**Assignment 1**

**Name – Suraj Kumar Yadav**

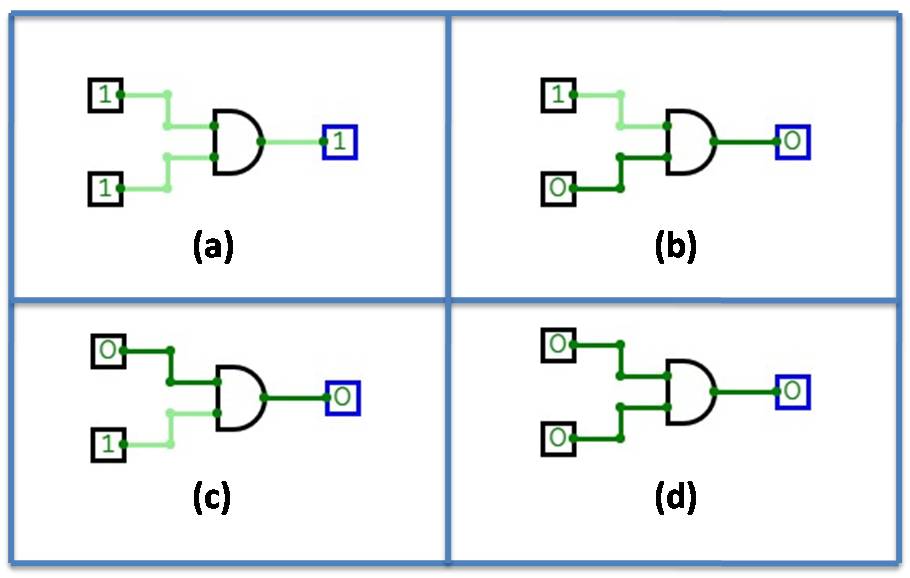
**Exam Roll No. - 20220PHY014**

**Draw the logic diagram on CircuitVerse for the following:**

1. Verify the truth table of AND, OR and NOT gates on CircuitVerse.
   1. **Truth table of AND gate**

**Boolean Expression 🡪 A . B = X**

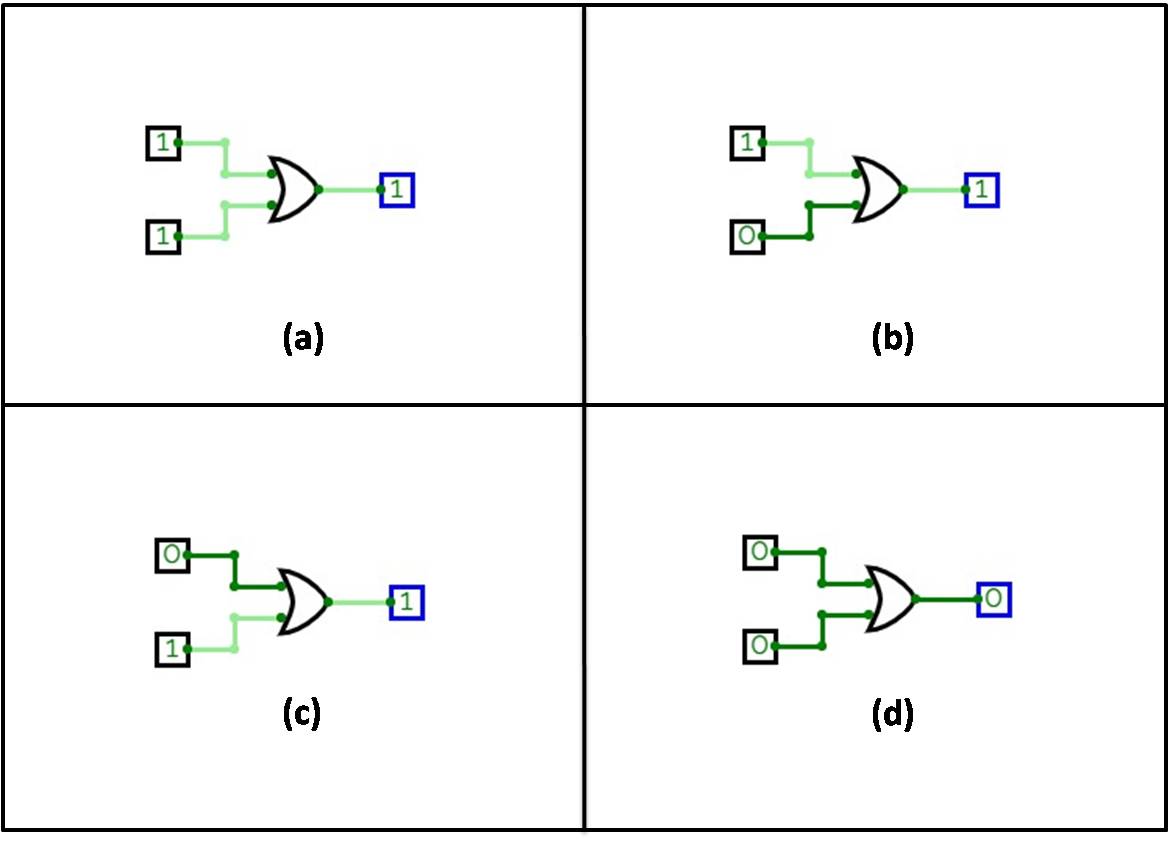
|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X(output) |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **0** |
| (c) | **0** | **1** | **0** |
| (d) | **0** | **0** | **0** |



