PDS0101

Introduction to Digital Systems

Shift Registers

MULTIMEDIA UNIVERSITY

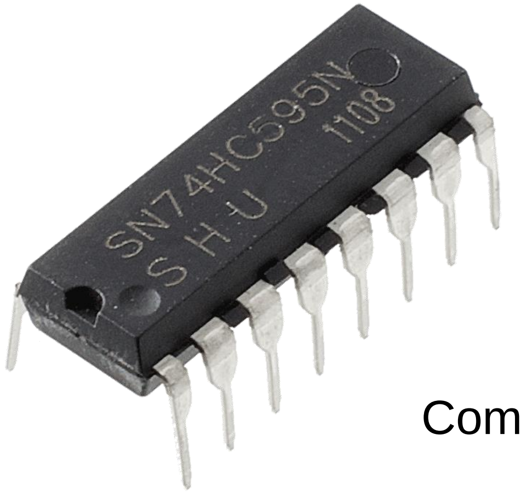
Lecture outcome

- By the end of today's lecture you should be able to
  - identify basic forms of data movement in shift registers
  - explain serial/parallel in/out combinations of shift registers
  - determine sequence of a Johnson counter
  - construct a ring counter from a shift register

- **NOTE**: contents in this set of slides are intentionally incomplete and content will be shown in class as examples of how they are derived

What are shift registers?

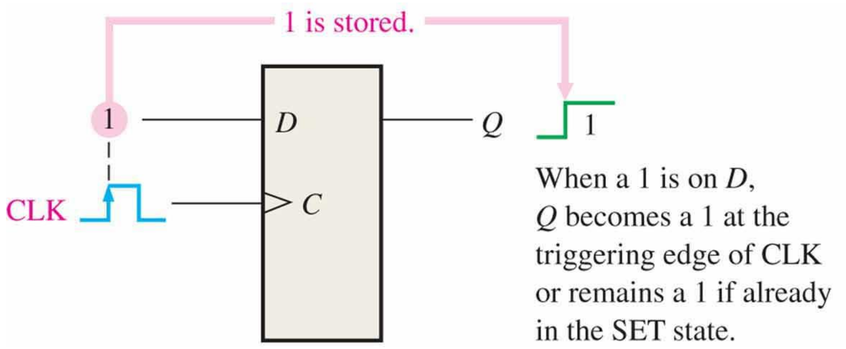
- Shift registers are a type of <u>sequential logic circuit</u> closely related to counters → Consist of an arrangement of FFs
- Important in functions involving storage and transfer of data in a digital system
- Unlike counters, registers typically have no specific sequence of states except in special circumstances
- The register stores and shifts data (binary bits) entered into in from external sources



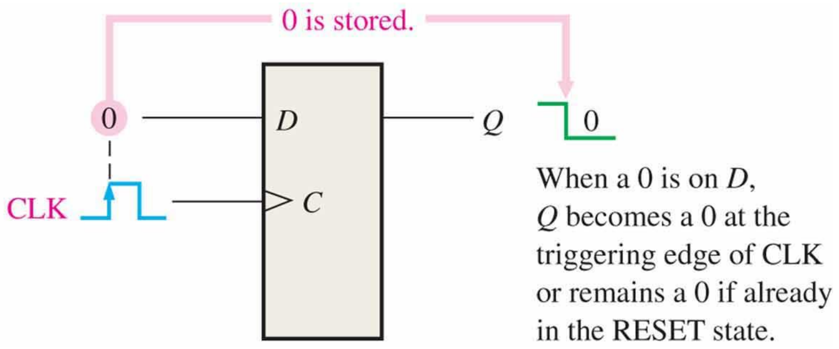
Common 8-bit parallel out shift register

Recap – FF as memory device

- FFs were taught earlier as a form of a storage device to store the value of a single bit
- The figure below recaps the concept of storing a 1 or 0 in a D FF by setting or resetting the FF



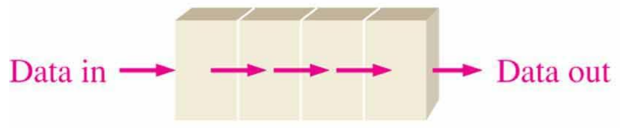
When a 1 is on *D*, *Q* becomes a 1 at the triggering edge of CLK or remains a 1 if already in the SET state.



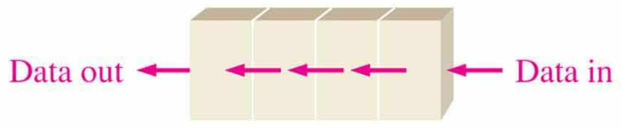
When a 0 is on *D*, *Q* becomes a 0 at the triggering edge of CLK or remains a 0 if already in the RESET state.

Shift register operations

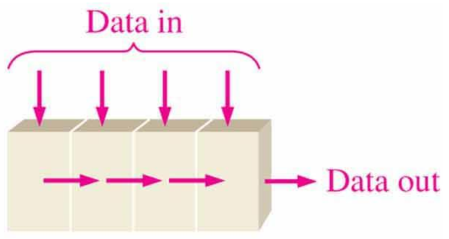
- A register can consist of one or more FFs used to store and shift data
- The storage capacity of a register is the total number of bits it can retain – each bit/FF is usually referenced as a *stage*
- The *shift* capability of a register allows the movement of data from stage to stage within or in/out of the register upon a clock pulse trigger
- The next slide shows the types of data movement possible in shift registers
    - The block represent an arbitrary 4-bit/stage registers and arrows indicate movement



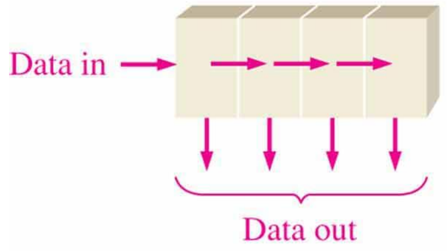
(a) Serial in/shift right/serial out



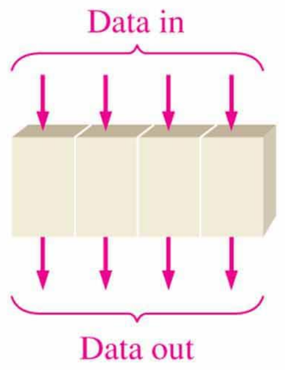
(b) Serial in/shift left/serial out



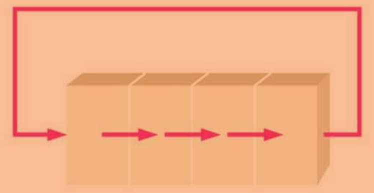
(c) Parallel in/serial out



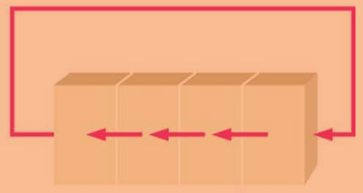
(d) Serial in/parallel out



(e) Parallel in/parallel out



(f) Rotate right

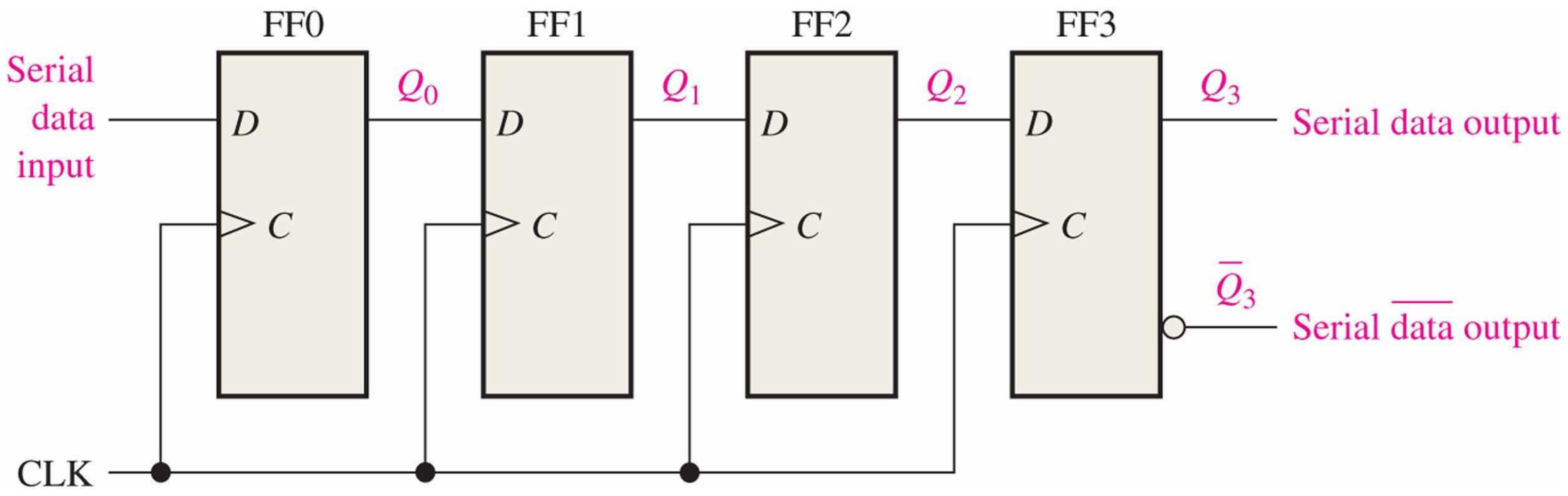


(g) Rotate left

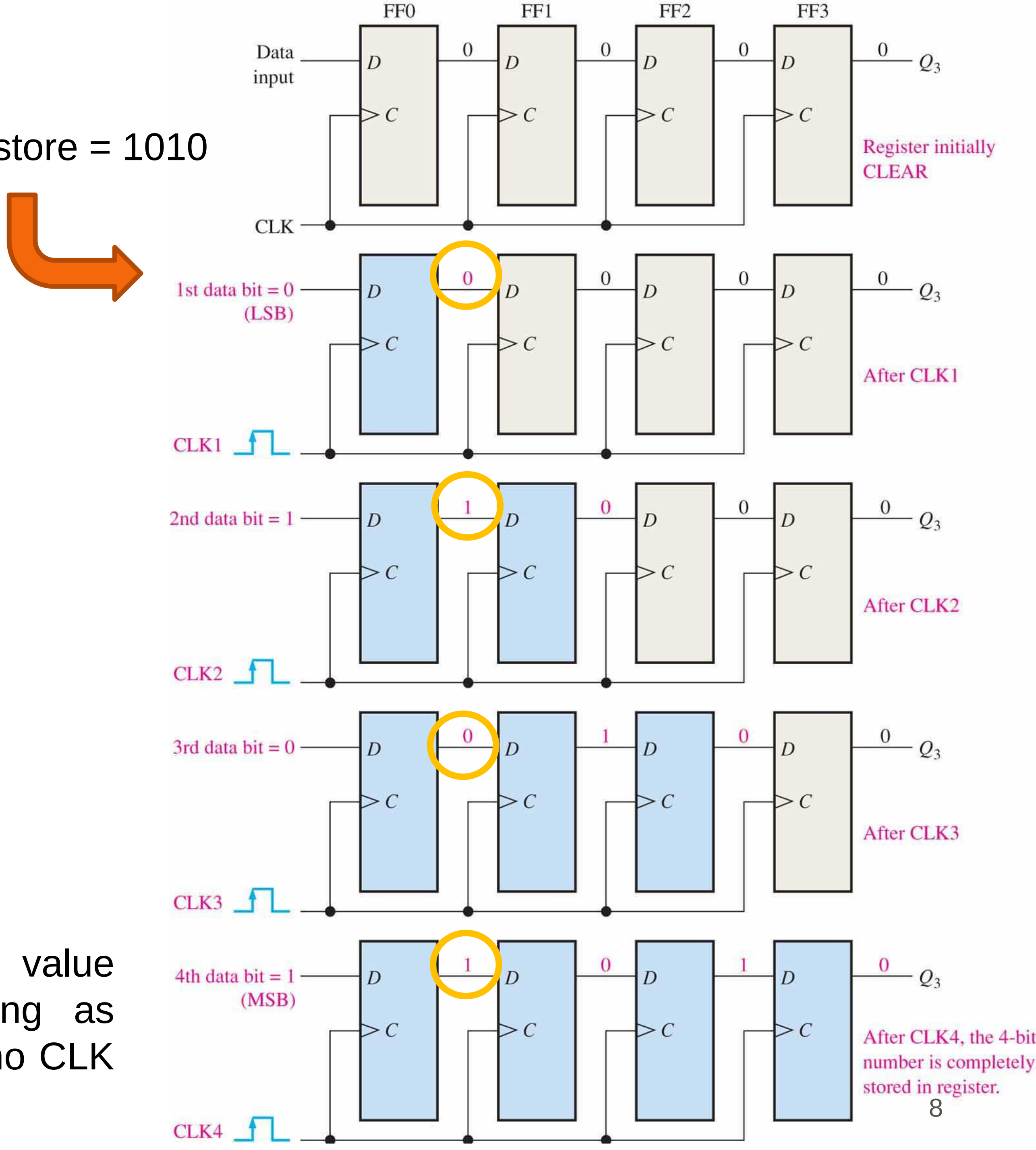
NOT IN SYLLABUS

Serial in/out Shift Registers

- Shift registers are available in IC form or can be constructed from discrete flip-flops as shown here with a four-bit serial-in serial-out register
- Each clock pulse will move an input bit to the next flip-flop.
- The next slide shows the example of how bits are stored by shifting individual bits into the register



