

SHIFT REGISTERS

CHAPTER OBJECTIVES

- Identify the basic forms of data movement in shift registers
- Explain how serial in/serial out, serial in/parallel out, parallel in/serial out, and parallel in/parallel out shift registers operate
- Describe how a bidirectional shift register operates
- Determine the sequence of a Johnson counter
- Set up a ring counter to produce a specified sequence
- Construct a ring counter from a shift register
- Use a shift register as a time-delay device

- Use a shift register to implement a serial-to-parallel data converter
- Implement a basic shift-register-controlled keyboard encoder

INTRODUCTION

Shift registers are a type of sequential logic circuit closely related to digital counters. Registers are used primarily for the storage of digital data and typically do not possess a characteristic internal sequence of states as do counters. There are exceptions, however, and these are covered in Section 9-7.

In this chapter, the basic types of shift registers are studied and several applications are presented.

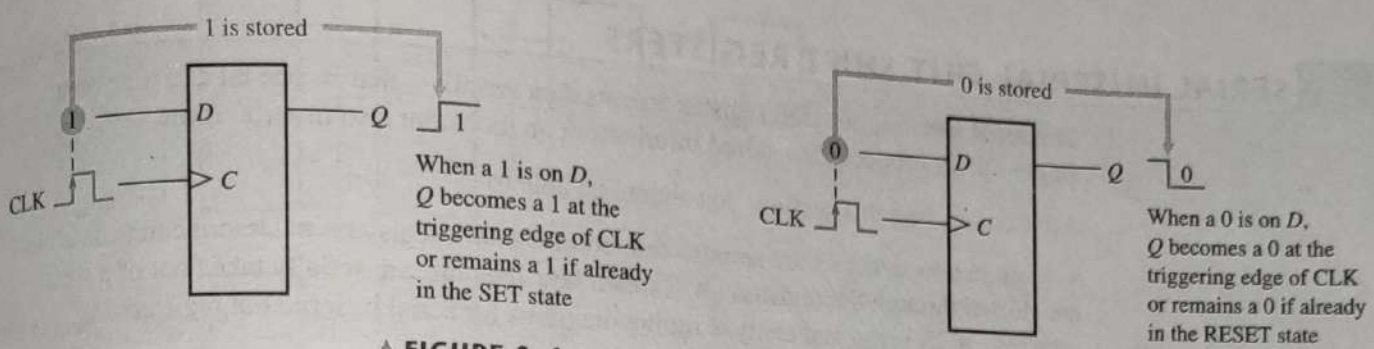
9-1 BASIC SHIFT REGISTER FUNCTIONS

Shift registers consist of an arrangement of flip-flops and are important in applications involving the storage and transfer of data in a digital system. A register, unlike a counter, has no specified sequence of states, except in certain very specialized applications. A register, in general, is used solely for storing and shifting data (1s and 0s) entered into it from an external source and typically possesses no characteristic internal sequence of states.

After completing this section, you should be able to

- Explain how a flip-flop stores a data bit
- Define the storage capacity of a shift register
- Define the shifting capability of a register

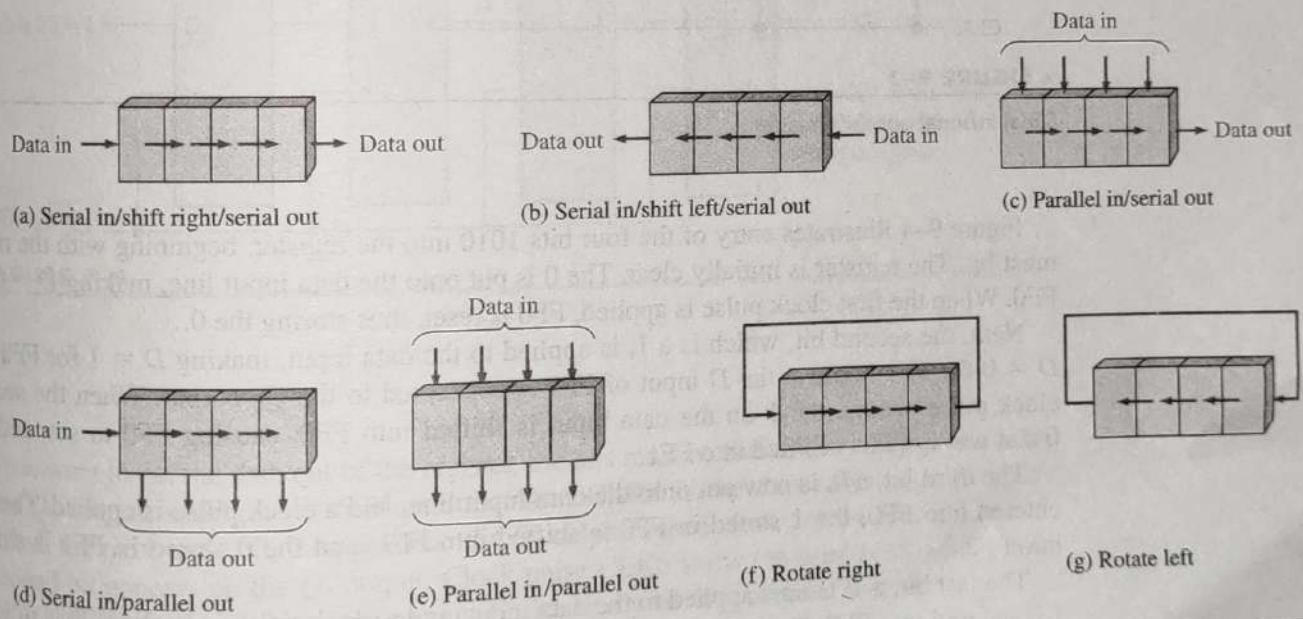
A register is a digital circuit with two basic functions: data storage and data movement. The storage capability of a register makes it an important type of memory device. A register can consist of one or more flip-flops used to store and shift data. Figure 9-1 illustrates the concept of storing a 1 or a 0 in a D flip-flop. A 1 is applied to the data input as shown, and a clock pulse is applied that stores the 1 by setting the flip-flop. When the 1 on the input is removed, the flip-flop remains in the SET state, thereby storing the 1. A similar procedure applies to the storage of a 0 by resetting the flip-flop, as also illustrated in Figure 9-1.



▲ FIGURE 9-1
The flip-flop as a storage element.

The storage capacity of a register is the total number of bits (1s and 0s) 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.

