United International University

Department of CSE

Course Code: CSE 1326

Course Name: Digital Logic Design Laboratory

Experiment no: 05

Experiment Name: Multiplexer

Submitted by:

Name: Talha Jubayer

Student ID: 0112410062

Submitted to:

Fahim Hafiz [Lecturer, Dept. of CSE]

Date of Performance: 27-11-2024

Date of Submission: 16-12-2024

Objective:

- Learn about Multiplexer
- Implementing logic function using multiplexer

The primary objective of this report is to explore the design and functionality of multiplexers and their applications in digital systems. This includes understanding how multiplexers can be utilized to implement various logic functions, demonstrating their effectiveness in simplifying and optimizing combinational logic designs. The report aims to provide a detailed understanding of how smaller multiplexers can be combined to create larger ones, how the select key operates, and the significance of multiplexer ICs. Through this exploration, the report will highlight the versatility of multiplexers in logic design and their critical role in enabling efficient data routing and decision-making processes within digital systems.

Apparatus/Instruments used in the Lab:

● Trainer Board

● Jumper wires

● Power supply

● NOT Gate (74LS04)

● AND Gate (74LS08)

● OR gate (74LS32)

● 2-to-1 multiplexer

● 4-to-1 multiplexer

● 8-to-1 multiplexer

● 74LS153

● 74LS151

● Logisim

Theory:

Multiplexing means transmitting a large number of information units over a smaller number of channels or lines. A digital multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line. The selection of a particular input line is controlled by a set of selection lines. Normally, there are 2^n input lines and n selection lines whose bit combinations determine which input is selected. A multiplexer is also called a data selector, since its selecting data pin based on what value put in its selecting pin. The AND gates and inverters in a multiplexer function similarly to a decoder as they interpret the selection input signals. Typically, a multiplexer with 2^n input lines is built using a decoder with n inputs and 2^n outputs. Each input line is connected to a corresponding AND gate within the decoder. The outputs of these AND gates are then combined through a single OR gate to generate the final output. The size of a multiplexer is defined by its 2^n input lines and one output line, along with the n selection lines required. Multiplexers are commonly abbreviated as MUX.

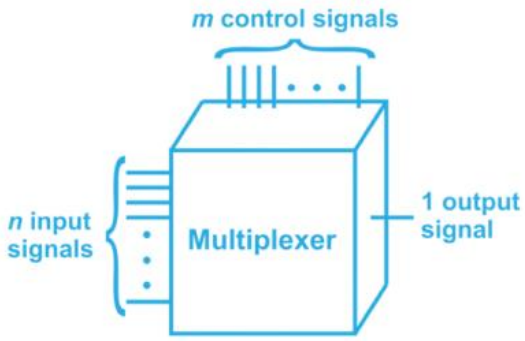

Figure 1 Block diagram of Multiplexer-SRC:prepbytes.com

Types of Multiplexers in Digital Electronics

There are several types of multiplexers in digital electronics, which are as follows:

1. **2X1 Multiplexer:**

A 2×1 multiplexer is a combinational logic circuit with two inputs, labeled A_0 and A_1, one selection line, labeled S_0, and a single output, labeled Y. Depending on the state of the selection line S_0, one of the two inputs is directed to the output line. This type of multiplexer is the simplest form used in digital electronics. The block diagram and truth table of the 2×1 multiplexer are illustrated below.

Inputs	Outputs
S_0	Y
0	A_0
1	A_1

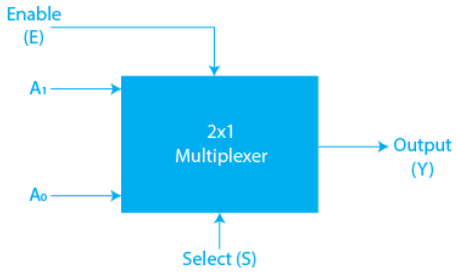

Figure 2 Truth table and block diagram of 2X1 MUX-SRC: prepbytes.com

2. 4X1 Multiplexer:

A 4×1 multiplexer is a combinational logic circuit that selects one input out of four possible inputs (A_0, A_1, A_2, A_3) and directs it to the output line (Y). The selection is determined by two control signals or selection lines, S_0, S_1. The block diagram and truth table for the 4×1 multiplexer are provided below.

