ASSIGNMENT 1:

Title: Introduction to digital Fundamental Circuit Design using Proteus

PRN: 22510034 Date:09-09-23

Aim:

a. Implement basic gates - OR, NAND, NOR, Ex-OR, Ex-NOR. Verify truth table for each gate.

b. Demonstrate universal gates – NAND and NOR. Verify the truth table.

Hardware Used: None

Software used: Proteus 8 Professional

Procedure:

Basically, I first understood the workings of all the logic gates then according to their IC numbers searched them in Proteus software and built the circuits accordingly and then assuming some inputs found out the outputs by simulating it in the software itself.

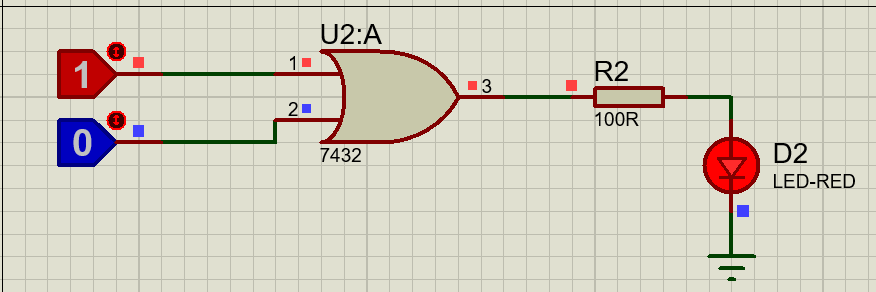
Results:

a)

0)AND GATE

TRUTH TABLE:

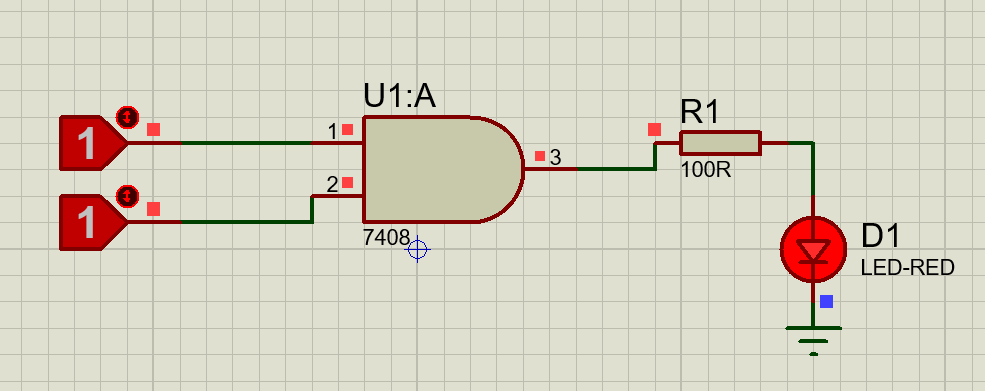
|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



1. OR GATE

TRUTH TABLE:

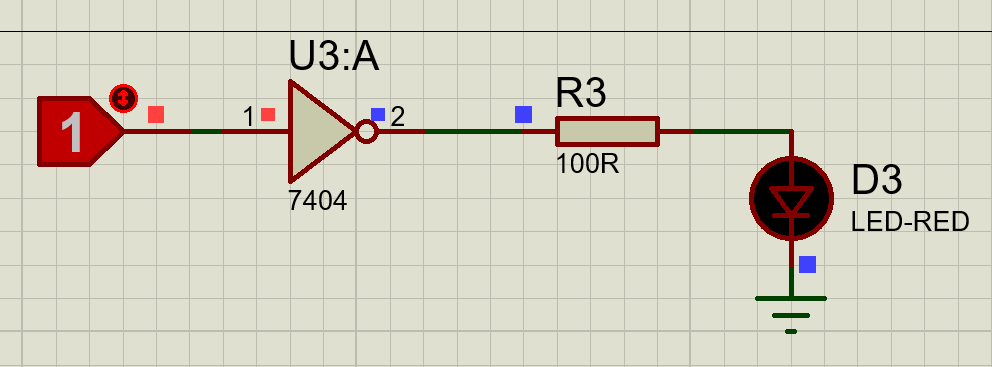
|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



