Chapter: Shift Registers

Paper: Digital Systems and Applications

Lesson Developer: Ms. Chanchal Yadav

College: Rajdhani College, University of Delhi

Table of Contents

Chapter: Shift Registers

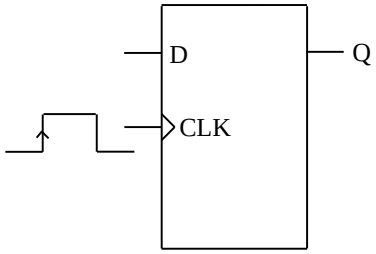
1 Introduction

Registers are the sequential logic circuits which are used mainly to store binary data (multiple 1's and 0's). They are also used to transfer data in digital systems. Registers do not have specific sequence of states. They shift the contents of registers once in every clock cycle so called as shift registers.

Shift registers consist of an arrangement of flip-flops. As a flip-flop has one bit memory so the storage capacity of a register is determined by the number of flip-flops used. An n-bit register is a collection of n D flip-flops or stages with a common clock input. So a 4-bit register contains four flip-flops and used to store four bits. Shift registers allow transfer of data from a stage to another within the register or into or out of the register, as per the clock input.

Basic concept

Clocked D flip-flop

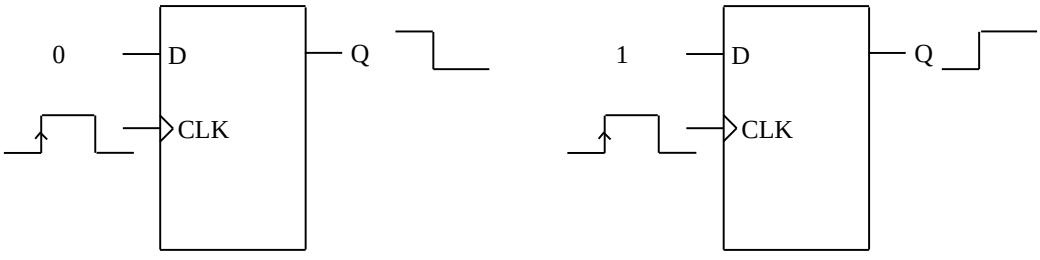


CLK	D	Q(n+1)
↑	0	0
↑	1	1

Positive edge triggered JK flip-flop symbol and truth table.

- If D=1 is applied at the positive going edge of clock then Q becomes 1 (flip-flop is set, (a)) or remains same if it was already 1.
- If D=0 is applied at the positive going edge of clock then Q becomes 0 (flip-flop is reset, (b)) or remains same if it was already 0.

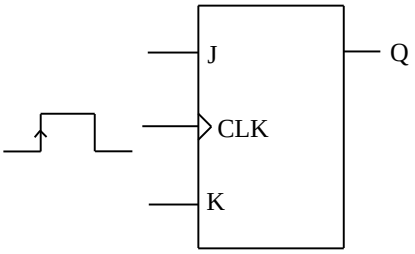
Hence, D flip-flop stores one bit of data and can be called as a single-bit register.



D flip-flop as a single-bit register.

Basic concept

Clocked JK flip-flop

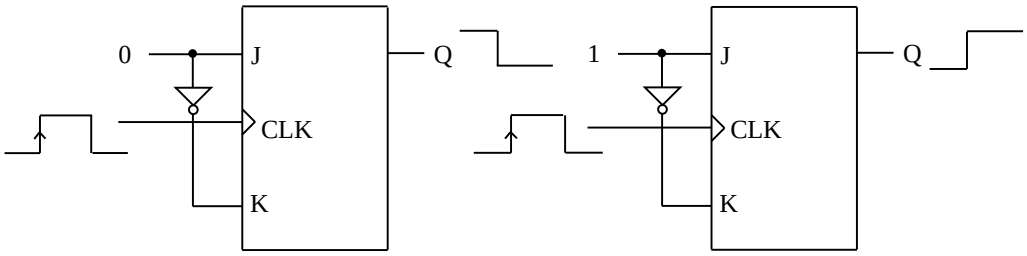


CLK	J	K	Q(n+1)
↑	0	0	Q(n) (no change)
↑	0	1	0
↑	1	0	1
↑	1	1	$\overline{Q}(n)$ (toggles)

Positive edge triggered JK flip-flop symbol and truth table.

- If J=1 and K=0 is applied at the positive going edge of clock then Q becomes 1 (flip-flop is set, (a)) or remains same if it was already 1.
- If J=0 and K=1 is applied at the positive going edge of clock then Q becomes 0 (flip-flop is reset, (b)) or remains same if it was already 0.

Hence, JK flip-flop stores one bit of data and can be called as a single-bit register.



JK flip-flop as a single-bit register.

2 Type of shift registers

Shift registers can be classified according to the way in which data is entered into the register for storage and the way in which data is outputted from the register (Figure 1). Various types of shift registers are listed below:

1. Serial Input/Serial Output shift register (SISO)
2. Serial Input/Parallel Output shift register (SIPO)
3. Parallel Input/Serial Output shift register (PISO)
4. Parallel Input/Parallel Output shift register (PIPO)

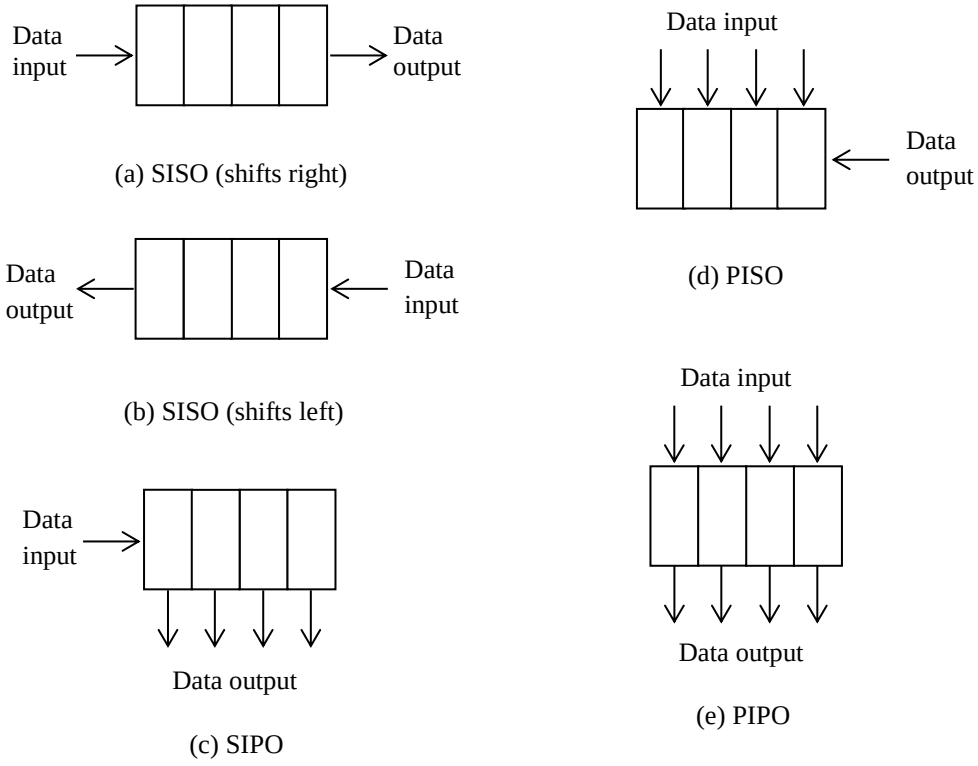


Figure 1. Different ways to transfer data in registers.

2.1 SISO shift register

Data input comes in serially means one bit at a time and stored output goes out serially, bit by bit on one line.

2.1.1 Construction

Let us consider a 4-bit SISO shift register designed using four positive edge triggered D flip-flops as shown in Figure 2. Serial data input is placed on D_0 input of first flip-flop (FF0), Q_0 output of first flip-flop goes to D_1 input of second flip-flop (FF1), Q_1 output of second flip-flop goes to D_2 input of third flip-flop (FF2), Q_2 output of third flip-flop goes to D_3 input of fourth flip-flop (FF3) and Q_3 output of fourth flip-flop is taken out. A 4-bit SISO shift register has one serial input (D_0) and one serial output (Q_3).

2.1.2 Working

Assume that the register is initially clear (means $Q_3=Q_2=Q_1=Q_0=0$, Figure 3(a)).

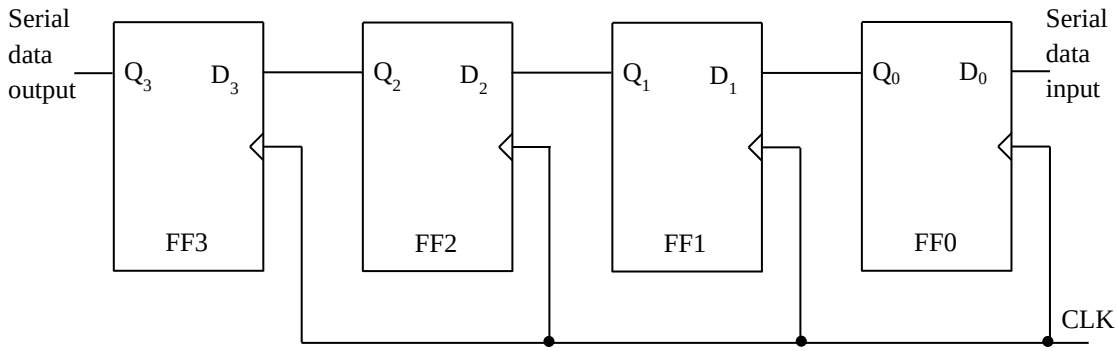


Figure 2. Logic diagram of 4-bit SISO shift register using D flip-flops.

The steps to store four bits (1010) into the register are:

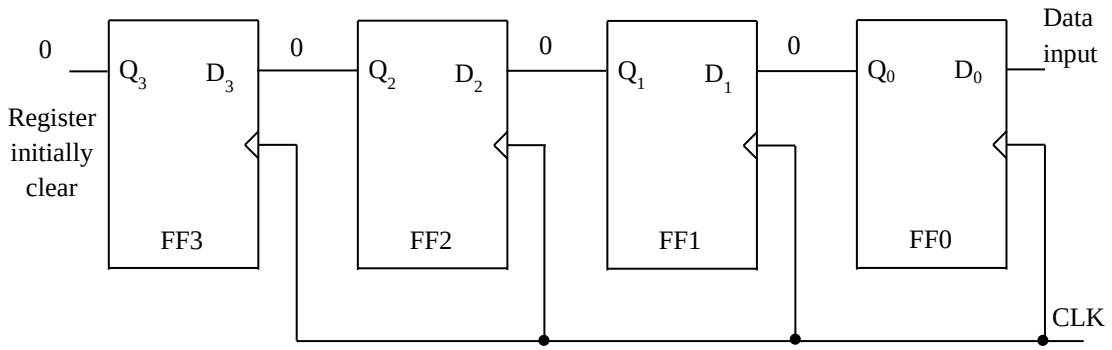
1. Let us place bit by bit from left to right starting with left most bit. So first of all, 1 is placed on the data input line making $D_0=1$ for FF0 (Figure 3(b)). The first positive going edge of clock sets first flip-flop (means FF0 stores the data input i.e., $Q_0=1$).

2. Next, the second bit which is a 0, is placed on the data input line making $D_0=0$ for FF0 and $D_1=1$ for FF1 as D_1 input of FF1 is connected to Q_0 output (Figure 3(c)). The second positive going edge of clock sets second flip-flop (means FF1 stores the output of FF0 i.e., $Q_1=1$) and resets first flip-flop (means FF0 stores the data input i.e., $Q_0=0$). So the data stored in FF0 is shifted to FF1 and FF0 stores the input data.

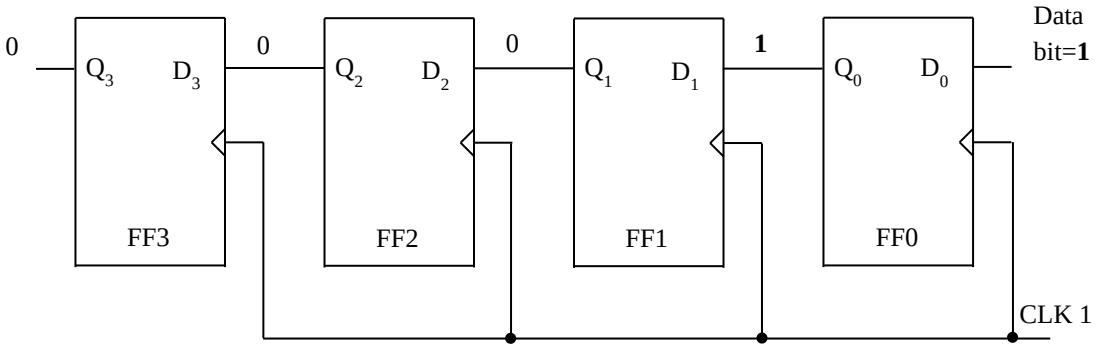
3. Again, the third bit which is a 1, is placed on the data input line making $D_0=1$ for FF0, $D_1=0$ for FF1 as D_1 input of FF1 is connected to Q_0 output and $D_2=1$ for FF2 as D_2 input of FF2 is connected to Q_1 output (Figure 3(d)). The third positive going edge of clock sets third flip-flop (means FF2 stores the output of FF1 i.e., $Q_2=1$), resets second flip-flop (means FF1 stores the output of FF0 i.e., $Q_1=0$) and sets first flip-flop (means FF0 stores the data input i.e., $Q_0=1$). So the data stored in FF1 is shifted to FF2, data stored in FF0 is shifted to FF1 and FF0 stores the input data.

