|  |  |  |
| --- | --- | --- |
| G:\nsu-logo.png  **North South University**  Department of Electrical & Computer Engineering    **LAB REPORT**  Course Name:CSE231L  Section: 01  Experiment Number: 7A,7B     |  | | --- | | Experiment Name: Introduction to Flip-Flops and Shift Registers  BCD to Seven Segment Display |     Experiment Date: 02/04/2021  Report Submission Date: 09/04/2021  Group Number: | |
| Student Name: Ishrat Jahan | Score |
| Student ID:1921909042 |  |
| Remarks: |

**Introduction to Flip-Flops and Shift Registers**

**BCD to Seven Segment Decoder**

**Objectives**

• Learning about the concept of states in digital logic and how Flip-Flop circuits can be used to store state information.

• Understand the internal logic of J-K Flip-Flops and implement one using basic logic gates.

• Understand the relationship between J-K, T and D Flip-Flops and observe the characteristics of all three.

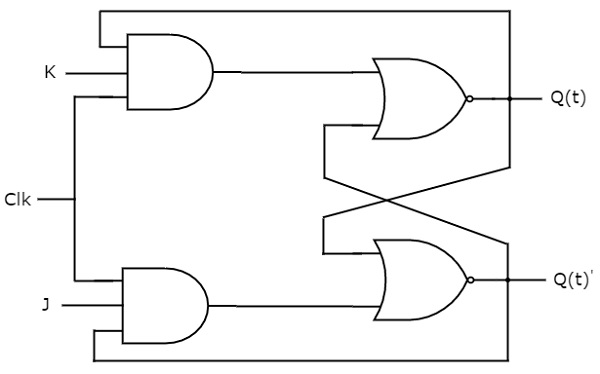
• Drawing the circuits of T and D Flip-Flops using Logisim.

**Theory**

There are two types of logic circuits; Combinational Logic, whose output depends on the present inputs and Sequential Logic, whose outputs are a function of both current and past inputs. In a sequential logic, past inputs are stored in memory elements and one of them is a Flip-Flop. The contents and information’s that are stored from past inputs at a given time, is referred as the circuits “state”.

**Flip-Flop** A flip flop is a sequential circuit which consists of a single binary state of information or data. It has two stable states. In a stable state, the output of a Flip-Flop is 1(High) and 0(low).The outputs can only be changed with a clock pulse. When a clock pulse is connected to a Flip-Flop, it determines the input to the Flip-Flop at that time and stores that value or the values stored in it beforehand. Flip-Flops are controlled by clock transition.

**JK Flip-Flop**

****

In a JK Flip-Flop J stands for Set and K stands for Reset. This circuit has two inputs J & K and two outputs Q & Q’.

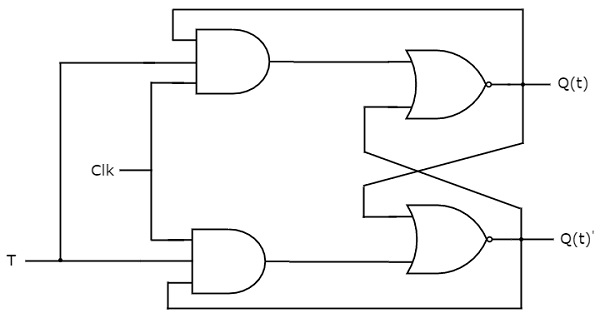
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Characteristic Table** | | |  | **Excitation Table** | | | |
| **J** | **K** | **Qnext** |  | **Q** | **Qnext** | **J** | **K** |
| **0** | **0** | **Q** |  | **0** | **0** | **0** | **X** |
| **0** | **1** | **0** |  | **0** | **1** | **1** | **X** |
| **1** | **0** | **1** |  | **1** | **0** | **X** | **1** |
| **1** | **1** | **Q’** |  | **1** | **1** | **X** | **0** |

Table B1: JK flip-flop: Characteristic and Excitation Tables

Qnext is the output for the given inputs if a clock pulse is provided and Q is the current output. The excitation table shows the values of J and K to go from Q to Qnext. When J = 0 and K = 0, the next output (Qnext) will be the same as the current output (Q). And if the current output (Q) is 0 and we want to change it to 1 (Qnext = 1), we will have to set J to 1 and provide a clock pulse. The value of K doesn’t matter because the output will change to 1 in either case (K = 0 or K = 1) if we see it from the characteristic table.

