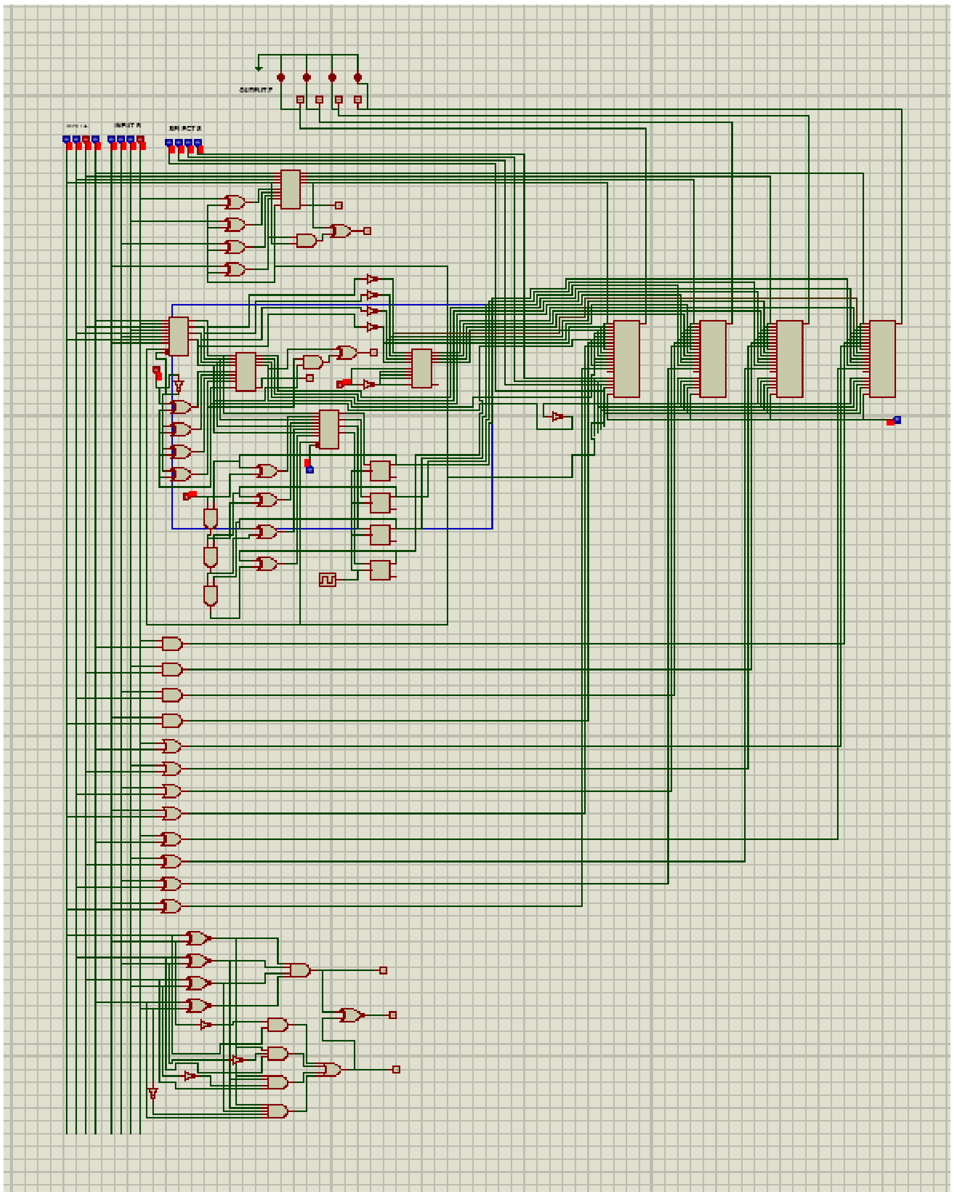
PROJECT REPORT

Arithmetic Logic Unit

By: Muhammad Umar Farooq

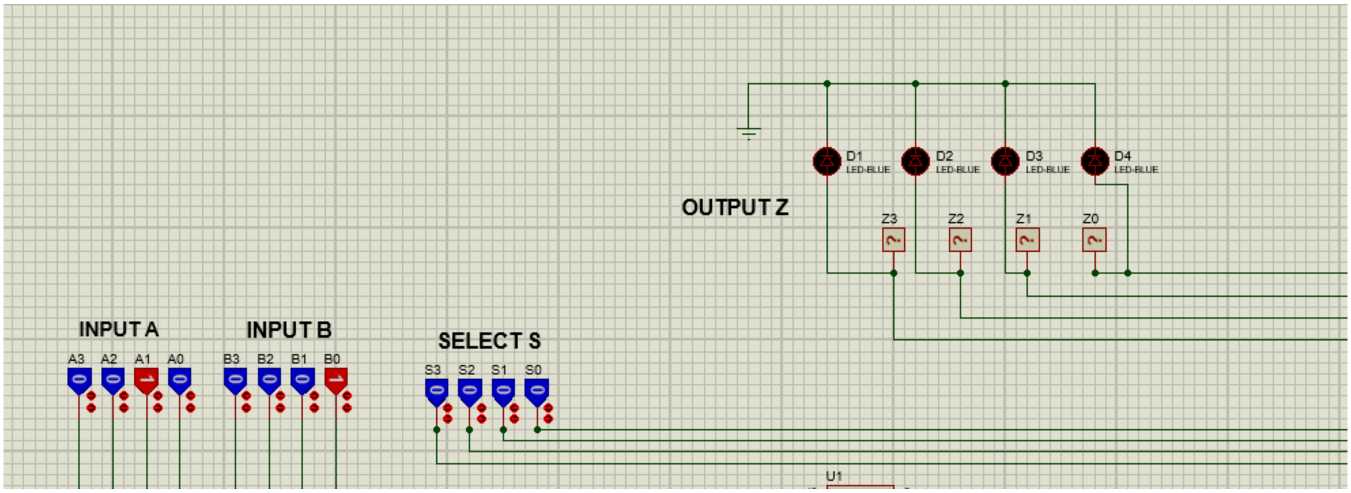
Abstract:
In the given project, we have designed an Arithmetic Logic Unit that takes two four-bit inputs and then performs certain arithmetic operations on it like addition, increment, comparison, etc. We can also select which operation's result we want to view on the output probe through a selector.



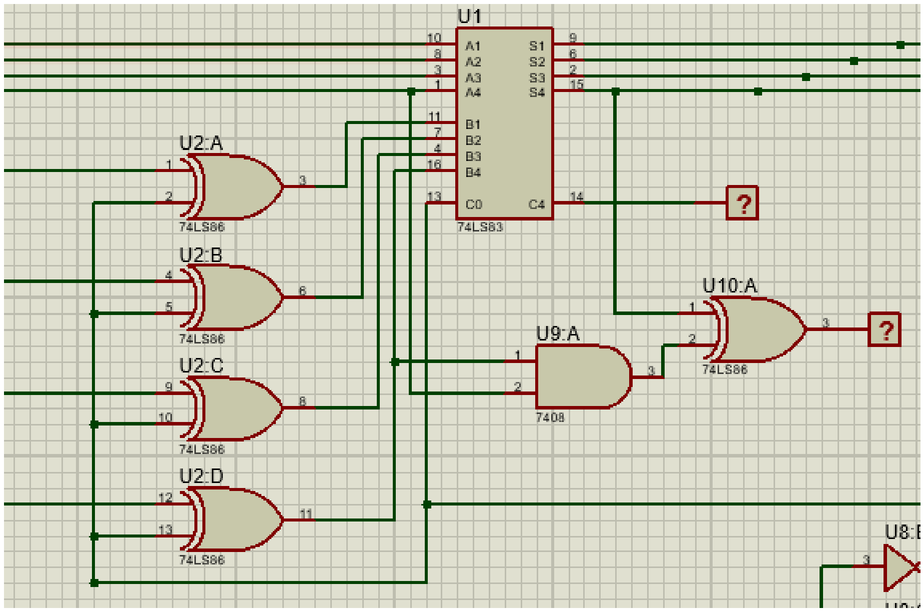
Introduction:

An ALU is a combinational digital electronic circuit that performs arithmetic and bitwise operations on integer binary numbers. The inputs to an ALU are the data to be operated on, called the operands, and a code indicating the operation to be performed; the ALU's output is the result of the performed operation.

1. **Parts of the ALU:**
   - **<u>Input Output Probes:</u>**
     Here our circuit takes the Input A and Input B. With that, we select the operation we want to perform from the SELECT S. The output will be shown at the output probes and also through the LEDs connected to the probes.

