DIGITAL LOGIC DESIGN EE(227)

Chapter No:8
Shift Registers

Outlines

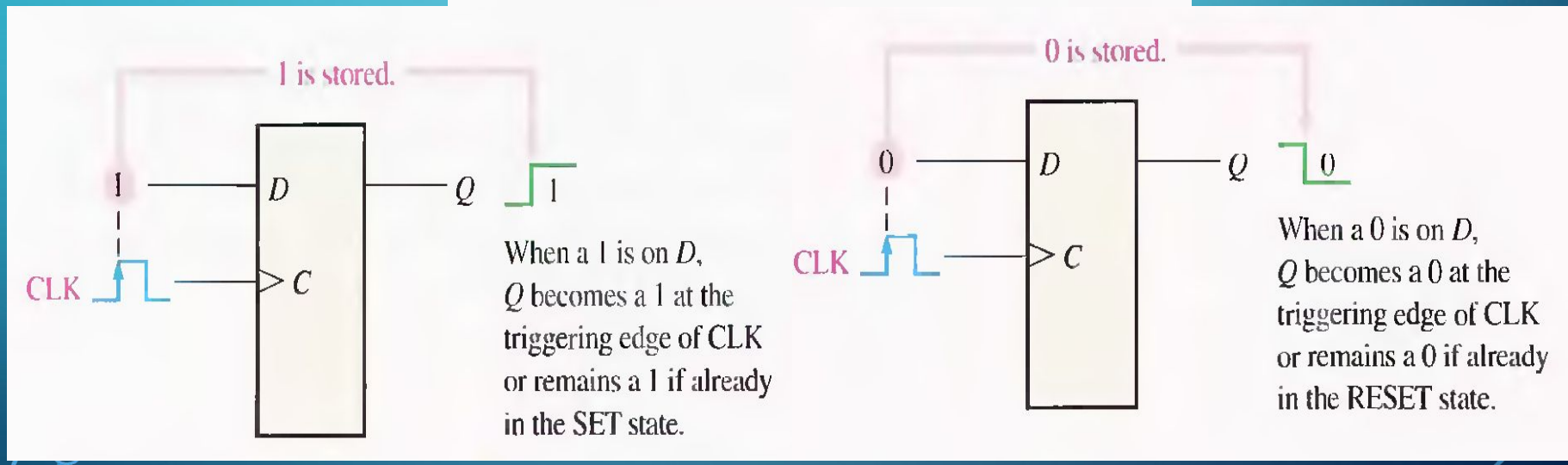
Shift Register Operations
Types of Shift Register Data I/Os
Bidirectional Shift Registers
Shift Register Counters
Shift Register Applications

BASIC SHIFT REGISTER FUNCTIONS

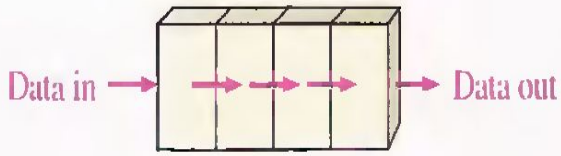
A **register** is a digital circuit with two basic functions: data storage and data movement. The storage capability of a register makes it an important type of memory device.

A register can consist of one or more flip-flops used to store and shift data.

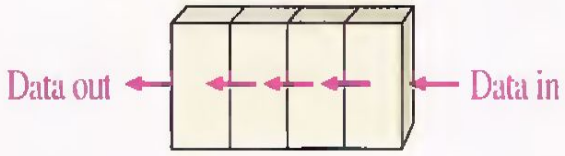
The flip-flop as a storage element.



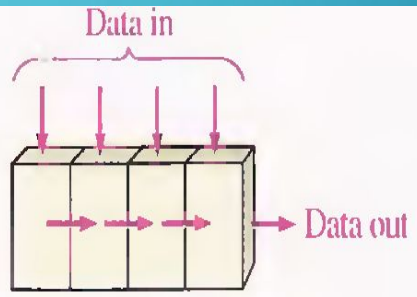
Basic data movement in shift registers. (Four bits are used for illustration. The bits move in the direction of the arrows.)



