01). ADDITION/ SUBTRACTION OF TWO 4-BIT NUMBERS

Addition of two 2-bit numbers

By using full adder concept we can add/ substract two 4-bit numbers and generate the output.

To generate a logic circuit, we can use AND gate, OR gate and XOR gate.

The truth table for the full adder is as follows which adds two 1-bit numbers,

A	B	Cin	Cout	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Cout = (A'.B.Cin)+(A.B'.Cin)+(A.B.Cin')+(A.B.Cin)

A \ B Cin	00	01	11	10
0			1	
1		1	1	1

$\boxed{\text{Cout=A.B+B.Cin+A.Cin}}$

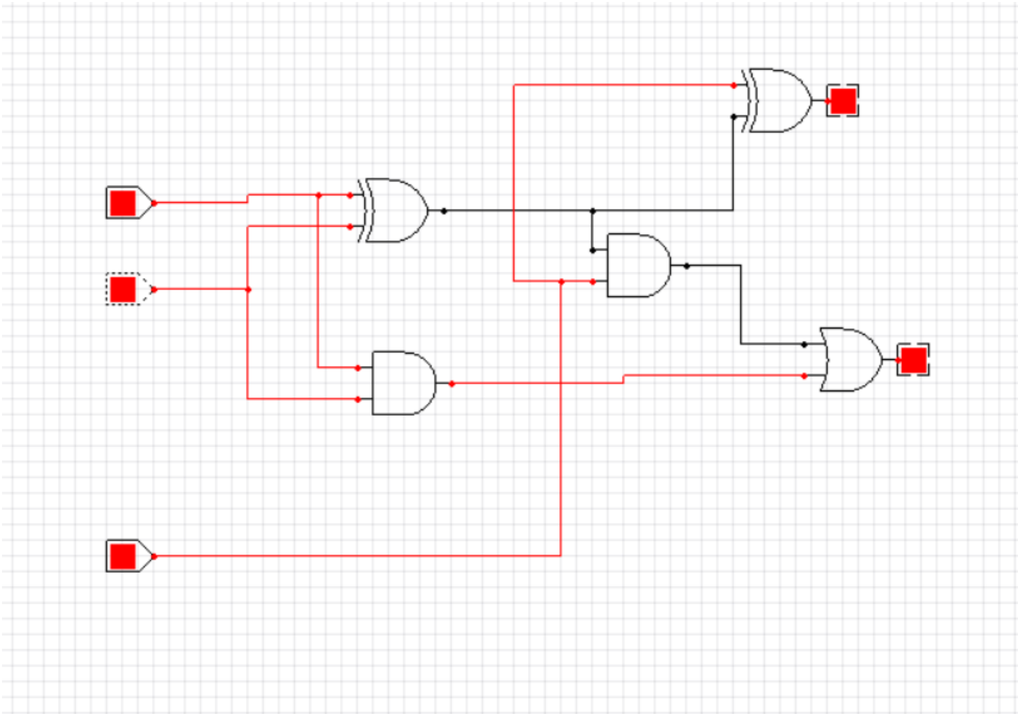
Sum =(A'.B'.Cin)+(A'.B.Cin')+(A.B'Cin')+(A.B.Cin)

= Cin(A'.B'+A.B)+Cin'(A'.B+A.B')

= Cin.1+Cin'(A$\oplus$B)

= $\underline{\text{Cin}\oplus\text{A}\oplus\text{B}}$

The logic circuit for the full adder,



Subtraction of two 2-bit numbers

The truth table for the subtraction of two numbers,

A	B	Bin	Bout	Sub
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$Bout = (A'.B'.Bin)+(A'.B'.Bin')+(A'.B.Bin)+(A.B.Bin)$$

