

Chapter - 7.

Sequential Logic

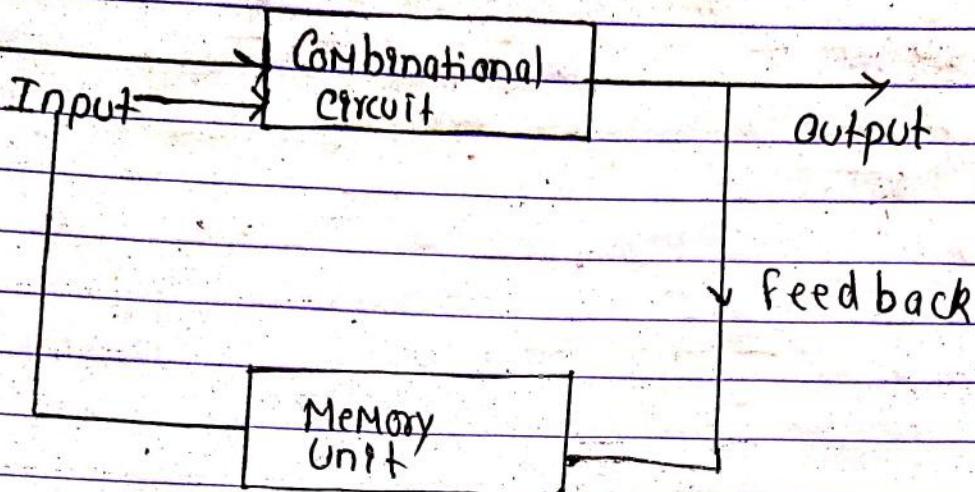


fig: Block Diagram of a Sequential circuit.

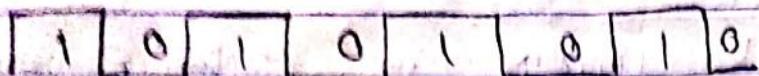
The logic circuit whose outputs at any instant depend not only on the present input but also on past outputs are called sequential circuits. A sequential circuit consists of a combinational circuit to which storage elements are connected to form a feedback path. The storage elements are device capable of storing binary information.

The binary information stored in the memory unit of sequential circuit is known as State. From the block diagram the next state and O/P. is the function of present state and I/Ps.

There are two types of sequential circuits;

1) Synchronous Sequential circuit.

It is a system whose behaviour is defined at discrete interval of time. The Synchronous ckt is synchronized by timing device called "master-clock generator". The Master clock generator generates a periodic train of clock pulse. as given as,



They are also known as clocked sequential ckt.

- The clockpulse is fed to each and every memory unit.
- The output of memory unit is affected at positive level or negative level or positive edge or negative edge of the clock pulse.
- In this way sequential ckt works only at discrete interval of time.
- The synchronous sequential ckt which use clock is known as 'clocked' Sequential ckt.
- The clocked Sequential ckt are most widely used in digital system.

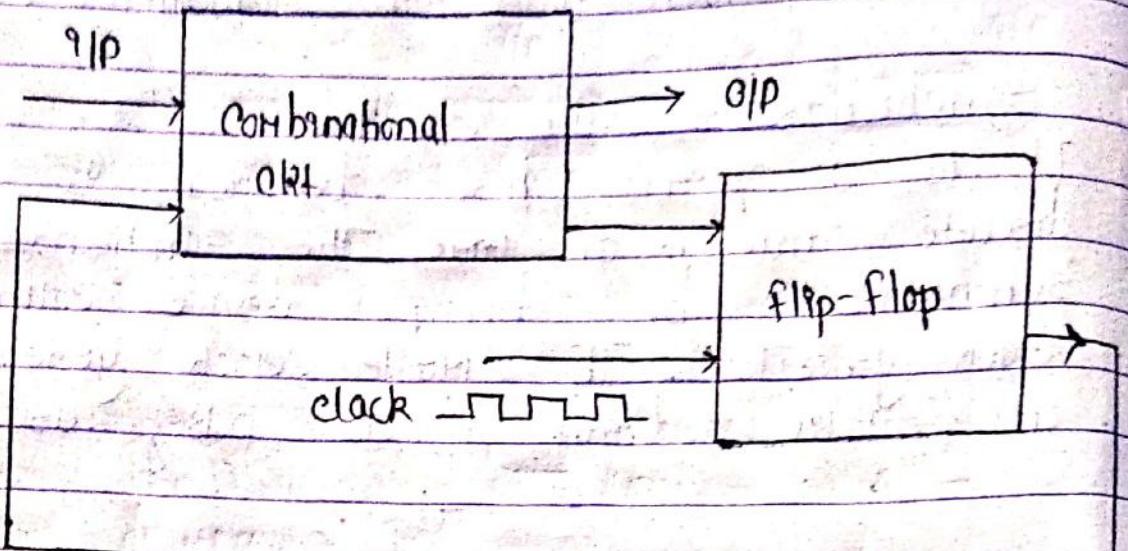


fig: Block diagram of clocked Sequential ckt.

- The block diagram of clocked Sequential ckt is similar to that of Sequential ckt with only difference in memory unit.
- In clocked Sequential ckt, memory element is known as flipflop (F/F).
- The F/F has has, the I/P from combinational ckt as well as from clock.
- The Sampling of the I/P in the F/F takes place only at the transition of clock. The consequence of the above statement is that the transition of state in the clocked Sequential ckt occurs only at the transition of clock.

~~CHAPTER~~ ~~QUESTION~~ ~~TOPIC~~

Asynchronous Sequential circuit

It is a system whose behaviour depends in the order in which its input signal change and can't be affected at any instant of time. Asynchronous Sequential ckt are very unstable and due to this fact it is very rarely used.