* 1. **Truth table of OR gate**

**Boolean Expression** 🡪 **A + B = X**

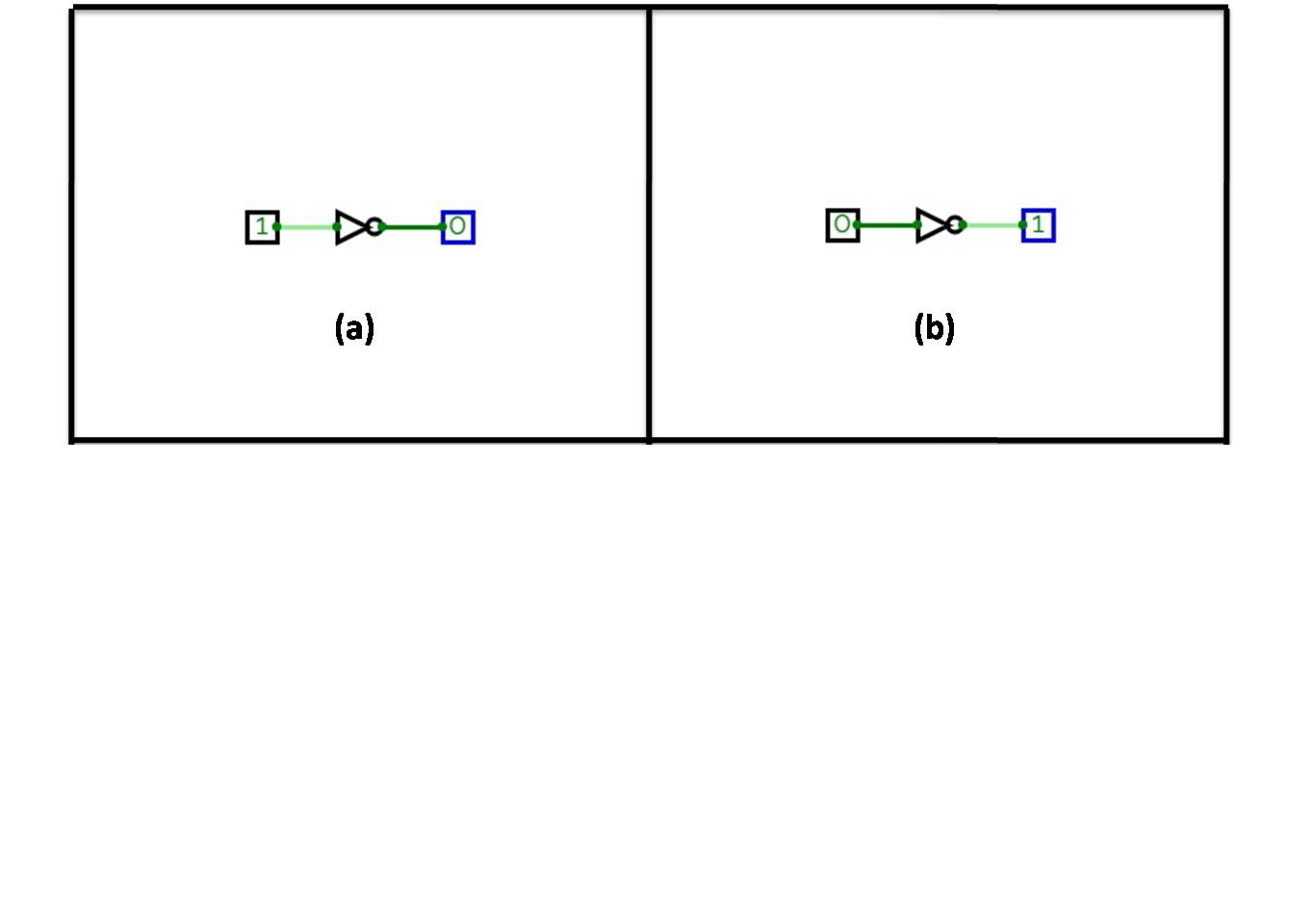
|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X(output) |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **1** |
| (c) | **0** | **1** | **1** |
| (d) | **0** | **0** | **0** |



* 1. **Truth table of NOT gate**

**Boolean Expression** 🡪 **Ā = X**

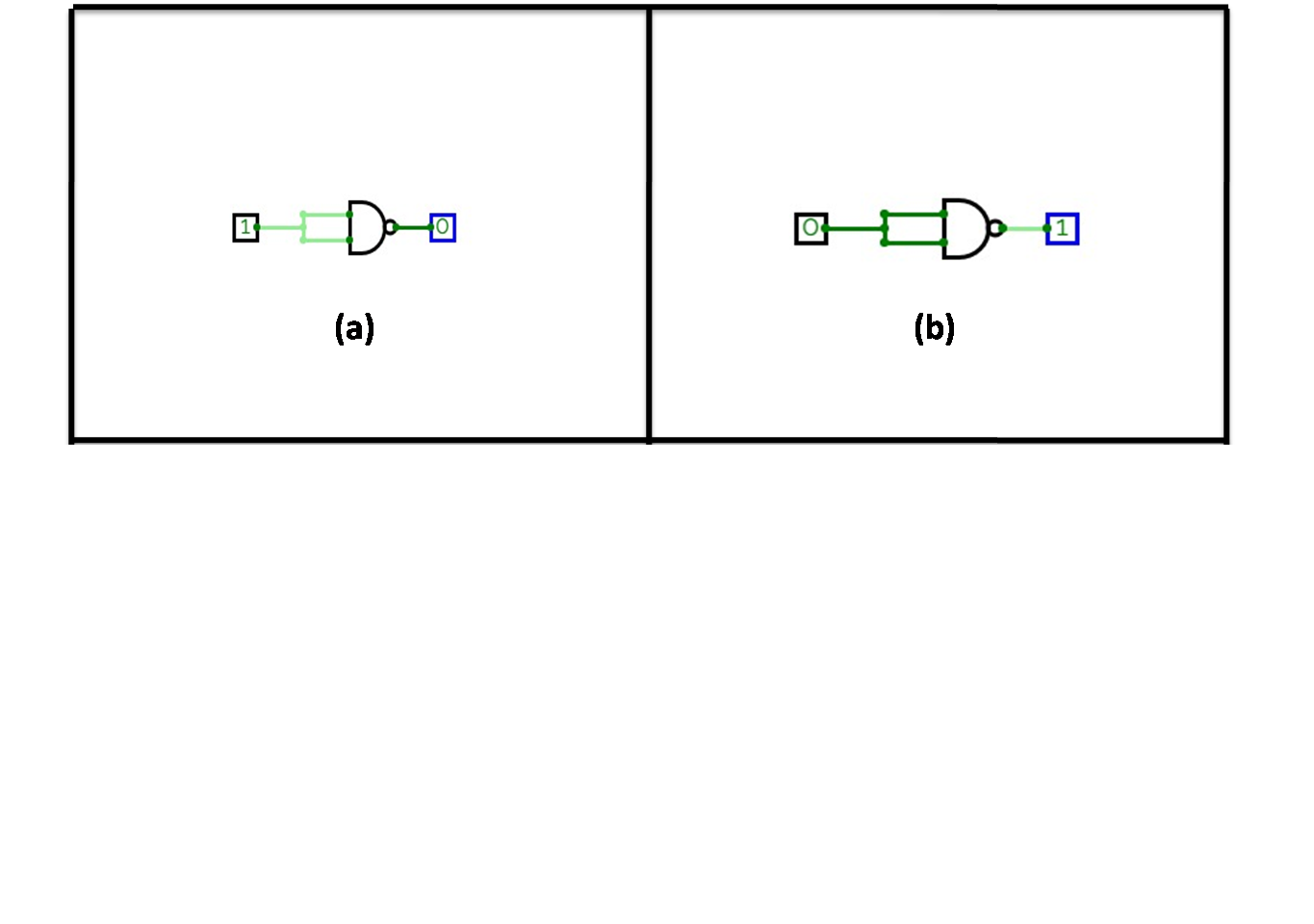
|  |  |  |
| --- | --- | --- |
|  | A | X(output) |
| (a) | **1** | **0** |
| (b) | **0** | **1** |