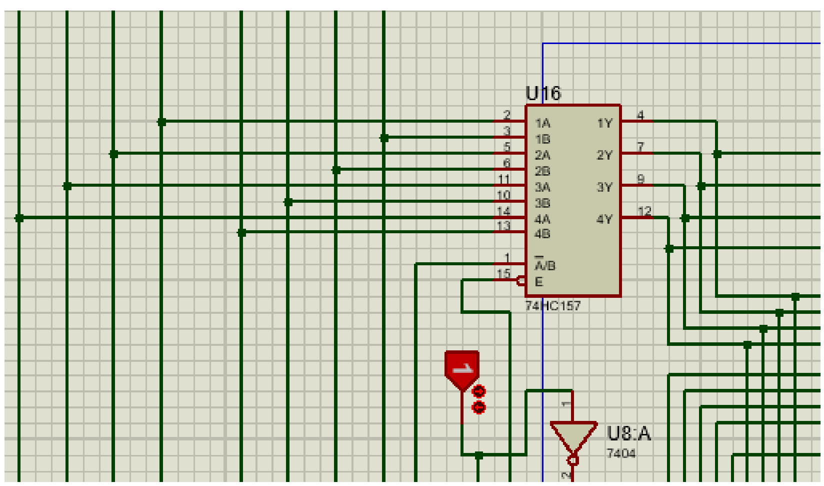


   - **<u>Addition/Subtraction:</u>**
     Here, we are using a ripple carry adder IC that consists of four full adders. The one IC can both add and subtract our two inputs. This is achieved by connecting the B input with XOR gates and making carry in 1 and one input of all XOR gates 1 which in turn takes the two's compliment of input B thus subtracts B while doing addition. Outflow is also shown and is calculated using expression:
     Outflow = (A3 AND B3) XOR S3.



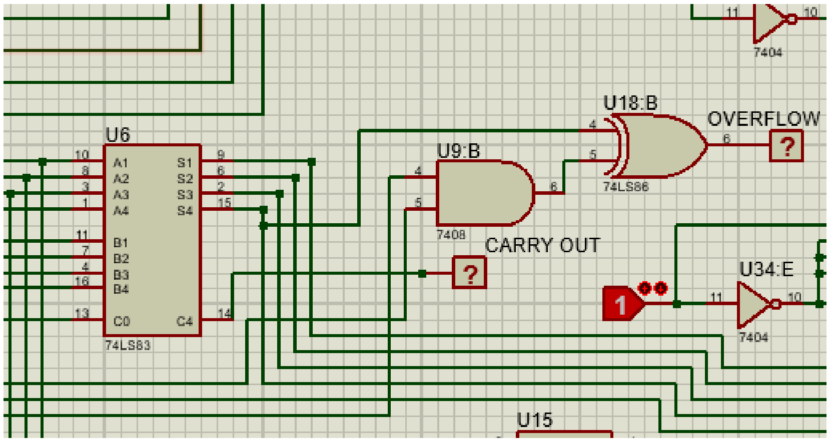
- **Input Selector**

We make use of a 2 to 1 multiplexer to select between inputs A and B. This selection is done to achieve efficiency so we may not have to use multiple ICs for same operations on different inputs.

