LAB 3
Roll No.: 200010003

Aim: Designing logic gates using CMOS (Complementary Metal Oxide Semiconductor) logic.

Summary of the experiment: Study of logic gates using CMOS logic (Digital ICs), verifying their functionality and rigging up of circuits.

Components used: HCC4007UB IC, 1K ohm resistor array, DIP switches, LED displays, breadboard, power supply.

Design Procedure and Circuit Diagram: ***(attached below)***

Results and Discussions:

1. We have constructed the required logic gates (AND, OR, XOR, NAND, NOR, XNOR) using CMOS logic, with the help of the HCC4007UB IC. We verified the functionality of the constructed circuit using truth tables of the logic gates.
2. With the use of the CMOS logic family, we have greatly reduced the circuit complexity, and the density of logic functions per chip.

Conclusion:

We constructed logic circuits for the basic (AND, OR), universal (NAND, NOR), and special (XOR, XNOR) logic gates using the CMOS logic family, while verifying their functionality via truth table.

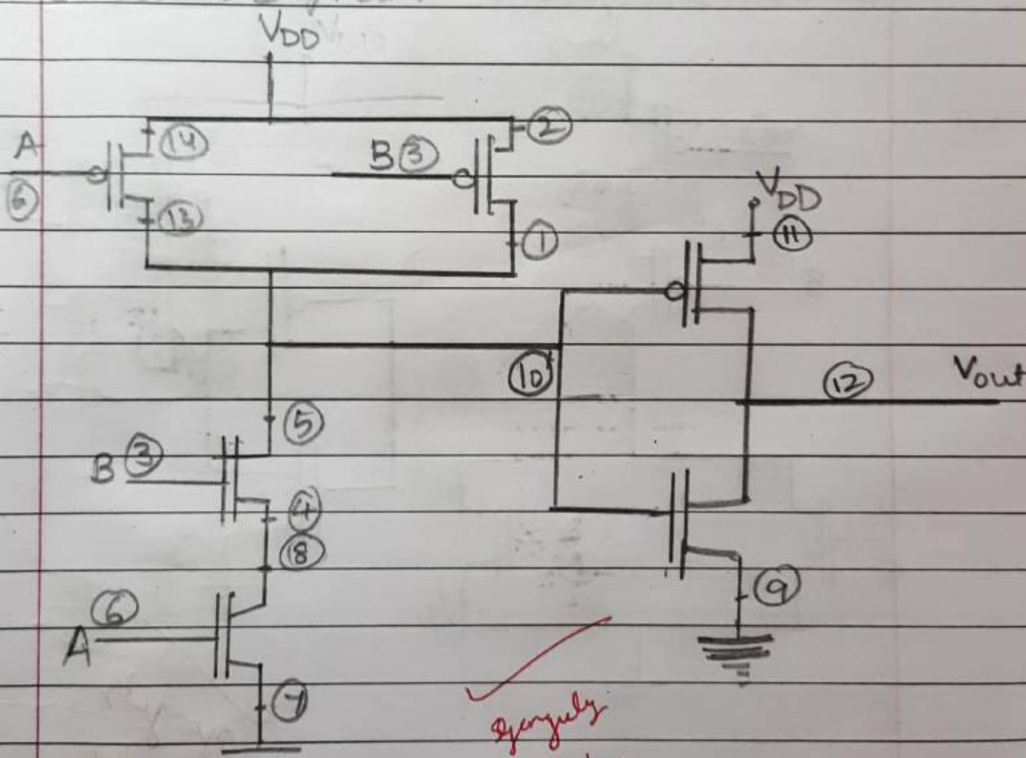
Design Procedure (Truth Table) and Circuit Diagrams

