Shift registers

1.0 Introduction

Shift registers are a type of sequential logic circuit, mainly for storage of digital data. They are a group of flip-flops connected in a chain so that the output from one flip-flop becomes the input of the next flip-flop. Most of the registers possess no characteristic internal sequence of states. All flip-flop is driven by a common clock, and all are set or reset simultaneously.

In these few lectures, the basic types of shift registers are studied, such as Serial In - Serial Out, Serial In - Parallel Out, Parallel In – Serial Out, Parallel In - Parallel Out, and bidirectional shift registers. A special form of counter - the shift register counter, is also introduced.

Register:

?? A set of n flip-flops
?? Each flip-flop stores one bit
?? Two basic functions: data storage (Figure 1.2) and data movement (Figure 1.1).

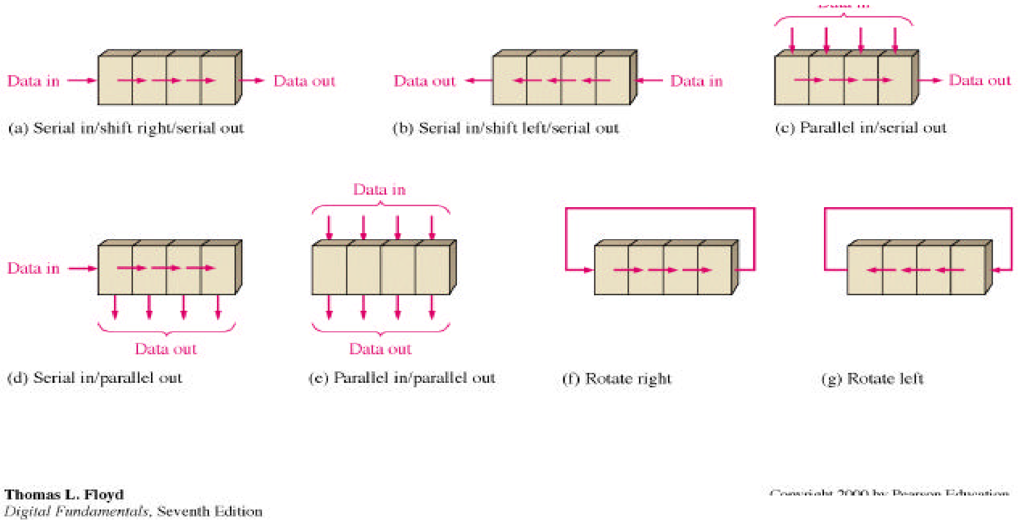
Shift Register:

?? A register that allows each of the flip-flops to pass the stored information to its adjacent neighbour
?? Figure 1.1 shows the basic data movement in shift registers.

Counter:

?? A register that goes through a predetermined sequence of states

Figure 1.1: Basic data movement in shift registers [Floyd]



(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

Thomas L. Floyd
Digital Fundamentals, Seventh Edition

Copyright 2000 by Pearson Education

Storage Capacity:

The storage capacity of a register is the total number of bits (1 or 0) of digital data it can retain. Each stage (flip flop) in a shift register represents one bit of storage capacity. Therefore the number of stages in a register determines its storage capacity.

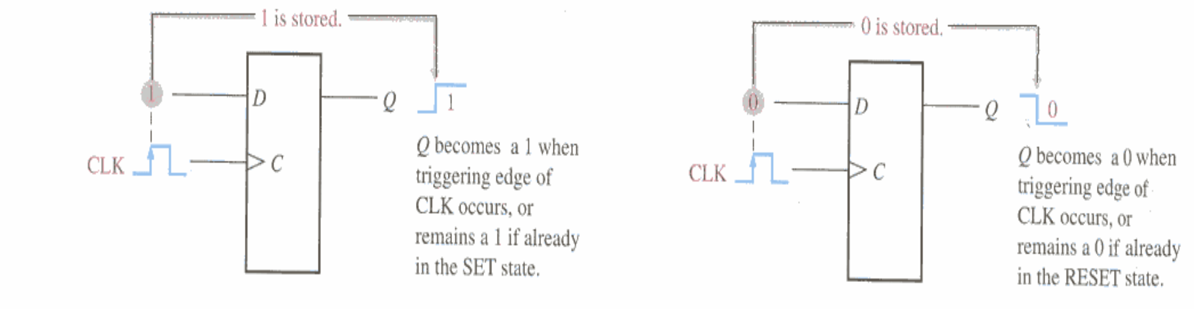

1 is stored.
1
D
Q
1
CLK
C
Q becomes a 1 when triggering edge of CLK occurs, or remains a 1 if already in the SET state.
0 is stored.
0
D
Q
0
CLK
C
Q becomes a 0 when triggering edge of CLK occurs, or remains a 0 if already in the RESET state.

Figure 1.2: The flip-flop as a storage element.

2.0 Serial In - Serial Out Shift Registers

The serial in/serial out shift register accepts data serially – that is, one bit at a time on a single line. It produces the stored information on its output also in serial form.

2.1 Example: Basic four-bit shift register

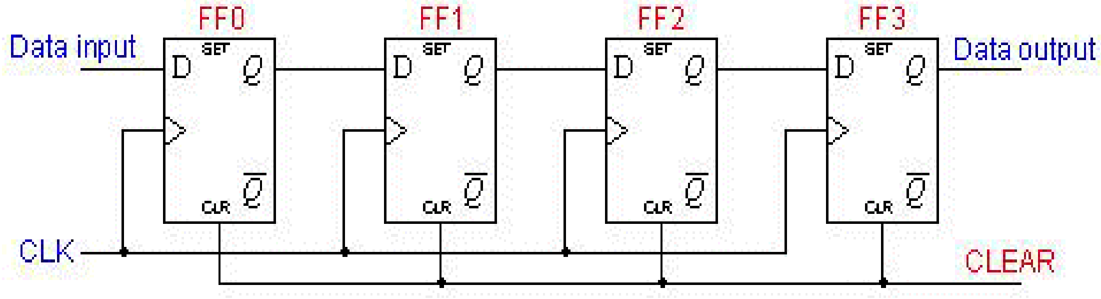


Figure 2.1

A basic four-bit shift register can be constructed using four D flip-flops, as shown in Figure 2.1.

The operation of the circuit is as follows.

?? The register is first cleared, forcing all four outputs to zero.

?? The input data is then applied sequentially to the D input of the first flip-flop on the left (FF0).

?? During each clock pulse, one bit is transmitted from left to right.

?? Assume a data word to be 1001.

?? The least significant bit of the data has to be shifted through the register from FF0 to FF3.

In order to get the data out of the register, they must be shifted out serially. This can be done destructively or non-destructively. For destructive readout, the original data is lost and at the end of the read cycle, all flip-flops are reset to zero.

FF0	FF1	FF2	FF3	
0	0	0	0	1001

The data is loaded to the register when the control line is HIGH (ie WRITE). The data can be shifted out of the register when the control line is LOW (ie READ).

Clear	FF0	FF1	FF2	FF3
1001	0	0	0	0

WRITE:

FF0	FF1	FF2	FF3	
1	0	0	1	0000

READ:

FF0	FF1	FF2	FF3	
1	0	0	1	1001

Figure 2.2 illustrates entry of the four bits 1010 into the register. Figure 2.3 shows the four bits (1010) being serially shifted out of the register and replaced by all zeros.