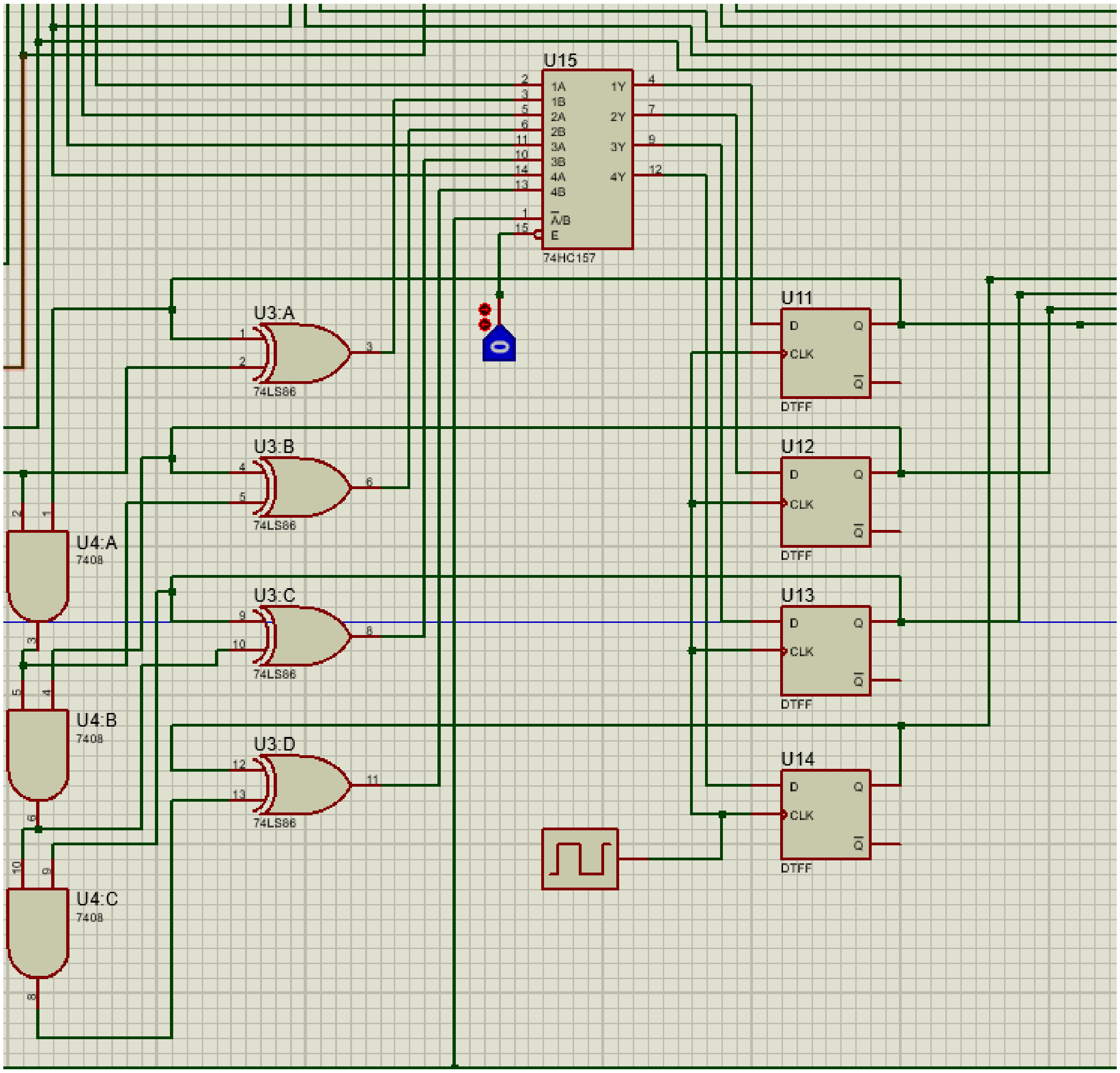


- **Increment/Decrement**

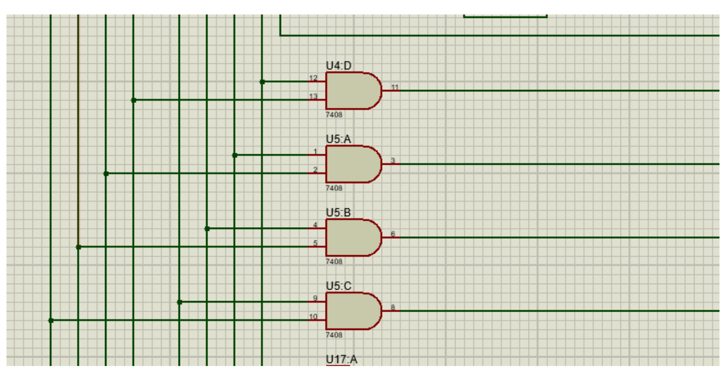
  We use 7483 adder IC in order to increment and decrement our inputs as show. Only one IC can perform both tasks and this is done by takin two's compliment 1 and then adding it when we want to decrement.



