

# EIE 2050 Digital Logic and Systems

## Chapter 8 : Shift Registers

Simon Pun, Ph.D.



# Last Week

## □ Latches

- ◆ Temp. storage devices with two stable states : SET (Q HIGH) & RESET :  $\bar{Q}$  HIGH
- ◆ Gated latches : Gated S-R latches, Gated D latches

## □ Flip-Flops

- ◆ Bistable devices whose output synchronously changes at the rising/positive or falling/negative edge of the CLK : D Flip-Flop and J-K Flip-Flop
- ◆ Asynchronous Preset and Clear Inputs; Applications: Counting

## □ Flip-Flop Operating Characteristics

- ◆ Prop. Delay/Set-up/Hold times, Max. clock freq., Pulse widths & Power dissipation

## □ One-Shots (555 Timers)

- ◆ Devices with only one stable state : when triggered, remains in its unstable state for a predetermined length of time before automatically returning to its stable state.

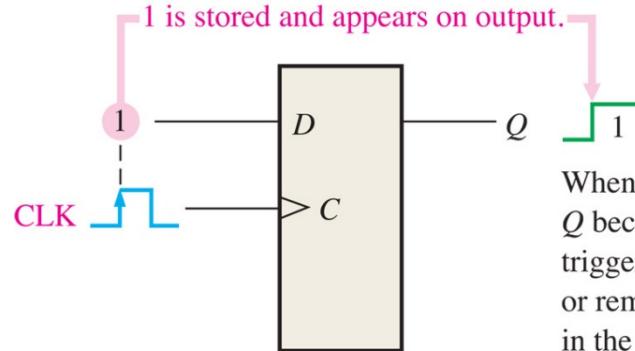
## □ The Astable Multivibrator

- ◆ Oscillates % two unstable states without any external triggering.

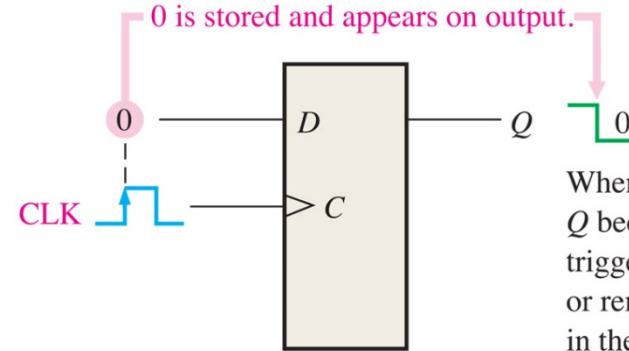


# Two Basic Functions of Shift Registers

## □ Data storage



When a 1 is on *D*, *Q* becomes a 1 at the triggering edge of CLK or remains a 1 if already in the SET state.



When a 0 is on *D*, *Q* becomes a 0 at the triggering edge of CLK or remains a 0 if already in the RESET state.

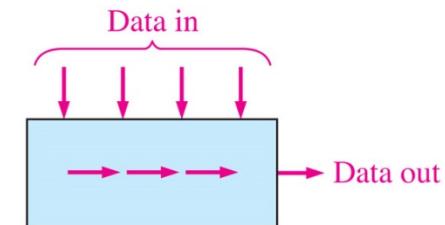
## □ Data movement



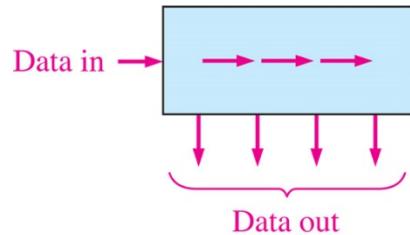
(a) Serial in/shift right/serial out



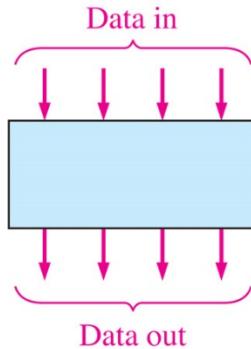
(b) Serial in/shift left/serial out



(c) Parallel in/serial out



(d) Serial in/parallel out



(e) Parallel in/parallel out



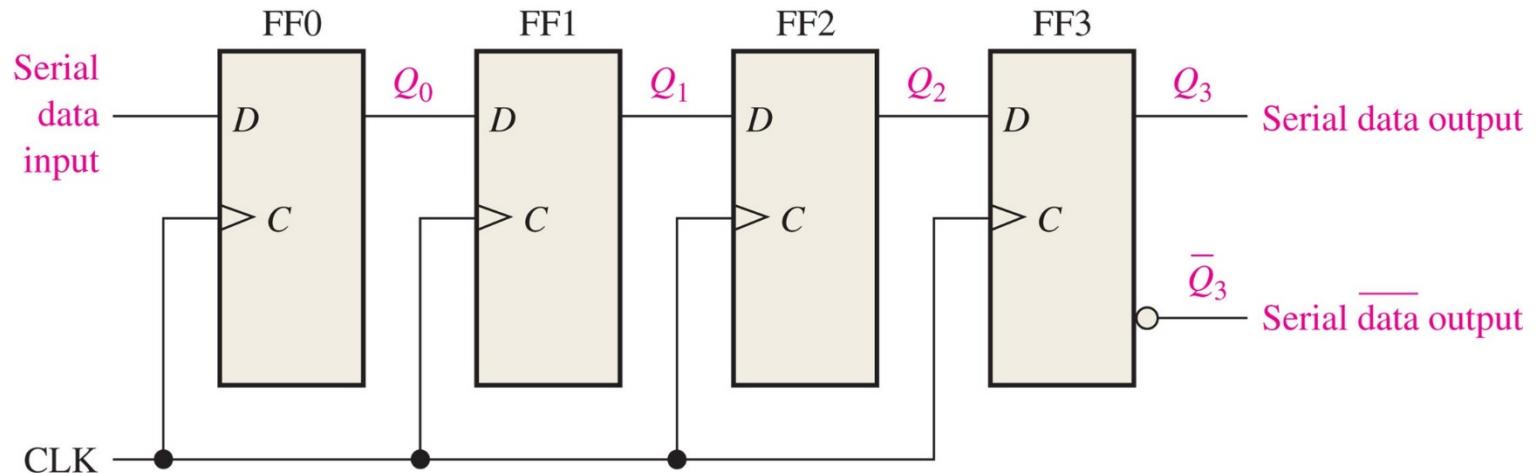
(f) Rotate right



(g) Rotate left



# Serial In/**Serial** Out Shift Registers



**TABLE 8-1**

Shifting a 4-bit code into the shift register in Figure 8–3. Data bits are indicated by a beige screen.

CLK	FF0 ( $Q_0$ )	FF1 ( $Q_1$ )	FF2 ( $Q_2$ )	FF3 ( $Q_3$ )
Initial	0	0	0	0
1	0	0	0	0
2	1	0	0	0
3	0	1	0	0
4	1	0	1	0

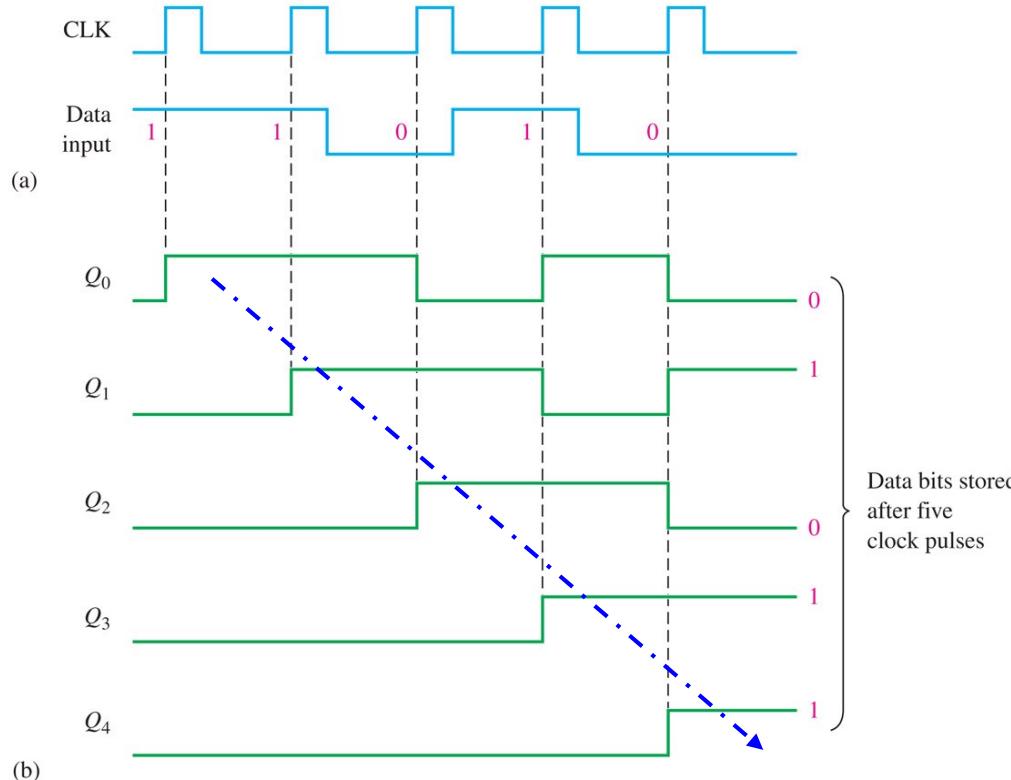
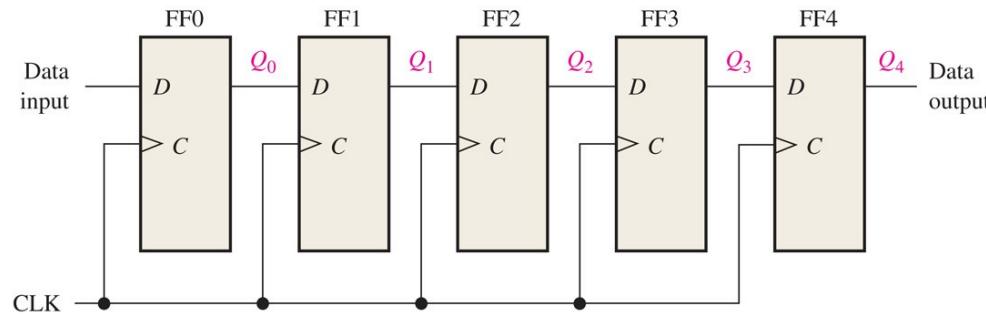
**TABLE 8-2**

Shifting a 4-bit code out of the shift register in Figure 8–3. Data bits are indicated by a beige screen.