4. The last bit which is a 0, is placed on the data input line making $D_0=0$ for FF0, $D_1=1$ for FF1 as D_1 input of FF1 is connected to Q_0 output, $D_2=0$ for FF2 as D_2 input of FF2 is connected to Q_1 output and $D_3=1$ for FF3 as D_3 input of FF3 is connected to Q_2 output (Figure 3(e)). The fourth positive going edge of clock sets fourth flip-flop (means FF3 stores the output of FF2 i.e., $Q_3=1$), resets third flip-flop (means FF2 stores the output of FF1 i.e., $Q_2=0$), sets second flip-flop (means FF1 stores the output of FF0 i.e., $Q_1=1$) and resets first flip-flop (means FF0 stores the data input i.e., $Q_0=0$). So the data stored in FF2 is shifted to FF3, data stored in FF1 is shifted to FF2, data stored in FF0 is shifted to FF1 and FF0 stores the input data. That's how four bits are serially entered into the shift register and stored.

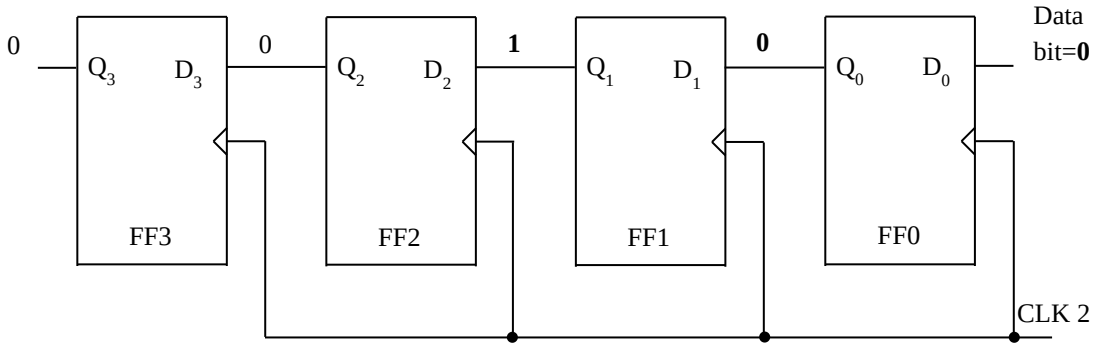
To get remaining data output, bits are further shifted serially out and become available at Q_3. After fourth positive going edge of clock, we have left most bit at Q_3 (means $Q_3=1$, Figure 3(e)). Similarly after fifth, sixth and seventh positive going edge of clock, we have $Q_3=0$ (Figure 3(f)), $Q_3=1$ (Figure 3(g)) and $Q_3=0$ (Figure 3(h)) respectively. After eighth positive going edge of clock, the register is clear again (Figure 3(i)).


0
Register initially clear
Q3 D3
FF3
0
Q2 D2
FF2
0
Q1 D1
FF1
0
Q0 D0
FF0
Data input
CLK

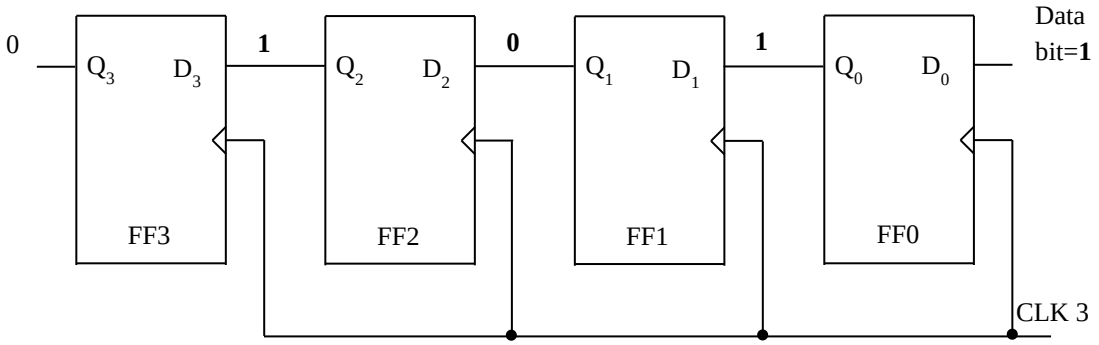
(a)


0
Q3 D3
FF3
0
Q2 D2
FF2
0
Q1 D1
FF1
1
Q0 D0
FF0
Data bit=1
CLK 1

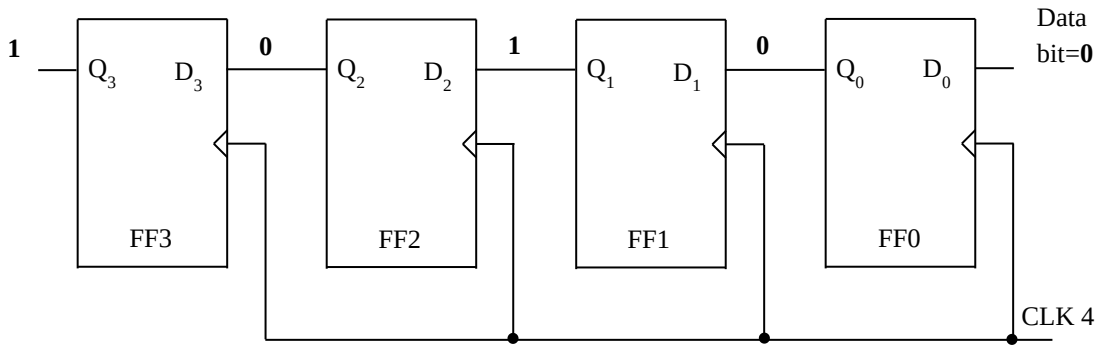
(b)


0
Q3 D3
FF3
0
Q2 D2
FF2
1
Q1 D1
FF1
0
Q0 D0
FF0
Data bit=0
CLK 2

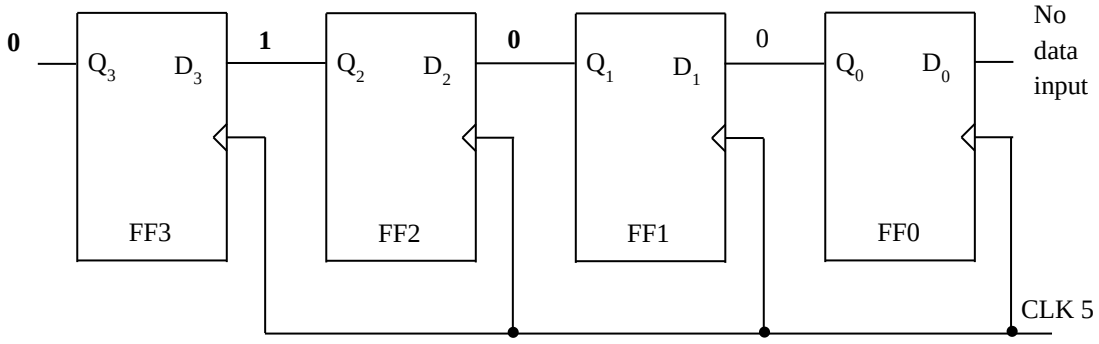
(c)


0
Q3 D3
FF3
1
Q2 D2
FF2
0
Q1 D1
FF1
1
Q0 D0
FF0
Data bit=1
CLK 3

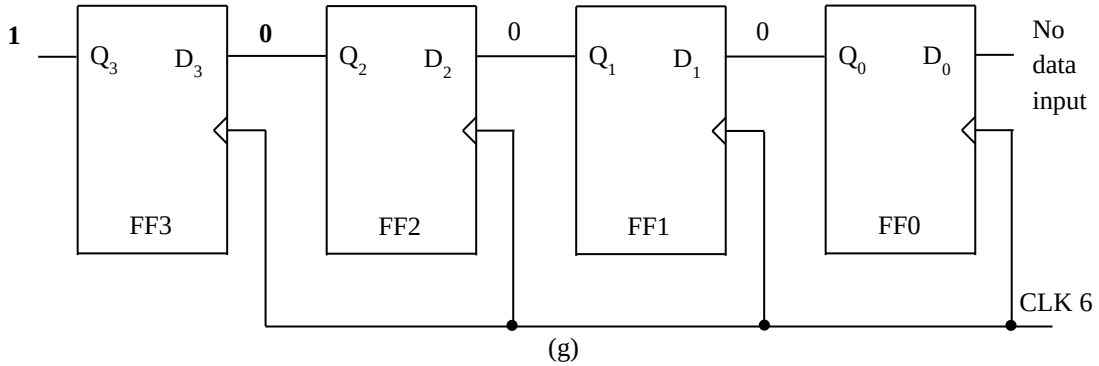
(d)


1
0
1
0
Data bit=0
Q3 D3
Q2 D2
Q1 D1
Q0 D0
FF3
FF2
FF1
FF0
CLK 4

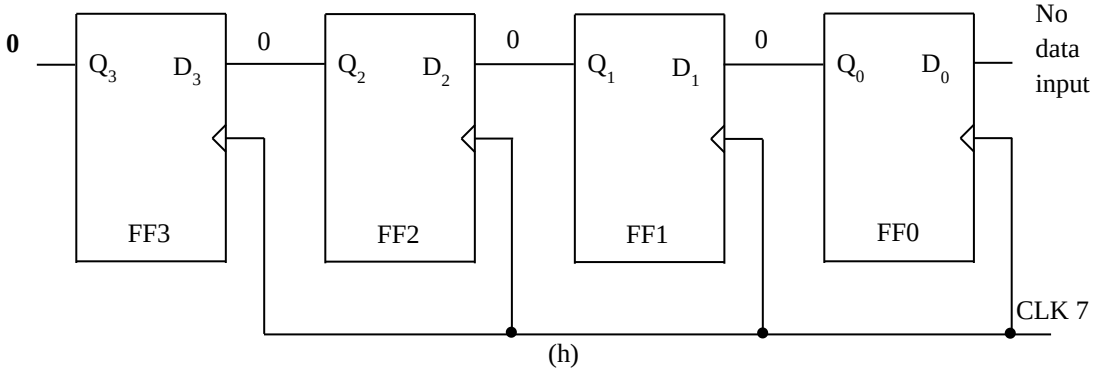
(e)


0
1
0
0
No data input
Q3 D3
Q2 D2
Q1 D1
Q0 D0
FF3
FF2
FF1
FF0
CLK 5

(f)


1
0
0
0
No data input
Q3 D3
Q2 D2
Q1 D1
Q0 D0
FF3
FF2
FF1
FF0
CLK 6

(g)


0
0
0
0
No data input
Q3 D3
Q2 D2
Q1 D1
Q0 D0
FF3
FF2
FF1
FF0
CLK 7

(h)

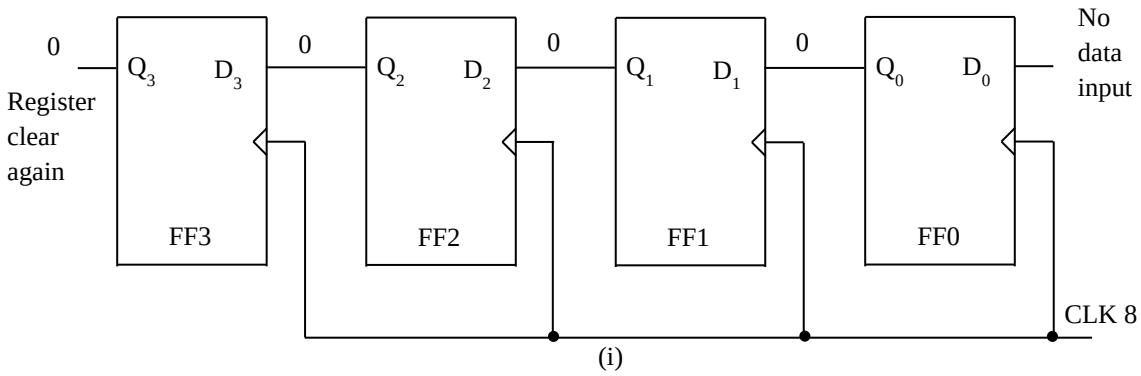


Figure 3. Data flow into and out of 4-bit SISO shift register.

