

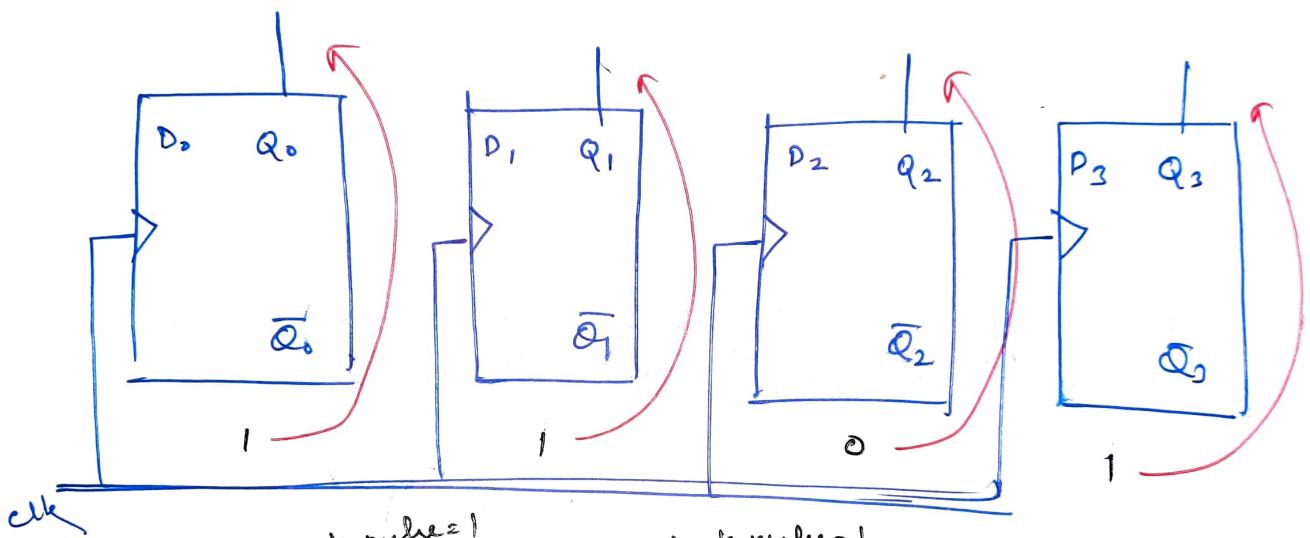
# REGISTERS:

- It is a group of flip flops.
- used for storing information.

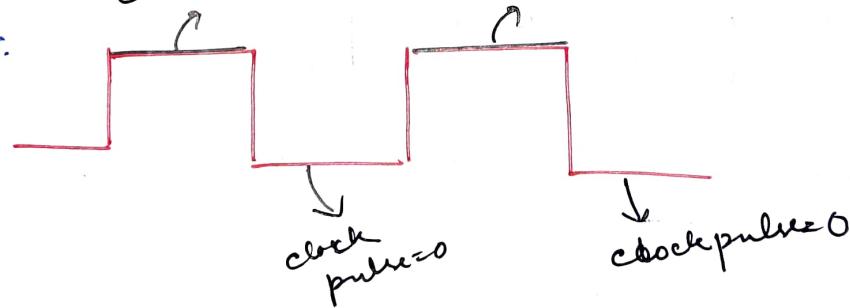
If there are  $n$  bit register, then it will contain  $n$  - flip flops and will store  $n$  - bit words

ex. 4-bit register:

$$\Rightarrow \text{No. of FF's} = 4.$$



Note:



If clock pulse = 1;

then; 1/pulse will move to  $Q_0, Q_1, Q_2, Q_3$ .

ex. 1101. (see. fig)

If clock pulse=0; then, it will remain in the prev. state. ; value will not move.

# Shift Register:

nee know:

latch, FF's store 1 bit.

