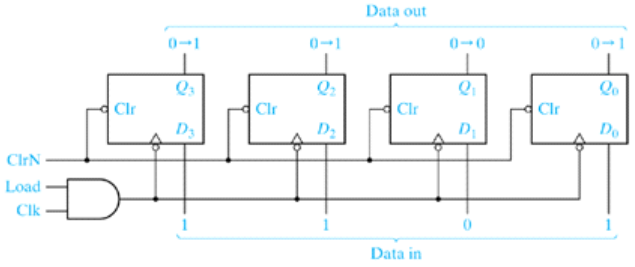
Topic 15: Modular sequential logic: registers

Thursday, January 31, 2019 4:45 PM

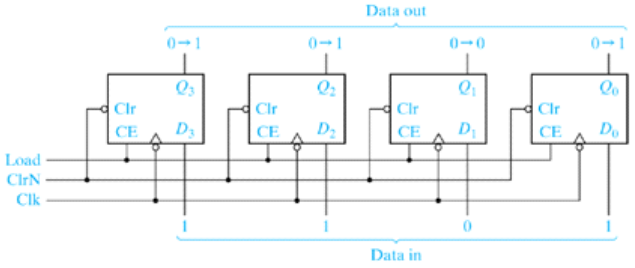
I. Registers

1. **Definition:** a synchronous sequential circuit that will store and move n-bit data, either serially or in parallel, in n flip-flops

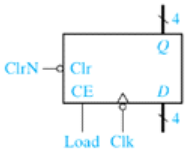
   e.g. 4-bit D flip-flop with clear, load and clock inputs



(a) Using gated clock



(b) With clock enable

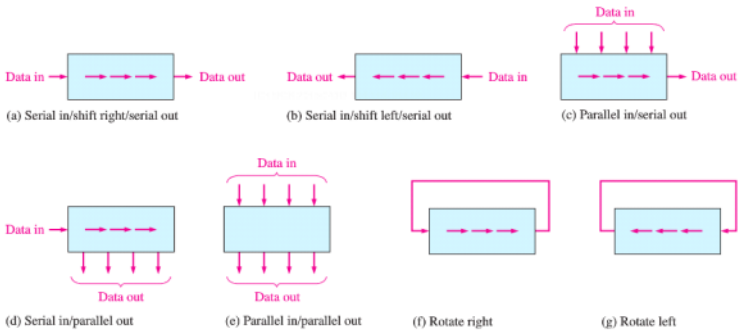


(c) Symbol

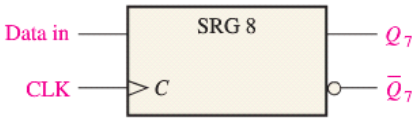
II. Shift registers

1. **SRGn:** n-bit shift register. e.g. SRG4 indicates a 4-bit shift register
2. **Type of shift register:**
   a. **Serial shifting:** movement of data from one end of a shift register to the other at a rate of one bit per clock pulse
   b. **Parallel shifting:** movement of data into all flip-flops of a shift register at the same time
   c. **Rotation:** in serial shift register, the output of the last FF is connected back to the synchronous input of the first FF. This results in continuous circulation of the same data
   d. Right shift, left shift, bidirectional shift register
   e. **Parallel-load shift register:** a shift register that can be preset to any value by directly loading a binary number into its internal flip-flops
   f. **Universal shift register:** a shift register that can operate with any combination of serial and parallel inputs and outputs

      e.g. serial in/serial out, serial in/parallel out, parallel in/serial out, parallel in/parallel out, and bidirectional operation

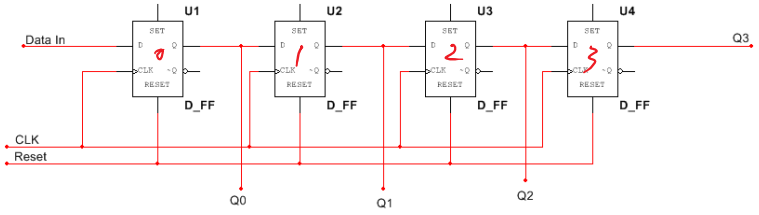


3. **Serial in/serial out shift register**
   - Symbol of a serial in/serial out SRG8