CLK	FF0 ( $Q_0$ )	FF1 ( $Q_1$ )	FF2 ( $Q_2$ )	FF3 ( $Q_3$ )
Initial	1	0	1	0
5	0	1	0	1
6	0	0	1	0
7	0	0	0	1
8	0	0	0	0



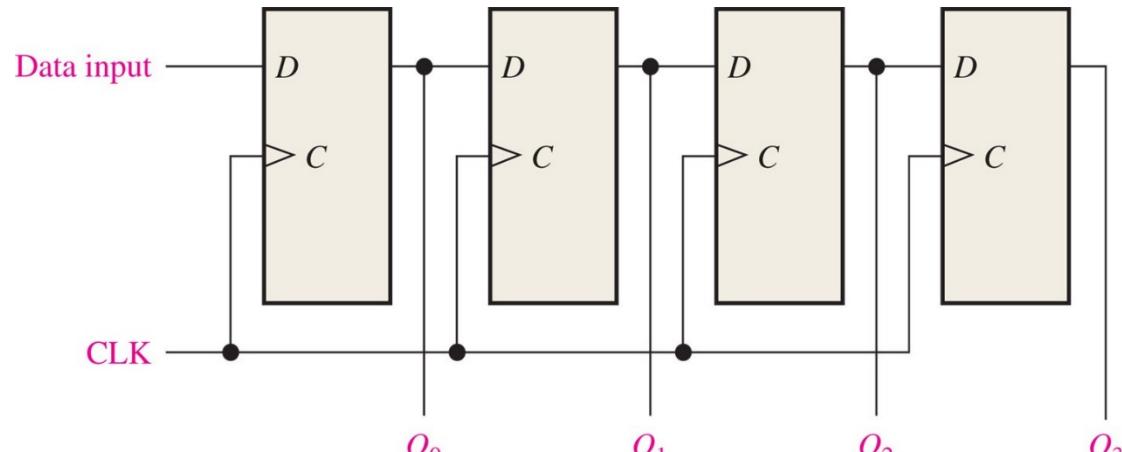
# Example 8-1



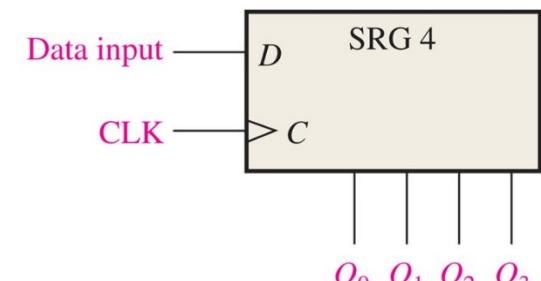
- The first data bit (1) is entered into the register on the first clock pulse and then shifted from left to right as the remaining bits are entered and shifted.
- The register contains  $Q_4Q_3Q_2Q_1Q_0 = 11010$  after five clock pulses.



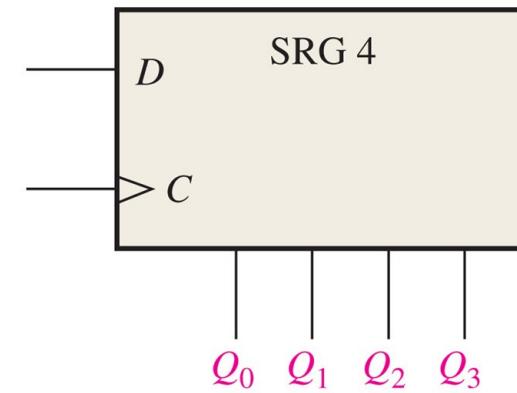
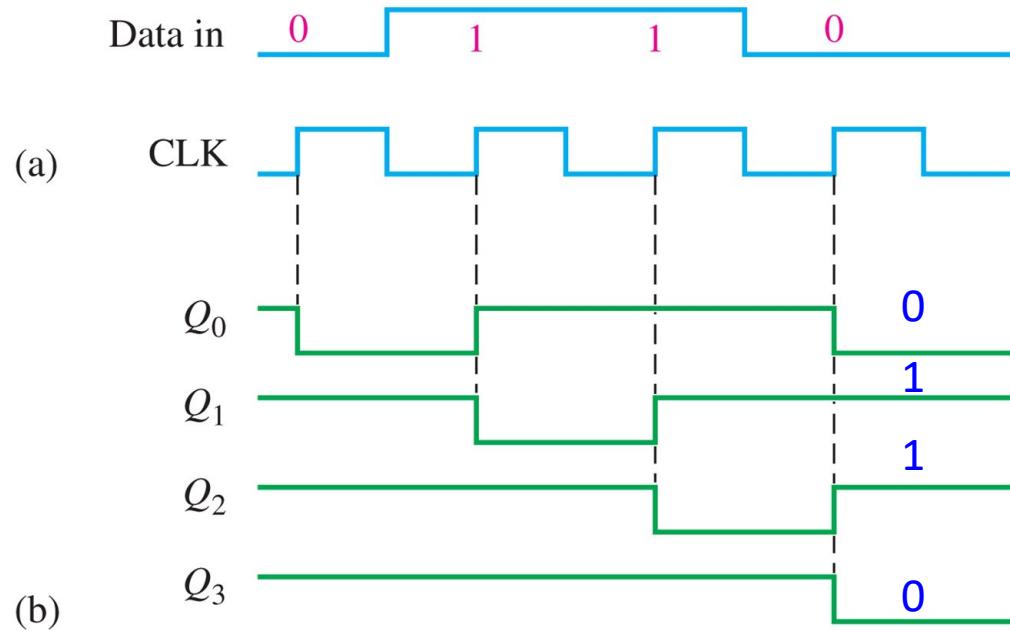
# Serial In/**Parallel** Out Shift Registers



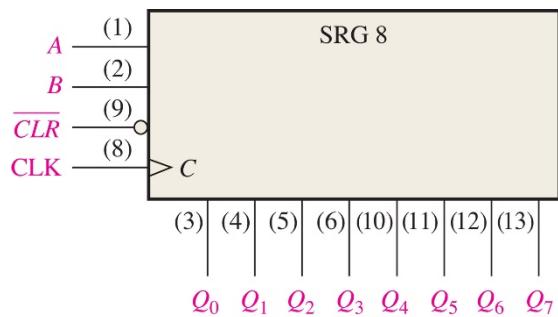
(a)



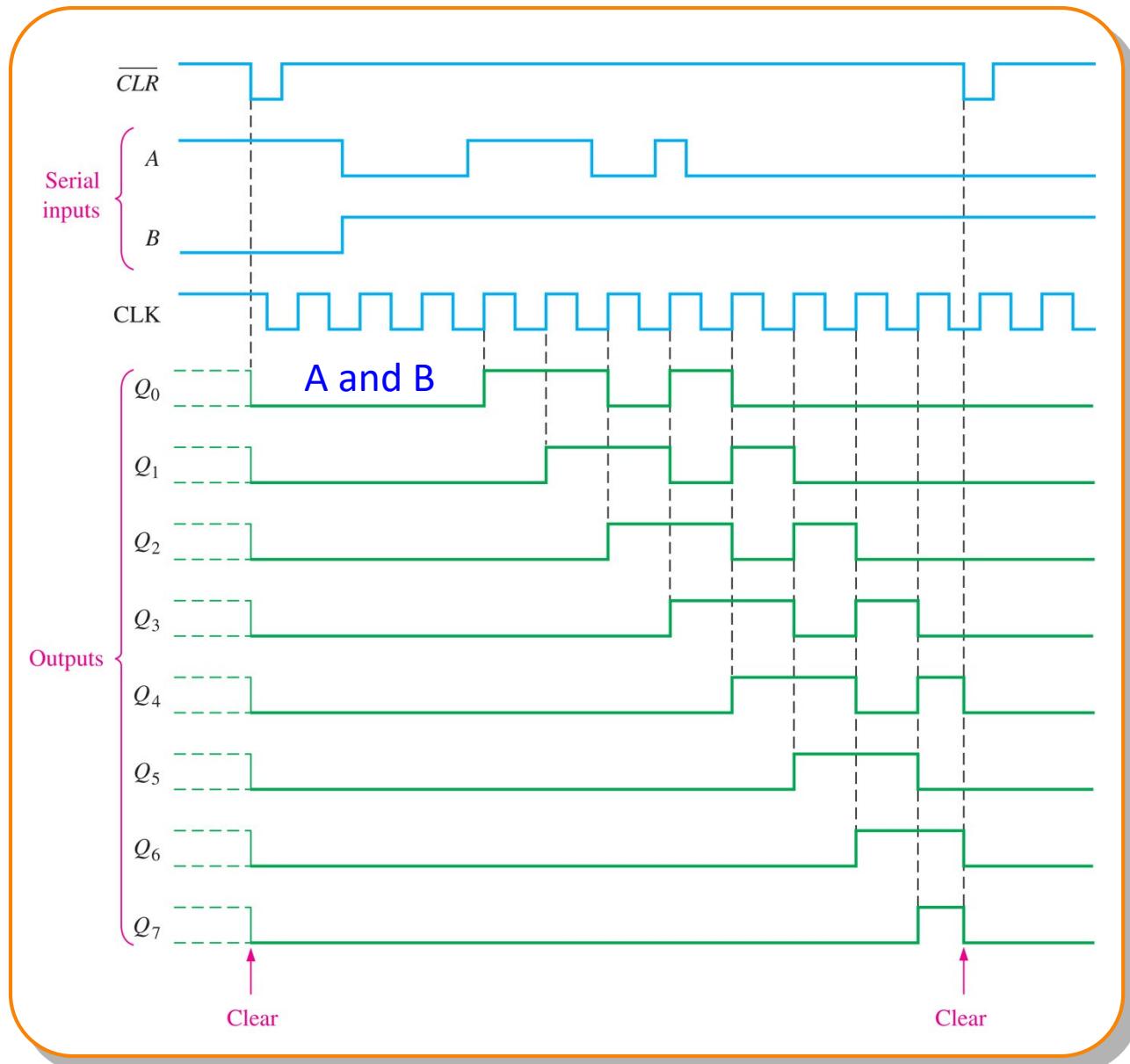
(b)



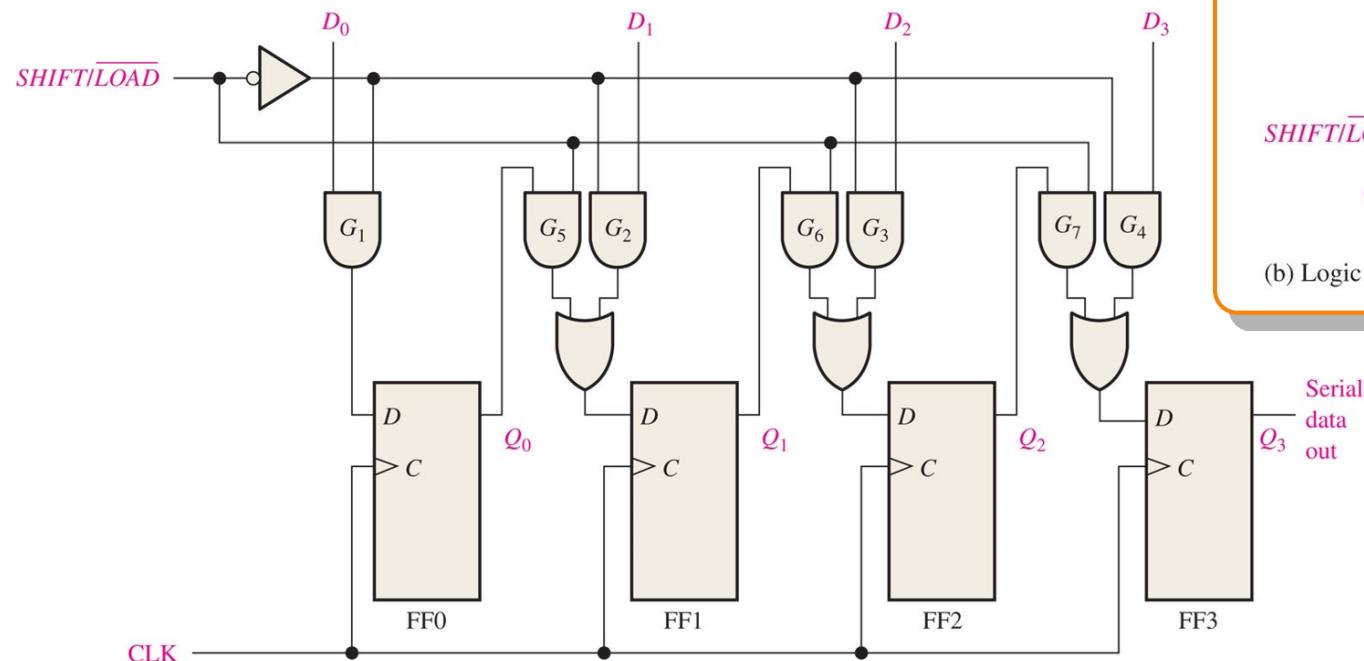
# 8-Bit Serial In/Parallel Out Shift Register



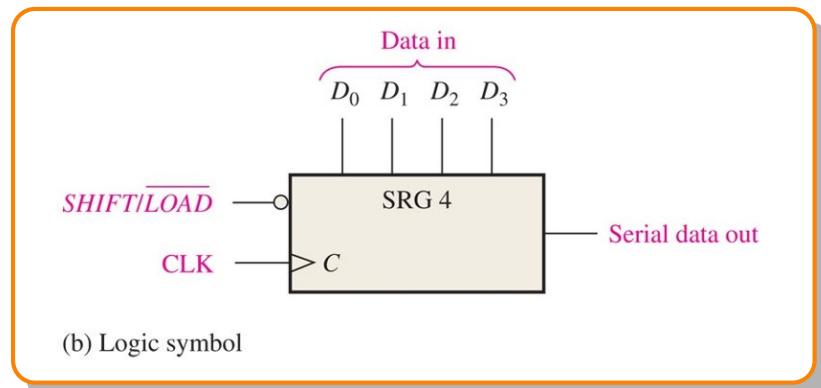
- 74HC164: Fixed-function IC shift register with serial in/parallel out
- Two gated serial inputs: A and B, & an asynchronous clear (CLR) input
- Parallel outputs : Q<sub>0</sub> ~ Q<sub>7</sub>.