1. **Draw the logic diagram to implement basic logic gates (AND, OR, NOT) using NAND only.**
   1. **NAND to NOT**

Boolean expression for NAND gate is

To convert NAND gate to NOT gate let A = B. Then, which gives the NOT gate as shown in figure



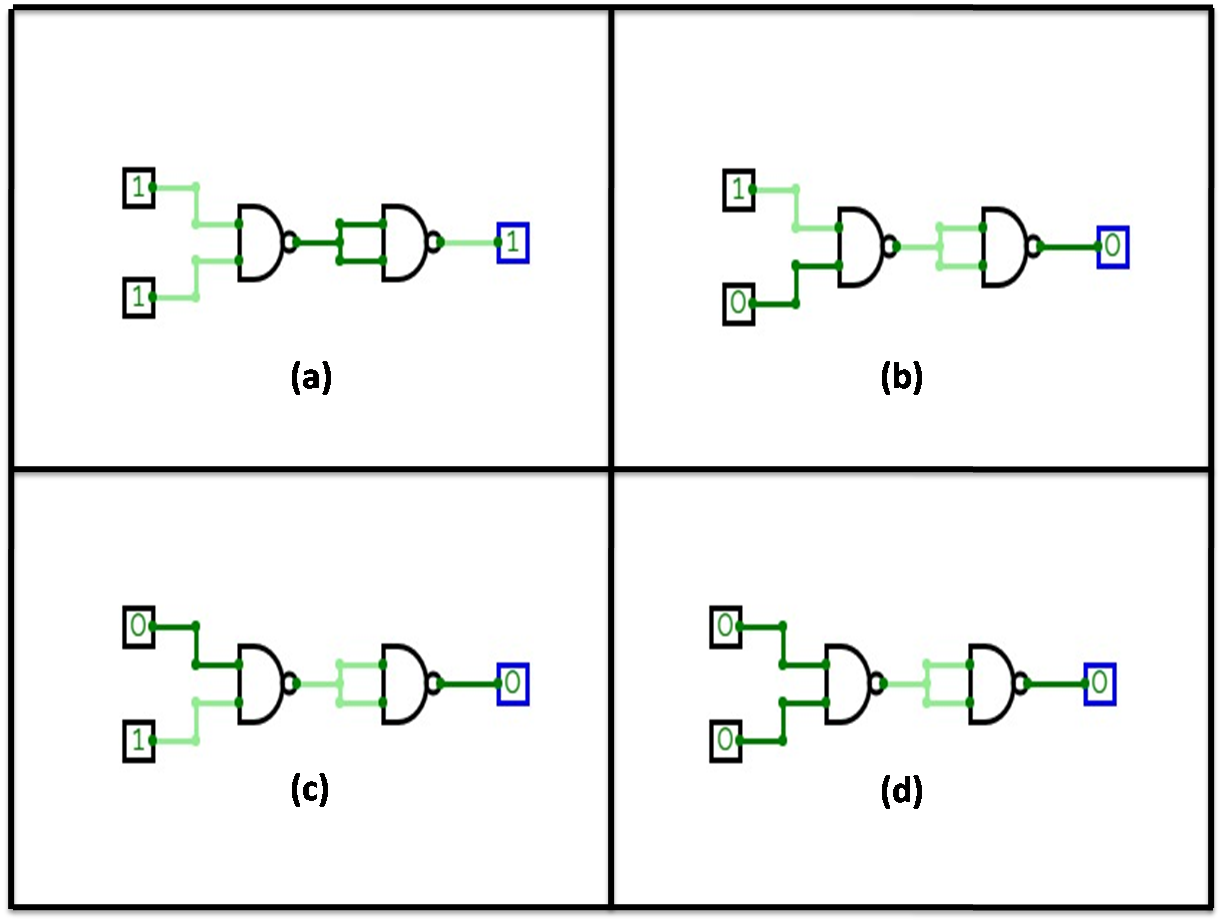
**Truth Table:-**

|  |  |  |
| --- | --- | --- |
|  | A | X (for NOT gate) |
| (a) | **1** | **0** |
| (b) | **0** | **1** |

* 1. **NAND to AND**

Boolean expression for NAND gate is

To convert NAND gate to NOT gate, we have to connect NAND gate as NAND + NOT (from NAND) as shown in figure