2.1.3 Timing diagram

The timing diagram of 4-bit SISO shift register is shown in Figure 4 and its stepwise description is as follows:

1. The first positive going edge of clock changes Q_0 output of FF0 from 0 to 1 (as data input D_0=1) and Q_1 output of FF1, Q_2 output of FF2 as well as Q_3 output of FF3 remain unchanged.

2. The second positive going edge of clock changes Q_0 output of FF0 from 1 to 0 (as data input D_0=0), Q_1 output of FF1 changes from 0 to 1 (as D_1=1) and Q_2 output of FF2 as well as Q_3 output of FF3 remain unchanged.

3. The third positive going edge of clock changes Q_0 output of FF0 from 0 to 1 (as data input D_0=1), Q_1 output of FF1 changes from 1 to 0 (as D_1=0), Q_2 output of FF2 changes from 0 to 1 (as D_2=1) and Q_3 output of FF3 remains unchanged.

4. The fourth positive going edge of clock changes Q_0 output of FF0 from 1 to 0 (as data input D_0=0), Q_1 output of FF1 changes from 0 to 1 (as D_1=1), Q_2 output of FF2 changes from 1 to 0 (as D_2=0) and Q_3 output of FF3 changes from 0 to 1 (as D_3=1).

5. The fifth positive going edge of clock leaves Q_0 output of FF0 unchanged (as no data input), changes Q_1 output of FF1 from 1 to 0 (as D_1=0), Q_2 output of FF2 changes from 0 to 1 (as D_2=1) and Q_3 output of FF3 changes from 1 to 0 (as D_3=0).

6. The sixth positive going edge of clock leaves Q_0 output of FF0 as well as Q_1 output of FF1 unchanged (as no data input), Q_2 output of FF2 changes from 1 to 0 (as D_2=0) and Q_3 output of FF3 changes from 0 to 1 (as D_3=1).

7. The seventh positive going edge of clock leaves Q_0 output of FF0, Q_1 output of FF1 as well as Q_2 output of FF2 unchanged (as no data input) and changes Q_3 output of FF3 from 1 to 0 (as D_3=0).

8. The eighth positive going edge of clock leaves Q_0 output of FF0, Q_1 output of FF1, Q_2 output of FF2 as well as Q_3 output of FF3 unchanged (as no data input) so the register is clear again.

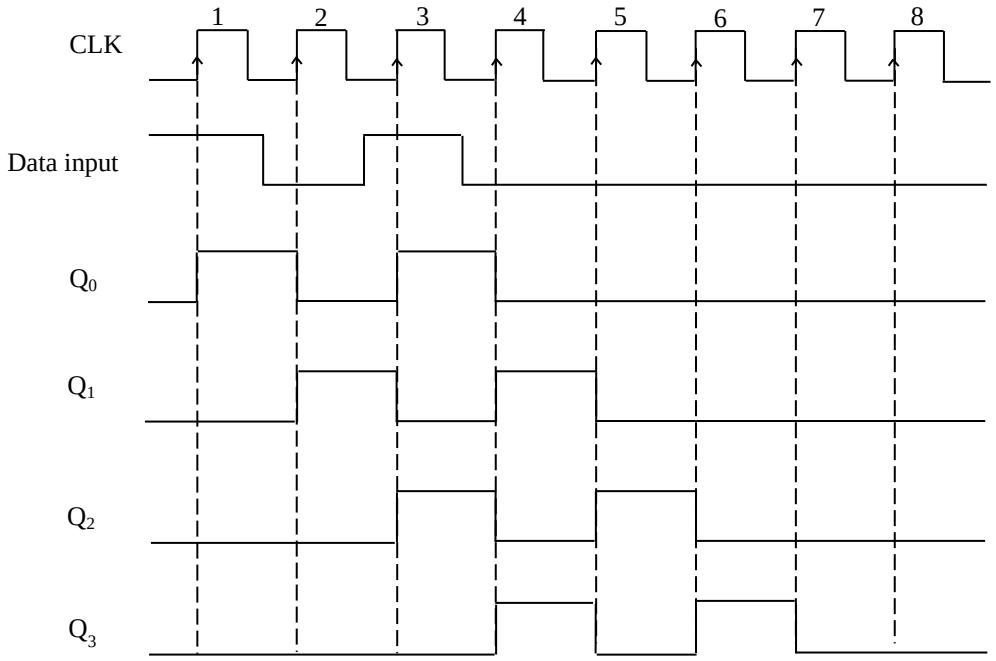


Figure 4. Timing diagram of 4-bit SISO shift register using D flip-flops.

2.1.4 Logic symbol

Figure 5 shows the logic symbol for 4-bit SISO shift register where D_0 is the serial data input and Q_3 is the serial data output.

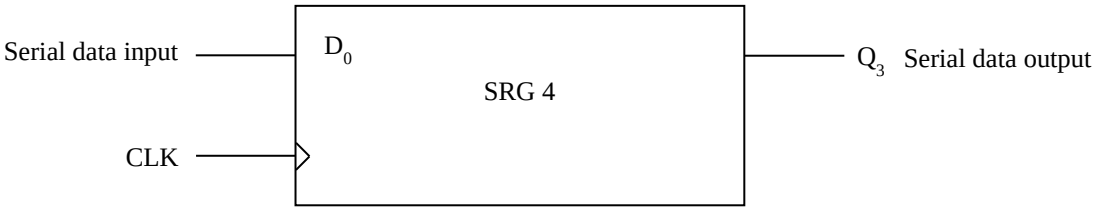


Figure 5. Logic symbol for 4-bit SISO shift register.

Interesting fact

- For n-bit SISO shift register, serial output is nothing but serial input delayed by n clock cycle. The binary data is entered serially into the register and shifts through it with the clock. Data is available serially as clock progresses.
- Data input given at D_0 input is not directly available at Q_3 output because of finite propagation delay of flip-flop but becomes available after a finite number of clock cycles.

2.1.5 Application: Ring counter

A ring counter is designed by modifying a SISO shift register. Ring counter is a shift register counter which uses feedback connection to connect the serial output of last flip-flop to the serial input of first flip-flop means Q_3 output of last flip-flop goes to D_0 input of first flip-flop. It has a sequence of states so called as counter.

A mod-n ring counter uses n flip-flops. It is also called as divide by n counter where n is the number of stages. The logic diagram of mod-4 ring counter using four positive edge triggered D flip-flops is shown in Figure 6. The output of each flip-flop is connected to the input of next flip-flop means Q_0 output of first flip-flop (FF0) goes to D_1 input of second flip-flop (FF1), Q_1 output of FF1 goes to D_2 input of third flip-flop (FF2) and Q_2 output of FF2 goes to D_3 input of fourth flip-flop (FF3) except last stage where Q_3 output of FF3 goes to D_0 input of FF0.

Mostly, a ring counter starts with only one flip-flop in 1 state and others in 0 state means one flip-flop must be preset and others should be clear in the counter. Initially assume that first flip-flop is preset and other flip-flops are clear (means $Q_0=1$ and $Q_1=Q_2=Q_3=0$), so 0001 is the starting state.

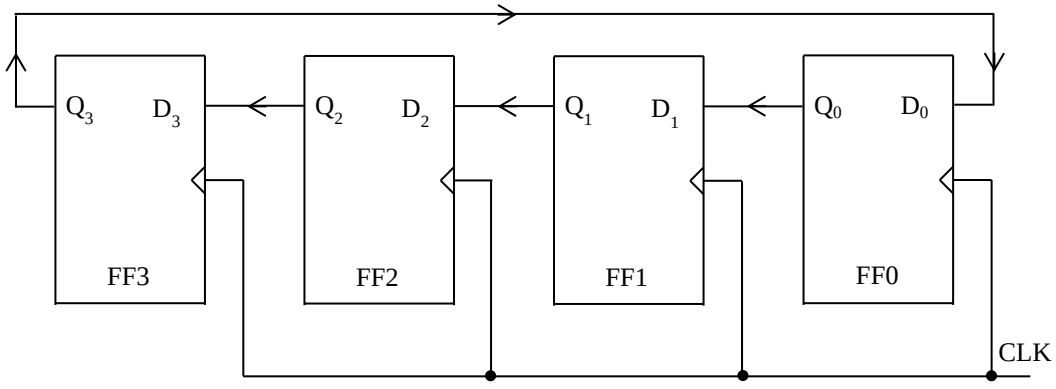


Figure 6. Logic diagram of mod-4 ring counter using D flip-flops.

The sequence of states of a ring counter with the clock is as follows:

1. As D_0 input of FF0 is connected to Q_3 output of FF3 so D_0=0 and as D_1 input of FF1 is connected to Q_0 output of FF0 so D_1=1. The first positive going edge of clock resets first flip-flop, sets second flip-flop and leaves third as well as fourth flip-flop unchanged (means the 1 goes from FF0 to FF1 i.e., Q_1=1 and Q_0=Q_2=Q_3=0).

2. As D_1 input of FF1 is connected to Q_0 output of FF0 so D_1=0 and as D_2 input of FF2 is connected to Q_1 output of FF1 so D_2=1. The second positive going edge of clock resets second flip-flop, sets third flip-flop and leaves first as well as fourth flip-flop unchanged (means the 1 goes from FF1 to FF2 i.e., Q_2=1 and Q_0=Q_1=Q_3=0).

3. As D_2 input of FF2 is connected to Q_1 output of FF1 so D_2=0 and as D_3 input of FF3 is connected to Q_2 output of FF2 so D_3=1. The third positive going edge of clock resets third flip-flop, sets fourth flip-flop and leaves first as well as second flip-flop unchanged (means the 1 goes from FF2 to FF3 i.e., Q_3=1 and Q_0=Q_1=Q_2=0).

4. As D_3 input of FF3 is connected to Q_2 output of FF2 so D_3=0 and as D_0 input of FF0 is connected to Q_1 output of FF1 so D_0=1. The fourth positive going edge of clock sets first flip-flop, resets fourth flip-flop and leaves second as well as third flip-flop unchanged (means the 1 goes from FF3 to FF0 i.e., Q_0=1 and Q_1=Q_2=Q_3=0).

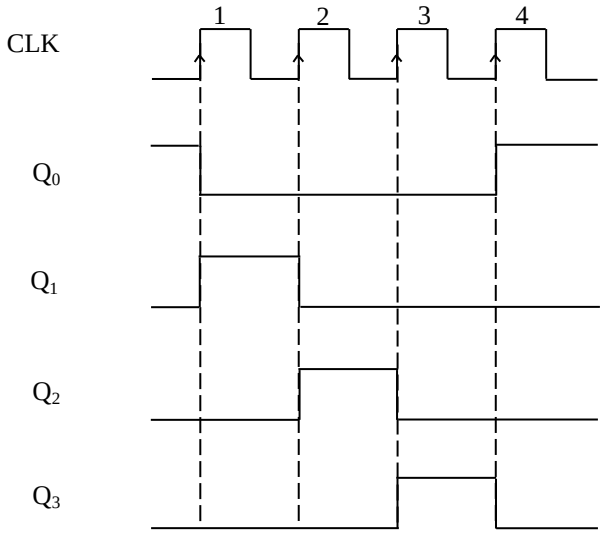


Figure 7. Timing diagram of mod-4 ring counter using D flip-flops.

Figure 7 shows the timing diagram of mod-4 ring counter and its stepwise description is as follows:

1. The first positive going edge of clock changes Q_0 output of FF0 from 1 to 0, Q_1 output of FF1 changes from 0 to 1 and Q_2 output of FF2 as well as Q_3 output of FF3 remain unchanged. So after first positive going edge of clock Q_0=0, Q_1=1, Q_2=0 and Q_3=0 (means 0010 state).

2. The second positive going edge of clock changes Q_1 output of FF1 from 1 to 0, Q_2 output of FF2 from 0 to 1 and Q_0 output of FF0 as well as Q_3 output of FF3 remain unchanged. So after second positive going edge of clock Q_0=0, Q_1=0, Q_2=1 and Q_3=0 (means 0100 state).

3. The third positive going edge of clock leaves Q_0 output of FF0 as well as Q_1 output of FF1 unchanged, Q_2 output of FF2 changes from 1 to 0 and Q_3 output of FF3 changes from 0 to 1. So after third positive going edge of clock Q_0=0, Q_1=0, Q_2=0 and Q_3=1 (means 1000 state).

4. The fourth positive going edge of clock changes Q_0 output of FF0 from 0 to 1, Q_3 output of FF3 changes from 1 to 0 and Q_1 output of FF1 as well as Q_2 output of FF2 remain unchanged. So after fourth positive going edge of clock Q_0=1, Q_1=0, Q_2=0 and Q_3=0 (means 0001 state), which is the initial state.

Each positive going edge of clock advances 1 in the register by one stage (Table 1). It allows data to shift from right to left and back around from Q_3 to Q_0 (circular shift). As 1 circulates around in the counter so it is called as ring counter. No decoding gates are required in case of a ring counter as its output is unique for each state.

