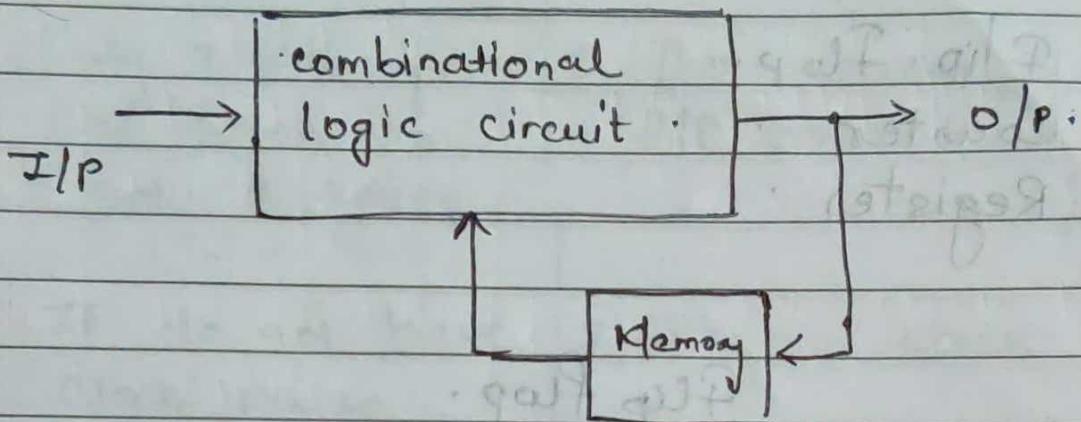


To give clock signal = Pulse Triggering Method
= High level → +ve edge,
= low level. → -ve.

UNIT: 4

Sequential data logic circuit



Classification of Sequential

Synchronous

Asynchronous

→ The memory elements → It is unclocked f/f are clocked f/f.

or Time delay element

→ If I/P change then it affect the clock Signal of memory element

→ Slower.

→ not difficult.

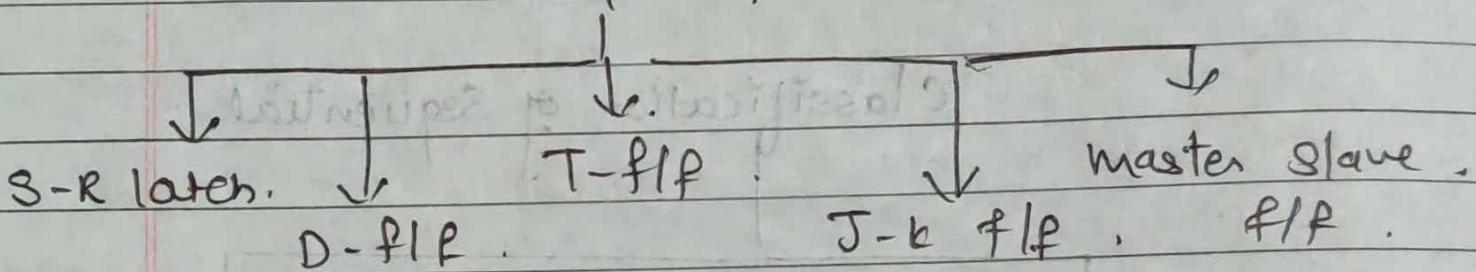
→ Faster operation as no clock is used.

→ difficult to design.

Types of Sequential

- flip flop
- counter
- Register

Flip flop



Counter

Asynchronous
counter

Synchronous
Counter

Register

→ shift Register

→ Ring Counter

→ Ripple Counter

MOD - 4, 8, 10 Counter

Latches :-

1. Only 1 latch i.e S-R latch.
= Set & Reset

2. It do not have clock pulse.

3. The I/p of 1 bit data is stored in latch.

4. Latch can change its o/p immediately based on applied I/p.

Flip flop

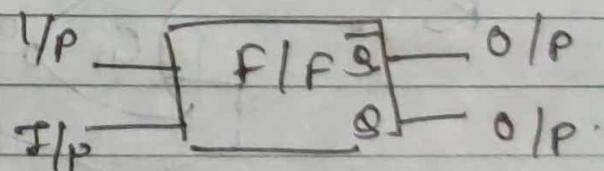
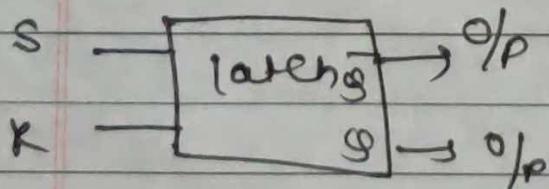
1. There many types of F/F available.

2. It has clock pulse.

3. 1 bit data is stored.

4. It is both clocked & unclocked.

Clocked F/F = Synch.
unclocked F/F = Async.



→ Disadvantage:- It is must to know which gate latch we are using.