Inputs		Outputs
S_1	S_0	Y
0	0	A_0
0	1	A_1
1	0	A_2
1	1	A_3

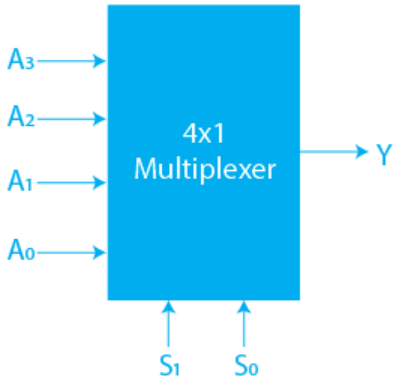

Figure 3 TT and BG of 4X1 MUX-SRC: prepbytes.com

3. 8X1 Multiplexer:

An 8×1 multiplexer is a combinational logic circuit that selects one of eight input lines and directs it to the output line based on the states of the selection lines. It includes eight input lines, one output line, and three selection lines. The block diagram and truth table for the 8×1 multiplexer are provided below.

Inputs			Outputs
S_2	S_1	S_0	Y
0	0	0	A_0
0	0	1	A_1
0	1	0	A_2
0	1	1	A_3
1	0	0	A_4
1	0	1	A_5
1	1	0	A_6
1	1	1	A_7

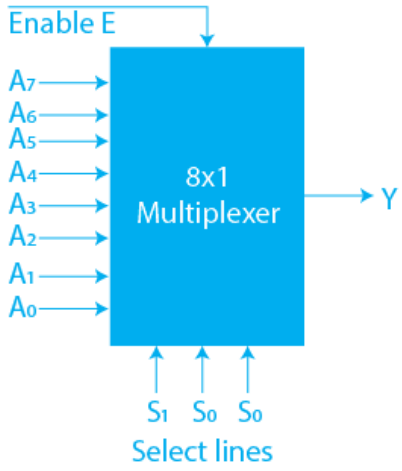

Figure 4 TT and BG of 8x1 MUX-SRC: prepbytes.com

4. 8X1 Multiplexer using 4X1 and 2X1 Multiplexer:

We can use a lower-order multiplexer to implement the 8X1 multiplexer. We need two 4X1 multiplexers and one 2X1 multiplexer to implement the 8X1 multiplexer. There are two selection lines, four inputs, and one output on the 4X1 multiplexer. There is only one selection line on the 2X1.

Two 4X1 multiplexers are required to receive 8 data inputs. The 4X1 multiplexer generates a single output. Thus we need a 2X1 multiplexer to get the final output. The block diagram of an 8X1 multiplexer made up of 4X1 and 2X1 multiplexers is shown below.

