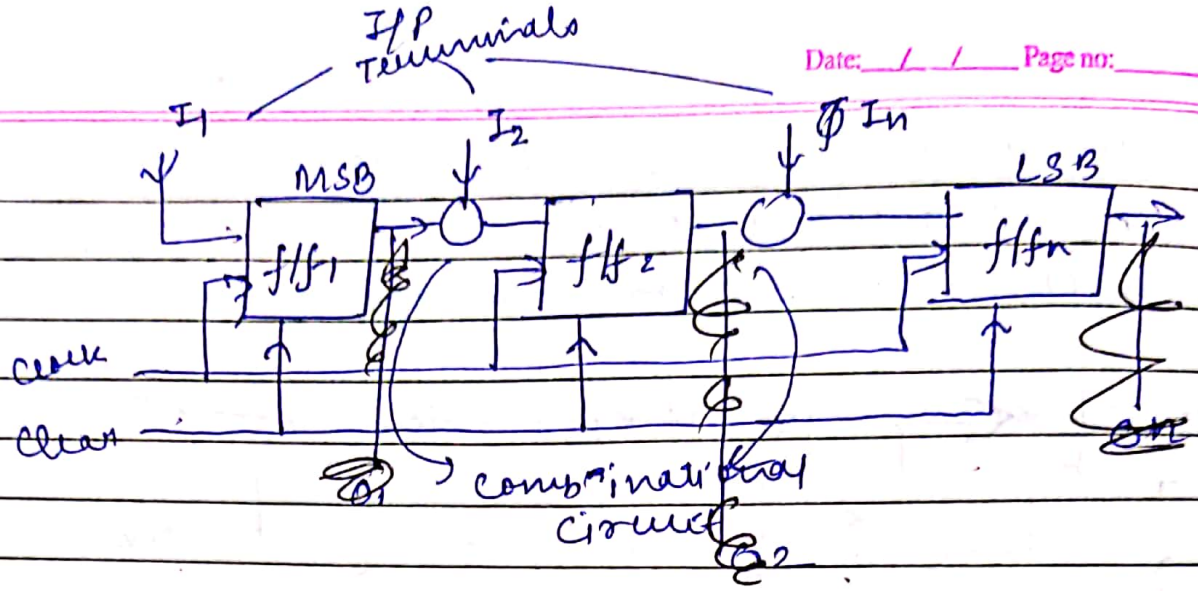
Sol:

Clk	I/p	T_2	T_1	T_0	Q_2	Q_1	Q_0
	$Q_1 \oplus Q_0$	Q_0	~~I/p~~ I/P				
0	–	–	–	–	0	0	0
1	1	0	0	1	0	0	1
2	0	1	1	0	1	1	1
3	1	0	0	1	1	0	0
4	0	0	0	0	1	0	0
5	1	0	0	1	1	0	1
6	1	1	1	1	0	1	0
7	0	1	0	0	1	1	0
8	0	1	0	0	0	1	0

→ Ans.

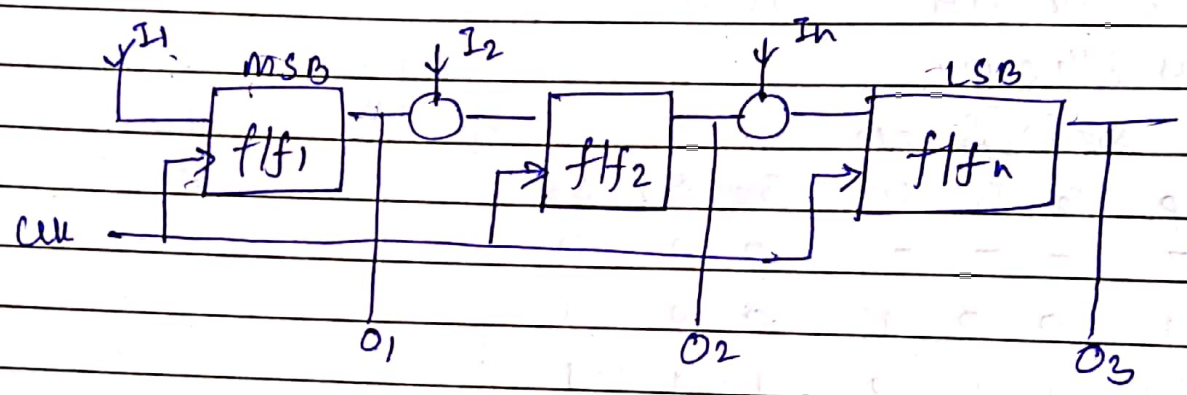
(iii) Parallel in ~~parallel~~ serial out: We can supply i/p to all the ffs parallelly and we can get 1 bit o/p from MSB or LSB