# Parallel In/ **Serial** Out Shift Registers



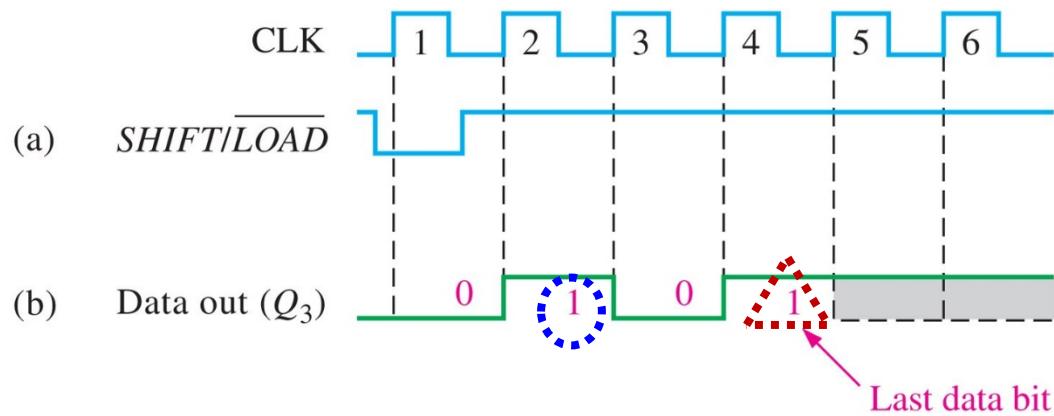
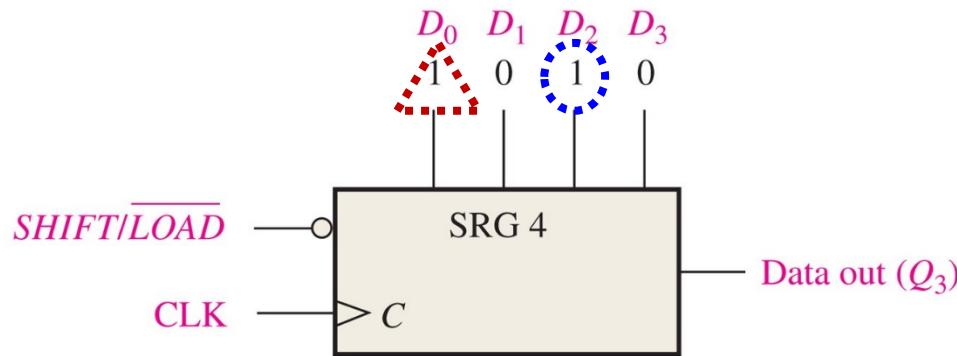
(a) Logic diagram



- When SHIFT/LOAD is LOW, G<sub>1</sub> ~ G<sub>4</sub> are enabled → each data bit is applied to the D input of its respective flip-flop;
- When SHIFT/LOAD is HIGH, G<sub>1</sub> ~ G<sub>4</sub> are disabled while G<sub>5</sub> ~ G<sub>7</sub> are enabled → data bits are shifted right from one stage to the next.



# Example 8–3

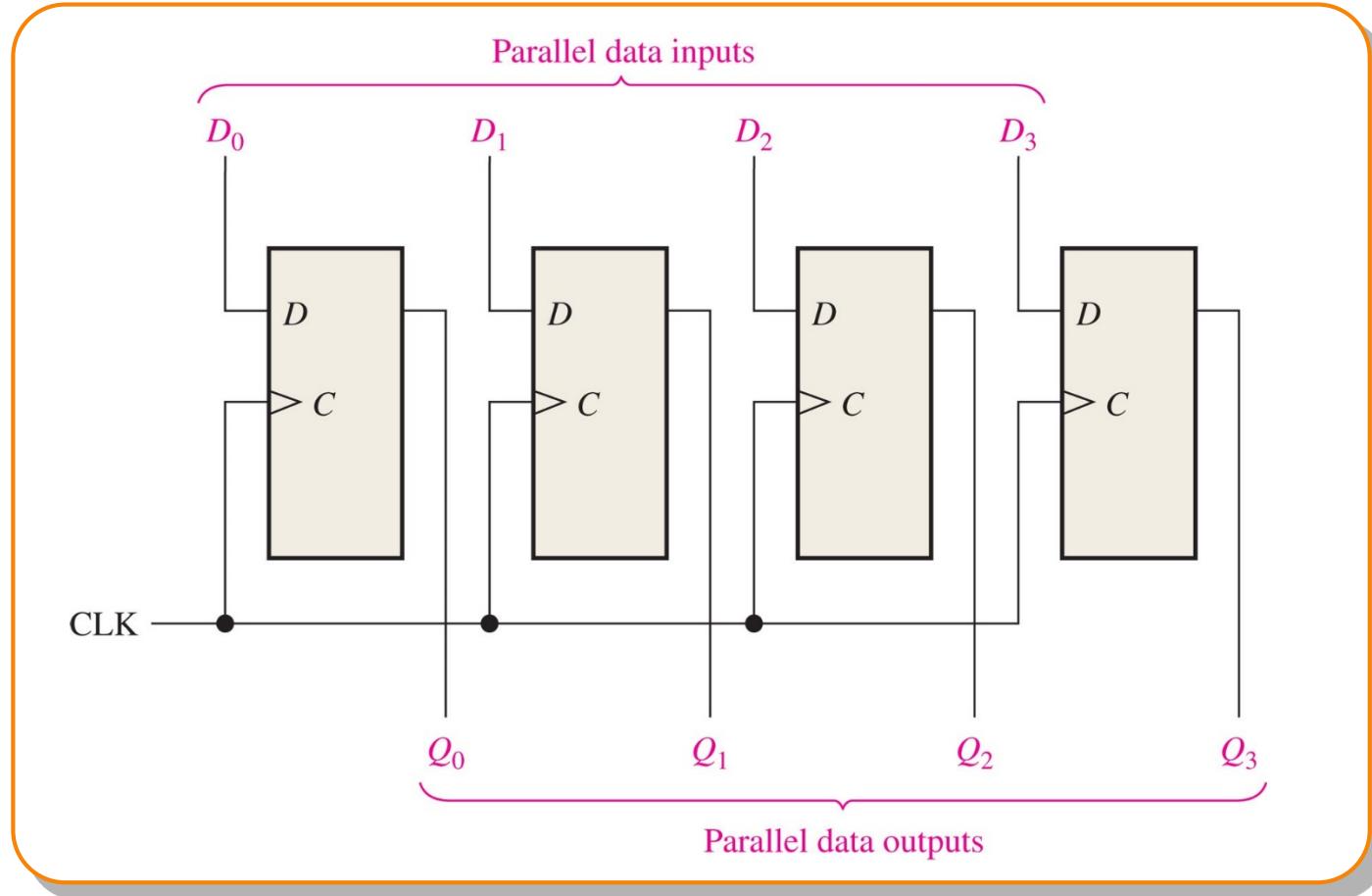


- On clock pulse 1, the parallel data ( $D_0D_1D_2D_3 = 1010$ ) are loaded into the register, making  $Q_3$  a 0.
- On clock pulse 2 the 1 from  $Q_2$  is shifted onto  $Q_3$ ;
- On clock pulse 3 the 0 is shifted onto  $Q_3$ ;
- On clock pulse 4 the last data bit (1) is shifted onto  $Q_3$ ;

- On clock pulse 5, all data bits have been shifted out, and only 1's remain in the register (assuming the  $D_0$  input remains as 1).



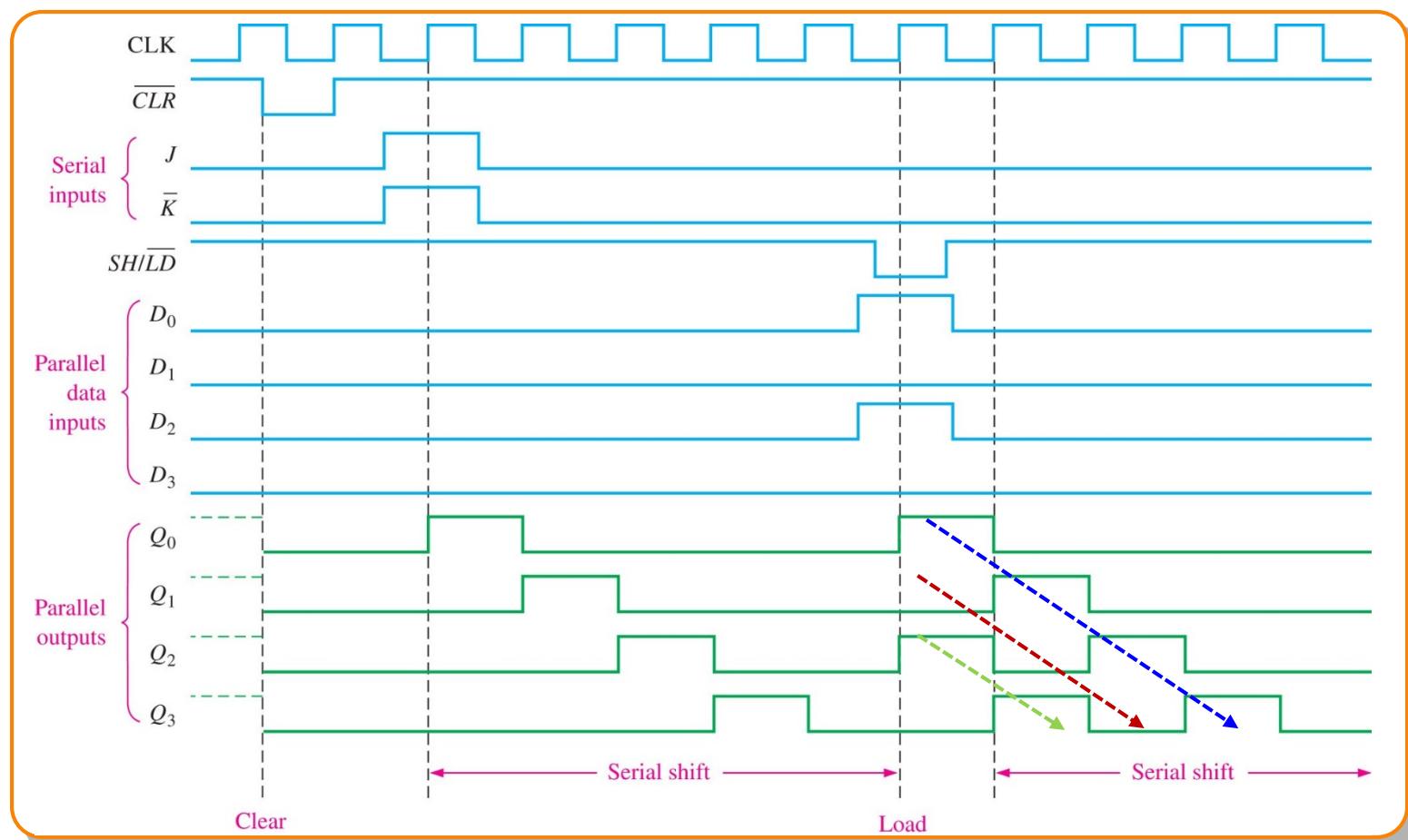
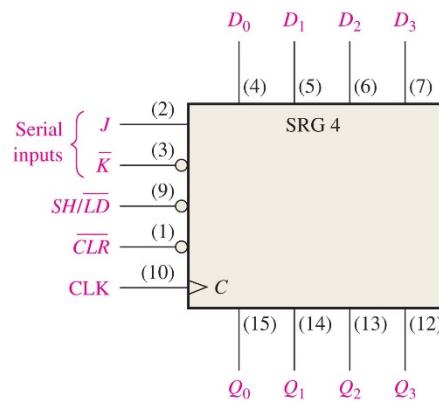
# Parallel In/Parallel Out Shift Registers



- The parallel in/parallel out register employs parallel entry and parallel output.
- Immediately following the simultaneous entry of all data bits, the bits appear on the parallel outputs.



