# Department of Computer and Software Engineering – ITU

# SE201L: Digital Logic Design Lab

| | |
|---|---|
| **Course Instructor: Ms Aqsa Khalid** | **Dated: 02-10-2024** |
| **Lab Engineer: Muhammad Kashif** | **Semester: Fall 2024** |
| **Batch: BSSE 23 A** | |

## LAB 7 Digital Multiplexers and Their Use in Combinational Logic Circuits

| Name | Roll Number | Lab Marks |
|---|---|---|
| BSSE23058 | BSSE23058 | |

Checked on: _____

Signature: _____

# Digital Multiplexers and Their Use in Combinational Logic Circuits

## 7.1. Introduction

This lab exercise introduces students to multiplexers, implementation of logic functions using multiplexers and cascading of smaller multiplexers to build larger multiplexer.

## 7.2. Objectives

This lab will enable the students to achieve the following:

- Learn about the internal structure and the use of multiplexers
- Cascade smaller size multiplexers to form large size multiplexers
- Implement Boolean functions using multiplexers
- Assess the size requirement of multiplexers to implement Boolean functions

## 7.3. Conduct of Lab

1. Task 1 of this lab has to be performed using ModelSim.
2. Bring printout of this lab manual when you come to perform the lab.
3. You can work and get evaluated in groups of two. However, manual submission has to be separate.
4. If there is difficulty in understanding any aspect of the lab, please seek help from the lab engineer or the TA.
5. If a lab task contains an instruction to show the work to lab engineer, make sure that the lab engineer evaluates and marks on your manual for that task. If your manual is unmarked for this task, it can result in mark deduction.

## 7.4. Theory and Background

### 7.4.1. Multiplexer

A multiplexer is a device that has multiple inputs, a few select lines, and one output. It selects one of the inputs according to the combination given to select lines and forwards it to the single output. A $2^n$ input digital multiplexer has n select lines.

Multiplexers are generally abbreviated as MUX. A 2-to-1-line MUX along with its symbol is shown Figure 7.1:

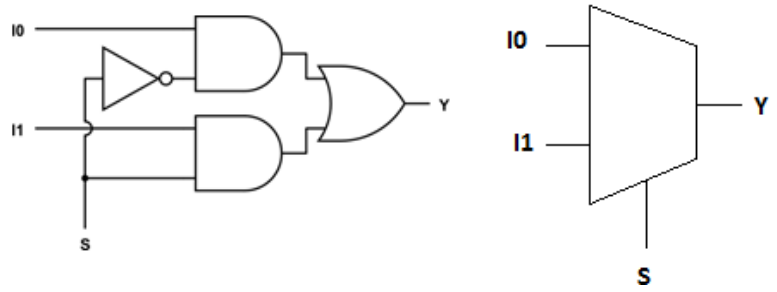



**Figure 7.1: A 2-to-1 MUX circuit with its symbol**

The truth table of the above is as follows:

**Table 7.1: Truth table of a 2-to-1 MUX**

| S | I0 | I1 | Y |
|---|---|---|---|
| **0** | 0 | X | 0 |
| **0** | 1 | X | 1 |
| **1** | X | 0 | 0 |
| **1** | X | 1 | 1 |

A 4-1 MUX will have 2 select lines and an 8-1 MUX will have 3 select lines, and so on.

Digital Multiplexers find applications in high frequency switching, parallel to serial data transmission etc. Use of multiplexers also saves hardware resources (copper wires) used to transmit data between memory unit and other parts of the computer.