**T Flip-Flop**

T Flip-Flop is a simplified version of JK Flip-Flop.  It is obtained by connecting the same input ‘T’ to both inputs of JK flip-flop. It operates with only positive clock transitions or negative clock transitions.

****

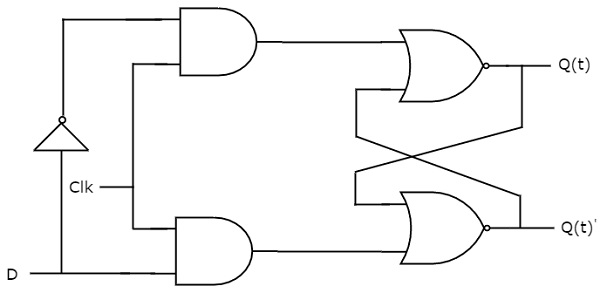
The T flip-flop changes state ("toggles") whenever the input T is high and the clock input is strobed. If the T input is low, the flip-flop holds the previous value when given a clock pulse.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Characteristic Table** | |  | **Excitation Table** | | |
| **T** | **Qnext** |  | **Q** | **Qnext** | **T** |
| **0** | **Q** |  | **0** | **0** | **0** |
|  |  |  | **0** | **1** | **1** |
| **1** | **Q’** |  | **1** | **0** | **1** |
|  |  |  | **1** | **1** | **0** |

Table B2: T flip-flop: Characteristic and Excitation Tables

**D Flip-Flop**

D flip-flop operates with only positive clock transitions or negative clock transitions. That means, the output of D flip-flop is insensitive to the changes in the input D,except for active transition of the clock signal.

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Characteristic Table** | |  | **Excitation Table** | | |
| **D** | **Qnext** |  | **Q** | **Qnext** | **D** |
| **0** | **0** |  | **0** | **0** | **0** |
|  |  |  | **0** | **1** | **1** |
| **1** | **1** |  | **1** | **0** | **0** |
|  |  |  | **1** | **1** | **1** |

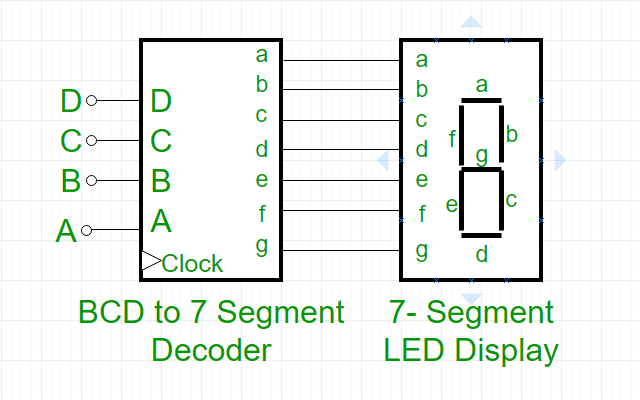
Table B3: D flip-flop: Characteristic and Excitation Tables

**Registers**

A register is a group of flip-flops. Each flip-flop is capable of storing one bit of information. An n-bit register contains a group of n flip-flops capable of storing n bits of binary information. A register have combinational gates that perform certain data processing tasks. And it consists of a group of flip-flops and gates that effect their transition. The flip-flops hold binary information and the gates determine how the information is transferred into the registers.

