MODULE 5

SHIFT REGISTERS

Introduction

- Data may be available in parallel form or serial form
- Multi-bit data is said to be in parallel form when all the bits are available simultaneously.
- The data is said to be in serial form when the data bit appears sequentially on e after the other ,in time at a single terminal.

- Flip flop – store only one bit of data. Single bit register.
- Register – set of FFs used to store binary data.
- Loading a register – inputting data into register. Ie; setting or resetting the individual FFs.
- May be serial or parallel.

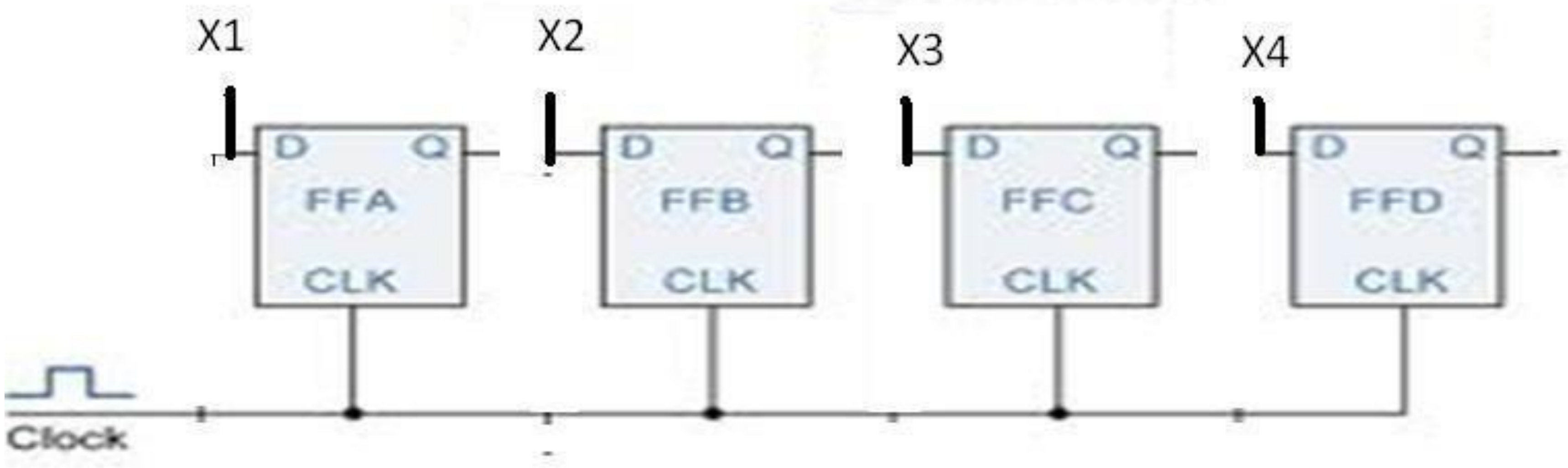
Shift register

- A type of logic circuits closely related to counters.
- Used for storage and transfer of digital data.
- If the register is capable of shifting bits either towards right hand side or towards left hand side is known as **shift register**.
- An “N‟ bit shift register contains “N‟ flip-flops.
- Difference between counter and shift register – shift register has no specified sequence of states whereas counter has.

Buffer register

- The buffer register is the simple set of registers.
- It is simply stores the binary word. The buffer may be controlled buffer.
- Most of the buffer registers used D Flip-flops.

Figure: logic diagram of 4-bit buffer register



- The figure shows a 4-bit buffer register.
- The binary word to be stored is applied to the data terminals.
- On the application of clock pulse, the output word becomes the same as the word applied at the terminals. i.e., the input word is loaded into the register by the application of clock pulse.
- When the positive clock edge arrives, the stored word becomes:

$$Q4Q3Q2Q1=X4X3X2X1$$

$$Q=X$$

Controlled buffer register

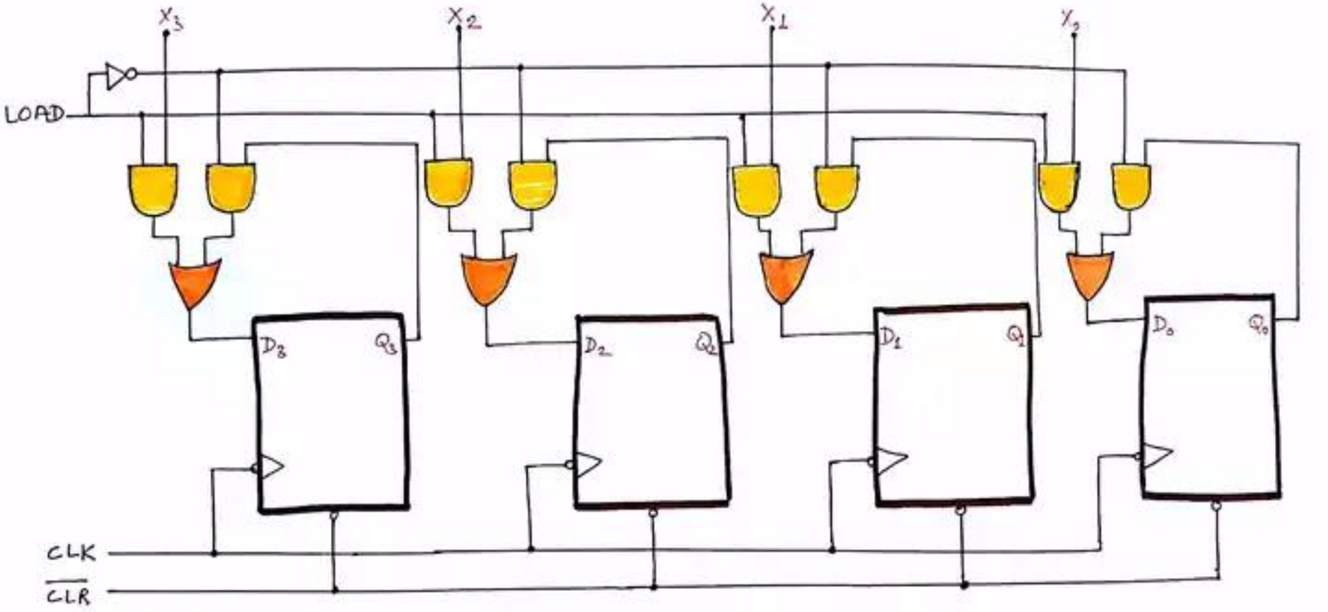
- If goes LOW, all the FFs are RESET and the output becomes, Q=0000.
- When is HIGH, the register is ready for action. LOAD is the control input.
- When LOAD is HIGH, the data bits X can reach the D inputs of FF‘s.

  Q4Q3Q2Q1=X4X3X2X1

  Q=X
- When load is low, the X bits cannot reach the FF‘s. At the same time, the inverted signal LOAD is $\overline{\text{HIGH.}}$

- This forces each flip-flop output to feed back to its data input.
- Therefore, the data is circulated or retained as each clock pulse arrives.
- In other words, the contents of the register remain unchanged in spite of the clock pulses.
- Longer buffer registers can be built by adding more FFs.

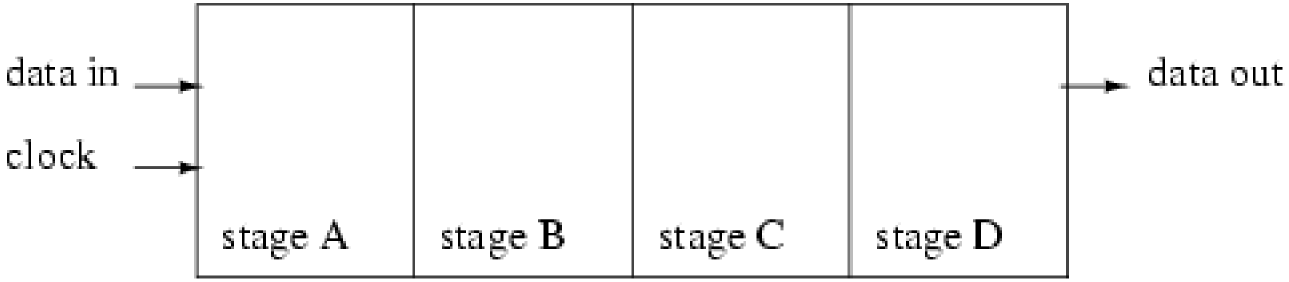
CONTROLLED BUFFER REGISTER



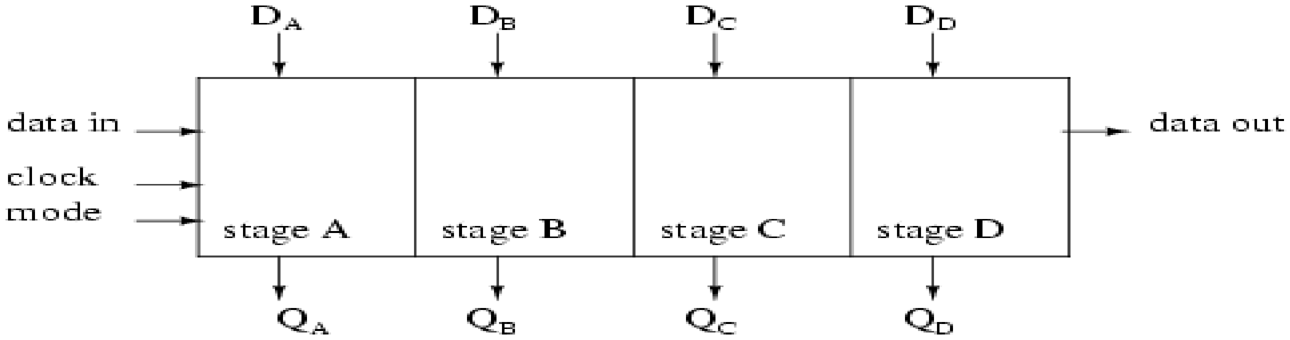
Data transmission in shift registers

- A number of FFs connected together such that data may be shifted into and shifted out of them is called shift register.
- Data may be shifted into or out of the register in serial form or in parallel form.
- There are four basic types of shift registers.

