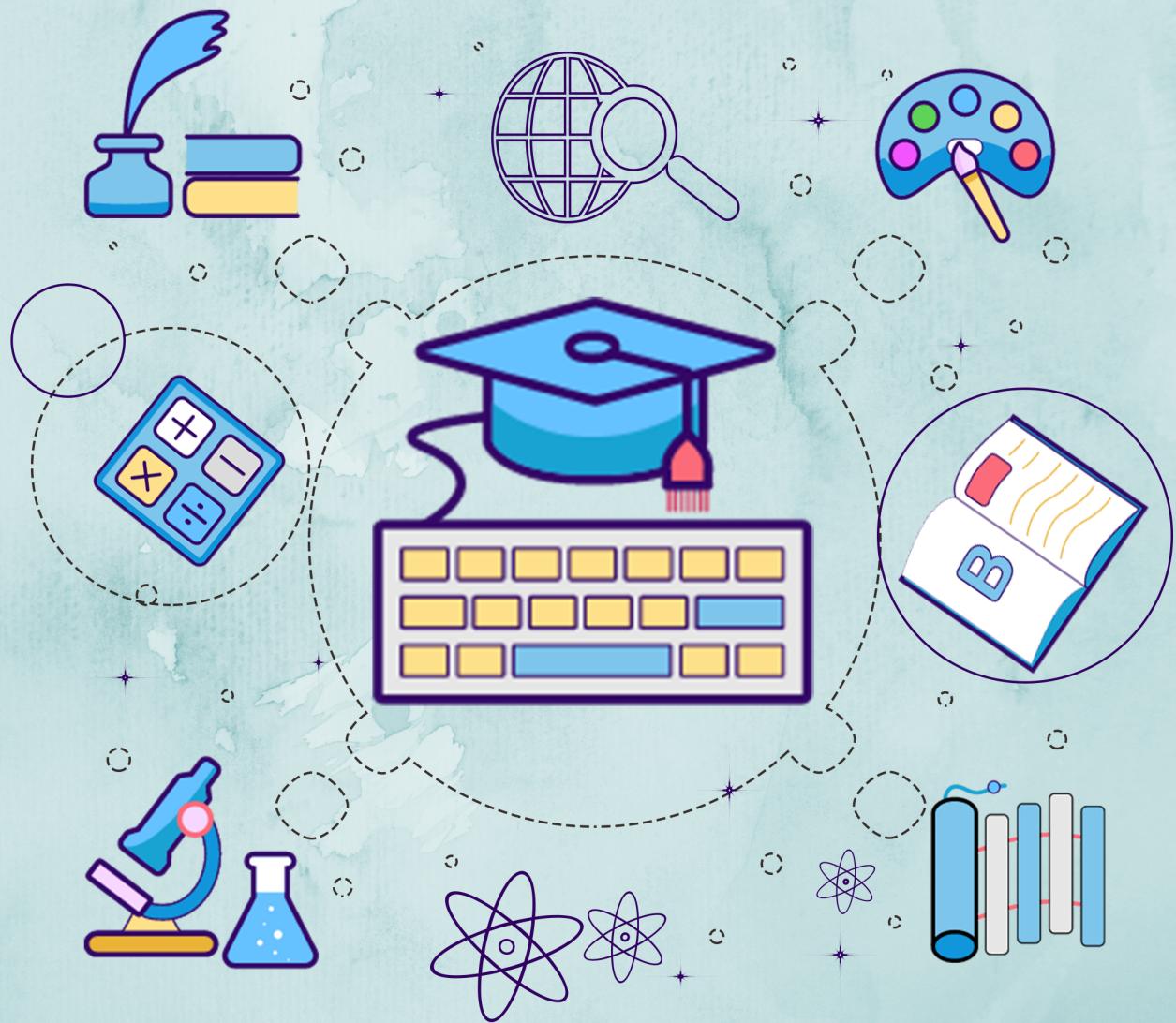


Kerala Notes



SYLLABUS | STUDY MATERIALS | TEXTBOOK

PDF | SOLVED QUESTION PAPERS



KTU STUDY MATERIALS

LOGIC SYSTEM DESIGN

CST 203

Module 4

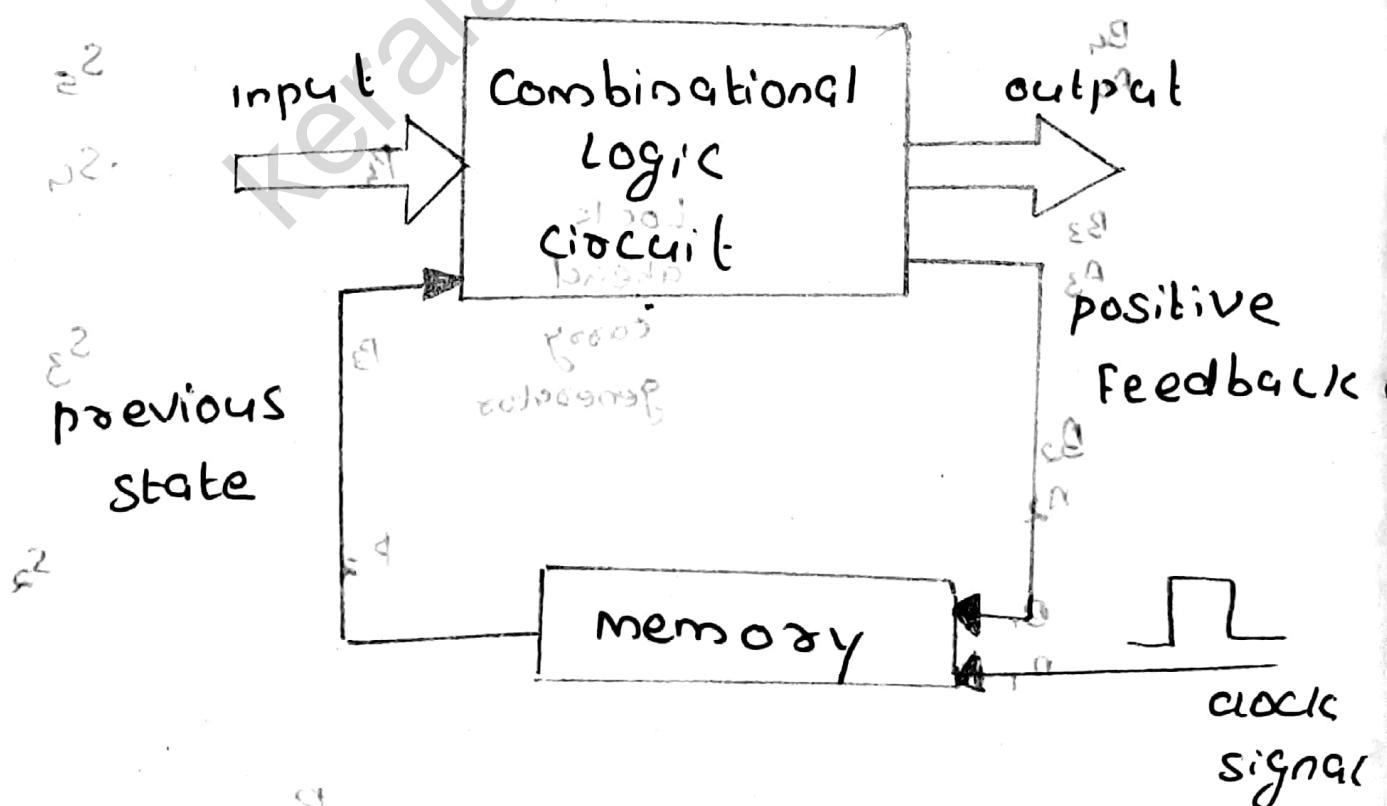
[Related Link :](#)

- KTU S3 STUDY MATERIALS
- KTU S3 NOTES
- KTU S3 SYLLABUS
- KTU S3 TEXTBOOK PDF
- KTU S3 PREVIOUS YEAR
SOLVED QUESTION PAPER

MODULE : 4

Sequential Circuit

- Sequential circuit is a combinational circuit with memory
- The o/p of sequential circuit depends upon the present i/p and past o/p
- Information stored in circuit represent present state
- the present state and present i/p will define the output and next state

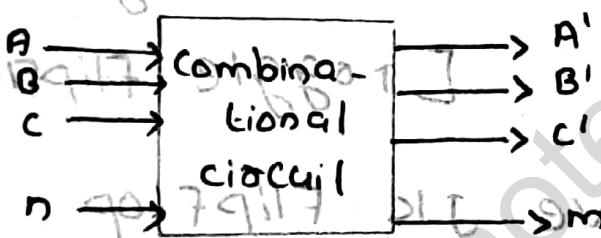


Output is depends on the present input
memory Element is Absent -

[No clock signal is Applied]

Applied

[Digital logic]



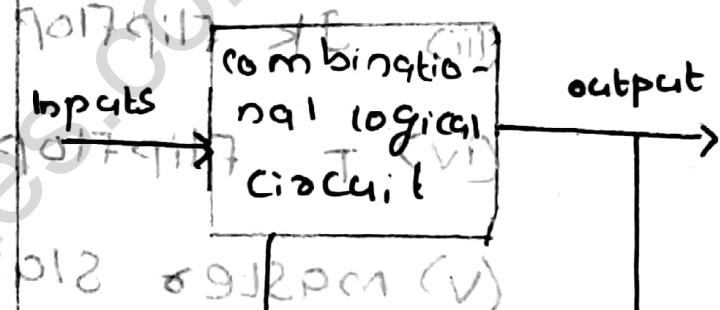
Eg: Half Adder,
full Adder,
Multiplexers

$X \oplus A$

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

output depends on present input and past output presence of memory Element

[clock signal is generated]



Eg: Flipflop, counters and Registers

$$i = s, o = r$$

$$s + p \oplus r \oplus s [o = s, i = p]$$

Types of Sequential Circuit

(i) Flip Flop

(ii) Counter

(iii) Shift Register

Types of Flip Flop

(i) SR flip flop

(ii) D flip flop

(iii) JK flip flop

(iv) T flip flop

(v) master slave JK flip flop

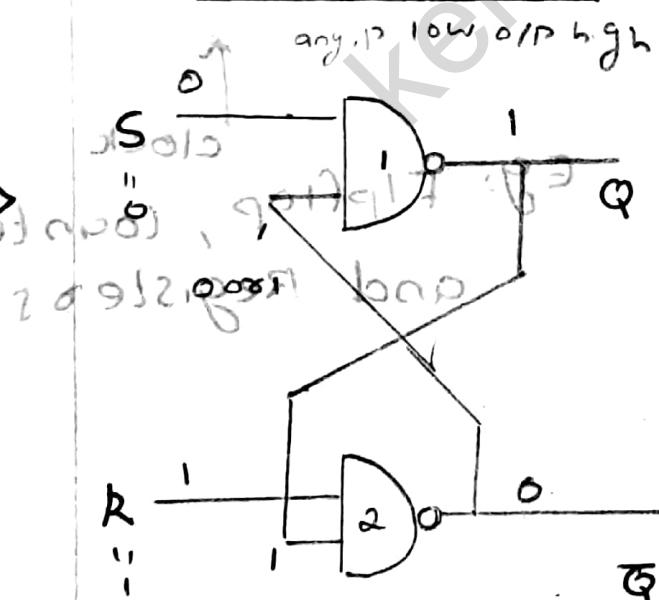
[set & reset flip flop]

[delay flip flop]

[Jack Kilby flip flop]

[Toggle flip flop]

SR Latch



$$S=0, R=1$$

$$Q=1 \quad \bar{Q}=0$$

SET state

NAND

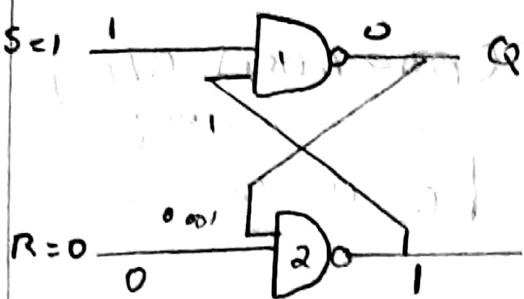
NOR

σ global truth : P

σ global 11, 2 -
σ configuration

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

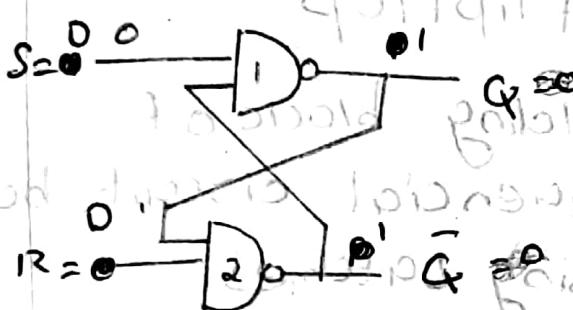
my one
input is
low
the o/p
will be
high

$S=1, R=0$


$$Q = 0 \quad \bar{Q} = 1$$

Reset state

$\Rightarrow S = 0, R = 0$

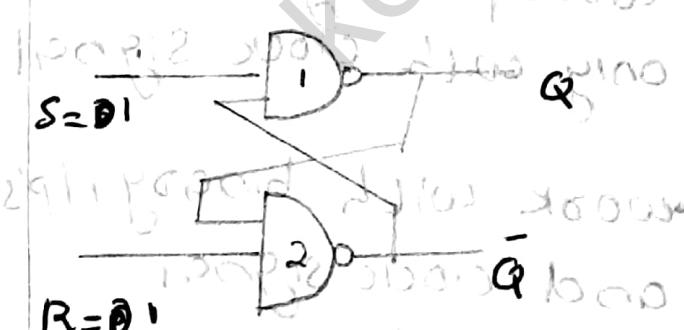


Invalid

because Q & \bar{Q} cannot be same since \bar{Q} is the complement of Q

loop please start all steps.

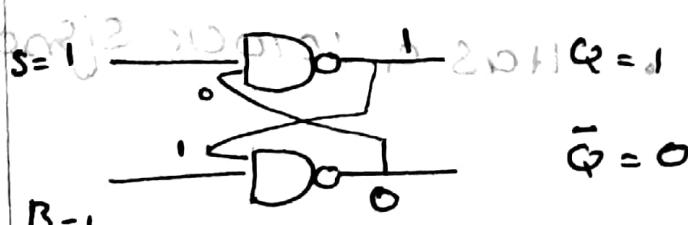
$\Rightarrow S = 1, R = 1$



Let us assume

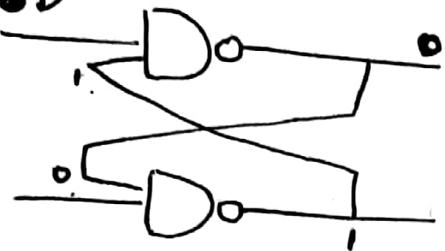
MEMORY state

(previous O/P)



Let us assume $Q = 0^0, \bar{Q} = 1$

$$Q = 0^0$$



$$Q = 1$$

$$Q = 0$$

(Previous O/P)

$$\boxed{Q = 0 \quad \bar{Q} = 1}$$

$$\bar{Q} = 1$$

Latch Vs Flip Flop

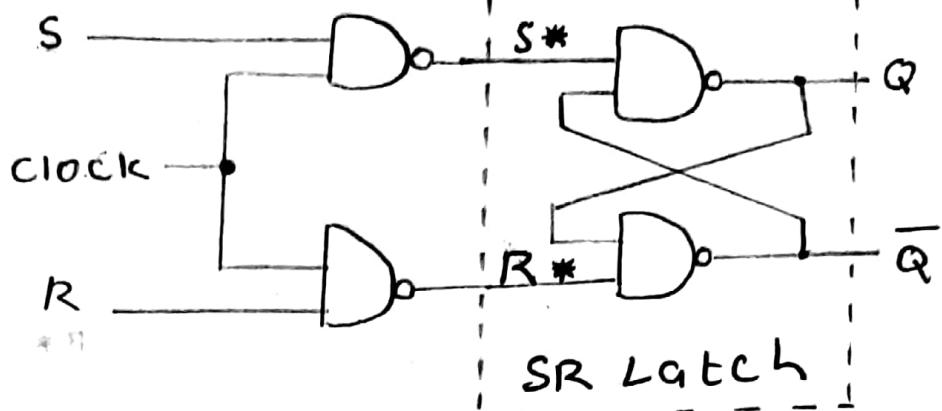
Latch

- Building blocks of sequential circuit, built using logic states
- check i/p continuously and changes o/p correspondingly
- work with only binary i/p's
- can not be used as a tester
- no clock signal

flip flop

- Building blocks of sequential circuit built using latches
- check i/p continuously and checks the o/p in a corresponding manner only with clock signal
- work with binary i/p's and clock signals
- can be used as a tester
- has a clock signal