A \ B.Bin	00	01	11	10
0	1	1	1	
1			1	

$$\boxed{Bout = A'.B'+B.Bin}$$

$$Sub = (A'.B'.Bin)+(A'.B.Bin')+(A.B'.Bin')+(A.B.Bin)$$

$$= Bin(A'.B'+A.B)+Bin'(A'.B+A.B')$$

$$= Bin.1+Bin'(A \oplus B)$$

$$\underline{\underline{= Bin \oplus A \oplus B}}$$

Addition of two 4-bit numbers

Below truth table show some examples for the addition of two 4-bit numbers

Cin	A3	A2	A1	A0	B3	B2	B1	B0	Sum3	Sum2	Sum1	Sum0	Cout
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	1	0	0	1	0	0
0	0	0	1	0	0	0	1	0	0	1	0	0	0
0	0	0	1	1	0	0	1	1	0	1	1	0	0
0	0	1	0	0	0	1	0	0	1	0	0	0	0
0	0	1	0	1	0	1	0	1	1	0	1	0	0
0	0	1	1	0	0	1	1	0	1	1	0	0	0
0	0	1	1	1	0	1	1	1	1	0	1	0	0
0	1	0	0	0	1	0	0	0	0	0	0	0	1
0	1	0	0	1	1	0	0	1	0	0	1	0	1
0	1	0	1	0	1	0	1	0	0	1	0	0	1
0	1	0	1	1	1	0	1	1	0	1	1	0	1
0	1	1	0	0	1	1	0	0	1	0	0	0	1
0	1	1	0	1	1	1	0	1	1	0	1	0	1
0	1	1	1	0	1	1	1	0	1	1	0	0	1
0	1	1	1	1	1	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1

Subtraction of two 4-bit numbers

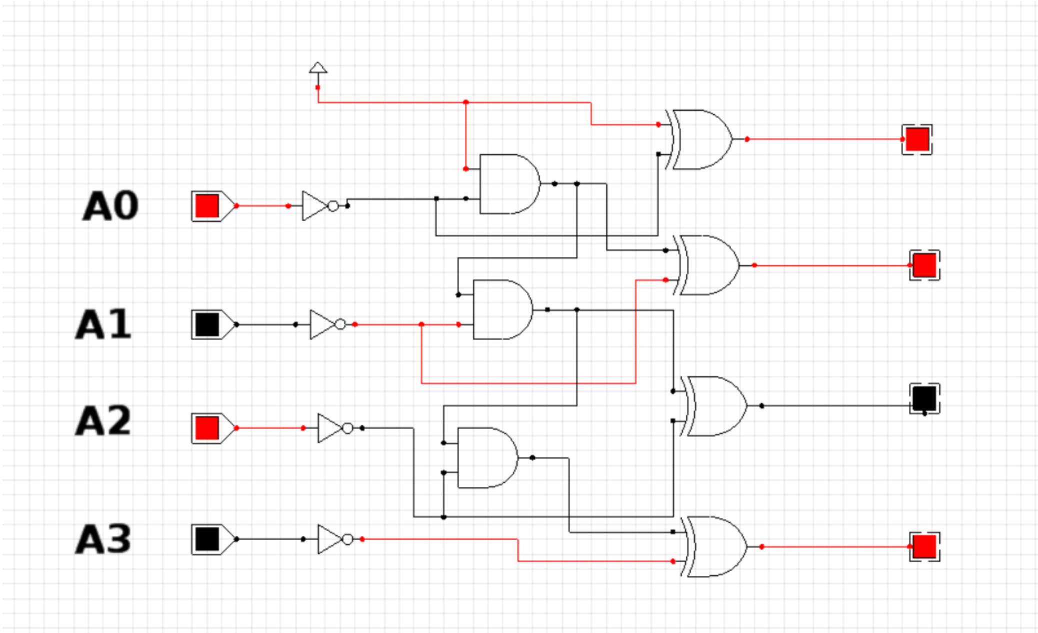
Truth table for a 4-bit number in 2's complement

To find out the 2's complement of a binary number first we can convert the number to the 1's complement.

And then we can add 1 to the least significant bit and convert the number in to the 2's complement.

B3	B2	B1	B0	L	M	N	O
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

Logic circuit for the 2’s complement



Without using this circuit, we can simply design a subtractor in the same circuit as the adder by using binary adder – subtractor as shown in the below circuit.