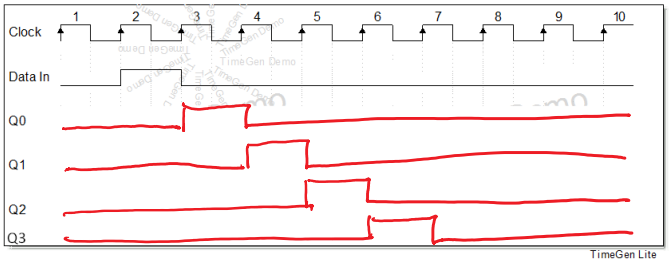


Commercial product: SN7491A -- 8-bit serial-in, serial-out shift register

   - Diagram of a 4-bit serial in/serial out shift register:



Q1: Given the Clock and data waveforms, draw the output waveforms of each flip-flop (Q0 - Q3)



Now thinking we have a series of data (1010) to store: start from the LSB

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**

To read the data out, another 4 clock pulse is needed

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**

Readout.

Data bits are indicated by a beige screen.

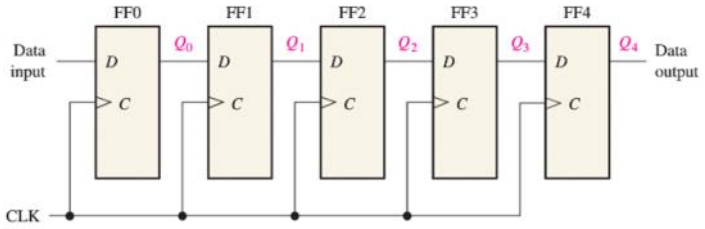
CLK	FF0 (Q_0)	FF1 (Q_1)	FF2 (Q_2)	FF3 (Q_3)
Initial	1	0	1	0
5	0	1	0	1
6	0	0	1	0
7	0	0	0	1
8	0	0	0	0

Readout.

Q2: in the following 5-bit serial in/serial out shift register (SGR5), a series of data (01011) is going to be stored (the rightmost bit goes in first).

a. **What data is stored in the output of each flip-flop (from Q_0 to Q_4) after 4 clock cycles?**

b. **If the data input is fixed to logic 1 after the 5-bit data all go into the register, what data is stored in the output of each flip-flop (from Q_0 to Q_4) after 7 clock cycle?**



4. **Serial in/parallel out shift register**

n clk cycle to store
1 clk cycle to read

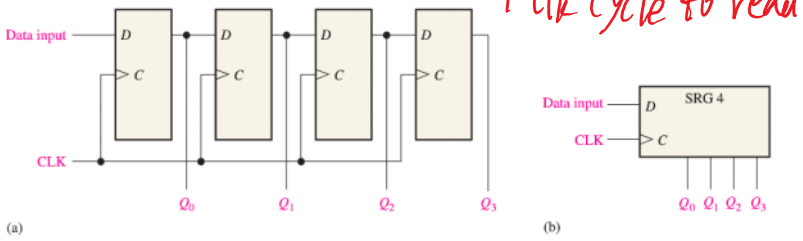
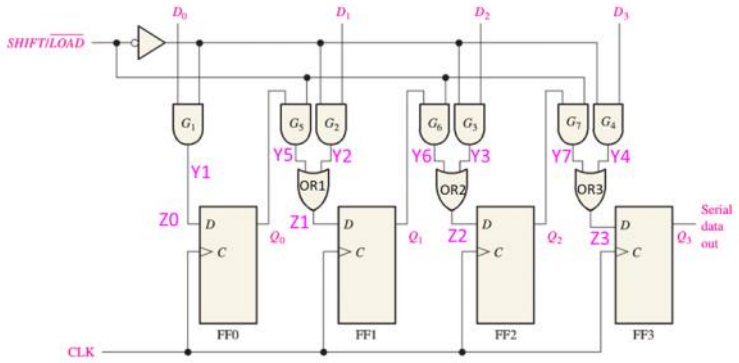


FIGURE 8–6 A serial in/parallel out shift register.