SR FlipFlop

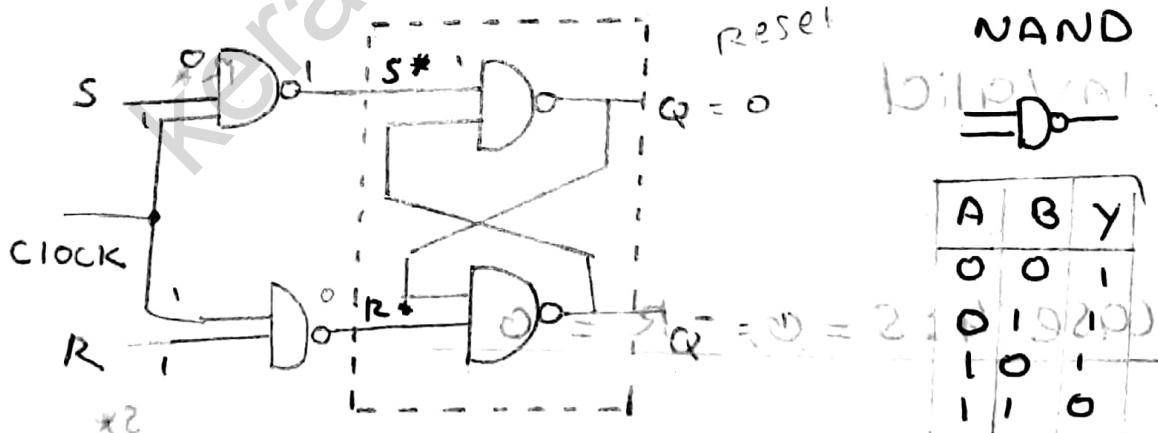


S*	R*	Q	\bar{Q}	State
0	0	X	X	Invalid
0	1	0	1	Set
1	0	0	1	Reset
1	1	Nc	Nc	Memory

Assum

$$\text{Clock} = 1$$

Case 1 $S=0, R=1$



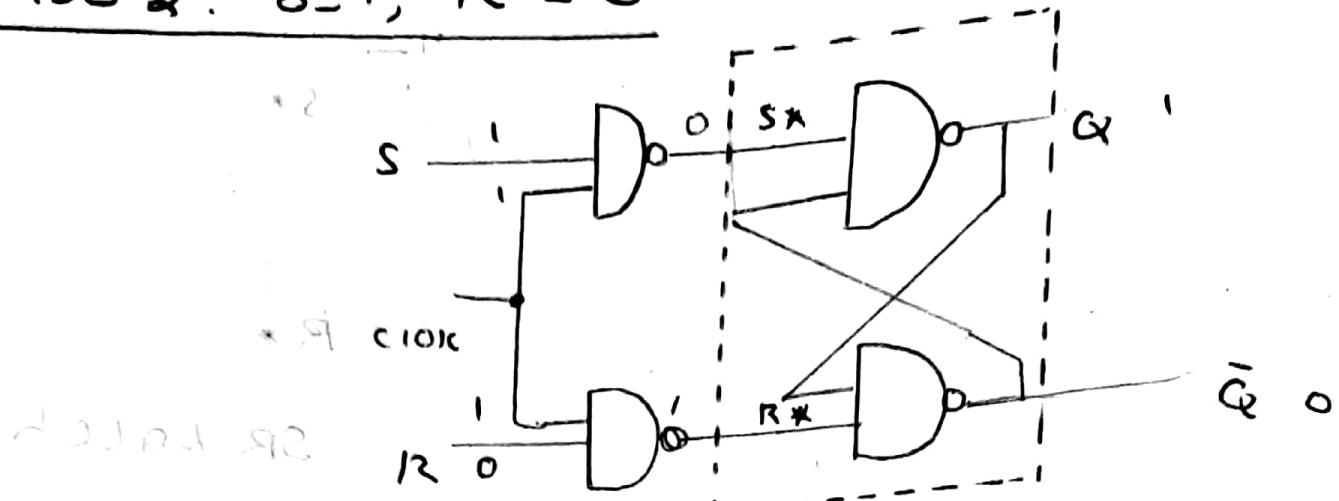
A	B	Y
0	0	1
0	1	0
1	0	1
1	1	0

$$Q=0, \bar{Q}=1$$

Reset goes on

when $S=0$ and $R=1$ Q will be 0 and \bar{Q} will be 1 so the state is Reset

Case 2: $S=1$, $R=0$



$$\left[Q = 1, \bar{Q} = 0 \right] \text{Set}$$

when $Q_e = 1$ and $\bar{Q} = 0$ The state will be set

case 3: $S = 1, R =$

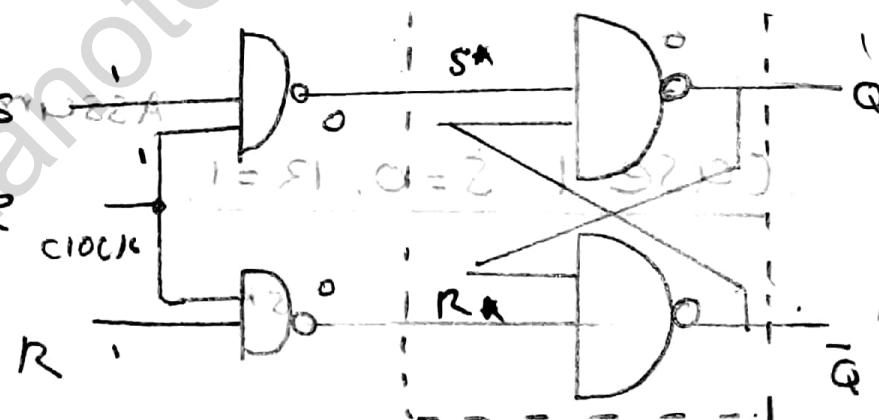
Here we get

$$C_0 = 1, \bar{C} = 1$$

but it not possible

Chancery

Invalid



$$\text{Case 4: } S = 0, \quad R = 0$$

$$S^* = 1, R^* = 1$$

No change

1000 samples

Memory

Memory - 2000 D = 2.0950
1923521802 90 02 1961 H.W. 5

no change

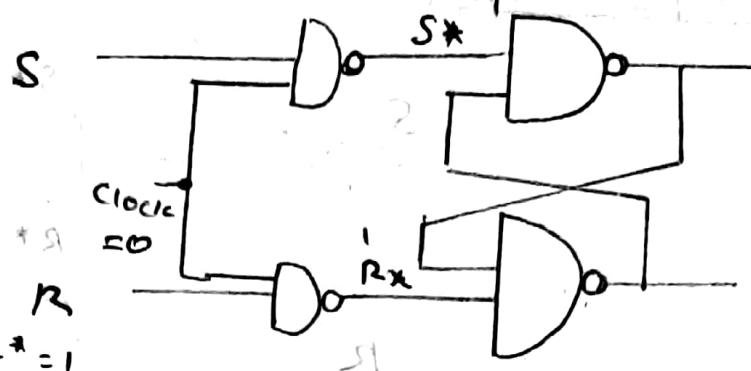
* Let us assume $CLOCK = 0$

whatever

value for

S & R

it will be $S^* = 1, R^* = 1$



if clock is 0 the S^* will be 1
 R^* will be 1

memory

IMP

CLOCK	S	R	Q	\bar{Q}	S^*	STATE
0	x	x	Nc	Nc		memory
1	0	0	Nc	Nc		memory
1	0	1	0	1	0	Reset
1	1	0	1	0	1	set
1	1	1	x	x		invalid

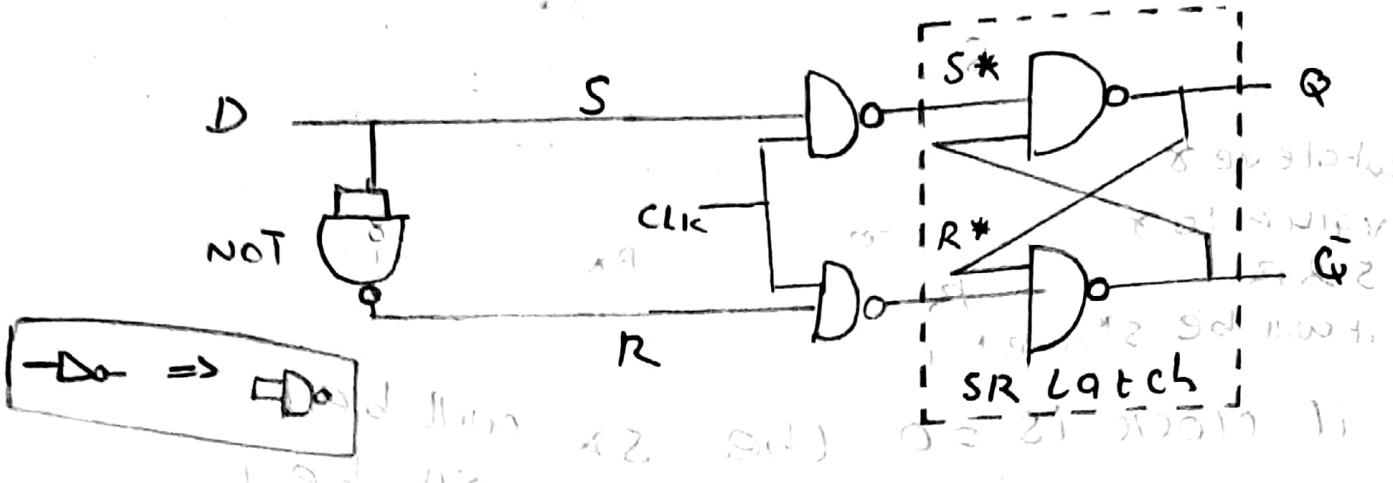
Actual truth table

CLOCK	S	R	Q_{n+1}	STATE
0	x	x	Q_n	memory
1	0	0	Q_n	memory
1	0	1	0	Reset
1	1	0	1	set
1	1	1	x	invalid

next book
 page 38

Q_{n+1}
 previous state Q_n

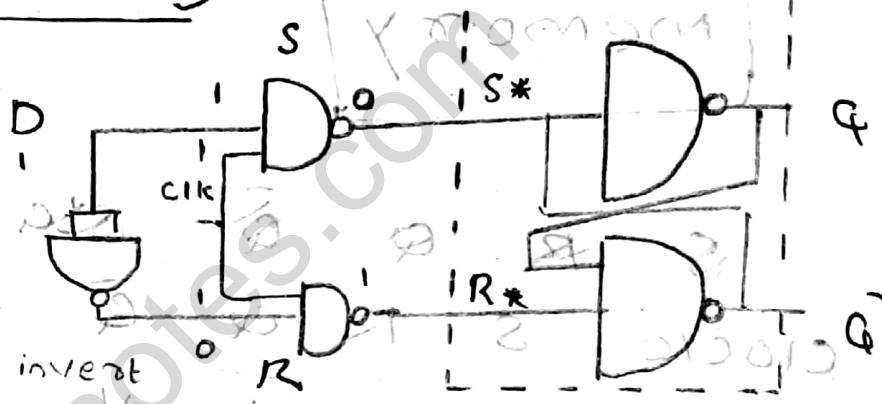
D-Flip Flop (Delay / Data flip flop)



case 1 : D=1 (CLK=1)

$$\begin{array}{l} Q = 1 \\ \bar{Q} = 0 \end{array}$$

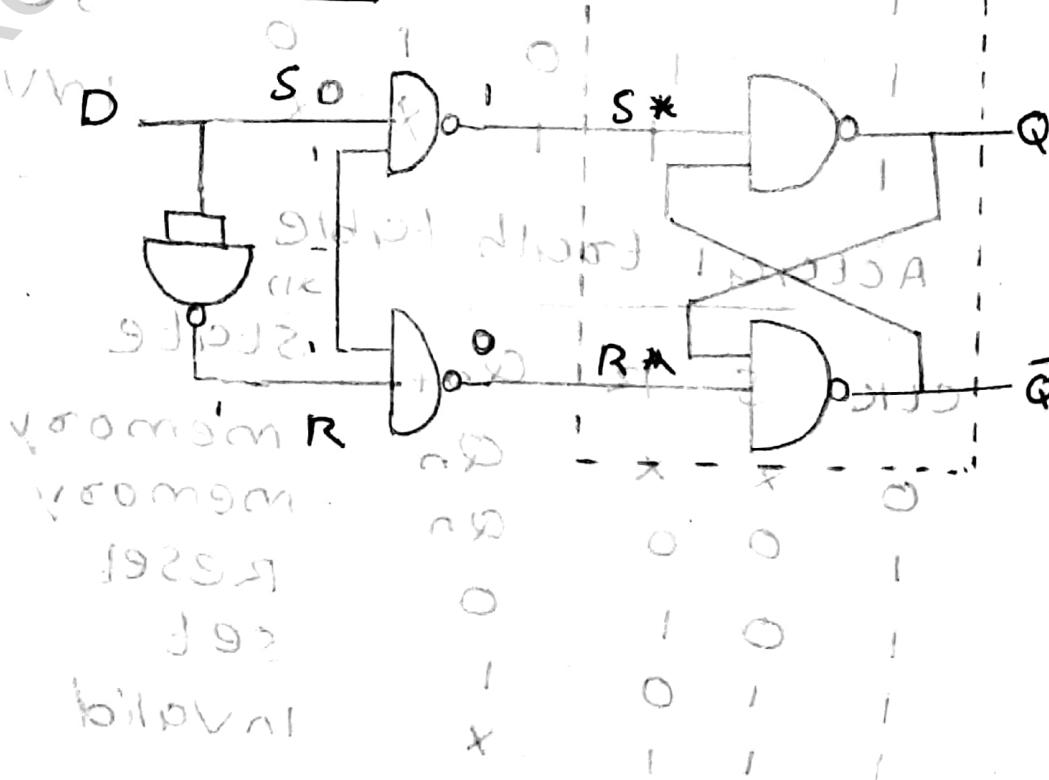
Set



case 2 : D=D C CLK=1

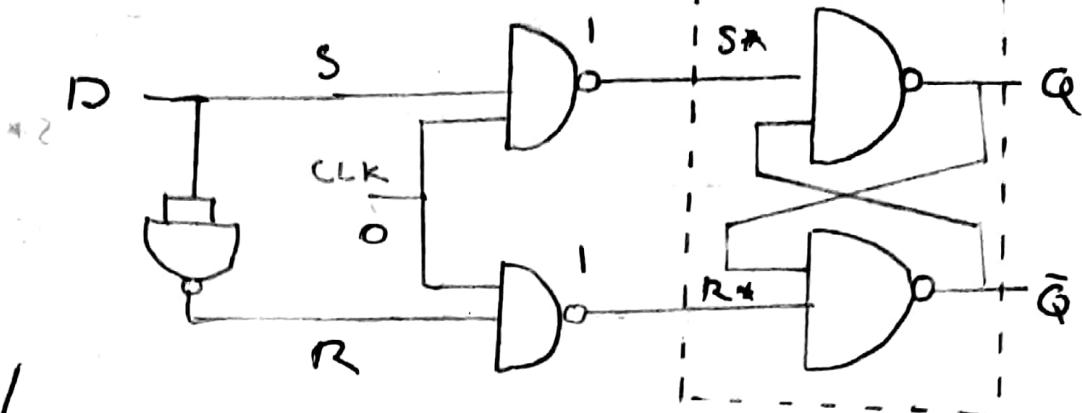
$$\begin{array}{l} Q = 0 \\ \bar{Q} = 1 \end{array}$$

