**Department of Electrical Engineering**

**Faculty Member:** Arshad Nazir **Dated:** 26/11/2021

**Semester:** 3rd **Section:** BEE 12C

**Group No.: 7**

**EE-221: Digital Logic Design**

Lab 9: XS – 3 To Gray Code Using NAND Gates

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **PLO4/CLO4** | **PLO4/CLO4** | **PLO5/CLO5** | **PLO8/CLO6** | **PLO9/CLO7** |  |
| **Name** | **Reg. No** | **Viva / Lab Performance** | **Analysis of data in Lab Report** | **Modern Tool Usage** | **Ethics and Safety** | **Individual and Teamwork** | **Total marks Obtained** |
|  |  | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** | **25 Marks** |
| **Muhammad Adil Azeem** | **342459** |  |  |  |  |  |  |
| **Muhammad Umer** | **345834** |  |  |  |  |  |  |
| **Raahim Ahmad Waqar** | **342287** |  |  |  |  |  |  |
| **Saad Bakhtiar** | **341150** |  |  |  |  |  |  |
| **Syed Aun Ali Kazmi** | **342384** |  |  |  |  |  |  |

## Lab 9: XS – 3 To Gray Code

This Open-ended Lab has been divided into two parts:

In first part you are required to **design** and **implement** a Excess-3 to gray code converter.

The next part is the Verilog Modeling and **Simulation** of the Circuit you implemented in you first part.

**Objectives:**

* Understand steps involved in design of combinational circuits
* Understand binary codes for decimals and their hardware realization
* Write code for combinational circuits using Verilog Gate Level Modeling
* Design a circuit in Verilog by calling different modules

**Lab Instructions**

* This lab activity comprises three parts, namely Pre-lab, Lab tasks, and Post-Lab Viva session.
* The lab report will be uploaded on LMS three days before scheduled lab date. The students will get hard copy of lab report, complete the Pre-lab task before coming to the lab and deposit it with teacher/lab engineer for necessary evaluation. Alternately each group to upload completed lab report on LMS for grading.
* The students will start lab task and demonstrate design steps separately for stepwise evaluation (course instructor/lab engineer will sign each step after ascertaining functional verification)
* Remember that a neat logic diagram with pins numbered coupled with nicely patched circuit will simplify trouble-shooting process.
* After the lab, students are expected to unwire the circuit and deposit back components before leaving.
* The students will complete lab task and submit complete report to Lab Engineer before leaving lab.
* **The Total duration for the lab is 3 hrs. After lab duration, a deduction of 5 marks per day will be done for late submission.**
* **A lab with in-complete lab tasks will not be accepted.**
* There are related questions at the end of this activity. Give complete answers.

**Pre – Lab Tasks**

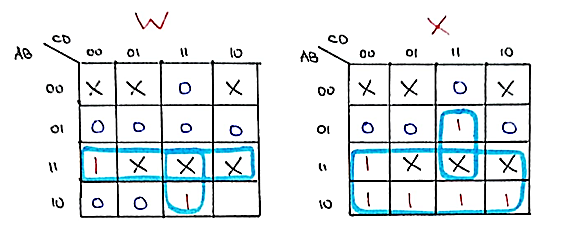
**Pre – Lab Task 1**

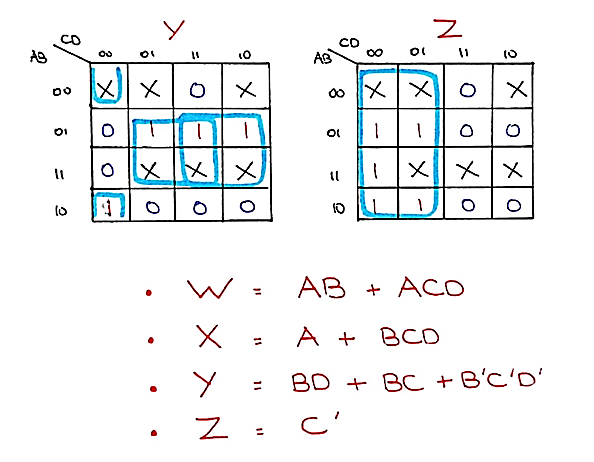
**In the lab you would be implementing an Excess-3 to gray code converter. Make a truth table for both the codes by filling in the following tables and simplify the expressions for W, X, Y, Z in terms of A, B, C, D. (Use backside of the page if necessary).**