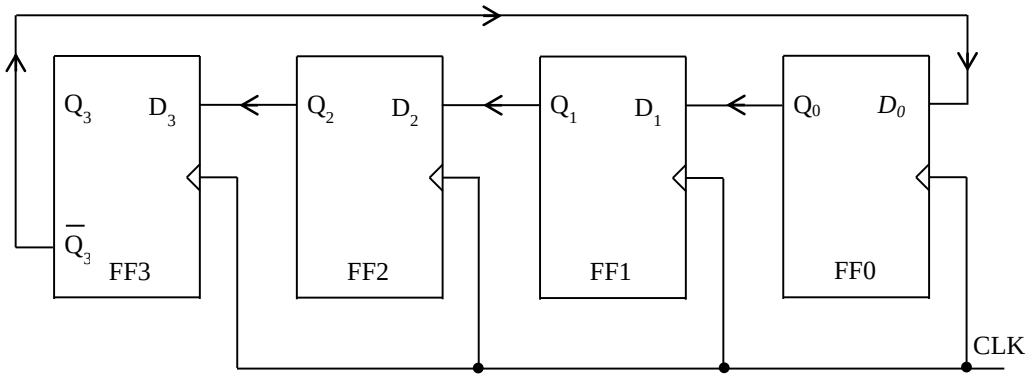
Table 1. Sequence of states for mod-4 ring counter.

After clock edge	Q_3	Q_2	Q_1	Q_o
0	0	0	0	**1**
1	0	0	**1**	0
2	0	**1**	0	0
3	**1**	0	0	0
4	0	0	0	**1**

Interesting fact

Johnson counter

- Johnson counter is also known as twisted ring counter. Johnson counter is a shift register based counter designed by modifying a SISO shift register such that $\overline{Q}_3$ output of last flip-flop goes to D_0 input of first flip-flop.
- The output of each flip-flop is connected to the input of next flip-flop means Q_0 output of FF0 goes to D_1 input of FF1, Q_1 output of FF1 goes to D_2 input of FF2 and Q_2 output of FF2 goes to D_3 input of FF3 except last stage where $\overline{Q}_3$ output of FF3 goes to D_0 input of FF0.
- A johnson counter having n flip-flops, has a modulus of 2n.



Logic diagram of 4-bit johnson counter using D flip-flops.

Sequence of states for 4-bit johnson counter.				
After clock edge	**Q_3**	**Q_2**	**Q_1**	**Q_o**
0	0	0	0	**1**
1	0	0	**1**	**1**
2	0	**1**	**1**	**1**
3	**1**	**1**	**1**	**1**
4	**1**	**1**	**1**	0
5	**1**	**1**	0	0
6	**1**	0	0	0
7	0	0	0	0
8	0	0	0	**1**

2.2 SIPO shift register

Data input comes in serially means one bit at a time and stored output goes out in parallel means output of each stage is available. All bits are available simultaneously at the output line after the data is stored.

2.2.1 Construction

Let us consider a 4-bit SIPO shift register designed using four positive edge triggered D flip-flops as shown in Figure 8. Serial data input is placed on D_0 input of first flip-flop (FF0), Q_0 output of first flip-flop goes to D_1 input of second flip-flop (FF1), Q_1 output of second flip-flop goes to D_2 input of third flip-flop (FF2), Q_2 output of third flip-flop goes to D_3 input of fourth flip-flop (FF3) and output of each flip-flop is taken out. A 4-bit SIPO shift register has one serial input (D_0) and four parallel output (Q_3, Q_2, Q_1 and Q_0).

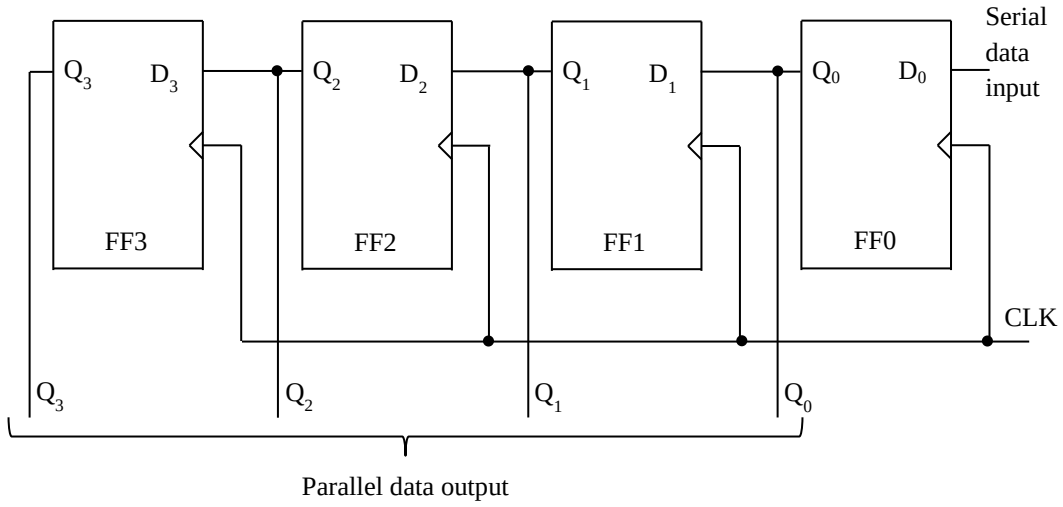


Figure 8. Logic diagram of 4-bit SIPO shift register using D flip-flops.

2.2.2 Working

Assume that the register is initially clear (means $Q_3=Q_2=Q_1=Q_0=0$). The steps to store four bits (1110) into the register are:

1. Let us place bitwise starting with left most bit. So first of all, 1 is placed on the data input line making $D_0=1$ for FF0. The first positive going edge of clock sets first flip-flop (means FF0 stores the data input i.e., $Q_0=1$).

2. Next, the second bit which is a 1, is placed on the data input line making $D_0=1$ for FF0 and $D_1=1$ for FF1 as D_1 input of FF1 is connected to Q_0 output. The second positive going edge of clock sets second flip-flop (means FF1 stores the output of FF0 i.e., $Q_1=1$) and leaves first flip-flop unchanged (means FF0 stores the data input i.e., $Q_0=1$). So the data stored in FF0 is shifted to FF1 and FF0 stores the input data.

3. Again, the third bit which is a 1, is placed on the data input line making $D_0=1$ for FF0, $D_1=1$ for FF1 as D_1 input of FF1 is connected to Q_0 output and $D_2=1$ for FF2 as

D_2 input of FF2 is connected to Q_1 output. The third positive going edge of clock sets third flip-flop (means FF2 stores the output of FF1 i.e., Q_2=1) and leaves first as well as second flip-flop unchanged (means FF1 stores the output of FF0 and FF0 stores the data input i.e., Q_0=Q_1=1). So the data stored in FF1 is shifted to FF2, data stored in FF0 is shifted to FF1 and FF0 stores the input data.

4. The last bit which is a 0, is placed on the data input line making D_0=0 for FF0, D_1=1 for FF1 as D_1 input of FF1 is connected to Q_0 output, D_2=1 for FF2 as D_2 input of FF2 is connected to Q_1 output and D_3=1 for FF3 as D_3 input of FF3 is connected to Q_2 output. The fourth positive going edge of clock sets fourth flip-flop (means FF3 stores the output of FF2 i.e., Q_3=1), leaves second as well as third flip-flop unchanged (means FF2 stores the output of FF1 and FF1 stores the output of FF0 i.e., Q_1=Q_2=1) and resets first flip-flop (means FF0 stores the data input i.e., Q_0=0). So the data stored in FF2 is shifted to FF3, data stored in FF1 is shifted to FF2, data stored in FF0 is shifted to FF1 and FF0 stores the input data. That's how four bits are serially entered into the shift register and stored. So after fourth positive going edge of clock, data output is available at Q_3, Q_2, Q_1 and Q_0.

2.2.3 Timing diagram

The timing diagram of 4-bit SIPO shift register is shown in Figure 9 and its stepwise description is as follows:

1. The first positive going edge of clock changes Q_0 output of FF0 from 0 to 1 (as data input D_0=1) and Q_1 output of FF1, Q_2 output of FF2 as well as Q_3 output of FF3 remain unchanged.

2. The second positive going edge of clock changes Q_1 output of FF1 from 0 to 1 (as D_1=1) and Q_0 output of FF0, Q_2 output of FF2 as well as Q_3 output of FF3 remain unchanged.

3. The third positive going edge of clock changes Q_2 output of FF2 from 0 to 1 (as D_2=1) and Q_0 output of FF0, Q_1 output of FF1 as well as Q_3 output of FF3 remain unchanged.

4. The fourth positive going edge of clock changes Q_0 output of FF0 from 1 to 0 (as data input D_0=0), Q_1 output of FF1 as well as Q_2 output of FF2 remain unchanged and Q_3 output of FF3 changes from 0 to 1 (as D_3=1). So after fourth positive going edge of clock complete output is available at Q_3, Q_2, Q_1 and Q_0. The left most bit is available at Q_3 and the right most bit is at Q_0 (means Q_3=1, Q_2=1, Q_1=1 and Q_0=0).

5. The fifth positive going edge of clock leaves Q_0 output of FF0 unchanged (as no data input), Q_1 output of FF1 changes from 1 to 0 (as D_1=0) and leaves Q_2 output of FF2 as well as Q_3 output of FF3 unchanged (as D_2=D_3=1).

6. The sixth positive going edge of clock leaves Q_0 output of FF0 as well as Q_1 output of FF1 unchanged (as no data input), changes Q_2 output of FF2 from 1 to 0 (as D_2=0) and leaves Q_3 output of FF3 unchanged (as D_3=1).

7. The seventh positive going edge of clock leaves Q_0 output of FF0, Q_1 output of FF1 as well as Q_2 output of FF2 unchanged (as no data input) and changes Q_3 output of FF3 from 1 to 0 (as D_3=0).

8. The eighth positive going edge of clock leaves Q_0 output of FF0, Q_1 output of FF1, Q_2 output of FF2 as well as Q_3 output of FF3 unchanged (as no data input) so the register is clear again.

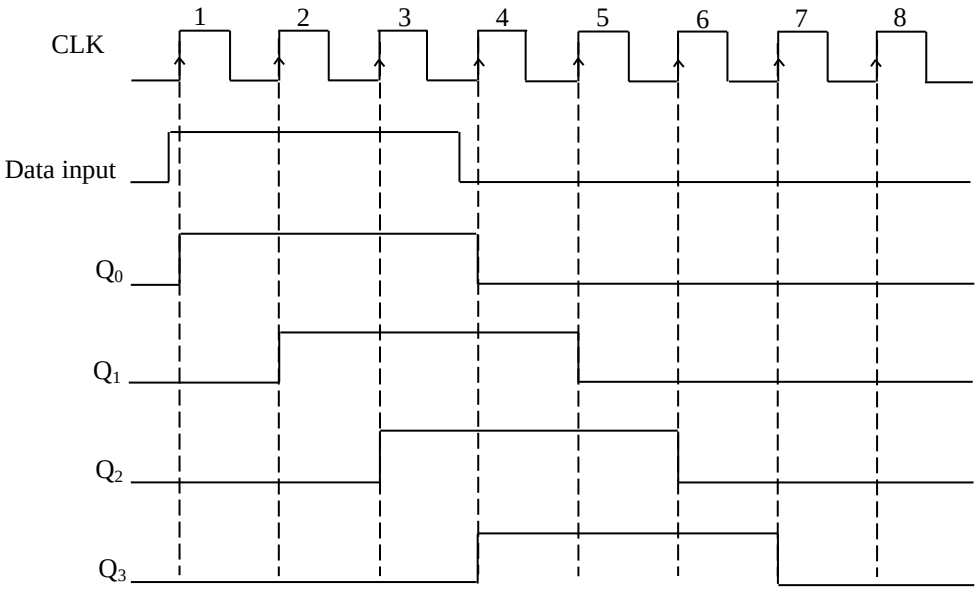


Figure 9. Timing diagram of 4-bit SIPO shift register using D flip-flops.

2.2.4 Logic symbol

Figure 10 shows the logic symbol for 4-bit SIPO shift register where D_0 is the serial data input and Q_0, Q_1, Q_2 and Q_3 are the parallel data output

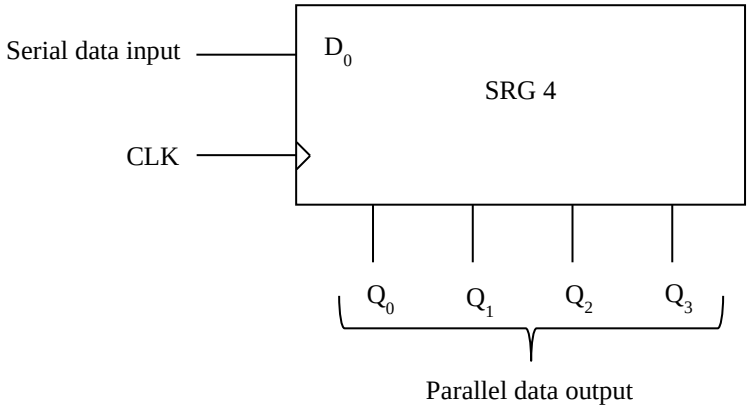


Figure 10. Logic symbol for 4-bit SIPO shift register.