Bits to store = 1010



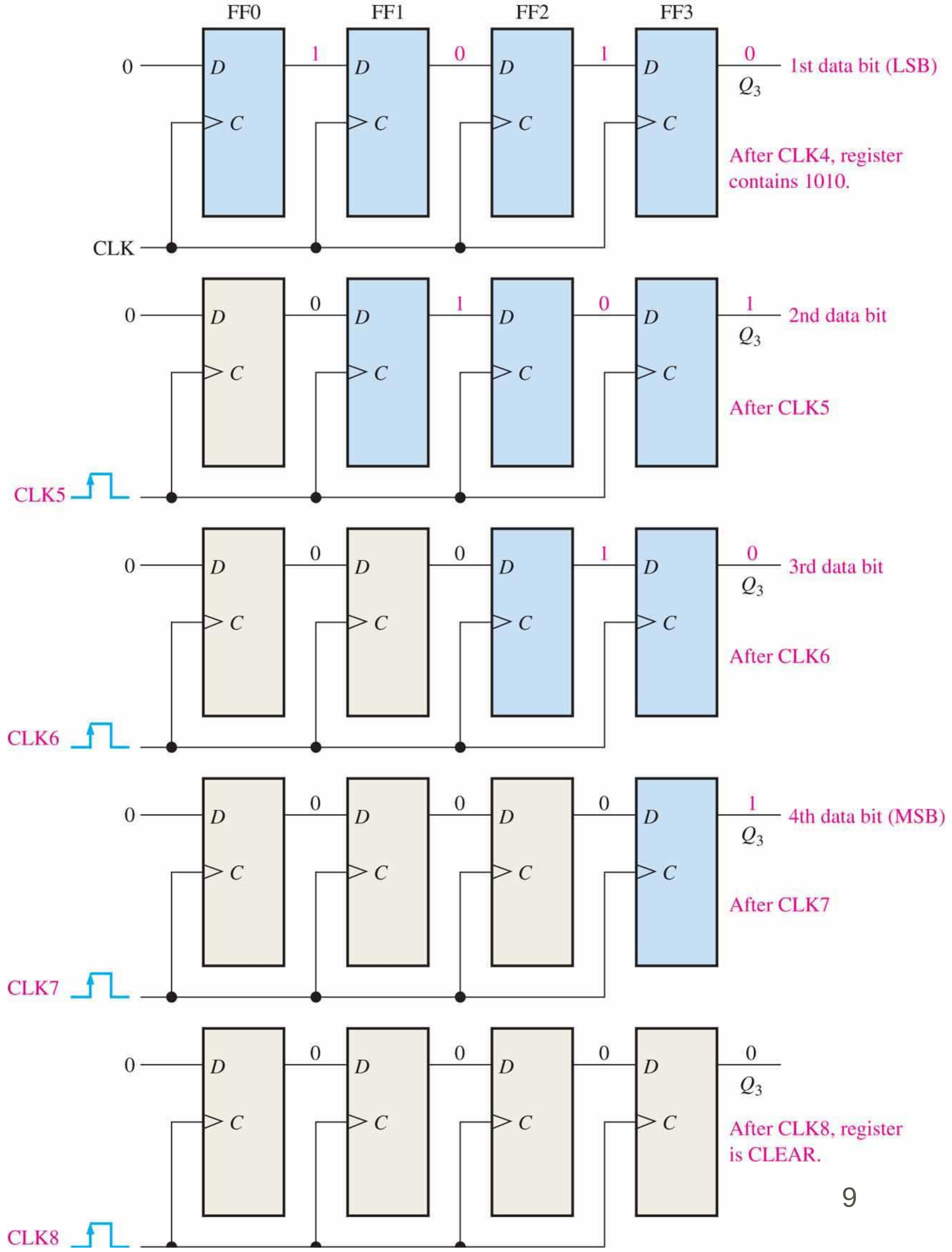
Register now stores the value 1010 indefinitely (as long as there is DC supply) and no CLK applied

The same process is used to shift data out of the register. After CLK4 (from previous slide), the value of LSB can be obtained from the output at Q3

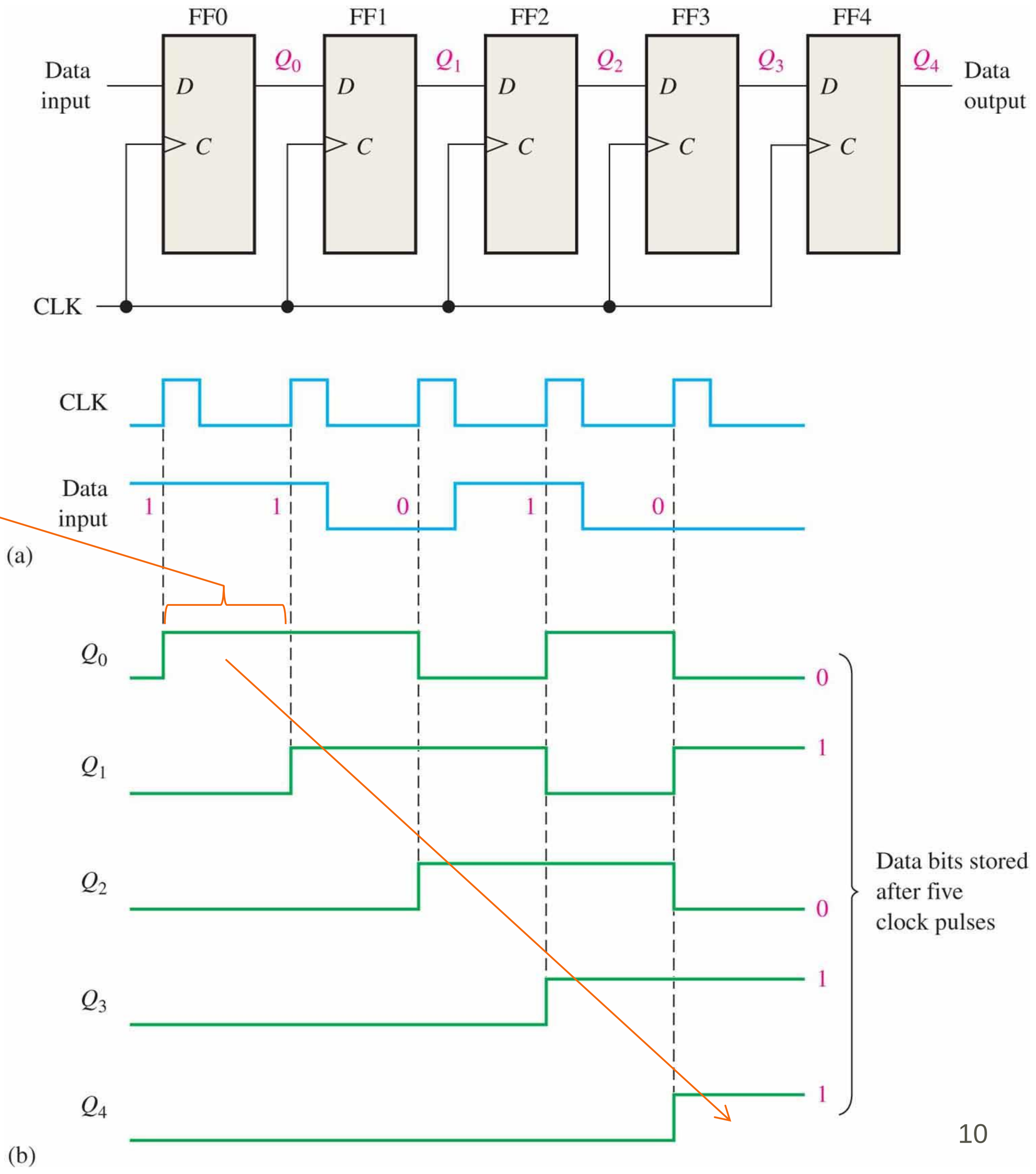
CLK5 pulse shifts the bits, and the second bit appears

CLK6 and CLK7 shift to reveal third and fourth/final bit

Note that while the bits are shifted *out* new bits can be shifted *in* but the diagram shows only zeros shifted in

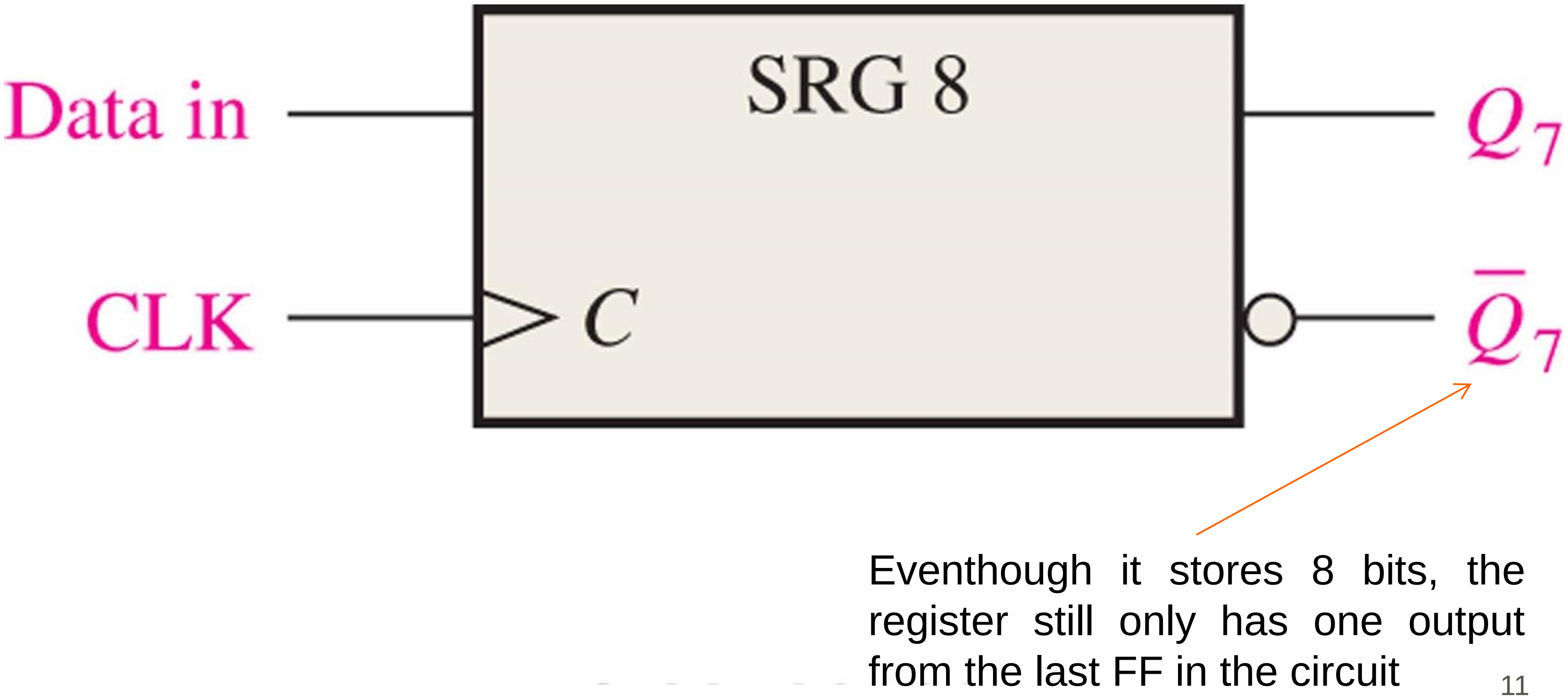


Note how the ‘first’ bit gets shifted from one output to another along the register



