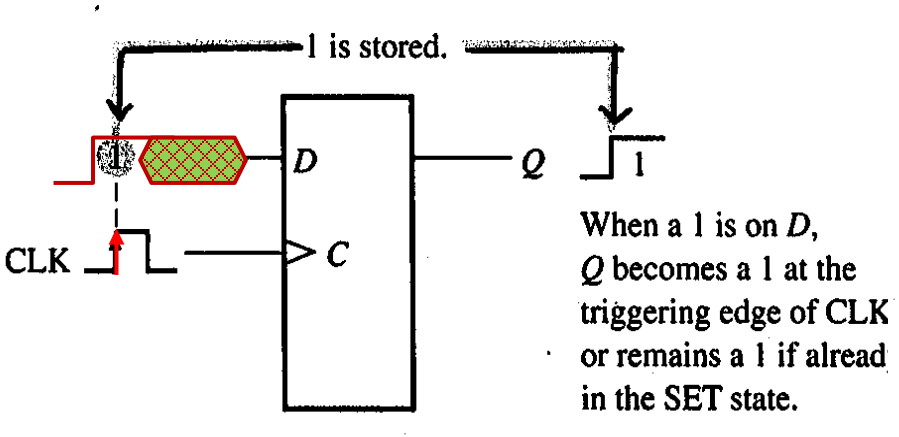
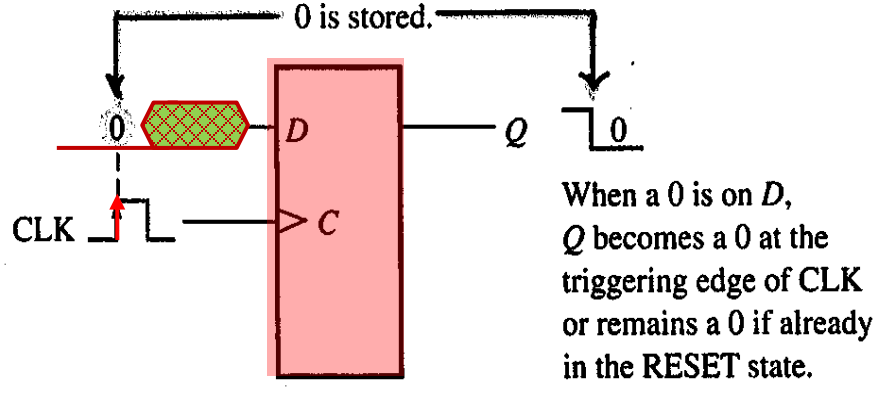
7 REGISTERS

Registers

- A register is a circuit that receives and maintains the data;
- And/or transfers the data to another circuit.
- May be called a memory circuit;
- Made from a number of D Flip-flops;
- Memory Capacity of *n* bits requires *n* Flip-flops.



Set to 1

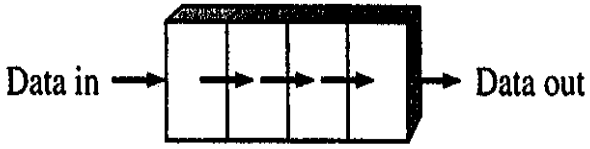


Set to 0

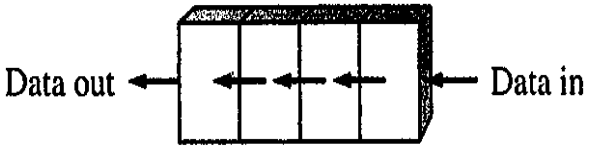
Register Types

Categorized by the method of data import/export:

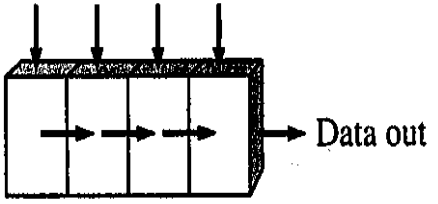
- Serial Data Transfer: Need a number of clocks to trigger the data transfer.
- Parallel Data Transfer: Need only one clock to trigger the transfer.



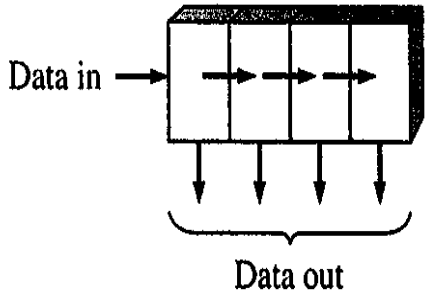
(a) Serial in/shift right/serial out



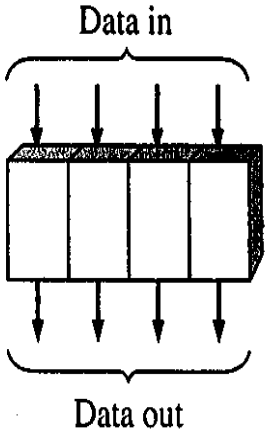
(b) Serial in/shift left/serial out



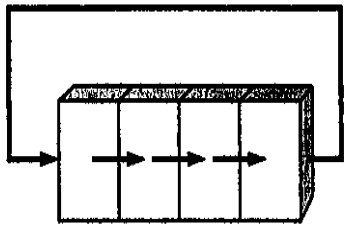
(c) Parallel in/serial out



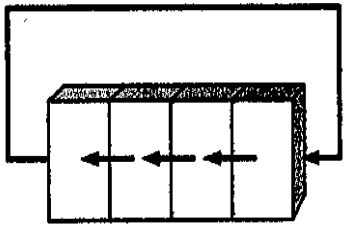
(d) Serial in/parallel out



(e) Parallel in/parallel out



(f) Rotate right



(g) Rotate left

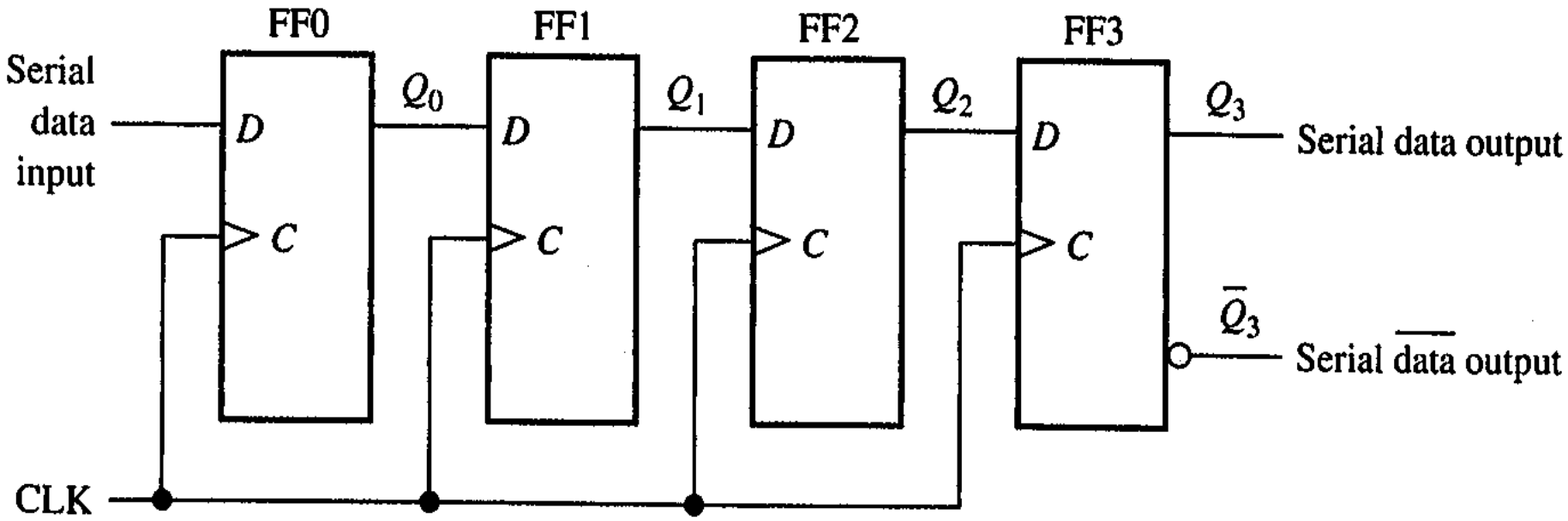
Shift Register

Shift Register transfers data (bit by bit) in serial both input and output;

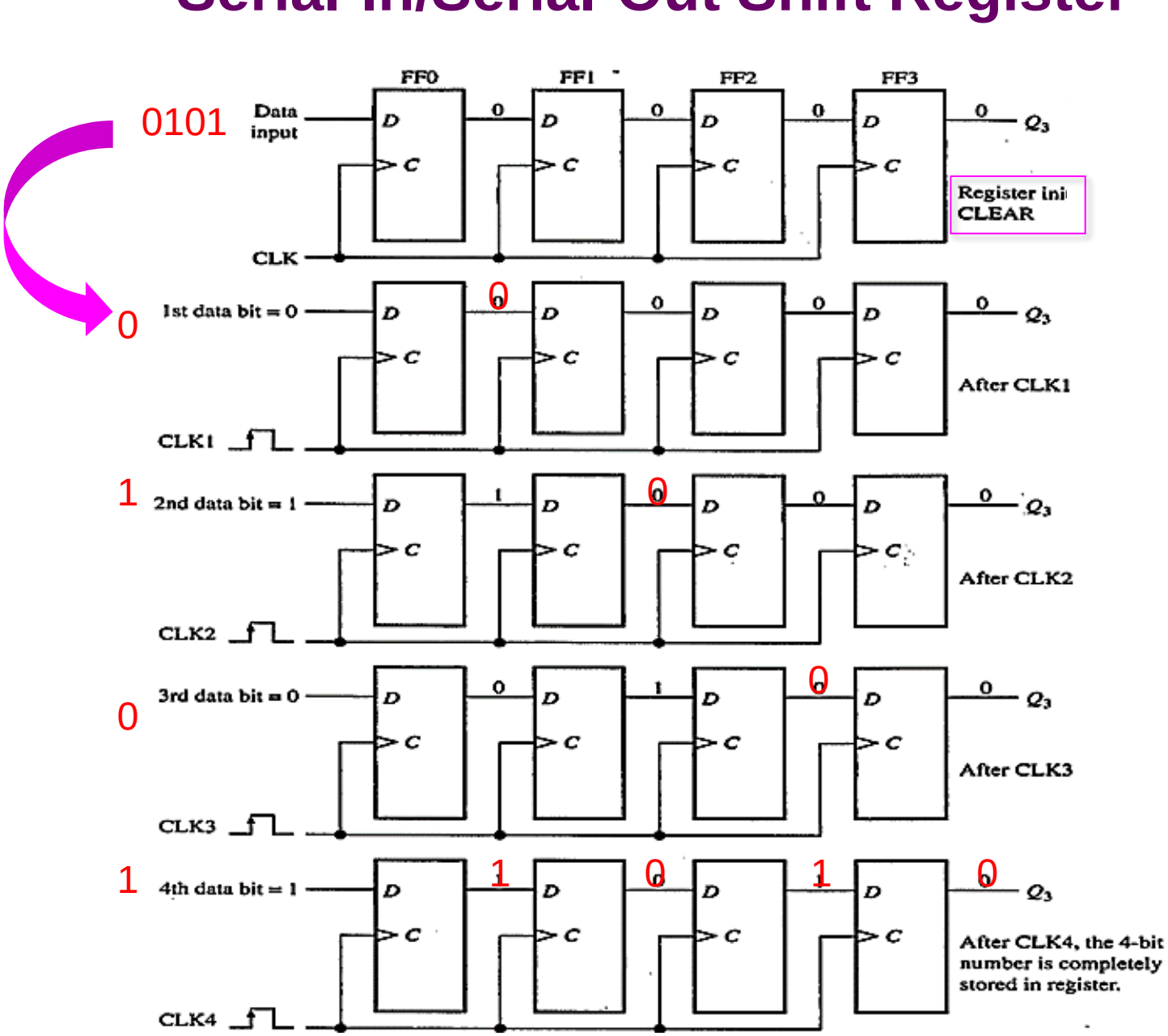
Categorized by orientation of the data transfer:

- Shift-Right Register (SRR)
- data is imported from the left.
- Keep transferring to the right until it reaches the end of the data.
- Shift-Left Register (SLR)
- input data from the right.
- Keep transferring to the left until it reaches the end of the data.

