

## ***LOGISIM* SIMULTOR**

### **21. HALF ADDER**

#### **EXP.NO: 21**

**AIM:** To design and implement the two bit half adder using Logisim simulator.

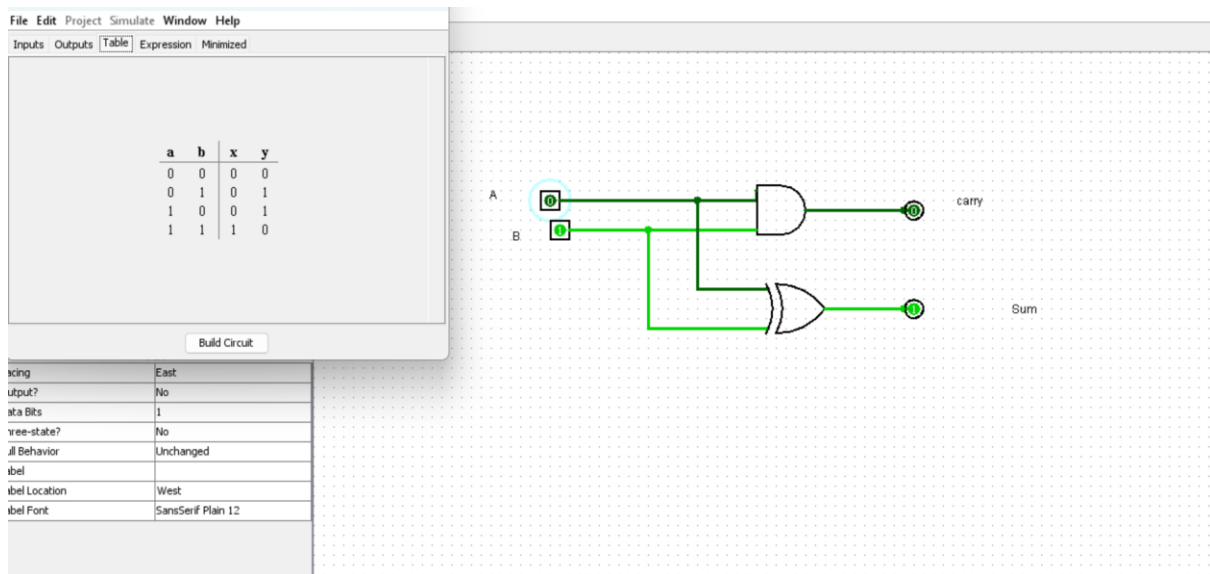
#### **PROCEDURE:**

- 1) Pick and place the necessary gates
- 2) Insert 2 inputs into the canvas.
- 3) Connect the inputs to the XOR gate and AND gate.
- 4) Insert 2 outputs into the canvas.
- 5) Make the connections using the connecting wires.
- 6) Verify the truth table.