**XS - 3 to Gray Converter**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dec** | **XS - 3** | | | | **Gray Code** | | | |
| **A** | **B** | **C** | **D** | **W** | **X** | **Y** | **Z** |
|  | 0 | 0 | 0 | 0 | X | X | X | X |
|  | 0 | 0 | 0 | 1 | X | X | X | X |
|  | 0 | 0 | 1 | 0 | X | X | X | X |
| **0** | **0** | **0** | **1** | **1** | **0** | **0** | **0** | **0** |
| **1** | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **1** |
| **2** | **0** | **1** | **0** | **1** | **0** | **0** | **1** | **1** |
| **3** | **0** | **1** | **1** | **0** | **0** | **0** | **1** | **0** |
| **4** | **0** | **1** | **1** | **1** | **0** | **1** | **1** | **0** |
| **5** | **1** | **0** | **0** | **0** | **0** | **1** | **1** | **1** |
| **6** | **1** | **0** | **0** | **1** | **0** | **1** | **0** | **1** |
| **7** | **1** | **0** | **1** | **0** | **0** | **1** | **0** | **0** |
| **8** | **1** | **0** | **1** | **1** | **1** | **1** | **0** | **0** |
| **9** | **1** | **1** | **0** | **0** | **1** | **1** | **0** | **1** |
|  | 1 | 1 | 0 | 1 | X | X | X | X |
|  | 1 | 1 | 1 | 0 | X | X | X | X |
|  | 1 | 1 | 1 | 1 | X | X | X | X |