# 4-Bit Parallel-Access Shift Register

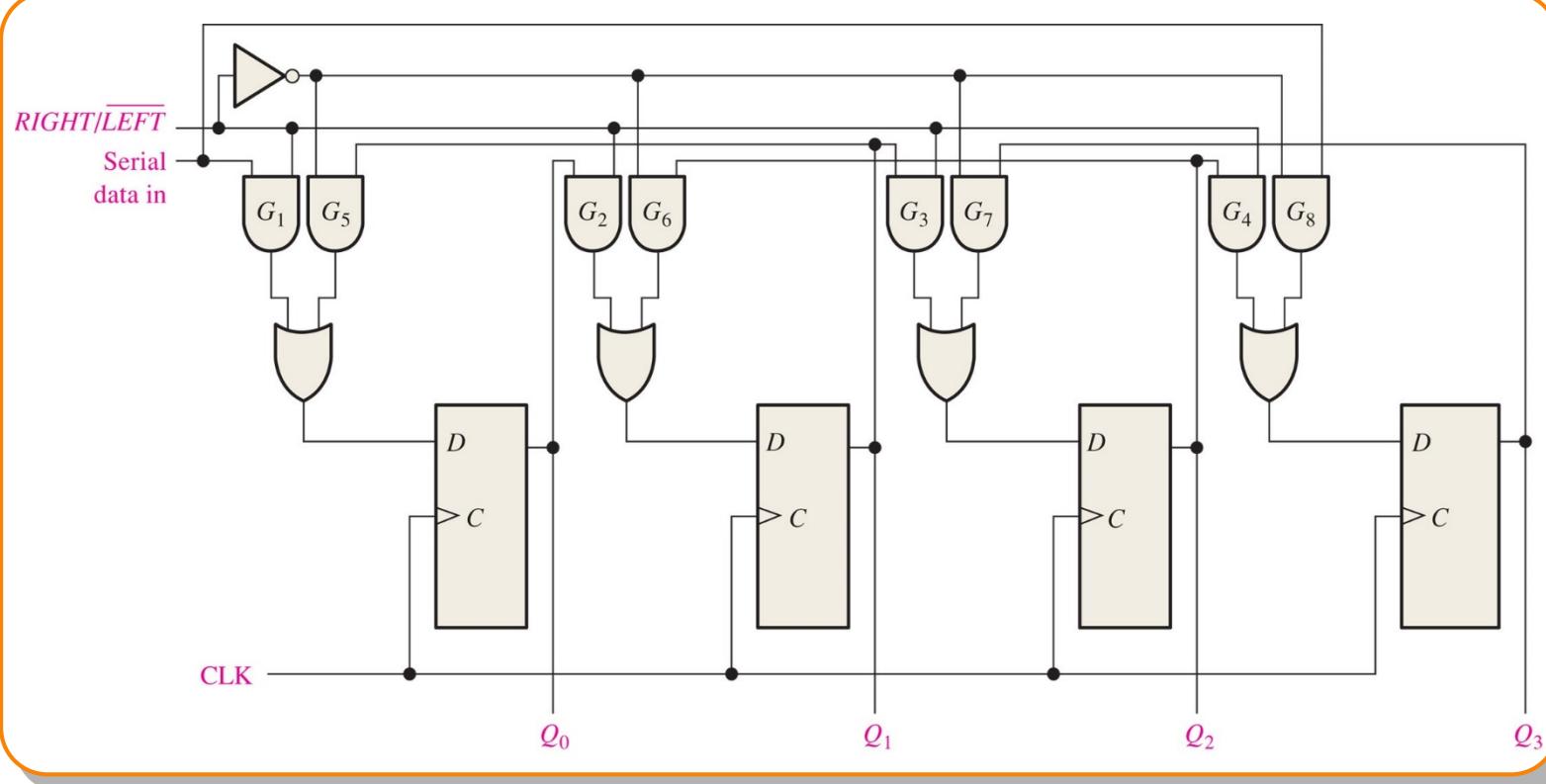


- SHIFT/LOAD : LOW  $\rightarrow$  data on the parallel inputs are entered; HIGH  $\rightarrow$  stored data will shift right ( $Q_0 \sim Q_3$ ) synchronously on the rising edge of the clock.
- Inputs J and K are the serial data inputs to the first stage of the register ( $Q_0$ );  $Q_3$  can be used for serial output data.



# Bidirectional Shift Registers

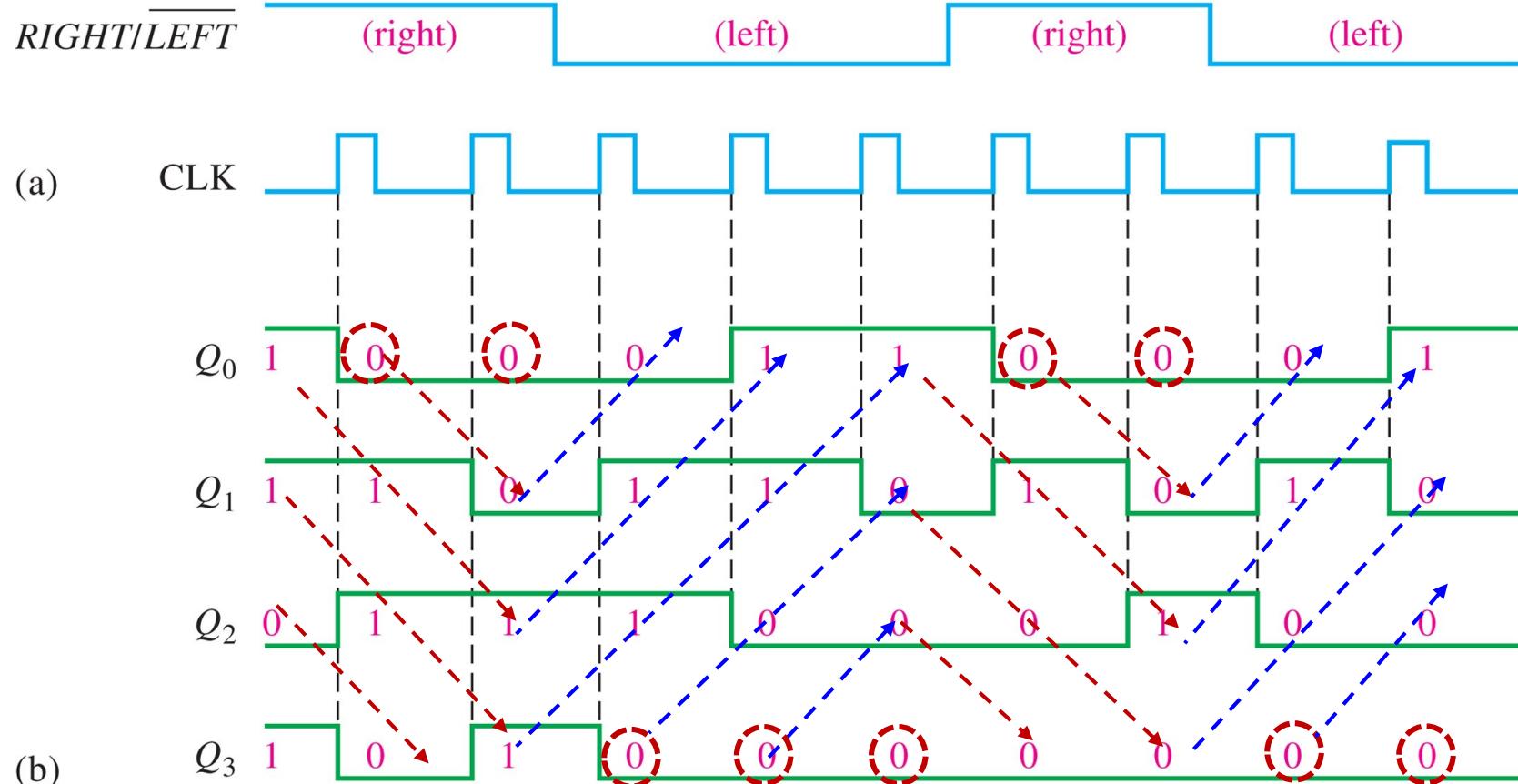
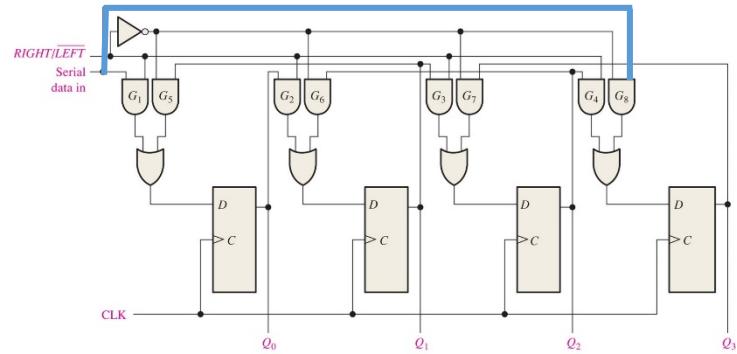
- RIGHT/LEFT controls data bits inside the register to be shifted left or right



- RIGHT/LEFT HIGH:  $G_1 \sim G_4$  are enabled, and the state of the **Q output** of each flip-flop is passed through to the **D input** of the **following** flip-flop.
- RIGHT/LEFT LOW:  $G_5 \sim G_8$  are enabled, and the **Q output** of each flip-flop is passed through to the **D input** of the **preceding** flip-flop.