The shifting capability of a register permits the movement of data from stage to stage within the register or into or out of the register upon application of clock pulses. Figure 9-2 illustrates the types of data movement in shift registers. The block represents any arbitrary 4-bit register, and the arrows indicate the direction of data movement.



▲ FIGURE 9-2
Basic data movement in shift registers.

SECTION 9-1 REVIEW

1. Generally, what is the difference between a counter and a shift register?
2. What two principal functions are performed by a shift register?

9-2

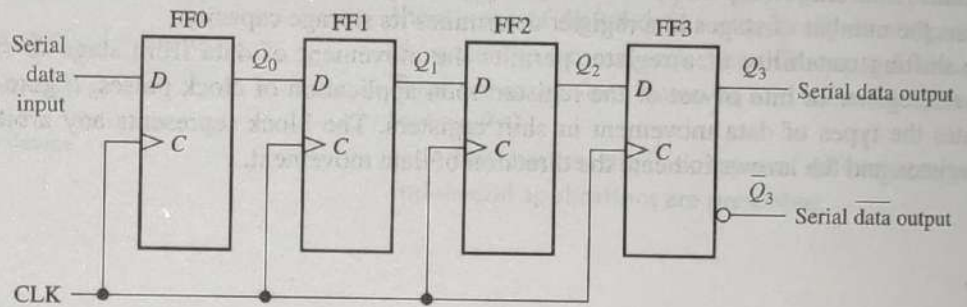
SERIAL IN/SERIAL OUT SHIFT REGISTERS

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.

After completing this section, you should be able to

- Explain how data bits are serially entered into a shift register
- Describe how data bits are shifted through the register
- Explain how data bits are serially taken out of a shift register
- Develop and analyze timing diagrams for serial in/serial out registers

Let us first look at the serial entry of data into a typical shift register. Figure 9-3 shows a 4-bit device implemented with D flip-flops. With four stages, this register can store up to four bits of data.



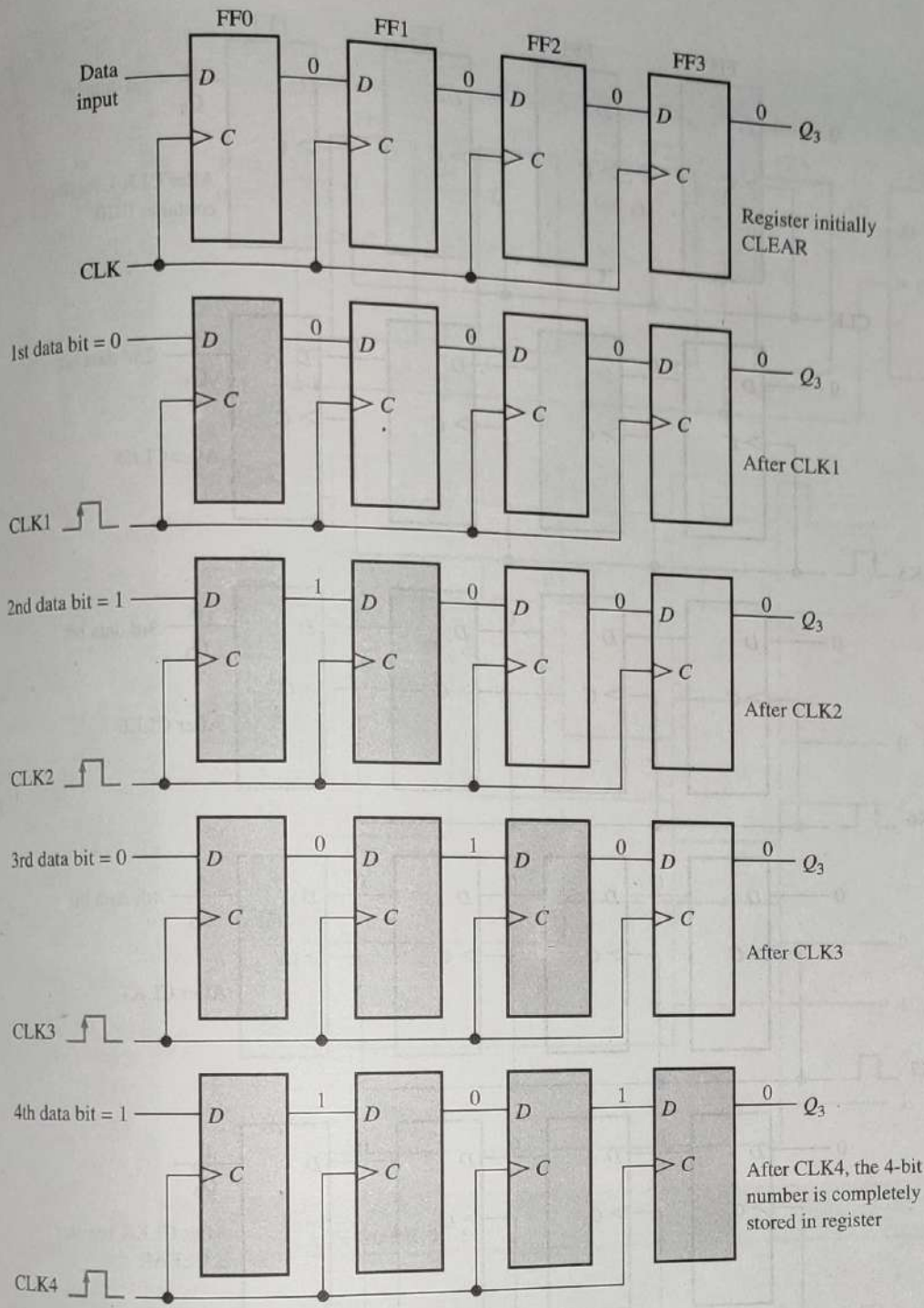
▲ FIGURE 9-3
Serial in/serial out shift register

Figure 9-4 illustrates entry of the four bits 1010 into the register, beginning with the right-most bit. The register is initially clear. The 0 is put onto the data input line, making $D = 0$ for FF0. When the first clock pulse is applied, FF0 is reset, thus storing the 0.

Next, the second bit, which is a 1, is applied to the data input, making $D = 1$ for FF0 and $D = 0$ for FF1 because the D input of FF1 is connected to the Q_0 output. When the second clock pulse occurs, the 1 on the data input is shifted into FF0, causing FF0 to set; and the 0 that was in FF0 is shifted into FF1.