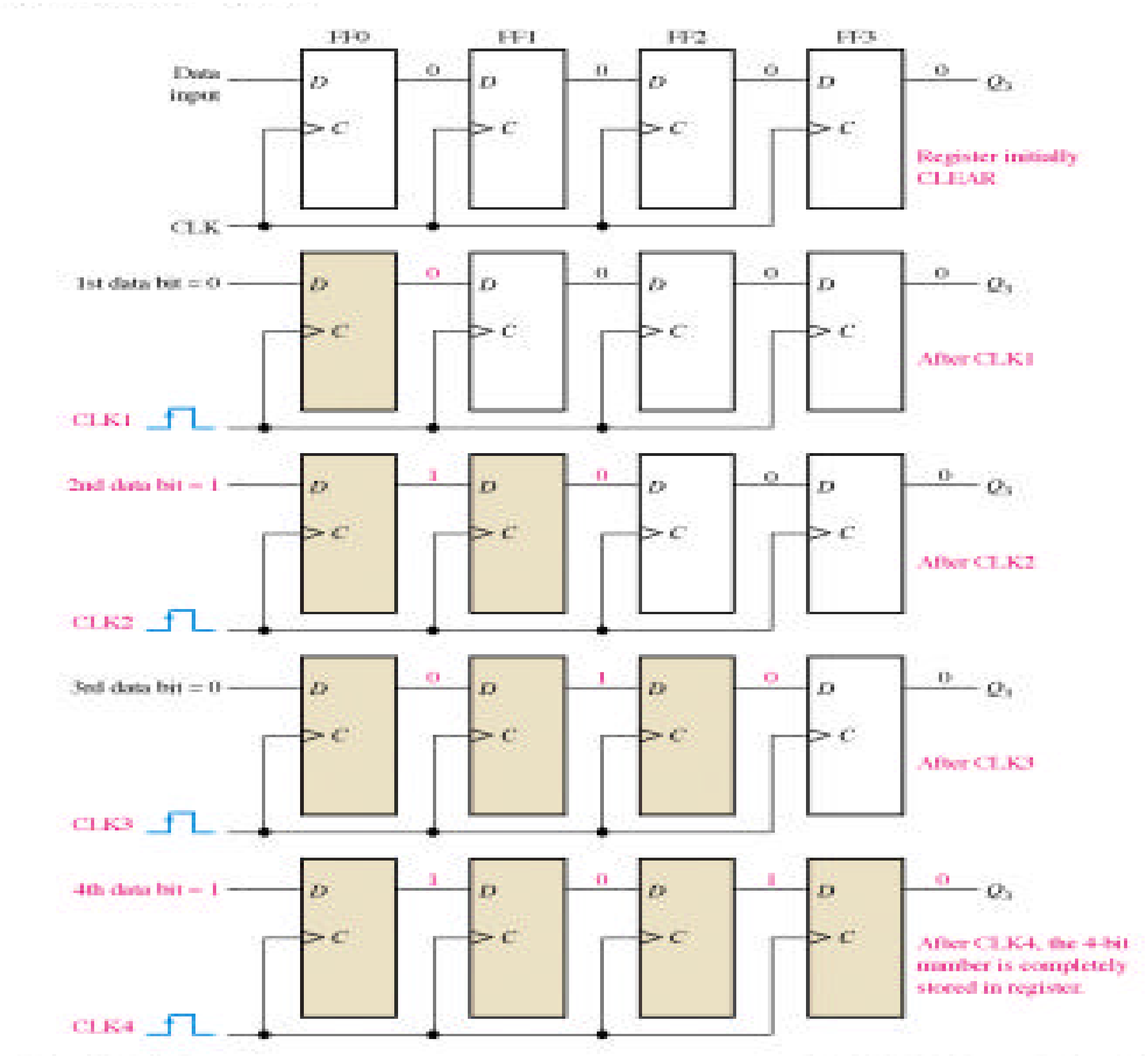


Figure 2.2: Four bits (1010) being entered serially into the register.

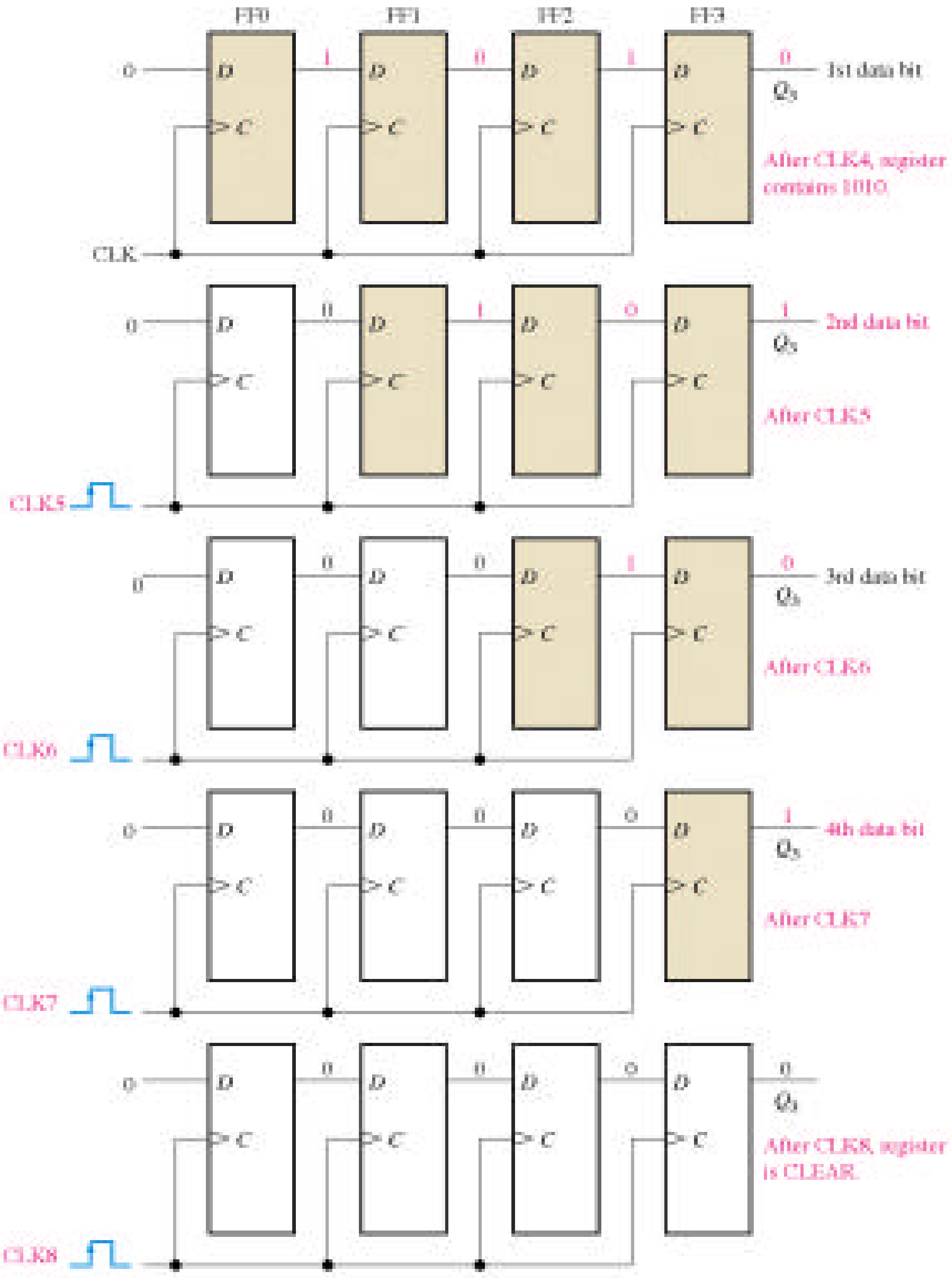


Figure 2.3: Four bits (1010) being serially shifted out of the register and replaced by all zeros

2.2 5-bit serial in/serial out shift registers

Figure 2.4 illustrates entry of the five bits 11010 into the register.