**Karnaugh Maps for Simplified Expressions**



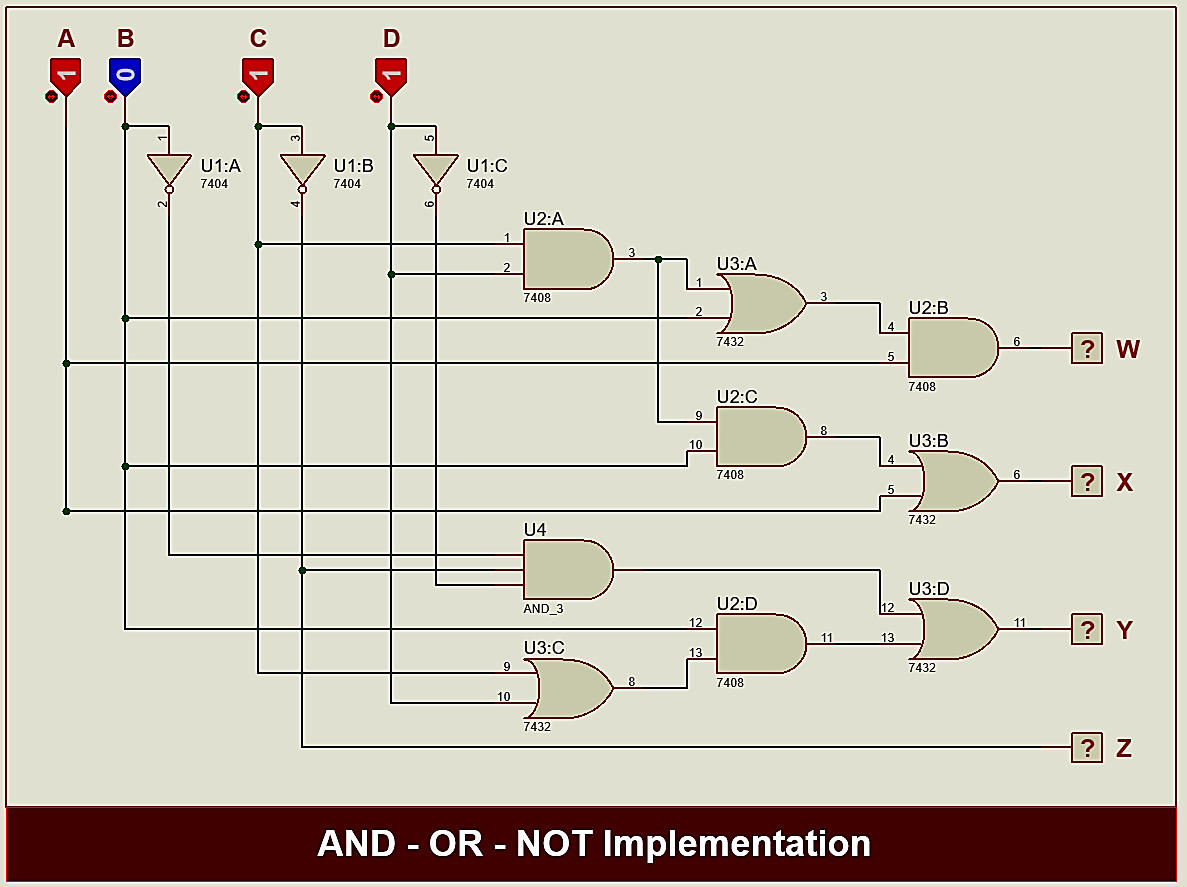


**Simplified Expressions**

|  |  |  |
| --- | --- | --- |
| **W** | **AB + ACD** | **A (B + CD)** |
| **X** | **A + BCD** | **A + B (CD)** |
| **Y** | **B’C’D’ + BD + BC** | **B’C’D’ + B (C + D)** |
| **Z** | **C’** | **C’** |