Differences between:

Combinational logic

i) Combinational logic ckt consists of i/p variables, o/p variables, logic gates interconnections.

ii) Logic gates are basic elements used.

iii) No Memory element present.

iv) O/p at any instant can be calculated from the function with i/p's.

v) Input triggers the Combinational logic circuit.

Sequential logic

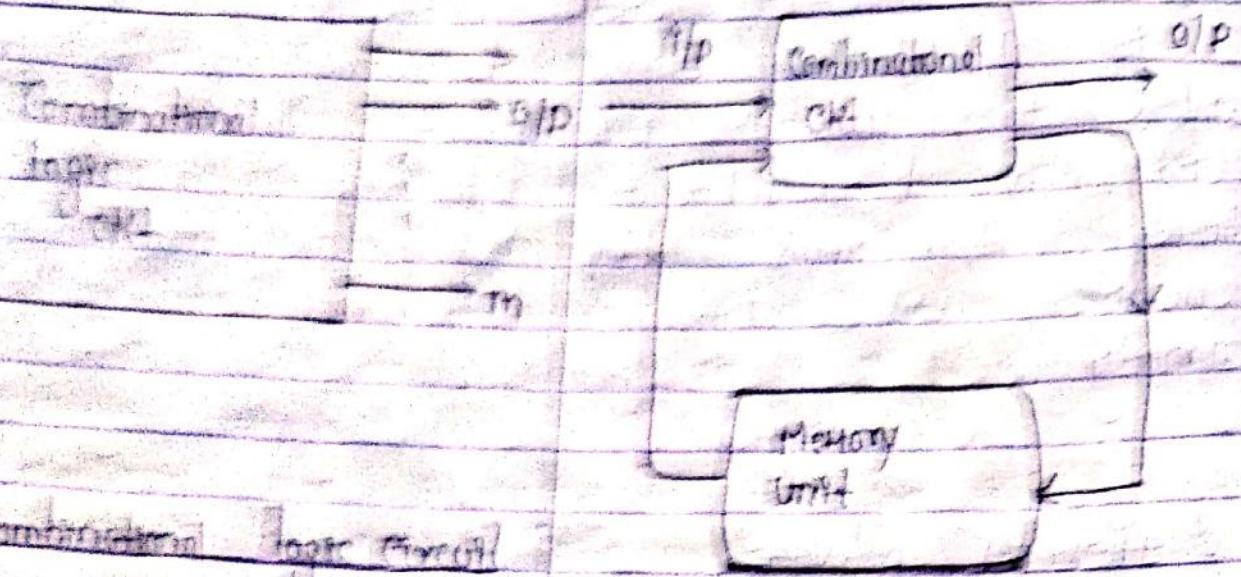
It consists of inputs variables, o/p variables, logic gates, interconnections (i.e. Combinational logic ckt) and memory unit.

Flip-flops are the basic elements used.

Memory element present for the storage.

O/p depends upon the present i/p's and past o/p's (i.e. feedback paths).

Clock pulse (C_p) and previous o/p triggers the ckt.



By Sequential logic O/P

Advantages

- 1) No clock pulse.
- 2) Asynchronous.
- 3) Used as buffer.
- 4) Works with clock signal but works with a latched signal. Input is level signal. It is one bit storage element and works with a clock signal. It is edge triggered device.
- 5) Memory latches are used but flip-flops are preferred.

Disadvantages

Clock pulse is provided.

Synchronous.

Used as storage element.

It is one bit storage element and works with a clock signal.

It is edge triggered device.

But flip-flops are preferred.

Flipflops

The memory elements used in clocked sequential circuits are called flipflops. It is used for storing binary information or maintaining binary state indefinitely until directed by an input signal. Flipflops has two outputs or levels; one for the normal value and next for the complement value of the bit stored in it. Therefore, they are called bistable circuits.

Flip-flops are of different types which depends upon the number of inputs and its effect in binary state.

latch

Flipflops with no control or clocked pulse or signal is called latch. It is also a bistable device which holds or latch into two stable state. The basic latch is constructed from the cross coupled NAND or NOR gates. Such basic latch is known as SR-latch. Latch is also a storing or memory unit.

SR latch

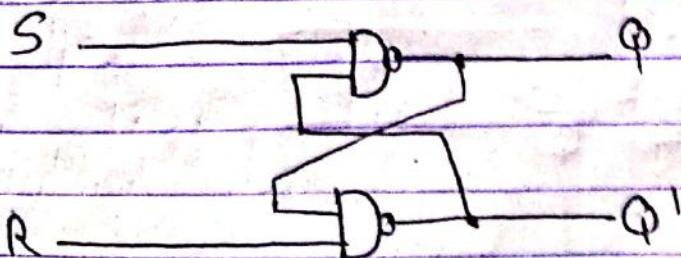


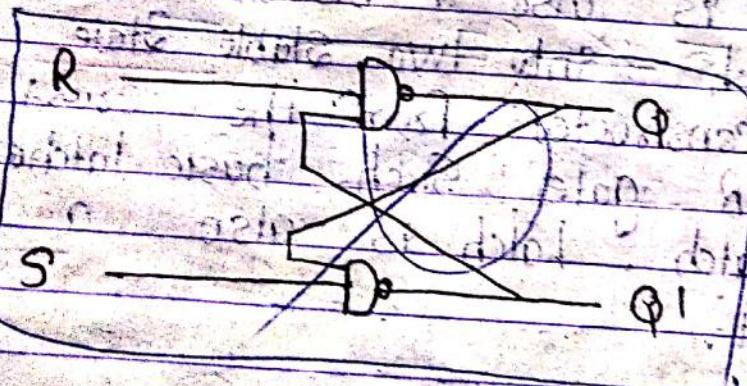
Fig: Basic SR latch constructed from NAND gate.