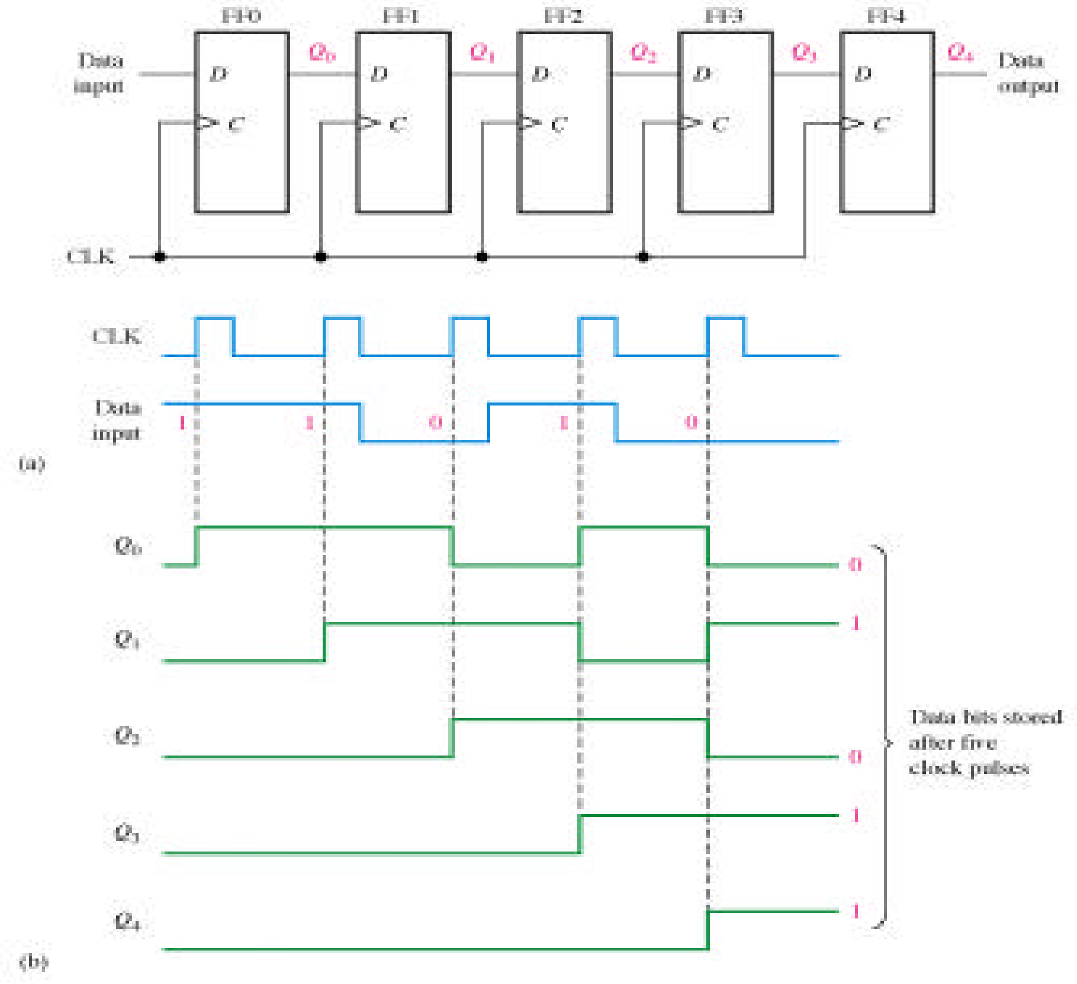
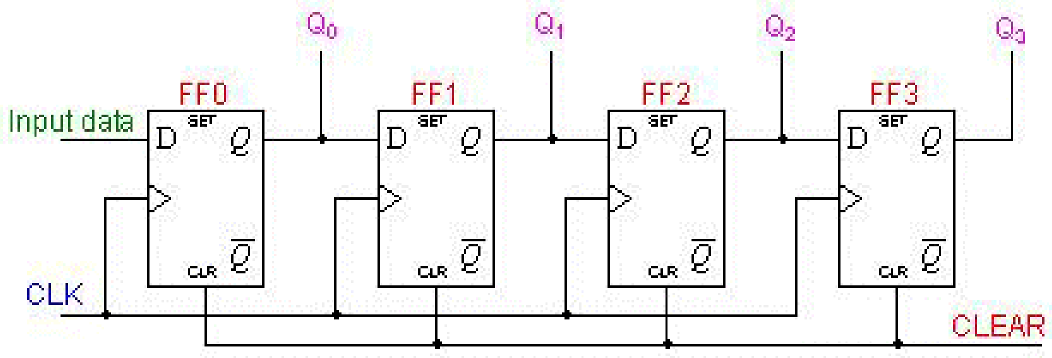


Figure 2.4

3.0 Serial In - Parallel Out Shift Registers

For this kind of register, data bits are entered serially in the same manner as discussed in the last section. The difference is the way in which the data bits are taken out of the register. Once the data are stored, each bit appears on its respective output line, and all bits are available simultaneously. A construction of a four-bit serial in - parallel out register is shown below.

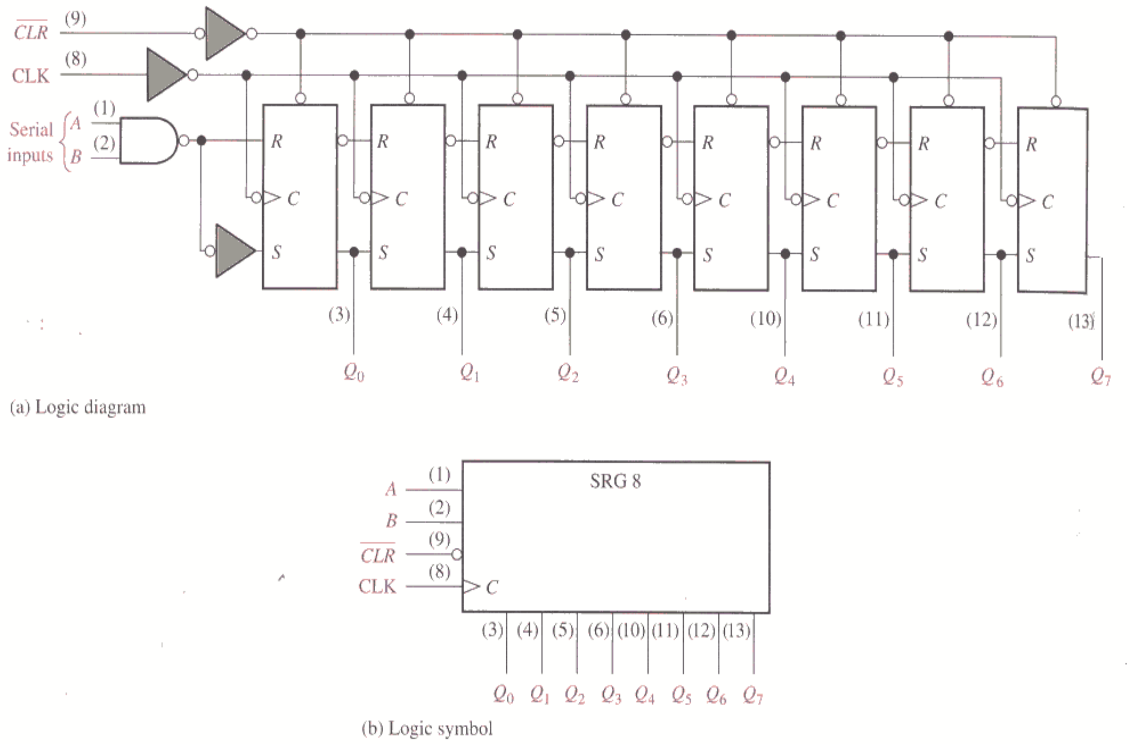


In the table below, we can see how the four-bit binary number 1001 is shifted to the Q outputs of the register.

Clear	FF0	FF1	FF2	FF3
1001	0	0	0	0
	1	0	0	0
	0	1	0	0
	0	0	1	0
	1	0	0	1

3.1 An 8-bit serial in/parallel out shift register (74HC164)

The 74HC164 is an example of an IC shift register having serial in/parallel out operation. The logic diagram and logic block are shown in Figure 3.1 (a),(b).