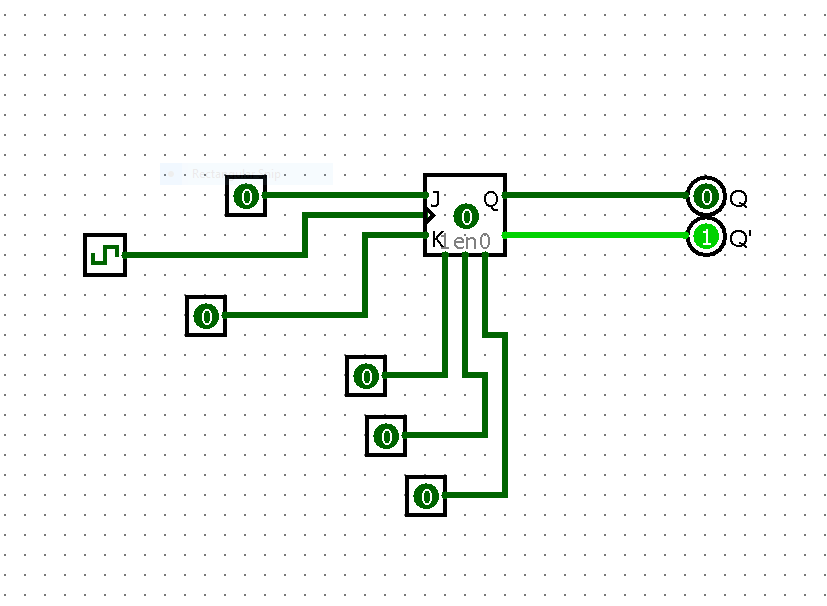
**Seven Segment Decoder**

**Seven segment** display is an electronic device which consists of seven Light Emitting Diodes (LEDs) arranged in a some definite pattern which is used to display Hexadecimal numbers, in this case decimal numbers, as input is BCD 0-9.  First, the decimal number is changed to its BCD equivalent signal then BCD to seven segment decoder converts that signals to the form which is fed to seven segment display.

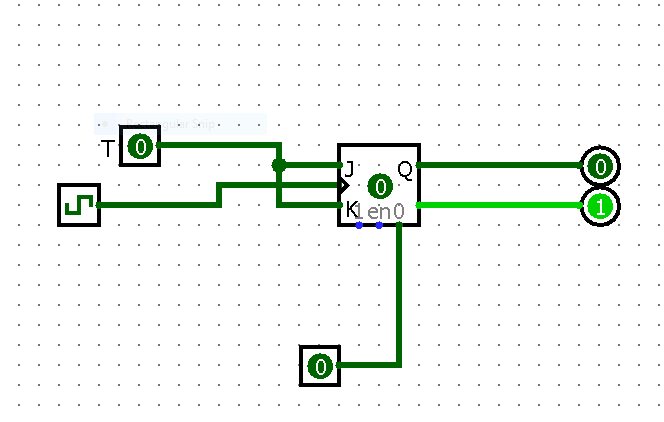
****

**Circuit Diagrams**

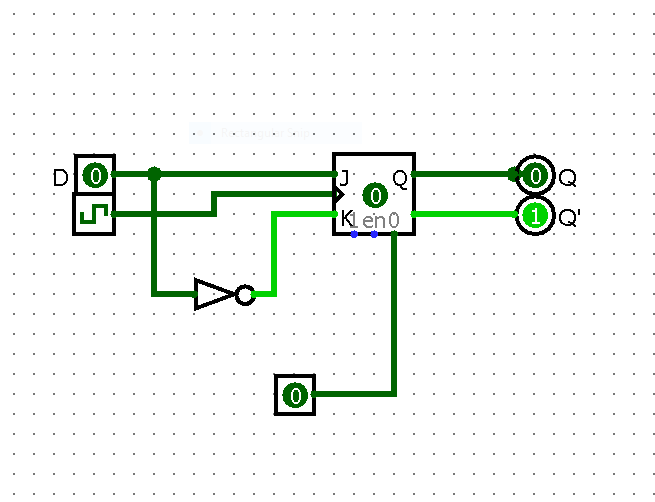
**Experiment 1: Constructing a JK Flip-Flop using AND and NOR gates**