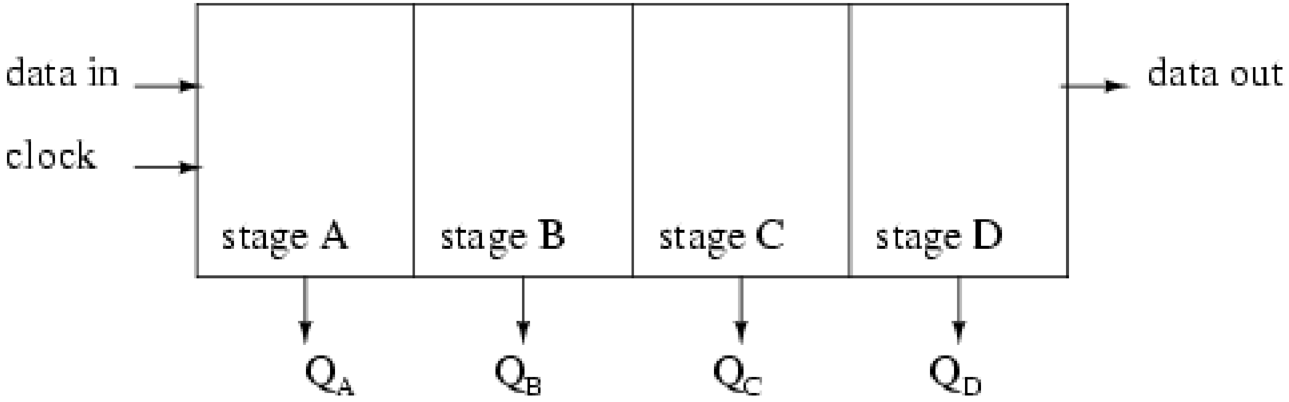
1. Serial in, serial out, shift right/left, shift registers
2. Serial in, Parallel out, shift registers
3. Parallel in, serial out shift registers
4. Parallel in, parallel out shift registers



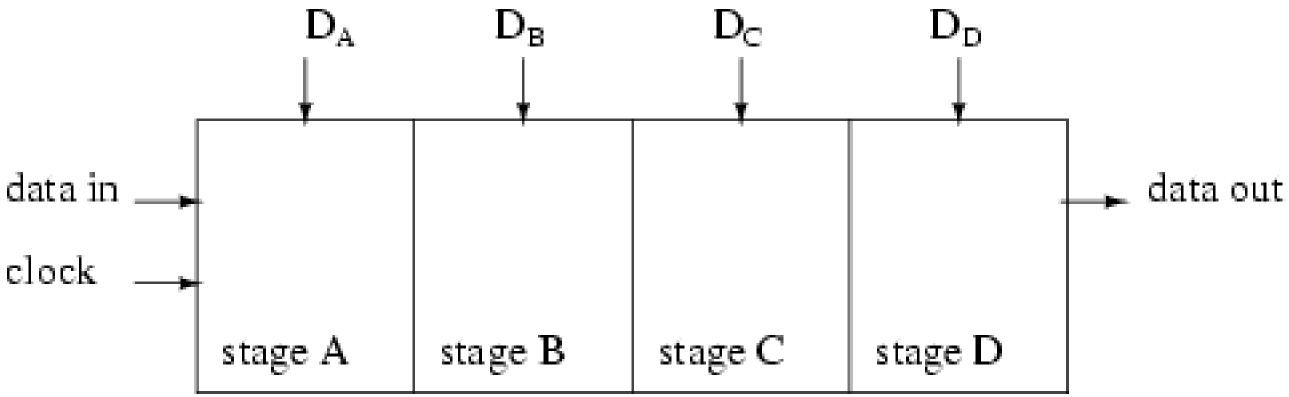
Serial-in, serial-out shift register with 4-stages



Parallel-in, parallel-out shift register with 4-stages



Serial-in, parallel-out shift register with 4-stages



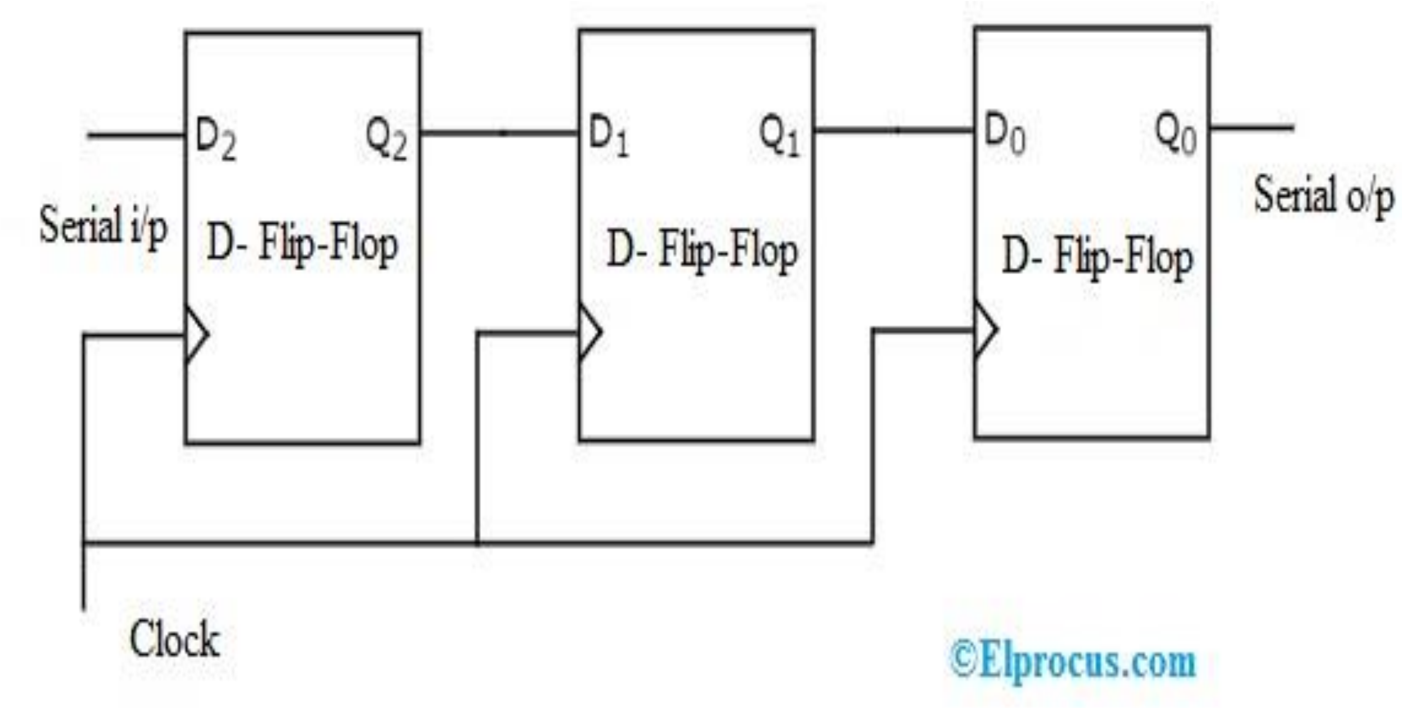
Parallel-in, serial-out shift register with 4-stages

1.Serial In Serial Out (SISO) shift registers

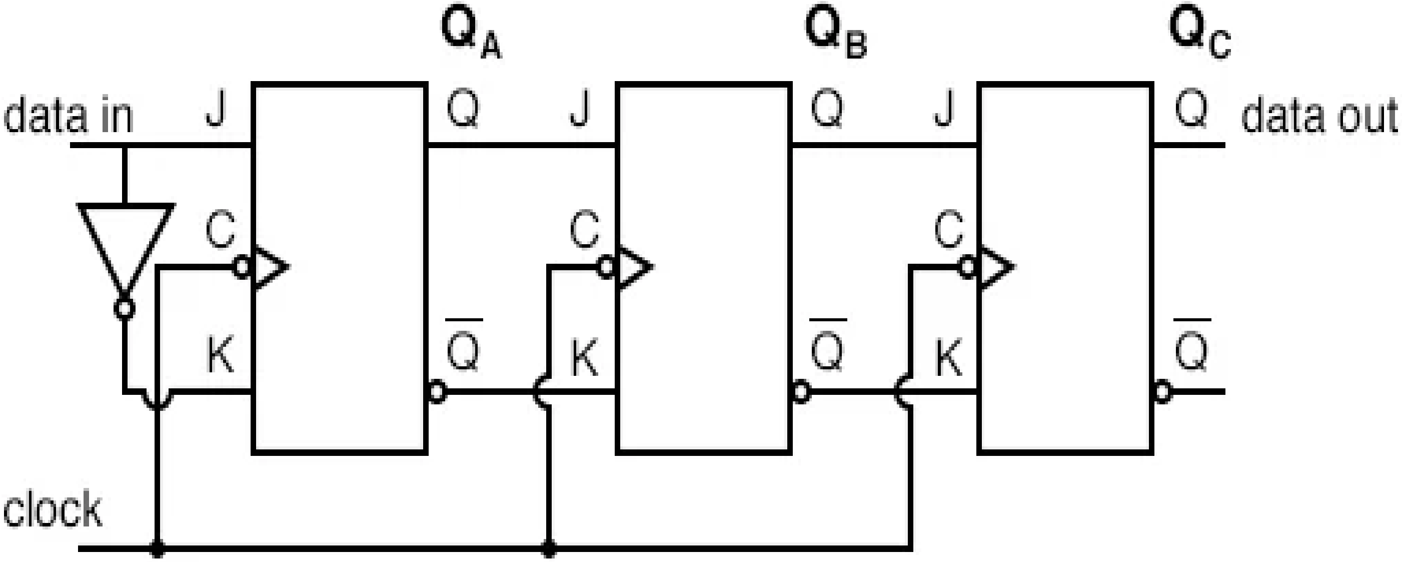
- The shift register, which allows serial input and produces serial output is known as Serial In – Serial Out **(SISO)** shift register.
- **Serial In Serial Out (SISO) shift registers** are a kind of shift registers where both data loading as well as data retrieval to/from the shift registers occurs in serial-mode.
- The logic diagram of 3-bit serial in serial out, right shift register with four stages.
- The register can store four bits of data.
- Serial data is applied at the input D of the first FF.

- The Q output of the first FF is connected to the D input of another FF.
- The data is outputted from the Q terminal of the last FF.
- When serial data is transferred into a register, each new bit is clocked into the first FF at the positive going edge of each clock pulse.
- The bit that was previously stored by the first FF is transferred to the second FF.
- The bit that was stored by the Second FF is transferred to the third FF.
- All these flip-flops are synchronous with each other since, the same clock signal is applied to each one.

3-bit serial-in,serial-out, shift-right, shift-register.



