**Experiment / Assignment / Tutorial No. 3**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

|  |
| --- |
| **Batch: A1 Roll No.: 1911004 Experiment / assignment / tutorial No.: 3** |

|  |
| --- |
| **Title: Design 4:1 Multiplexer and 1: 4 De-multiplexer** |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Objective: To design and implement a 4:1 multiplexer & 1:4 de-multiplexer using logic gates & MUX IC**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected Outcome of Experiment:**

**CO2: Use different minimization technique and solve combinational circuits, synchronous & asynchronous sequential circuits.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

* **VLab Links:** [**http://vlabs.iitb.ac.in/vlabs-dev/labs/dldesignlab/experimentlist.html**](http://vlabs.iitb.ac.in/vlabs-dev/labs/dldesignlab/experimentlist.html)
* **R. P. Jain, “Modern Digital Electronics”, Tata McGraw Hill**
* **M .Morris Mano, “Digital Logic & computer Design”, PHI**
* [**https://wiki.engr.illinois.edu/download/attachments/84770821/08**](https://wiki.engr.illinois.edu/download/attachments/84770821/08-Multiplexers.pdf?version=2&modificationDate=1285128827000)**-** [**Multiplexers.pdf?version=2&modificationDate=128512882700**](https://wiki.engr.illinois.edu/download/attachments/84770821/08-Multiplexers.pdf?version=2&modificationDate=1285128827000)**0**

**Pre Lab/ Prior Concepts:**

**Multiplexer:** Multiplexer is a special type of combinational circuit. It is a digital circuit which selects one of the n data inputs and routes it to the output. The selection of one of the n inputs is done by the select lines. To select n inputs we require m select lines, such that 2m=n. Depending on the digital code applied at the select inputs, one out of the n data sources is selected and transmitted to a single output . E is called as the strobe or enable input which is useful for cascading. It is generally on active low terminal that means it will perform the required operation when it is low. The multiplexer act like a digitally controlled single pole, multiple way switches. The output gets connected to only one input at a time. In most of the electronic system the digital data is available on more than one line. It is necessary to route the data over a single line, under such circumstances input at a time

Types of Multiplexer:

1. 2:1 Multiplexer
2. 4:1 Multiplexer
3. 8:1 Multiplexer
4. 16:1 Multiplexer
5. 32:1 Multiplexer

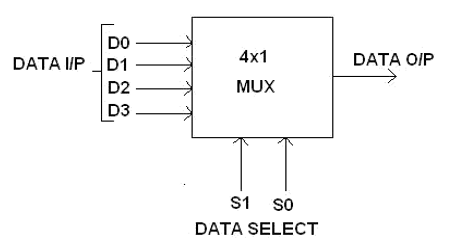
De-multiplexer: It has only one input, n output and m select lines. A demultiplexer performs the reverse operation of a multiplexer i.e. it receives one input and distributes it over several outputs. The demultiplexer converts a serial data signal at the input to a parallel data at its output lines. The relation between the output lines and select lines is as follows: N=2m

Types of Demultiplexers:

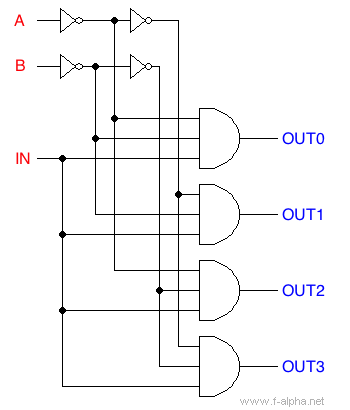
1. 1:2 DEMUX
2. 1:4 DEMUX
3. 1:8 DEMUX
4. 1:16 DEMUX