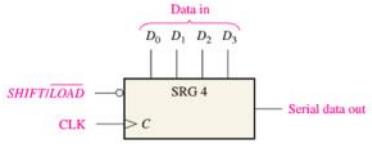
The store process is the same as serial in/serial out shift register

Commercial product: SN74164 -- 8-bit serial-in, serial/parallel -out shift register; SN796 -- 5-bit serial-in, serial-parallel - out shift register with asynchronous common clear and preset

5. **Parallel in/serial out shift register**



(a) Logic diagram

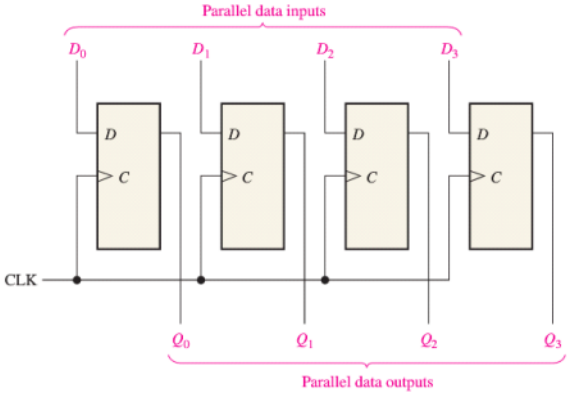


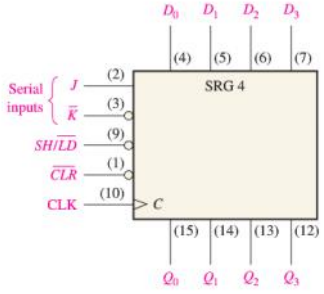
(b) Logic symbol

Q3: If SHIFT/$\overline{LOAD}$ = 0 , what are the outputs of the AND gates Y1~Y7? What are the outputs of the OR gates Z1~Z3? When a clock rising edge is coming, what are the outputs of the flip-flops Q0~Q3? Do you think the registers are doing loading or shifting function?

Q4: If SHIFT/$\overline{LOAD}$ = 1 , what are the outputs of the AND gates Y1~Y7? What are the outputs of the OR gates Z1~Z3? When a clock rising edge is coming, what are the outputs of the flip-flops Q0~Q3? Do you think the registers are doing loading or shifting function?

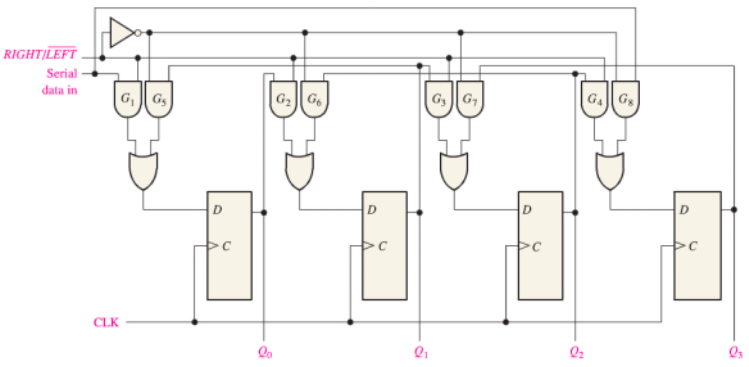
6. **Parallel in/parallel out shift register**



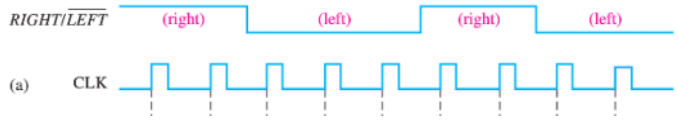


The 74HC195 4-bit parallel access shift register.