(1) AND

→ Truth Table

X	Y	X·Y
0	0	0
0	1	0
1	0	0
1	1	1

→ Circuit Diagram



31/12/24

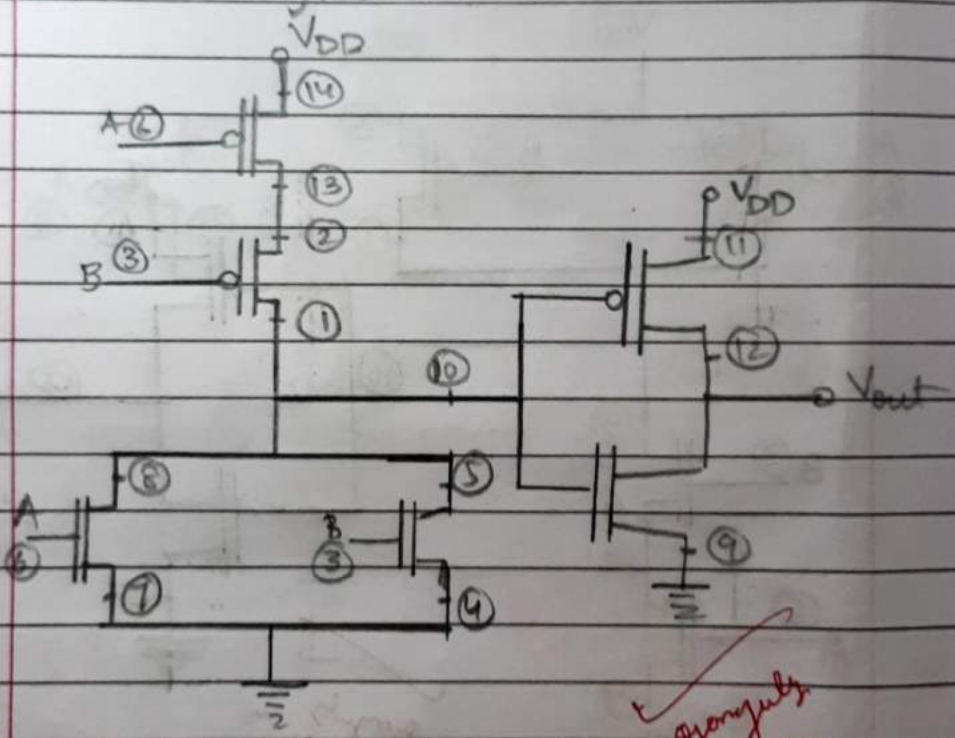
Page No.
Date / /

(2) OR

→ Truth Table

X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1

→ Circuit Diagram



Siyonguly
31/12/21

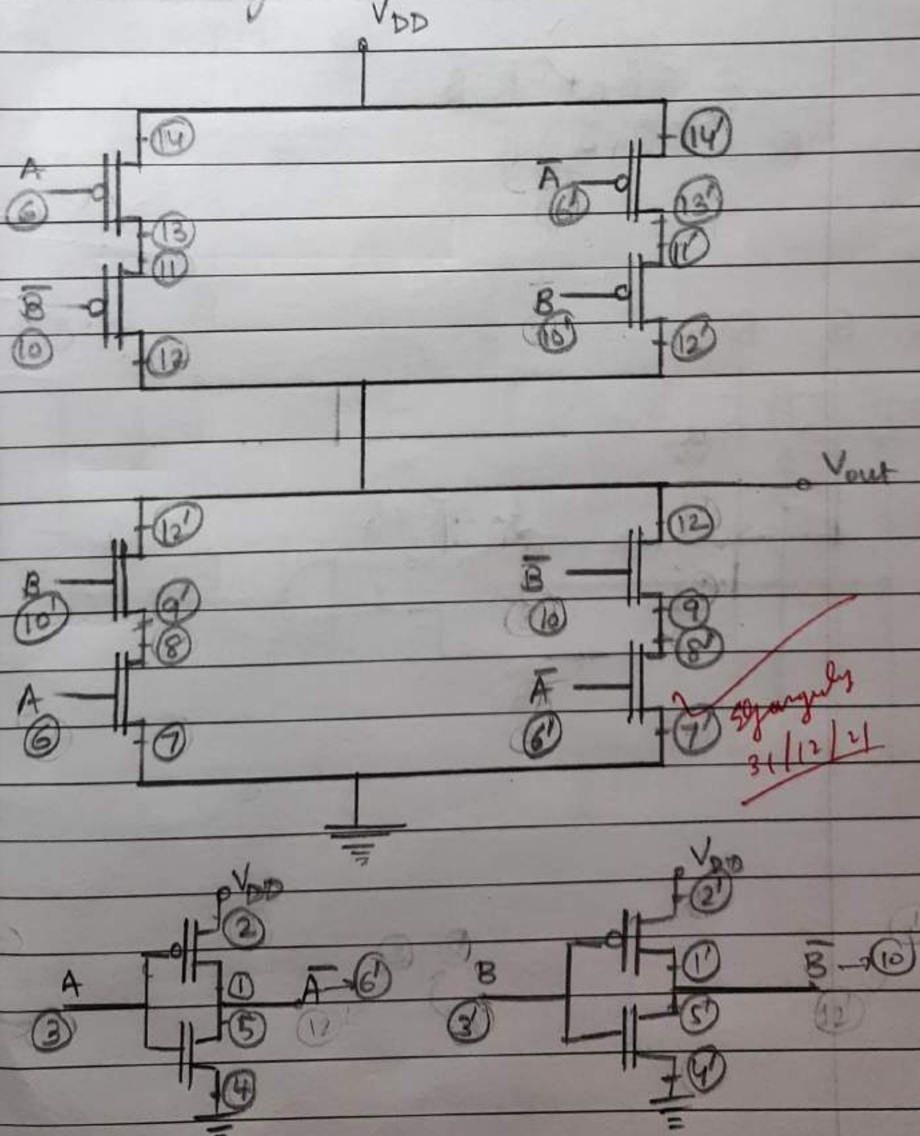
② XOR

→ Truth Table

X	Y	X ⊕ Y
0	0	0
0	1	1
1	0	1
1	1	0

$X \oplus Y = \overline{X}Y + X\overline{Y}$

→ Circuit Diagram



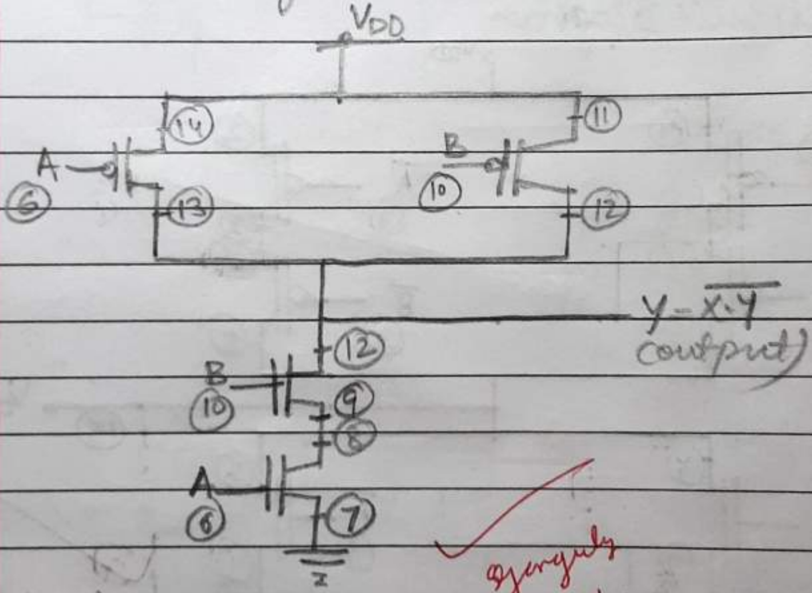
Page No.
Date

(4) NAND

→ Truth Table

X	Y	$\overline{X \cdot Y}$
0	0	1
0	1	1
1	0	1
1	1	0

→ Circuit Diagram



31/12/21

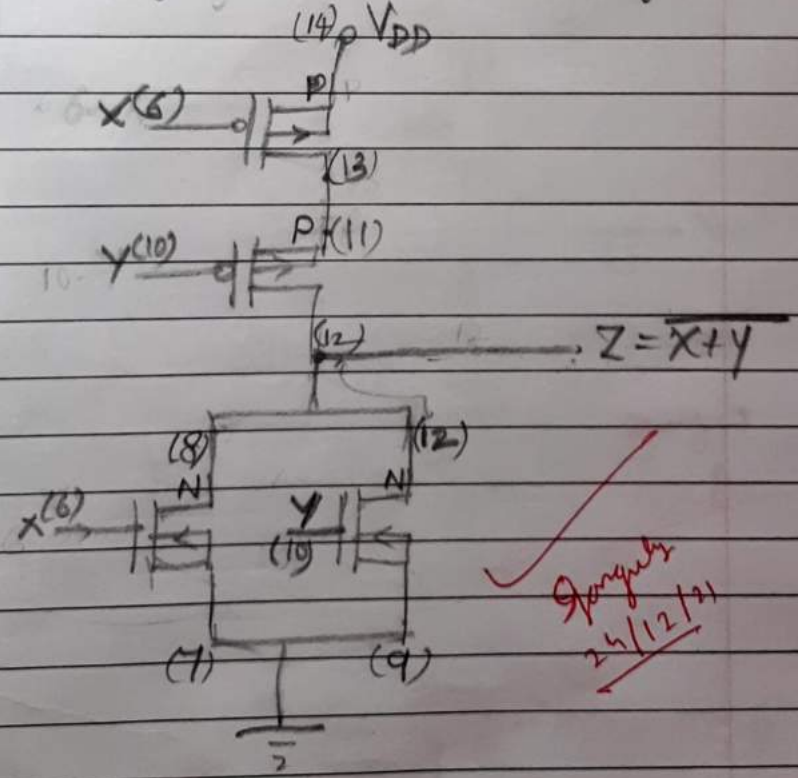
Page No.
Date

(5) NOR

→ Truth table

X	Y	$\overline{X+Y}$ = (X NOR Y)
0	0	1
0	1	0
1	0	0
1	1	0

→ Circuit Diagram (CMOS) IC (using HCF4007)



Gangadhar 24/12/21

X → pin 6
Y → pin 10

Page No.
Date

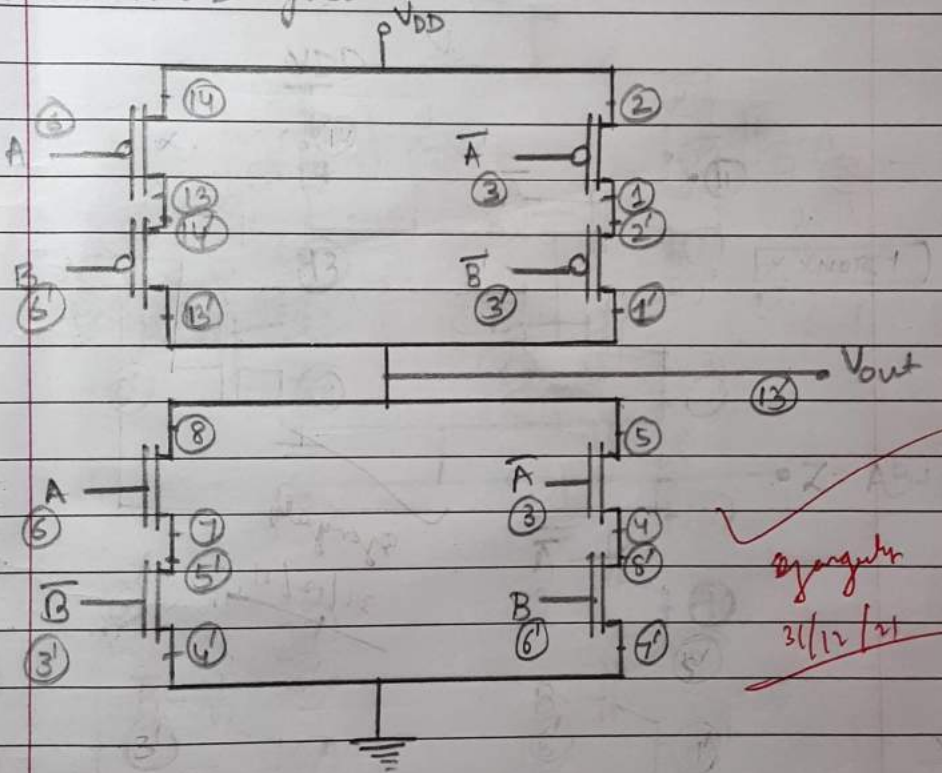
(6) XNOR

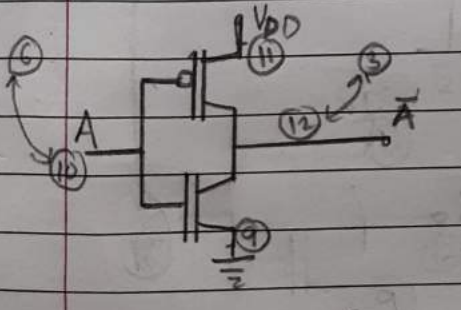
→ Truth Table

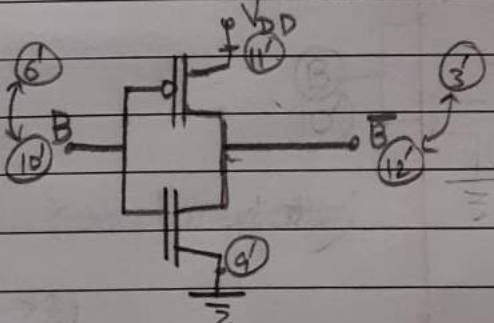
X	Y	$X \odot Y$ = X XNOR Y
0	0	1
0	1	0
1	0	0
1	1	1