Interesting fact

Direction of shifting

- If we consider Q_3 to be most significant bit (MSB) and Q_0 as least significant bit (LSB) then data bits can shift from right to left in the register with the clock.

Table showing data shifting from right to left.					
After clock edge	**Data input**	**Q_3(MSB)**	**Q_2**	**Q_1**	**Q_o(LSB)**
0	-	0	0	0	0
1	1	0	0	0	**1**
2	0	0	0	**1**	**0**
3	0	0	**1**	**0**	**0**
4	1	**1**	**0**	**0**	**1**
5	0	**0**	**0**	**1**	**0**
6	1	**0**	**1**	**0**	**1**
7	1	**1**	**0**	**1**	**1**
8	0	**0**	**1**	**1**	**0**

- Whereas data bits shift from left to right, if Q_0 is considered as MSB and Q_3 as LSB.

Table showing data shifting from left to right.					
After clock edge	**Data input**	**Q_o(MSB)**	**Q_1**	**Q_2**	**Q_3(LSB)**
0	-	0	0	0	0
1	1	**1**	0	0	0
2	0	**0**	**1**	0	0
3	0	**0**	**0**	**1**	0
4	1	**1**	**0**	**0**	**1**
5	0	**0**	**1**	**0**	**0**
6	1	**1**	**0**	**1**	**0**
7	1	**1**	**1**	**0**	**1**
8	0	**0**	**1**	**1**	**0**

Multiplication and division by shifting

Mathematically, shifting operation is equivalent to multiplication or division depending on the direction of shifting.

- Shifting of data to the left by n bits is same as multiplying by 2^n. Let us consider an example, as 3*2=6 and in binary form $3)_{10}=0011)_2$ so shifting $0011)_2$ to left once, gives $0110)_2$ which is decimal equivalent of 6.

- Shifting of data to the right by n bits is same as dividing by 2^n. Let us consider an example, as 8/4=2 and in binary form $8)_{10}=1000)_2$ so shifting $1000)_2$ to right twice, gives $0010)_2$ which is decimal equivalent of 2.

2.3 PISO shift register

Data input comes in parallel means bits are entered simultaneously into respective stages and stored output goes out serially, bit by bit on one line.

2.3.1 Construction

Let us consider a 4-bit PISO shift register designed using four positive edge triggered D flip-flops as shown in Figure 11. A 4-bit PISO shift register has four parallel input (D_3, D_2, D_1 and D_0) and one serial output (Q_3).

Level of $\text{Shift}/\overline{\text{Load}}$ signal allows four bits of data to load in parallel or shift in the register:

1. When $\text{Shift}/\overline{\text{Load}} = 0$, data input comes in and the positive going edge of clock sets or resets flip-flops according to the input means data present at D_3, D_2, D_1 and D_0 input is loaded in parallel into the register simultaneously.

2. When $\text{Shift}/\overline{\text{Load}} = 1$, shifting operation takes place and Q_0 output of first flip-flop goes to D_1 input of second flip-flop (FF1), Q_1 output of second flip-flop goes to D_2 input of third flip-flop (FF2) and Q_2 output of third flip-flop goes to D_3 input of fourth flip-flop (FF3).

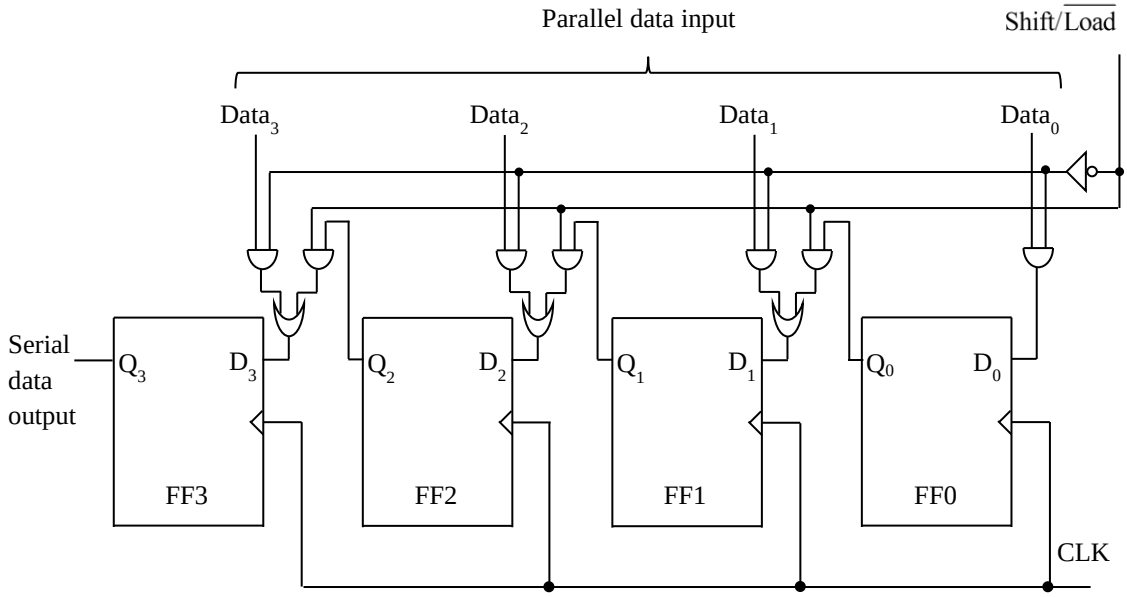


Figure 11. Logic diagram of 4-bit PISO shift register using D flip-flops.

2.3.2 Working

Assume that the register is initially clear (means $Q_3=Q_2=Q_1=Q_0=0$) and $\text{Shift}/\overline{\text{Load}} = 1$. The steps to store four bits (0101) into the register are:

1. Let us place each bit in parallel into the register. The left most bit which is a 0, is placed on $Data_3$ input line. The second bit which is a 1, is placed on $Data_2$ input line. The third bit which is a 0, is placed on $Data_1$ input line and the right most bit which is a 1, is placed on $Data_0$ input line.

2. As $Shift/\overline{Load}$ signal goes low, bits placed on $Data_3$, $Data_2$, $Data_1$ and $Data_0$ input line can reach D_3, D_2, D_1 and D_0 input of corresponding flip-flop making $D_3=0$, $D_2=1$, $D_1=0$ and $D_0=1$. That's how four bits are entered in parallel into the register.

3. The first positive going edge of clock resets fourth flip-flop (means FF3 stores D_3 input i.e., $Q_3=0$), sets third flip-flop (means FF2 stores D_2 input i.e., $Q_2=1$), resets second flip-flop (means FF1 stores D_1 input i.e., $Q_1=0$) and sets first flip-flop (means FF0 stores D_0 input i.e., $Q_0=1$). That's how four bits of data are stored in the register and left most bit is available at Q_3. To get remaining data output, bits are further shifted serially out and become available at Q_3.

4. The moment $Shift/\overline{Load}$ signal goes high, data output Q_0, Q_1, Q_2 and Q_3 remain same and data input D_0, D_1, D_2 and D_3 change according to the previous stage output as Q_0 output of first flip-flop can reach D_1 input of second flip-flop (means $D_1=Q_0$), Q_1 output of second flip-flop can reach D_2 input of third flip-flop (means $D_2=Q_1$) and Q_2 output of third flip-flop can reach D_3 input of fourth flip-flop (means $D_3=Q_2$).

5. The second positive going edge of clock sets fourth flip-flop (means FF3 stores the output of FF2 i.e., $Q_3=1$), resets third flip-flop (means FF2 stores the output of FF1 i.e., $Q_2=0$), sets second flip-flop (means FF1 stores the output of FF0 i.e., $Q_1=1$) and resets first flip-flop (as no data input so $Q_0=0$). So the data stored in FF2 is shifted to FF3, data stored in FF1 is shifted to FF2 and data stored in FF0 is shifted to FF1. That's how second bit is available at Q_3.

6. The third positive going edge of clock resets fourth flip-flop (means FF3 stores the output of FF2 i.e., $Q_3=0$), sets third flip-flop (means FF2 stores the output of FF1 i.e., $Q_2=1$), resets second flip-flop (as no data input so $Q_1=0$) and leaves first flip-flop unchanged (as no data input so $Q_0=0$). So the data stored in FF2 is shifted to FF3 and data stored in FF1 is shifted to FF2. That's how third bit is available at Q_3.

7. The fourth positive going edge of clock sets fourth flip-flop (means FF3 stores the output of FF2 i.e., $Q_3=1$), resets third flip-flop (as no data input so $Q_2=0$) and leaves first as well as second flip-flop unchanged (as no data input so $Q_0=Q_1=0$). So the data stored in FF2 is shifted to FF3 and the right most bit is available at Q_3.

8. The fifth positive going edge of clock resets fourth flip-flop (as no data input so $Q_3=0$) and leaves first, second as well as third flip-flop unchanged (as no data input so $Q_0=Q_1=Q_2=0$) so the register is clear again.

2.3.3 Timing diagram

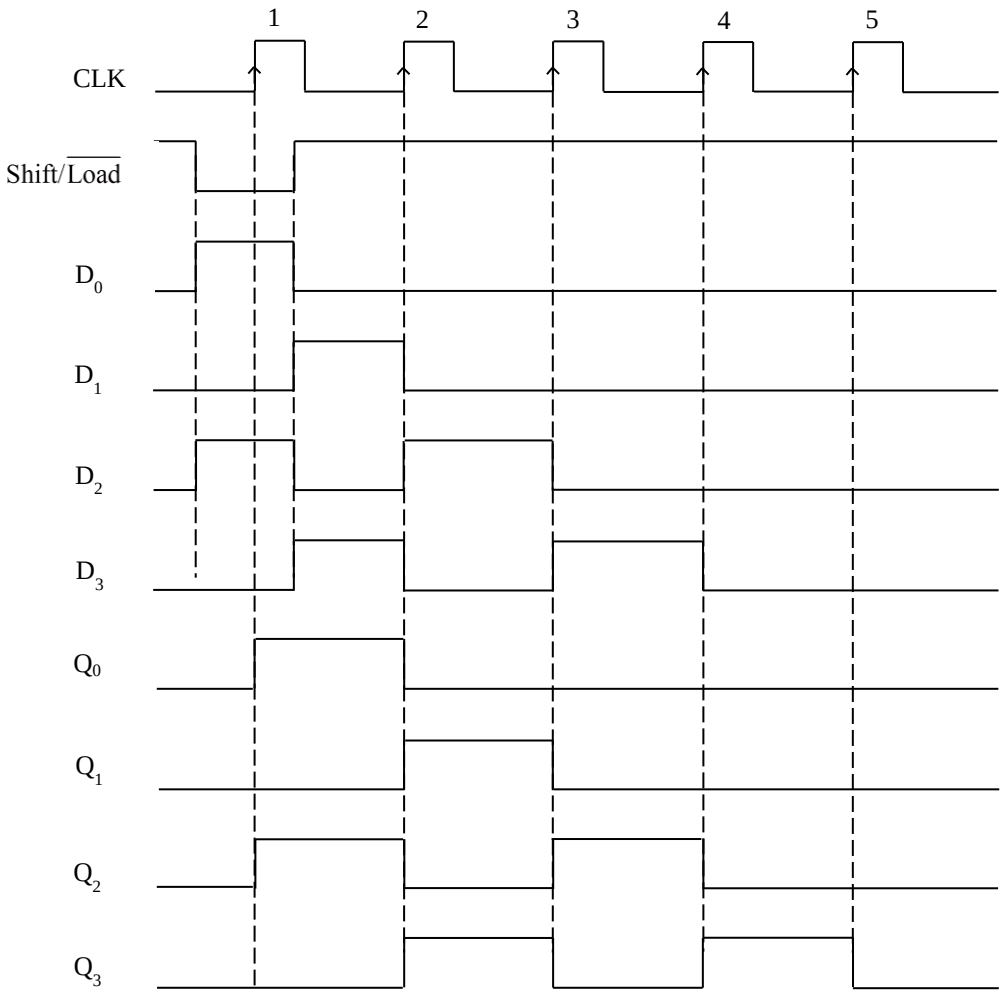


Figure 12. Timing diagram of 4-bit PISO shift register using D flip-flops.

The timing diagram of 4-bit PISO shift register is shown in Figure 12 and its stepwise description is as follows:

1. As soon as Shift/$\overline{\text{Load}}$ becomes 0, data bits enter into the register making D_3=0, D_2=1, D_1=0 and D_0=1.

2. The first positive going edge of clock changes Q_0 output of FF0 from 0 to 1 (as data input D_0=1), Q_1 output of FF1 remains unchanged (as data input D_1=0), Q_2 output of FF2 changes from 0 to 1 (as data input D_2=1) and Q_3 output of FF3 remains unchanged (as data input D_3=0).