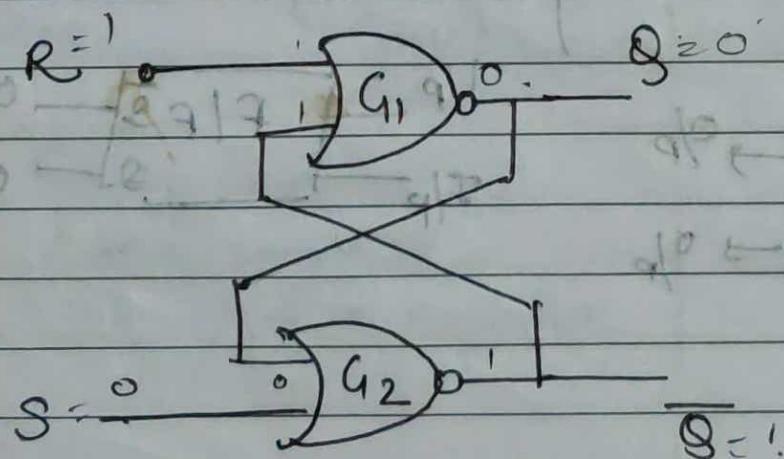
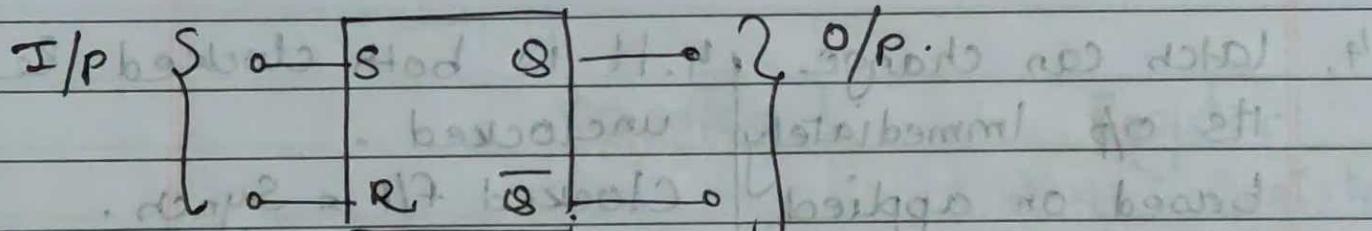
S-R latch

- $S = \text{Set}$. $R = \text{Reset}$.
- It has 2 I/p = S, R .
- It has 2 O/p = $Q_n \text{ & } \bar{Q}_{n+1}$.
- It Responds to level $Q = \text{High} = 1$.
 $= \text{Low} = 0$.

$\bar{Q}_n = \text{Compliment of } Q$.

- It can be Implemented by NOR & NAND.

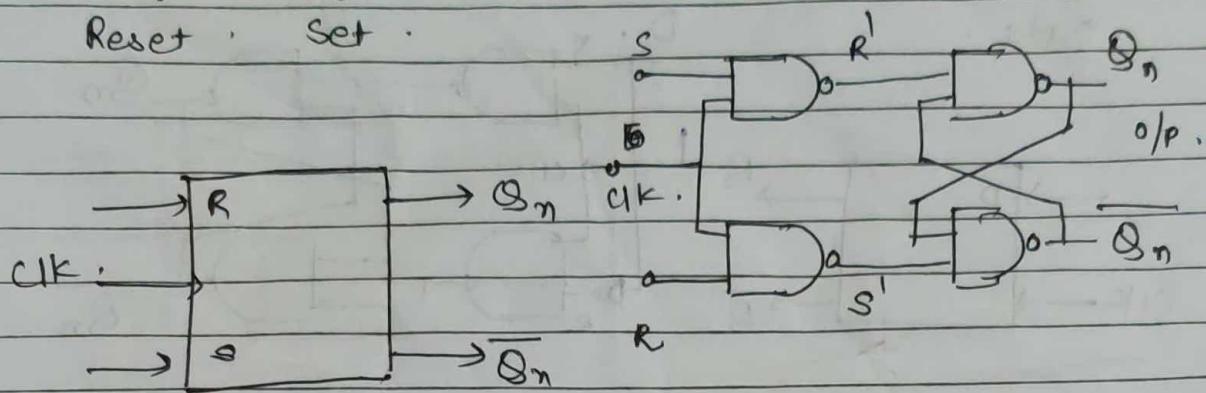
* Block Diagram . — Using NOR gate .



logic diagram .

R-S flip flop

Reset Set



CLK. S R. Qn.

0 x x Qn. → Memory state / No. change.

1 0 0 Qn → Memory state / No. change.

1 0 1 0. → Reset Condition.

1 1 0 1 → Set Condition.

1 1 1 Race → Avoid

condition.

$R = S = 1$, this case is not allowed.

When $S = 1$

$R = 0$, the gate having 1 will become active.

So, $S = 1$ Active first

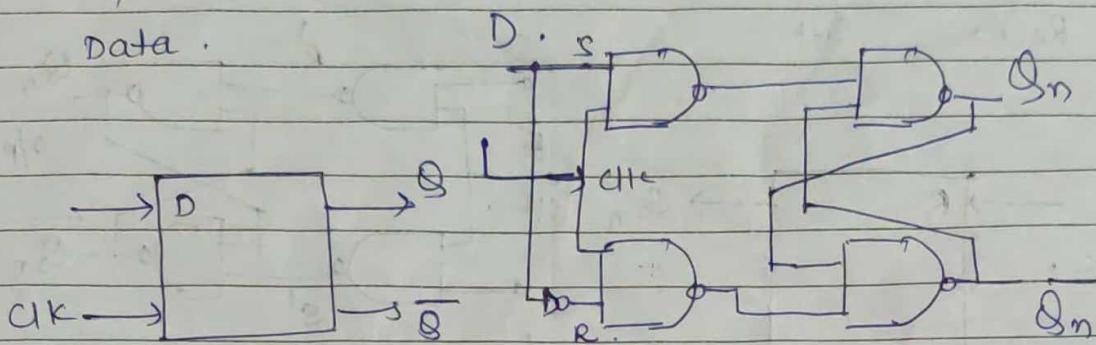
When $S = 0$

$R = 1$, R will become active.

D flip flop. = Delay flip flop.



Data .



