EXPERIMENT – 6

DESIGN OF COMPUTATIONAL PROCESSING SYSTEM FOR ARITHMETIC AND LOGICAL OPERATIONS

Aim: To design and verify the operation of Arithmetic and Logic Unit with various operations using Logisim.

Tools Required: Logisim

Pre-lab:

1. **What are the basic arithmetic operations performed by an ALU?**

   Answer: Addition, Subtraction, Multiplication, and Division.

2. **What are some common logic operations performed by an ALU?**

   Answer: AND, OR, NOT, XOR, XNOR (answers might vary depending on specific ALU design).

3. **List the components you will need from the Logisim library to build the ALU.**

   Answer: Arithmetic components: Adders, Subtractors (might be built from full adders/subtractors)

   Logic gates: AND, OR, NOT, XOR (depending on chosen logic operations)

   Multiplexers: To select between arithmetic and logic operations

   Decoders: To decode control signals for different operations

   Registers: To hold operands and results

4. **How will you verify the logic operations of your ALU (e.g., AND)?**

   Answer: Define test cases with different input bits (0, 1) and verify the expected output based on the chosen logic operation (e.g., 0 for AND if any input is 0).

Need and Purpose:

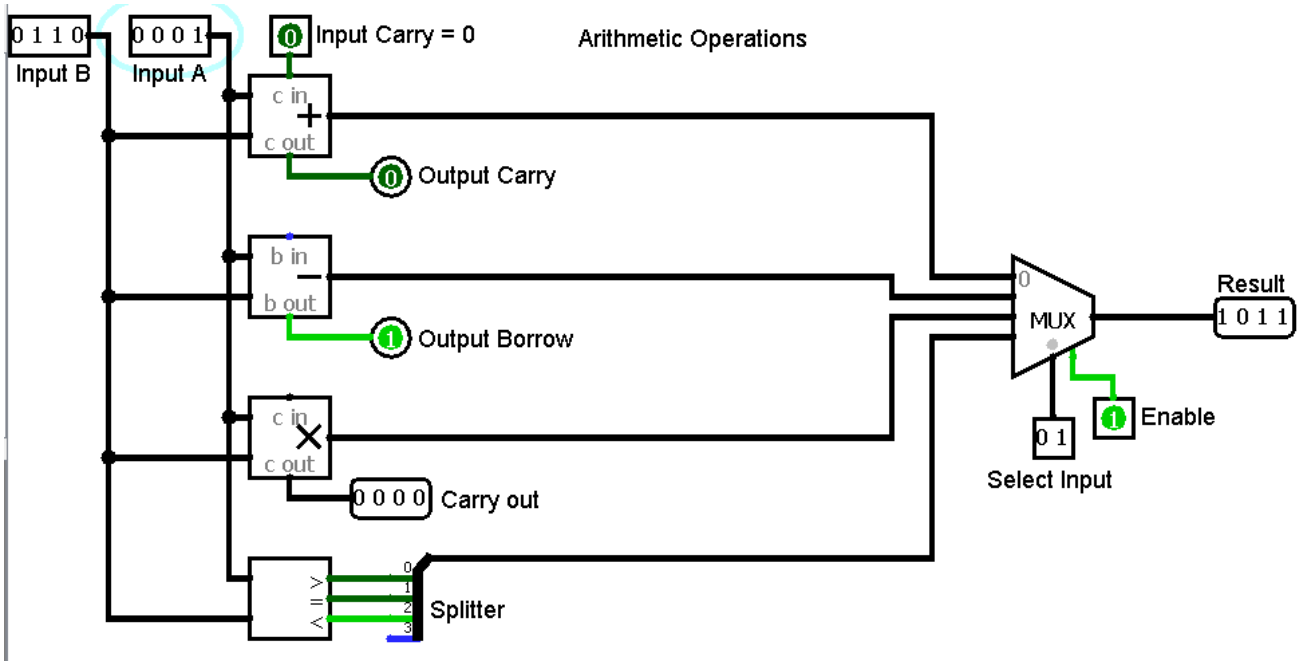
ALU constitutes efficient alternate modes for data processing for multiple data sizes. ALU is a crucial module used in computational systems like digital systems, image processing, signal processing, cryptography etc., to carry out arithmetic and logical operations.