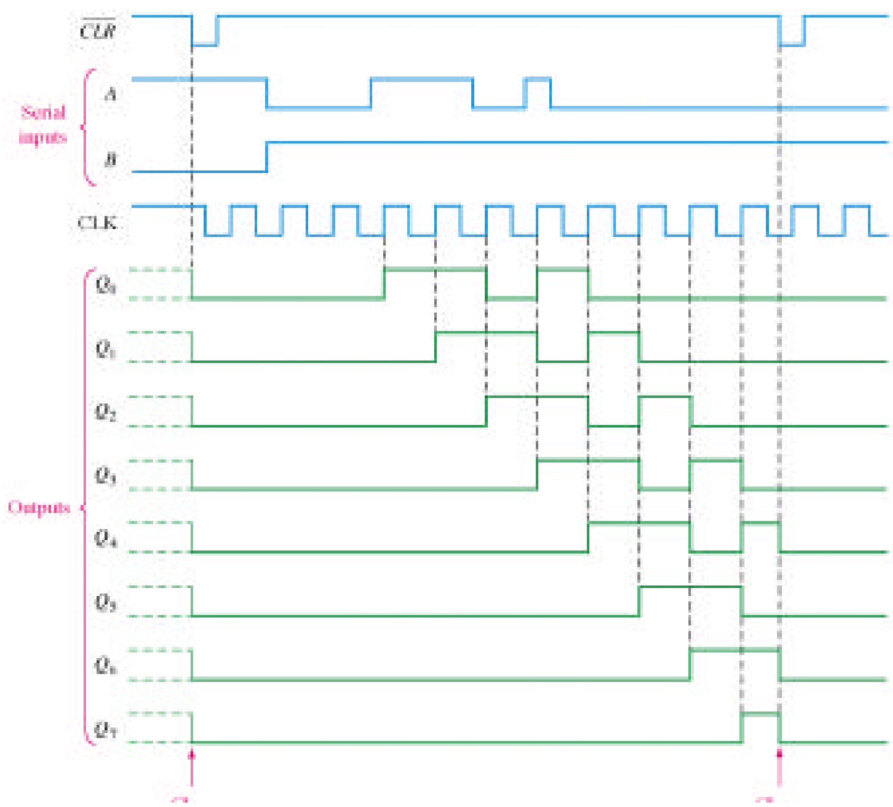


(a) Logic diagram

(b) Logic symbol

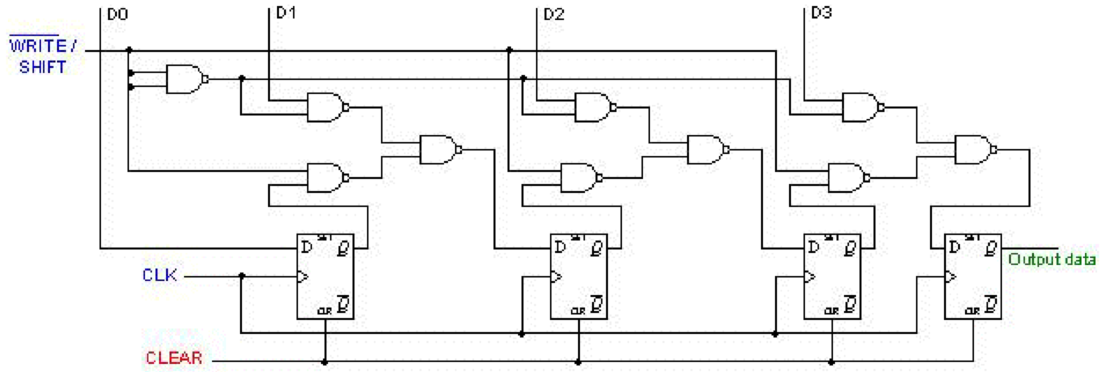
Figure 3.1: The logic diagram and logic block of 74HC164

Figure 3.2: The timing diagram of 74HC164



4.0 Parallel In - Serial Out Shift Registers

A four-bit parallel in - serial out shift register is shown below. The circuit uses D flip-flops and NAND gates for entering data (ie writing) to the register.

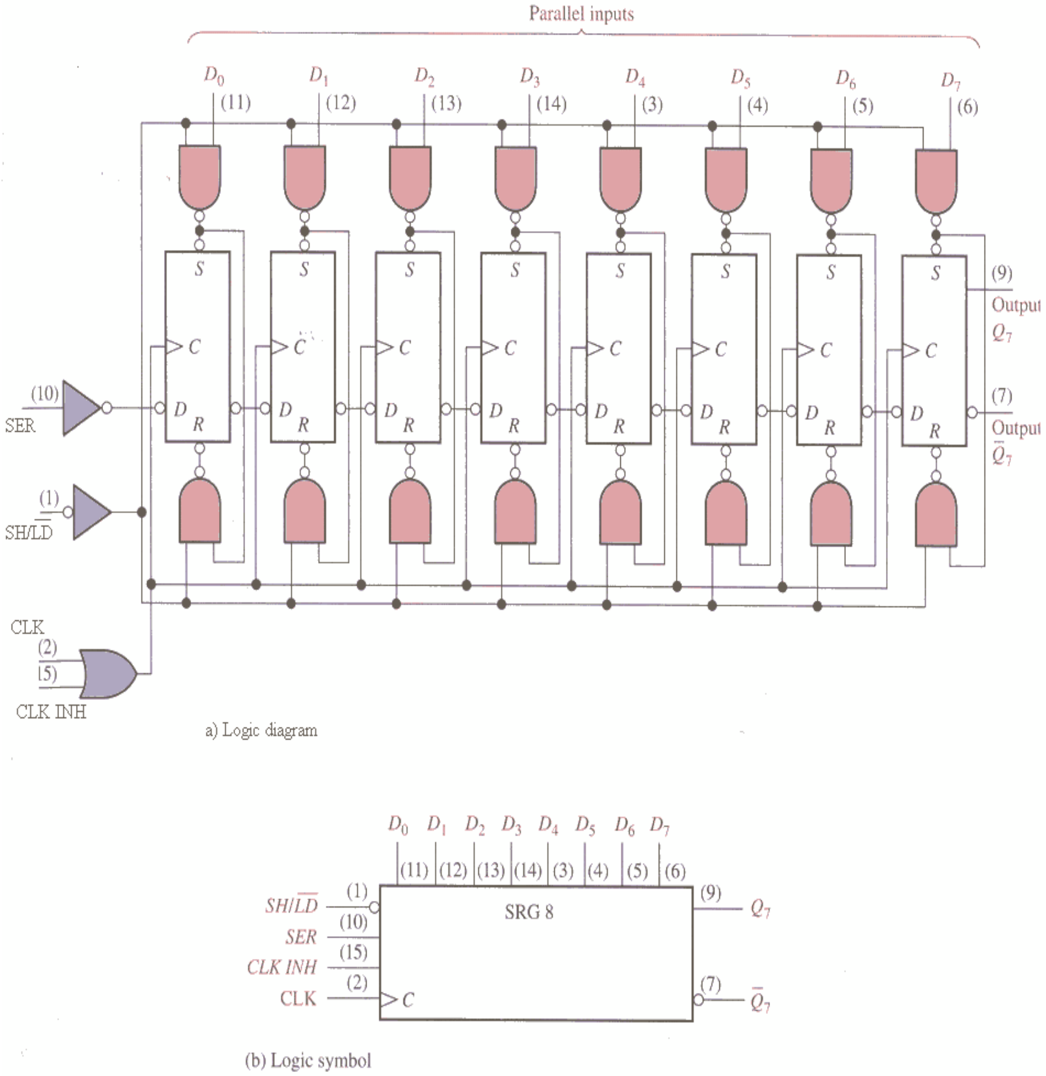


D0, D1, D2 and D3 are the parallel inputs, where D0 is the most significant bit and D3 is the least significant bit. To write data in, the mode control line is taken to LOW and the data is clocked in. The data can be shifted when the mode control line is HIGH as SHIFT is active high. The register performs right shift operation on the application of a clock pulse, as shown in the table below.

	Q_0	Q_1	Q_2	Q_3	
Clear	0	0	0	0	
Write	1	0	0	1	
Shift	1	0	0	1	
	1	1	0	0	1
	1	1	1	0	01
	1	1	1	1	001
	1	1	1	1	1001

4.1 8-bit Parallel Load Shift Register (74HC165)

The 74HC165 is an example of an IC shift register that has a parallel in/serial out operation. It can also be operated as serial in/serial out. Figure 4.1 shows the logic diagram and logic symbol of 74HC165.



a) Logic diagram

(b) Logic symbol

Figure 4.1: the logic diagram and logic symbol of 74HC165.

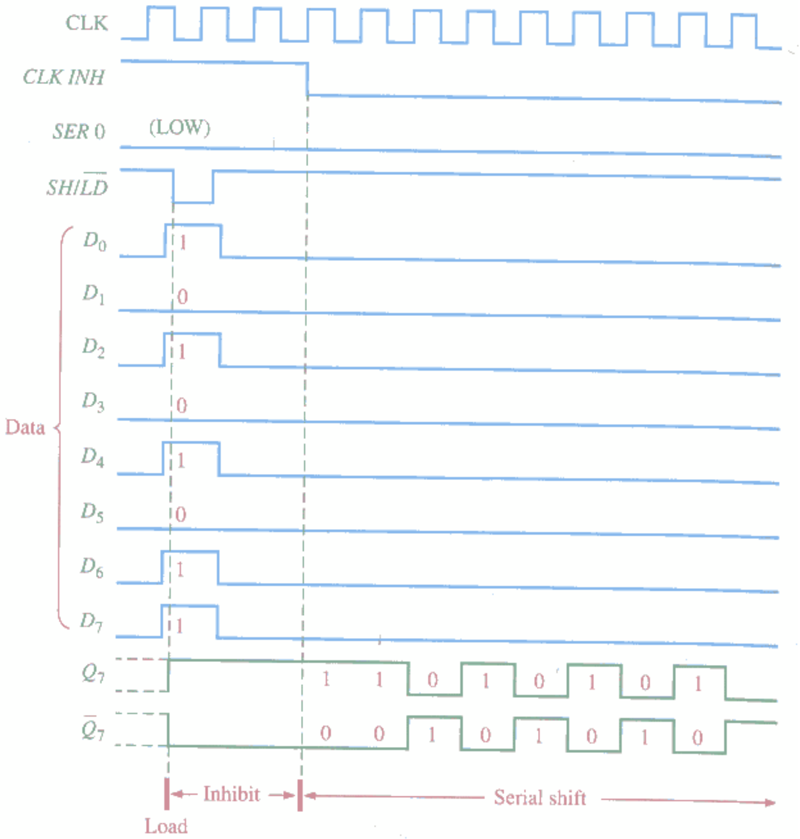


Figure 4.2: The timing diagram of 74HC165.