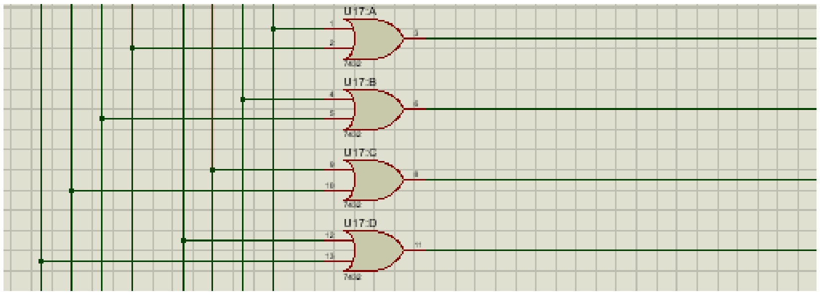
- **Loader/Counter**
  We are using a multiplexer to load values into the counter. The counter then makes use of D flip flops to count forward numbers in order.



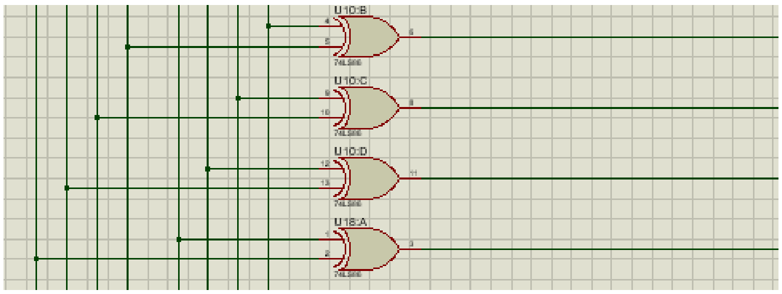
- **Bitwise AND**
  Four AND gates are used here to find and of individual input A’s bits with B’s bits.



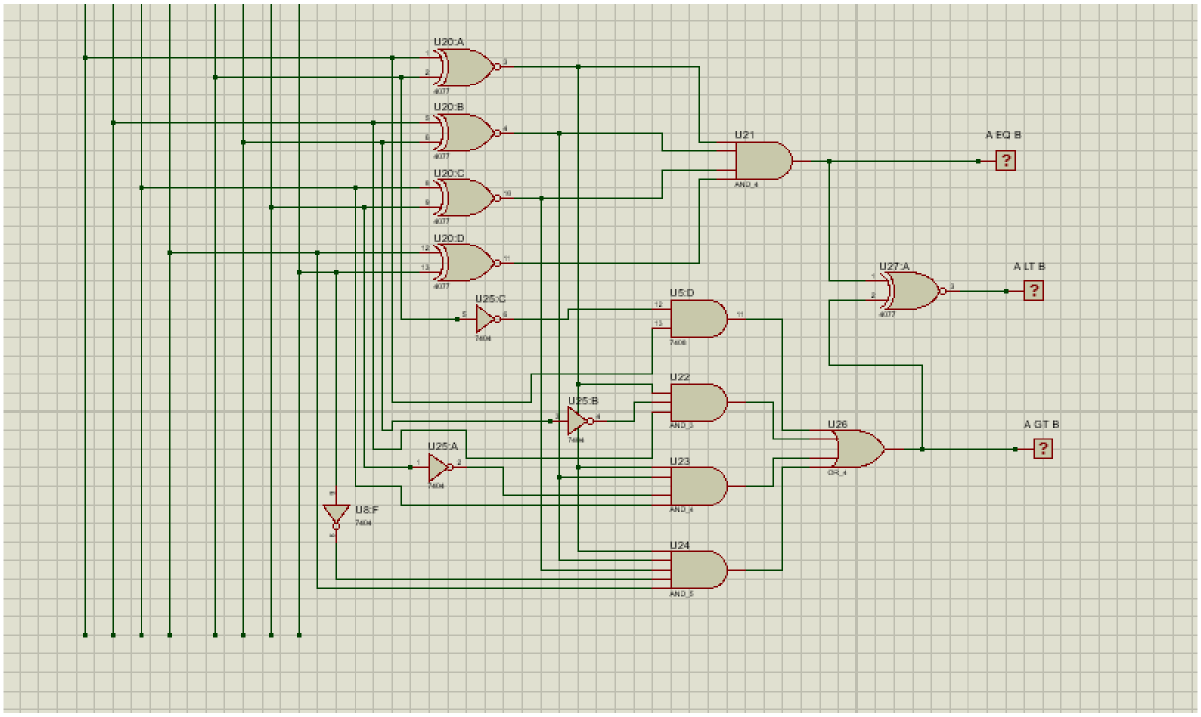
- **<u>Bitwise OR</u>**
  Four OR gates are used here to find or of individual input A’s bits with B’s bits.