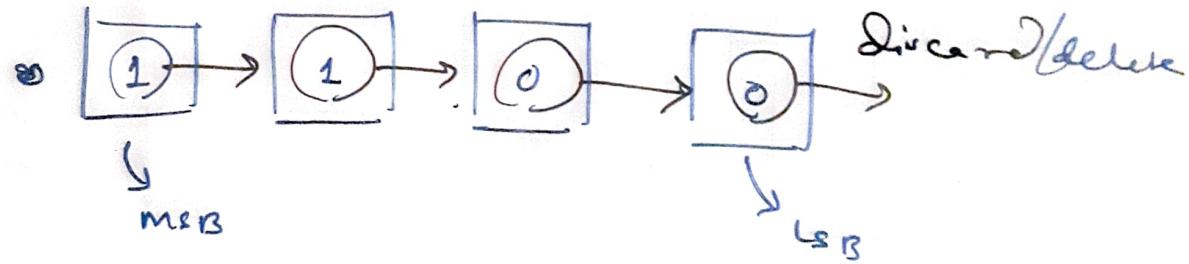
for storing bits ( $> 1$ ), we require shift register.

Types:

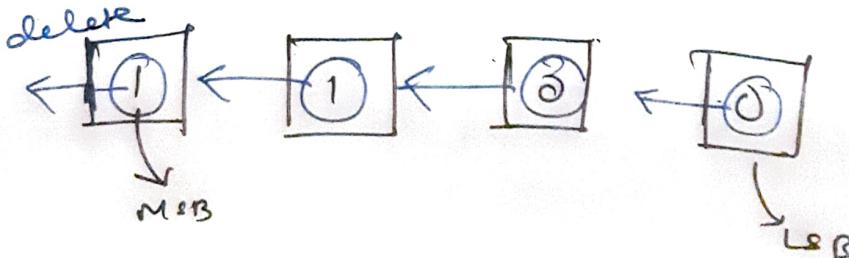
- i) Serial In parallel Out (SISO)
- ii) Serial In parallel Out (SIPO)
- iii) Parallel In Serial Out (PISO)
- iv) Parallel In Parallel Out (PIPO).

## SISO:

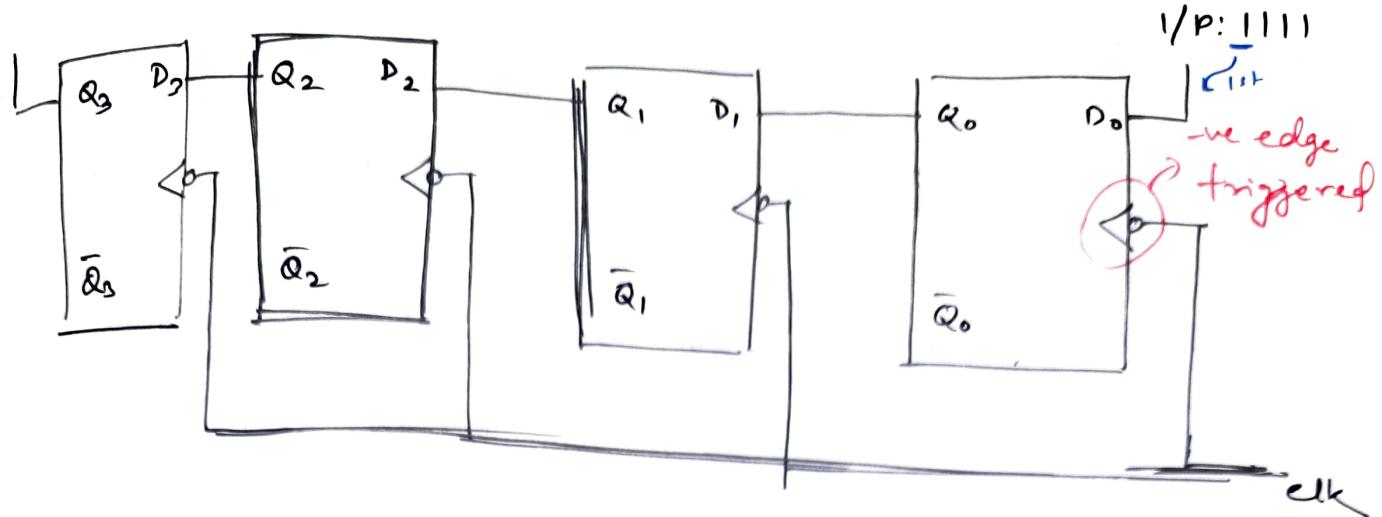
\* Right shift:



\* Left shift



\* Left Shift SISO :