3. As soon as Shift/$\overline{\text{Load}}$ becomes 1, data input change according to the previous stage output making D_3=Q_2=1, D_2=Q_1=0, D_1=Q_0=1 and D_0=0.

4. The second positive going edge of clock changes Q_0 output of FF0 from 1 to 0 (as no data input), Q_1 output of FF1 changes from 0 to 1 (as D_1=1), Q_2 output of FF2 changes from 1 to 0 (as D_2=0) and Q_3 output of FF3 changes from 0 to 1 (as D_3=1).

5. The third positive going edge of clock leaves Q_0 output of FF0 unchanged (as no data input), Q_1 output of FF1 changes from 1 to 0 (as D_1=0), Q_2 output of FF2 changes from 0 to 1 (as D_2=1) and Q_3 output of FF3 changes from 1 to 0 (as D_3=0).

6. The fourth positive going edge of clock leaves Q_0 output of FF0 as well as Q_1 output of FF1 unchanged (as no data input), Q_2 output of FF2 changes from 1 to 0 (as no data input) and Q_3 output of FF3 changes from 0 to 1 (as D_3=1).

7. The fifth positive going edge of clock leaves Q_0 output of FF0, Q_1 output of FF1 as well as Q_2 output of FF2 unchanged (as no data input) and Q_3 output of FF3 changes from 1 to 0 (as no data input) so the register is clear again.

2.3.4 Logic symbol

Figure 13 shows the logic symbol for 4-bit PISO shift register where D_0, D_1, D_2 and D_3 are the parallel data input and Q_3 is the serial data output.

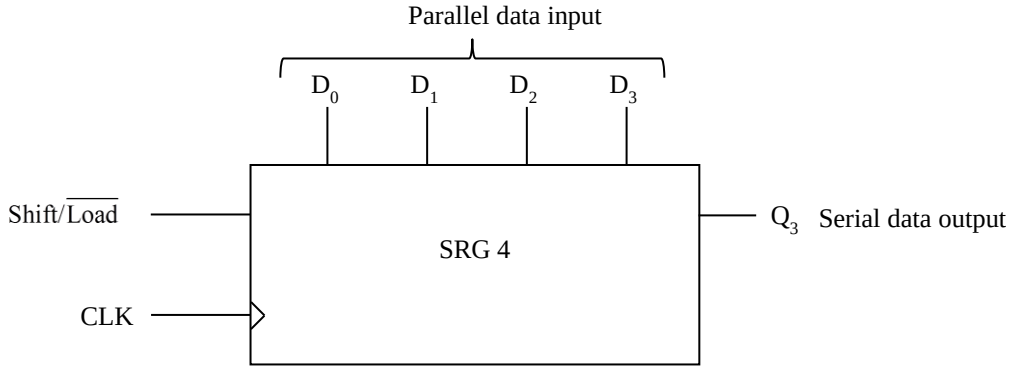


Figure 13. Logic symbol for 4-bit PISO shift register.

2.4 PIPO shift register

Data input comes in parallel means bits are entered simultaneously into respective stages and stored output goes out in parallel means output of each stage is available.

2.4.1 Construction

Let us consider a 4-bit PIPO shift register designed using four positive edge triggered D flip-flops as shown in Figure 14. A 4-bit PIPO shift register has four parallel input (D_3, D_2, D_1 and D_0) and four parallel output (Q_3, Q_2, Q_1 and Q_0).

Level of $\text{Shift}/\overline{\text{Load}}$ signal allows four bits of data to load in parallel or shift in the register:

1. When $\text{Shift}/\overline{\text{Load}} = 0$, data input comes in and the positive going edge of clock sets or resets flip-flops according to the input means data present at D_3, D_2, D_1 and D_0 input is loaded in parallel into the register simultaneously.

2. When $\text{Shift}/\overline{\text{Load}} = 1$, shifting operation takes place and Q_0 output of first flip-flop goes to D_1 input of second flip-flop (FF1), Q_1 output of second flip-flop goes to D_2 input of third flip-flop (FF2) and Q_2 output of third flip-flop goes to D_3 input of fourth flip-flop (FF3).

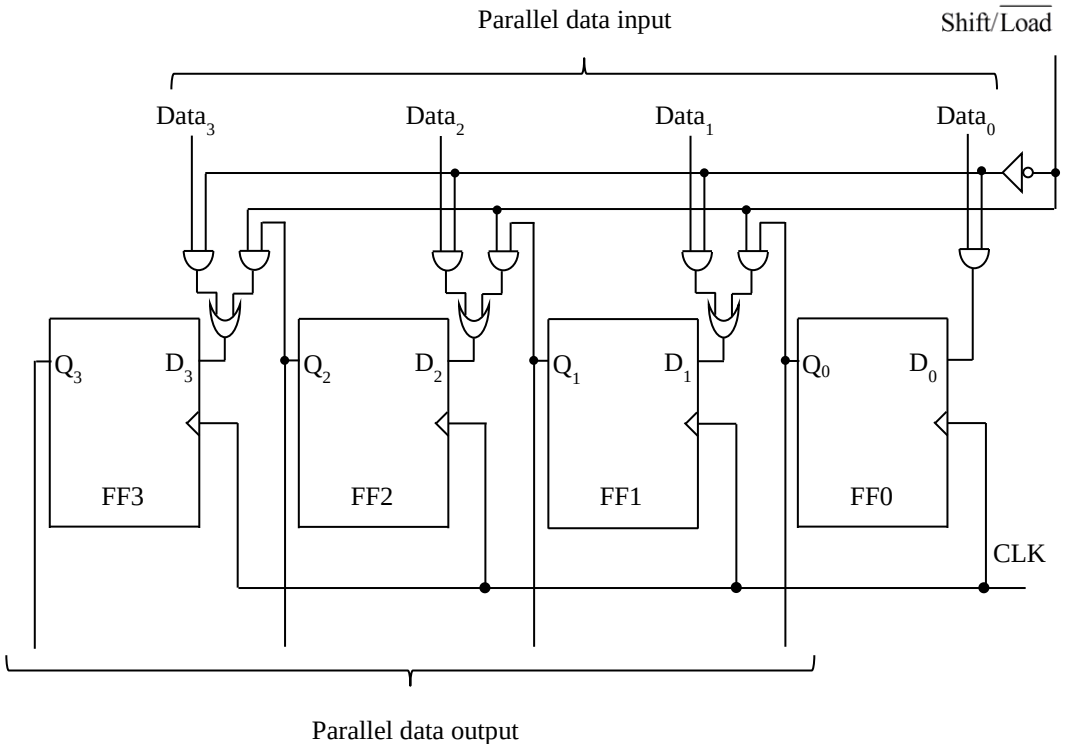


Figure 14. Logic diagram of 4-bit PIPO shift register using D flip-flops.

Figure 15 shows the simplified 4-bit PIPO register using D flip-flops where D_0 is the data input and Q_0 is the output of first flip-flop (FF0). For second flip-flop (FF1), D_1 is the data input and Q_1 is the output. In case of third flip-flop (FF2), D_2 is the data input and Q_2 is the output and D_3 is the data input and Q_3 is the output of fourth flip-flop (FF3).

2.4.2 Working

Assume that the register is initially clear (means Q_3=Q_2=Q_1=Q_0=0) and $\text{Shift}/\overline{\text{Load}} = 1$. The steps to store four bits (1101) into the register are:

1. Let us place each bit in parallel into the register. The left most bit which is a 1, is placed on $Data_3$ input line. The second bit which is a 1, is placed on $Data_2$ input line. The third bit which is a 0, is placed on $Data_1$ input line and the right most bit which is a 1, is placed on $Data_0$ input line.

2. As $\text{Shift}/\overline{\text{Load}}$ signal goes low, bits placed on $Data_3$, $Data_2$, $Data_1$ and $Data_0$ input line can reach D_3, D_2, D_1 and D_0 input of corresponding flip-flop making $D_3=1$, $D_2=1$, $D_1=0$ and $D_0=1$. That's how four bits are entered in parallel into the register.

3. The first positive going edge of clock sets fourth flip-flop (means FF3 stores D_3 input i.e., $Q_3=1$), sets third flip-flop (means FF2 stores D_2 input i.e., $Q_2=1$), resets second flip-flop (means FF1 stores D_1 input i.e., $Q_1=0$) and sets first flip-flop (means FF0 stores D_0 input i.e., $Q_0=1$). That's how four bits of data are stored in the register and are available at Q_3, Q_2, Q_1 and Q_0.

4. The moment $\text{Shift}/\overline{\text{Load}}$ signal goes high, data output Q_0, Q_1, Q_2 and Q_3 remain same and data input D_0, D_1, D_2 and D_3 change according to the previous stage output as Q_0 output of first flip-flop can reach D_1 input of second flip-flop (means $D_1=Q_0$), Q_1 output of second flip-flop can reach D_2 input of third flip-flop (means $D_2=Q_1$) and Q_2 output of third flip-flop can reach D_3 input of fourth flip-flop (means $D_3=Q_2$).

5. The second positive going edge of clock leaves fourth flip-flop unchanged (means FF3 stores the output of FF2 i.e., $Q_3=1$), resets third flip-flop (means FF2 stores the output of FF1 i.e., $Q_2=0$), sets second flip-flop (means FF1 stores the output of FF0 i.e., $Q_1=1$) and resets first flip-flop (as no data input so $Q_0=0$). So the data stored in FF2 is shifted to FF3, data stored in FF1 is shifted to FF2 and data stored in FF0 is shifted to FF1.

6. The third positive going edge of clock resets fourth flip-flop (means FF3 stores the output of FF2 i.e., $Q_3=0$), sets third flip-flop (means FF2 stores the output of FF1 i.e., $Q_2=1$), resets second flip-flop (as no data input so $Q_1=0$) and leaves first flip-flop unchanged (as no data input so $Q_0=0$). So the data stored in FF2 is shifted to FF3 and data stored in FF1 is shifted to FF2.

7. The fourth positive going edge of clock sets fourth flip-flop (means FF3 stores the output of FF2 i.e., $Q_3=1$), resets third flip-flop (as no data input so $Q_2=0$) and leaves first as well as second flip-flop unchanged (as no data input so $Q_0=Q_1=0$). So the data stored in FF2 is shifted to FF3.

8. The fifth positive going edge of clock resets fourth flip-flop (as no data input so $Q_3=0$) and leaves first, second as well as third flip-flop unchanged (as no data input so $Q_0=Q_1=Q_2=0$) so the register is clear again.

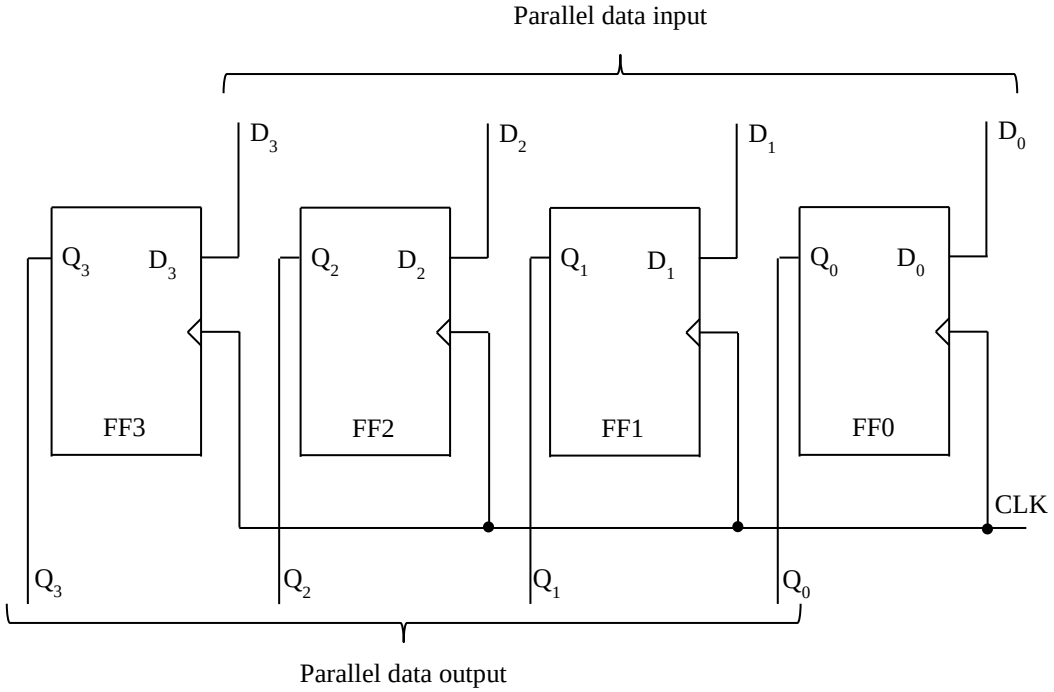


Figure 15. Simplified 4-bit PIPO register using D flip-flops.