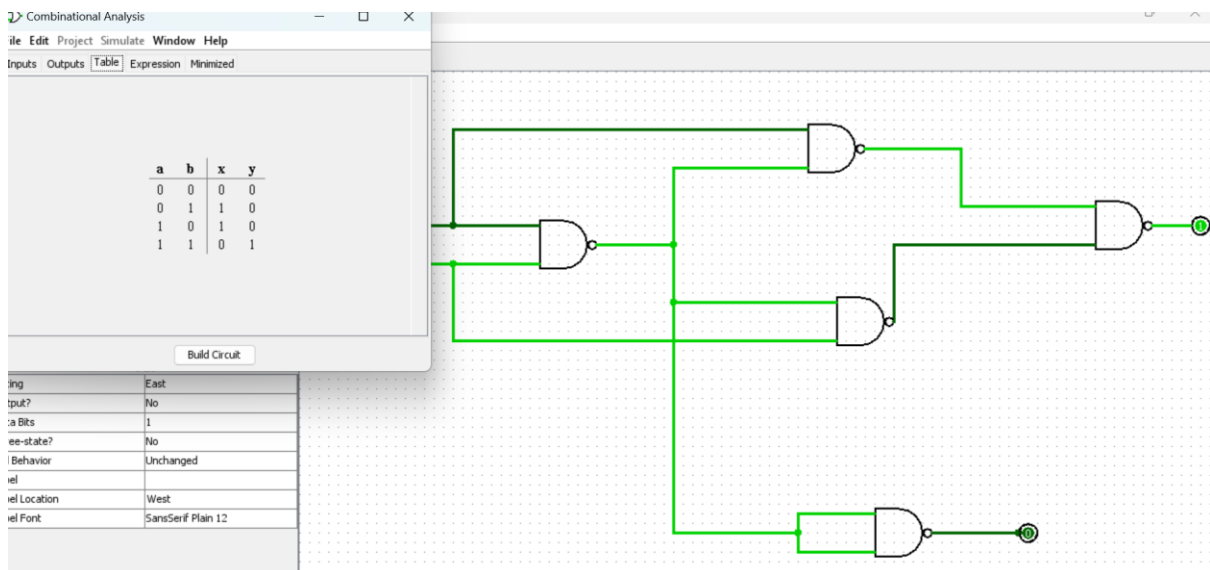
#### **TRUTH TABLE**

Truth Table			
Input		Output	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

## Logical Diagram:



## Half Adder using NAND Gates:



The screenshot shows the LogicWorks software interface. On the left, a window titled "a b x y" displays a truth table with four rows of data. Below the table is a "Build Circuit" button. On the right, a logic circuit diagram is shown on a grid background. The circuit has four inputs (a, b, x, y) and two outputs (z, w). The circuit consists of several logic gates: two 3-input AND gates, two 2-input OR gates, and one 3-input OR gate. The inputs are connected to the gates as follows: input 'a' is connected to the top input of the first 3-input AND gate and the top input of the first 2-input OR gate; input 'b' is connected to the middle input of the first 3-input AND gate and the bottom input of the first 2-input OR gate; input 'x' is connected to the bottom input of the first 3-input AND gate and the top input of the second 2-input OR gate; input 'y' is connected to the top input of the second 3-input AND gate and the bottom input of the second 2-input OR gate. The outputs of the first 3-input AND gate and the first 2-input OR gate are connected to the top input of the 3-input OR gate. The outputs of the second 3-input AND gate and the second 2-input OR gate are connected to the bottom input of the 3-input OR gate. The output of the 3-input OR gate is connected to output 'z'. The output of the first 2-input OR gate is connected to output 'w'.

a	b	x	y
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Build Circuit

## 22. TWO BIT HALF SUBTRACTOR

**AIM:** To design and implement the two bit half subtractor using Logisim simulator.

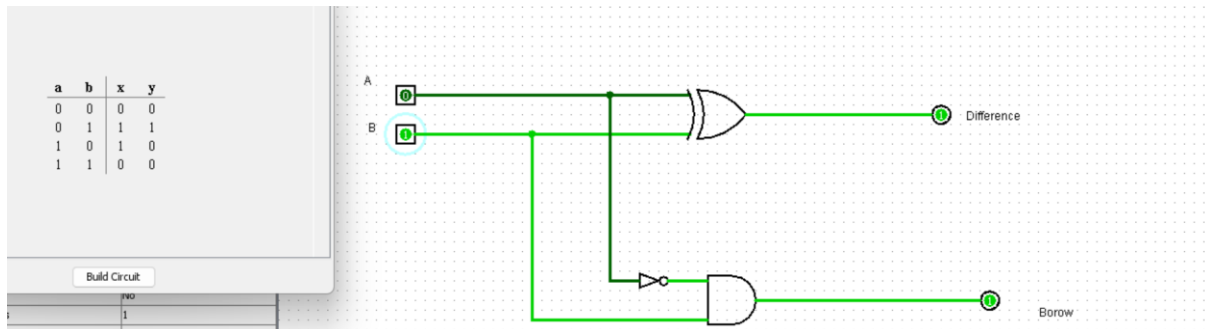
- 1) Pick and place the necessary gates.
- 2) Insert 2 inputs into the canvas.
- 3) Connect the inputs to the OR gate, AND gate and NOT gate.
- 4) Insert 2 outputs into the canvas.
- 5) Make the connections using the connecting wires.
- 6) Verify the truth table.