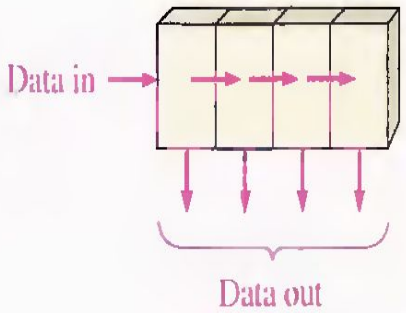
(a) Serial in/shift right/serial out



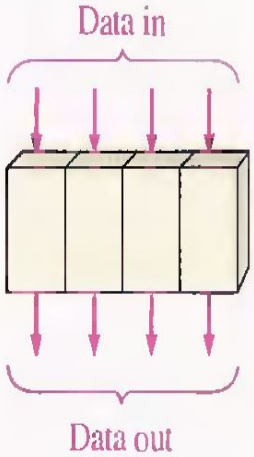
(b) Serial in/shift left/serial out



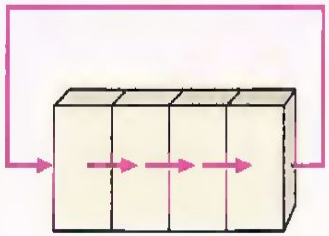
(c) Parallel in/serial out



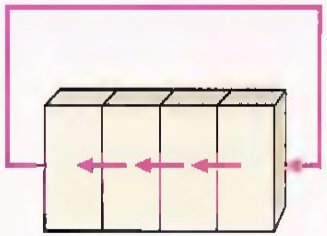
(d) Serial in/parallel out



(e) Parallel in/parallel out



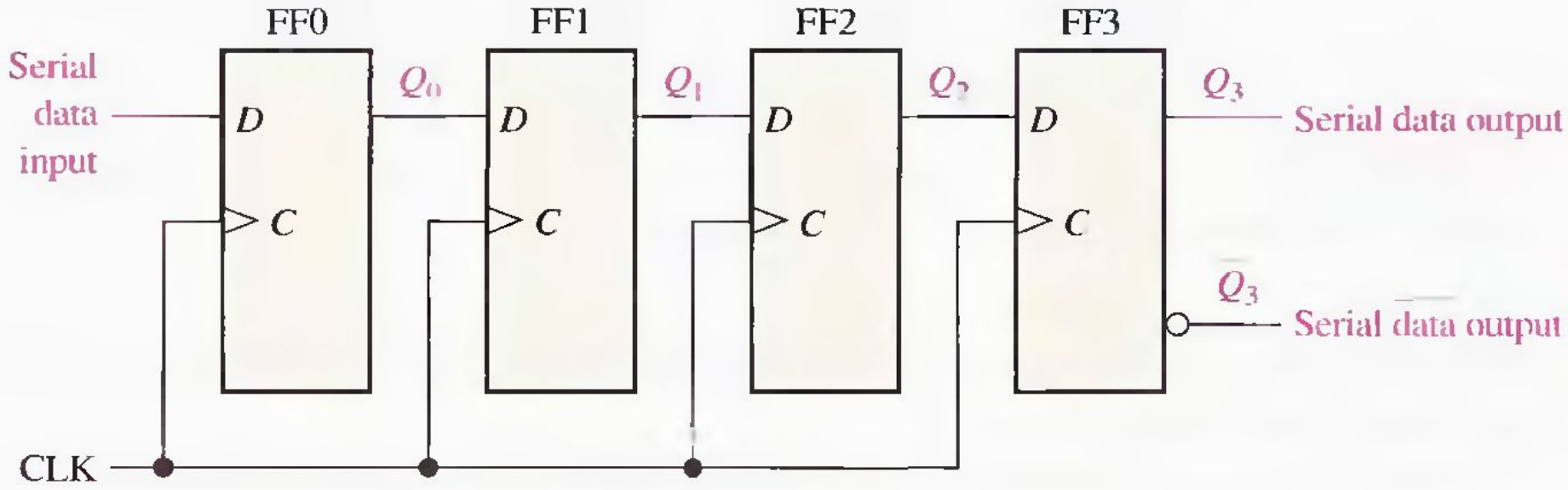
(f) Rotate right



(g) Rotate left

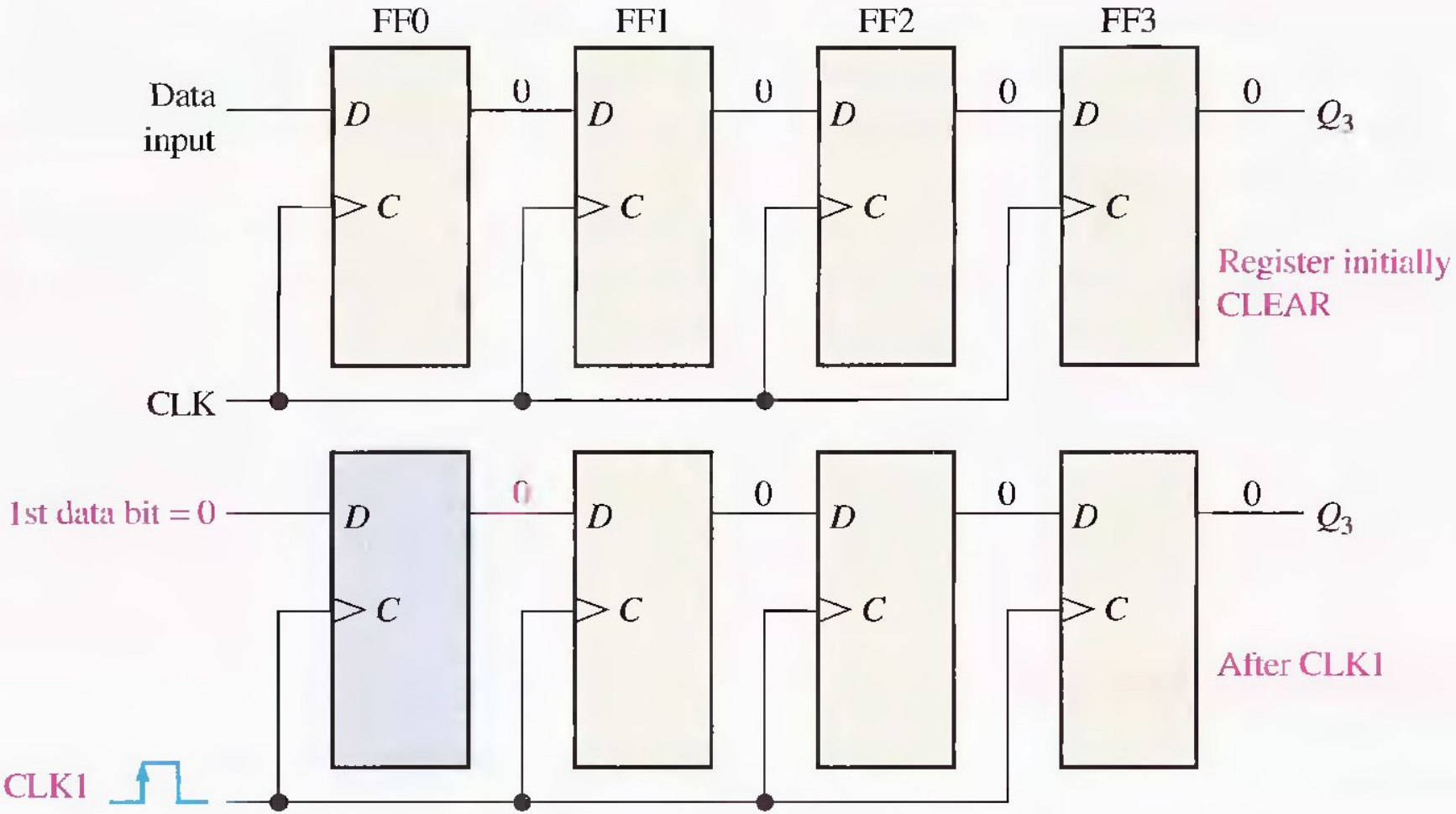
SERIAL IN/SERIAL OUT SHIFT REGISTERS

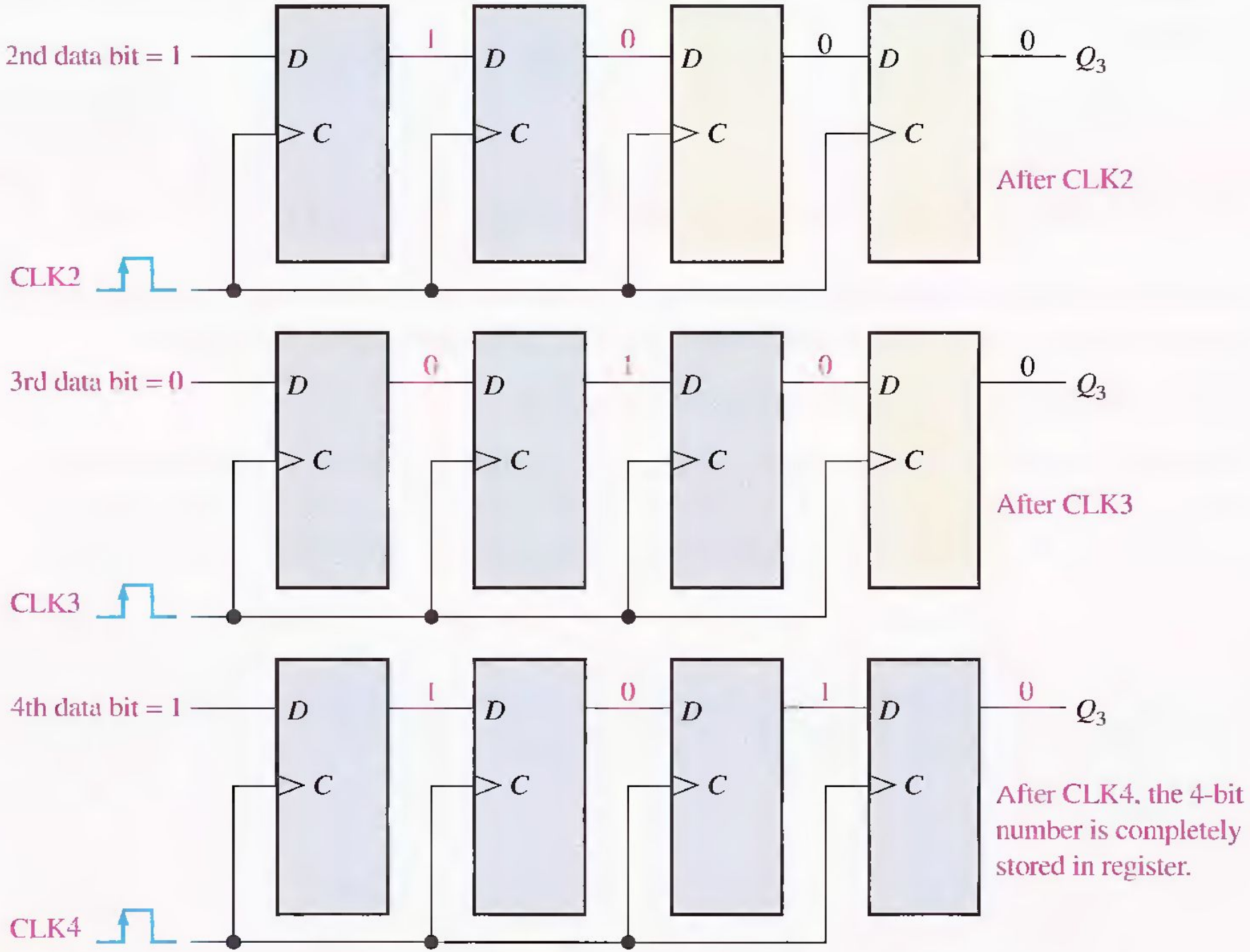
The serial in/serial out shift register accepts data serially—that is, one bit at a time on a single line. It produces the stored information on its output also in serial form.



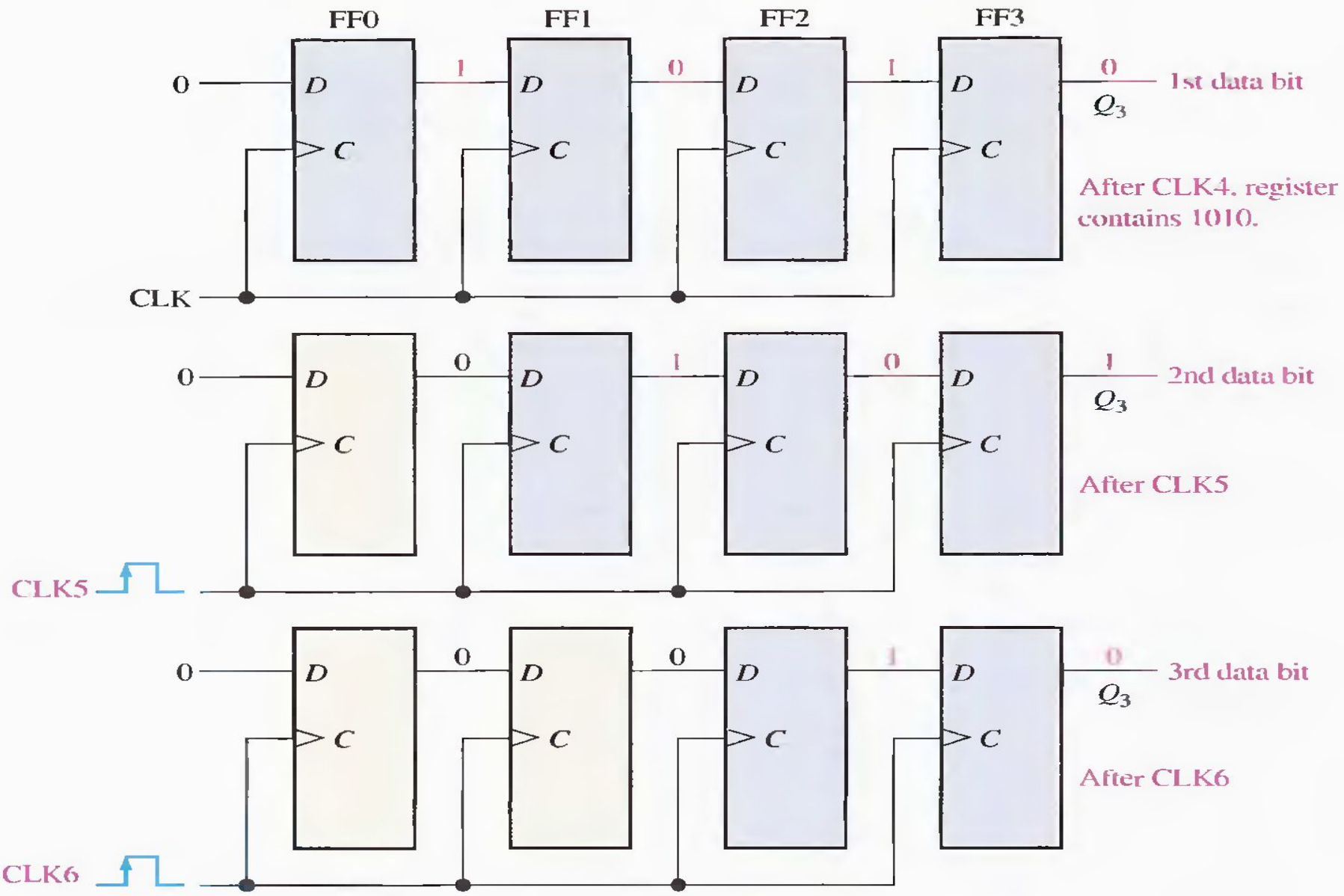
Four bits (1010) being entered serially into the register.