$(X \odot Y = XY + \bar{X}\bar{Y})$

→ Circuit Diagram

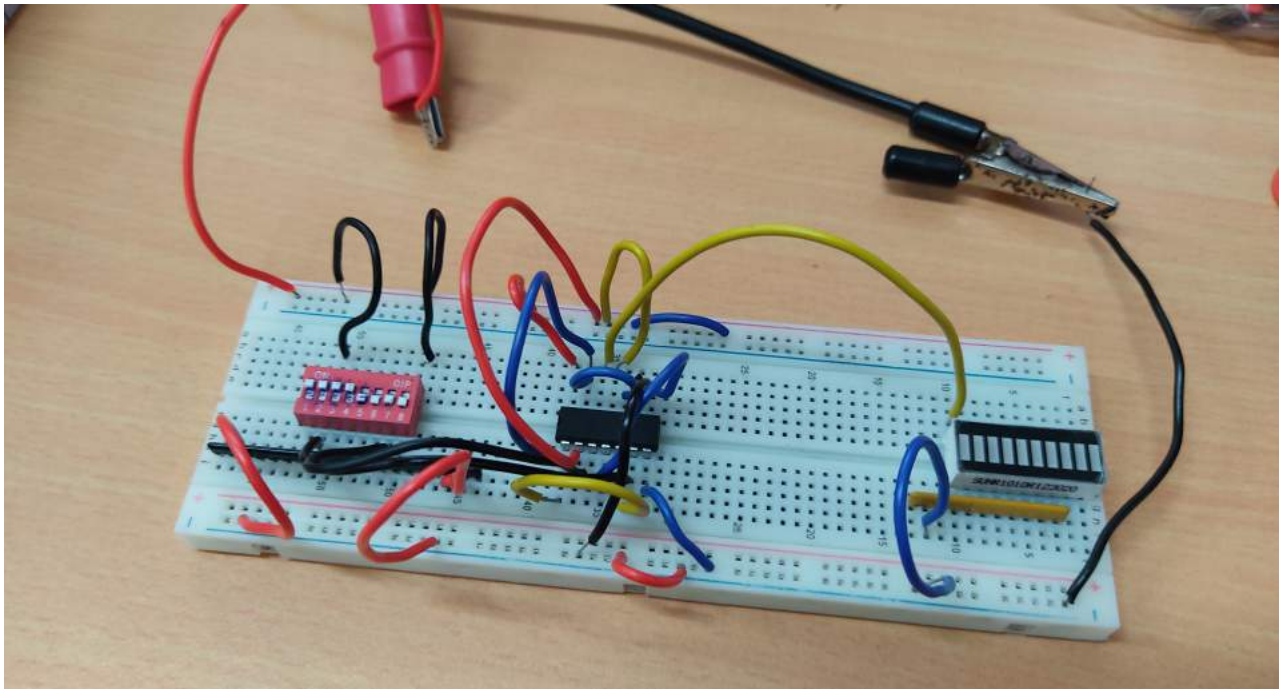




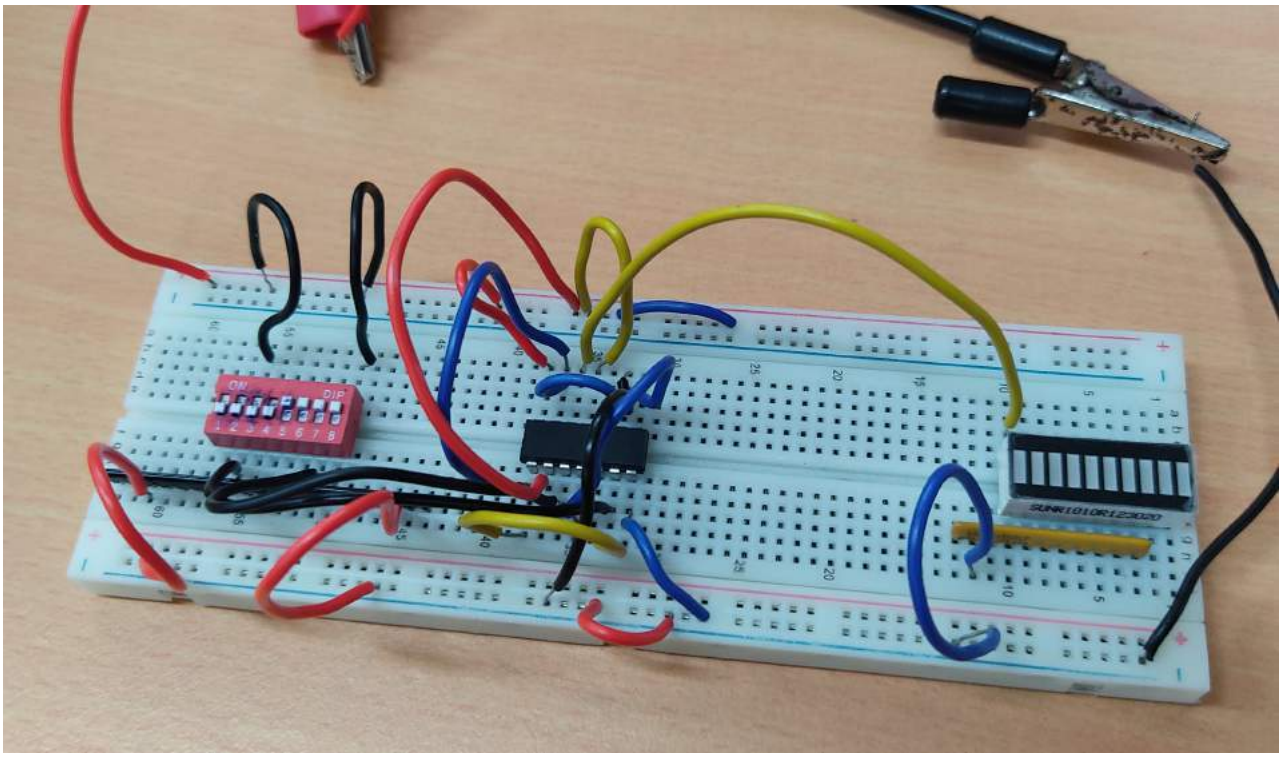


Circuit Snapshots:

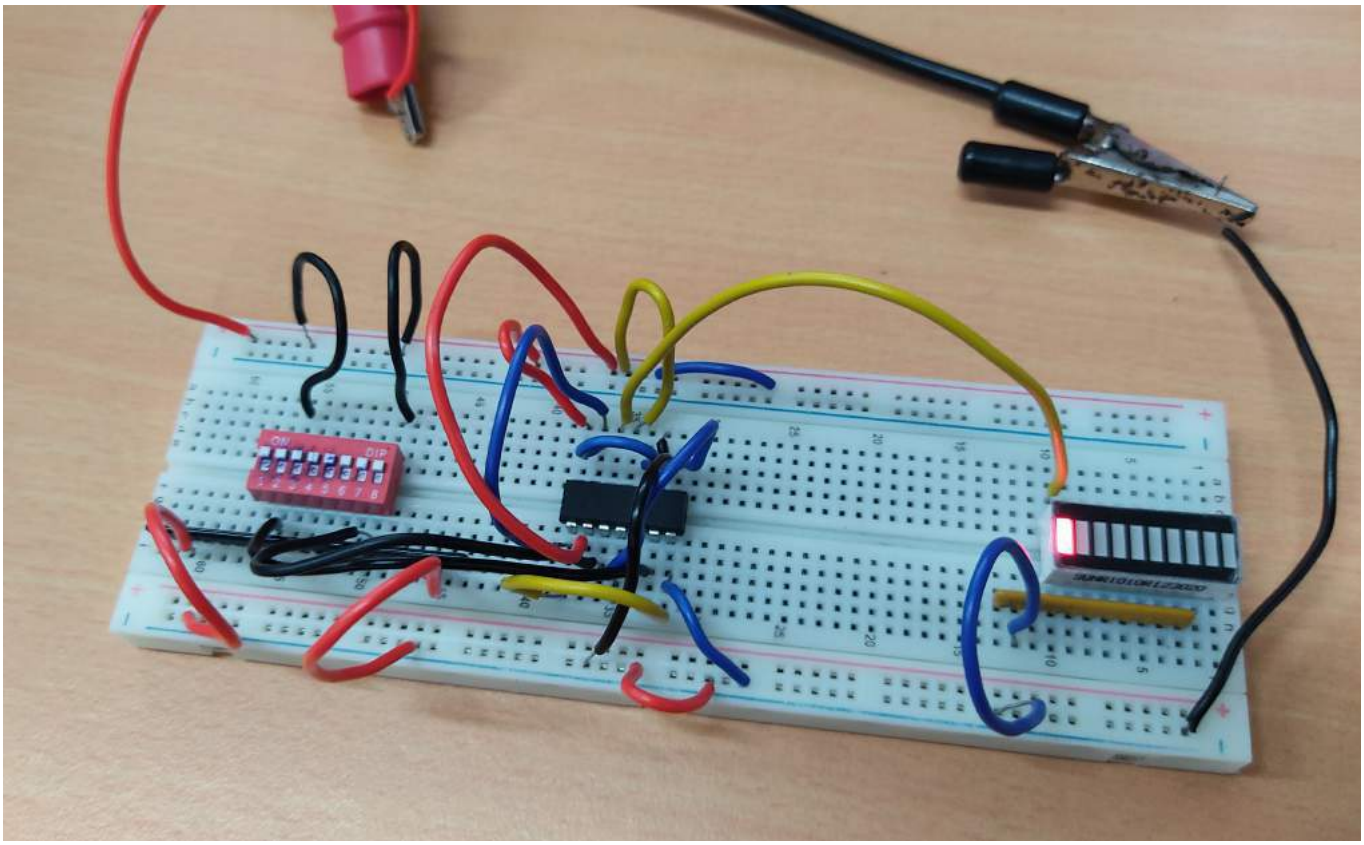
AND:



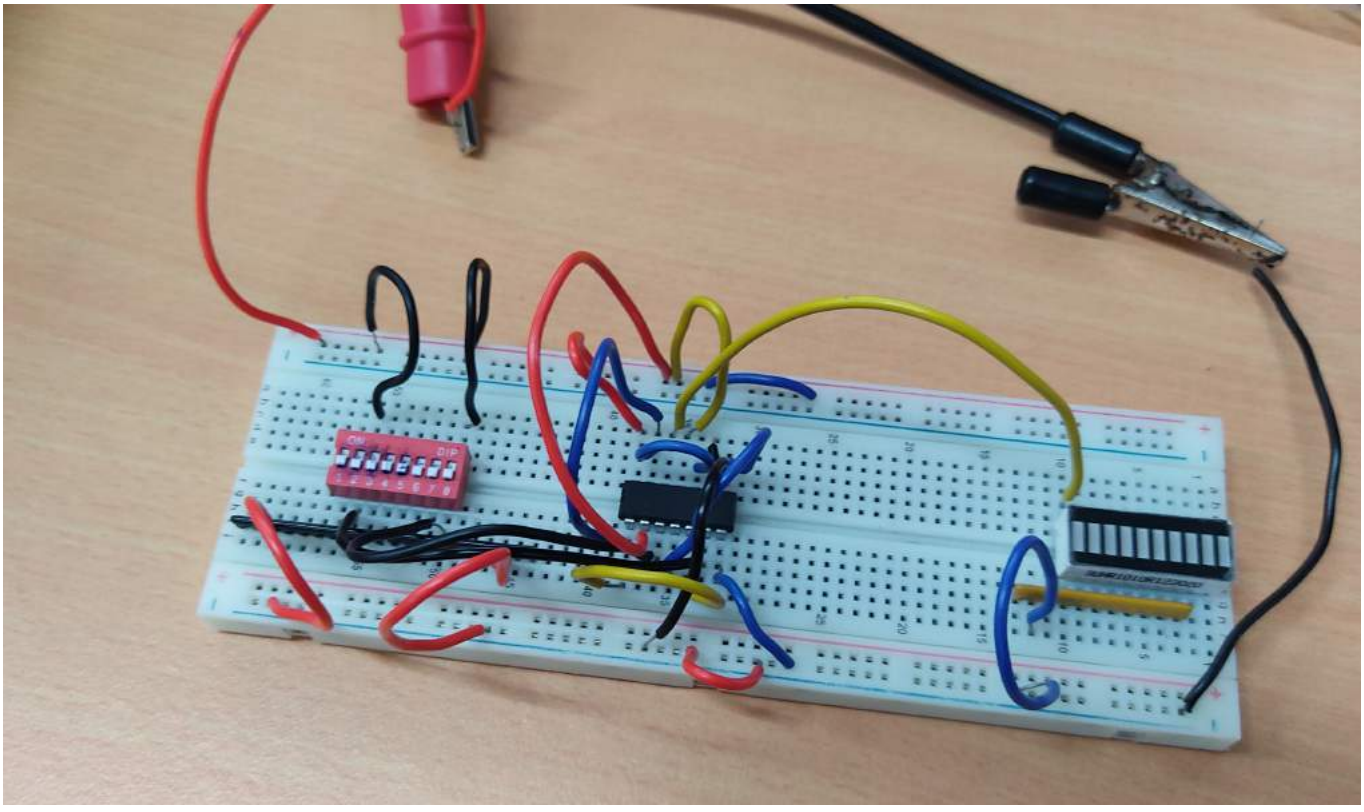
1_0 —> 0



0_1 —> 0

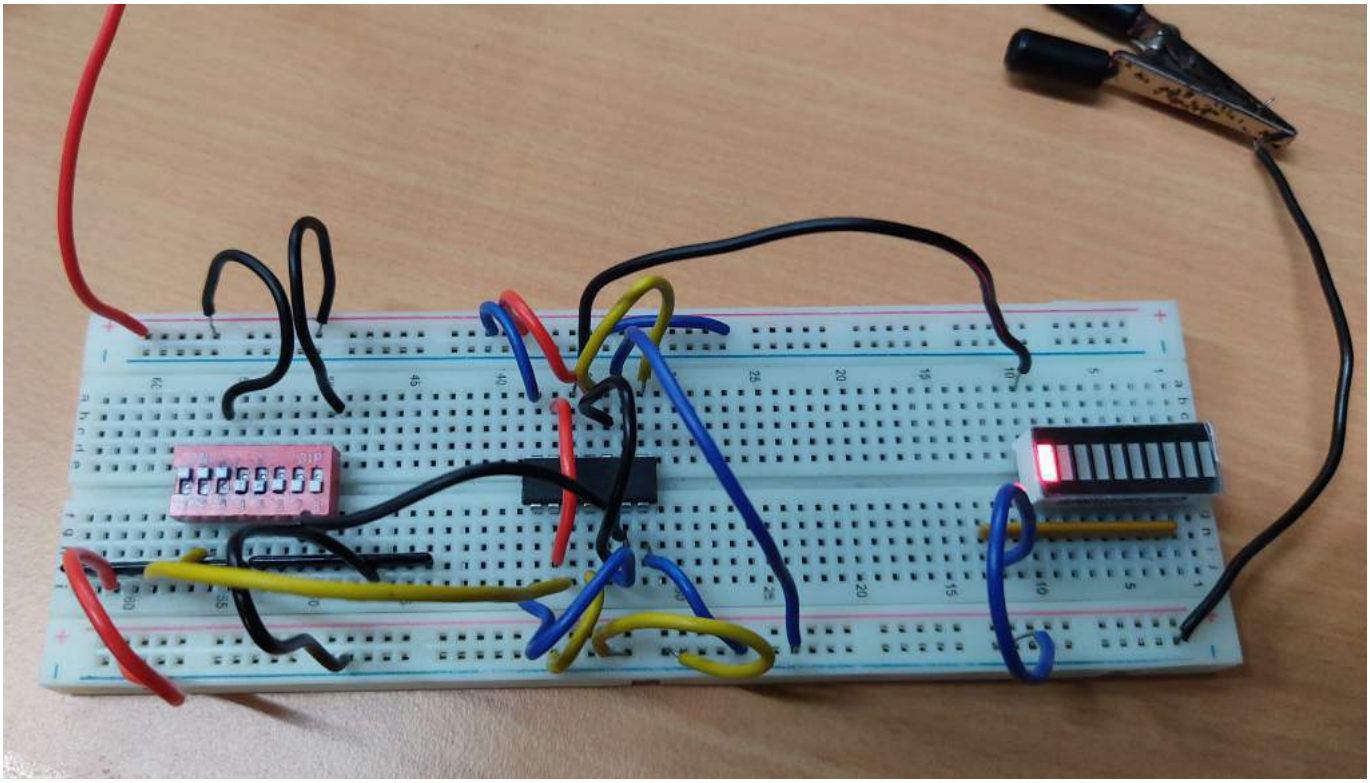


1_1 —> 1

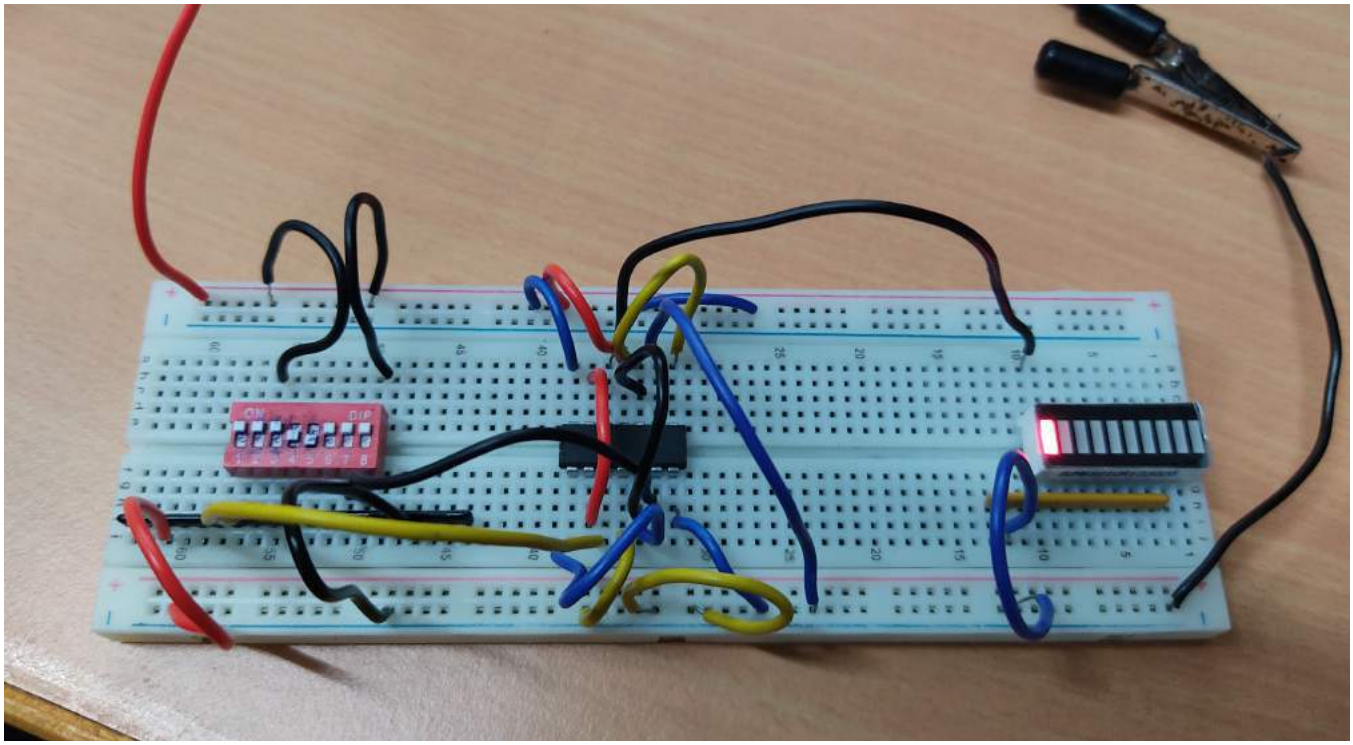


0_0 —> 0

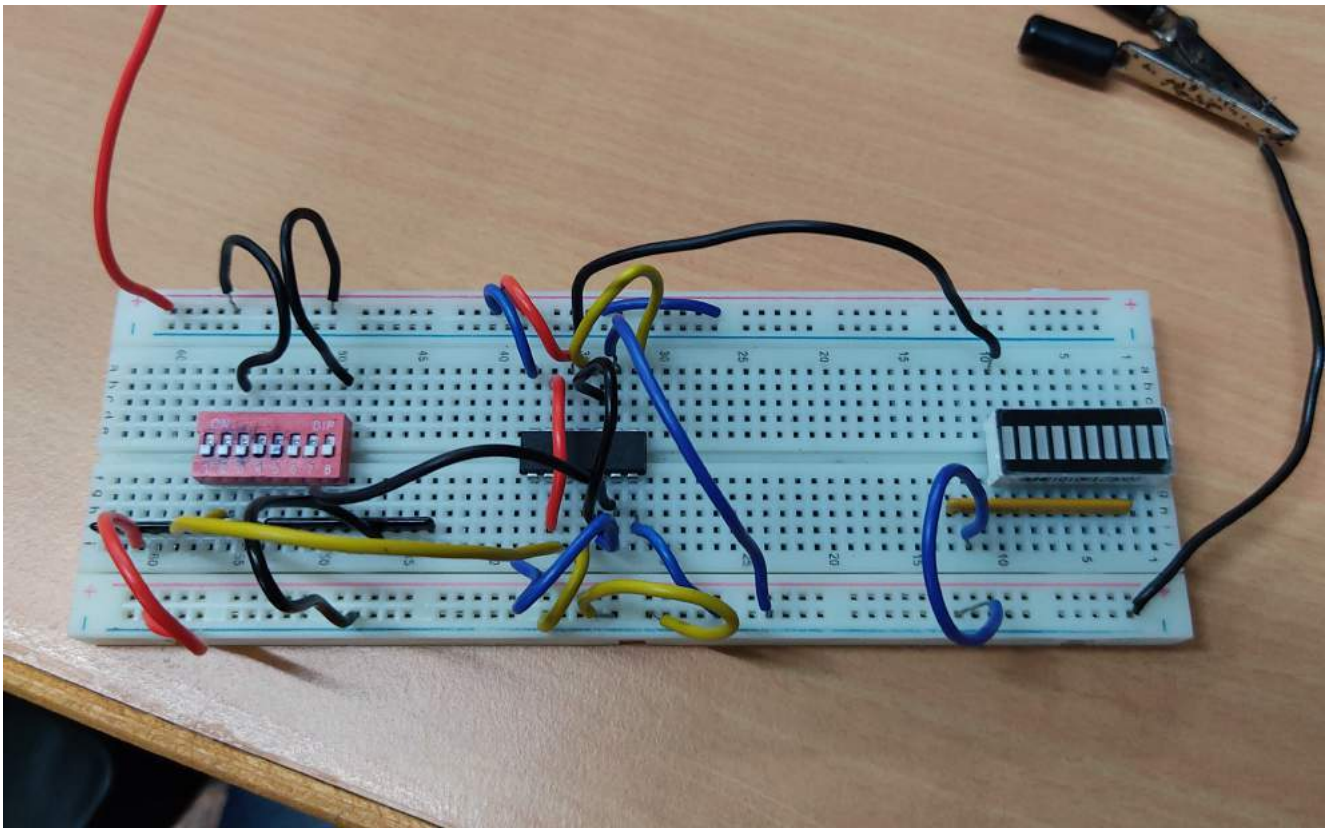
OR:



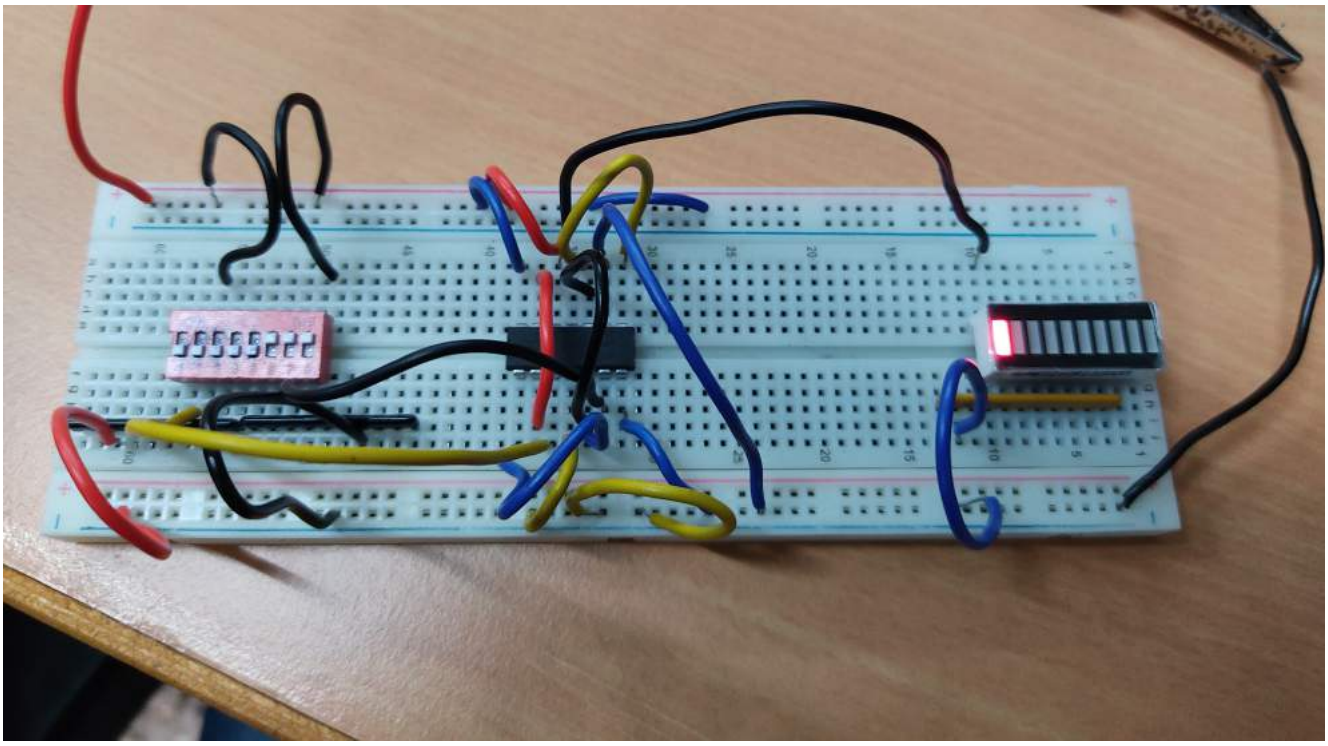
1_0 —> 1



1_1 —> 1

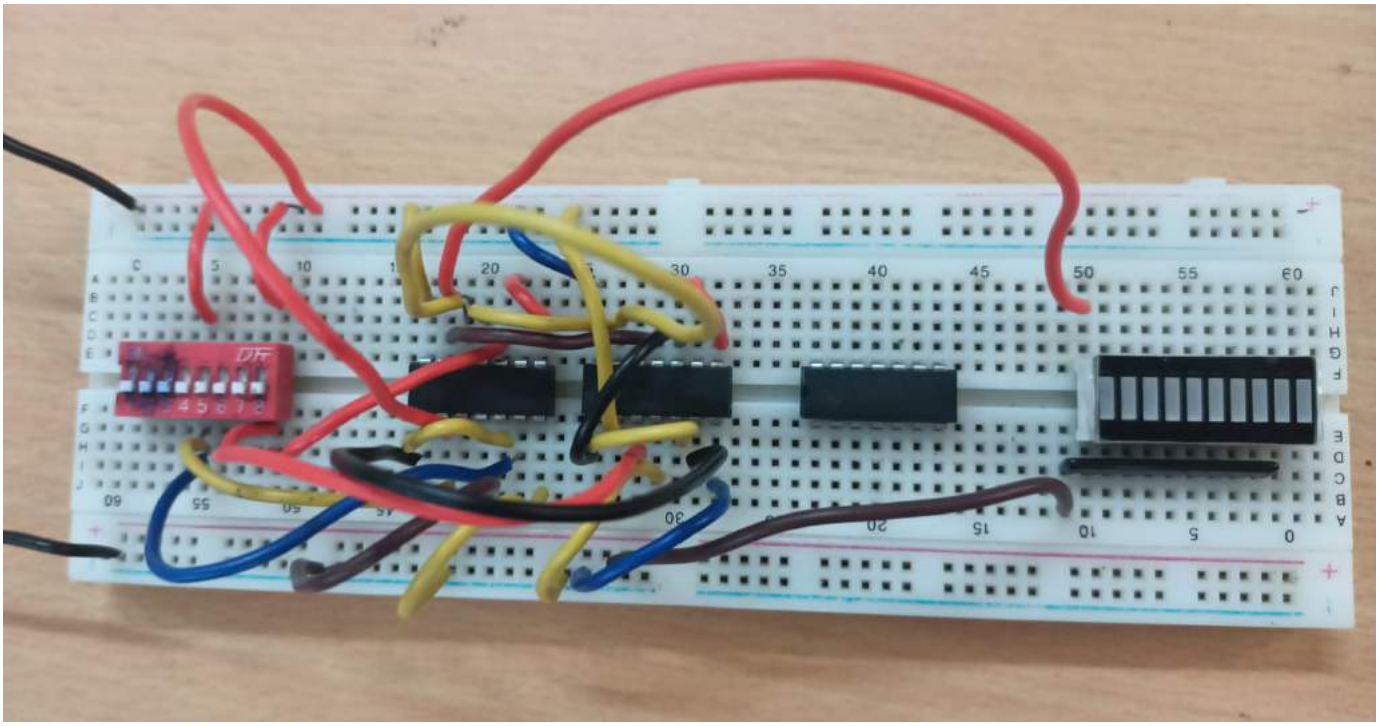


