

Muhammad Bilal

0134-BSCS-19-B

Task: 1: Design and verify the truth table of a 3-bit Odd Parity generator and checker.

Table Circuit and Diagram

Circuit:

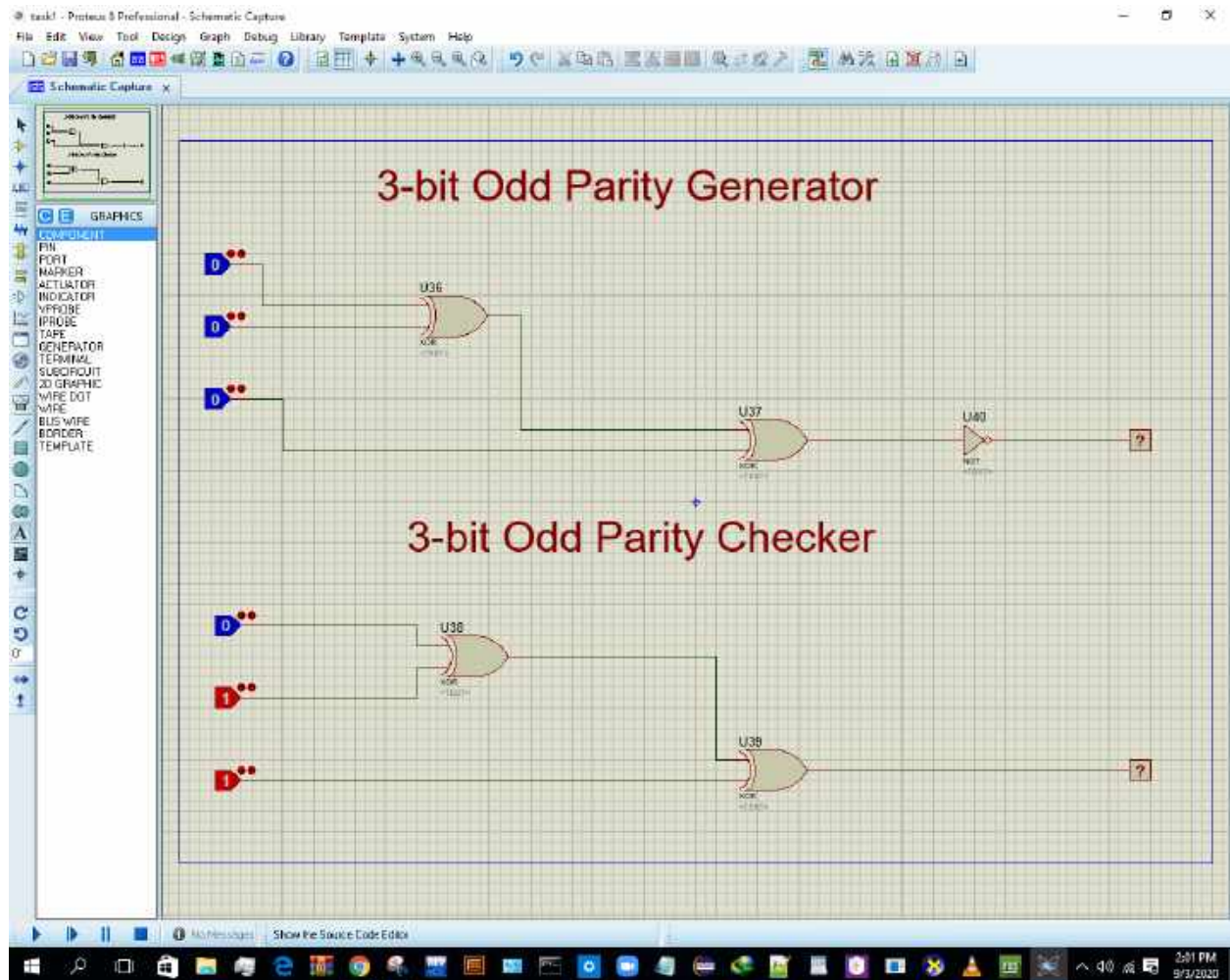


Table:

A\BC	00	01	1	10
0	1	0	1	0
1	0	1	0	1

Output Visualization

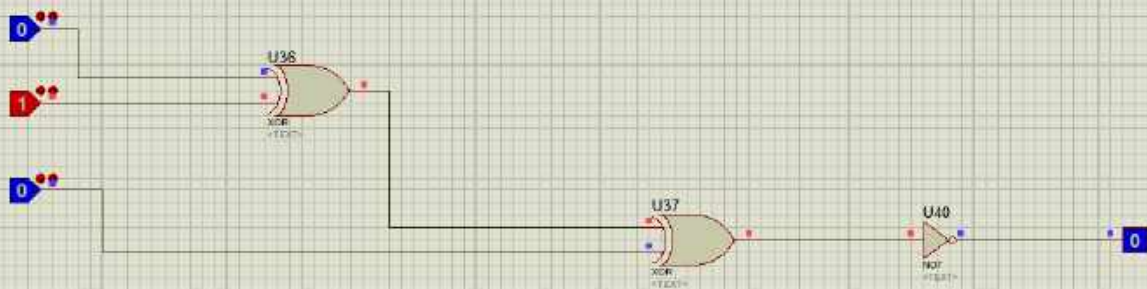
3-bit Odd Parity Generator



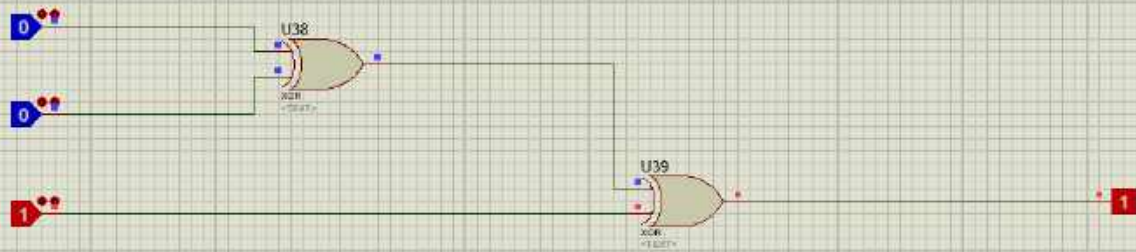
3-bit Odd Parity Checker



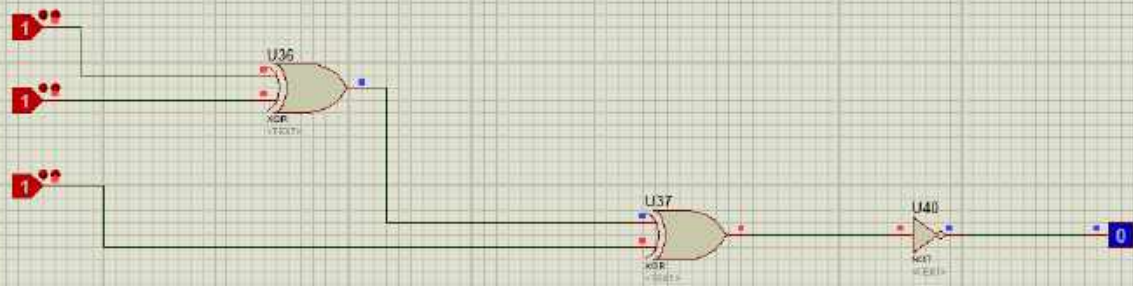
3-bit Odd Parity Generator



3-bit Odd Parity Checker



3-bit Odd Parity Generator



3-bit Odd Parity Checker

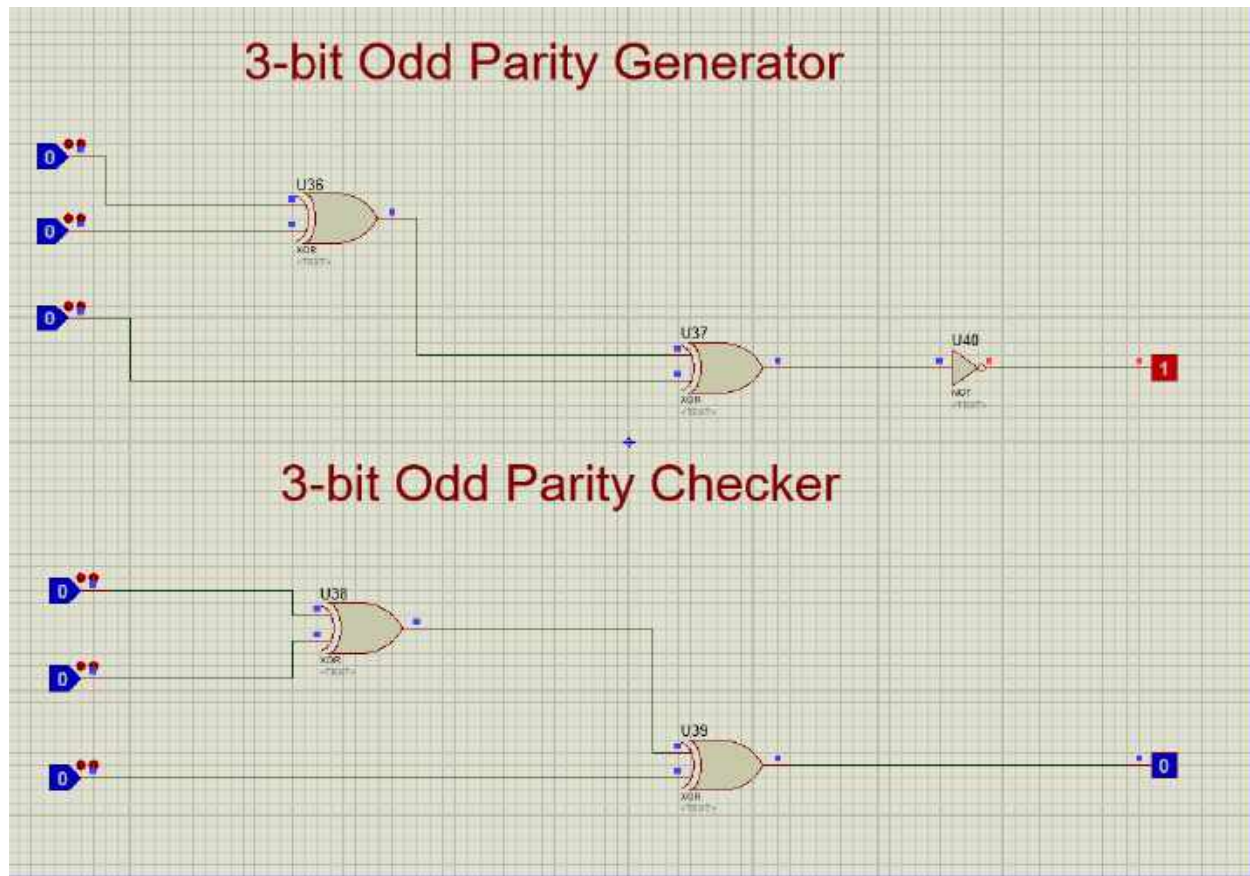


3-bit Odd Parity Generator



3-bit Odd Parity Checker





Viva Voce

(Viva Voce): What is the use of Parity generator and checker in digital logic design?