Reset



Case 3: CLIC = 0

$D = 0/1$



memory/

no change

$S \cdot$	$R \cdot$	Q	\bar{Q}
0	0	X	X
0	1	1	0
1	0	0	1

CLIC	D	Q	State
0	X	Nc	Memory
0	1	0	
1	0	0	Reset
1	1	1	Set

q11

0 0 0

1 0 0

standard

1 1 0

0 0 1

1 0 1

0 1 1

1 1 1

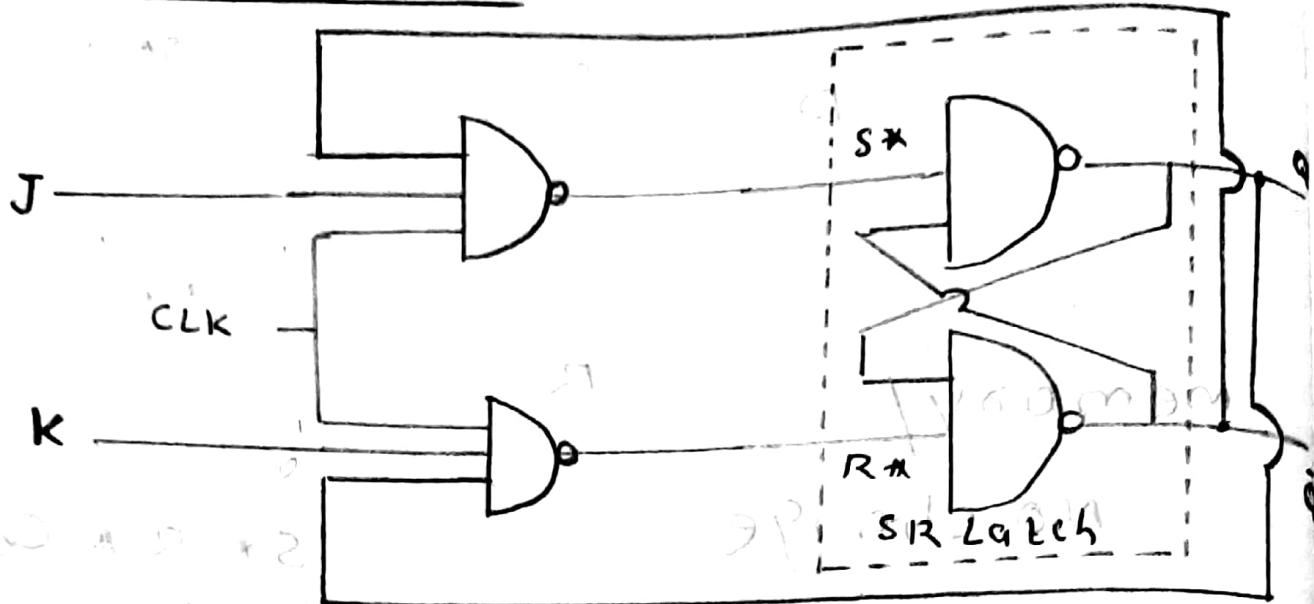


initial state

Actual truth table

CLK	D	Q_{n+1}	State
0	X	Q_n	Memory
1	0	0	Reset
1	1	1	Set

JK FLIPFLOP



The difference is that JK flip flop does not have invalid state, (when SSR is 1)

NAND



Three i/p

i/p	o/p	S*	R*	C ₁	C ₂	State
0 0 0	1	0	0	X	X	Invalid
0 0 1	1	0	1	1	0	Set
0 1 0	1	1	0	0	1	Reset
0 1 1	1	1	1	Nc	Nc	Memory
1 0 0	1					
1 0 1	1					
1 1 0	1					
1 1 1	0					

SR Latch

If any input is low output will be,

JS2SA

JS2

0 0 1

1 1 1

1. $J=0 \ K=0$

Assume $Q=0, \bar{Q}=1$

$Q=1, \bar{Q}=0$

$Q=0, \bar{Q}=1$ $J = 101 \rightarrow 1 - 111 = 0$ Memory
 $K = 100 \rightarrow 1 - 011 = 1$

Memory

$Q=1, \bar{Q}=0$

$J = 001 \rightarrow 1$ Memory
 $K = 101 \rightarrow 1$

2. $J=0 \ K=1$

Assume $Q=0, \bar{Q}=1$

$Q=1, \bar{Q}=0$

$Q=0, \bar{Q}=1$

$J = 101 \rightarrow 1$

$K = 110 \rightarrow 0$

$Q=0, \bar{Q}=1$

Reset

$S^* = 1, R^* = 0 \Rightarrow \text{Reset}$

$Q=1, \bar{Q}=0$

$J = 001 \rightarrow 1$
 $K = 110 \rightarrow 0$

3. $J=1, \ K=0$

$Q=0, \bar{Q}=1$

$Q=1, \bar{Q}=0$

$Q=1, \bar{Q}=0$ Set

Step

Step

4.

J=1, K=1 IMP

$$\underline{Q = 0, \bar{Q} = 1}$$

$$\begin{array}{l} J = 1 \ 1 \ 1 \rightarrow 0 \ 1 \ 0 \ 1 - 1 \\ K = 1 \ 1 \ 0 \rightarrow 1 \ 0 \ 0 \ 1 - 1 \end{array} \Rightarrow Q = 1, \bar{Q} = 0$$

we get

* previous output
complement

$$\underline{Q = 1 \ \bar{Q} = 0}$$

$$\underline{Q = 0, \bar{Q} = 1}$$

\Rightarrow Toggle condition

CLK	J	K	\bar{Q}	$\bar{\bar{Q}}$	state
1	0	0	Nc	Nc	memory
1	0	1	0	1	Reset
1	1	0	1	0	Set
1	1	1	T	T	Toggle
0	X	X	Nc	Nc	memory

Actual Truth table

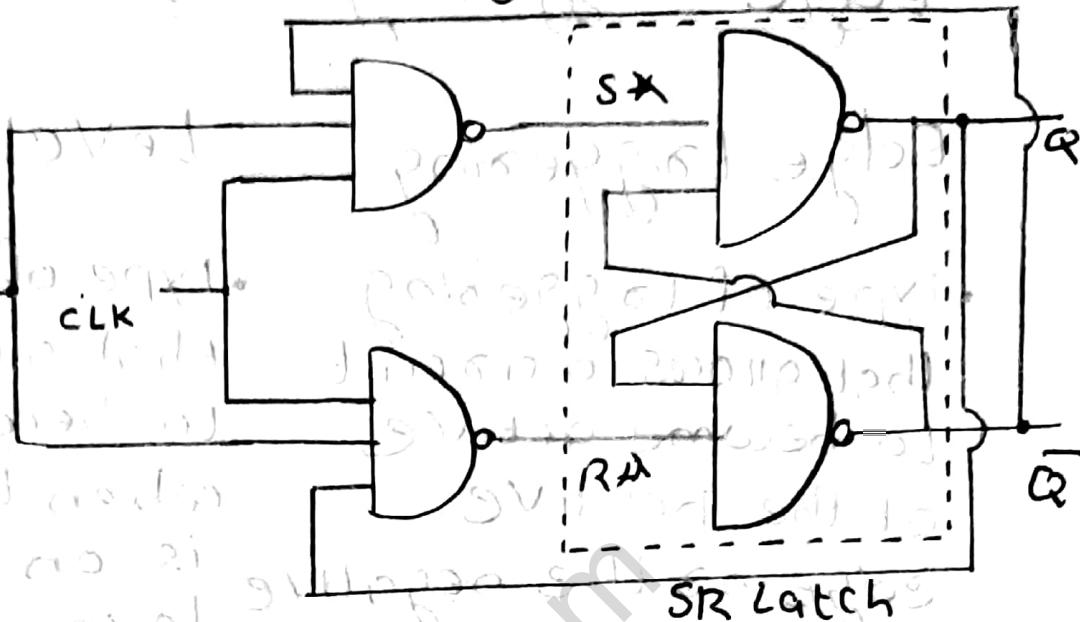
CLK	J	K	Q_{n+1}	state
0	X	X	Q_n	memory
1	0	0	Q_n	memory
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	\bar{Q}_n	Toggle

T-Flip Flop



KeralaNotes

- Connecting J and K together.



Truth Table

CLK	T	Q	\bar{Q}	State
0	X	Nc	Nc	Memory
1	0	Nc	Nc	Memory
1	1	T	T	Toggle

Actual Truth Table

CLK	T	Q_{n+1}	state
0	X	Q_n	memory
1	0	\bar{Q}_n	memory
1	1	\bar{Q}_n	Toggle

State Diagram of Flip Flop

last pg

Race around condition

Edge triggering Vs level triggering

Edge triggering

- Type of triggering that allows a circuit to become active at the positive edge or the negative edge of the clock signal

Level triggering

- Type of triggering that allows a circuit to become active when the clock pulse is on particular level

- An event occurs at the during the high voltage or low voltage level
- flipflop are edge triggered
- Latches are level triggered

Level triggering

• +ve

• -ve

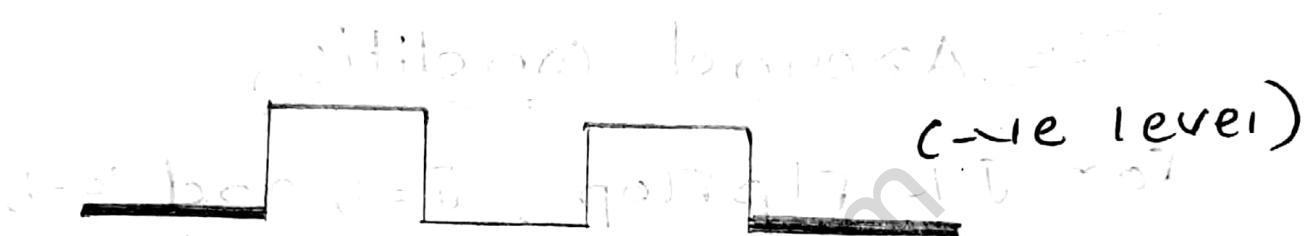
The clock signal is enabled only if active high levels are present

+ve level triggering



(+ve level) ↑

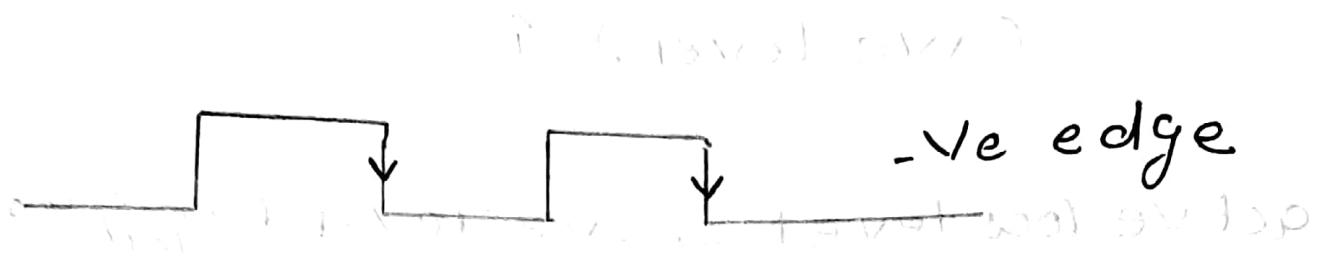
active low level → -ve level triggering



- Flip flop / clock is activated during high voltage level is called positive level triggering
- Flip flop / clock is activated during low level voltage level is called negative level triggering

Edge triggering

Clock is enabled during rising edge is called +ve edge triggering
 Clock is enabled during falling edge is called -ve edge triggering

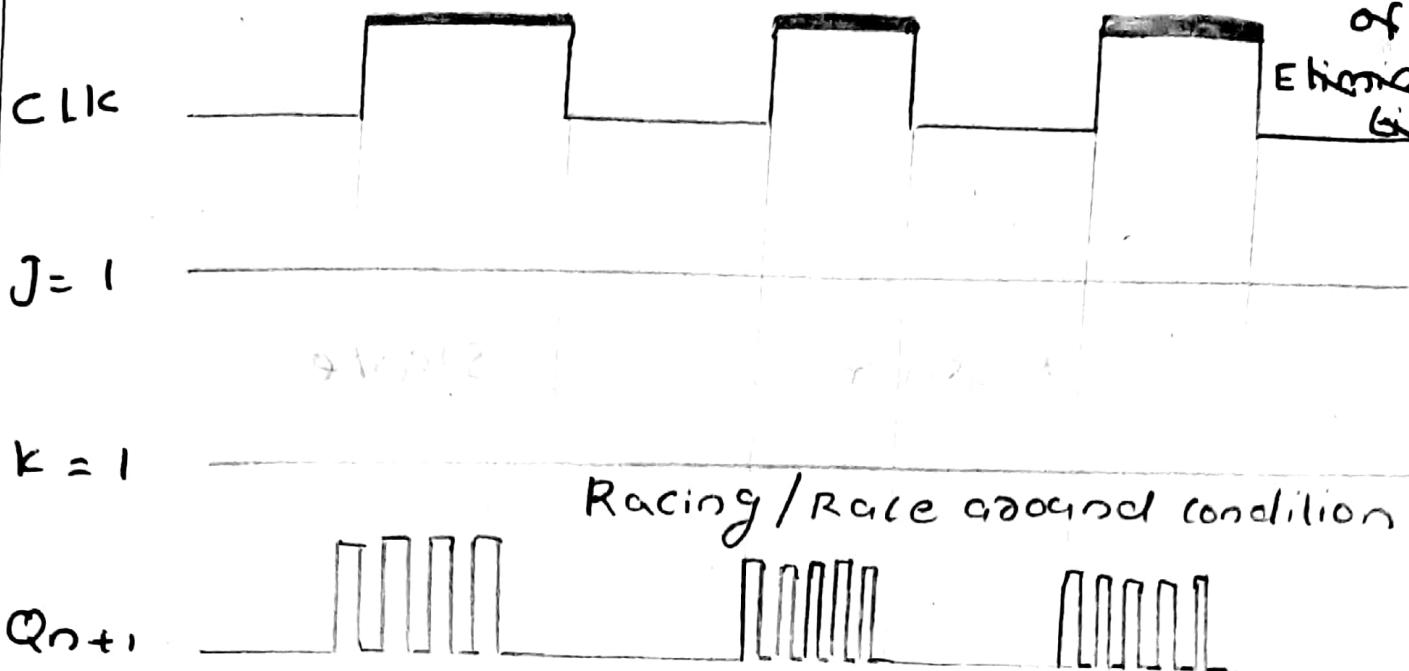


Race Around condition

For J K flipflop, J=1, and K=1, clock is too long then state of flipflop keeps on toggle which leads uncertainty in determining o/p state of flip flop.

