K.T.S.P.Mandal's

Hutatma Rajguru Mahavidyalaya, Rajgurunagar

Tal-Khed, Dist.-Pune 410505.

F.Y.BSc (Computer Science)

Semester-II

Electronics Paper-II

Subject- ELC-122 – Basics of Computer Organization

According to new CBCS syllabus w.e.f.2019-2020

Prof.Aparna P.Kulkarni

Department of Computer Science

Hutatma Rajguru Mahavidyalaya, Rajgurunagar.

Unit-II- Shift Registers and Counters

- Shift Registers: SISO, SIPO, PISO, PIPO shift registers, Ring Counter using D flip-flop.
- Counters: Synchronous and Asynchronous type, 3 bit Up-Down counter, Concept of modulus Counters.

Introduction:

A Flip-flop stores a single bit data. However when we need to work with storage and manipulation of data words, then group of specific flip-flops need to be connected. A D flip-flop is capable of storing a single bit. However when several D flip-flop are connected together we can store several bits at the same time. Such a group of D flip-flop is called a register.

Shift Registers:

Shift Register is a group of flip flops used to store multiple bits of data. The bits stored in such registers can be made to move within the registers and in/out of the registers by applying clock pulses. Shift registers are very useful when the format of data needs to be changed (from serial to parallel or parallel to serial). Shift registers do not count. They use only D flip-flop.

Types of Shift Registers:

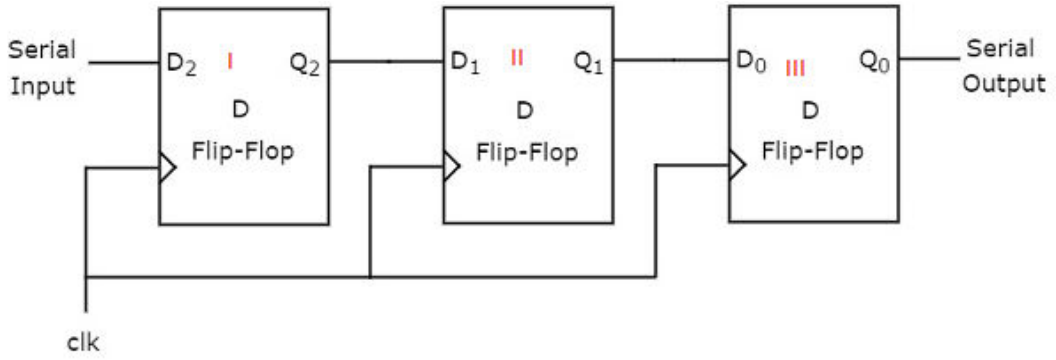
i. SISO (Serial In Serial Out)

ii. SIPO (Serial in Parallel Out)

iii. PISO (Parallel In Serial Out)

iv. PIPO (Parallel In Parallel Out)

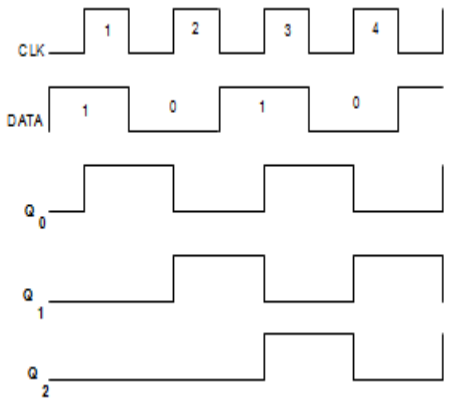
SISO Shift Register: (Right Shift)



S

Data is entered into flip-flop I. Data is shifted out of flip-flop III, after each clock pulse. Initially, all flip-flops are reset to 0. Suppose data is 1. After 1st falling edge of the clock $Q_2 = 1$. However Q_1 and Q_0 are 0. After 2nd falling edge $Q_2 = 1$, which acts as data to flip-flop (II), is shifted out to Q_1. So $Q_1 = 1$. After third clock pulse Q_1, which is 1, gets shifted out of Q_0.