**Implementation Details of 4:1 MUX**

**Block Diagram of 4:1 MUX**

****

**Circuit Diagram of 4:1 MUX**

****

**Truth table**

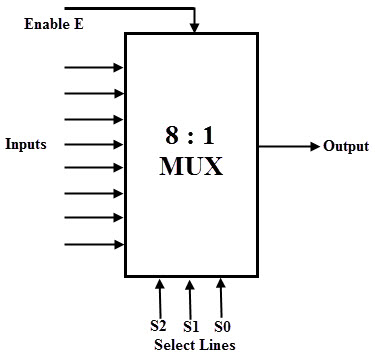
|  |  |  |
| --- | --- | --- |
| **S1** | **S0** | **Y** |
| **0** | **0** | **I0** |
| **0** | **1** | **I1** |
| **1** | **0** | **I2** |
| **1** | **1** | **I3** |

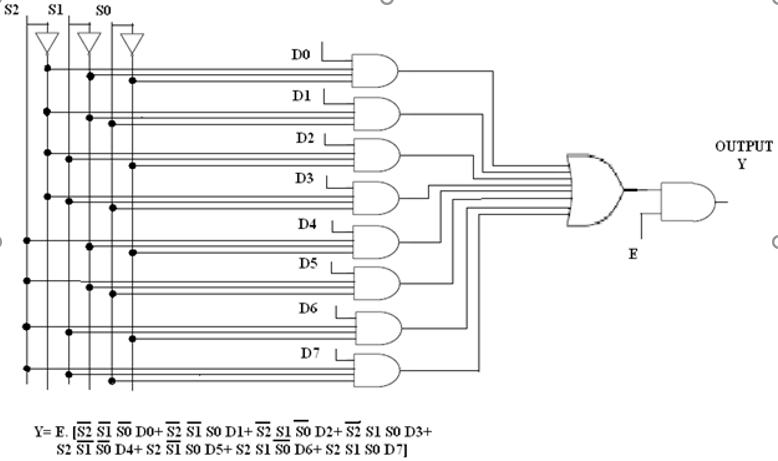
**From Truth Table:**

**Y = I 0 S 1 ’S 0 ’+ I 1 S 1 ’S 0 + I 2 S 1 S 0 ’+ I 3 S 1 S 0**

**Implementation Details of 8:1 MUX**

**Block Diagram & Circuit Diagram of 8:1 MUX**





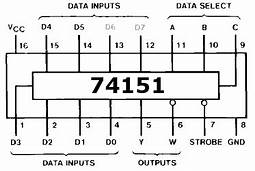
**Truth Table for 8:1 Multiplexer**

|  |  |  |  |
| --- | --- | --- | --- |
| **S2** | **S1** | **S0** | **Y** |
| **0** | **0** | **0** | **I0** |
| **0** | **0** | **1** | **I1** |
| **0** | **1** | **0** | **I2** |
| **0** | **1** | **1** | **I3** |
| **1** | **0** | **0** | **I4** |
| **1** | **0** | **1** | **I5** |
| **1** | **1** | **0** | **I6** |
| **1** | **1** | **1** | **I7** |

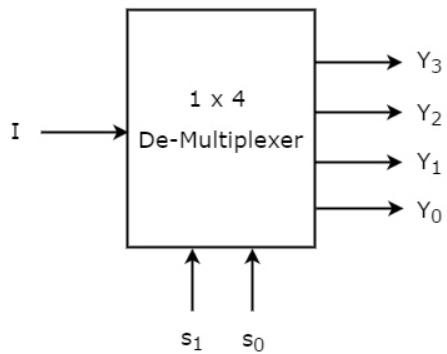
**From Truth Table:**

**Y = D0 S 2’S 1’S0’+ D1 S 2 ‘S 1 ‘S0 + D2 S 2 ‘S 1 S0 ‘+ D3 S 2 ‘S 1 S0 + D4 S 2 S 1 ‘S0 ‘+ D5 S 2 S 1 ‘ S0 + D6 S 2 S 1 S0’+ D7 S 2 S 1 S0**