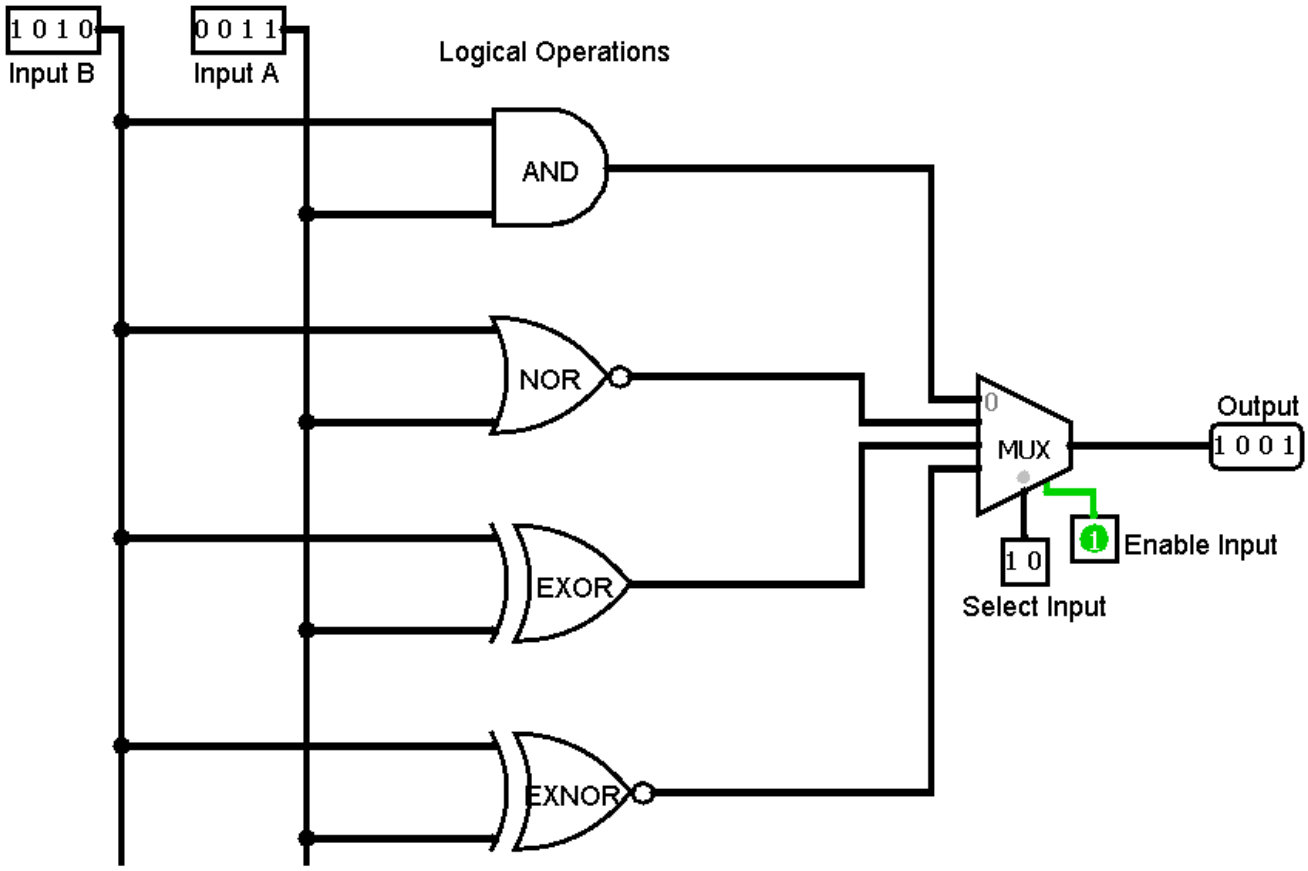
As ALU is the main part of many mathematical computational systems, the designing of ALU is the main designing parameter in the computational system. Without ALU fundamental building block central processing unit (CPU) of a computer cannot be a valid computational system. Here it is intended to design an ALU unit which will be implementing basic arithmetic and logical operations.

Functional table:

S. No	Type of operations	Name of operations
1	Arithmetic	1. Addition 2. Subtraction 3. Multiplication 4. Comparison
2	Logical	1. AND 2. XOR 3. XNOR 4. NOR

Circuit Diagram:





Procedure:

1. Set up the arithmetic and logical operational circuits as outlined in the circuit diagram provided.
2. Apply the specified 4-bit inputs, labeled as "A" and "B", to their respective inputs in the circuit.
3. Adjust the select inputs on the multiplexer to choose the desired arithmetic or logical operation.
4. Modify the data inputs as needed and note the variations in the output corresponding to the selected operation on the multiplexer.

Viva Questions and answers:

1. Explain the concept of a carry bit in an ALU's addition operation.
2. How does an ALU differentiate between addition and subtraction?
3. What are the advantages of using a multiplexer in ALU design?
4. Why did you choose specific logic gates for your ALU's logic operations? (e.g., AND, OR)
5. How did you decide on the data width for your ALU design?

6. Explain the decoding process for the control signals in your ALU.
7. Describe the challenges you might face when designing a multi-bit ALU compared to a single-bit ALU.

Result: The experiment successfully demonstrated the construction and operation of arithmetic and logical operations of ALU.