If there is a string of databits at the input, then at subsequent clock pulses new bits will be accepted while previously accepted data gets shifted out continuously. T

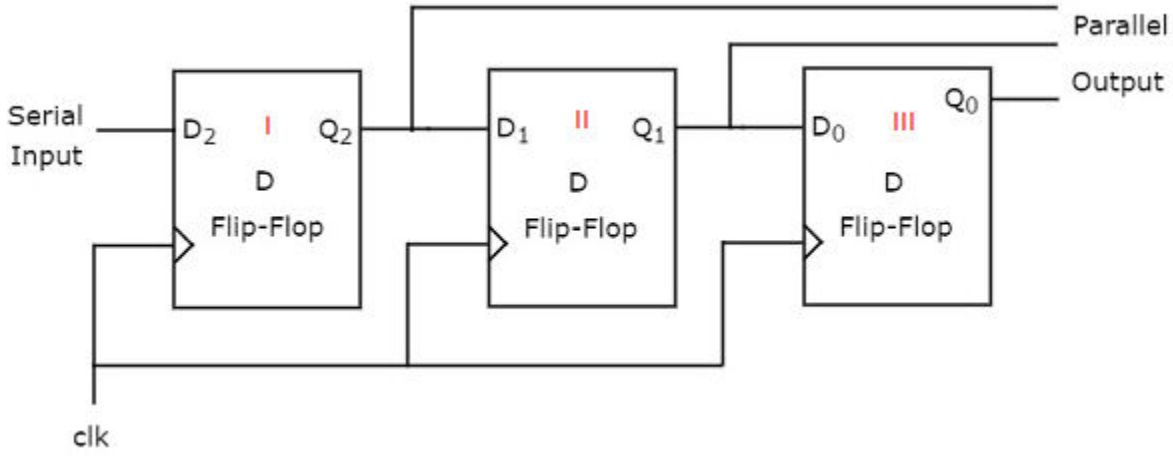


After clock	Data i/p	Q_0
1	1	X
2	1	X
3	0	1
4	X	1
5	X	0
6	X	X

SISO shift register will be the slowest of all registers. More the number of flip-flops in a register more will be the time taken for complete shifting of data. 'n' bits of data take '2n' seconds to be completely accepted, loaded and shifted out of the register. So the shifting time also depends on the number of bits to be accepted and shifted.

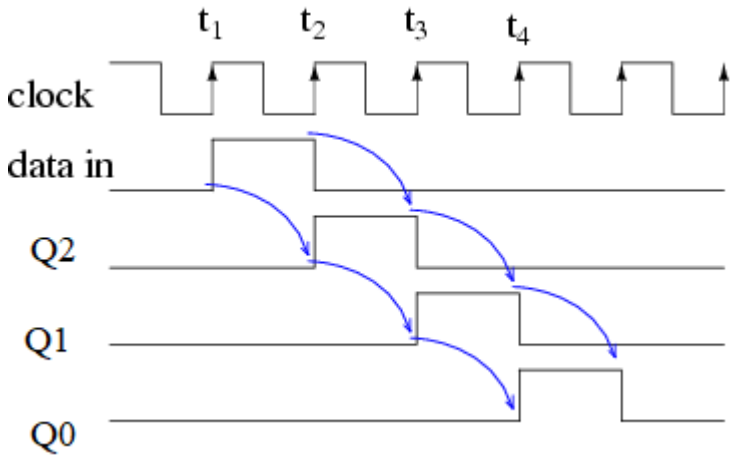
SIPO Shift Register:

Here data is entered serially bit by bit after each clock pulse. However all the outputs are seen simultaneously after each clock pulse. If we consider a 3 bit register, 3 clock pulses are required to load data, but at the end of 3rd clock pulse all the 3 bits of data are simultaneously available at the respective outputs. So it is called Serial in Parallel Out shift register.