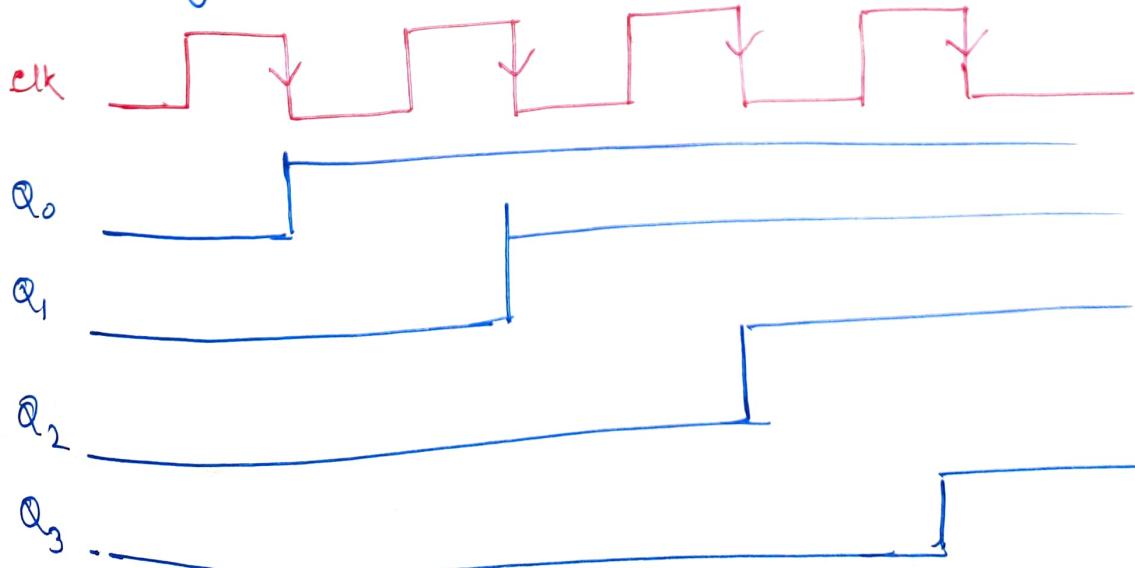


I/P<sub>c</sub>

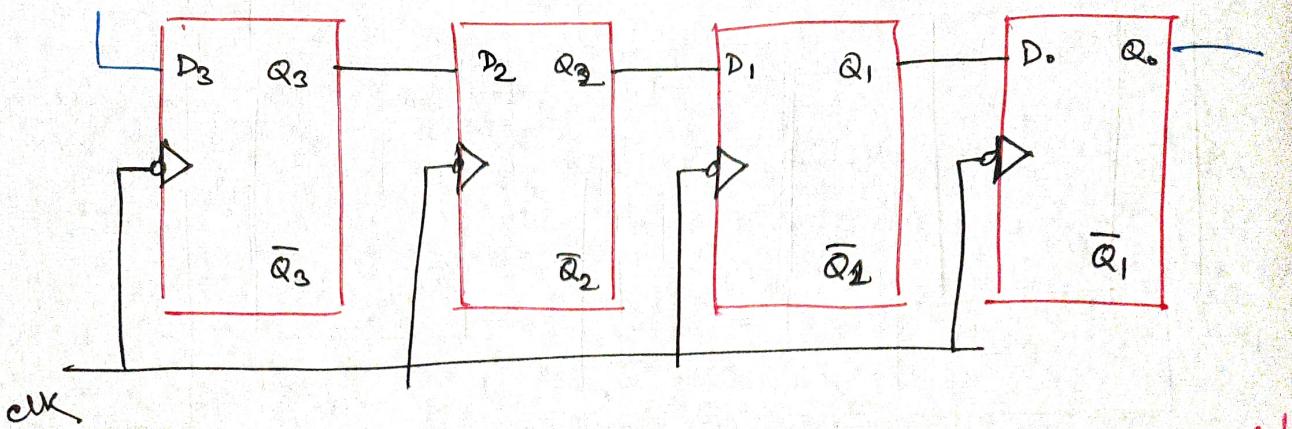
O/P<sub>s</sub>

<u>CP</u>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
2	0	0	1	1	0	0	1	1
3	0	1	1	1	0	1	1	1
4.	1	1	1	1	1	1	1	1

Time diagram:



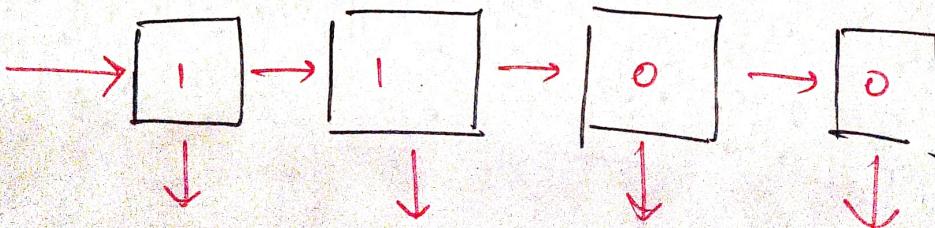
## \* Right Shift SISO:

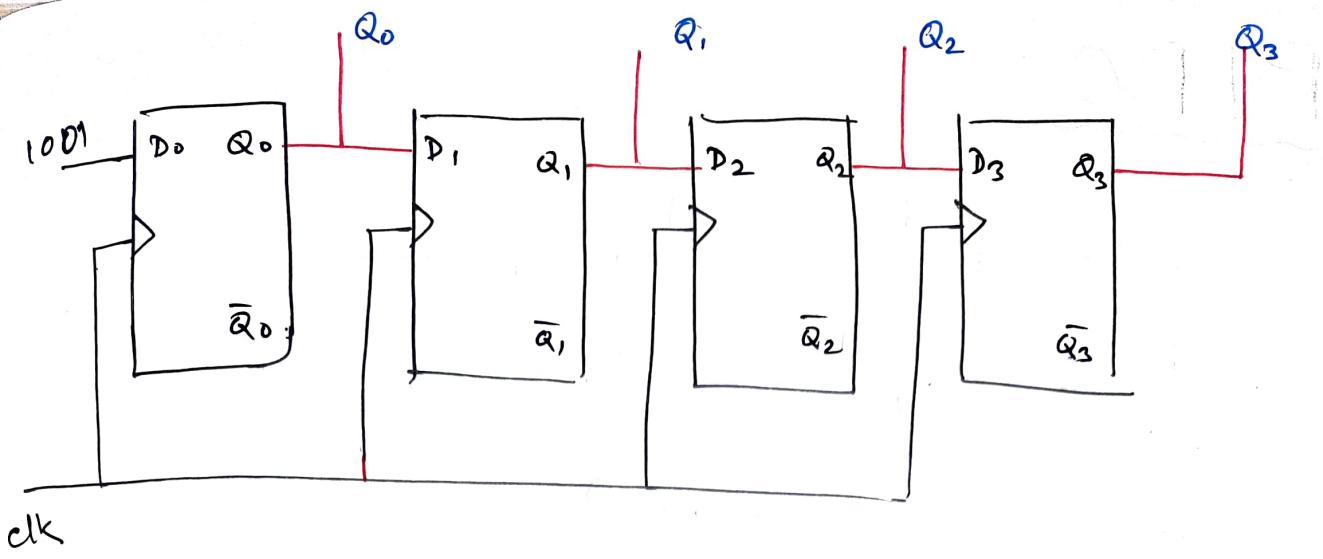


Note:

- \* In a  $n$ -bit shift register,
- No. of clock pulse required =  ~~$n$~~  <sup>to get single bit</sup>.
- To output data serially,
- No. of clock pulse required =  $n-1$
- To get all the bits at the end; No. of CP =  $2n-1$ .

S I P O :





CP	D <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>
	0	0	0	0	0	0	0	0
1	1	0	0	0	1	0	0	0
2	0	1	0	0	0	1	0	0
3	0	0	1	0	0	0	1	0
4	1	0	0	1	1	0	0	1

Note: At the end of 'n' clock pulse; we will get n-bit data 1/Ps; i.e. we are dependent on every IF's to get the O/P.

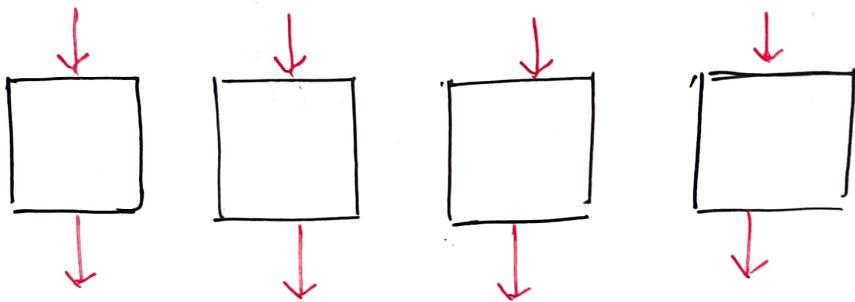
i.e. To get 1 bit data;

No. of clock pulse required = 1.