Scanned with CamScanner

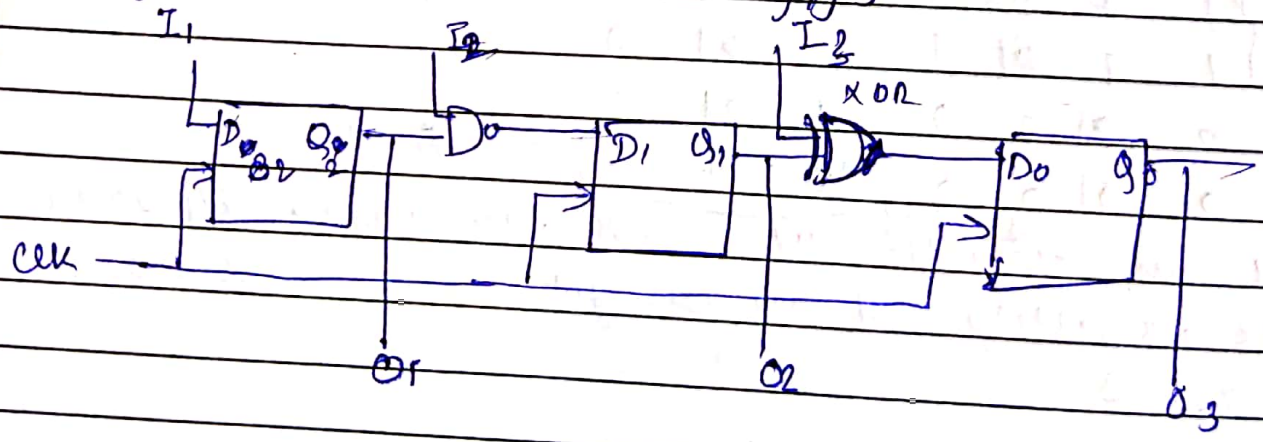


(4) Parallel in Parallel out :

We can supply i/p to all the ffs parallely and we can get o/p from all the ffs parallely.



Q) Consider the following parallel in parallel out register designed with D ff's



Scanned with CamScanner

28.11.2022
Monday.

If initial state not given then = 000

Input terminals I_1, I_2, I_3 supply 1 bit per clock from LSB. Data 101011, 101100, 101010 are supplied to i/p of terminals I_1, I_2 & I_3 respectively. Find the content of register after 6th clk.

Sol:

Clock	I_1	I_2	I_3	D_2	D_1	D_0	Q_2 Q_1 Q_0
0	–	–	–	0	0	0	
1	1	0	0	1	1	0	
2	1	0	1	1	1	0	
3	0	1	0	0	0	1	
4	1	1	1	1	1	1	
5	0	0	0	0	1	1	
6	1	1	1	1	1	0	