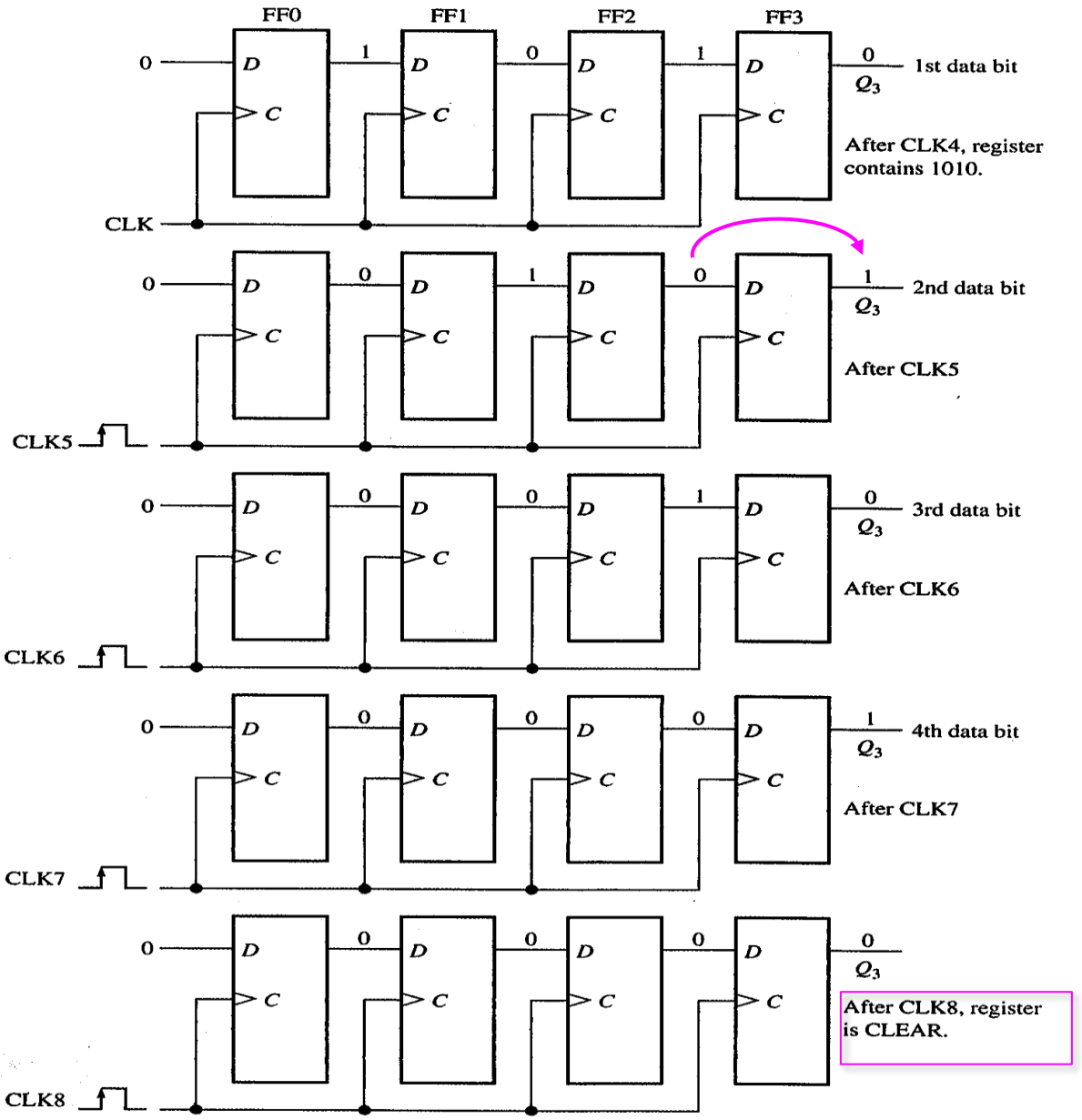
Serial-In/Serial-Out Shift Register



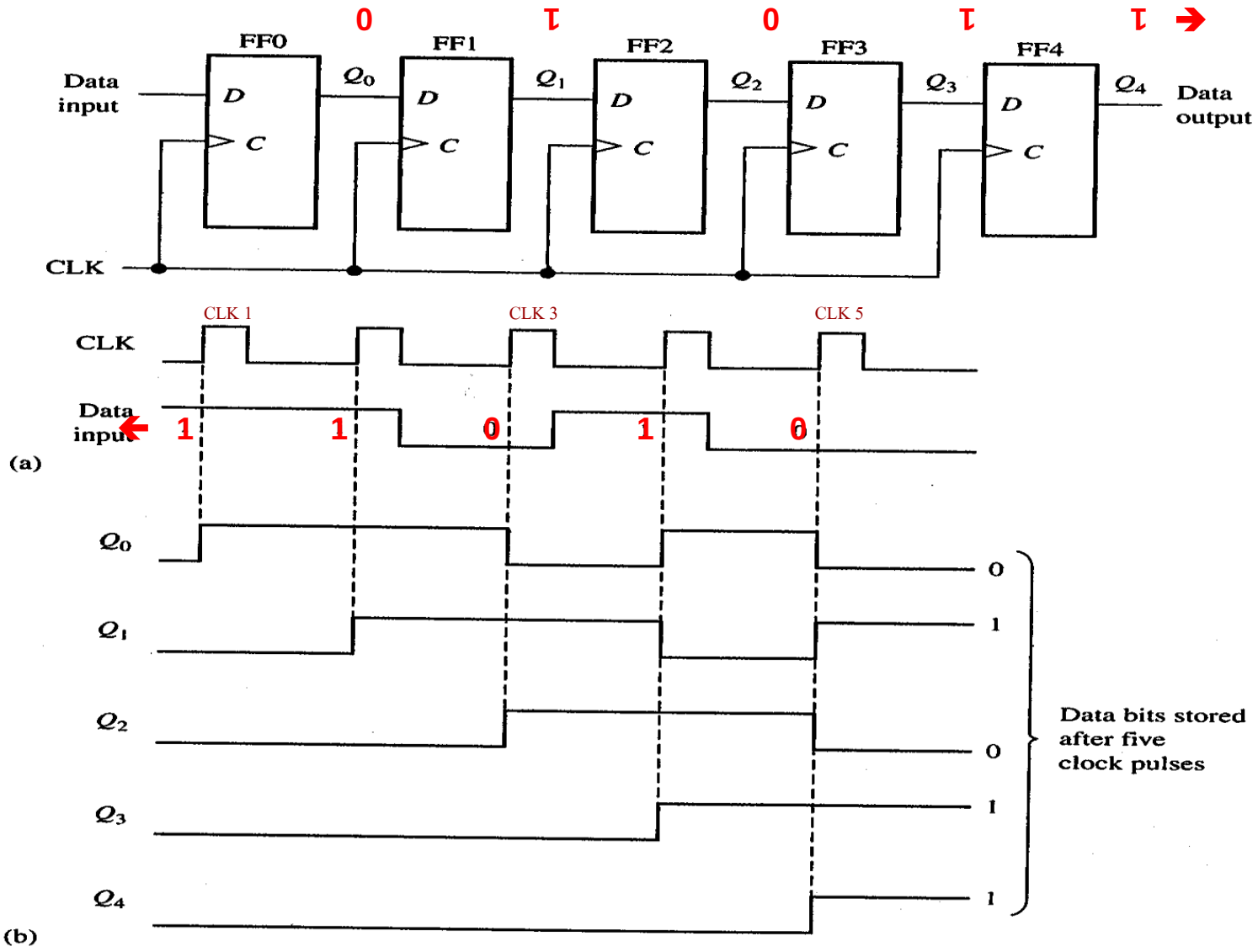
Serial-In/Serial-Out Shift Register



Serial-In/Serial-Out Shift Register

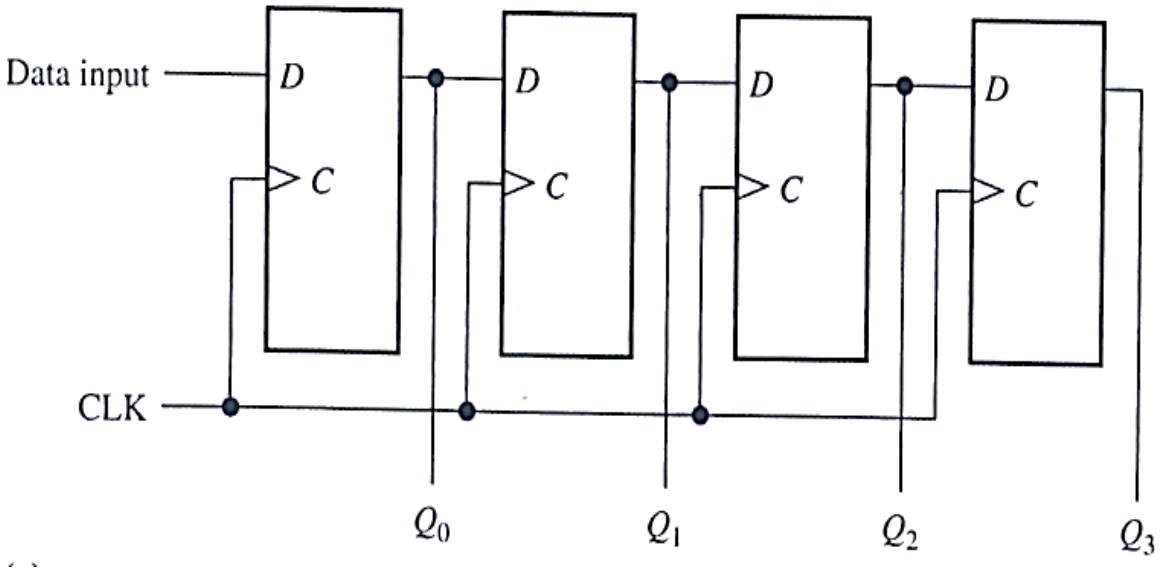


5-Bit Shift Register: Timing Diagram

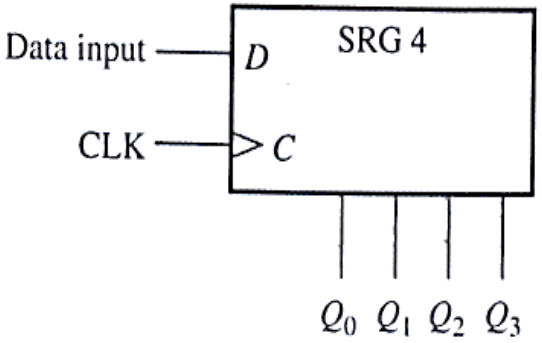


Serial-In Parallel-Out (SIPO) Register

- SIPO is a register that imports data in serial (the MSB first) and transfer data out parallelly (from the outputs of all Flip-Flops)
- Import the data of *n* bits requires *n* clocks;
- Once the all bits already Imported, the *n*-bit data will appear immediately at the output of individual Flip-flops.