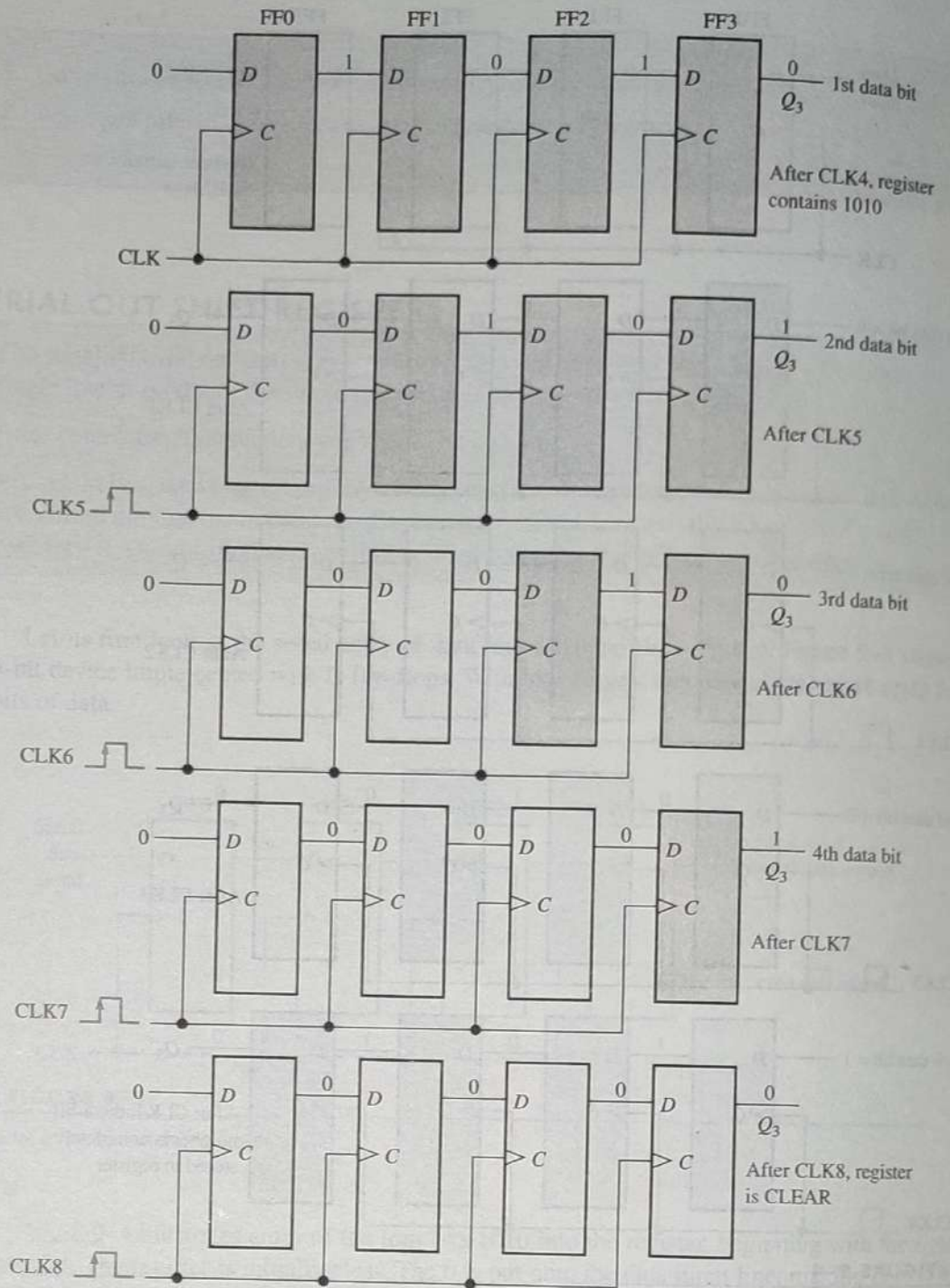
The third bit, a 0, is now put onto the data-input line, and a clock pulse is applied. The 0 is entered into FF0, the 1 stored in FF0 is shifted into FF1, and the 0 stored in FF1 is shifted into FF2.

The last bit, a 1, is now applied to the data input, and a clock pulse is applied. This time the 1 is entered into FF0, the 0 stored in FF0 is shifted into FF1, the 1 stored in FF1 is shifted into FF2, and the 0 stored in FF2 is shifted into FF3. This completes the serial entry of the four bits into the shift register, where they can be stored for any length of time as long as the flip-flops have dc power. Thus we see that for serial data, one bit at a time is transferred.