TruthTable

S	R	Q	Q'	State
0	1	1	0	Set
1	1	1	0	Nochange
1	0	0	1	Reset
1	1	0	1	Nochange
0	0	1	1	Invalid

To analyze the operation of ckt we should know the output of NAND Gate.

- ① For any one input '1' or both inputs '0' the output is 1.
- ② For both Input '1' the output is 0.



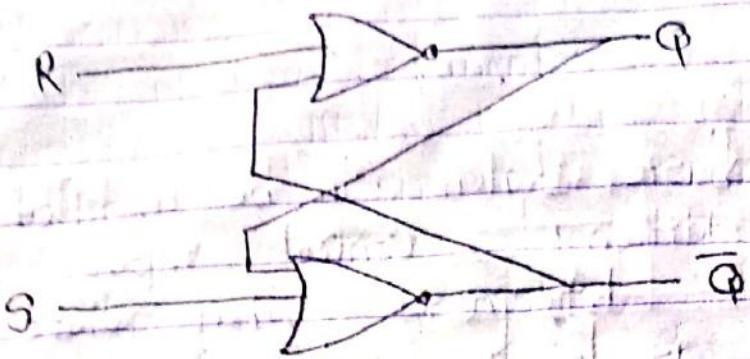


Fig: NOR based S-R latch.

Truth table

S	R	Q	Q'	State
1	0	1	0	set
0	0	1	0	no change
0	1	0	1	reset
0	0	0	1	no change
1	1	0	0	invalid

From the truth table, SR latch constructed from NAND is opposite and also called ex denoted by NAND's R latch. SR latch reminds us that '0' should be applied to 'S' to set the latch.

(2) Gated latch (Flipflop).

(1) RS flip flop

The operation of basic latch can be modified by providing an additional control input (Enable Signal) that determines when state of the circuit is to be changed.

As been discussed above the RS-Flipflop can be realized by both NAND based and NOR based.

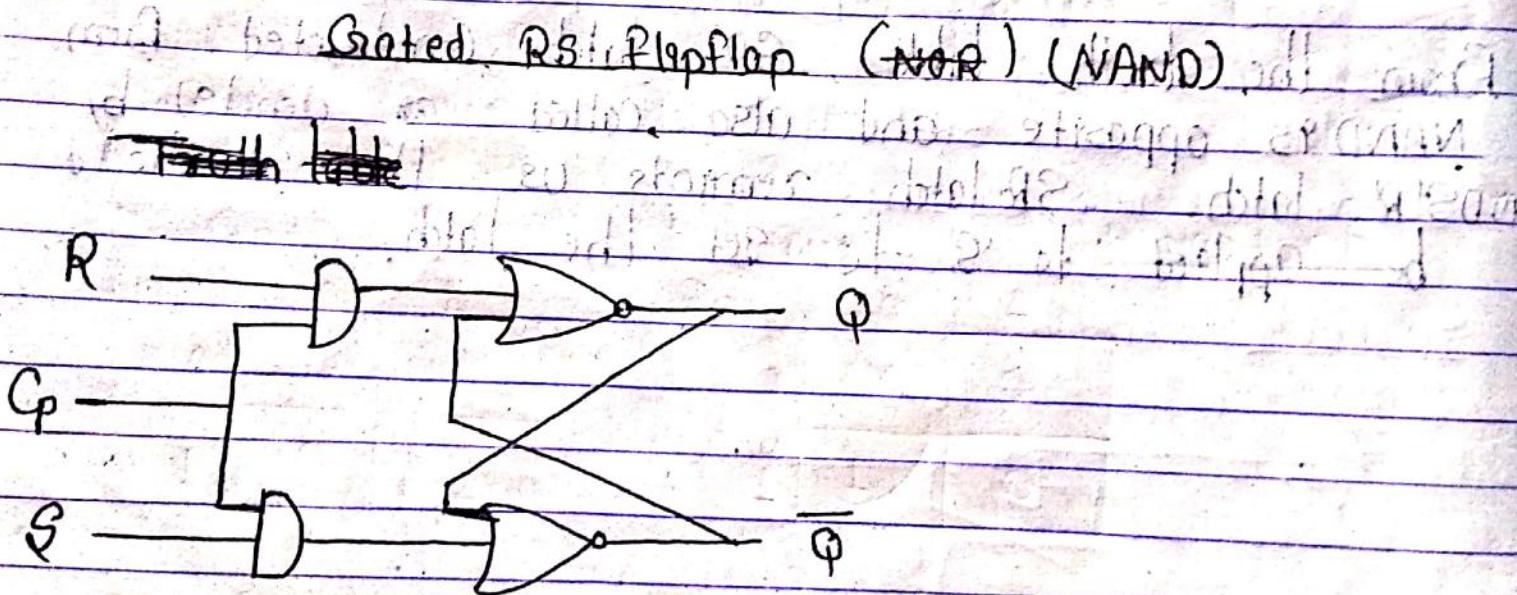
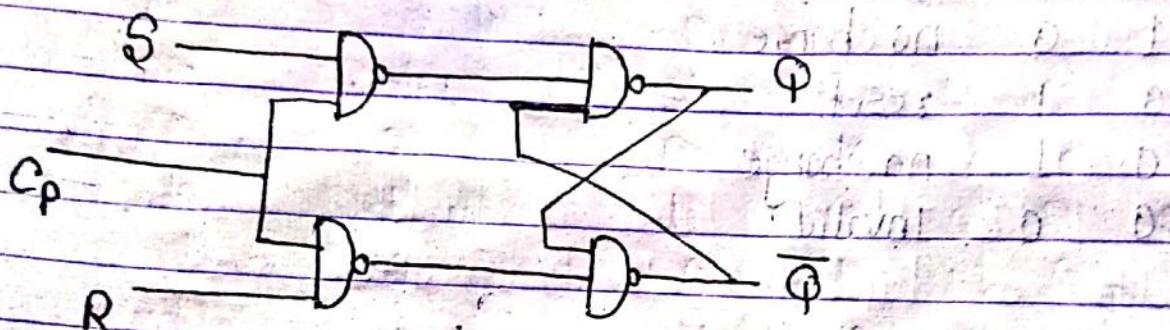
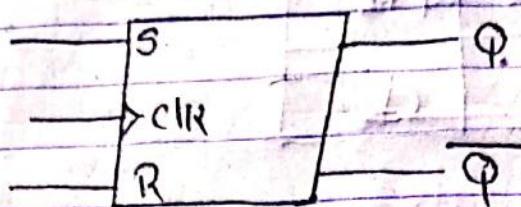