Using JK flip flop



Serial-in, serial out shift register using type "**JK**" storage elements

Left shift SISO Register

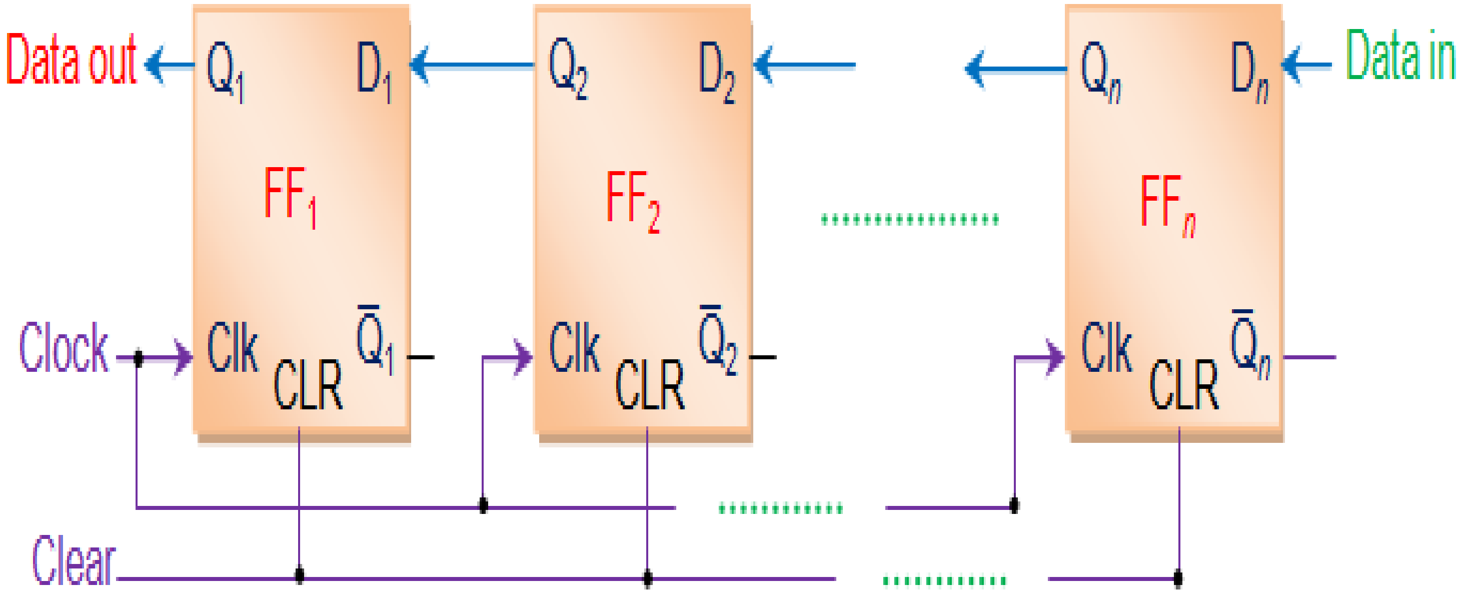
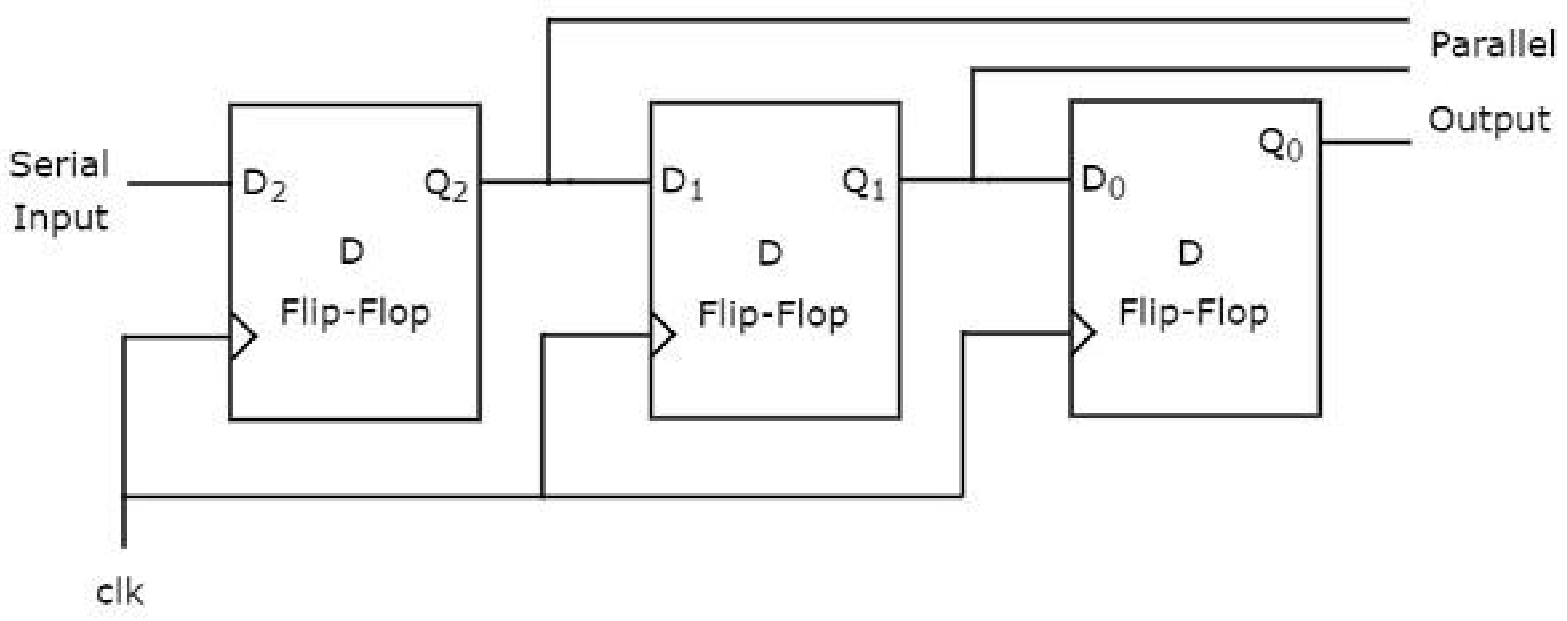


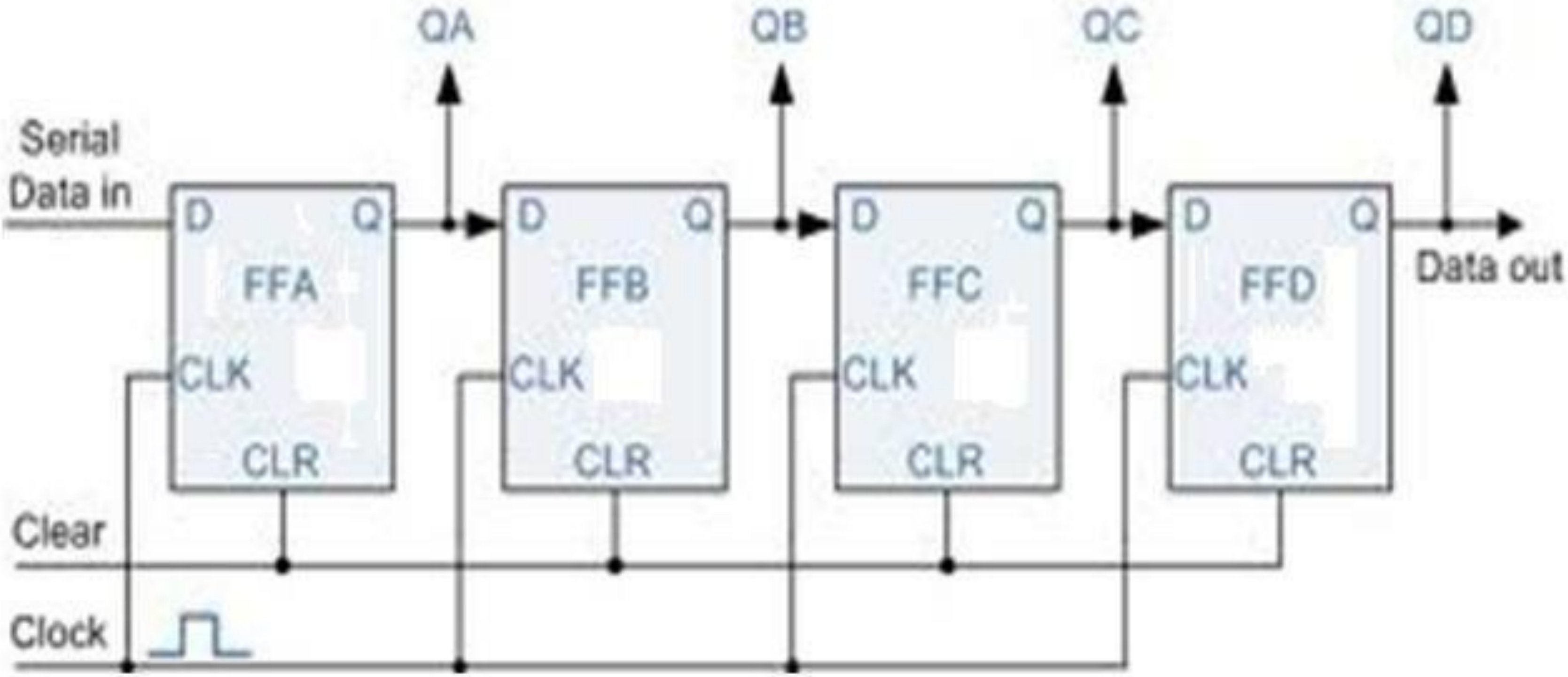
Figure 3 *n*-bit Left-Shift Serial-in Serial-Out Shift Register

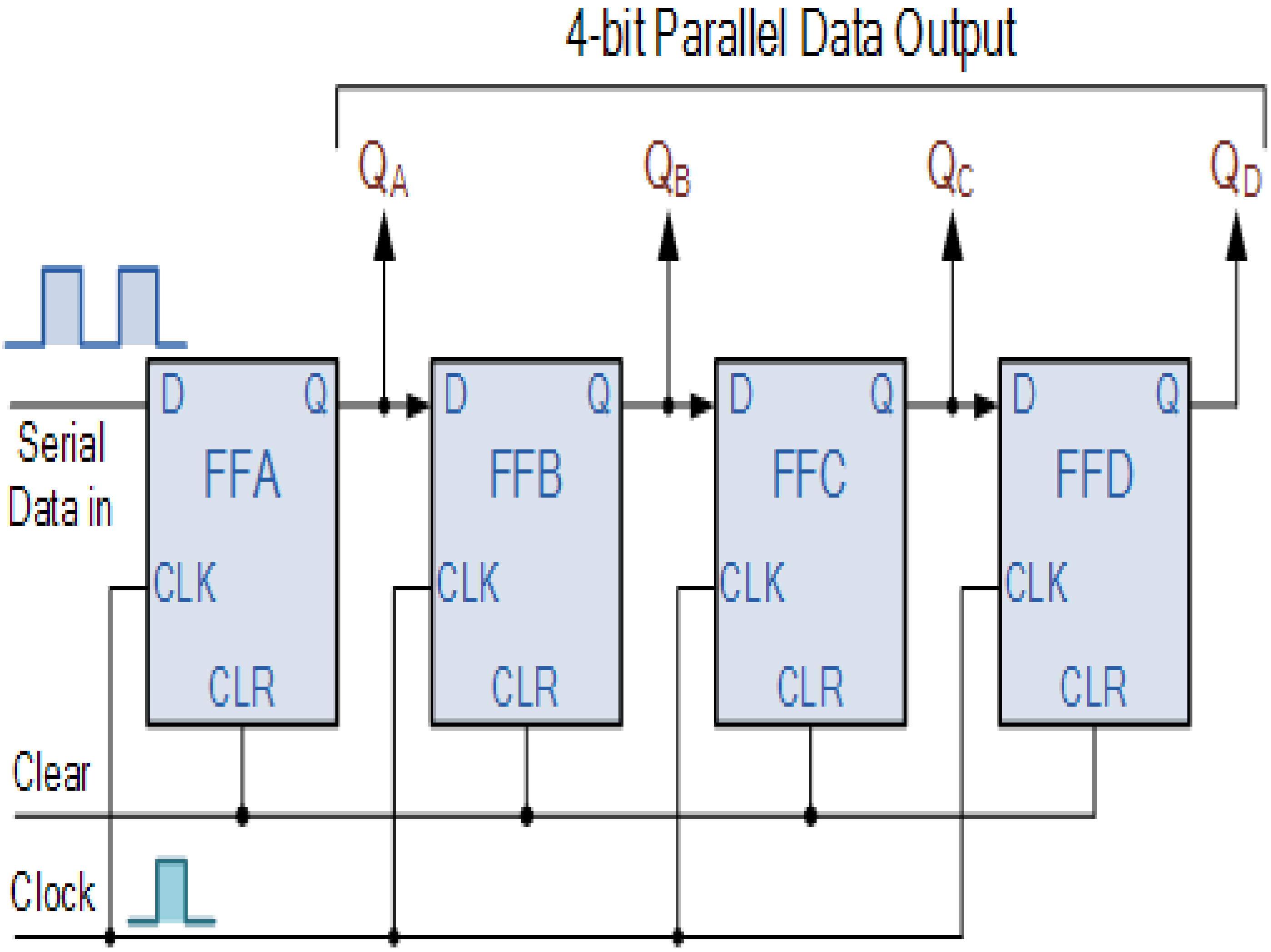
2.Serial-in, parallel-out, shift register

- The shift register, which allows serial input and produces parallel output is known as Serial In – Parallel Out **(SIPO)** shift register.