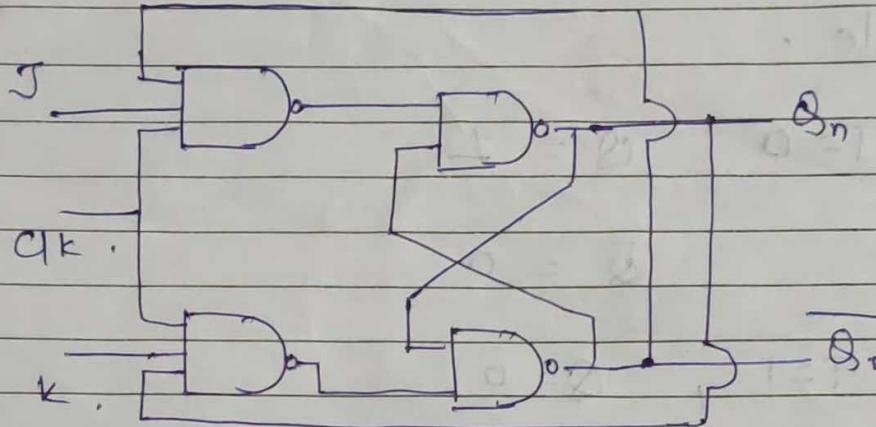
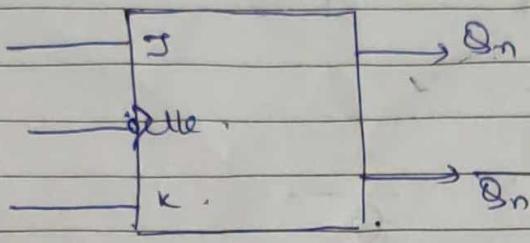
clk.	D.	Q_n .	
0.	x	Q_n .	Memory state .
1	0	0.	Reset . Condition .
1	1	1.	Set . Condition .

→ The two IP of S & R is combined together .

→ Not gate is Connected before R to toggle the input from $0 \rightarrow 1$ or $1 \rightarrow 0$.

→ The gate having 1 will Operate first .

J. k. flip flop



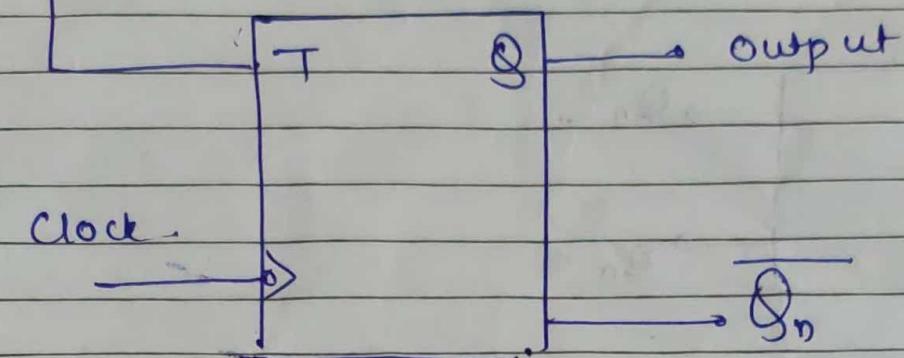
Clk.	J	K	Q_n	
0	x	x	Q_n	NC
1	0	0	Q_n	NC
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	$\overline{Q_n}$	Toggle

→ It is used to overcome the drawback of R & S flip flop.

When $R=S=1$ it do not work,
so when $J=K=1$, it toggle the memory data.

T flip flop.

0/1.



T = Toggle.

when $T = 0$, $Q = 1$

$$\bar{Q} = 0.$$

when $T = 1$, $Q = 0$

$$\bar{Q} = 1.$$

It is used in J.K flip flop.

to work at max of 650 MHz
• Qf Qf 2 3 8

slow tan ab A 1-2=3 minter
product at step f1, 1-3=5 minter
of 1010

(D)

Counter's

- Set of flip flop
- It measure the clock pulse.
- Flip flop are interconnected to get Combined state at any time .
- It Count pulses .
- Used as frequency divider , digital watch frequency counter etc.

Types of Counter's

(best notes)

① Asynchronous Counter :-

- Ripple Counter
 - up Counter
 - Down Counter
 - up/ Down Counter
 - MOD Counter's
- } Using T flip flop.

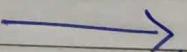
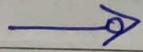
② Synchronous Counter :-

- up/ Down Counter
 - up Counter
 - Down Counter
 - mod Counter
 - BCD Counter
- } Using J-K flip flop

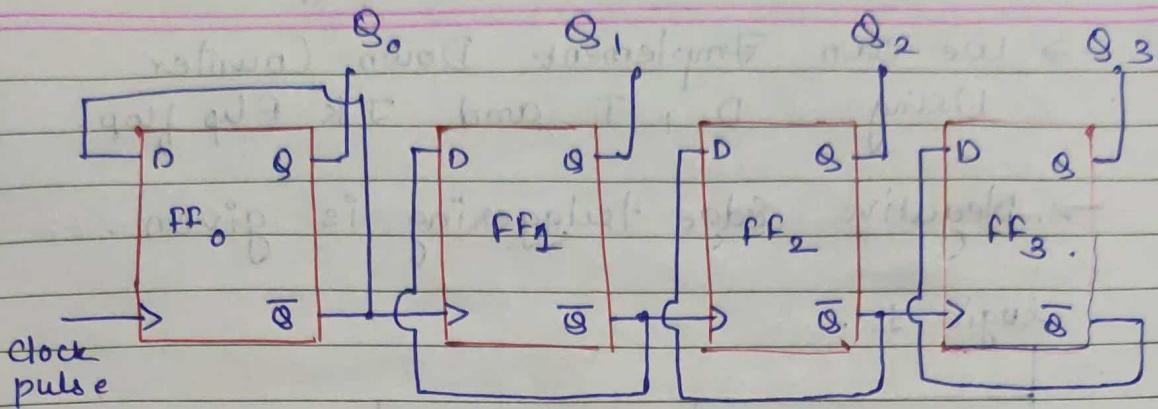
Async Counter can be provided positive edge trigger (clock pulse) or Negative Edge trigger (clock pulse).