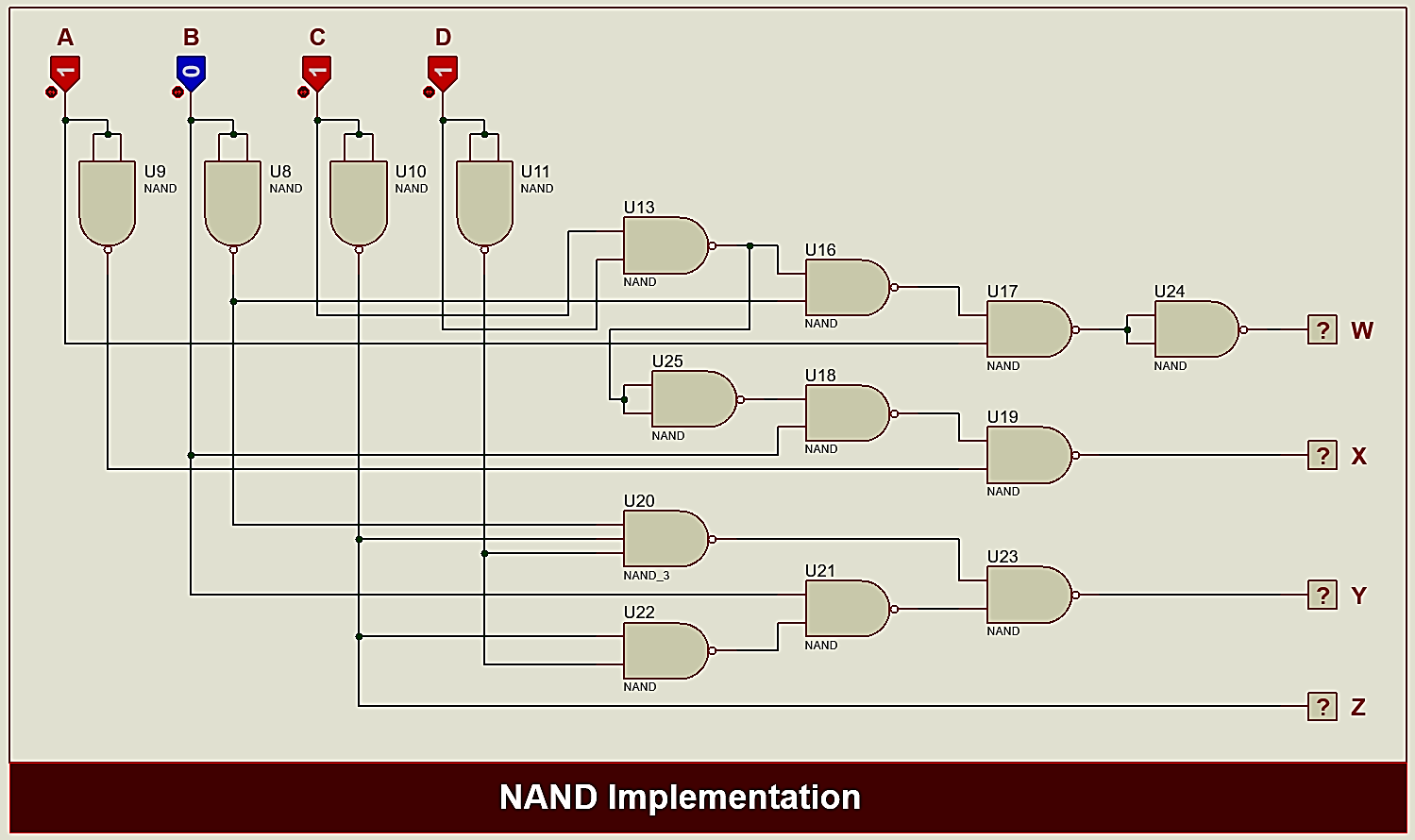
**Pre – Lab Task 2**

**Draw the logic diagram for Excess-3 to gray code converter using AND, OR and NOT gates in the space provided below. You can use 2, 3, 4 input gates if required.**