Binary adder - subtractor circuit

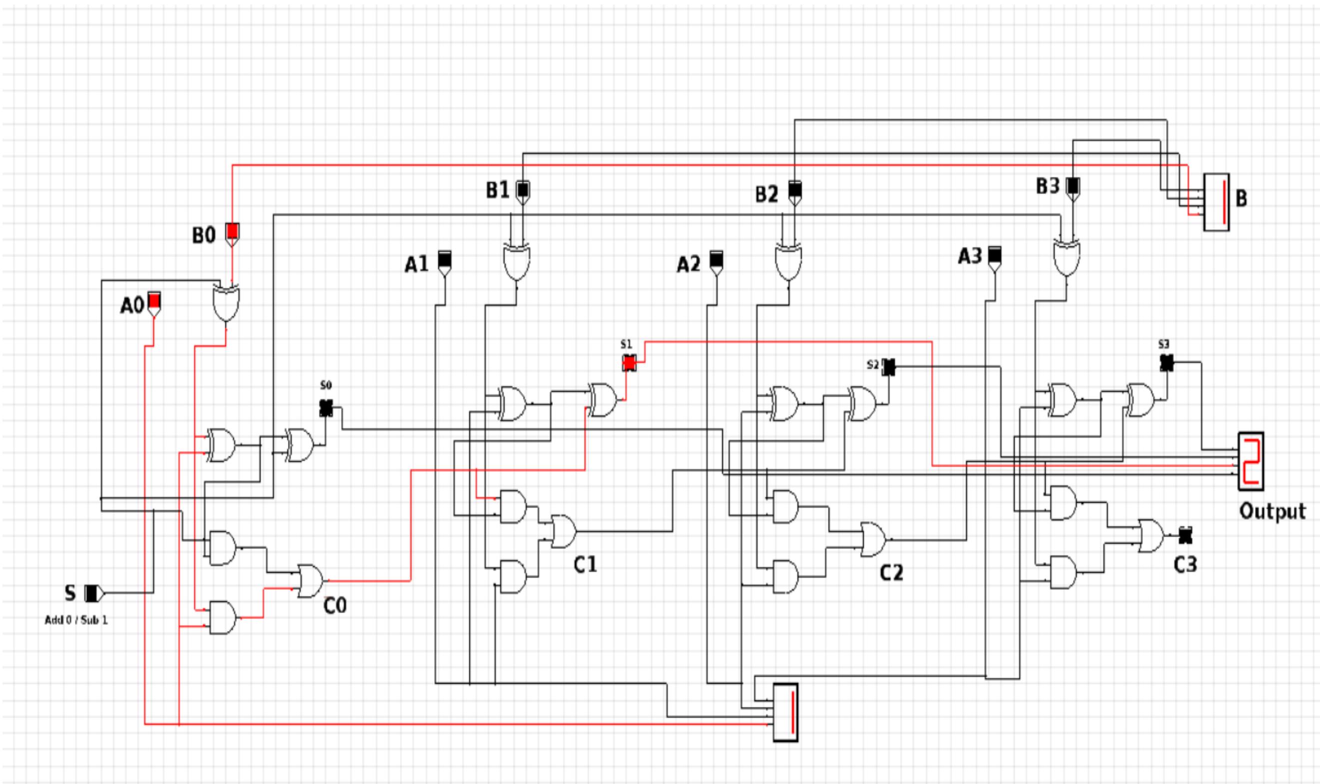
By using binary – subtractor circuit we can simply design both addition and subtraction in a single circuit.

We can control this operation by changing the value of the control signal line holds.

This control signal line holds the value of either 1 or 0 which determine what operation should be carried out.

If control signal line(S) = 1 the operation should be subtraction

If control signal line(S) = 0 the operation should be addition



As shown in this circuit, the control line directly connected to the C0 as the its input.

The value of the A0 is directly connected to the XOR gate in the L1 part.

The XOR of the B0 is connected to this XOR gate.

The outputs of C0 are the Sum and the Carry out.

When the value of the control line/ S=1, the output of the B0 would be B0’.

Operation would be (A+B0’).

Then the 2’s complement operation would be carried out.

02.) COMPARING TWO 4-BIT NUMBERS

Inputs: A (First 4-bit binary number)

A3, A2, A1, A0

B (Second 4-bit binary number)

B3, B2, B1, B0

Outputs: L -> when A>B

S -> when A<B

E -> when A=B

Truth table for comparing two numbers

A	B	A>B (L)	A<B (S)	A=B (E)
0	0	X	X	1
0	1	X	1	X
1	0	1	X	X
1	1	X	X	1

A>B(L)= A.B’

A<B(S)=A’.B

A=B(E) =A’B’+A.B

Compare two 4-bit numbers

Possibilities of comparison of two 4-bit numbers

A3B3	A2B2	A1B1	A0B0	A>B (L)	A<B (S)	A=B (E)
A3>B3	**X**	**X**	**X**	1	0	0
A3<B3	**X**	**X**	**X**	0	1	0
A3=B3	A2>B2	**X**	**X**	1	0	0
A3=B3	A2<B2	**X**	**X**	0	1	0
A3=B3	A2=B2	A1>B1	**X**	1	0	0
A3=B3	A2=B2	A1<B1	**X**	0	1	0
A3=B3	A2=B2	A1=B1	A0>B0	1	0	0
A3=B3	A2=B2	A1=B1	A0<B0	0	1	0
A3=B3	A2=B2	A1=B1	A0=B0	0	0	1