Fig: Clocked NOR based flipflop

NOR-based R-S flipflop consists of two additional AND gate at the S & R input side as in Fig:



When $C_p = 0$, R & S both, so it holds the information
When $C_p = 1$, O/p depends on the value of R & S

Fig: Graphical Symbol

Truth table

C_p	S	R	$Q(t+1)$	Remarks
0	X	X	Q_t (Memory)	No change
1	0	0	Q_t	No change
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	X	invalid

Characteristics table

$Q(t)$	S	R	$Q(t+1)$
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

Here, $Q(t+1)$ is the state of flipflop after the occurrence of a clock pulse.
(refers to next state)

K-map for Q_{n+1}

Q_n	00	01	11	10
Q_{n+1}	0		X	1
1	D		X	1

$Q_{n+1} = Q_n \bar{R} + S - Q_n$ is the characteristic equation.

Excitation table:

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

K-map for S

Q_n	0	1
Q_{n+1}	0	0
0	0	(D)
1	0	X

K-map for R

Q_n	0	1
Q_{n+1}	0	(X)
0	X	(1)
1	0	0

$$S = Q_{n+1}$$

$$R = Q_{n+1}$$

(99) Gated D-FlipFlop (Delay on Data flipflop).

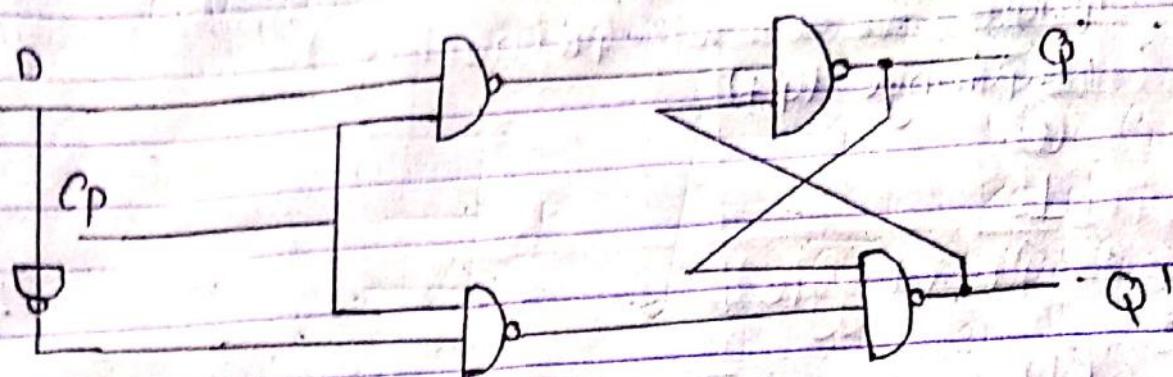


fig: logic Diagram.

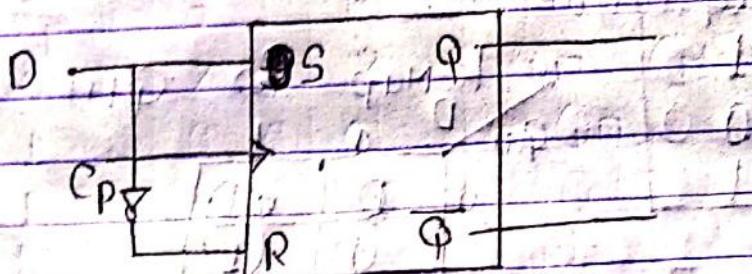


fig: Block diagram of D-flipflop

D-Flipflop eliminates the undesirable condition of the indeterminate state of SR flipflops (i.e. S=R=1). D-Flipflops has only two input : D (Data) and C (Control or Enable). and two output Q and Q'. From the above figure, 'D' input goes directly to the 'S' input and its complement is applied to the 'R' input.

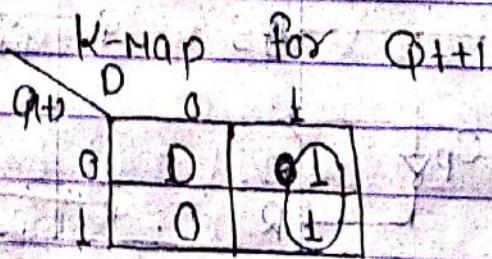
The output of 'Q' is same as 'D' input.

