Experiment 2 (Implementation of a Boolean Expression)

Aim

In this experiment, your knowledge to minimize the number of gates by using decoder and multiplexer gate which is explained in section 2.9 of the course book will be tested.

Problem

A circuit takes four inputs denoted as x_3, x_2, x_1, and x_0. This circuit has only one output y. The output is determined as shown in the following table.

x_3	x_2	y
0	0	GT2?(x_1, x_0)
0	1	ODD?(x_1, x_0)
1	0	PRIME?(x_1, x_0)
1	1	NEQ?(x_1, x_0)

We define GT2, ODD, PRIME, and NEQ as follows. x_1x_0 represents a two bit unsigned number.

$$GT2?(x_1, x_0) = \begin{cases} 1 & x_1x_0 \text{ is greater than 2} \\ 0 & \text{otherwise} \end{cases}$$

$$ODD?(x_1, x_0) = \begin{cases} 1 & x_1x_0 \text{ is odd} \\ 0 & \text{otherwise} \end{cases}$$

$$PRIME?(x_1, x_0) = \begin{cases} 1 & x_1x_0 \text{ is prime} \\ 0 & \text{otherwise} \end{cases}$$

$$NEQ?(x_1, x_0) = \begin{cases} 1 & x_1 \neq x_0 \\ 0 & \text{otherwise} \end{cases}$$

You are only allowed to use **one 2x4 Decoder, one 4x1 Multiplexer, at most 2 binary logic gates (AND, OR, XOR, XNOR, NAND, NOR) and infinite number of NOT gates** to complete the design of this circuit.

Preliminary Work

Before the experiment, you should prepare following materials:

1. Fill the truth table.
2. Draw the circuit of your design.
3. Write a verilog code for the circuit drawn in the previous step. Verilog code should have two components. First, you have to write behavioural level verilog code which implements the functionality of multiplexer and/or decoder. Then, you have to write the gate level verilog code (source.v) for your circuitry which uses the implemented components and additional **built-in** gates (i.e. *AND*, *OR*, *NAND*, *NOR*, *XOR*, or *XNOR*).

4. Write the verilog code for the testbench (testbench.v) in order to test all possible input combinations.

Then, submit the your code, and your report under the name `<StudentID>_PRE2.zip` through Moodle.