PART 2: DIGITAL DESIGN LAB1

INTRODUCTION

In this lab, you are required to use Vivado 2017.4 to design a simple logic circuit: Do the addition on two signed 2bit numbers, do the simulation and verify its function by simulation. You should submit the description of the operation steps, the Verilog code of the design and test-bench, the waveform from the simulation, and the function verification results.

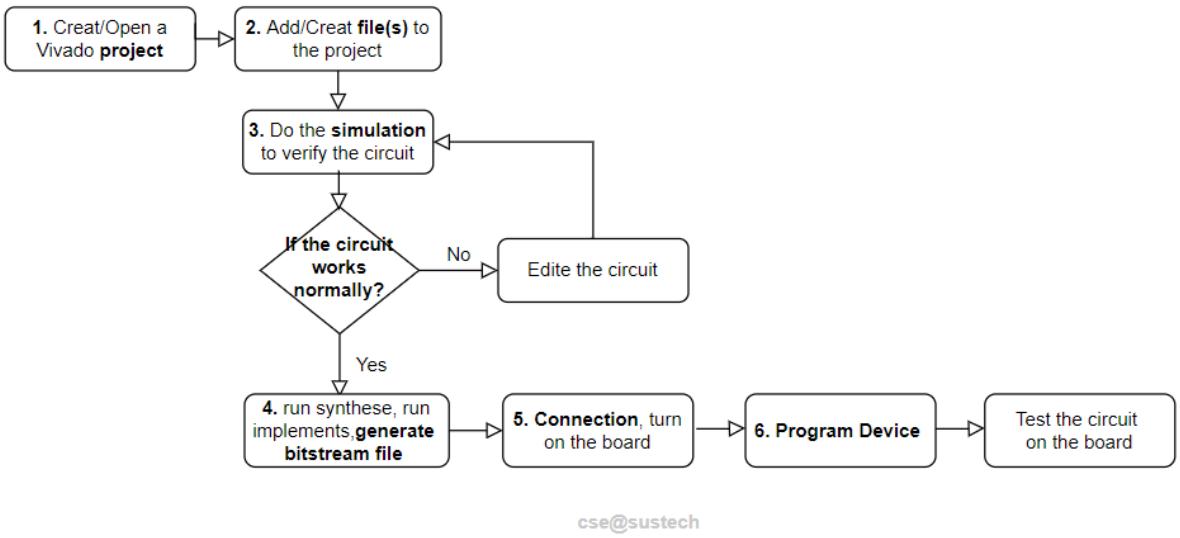
PREAMBLE

Before working on the coursework itself, you should master the following material. A separate tutorial document (on the Sakai site) has be provided to you which includes:

- Vivado: The Vivado software provides a complete design environment for system-on-a-programmable-chip (SOPC) design. Regardless of whether you use a personal computer or a Linux workstation, Vivado ensures easy design entry, fast processing, and straightforward device programming.
- Minisys/EGO1 Practice platform: a practice platform designed for Digital design, Principles of Computer Organization and many other courses. This platform include FPGA chip, storage chip and lots of Dial switches for input and lots of LEDs for output.
- Verilog : standardized as IEEE 1364, is a hardware description language (HDL) used to model electronic systems. It is most commonly used in the design and verification(test-bench) of digital circuits at the RTL(Register-Transfer-Level) of abstraction. It is also used in the verification of analog circuits and mixed-signal circuits, as well as in the design of genetic circuits. http://www.verilog.com/

EXERCISE SPECIFICATION

In practical part, you will focus on how to use Vivado to do the design, simulation, generate the bitstream file which is used to program the FPGA chip (a part of the Minisys/EGO1 practice platform), you will also learn some basic concepts of Verilog. The steps you need to follow are:



In all the tasks of assignment1, you should just focus on step 1~3. If you have got the practice board, your are strongly suggested to finish all the steps to implement your designed circuit on the FPGA chip and test it by the practice board.

IMPORTANT:

When writing the truth-table, the table must meet the following rules from top to bottom: the input value must start from 0 and increase by 1 and traverse all States. If this condition is not met, points will be deducted.

TASK1 (30 marks):

Create a project named as UnsignedMultiplier, design the source code to implement following function: **get the two input numbers of the multiplication, and output the product**. The bit-width of the inputs and 2, the bit-width of production is 4. **Do the simulation and to verify the function of the circuit design**.