Serial I/O shift register block symbol

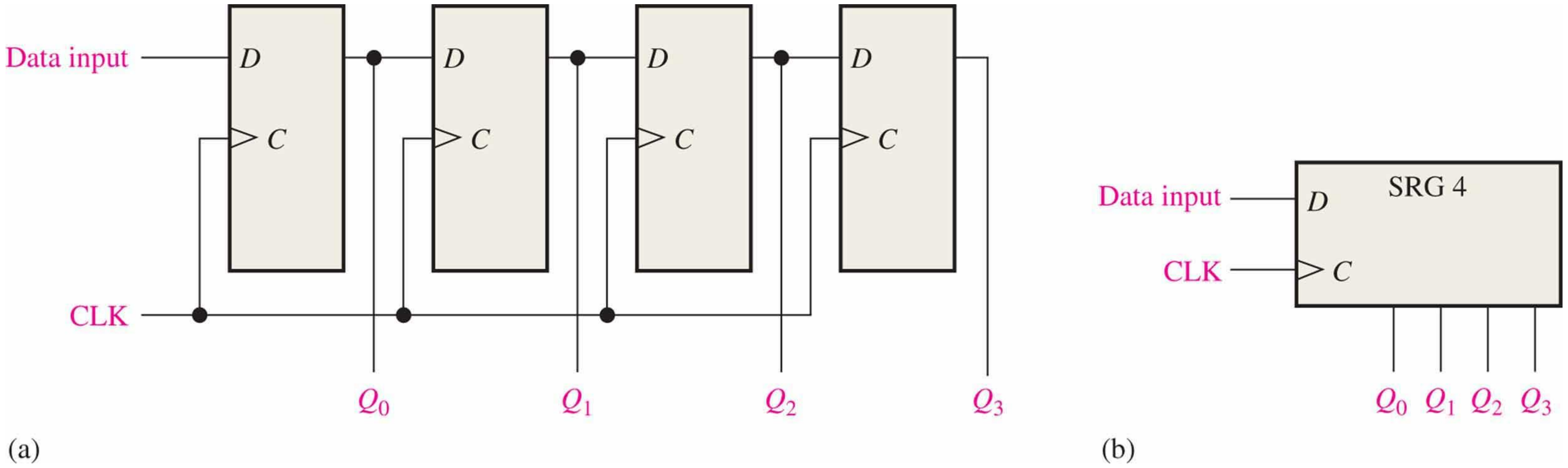
- The standard logic block symbol for a serial I/O shift register is shown below.
  - Note that “SRG 8” indicates a shift register with 8-bit capacity



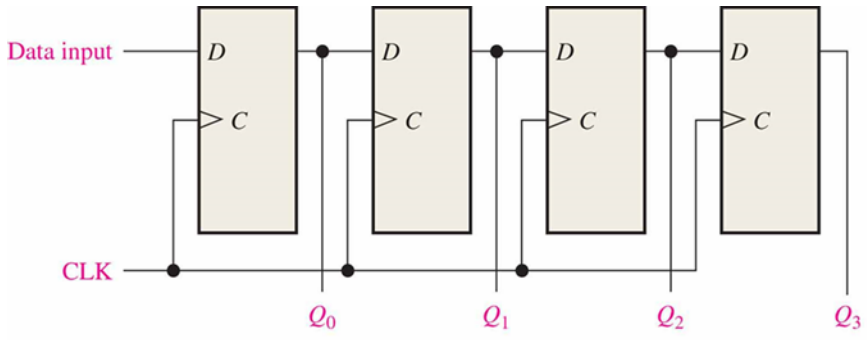
Eventhough it stores 8 bits, the register still only has one output from the last FF in the circuit

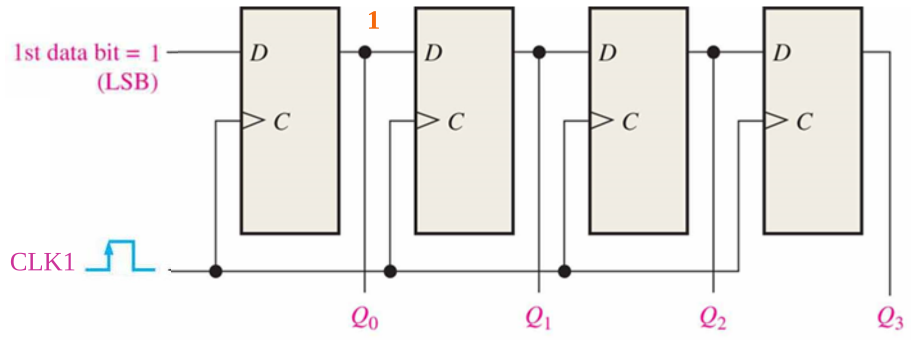
Serial In/Parallel Out Registers

- Data bits are entered serially from LSB first into this type of register similar to the earlier register
- Difference is the way bits are taken out from the register – the output of all stages are available simultaneously on its own respective output line.
  - An application of this type of shift register is the conversion of serial data to parallel form

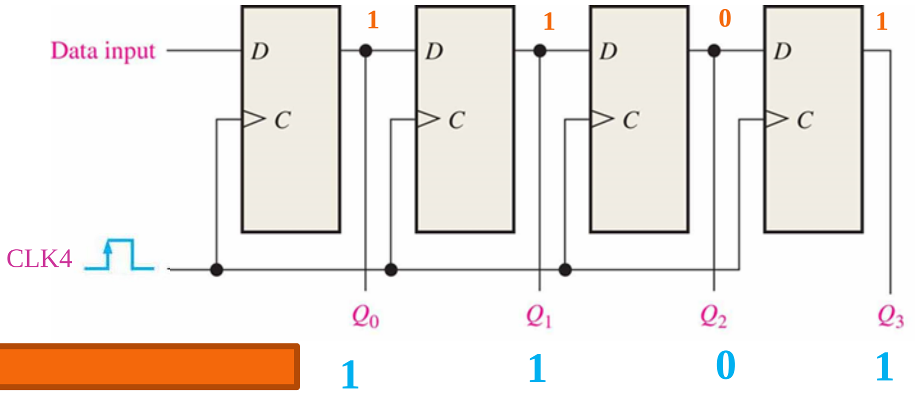


Bits to store = 1101

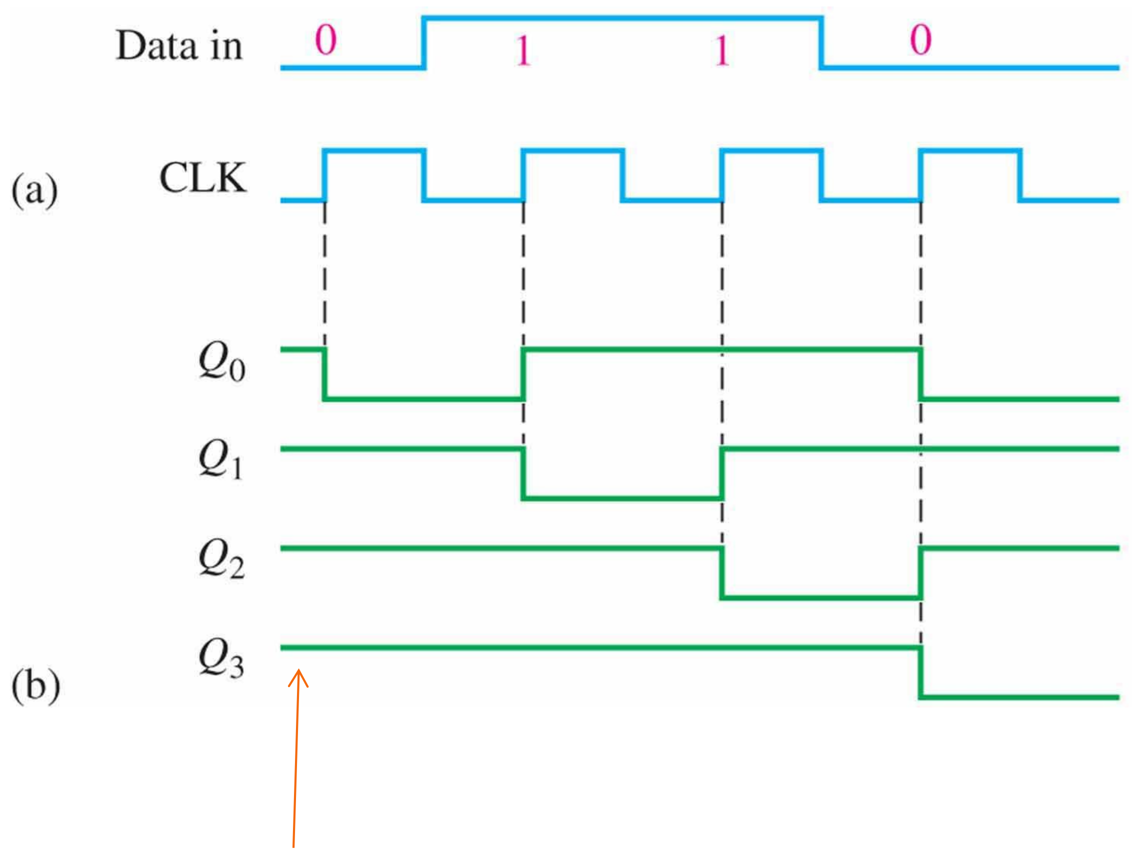


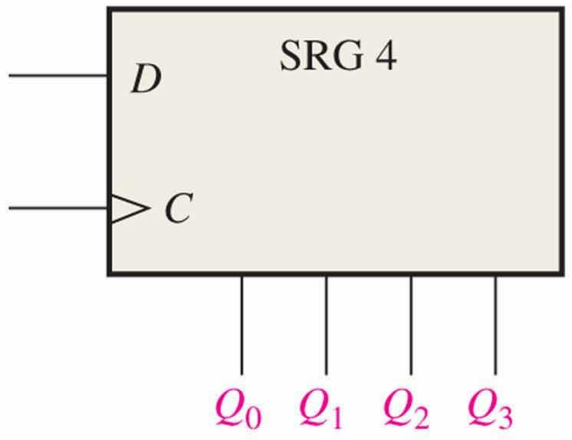


4-bits transferred in similar to serial I/O shift register



After 4 clock pulses, all 4 bits of the data is immediately available at the parallel output