5.0 Parallel In - Parallel Out Shift Registers

For parallel in - parallel out shift registers, all data bits appear on the parallel outputs immediately following the simultaneous entry of the data bits. The following circuit is a four-bit parallel in - parallel out shift register constructed by D flip-flops.

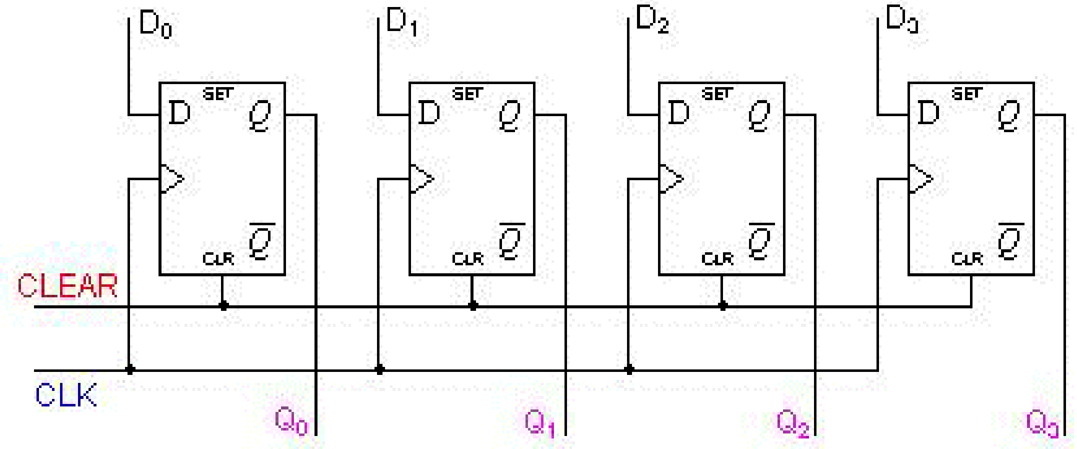


Figure 5.1

The D's are the parallel inputs and the Q's are the parallel outputs. Once the register is clocked, all the data at the D inputs appear at the corresponding Q outputs simultaneously.

5.1 4-bit Parallel-Access Shift Register (74HC195)

The 74HC195 can be used for parallel in/parallel out operation, serial in/serial out and serial in/parallel out operations. Q_3 is the output when it is used for parallel in/serial out operation.

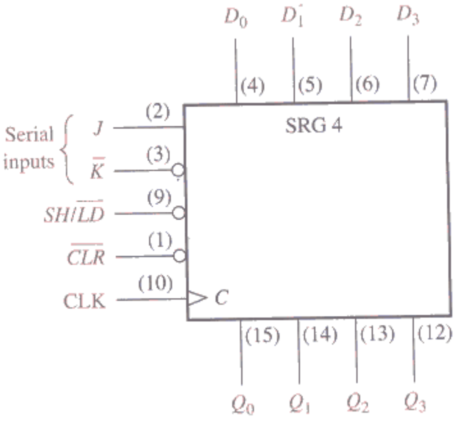


Figure 5.2: The 74LS195A 4-bit parallel access shift register

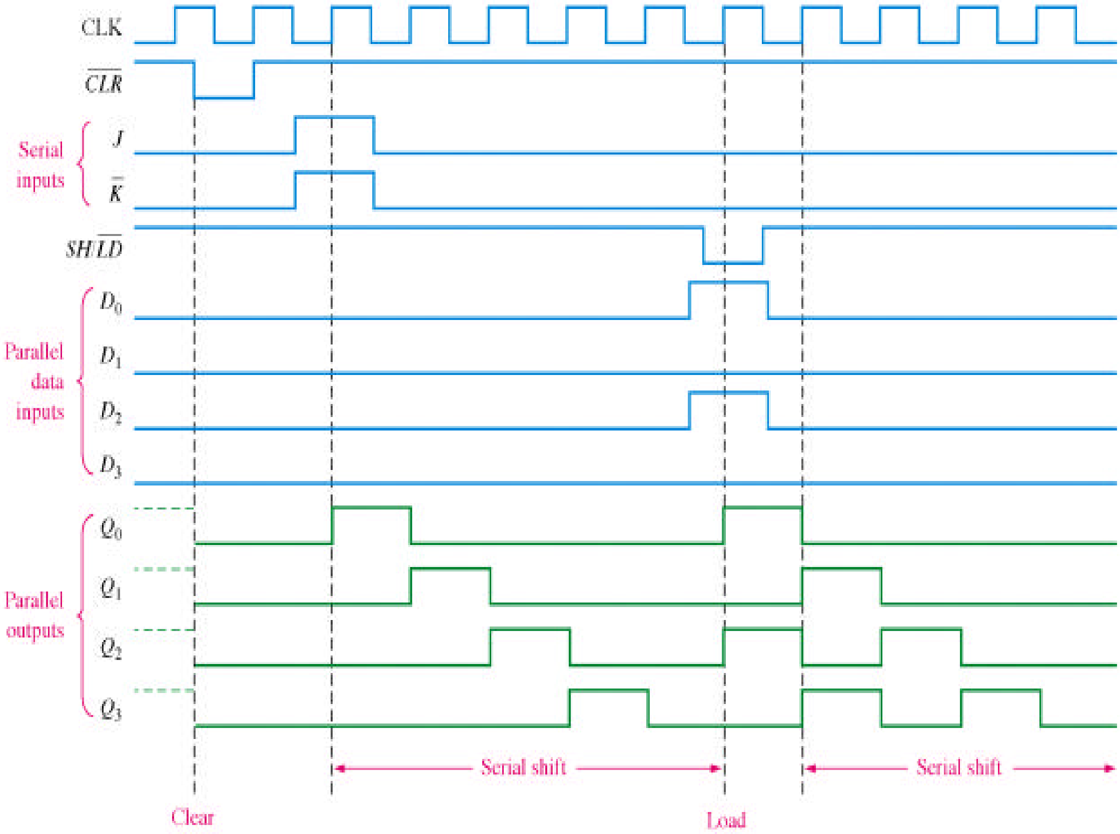
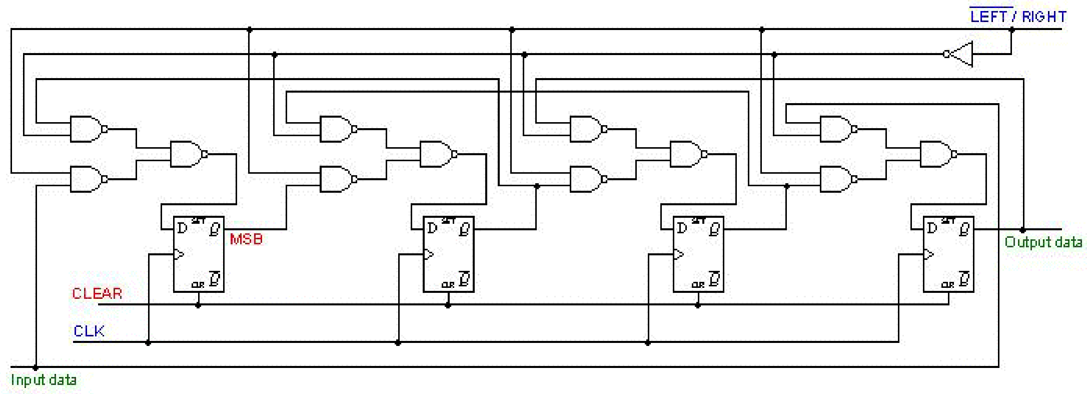