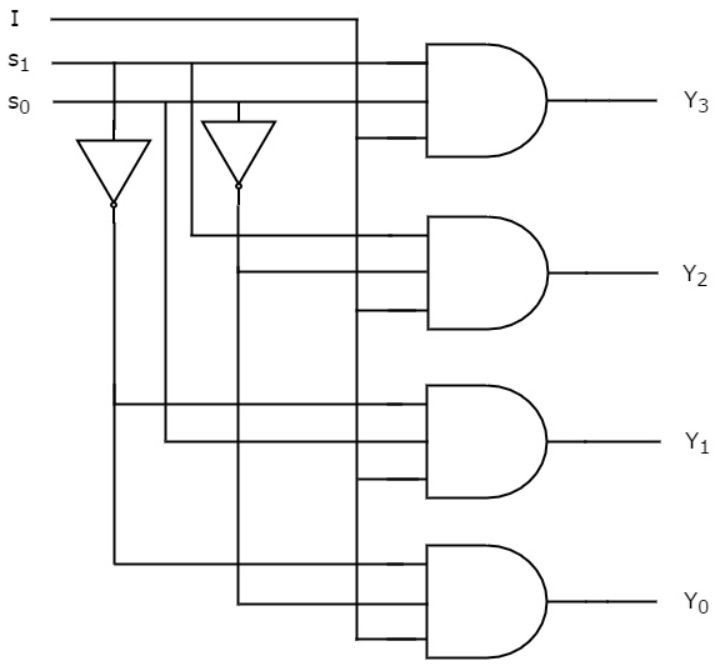
**Pin diagram: IC 74151**



**Block Diagram of 1:4 DE MUX**



**Circuit Diagram of 1:4 DE MUX**



**Truth Table for 1:4 Demultiplexers**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **INPUT** | **SELECT LINES** | | **OUTPUTS** | | | |
| **S1** | **S0** | **Y3** | **Y2** | **Y1** | **Y0** |
| **I** | **0** | **0** | **0** | **0** | **0** | **1** |
| **I** | **0** | **1** | **0** | **0** | **1** | **0** |
| **I** | **1** | **0** | **0** | **1** | **0** | **0** |
| **I** | **1** | **1** | **1** | **0** | **0** | **0** |

