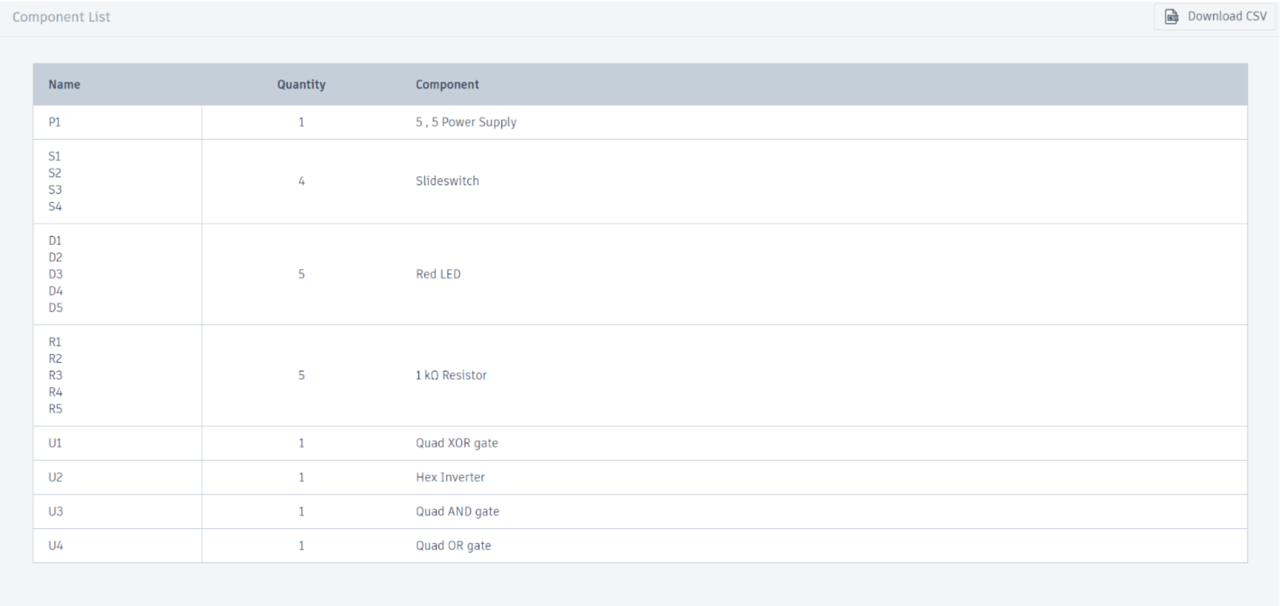
Magnitude Comparator

Aim: To perform magnitude comparator on the breadboard

Components:



Component List

Download CSV

Name	Quantity	Component
P1	1	5 , 5 Power Supply
S1 S2 S3 S4	4	Slideswitch
D1 D2 D3 D4 D5	5	Red LED
R1 R2 R3 R4 R5	5	1 kΩ Resistor
U1	1	Quad XOR gate
U2	1	Hex Inverter
U3	1	Quad AND gate
U4	1	Quad OR gate

Procedure:

Step 1: Understand the Concept

A 4-bit magnitude comparator compares two 4-bit binary numbers (A and B) and provides three outputs:

- A > B
- A < B
- A = B

Step 2: Log in to Tinker cad

1. Go to the Tinker cad website and log in to your account.
2. Open the "Circuits" workspace from the dashboard.

Step 3: Create a New Circuit

1. Click on "Create new Circuit."
2. Name your circuit project, for example, "4-bit Magnitude Comparator."

Step 4: Add Components

1. **Breadboard**: Add a breadboard to organize your circuit neatly.
2. **Logic Gates**: Add the necessary logic gates (AND, OR, NOT, XOR) from the component’s library.
3. **Switches**: Add 8 switches (4 for each binary number A and B).
4. **LEDs**: Add 3 LEDs to display the results (A > B, A < B, A = B).
5. **Resistors**: Add resistors for the LEDs (typically 220 ohms).