Figure 5.3: The timing diagram for 74LS195A shift register

6.0 Bidirectional Shift Registers

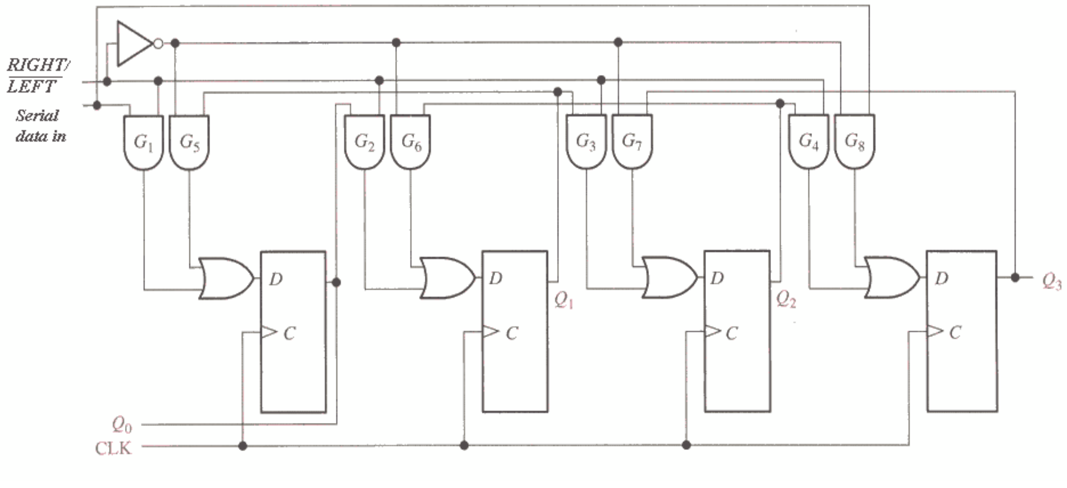
The registers discussed so far involved only right shift operations. Each right shift operation has the effect of successively dividing the binary number by two. If the operation is reversed (left shift), this has the effect of multiplying the number by two. With suitable gating arrangement a serial shift register can perform both operations.

A bidirectional, or reversible, shift register is one in which the data can be shift either left or right. A four-bit bidirectional shift register using D flip-flops is shown below.



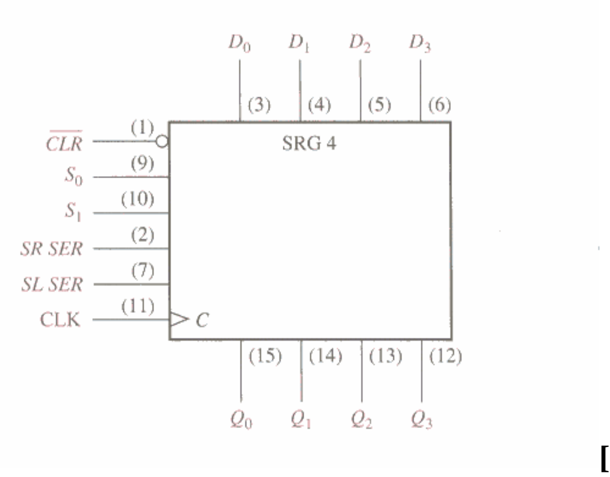
Here a set of NAND gates are configured as OR gates to select data inputs from the right or left adjacent bistables, as selected by the LEFT/RIGHT control line.

Alternative Circuit: [Floyd]



6.1 4-Bit Bidirectional Universal Shift Registers (74HC194)

The 74HC194 is a universal bi-directional shift register. It has both serial and parallel input and output capability.



[Floyd]

Figure 6.1:The 74HC194 4-bit bi-directional universal shift register

FIGURE 10–22 Sam

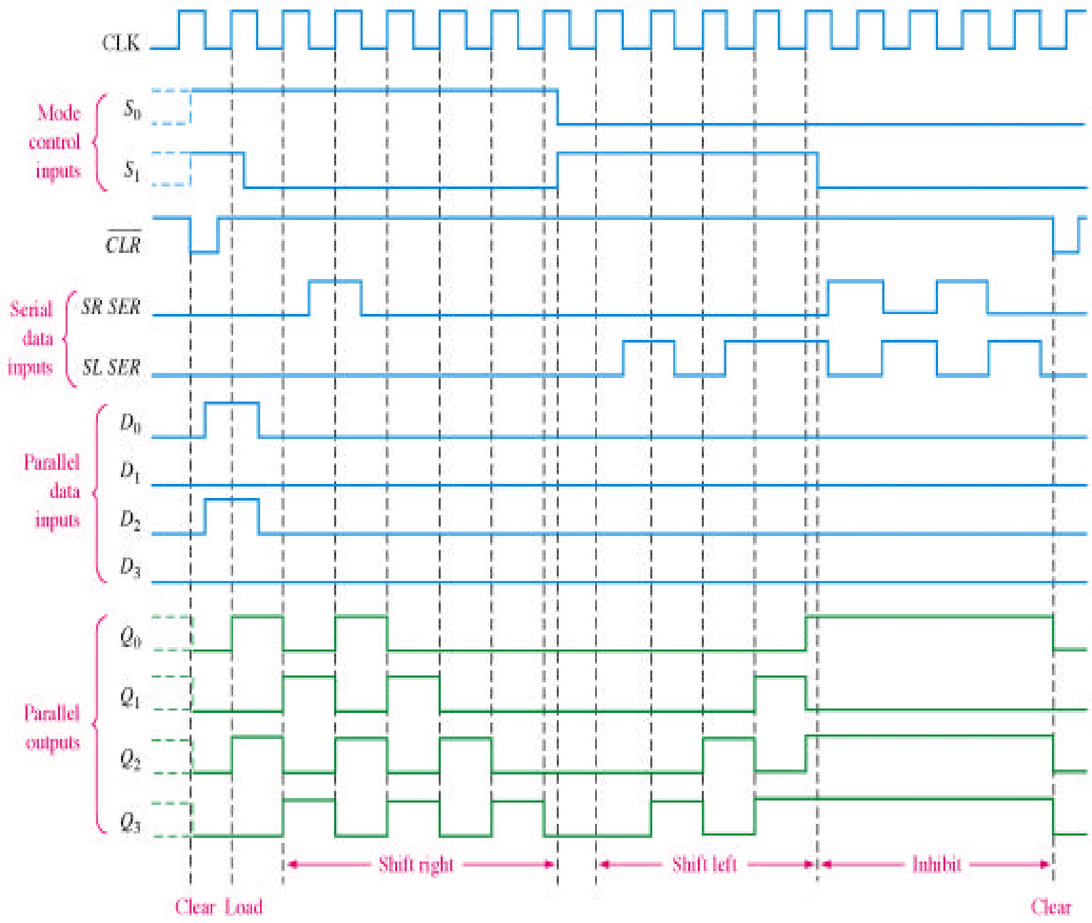


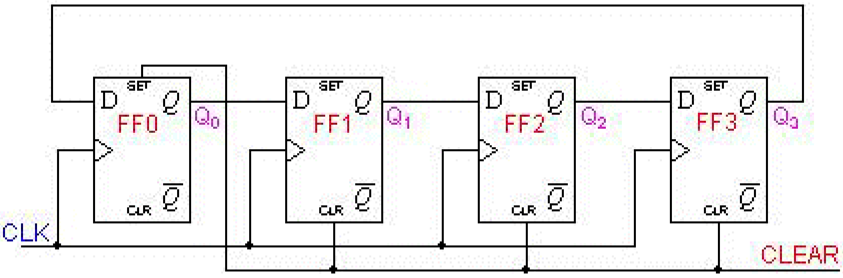
Figure 6.2:The timing diagram of 74HC194

7.0 Shift Register Counters

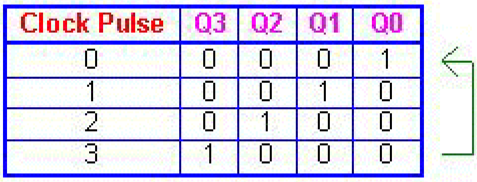
Two of the most common types of shift register counters are introduced here: the Ring counter and the Johnson counter. They are basically shift registers with the serial outputs connected back to the serial inputs in order to produce particular sequences. These registers are classified as counters because they exhibit a specified sequence of states.

7.1 Ring Counters

A ring counter is basically a circulating shift register in which the output of the most significant stage is fed back to the input of the least significant stage. The following is a 4-bit ring counter constructed from D flip-flops. The output of each stage is shifted into the next stage on the positive edge of a clock pulse. If the CLEAR signal is high, all the flip-flops except the first one FF0 are reset to 0. FF0 is preset to 1 instead.