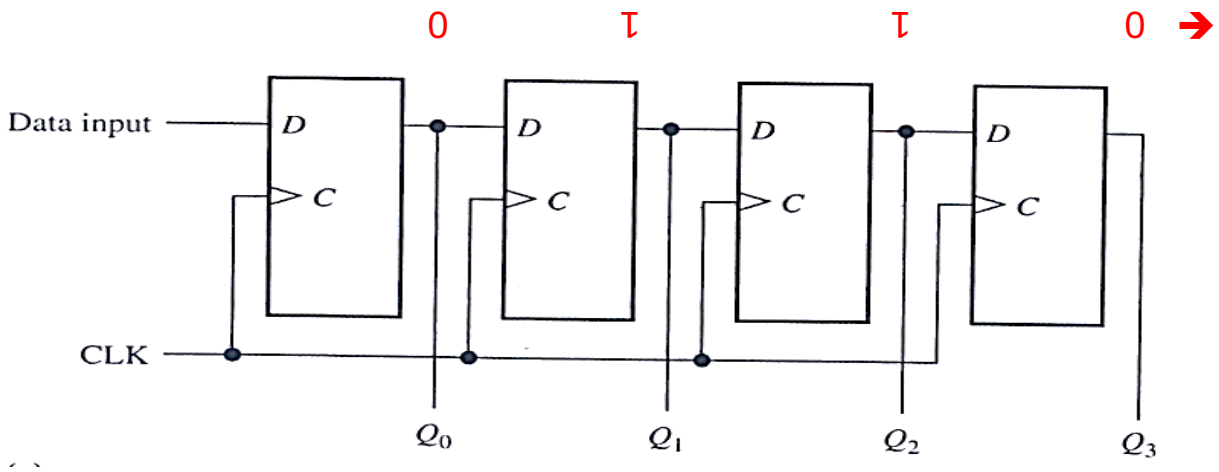


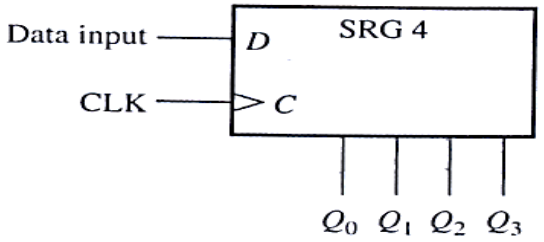
Block Diagram

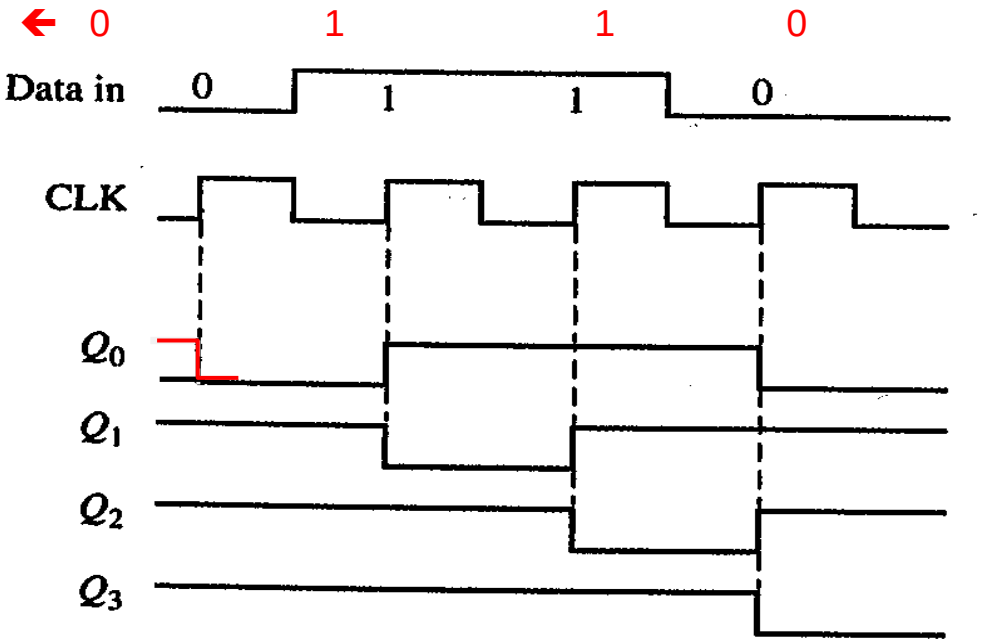


Symbol

Serial-In Parallel-Out Register: Timing Diagram

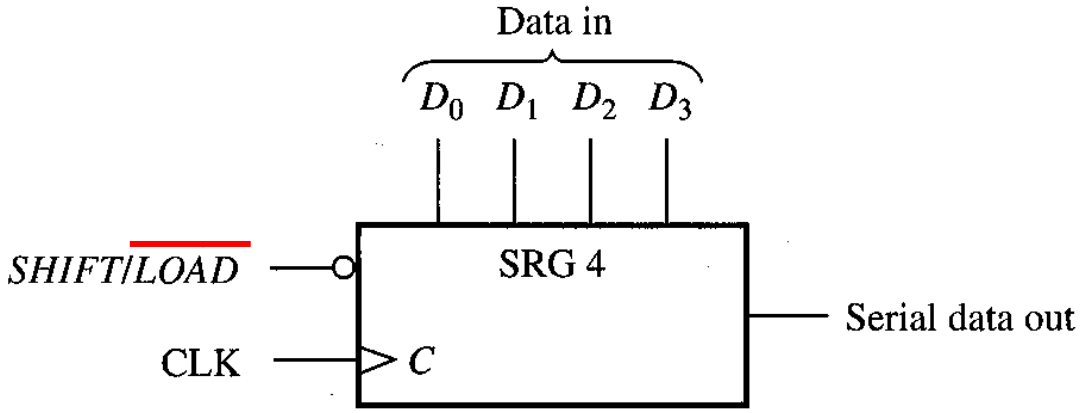






Parallel-In Serial-Out (PISO) Shift Register

- PISO is a register with multiple inputs of *n* Flip-Flops;
- Requires a control signal (SHIFT/LOAD) to indicate loading or shifting the data;
- For the data loading state: the *n-bit* data will be imported to be at the individual outputs of the Flip-Flops by *1* clocks;
- For the data shifting state: the *n-bit* data will be shifted to the right of the individual Flip-Flops by (*n* – *1*) clocks;

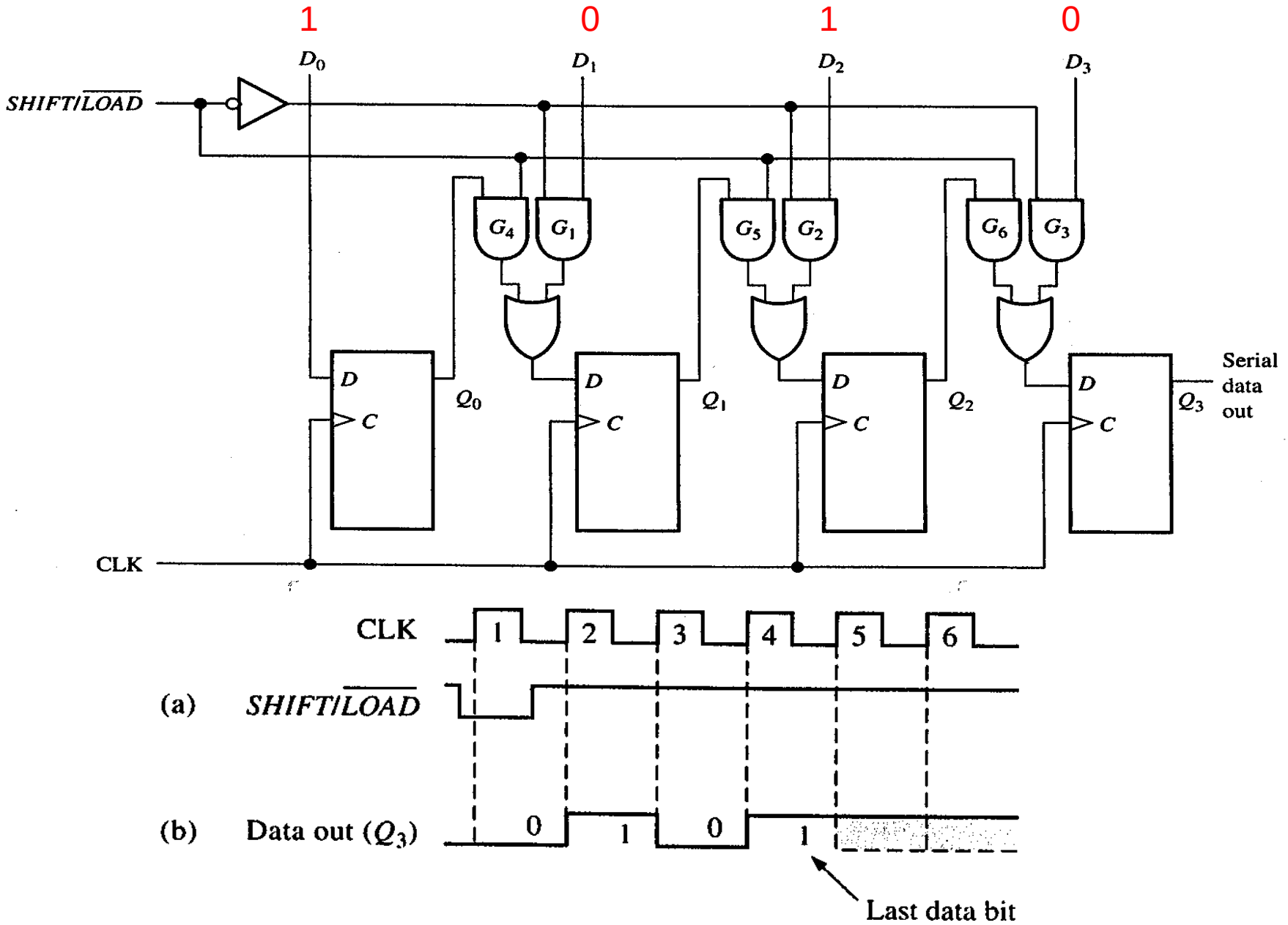


- If SHIFT/LOAD = 0, G_1 to G_3 are active:

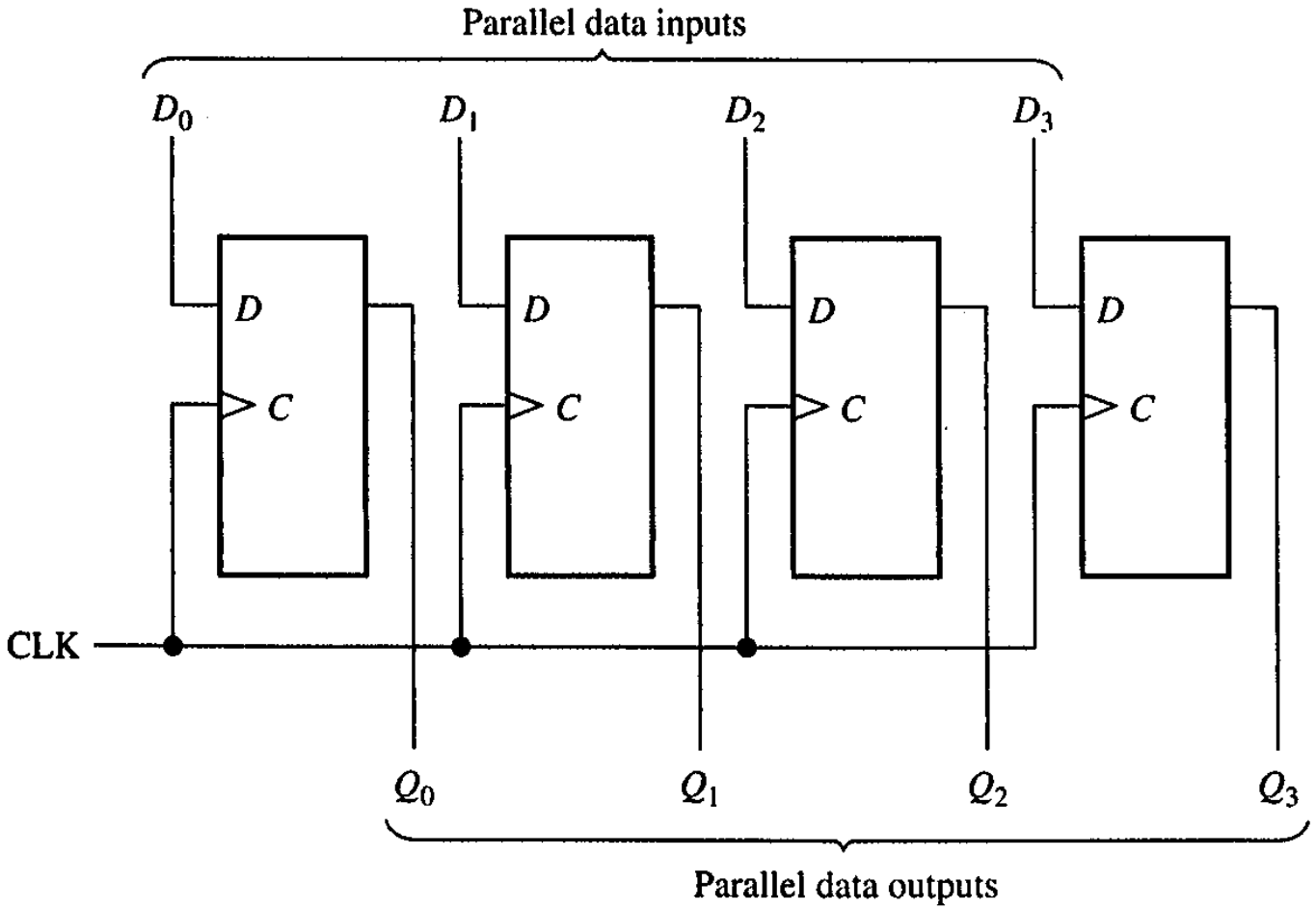
  ➔ the input data will be imported into the parallel register by *1* clocks.
- If SHIFT/LOAD = 0, G_4 to G_6 are active:

  ➔ the data will be shifted bit by bit to the right by (*n*) clocks.

Parallel-In Serial-Out Shift Register



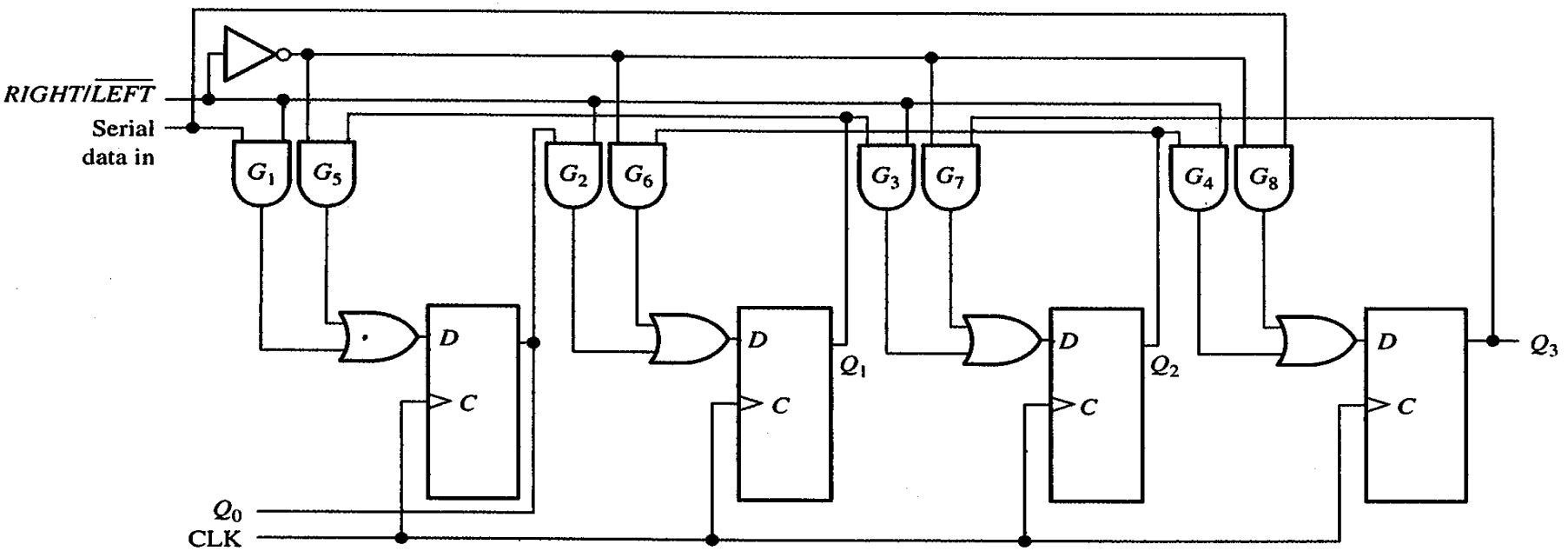
Parallel-In/Parallel-Out (PIPO) Register



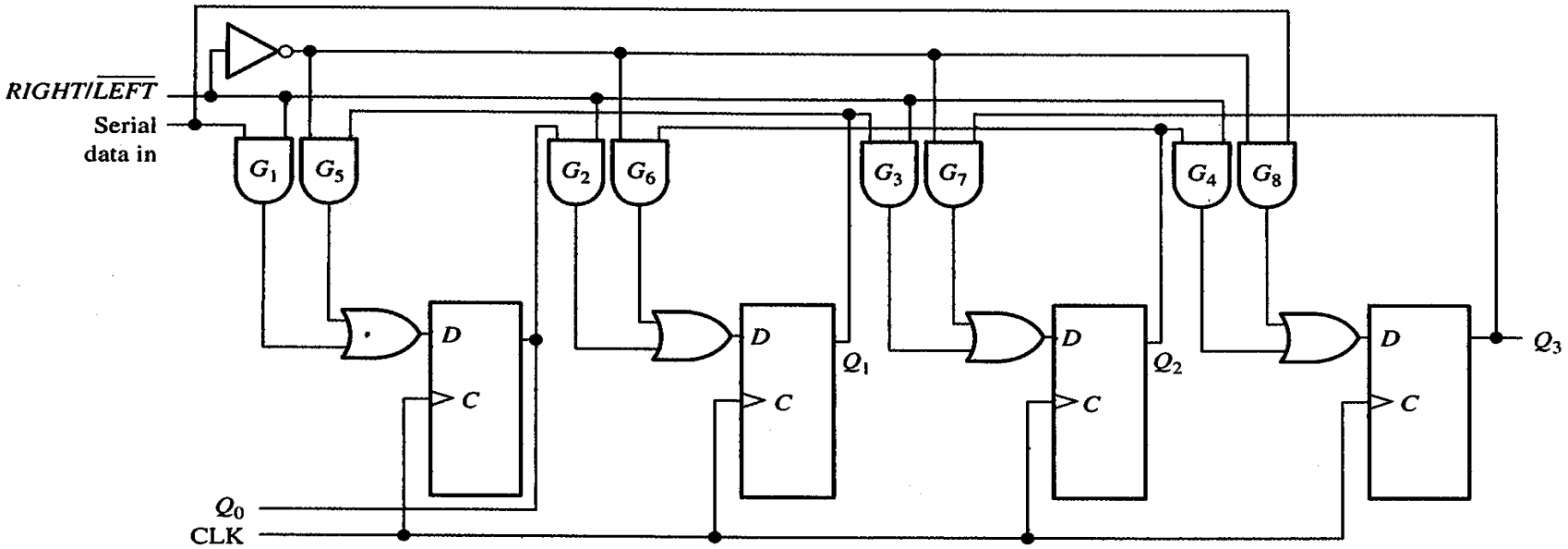
- The input data will be imported into the parallel register by *1* clocks.
- And ready to be exported for another register.

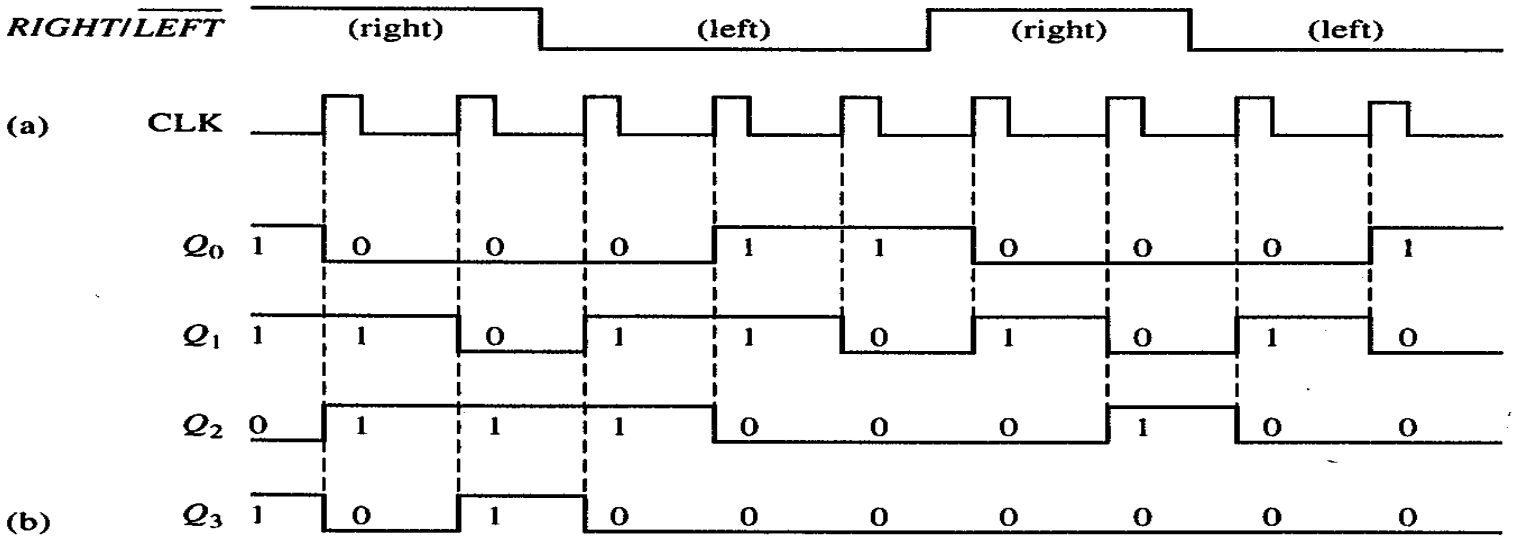
Bidirectional Shift Register

- If need to shift the data in the register either to the right or to the left.
- By inserting a gate circuit to controls the direction of shifting;
- to control by the RIGHT/$\overline{\text{LEFT}}$ pin:
  - If the pin is 1, the data shifting is to the right;
  - If the pin is 0, the data shifting is to the right left.