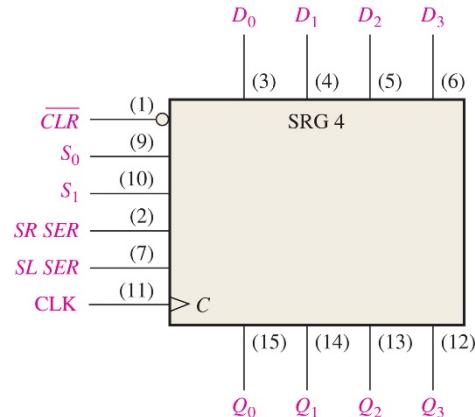
# Example 8–4



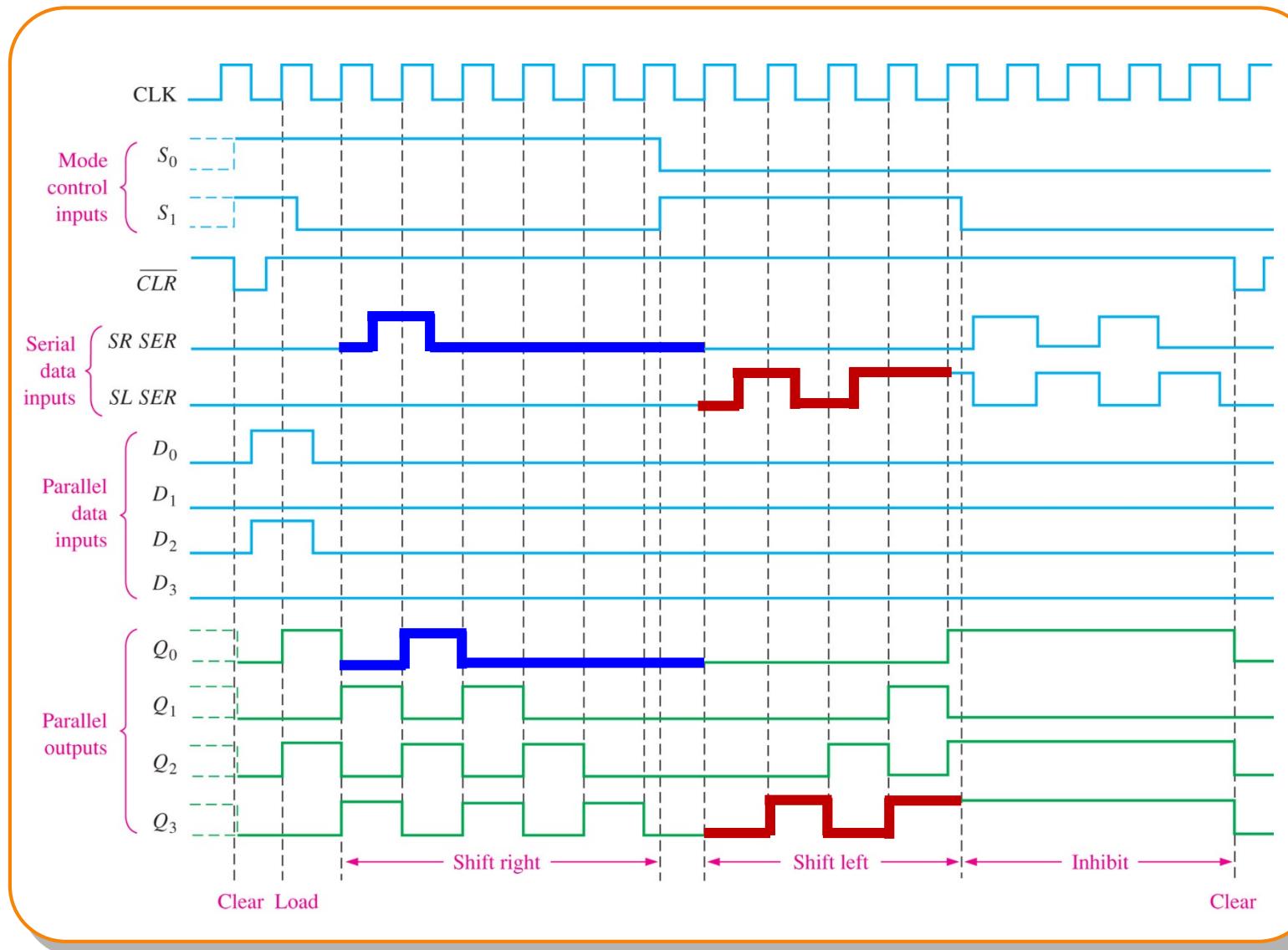
Assume that the serial data-input line is LOW.



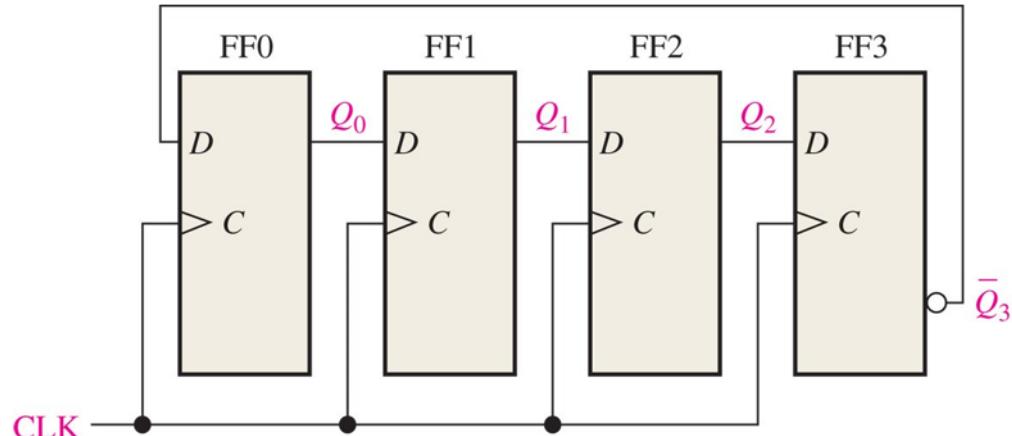
# 4-Bit Bidirectional Universal Shift Register



$S_0$	$S_1$	Func.
HIGH	HIGH	Parallel Loading
HIGH	LOW	Shift right
LOW	HIGH	Shift left
LOW	LOW	Inhibit



# Johnson counters



(a) Four-bit Johnson counter

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 ←

- Johnson counters: the complement of the output of the last flip-flop is connected back to the D input of the first flip-flop
- If the counter starts at 0, this feedback arrangement produces a characteristic sequence of states
- In general, a Johnson counter will produce a modulus of  $2n$ , where n is the number of stages in the counter, e.g. for  $n=4$  stages, 8 states.



# Ring Counters

$Q$  rather than  $\bar{Q}$  is fed back from the last stage

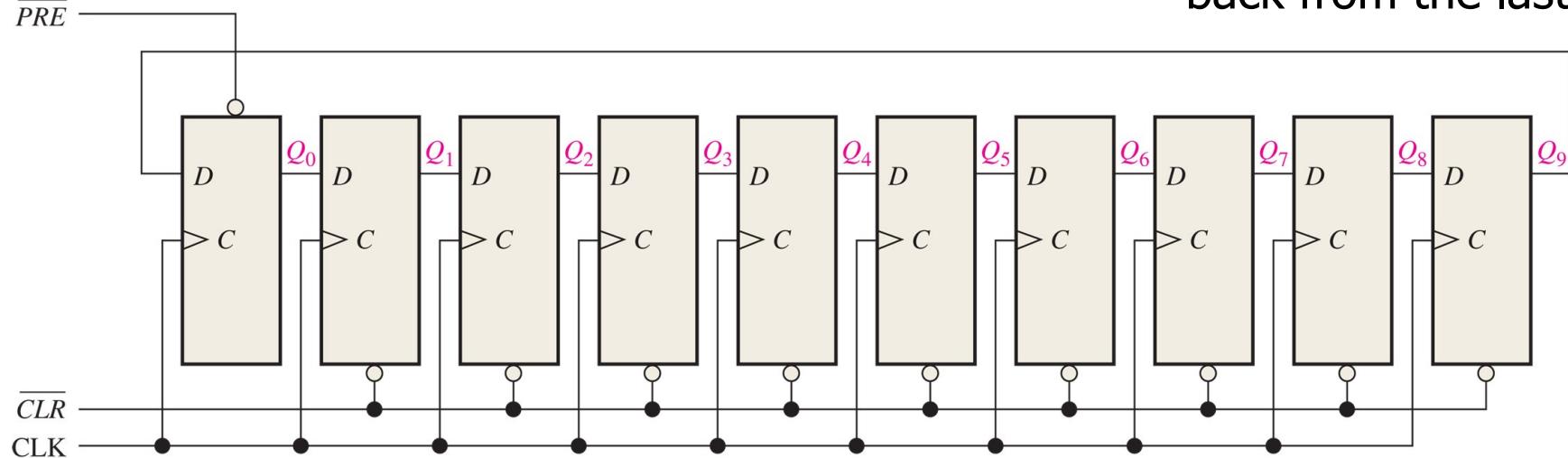


TABLE 8-5

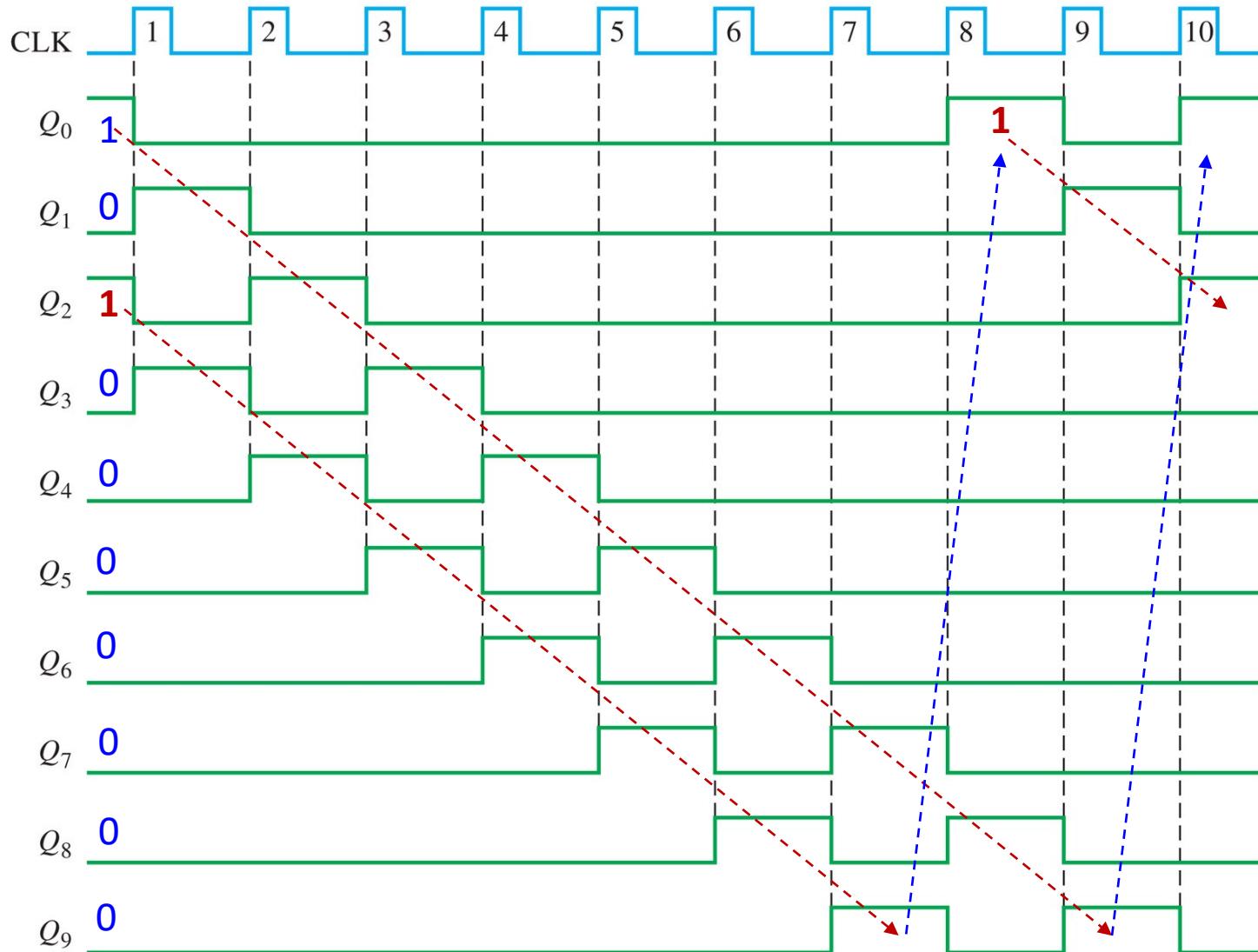
Ten-bit ring counter sequence.