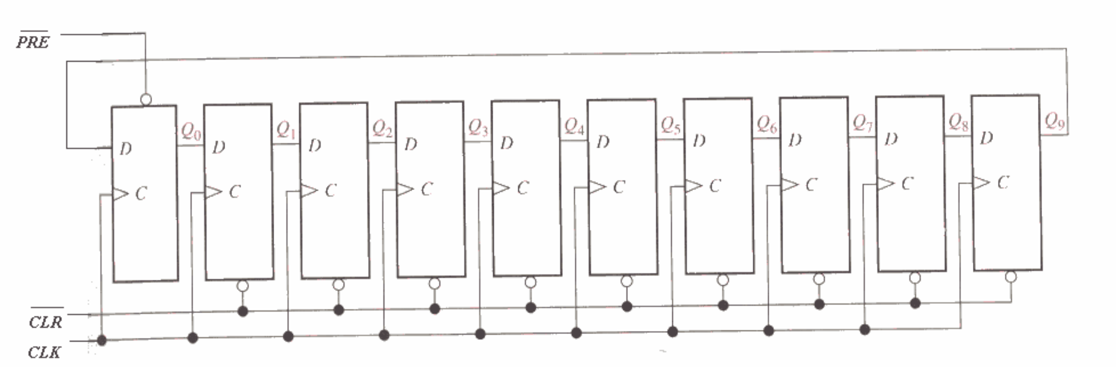


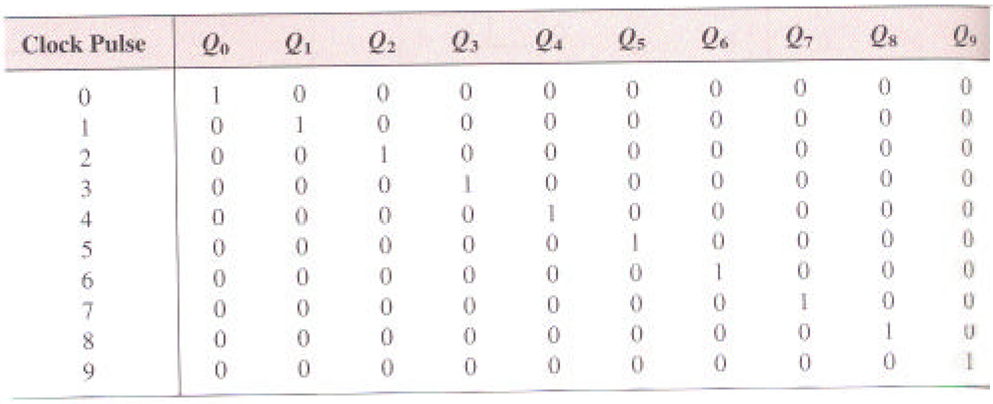
Since the count sequence has 4 distinct states, the counter can be considered as a mod-4 counter. Only 4 of the maximum 16 states are used, making ring counters very inefficient in terms of state usage. But the major advantage of a ring counter over a binary counter is that it is self-decoding. No extra decoding circuit is needed to determine what state the counter is in.



Clock Pulse	Q3	Q2	Q1	Q0
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	1	0	0	0

7.1.1 Example: A 10-bit Ring Counter [Flyod]





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

Figure 7.1: 10-bit ring counter & its sequence

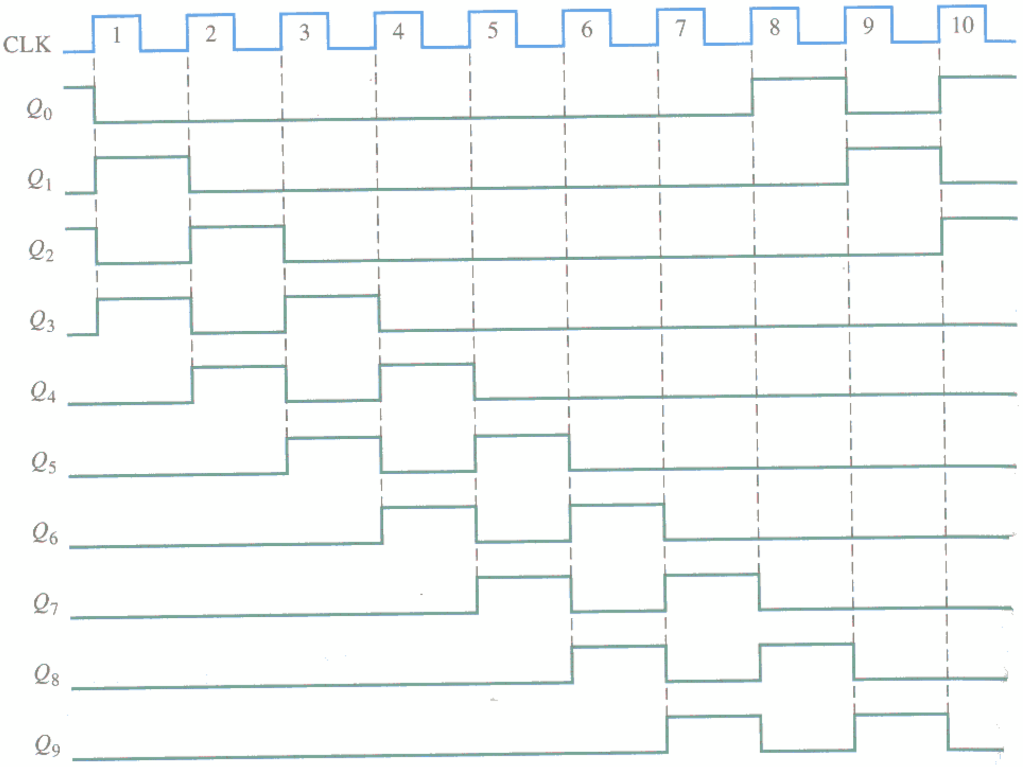
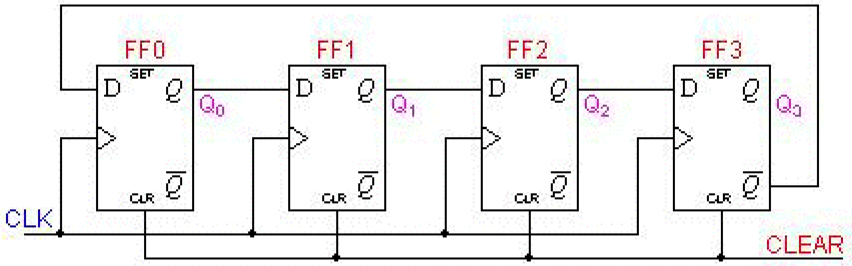
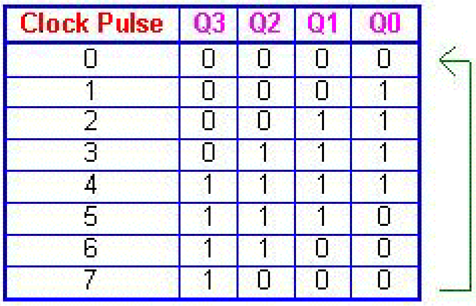


Figure 7.2: 10-bit ring counter waveform (initial state 1010000000)

7.2 Johnson Counters

Johnson counters are a variation of standard ring counters, with the inverted output of the last stage fed back to the input of the first stage. They are also known as twisted ring counters. An n-stage Johnson counter yields a count sequence of length 2n, so it may be considered to be a mod-2n counter. The circuit below shows a 4-bit Johnson counter. The state sequence for the counter is given in the table .