A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

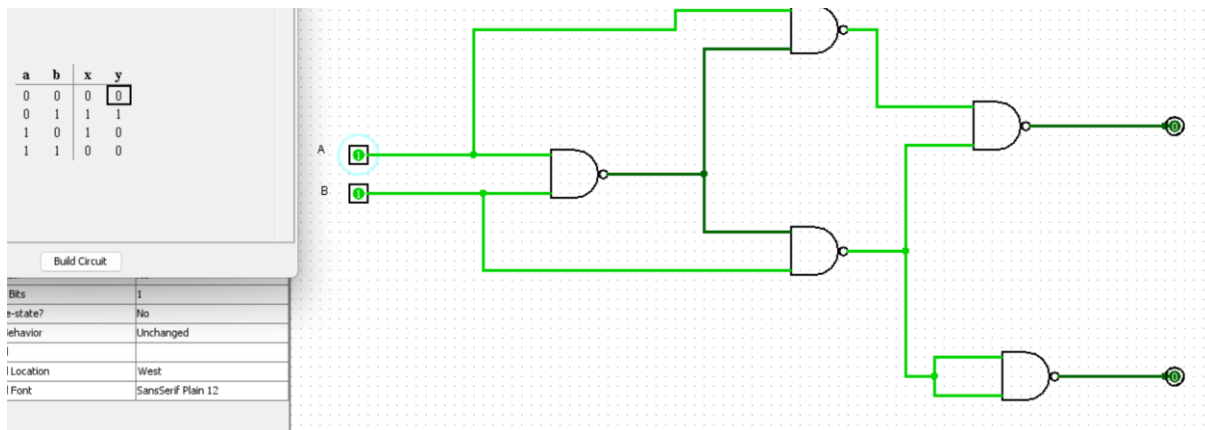
$$\text{Diff} = A'B + AB'$$

$$\text{Borrow} = A'B$$

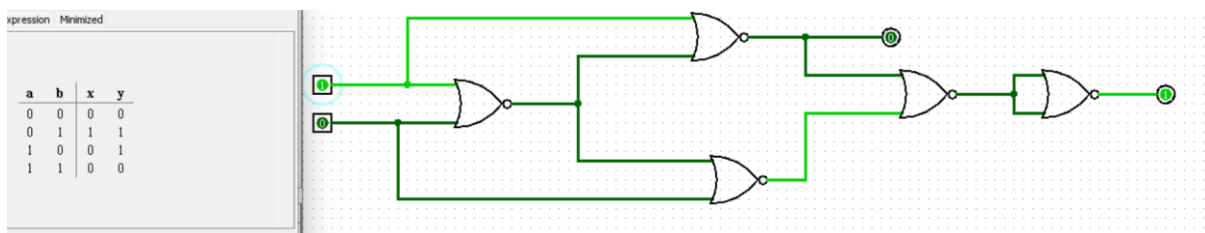
### Logical Diagram



### Half Subtractor using NAND Gates:



### Half Subtractor using NOR Gates:



**RESULT:** Thus 2-bit half subtractor has been designed and implemented successfully using logisim simulator

## 23. FULL ADDER

### EXP.NO: 23

**AIM:** To design and implement the full adder using Logisim simulator.

### PROCEDURE:

- 1) Pick and place the necessary gates.

- 2) Insert 3 inputs into the canvas.
- 3) Connect the inputs to the XOR gate, AND gate and OR gate.
- 4) Insert 2 outputs into the canvas.
- 5) Make the connections using the connecting wires.
- 6) Verify the truth table.