2.4.3 Timing diagram

The timing diagram of 4-bit PIPO shift register is shown in Figure 16 and its stepwise description is as follows:

1. As soon as Shift/$\overline{\text{Load}}$ becomes 0, data bits enter into the register making D_3=1, D_2=1, D_1=0 and D_0=1.

2. The first positive going edge of clock changes Q_0 output of FF0 from 0 to 1 (as D_0=1), Q_1 output of FF1 remains unchanged (as D_1=0), Q_2 output of FF2 changes from 0 to 1 (as D_2=1) and Q_3 output of FF3 changes from 0 to 1 (as D_3=1).

3. As soon as Shift/$\overline{\text{Load}}$ becomes 1, data input change according to the previous stage output making D_3=Q_2=1, D_2=Q_1=0, D_1=Q_0=1 and D_0=0.

4. The second positive going edge of clock changes Q_0 output of FF0 from 1 to 0 (as no data input), Q_1 output of FF1 changes from 0 to 1 (as D_1=1), Q_2 output of FF2 changes from 1 to 0 (as D_2=0) and Q_3 output of FF3 remains unchanged (as D_3=1).

5. The third positive going edge of clock leaves Q_0 output of FF0 unchanged (as no data input), Q_1 output of FF1 changes from 1 to 0 (as D_1=0), Q_2 output of FF2 changes from 0 to 1 (as D_2=1) and Q_3 output of FF3 changes from 1 to 0 (as D_3=0).

6. The fourth positive going edge of clock leaves Q_0 output of FF0 as well as Q_1 output of FF1 unchanged (as no data input), Q_2 output of FF2 changes from 1 to 0 (as no data input) and Q_3 output of FF3 changes from 0 to 1 (as D_3=1).

7. The fifth positive going edge of clock leaves Q_0 output of FF0, Q_1 output of FF1 as well as Q_2 output of FF2 unchanged (as no data input) and Q_3 output of FF3 changes from 1 to 0 (as no data input) so the register is clear again.

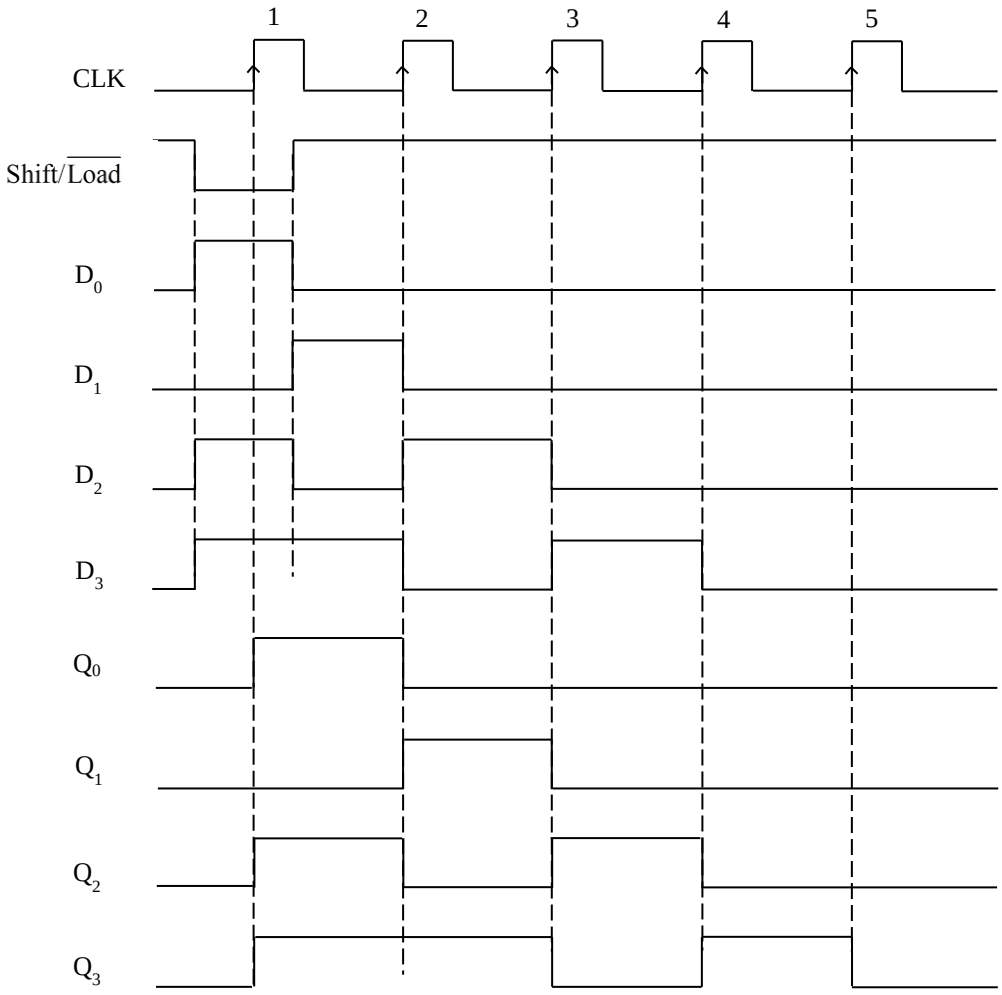


Figure 16. Timing diagram of 4-bit PIPO shift register using D flip-flops.

2.4.4 Logic symbol

Figure 17 shows the logic symbol for 4-bit PIPO shift register where D_0, D_1, D_2 and D_3 are the parallel data input and Q_0, Q_1, Q_2 and Q_3 are the parallel data output.

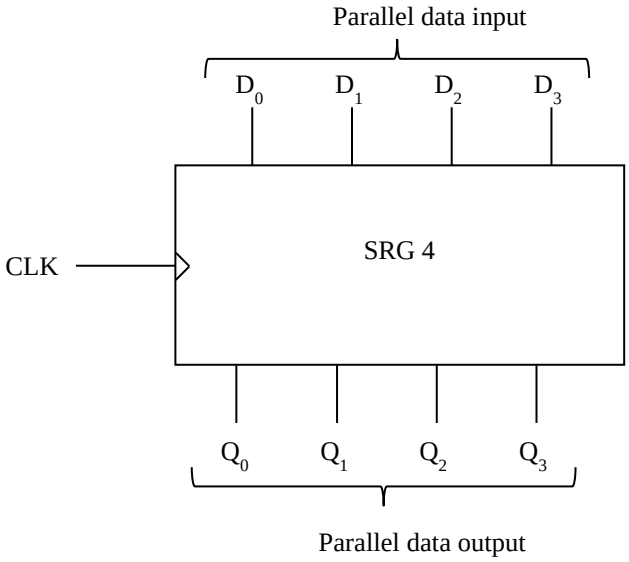


Figure 17. Logic symbol of 4-bit PIPO shift register.

Summary

- Registers consist of an arrangement of flip-flops and are used to store binary data or to transfer digital data.
- Shift registers shift contents of registers once in every clock cycle.
- Storage capacity of a register is determined by the number of flip-flops used as a flip-flop has one bit memory.
- In SISO shift register, data input comes in serially means one bit at a time and stored output goes out serially, bit by bit on one line.
- In SIPO shift register, data input comes in serially means one bit at a time and stored output goes out in parallel means output of each stage is available.
- In PISO shift register, data input comes in parallel means bits are entered simultaneously into respective stages and stored output goes out serially, bit by bit on one line.
- In PIPO shift register, data input comes in parallel means bits are entered simultaneously into respective stages and stored output goes out in parallel means output of each stage is available.
- A 4-bit SISO shift register has one serial input (D_0) and one serial output (Q_3).
- A 4-bit SIPO shift register has one serial input (D_0) and four parallel output (Q_3, Q_2, Q_1 and Q_0).
- A 4-bit PISO shift register has four parallel input (D_3, D_2, D_1 and D_0) and one serial output (Q_3).
- A 4-bit PIPO shift register has four parallel input (D_3, D_2, D_1 and D_0) and four parallel output (Q_3, Q_2, Q_1 and Q_0).
- Shifting operation is equivalent to multiplication or division depending on the direction of shifting.
- A mod-n ring counter uses n flip-flops.
- A ring counter is modified form of a SISO shift register in which data shifts from right to left and back around.
- No decoding gates are required in case of a ring counter as its output is unique for each state.

Exercise

Solved problems

Problem 1. Draw the timing diagram of 4-bit SISO shift register (Figure 2) for 1101 data input. Assume that the register is initially clear.

Solution: The left most bit which is a 1 goes into the register on first positive going edge of clock. As remaining bits are entered with the clock, they are shifted from right to left through the shift register. The shift register contains $Q_3Q_2Q_1Q_0$=1101 after four positive going edge of clock. Complete output is available at Q_3 in seven clock cycles.

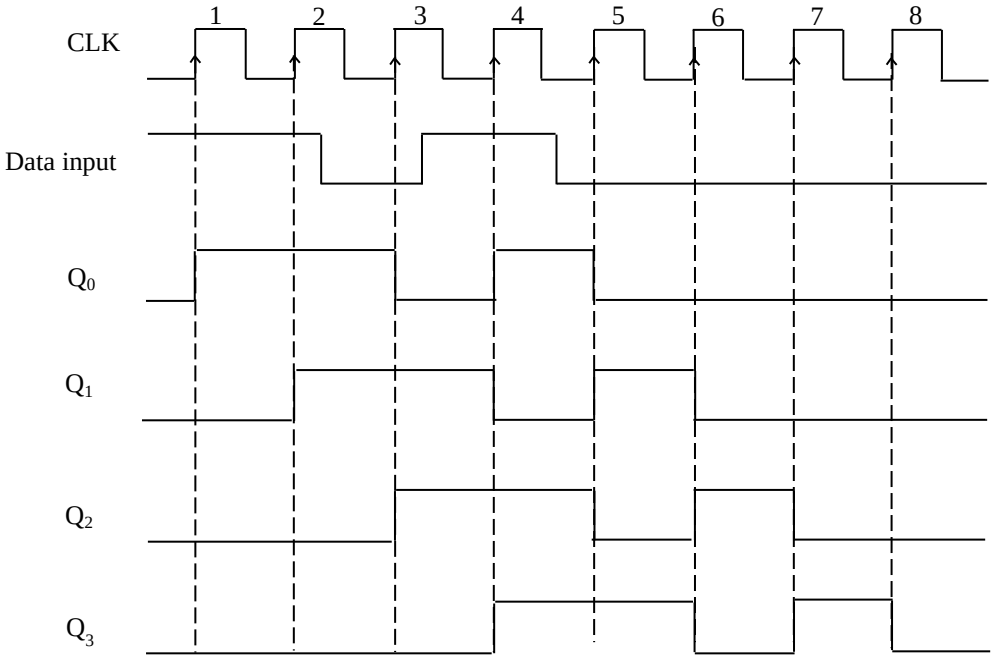


Figure 18. Timing diagram of 4-bit SISO shift register using D flip-flops.

Problem 2. Draw the logic diagram of 4-bit SISO shift register using JK flip-flops instead of D flip-flops.

Solution: At the positive going edge of clock, if J=1 and K=0 is applied then flip-flop stores a 1 and if J=0 and K=1 is applied then flip-flop stores a 0 so a 4-bit SISO shift register using JK flip-flops can be realized as shown in Figure 19.

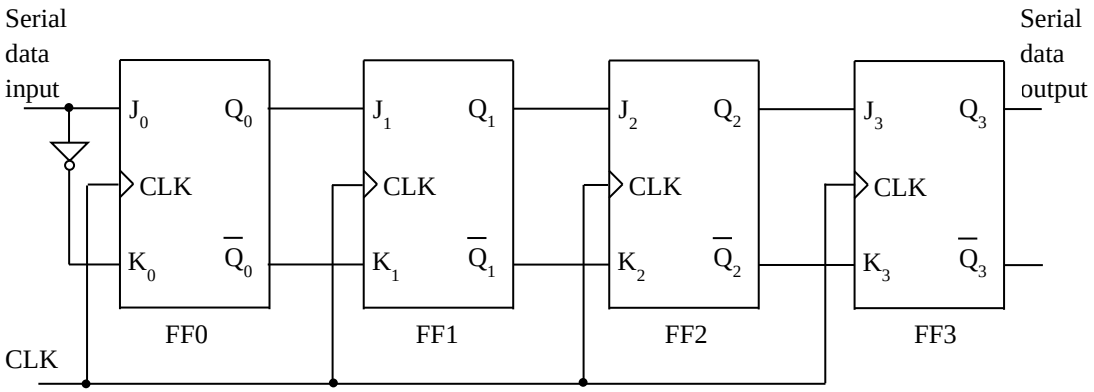


Figure 19. Logic diagram of 4-bit SISO shift register using JK flip-flops.

Problem 3. Draw the timing diagram of 4-bit SIPO shift register (Figure 8) for 0111 data input. Assume that the register initially contains all 1's.