0_0 —> 0

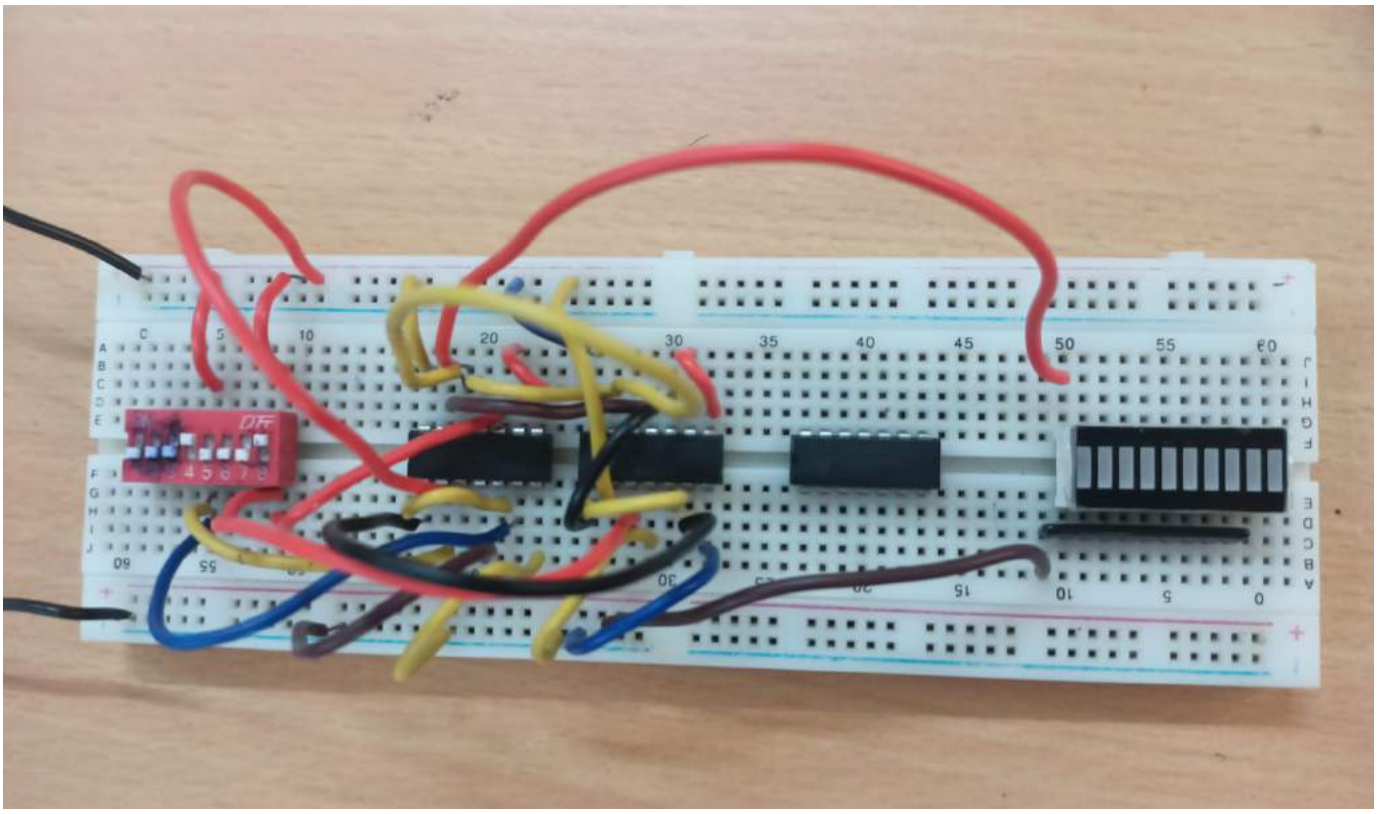


0_1 —> 1

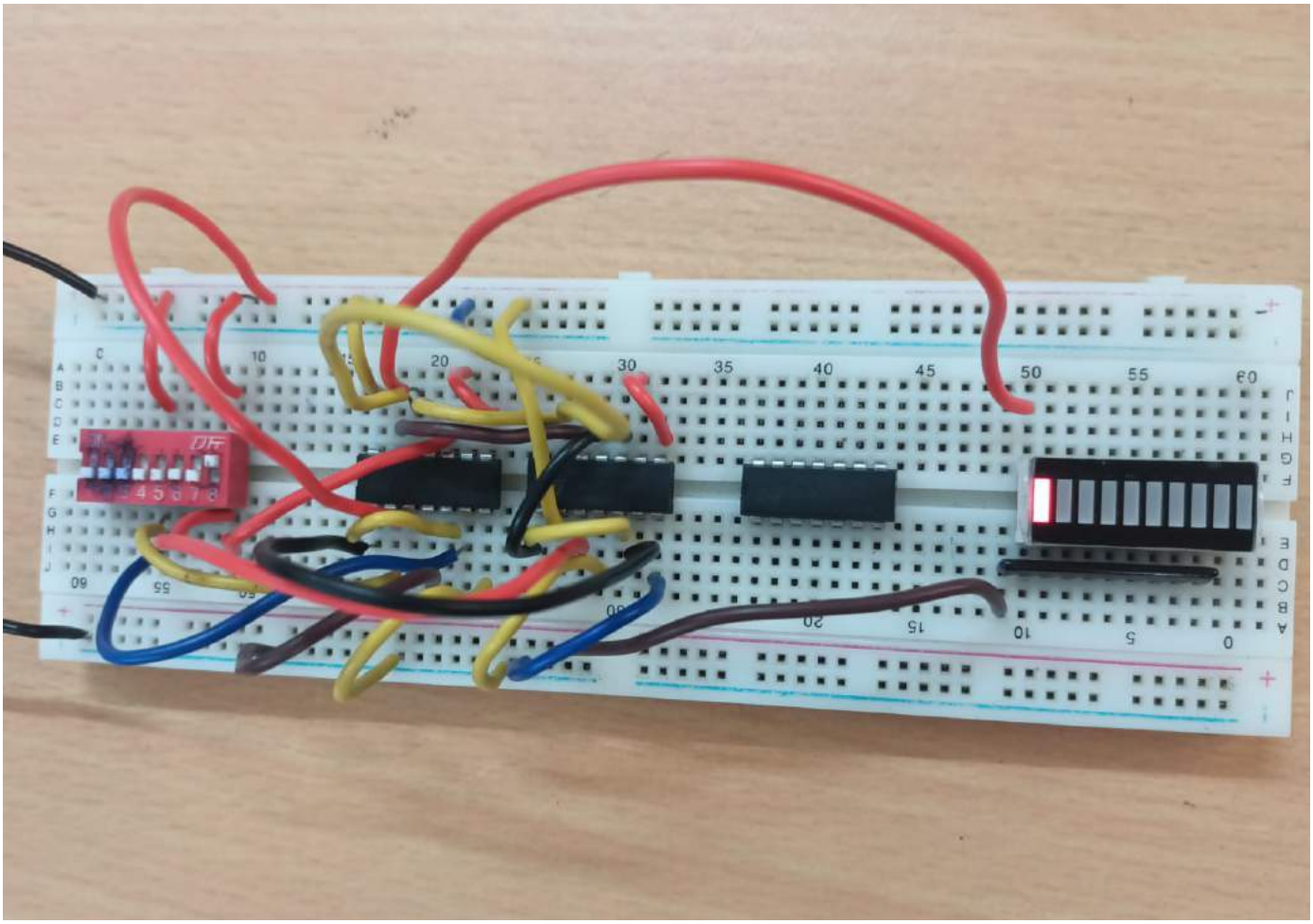
XOR:



0_0 —> 0

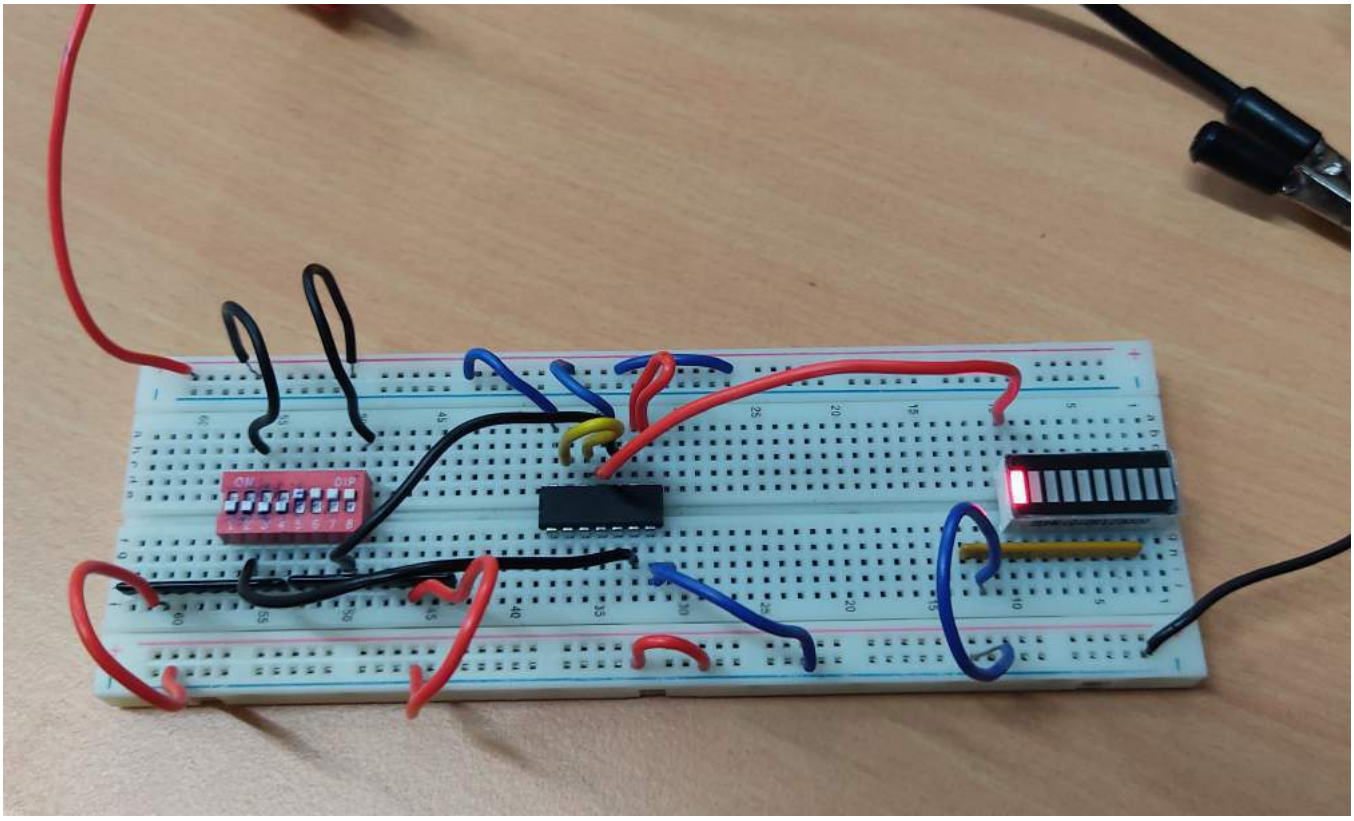


1_1 —> 0

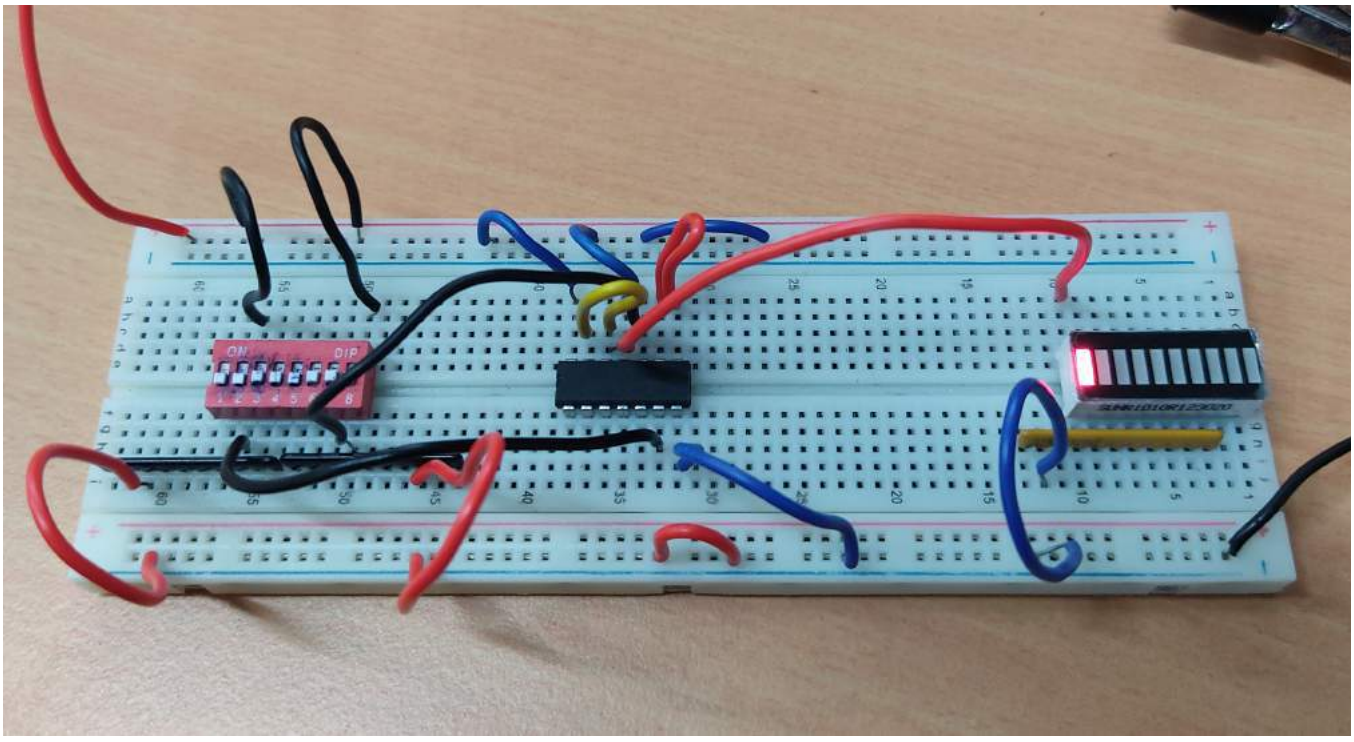


0_1 —> 1

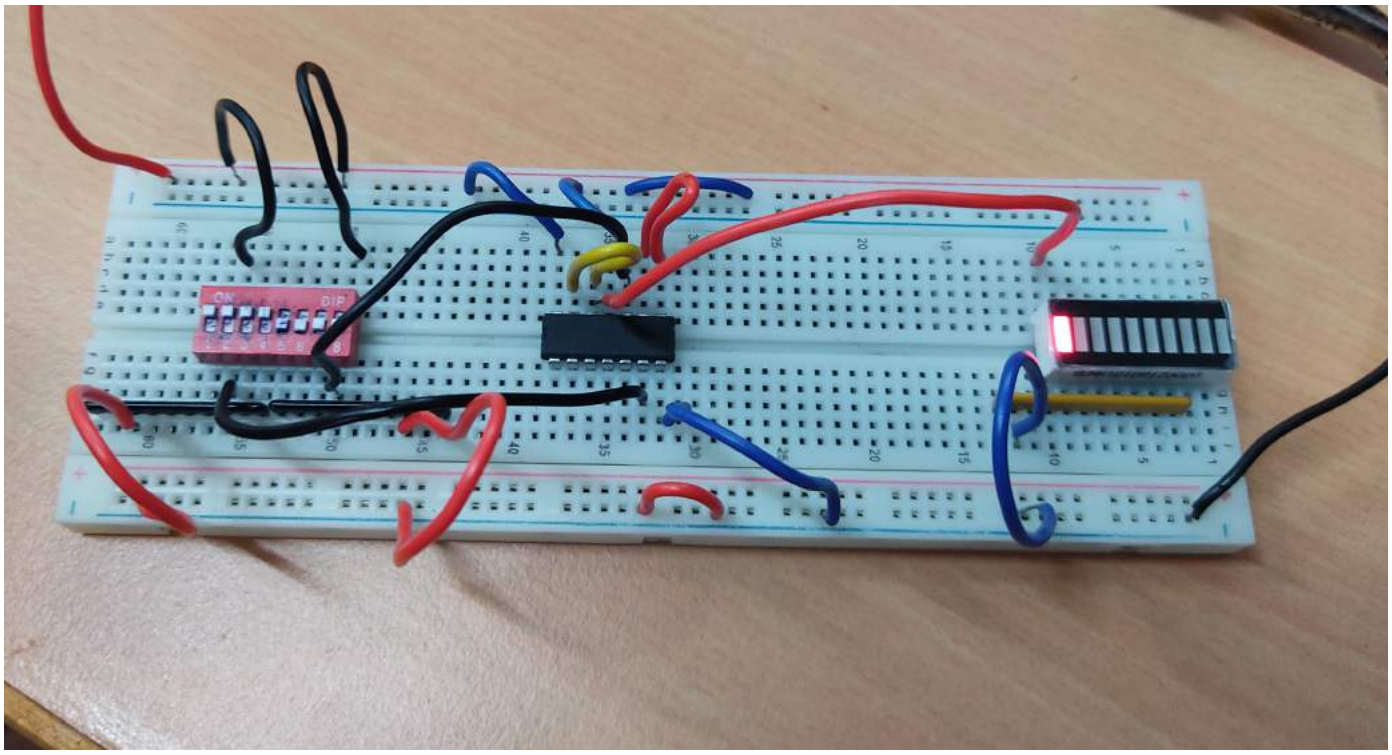