This problem is called Race Around condition

J	K	Q	\bar{Q}	State
0	0	N _c	N _c	memory
0	1	0	1	Reset
1	0	1	0	Set
1	1	T	T	Toggle



Method of eliminating Race around condition

1. use of -ve edge triggering
2. master slave JK flipflop

Master slave JK flipflop

- It consists of clocked JK flip flop as master and clocked JK flip flop as slave
- Output of master flip flop is fed as an input of slave flip flop
- Clock is connected directly to master flip flop, but it is connected through inverter of slave flip flop

QUESTION
ANSWER

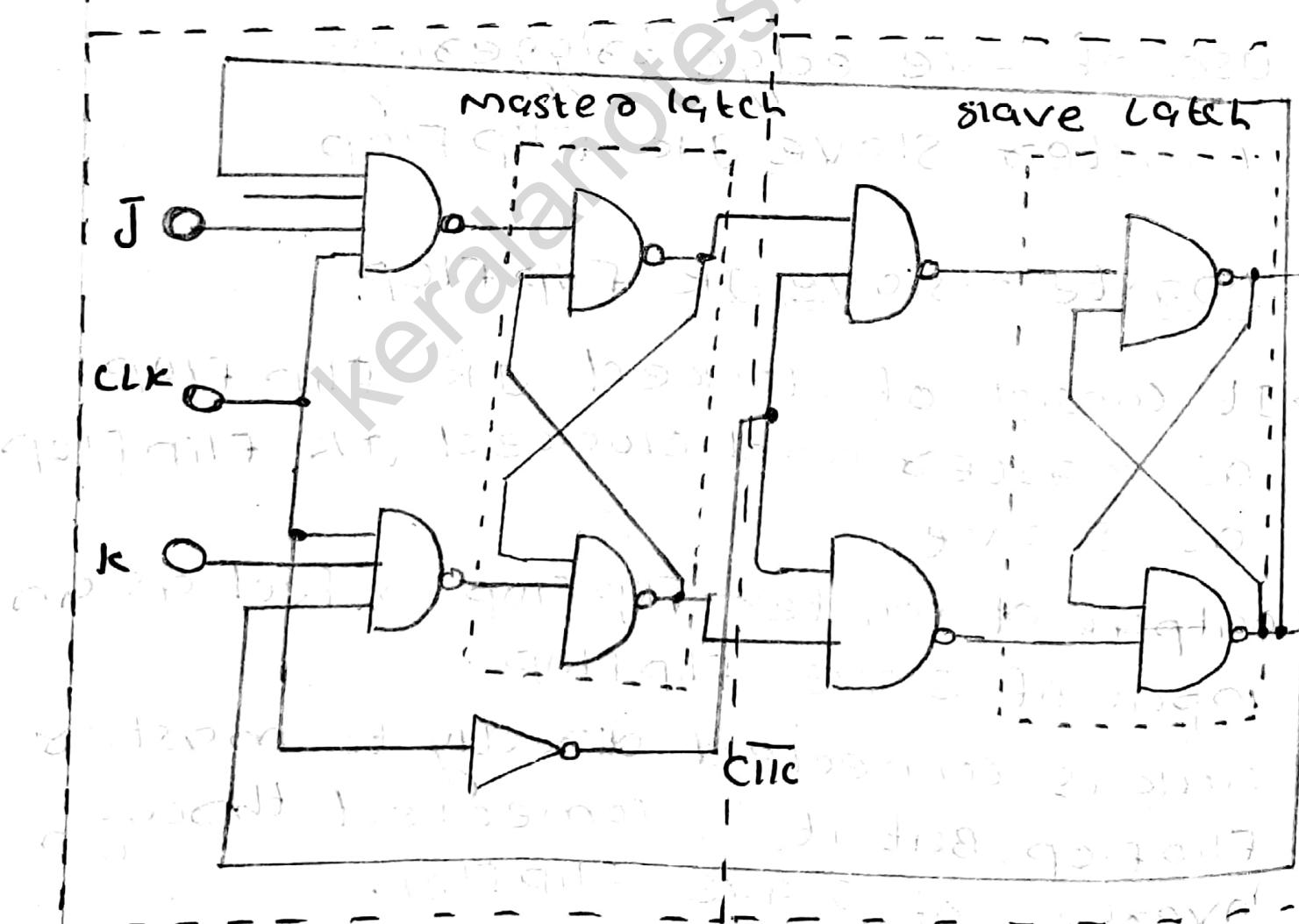
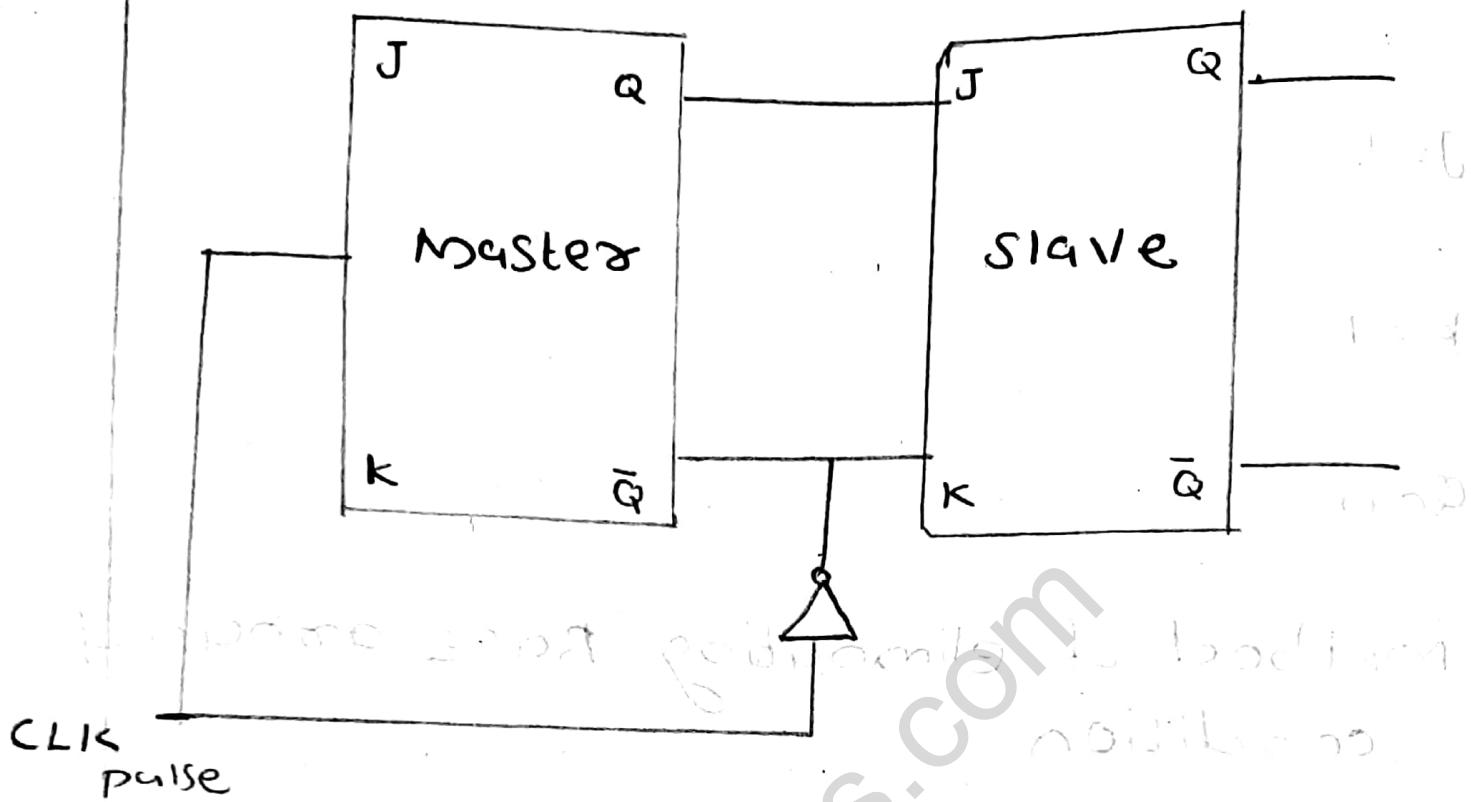
QUESTION ANSWER

ANSWER

ANSWER

ANSWER

ANSWER



SR Flipflop

Truth table only taking clk enable table

CLK	S	R	Q_{n+1}
0	X	X	Q_n (memory)
1	0	0	Q_n (memory)
1	0	1	0
1	1	0	1
1	1	1	Invalid

Characteristic

table

Note $S=0, R=0$, so it's memory
so its previous value is $Q_n = 0$

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

$Q_{n+1} \rightarrow$ Next state

$Q_n \rightarrow$ present state

Excitation table

Q_n	Q_{n+1}	S	R
0	0	0	X (since R is changing)
0	1	1	X 0
1	0	0	1
1	1	X	0

(since R is changing)

For calculating
characteristic table compare
SR of characteristic with

SR of characteristic with
SR of characteristic with
SR of characteristic with

in excitation to get
SR of characteristic

a and b with characteristic

characteristic
characteristic

Characteristic Equation

Using Kmap

and simplifying

Characteristic

Q_n	$S\bar{R}$	$\bar{S}R$	SR	$\bar{S}\bar{R}$
0	00	01	11	10
1	11	00	01	10

$$Q_{n+1} = S + Q_n \bar{R}$$

D flipflop

Truth table

CLK	D	Q_{n+1}
0	X	Q_n
1	0	0
1	1	1

Characteristic table

CLK = 1 [Always enable]

Q_n	D	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

if D=0
it will
be
 Q_{n+1}

Excitation table

Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Characteristic equation

$$\bar{Q}_n D + Q_n D$$

$$D(Q_n + \bar{Q}_n)$$

$$\Rightarrow Q_{n+1} = D$$

J. K flipflop

Truth table

CLK	J	K	Q_{n+1}
0	X	X	Q_n
1	0	0	Q_n (memory)
1	0	1	0
1	1	0	1
1	1	1	$\overline{Q_n}$

Toggle

Characteristic table

Q_n	J	K	Q_{n+1}
0	0	0	0 (previous value $Q_n = 0$)
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Characteristic equation

Q_n	JK			
	$\bar{J} \bar{K}$	$\bar{J} K$	$J \bar{K}$	$J \bar{K}$
00	0	0	1	1
01	1	0	0	1

$$\overline{Q_n} J + Q_n \bar{K}$$

$$Q_{n+1} = \overline{Q_n} J + Q_n \bar{K}$$

Excitation Table

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

T flipFlop

Truth table

CLK	T	Q_{n+1}
0	X	Q_n
1	0	Q_n
1	1	\bar{Q}_n

Characteristic Table

Q_n	T	Q_{n+1}
0	0	0
0	1	Q_1
1	0	1
1	1	0

Equivalent to XOR gate

Characteristic Equation

$$Q_{n+1} = Q_n \oplus T$$

$$(Q_n \bar{T} + \bar{Q}_n T)$$

Excitation table

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	⊕ 1
1	1	0

Binary counters

Counters

Counter is a sequential circuit which is used for digital circuit which is used for counting pulses is known as counter. Counting is the widest application of counter is the group of flip-flops with a clock signal applied.

Counters are of two types:

- (i) Asynchronous Counter / Ripple Counter
- (ii) Synchronous Counter

Synchronous Counter

KeralaNotes Asynchronous Counter

- In synchronous counter all flipflops are triggered with same clock simultaneously.
- Synchronous counter is faster than asynchronous counter in operation.
- Does not produce any decoding errors.
- Also known as parallel counter.
- Designing as well as implementation are complex due to increasing the no. of states.
- Will operate in any desired count sequence.
- Propagation delay is less.
- In asynchronous counter, different flipflop are triggered with different clock, not simultaneously.
- Slower than synchronous counter in operation.
- Produces decoding errors.
- Also known as serial counter.
- Designing as well as implementation is very easy.
- Operate only in fixed count sequence (up/down).
- Propagation delay is high.

Asynchronous Counter

IF the flip-flop do not receive the same clock signal, then that counter is called as Asynchronous Counter.

The output clock is applied as clock signal only to first flip-flop. The remaining flip-flop receive the clock signal from output of its previous stage flip-flop.

Hence, the output of all flip-flop counter in which output of first flip-flop act as a clock of next flip-flop

- (i) UP Counter
- (ii) Down Counter
- (iii) Modulo Counter

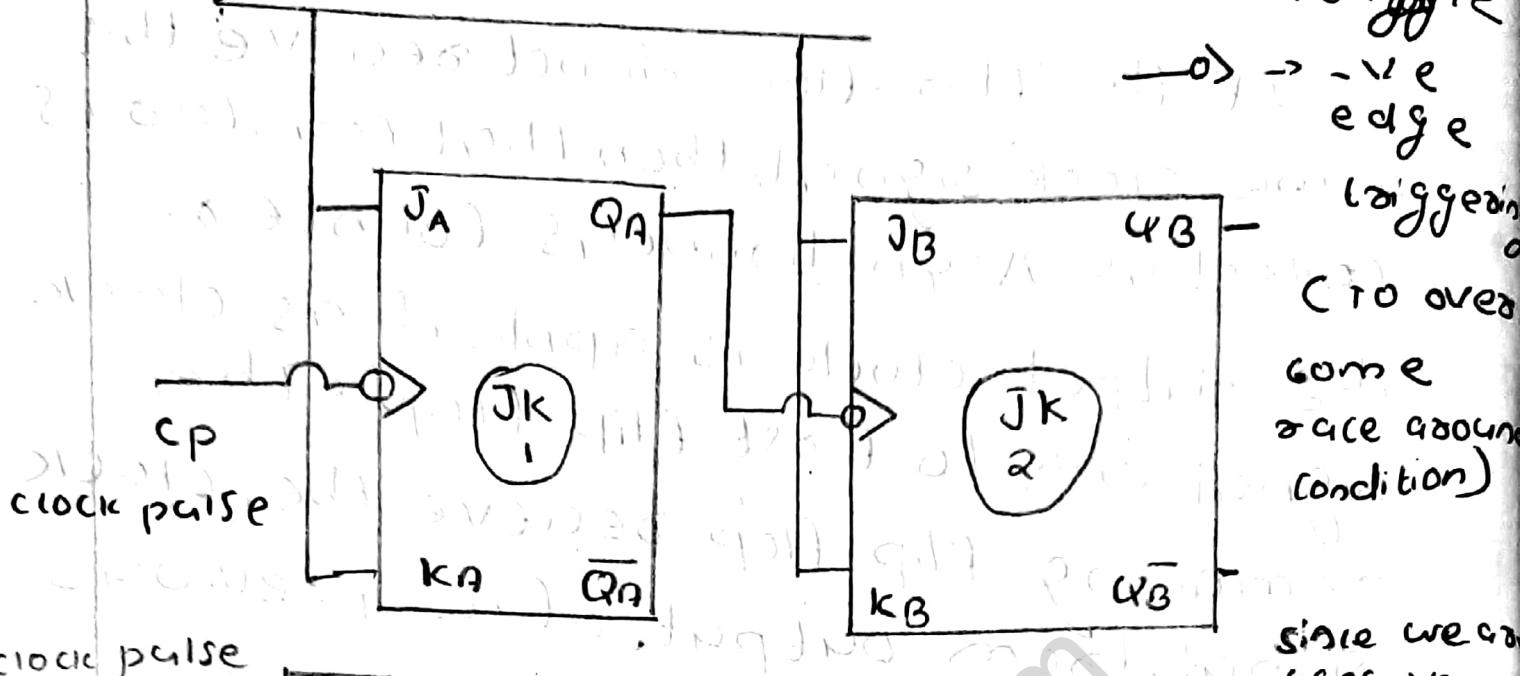
2 bit Asynchronous up counter

Initial condition
Q1 Q0 = 0 0

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

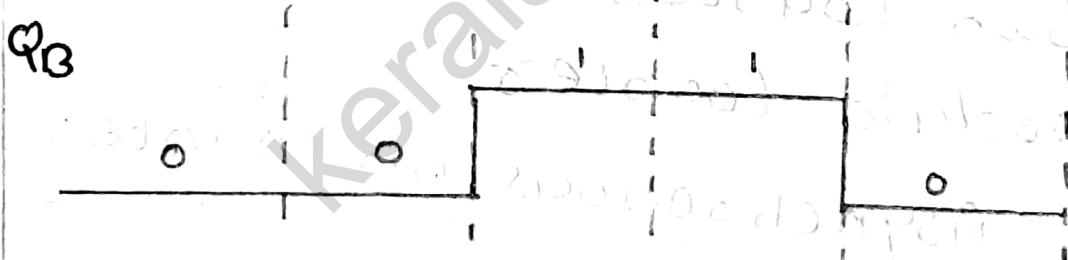
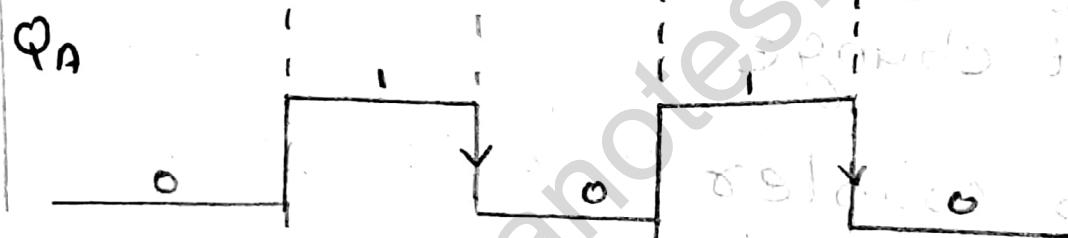
High