Solution: The left most bit which is a 0 goes into the register on first positive going edge of clock. As remaining bits are entered with the clock, they are shifted from right to left through the shift register. The register contains $Q_3Q_2Q_1Q_0$=0111 after four positive going edge of clock. Complete output is available at Q_3, Q_2, Q_1 and Q_0 after four clock cycles.

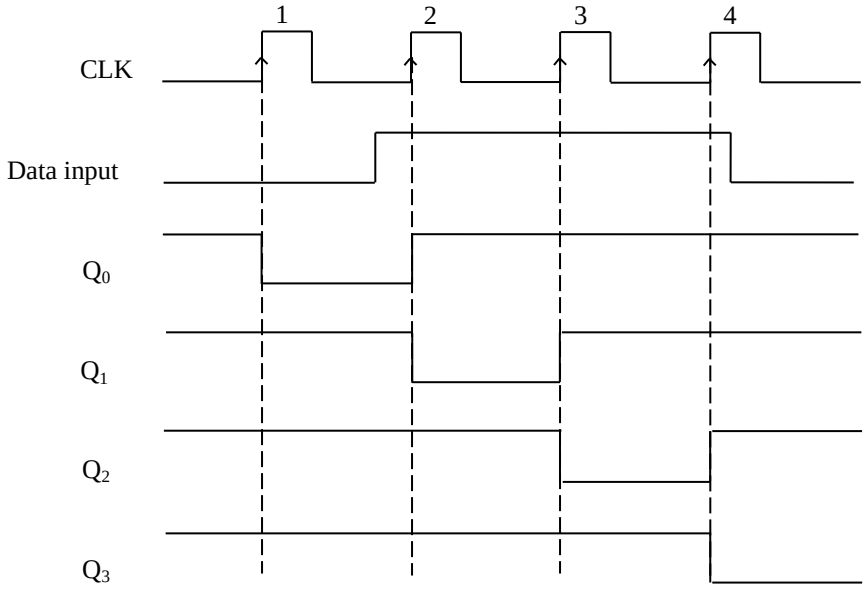


Figure 20. Timing diagram of 4-bit SIPO shift register using D flip-flops.

Problem 4. The data input 1101 is entered serially into a 4-bit SIPO shift register which is initially clear. What is the output after four clock cycles? If the data input remains 0 after fourth positive going edge of clock, what is the state of the register after (a) 2 additional clock edges and (b) 4 additional clock edges?

Solution:

Table 2. Sequence of states for 4-bit SIPO shift register.				
After clock edge	**Q_3 (MSB)**	**Q_2**	**Q_1**	**Q_o (LSB)**
0	0	0	0	0
1	0	0	0	**1**
2	0	0	**1**	**1**
3	0	**1**	**1**	**0**
4	**1**	**1**	**0**	**1**
5	**1**	**0**	**1**	0
6	**0**	**1**	0	0
7	**1**	0	0	0
8	0	0	0	0

Problem 5. How to convert 4-bit PIPO shift register as SISO shift register?

Solution: To convert 4-bit PIPO shift register as SISO shift register, the output of each flip flop is connected to the input of next flip-flop. Q_0 output of first flip-flop goes to D_1 input of second flip-flop, Q_1 output of second flip-flop goes to D_2 input of third flip-flop, Q_2 output of third flip-flop goes to D_3 input of fourth flip-flop. Serial data input is applied at D_0 and serial output is received at Q_3.

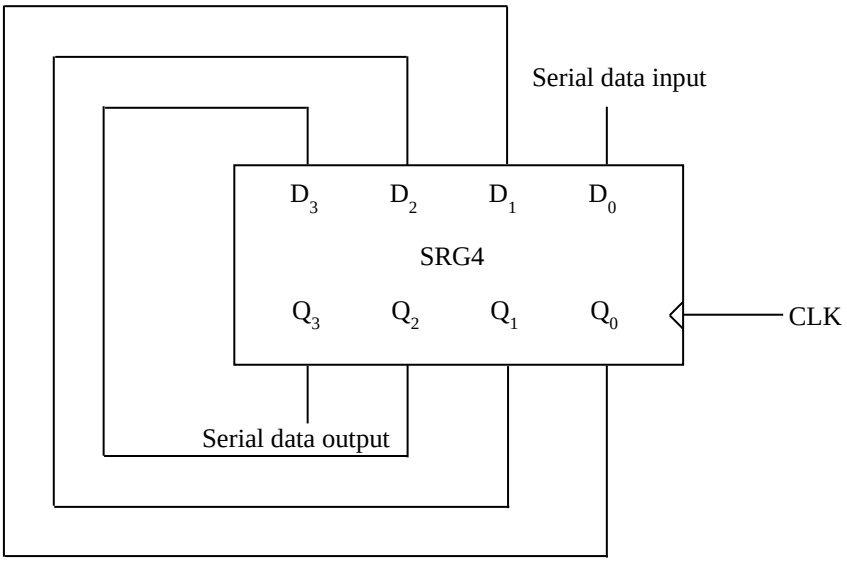


Figure 21. Connection diagram for SISO shift register using PIPO shift register.

Problem 6. Draw the timing diagram of 4-bit PIPO shift register (Figure 14) for 0101 data input. Assume that the register is initially clear.

Solution: Level of $\text{Shift}/\overline{\text{Load}}$ signal allows four bits of data to load in parallel or shift in the register. When $\text{Shift}/\overline{\text{Load}} = 0$, data input comes in and the positive going edge of clock sets or resets flip-flop according to the input means data present at D_3, D_2, D_1 and D_0 input is loaded in parallel into the register simultaneously. When $\text{Shift}/\overline{\text{Load}} = 1$, shifting operation takes place and Q_0 output of first flip-flop goes to D_1 input of second flip-flop (FF1), Q_1 output of second flip-flop goes to D_2 input of third flip-flop (FF2) and Q_2 output of third flip-flop goes to D_3 input of fourth flip-flop (FF3).

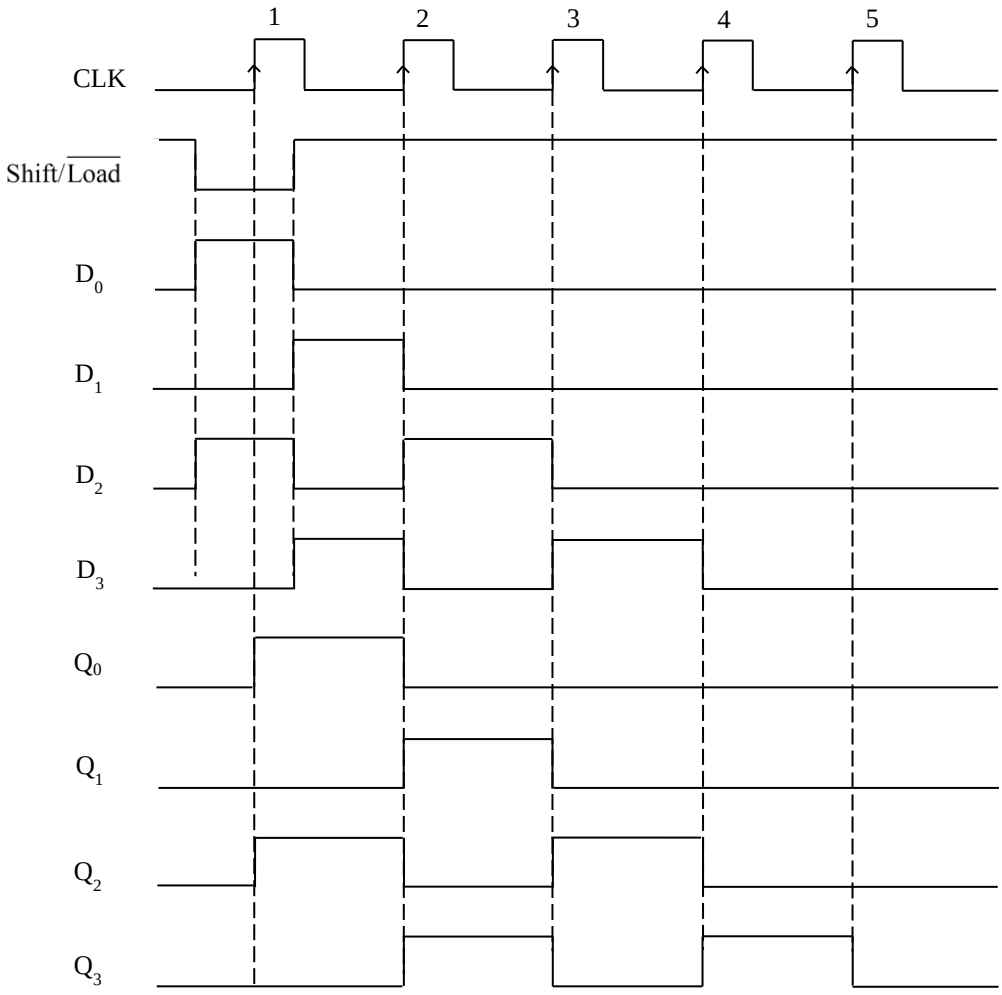


Figure 22. Timing diagram of 4-bit PIPO shift register using D flip-flops.

Objective problems

1. Storage capacity of a register is determined by

(A) Method of data input

(B) Method of data output

(C) Type of flip-flops used

(D) Number of flip-flops used

Answer: (D)

2. In SISO shift register

(A) Data input comes in serially and stored output goes out serially

(B) Data input comes in serially and stored output goes out in parallel

(C) Data input comes in parallel and stored output goes out serially

(D) Data input comes in parallel and stored output goes out in parallel

Answer: (A)

3. In PISO shift register

(A) Data input comes in serially and stored output goes out serially

(B) Data input comes in serially and stored output goes out in parallel

(C) Data input comes in parallel and stored output goes out serially

(D) Data input comes in parallel and stored output goes out in parallel

Answer: (C)

4. Data input comes in parallel means

(A) Data output of each stage is available

(B) Bits are entered bit by bit on one line

(C) Bits are entered simultaneously into respective stages

(D) None of the above

Answer: (C)

5. A 4-bit PISO shift register has

(A) One serial input (D_0) and one serial output (Q_3)

(B) One serial input (D_0) and four parallel output (Q_3, Q_2, Q_1 and Q_0)

(C) Four parallel input (D_3, D_2, D_1 and D_0) and one serial output (Q_3)

(D) Four parallel input (D_3, D_2, D_1 and D_0) and four parallel output (Q_3, Q_2, Q_1 and Q_0)

Answer: (C)

6. What is the modulus of a ring counter having 4 flip-flops

(A) 4

(B) 8

(C) 16

(D) None of the above

Answer: (A)

Short problems

1. Which shift register can have a complete binary number loaded into it at a time and have it shifted out one bit at a time? (PISO shift register)

2. Which type of shift register can have only one bit of data entered into it at a time but has all data bits available as output simultaneously? (SIPO shift register)

3. How many bit position shifting of binary data is required so that it multiplies by 4? (as shifting from right to left by one bit position is same as multiplication by 2 so to multiply by 4 we need to shift binary data to the left by two bit positions)

4. How many clock cycles are required to enter or store 2 bytes of data serially into a 16-bit shift register? (one clock cycle is required to store a bit and as a byte of data has 8 bits so to store 2 bytes =16 bits of data, we require 16 clock cycles)

5. In which type of shift register do we have access only to one flip-flop output? (serial output shift register)

6. What is the number of states for a ring counter having five flip-flops? (5 states)

True or false problems

1. Registers are used mainly to store binary data. (True)
2. A flip-flop has 4-bit memory. (False)
3. Shift registers shift content of registers once in every clock cycle. (True)
4. A 4-bit SISO shift register has 4-bit memory. (True)
5. A SIPO shift register can display all input bits at a time. (True)
6. A PISO shift register can display all input bits at a time. (False)
7. The output of a ring counter is always square wave. (False)
8. The output of a ring counter is unique for each state. (True)
9. The output of a johnson counter is always square wave. (True)
10. All flip-flops in a shift register have common clock input. (True)

Glossary

Shift register: shifts binary data stored in register once in every clock cycle

Serial input: one bit of input is entered at a time

Serial output: output available bit by bit on one line

Parallel input: input bits are entered simultaneously into respective stages

Parallel output: output of each stage is available simultaneously

SISO: serial input/serial output shift register

SIPO: serial input/parallel output shift register

PISO: parallel input/serial output shift register

PIPO: parallel input/parallel output shift register

Ring counter: shifts data from right to left and back around from Q_3 to Q_0

References

1. Digital fundamentals by Thomas L. Floyd and R.P. Jain, Eighth Edition, Pearson Education Inc., ISBN (978-81-7758-763-0)

2. Digital systems principles and applications by Ronald J. Tocci and Neal S. Widmer, Eigthh Edition,Prentice Hall Publication, ISBN (0-13-085634-7)

3. Fundamentals of digital circuits by A. Anand Kumar, Second Edition, PHI Learning Private Limited, ISBN (978-81-203-3679-7)