1. NOT GATE

TRUTH TABLE:

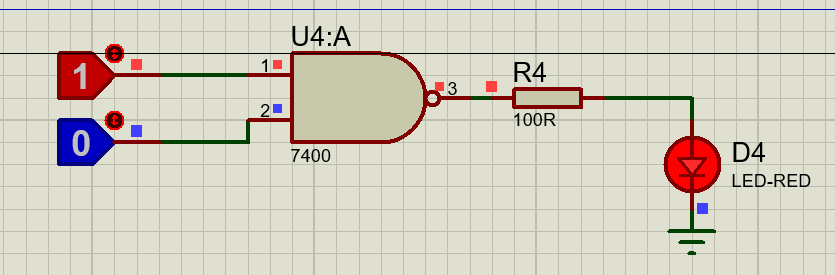
|  |  |
| --- | --- |
| A | Y |
| 1 | 0 |
| 0 | 1 |



1. NAND GATE

TRUTH TABLE:

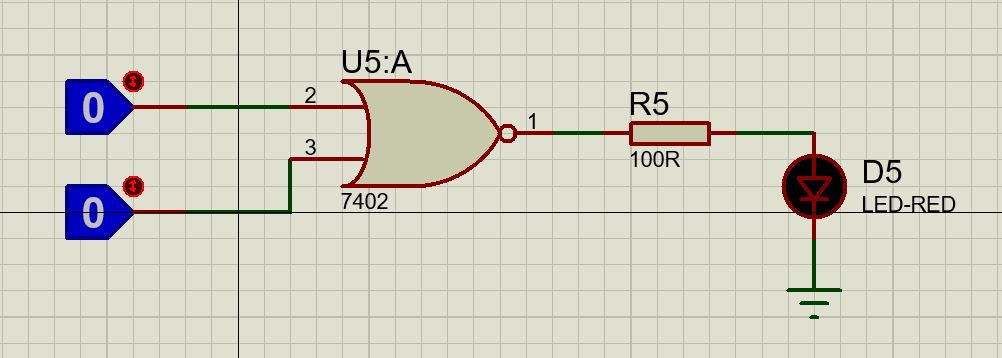
|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |



1. NOR GATE

TRUTH TABLE:

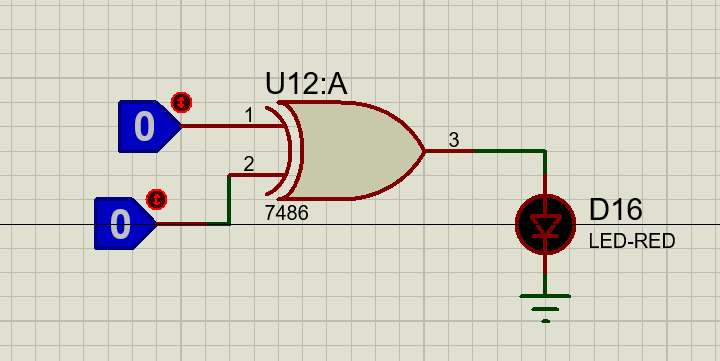
|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |



1. Ex-OR GATE

TRUTH TABLE:

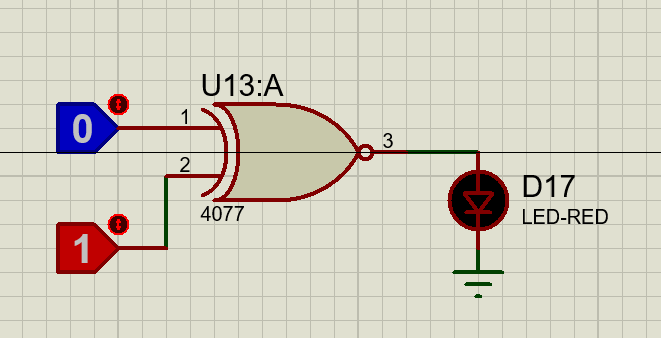
|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

.