If Q_A is set it will be
down counter



$\rightarrow 0 \rightarrow -ve$
edge
(triggered
C to over
come
race around
condition)

single we are
using -ve
edge so
clock will
be at
falling
edge



CP	Q_B	Q_A	
0	0	0	0
1	0	1	1
2	1	0	2
3	1	1	3
4			

4th, clock pulse will
be 1st clock pulse

3 bit asynchronous CP

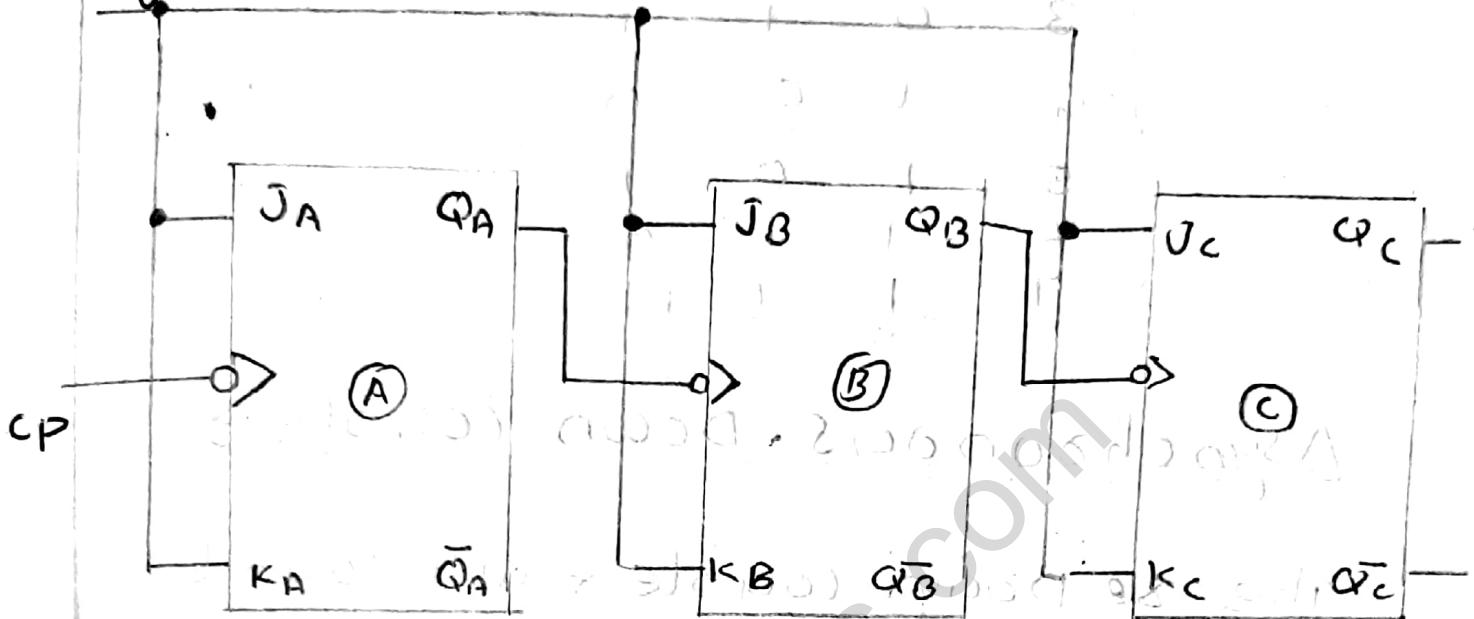
→ 3 J1C flip flop

Timing

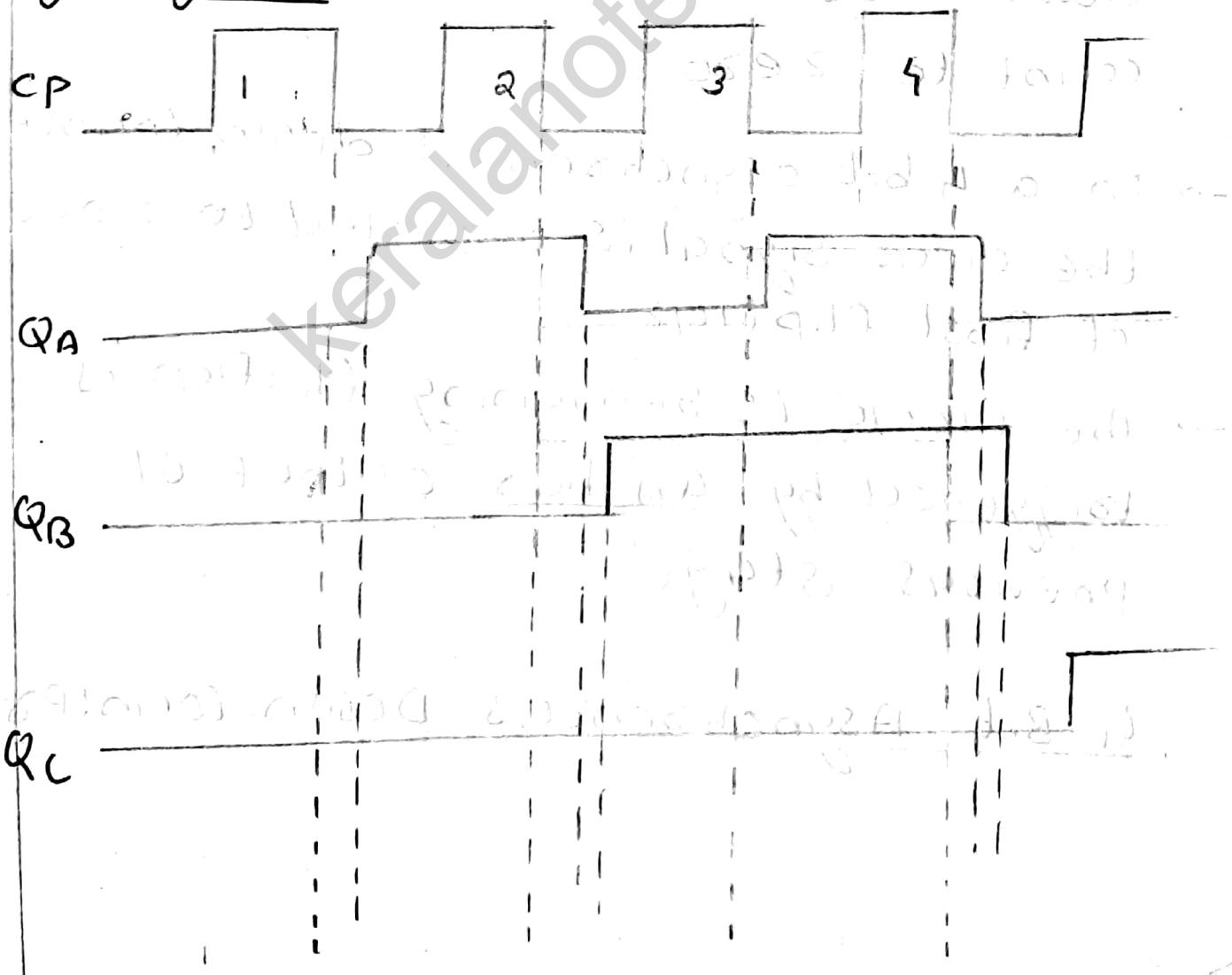
3 bit → 8 CLIC

$$2^3 = 2 \times 2 \times 2 \\ = 8.$$

High



Timing Diagram



Clock	Q _A	Q _B	Q _C
0	0	0	0
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

Asynchronous Down Counter

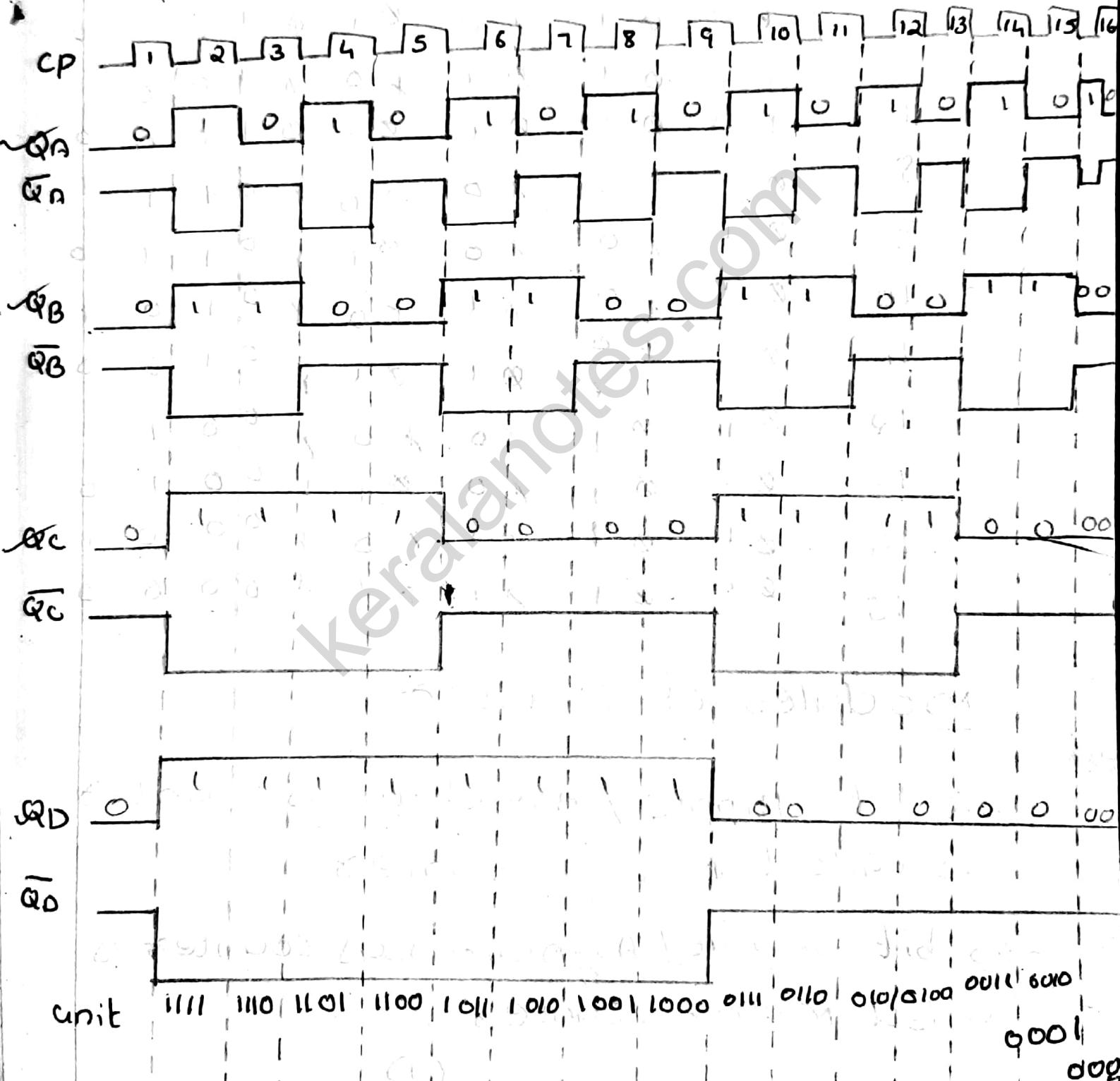
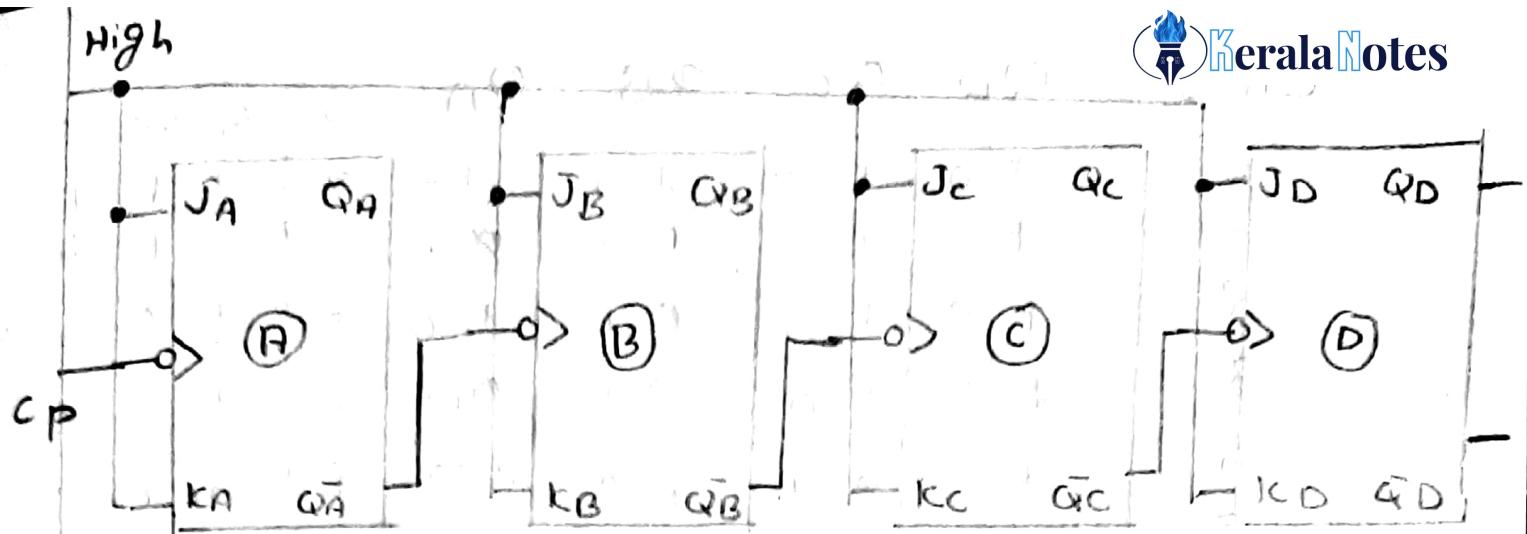
The ~~so~~ down counter will count downwards from a maximum count to zero.

→ In a 4 bit asynchronous down counter the clock signal is connected to clock of first flip flop.

→ The clock of remaining flip flop is triggered by an bar output of previous stage.

4 Bit Asynchronous Down Counter

$$2^4 = 2 \times 2 \times 2 \times 2 = 16$$



CLOCK	QD	QC	QB	QA	Q _D	Q _C	Q _B	Q _A
0	x 0	1 0	1 0	1 0	0 0	1 1	1 1	1 1
1	x 0	1 0	1 0	0 1	0 0	1 1	1 1	0 0
2	x 0	x 0	0 1	1 0	0 0	1 1	1 0	1 0
3	x 0	1 0	0 1	0 1	0 0	1 1	1 0	0 0
4	x 0	0 1	x 0	1 0	0 1	1 0	0 1	1 1
5	x 0	0 1	x 0	0 1	0 0	1 0	0 1	0 0
6	1 0	0 1	0 1	x 0	0 0	1 0	0 0	1 1
7	x 0	0 1	0 1	0 1	0 0	1 0	0 0	0 1
8	0 1	1 0	1 0	x 0	1 0	0 1	1 1	0 0
9	0 1	x 0	x 0	1 0	0 1	1 0	1 1	0 0
10	0 1	x 0	0 1	x 0	0 1	0 1	0 0	1 1
11	0 1	x 0	0 1	0 1	0 1	0 1	0 0	0 0
12	0 1	0 1	x 0	x 0	0 0	0 0	1 1	1 1
13	0 1	0 1	x 0	0 1	1 0	0 0	1 0	0 0
14	0 1	0 1	0 1	x 0	1 0	1 0	0 0	1 1
15	0 1	0 1	0 1	0 1	0 1	1 0 0	0 0	0 0

modules of counter

→ 2 bit Ripple / Asynchronous counter
is called MOD-4 counter

→ 3 bit Ripple / Asynchronous counter is
called MOD-8 counter

000 111 (7)
total 8.