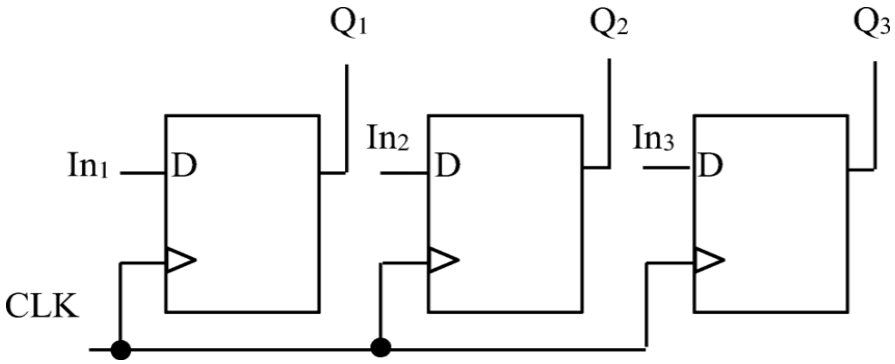
After Clock	Data i/p	Q_2	Q_1	Q_0	
1	1	1	0	0	

2	0	0	1	0	
3	1	1	0	1	→ Parallel data



PIPO Shift Register:

In this data is applied simultaneously to all the flip flops. The outputs of all the Flip-flops are viewed simultaneously. A single clock pulse is sufficient to load and Shift all bits of data. This is fastest type of shift register.

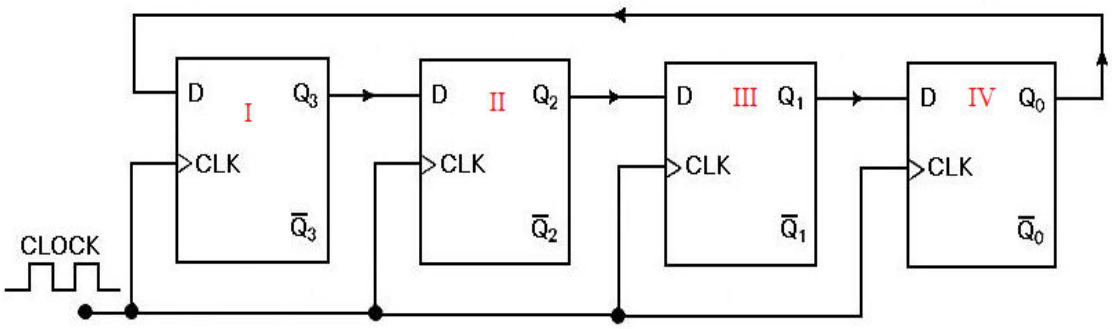


PISO Shift Register:

Here data is accepted parallely by all the flip-flops. However data moves serially through each flip-flop.

Ring Counter:

In a ring counter, the data once loaded will be continuously shifted within the flip-flops. No new data will be loaded; Output of last flipflop will act as data input to the first flipflop. Such arrangement is very useful for data sequencing and timing purposes and in applications like rolling display. They are also used to develop control section of a digital system.



Assume the data preloaded into the flip-flop as 0010 (I, II, III, and IV). After 1st clock pulse the data at the output of respective flip-flops are 0001 (Q_3 shifts to Q_2 , Q_2 shifts to Q_1 , Q_1 shifts to Q_0 and Q_0 shifts back to the Q_3). After 2nd clock pulse, the output data is 1000. After 3rd clock pulse, the output data is 0100 and after 4th clock pulse, the output data is 0010 and same movements will be repeated.

After clock	Loaded data			
	Q_3	Q_2	Q_1	Q_0
	0	0	1	0
1	0	0	0	1
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1

Applications of Shift Register:

- Shift register is used as Parallel to serial converter, which converts the parallel data into serial data. It is utilized at the transmitter section after Analog to Digital Converter ADC block.
- Shift register is used as Serial to parallel converter, which converts the serial data into parallel data. It is utilized at the receiver section before Digital to Analog Converter DAC block.
- Shift register along with some additional gates generate the sequence of zeros and ones. Hence, it is used as sequence generator.
- Shift registers are also used as counters. There are two types of counters based on the type of output from right most D flip-flop is connected to the serial input. Those are Ring counter and Johnson Ring counter.

Counters:

Counter is a sequential circuit. A digital circuit which is used for counting pulses is known counter. Counter is the widest application of flip-flops. It is a group of flip-flops with a clock signal applied. Apart from counting they can also be used for frequency division and sequencing. A high frequency clock pulse can be divided to get smaller frequency with counter.

Essentially, a counter employs JK flip-flop in toggle mode. The number of flip-flops decides the maximum events (clock pulses) that can be counted by a counter.