1. Ex-NOR GATE

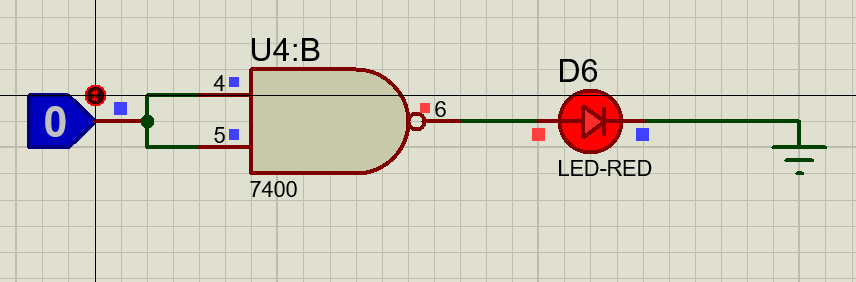
TRUTH TABLE

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



NOT USING NAND

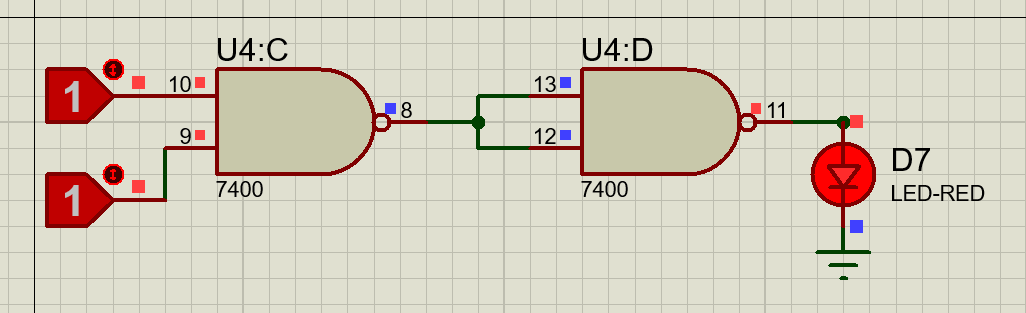
|  |  |
| --- | --- |
| A | Y |
| 1 | 0 |
| 0 | 1 |



AND USING NAND

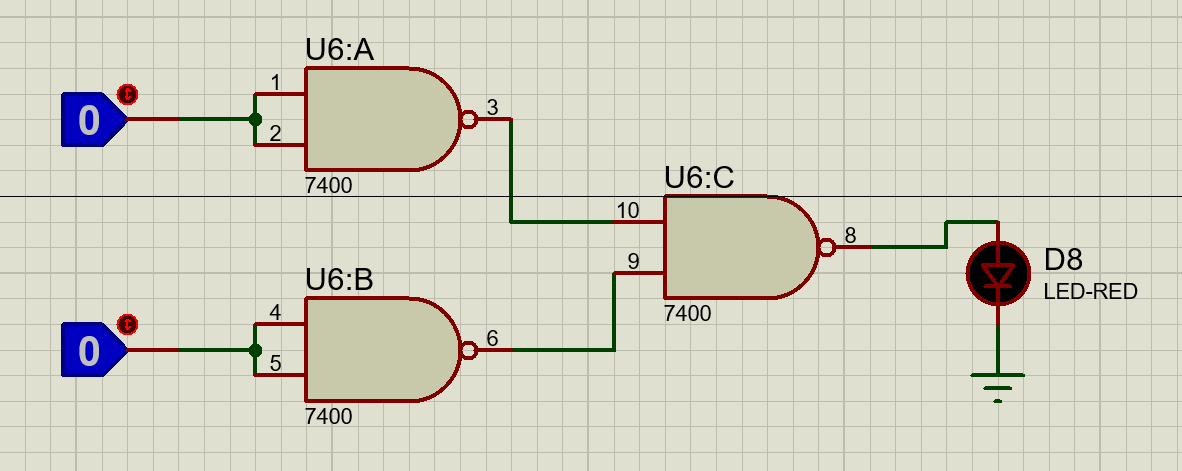
TRUTH TABLE :