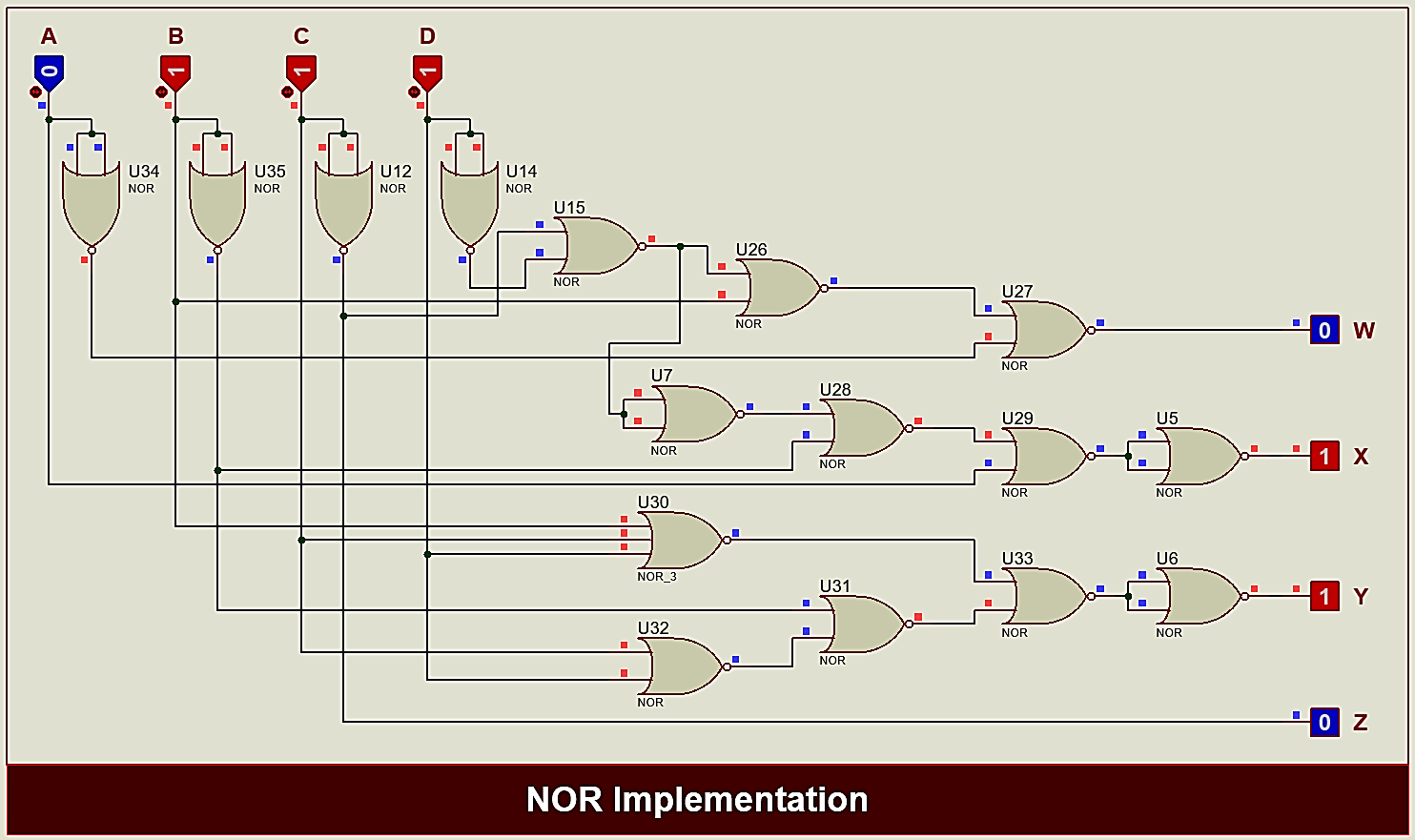
**Pre – Lab Task 3**

**Draw the logic diagram for Excess-3 to gray code converter using only NAND gates in the space provided below, you can use 2, 3, 4 input Nand gates if required.**



**Pre – Lab Task 4**

**Draw the logic diagram for Excess-3 to gray code converter using only NOR gates in the space provided below, you can use 2,3,4 input Nor gates if required.**