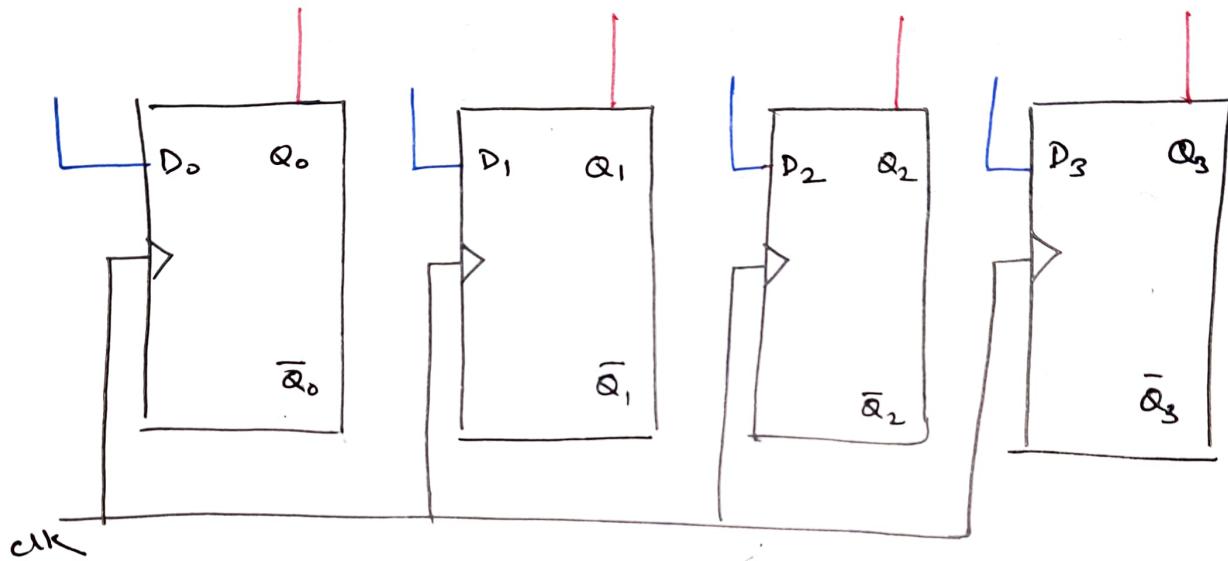
To get n-bit data;

No. of clock pulse required = n.

# PIPO:



(No connection b/w FF's).



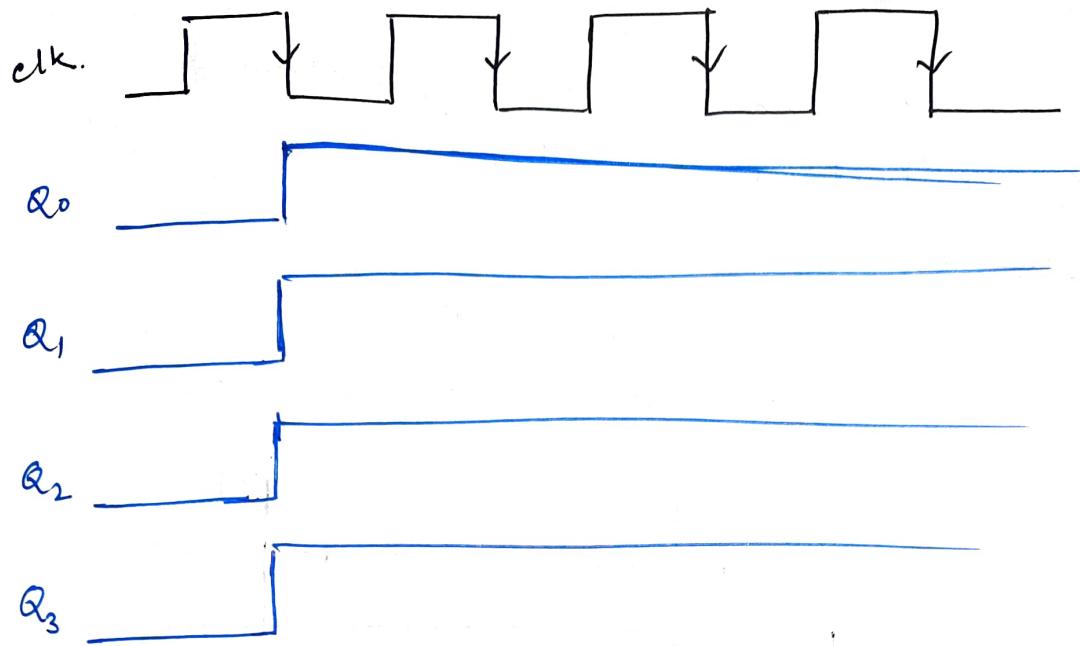
<u>CP</u>	D <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>
0	0	0	0	0	0	0	0	0
1	1	0	1	0	1	0	1	0