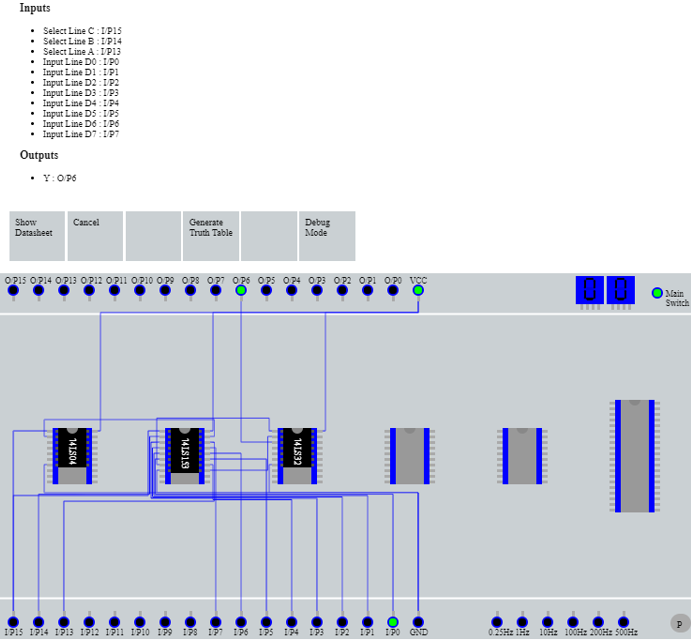
**From Truth Table:**

**Y3=S1 S0 I**

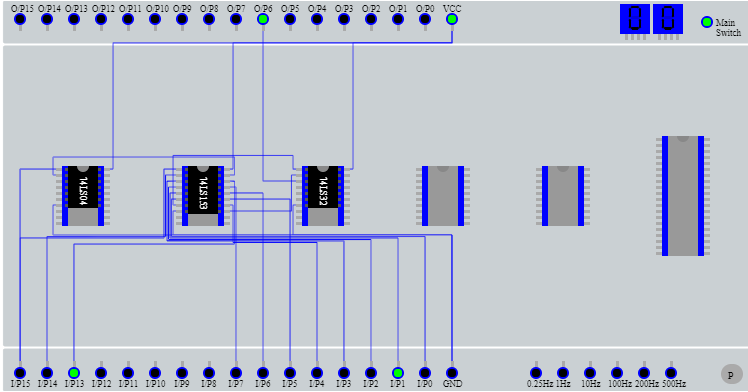
**Y2=** **S1 S0‘I**

**Y1= S1’ S0 I**

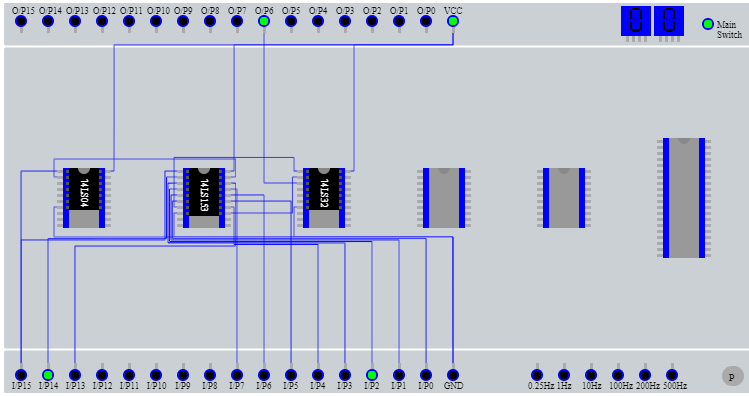
**Y0= S1’ S0’ I**

****

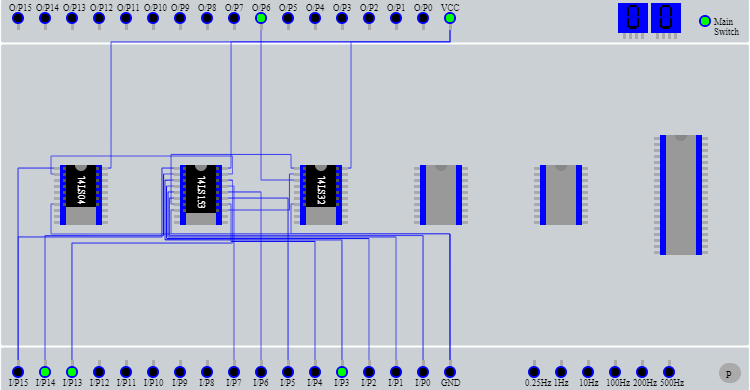
|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **0** | **0** | **0** | **D0 (I/P 0)** |

****

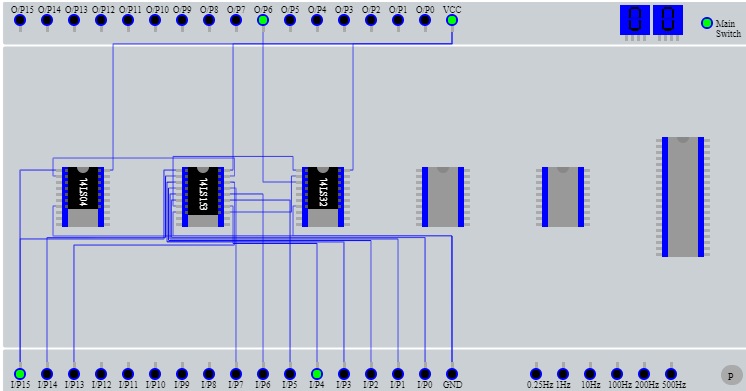
|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **0** | **0** | **1** | **D1 (I/P 1)** |