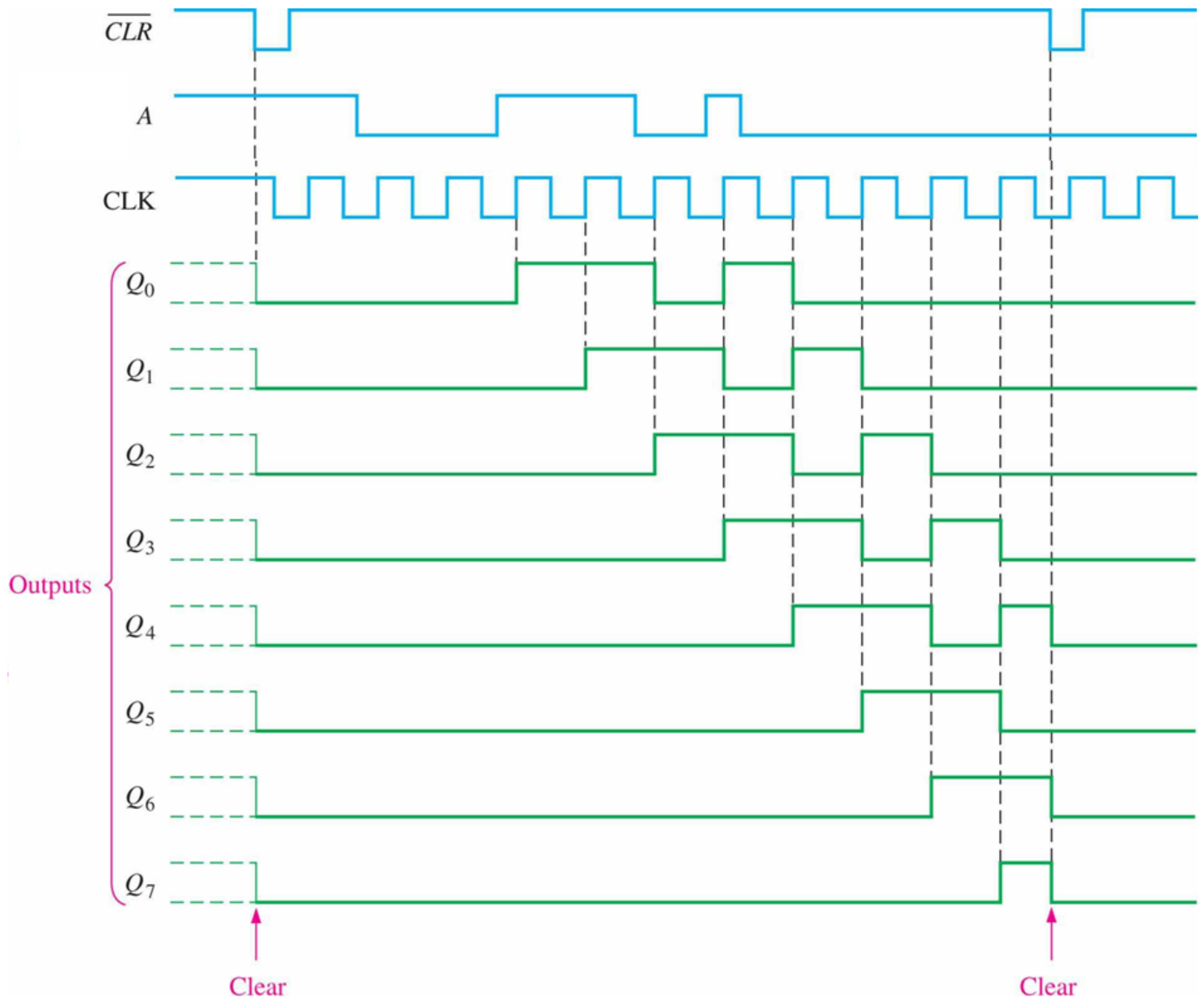




In this example, the register starts with having all 1s in each stage

Exercise

- Complete the timing diagram to show how an 8-bit serial in/parallel out shift register stores/retrieves the following bits

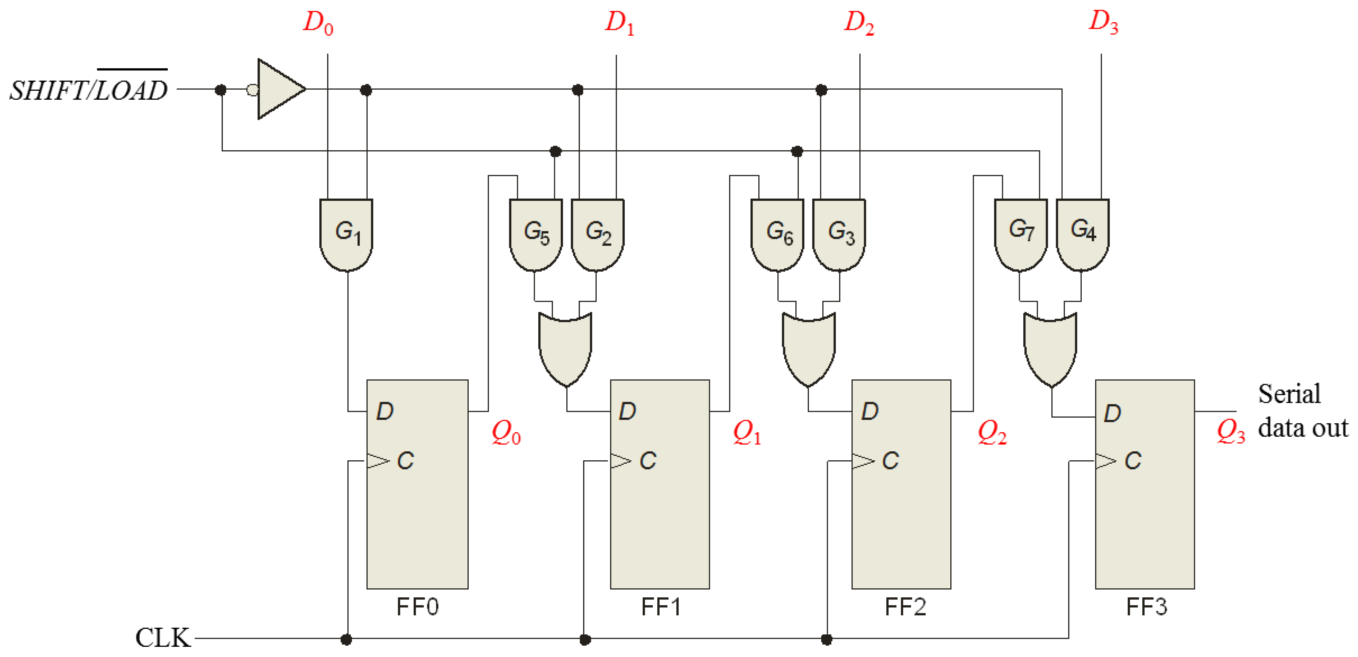


Parallel In/Serial Out Registers

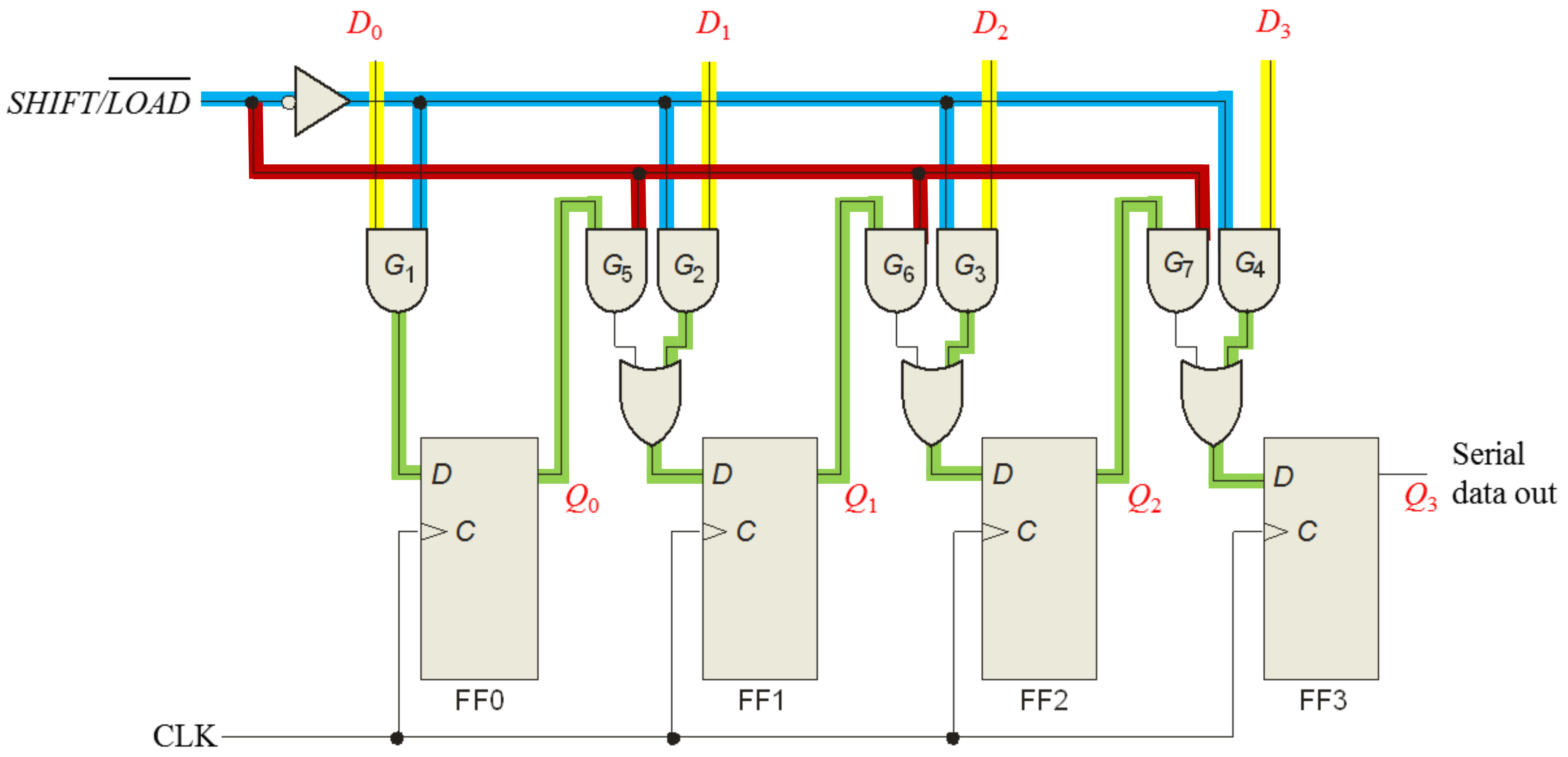
- For a register with parallel data inputs, the bits are entered simultaneously into their respective stages rather than one-by-one using one line inputs as with serial data input
- The serial output is the same as described earlier
- This type of register can be used to convert parallel data to serial form
- An example of a 4-bit parallel in/serial out register is shown on the following slides alongside its operation

