Implementation of 4-bit Magnitude Comparator

Objective:

- Draw the circuit that will act as a Magnitude Comparator. Your circuit should be able to compare two 4 bits number.
- Implement your circuit (for two 4-bit numbers)

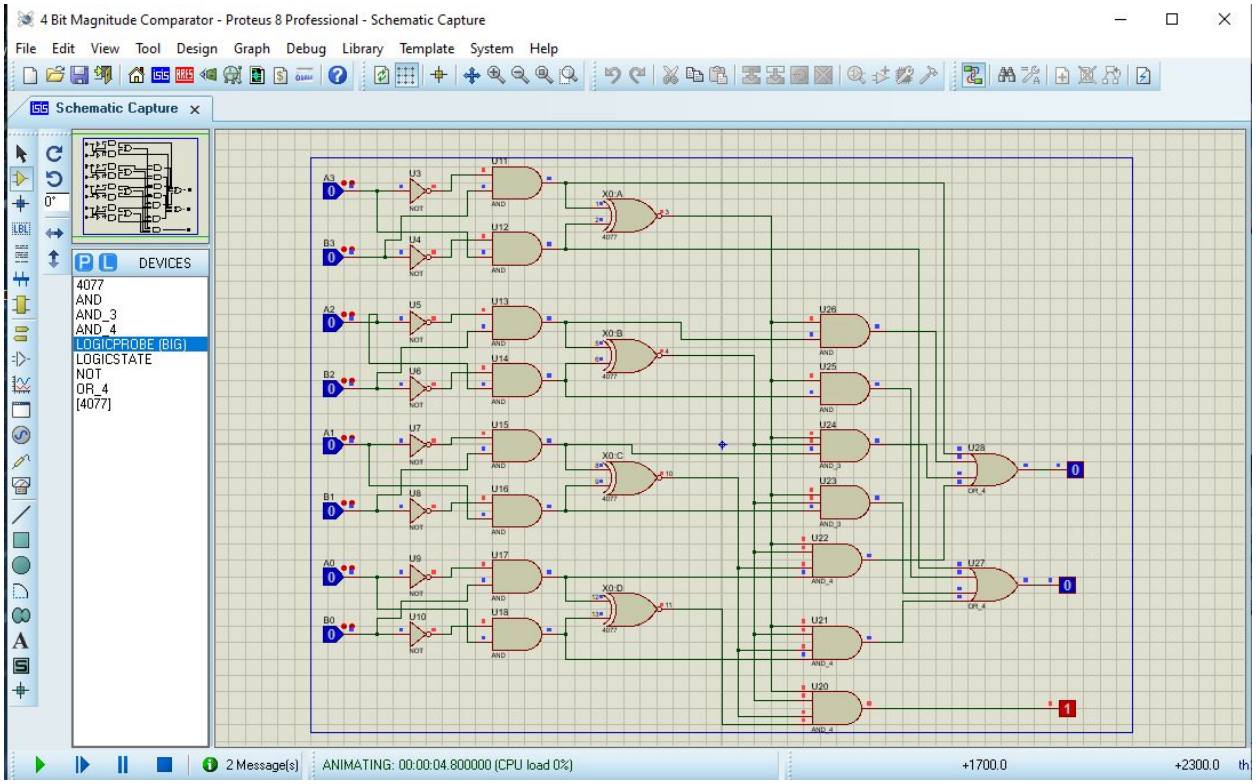
Required Components and Equipments:

- Logic State
- Logic Probe(Big)
- Not Gate, And 2/3/4 Gate, OR 4 Gate, XNOR(4077) Gate

Experimental Setup (No need to draw the IC configurations):