A single flip flop can count 2 states 1 and 0, 2 flip flops can count 00, 01, 10, 11, i.e., 4 states, 3 flip flops can count 000, 001, 010, 01, 100, 101, 110, 111, i.e., 8 states. So in general n flops can count 2^n states/clock pulses.

Counters are broadly classified as

1. Asynchronous or ripple counters.
2. Synchronous counters.

Further each counter can be sub classified as

1. Up counter
2. Down Counter

The up counter counts upwards with each clock pulse, i.e., 00 →99.

The down counter counts downwards with each clock pulse. The initial count is the biggest one, i.e., 99 → 00.

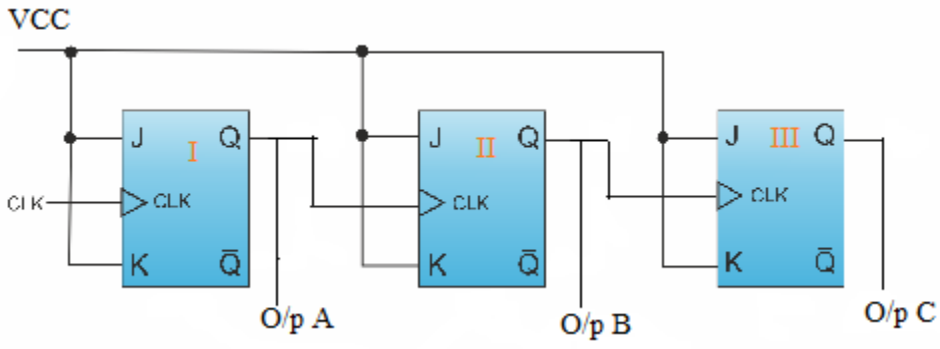
Many times both these facilities are combined together to get combined up down counter. Some counters also have provision for presetting or clearing the counts (Programmable counters).

Asynchronous or ripple counters:

This is also called a ripple binary counter as the clock is provided to each flip-flop just like ripples spreading in water.

The clock is connected only to the first flip-flop. The output of the 1st flip flop acts as clock to the second flip-flop. The output of 2nd flip-flop acts as clock to the third and so on. Thus second flip-flop can respond only after the first. Third flip flop can respond only after 1st and 2nd and so on. So we find that it is very slow. Moreover the speed further reduces due to addition of propagation delays of each flip-flop. This speed difference is considerable when a counter is used to count large sequences. These counters are used for low to medium frequency applications. However they are comparatively easy in construction using lesser hardware and have simpler operation.

3-Bit asynchronous Up Counter:



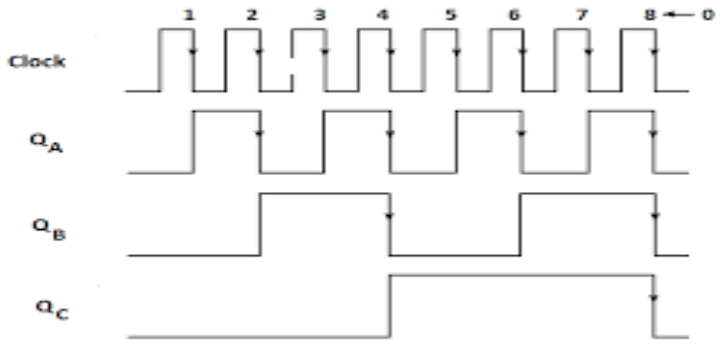
Let initially all flip-flops be reset to 0. So Q_C, Q_B, Q_A = 000. After 1st clock pulse's falling edge has arrived, flip-flop (I) toggles. So Q_A = 1. However Q_A which acts as clock to flip-flop (II) has changed from 0 to 1, (rising edge). So output of flip-flop (II) remains at 0.
Third flip-flop output also remains at 0. So after 1"clock pulse, Q_C, Q_B, Q_A = 001.

When 2nd clock pulse comes Q_A toggles again. It goes from 1 to 0. This acts as a falling clock edge for the flip-flop (II) so its output Q_B = 1. However Q_B which acts as clock to flip flop (III) was changed from 0 to 1. So flip-flop (Ill) maintains the previous output 0 as it is. After second clock pulse, Q_C, Q_B, Q_A =010.