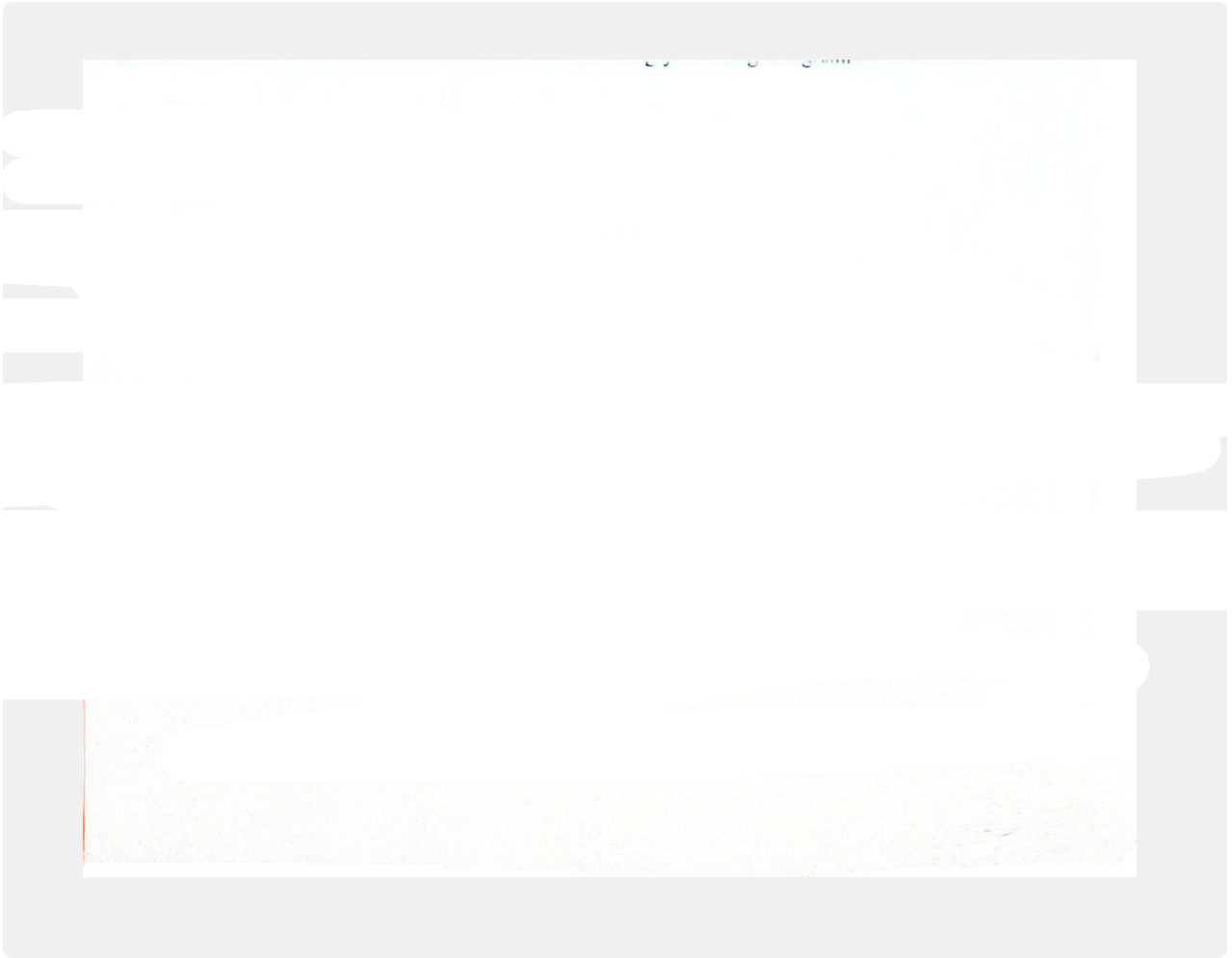
## 7.5. Parts and Equipment

1. Digital logic trainer
2. 74151: 8-line to 1-line Multiplexer
3. Connecting wires

## 7.6. Lab Tasks

### 7.6.1. Task 1: Cascading multiplexers [Marks: 30]

1. Suppose you only have 4-to-1-line multiplexers (as many as you want) and no logic gates. In the space given, propose a design diagram of implementing an 8-to-1 MUX using only 4-to-1 MUXes. Use the MUX device symbol ( such as the diagram on right of Figure 7.1 for 2-to-1 MUX) for the 4-to-1 line multiplexers. [10]
6. Give answer to question 1 of Analysis before drawing your design diagram.



7. Implement the 4-to-1 MUX in Verilog using behavioral modeling. [10]

```
module MUX_4_to_1 (
    input wire [1:0] sel,
    input wire [3:0] in,
    output wire out  );
```

```
    assign out = (sel == 2'b00) ? in[0] : (sel == 2'b01) ? in[1] : (sel == 2'b10) ? in[2] : in[3];
endmodule
```