▲ FIGURE 9-4

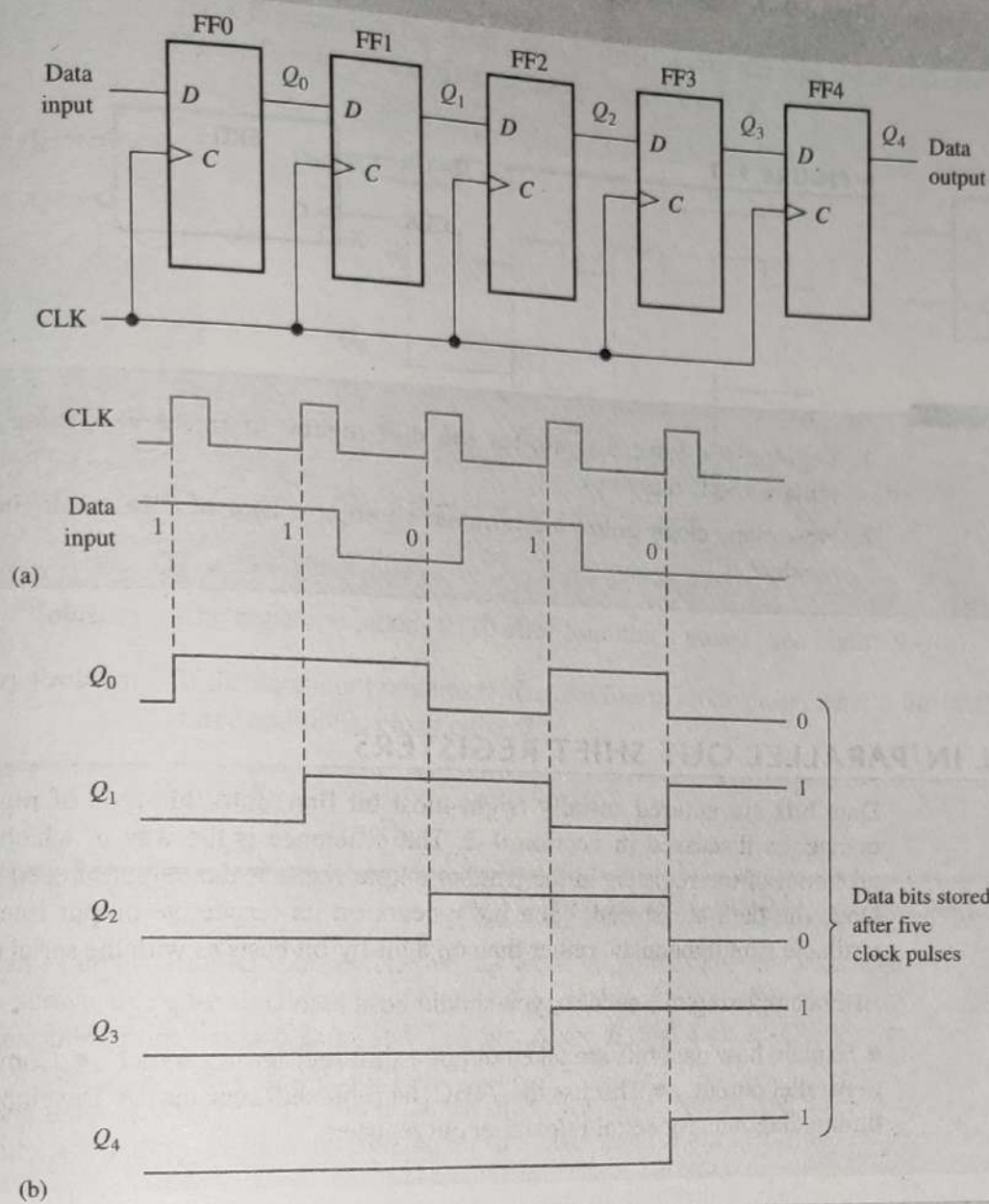
If you want to get the data out of the register, the bits must be shifted out serially and taken off the Q_3 output, as Figure 9-5 illustrates. After CLK4 in the data-entry operation just described, the right-most bit, 0, appears on the Q_3 output. When clock pulse CLK5 is applied, the second bit appears on the Q_3 output. Clock pulse CLK6 shifts the third bit to the output, and CLK7 shifts the fourth bit to the output. Notice that while the original four bits are being shifted out, more bits can be shifted in. All zeros are shown being shifted in.



▲ FIGURE 9-5

EXAMPLE 9-1

Show the states of the 5-bit register in Figure 9-6(a) for the specified data input and clock waveforms. Assume that the register is initially cleared (all 0s).



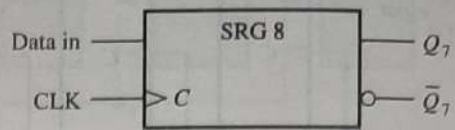
▲ FIGURE 9-6

Solution The first data bit (1) is entered into the register on the first clock pulse and then shifted from left to right as the remaining bits are entered and shifted. The register contains $Q_4Q_3Q_2Q_1Q_0 = 11010$ after five clock pulses. See Figure 9-6(b).

Supplementary Problem Show the states of the register if the data input is inverted. The register is initially cleared.

A traditional logic block symbol for an 8-bit serial in/serial out shift register is shown in Figure 9-7. The "SRG 8" designation indicates a shift register (SRG) with an 8-bit capacity.

► FIGURE 9-7



SECTION 9-2 REVIEW

1. Develop the logic diagram for the shift register in Figure 9-3, using J-K flip-flops to replace the D flip-flops.
2. How many clock pulses are required to enter a byte of data serially into an 8-bit shift register?

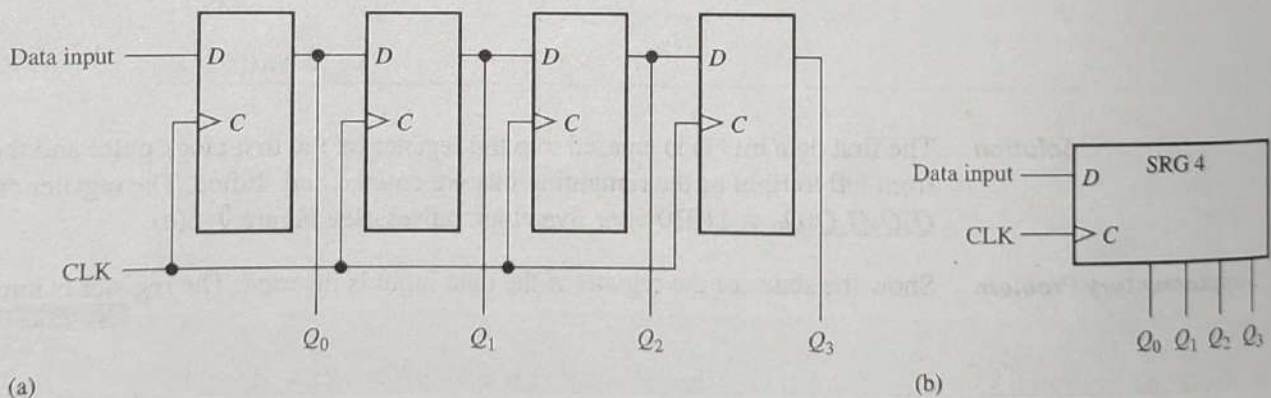
9-3 SERIAL IN/PARALLEL OUT SHIFT REGISTERS

Data bits are entered serially (right-most bit first) into this type of register in the same manner as discussed in Section 9-2. The difference is the way in which the data bits are taken out of the register; in the parallel output register, the output of each stage is available. Once the data are stored, each bit appears on its respective output line, and all bits are available simultaneously, rather than on a bit-by-bit basis as with the serial output.

After completing this section, you should be able to

- Explain how data bits are taken out of a shift register in parallel
- Compare serial output to parallel output
- Discuss the 74HC164 8-bit shift register
- Develop and analyze timing diagrams for serial in/parallel out registers

Figure 9-8 shows a 4-bit serial in/parallel out shift register and its logic block symbol.