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

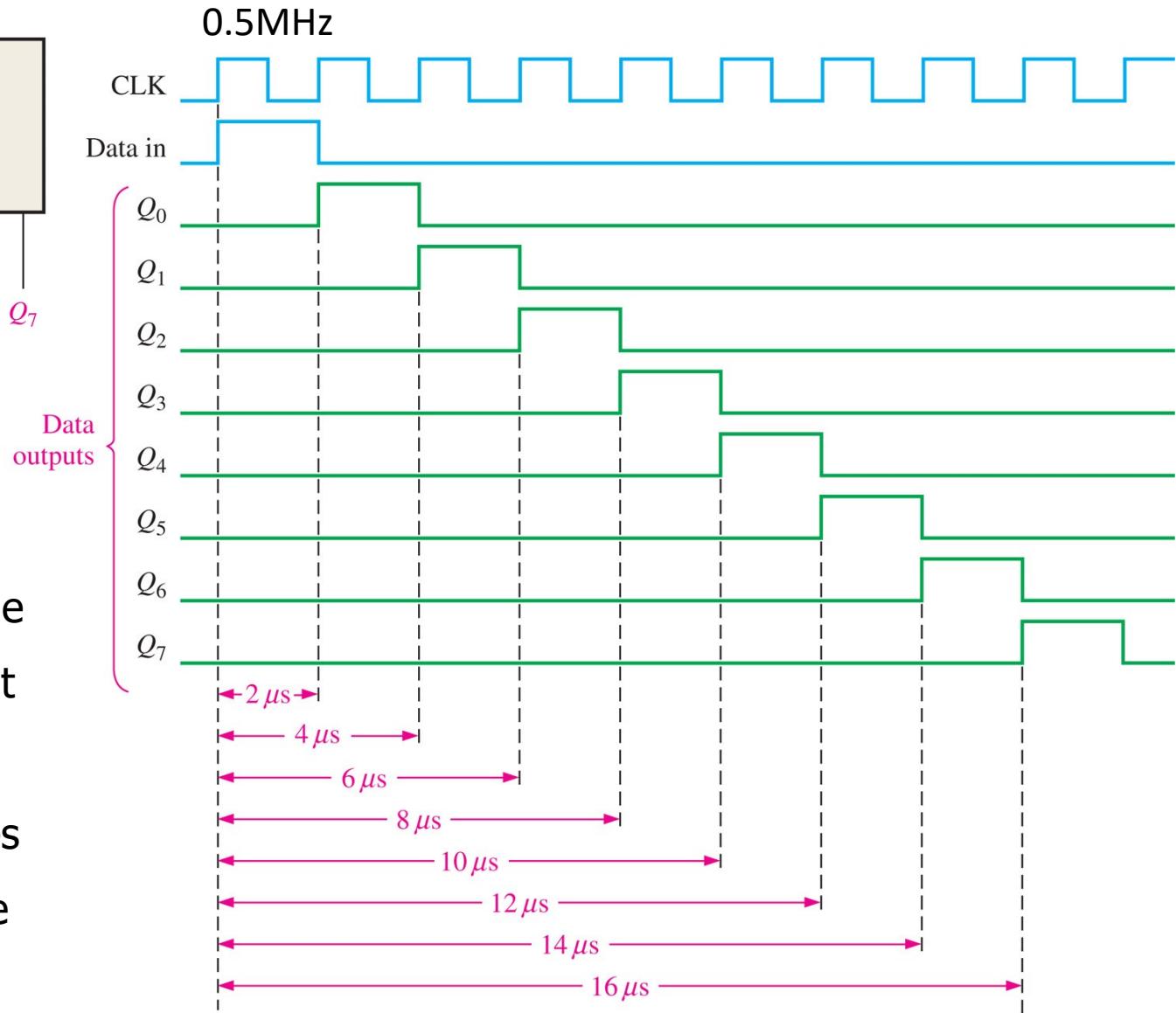
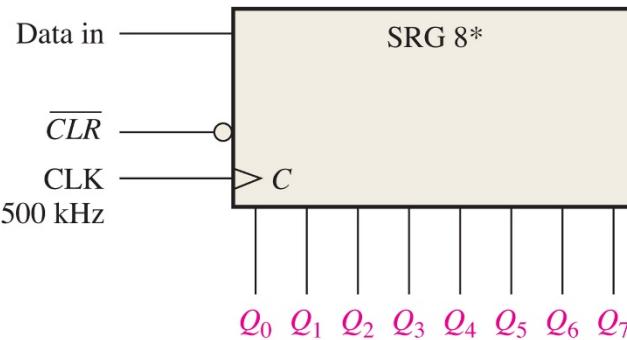
Initially, a 1 is preset into the first flip-flop, and the rest of the flip-flops are cleared.



# Example 8–5



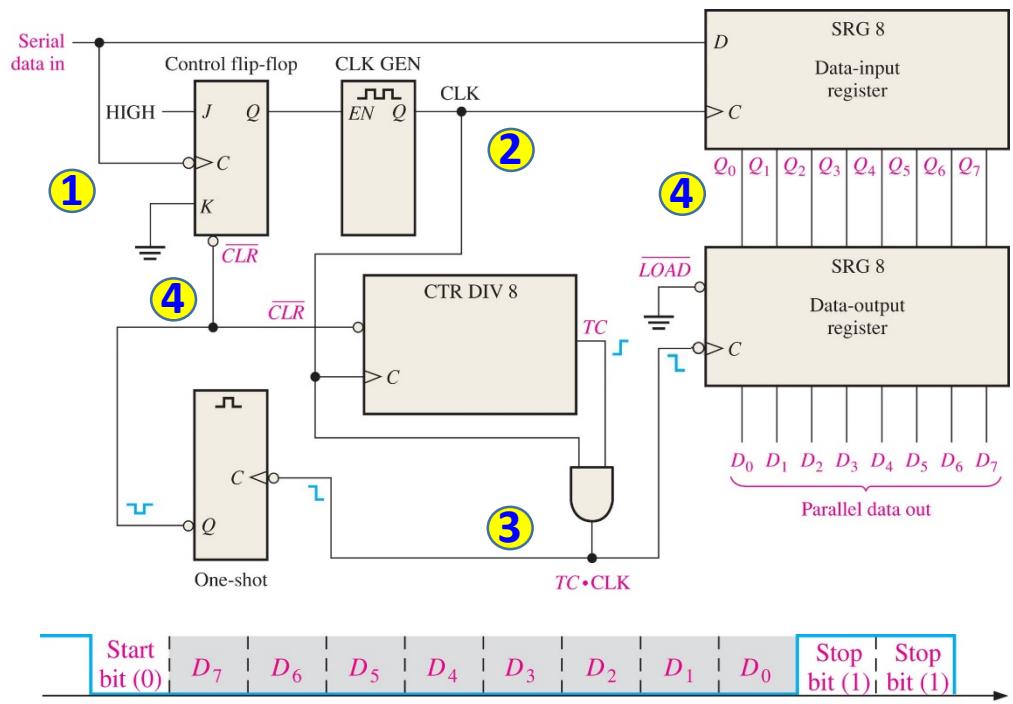
# Shift Register Applications: Time Delay



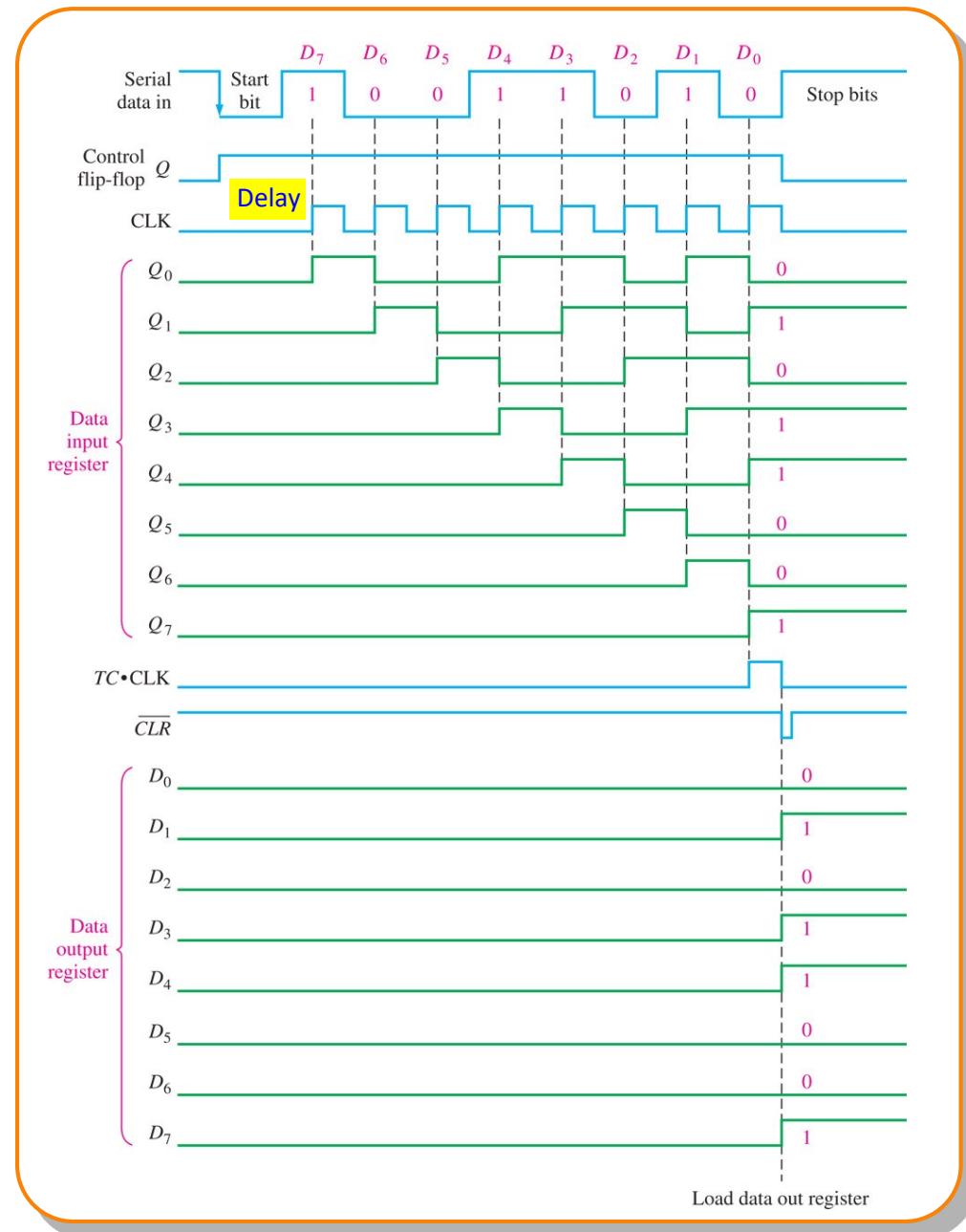
- A serial in/serial out shift register can provide a time delay from input to output
- The delay is a function of both the number of stages (n) in the register and the clock frequency.