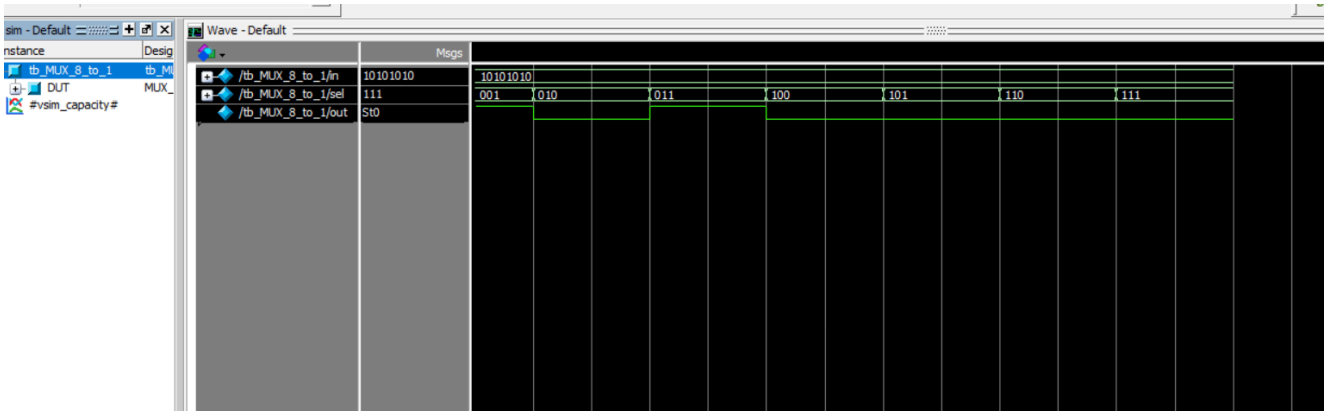
8. Use hierarchical modeling to construct the 8-to-1 MUX according to your proposed design. [5]

```
module MUX_8_to_1 (
    input wire [2:0] sel,
    input wire [7:0] in,
    output wire out
);
    wire out1, out2;

    MUX_4_to_1 mux1 (.sel(sel[1:0]), .in(in[3:0]), .out(out1));
    MUX_4_to_1 mux2 (.sel(sel[1:0]), .in(in[7:4]), .out(out2));

    MUX_4_to_1 mux3 (.sel({sel[2], 1'b0}), .in({2'b00, out2, out1}), .out(out));
endmodule
```

9. Code a test bench module to test the working of your 8-to-1 MUX. [5]

```
module tb_MUX_8_to_1;
    reg [7:0] in;
    reg [2:0] sel;
    wire out;
    MUX_8_to_1 DUT (.sel(sel), .in(in), .out(out));
    initial begin
        in = 8'b10101010;
        sel = 3'b000; #10;
        sel = 3'b001; #10;
        sel = 3'b010; #10;
        sel = 3'b011; #10;
        sel = 3'b100; #10;
        sel = 3'b101; #10;
        sel = 3'b110; #10;
        sel = 3'b111; #10;
        $stop;
    end
endmodule
```

10. Paste screenshots of code and simulation results in a WORD document on a single page. Print the document and attach it at the end of your lab manual.



### 7.6.2. Analysis [Marks: 5]

1. You might want to use 2-to-1 MUX to implement the required MUX, however, you only have 4-to-1 MUXes. How can you use a 4-to-1 MUX as a 2-to-1 MUX? Describe below with supporting diagram. [5]

# Assessment Rubric

**Method:**
Lab report evaluation and instructor observation during lab sessions.

| Performance | CLO | Able to complete the tasks over 80% (4-5) | Able to complete the tasks 50 – 80% (2-3) | Tasks completion below 50% (0-1) | Marks |
|---|---|---|---|---|---|
| 1. Teamwork | 1 | Actively engages and cooperates with other group members in an effective manner | Cooperates with other group members in a reasonable manner | Distracts or discourages other group members from conducting the experiments | |
| 2. Laboratory safety and disciplinary rules | 1 | Observes lab safety rules; handles the development board and other components with care and adheres to the lab disciplinary guidelines aptly | Observes safety rules and disciplinary guidelines with minor deviations | Disregards lab safety and disciplinary rules | |
| 3. Realization of experiment | 3 | Conceptually understands the topic under study and develops the experimental setup accordingly | Needs guidance to understand the purpose of the experiment and to develop the required setup | Incapable of understanding the purpose of the experiment and consequently fails to develop the required setup | |
| 4. Conducting experiment | 3 | Sets up hardware/software properly according to the requirement of experiment and examines the output carefully | Makes minor errors in hardware/software setup and observation of output | Unable to set up experimental setup, and perform the procedure of experiment | |
| 5. Data collection | 3 | Completes data collection from the experiment setup by giving proper inputs and observing the outputs, complies with the instructions regarding data entry in manual | Completes data collection with minor errors and enters data in lab report with slight deviations from provided guidelines | Fails at collecting data by giving proper inputs and observing output states of experiment setup, unable to fill the lab report properly | |
| 6. Data analysis | 3 | Analyzes the data obtained from experiment thoroughly and accurately verifies it with theoretical understanding, accounts for any discrepancy in data from theory with sound explanation, where asked | Analyzes data with minor error and correlates it with theoretical values reasonably. Attempts to account for any discrepancy in data from theory | Unable to establish the relationship between practical and theoretical values and lacks the theoretical understanding to explain any discrepancy in data | |
| 7. Computer use | 3 | Successfully uses lab PC and internet to look for relevant datasheets, carry out calculations, or verify results using simulation | Requires assistance in looking for IC datasheets and carrying out calculation and simulation tasks | Does not know how to use computer to look up datasheets or carry out calculation and simulation tasks | |
| | | | | **Total** (out of 35) | |