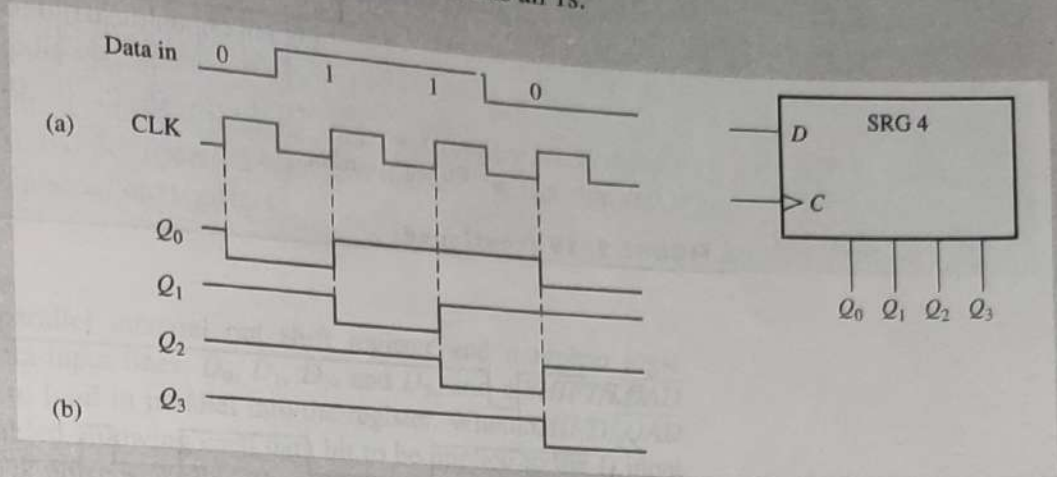
▲ FIGURE 9-8

A serial in/parallel out shift register

EXAMPLE 9-2

► **FIGURE 9-9**

Show the states of the 4-bit register (SRG 4) for the data input and clock waveforms in Figure 9-9(a). The register initially contains all 1s.



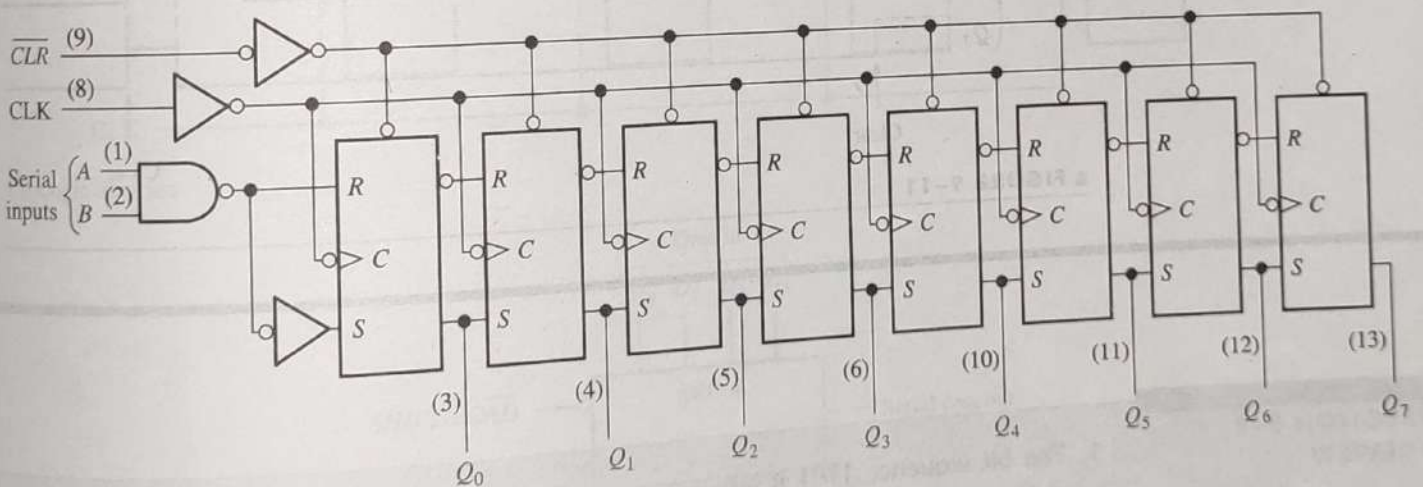
Solution The register contains 0110 after four clock pulses. See Figure 9-9(b).

Supplementary Problem If the data input remains 0 after the fourth clock pulse, what is the state of the register after three additional clock pulses?

AN 8-BIT SERIAL IN/PARALLEL OUT SHIFT REGISTER

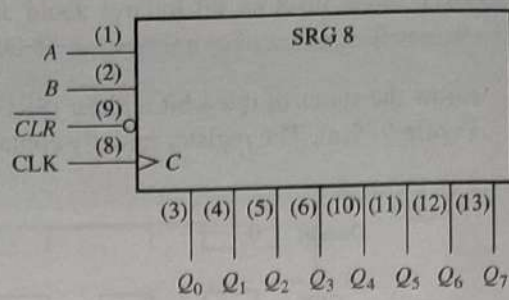
The 74HC164 is an example of an IC shift register having serial in/parallel out operation. The logic diagram is shown in Figure 9-10(a), and a typical logic block symbol is shown in part (b). Notice that this device has two gated serial inputs, A and B, and a clear (\overline{CLR}) input that is active-LOW. The parallel outputs are Q_0 through Q_7 .

A sample timing diagram for the 74HC164 is shown in Figure 9-11. Notice that the serial input data on input A are shifted into and through the register after input B goes HIGH.