Serial-in, parallel-out, shift register:

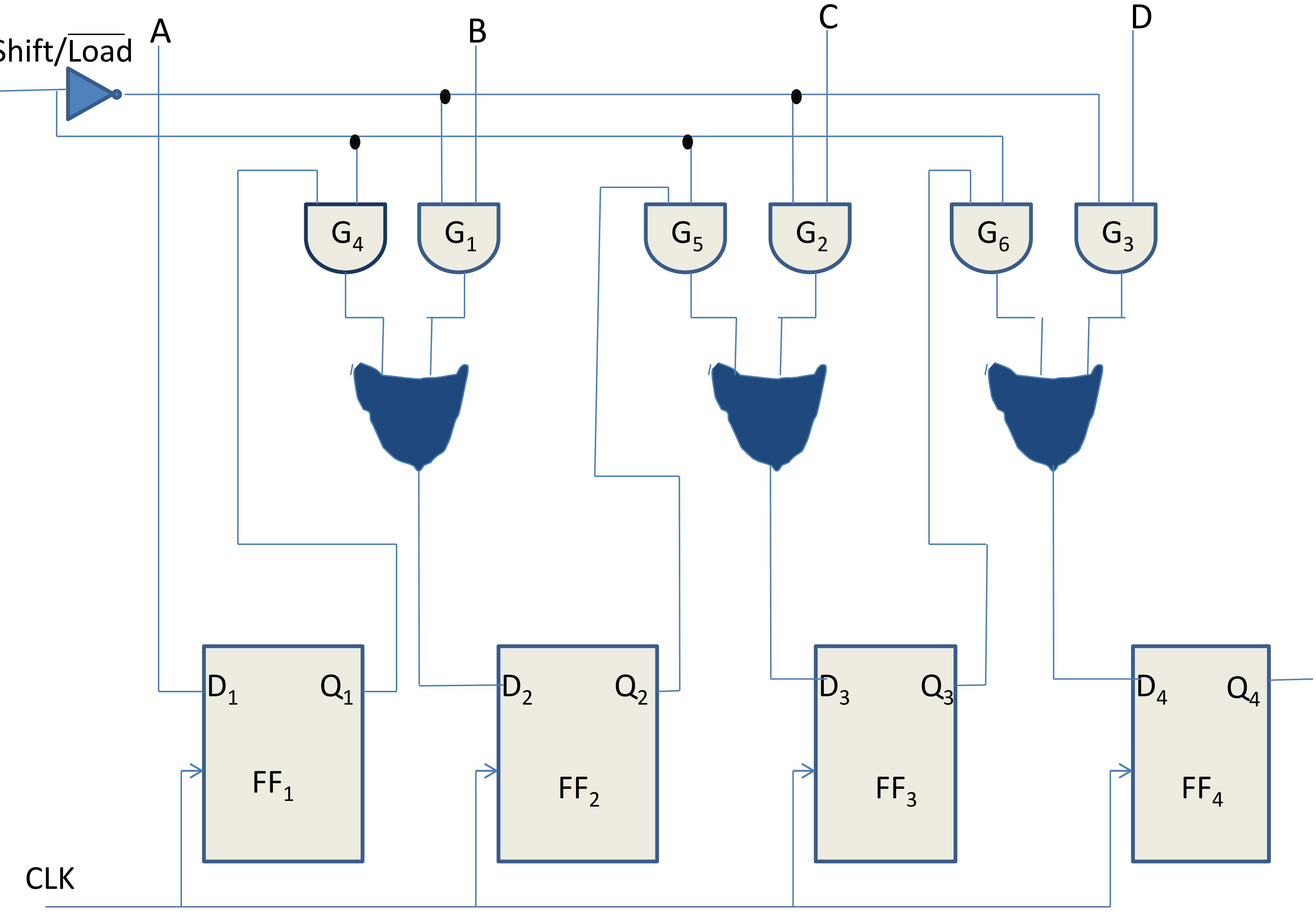



4-bit Parallel Data Output
Q_A
Q_B
Q_C
Q_D
D
Q
FFA
CLK
CLR
D
Q
FFB
CLK
CLR
D
Q
FFC
CLK
CLR
D
Q
FFD
CLK
CLR
Serial Data in
Clear
Clock

- In this type of register, the data bits are entered into the register serially, but the data stored in the register is shifted out in parallel form.
- Once the data bits are stored, each bit appears on its respective output line and all bits are available simultaneously, rather than on a bit-by-bit basis with the serial output.
- For every positive edge triggering of clock signal, the data shifts from one stage to the next.
- In this case, we can access the outputs of each D flip-flop in parallel. So, we will get **parallel outputs** from this shift register.

- The serial-in, parallel out, shift register can be used as serial-in, serial out, shift register if the output is taken from the Q terminal of the last FF.

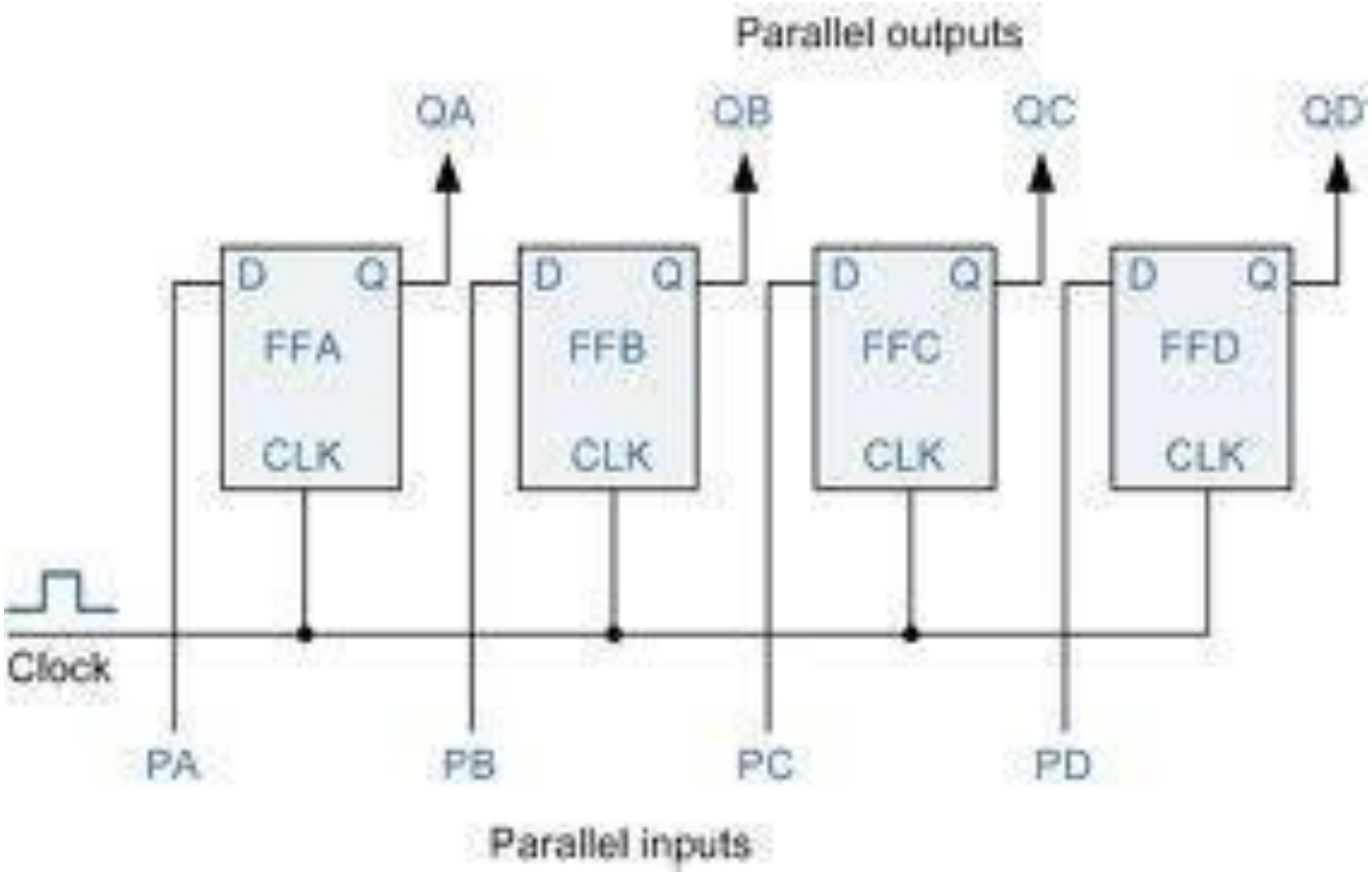
3.Parallel-in, Serial-out, Shift register



LOGIC DIAGRAM

- For a parallel-in, serial out, shift register, the data bits are entered simultaneously into their respective stages on parallel lines, rather than on a bit-by-bit basis on one line as with serial data bits are transferred out of the register serially.
- On a bit-by-bit basis over a single line.
- There are four data lines A,B,C,D through which the data is entered into the register in parallel form.
- The signal shift/ load allows the data to be entered in parallel form into the register and the data is shifted out serially from terminal Q4.

