CSc 21100 (Fall 2019)
Project 02 (30 points)

IMPORTANT!

Please follow the **submission guidelines** below or your submission will be rejected.

1. You are expected to submit both a lab report and the source files to Blackboard in a single submission attempt.
2. The VHDL project needs to be exported from Xilinx ISE Design Suite. To export VHDL project file, please refer to Blackboard -> Content -> Lab -> **Exporting VHDL Project Files**.
3. Naming convention:

   Report: "FirstName_LastName_Project_**XX**_MM**Y**.pdf"

   Project: "FirstName_LastName_Project_**XX**_MM**Y**.zip"

   Replace "XX" and "Y" with the actual project number and section number, respectively.

In this project, students are expected to use the Xilinx ISE Design Suite (Webpack edition) 14.7 to complete the following tasks.

Please read the instructions carefully. Failing to follow the instructions would lead to significant point deductions.

Task 1: 2-to-4 Decoder (15 points)

A 2-to-4 decoder operates according to the following function table.

Inputs			*Outputs*			
EN	I1	I0	Y3	Y2	Y1	Y0
0	x	x	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

Table 6-4

Truth table for a 2-to-4 binary decoder.

Implement a 2-to-4 decoder in VHDL using *structural design*: Please adopt the following as the entity declaration.

```
entity v2to4dec is
   port ( EN : in    std_logic;
          I0 : in    std_logic;
          I1 : in    std_logic;
          Y0 : out   std_logic;
          Y1 : out   std_logic;
          Y2 : out   std_logic;
          Y3 : out   std_logic);
end v2to4dec;
```

Write a test-bench program and run simulations to validate your design: Use the given test cases below in your test-bench program. Pay attention to the signal names, signal values, and the time.

```
-- insert stimulus here
EN <= '0';
I1 <= '0'; I0 <= '0'; wait for 10 ns;
I1 <= '0'; I0 <= '1'; wait for 10 ns;
I1 <= '1'; I0 <= '0'; wait for 10 ns;
I1 <= '1'; I0 <= '1'; wait for 10 ns;
EN <= '1';
I1 <= '0'; I0 <= '0'; wait for 10 ns;
I1 <= '0'; I0 <= '1'; wait for 10 ns;
I1 <= '1'; I0 <= '0'; wait for 10 ns;
I1 <= '1'; I0 <= '1'; wait for 10 ns;
```

Requirement(s):

In your VHDL implementation, please follow the structural design method.

Deliverable(s):

Your report:

1. Use your own language to describe the function of the module to be implemented in VHDL. (2 point)
2. Draw a circuit diagram of the module to show the design. (2 point)
3. Include your VHDL entity declaration(s), architecture definition(s) and the testbench program. (2 points)
4. Show simulation results (e.g. the waveforms). Explain the outcome of **each** test case with screenshots. Show why the simulation result is correct. (5 points)

Your project file(s):

1. Can compile without any errors. (2 points)
2. Can run simulations without any errors. (2 points)

Note: no points will be given if requirements are not satisfied.

Task 2: 3-to-8 Decoder (15 points)

Implement a 3-to-8 decoder using the 2-to-4 decoder you have implemented in Task 1. The input to the 3-to-8 decoder should be labeled as ***A***. It needs to be a 3-bit bus. The output should be labeled as ***O***, which is an 8-bit bus. Please use the following entity declaration for the 3-to-8 decoder.

```
entity v3to8dec is
   port ( A  : in    std_logic_vector (2 downto 0);
          EN : in    std_logic;
          O  : out   std_logic_vector (7 downto 0));
end v3to8dec;
```

Note that: (1) please adopt the *structural design approach*; (2) in addition to the 2-to-4 decoders, you also need to include some logic gates to make it work.

Write a test-bench program and run simulations to validate your design. Use the given test cases below in your test-bench program. Pay attention to the signal names, signal values, and the time.

```
EN<='0'; A<="000"; wait for 10 ns;
EN<='0'; A<="001"; wait for 10 ns;
EN<='0'; A<="010"; wait for 10 ns;
EN<='0'; A<="011"; wait for 10 ns;
EN<='0'; A<="100"; wait for 10 ns;
EN<='0'; A<="101"; wait for 10 ns;
EN<='0'; A<="110"; wait for 10 ns;
EN<='0'; A<="111"; wait for 10 ns;

EN<='1'; A<="000"; wait for 10 ns;
EN<='1'; A<="001"; wait for 10 ns;
EN<='1'; A<="010"; wait for 10 ns;
EN<='1'; A<="011"; wait for 10 ns;
EN<='1'; A<="100"; wait for 10 ns;
EN<='1'; A<="101"; wait for 10 ns;
EN<='1'; A<="110"; wait for 10 ns;
EN<='1'; A<="111"; wait for 10 ns;
```

Requirement(s):
In your VHDL implementation
(1) Please follow the structural design method;
(2) Make use of the module you implemented in Task1.

Deliverable(s):

Your report:

1. Use your own language to describe the function of the module to be implemented in VHDL. (2 point)
2. Draw a circuit diagram of the module to show the design. (2 point)
3. Include your VHDL entity declaration(s), architecture definition(s) and the testbench program. (2 points)
4. Show simulation results (e.g. the waveforms). Explain the outcome of **each** test case with screenshots. Show why the simulation result is correct. (5 points)

Your project file(s):

1. Can compile without any errors. (2 points)
2. Can run simulations without any errors. (2 points)

Note: no points will be given if requirements are not satisfied.