6. **Wires**: Use wires to connect components.

Step 5: Design the Comparator Circuit

Step 5.1: Comparator Logic

The 4-bit magnitude comparator can be implemented using combinational logic circuits. The equations for the outputs are:

- A=BA = BA=B: The bits of A are equal to the bits of B.
- A>BA > BA>B: The most significant bit of A is greater than the most significant bit of B, or if the bits are equal, the next significant bit is compared, and so on.
- A<BA < BA<B: The most significant bit of A is less than the most significant bit of B, or if the bits are equal, the next significant bit is compared, and so on.

Step 5.2: Equality Comparison (A = B)

1. Use XNOR gates to compare each bit of A and B.
2. Combine the outputs of the XNOR gates using an AND gate to get the final A=BA = BA=B output.

Step 5.3: Greater Than Comparison (A > B)

1. Use a combination of AND, OR, and NOT gates to compare the bits.
2. Start from the most significant bit and move to the least significant bit. Use logic to check if A is greater than B at any bit position while all more significant bits are equal.

Step 5.4: Less Than Comparison (A < B)

1. Similar to the greater than comparison, but check if B is greater than A.

Step 6: Connect Components

1. **Switches to Logic Gates**: Connect the switches representing A and B to the respective inputs of the XNOR, AND, OR, and NOT gates.
2. **Logic Gates to LEDs**: Connect the outputs of the logic gates to the LEDs to display the results.
3. **Power and Ground**: Ensure all components are connected to power (Vcc) and ground (GND).

Step 7: Simulation

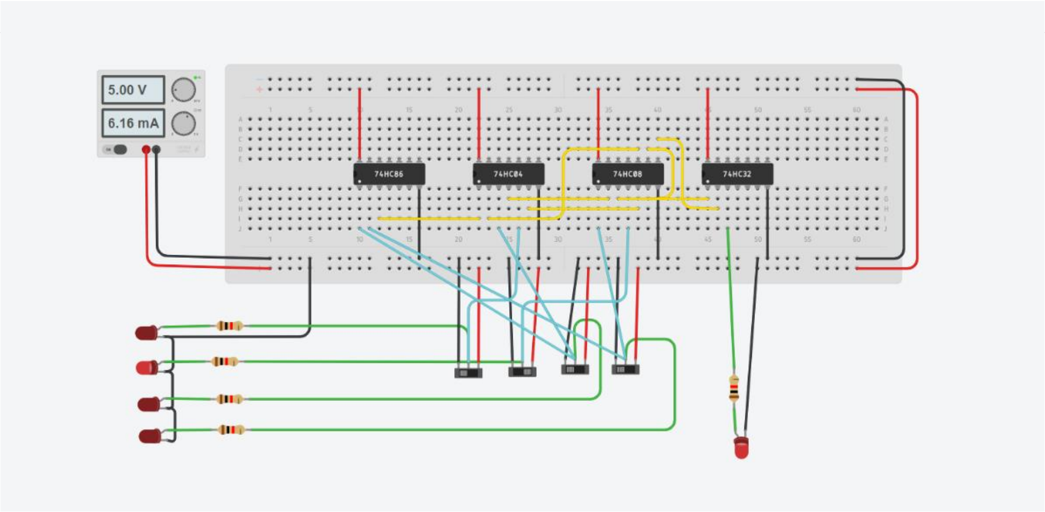
1. After connecting all components, start the simulation.
2. Toggle the switches to provide different 4-bit values for A and B.
3. Observe the LEDs to check if the circuit correctly indicates A > B, A < B, or A = B.