Clock pulse:-

Negative edge Trigger positive Edge Trigger



Asynchronous 4-Bit UP Counter

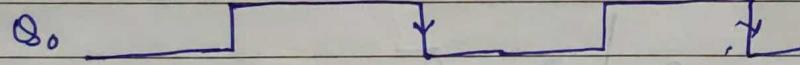
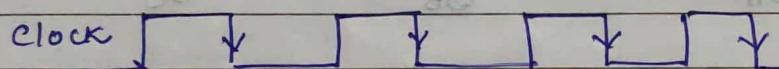


\Rightarrow D-flip flop is used.

\Rightarrow Clock pulse is applied to 1st flip flop in Async.

\Rightarrow Other flip flop are received \bar{Q} as clock pulse

\Rightarrow .



Q ₁	0	1	1	1	0	0	0	0	1	1	0	1	0	0	0	1	1	0	0	0
Q ₀	0	1	1	1	0	0	0	0	1	1	0	1	0	0	0	1	1	0	0	0
	0	1	1	1	0	0	0	0	1	1	0	1	0	0	0	1	1	0	0	0
	0	0	1	1	0	0	0	0	1	1	0	1	0	0	0	1	1	0	0	0
	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0	1	1	0	0	0
	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	1	1	0	0	0
	1	0	0	0	0	1	0	0	1	1	0	1	1	0	0	1	1	0	0	0
	0	0	0	1	1	0	0	0	1	1	0	1	1	0	0	1	1	0	0	0

Shows pulse does not occur during setup and hold time
Setup time of half adder must be less than or equal to 1

Setup time of half adder must be less than or equal to 1

$$\text{Setup} = 5\% \quad \text{Hold} = 3\%$$

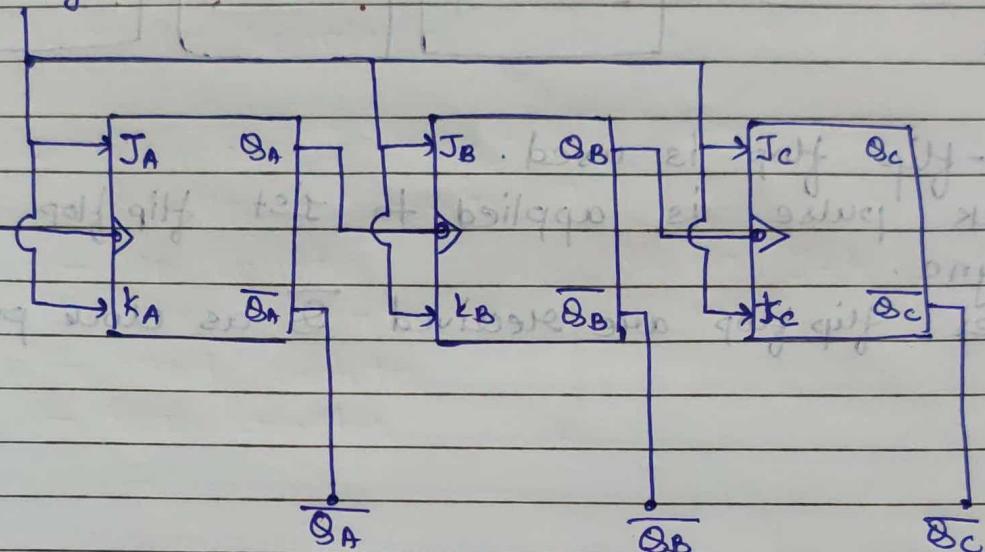
Asynchronous Down Counter

(3 Bit) = 8 Input.

→ We can Implement Down Counter
Using D, T and JK flip flop.

→ Negative Edge triggering is given.

logic 1.



State Table :-

Clock	Q_A	Q_B	Q_C	\bar{Q}_A	\bar{Q}_B	\bar{Q}_C
initial	0	0	0	1	1	1
1 st	0	0	1	1	1	0
2 nd	0	1	0	1	0	1
3 rd	0	1	1	1	0	0
4 th	1	0	0	0	1	1
5 th	1	0	1	0	1	0
6 th	1	1	0	0	0	1
7 th	1	1	1	0	0	0

→ The clock pulse having 50% duty cycle
is given to 1st flip flop.

→ The output Q of every flip flop is used
as clock pulse to next flip flop.

→ $Q_A = \text{LSB}$ / $Q_C = \text{MSB}$.

Counters



Async Down Counter:-

→ If we take 2 bit counter, it requires 2 flip flop.

2 Bit Counter = 2 flip flop

3 Bit Counter = 3 flip flop

4 Bit Counter = 4 flip flop.

→ We can also consider Q_A , Q_B and Q_C as output and take $\overline{Q_A}$, $\overline{Q_B}$ and $\overline{Q_C}$ as clock to next flip flop.

→ In Negative edge triggering ($\rightarrow \overline{\cdot}$) is used where counter state transition occurs only at falling edge of clock pulse.

→ At Initial stage $Q_A = 0$
 $Q_B = 0$
 $Q_C = 0$.

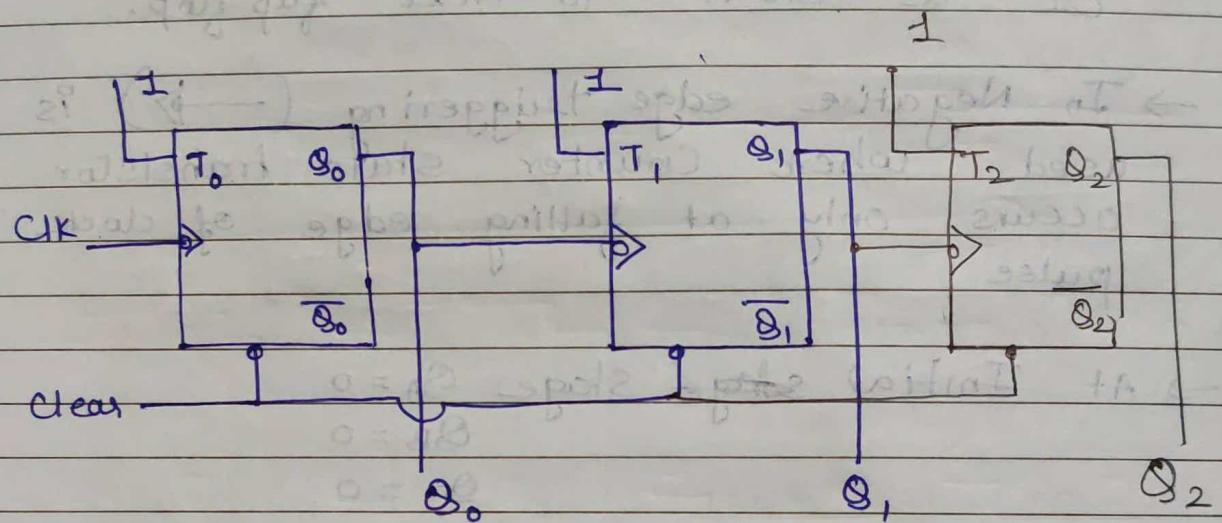
→ It starts counting from 7, 6, 5, ... to 0
In decreament manner