Note:

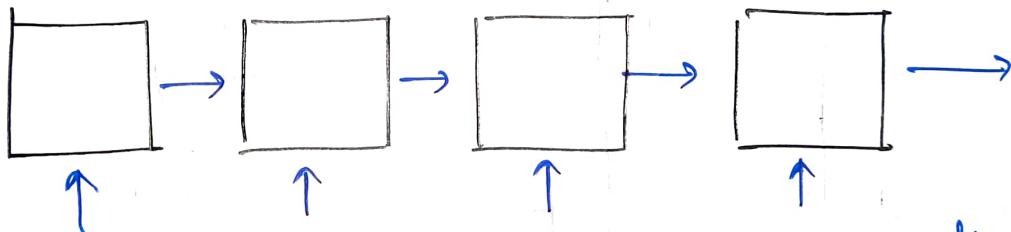
To get 1 bit data; No. of clock pulse required = 1.

To get  $n$  bit data; No. of clock pulse required = 1

Time diagram:



PISO :-

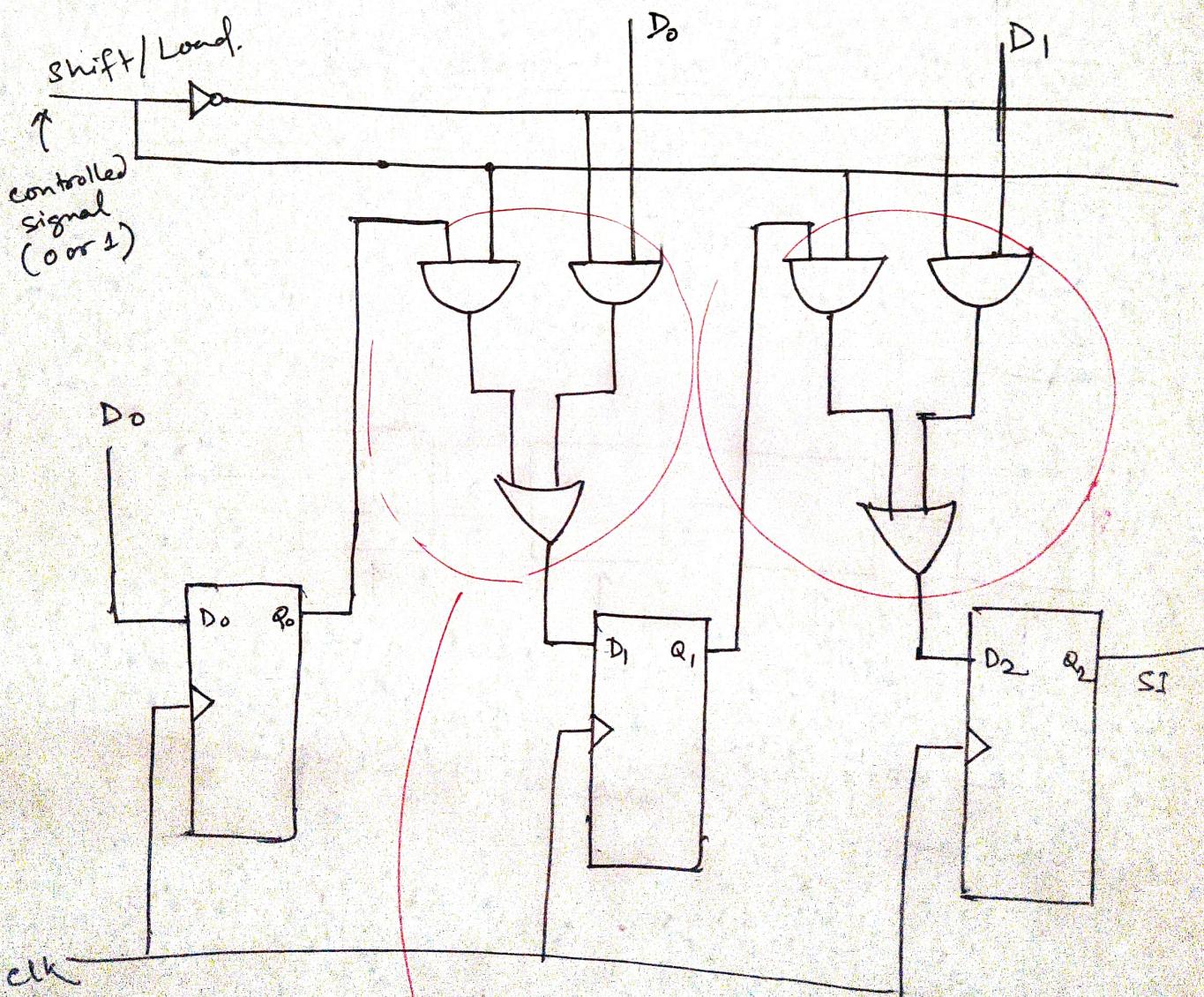


Here; multiple inputs were given at a time;  
but we will give 1 o/p. at a time . So;  
use there will be use of Multiplexers.

\* For parallel in → Loading ✓  