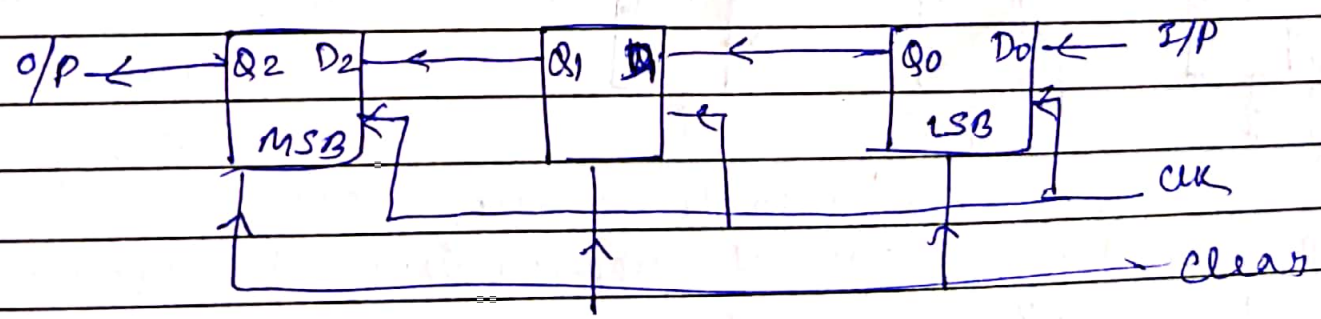
$D_2 = Q_2$
$D_1 = Q_1$
$D_0 = Q_0$

In this ques. we have previous state I_1 & present state I_1

Synchronous → Take previous only even present is there because Synchronous works only when all the i/ps ~~(of same)~~ are present at a time.

Shift Register: It is SISO register that shifts the content either left or right dirn.

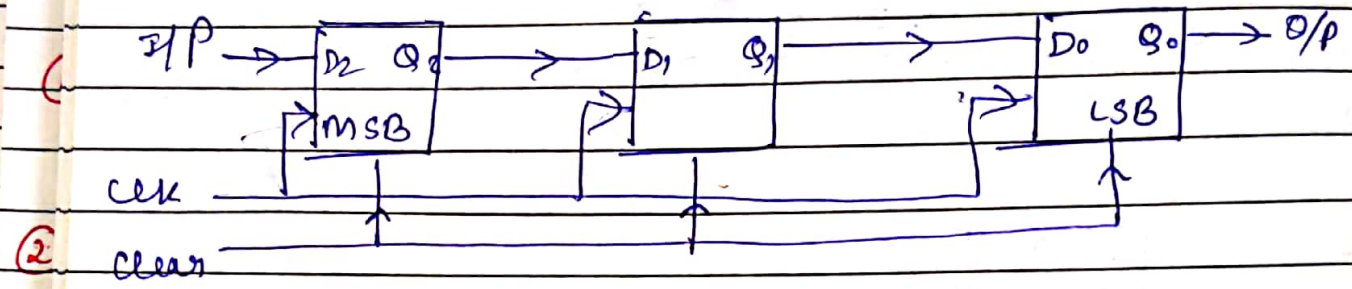
Left Shift Register: Content will be shifted in left dirn (i/p at LSB and o/p at MSB)



Scanned with CamScanner

Right Shift Register : content will be shifted in Right direction.

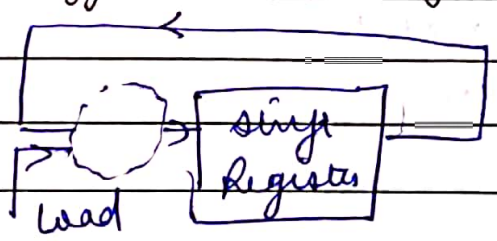
(i/p at MSB & o/p at LSB)



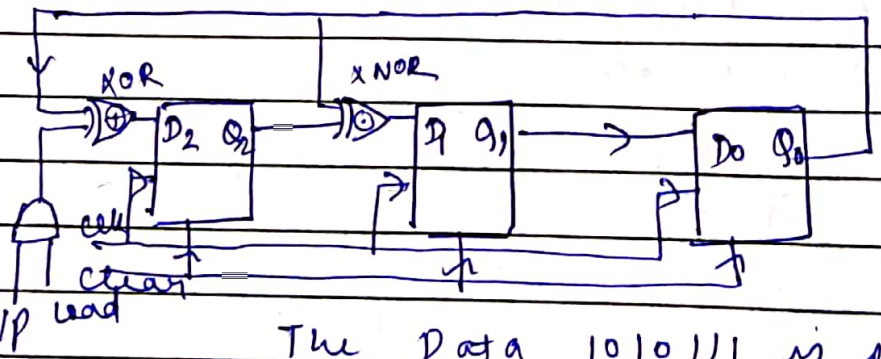
Ring Counter ; It is a special-type of shift register where o/p is connected to i/p.