Parallel In/Serial Out Registers

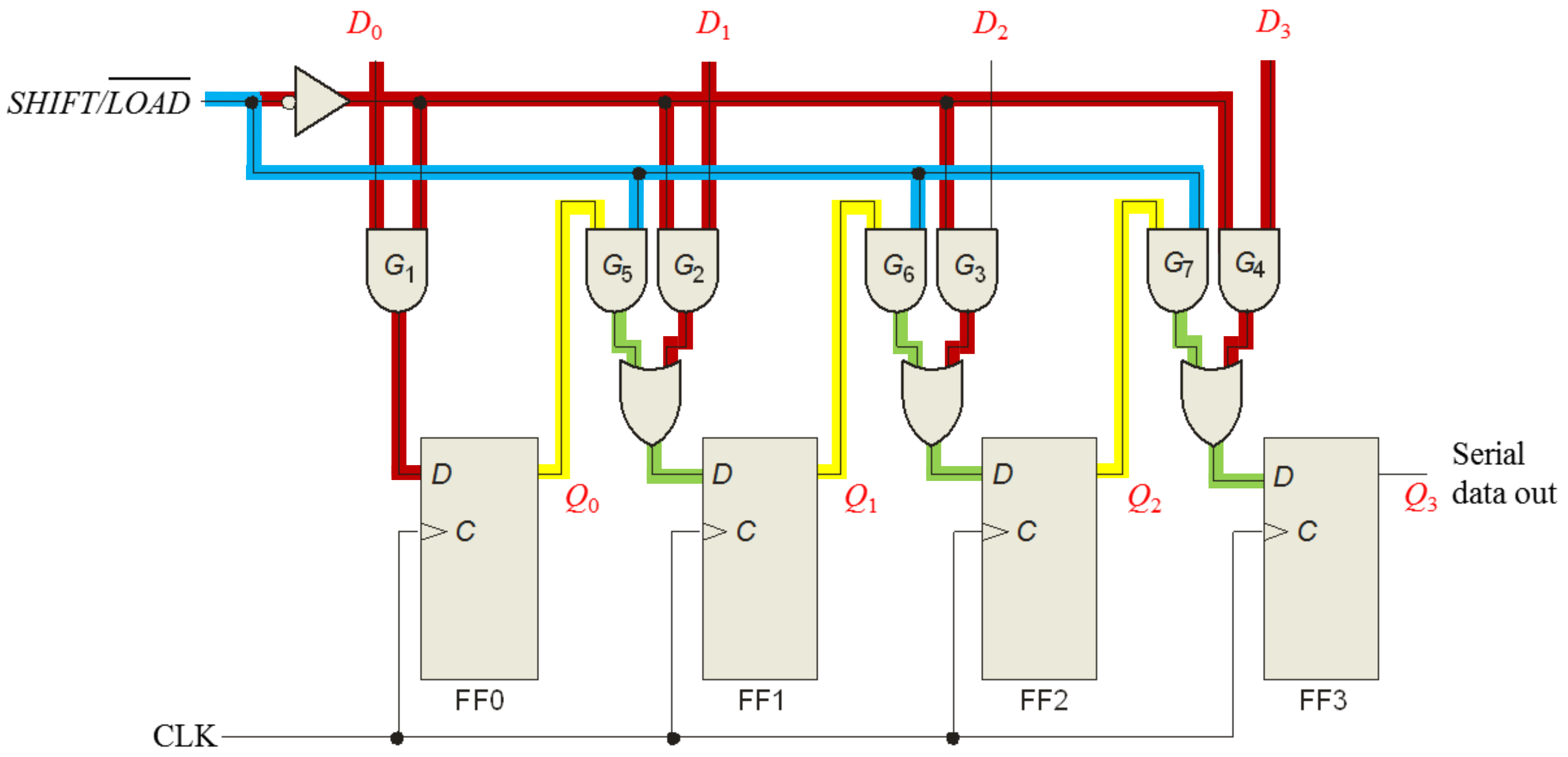
- A logic diagram for this type of register is shown below
  - The 4-bit parallel in/serial out register has 4 data input lines D_0, D_1, D_2 and D_3 along with a SHIFT/LOAD input
  - The OR gates allow either the normal shifting operation or parallel data-entry operation, depending on which AND gates are enabled by the level on the SHIFT/LOAD input



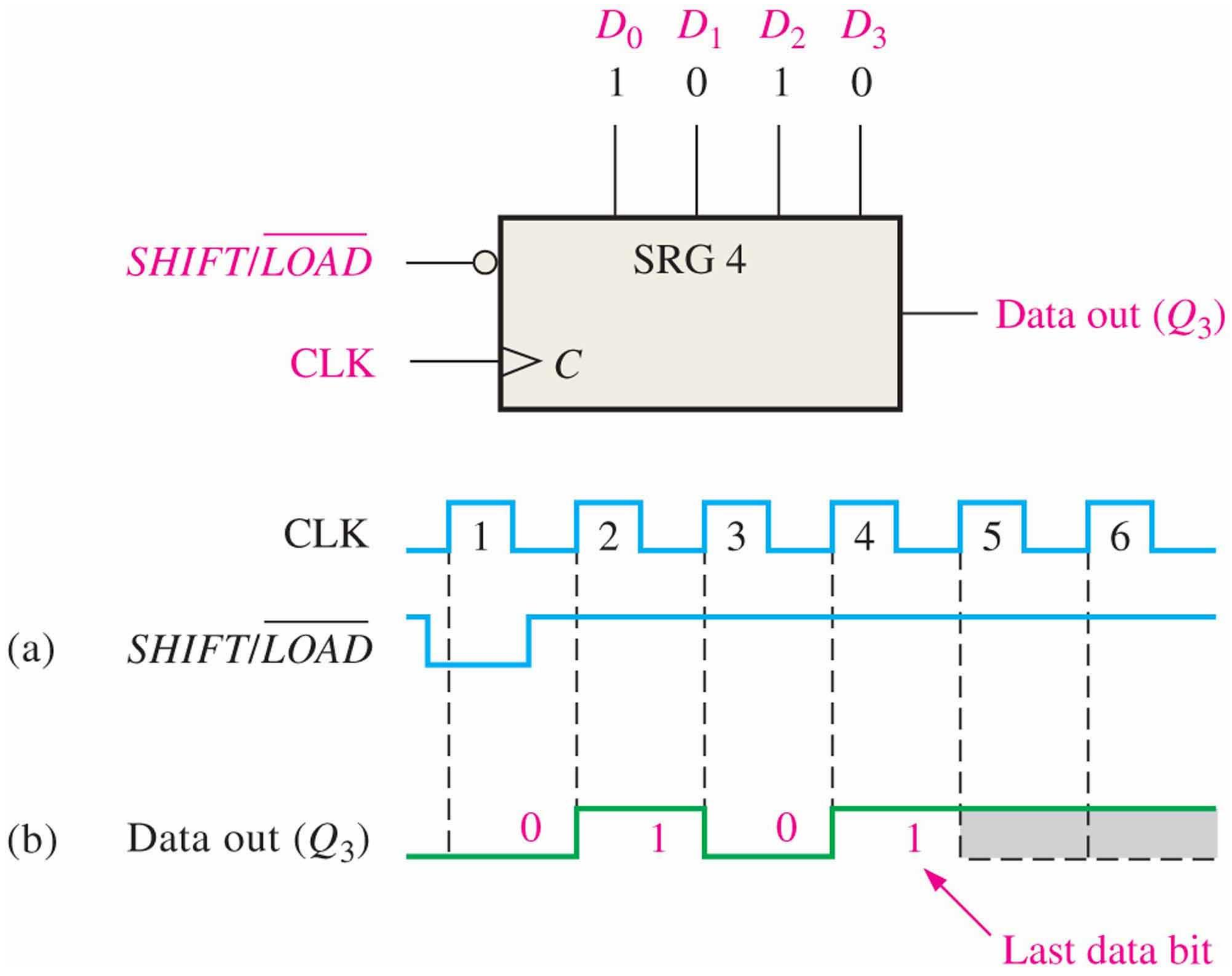
- When SHIFT/LOAD is LOW, gates G1 through G3 are enabled thus allowing the data bit signals to be applied to the D inputs of each respective FF
- When a clock pulse is applied, the flip-flops with D=1 will be set and those with D=0 will be reset, thereby storing all four bits simultaneously



- When SHIFT/LOAD is HIGH, gates G1 through G3 are disabled and G5 through G7 are enabled, allowing the data bits to shift right from one stage to the next
- It will take 4 CLK cycles to take the data out from the register – with each pulse cycle shifting the bits one stage towards Q_3

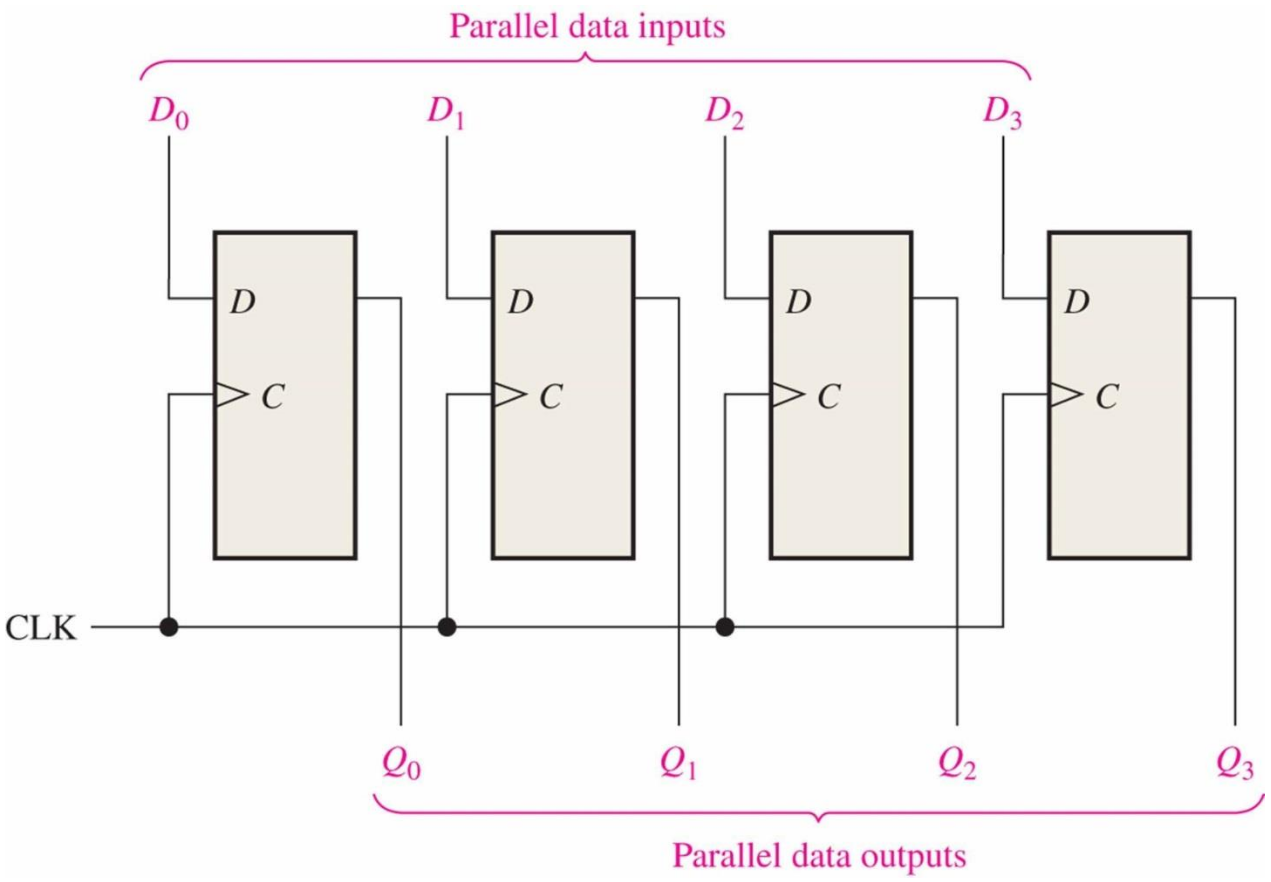


Parallel In/Serial Out Registers



Parallel In/Out Registers

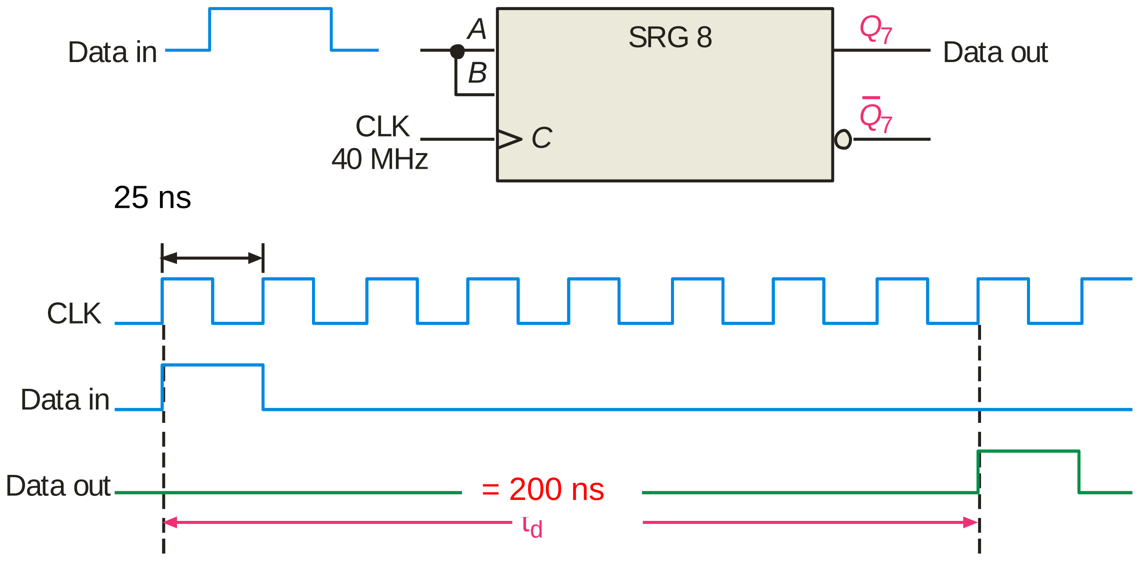
- As before, the parallel in/out registers stores and transfers multibit data in/out of the register in parallel
- Immediately following the simultaneous entry of all data bits, the bits appear on the parallel outputs



Applications of shift registers

- Shift registers can be used to delay a digital signal by a predetermined amount
- Example
  An 8-bit serial in/serial out shift register has a 40 MHz clock. What is the total delay through the register?