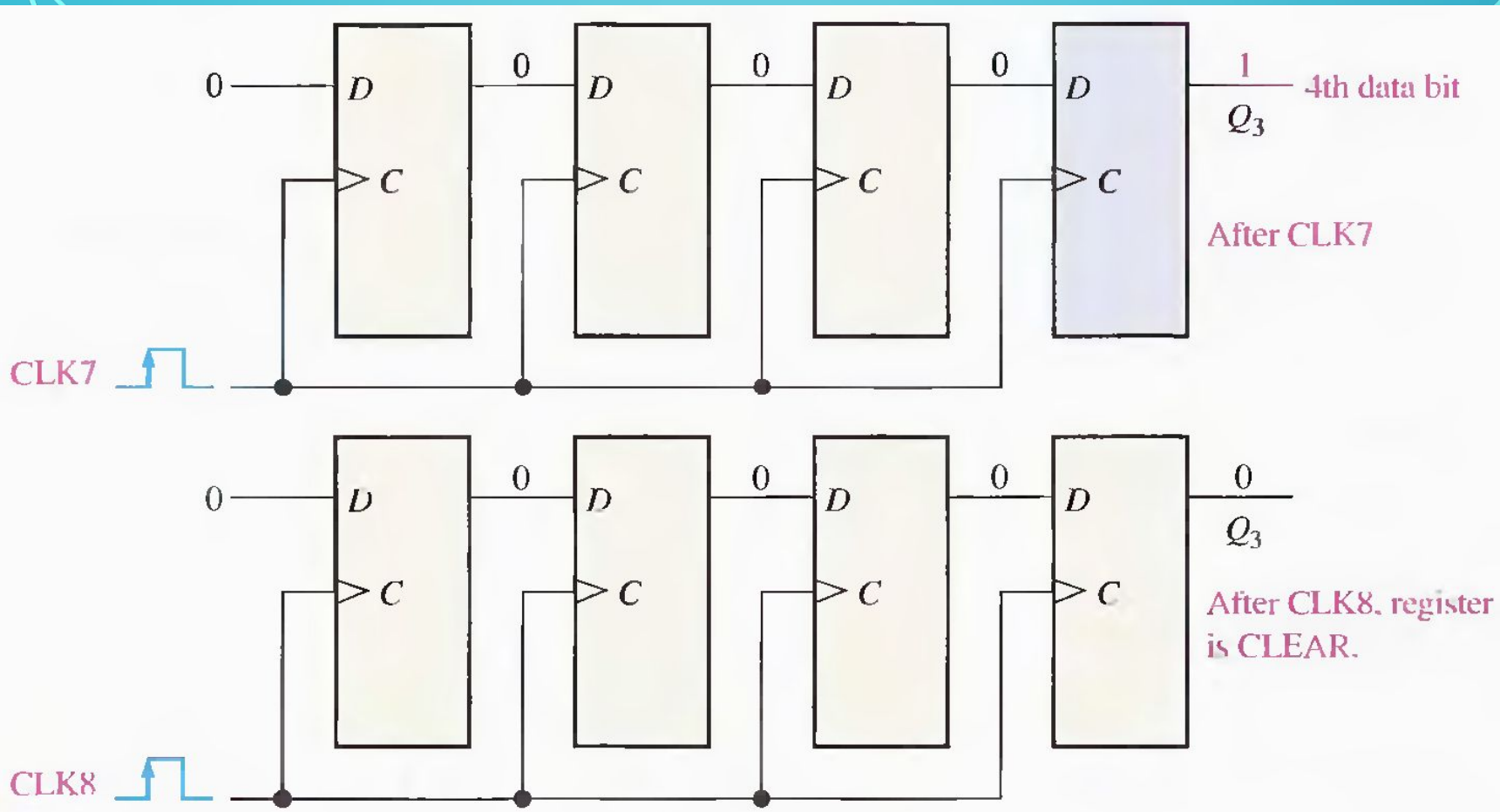
For serial data, one bit at a time is transferred.



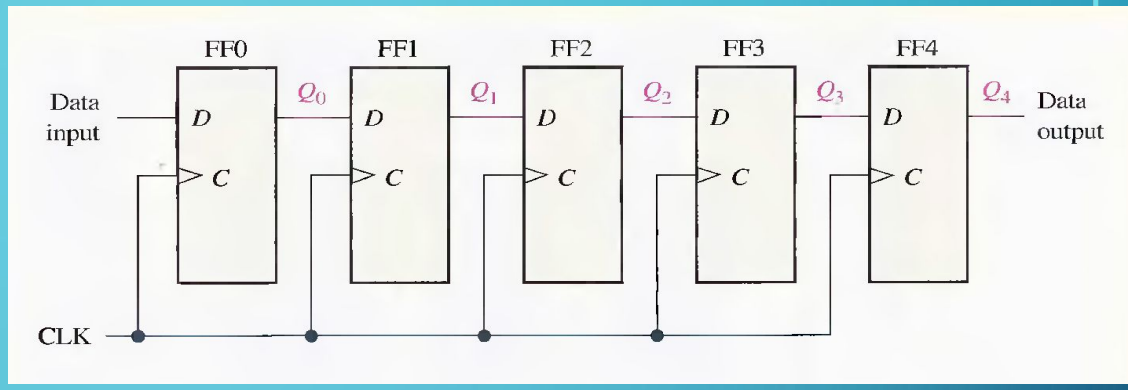

2nd data bit = 1
D
C
1
0
0
0
Q_3
After CLK2
CLK2
3rd data bit = 0
0
1
0
0
Q_3
After CLK3
CLK3
4th data bit = 1
1
0
1
0
Q_3
After CLK4, the 4-bit number is completely stored in register.
CLK4

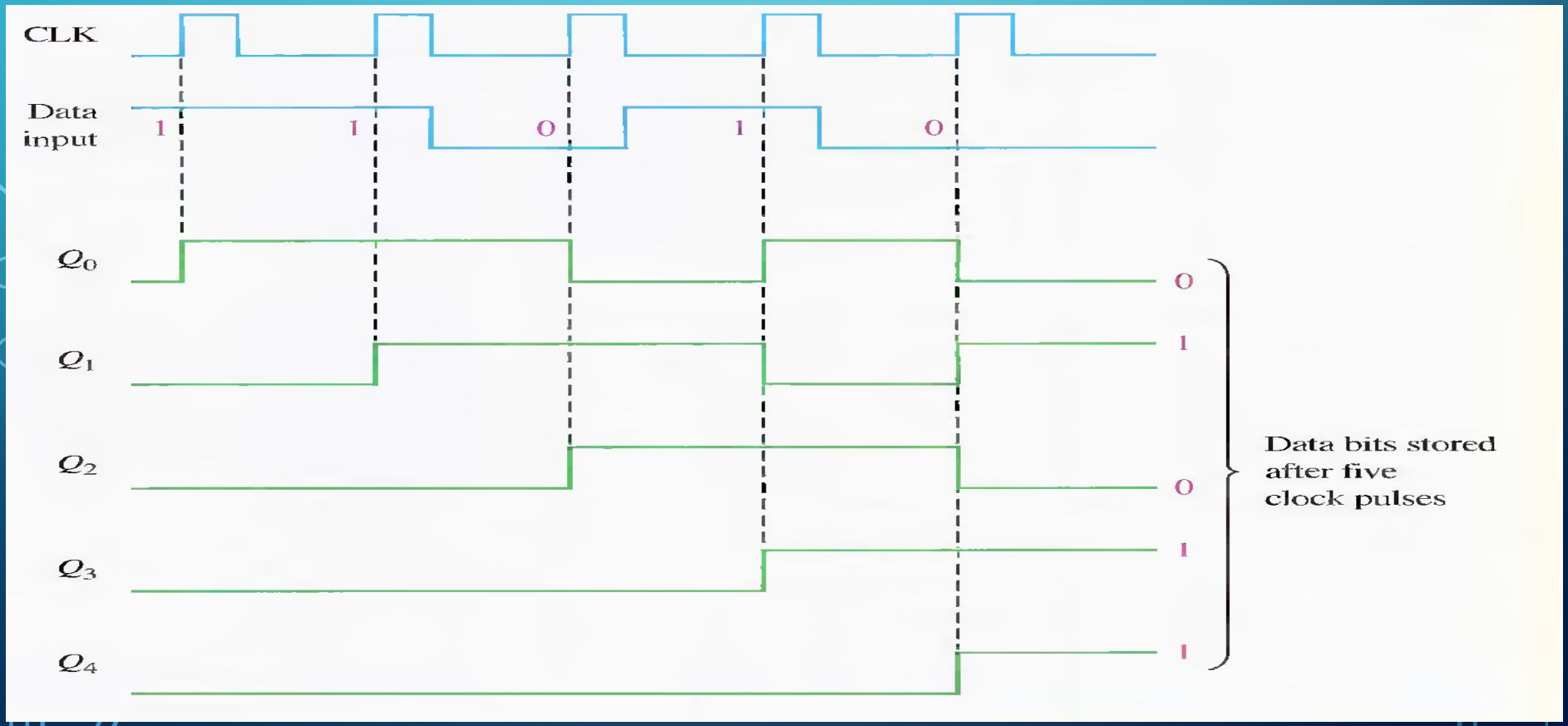
Four bits (1010) being serially shifted out of the register and replaced by all zeros.



