**6. Sequence Generator**

**AIM :** To design and implement sequence generator with and without bushing using

JK Flip flop IC 7476 & Shift Register IC 74194.

**OBJECTIVE :** To understand sequence generator, one of the sequential circuit.

**IC’s USED :** IC 74194,7408 (AND-gate), 7432 (OR-gate).

# THEORY :

# Part A. Sequence Generator with Flip flop

A sequential circuit which generates a prescribed sequence of bits in synchronism with a clock is referred to as a sequence generator. These pulse trains or sequence of bits can be used to open valves, close gates, turn on lights, and turn off machines and other variety of jobs.

For the design of sequence generator, we first determine the required no. of flip flops and the logic circuit for the next state decoder.

No. of flip flops required to generate particular sequence can be determined as follows.

1. Find the no. of 1’s in the sequence.
2. Find the no. of 0’s in the sequence.
3. Take the maximum out of two.
4. If N is the required no. of flip flops, choose minimum value of ‘n’ to satisfy equation given below.

Max (0’s , 1’s) ≤ 2n-1

The sequence generator can be classified as

1. sequence generator without bushing
2. sequence generator with bushing

The aim in this experiment is to design a sequence generator to generate a sequence of bit i.e. 10101.

# Part B. Sequence Generator using Shift Register IC 74194

Theory :**IC 74194 : 4 bit bidirectional Shift Register:**

This bidirectional shift register is designed to incorporate virtually all the features a system designer may want in a shift register; they feature Parallel inputs, parallel outputs, right shift, left shift serialinputs, operating mode control inputs,and a direct overriding clear line. The register has four distinct modes of operations,namely:

Parallel load,

Shift right (in the direction QA towards QD)

Shift left (in the direction QD towards QA)

Inhibit clock (do nothing)

Synchronous parallel loading is accomplished by applying the four bits of data and taking both mode control inputs, S0 and S1,High. The data is loaded in to the associated flip-flops and appear at the outputs after positive transition of the clock input. During loading, serial data flow is inhibited.

Shift right is accomplished synchronously with the rising edge of the clock pulse when S0 is High and S1 is Low. Serial data for this mode is entered at the shift right data input. When S0 is Low and S1 is High, data shifts left synchronously and new data is entered at the shift left serial input.

Clocking of the flip flop is inhibited when both mode control inputs are LOW.

|  |  |  |
| --- | --- | --- |
| Mode Control Input | | Operation |
| S1 | S0 |
| 0 | 0 | Clock Inhibit |
| 0 | 1 | Shift right |
| 1 | 0 | Shift Left |
| 1 | 1 | Parallel loading |

**Function Table of IC 74194**

**Design**:

The minimum number of flip-flops, N, required to generate a sequence of length S is given by S ≤ 2N – 1.

In this case S=7, therefore the minimum value of N, which may generate in this sequence is 3. However, it is not guaranteed to lead to a solution. If the given sequence leads to seven distinct states, then only three flip flops are sufficient otherwise we have to increase the number of flip flops. We write the states of circuit as given in table 1. The prescribed sequence is listed under QA and the sequence listed under QB and QC are the same sequence delayed by one and two clock pulses respectively. From the table we observe that all the states are not distinct, which means N=3 is not sufficient. Next we assume that N=4 and prepare table 2. The last column gives the required serial input for getting the desired change of state when a clock pulse is applied. This is obtained by assuming D type flip flop and looking at the QA output. For example, at the falling edge of first clock pulse, QA=1. The second clock pulse must result in QA=1 which requires its D input to be 1. In the same manner, all the entries in column Y are determined.

Table 1. State Assignment Table of Sequence Generator (N=3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Number of Clock Pulses | Flip Flop Inputs | | | States |
| QA | QB | QC |
| 1 | 1 | 1 | 1 | 7 \* |
| 2 | 1 | 1 | 1 | 7 \* |
| 3 | 0 | 1 | 1 | 3 |
| 4 | 1 | 0 | 1 | 5 \* |
| 5 | 0 | 1 | 0 | 2 |
| 6 | 1 | 0 | 1 | 5 \* |
| 7 | 1 | 1 | 0 | 6 |

Excitation table of D Flip-Flop

|  |  |  |
| --- | --- | --- |
| Present State Qn | Next State Qn+1 | Flip-Flop Input D |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Table 2.State Assignment Table of Sequence Generator (N=4)