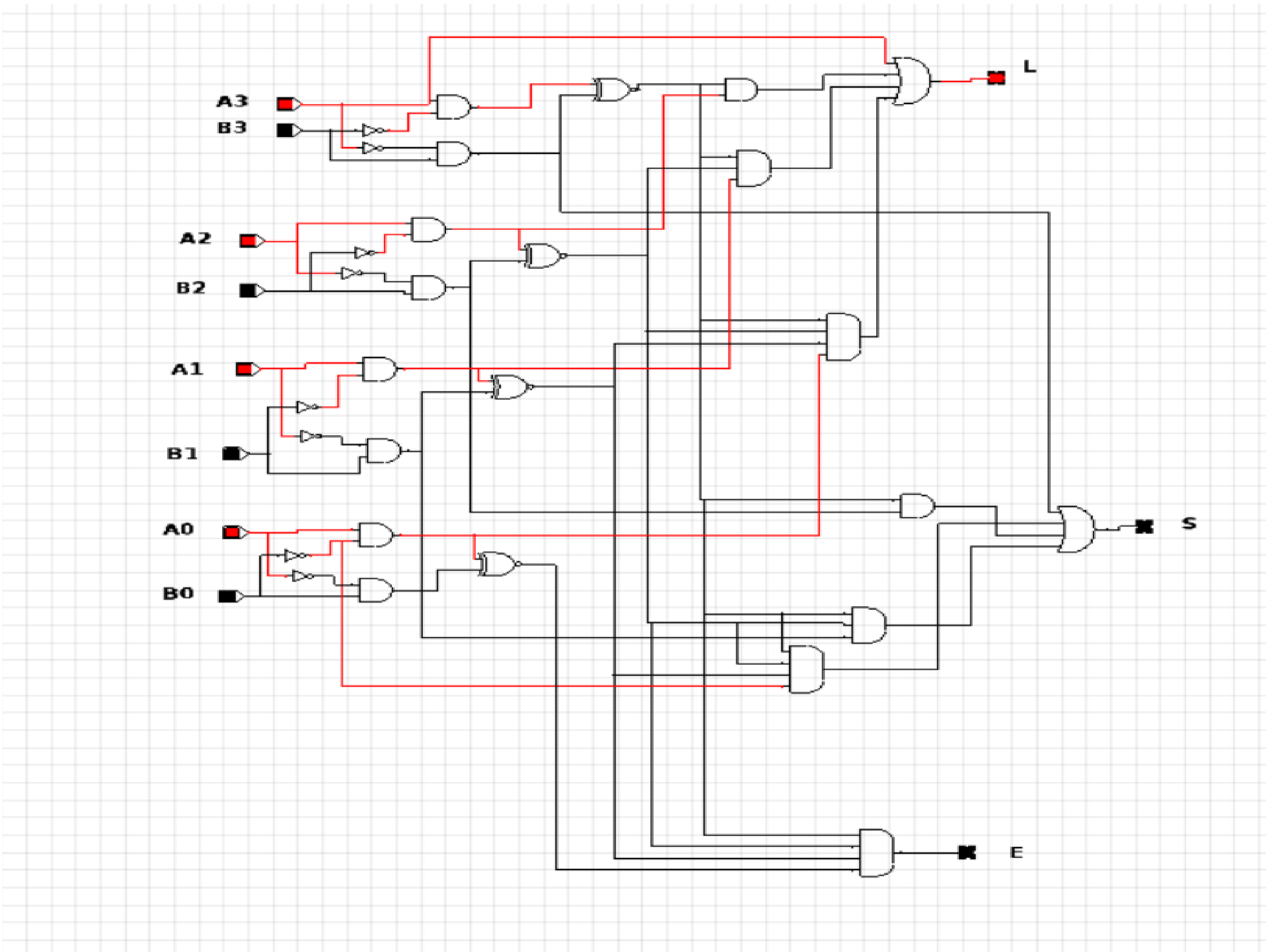
Here first we compare the most significant bit of the two 4-bit numbers.

If and only if those first most significant bits are equal, we have to compare next two most significant bits.

If most significant bit of A is greater than the most significant bit of B, then A is greater than B.

Otherwise, B is greater than A.