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



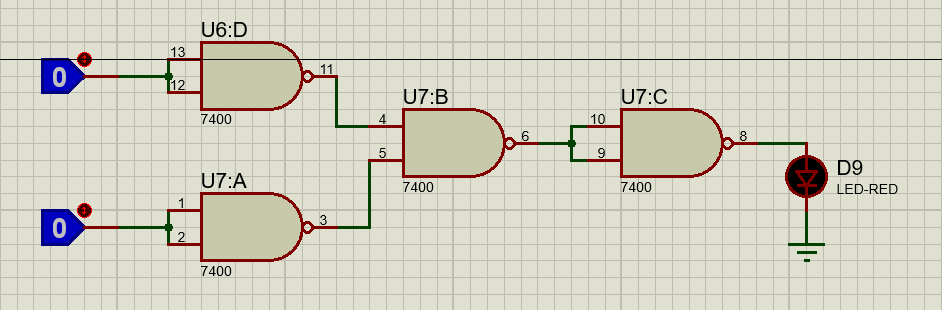
OR USING NAND

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



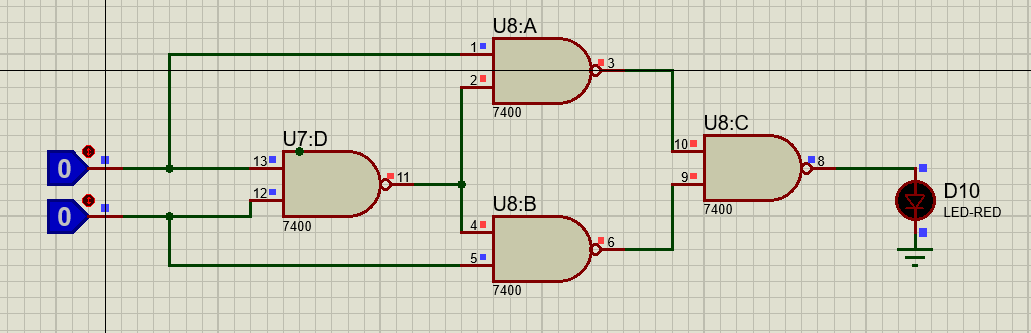
NOR USING NAND

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |



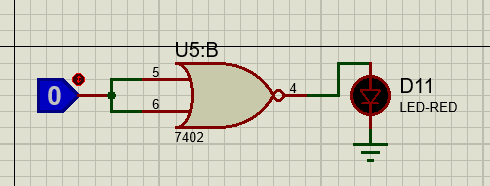
Ex-OR USING NAND

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |



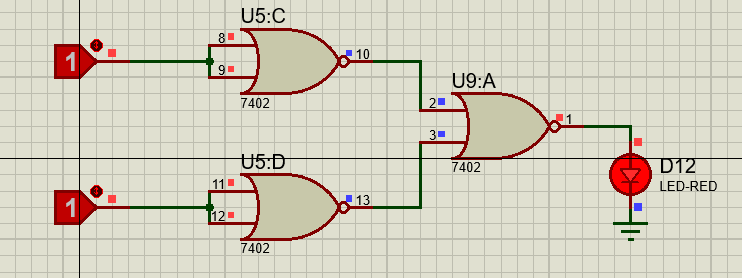
NOT USING NOR

|  |  |
| --- | --- |
| A | Y |
| 1 | 0 |
| 0 | 1 |



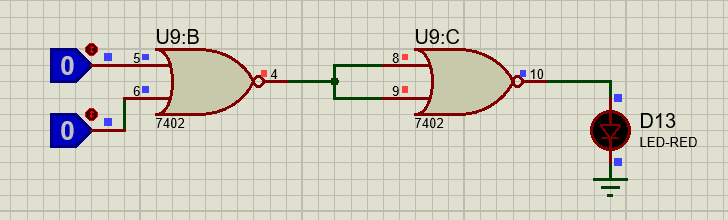