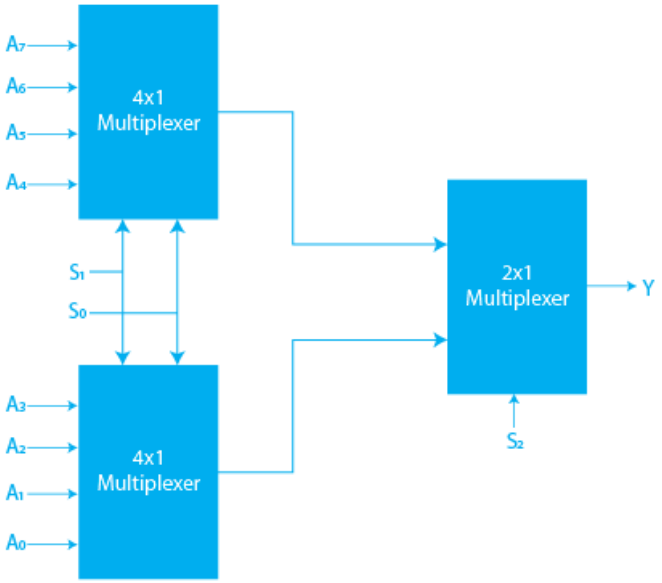

Figure 5 Block Diagram

5. 16X1 Multiplexer

A 16×1 multiplexer is a combinational logic circuit with 16 input lines (A_0, A_1….. A_{15}) four selection lines (S0, S1, S2, S3), and a single output (Y). Depending on the combination of values on the selection lines, one of the 16 inputs is routed to the output. The block diagram and truth table for the 16×1 multiplexer are shown below.

Inputs				Outputs
S_0	S_1	S_2	S_3	Y
0	0	0	0	A_0
0	0	0		A_1
0	0	1	0	A_2
0	0	1	1	A_3
0	1	0	0	A_4
0	1	0	1	A_5
0	1	1	0	A_6
0	1	1	1	A_7
1	0	0	0	A_8
1	0	0	1	A_9
1	0	1	0	A_{10}
1	0	1	1	A_{11}
1	1	0	0	A_{12}

1	1	0	1	A_{13}
1	1	1	0	A_{14}
1	1	1	1	A_{15}

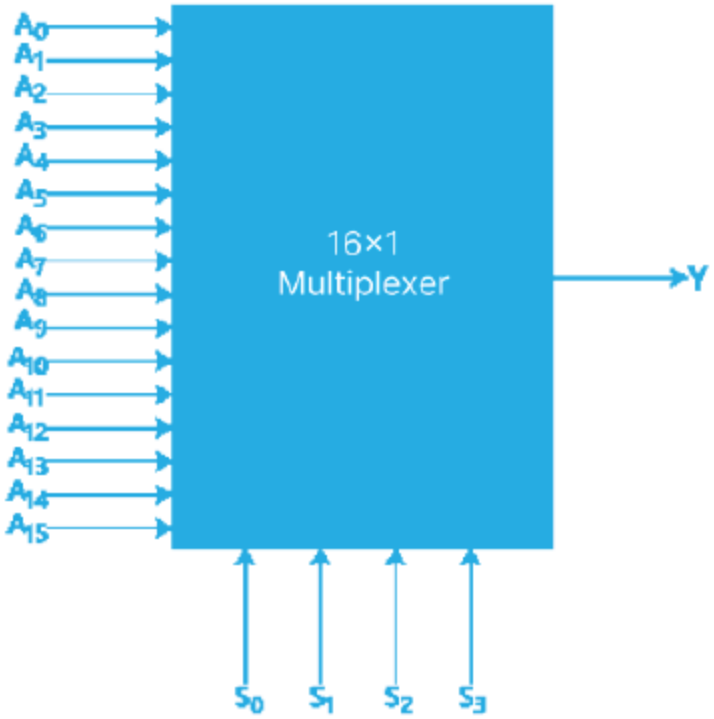

Figure 6 TT and BG of 16X1 MUX- SRC: prepbytes.com

6. 16X1 Multiplexer using 8X1 and 2X1 Multiplexer:

We can use a lower-order multiplexer to construct the 16X1 multiplexer. Two 8X1 multiplexers and one 2X1 multiplexer are required to implement the 8X1 multiplexer. The 8X1 multiplexer has three selection lines, four inputs, and one output. There is only one selection line on the 2×1.

We need two 8X1 multiplexers to get 16 data inputs. The 8X1 multiplexer generates a single output. Thus, a 2X1 multiplexer is required to obtain the final output. The following is a block diagram of a 16X1 multiplexer made up of 8X1 and 2X1 multiplexers.