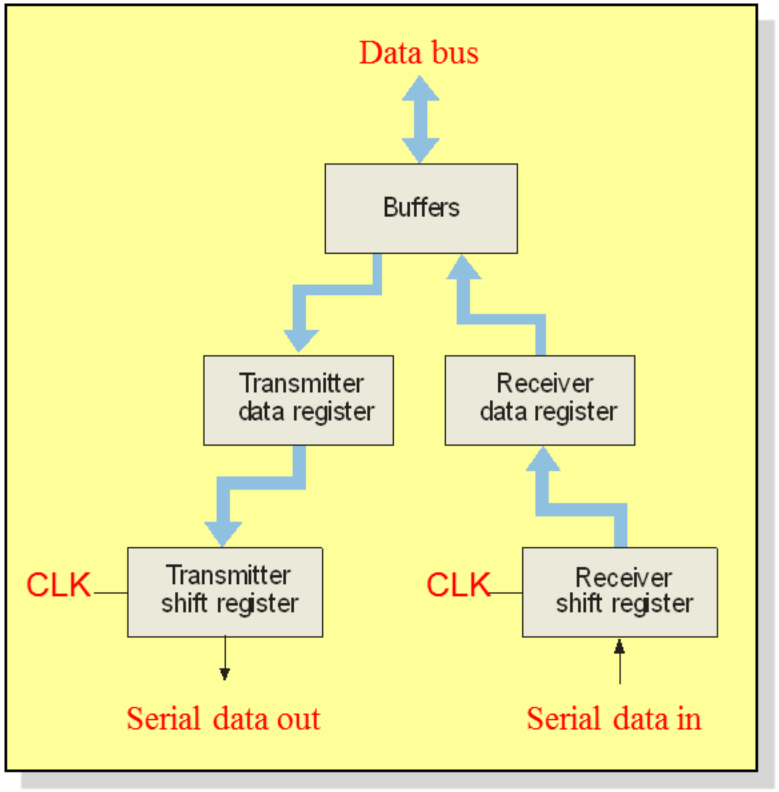
The delay for each clock is
1/40 MHz = 25 ns



The total delay is
8 x 25 ns = 200 ns

UARTs

- A UART (Universal Asynchronous Receiver Transmitter) is a serial-to-parallel converter and a parallel to serial converter.
- UARTs are commonly used in small systems where one device must communicate with another.
- Parallel data is converted to asynchronous serial form and transmitted.

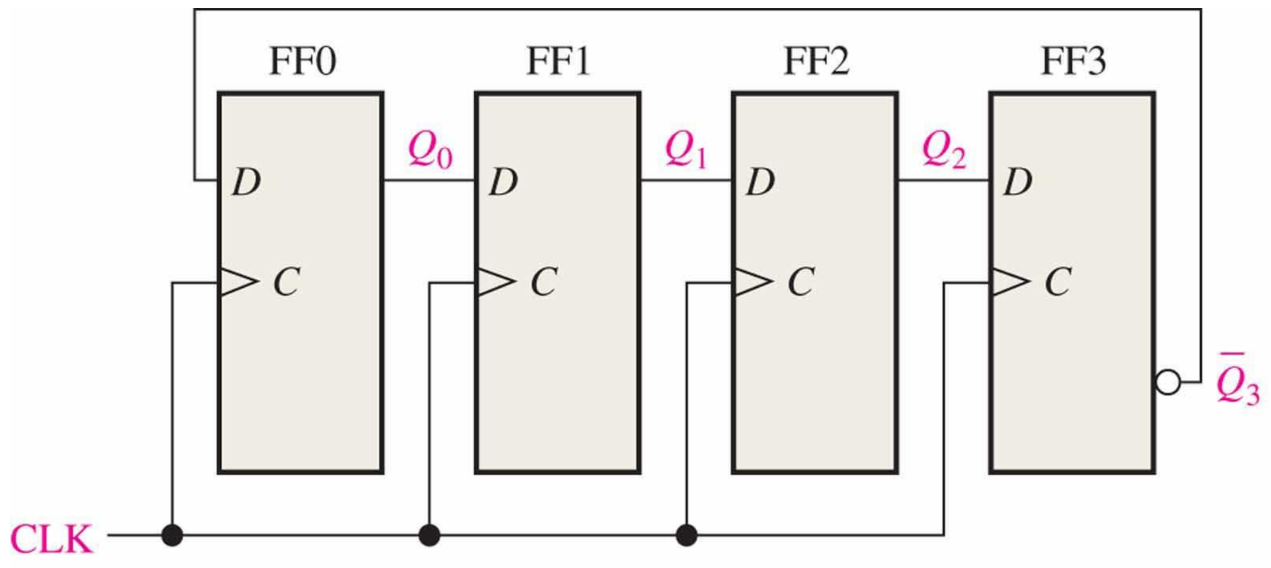


Registers as counters

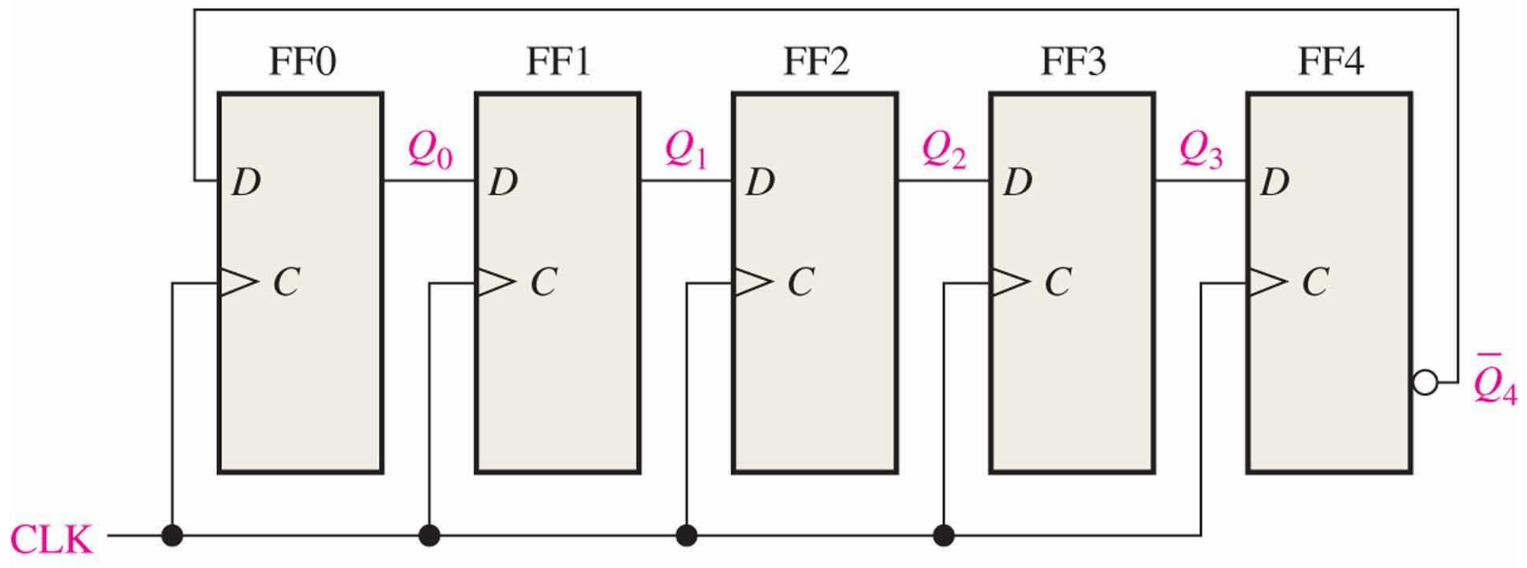
- Shift registers can form useful counters by recirculating a pattern of 0’s and 1’s
  - A shift register counter is basically a shift register with its **serial** output connected back to the serial input of the same register
- Two important shift register counters are the *Johnson counter* and the *ring counter*
- The Johnson and ring counter cycle through states of a particular sequence (unlike the numerical order seen in earlier counters)

Johnson counter

- The Johnson counter can be built using a normal shift register of D flip-flops with its *complemented* output of the last FF fed back to the D input of the first FF
  - Note: It can also be built with other FFs if necessary
- The next slide shows a typical arrangement in the JC
- If the counter starts at zero, this feedback arrangment produces the characteristic sequence of states of a Johnson counter
- The timing diagrams on the following slide show the sequence of a 4- and 5-bit Johnson counter
  - Note that a 4-bit JC creates 8 states whilst the 5-bit JC creates 10 states → the JC produces modulus of 2n states where n is the number of stages in the counter
  - Because of the fed back complemented output, the counter ‘starts’ itself upon application of the clock pulse