Asynchronous up/Down Counter

→ It can be Implemented Using T, D and JK flip flop.

→ This Counter is Combination of both up and Down Counting of clock pulse in a Single Counter.

→ Implementing Using Negative edge Triggering.
 → let us take example of 3 bit.



→ The Control Input is used to select the Up Counter as well as Down Counter.

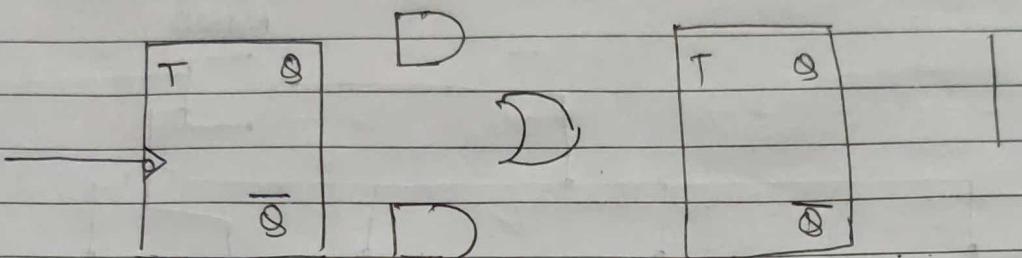
Count up . Count down .

State	Q ₂	Q ₁	Q ₀	State	Q ₂	Q ₁	Q ₀
0	0	0	0	7	1	1	1
1	0	0	1	6	1	1	0
2	0	1	0	5	1	0	1
3	0	1	1	4	1	0	0
4	1	0	0	3	0	1	1
5	1	0	1	2	0	1	0
6	1	1	0	1	0	0	1
7	1	1	1	0	0	0	0

⇒ The up counter selected then it starts counting from 0, 1, 2, ..., 7.

⇒ In down counter, it uses n number of flip flop & starts counting downward i.e. $(2^n - 1)$ to zero,

7, 6, 5, ..., 0.



⇒ When Count-up/Down line = High (1)

Then Lower AND gate = Disabled
And output = zero.

so it won't affect the o/p of OR gate.

⇒ When Count-up/Down line = Low (0).

Then upper AND gate = disabled
lower AND gate = Enabled.

⇒ Then It allow Q_A & Q_B to pass through clock input.

⇒ Thus at this case, Counter count in Down mode.

SYNCHRONOUS COUNTER.

Decade Counter:- OR BCD Counter.

→ It counts from 0 to 9.

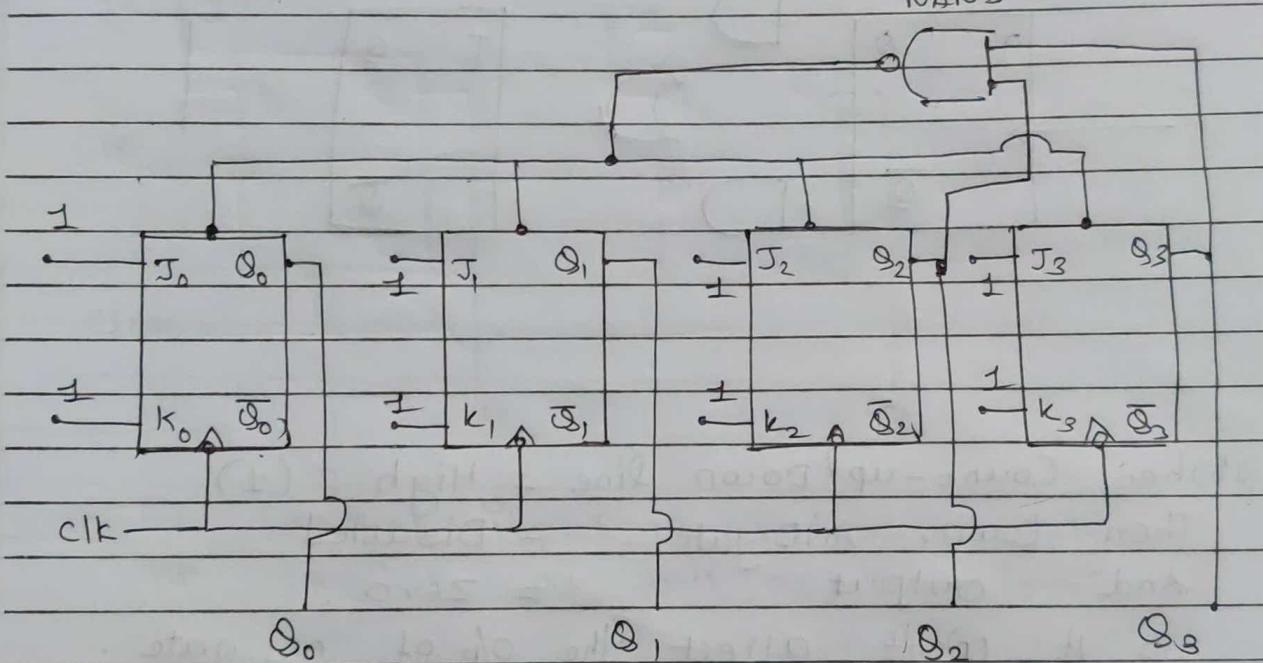
→ I.e. It counts the clock pulse reach upto count of 10.