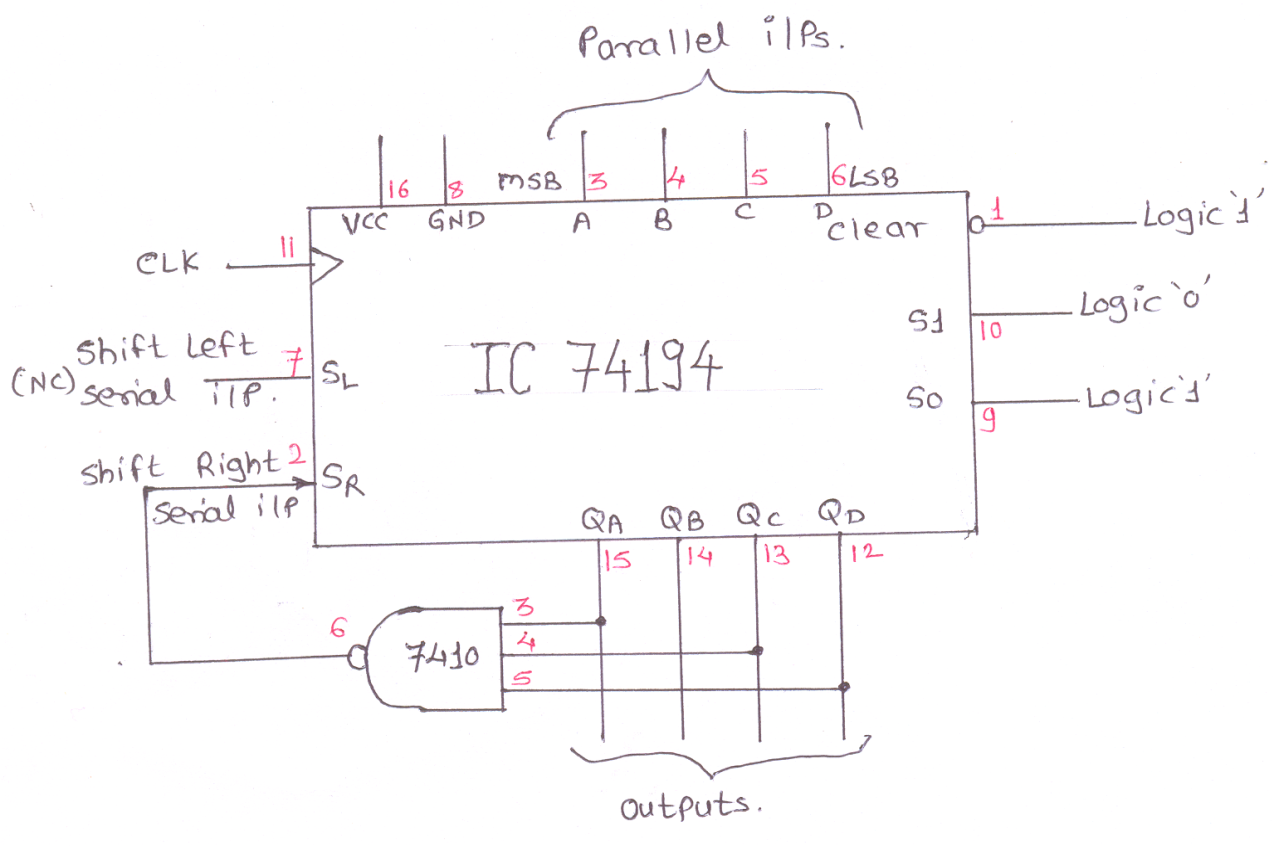
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Number of Clock Pulses | Flip Flop Inputs | | | | States | Y  Serial i/p |
| QA | QB | QC | QD |
| 1 | 1 | 1 | 1 | 0 | 14 | 1 |
| 2 | 1 | 1 | 1 | 1 | 15 | 0 |
| 3 | 0 | 1 | 1 | 1 | 7 | 1 |
| 4 | 1 | 0 | 1 | 1 | 11 | 0 |
| 5 | 0 | 1 | 0 | 1 | 5 | 1 |
| 6 | 1 | 0 | 1 | 0 | 10 | 1 |
| 7 | 1 | 1 | 0 | 1 | 13 | 1 |

**K-Map Simplification:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | X | X | X | X |
| 01 | X | 1 | 1 | X |
| 11 | X | 1 | 0 | 1 |
| 10 | X | X | 0 | 1 |

Y= QA’ + QB’ + QC’ = QA.QB.QC

**LOGIC DIAGRAM:**

****

**HARDWARE REQUIREMENTS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl No.** | **ICs** | **Description** | **Quantity** |
| 1 | 74194 | 4 bit bidirectional universal shift register | 1 |
| 2 | 7410 | 3 input NAND Gate | 1 |

**Conclusion:** In this way sequence generator using JK flip flips & shift register is designed and implemented.

**Enhancements / Modifications –** Sequence generator can also be implemented with shift register instead of flip flops. Use IC 7495 universal shift register IC and try to implement sequence generator.

**FAQs :**

1. What is sequential logic circuit?

A sequential logic circuit consists of a memory elements in addition to

combinational circuit. Its output at any instant of time depends upon

the present input as well as present state of memory element.

1. What is meant by delay line?

It is used to introduce time delays in digital signals.

1. What is meant by following terms

a) Synchronous preset b) Asynchronous preset

c) Synchronous clear d) Asynchronous clear

* + a)Preset operation is synchronised with the clock
  + b)Preset operation is independent of the clock
  + c) clear is performed in synchronous with clock
  + d) clear is performed independent with clock

1. Is asynchronous counter faster than synchronous counter ?

In a synchronous counter the time required for change of any state is same and is equal to delay time of one flip flop where as in asynchronous counter all flip flops are not clocked simultaneously, hence time required is not same.

1. What is mean by lockout in counter?

In a counter design for a fewer state than the maximum possible state some time it may so happen that counter enters in unused state and goes from one unused state to another unused state and never comes to used state.

1. What is mean by state table?

It consists of complete information about present state and next state and outputs of a sequential system.

1. What is mean by state diagram?

The information available in a state table can be represented as graphically. the graphical representation is known as state diagram.

1. What is the advantage of state reduction in the design of sequential circuit?

It reduces the number of flip flops

1. What is meant by excitation table?

This gives information about what should be the flip flop inputs if outputs are specified before and after the clock pulse.

1. How many flip flops are required to design sequence generator using Counters:

max (0’S,1’S) in a given sequence <= 2(N - 1)

Where, N = Number of flip flops

1. How many flip flops are required to design sequence generator using shift registers:

S<= 2N - 1

Where, N=Number of flip flops

S= Length of sequence

1. What is Lock out condition? How it is avoided?

When counter enters into one of the invalid state and after application of

pulses remains in invalid states only i. e. counter gets locked into invalid state

& this is called as lock out. Lock out can be avoided by providing bushing to

all the invalid states in such a way that after application of one or more clock

pulses counter will fall into one of the valid state.