Bidirectional Shift Register

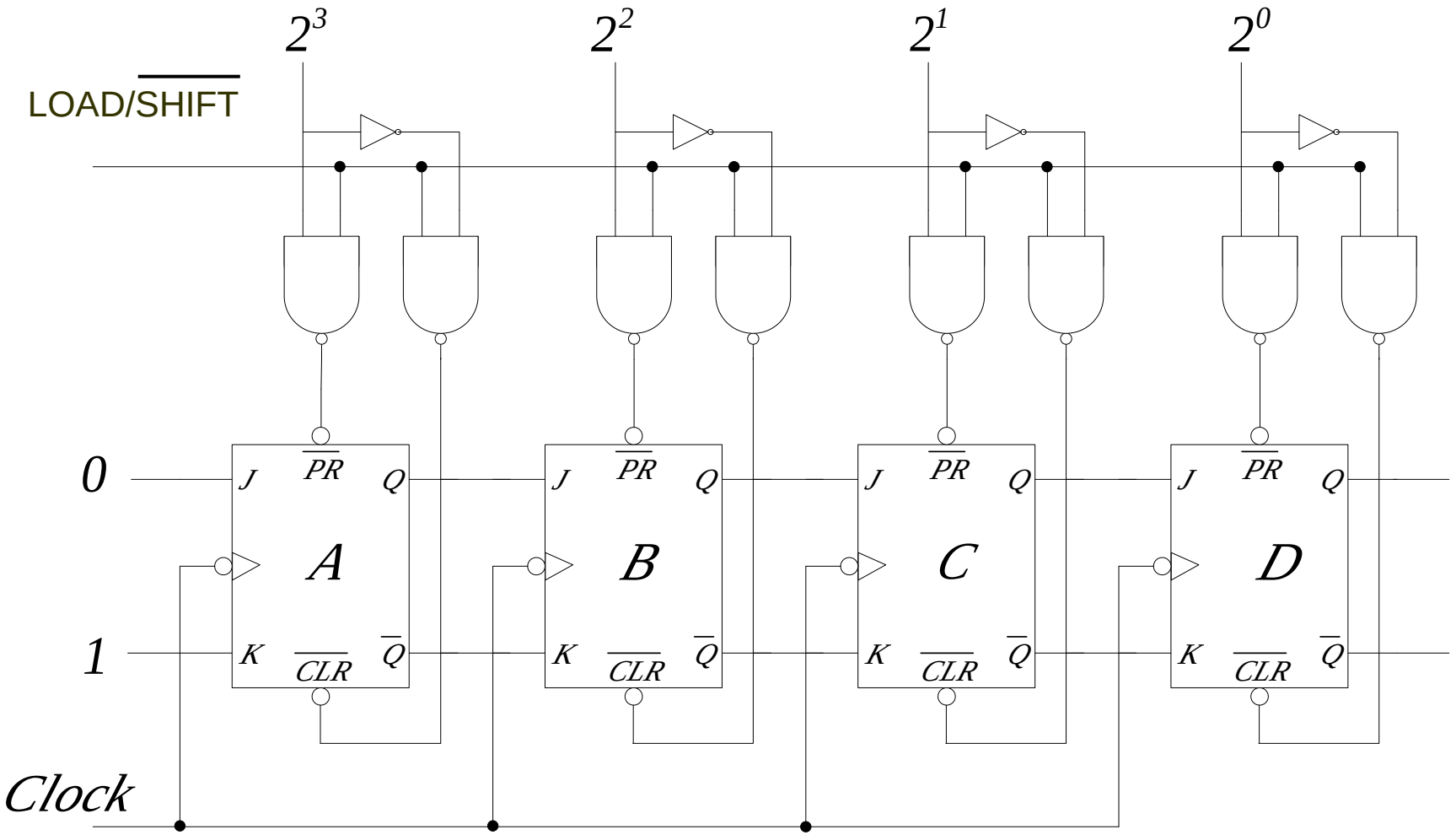




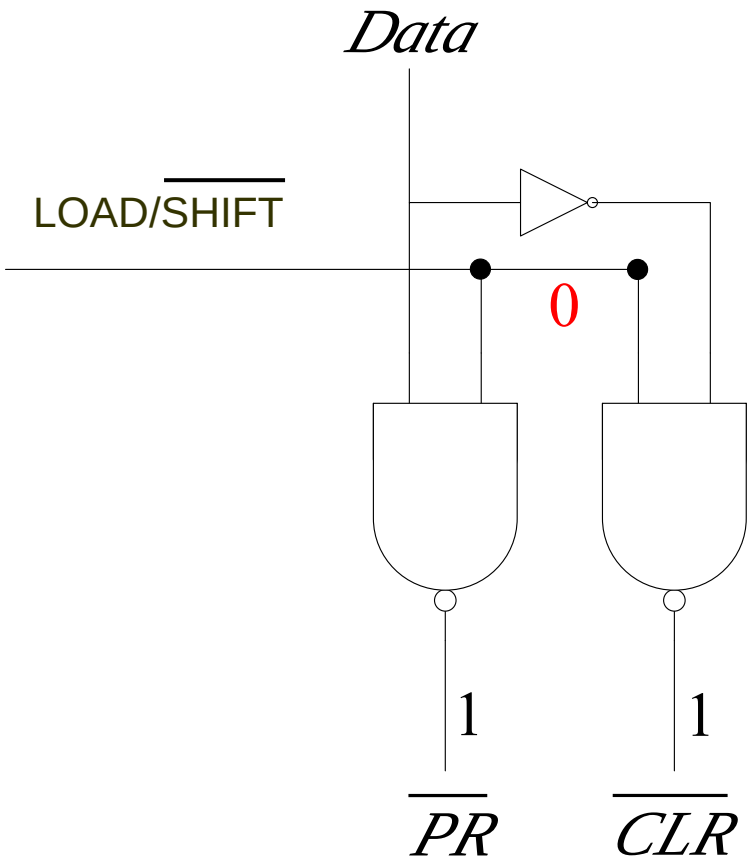
Parallel-In Serial-Out (PISO) Shift Register

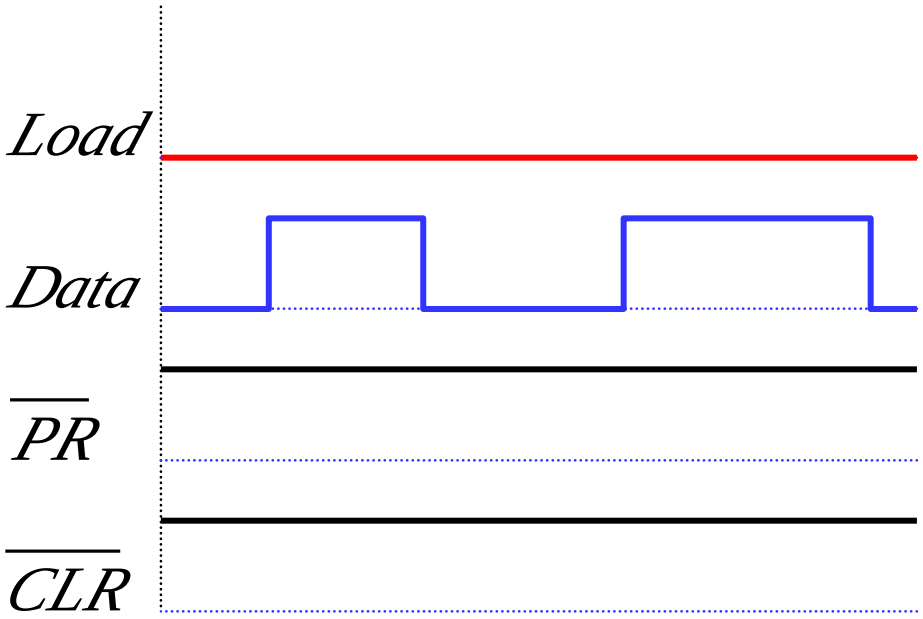
Typically, we use D Flip-Flop to create registers.

However, we may use JK Flip-Flops with Preset and Clear too.