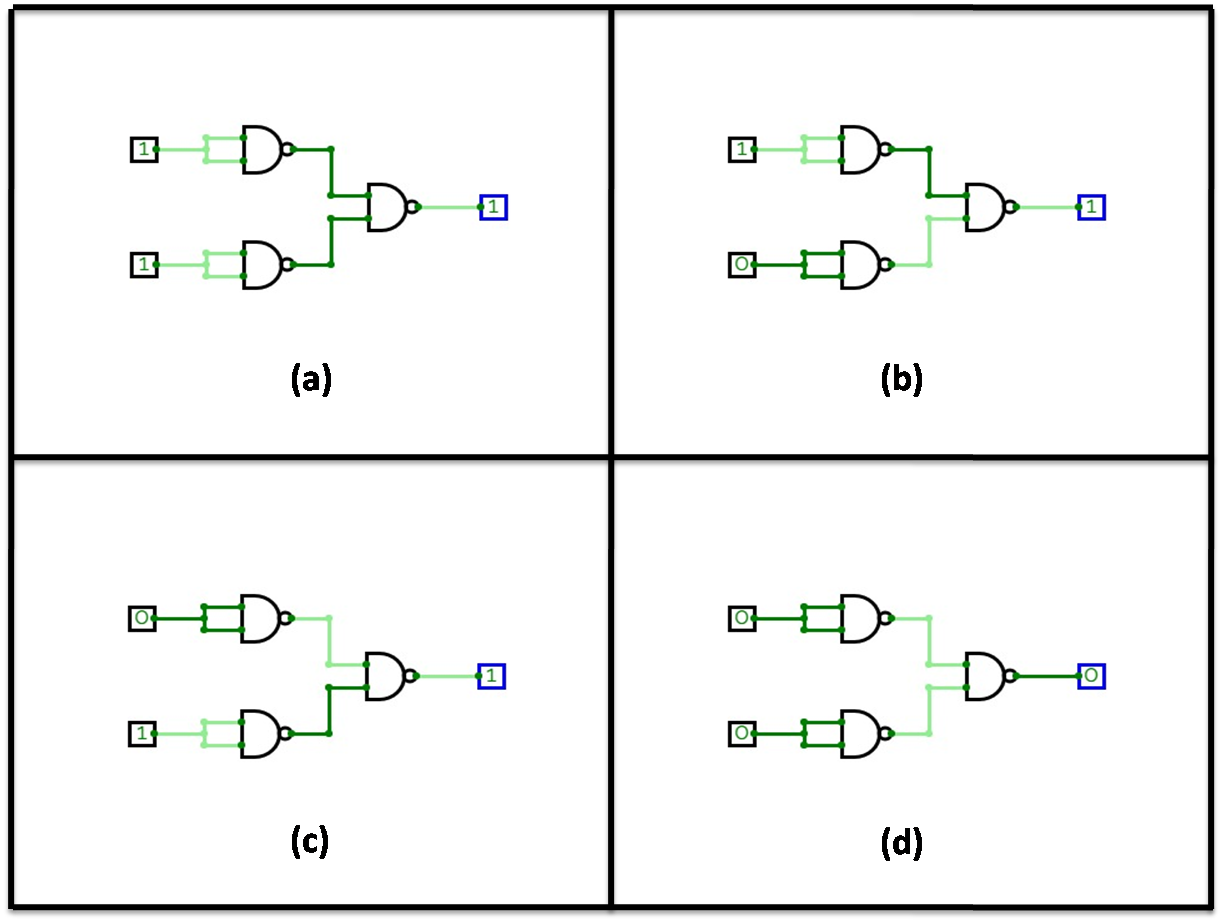
**Truth Table:-**

|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X (for AND gate) |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **0** |
| (c) | **0** | **1** | **0** |
| (d) | **0** | **0** | **0** |

* 1. **NAND to OR**

Boolean expression for NAND gate is

To convert NAND gate to OR gate, we have to connect the two inputs of NAND gate to two different NOT gate (from NAND) as shown in figure.



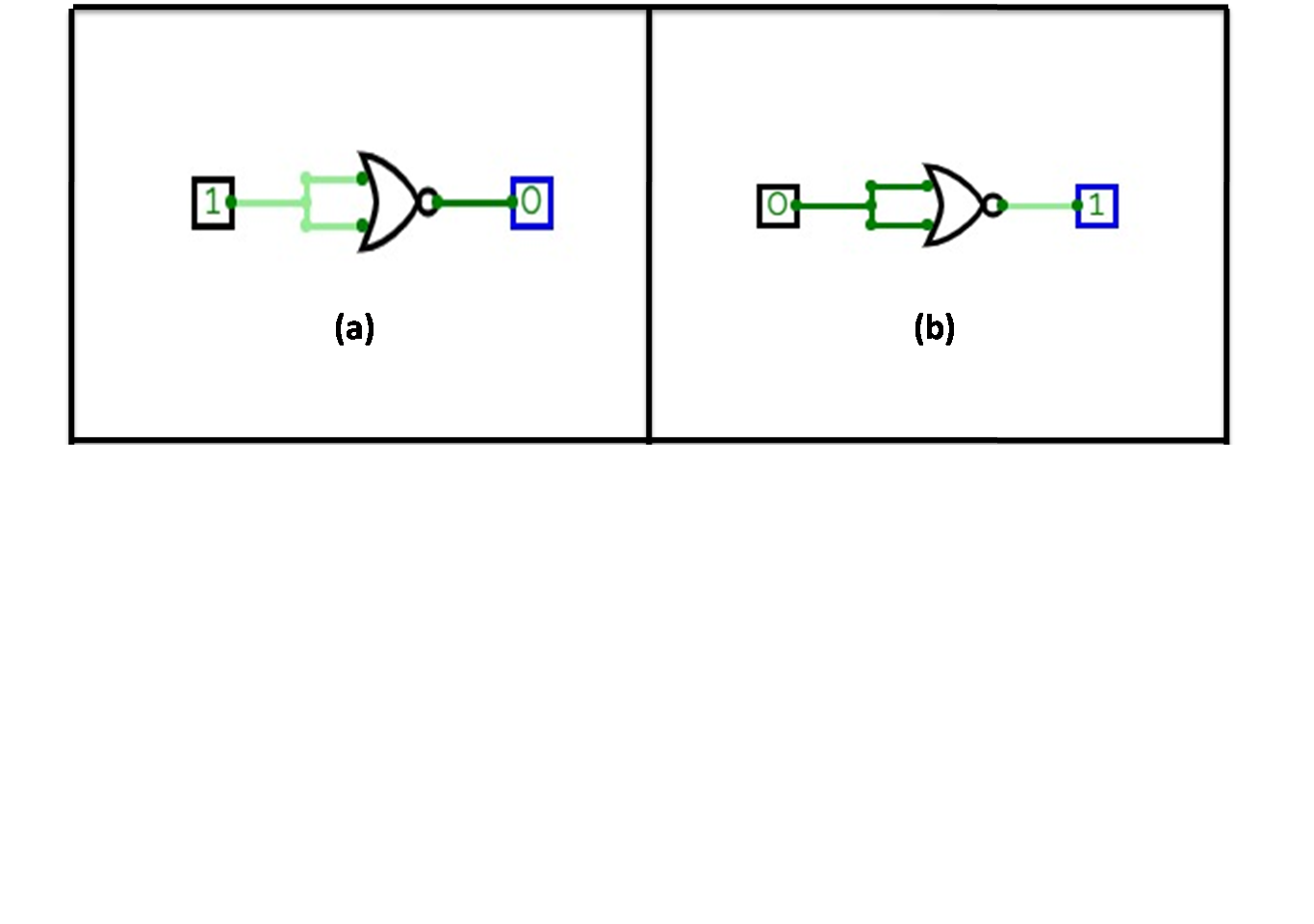
**Truth Table:-**

|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X (for OR gate) |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **1** |
| (c) | **0** | **1** | **1** |
| (d) | **0** | **0** | **0** |