Truth table

C	D	$Q(t+n)$
0	x	No change ($Q+1$)
1	0	0
1	1	1

characteristics table

$Q(t)$	D	$Q(t+n)$
0	0	0
0	1	1
1	0	0
1	1	1

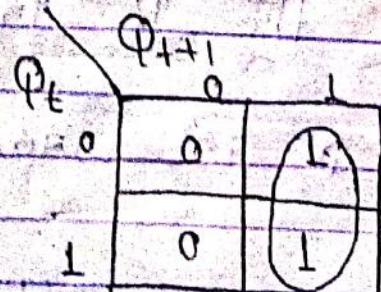


excitation table

$$Q(t+1) = D$$

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

R-map for D



$$D = Q(t+1)$$

(iii) JK-flipflop

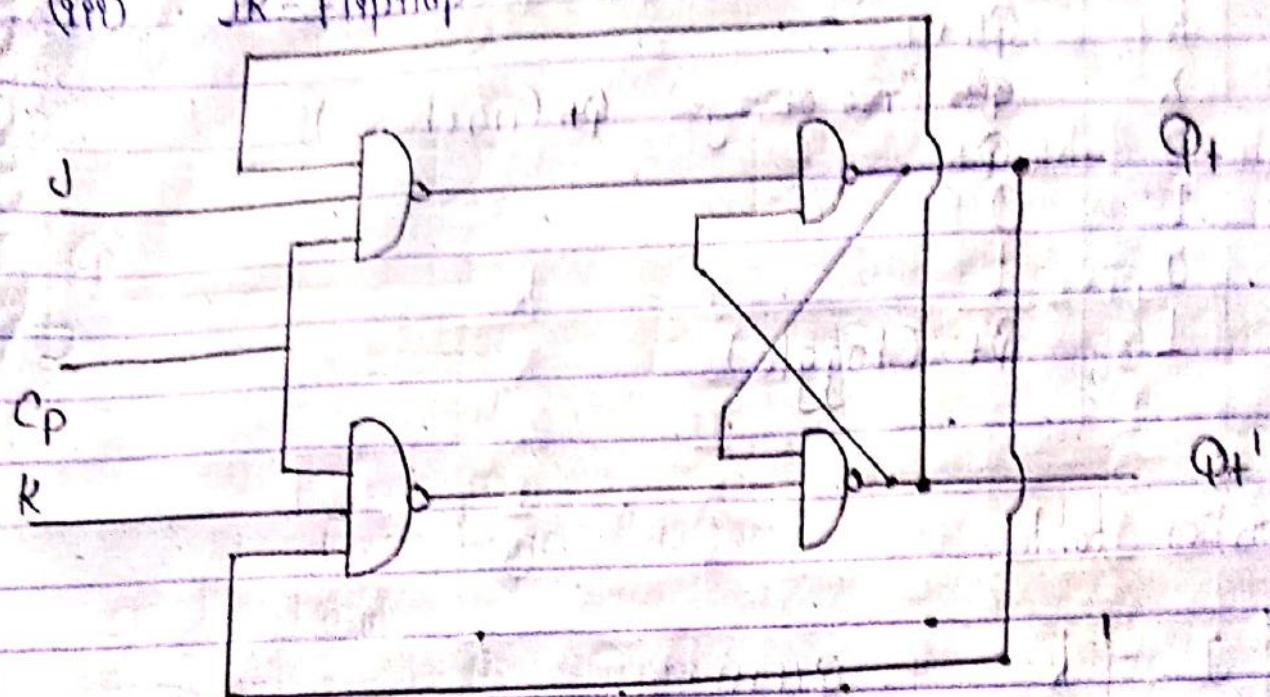


Fig: logic diagram



Fig: block diagram

- It is refinement of RS-flipflop as the indeterminate state of RS is defined as 'Toggle' in JK.
- In J-K flipflop input J and K behave like inputs 'S' and 'R' to set and reset the flipflops.

Truth table

J	K	Q_{t+1}	Q_t
0	X	X	(no change)
0	0	0	(no change)
0	1	1	(no change)
1	0	1	(Toggle)
1	1	0	(Toggle)

Characteristic table

Q_t	J	K	Q_{t+1}
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

K-map for Q_{t+1}

Q_t	00	01	10	11
0	0	0	1	1
1	1	0	0	1

$$Q_{t+1} = J Q_t + K_i Q_t$$

Case 1: When $J=0, K=0$,
Then the flipflop retains previous value that
is $Q(t)$.

Case 2: When $J=0, K=1$, then the flipflop is
reset (~~set~~) also when previous condition of flipflop
is reset (i.e. $Q_t=0$).

Case 3: When $J=0, K=1$ and previous condition of
flipflop is set (i.e. $Q_t=1$), then the flipflop is
in reset state.

Case 4: When $J=1, K=0$ and previous condition
of flipflop is reset (i.e. $Q_t=0$), then the flipflop is
in set state. on the application of control input.

Case 5: When $J=1, K=0$ and previous condition
of flipflop is set (i.e. $Q_t=1$) then, the flipflop
will be in previous state. (Set State).

Case 6: When $J=1, K=1$ and previous state is set ($Q_t=1$).
Then the flipflop toggle from Set to Reset
state.

Also, if $J=1, K=1$ and previous state is

reset i.e. $Q_t = 0$ then the flip-flop toggles from Reset to Set state.

Excitation table

Q_t	Q_{t+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

K-map for J

Q_t	0	1
0	0	1
1	X	X

$$J = Q_t + 1$$

K-map for K

Q_t	0	1
0	X	X
1	1	0

$$K = \overline{Q_{t+1}}$$

(iv) T flip flop

The T flip flop is a sym. Single input version of JK flip flop.