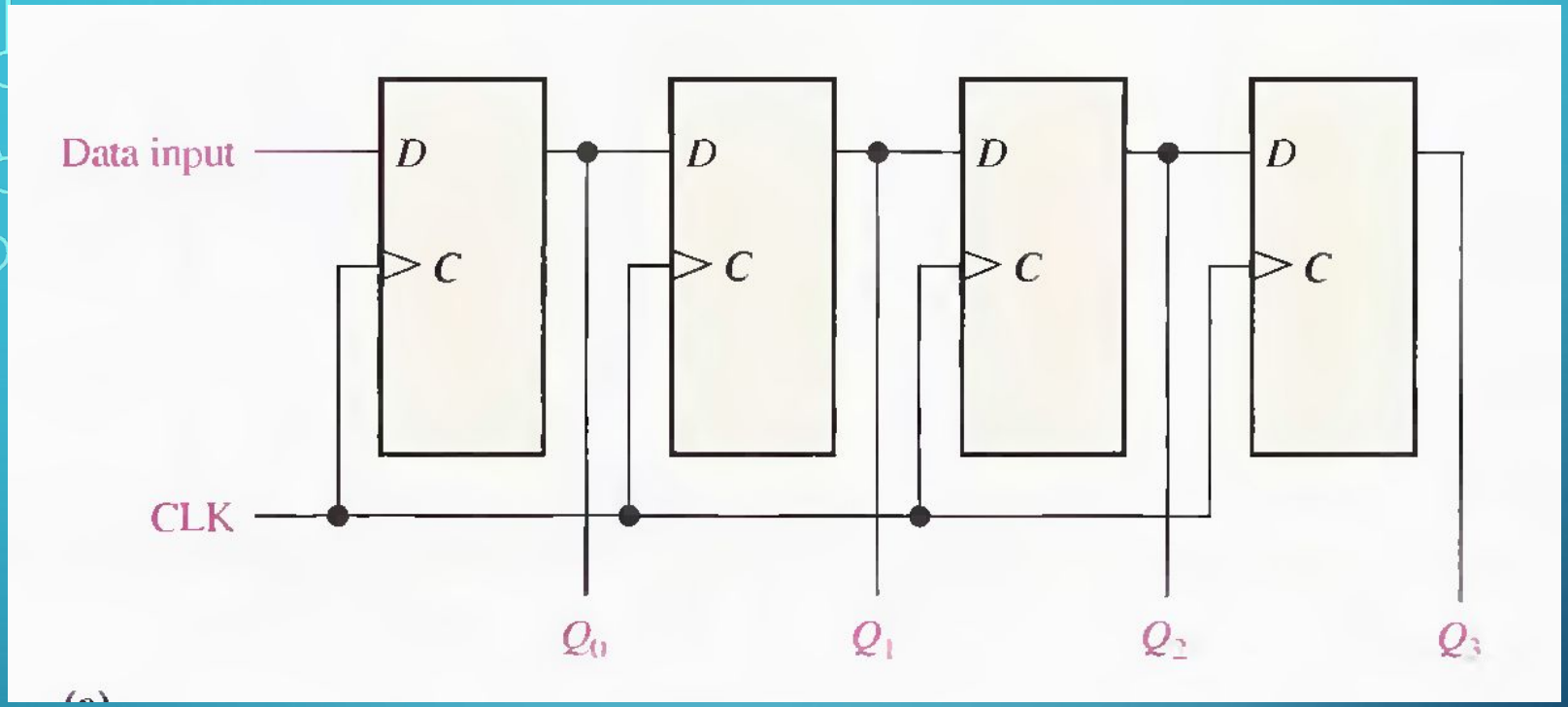

0
D
C
0
D
C
0
D
C
0
D
C
1
Q_3
4th data bit
After CLK7
CLK7
0
D
C
0
D
C
0
D
C
0
D
C
0
Q_3
After CLK8, register is CLEAR.
CLK8

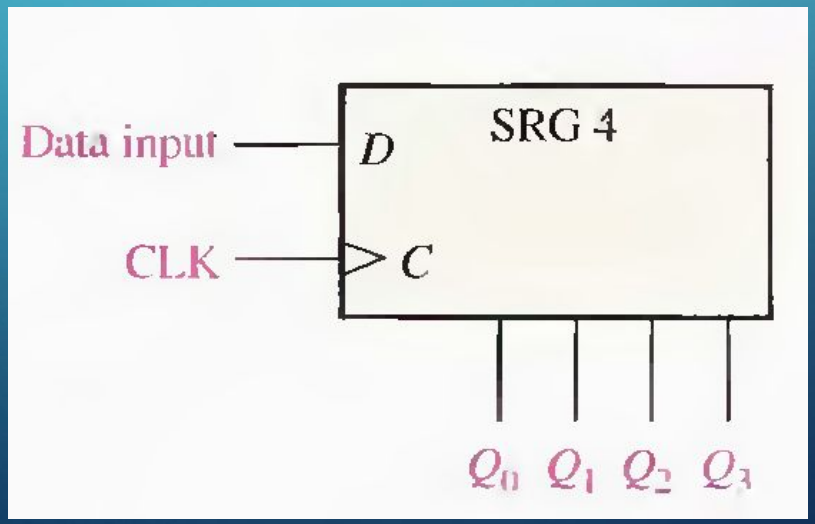
Show the states of the 5-bit register in Figure, for the specified data input and clock waveforms. Assume that the register is initially cleared (all Os).





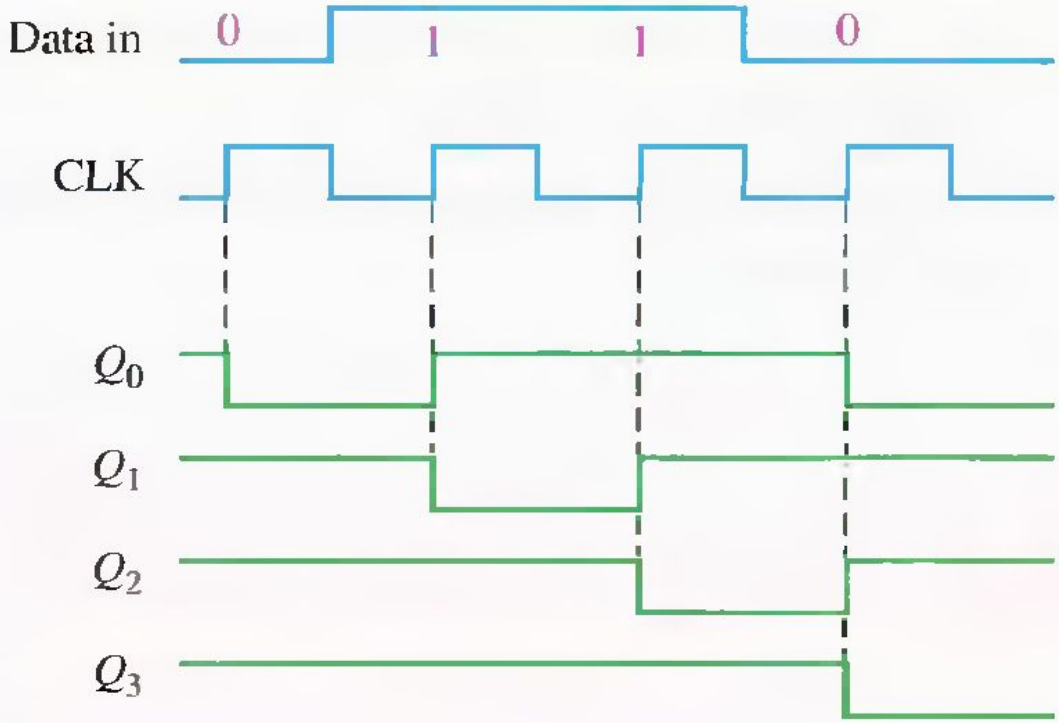
SERIAL IN/PARALLEL OUT SHIFT REGISTERS

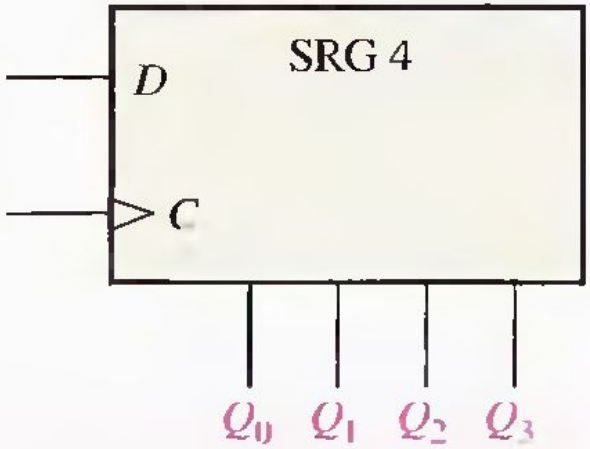




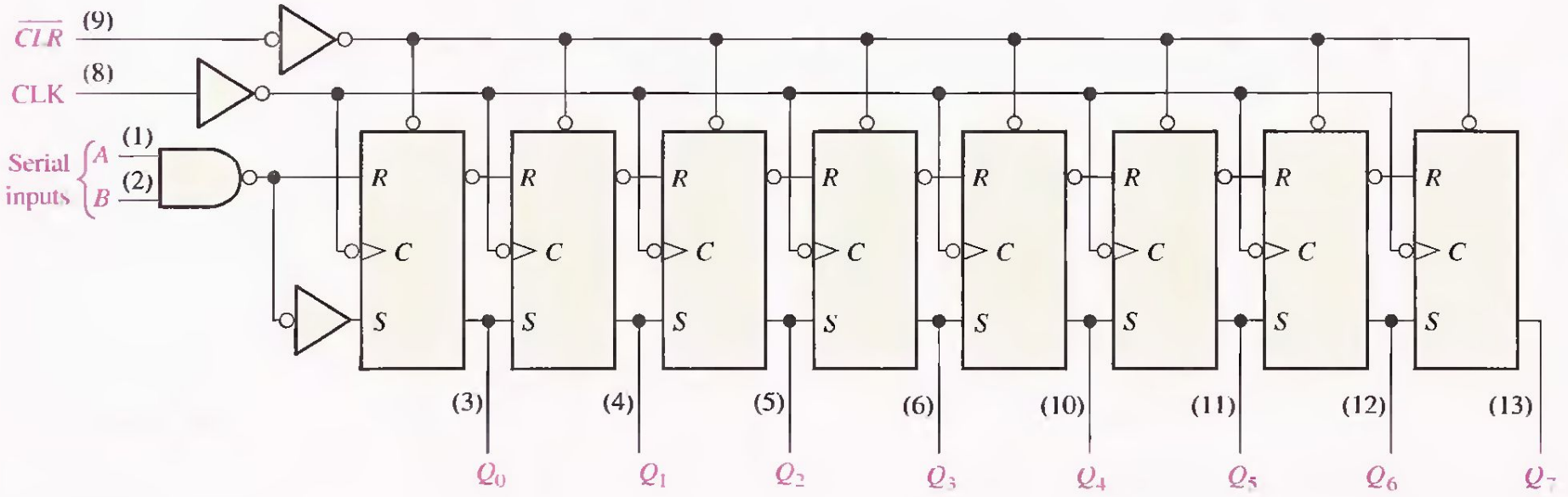
Show the states of the 4-bit register (SRG 4) for the data input and clock waveforms in Figure 9–9(a). The register initially contains all 1s.