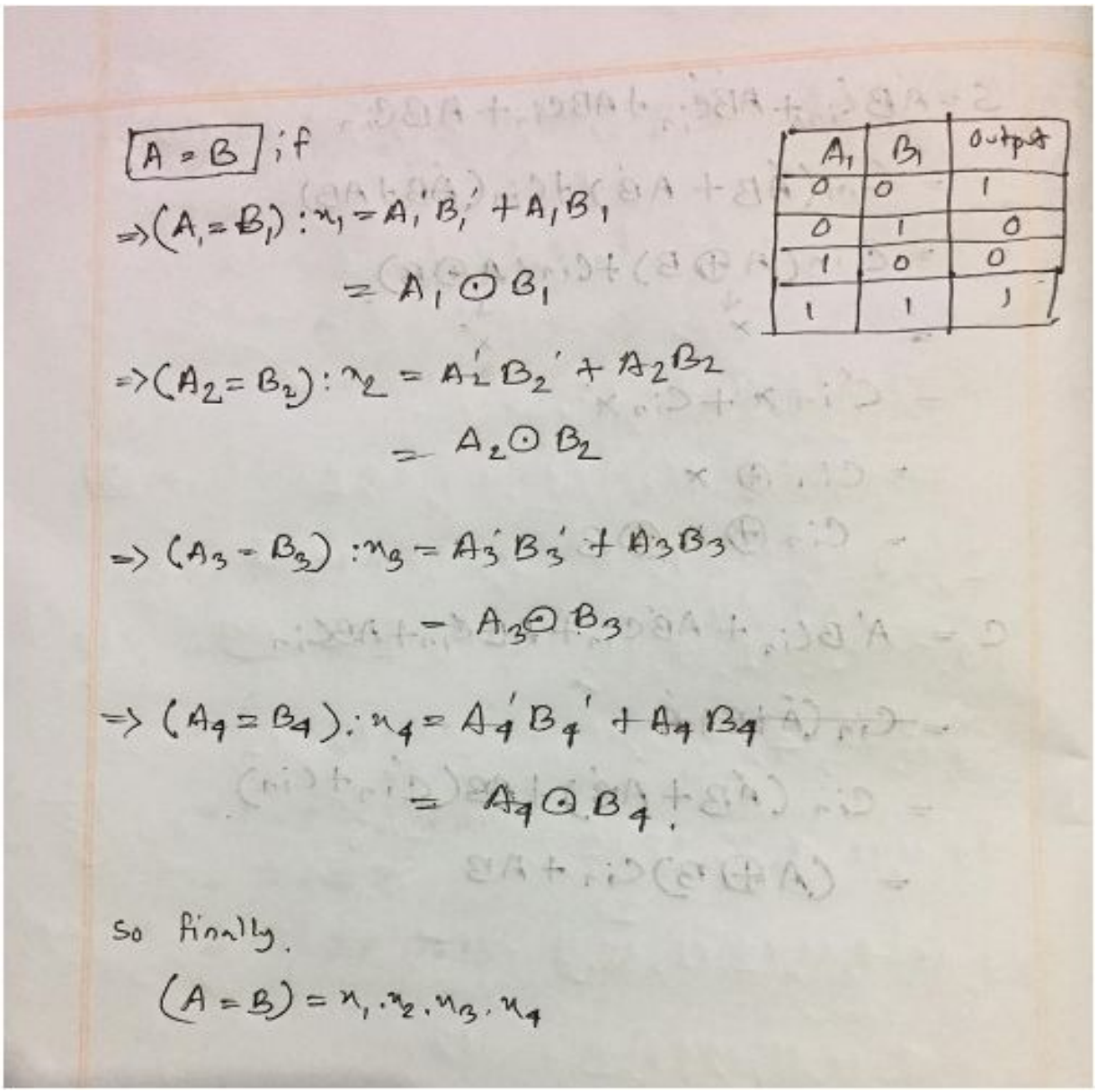
- From parts, I picked Not Gate, And 2-input/3-input/4-input Gate, OR 4-input Gate, XNOR(4077) Gate, Logic state, Logic Probe(Big).
- After picking out the components I placed 8 logic states for A and B. Named them A0, B0, A1, B1, A2, B2, A3, B3.
- Also, I placed 8 Not gates to make A' and B'.
- Placed 8 And Gates. For A0, B0 pair I added A0' with B0 by an AND gate. And B0' with A0 with another AND gate. I followed this same procedure for the other 3 pair and got U11, U12, U13, U14, U15, U16.
- For A0, B0 I have two and gates. U17 and u18. I added them with an XOR(4077) Gate which is X0-D. And also did the same for the other three pairs and got X0-C, X0-B, X0-A.
- Then I added three 4-input AND gate(U20, U21, U22), two 3-input AND gate(U23, U24), and two basic AND gate(U25, U26).
- I added U26 with X0-A and U13. U25 with X0-A and U14.
- U24 with X0-A, X0-B, and U15. U23 with X0-A, X0-B, and U16.
- U22 with X0-A, X0-B, X0-C, and U17.
- U21 with X0-A, X0-B, X0-C, and U18.
- U20 with X0-A, X0-B, X0-C, and X0-D.
- Then I placed two 4-input OR gate(U27, U28).
- I added U27 with U12, U25, U23, and U21.
- I added U28 with U11, U26, U24, and U22.
- Finally, I added three logic probes with U20, U27, U28 to see my expected output.
- U20 indicates if A and B are equal. A=B.
- U28 indicates if B>A.
- U27 indicates if A>B.

