**GIFT UNIVERSITY GUJRANWALA**



**LAB MANUAL**

**DIGITAL LOGIC DESIGN**

**DEPARTMENT OF Computer Science**

**Submitted By**:

Badar Rasheed Butt

211370113

Section-C

**Submitted To:**

Engr. Umair Ahsan

**Boolean Function Implementation Using**

**Universal Logic Gates**

**EXPERIMENT NO. 06**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Report Marks (04)** | **Lab Performance (08)** | **Viva Marks**  **(03)** | **Total**  **(15)** |
|  |  |  |  |  |

=========================================================================================

**Lab Tasks**

=========================================================================================

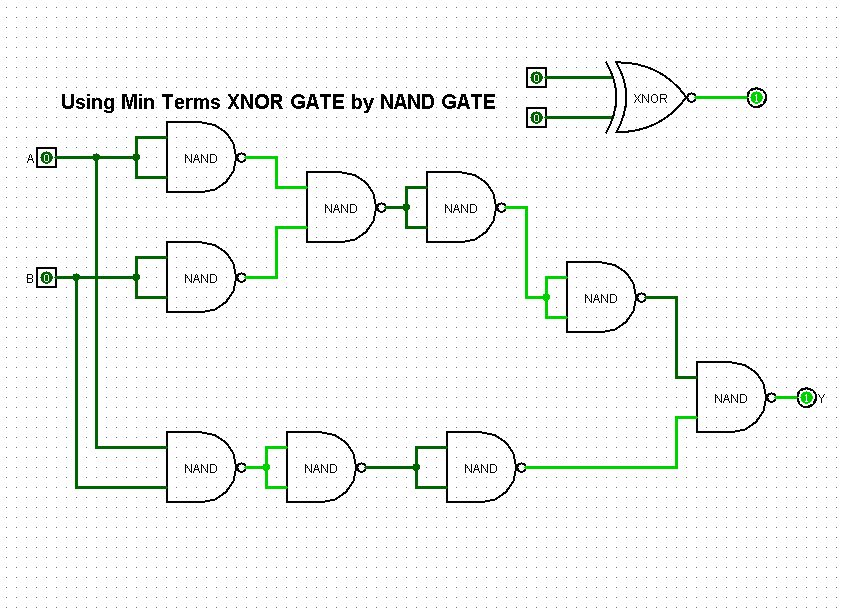
1. Implement XNOR gate using NAND gates on LOGISIM & Write truth table and draw circuit diagram. **(02 Marks)**
2. Implement XNOR gate using NOR gates on LOGISIM & Write truth table and draw circuit diagram. **(02 Marks)**
3. Implement XOR gate using NAND and NOR gates on LOGISIM & Write truth table and draw circuit diagram. **(04 Marks)**

**LAB Task # 01**

**Fill in the truth table for the XNOR function using NAND Gate: (0.5 Marks)**

|  |  |  |
| --- | --- | --- |
| **Input A** | **Input B** | **Output F** |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Write Boolean function for XNOR gate: **F** = **(A’.B’) + (A.B)** **(0.5 Marks)**