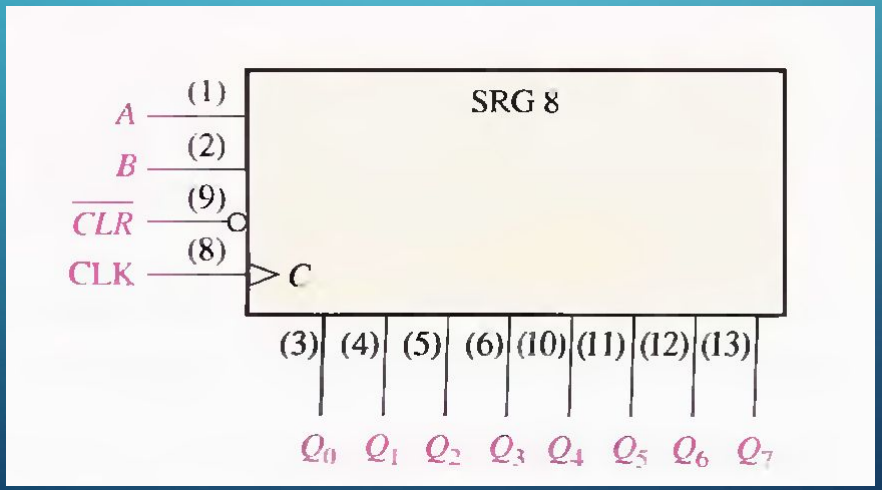
The register contains 0110 after four clock pulses. See Figure 9–9(b).

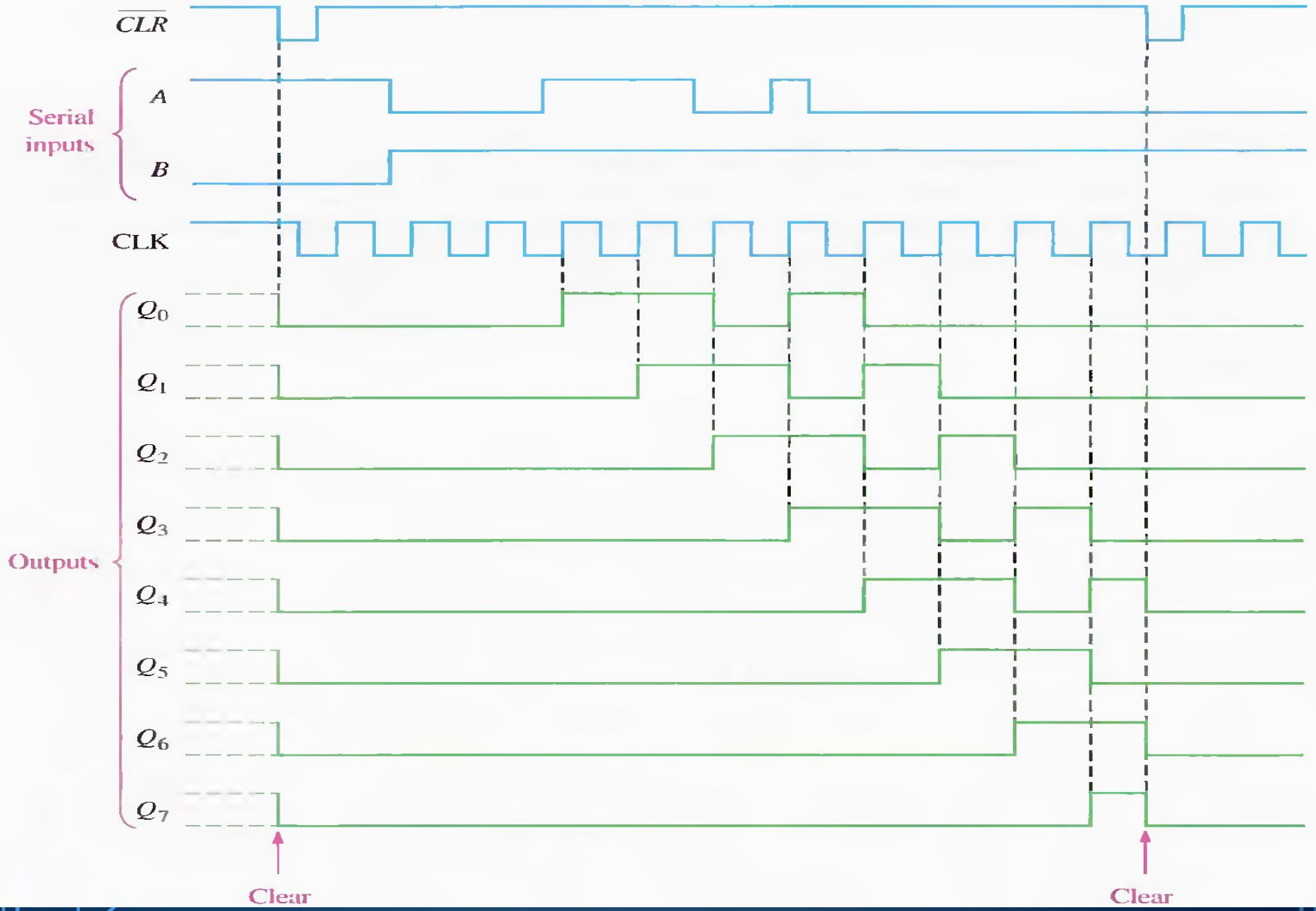




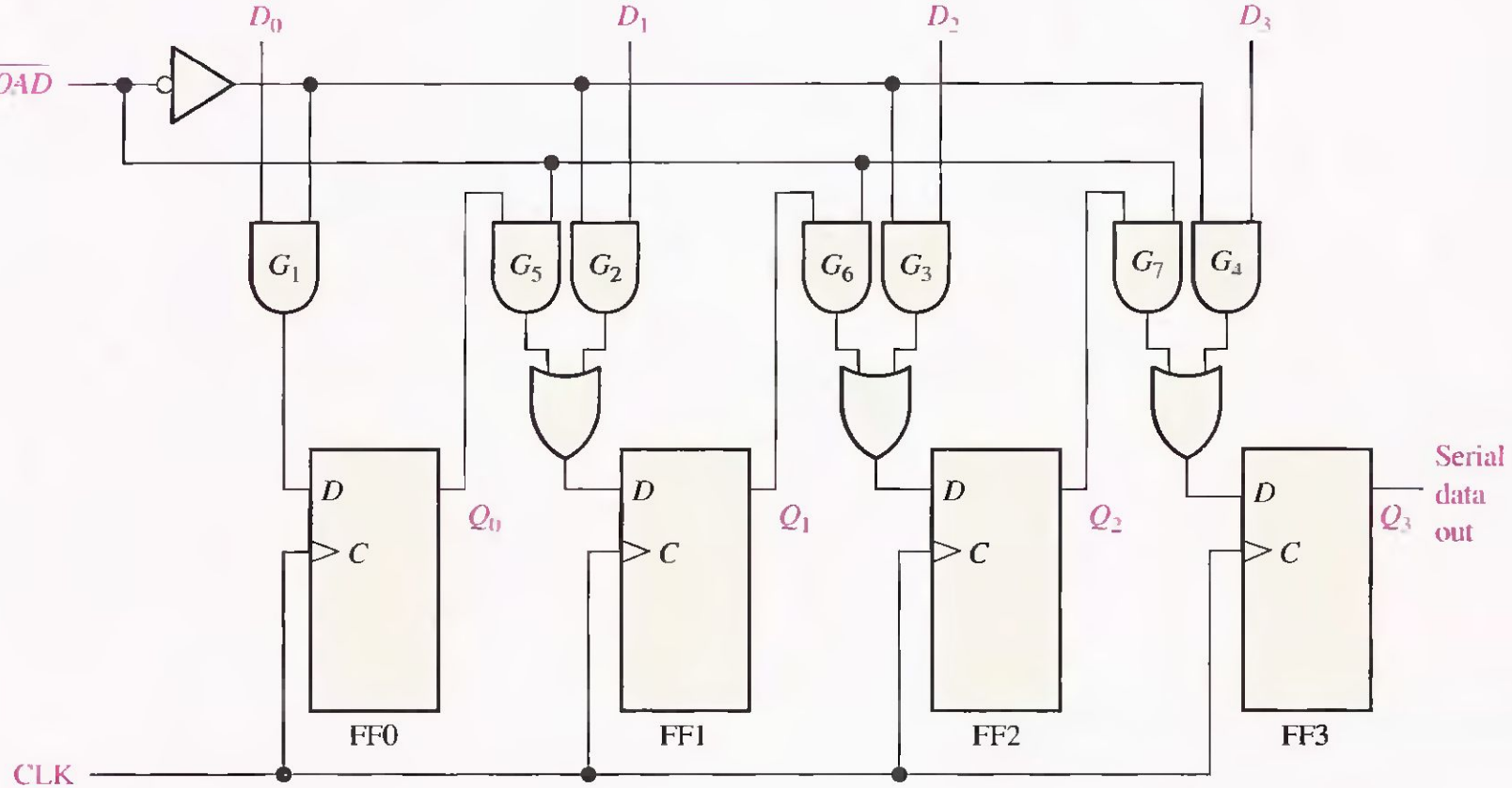
THE 74HC164 8-BIT SERIAL IN/PARALLEL OUT SHIFT REGISTER