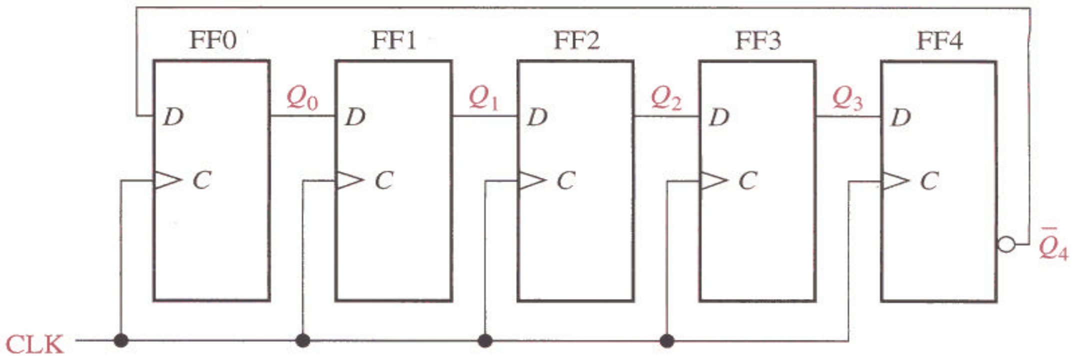


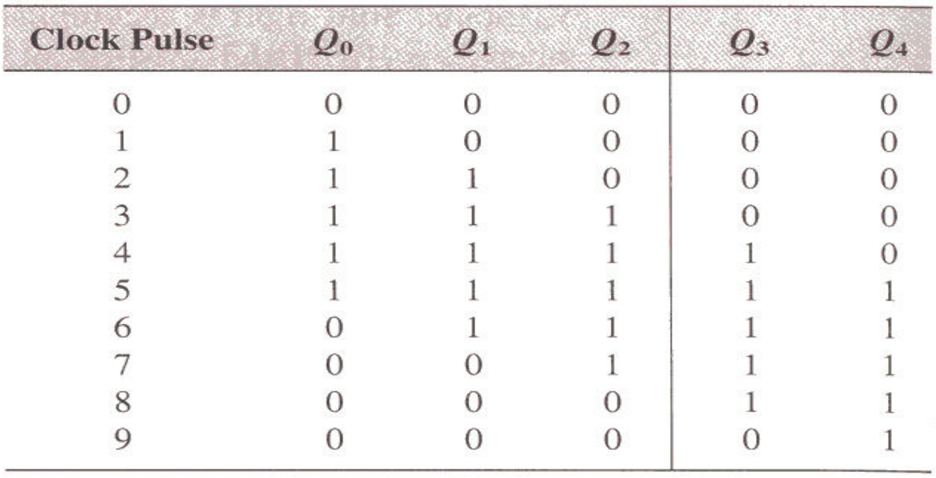
Clock Pulse	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	1	1	1
4	1	1	1	1
5	1	1	1	0
6	1	1	0	0
7	1	0	0	0

?? Again, the apparent disadvantage of this counter is that the maximum available states are not fully utilized. Only eight of the sixteen states are being used.

?? Beware that for both the Ring and the Johnson counter must initially be forced into a valid state in the count sequence because they operate on a subset of the available number of states. Otherwise, the ideal sequence will not be followed.

7.2.1 Example: 5 bit Johnson Counter [Flyod]





Clock Pulse	Q_0	Q_1	Q_2	Q_3	Q_4
0	0	0	0	0	0
1	1	0	0	0	0
2	1	1	0	0	0
3	1	1	1	0	0
4	1	1	1	1	0
5	1	1	1	1	1
6	0	1	1	1	1
7	0	0	1	1	1
8	0	0	0	1	1
9	0	0	0	0	1

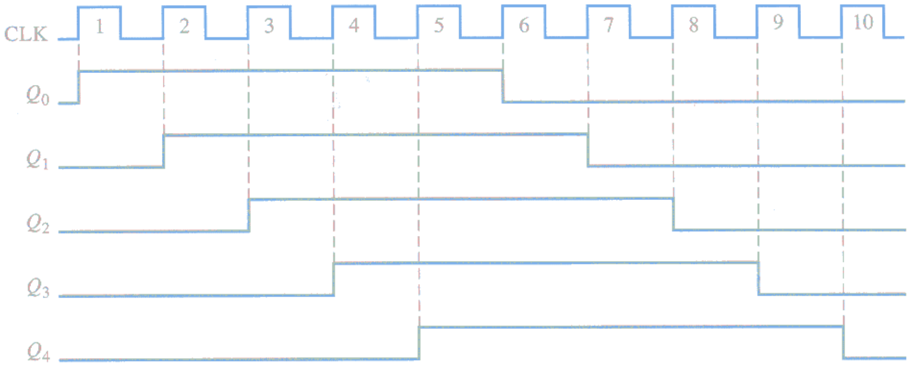


Figure 7.3: 5-bit Johnson Counter, its sequence and waveform

8.0 Applications

Shift registers can be found in many applications. Here is a list of a few.

?? To produce time delay

The serial in -serial out shift register can be used as a time delay device. The amount of delay can be controlled by:

1. the number of stages in the register
2. the clock frequency

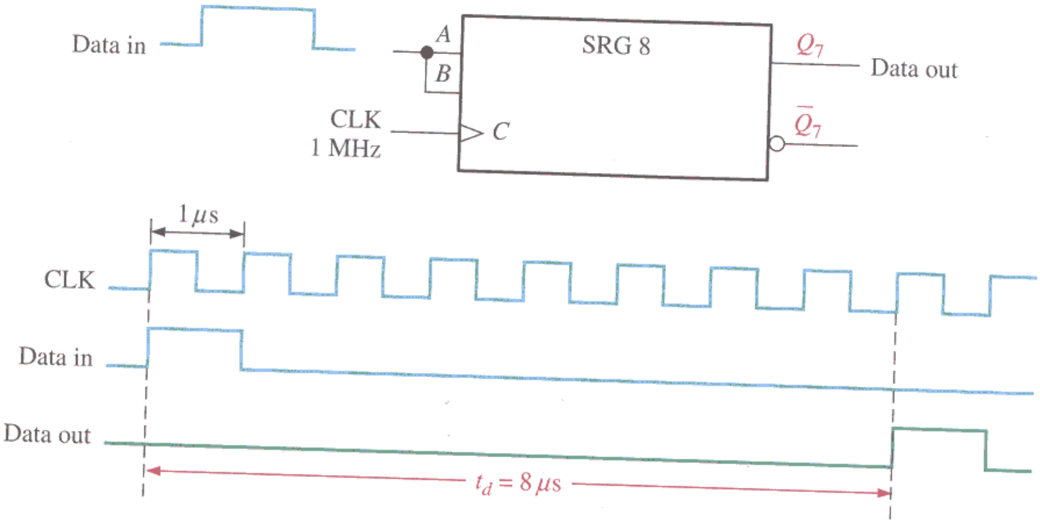


Figure 8.1: The shift register as a time-delay device.

?? **To simplify combinational logic**

The ring counter technique can be effectively utilized to implement synchronous sequential circuits. A major problem in the realization of sequential circuits is the assignment of binary codes to the internal states of the circuit in order to reduce the complexity of circuits required. By assigning one flip-flop to one internal state, it is possible to simplify the combinational logic required to realize the complete sequential circuit. When the circuit is in a particular state, the flip-flop corresponding to that state is set to HIGH and all other flip-flops remain LOW.

?? **To convert serial data to parallel data**

A computer or microprocessor-based system commonly requires incoming data to be in parallel format. But frequently, these systems must communicate with external devices that send or receive serial data. So, serial-to-parallel conversion is required. As shown in the previous sections, a serial in - parallel out register can achieve this.

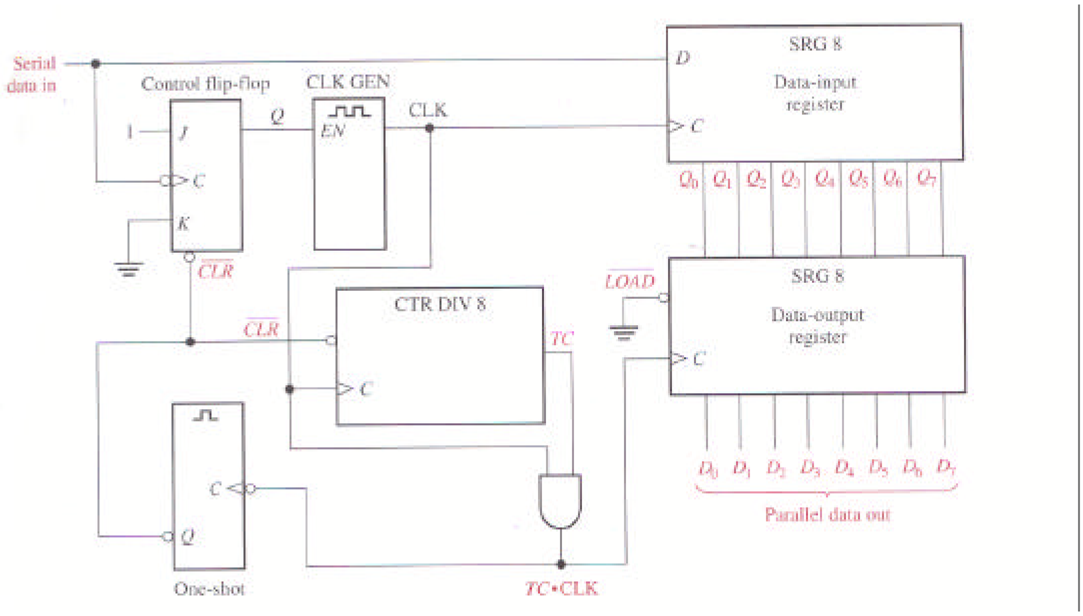


Figure 8.2: Simplified logic diagram of a serial-to-parallel converter