A combined circuit or devices of parity generators and parity checkers are commonly used in digital systems to detect the single bit errors in the transmitted data word caused by noise or other disturbances. The sum of the data and parity bits can either be even or odd.

Task: 4: Design and verify the SR-Gated Latch using Universal Gates (NAND or NOR Gates).

Table Circuit and Equation

Circuit:

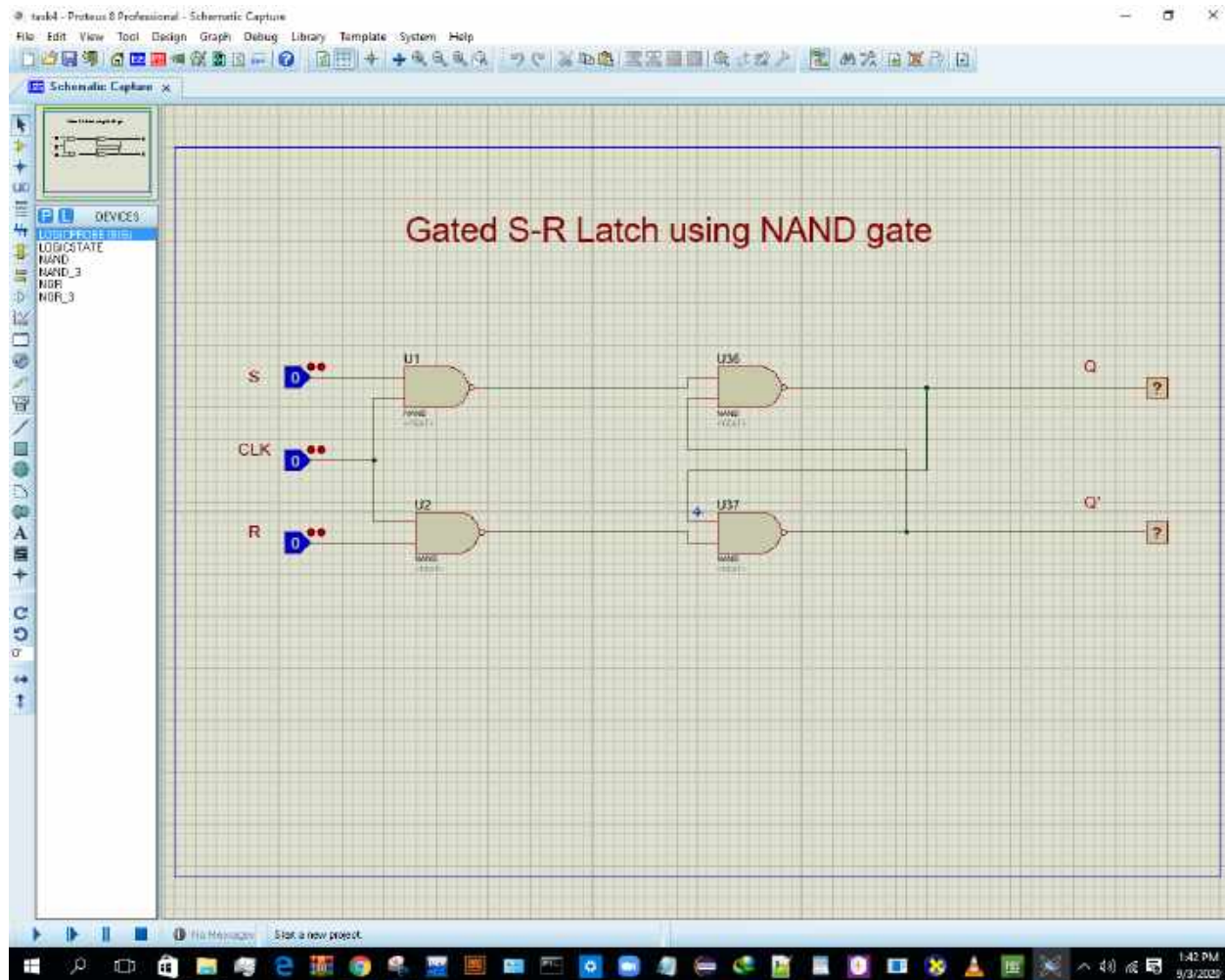
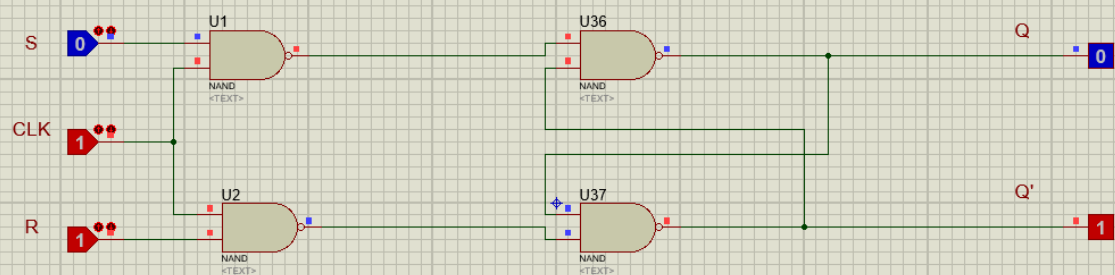


Table:

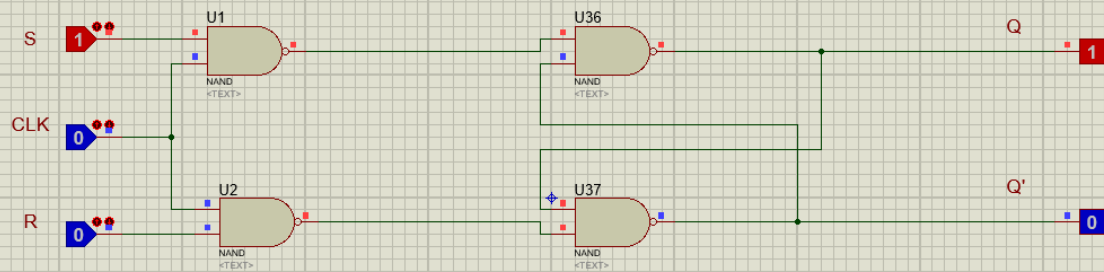
CLK	S	R	Q+
0	x	x	Q
1	0	0	Q
1	0	1	0
1	1	0	1
1	1	1	x