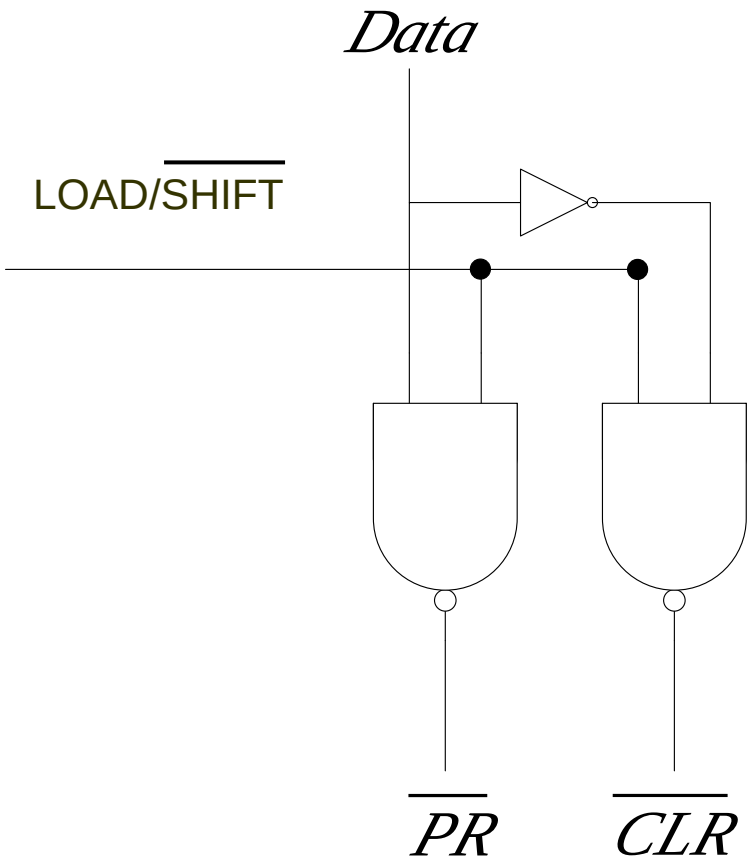


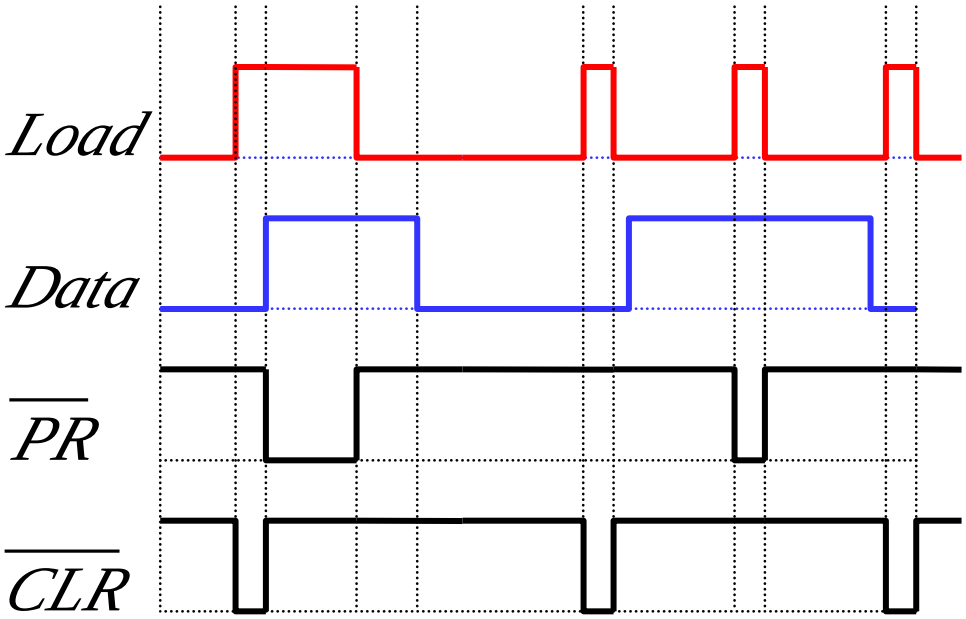
Parallel-In Serial-Out Shift Register



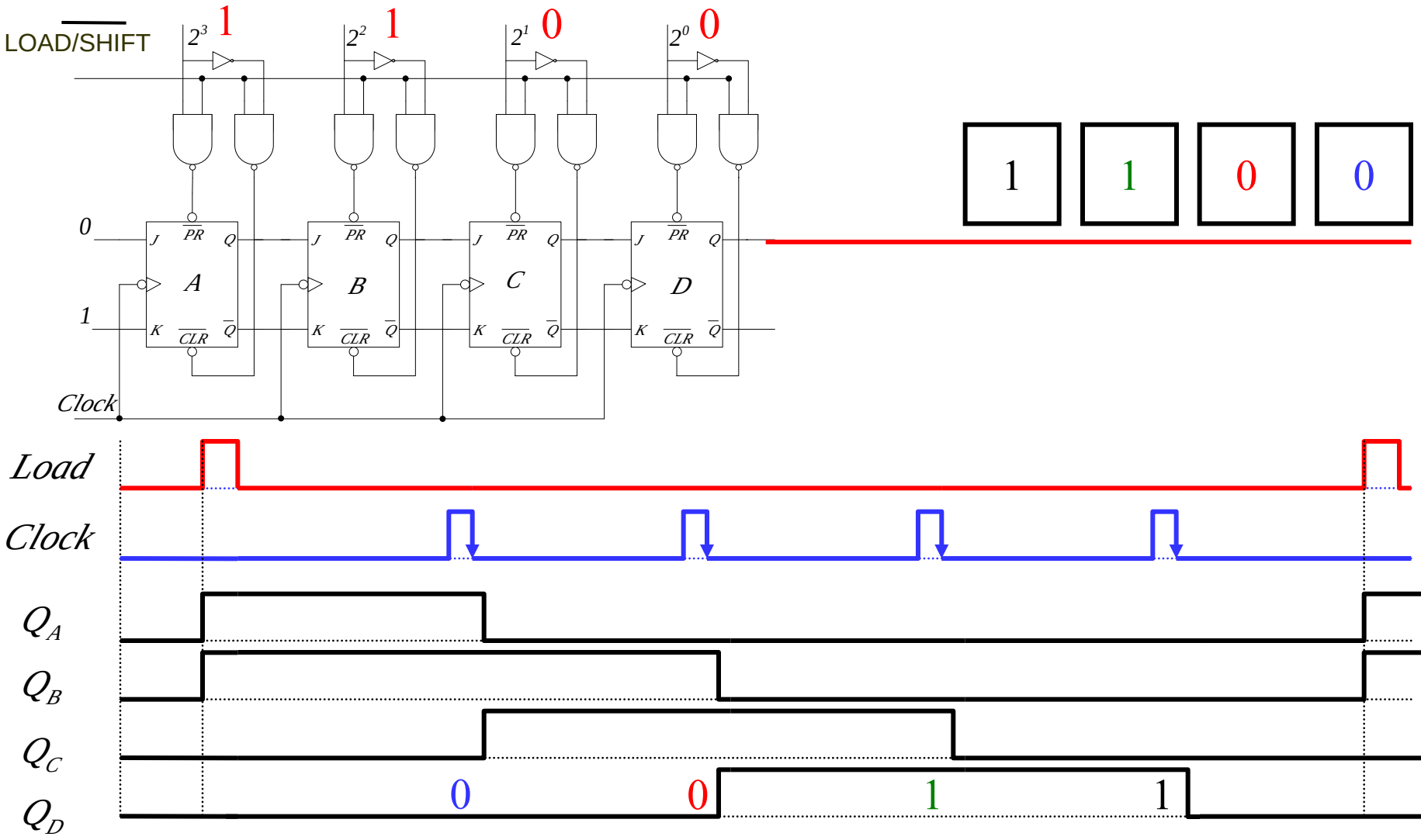


Parallel-In Serial-Out Shift Register



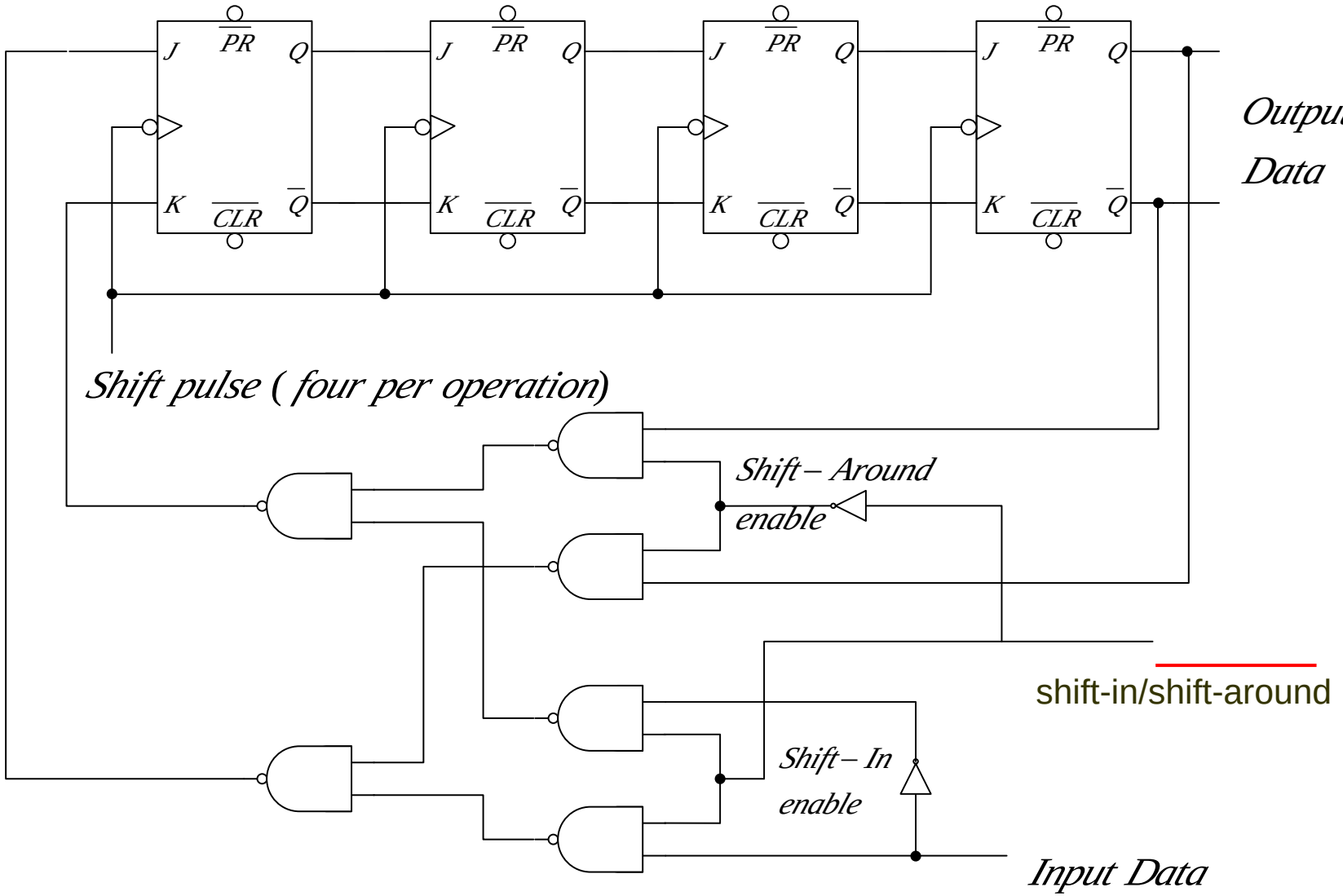


Parallel-In Serial-Out (PISO) Shift Register

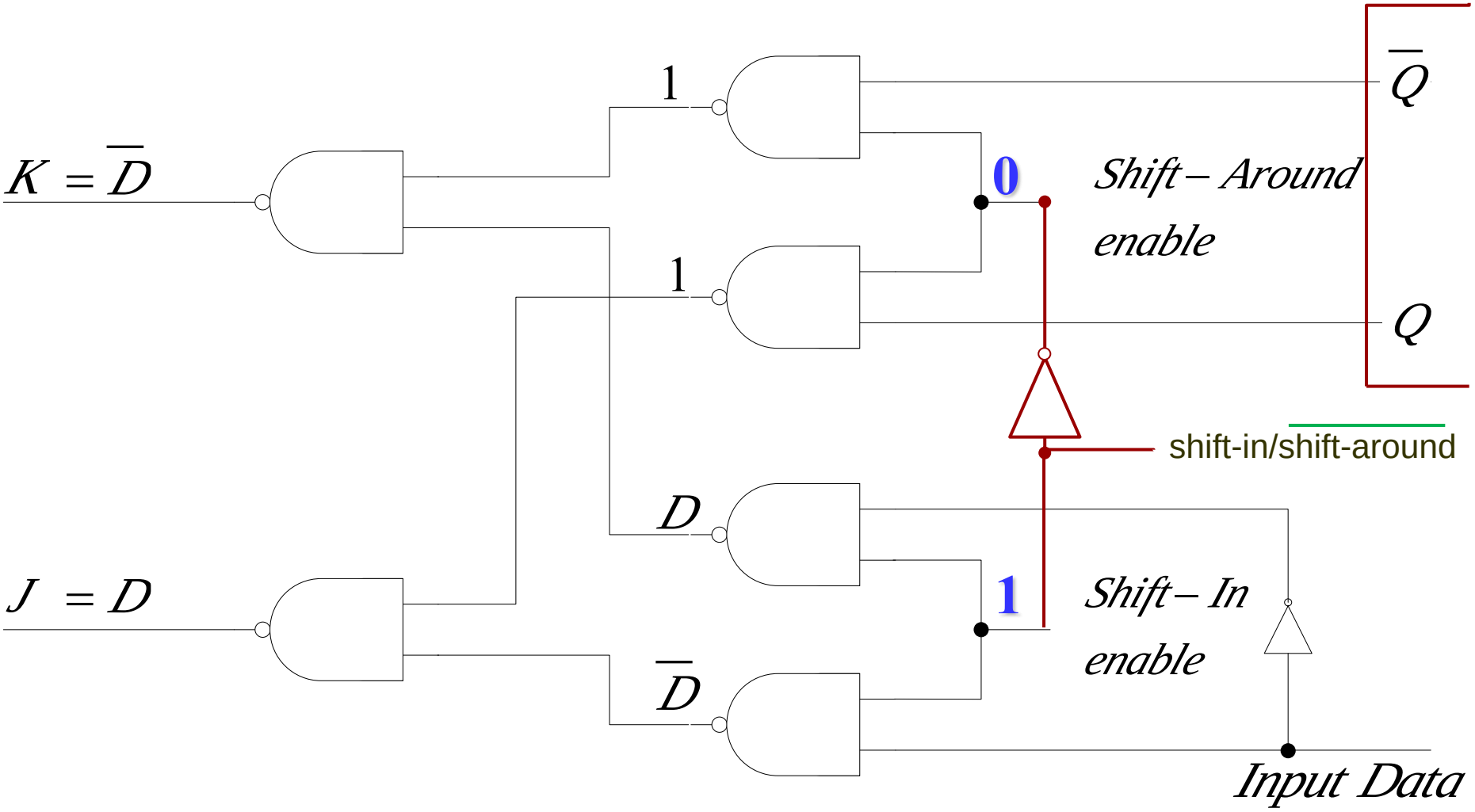


Shift-in Shift-around Register

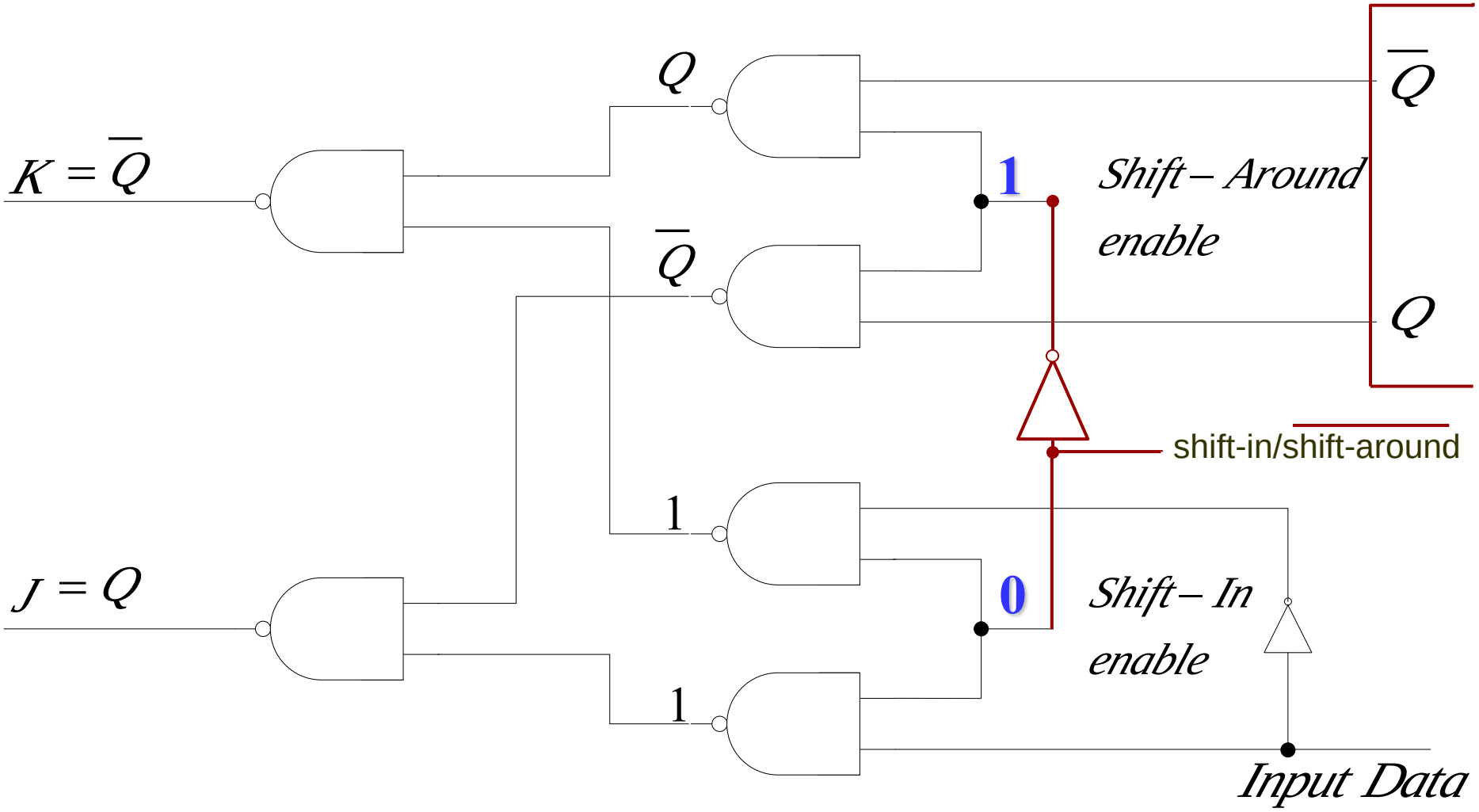
- This register can import the data from outside (shift-in) or from itself (shift-around).
- Need a control signal for gating the data either from outside or from itself.



Shift-in Shift-around Register



Shift-in Shift-around Register



Shift Register Counter

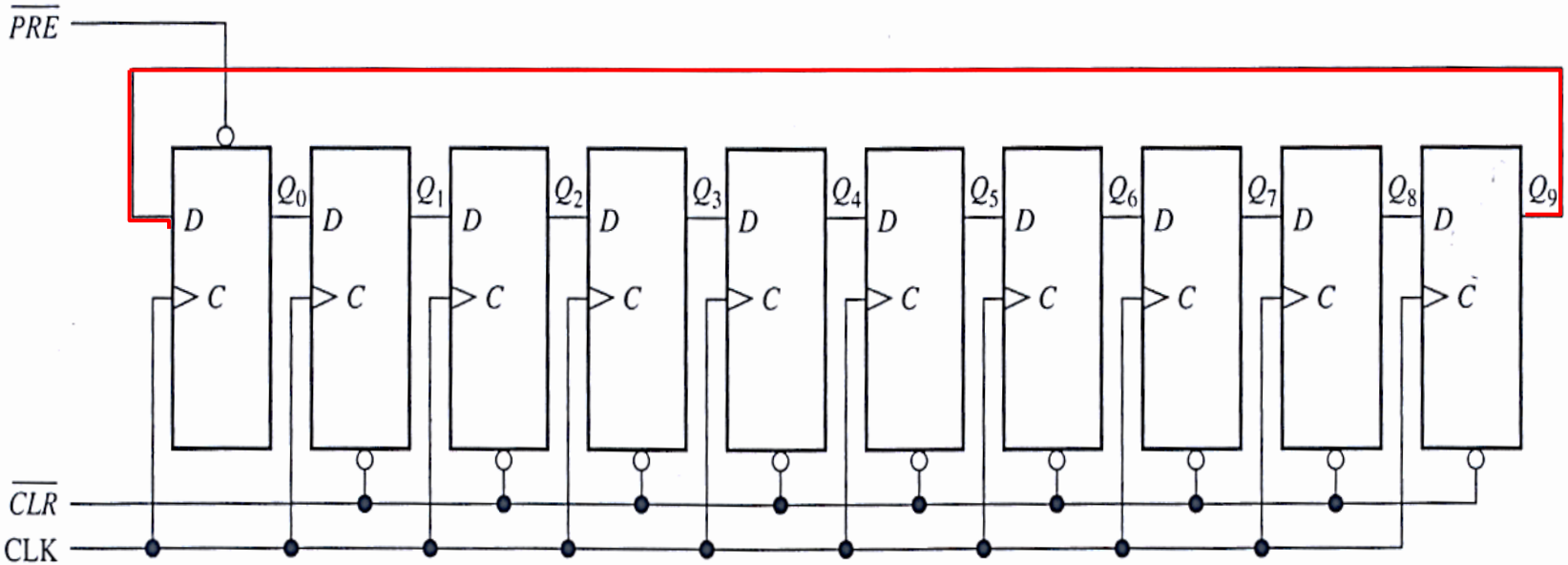
- Shift Register Counter is a Shift-around register with feeding the last output back to the first input.