Proteus ScreenShot:



Results and Discussions:

A=B:



A = B ; if

$\Rightarrow (A_1 = B_1) : x_1 = A_1' B_1' + A_1 B_1$

$= A_1 \odot B_1$

A_1	B_1	output
0	0	1
0	1	0
1	0	0
1	1	1

$\Rightarrow (A_2 = B_2) : x_2 = A_2' B_2' + A_2 B_2$

$= A_2 \odot B_2$

$\Rightarrow (A_3 = B_3) : x_3 = A_3' B_3' + A_3 B_3$

$= A_3 \odot B_3$

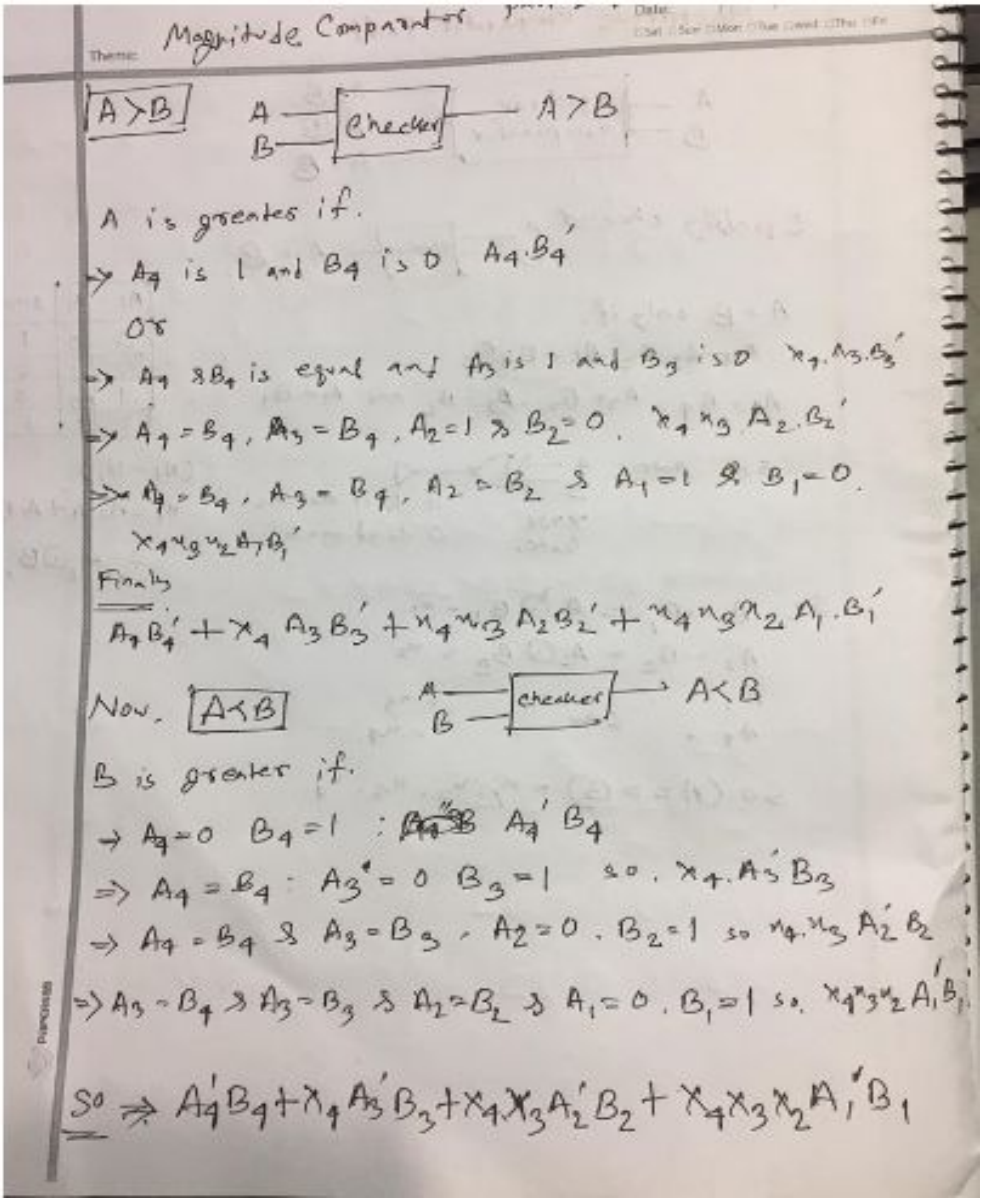
$\Rightarrow (A_4 = B_4) : x_4 = A_4' B_4' + A_4 B_4$