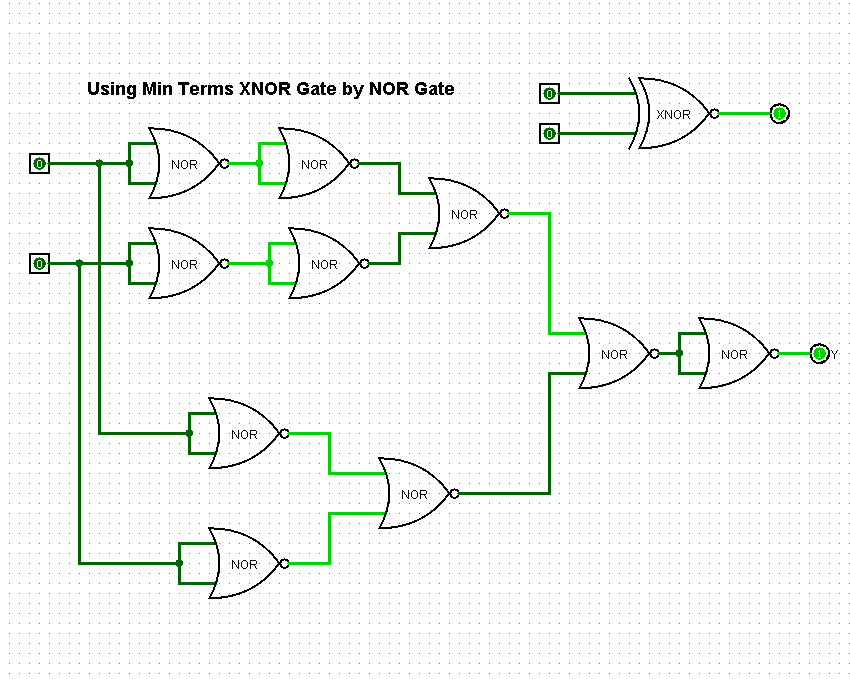
**NAND logic diagram for the XNOR gate (LOGISIM screenshot): (01 Mark)**

**LAB Task # 02**

**Fill in the truth table for the XNOR function using NOR Gate: (0.5 Marks)**

|  |  |  |
| --- | --- | --- |
| **Input A** | **Input B** | **Output F** |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Write Boolean function for XNOR gate: **F** = **(A’.B’) + (A.B)** **(0.5 Marks)**