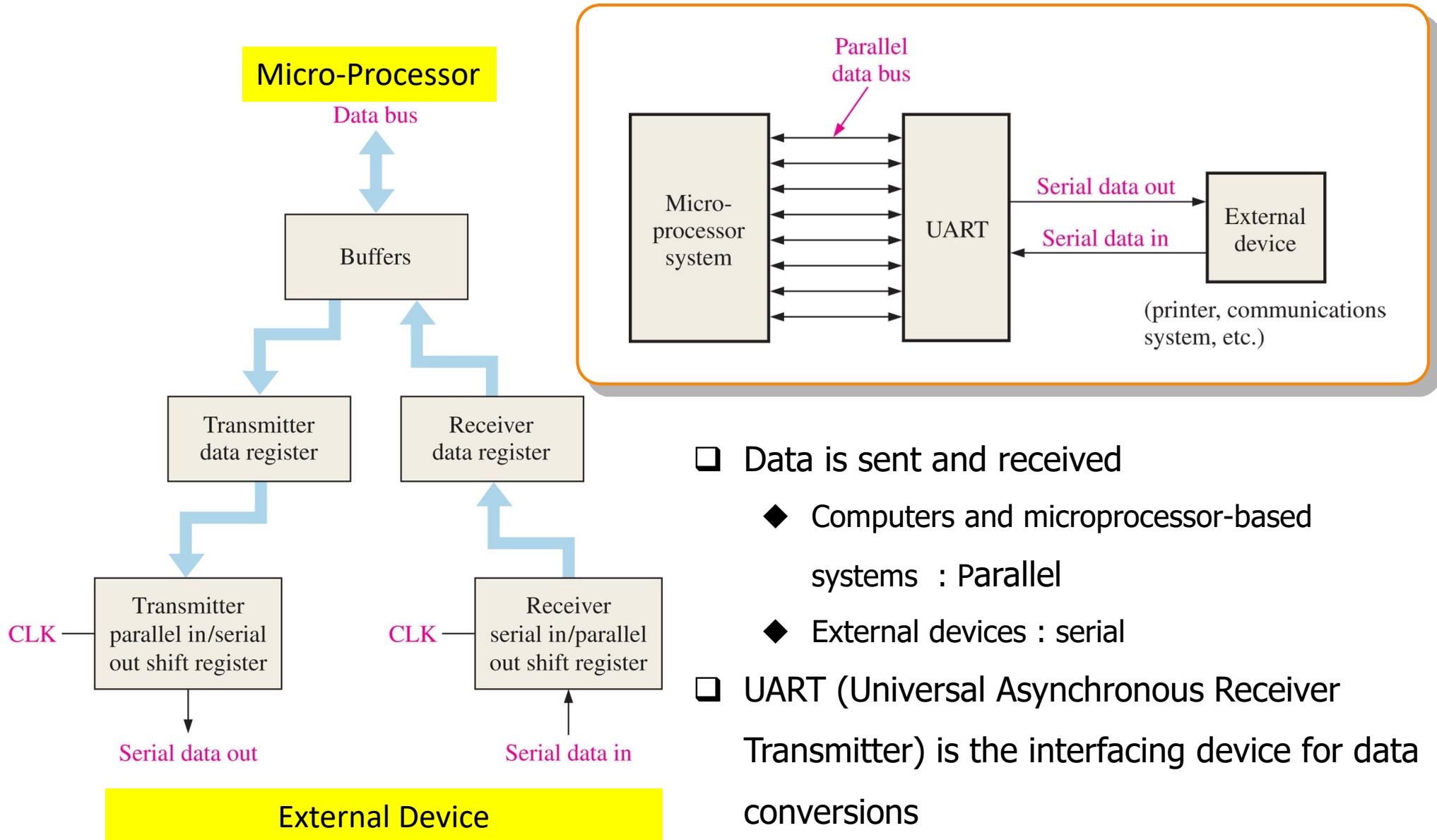
# Shift Register Applications: S/P Data Converter



- Serial-to-Parallel Data Converter: Serial data transmission can reduce the number of wires in the transmission line.
- USB (universal serial bus) is used to connect keyboards printers, scanners, and more to the computer.



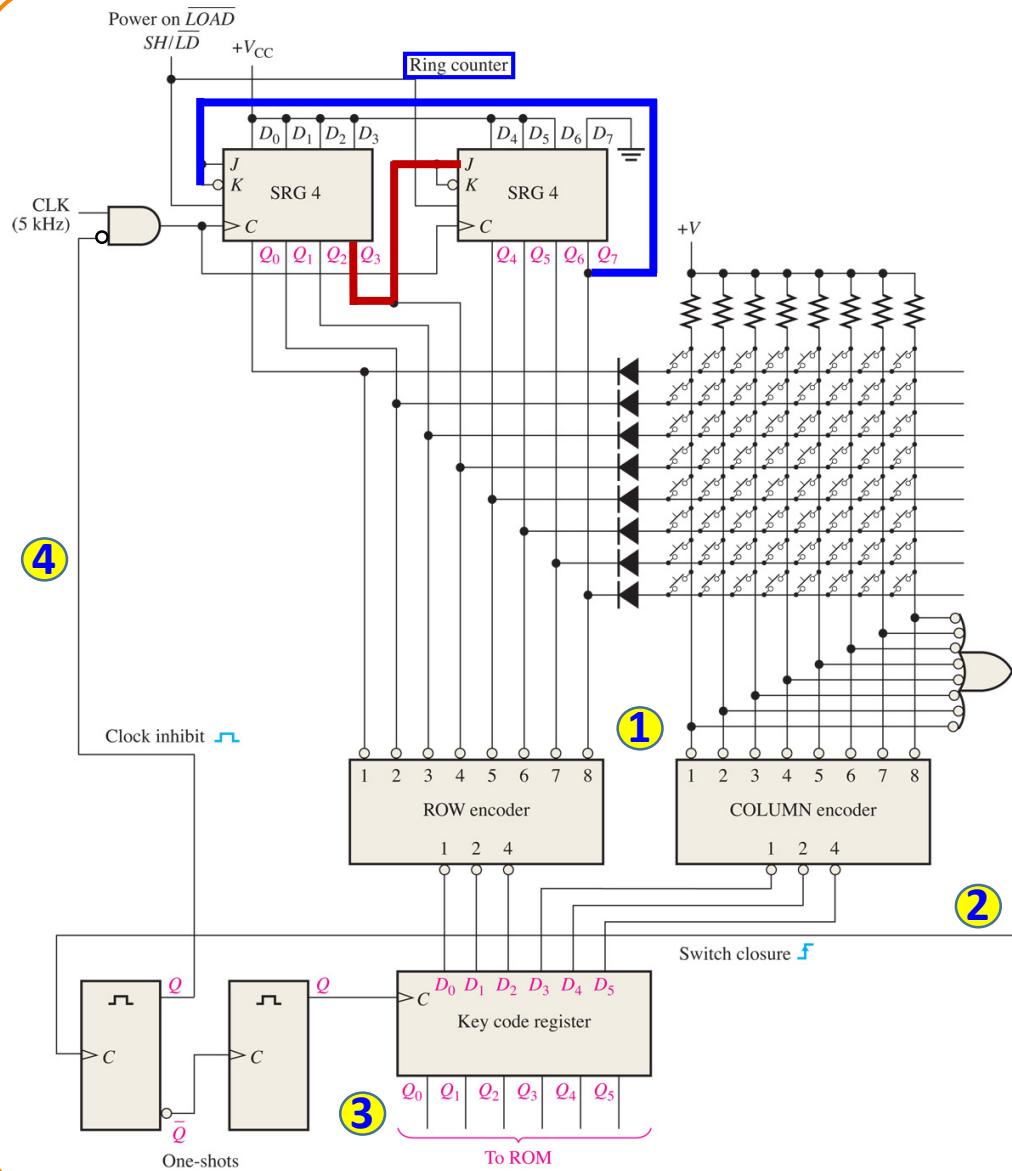
# Universal Asynchronous Receiver Transmitter



- Data is sent and received
  - ◆ Computers and microprocessor-based systems : Parallel
  - ◆ External devices : serial
- UART (Universal Asynchronous Receiver Transmitter) is the interfacing device for data conversions



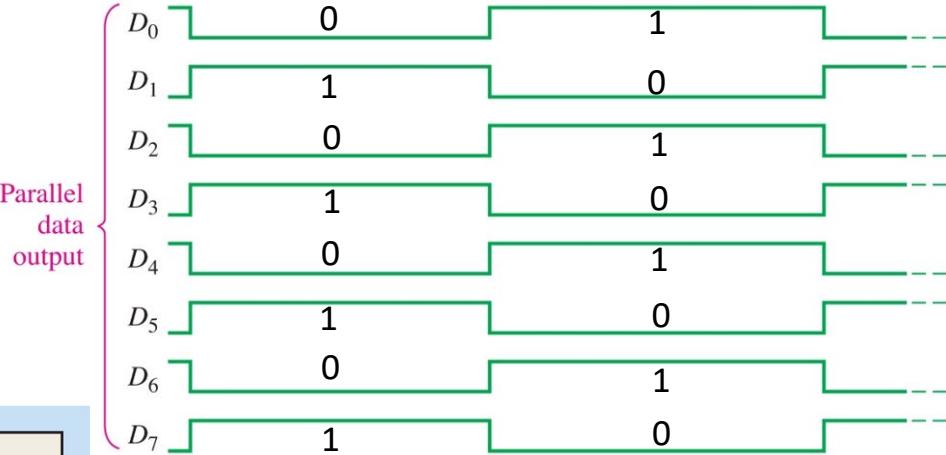
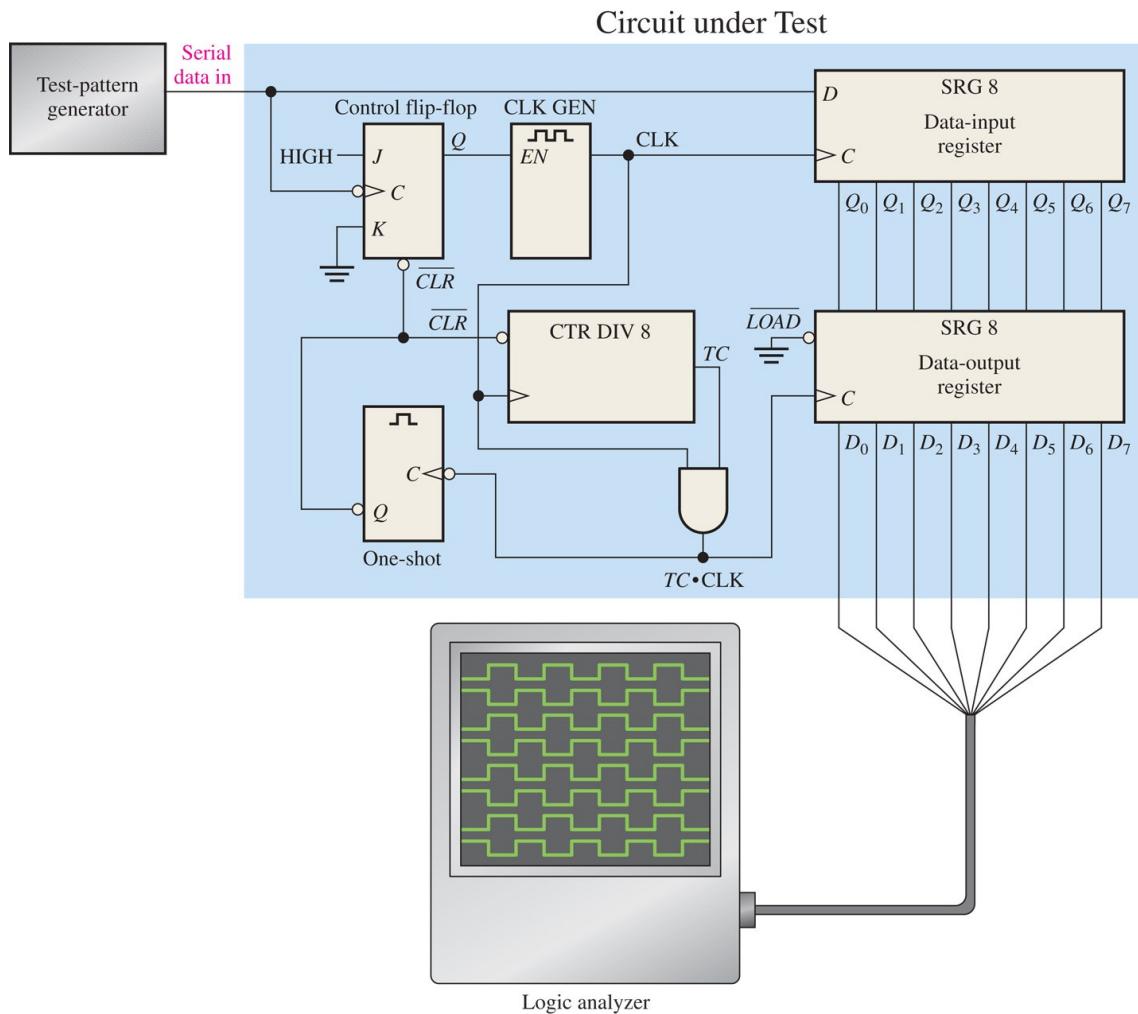
# Keyboard Encoder



- ❑ Initially, **11111110** is loaded into the ring counter when Power ON;
- ❑ Then, the ring counter “scans” the rows for a key closure as the clock signal shifts the 0 around the counter at a 5 kHz rate.
- ❑ If a key closure occurs, one COLUMN line is connected to one ROW line. When the ROW line is taken LOW by the ring counter, that particular COLUMN line is also pulled LOW.
- ❑ The 3-bit ROW code plus the 3-bit COLUMN code uniquely identifies the key that is closed.
- ❑ When a key is closed, the two one-shots produce a **delayed** clock pulse to parallel-load the 6-bit code into the key code register.
- ❑ The first one-shot output inhibits the ring counter to prevent it from scanning.