$= A_4 \odot B_4$;

So finally,

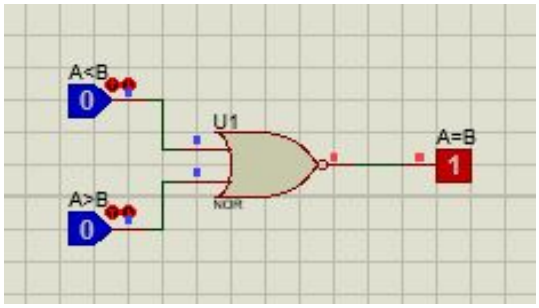
$(A = B) = x_1 . x_2 . x_3 . x_4$

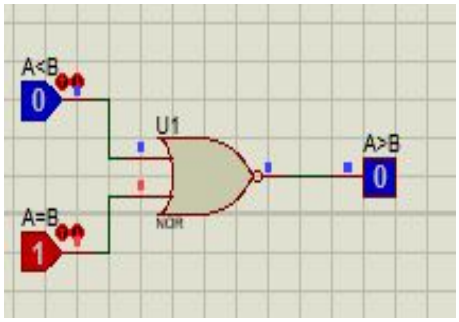
A>B and A<B:

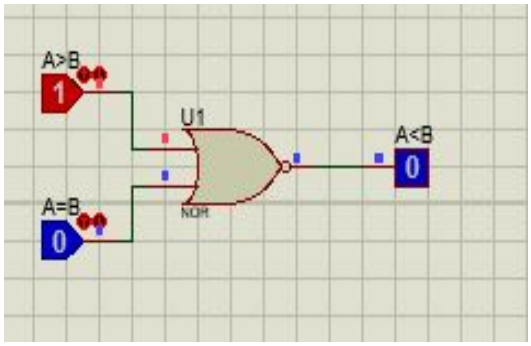


What changes have to made in your design to find the third result from any two of the three results (A = B, A >B, A< B):

To find out the third value, we can simply construct a NOR gate. As we know two of the value, we feed the values into the 2 input NOR gate. These are the three types of situations we will face.


A<B
0
A>B
0
U1
NOR
A=B
1


A<B
0
A=B
1
U1
NOR
A>B
0


A>B
1
A=B
0
U1
NOR
A<B
0