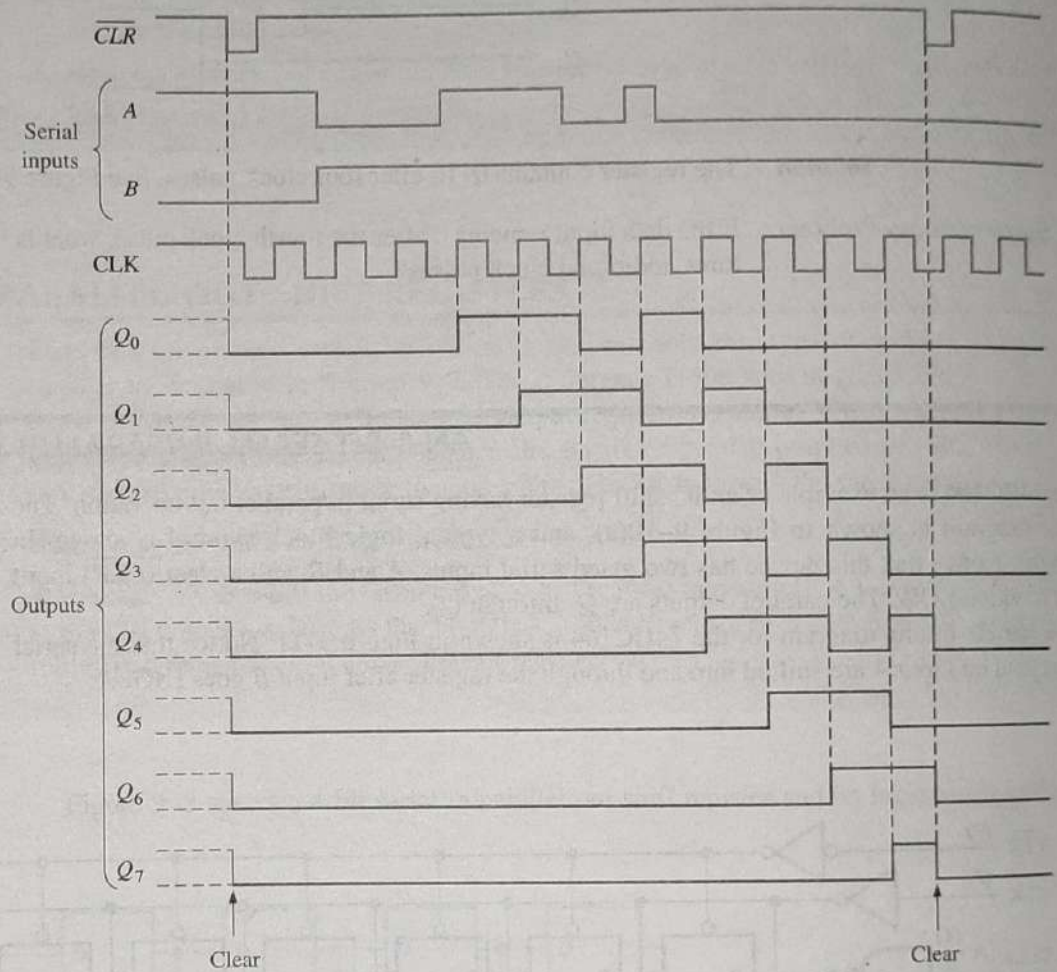
(a) Logic diagram

▲ **FIGURE 9-10**



(b) Logic symbol

▲ FIGURE 9-10 (continued)



▲ FIGURE 9-11

**SECTION 9-3
REVIEW**

1. The bit sequence 1101 is serially entered (right-most bit first) into a 4-bit parallel out shift register that is initially clear. What are the Q outputs after two clock pulses?
2. How can a serial in/parallel out register be used as a serial in/serial out register?

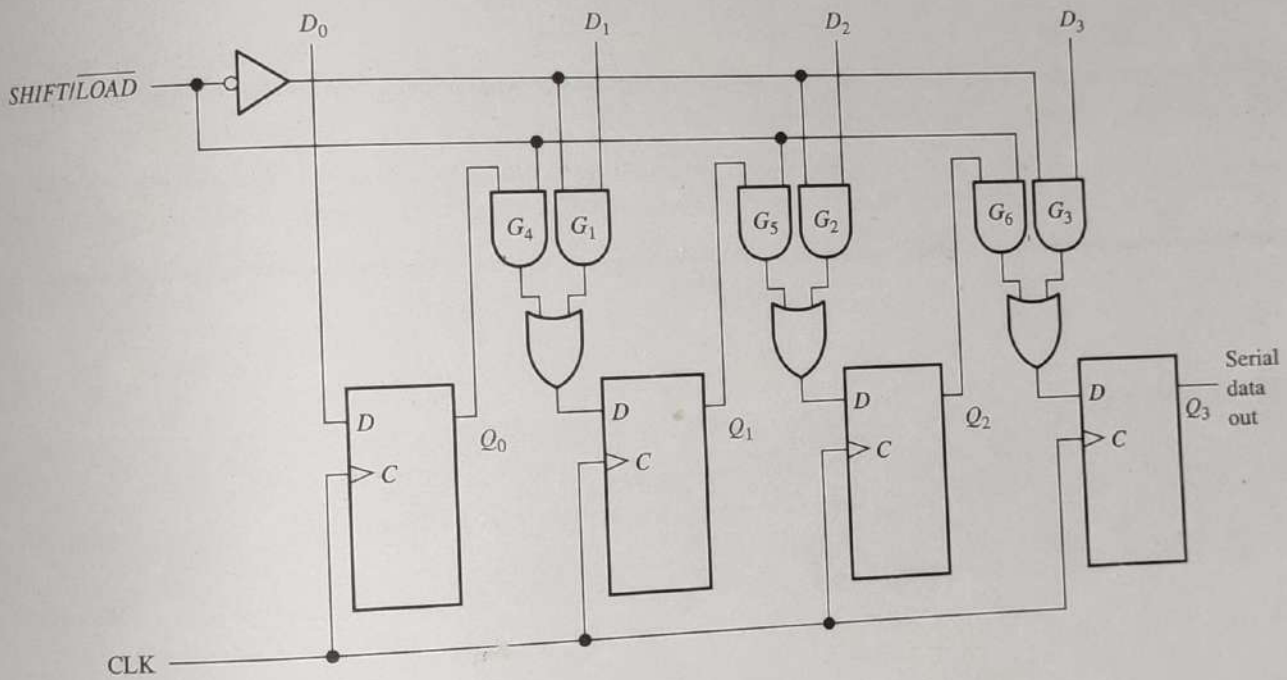
9-4 PARALLEL IN/SERIAL OUT SHIFT REGISTERS

For a register with parallel data inputs, the 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 inputs. The serial output is the same as described in Section 9-2, once the data are completely stored in the register.

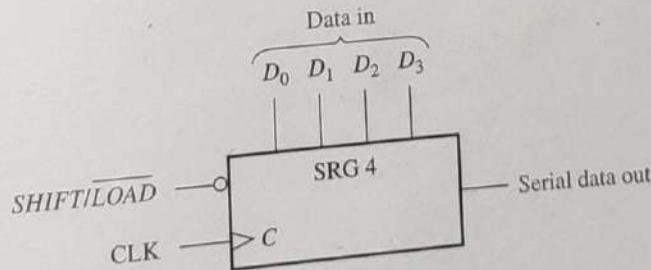
After completing this section, you should be able to

- Explain how data bits are entered into a shift register in parallel
- Compare serial input to parallel input
- Discuss the 74HC165 8-bit parallel-load shift register
- Develop and analyze timing diagrams for parallel in/serial out registers

Figure 9-12 illustrates a 4-bit parallel in/serial out shift register and a typical logic symbol. Notice that there are four data-input lines, D_0 , D_1 , D_2 , and D_3 , and a $\overline{SHIFT/LOAD}$ input, which allows four bits of data to load in parallel into the register. When $\overline{SHIFT/LOAD}$ is LOW, gates G_1 through G_3 are enabled, allowing each data bit to be applied to the D input of its respective flip-flop. When a clock pulse is applied, the flip-flops with $D = 1$ will set and those with $D = 0$ will reset, thereby storing all four bits simultaneously.