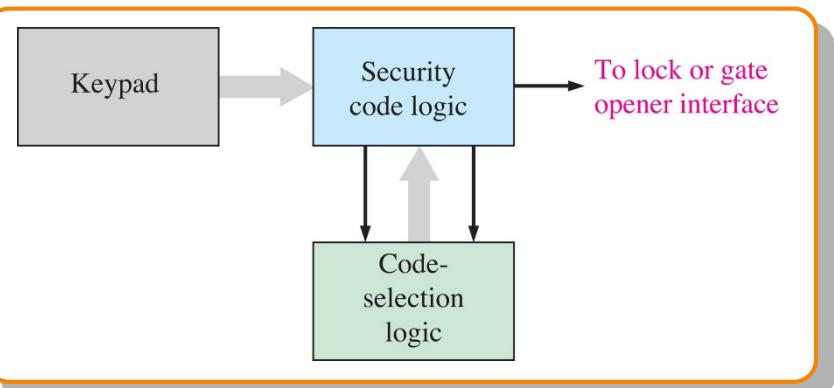
# Troubleshooting



- Test patterns: To force all elements (flip-flops and gates) into all of their states to be certain that nothing is stuck in a given state as a result of a fault.
- The input test pattern must be designed to force each flipflop in the registers into both states.

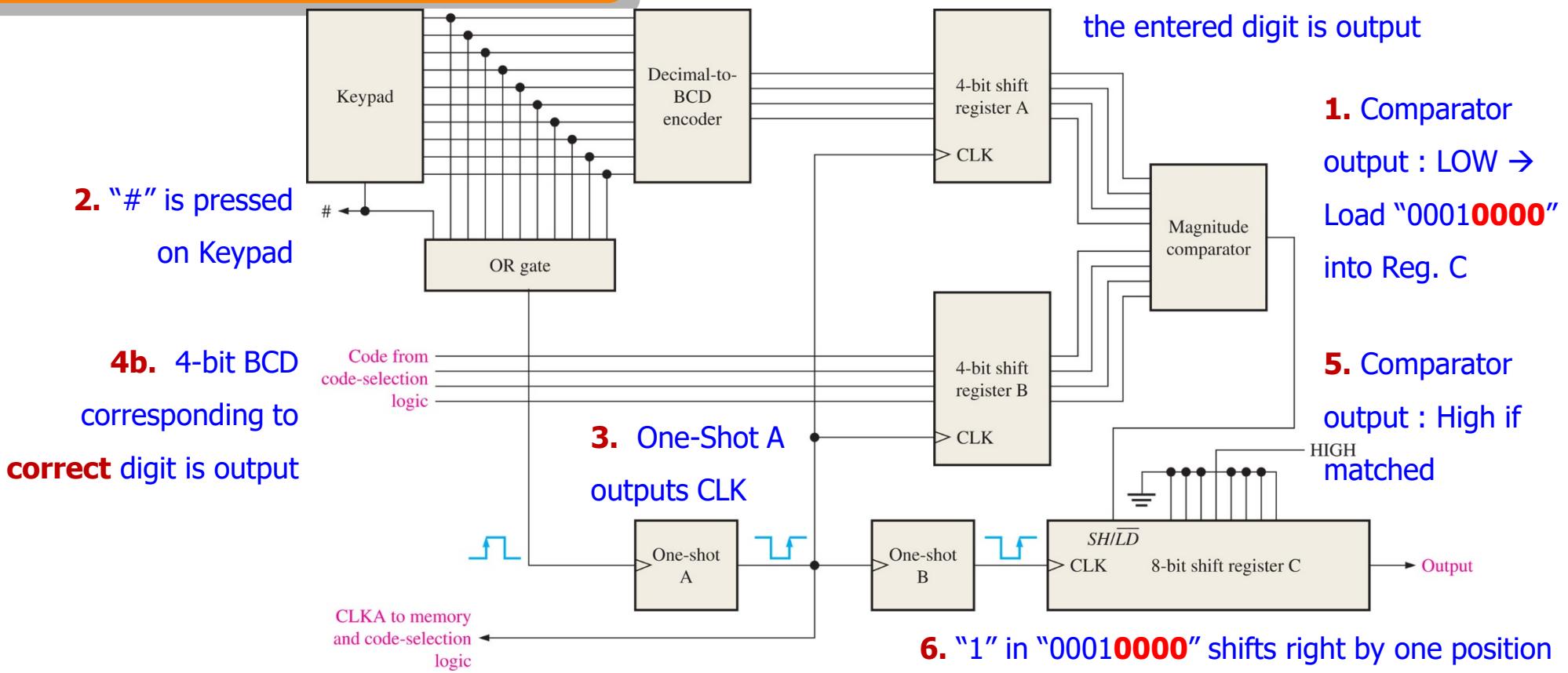


# Applied Logic : Security System



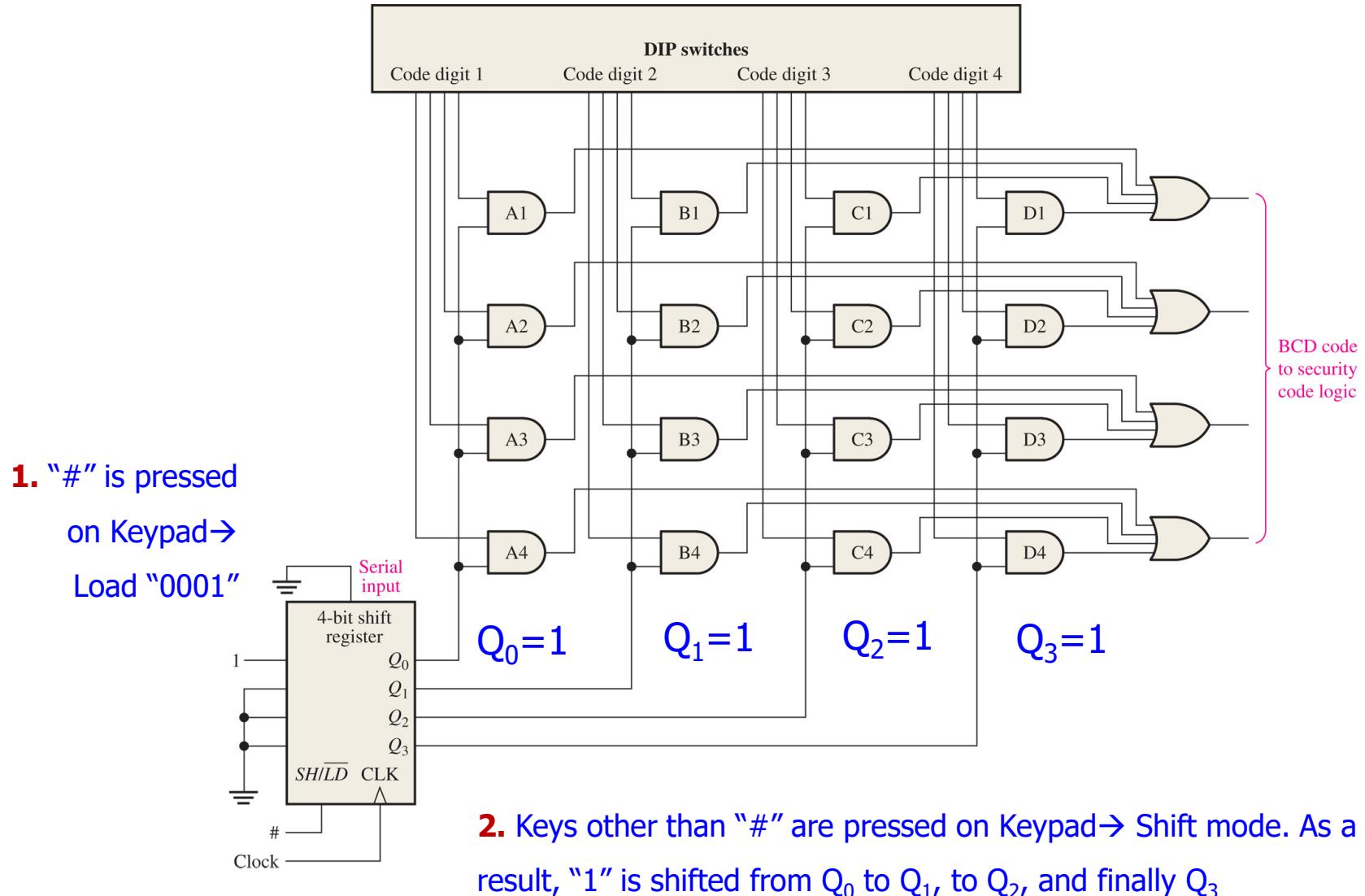
- A 4-digit security code is stored in the system
- Access is achieved by entering the correct code on a keypad.

**4a.** For every digit entered, a 4-bit BCD corresponding to the entered digit is output



# Code Selection Logic

4-bit BCD corresponding each digit of the correct 4-bit code



# Chapter Review

## □ Shift Registers

- ◆ Data storage
- ◆ Data movement
- ◆ Serial/Parallel In - Serial/Parallel Out

## □ Counters

- ◆ Johnson counters
- ◆ Ring Counters

## □ Shift Register Applications

- ◆ Time Delay
- ◆ Serial-to-Parallel Data Converter
- ◆ UART (Universal Asynchronous Receiver Transmitter)
- ◆ Keyboard Encoder



# True/False Quiz

- Shift registers consist of an arrangement of flip-flops.
- A shift register cannot be used to store data.
- A serial shift register accepts one bit at a time on a single line.
- All shift registers are defined by specified sequences.
- A shift register counter is a shift register with the serial output connected back to the serial input.
- A shift register with four stages can store a maximum count of fifteen.
- The Johnson counter is a special type of shift register.
- The modulus of an 8-bit Johnson counter is eight.
- A ring counter uses one flip-flop for each state in its sequence.
- A shift register cannot be used as a time delay device.



# True/False Quiz

- Shift registers consist of an arrangement of flip-flops.
- A shift register cannot be used to store data.
- A serial shift register accepts one bit at a time on a single line.
- All shift registers are defined by specified sequences.
- A shift register counter is a shift register with the serial output connected back to the serial input.
- A shift register with four stages can store a maximum count of fifteen.
- The Johnson counter is a special type of shift register.
- The modulus of an 8-bit Johnson counter is eight.
- A ring counter uses one flip-flop for each state in its sequence.
- A shift register cannot be used as a time delay device.