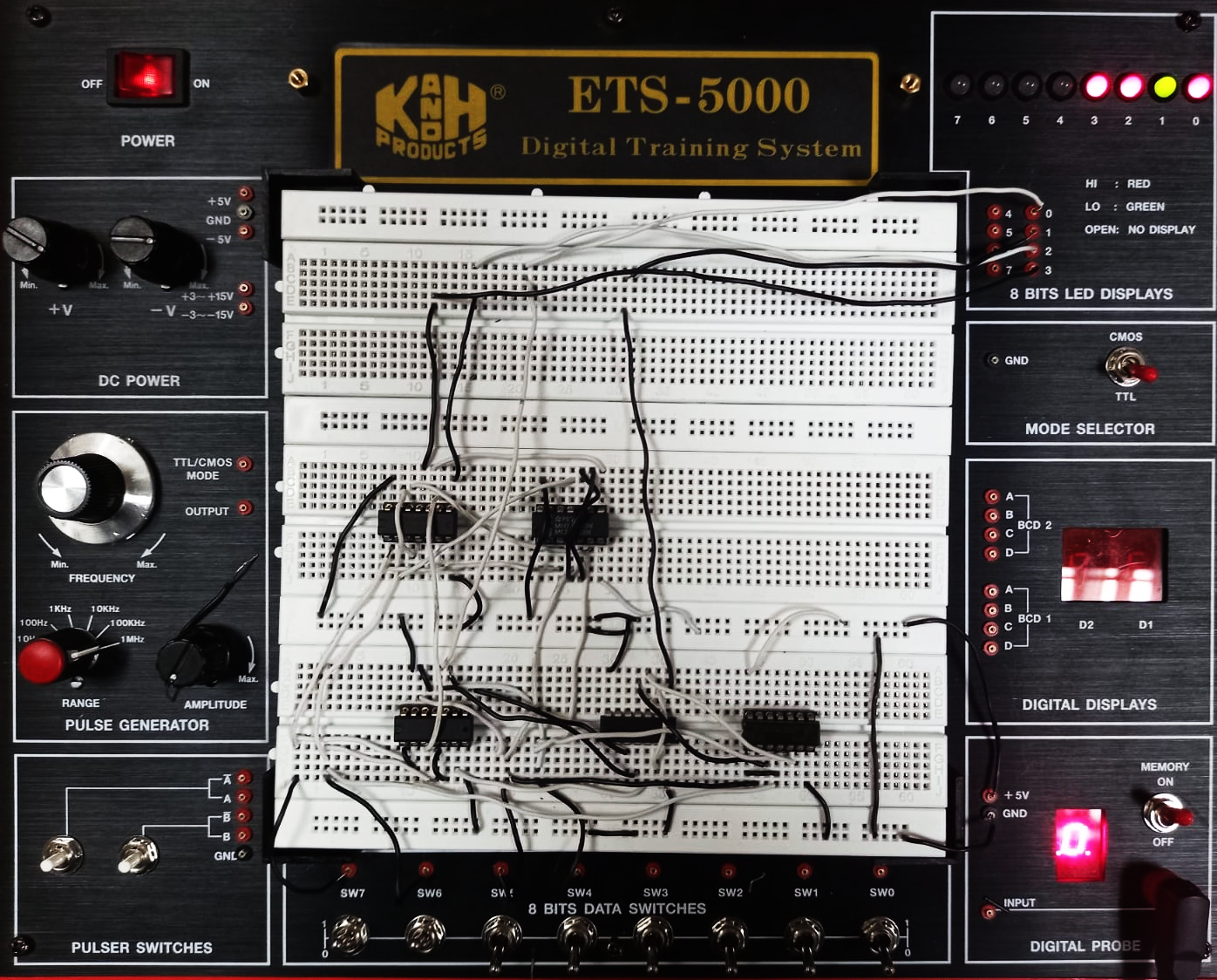


**Lab Tasks**

**Lab Task 1**

**Implement Excess-3 to gray code converter using only NAND gates on hardware. Paste the complete circuit diagram, depicting hardware results.**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dec** | **XS - 3** | | | | **Gray Code (Hardware)** | | | |
| **A** | **B** | **C** | **D** | **W** | **X** | **Y** | **Z** |
| **0** | **0** | **0** | **1** | **1** | **0** | **0** | **0** | **0** |
| **1** | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **1** |
| **2** | **0** | **1** | **0** | **1** | **0** | **0** | **1** | **1** |
| **3** | **0** | **1** | **1** | **0** | **0** | **0** | **1** | **0** |
| **4** | **0** | **1** | **1** | **1** | **0** | **1** | **1** | **0** |
| **5** | **1** | **0** | **0** | **0** | **0** | **1** | **1** | **1** |
| **6** | **1** | **0** | **0** | **1** | **0** | **1** | **0** | **1** |
| **7** | **1** | **0** | **1** | **0** | **0** | **1** | **0** | **0** |
| **8** | **1** | **0** | **1** | **1** | **1** | **1** | **0** | **0** |
| **9** | **1** | **1** | **0** | **0** | **1** | **1** | **0** | **1** |



**Lab Task 2**

**Design and simulate the circuit K – Map equations you obtained in Pre – Lab task 1 in Verilog dataflow modeling. Give the code and testbench and waveform in the space provided below.**

**Verilog Code**

module xs3\_to\_gray(w, x, y, z, a, b, c, d);

    input a, b, c, d;

    output w, x, y, z;

    assign w = a & (b | (c & d));

    assign x = a | (b & (c & d));

    assign y = (~b & ~c & ~d) | (b & (c | d));

    assign z = ~c;

endmodule

module testbench();

    reg A, B, C, D;

    wire W, X, Y, Z;

    xs3\_to\_gray t1(W, X, Y, Z, A, B, C, D);

    initial begin

        #100 A = 0; B = 0; C = 1; D = 1;

        #100 A = 0; B = 1; C = 0; D = 0;

        #100 A = 0; B = 1; C = 0; D = 1;

        #100 A = 0; B = 1; C = 1; D = 0;

        #100 A = 0; B = 1; C = 1; D = 1;

        #100 A = 1; B = 0; C = 0; D = 0;

        #100 A = 1; B = 0; C = 0; D = 1;

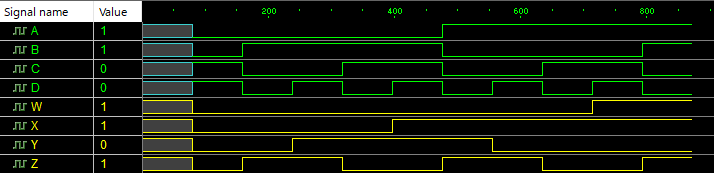
        #100 A = 1; B = 0; C = 1; D = 0;

        #100 A = 1; B = 0; C = 1; D = 1;

        #100 A = 1; B = 1; C = 0; D = 0;

    end

endmodule

****