Output Visualization

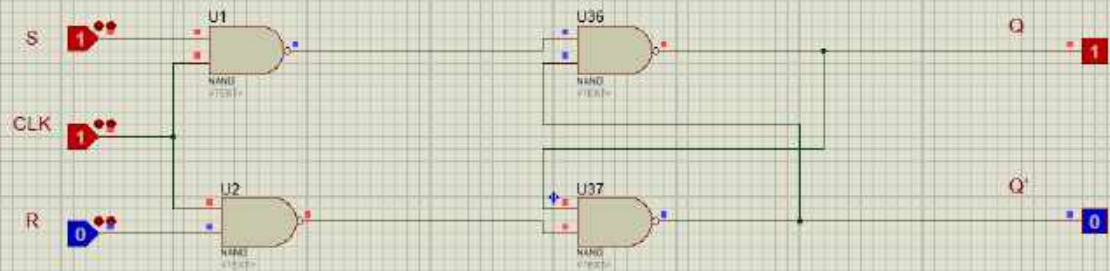
Gated S-R Latch using NAND gate



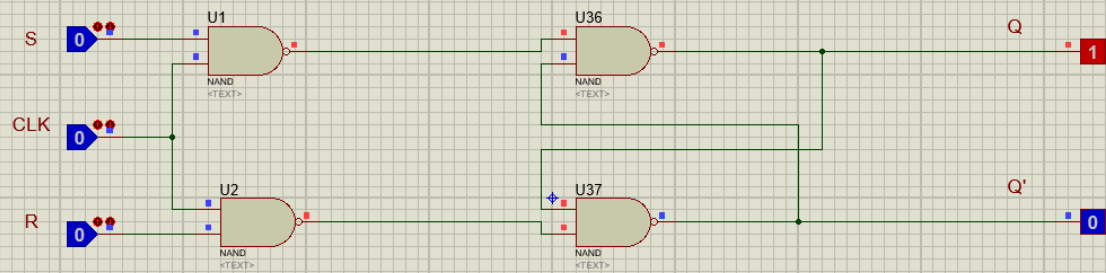
Gated S-R Latch using NAND gate



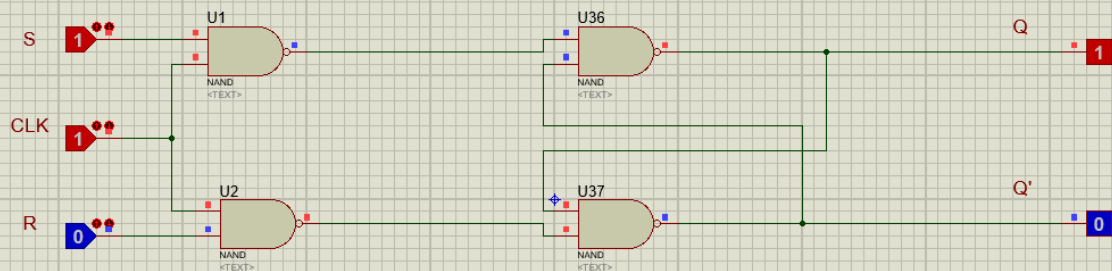
Gated S-R Latch using NAND gate



Gated S-R Latch using NAND gate



Gated S-R Latch using NAND gate



Viva Voce

(Viva Voce): Why race condition occurs in SR-Latch and how to overcome it?

The **race condition** occurs when both 'S' and 'R' are HIGH so the input changes to any other condition. And due to this the output becomes unpredictable.

This racing condition can be avoided if the HIGH time or clock ON is less than the propagation delay of the flip flop. If the flip flop is made to toggle over one clock period then racing can be avoided. Edge triggering instead of level triggering is used to achieve this.

.....

Task: 5: Implement the following Boolean expression using NAND gate Only?