Parallel-in, parallel-out, shift register

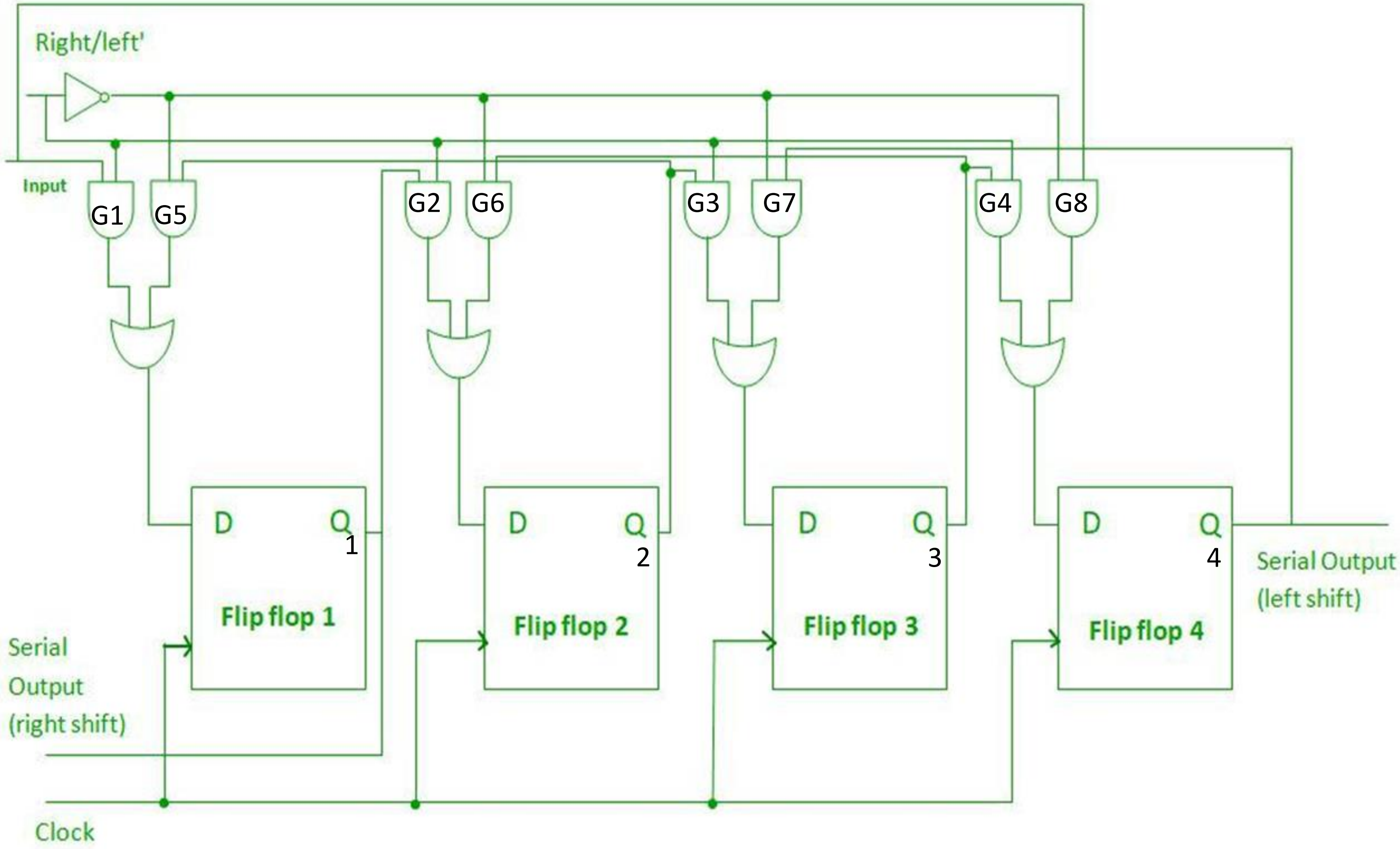


- In a parallel-in, parallel-out shift register, the data is entered into the register in parallel form, and also the data is taken out of the register in parallel form.
- Data is applied to the D input terminals of the FF‘s.
- When a clock pulse is applied, at the positive going edge of the pulse, the D inputs are shifted into the Q outputs of the FFs.
- The register now stores the data.
- The stored data is available instantaneously for shifting out in parallel form.

Bidirectional shift register

- A bidirectional shift register is one which the data bits can be shifted from left to right or from right to left.
- A fig shows the logic diagram of a 4-bit serial-in, serial out, bidirectional shift register.
- Right/left is the mode signal, when right /left is a 1, the logic circuit works as a shift-register.
- The bidirectional operation is achieved by using the mode signal and two NAND gates and one OR gate for each stage.

Figure: logic diagram of a 4-bit bidirectional shift register



- A HIGH on the Right/$\overline{\text{Left}}$ control input enables the AND gates G1, G2, G3 and G4 and disables the AND gates G5,G6,G7 and G8, and the state of Q output of each FF is passed through the gate to the D input of the following FF.
- When a clock pulse occurs, the data bits are then effectively shifted one place to the right.
- A LOW on the Right/$\overline{\text{Left}}$ control inputs enables the AND gates G5, G6, G7 and G8 and disables the And gates G1, G2, G3 and G4 and the Q output of each FF is passed to the D input of the preceding FF.
- When a clock pulse occurs, the data bits are then effectively shifted one place to the left. Hence, the circuit works as a bidirectional shift register

Universal shift register:

Universal shift register:

- A register capable of shifting in one direction only is a unidirectional shift register.
- One that can shift both directions is a bidirectional shift register.
- If the register has both shifts and parallel load capabilities, it is referred to as a universal shift registers.
- So a Universal shift register is a bidirectional register, whose input can be either in serial form or in parallel form and whose output also can be in serial form or in parallel form.
- The most general shift register has the following **capabilities**:

1. A clear control to clear the register to 0

2. A clock input to synchronize the operations

3. A shift-right control to enable the shift-right operation and serial input and output lines associated with the shift-right

4. A shift-left control to enable the shift-left operation and serial input and output lines associated with the shift-left

5. A parallel loads control to enable a parallel transfer and the n input lines associated with the parallel transfer

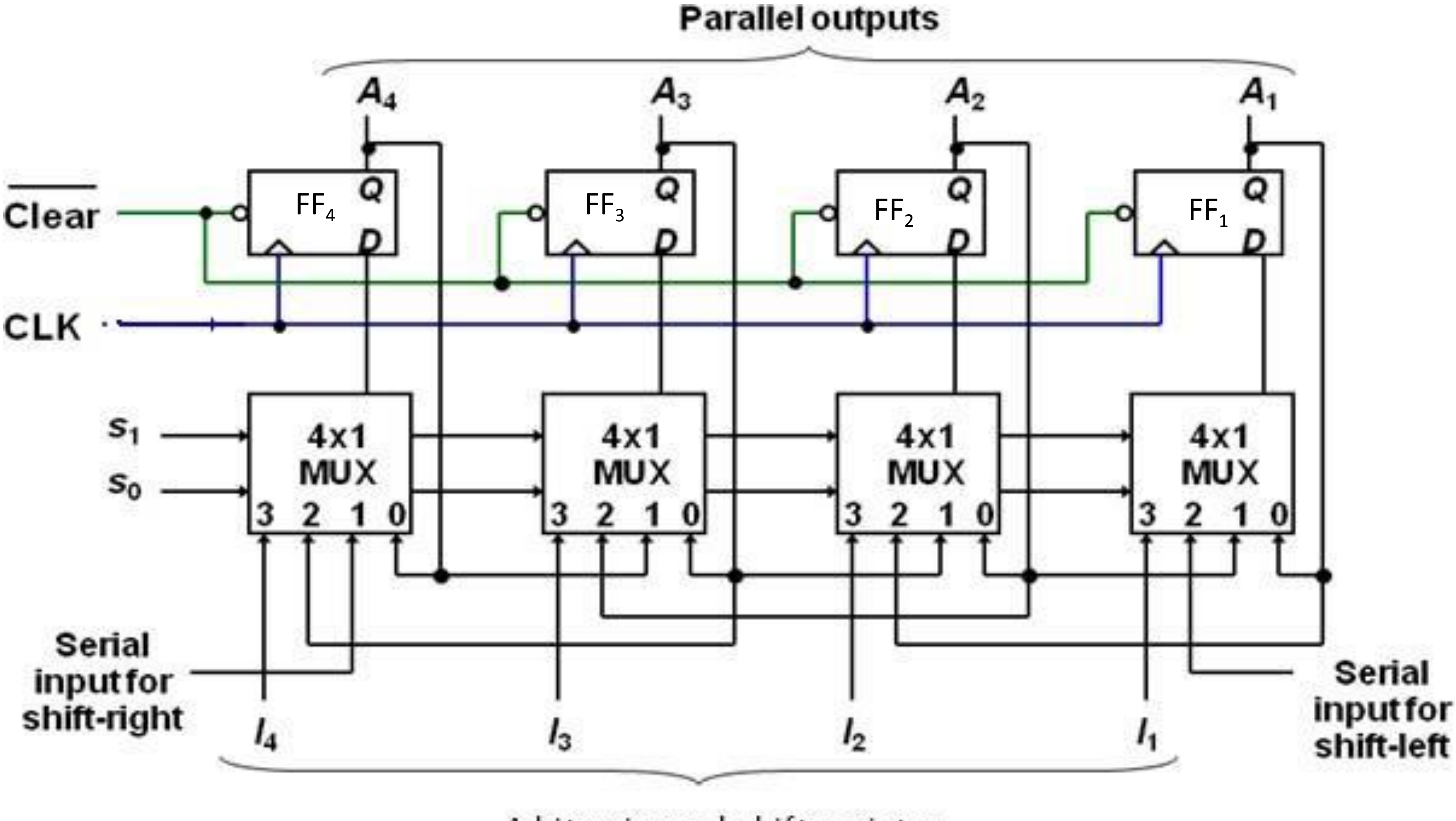
6. N parallel output lines

7. A control state that leaves the information in the register unchanged in the presence of the clock.

- A universal shift register can be realized using multiplexers.
- The below fig shows the logic diagram of a 4-bit universal shift register that has all capabilities.
- It consists of 4 D flip-flops and four multiplexers.
- The four multiplexers have two common selection inputs s1 and s0.
- Input 0 in each multiplexer is selected when S1S0=00, input 1 is selected when S1S0=01 and input 2 is selected when S1S0=10 and input 4 is selected when S1S0=11.
- The selection inputs control the mode of operation of the register according to the functions entries.

- When S1S0=0, the present value of the register is applied to the D inputs of flip-flops.
- The condition forms a path from the output of each flip-flop into the input of the same flip-flop.
- The next clock edge transfers into each flip-flop the binary value it held previously, and no change of state occurs.
- When S1S0=01, terminal 1 of the multiplexer inputs have a path to the D inputs of the flip-flop. This causes a shift-right operation, with serial input transferred into flip-flop A4(FF4).
- When S1S0=10, a shift left operation results with the other serial input going into flip-flop A1(FF1).
- Finally when S1S0=11, the binary information on the parallel input lines is transferred into the register simultaneously during the next clock edge.

Figure: logic diagram 4-bit universal shift register



4-bit universal shift register

Function table for theregister

mode control

S0	**S1**	**register operation**
0	0	No change
0	1	Shift Right
1	0	Shift left
1	1	Parallel load

APPLICATIONS OF SHIFT REGISTERS

- Registers are often used to momentarily store binary information appearing at the output of an encoding matrix.
- A register might be used to accept input data from an alphanumeric keyboard and then present the data at the input of a microprocessor chip.
- Shift Registers are often used to momentarily store binary data at the output of a decoder.
- A shift register also forms the basis for some very important arithmetic operations.
- A shift register can also be connected to form a number of different types of counters. These counters offer some very distinct advantages.

APPLICATIONS OF SHIFT REGISTERS

- Time delays : In many digital systems, it is necessary to delay the transfer of data until such time as operations on other data have been completed, or to synchronize the arrival of data at a subsystem where it is processed with other data. A shift register can be used to delay this arrival of serial data.
- Serial/Parallel data conversion
- Ring counters (Shift Register Counters)
- Universal Asynchronous Receiver Transmitter(UART): A UART is a specially designed integrated circuit that contains all the registers and synchronizing circuitry necessary to receive data in serial form and to convert and transmit it in parallel form and vice versa.

Shift register counters

- One of the applications of shift register is that they can be arranged to form several types of counters.
- The most widely used shift register counter is ring counter as well as the twisted ring counter.
- **Ring counter:** this is the simplest shift register counter.
- The basic ring counter using D flip flops is shown in fig. the realization of this counter using JK FFs.
- The Q output of each stage is connected to the D flip-flop connected back to the ring counter.