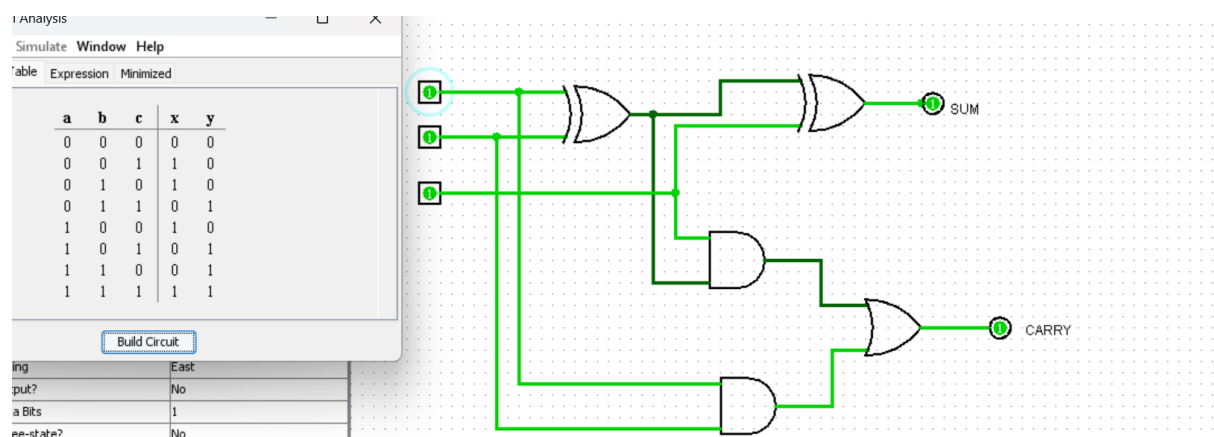
### TRUTH TABLE

Inputs			Outputs	
A	B	C <sub>in</sub>	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