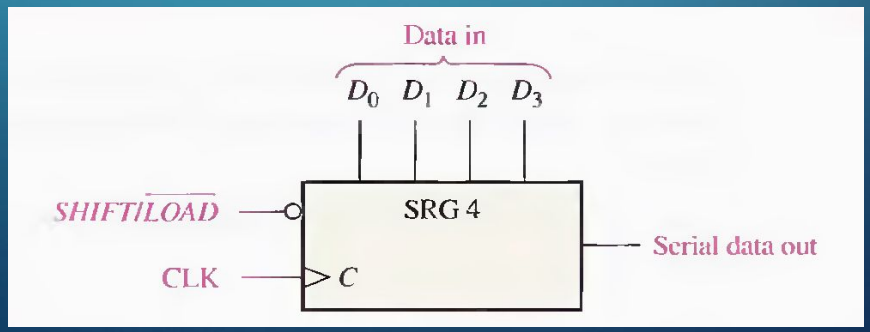




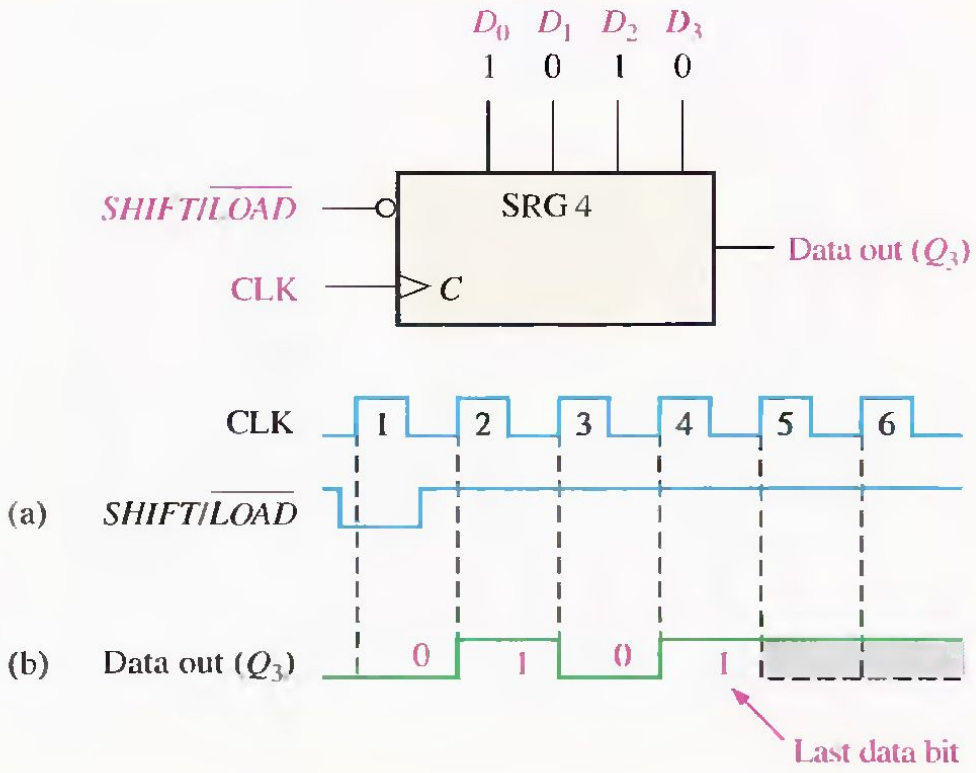

$\overline{CLR}$
Serial inputs
A
B
CLK
Outputs
Q_0
Q_1
Q_2
Q_3
Q_4
Q_5
Q_6
Q_7
Clear
Clear

PARALLEL IN/SERIAL OUT SHIFT REGISTERS



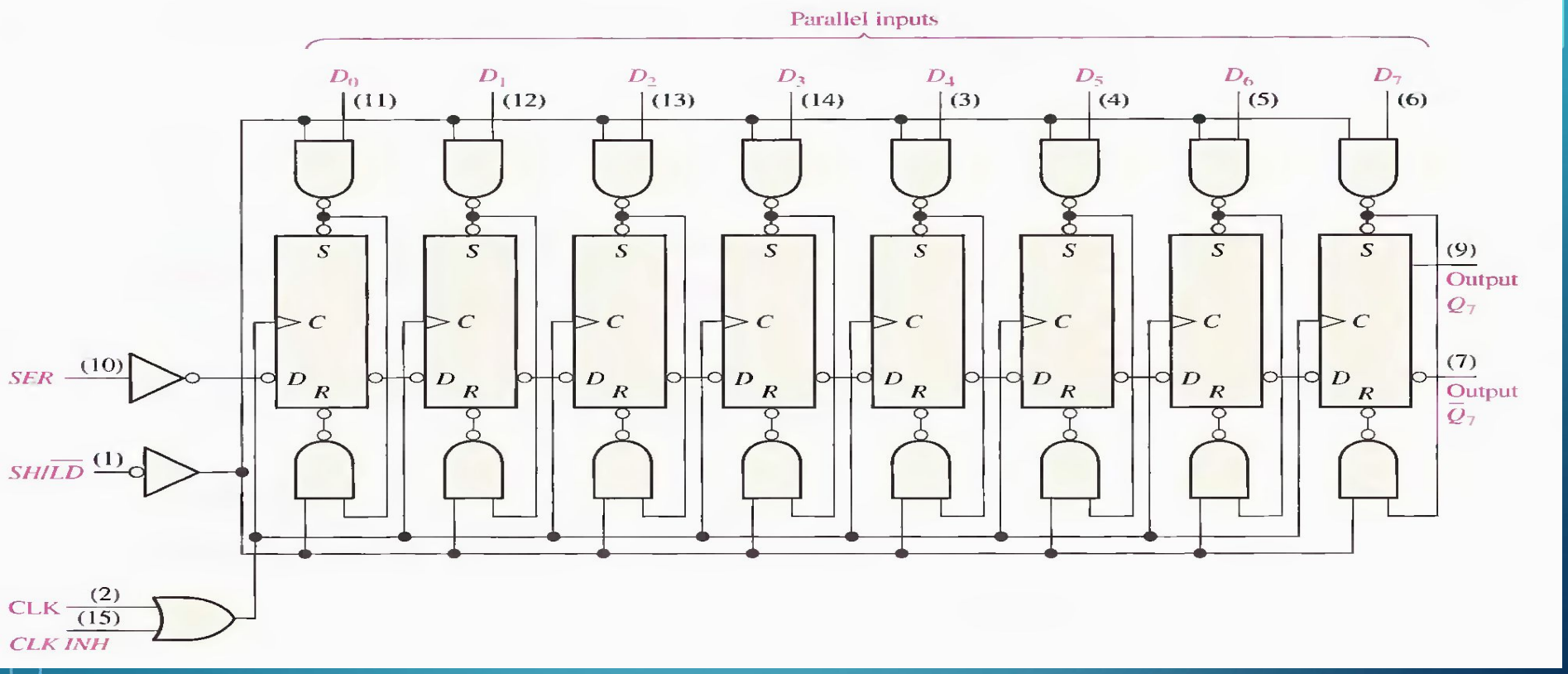


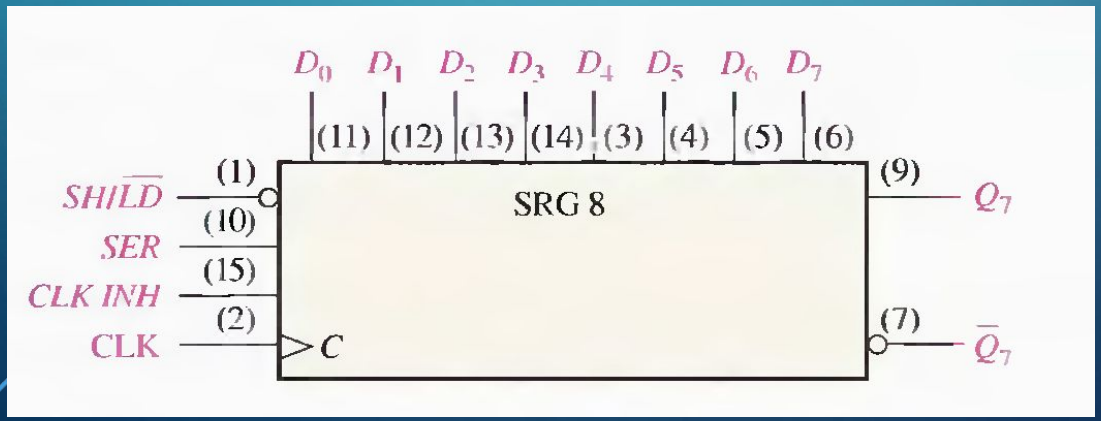
Show the data-output waveform for a 4-bit register with the parallel input data and the clock and *SHIFT*/$\overline{LOAD}$ waveforms given in Figure 9–13(a). Refer to Figure 9–12(a) for the logic diagram.