→ It is a serial digital Counter that counts ten digits.

→ After counting 10, it restart again from 0.

→ 4 Bit Binary Counter will act as decade Counter.

NAND



⇒ It is constructed using JK flip-flop.

J=1, K=1

⇒ The clock pulse is connected to each flip flop.

⇒ The NAND gate connected is a parallel to all input called as clear. CLR.

(3)

Decade Counter Truth Table:-

Input

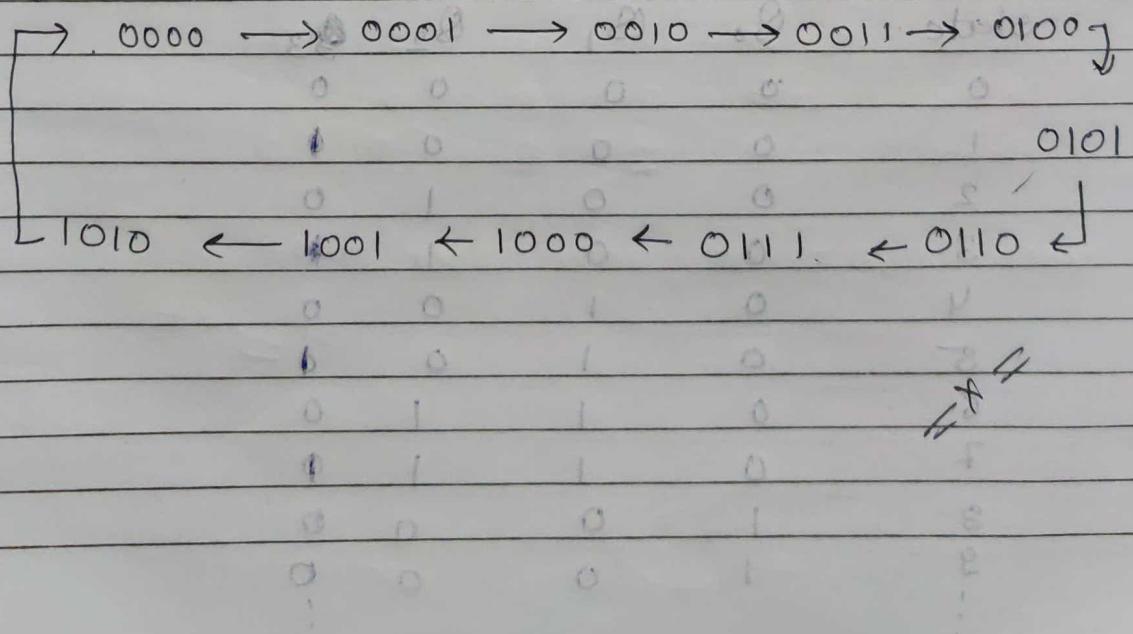
Input pulse	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	0	0	0	0

Restart 0 0 0 0 0

\Rightarrow The NAND gate output is = 0 when Count Reaches 10 = 1010.

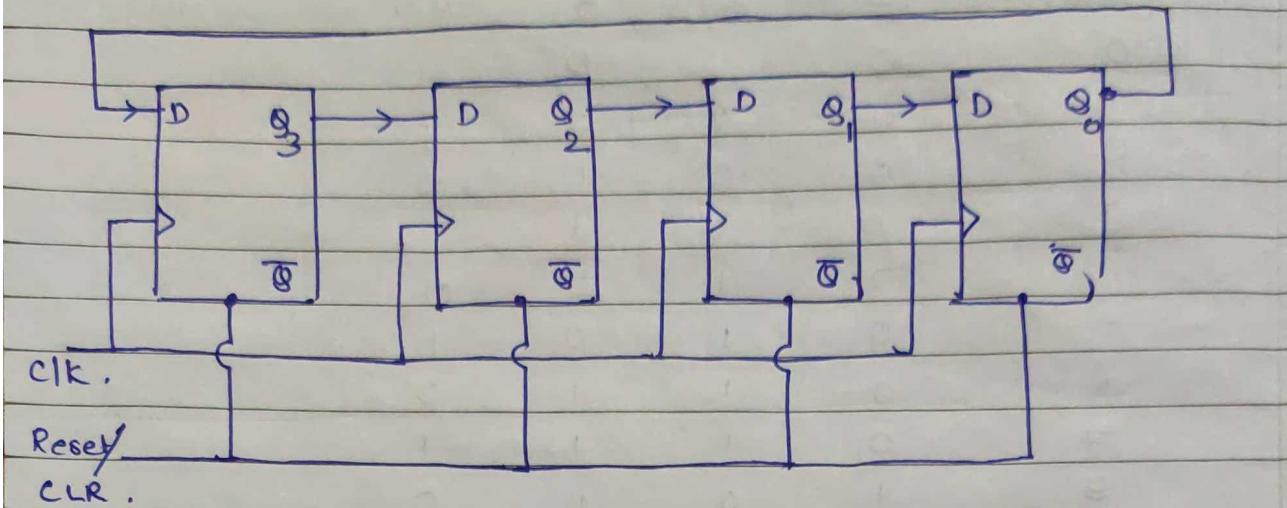
* State Diagram of Decade Counter.

Count



Synchronous Counter

The Johnson Counter OR Modified Ring Counter.



⇒ All flip flop are Cascaded with other.

⇒ The Output of proceeding flip flop is fed back as input to the next flip flop.

⇒ The Inverted Output from last flip flop is Connected to the Input of first flip flop.