4-bit Ring Counter State diagram

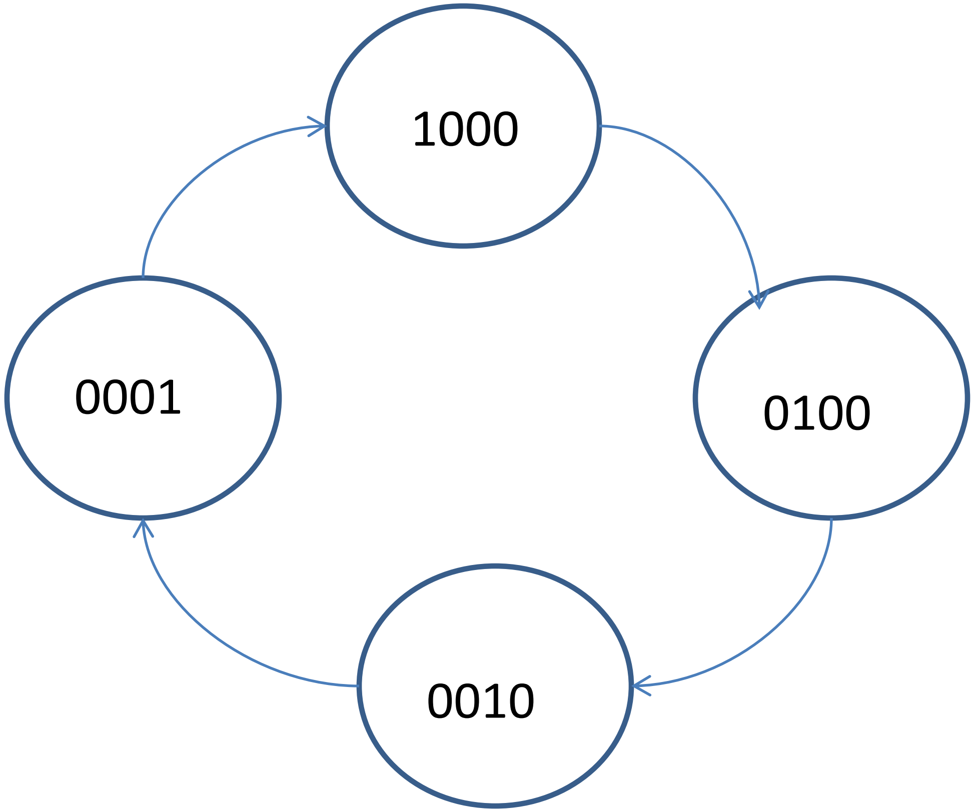
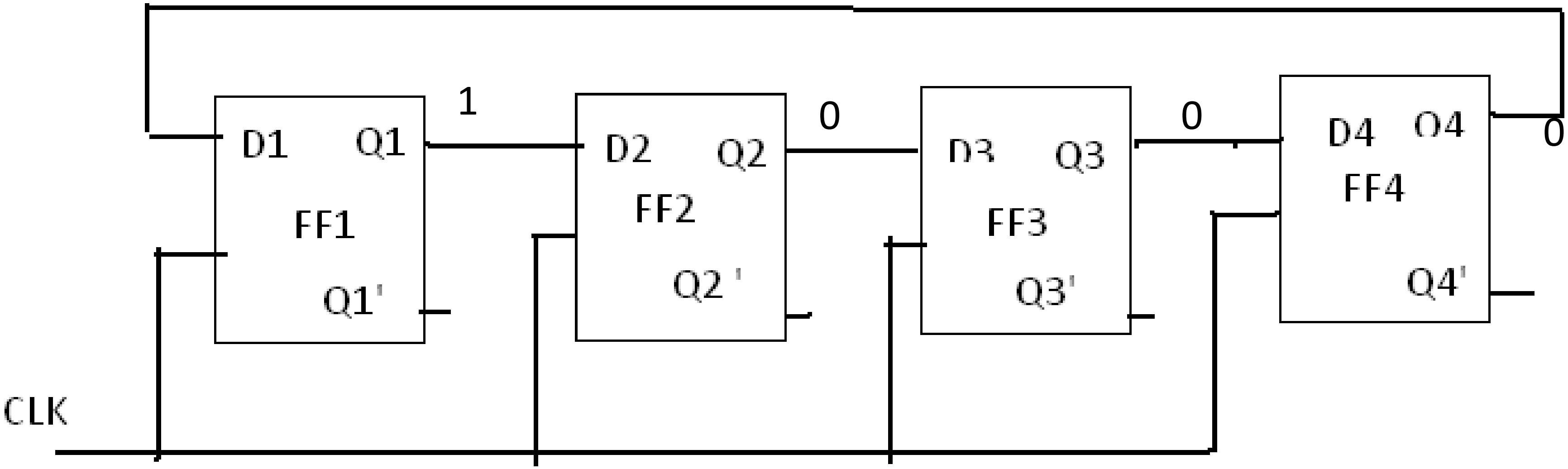


FIGURE: logic diagram of 4-bit ring counter using D flip-flops



CLK=0	Q1Q2Q3Q4	1 0 0 0
CLK=1		0 1 0 0
CLK=2		0 0 1 0
CLK=3		0 0 0 1
CLK=4		1 0 0 0
CLK=5		

SEQUENCE TABLE

AFTER CLOCK PULSE	Q1	Q2	Q3	Q4
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	0	0
5	0	1	0	0
6	0	0	1	0
7	0	0	0	1

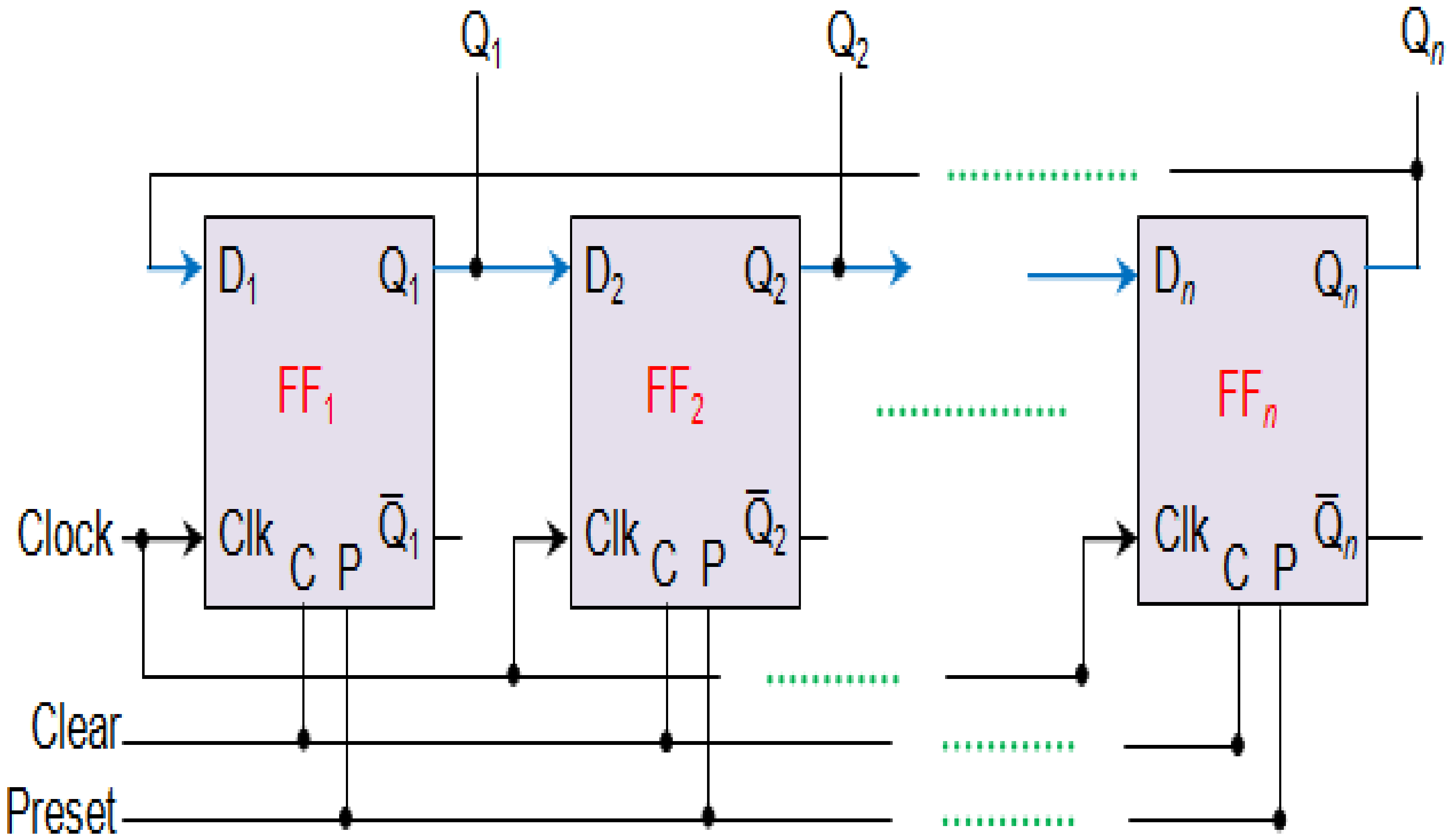


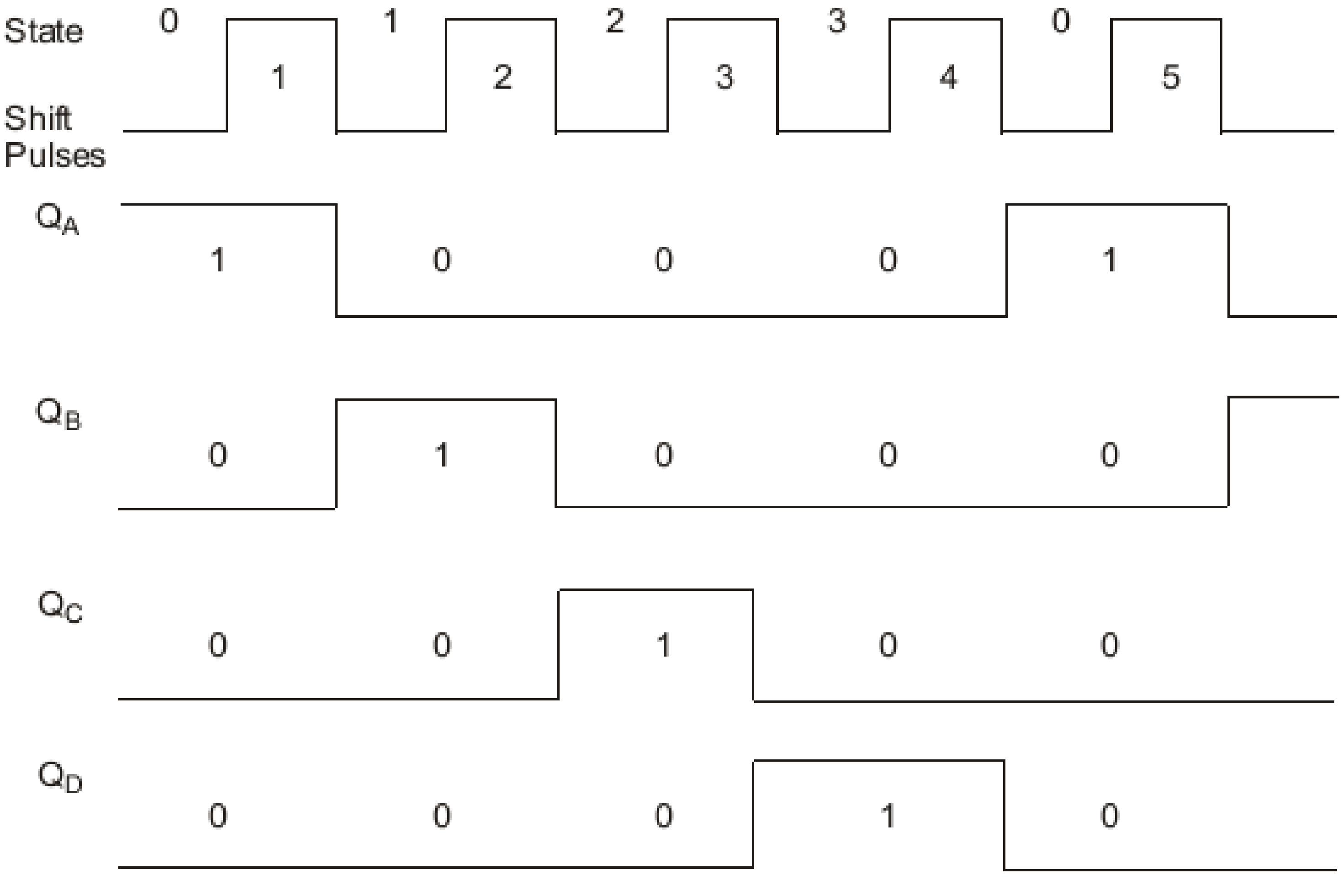
Figure 1 *n*-bit Ring Counter Designed Using D Flip-Flops