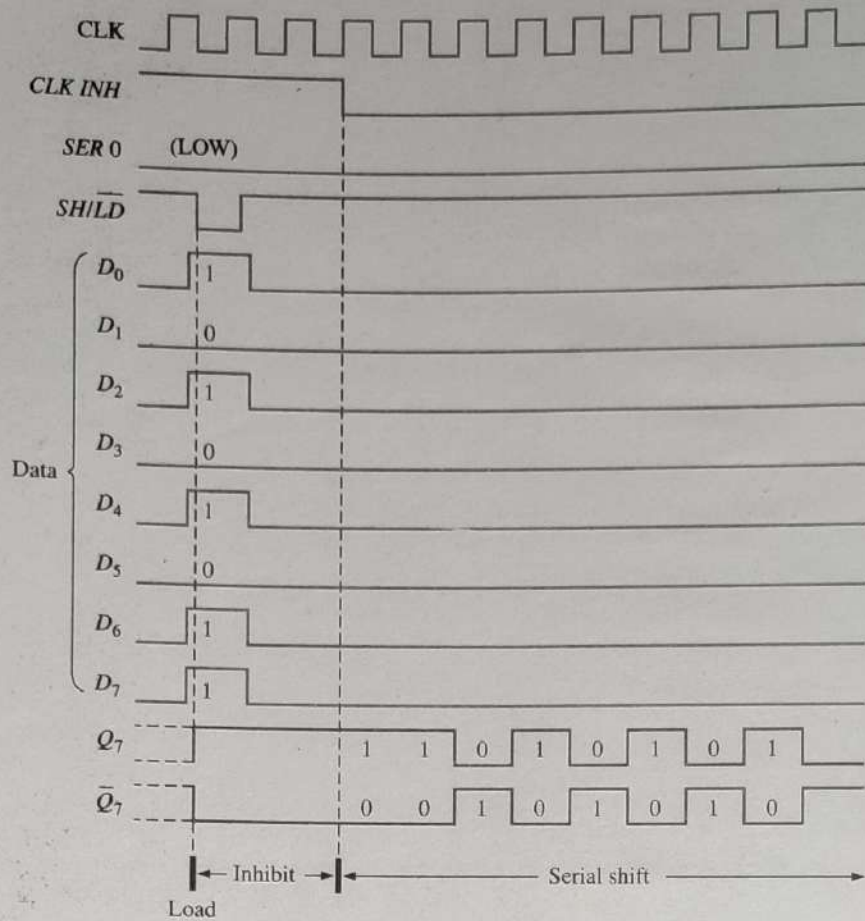


(a) Logic diagram



(b) Logic symbol

▲ FIGURE 9-12



▲ FIGURE 9-15

**SECTION 9-4
REVIEW**

1. Explain the function of the $SHIFT/LOAD$ input.
2. Is the parallel load operation in a 74HC165 shift register synchronous or asynchronous? What does this mean?

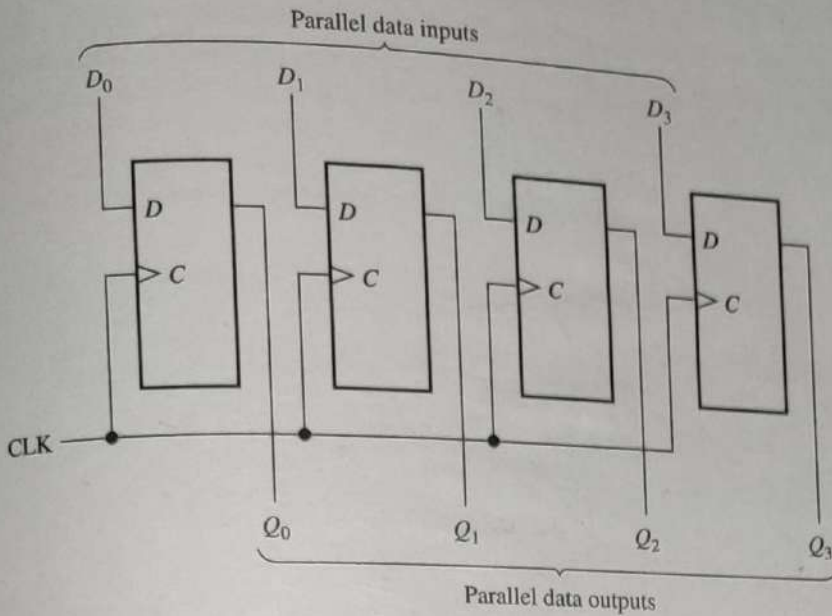
9-5 PARALLEL IN/PARALLEL OUT SHIFT REGISTERS

Parallel entry of data was described in Section 9-4, and parallel output of data has also been discussed previously. The parallel in/parallel out register employs both methods. Immediately following the simultaneous entry of all data bits, the bits appear on the parallel outputs.

After completing this section, you should be able to

- Discuss the 74HC195 4-bit parallel-access shift register
- Develop and analyze timing diagrams for parallel in/parallel out registers

Figure 9-16 shows a parallel in/parallel out register.

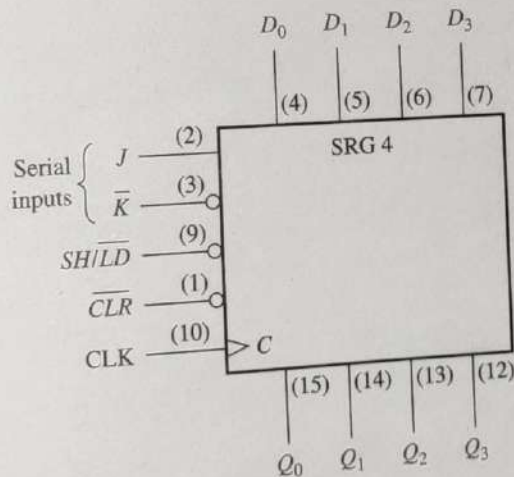


▲ FIGURE 9-16

A 4-BIT PARALLEL-ACCESS SHIFT REGISTER

The 74HC195 can be used for parallel in/parallel out operation. Because it also has a serial input, it can be used for serial in/serial out and serial in/parallel out operations. It can be used for parallel in/serial out operation by using Q_3 as the output. A typical logic block symbol is shown in Figure 9-17.