****

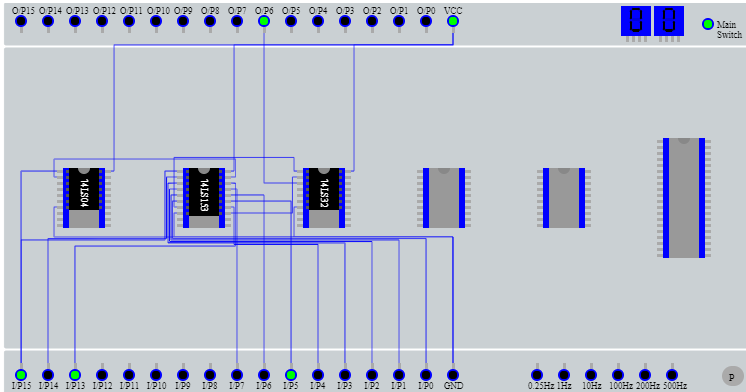
|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **0** | **1** | **0** | **D2 (I/P 2)** |

****

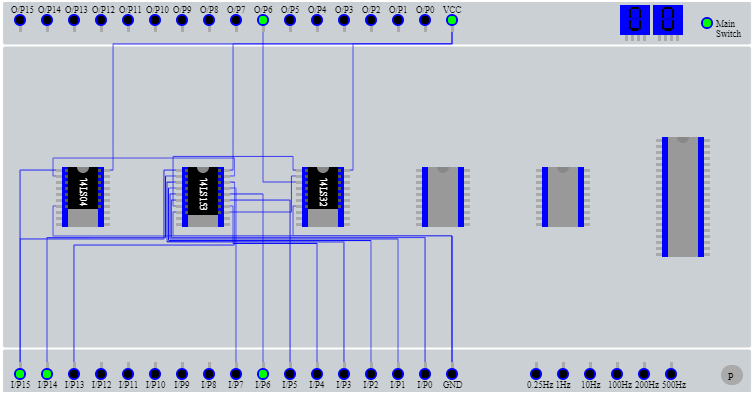
|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **0** | **1** | **1** | **D3 (I/P 3)** |

****

|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **1** | **0** | **0** | **D4 (I/P 4)** |

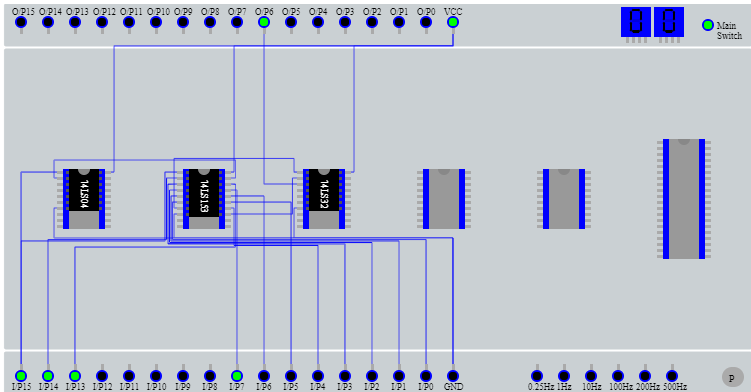
****

|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **1** | **0** | **1** | **D5 (I/P 5)** |