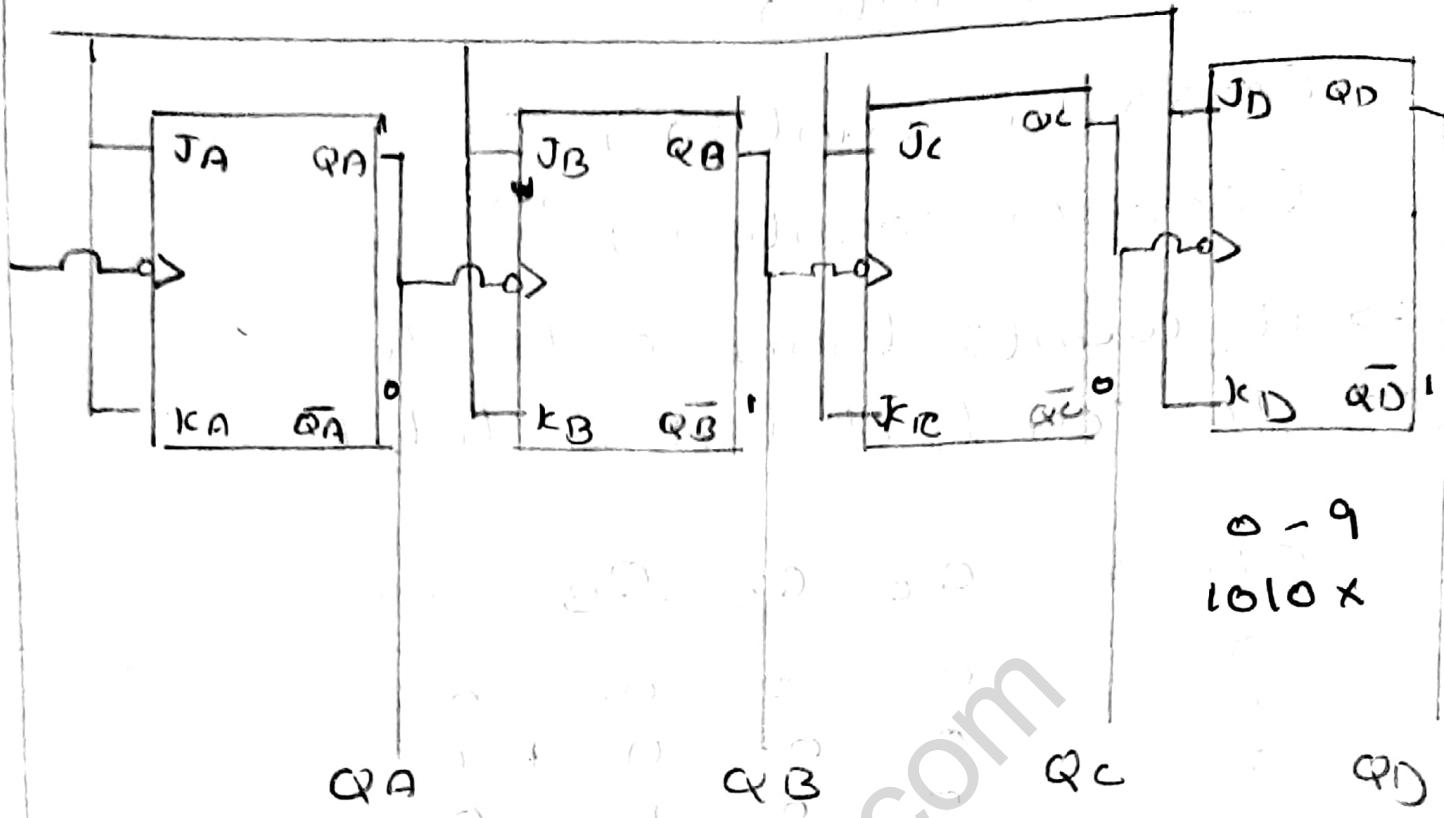
BCD counter

(MOD-10 counter)

(Decode counter)

- It counts from 0 to 9
- To count, it takes 4 bits

CLOCK	Q ₀	Q ₁	Q ₂	Q ₃
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
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



0 - 9
1010 X

QA QB QC QD

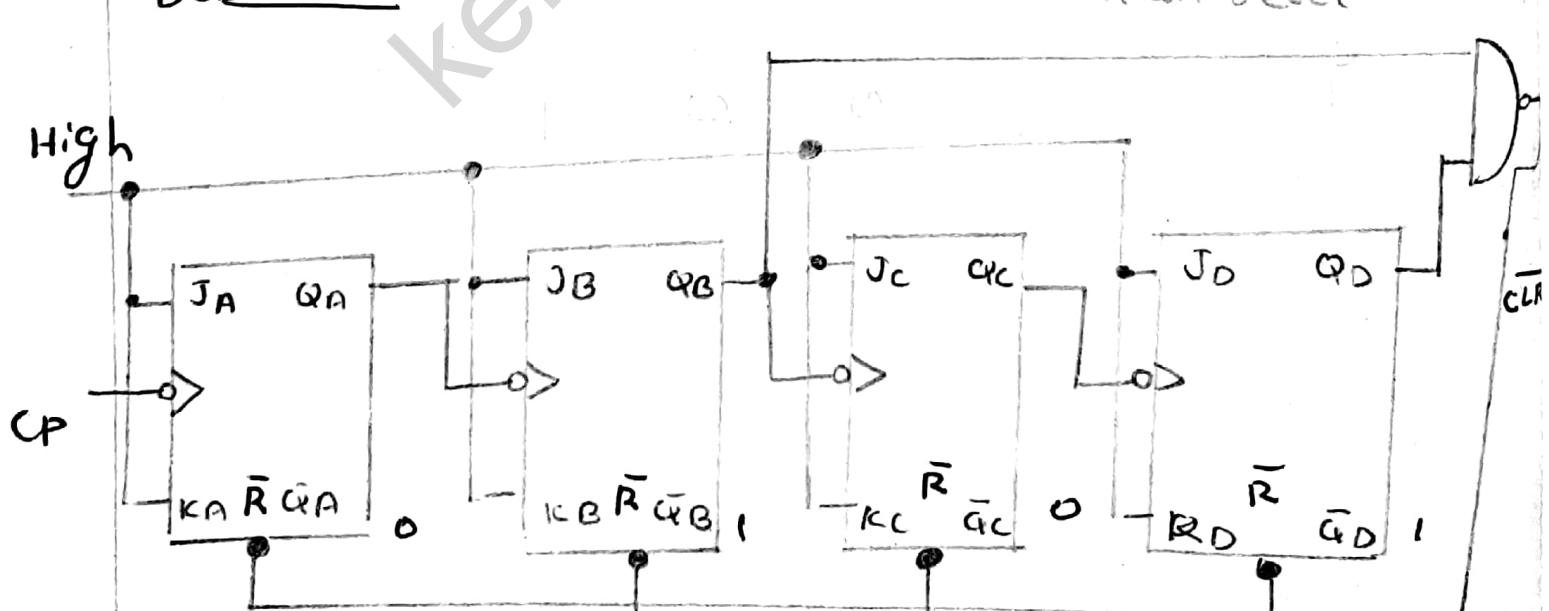
0000, 0001, 0010, 0011, 0100, 0101, 0110,
0111, 1000, 1001, 1010

when it become enable it will clear and
reset to first

0 - 9 1010 - X [10]

when it become 1010
it will reset

BCD counted:



we have to delete (1) so we
connect it to a NAND Gate

mod 16
mod 10
mod 6
101

Synchronous counter

Clock
Simultaneously to all
flipflop

Steps:

1. Decide the number of flipflop
 2. Excitation table of flipflop
 3. State diagram and circuit
excitation table
 4. Obtain simplified equation using k-mup
 5. Draw the logic Diagram
- *—*
- (i) synchronous up counter
 - (ii) synchronous Down counter
 - (iii) synchronous up-down counter
 - (iv) synchronous BCD counter

JK Excitation table

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

T-excitation table

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

Q Design 3 bit synchronous up counter using T-flipflop

ans: 1. No. of flipflop = 3

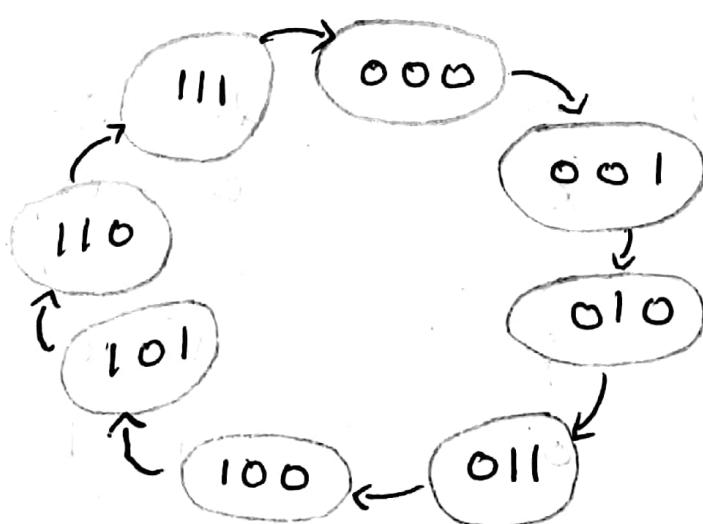
Flip flop used = T-flip flop

2. Excitation table

Present state		Next state	
Q_0	Q_{0+1}	\bar{T}	T
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

3. State diagram

3 bit have 8 state



Circuit excitation table

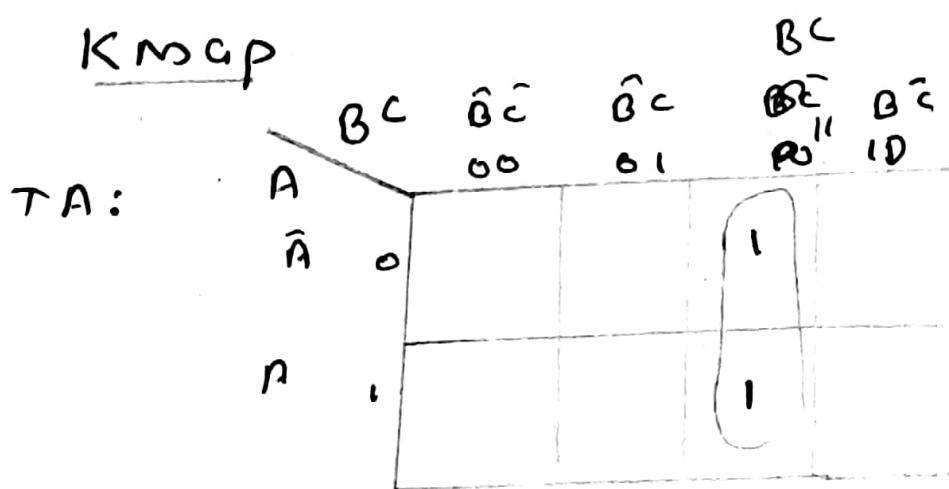
Present state			Next state			Flip flop 1/p		
(A)	B	C	(A+)	B+	C+	T _A	T _B	T _C
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

⇒ if present state is 000 next state will be 001

T_A = comparing A, A+

4. obtain boolean Expression using

Kmap



$$T_A = \underline{\underline{B}C}$$

$TB : A \setminus B^C$

\bar{B}^C	\bar{B}^C	B^C	B^C
00	01	11	10

\bar{A}	0	1	1	0
A	1	0	1	1

$$TB = C$$

$TC : A \setminus B^C$

\bar{B}^C	\bar{B}^C	B^C	B^C
00	01	11	10

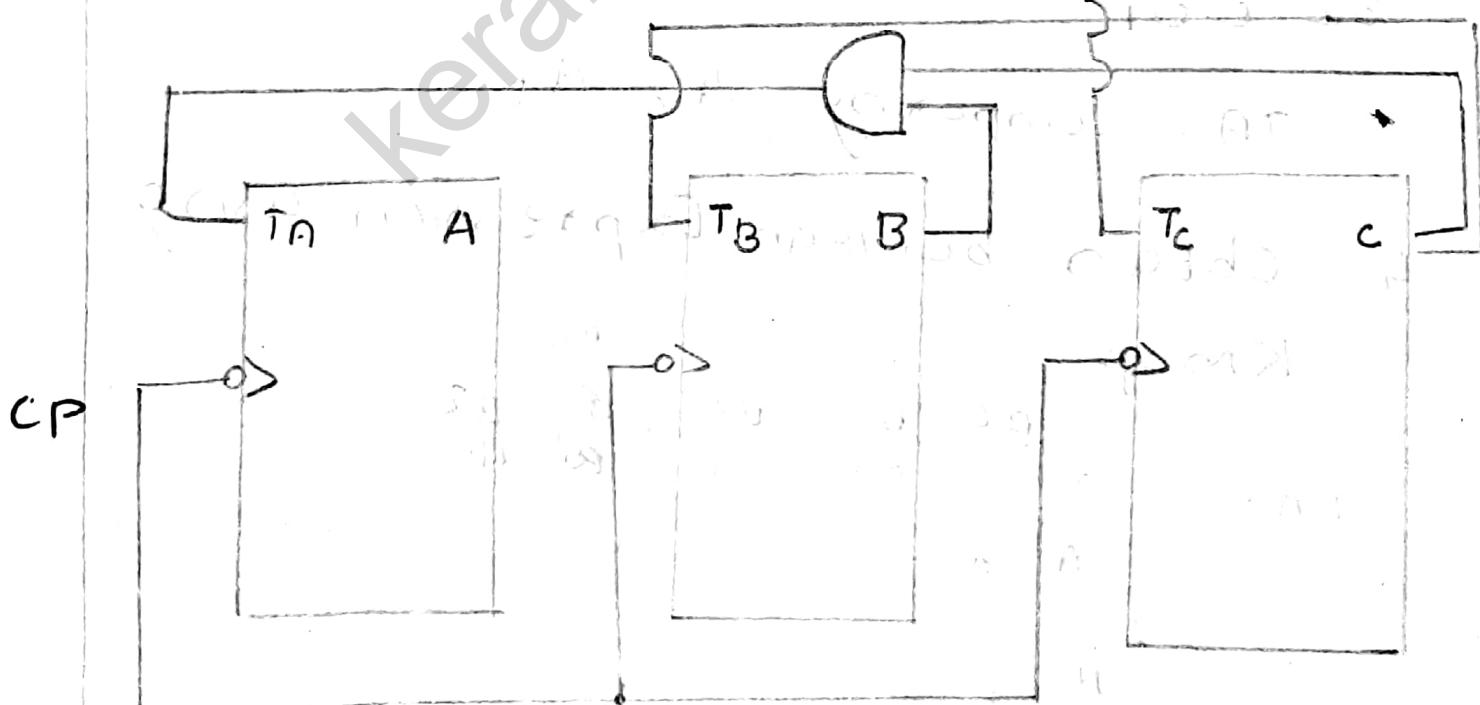
\bar{A}	0	1	1	1
A	1	1	1	1

$$TC = 1$$

5. Draw logic Diagram

$$TA = B^C, TB = C, TC = 1$$

High(1)



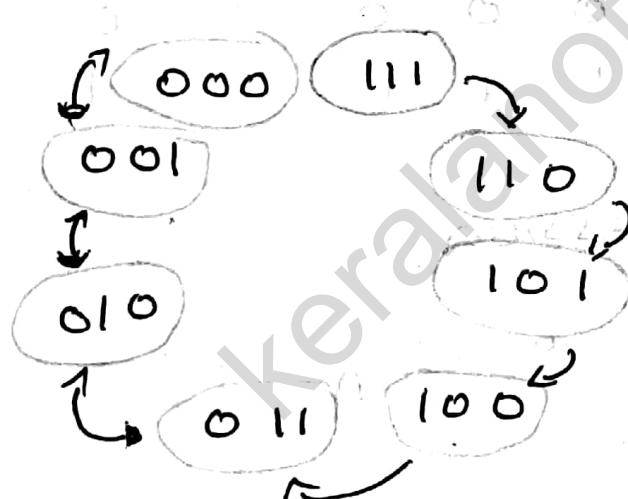
Q) Draw 3 bit synchronous counters using T-flip flop

1. 3

2. Excitation table

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

3. State diagram



8	7	6	5
4	3	1	2

Circuit Excitation table

Present state			Next state			flipflop i/p		
A	B	C	A+	B+	C+	τ_A	τ_B	τ_C
1	1	1	1	1	0	0	0	1
1	1	0	1	0	1	0	1	1
1	0	1	1	0	0	0	0	1
1	0	0	0	1	1	1	1	1
0	1	1	0	1	0	0	0	1
0	1	0	0	0	1	0	0	1
0	0	1	0	0	0	0	0	1
0	0	0	1	1	1	1	1	1