When 3rd clock pulse arrives Q_A toggles back to 1. Q_B does not change because Q_A (clock for flip-flop (II)) has changed from 0 to 1. Q_C also does not change. After 3rd clock pulse we have Q_C, Q_B, Q_A = 011.

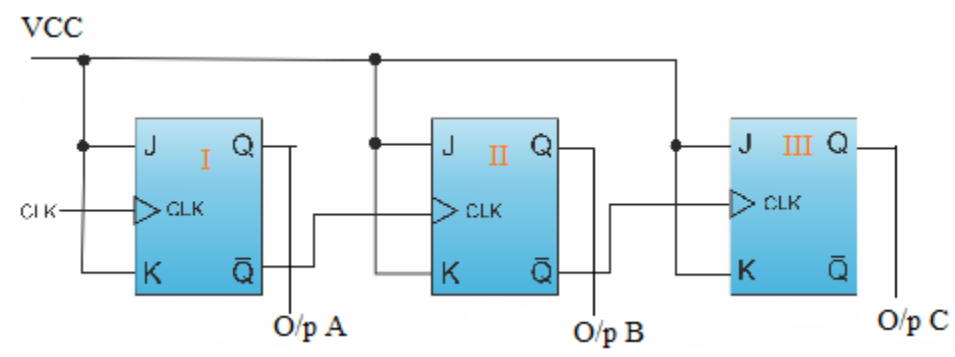
When 4th clock pulse arrives Q_A changes from 1 to 0. So now Q_B also changes from 1 to 0. This Q_B acts as clock to flip-flop (III). So now third flip-flop responds to the clock and its output changed to 1. So after 4th clock pulse, Q_C, Q_B, Q_A = 100. In this way, all the 8 states (000 to 111) are obtained. After 111, the counter automatically comes back to 000 and counting is repeated.

C	B	A	Counted state
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7
0	0	0	0



3-Bit asynchronous Down Counter:

Use $\overline{Q_A}$, $\overline{Q_B}$ and Q_C to act as clocks to the next flip-flops instead of Q_A Q_B and Q_C. The outputs will be seen at Q_A Q_B and Q_C.



Let Q_A Q_B Q_C be reset to 0. So Q_C $Q_B Q_A$ = 000. However Q_A, Q_B, $\overline{Q_C}$ = 111. When 1st clock comes, Q_A toggles from 0 to 1, so Q_A will go from 1 to 0. Flip-flop (II) receives a falling clock edge. So its output changes from 0 to 1. Since Q_B, has changed from 0 to 1, Q_B changes from 1 to 0. So flipflop (III) also receives a falling edge. Its output Q_C also goes to 1. So after 1st clock pulse, Q_C $Q_B Q_A$ = 111.

When 2nd clock pulse arrives, Q_A toggles from 1 to 0. Q_A change from 0 to 1. Flip-flops (II) and (III) will not change state. So after 2nd clock pulse, we have Q_C $Q_B Q_A$ = 110.

When 3rd pulse arrives, Q_A toggles back to 1. Q_A changes from 1 to 0. Now flip flop (II) responds. Its output Q_B changes from 1 to 0. However, flip flop (III) does not respond as Q_B has changed from 0 to 1. Thus after 3rd clock pulse we have Q_C $Q_B Q_A$ = 101.
In this way, the counter proceeds towards 000 after 8 clock pulses. In the next clock pulse, the output goes back to 111 and the same sequence of counting is repeated.