7. **Bidirectional shift register**

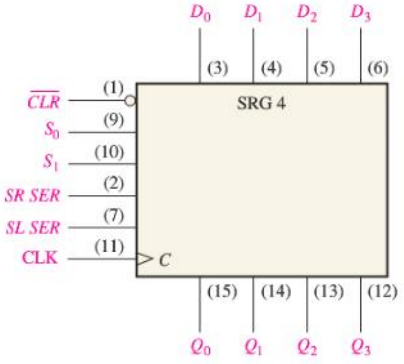


Q5: Determine the state of the bidirectional shift register after each clock pulse for the given control input waveform. Assume that Q0=1, Q1=1, Q2=0, Q3=1 and the serial data input is LOW.



Commercial product: SN74164 -- 4-bit bidirectional serial-in, parallel-out shift register

8. **Universal register -- 74HC194 4-bit universal shift register**



D Q*
0 0
1 1

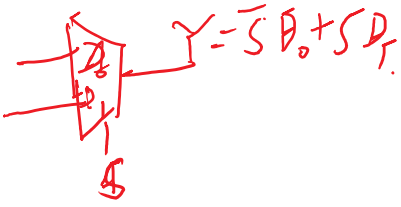
Q6: Design a 4-bit SRG with control input SH (Shift) and L (Load) using D flip-flops. If both SH and L are 0, the stored data in the SRG will not change; if L is 1, the SRG will do parallel loading from data input I3~I0; if SH is 1 and L is 0, the SRG should shift to the right.

shift R

a. Complete the state table

L	SH	Q3*	Q2*	Q1*	Q0*
0	0	Q_3	Q_2	Q_1	Q_0
1	X	I_3	I_2	I_1	I_0
0	1	0	Q_3	Q_2	Q_1

D3 D2 D1 D0.
Q3 Q2 Q1 Q0
I3 I2 I1 I0
0 Q3 Q2 Q1.



b. Generate the Boolean function for D3~D0

(Input circuit for FFs)

$D_3 = L'SH'Q_3 + LI_3$.
$D_2 = L'SH'Q_2 + LI_2 + L'SHQ_3$.
$D_0 = L'SH'Q_0 + LI_0 + L'SHQ_0$.

$D_3 = L'SH'Q_3 + LI_3$.

$D_2 = L'SH'Q_2 + LI_2 + L'SHQ_3$.

$D_1 = L'SH'Q_1 + LI_1 + L'SHQ_2$.

$D_0 = L'SH'Q_0 + LI_0 + L'SHQ_0$.

III. Shift register counter

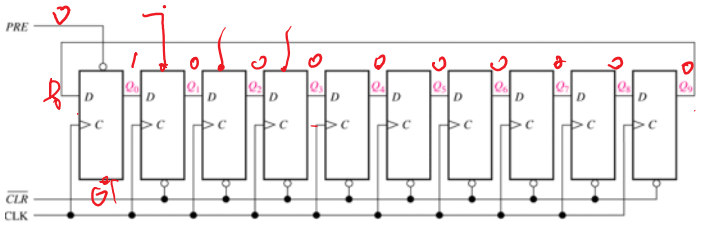
- **Counter:** a sequential circuit that has a number of binary outputs whose states progress through a fixed sequence, either ascending or descending or even nonlinear. **A counter goes back to its initial state after counting the last state.**

  e.g. 0, 1, 2, 3, 0, 1, 2, 3 …
  3, 2, 1, 0, 3, 2, 1, 0…
  1, 3, 5, 7, 1, 3, 5, 7….