4. Boolean Expression

$\tau_A :$	BC	00	01	11	10	
	A	1				
	B	1				

$$\tau_A = \bar{B}\bar{C}$$

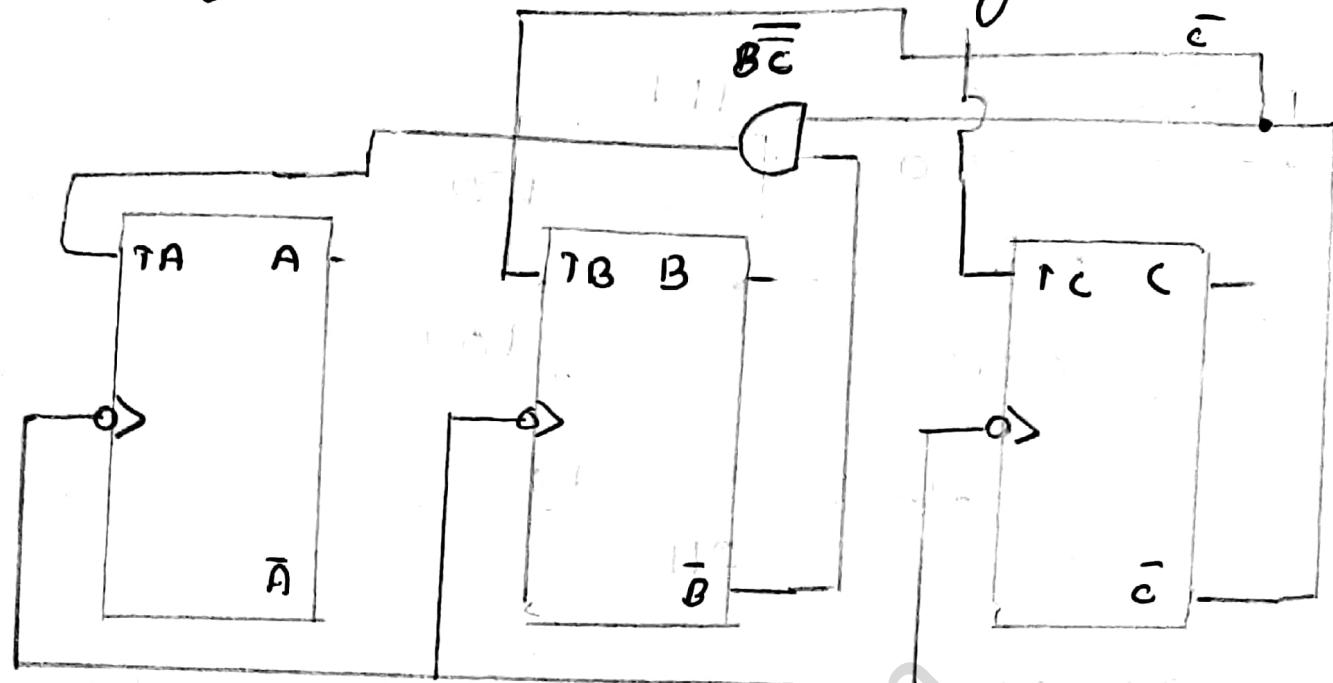
$\tau_B :$	BC	00	01	11	10	
	A	1				
	B	1				

$$\tau_B = \bar{C}$$

$\tau_C :$	BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$	
	A	0	1	1	0	
	B	0	1	1	1	

$$\tau_C = 1$$

5. Logic Diagram



Q Design 3 bit Synchronous Down
counter using JK flipflop

Ans: Step1: Determine Number of flipflop

it require 3 JK flipflop

Step2: The Excitation Table

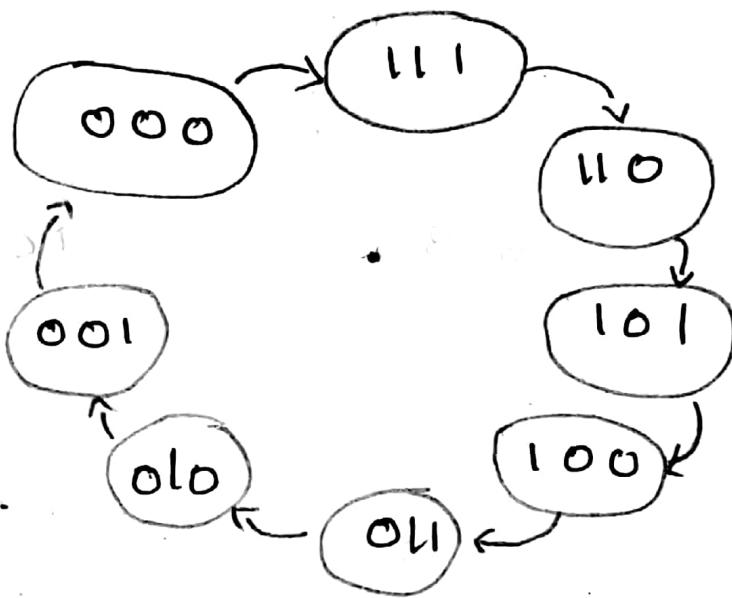
Q_n	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

Step 3: Draw the state Diagram



Kerala

Notes



Step 3b: circuit Excitation table

Present state			Next state			Required Excitation					
Q_3	Q_2	Q_1	Q_3^*	Q_2^*	Q_1^*	\bar{J}_3	K_3	J_2	K_2	J_1	K_1
1	1	1	1	1	0	x	0	x	0	x	1
1	1	0	1	0	1	x	0	x	1	1	x
1	0	1	1	0	0	x	0	0	x	x	1
1	0	0	0	1	1	x	1	1	x	1	x
0	1	1	0	1	0	0	x	x	0	x	1
0	1	0	0	0	1	0	x	x	1	1	x
0	0	1	0	0	0	0	x	0	x	x	1
0	0	0	1	1	1	1	x	1	x	1	x

Q	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1

Step 4: Simplified Boolean Expressions

KCMGIP

		Q ₂ Q ₁	Q ₃		
		00	01	11	10
Q ₃	0	1			
	1	X	X X	X	X

$$J_3 = \overline{Q_2} \overline{Q_1}$$

		Q ₂ Q ₁	Q ₃		
		00	01	11	10
Q ₃	0	X	X	X	X
	1	1			

$$K_3 = \overline{Q_2} \overline{Q_1}$$

		Q ₂ Q ₁	Q ₃		
		00	01	11	10
Q ₃	0	1			
	1	1	X	r	X

$$J_2 = \overline{Q_1}$$

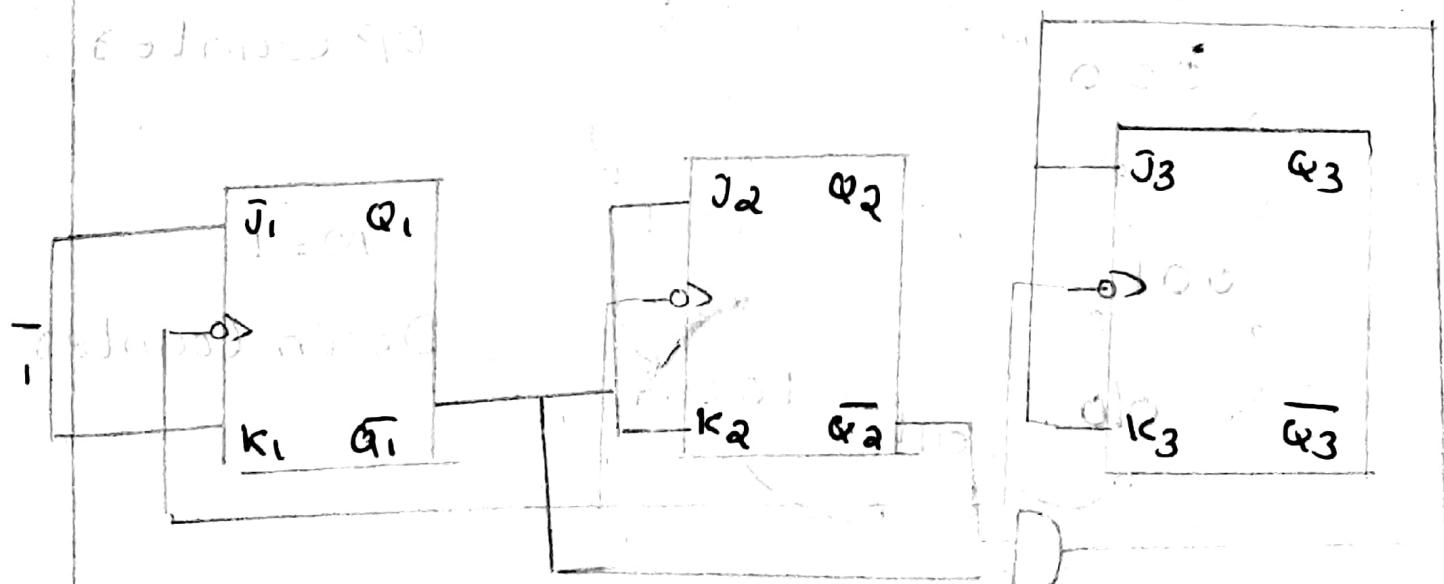
		Q ₂ Q ₁	Q ₃		
		00	01	11	10
Q ₃	0	X			
	1	X			

$$K_2 = \overline{Q_1}$$

$$J_1 = 1$$

$$K_1 = 1$$

Step 5: Logic Diagram



Q Design 3 bit synchronous  **KeralaNotes**
up-down counter using T-Flipflop

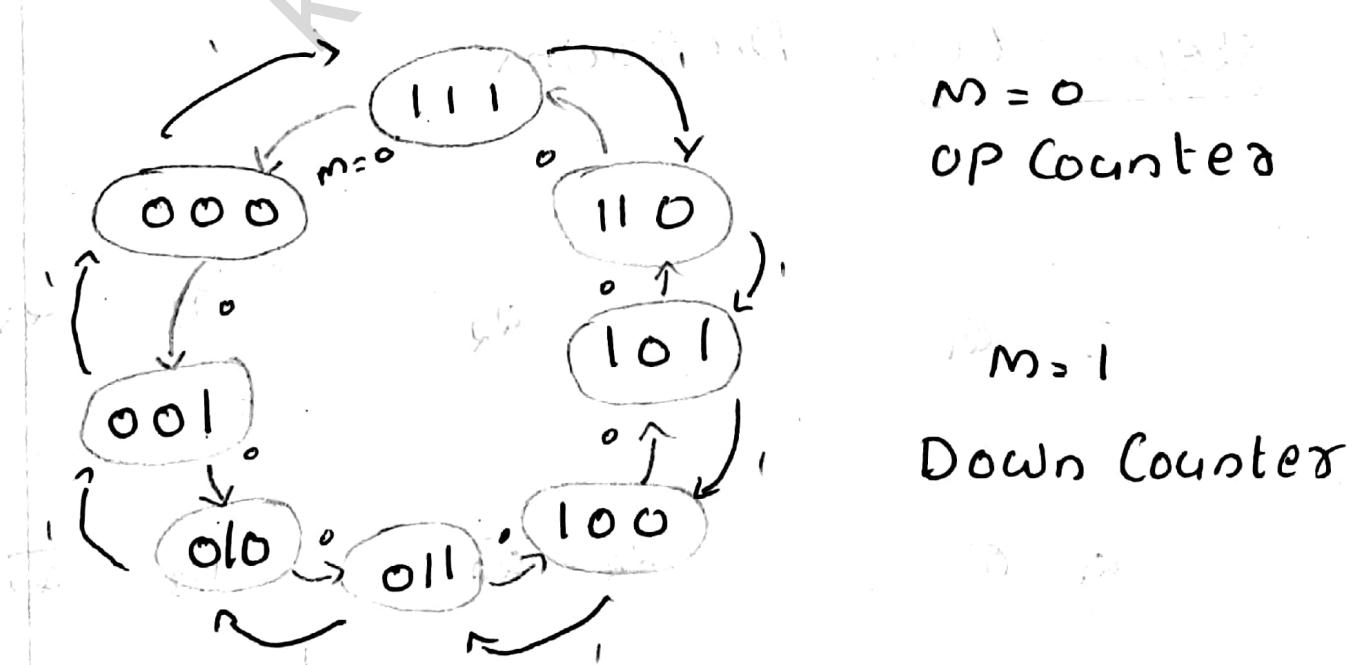
Ans: Step 1: Determine number of flipflop

It requires 3 - T Flipflop

Step 2: The Excitation Table

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

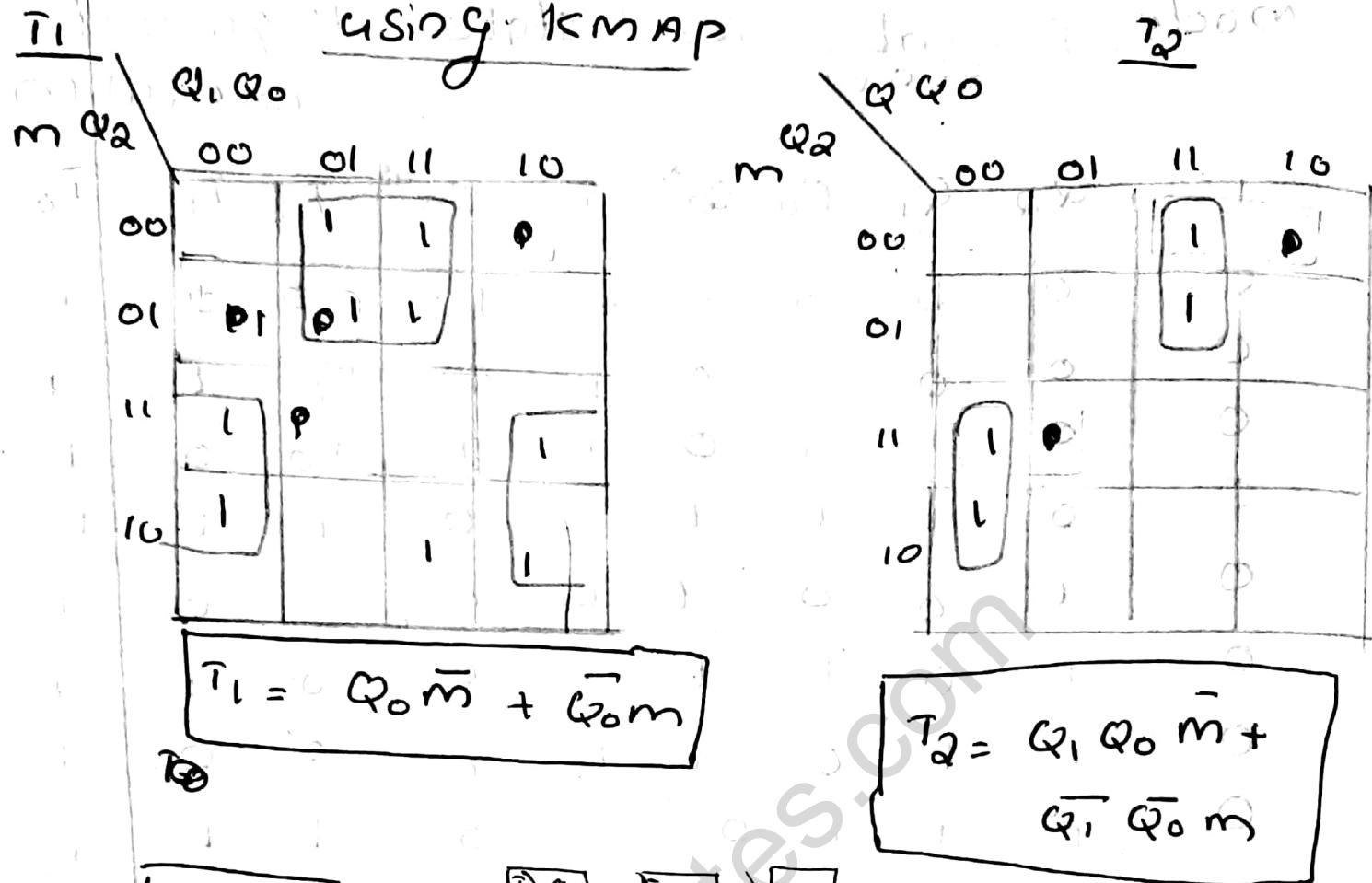
Step 3: Draw the state diagram



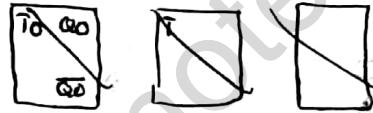
Step 3 b : The circuit Excitation KeralaNotes

Mode	Present State			Next State			Required Excitation		
	Q_2	Q_1	Q_0	Q_2^*	Q_1^*	Q_0^*	T_2	T_1	T_0
Up counter	0	0	0	0	0	1	0	0	1
	0	0	0	1	0	1	0	1	1
	0	0	1	0	0	1	0	0	1
	0	1	0	0	1	1	0	0	1
	0	1	1	1	0	0	1	1	1
	1	0	0	1	0	1	0	0	0
	0	1	0	1	1	1	0	1	1
	0	1	1	0	1	1	0	0	1
	0	0	1	1	0	0	1	1	1
	1	1	0	0	0	0	1	1	1
Down counter	1	1	1	0	0	0	1	1	1
	1	1	0	1	0	0	0	0	0
	1	0	0	0	1	0	0	0	0
	1	0	1	0	0	1	0	1	1
	1	0	0	1	0	0	0	0	1
	1	1	0	0	1	0	1	1	1
	1	1	1	0	0	1	0	0	1
	1	1	1	1	0	0	0	0	1
	1	1	1	1	1	0	0	0	0
	1	1	1	1	1	1	0	0	0