► FIGURE 9-17



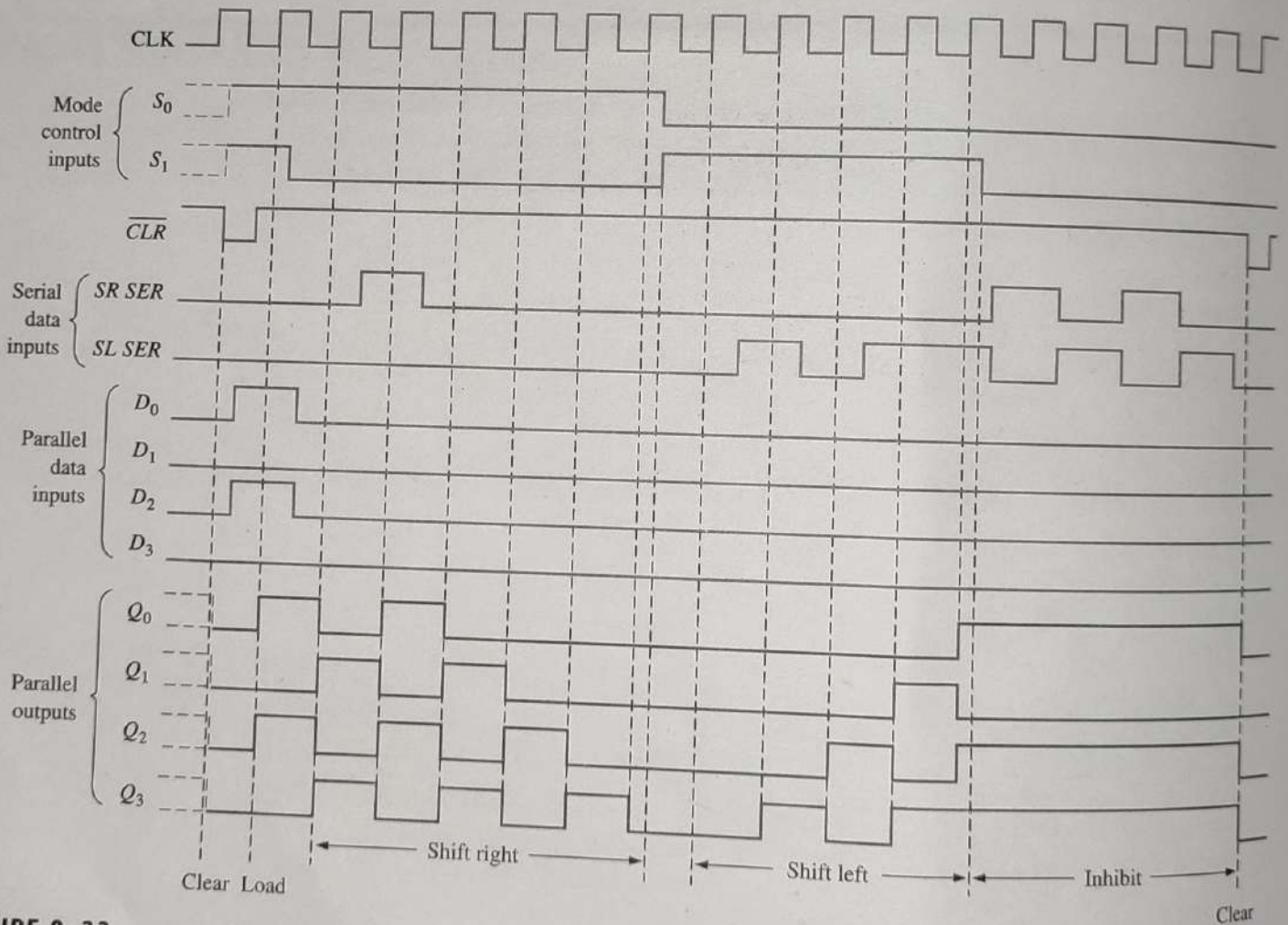
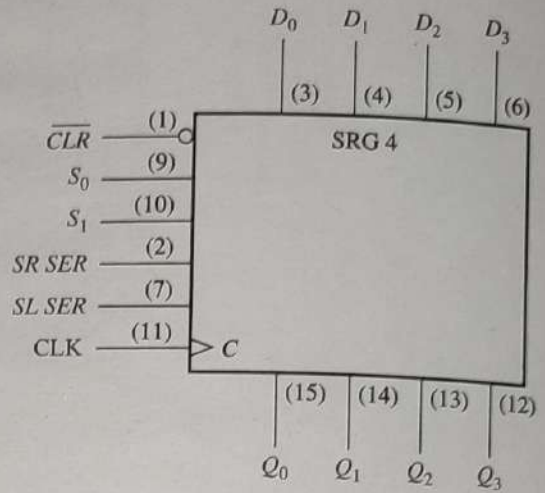
When the *SHIFT/LOAD* input (SH/\overline{LD}) is LOW, the data on the parallel inputs are entered synchronously on the positive transition of the clock. When SH/\overline{LD} is HIGH, stored data will shift right (Q_0 to Q_3) synchronously with the clock. Inputs J and \overline{K} are the serial data inputs to the first stage of the register (Q_0); Q_3 can be used for serial output data. The active-LOW clear input is asynchronous.

The timing diagram in Figure 9-18 illustrates the operation of this register.

A 4-BIT BIDIRECTIONAL UNIVERSAL SHIFT REGISTER

The 74HC194 is an example of a universal bidirectional shift register in integrated circuit form. A **universal shift register** has both serial and parallel input and output capability. A logic block symbol is shown in Figure 9-21, and a sample timing diagram is shown in Figure 9-22.

► FIGURE 9-21



▲ FIGURE 9-22

Parallel loading, which is synchronous with a positive transition of the clock, is accomplished by applying the four bits of data to the parallel inputs and a HIGH to the S_0 and S_1 inputs. Shift right is accomplished synchronously with the positive edge of the clock when S_0 is HIGH and S_1 is LOW. Serial data in this mode are entered at the shift-right serial input ($SR\ SER$). When S_0 is LOW and S_1 is HIGH, data bits shift left synchronously with the clock, and new data are entered at the shift-left serial input ($SL\ SER$). Input $SR\ SER$ goes into the Q_0 stage, and $SL\ SER$ goes into the Q_3 stage.