Depending upon content loaded in shift register ring counter counts sequences.

We can also load new content in Shift Register inorder to get different counting sequences



Q) Consider the following Ring counter designed with D flfs and initial state 101



clk	Q_2	Q_1	Q_0
0	-	-	-
1	1	0	0
2	0	0	1
3	1	0	0

again repeats

∴ Load changes i/p to get new sequence

The Data 1010111 is supplied at i/p terminal. I/P terminal supplies data 1 bit per clock from LSB

Always put clk

Date: / / Page no:

The load terminal sets high at every 3rd clock in order to load new content otherwise low. Find the content of shift Register at the end of 7th clock.

Sol: Clk i/p. D_2 D_1 D_0 Q_2 Q_1 Q_0 $Q_0 \oplus$ (i/p · Load) $Q_0 \odot Q_2$ Q_1 1 0 1 0 1 1 1 ← LSB

Clk	i/p	Load	D_2	D_1	D_0	Q_2	Q_1	Q_0	
–	–	–	–	–	–	1	0	1	Initial state.
1	1	0	1	1	0	1	1	0	
2	1	0	0	0	1	0	0	1	
3	1	1	0	0	0	0	0	0	
4	0	0	0	1	0	0	1	0	←
5	1	0	0	1	1	0	1	1	
6	0	1	1	0	1	1	0	1	
7	1	0	1	1	0	1	1	0	

↳ Given every 3rd bit = 1

Content after 7th clock: 110

[Also after every 3 bit content sequence same i.e. might be = 1 every 3rd]

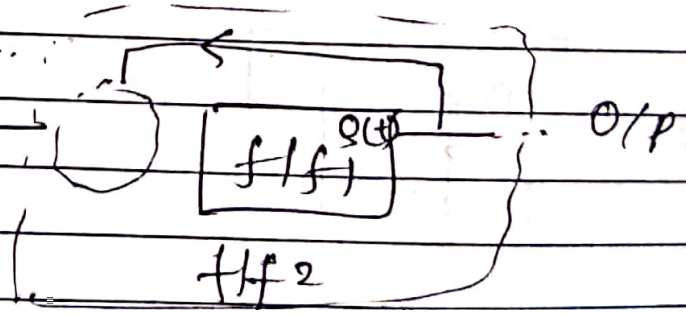
Flip Flop Conversion :-

Characteristic Table of final flf तो बनानी है

Excitation Table of flf initial फ्लिप-फ्लॉप बनानी है

JK to D flf.

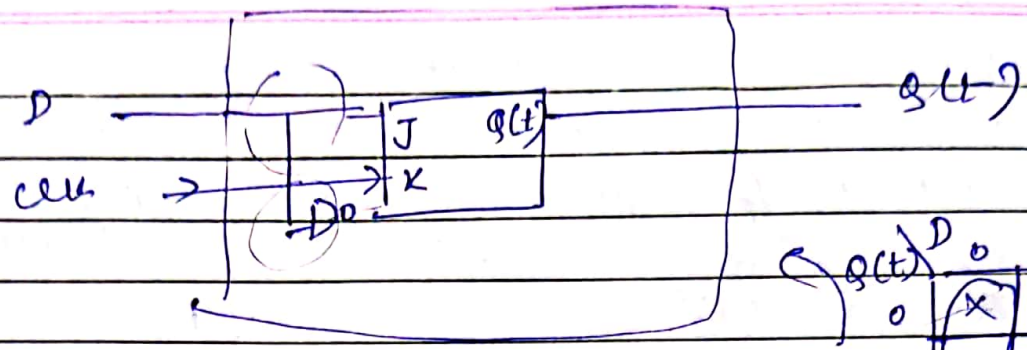
Excitation table — Characteristic table



Scanned with CamScanner

We can't convert JK, T, D to SR
[don't care

Date: / / Page no:



(1)

Q(t)	D	Q(t+1)	J	K
0	0	0	0	X
0	1	1	1	X
1	0	0	X	1
1	1	1	X	0

Q(t) \ D	0	1
0	X	X
1	1	

$K = \overline{D}$

Q(t) \ D	0	1
0		1
1	1	X

$J = D$

convert SR to JK

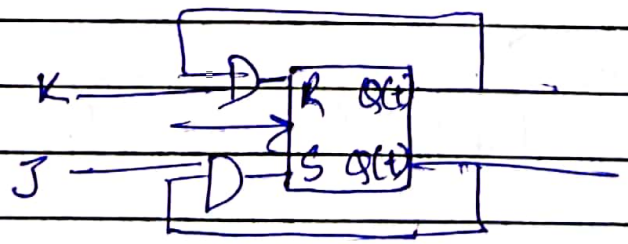
Q(t)	J	K	Q(t+1)	S	R	
0	0	0	0	0	X	(0)
0	0	1	0	0	X	(1)
0	1	0	1	1	0	(2)
0	1	1	1	1	0	(3)
1	0	0	1	X	0	(4)
1	0	1	0	0	1	(5)
1	1	0	1	X	0	(6)
1	1	1	0	0	1	(7)

Q(t) \ JK	00	01	11	10
0	0	1	1 3	1 2
1	X 4	5	7	X 6

$S = \overline{Q(t)} \cdot J$

Q(t) \ JK	00	01	11	10
0	X 0	X 1	3	2
1	4	1 5	1 7	6

$R = Q(t) \cdot K$



we can draw SR in opp. order
S above
R below

Minitest Monday (11:00 AM to 12:00 PM) Representation of real no.

Date: / / Page no:

Simplification of Boolean Function

(3) Tabulation Method (Quine McCluskey Method)

It is used to simplify higher order boolean f^n.

$f(A,B,C,D) = \Sigma(1,4,6,7,8,9,10,11,15)$

Step.1. Group the minterms based on the no. of 1's appear in their binary representation.

ex. '0' no. of 1's, '1' no. of 1's, ..., It will be level 1

		A	B	C	D		A	B	C	D
		level-1					level-2			
✓	1	0	0	0	1	(1,9)	–	0	0	1
✓	4	0	1	0	0	(4,6)	0	1	–	0
✓	8	1	0	0	0	✓(8,9)	1	0	0	–
✓	6	0	1	1	0	✓(8,10)	1	0	–	0
✓	9	1	0	0	1	(6,7)	0	1	1	–
✓	10	1	0	1	0	✓(9,11)	1	0	–	1
✓	7	0	1	1	1	✓(10,11)	1	0	1	–
✓	11	1	0	1	1	(7,15)	–	1	1	1
✓	15	1	1	1	1	(11,15)	1	–	1	1

Step.2 Find '1' bit difference ~~between~~ at any position betn minterms of adjacent groups. This will be level-2

Step.3 Put a check mark for all the minterms or pair of minterms of level-1 which are present at level 2

At each next level there will be atmost $n-1$ groups ~~terms~~

$^nC_1$ groups of previous level

Step.4 Again repeat step 2 and find '1' bit difference b/w pair of minterms of adjacent groups of level-2. It will be 2 bit difference from level-1

Check for – in two pos in two group

Scanned with CamScanner

Date: / / Page no:

Level-3

	A	B	C	D
(8, 9, 10, 11)	1	0	–	–
(8, 10, 9, 11)	1	0	–	–

(1)

Step 5 : Repeat Step 3 ~~for~~ pair of minterms which are present at Level-3.

(2) Step 6 : We cannot move ~~from~~ for level 4
(we can move for level (i+1) when level 'i' has more that one group)

Step 7 : ~~Now find~~ When we can't move for next level, then we find ~~to~~ minterms or pair of terms which are unchecked at any level ~~such to~~ and write corresponding terms.
Such terms are called Prime Implicants.