C	B	A	Counted state
0	0	0	0

Truth Table:

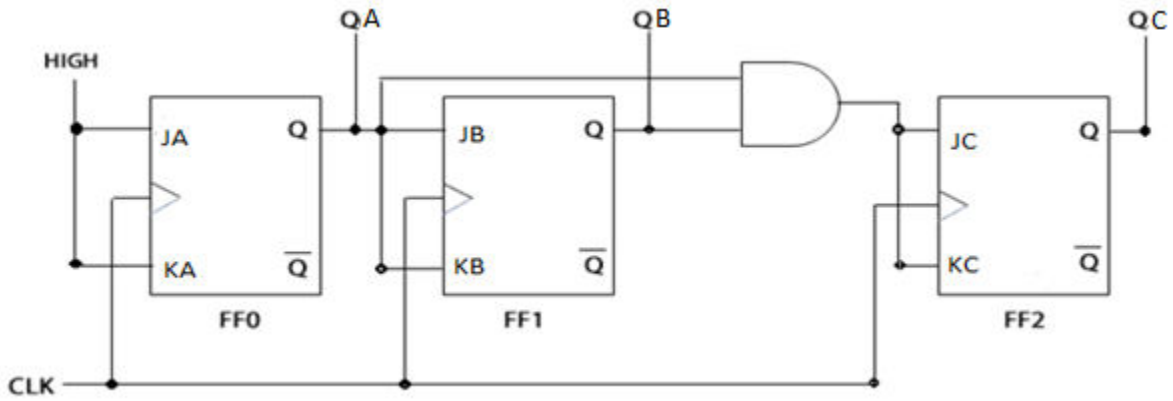
1	1	1	7
1	1	0	6
1	0	1	5
1	0	0	4
0	1	1	3
0	1	0	2
0	0	1	1
0	0	0	0

Note: Q_A toggles with the clock. Q_B changes only when Q_A changes from 1 to 0, i.e. Q_A changes from 0 to 1 (rising edge of Q_A). Similarly Q_C changes only when Q_B changes from 1 to 0 i.e. Q_B changes from 0 to 1(rising edge of Q_B).

Synchronous Counter:

They are also called as parallel counters as clock pulses are provided simultaneously to all flip-flops. All flip-flops are responds simultaneously and not one by one. As a result, the process of counting is speeded up. The total propagation delay here will be that of any one flip-flop as all of them respond at once. Circuit becomes more complex with addition of extra logic gates. These counters are used in high frequency applications.

3 Bit Synchronous Up Counter:



The first flipflop toggles for every falling edge of the clock as J_A and K_A both are connected to Vcc (Logic high). The 2nd flipflop will toggle when $Q_A = 1$ and clock goes from high to low simultaneously. The 3rd flipflop toggles when A and B both are high and clock goes from high to low simultaneously. Initially let ABC will be 000. At that instant flipflop (1) sees J_A and K_A as 1. At that instant flipflop (2) sees inputs as 0, i.e., J_B and K_B= 0 as Q_A was 0. Similarly Q_A and Q_B both were 0. So J_C and K_C both are 0. When the first falling edge of clock comes A = 1 B = 0 and C= 0 so we get 001. After the 1st clock pulse has passed, the inputs seen by each flipflop is as follows.

Flipflop (1) sees $J_A = K_A = 1$.

Flipflop (2) sees $J_B = K_B = A$ i.e. $J_B = K_B = 1$

Flipflop (3) sees $J_C = K_C = Q_A.Q_B = 1.0 = 0$

When the next falling edge of clock comes, A becomes 0, B = 1 and C = 0. (Previous state as $J_C = K_C = 0$) so we get 010.

After 2nd clock pulse the inputs seen by different flip-flops are as

Flipflop (1) sees $J_A = K_A = 1$

Flipflop (2) sees $J_B = K_B = A = 0$

Flipflop (3) sees $J_C = K_C = A.B = 0.1 = 0$

When the third falling edge of clock comes A = 1 (toggles), B = 1 (previous state), C = 0 (previous state). So we get 011.

After 3rd clock pulse has passed the inputs seen by different flip-flops are

Flipflop (1) sees $J_A = K_A = 1$

Flipflop (2) sees $J_B = K_B = A = 1$

Flipflop (3) sees $J_C = K_C = A.B = 1.1 = 1$

When the next falling edge comes A = 0, B = 0 and C = 1. So we get 100.