$$D = \bar{A}C + A\bar{C} + B$$

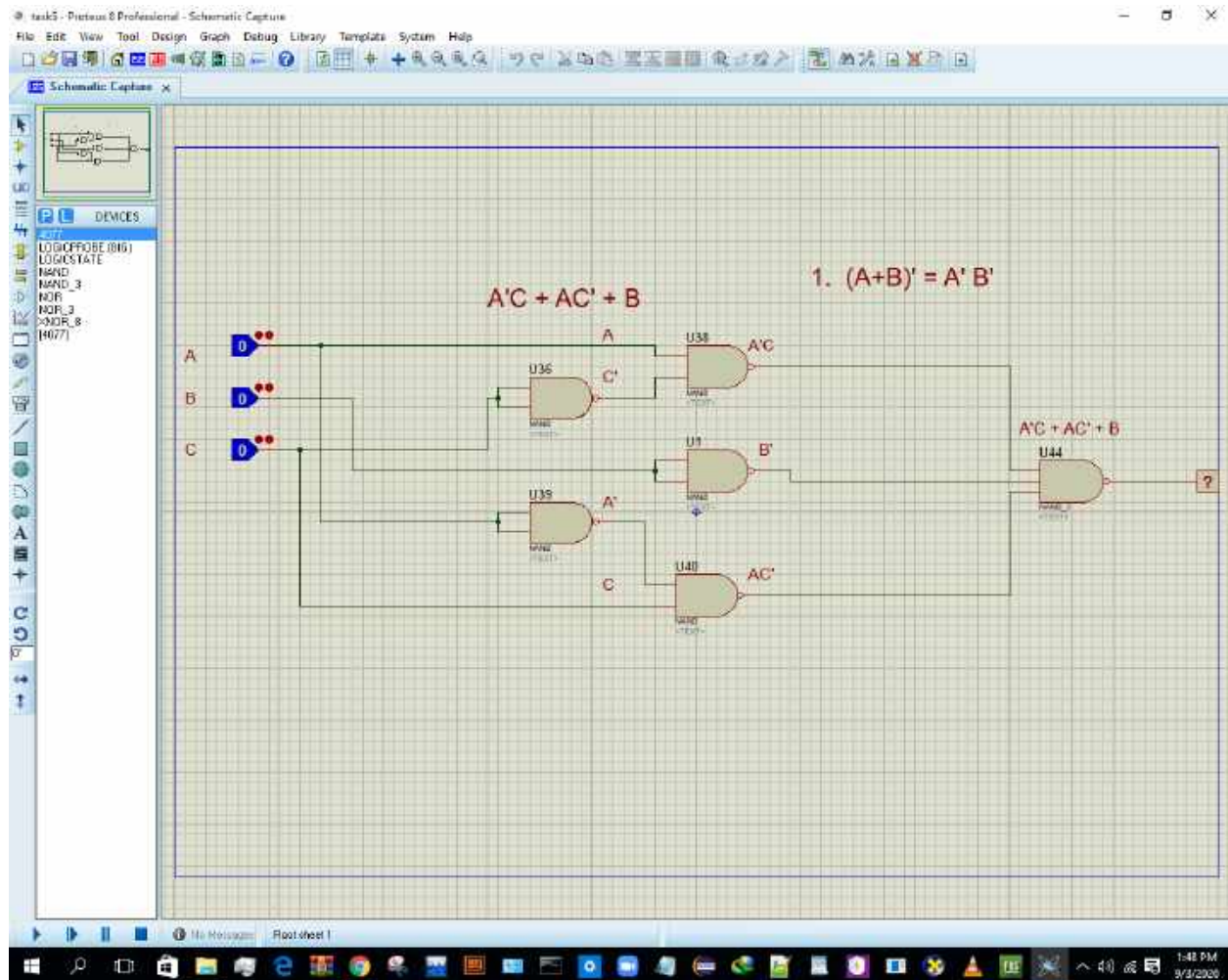
Table Circuit and Equation

Equation: $X = A'C + AC' + B$

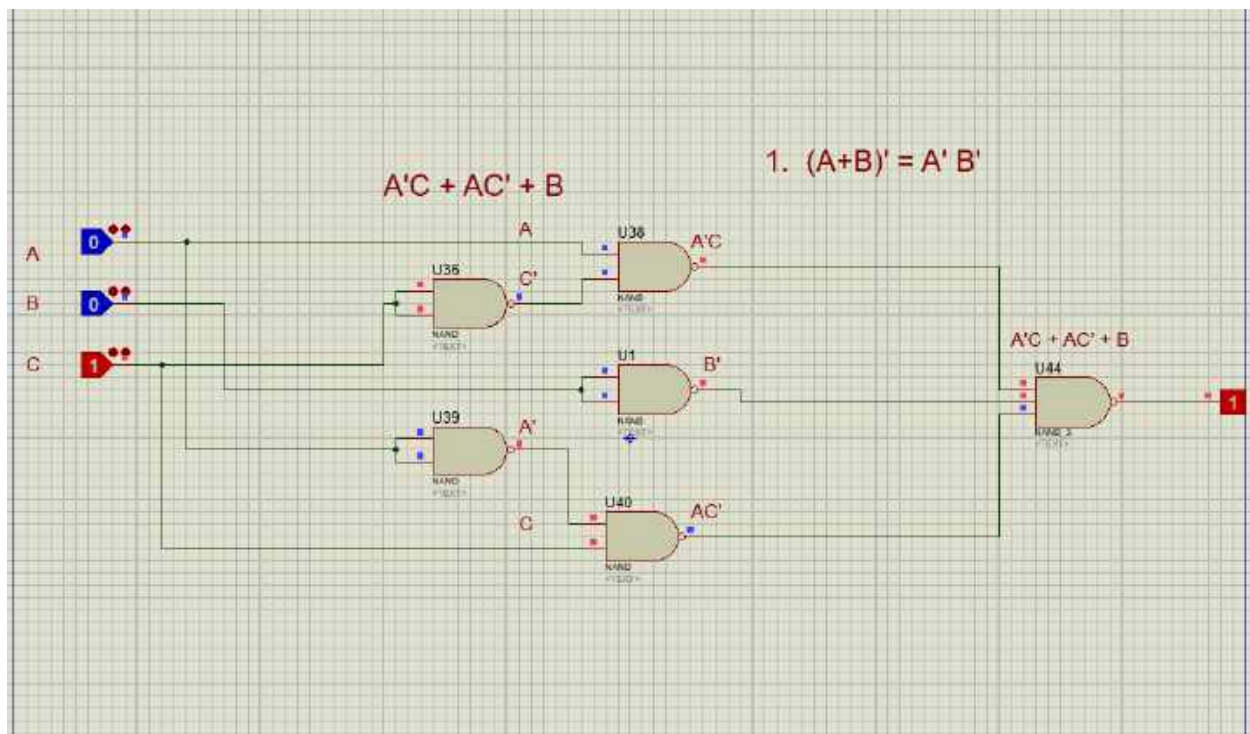
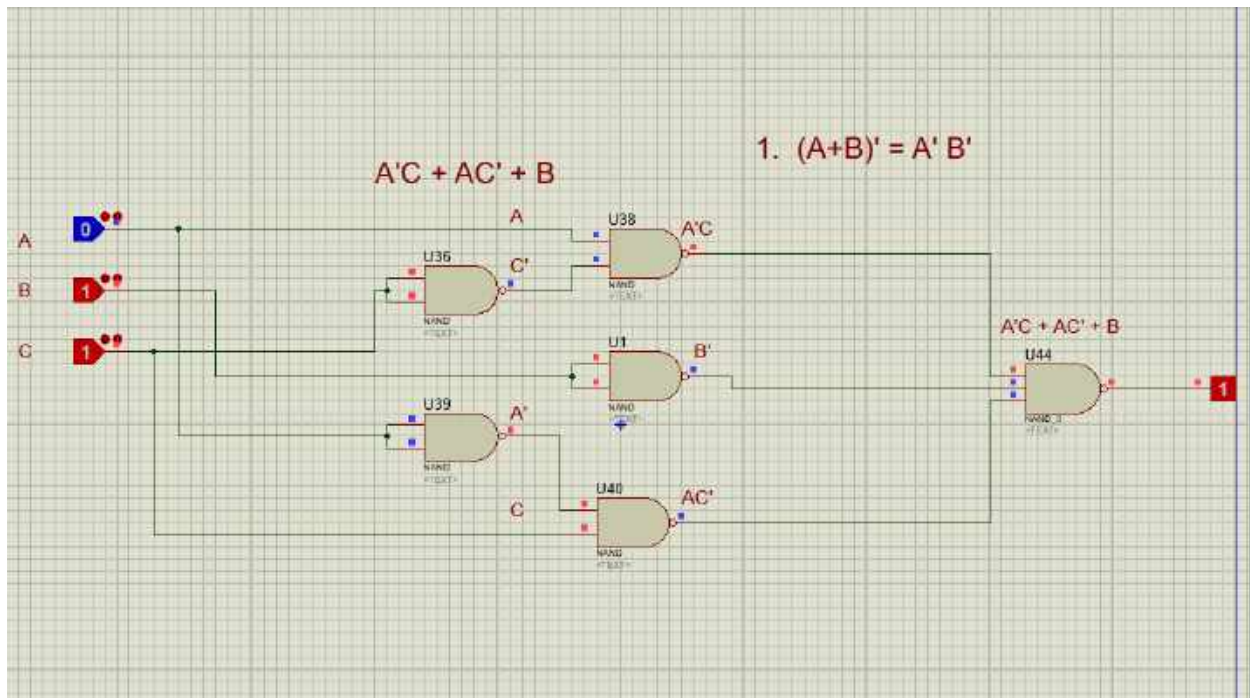
Table:

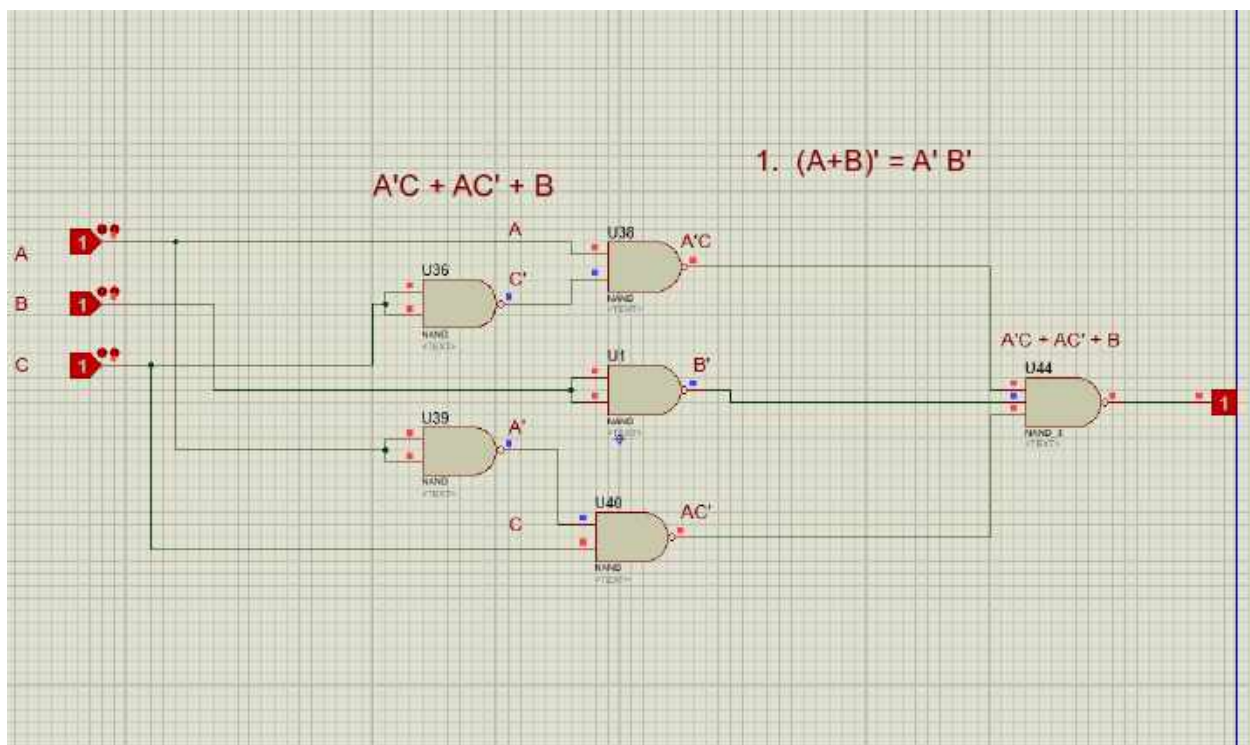
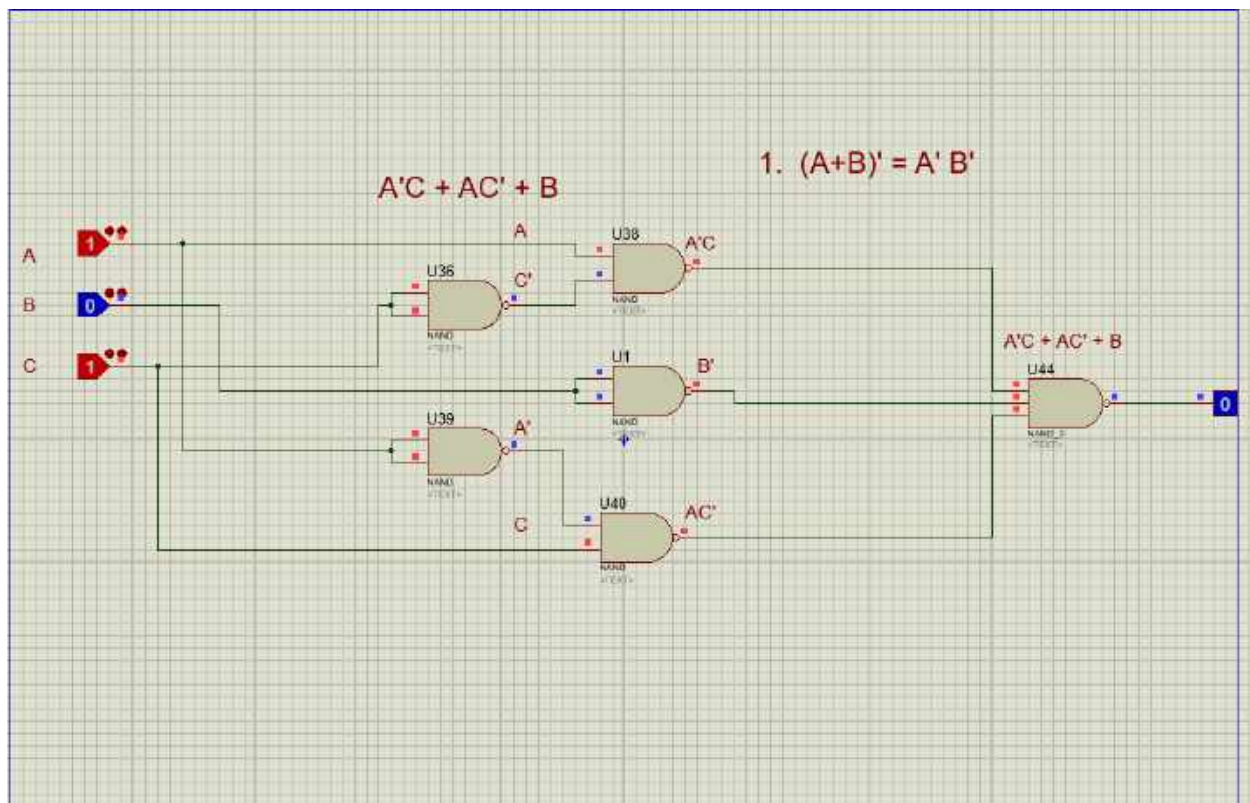
A	B	C	X
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