****

**Experiment 2: Using a JK Flip-Flop to construct T Flip-Flop and D Flip-Flop**

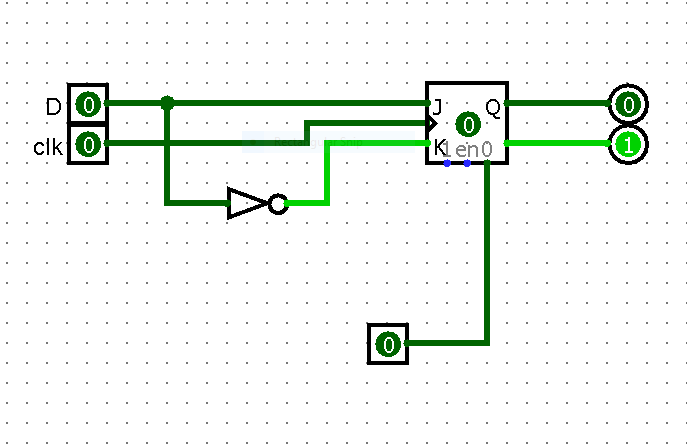
****

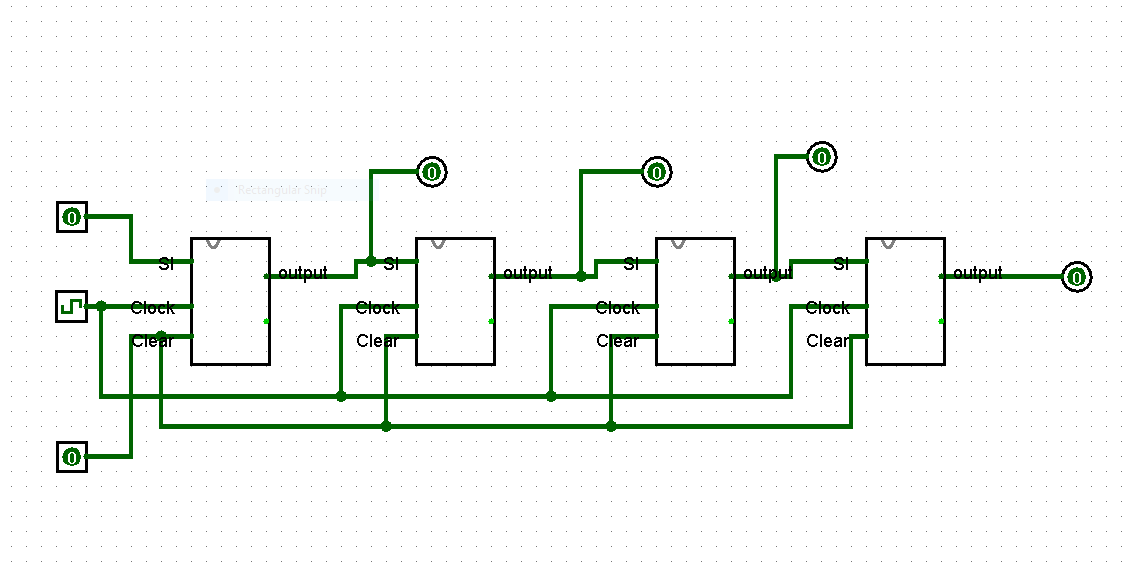
**T Flip-Flop**

****

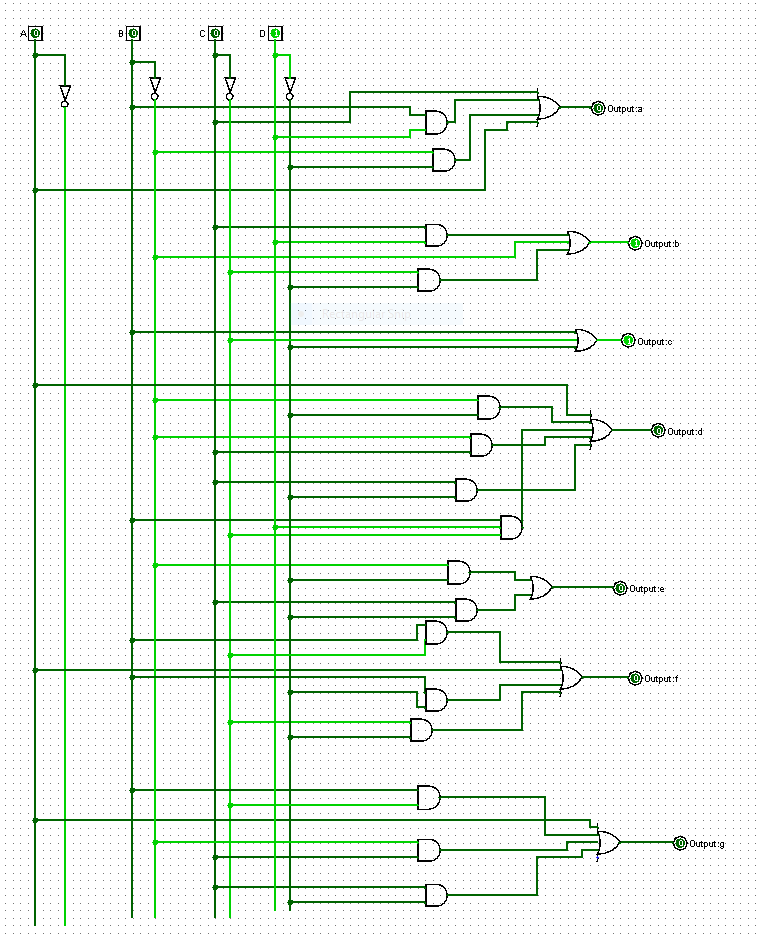
**D Flip-Flop**

**Experiment 3: Constructing a right shift register using D Flip-Flops**

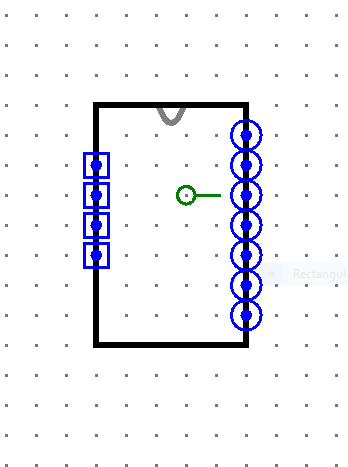
****

****

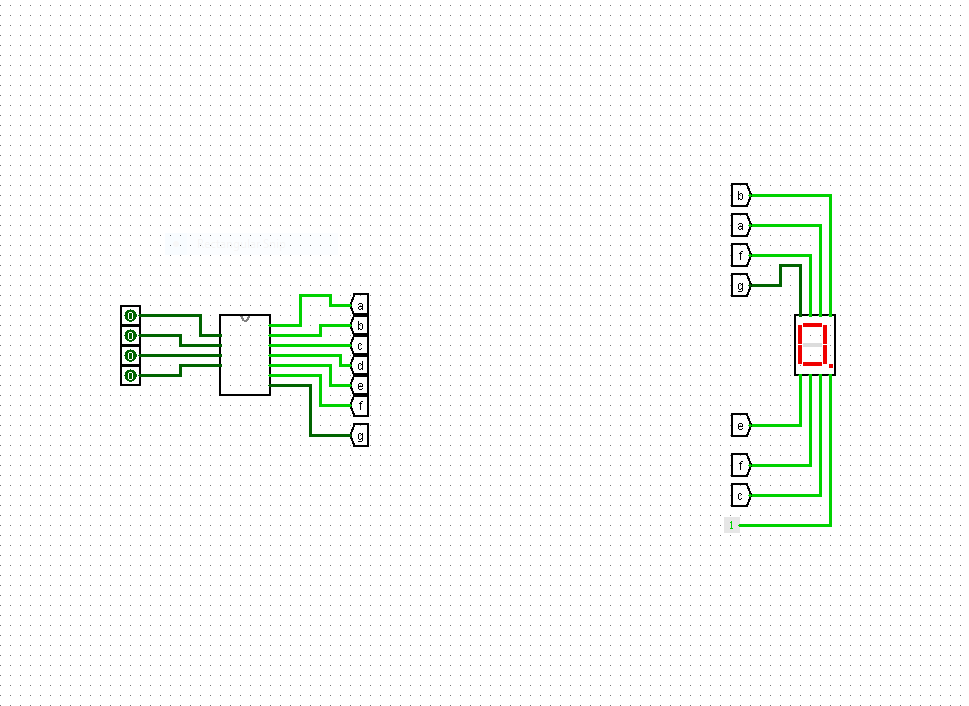
**Figure D.3.1: Right Shift Register**

****

**Figure: Seven Segment Decoder circuit**

****

**Fig: Sub-circuit**

****

**Fig: Seven Segment Decoder**

**Truth Table**

**F.1 Experimental Data: J-K Flip-Flop using AND and NOR gates**

|  |  |  |  |
| --- | --- | --- | --- |
| **J** | **K** | **Q** | **Q’** |
| **1** | **0** | **1** | **0** |
| **0** | **0** | **1** | **0** |
| **0** | **1** | **0** | **1** |
| **0** | **0** | **1** | **0** |
| **1** | **1** | **1** | **0** |
| **1** | **0** | **1** | **0** |
| **1** | **0** | **1** | **0** |
| **1** | **1** | **0** | **1** |

**Table F.1.1**

**F.2 Experimental Data: T and D Flip-Flops using J-K Flip-Flops**

|  |  |
| --- | --- |
| **T** | **Q** |
| **0** | **1** |
| **1** | **0** |