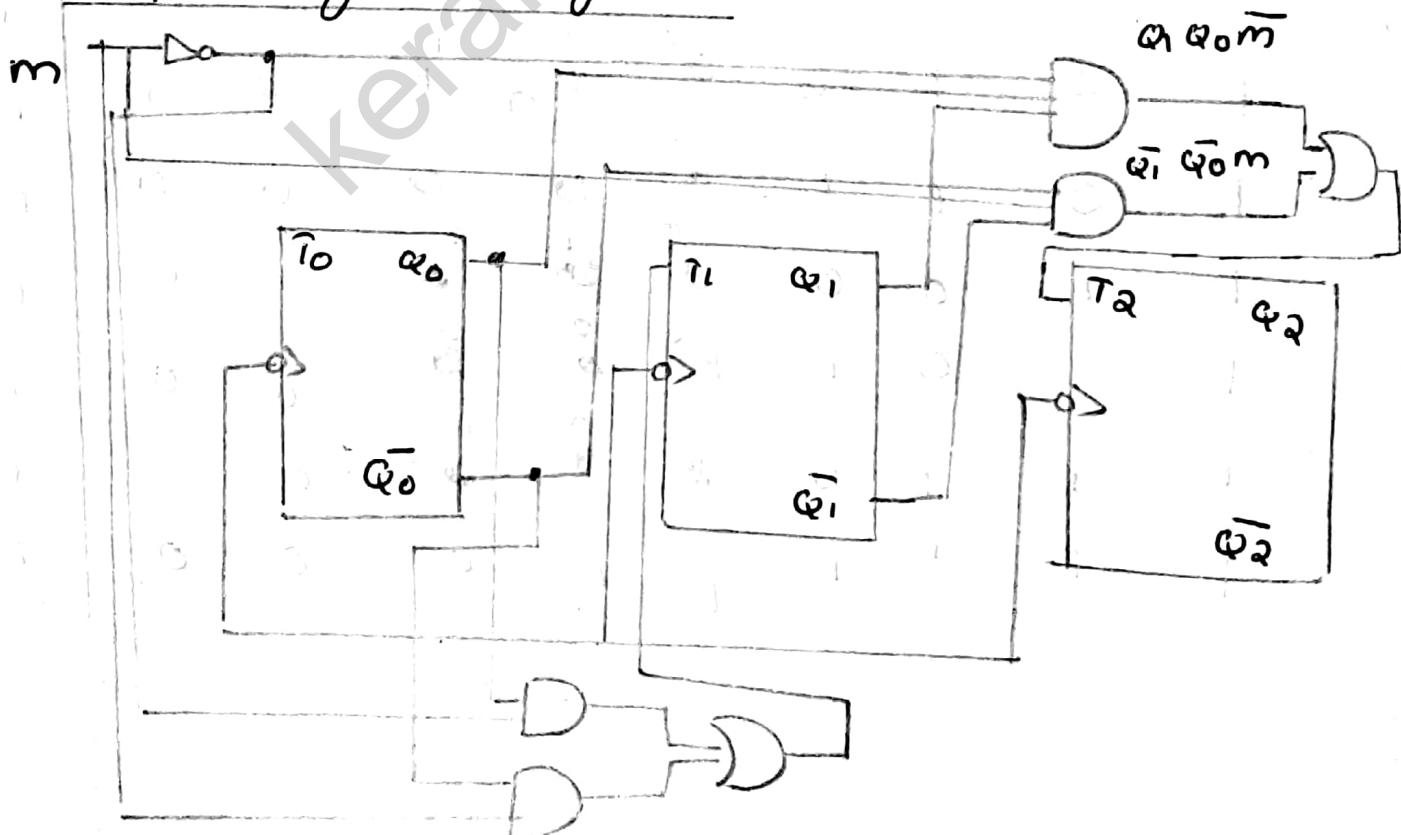
Step 4 : Simplified Boolean Expressions



$$T_0 = 1$$



Step 5 : Logic Diagram



ans: Using T flipflop

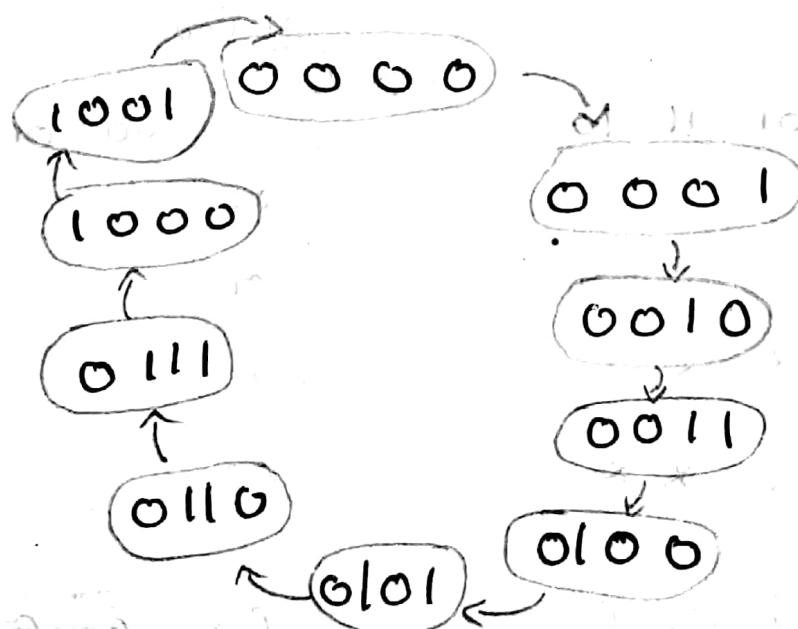
ans: Step 1: Determine number of flipflop

4 T flipflop

Excitation Table

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

State Diagram



Circuit Excitation Table

Present state				Next state				Required state			
Q_3	Q_2	Q_1	Q_0	Q_3^*	Q_2^*	Q_1^*	Q_0^*	T_3	T_2	T_1	T_0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	0	1
0	1	1	0	0	1	1	0	0	0	0	1
0	1	1	1	1	0	0	0	1	0	1	0
1	0	0	0	1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	1	0	0	0	1

Boolean Expression

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

$$T_3 =$$

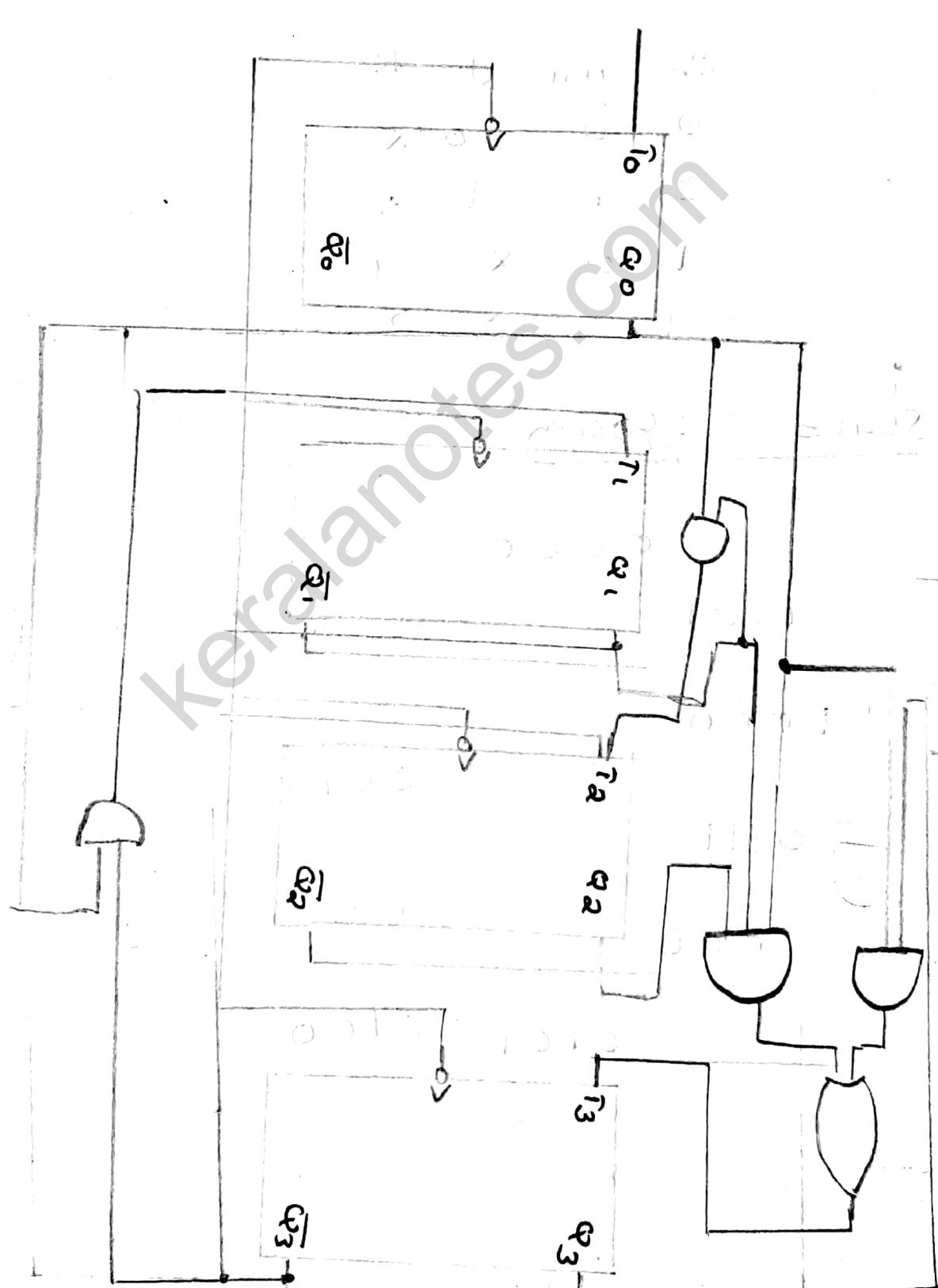
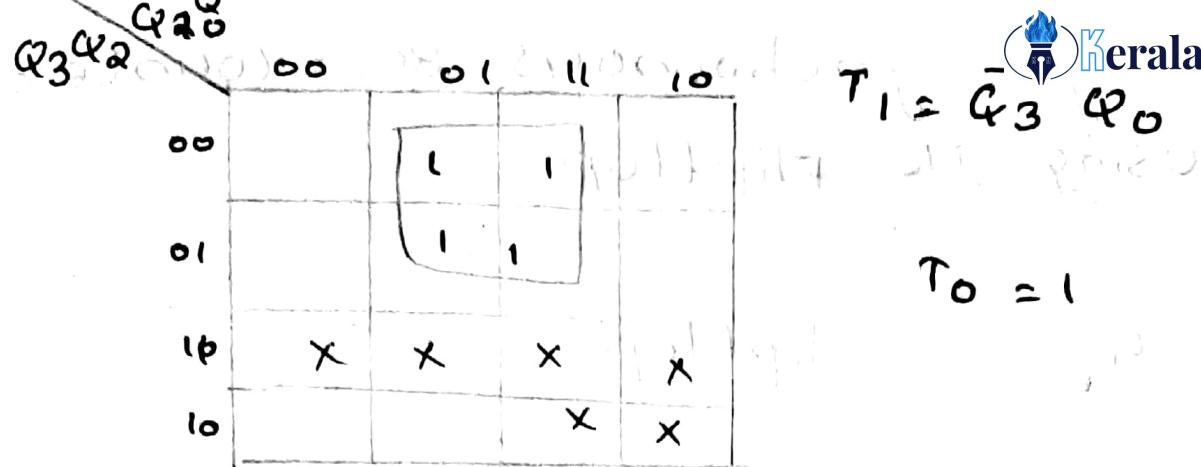
$$Q_2 Q_1 Q_0 + Q_3 Q_0$$

$$T_3 = Q_3 \bar{Q}_2 \bar{Q}_1 Q_0 +$$

$$Q_3 Q_2 Q_1 Q_0$$

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

$$T_2 = \bar{Q}_3 \bar{Q}_2 Q_1 Q_0$$



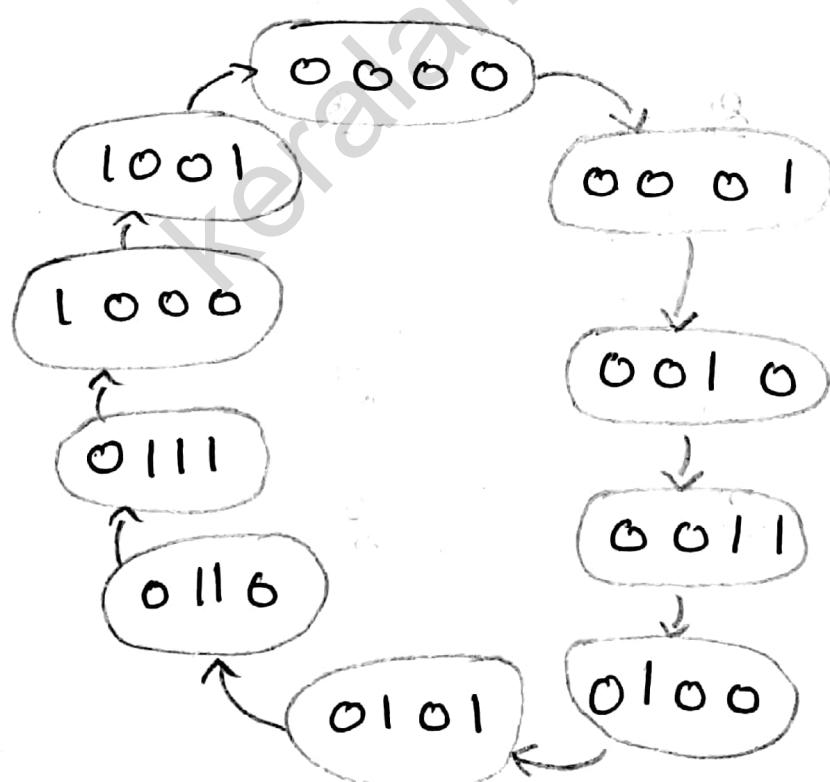
2. Design Synchronous BCD Using JK flipflop

↳ JK - flipflop

Excitation Table

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

State Diagram



Circuit Excitation Table

Present state				Next state				Required state								
Q3	Q2	Q1	Q0	Q3	Q2	Q1	Q0	J3	K3	J2	K2	J1	K1	J0	K0	
0	0	0	0	0	0	0	0	1	0	x	0	x	0	x	1	x
0	0	0	1	0	0	1	0	0	0	x	0	x	1	x	x	1
0	0	1	0	0	0	1	1	0	x	0	x	x	0	1	x	
0	0	1	1	0	1	0	0	0	0	x	1	x	x	1	x	
0	1	0	0	0	1	0	1	0	x	x	0	0	x	1	x	
0	1	0	1	0	1	1	1	0	0	x	0	1	x	x	1	
0	1	1	0	0	0	1	1	1	0	x	x	0	x	1	x	
0	1	1	1	1	1	0	0	0	1	x	x	1	x	1	x	
1	0	0	0	1	0	0	0	1	x	0	0	x	0	x	1	
1	0	0	1	0	0	0	0	0	0	x	1	x	0	x	1	

Boolean Expression

0, 9, BCD
Others C(0, 15)
Don't care

		Q1 Q0				
		Q3 Q2	00	01	11	10
		Q3 Q2	00	01	11	10
00						
01						
11		x	x	x	x	x
10		x	x	x	x	x

$$J_3 = \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0$$

		Q1 Q0				
		Q3 Q2	00	01	11	10
		Q3 Q2	00	01	11	10
00			x	x	x	x
01			x	x	x	x
11			x	x	x	x
10			0	1	x	x

$$K_3 = Q_0$$

Q_1, Q_0
 Q_3, Q_2

	00	01	1L	10
00				
01	X	X	X	X
1L	X	X	X	X
10		X		X

 Q_1, Q_0
 Q_3, Q_2

	00	01	1L	10
00		X	X	X
01	O	O	1	O
1L	X	X	X	X
10	X	X	X	X

$$J_2 = \overline{Q_3} Q_2 + Q_0$$

 Q_1, Q_0

$$K_2 = Q_1 Q_0$$

 Q_1, Q_0

	00	01	1L	10
00		1	X	X
01		1	X	X
1L	X	X	X	X
10		X	X	

 Q_1, Q_0
 Q_3, Q_2

	00	01	1L	10
00	X	X	1	
01	X	X	1	
1L	X	X	X	X
10	X	X	X	X

$$J_1 = \overline{Q_3} Q_0$$

~~$$J_2 = \overline{Q_3} Q_0$$~~

$$J_0 = 1$$

$$K_1 = Q_0$$

$$K_0 = 1$$