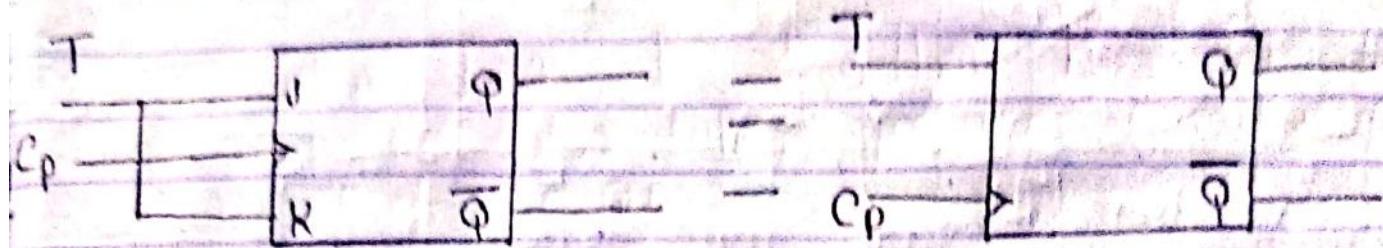


Fig: Logic symbol

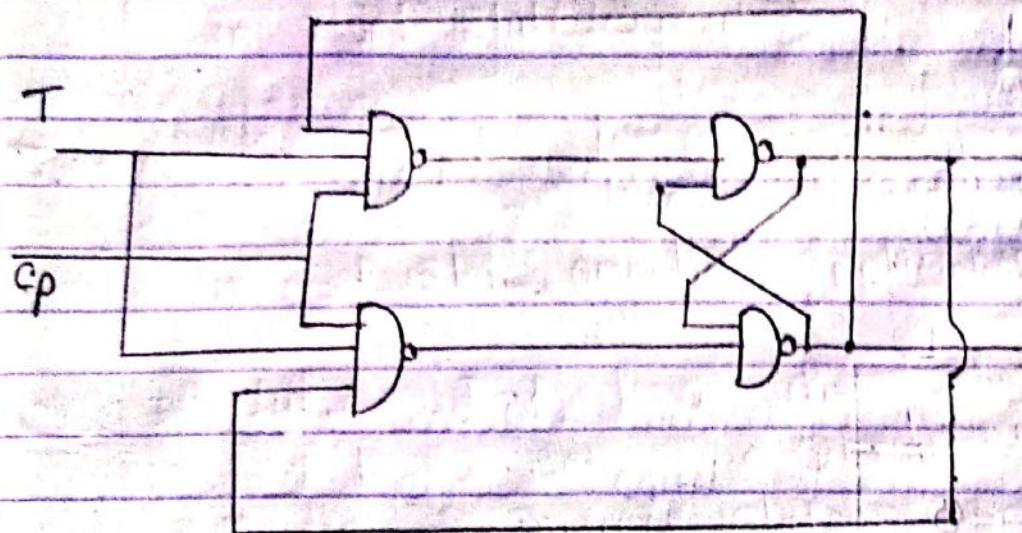


Fig: Logic diagram of T flip flop

- In this flf when $T=0$, the flf retains the present state.
- When $T=1$, complement of present state is obtained in the next state i.e. the state is toggled.

Truth table

C_p	T	$Q(t+1)$
0	X	MEMORY $Q(t)$
1	0	$Q(t)$
1	1	$\bar{Q}(t)$

characteristic table

$Q(t)$	T	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

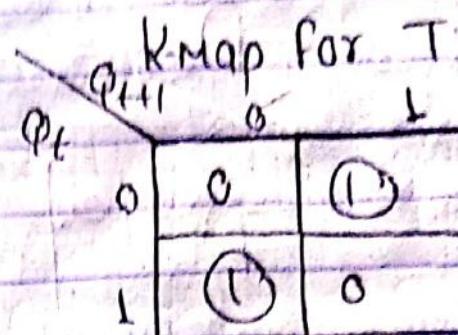
K-map for $Q(t+1)$

$Q_t \backslash T$	0	1	?
0	0	1	
1	1	0	

$$Q(t+1) = Q_t T' + Q_t' T \text{ is characteristic equation}$$

Exciation table

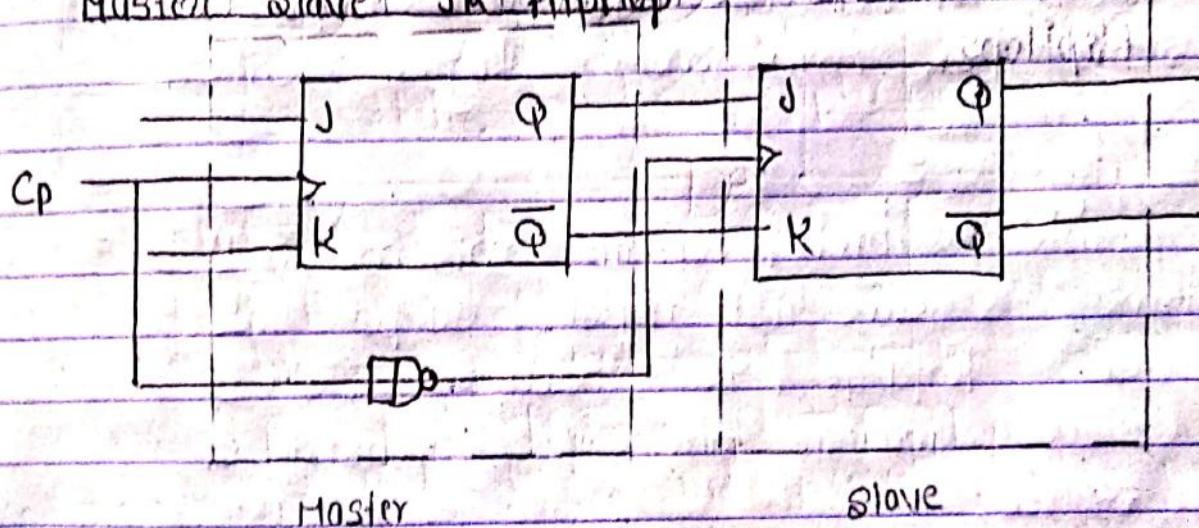
Q_t	$Q_{t+1} \text{ L}$	T
0	0	0
0	1	1
1	0	1
1	1	0