- **Type I:** Ring counter

  Definition: a shift register with the serial output connected back to the serial input to produce special sequence

  e.g. 10-bit ring counter



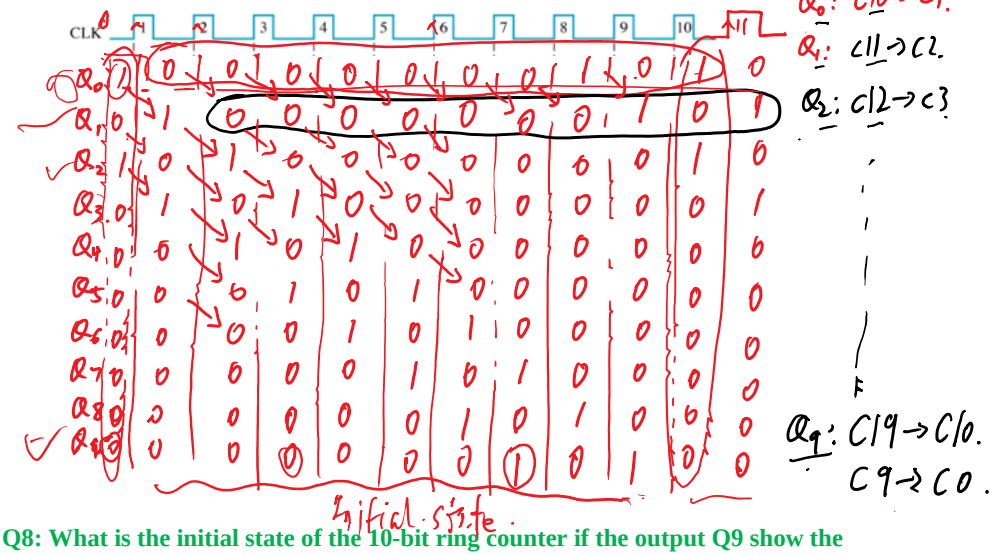
FFs.
↑
20-bit RC.
↑
20 state

10-bit RC
10 state

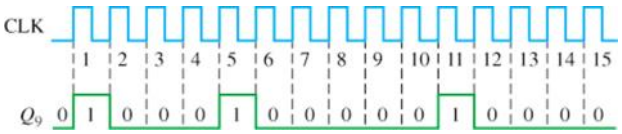
TABLE 8–5
Ten-bit ring counter sequence.

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

Q7: If a 10 bit ring counter has an initial state Q0:Q9 = 1 010 000 000 determine the count sequence for the Q outputs for the 1st 10 clock cycles.



Q8: What is the initial state of the 10-bit ring counter if the output Q9 show the following waveform



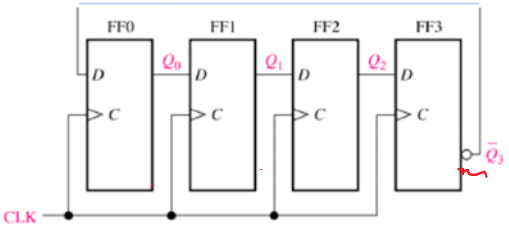
- **Type II:** Johnson counter

  The complement of the output of the last FF is connected back to the D input of the first FF.

  A n-bit (n-FFs) Johnson counter can realize modulus-2n counter

  e.g. 4-bit Johnson counter

4-bit Johnson → 8 state.



Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1

(4)bit Johnson counter → (8) states mod-8.

3-bit → 6 state.

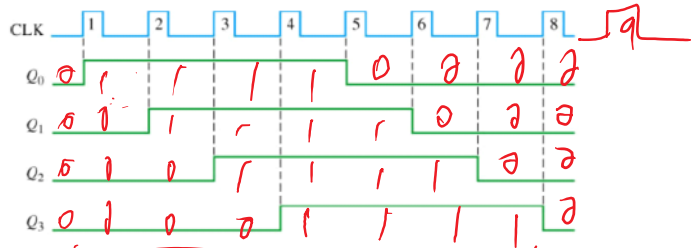
Q_0	Q_1	Q_2
0	0	0
1	0	0
1	1	0
1	1	1
0	1	1
0	0	1

n-bit → mod-2n.

n-bit → 2n state

↓

n-FFs.



n of states.

Q9: How many flip-flops are required to implement a:

a) **Modulus-6 Ring Counter** 6 FFs.

b) **Modulus-10 Johnson Counter** 5 FFs.

c) **Modulus-14 Ring Counter**

d) **Modulus-16 Johnson Counter**

e) **Modulus-6 Binary Counter**