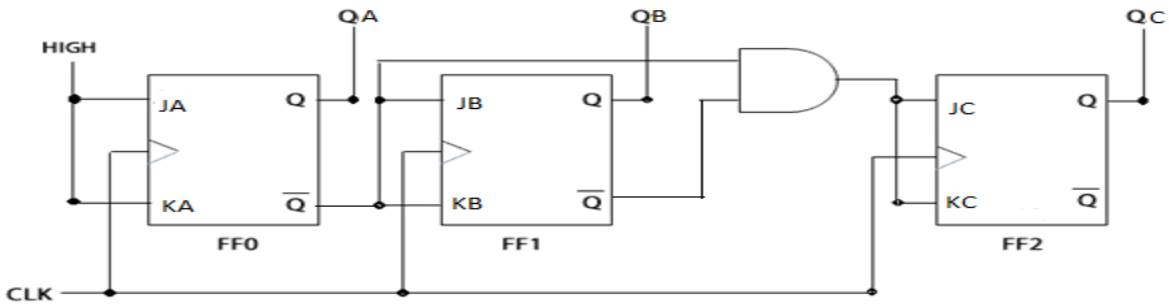
In this way the counts keep proceeding until 111. In the next clock pulse, the output goes back to 000 and the same sequence is repeated.

Truth Table:

C	B	A	Counted state
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7
0	0	0	0

3 Bit Synchronous Down Counter:

In three bit synchronous down counter, the counts proceed in the following way 000, 111, 110, 101....000, 111.



Outputs are taken at A, B and C but inputs to the flip-flops are from Q_A, Q_B and $\overline{Q}_C$. Let the outputs Q_A, Q_B and $Q_C = 0$ initially. So Q_A, Q_B and $Q_C = 1$. Before the clock pulse comes the inputs seen by various flip-flops are

Flipflop 1: $J_A = K_A = 1$

Flipflop 2: $J_A = K_B = Q_A = 1$

Flipflop 3: $J_C = K_C = Q_A \cdot Q_B = 1.1 = 1$

So when the falling edge of clock comes A = 1 (toggles), B = 1(toggles), C = 1 (toggles). So we get 111.

After the 1st clock pulse A =1, B=1, C = 1. So Q_A=0, $\overline{Q_B}$=0 and $\overline{Q_C}$=0. The new inputs seen by different flip-flops are

Flipflop 1: $J_A = K_A = 1$

Flipflop 2: $J_A = K_B = Q_A = \overline{0}$

Flipflop 3: $J_C = K_C = Q_A \cdot \overline{Q_B} = \overline{0}.0 = 0$

When the next clock pulse comes A = 0 (toggles), B=1 (previous state), C = 1 (previous state). So we get state 110.

Now $\overline{Q_A}$=1, $\overline{Q_B}$=0 and $\overline{Q_C}$=0. Now after the clock has passed next input seen by each flipflop is

Flipflop 1: $J_A = K_A = 1$

Flipflop 2: $J_A = K_B = Q_A = \overline{0}$

Flipflop 3: $J_C = K_C = Q_A \cdot \overline{Q_B} = \overline{1}.0 = 0$

When the next falling edge of clock comes J_A = 1 (toggles), J_B = 0 (toggles), J_C = 1 (state).So we get state 101.

In this way the counting sequence decrements after each clock pulse until we get 000 and the same sequence repeats.

Truth Table:

C	B	A	Counted state
0	0	0	0
1	1	1	7
1	1	0	6
1	0	1	5
1	0	0	4
0	1	1	3
0	1	0	2
0	0	1	1
0	0	0	0

Modulus of a Counter:

Modulus of a counter tells us the number of states the counter can count. When we say mod 10 counter, it means it can count 10 states. Similarly mod 16 counter can count 16 states.

If 'm' is the number of states to be counted and 'n' be the number of flip-flops then $m < 2^n$.

Thus 4 flip-flops are required for mod 9 to mod 16. 3 flip-flops are required for mod 5 to mod 8.

If we have 4 flip-flops we can count 16 states. However if we want to count only 10 states then the remaining six states must be skipped. So immediately after counting 10^{th} state, all the flip-flops will be reset to 0 and the same 10 states will be counted again and again.

IC 7490 can be used mod 2 to mod 10 counter.

IC 7493 is a mod 16 counter. it can count 16 states from 0000 to 1111.

References:

1. A text book of Basics of Computer Organization of vision publication by H.R.Arvind.
2. A text book of Basics of Computer Organization of Nirali publication by Dr.J.A.Bangali
3. www.google.com