**NOR logic diagram for the XNOR gate (LOGISIM screenshot): (01 Mark)**

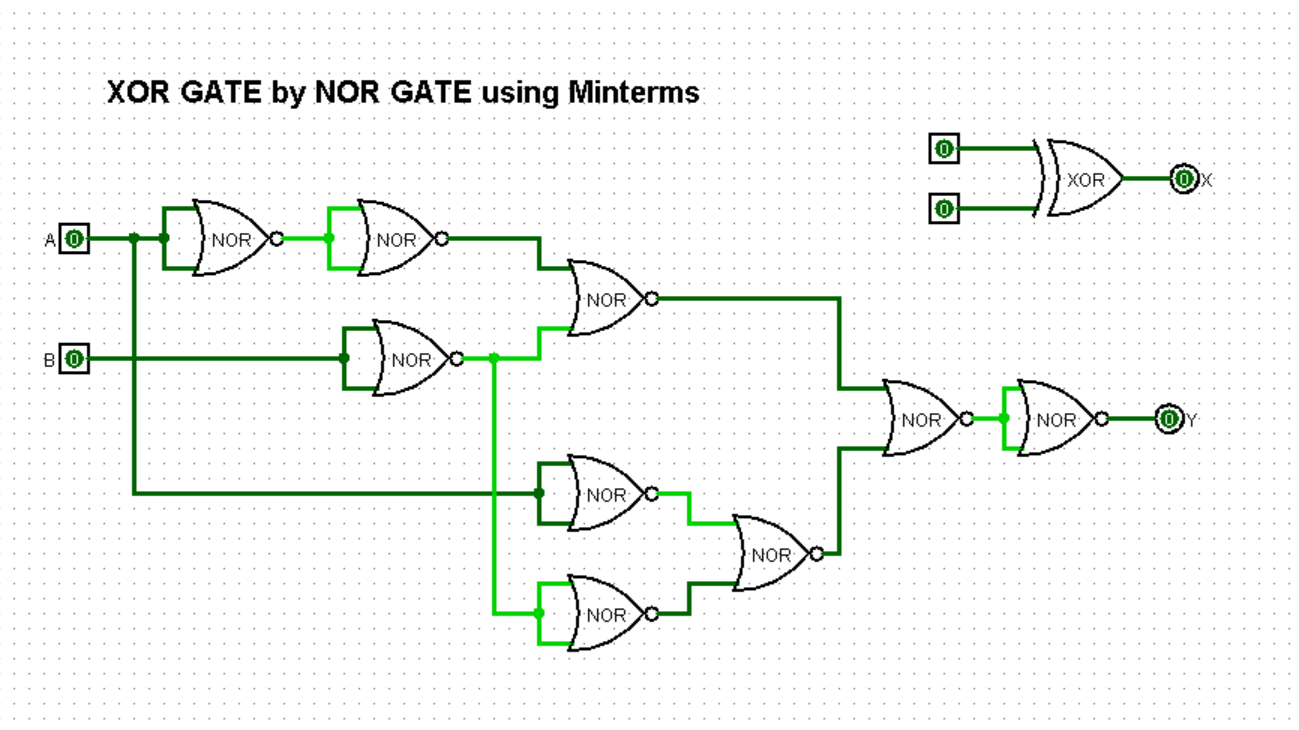
**LAB Task # 03**

**Fill in the truth tables for the XOR function using NOR & NAND Gates: (0.5 Marks)**

**NOR Logic**

|  |  |  |
| --- | --- | --- |
| **Input A** | **Input B** | **Output F** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Write Boolean function for XOR gate NOR Logic : **F** = **(A’.B) + (A.B’)** **(0.5 Marks)**

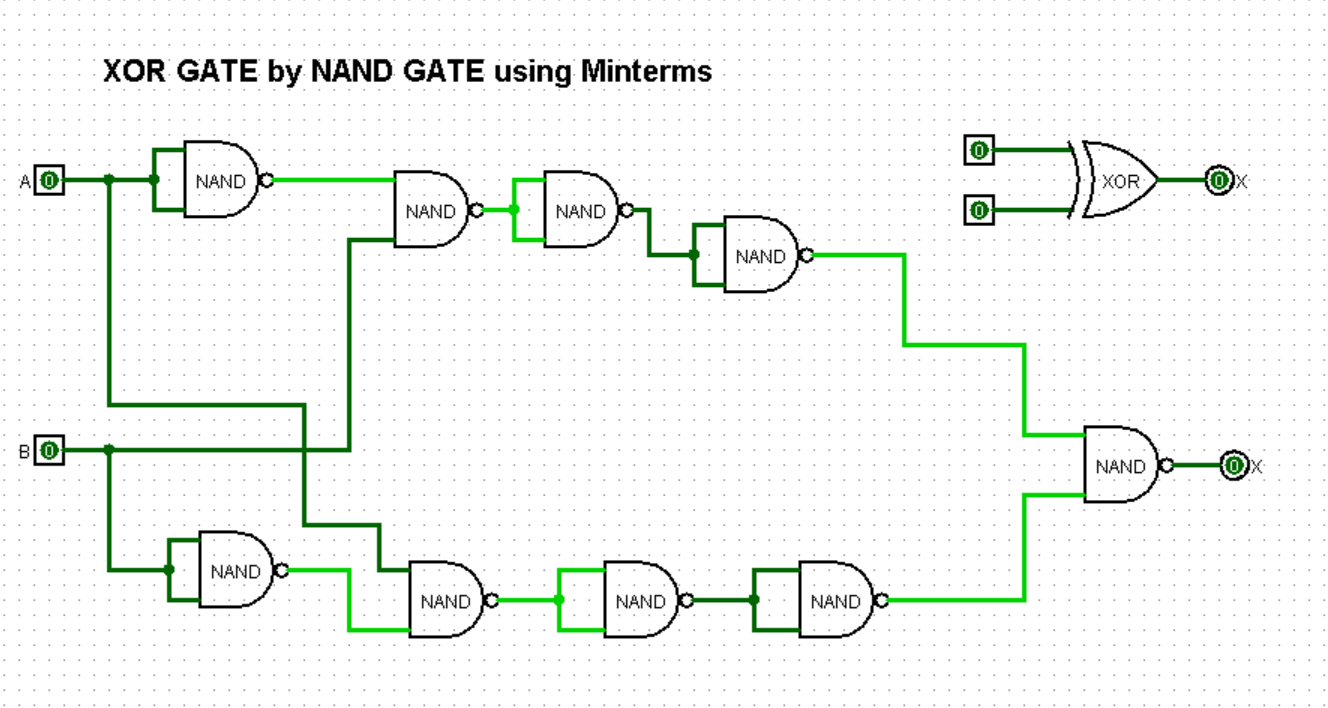
**NOR logic diagram for the XOR gate (LOGISIM screenshot): (01 Mark)**