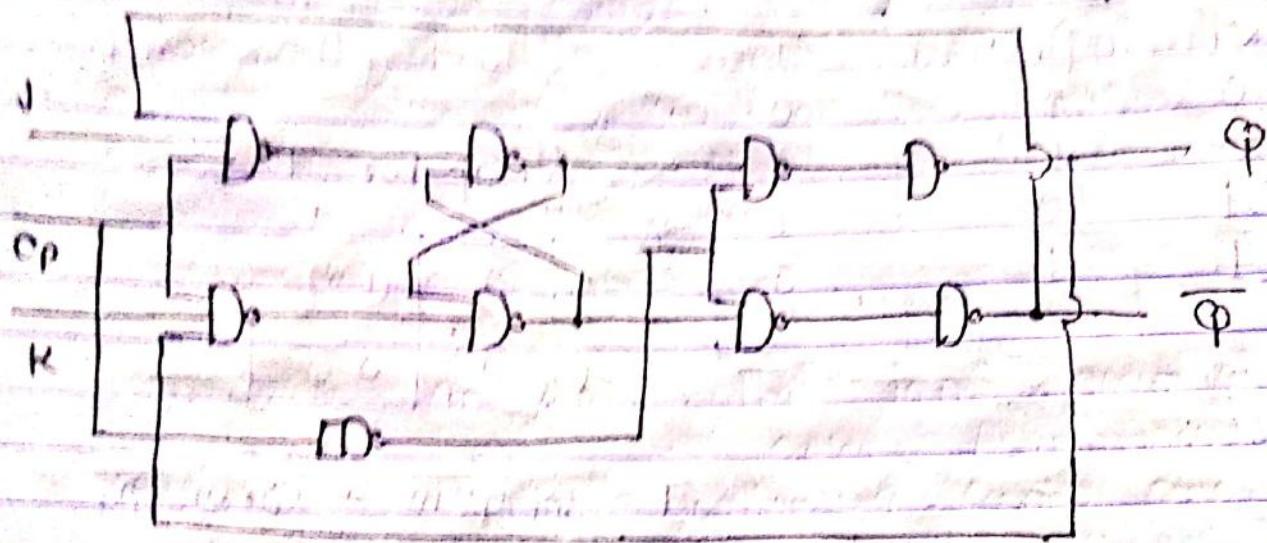


$$T = Q_t \overline{Q_{t+1} + D} + Q_t + D \overline{Q_t}$$

$$= Q_t + \overline{Q_t + D}$$

Mastor slave Jk flipflop





Logic diagram

A master-slave D/F is constructed from two separate flipflops.

Analysing Sequential Circuits

The behaviour of the sequential circuit is determined from its input, output and the state of flipflop.

The output and the next state are both the function of its input and its present state. The circuit consists of one type of flipflop and logic gates for combination logic circuit.

The analysis of sequential ckt is to obtain table or diagram for the time sequence of inputs outputs.

- a) Initial states.
- b) Representing table with diagram.
- c) Describing Boolean Expressions.
- d) Realizing expression using flipflops and combinational circuit.

Sequential Logic Design.

There are two types of sequential logic.
In synchronous logic flipflops are triggered by a common phase (cp). The common phase triggers all the flipflops simultaneously rather than one at a time in succession.

e.g: Ripple Counter.

Similarly an event that does not occur at the same time is called Asynchronous. Therefore flipflops within a counter can be made to change their states exactly at the same time.

Flipflops are triggered by triggers. They are independent of clock signal.
e.g → D latch or Asynchronous logical functions.

State Table

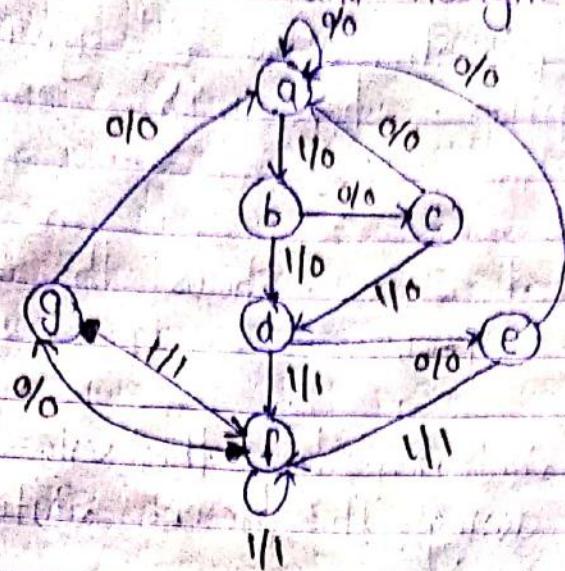
The table which contains information about the time sequence of input, output and state of flipflop is known as state table, (state analysis table). It consist of three section;

- a) present state → The present state designates the state of flip flop before the occurrence of clock pulse.
- b) next state → The next state shows the state of flip flop after the application of a clock pulse.
- c) output section → Output section list the values of the output variables during the present state.

State Diagram

The information available in a state table can be available in a state table can be represented graphically in the form of state diagram (or state transition diagram). In State Diagram, a state is represented by a circle and the transition between the states is indicated by directed lines connecting the circles.