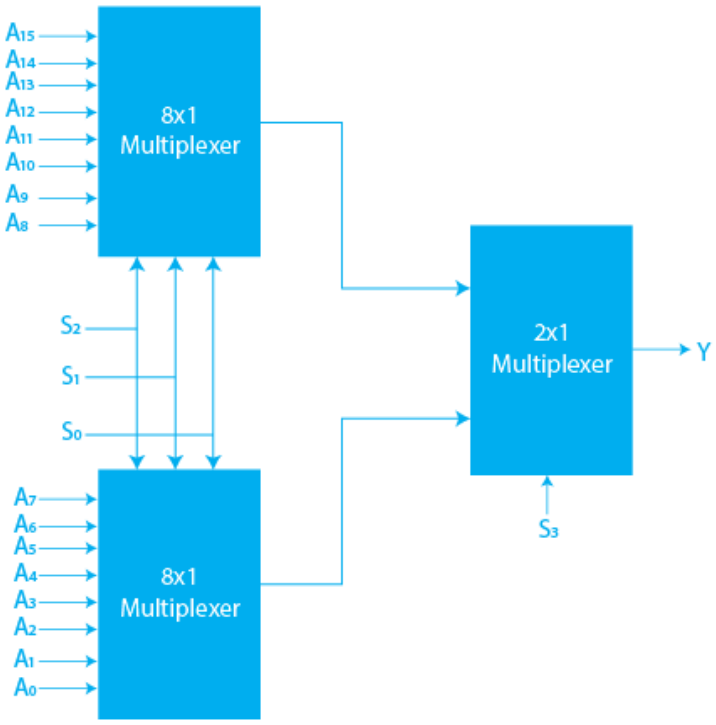

Figure 7 Block Diagram of two 8X1 and one 2X1 MUX

Solution of the mentioned problems:

1) Pin diagram of 74151 and 74153

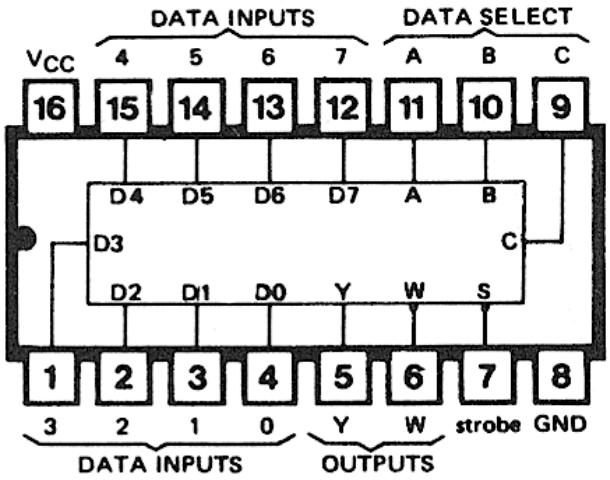

Figure 9 74LS151

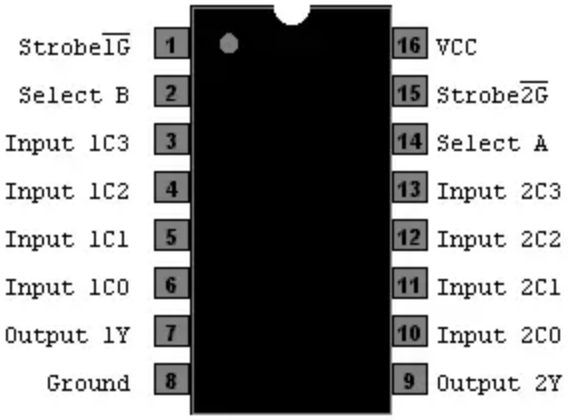

Figure 8 74LS153

2) Truth table, Input functions and circuit diagrams of implementation of Boolean function f(A,B,C) = ∑m (0,2, 5, 7) using-