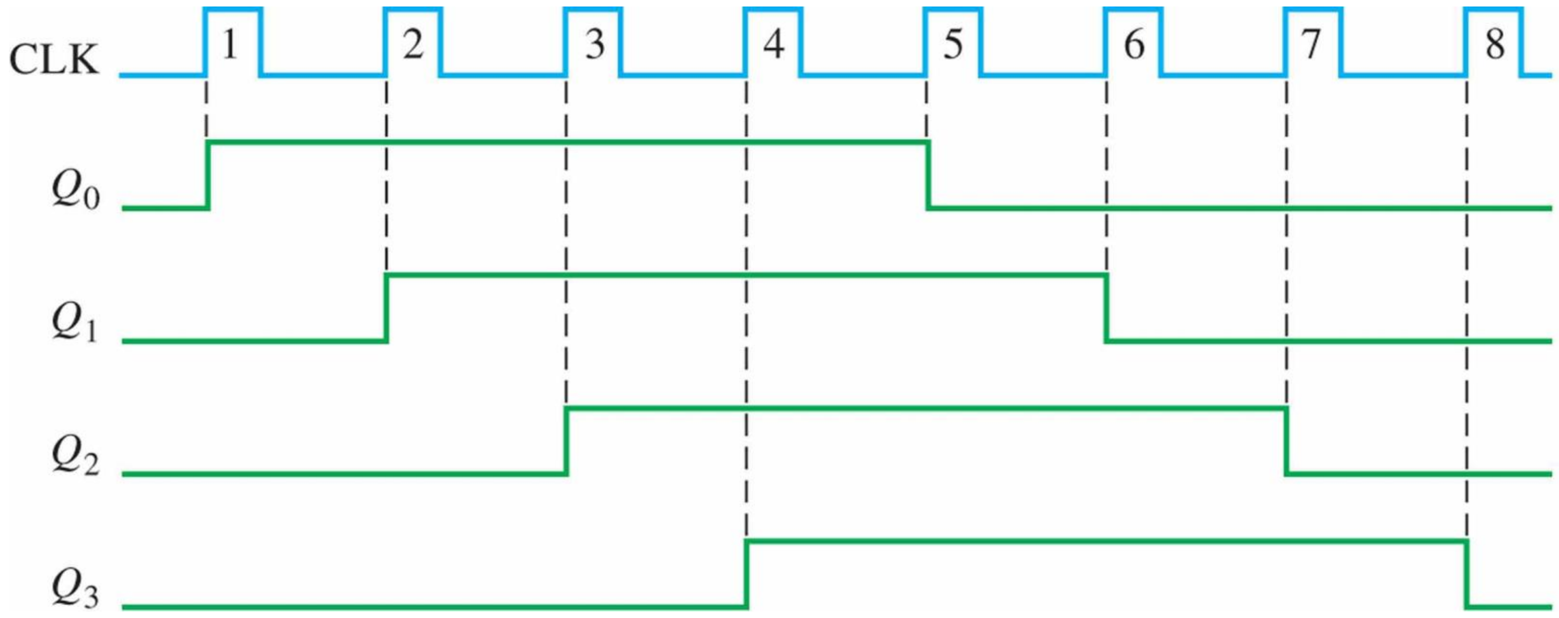


(a) Four-bit Johnson counter

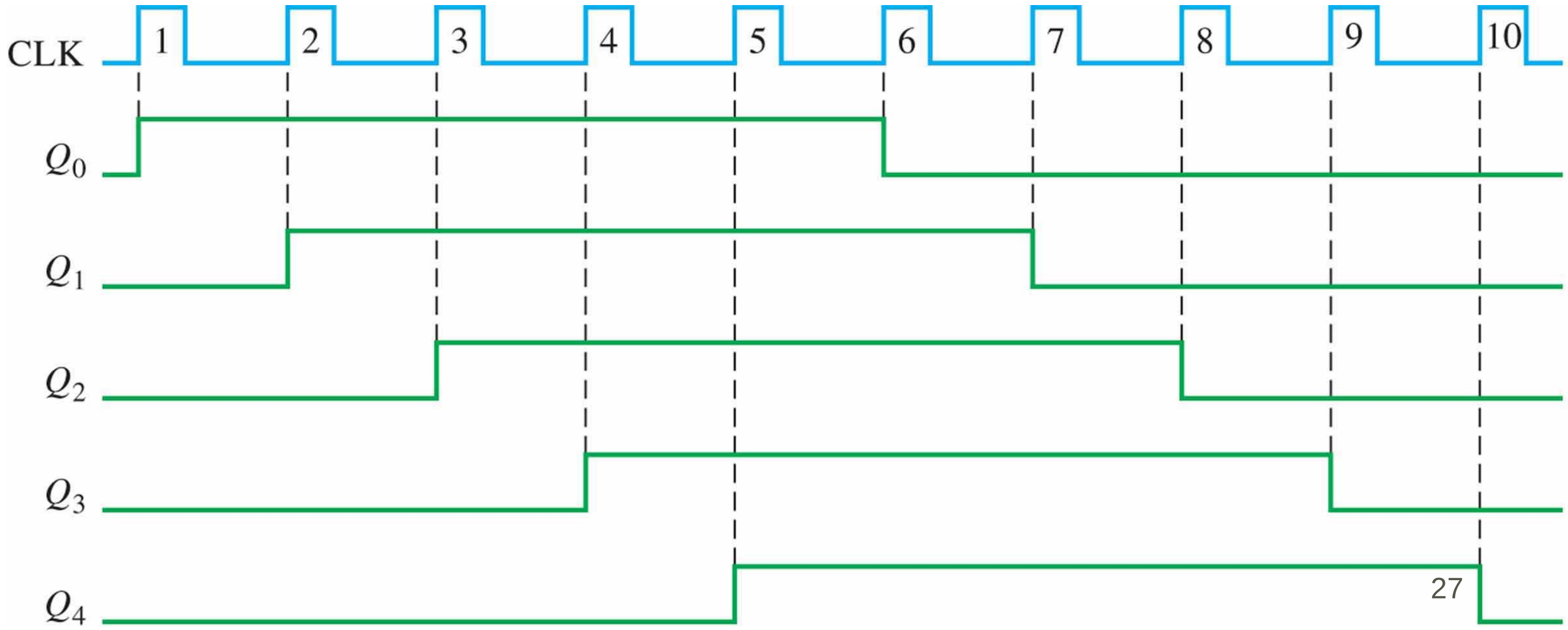


(b) Five-bit Johnson counter

4-bit Johnson counter

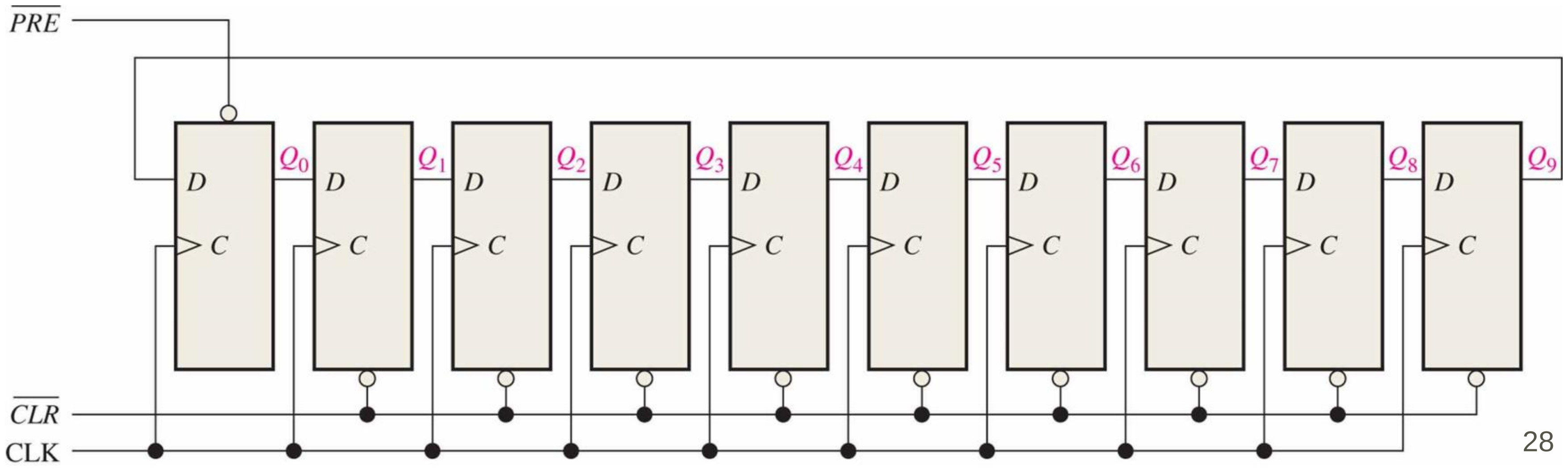


5-bit Johnson counter



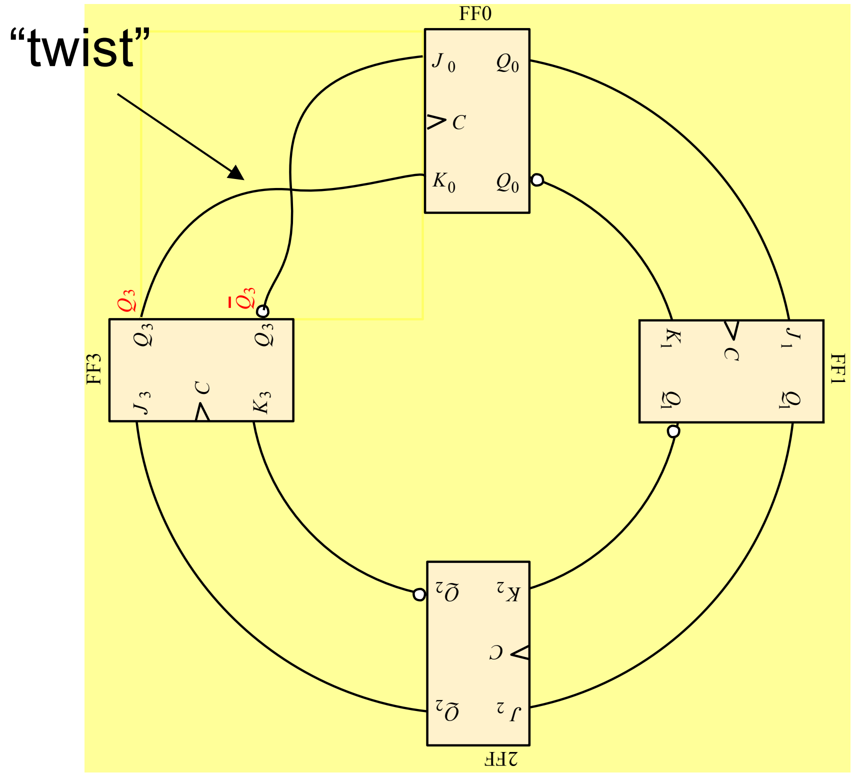
Ring counter

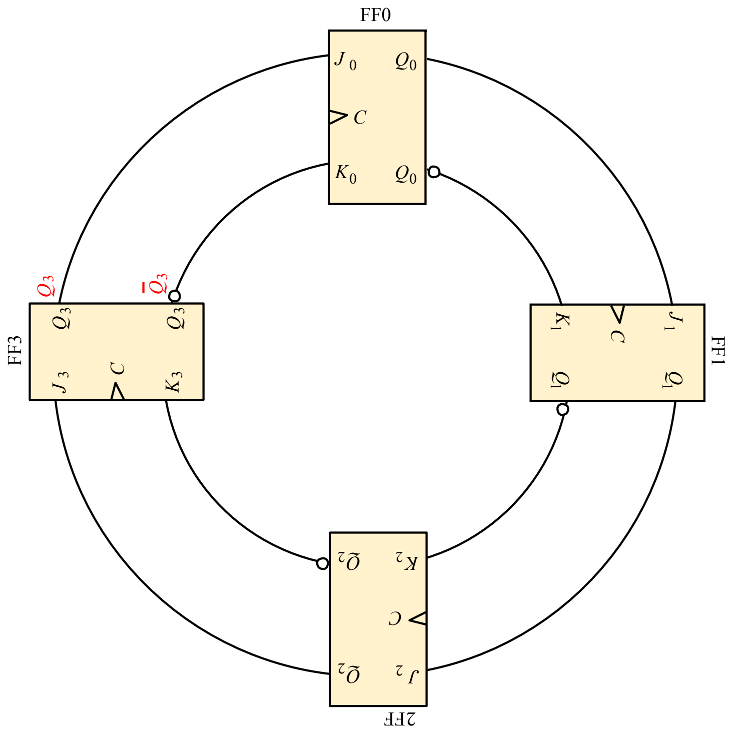
- Similar to the JC, the ring counter also cycles through a specific sequence of states
  - However, unlike the JC, the RC only uses one (1) FF for each state in its sequence → This means each state in the counter is indicated by a particular FF which can be used as an output/trigger
  - So a 10-stage RC has 10 states, 5-stage has 5 states etc.
- The logic diagram of a typical 10-bit RC is shown below
  - Note that the output is also fed back similar to the JC but it is Q instead of Q'



Why is it called ring counter?