- This will create a specific sequence of data,
- And be called as a Counter i.e. Shift Register Counter.
- There are 2 types of Shift Register Counter:

❑ Ring Counter: Feed the last output Q to the first input D

❑ Johnson Counter : Feed the last invert output $\overline{Q}$ to the first input D

Ring Counter

- The counter that uses *n* Flip-Flops to count *n* states;
- It will repeat the specific number for every *n* cycles;
- Therefore, do not need gates to help process the output at all;
- For example, for 10-bit Ring Counter, the circuit will repeat its initial number in every 10 clock cycles.

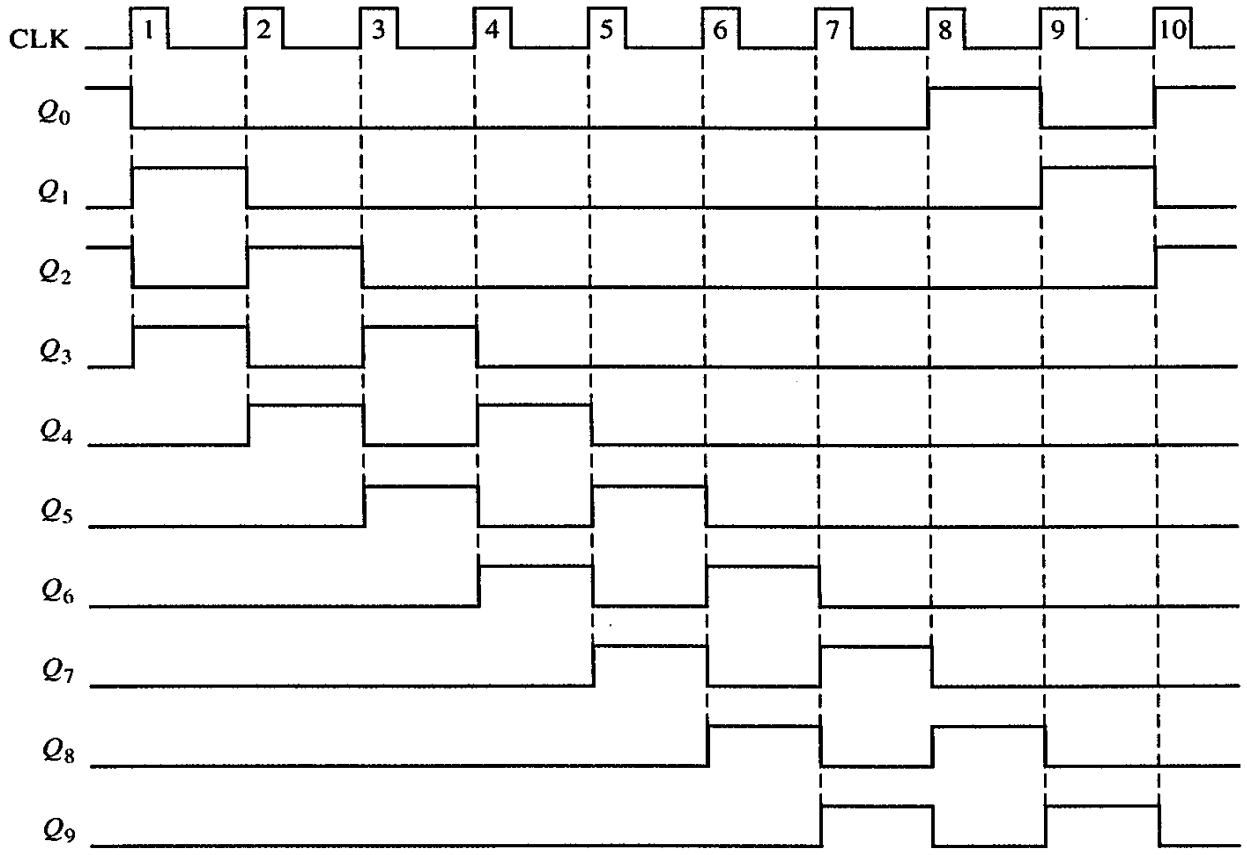


Ring Counter

CLOCK PULSE	Q_0	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8	Q_9
0	1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	0	0	0	1