****

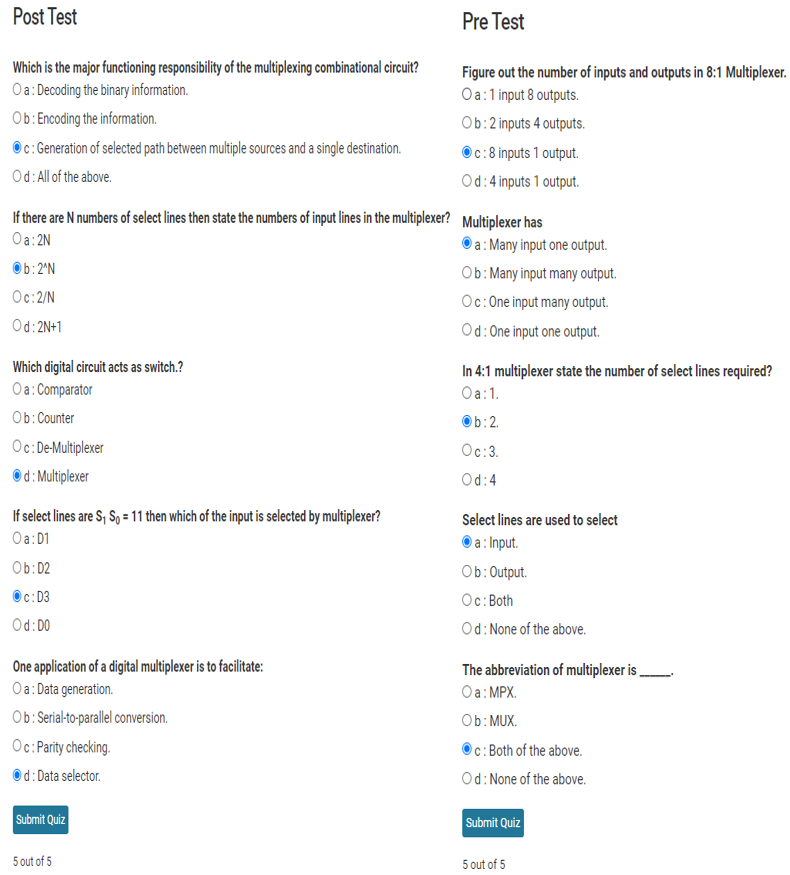
|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **1** | **1** | **0** | **D6 (I/P 6)** |

|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **1** | **1** | **1** | **D7 (I/P 7)** |

****

**TRUTH TABLE FOR VLAB**

|  |  |  |  |
| --- | --- | --- | --- |
| **S2 (I/P 15)** | **S1 (I/P 14)** | **S0 (I/P 13)** | **Y (O/P 6)** |
| **0** | **0** | **0** | **D0 (I/P 0)** |
| **0** | **0** | **1** | **D1 (I/P 1)** |
| **0** | **1** | **0** | **D2 (I/P 2)** |
| **0** | **1** | **1** | **D3 (I/P 3)** |
| **1** | **0** | **0** | **D4 (I/P 4)** |
| **1** | **0** | **1** | **D5 (I/P 5)** |
| **1** | **1** | **0** | **D6 (I/P 6 )** |
| **1** | **1** | **1** | **D7 (I/P 7)** |

****

**Conclusion:**

**We understood about the multiplexer (4:1 MUX, 8:1 MUX ) & de-multiplexer**

**(1:4 DEMUX) along with their circuits, truth table , Boolean Expression & implemented its concept in virtual lab and thus successfully verified their working for various test cases.**

**Post Lab Descriptive Questions**

1. **How many select lines are required for 64:1 MUX?**

**ANS :**

**We know that for 2n : 1 we need n select lines so thus by this for 64 :1 MUX we would need 6 Select lines from S0 - S5.**

1. **State some applications of MUX and DEMUX.**
2. **MUX**

* **In Computer Memory, Multiplexer (MUX) is used to maintain and store large amount of data & reduce the number of copper lines needed to connect the computer memory to other parts of the Computer System.**
* **In Communication systems, a Multiplexer is used to increase the efficiency of the communication system from different channels in cables & wires in Transmission & Communication Network System.**
* **It is used to generate logic function (i.e. logical expression & Boolean Algebraic Functions can be generated instead of logic gates).**

1. **DEMUX**

* **In Arithmetic logic unit (ALU), the output of ALU can be stored in storage unit (multiple registers) by using Demultiplexer. In this process, the output of ALU is connected as input to the Demultiplexer and the output of Demultiplexer connected to the registers to store the data.**
* **Demultiplexer is also used in serial to parallel converter. In this process, serial data has been connected as input to the demultiplexer at a regular interval. In addition, a counter is attached as control input to the Demultiplexer to detect the data signal of Demultiplexer outputs**
* **Demultiplexer (DEMUX) along with Multiplexer (MUX) is used in Communication systems to carry multiple data signals using single line for transmission. In this process, Demultiplexer receive the output data of Multiplexer (as a receiver) & coverts it back them to the original form .**

1. **Build a 4:1 MUX using only 2:1 MUX.**