⇒ It is Implemented Using D or JK flip flop.

⇒ Also called as Inverted feedback counter.

⇒ It has preset and clear pins.

→

State	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	0
---	---	---	---	---

⋮	⋮	⋮	⋮	⋮
---	---	---	---	---

When clock = 1, O/P = 1000

When clock = 2, O/P = 1100

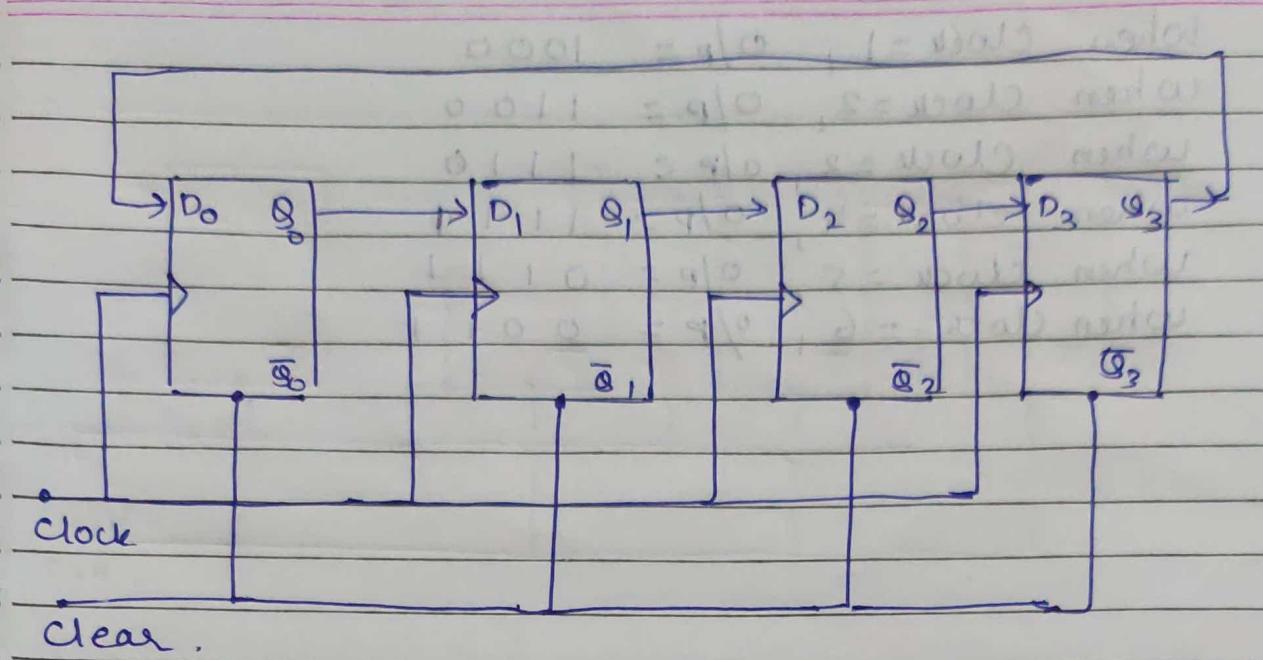
When clock = 3, O/P = 1110

When clock = 4, O/P = 1111

When clock = 5, O/P = 0111

When clock = 6, O/P = 0011

Synchronous Counter Ring Counter



When $PR = 0, Q = 1$
 $CLR = 0, Q = 0$.

The output is 1 when pre-set = 0.

Both preset & clear work in 0 value,
 i.e. active low.

clear. clk. $Q_0 \ Q_1 \ Q_2 \ Q_3$

low	X	1	0	0	0
High	Low	0	1	0	0
High	Low	0	0	1	0
High	Low	0	0	0	1
High.	Low	1	0	0	0

When $CLR = \text{low}$, No clock pulse.

=x=

Shift Register's

(4)

⇒ Register's is a group of flip flop for storing in number of bits more than 1 bit.

⇒ It stores Temporary Data.

Shift Register's

- SISO / serial I/p serial O/p
- SIPO / serial I/p parallel O/p
- PIPO / parallel I/p serial O/p
- PIPO / parallel I/p parallel O/p

⇒ All flip flop are connected in series.

1. Serial In Serial O/p

↳ Input = serial Transmission.
O/p = serial.

2. Serial In parallel O/p.

↳ Input = serial
Output = parallel.

3. Parallel In Serial O/p.

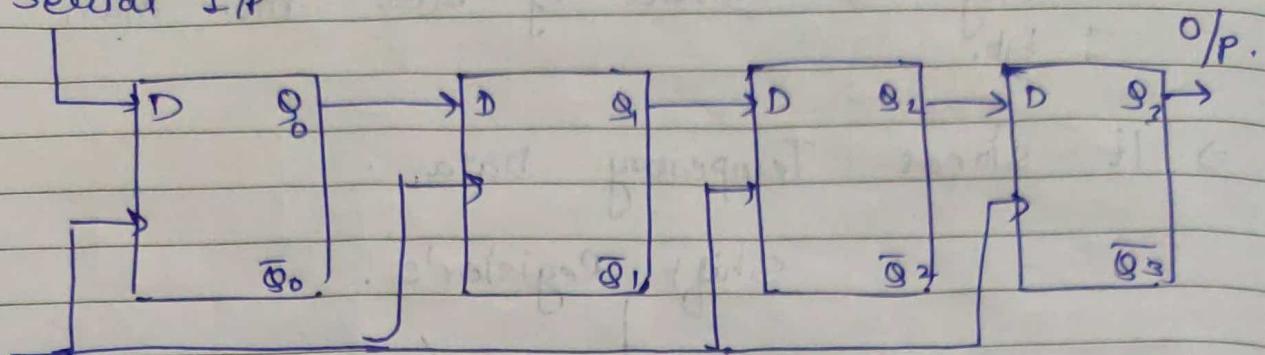
↳ Input = parallel
Output = serial.