1. **Implement basic logic gates (AND, OR, NOT) using NOR only.**
   1. **NOR to NOT**

Boolean expression for **NOR** gate is

To convert NAND gate to NOT gate let A = B. Then, which gives the NOT gate as shown in figure

****

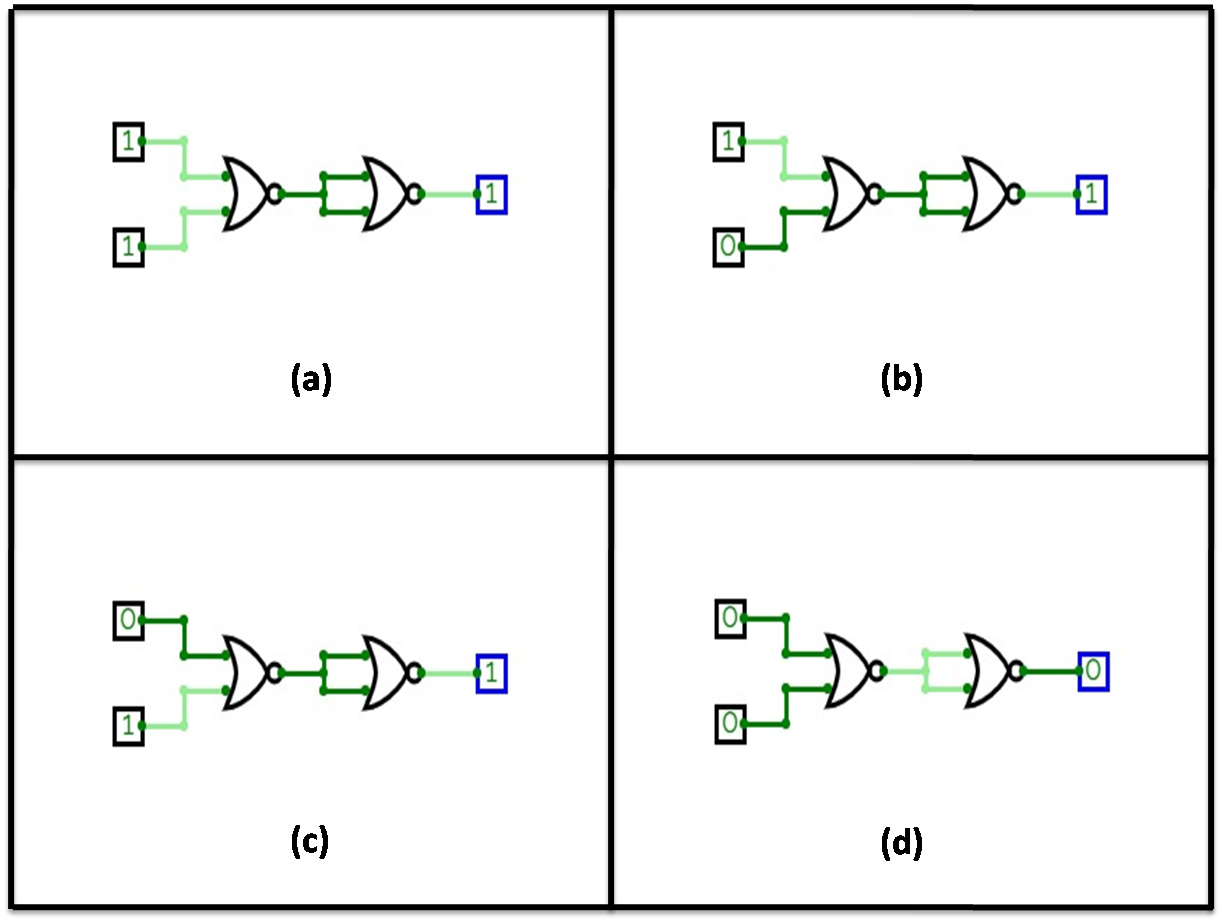
**Truth Table:-**

|  |  |  |
| --- | --- | --- |
|  | A | X (for NOT gate) |
| (a) | **1** | **0** |
| (b) | **0** | **1** |

* 1. **NOR to OR**

Boolean expression for NOR gate is

To convert NOR gate to OR gate, we have to connect NOR gate as NOR + NOT (from NOR) as shown in figure