State reduction and Assignment;



State diagram

Present state (PS)	Next state (NS)		Output (Y)	
	$x=0$	$x=1$	$x=0$	$x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f/d	0	1
e	a	f/d	0	1
f/d	g/e	f/d	0	1
g/e	a	f	0	1

present state e and g have same nextstate and output. So, one can be discarded.

Here we replace g by e.
 $e = g$.

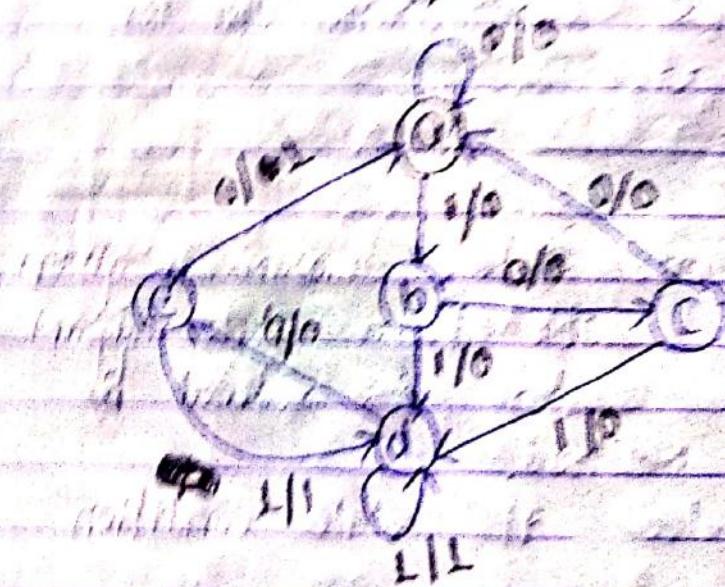
After being ticked all transitions will be made
from 0 to 1 and the only path leading to
1 is from 0.

So we get 000 as initial state.

	0	1	2	3
0	0	1	2	3
1	0	1	2	3
2	0	1	2	3
3	0	1	2	3

Hence State 0 is useless.

Dashed State means 000.



Two states are equivalent if each member of inputs give same outputs. When two states are equivalent one of them can be removed.

As it has been seen from above example State reduction is a technique to reduce the number of States in State table while keeping the input/output same. State reduction technique reduces number of flipflops, gates which will reduce the cost of design of final CLK.

State Assignment

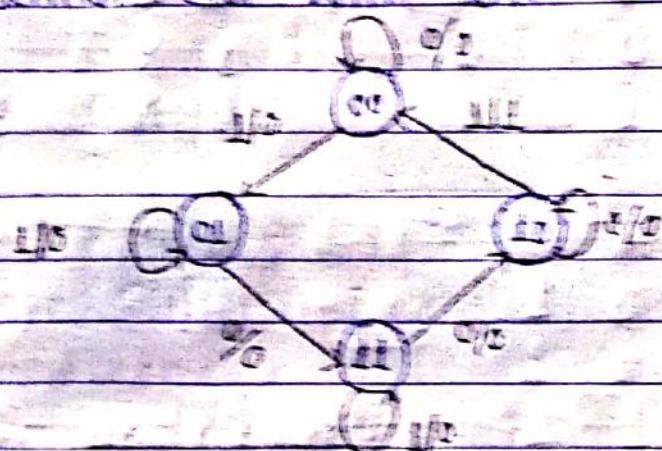
State Assignment is to represent the state variable in terms of binary value.

It is helpful in minimizing the number of combinational gates and to reduce the cost of combinational CLK that drives for the flipflop. Assignment is chosen to produce simple combinational CLK for the flipflop input.

Design procedure

- 1) CLK behaviour and state diagram is given.
- 2) Obtain the state table from state diagram.
- 3) The number of States is reduced by state reduction Method.
- 4) Assign binary value of state condition is symbolic.
- 5) choose the type of flipflop.

- ii) For the state table done the output and
enable table according to the following case
- iii) Find the complement of functions using the
method of other substitution method.
- iv) Draw logic diagram.
- v) For the given state diagram design a
sequential circuit with 8085



State table

Q1	Q2	Q3	Q1'Q2'Q3'	Q1Q2'Q3'	Q1Q2Q3'	Q1Q2Q3	S	V
0	0	0	0 0 0	0 1 0	1 0 0	1 1 0	0	0
0	1	1	1 1 1	0 1 1	0 0 1	0 0 0	0	0
1	0	1	1 0 0	0 0 0	0 0 0	0 0 0	1	1
1	1	0	0 1 1	1 1 1	1 0 1	0 1 0	0	0

Excitation table

Q_A	Q_B	X	Q_A^N	Q_B^N	Output	R_A	SRA	P_F	$11ps$	P_B	SRA
0	0	0	0	0	0	X	0			X	0
0	0	1	0	1	0	X	0	0	0	X	1
0	1	0	1	1	0	0	1	0	0	X	
0	1	1	0	1	0	X	0	0	X	X	
1	0	0	1	0	0	0	X	X	X	0	
1	0	1	0	0	1	0	1	0	0	X	
1	1	0	1	0	0	0	X	0	1	0	
1	1	1	1	1	0	0	0	X	0	X	

$$Y = Q_A Q_B' X$$

K-Map for R_A

$Q_B X$	00	01	10	11	
Q_A	0	X	(X)	X	0
	1	0	(1)	0	0

K-Map for R_B

$Q_B X$	00	01	11	10	
Q_A	0	X	0	0	0
	1	(X)	X	0	(1)

$$R_A = Q_B X$$

$$R_B = Q_A X' 1$$

K-Map for S_A

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

K-Map for S_B

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

$$S_A = Q_B \times 1$$

$$S_B = Q_A' X$$

C_P

Logic diagram