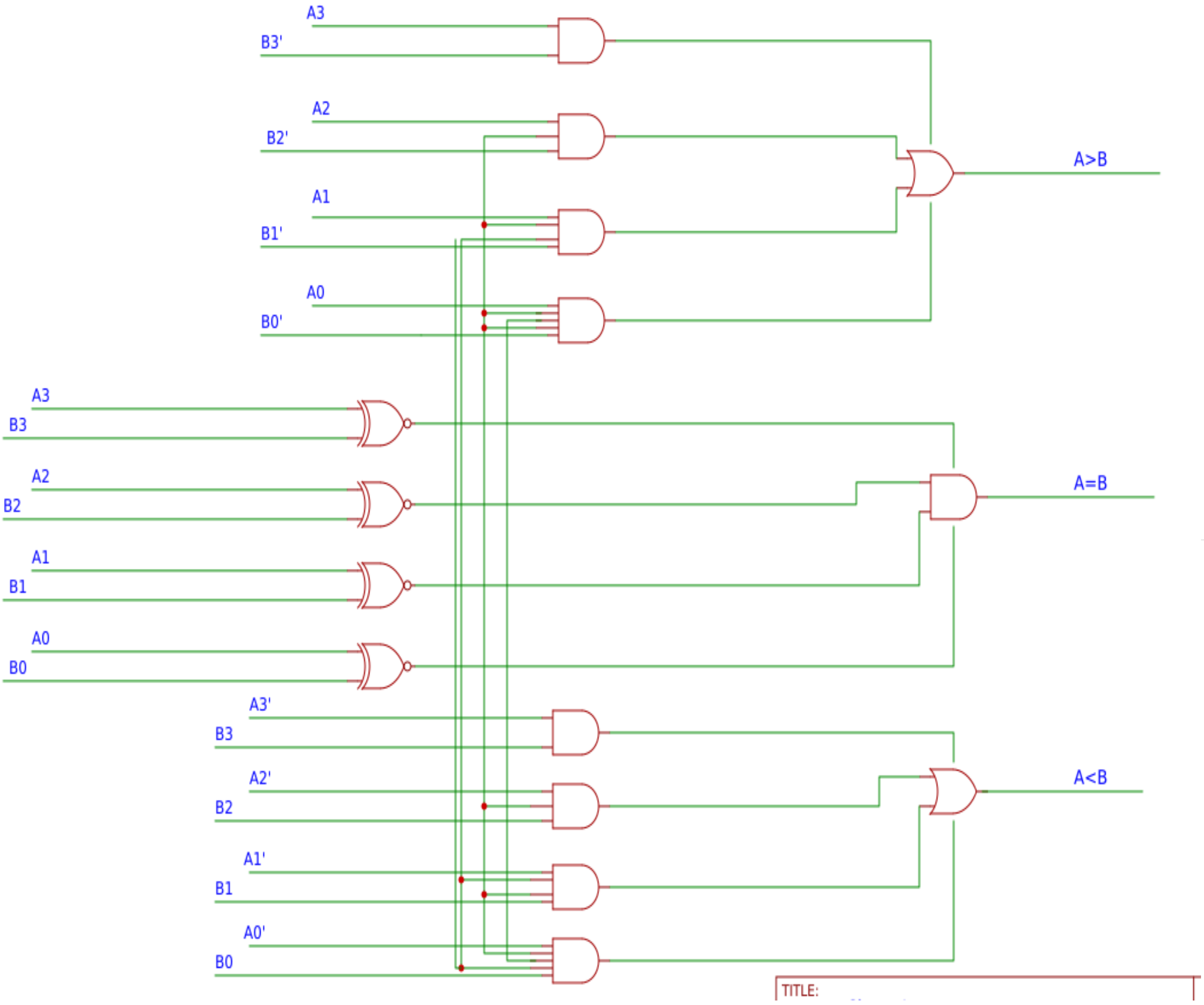
Step 8: Troubleshooting

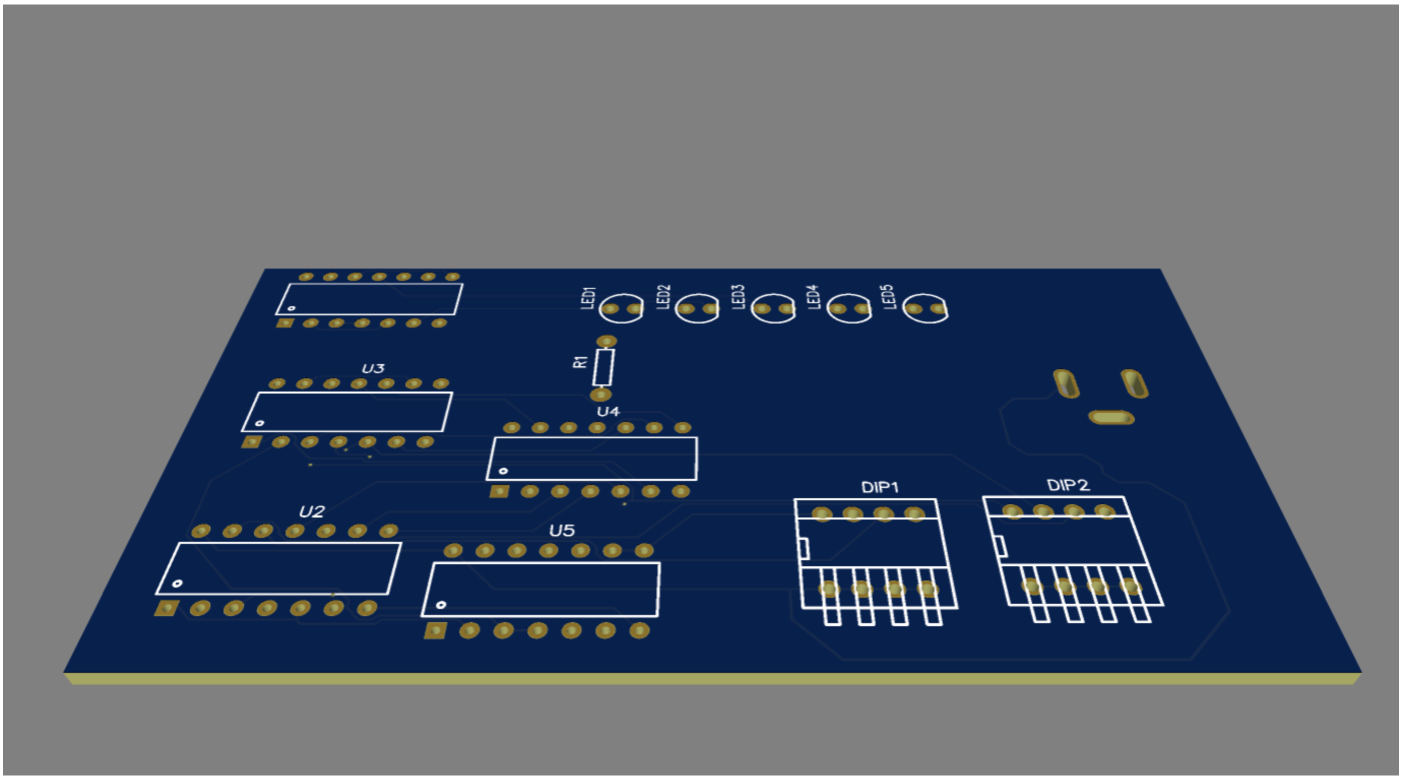
If the circuit is not working as expected:

1. Check all connections to ensure they are correct.
2. Verify the logic gates' inputs and outputs.
3. Make sure there are no short circuits or loose connections.

Schematic Circuit Diagram:







Result:

Setting up and programming a magnitude comparator with an Arduino is a straightforward and rewarding project that enhances the understanding of hardware and software integration.

TAMMALI KARTHIK: BU21EECE0100164

MARIA PUNYA: BU21EECE0100309

ALLI GOPI: BU21EECE0100161