****

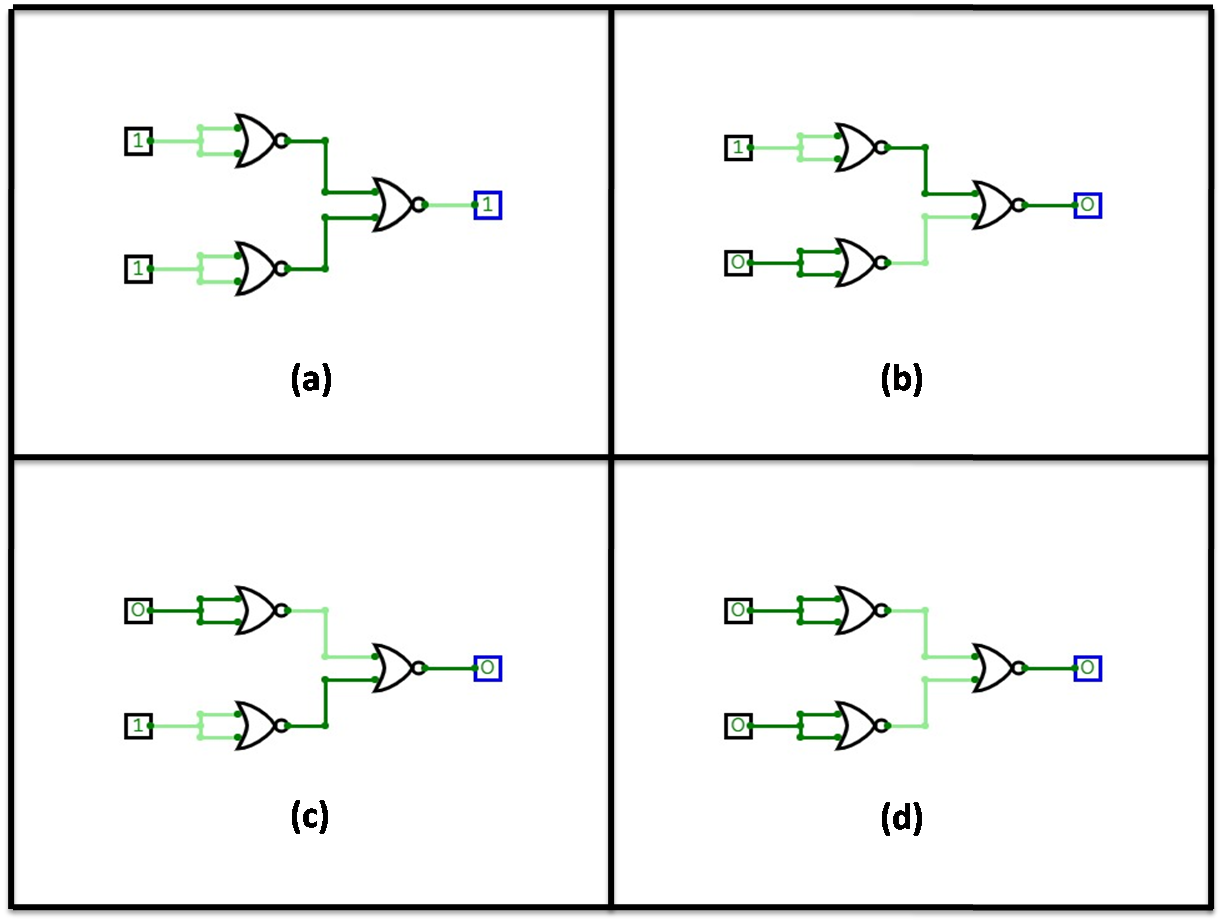
**Truth Table:-**

|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X (for OR gate) |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **1** |
| (c) | **0** | **1** | **1** |
| (d) | **0** | **0** | **0** |

* 1. **NOR to AND**

Boolean expression for NOR gate is

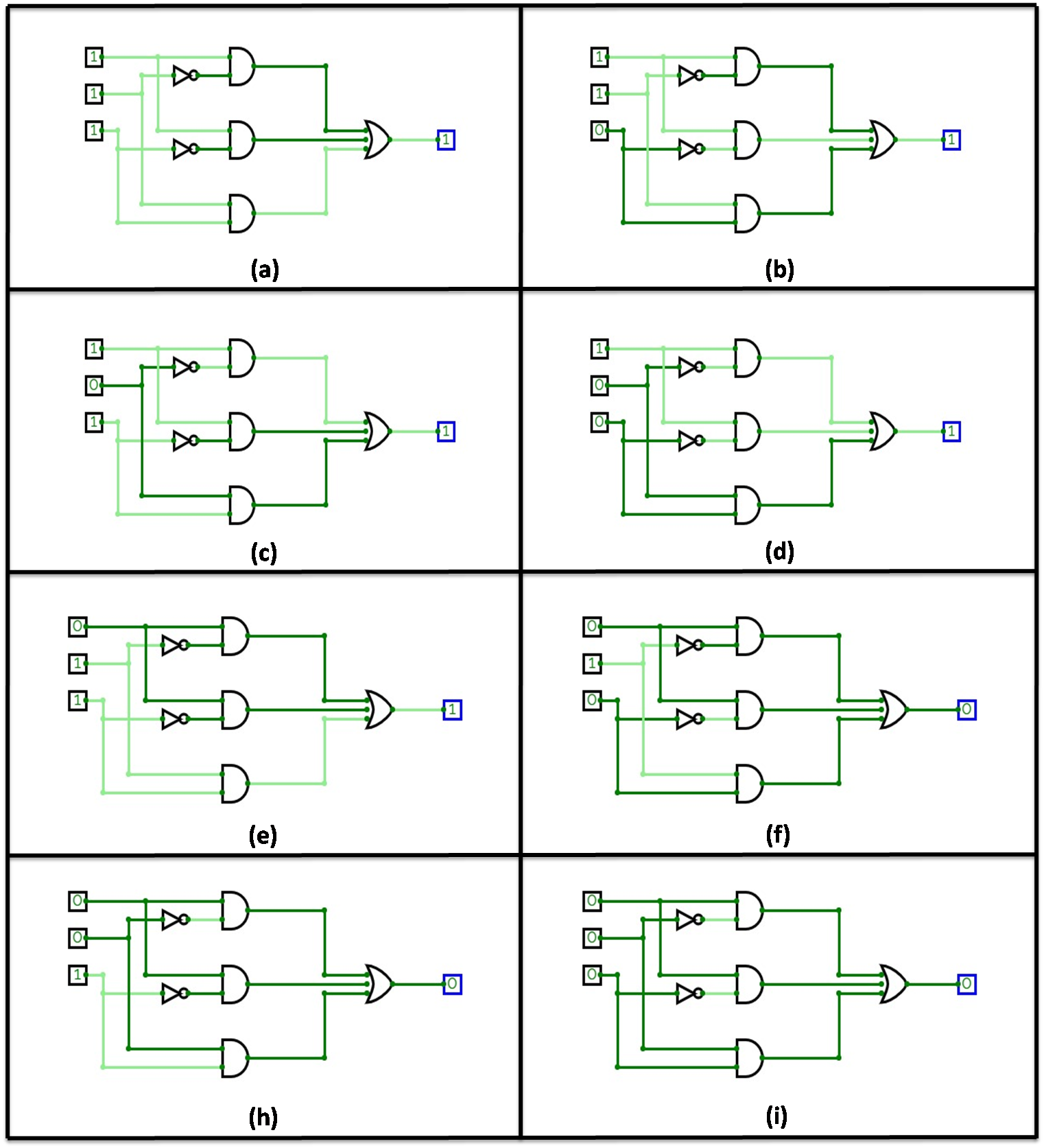
To convert NOR gate to AND gate, we have to connect the two inputs of NOR gate to two different NOT gate (from NOR) as shown in figure.