The circuit for comparing two 4-bit numbers,

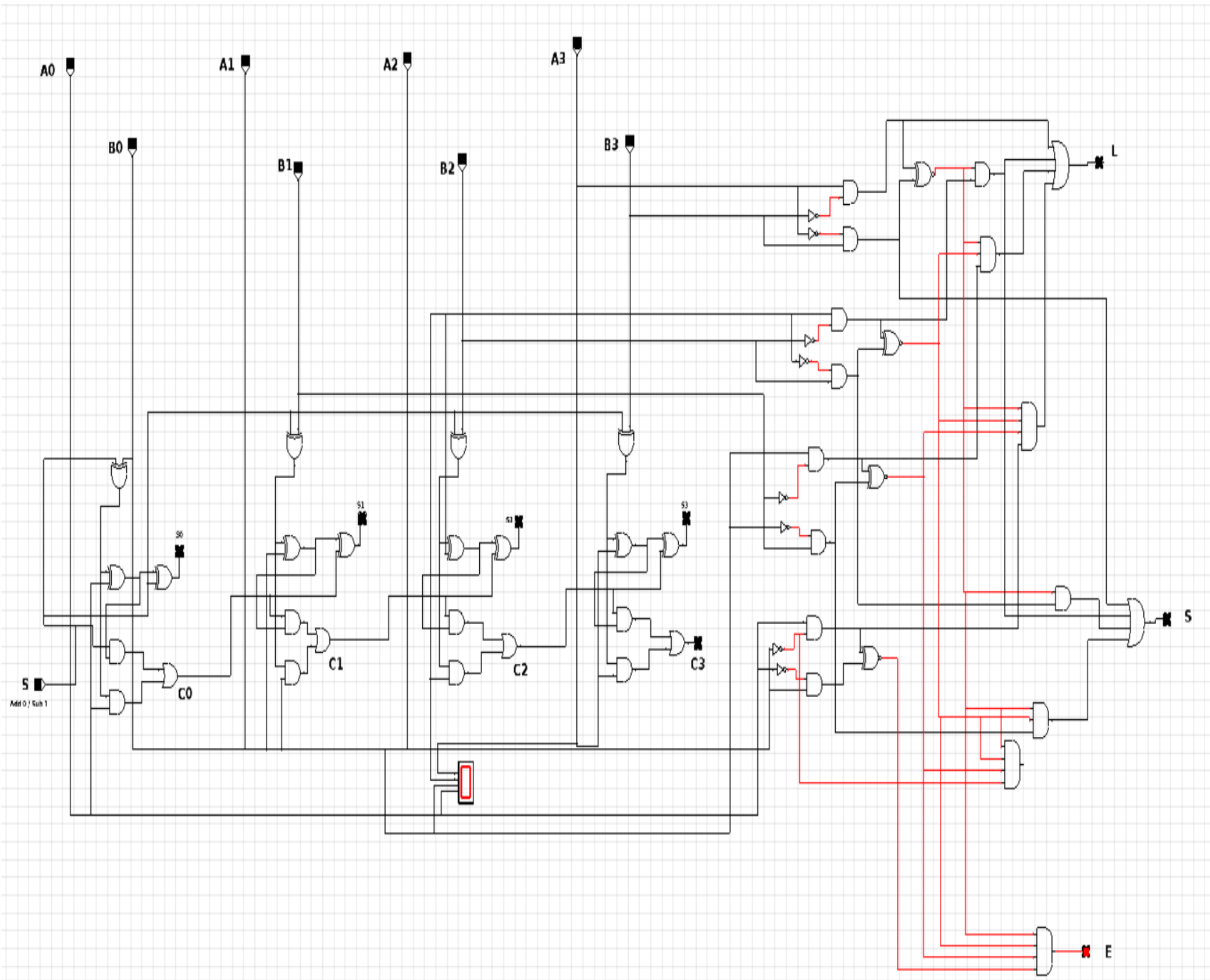


In here we have used eight inputs as A0,B0, A1,B1, A2,B2, A3,B3 and three inputs as E, S, L to compare two 4-bit numbers to generate equal to, smaller than, larger than between two numbers.

Here AND gates, OR gates, NOT gates, XOR gates are used to design the circuit.

03.) COMPLETE TWO 4-BIT ALU CIRCUIT

The below circuit shows the addition/ subtraction, comparing of two 4-bit numbers.



References

- Schaums Outlines-Digital Principles 3rd Edition
- 4-bit binary Adder-Subtractor - GeeksforGeeks
- Magnitude Comparator in Digital Logic - GeeksforGeeks
- https://youtu.be/1I5ZMmrOfnA