**Table F.2.1**

|  |  |
| --- | --- |
| **D** | **Q** |
| **0** | **1** |
| **1** | **0** |

**Table F.2.2**

**F.3 Experimental Data: Right shift register using D Flip-Flops**

|  |  |  |
| --- | --- | --- |
| **States** | **Input** | **Output** |
| **Initial State** | **X** | **XXXX** |
| **T1** | **1** | **1XXX** |
| **T2** | **0** | **01XX** |
| **T3** | **1** | **101X** |
| **T4** | **0** | **0101** |

**Table F.3.1**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Decimal** | **Inputs** | | | | **Outputs** | | | | | | |
| **D** | **C** | **B** | **A** | **a** | **b** | **c** | **d** | **e** | **f** | **g** |
| **0** | **0** | **0** | **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **0** |
| **1** | **0** | **0** | **0** | **1** | **0** | **1** | **1** | **0** | **0** | **0** | **0** |
| **2** | **0** | **0** | **1** | **0** | **1** | **1** | **0** | **1** | **1** | **0** | **1** |
| **3** | **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **0** | **0** | **1** |
| **4** | **0** | **1** | **0** | **0** | **0** | **1** | **1** | **0** | **0** | **1** | **1** |
| **5** | **0** | **1** | **0** | **1** | **1** | **0** | **1** | **1** | **0** | **1** | **1** |
| **6** | **0** | **1** | **1** | **0** | **1** | **0** | **1** | **1** | **1** | **1** | **1** |
| **7** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **0** | **0** | **0** | **0** |
| **8** | **1** | **0** | **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |
| **9** | **1** | **0** | **0** | **1** | **1** | **1** | **1** | **1** | **0** | **1** | **1** |
| **10** | **1** | **0** | **1** | **0** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |
| **11** | **1** | **0** | **1** | **1** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |
| **12** | **1** | **1** | **0** | **0** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |
| **13** | **1** | **1** | **0** | **1** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |
| **14** | **1** | **1** | **1** | **0** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |
| **15** | **1** | **1** | **1** | **1** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |

**Table: Seven Segment Decoder**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **1** | **0** | **1** | **1** |
| **01** | **0** | **1** | **1** | **1** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **1** | **X** | **X** |