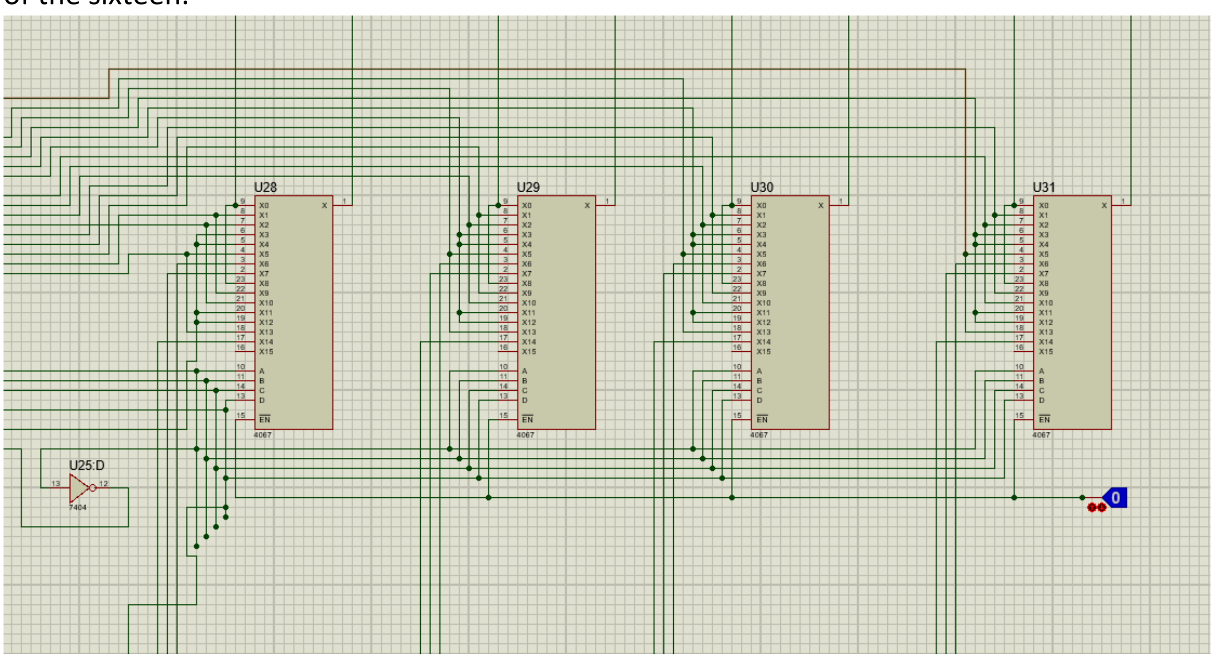
- **<u>Bitwise XOR</u>**
  Four XOR gates are used here to find and xor individual input A’s bits with B’s bits.



- **<u>COMPARSON</u>**
  We make use of XOR, And, OR and NOT gates to build two blocks which take input A and B. The first block checks weather both the inputs are equal and gives out output 1 and the second block checks if A is greater than B by checking from most significant bit to least significant bit and gives out A if it is true. Else they give out 0.



- **<u>Output Selector</u>**
  Four 16 to 1 multiplexers are used in this part. They take input from the outputs of the sixteen operations being performed in the circuit and each multiplexer's output bit is a part of the 4 bit final output and according to the selector gives out the one required output out of the sixteen.



2. Operations Selection Table

0000	Addition
1000	Subtraction
0001	A’s 1 Complement
1001	B’s 1 Complement
0010	A’s 2 Complement
1010	B’s 2 Complement
0011	A Increment
1011	B Increment
0100	A Decrement
1100	B Decrement
0101	Loader
1101	Counter
0110	Bitwise AND
0111	Bitwise OR
1110	Bitwise XOR
1111	Do not Care

Conclusion:

After completion of this project by stimulating it on software Proteus, we see that the design implemented by us gives negligible errors. Proving that an ALU is a combinational digital electronic circuit that performs arithmetic and bitwise operations on integer binary numbers.