AND USING NOR

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



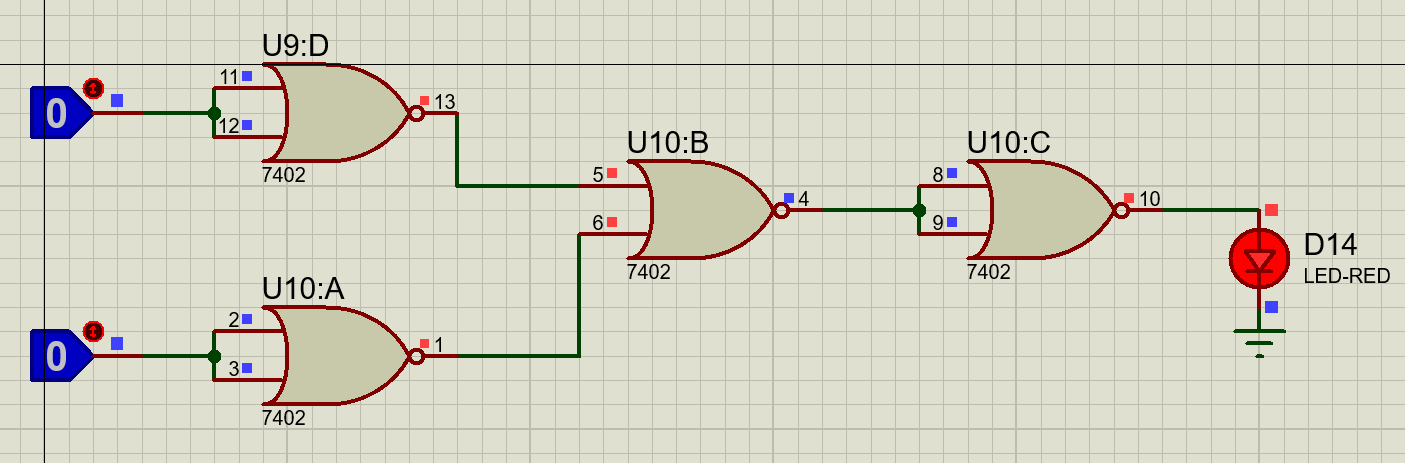
OR USING NOR

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

.

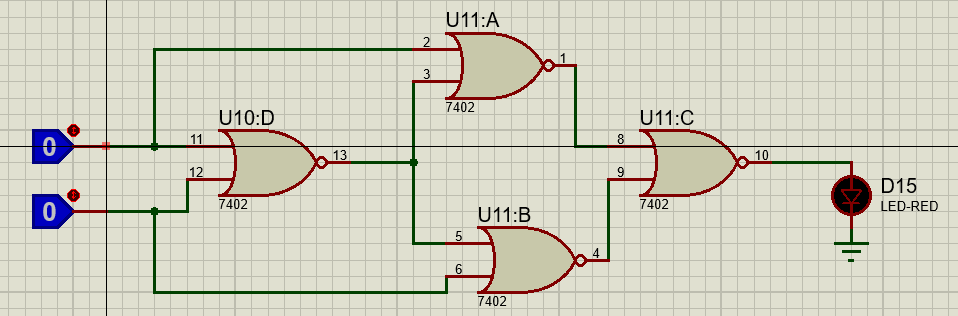
NAND USING NOR

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |



Ex-NOR USING NOR

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



Conclusion:

By using the proteus software the functionality of all the logic gates can be verified as per the theoretical truth tables and logic equations successfully.