Step 8 : Prepare a table with row headers as Prime Implicants and columns headers as minterms.

Prime Implicants (Terms)		1	4	6	7	8	9	10	11	15
(1,9)	$\bar{B}\bar{C}D$	✓					✓			
(4,6)	$\bar{A}B\bar{D}$		✓	✓						
(6,7)	$\bar{A}BC$			✓	✓					
(7,15)	BCD				✓					✓
(11,15)	ACD								✓	✓
(8,9,10,11)	$A\bar{B}$					✓	✓	✓	✓	

Step 9 : Put a check mark for minterms that is part of Prime Implicants.

Scanned with CamScanner

~~Step 10.~~ Then find single check under the column corresp-
- onding prime implicants will be an essential prime implicants.

Step 10 Now write all the terms of essential prime implicants and check minterms which are covered by essential prime implicants.

↳ sometimes they cover sometimes don't

If ~~they~~ all are not covered then select proper prime implicant(s) that can cover remaining minterms

$f(A,B,C,D) = \bar{B}\bar{C}D + \bar{A}B\bar{D} + A\bar{B}$

↓ goto table

(7, 15) unmarked

$= \bar{B}\bar{C}D + \bar{A}B\bar{D} + A\bar{B} + BCD$

↳ (7, 15)

Verify using K map.

AB \ CD	00	01	11	10
00	0	1	3	2
01	1	5	1	1
11	12	13	1	14
10	1	1	1	1

$A\bar{B} + \bar{B}\bar{C}D + \bar{A}B\bar{D} + BCD$

Q) $f(A,B,C,D) = \Sigma(1, 2, 3, 4, 5, 7, 8, 10, 11, 15)$

0001 0010 0011 0100 0101 0111 1000 1010 1011 1111

Date: / / Page no.

Unit-1

	A	B	C	D
✓1	0	0	0	1
✓2	0	0	1	0
✓4	0	1	0	0
✓8	1	0	0	0
✓3	0	0	1	1
✓5	0	1	0	1
✓10	1	0	1	0
✓7	0	1	1	1
✓11	1	0	1	1
✓15	1	1	1	1

Unit-2

	A	B	C	D
✓(1,3)	0	0	–	1
✓(1,5)	0	–	0	1
✓(2,3)	0	0	1	–
✓(2,10)	–	0	1	0
(4,5)	0	1	0	–
(8,10)	1	0	–	0
✓(3,7)	0	–	1	1
✓(3,11)	–	0	1	1
✓(5,7)	0	1	–	1
✓(10,11)	1	0	1	–
✓(7,15)	–	1	1	1
✓(11,15)	1	–	1	1

Unit-3

	A	B	C	D
(1,3,5,7)	0	–	–	1
(1,5,3,7)	0	–	–	1
(2,3,10,11)	–	0	1	–
(2,10,3,11)	–	0	1	–
~~(4,5,~~				
(3,7,11,15)	–	–	1	1
(3,11,7,15)	–	–	1	1

Unit-4

	A	B	C	D
Nothing				

Prime Implicants		1	2	3	4	5	7	8	10	11	15
4,5	$\bar{A}B\bar{C}$				✓	✓					
8,10	$A\bar{B}\bar{D}$							✓	✓		
(1,3,5,7)	$\bar{A}D$	✓		✓		✓	✓				
~~(1,5,3,7)~~	~~$\bar{A}D$~~										
(2,3,10,11)	$\bar{B}C$		✓	✓					✓	✓	
(3,7,11,15)	CD			✓			✓			✓	✓

$\bar{A}B\bar{C} + A\bar{B}\bar{D} + \bar{A}D + \bar{B}C + CD$

All cover ∴ No Prime Implicants

$\bar{B}C + \bar{A}D + \bar{A}B\bar{C} + A\bar{B}\bar{D} + CD$

Correct

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

Scanned with CamScanner