**Truth Table:-**

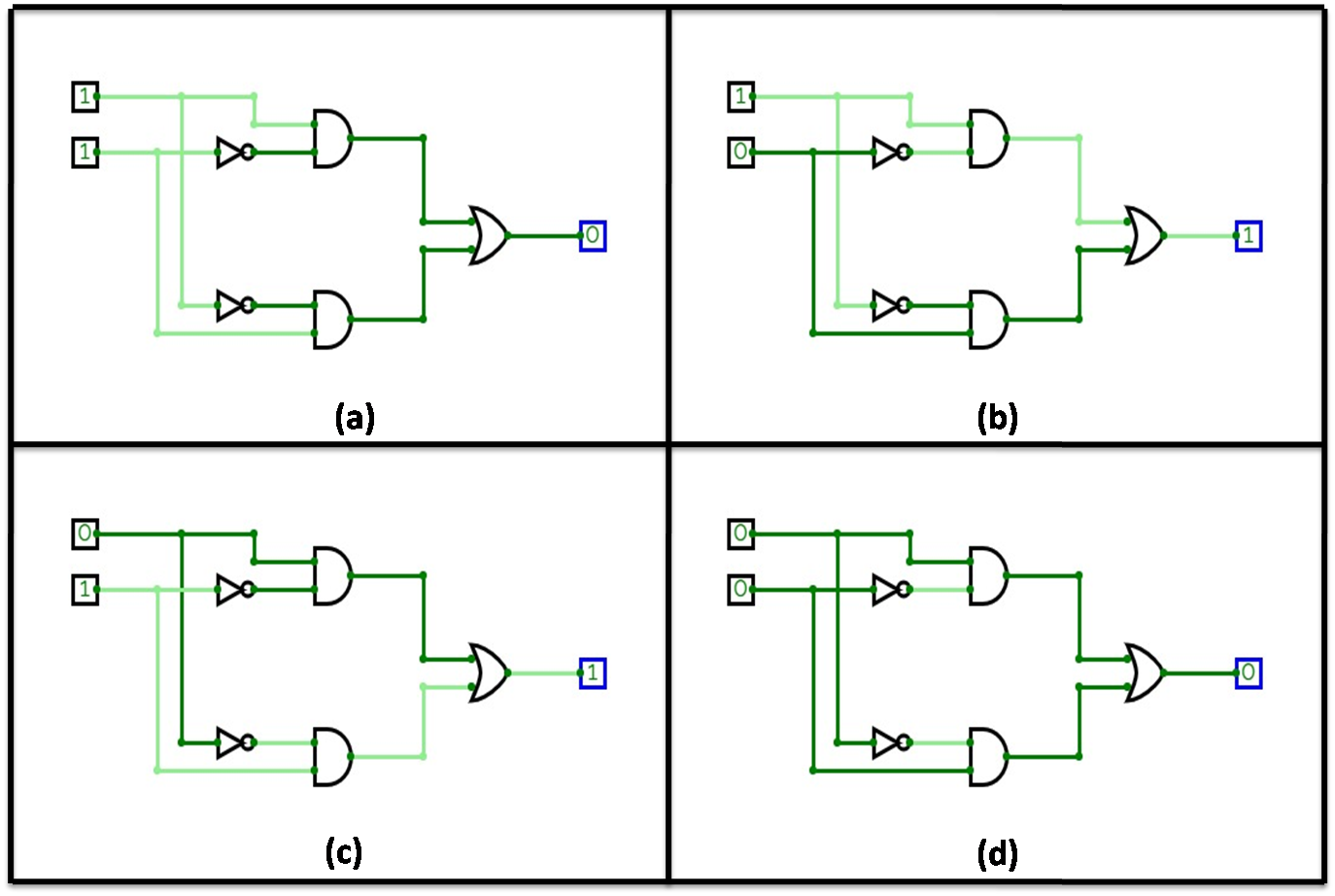
|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X (for AND gate) |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **0** |
| (c) | **0** | **1** | **0** |
| (d) | **0** | **0** | **0** |