4. parallel In parallel O/p.

↳ Input = parallel
Output = parallel.

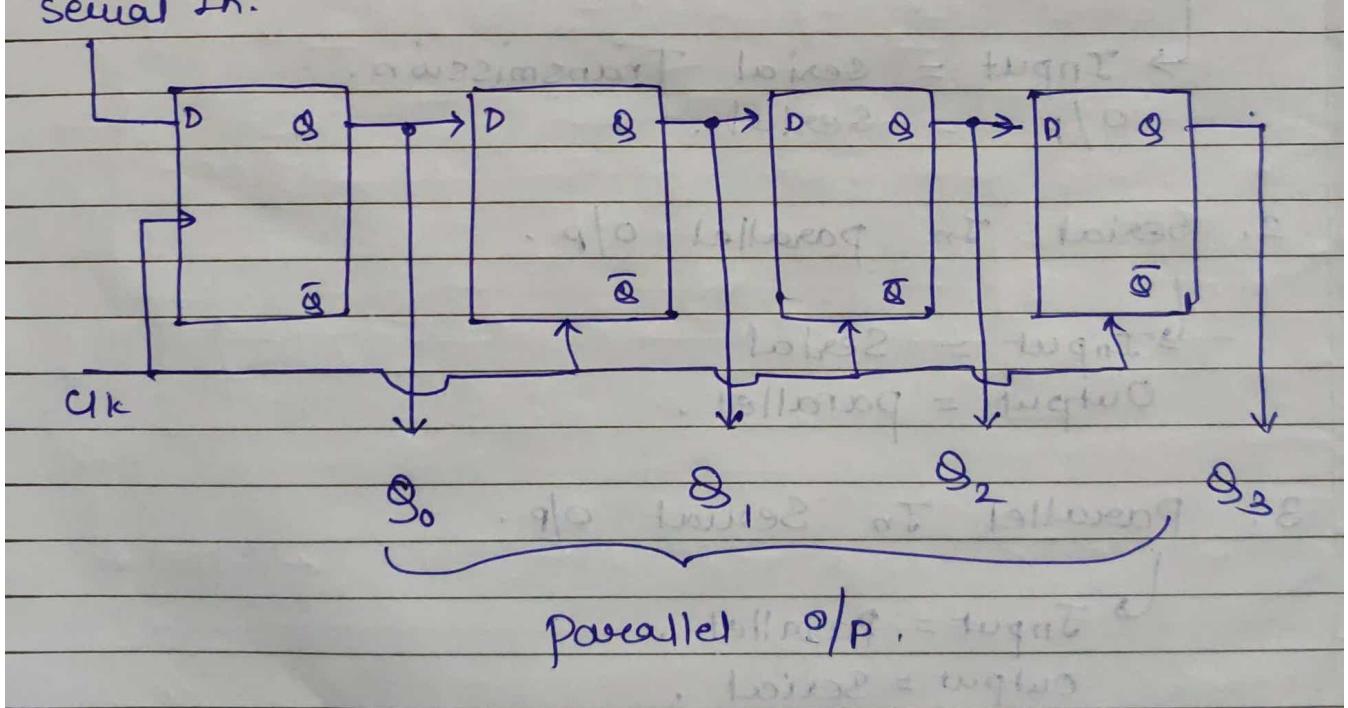
SISO

Serial I/P:

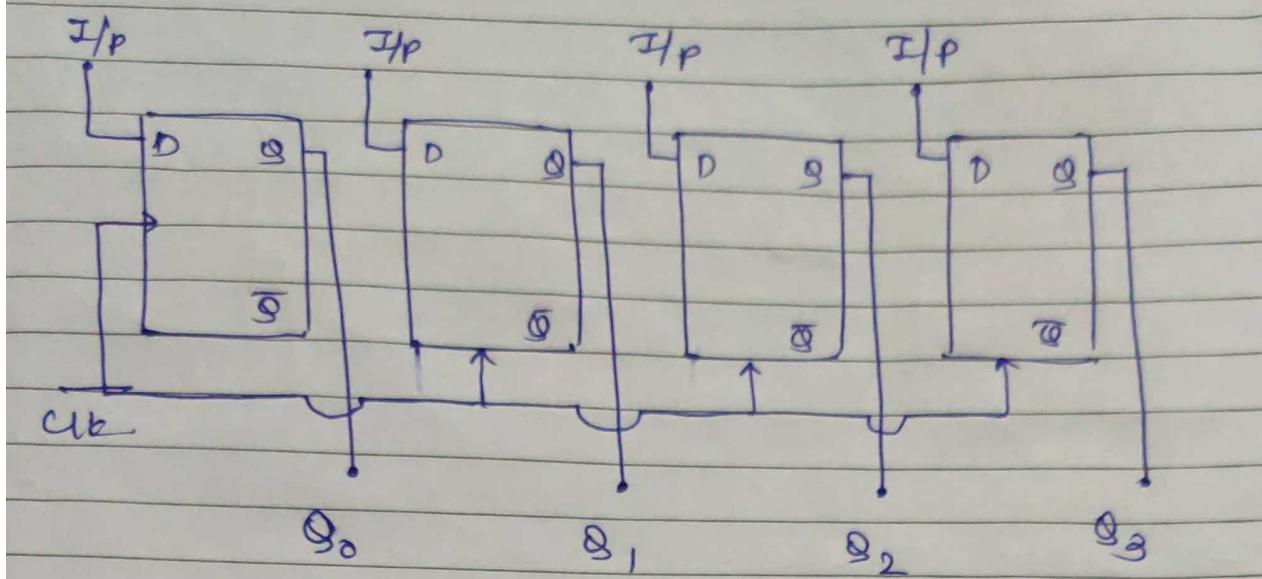


⇒ Delay process

serial In.



PIPO .



= faster process .
[PISO] .