- Only a single 1 is in the register and is made to circulate around the register as long as clock pulses are applied.
- Initially the first FF is present to a 1. So, the initial state is 1000, i.e.,

  Q1=1, Q2=0,Q3=0,Q4=0.
- After each clock pulse, the contents of the register are shifted to the right by one bit and Q4 is shifted back to Q1.
- The sequence repeats after four clock pulses. The number of distinct states in the ring counter, i.e., the mod of the ring counter is equal to number of FFs used in the counter.

- An n-bit ring counter can count only n bits, where as n-bit ripple counter can count 2n bits.
- So, the ring counter is uneconomical compared to a ripple counter but has advantage of requiring no decoder, since we can read the count by simply noting which FF is set.
- Since it is entirely a synchronous operation and requires no gates external FFs, it has the further advantage of being very fast.

Timing diagram:



Twisted Ring counter (Johnson counter)

- This counter is obtained from a serial-in, serial-out shift register by providing feedback from the inverted output of the last FF to the D input of the first FF.
- The Q output of each is connected to the D input of the next stage, but the Q‘ output of the last stage is connected to the D input of the first stage, therefore, the name **twisted ring counter.**
- This feedback arrangement produces a unique sequence of states.

- Let initially all the FFs be reset, i.e., the state of the counter be 0000.
- After each clock pulse, the level of Q1 is shifted to Q2, the level of Q2to Q3, Q3 to Q4 and the level of Q4‘to Q1 and the sequences given in fig.
- The logic diagram of a 4-bit Johnson counter using D FF is shown in fig. the realization of the same using J-K FFs is shown in fig.
- The state diagram and the sequence table are shown in figure.
- The timing diagram of a Johnson counter is shown in figure.

State Diagram for Twisted Ring Counter

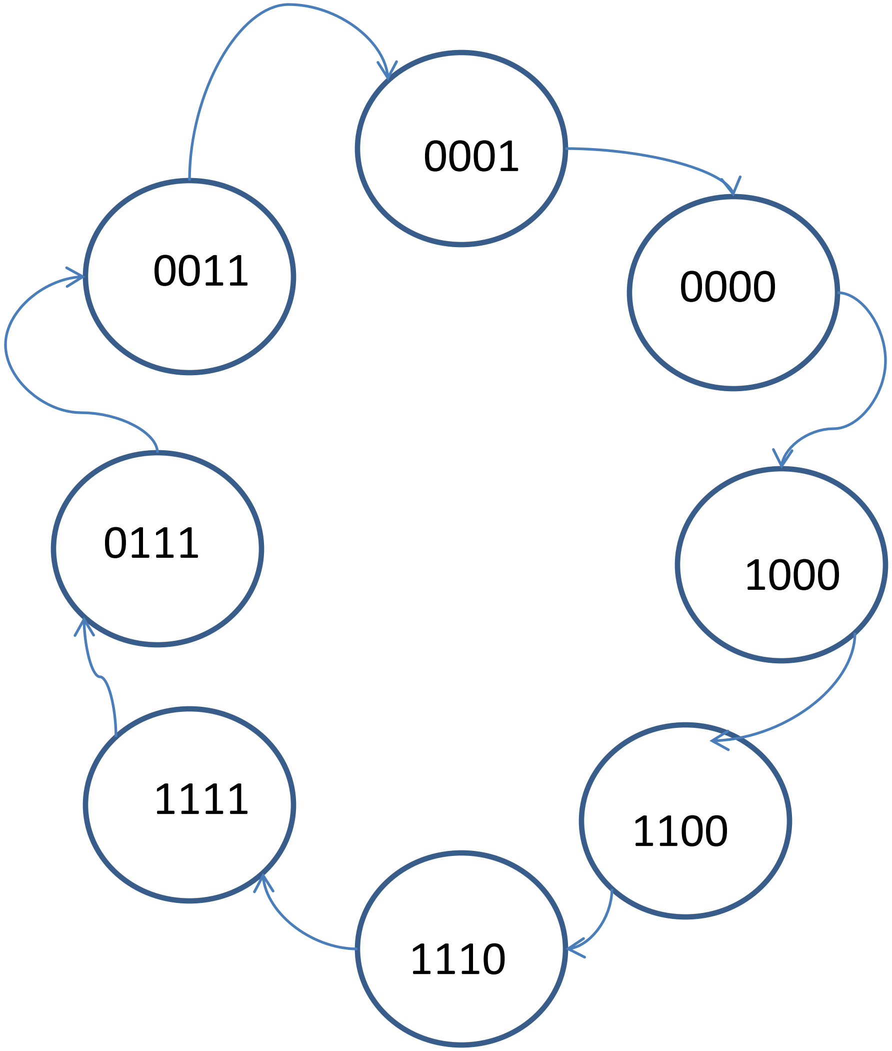
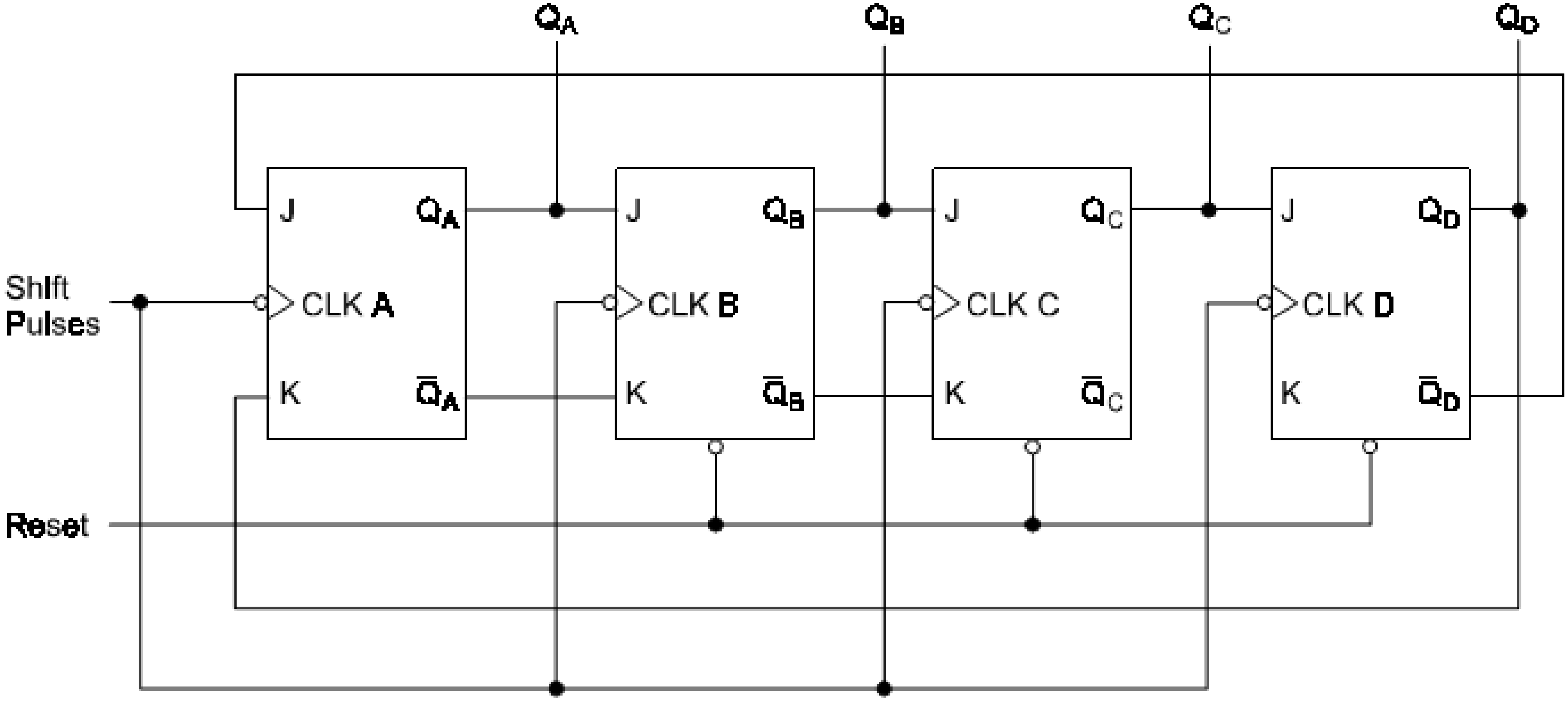
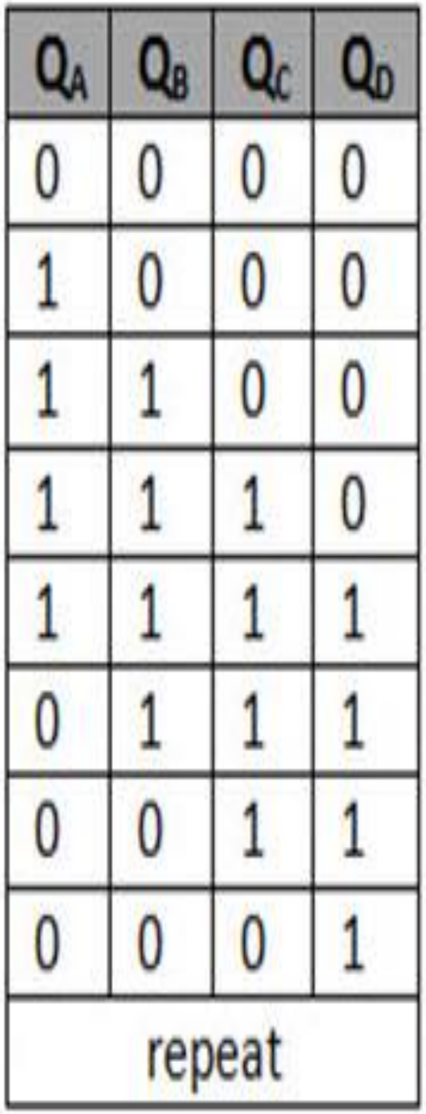
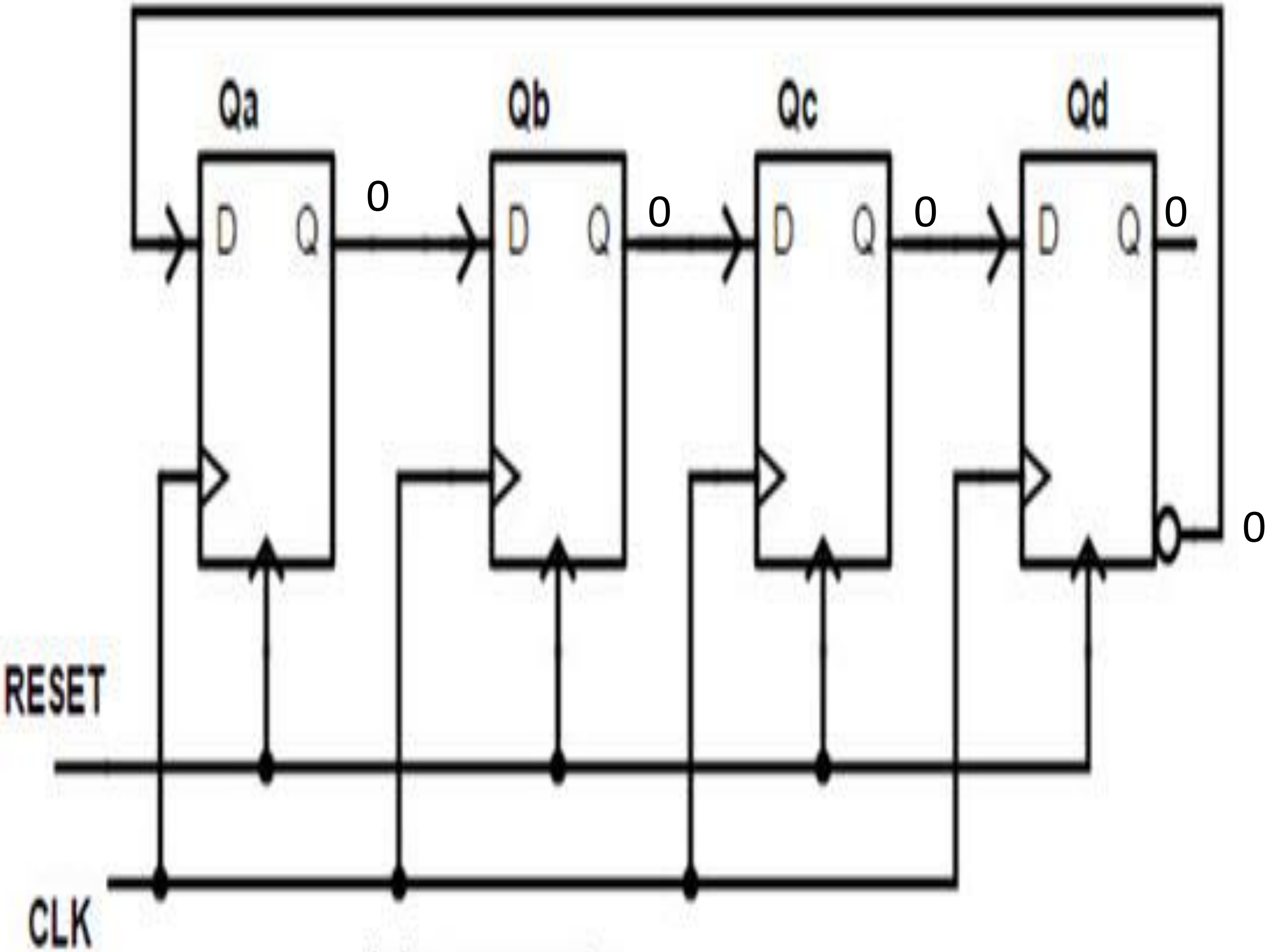