- **4 input (4:1) MUX and necessary basic gates**

A	B	C	F
0	0	0	**1**
0	0	1	**0**
0	1	0	**1**
0	1	1	**0**
1	0	0	**0**
1	0	1	**1**
1	1	0	**0**
1	1	1	**1**

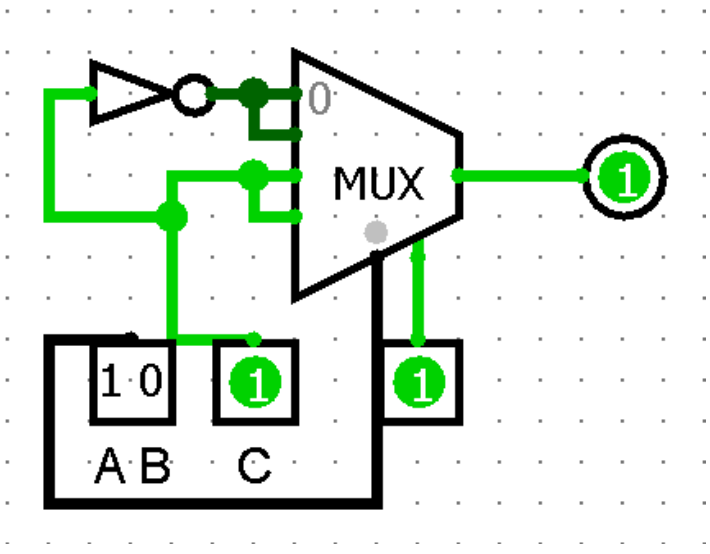

Figure 10 TT and Circuit Diagram

Input function : F (A, B, C) = A'B'C' + A'BC' + AB'C + ABC

- **2-input (2:1) MUX and necessary basic gates**

A	B	C	F
0	0	0	**1**
0	0	1	**0**
0	1	0	**1**
0	1	1	**0**
1	0	0	**0**
1	0	1	**1**
1	1	0	**0**
1	1	1	**1**

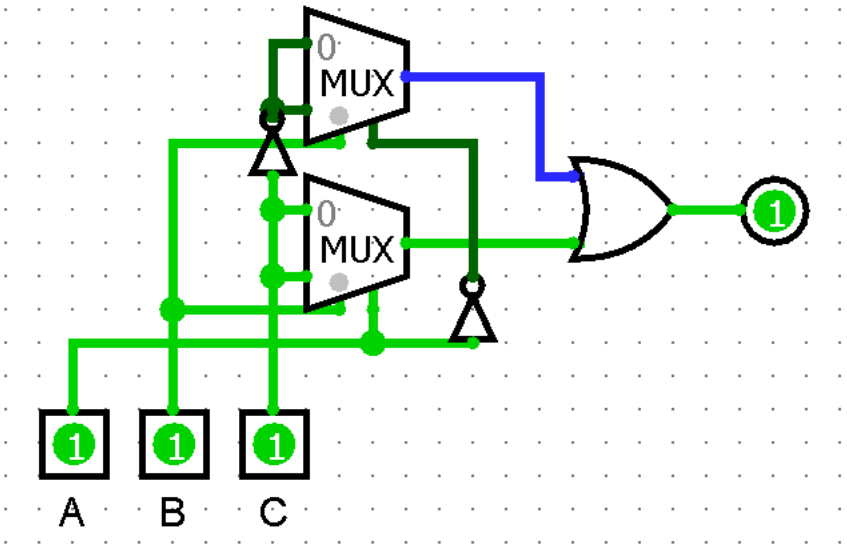

Figure 11 TT and Circuit Diagram

Input function : F (A, B, C) = A'B'C' + A'BC' + AB'C + ABC

Discussion:

Through this experiment, I learned a lot of new things about multiplexers. I found out how to build larger multiplexers by combining smaller ones and understood how the select key works. I also discovered some basics about the multiplexer IC and how to implement a multiplexer on a trainer board without using an IC. Additionally, I now have a clearer understanding of how and why multiplexers are used. Lastly, I learned how to design a logic circuit and write output equations based on a multiplexer's truth table.