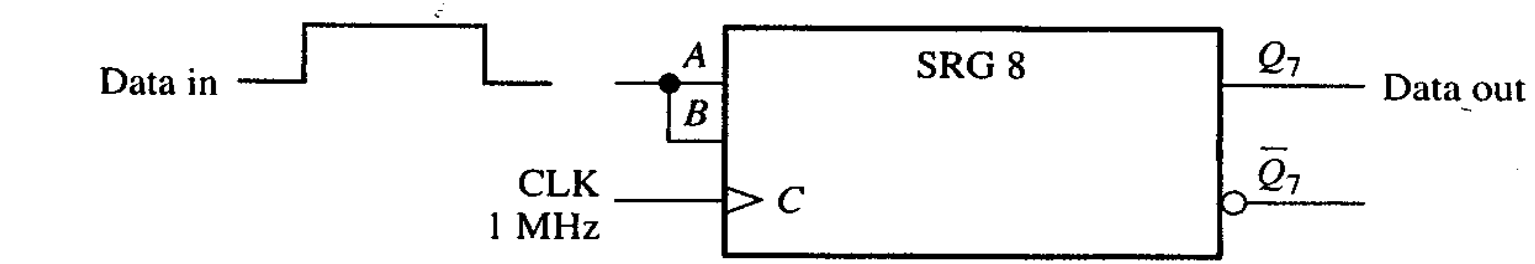
Ring Counter

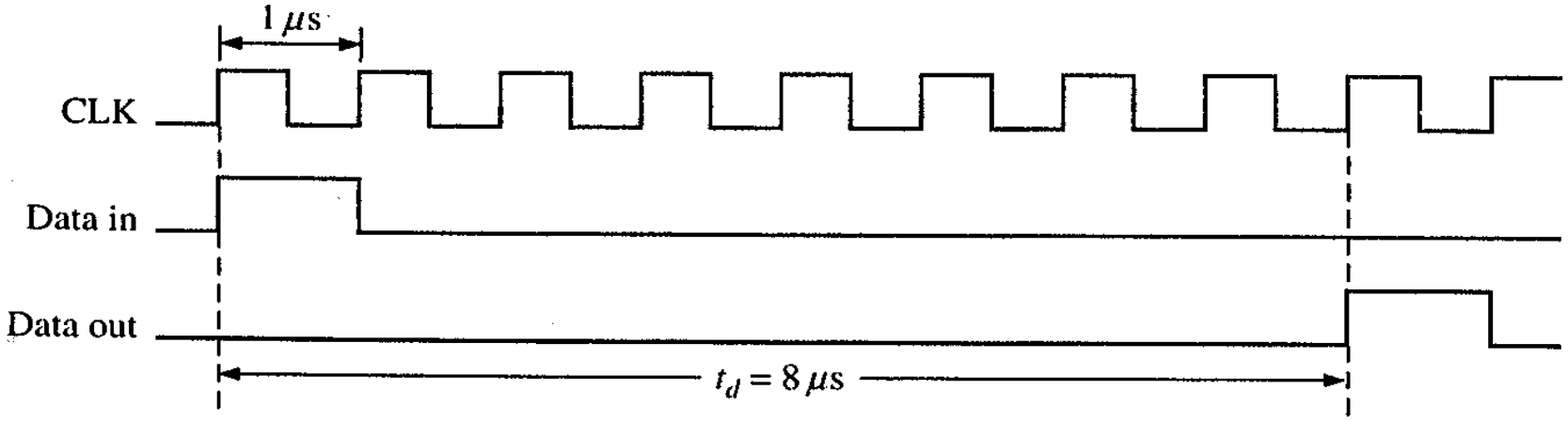
10-bit Ring Counter with an initial value of 0000000101



Shift Register: Applications

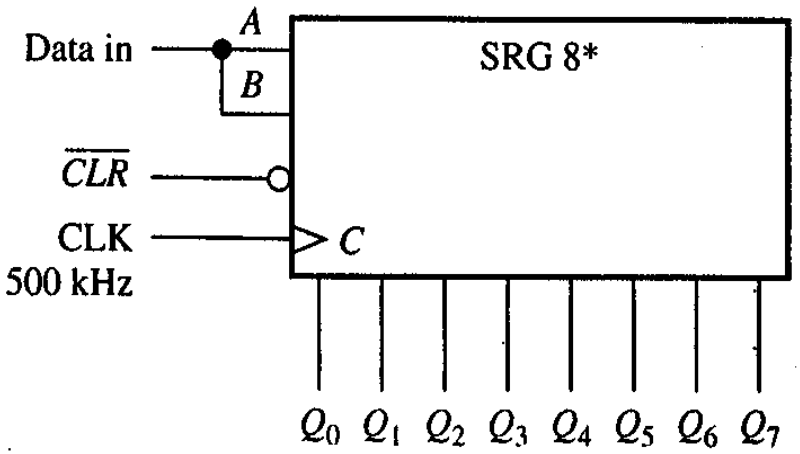
- **Time Delay**





Shift Register: Applications

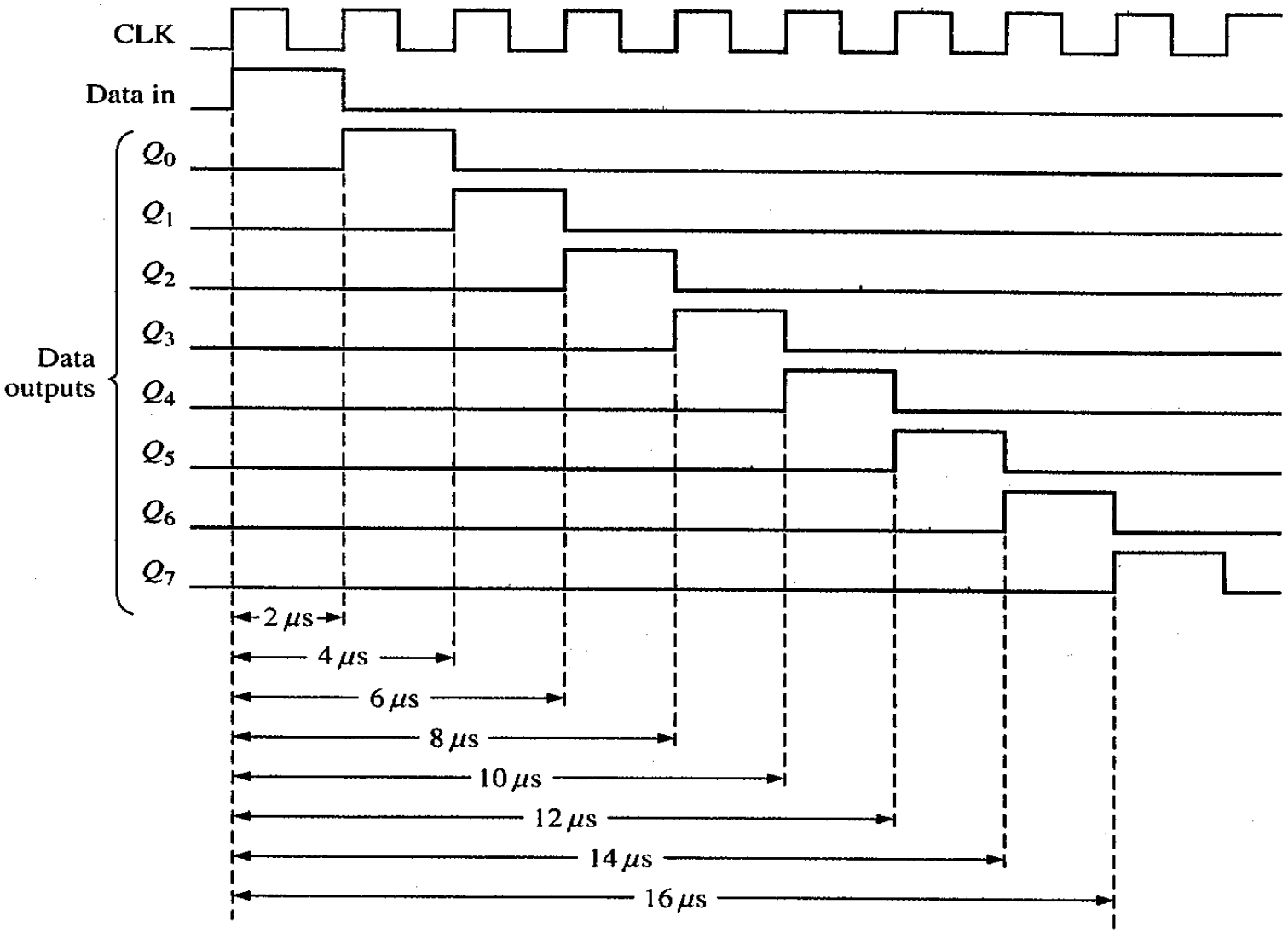
Example: Calculate the time delay when the clock runs at 500 kHz.



* Data shifts from Q_0 toward Q_7.

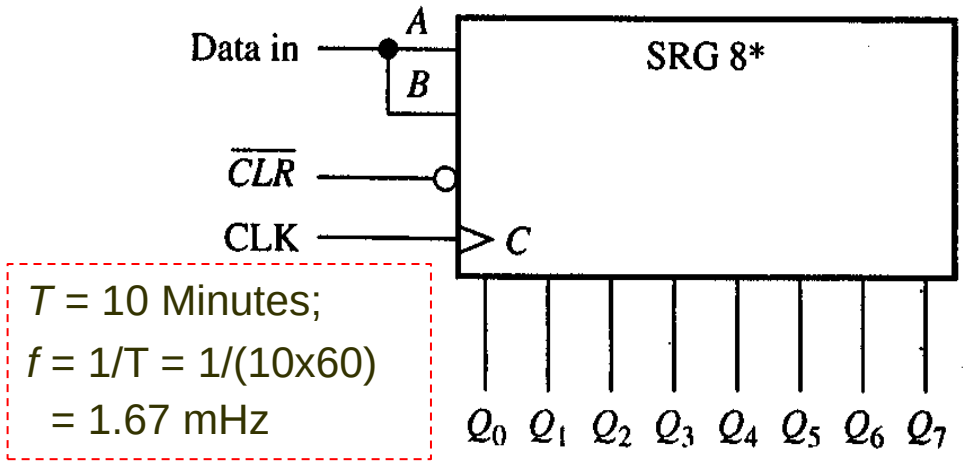
The time delay = $1/f = 1/(500 \times 10^3) = 2 \times 10^{-6}$ second = 2 μs.

Shift Register: Applications



Shift Register: Applications

Example: Find the clock that provides 10 minutes duration step for the morning routine below.



T = 10 Minutes;
f = 1/T = 1/(10x60)
= 1.67 mHz

Q_0**:** Turn-off the air-conditioner;

Q_1**:** Turn-on heater for the electric shower;

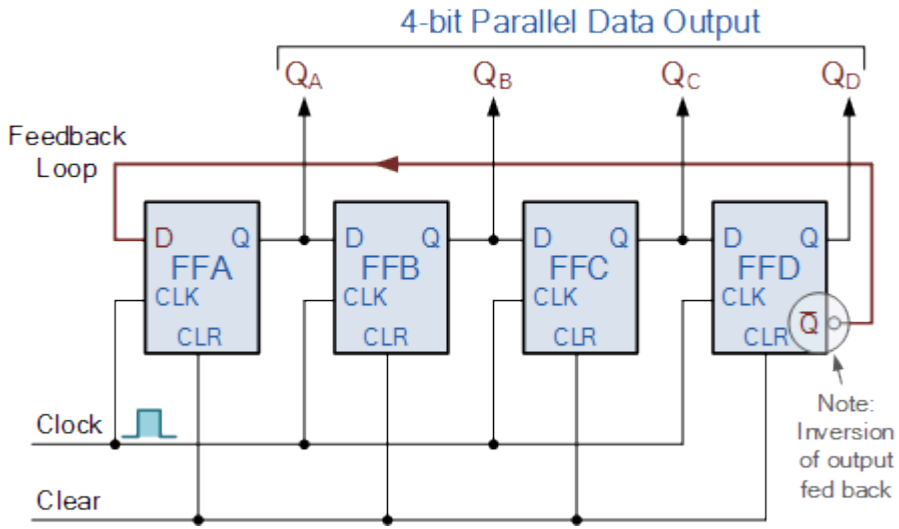
Q_2**:** Alarm to wake up;

Q_3**:** Boil the water pot for tea/coffee;

Q_7**:** Turn off the circuit braker.

Johnson Counter

4-bit Johnson Counter



Truth Table

Clock Pulse No	FFA	FFB	FFC	FFD
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1

- The counter that uses *n* Flip-Flops to generate 2*n* states;
- It will repeat its initial number for every 2*n* cycles;