**(0.5 Marks)**

**NAND Logic**

|  |  |  |
| --- | --- | --- |
| **Input A** | **Input B** | **Output F** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Write Boolean function for XOR gate NAND Logic: **F** = **(A’.B) + (A.B’)** **(0.5 Marks)**

**NAND logic diagram for the XOR gate: (01 Mark)**

**Observations: (01 Mark)**

In this lab we learn how to use **Universal gates** and derive there expressions, we learn how to design a gate using other gates. In this lab we design circuits using only **NAND** and **NOR** gates,

**For example**

We can design XOR using NAND

We can design XOR using NOR

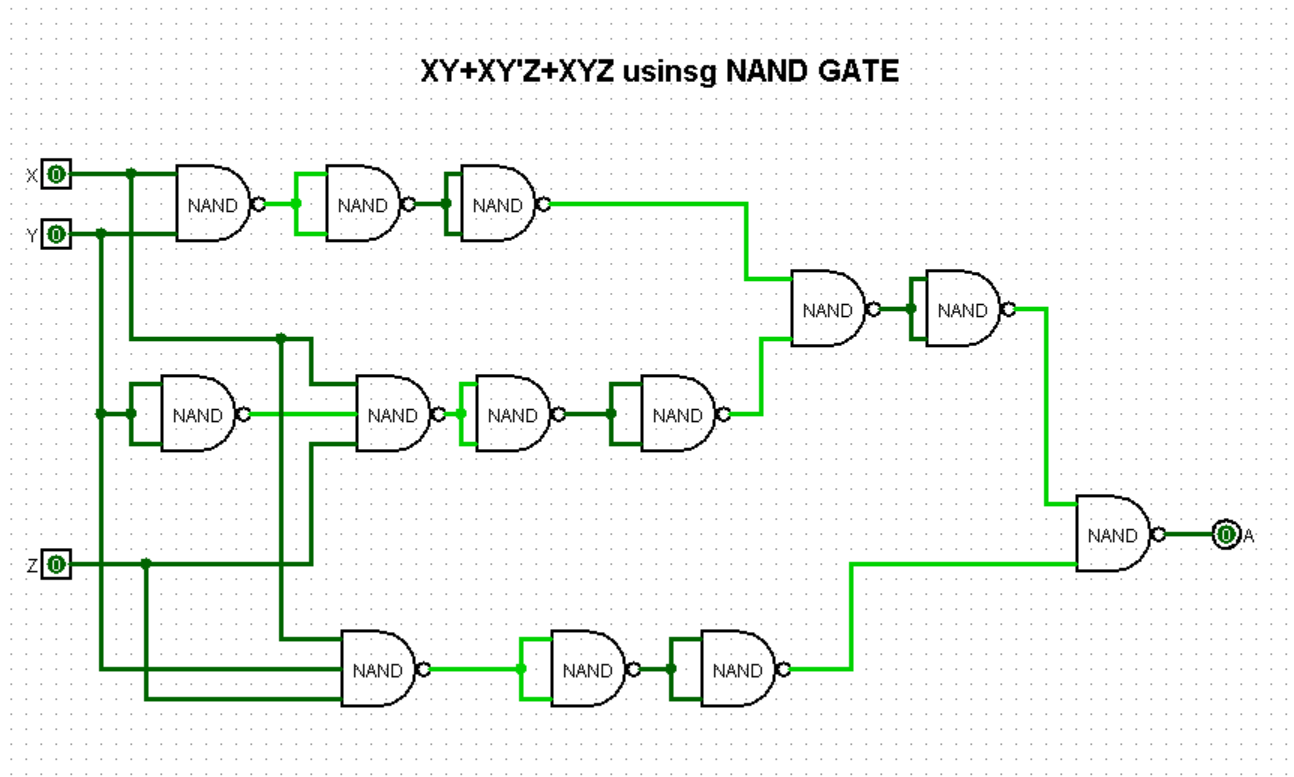
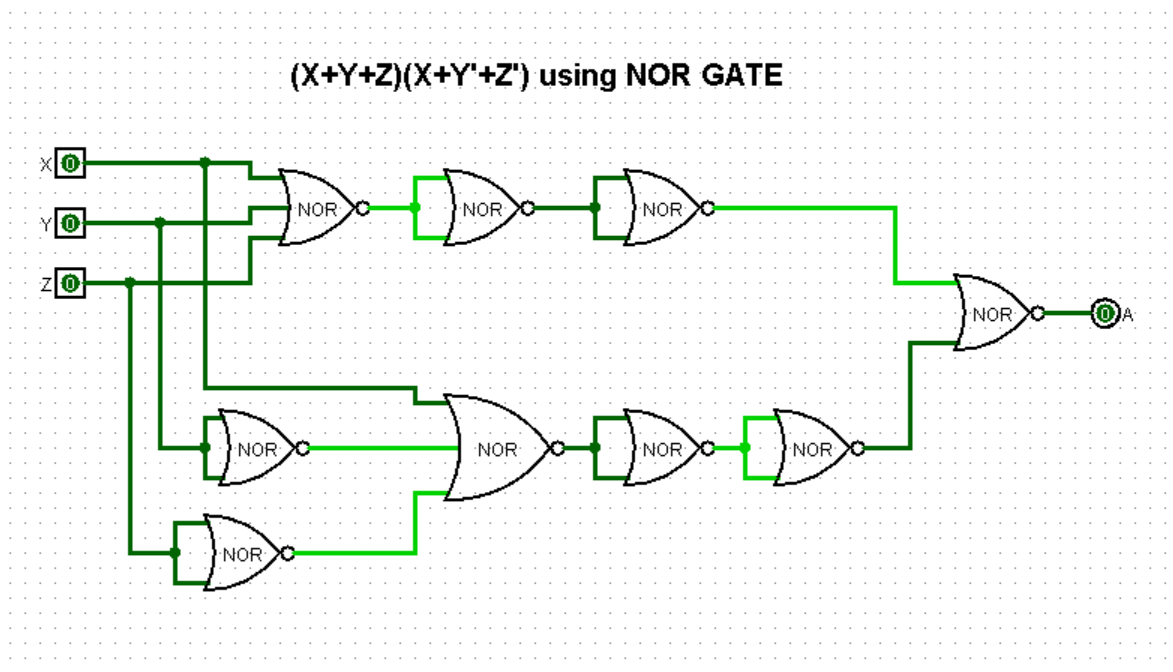
We can design XNOR using NOR

We can design XNOR using NAND

**Review Questions**

1. **Why are the NAND and NOR gates more popular? (01 Mark)**

These gates are popular because they are less expensive and easy to design, by using **NAND** and **NOR** gates we can design any other gate, that is why these gates are called **Universal gates**.

1. **Draw logical circuit for following using NAND only XY+XY’Z+XYZ (01 Mark)**
2. **Draw the logical function using NOR only (X+Y+Z).(X+Y’+Z’) (01 Mark)**