COL215 Assignment 1 part b
Hardware Assignment

Deadline: 10th August 2024

1 Introduction

The objective of the assignment is to make 2x1 multiplexer using basic gates (AND, OR and NOT gate) and extend it to make 4x1 multiplexer.

2 Overview of 2x1 Multiplexer

A multiplexer (Mux) is a logic device with multiple input ports and 1 output port, that directs one of its input ports onto the output port. The input to be sent to the output is indicated by selector ports. A 2x1 multiplexer has 2 input ports, 1 select port, and 1 output port, as shown in Figure 1. The logic value at either $d1$ or $d2$ is sent to o, depending on whether s is 0 or 1.

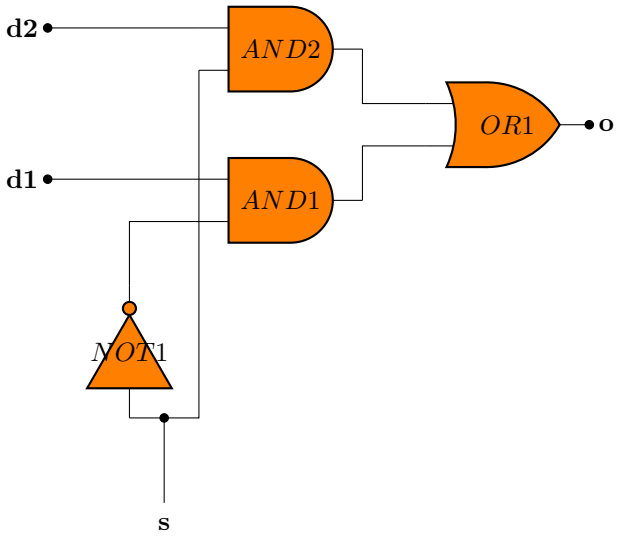

Figure 1: Circuit diagram of 2x1 MUX

3 Port mapping in VHDL

To connect different entities in VHDL, you need to declare the entities as a components in the architecture of the *top* entity, create *instances* of the entities, and further connect the ports using *signals* through VHDL *port mapping* statements

Using the testbench from the previous assignment as an example, we declare **AND_gate** as a component in the architecture of testbench, and instantiate it using a port mapping statement. The first way to connect is

called *named mapping* where **DUT1** is the label for the instantiated AND_gate and the signal **x** is connected to input port **a**; signal **z** to output port **c**; and signal **y** to input port **b**. Another way to connect is through *positional mapping* shown below; the order of ports determines the signal/port connection. The order of port declarations in the component is: **a, b, c**. So, **x** is mapped to **a**, **y** to **b**, and **c** to **z**.

```
architecture tb of AND_gate_tb is
    component AND_gate
      Port ( a : in STD_LOGIC;
             b : in STD_LOGIC;
             c : out STD_LOGIC);
    end component;
    signal x, y : std_logic;  -- inputs
    signal z : std_logic;  -- outputs
....
    -- connecting testbench signals with AND_gate.vhd
    DUT1 : AND_gate port map (a => x, c => z, b => y);

    DUT2 : AND_gate port map (x, y, z);
```

Another example port mapping for two entities is shown in Figure 2. We have two entities instantiated as **DUT1** and **DUT2**. You can instantiate and port map in two ways, as shown in the code snippet. For instance DUT1, the order of the port mapping clauses is different from the port order declared in the component **new_entity**, whereas for instance DUT2, signals are port-mapped based on the port declaration order: **m** to **x**, **C** to **y**, and **Output** to **z**.

```
entity top_entity is
    Port ( A : in STD_LOGIC;
           B : in STD_LOGIC;
           C : in STD_LOGIC;
           Output : out STD_LOGIC);
end top_entity;

architecture tb of top_entity is
    component new_entity
      Port ( x : in STD_LOGIC;
             y : in STD_LOGIC;
             z : out STD_LOGIC);
    end component;
    signal m : std_logic;  -- inputs
begin
    DUT1 : new_entity port map (z => m, y => B, x => A);

    DUT2 : new_entity port map (m, C, Output);
end
```

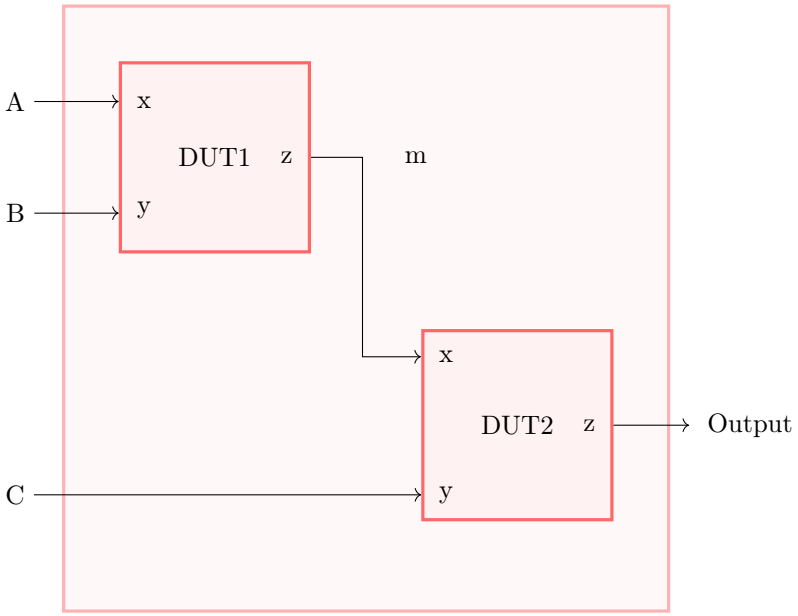

Figure 2: Port mapping example (top_entity)

4 Lab task

There are two tasks in the current assignment:

1. Implement a 2x1 Mux using AND, OR, and NOT modules created in the previous lab.
2. Implement 4x1 Mux using 2x1 Muxes.

On the Basys 3 board, you need to map two input switches to the input signals of a Mux, one switch to the Mux select signal, and one LED to the Mux's output signal. Determine number of switches and LED for the 4x1 Mux. You are free to select any switches and LED on the board.

Note: The VHDL model needs to be *structural*, that is, it consists of entity instantiations. Statements such as *process*, *if-else*, or *case*, are not allowed.

5 Assignment Submission Instructions

General assignment instructions that need to be followed for all assignments: only one partner needs to submit. Mention all team member names and entry IDs during the submission.

1. Name the submission file as entryNumber1_entryNumber2.zip or entryNumber1.zip
2. Go to Gradescope via moodle and upload the file under Hardware Assignment 1.
3. Only one submission per group is required. Gradescope will allow you to select the group partner.
4. The following files should be part of the zip folder:
   - Source files (.vhd) including test bench files.
   - Constraint Files (.xdc) for 2x1 and 4x1 Mux.

- Bit files (.bit) for 2x1 and 4x1 Mux.
- Report as a .pdf file (handwritten report will be rejected). The report needs to state:
  - your design decisions (if any)
  - simulation snapshots for AND, OR, NOT, 2x1 mux, and 4x1 mux
  - schematic snapshot for 2x1 mux and 4x1 mux
  - resource utilization table (including resource counts: Flip-flops, LUTs, BRAMs, and DSPs)