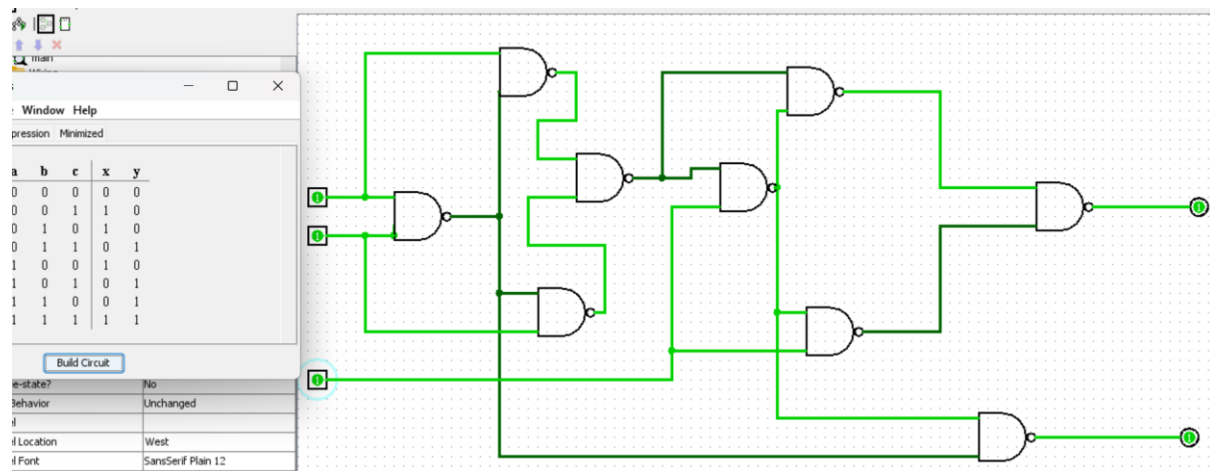
$$\text{Sum} = (A \oplus B)$$

$$\oplus \text{Cin Carry} = A.B + (A \oplus B)$$

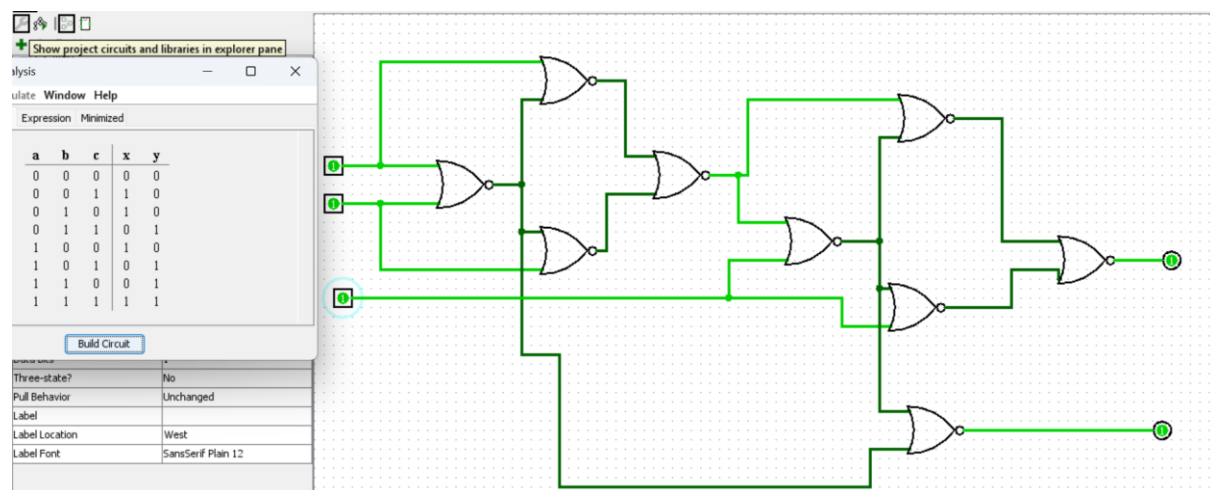
### Logical Diagram:



### Full adder using NAND Gates:



### Full adder using NOR Gates:



**RESULT:** Thus full adder has been designed and implemented successfully using logisim simulator.

## 24. FULL SUBTRACTOR

### EXP.NO: 24

**AIM:** To design and implement the full subtractor using Logisim simulator.

### PROCEDURE:

- 1) Pick and place the necessary gates.
- 2) Insert 3 inputs into the canvas.
- 3) Connect the inputs to the XOR gate, AND gate and OR gate.

- 4) Insert 2 outputs into the canvas.
- 5) Make the connections using the connecting wires.
- 6) Verify the truth table.

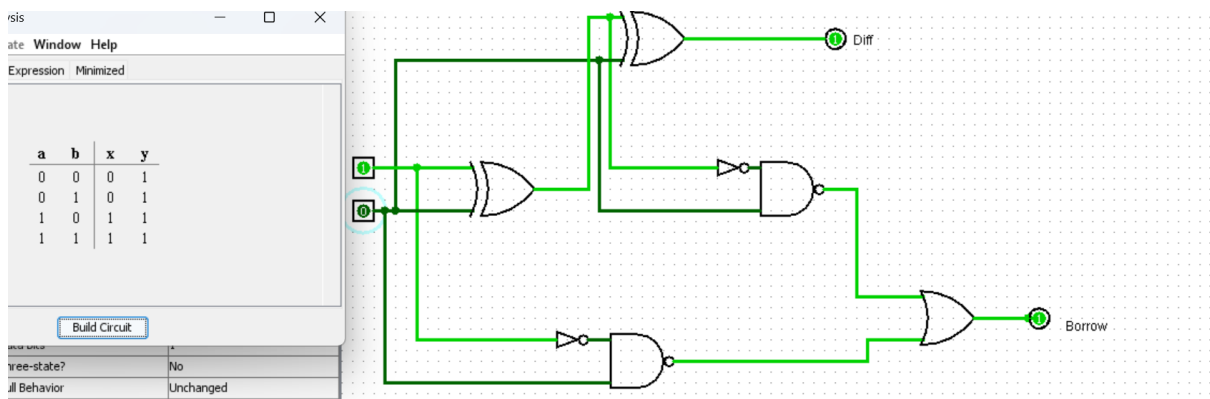
### TRUTH TABLE:

INPUT			OUTPUT	
A	B	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

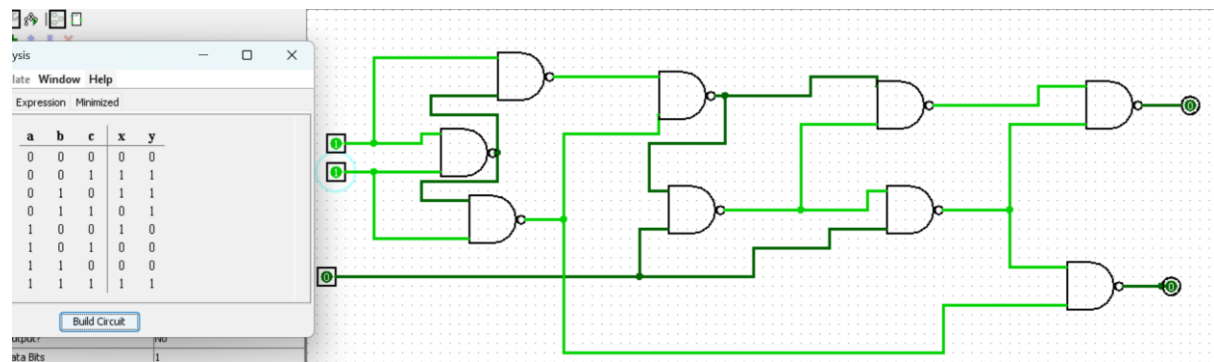
$$\text{Diff} = (A \oplus B) \oplus \text{Bin}$$

$$\text{'Borrow in' Borrow} = A'.B + (A \oplus B)'$$

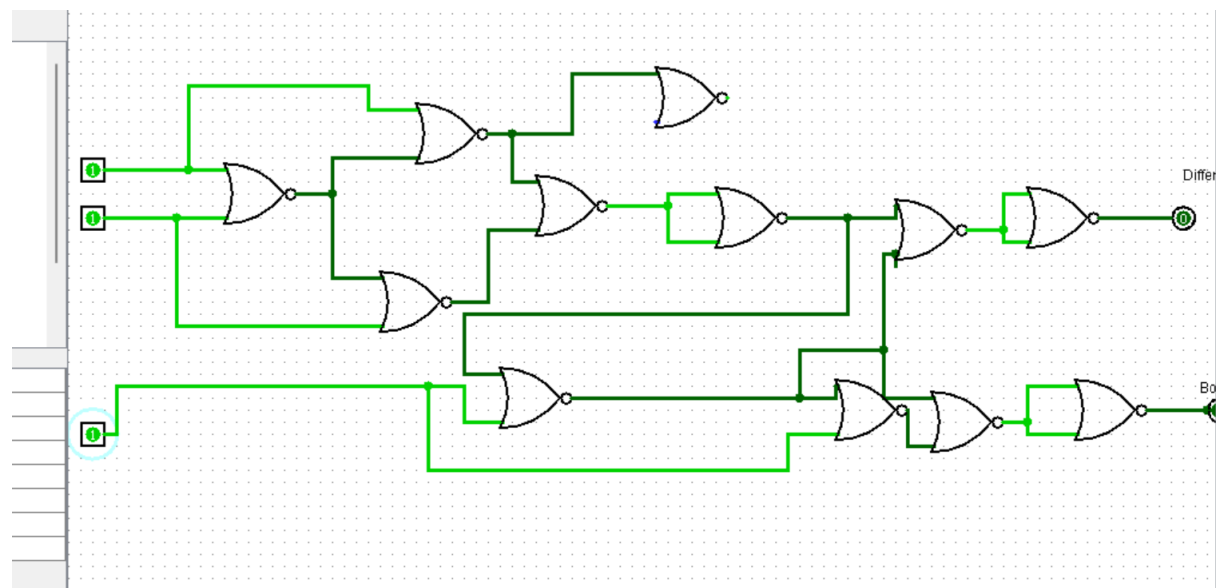
### Logic Diagram:



### Full Subtractor using NAND Gates:



### Full Subtractor using NOR Gates:



**RESULT:** Thus full subtractor has been designed and implemented successfully using logisim simulator