1. **Draw the logic diagram for the function**

****

1. **Draw logic diagrams for XOR and XNOR using basic gates**
   1. **XOR**

Boolean expression for XOR gate is

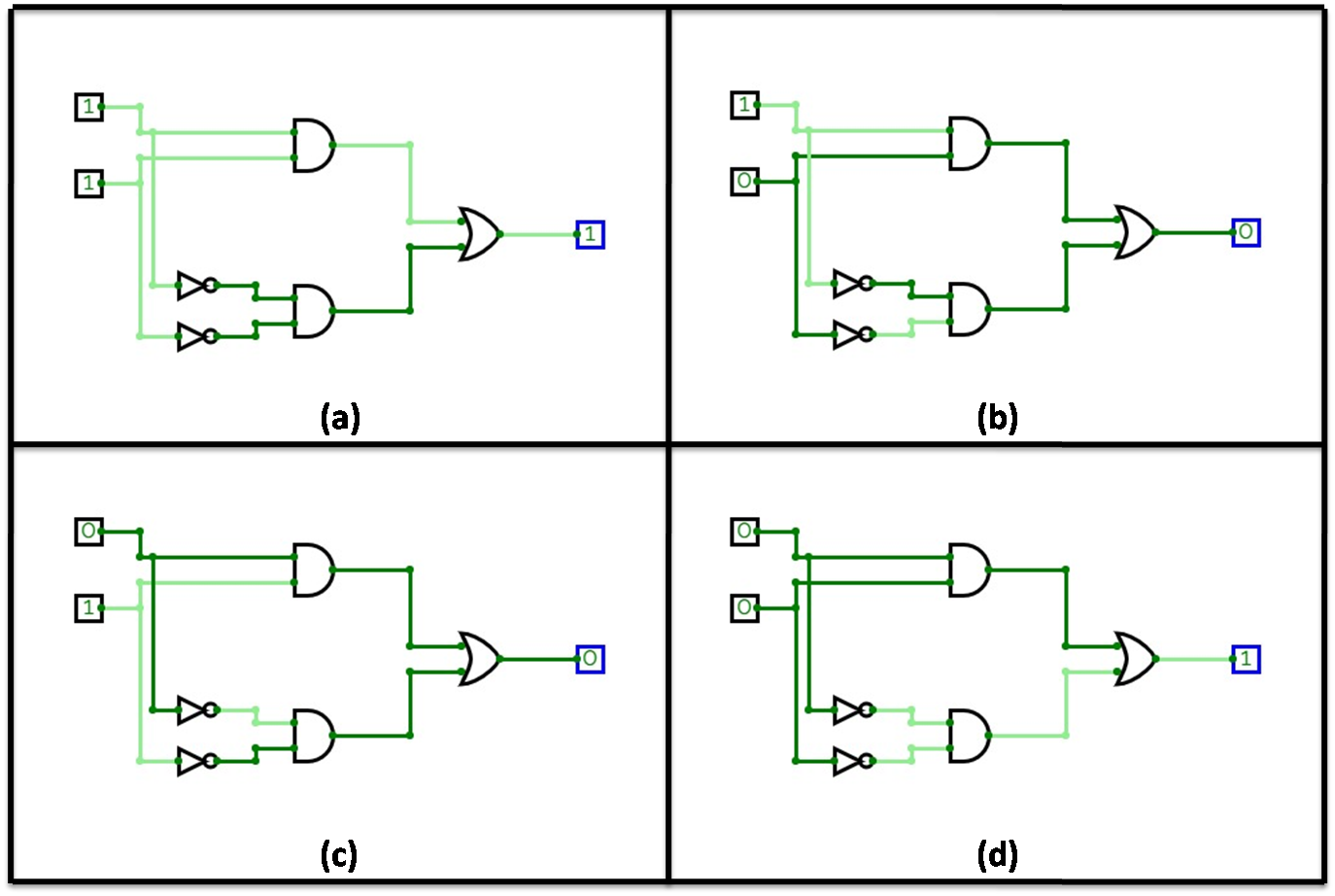
****

**Truth Table:-**

|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X |
| (a) | **1** | **1** | **0** |
| (b) | **1** | **0** | **1** |
| (c) | **0** | **1** | **1** |
| (d) | **0** | **0** | **0** |

* 1. **XNOR gate**

Boolean Expression for XNOR gate is



**Truth Table:-**

|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | X |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **0** |
| (c) | **0** | **1** | **0** |
| (d) | **0** | **0** | **1** |

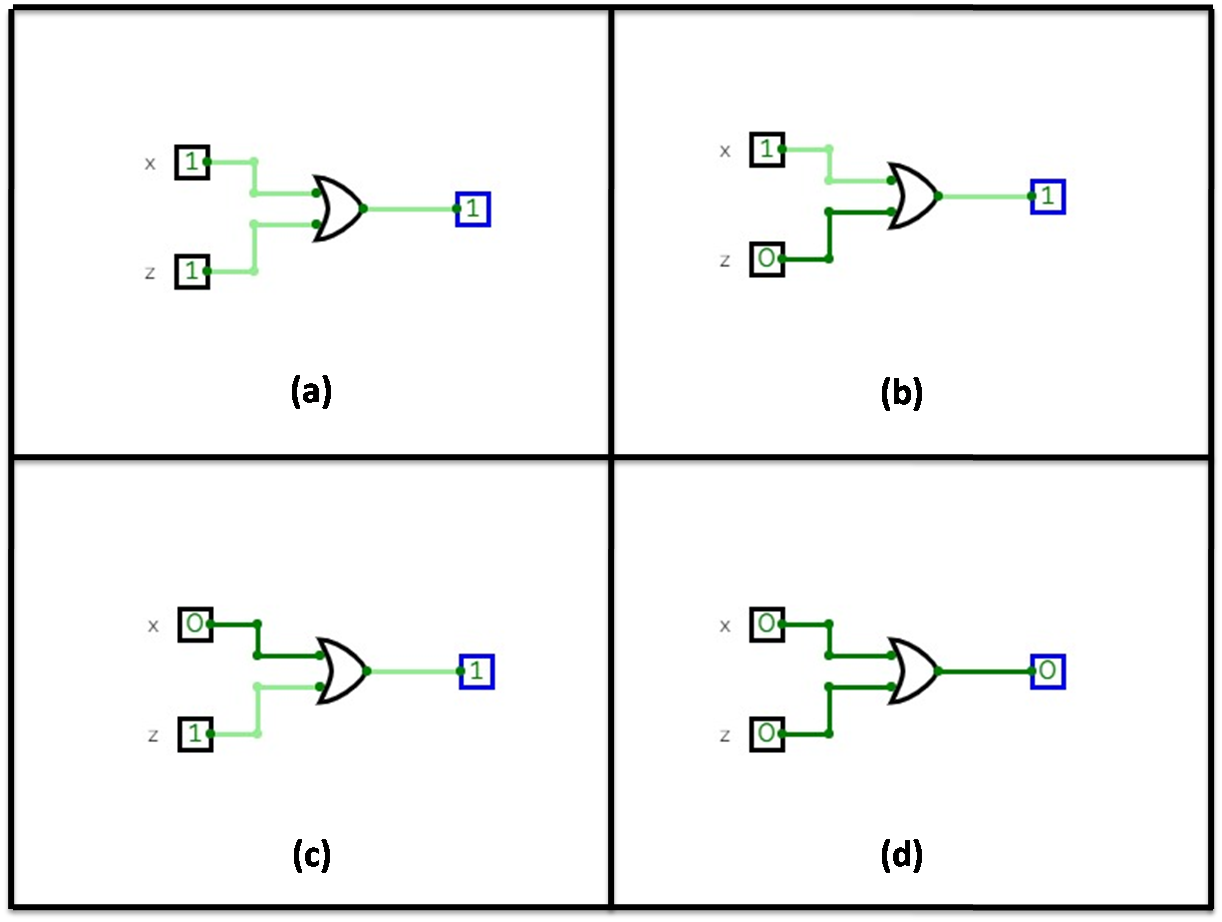
1. **Simplify the function with the minimum number of literals then draw the logic diagram of the simplified expression.**

Given function is

**🡪 Function f with minimum number of literals**

Truth Table:-

|  |  |  |  |
| --- | --- | --- | --- |
|  | x | z | F |
| (a) | **1** | **1** | **1** |
| (b) | **1** | **0** | **1** |
| (c) | **0** | **1** | **1** |
| (d) | **0** | **0** | **0** |



1. **Write the Boolean function of Question 6 in Product of sums form and implement it using NOR gates only.**

Boolean function of above question is

Now, this in POS (Product Of Sum) form can be written as:-

(From the truth table)

Its implementation using NOR gates only is shown in figure