**a=A+C+BD+B’D’**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **1** | **1** | **1** | **1** |
| **01** | **1** | **0** | **1** | **0** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **1** | **X** | **X** |

**b= B’+C’D’+CD**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **1** | **1** | **1** | **0** |
| **01** | **1** | **1** | **1** | **1** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **1** | **X** | **X** |

**c=B+C’+D**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **1** | **0** | **1** | **1** |
| **01** | **0** | **1** | **0** | **1** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **1** | **X** | **X** |

**d=A+B’D’+B’C+CD’+BC’D**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **1** | **0** | **0** | **1** |
| **01** | **0** | **0** | **0** | **1** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **0** | **X** | **X** |

**e=B’D’+CD’**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **1** | **0** | **0** | **0** |
| **01** | **1** | **1** | **0** | **1** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **1** | **X** | **X** |

**f=A+BC’+BD’+C’D’**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AB/CD** | **00** | **01** | **11** | **10** |
| **00** | **0** | **0** | **1** | **1** |
| **01** | **1** | **1** | **0** | **1** |
| **11** | **X** | **X** | **X** | **X** |
| **10** | **1** | **1** | **X** | **X** |

**g= A+BC’+B’C+CD’**

**Figure: K: Maps a-f for Seven Segment Decoder**

**Discussion**

In Lab-7 we studied Flip-Flops and Shift Registers. Firstly, we studied what are Flip-Flops and how they can be used to store data, types of Flip-Flops; JK, D (“Data” or “Delay”) and T (“Toggle”) are three types of flip-flops used in digital logic circuit and how we use a clock pulse to change the output in a flip-flop. Then, we were taught about the Characteristic Table and Excitation Table for JK Flip-Flop. The characteristic table represents the state the flip-flop will move to based on the current state and other inputs. The J and K are the inputs and Qnext is the output that we get when a clock pulse is added to a flip-flop circuit. On the other hand, the excitation table generates specific Qnext when the current state (Q) is known. For example: According to the characteristic table if J=0 and K=0, Qnext will be same as Q and if J=0 and K=1,Qnext will be 1 as well. And from the excitation table we can see that if J=0 and K is either 0/1, Qnext is same as Q. Following these we carried out Experiment-1 where we constructed a JK Flip-Flop in Logisim. For the construction. we used the JK Flip-Flop sub-circuit from the memory section and then completed Table F.1.1.Secondly,we studied the characteristic table and excitation table for both D Flip-Flop and T Flip-Flop. Both of them have one input .For T flip-flop, when T=0,Qnext is equal to Q’s value whatever it is in the excitation table. And for the D Flip-flop, when D=0, Qnext=0 and D=1 Qnext equals to 1. From the excitation table we can see that, the value of Qnext depends on the value of D. Then we completed Experiment-2, where we constructed T and D flip-flop using a JK Flip-flop and completed Tables F.2.1 and F.2.2.Lastly, we studied registers and how we can create one, by connecting group of Flip-Flops which will perform data processing tasks. Following that, we completed Experiment -3, where we created a right shift register using D flip- flop and completed table F.3.1.In the lab manual 7b ,we studied the seven segment decoder. We learned what a seven segment decoder is, how we can display a BCD number between 0-9 using the seven outputs a,b,c,d,e,f,g in the decoder ,each of which lights when a logic low is applied to its corresponding input pin.Then we completed the table provided, where each decimal number’s BCD forms were given and we had to fill up the output section,how we can display each of the numerals in the seven segment display. Following that,we constructed the circuit for the seven segment decoder by drawing K-maps of each of the outputs and constructed each of its equation in Sum of Minterms forms.We then used the sub-circuit,to create a circuit,where we used tunnels in the output section.Lastly,we used the seven segment decoder sub-circuit available in Logisim to draw the circuit with tunnels in the output and that displays the numbers. One drawback of the experiment was that creating flip-flops in Logisim showed error for correct circuits so we had to reopen files again and again.