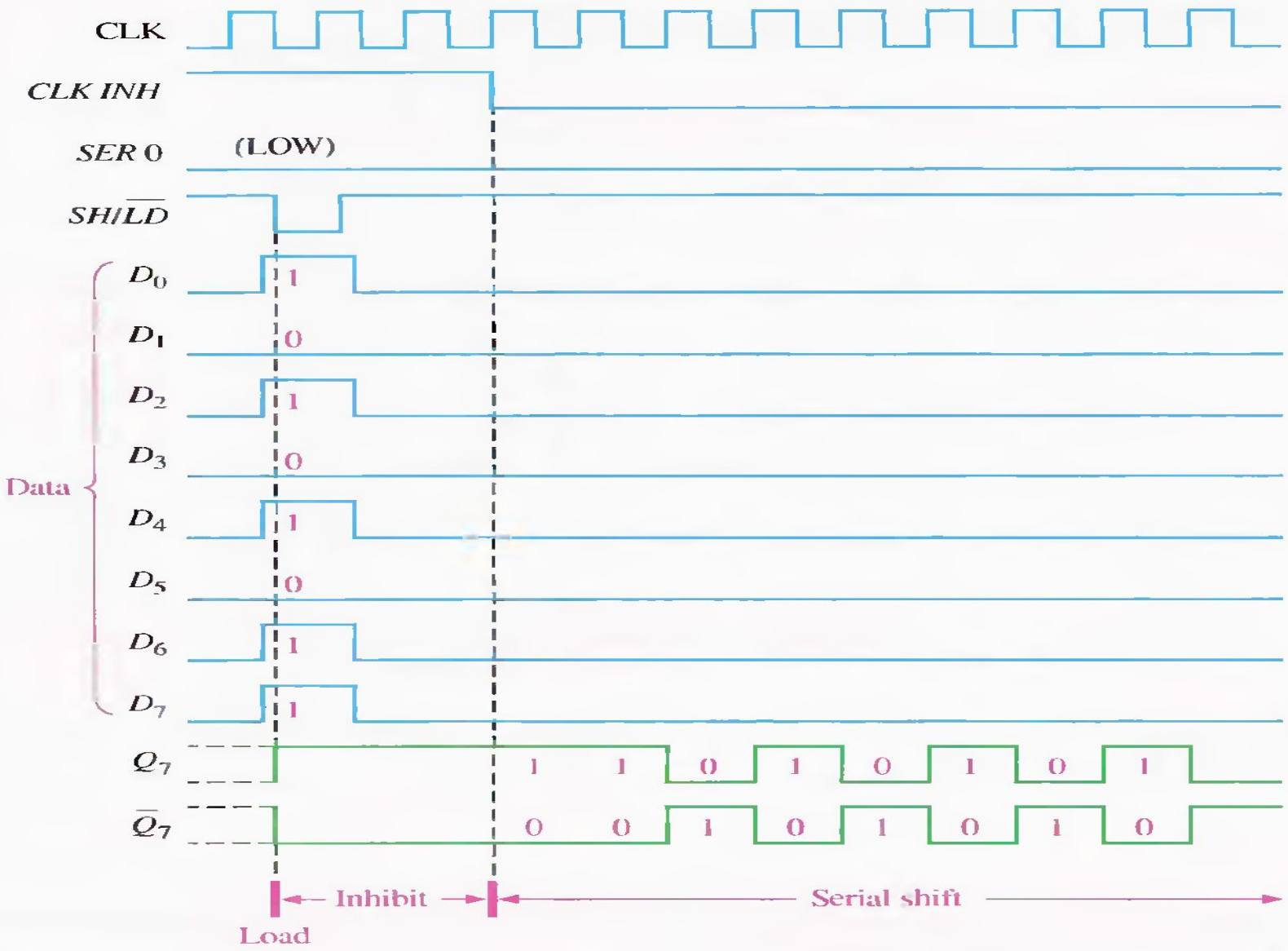


On clock pulse 1, the parallel data ($D_0D_1D_2D_3 = 1010$) are loaded into the register, making Q_3 a 0. On clock pulse 2 the 1 from Q_2 is shifted onto Q_3; on clock pulse 3 the 0 is shifted onto Q_3; on clock pulse 4 the last data bit (1) is shifted onto Q_3; and on clock pulse 5, all data bits have been shifted out, and only 1s remain in the register (assuming the *D* input remains a 1). See Figure 9–13(b).

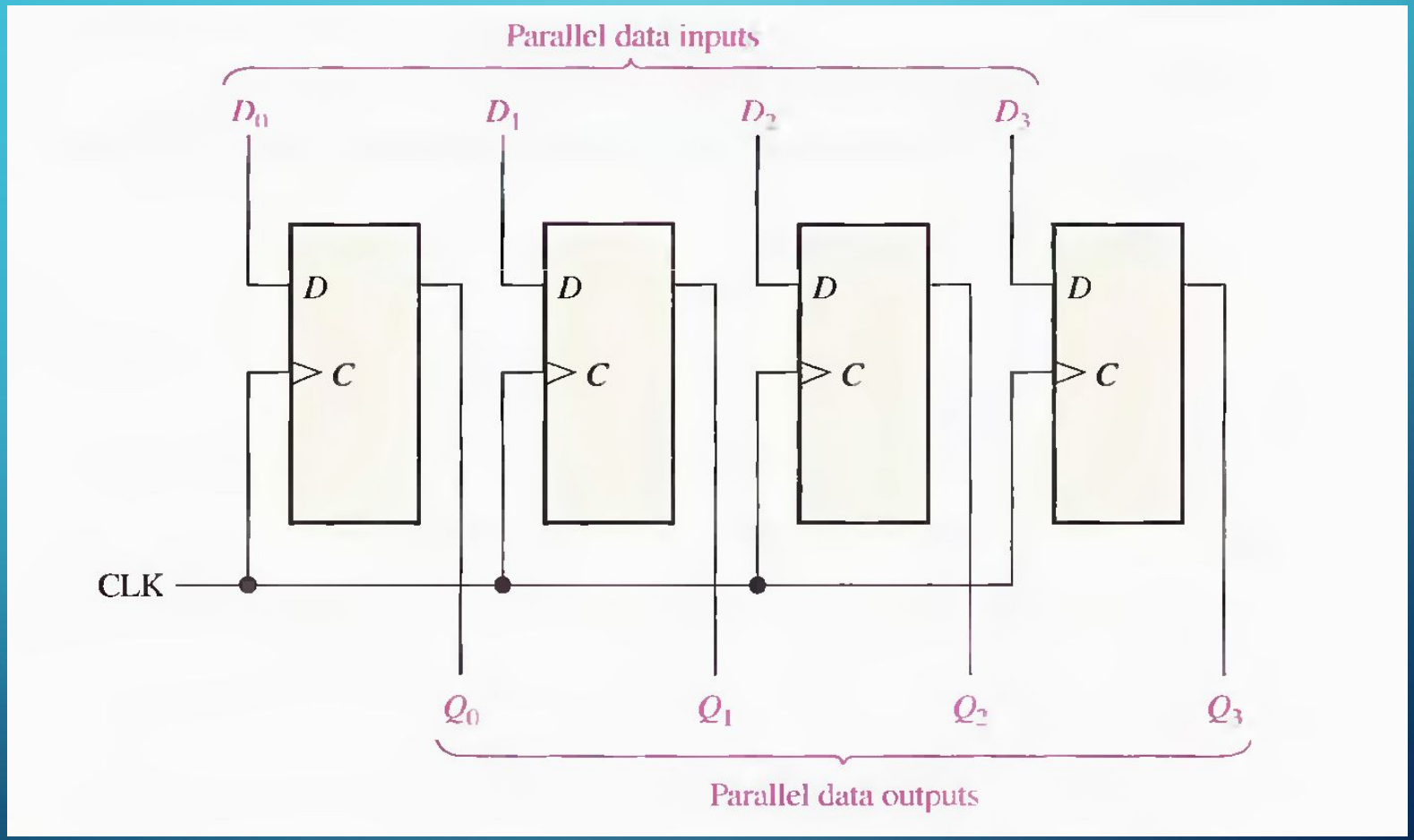
THE 74HC165 8-BIT PARALLEL LOAD SHIFT REGISTER






CLK
CLK INH
SER 0
(LOW)
SH/$\overline{LD}$
D_0
D_1
D_2
D_3
D_4
D_5
D_6
D_7
Data
1
0
1
0
1
0
1
1
Q_7
1 1 0 1 0 1 0 1
$\overline{Q}_7$
0 0 1 0 1 0 1 0
Inhibit
Serial shift
Load

PARALLEL IN/PARALLEL OUT SHIFT REGISTERS



Bidirectional Shift Register

Bidirectional shift registers can shift the data in either direction using a *RIGHT/$\overline{LEFT}$* input.

The logic analyzer simulation shows a bidirectional shift register such as the one shown in Figure 9-19 of the text. Notice the HIGH level from the Serial data in is shifted at first from Q_3 toward Q_0.

Bidirectional Shift Register

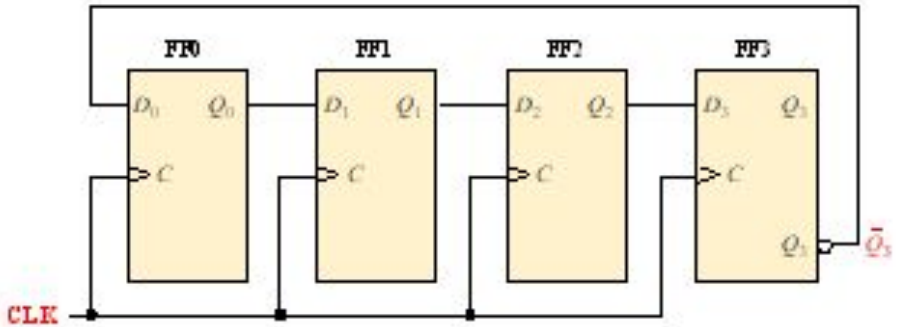
Question How will the pattern change if the *RIGHT/$\overline{LEFT}$* control signal is inverted?