- Redrawing the Johnson counter (note CLK is not shown) illustrates why it is sometimes called as a “twisted-ring” counter
- Whereas the ring counter resembles a “ring” – hence the name

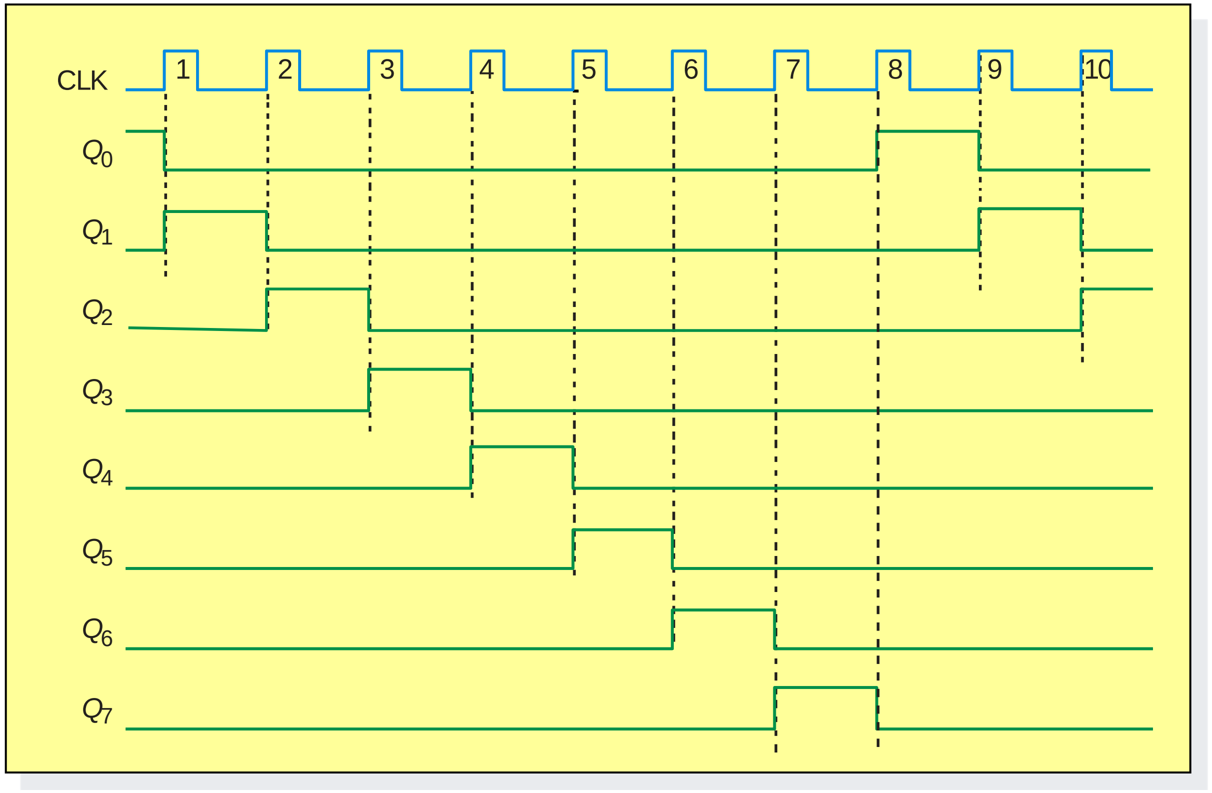




- Note: diagrams show JK FFs to better illustrate the *twisted* effect

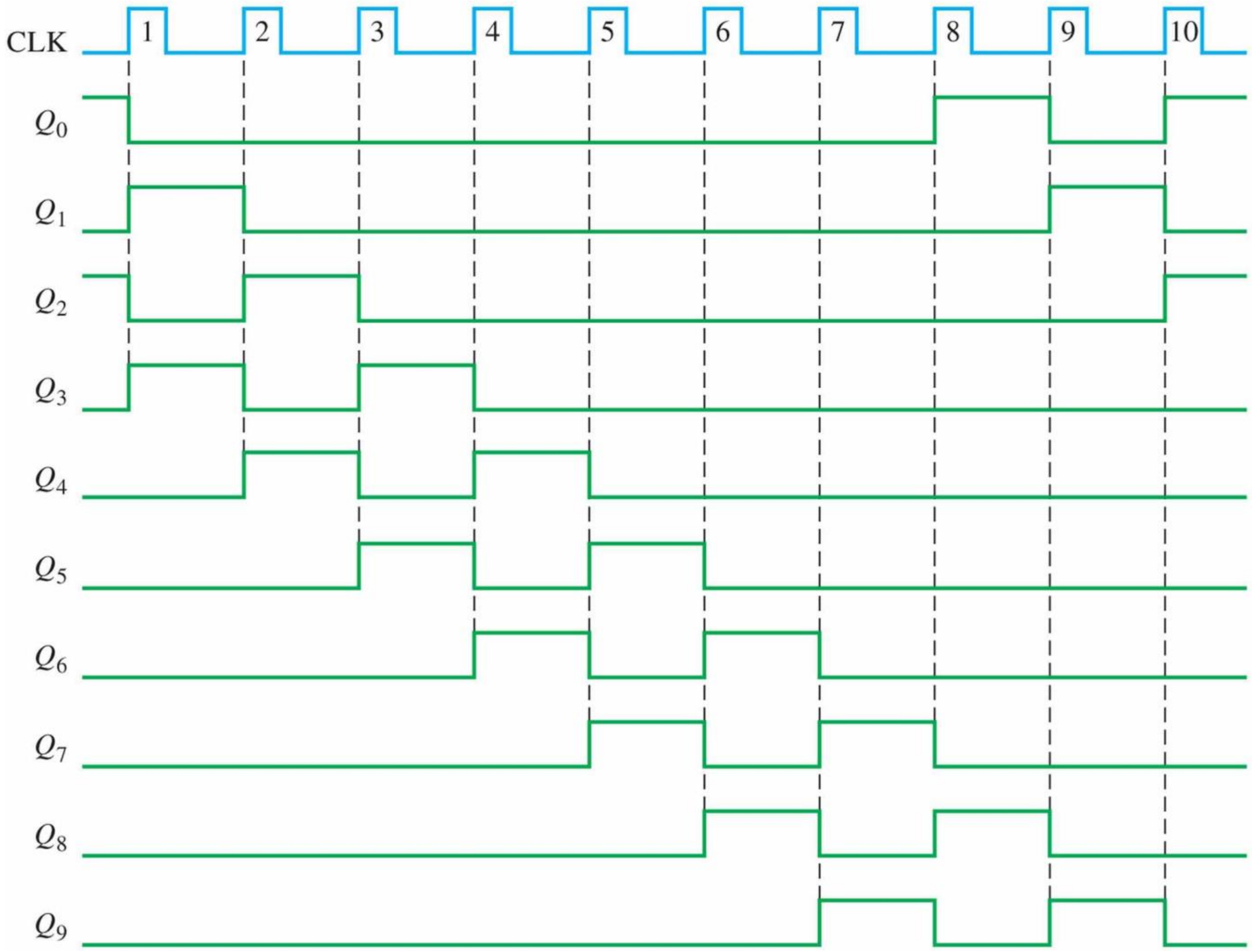
Ring counter

- Initially a 1 is preset into the first FF and the rest of the FFs are cleared
- Each time a CLK pulse is applied, the '1' shifts around the ring counter
  - Example: In a 10-bit RC, the 10 outputs of the counter can indicate directly the decimal count of the clock pulse
  - For example, a '1' on Q0 represents zero, '1' on Q1 represents one and so on



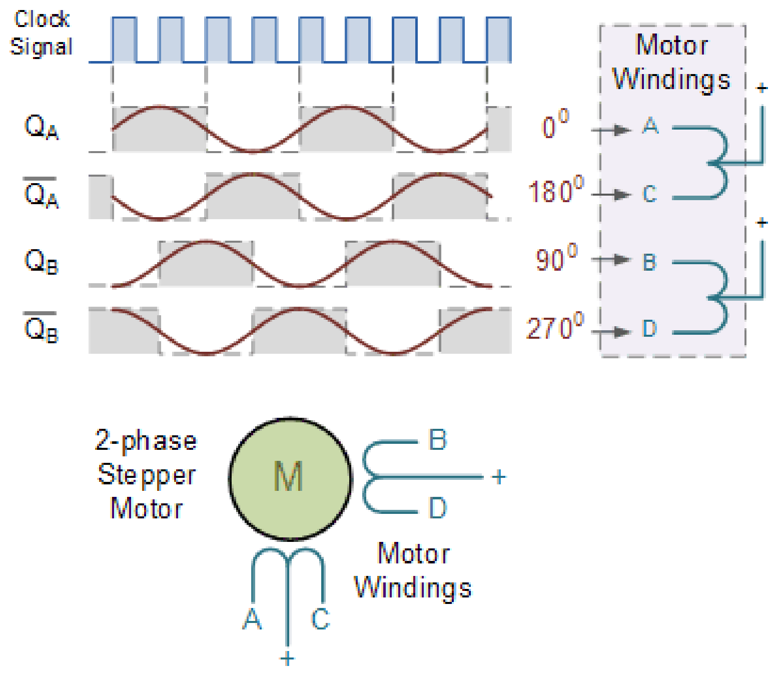
Exercise

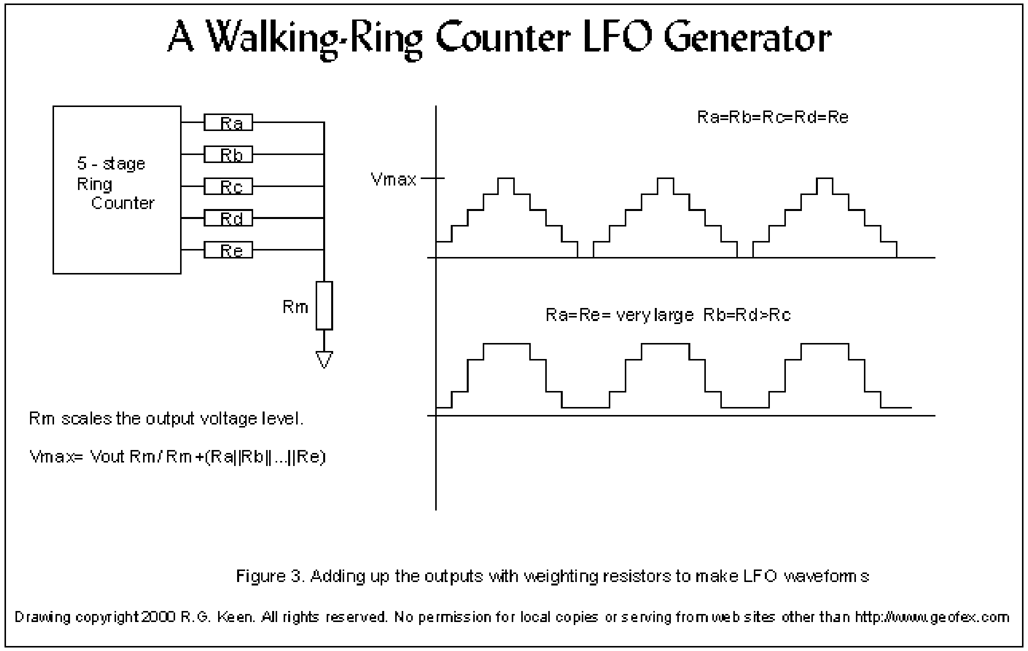
- If a 10-bit ring counter has the initial state 1010000000 determine the waveform in the timing diagram below for each of the Qx outputs



Uses of register counters

- Although similar to regular counters that can be used to 'count', register counters have a unique capability in certain applications. Example:
  - Control of stepper motors
  - Simplie sine-wave generator





A Walking-Ring Counter LFO Generator
5 - stage Ring Counter
Ra
Rb
Rc
Rd
Re
Rm
Vmax
Ra=Rb=Rc=Rd=Re
Ra=Re= very large Rb=Rd>Rc
Rm scales the output voltage level.
Vmax= Vout Rm/ Rm+(Ra||Rb||...||Re)

Figure 3. Adding up the outputs with weighting resistors to make LFO waveforms

Summary

- Shift registers can store and read data in serial or parallel mode
- Shift register counters are shift registers with feedback and exhibit special bit sequences
- The JC has 2n states in its sequence where n is the number of stages
- The RC has n states and n stages