1. the truth-table while the bit-width of inputs is 2bit.	8 marks
2. Design in Verilog	12 marks
3. Test bench in Verilog	5 marks

4. simulation result(waveform and its description)	5 marks

Note

(1) Two input and output numbers here should be treated as **unsigned number**. For example, the value of $(11)_2$ is $(3)_{10,}$ the value of $(10)_2$ is $(2)_{10.}$

(2) There should be **two inputs and one output** (the output is used to demonstrate the value of the sum of two inputs) //following module is just a simplified demo. **It is NOT allowed using the operator "*" of Verilog in the design of TASK1**.

```
module UnsignedMultiplier(in1, in2, product_led);
input [1: 0] in1;
Input [1: 0] in2;
output [?: 0]   product_led;
……
Endmodule
```

(3) The Simulation is asked to cover all the test cases, explain every test case according to the waveform with the corresponding arithmetic expression.

For example, from the waveform, the inputs are $(10)_2$ and $(11)_2$, the output is $(110)_2$. This is ok, because $2((10)_2) * 3((11)_2) = 6((110)_2)$.

(4) Redundant simulation is asked to be avoid. If the needed simulation time is smaller than 1000ns(the default simulation time), the simulation should NOT last for 1000ns. For example, if there are 4 test-cases in total, each test-case last for 10ns, the simulation time should be 40ns instead of 1000ns.

TASK2 (45 marks):

1) Do the design using data flow(While doing the design with data flow style, both the keyword "assign" and the operators in verilog are suggested) and **structured style respectively** (While doing the design with structured style, using primitive gate is suggested) **to prove that the DeMorgan theorem holds** (you can find the design on the lab3 courseware as a reference) while both the operands are 2bits width:

DeMorgan: $(A + B)' = A'B'$ and $(AB)' = A' + B'$

2) Create a test bench, do the simulation to verify the function of the design.

3) Answer the question: is the DeMorgan theorem still true while the operands are 2bits width?

1: The truth-table(2bit bit-width inputs)	8 marks
2: Design with data-flow style	10 marks
3: Design with structure-design style	10 marks
4: Test bench in Verilog (1 testbnech is enough)	5 marks
5: simulation result(waveform and its description), and your answer about the DeMorgan theorem.	10+2 marks

NOTE:

1) Naming the designed file and the module:

   a. For data flow design, the source file should be DeMorgan2bit_df.v

   b. For structured design, the source file should be DeMorgan2bit_sd.v

2) All the design should share the same test bench file.

3) **The Simulation is asked to cover all the test cases, while redundant simulation is asked to be avoid**. If the needed simulation time is smaller than 1000ns(the default simulation time), the simulation should NOT last for 1000ns. For example, if there are 4 test-cases in total, each test-case last for 10ns, the simulation time should be 40ns instead of 1000ns.

TASK3 (25 marks):

A circuit is expressed as sum of Minimum term: F(A, B, C, D) = $\sum$(0, 2, 3, 6, 7, 10, 11, 12, 13, 15) (expression1), write down its product of Maximum item(expression2), using a Karnaugh map to simplify the circuit in sum-of-product(expression3).

1) Create a circuit with 4 inputs(A,B,C,D), 3 outputs(each one output for one expression described above), using data flow(While doing the design with data flow style, both the keyword “assign” and the operators in verilog are suggested) are suggested.

2) Create a test bench, do the simulation to improve the 3 expression(expression1, expression2 and expression3) are the same circuit.

1: The 3 expressions and Design with data-flow style	10 marks
2: Test bench in Verilog (1 testbnech is enough)	5 marks
3: Simulation result(waveform and its description)	10 marks

The template for the report is provided in the next pages.

DIGITAL DESIGN

ASSIGNMENT REPORT

ASSIGNMENT ID : xxxx

Student Name: xxxx

Student ID: xxxx

PART 1: DIGITAL DESIGN THEORY

Provide your answers here:

PART 2: DIGITAL DESIGN LAB (TASK1)

DESIGN

Describe the design of your system by providing the following information:

- *Verilog design (provide the Verilog code)*
- *Truth-table*

SIMULATION

Describe how you build the test bench and do the simulation.

- *Using Verilog(provide the Verilog code)*
- *Wave form of simulation result (provide screen shots)*
- *The description on whether the simulation result is same as the truth-table, is the function of the design meet the expectation.*

THE DESCRIPTION OF OPERATION

Describe the problem occurred while in the lab and your solution. Any suggestions are welcomed.

- *Problems and solutions*

PART 2: DIGITAL DESIGN LAB (TASK2)

DESIGN

Describe the design of your system by providing the following information:

- *Verilog design while using data flow (provide the Verilog code)*
- *Verilog design while using structured design (provide the Verilog code)*

- *Truth-table*

SIMULATION

Describe how you build the test bench and do the simulation.

- *Using Verilog (provide the Verilog code)*
- *Wave form of simulation result (provide screen shots)*
- *The description on whether the simulation result is same as the truth-table, is the function of the design meet the expectation*

THE DESCRIPTION OF OPERATION

Describe the problem occurred while in the lab and your solution. Any suggestions are welcomed.

- *Problems and solutions*