 Serial out → Shifting ✓

during loading: controlled input / controlled signal = 0 and thus it will act as ~~shifting~~ shifting

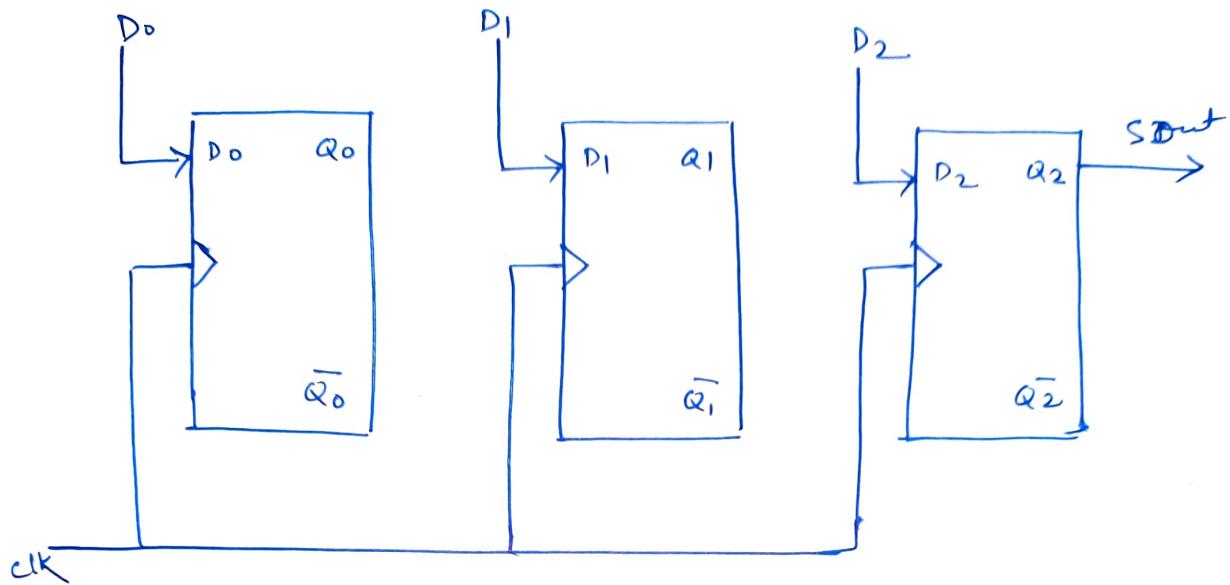
during shifting, controlled input / signal = 1; and then it works similar to Right shift SISO.



*multiplexer*

when controlled signal = 0

↓  
Loading (Parallel In)



when controlled i/p = 1

↓  
shifting (Serial Out)