Answer See display

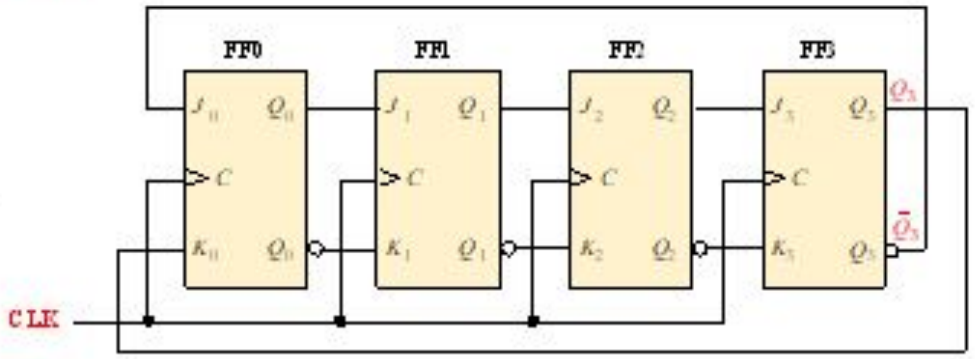
Shift Register Counters

Shift registers can form useful counters by recirculating a pattern of 0's and 1's. Two important shift register counters are the *Johnson counter* and the *ring counter*.

The Johnson counter can be made with a series of D flip-flops

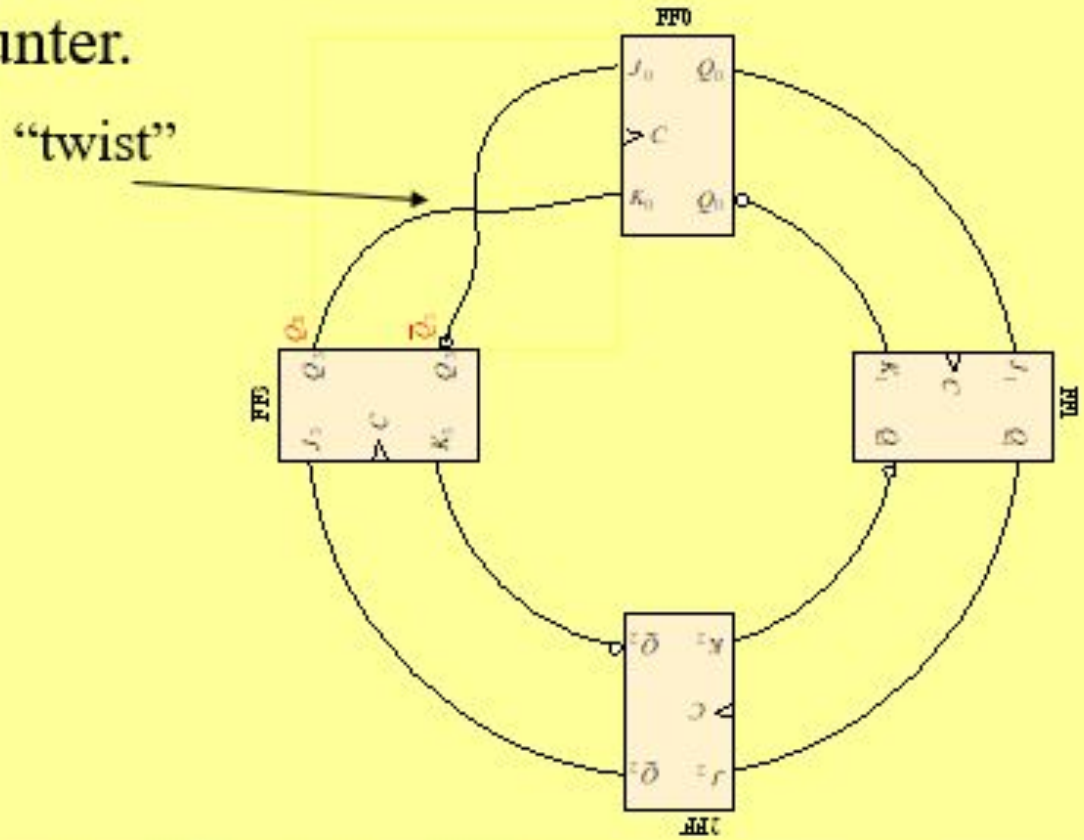


… or with a series of J-K flip flops. Here Q_3 and $\overline{Q}_3$ are fed back to the J and K inputs with a "twist".



Johnson Counter

Redrawing the same Johnson counter (without the clock shown) illustrates why it is sometimes called as a "twisted-ring" counter.



Johnson Counter

The Johnson counter is useful when you need a sequence that changes by only one bit at a time but it has a limited number of states ($2n$, where n = number of stages).

The first five counts for a 4-bit Johnson counter that is initially cleared are:

CLK	Q_0	Q_1	Q_2	Q_3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1

Question

What are the remaining 3 states?

TABLE 8–3

Four-bit Johnson sequence.

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

TABLE 8–4

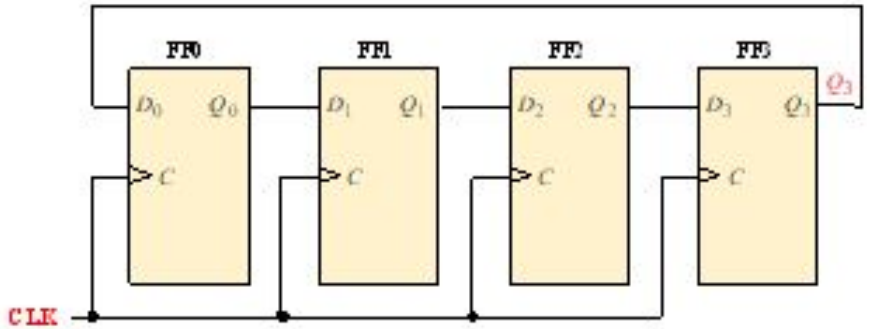
Five-bit Johnson sequence.

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

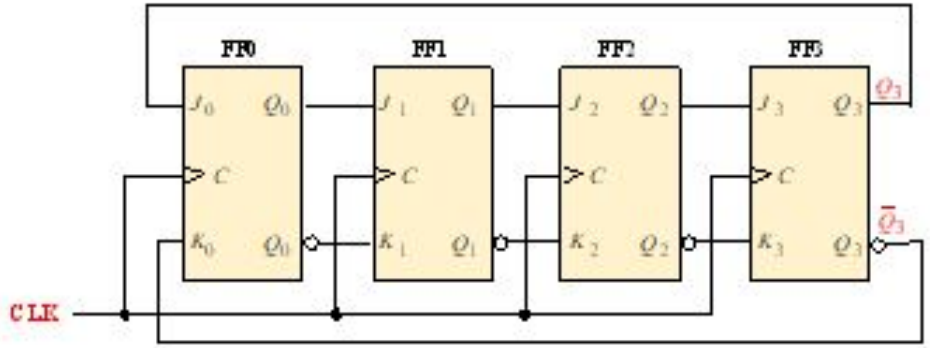
Ring Counter

The ring counter can also be implemented with either D flip-flops or J-K flip-flops.

Here is a 4-bit ring counter constructed from a series of D flip-flops. Notice the feedback.



Like the Johnson counter, it can also be implemented with J-K flip flops.

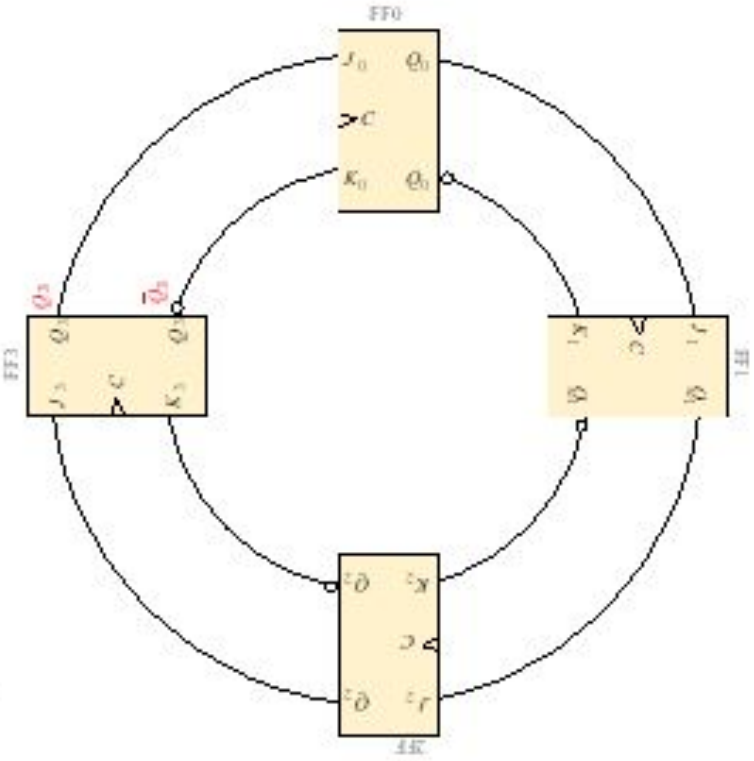


Ring Counter

Redrawing the Ring counter (without the clock shown) shows why it is a "ring".

The disadvantage to this counter is that it must be preloaded with the desired pattern (usually a single 0 or 1) and it has even fewer states than a Johnson counter (n, where n = number of flip-flops.

On the other hand, it has the advantage of being self-decoding with a unique output for each state.



Ring Counter

CLK

Q_0

Q_1

Q_2

Q_3

Q_4

Q_5

Q_6

Q_7