Figure: Johnson counter with JK flip-flops





Q_A	Q_B	Q_C	Q_D
0	0	0	0
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1
repeat			



Johnson counter

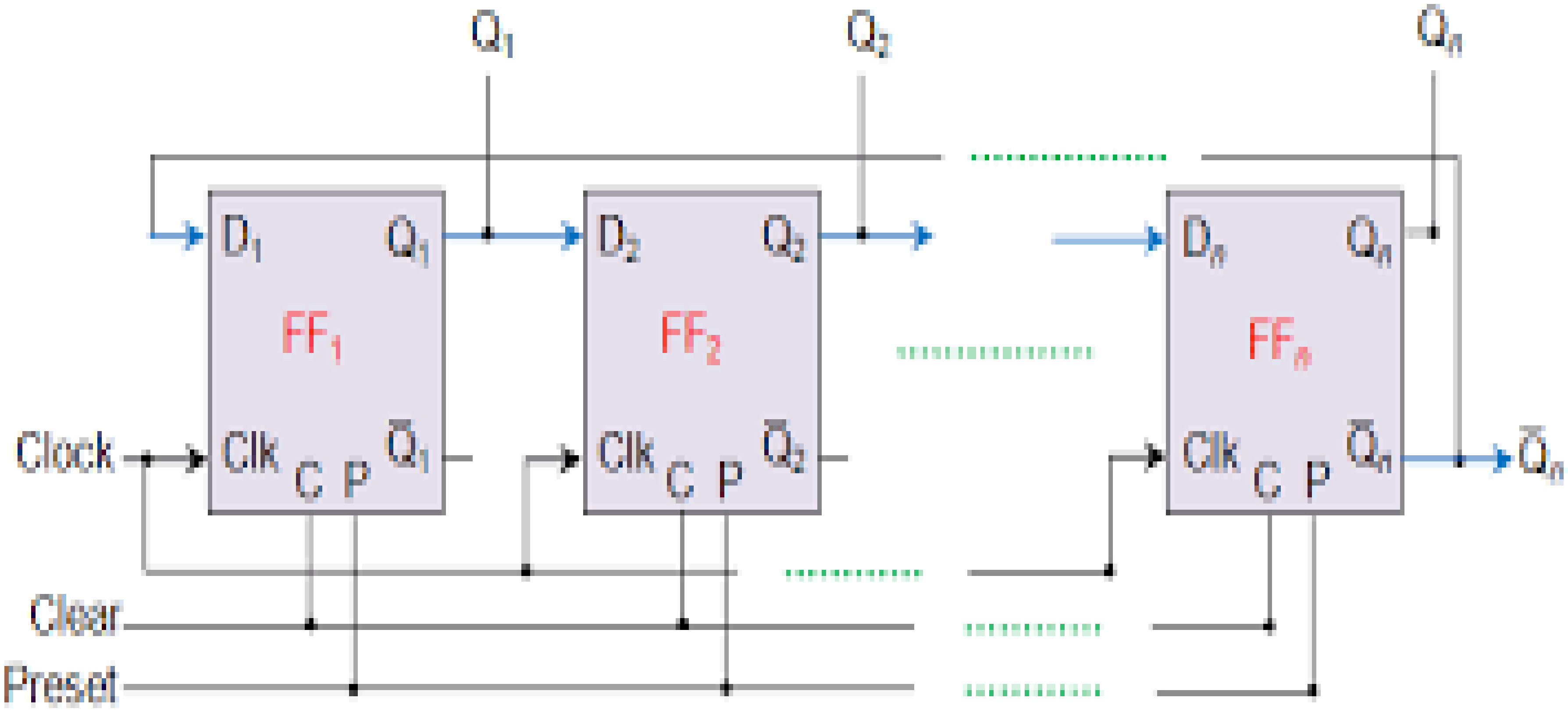
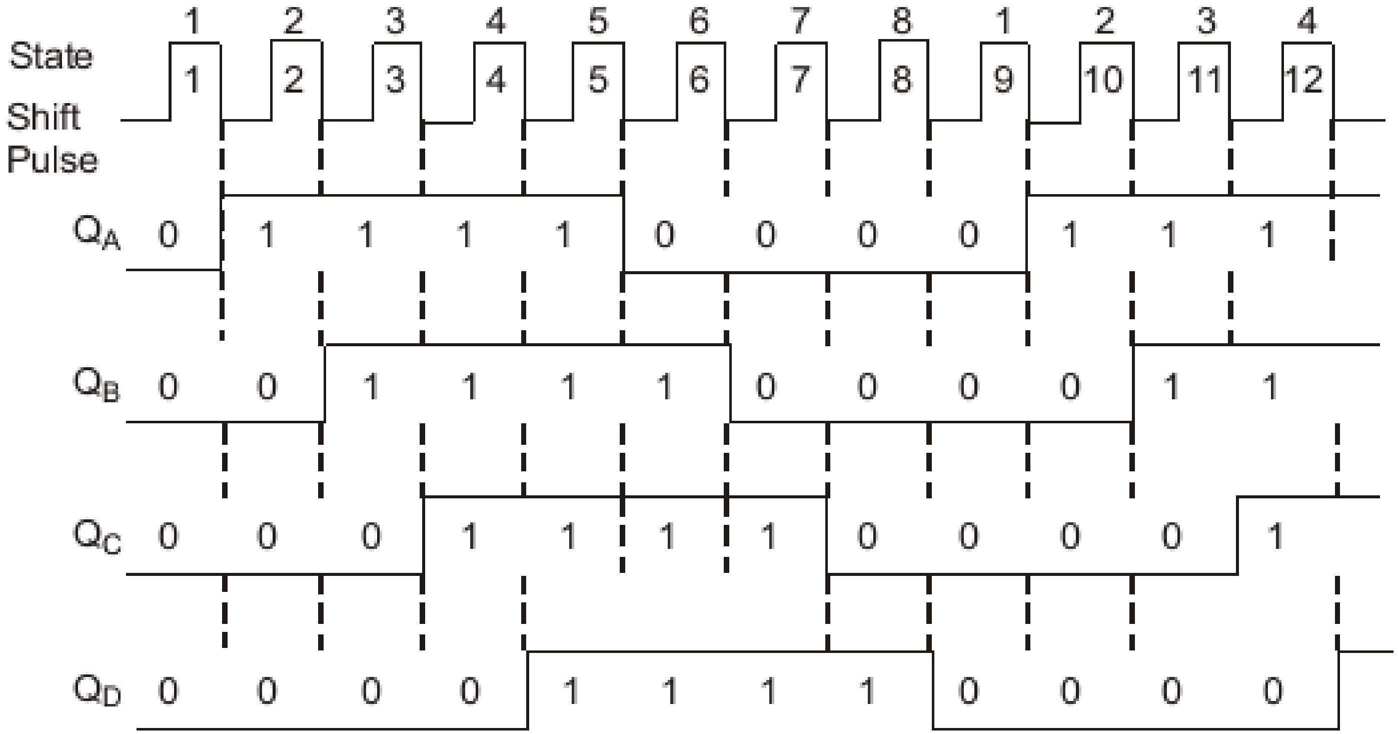


Figure 1 *n*-bit Johnson Counter Designed Using D Flip-Flops

Figure: timing diagram


1 2 3 4 5 6 7 8 1 2 3 4
State
1 2 3 4 5 6 7 8 9 10 11 12
Shift Pulse
Q_A 0 1 1 1 1 0 0 0 0 1 1 1
Q_B 0 0 1 1 1 1 0 0 0 0 1 1
Q_C 0 0 0 1 1 1 1 0 0 0 0 1
Q_D 0 0 0 0 1 1 1 1 0 0 0 0