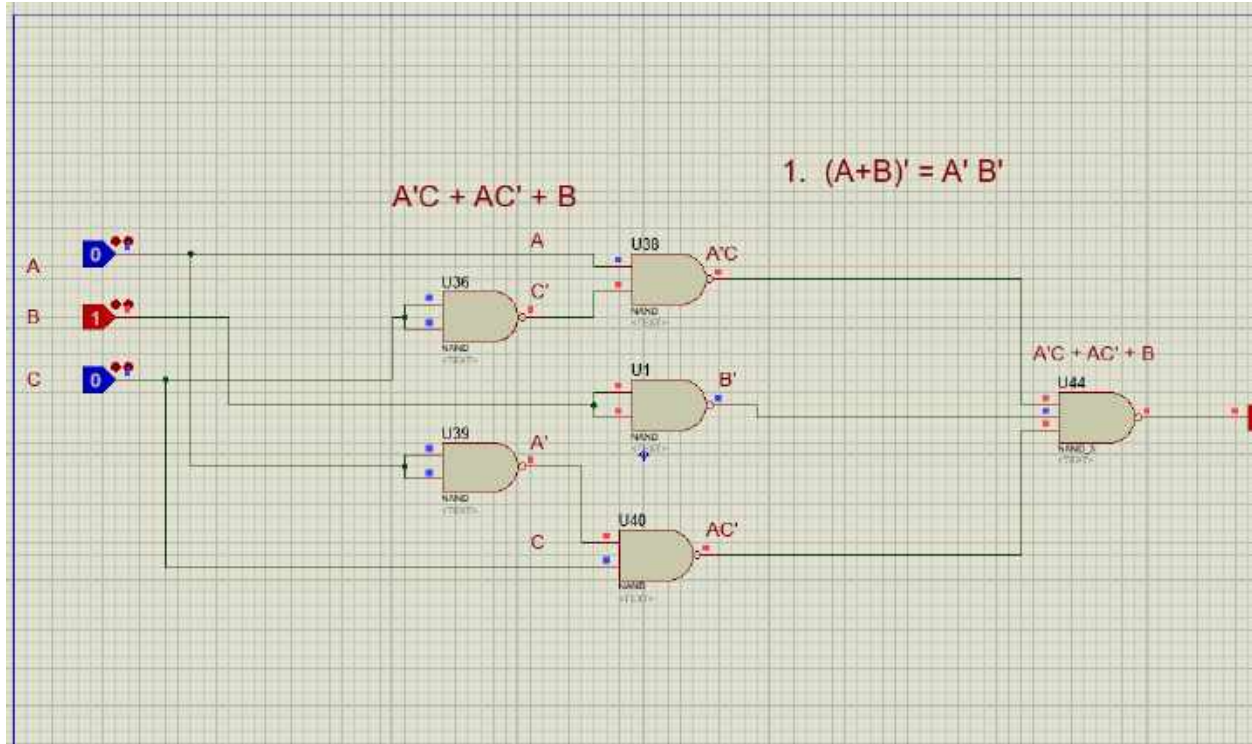
Circuit:



Output Visualization







Viva Voce

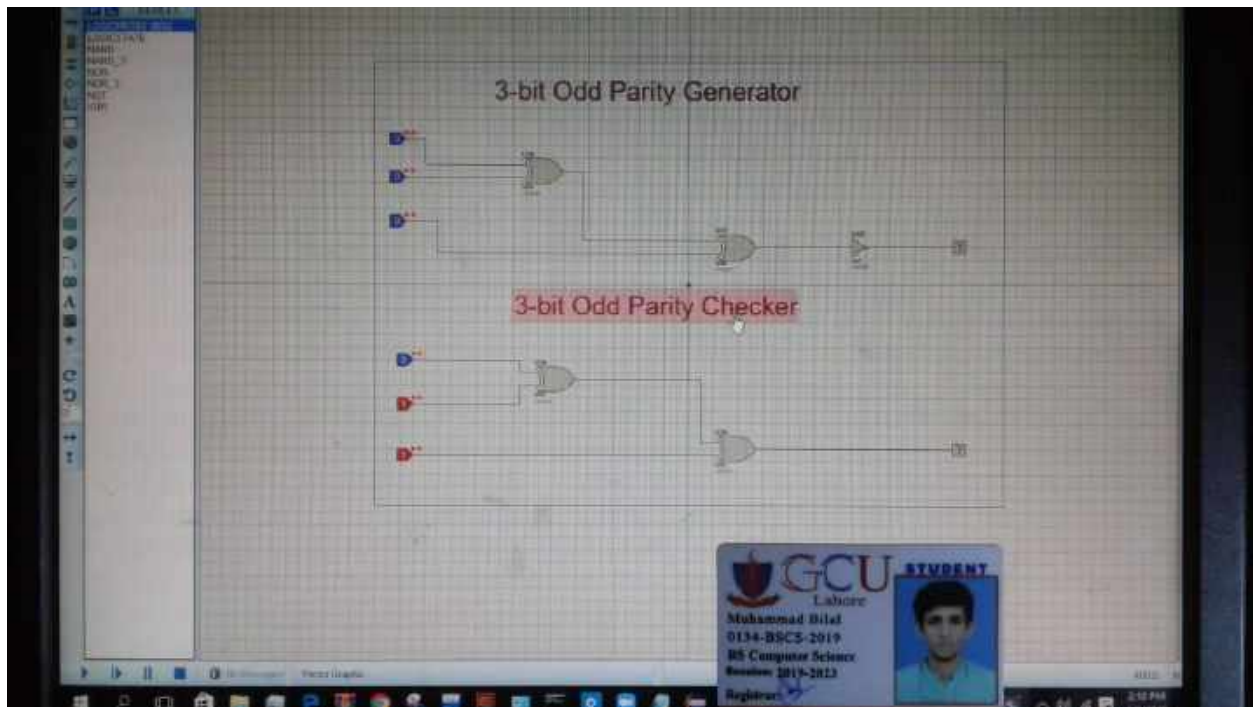
(Viva Voce): Why use only NAND gates or only NOR gates?

Because they have a unique significance and they are also called “Universal gates” because we can implement all other Boolean functions/operation by just using the combination of these two gates. This property is called functional completeness.

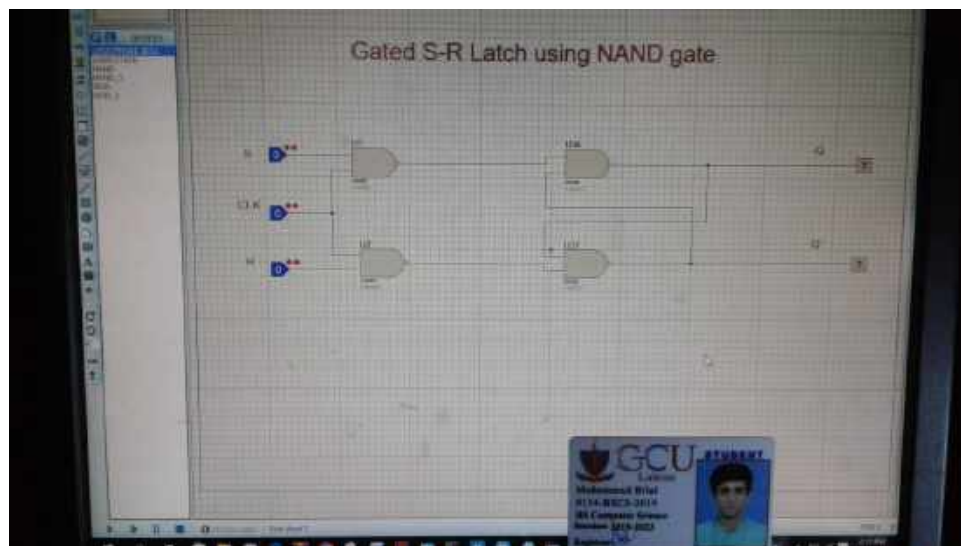
.....

Pictures

Task:1



Task:4



Task:5