NAND:



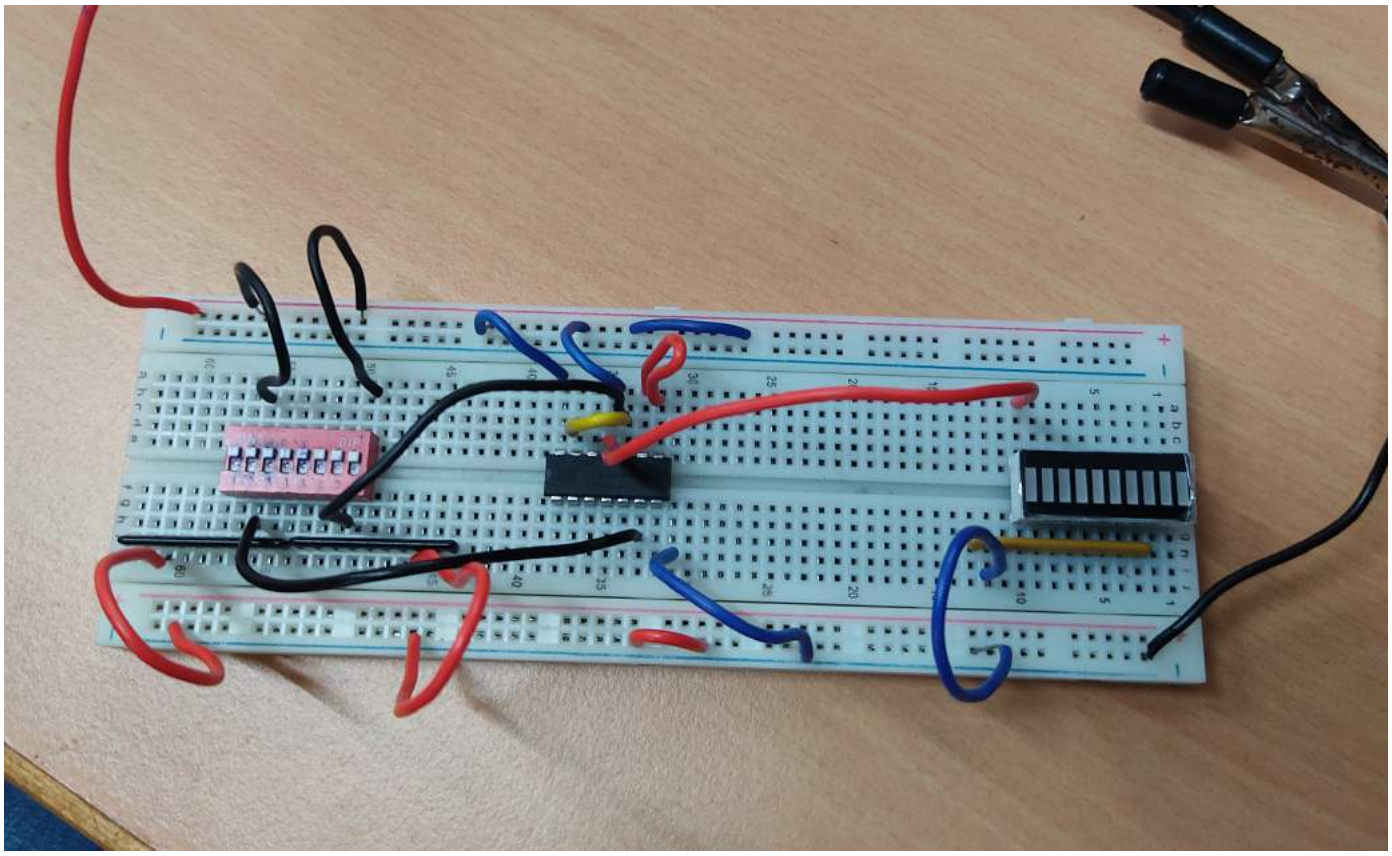
0_1 —> 1



0_0 —> 1

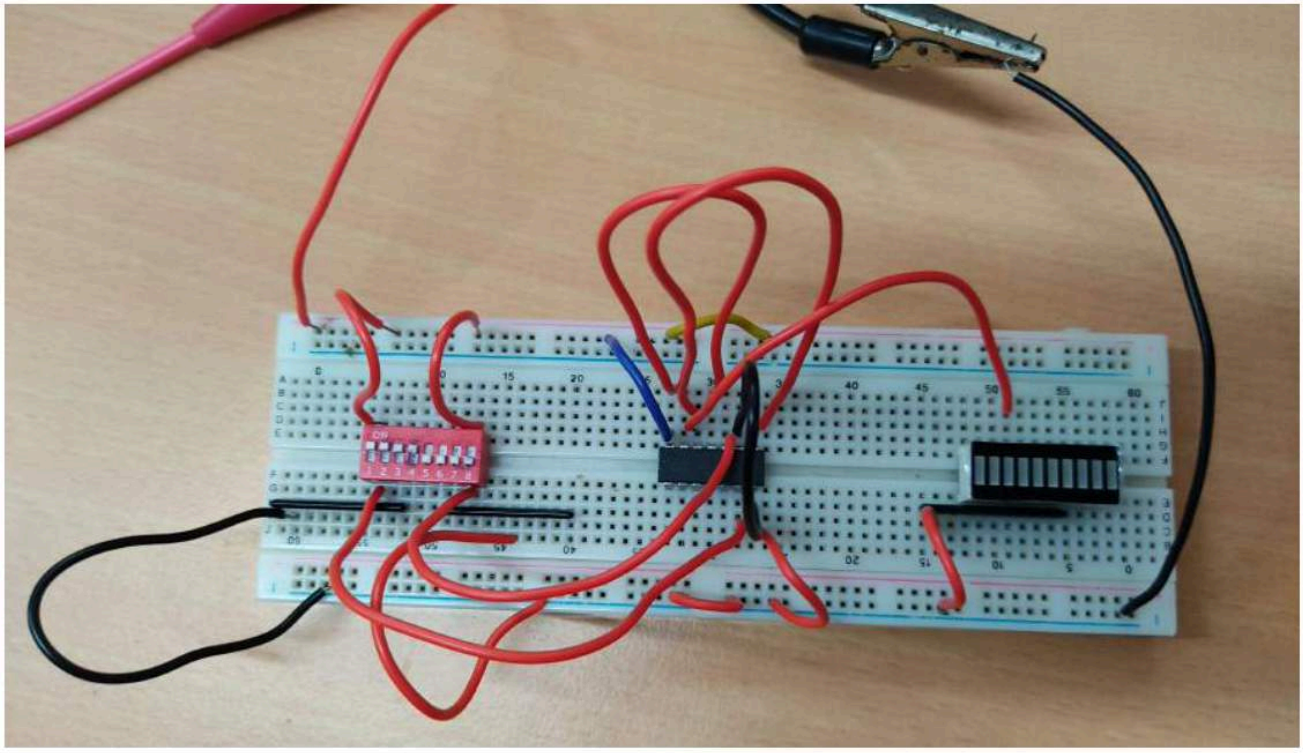


1_0 —> 1

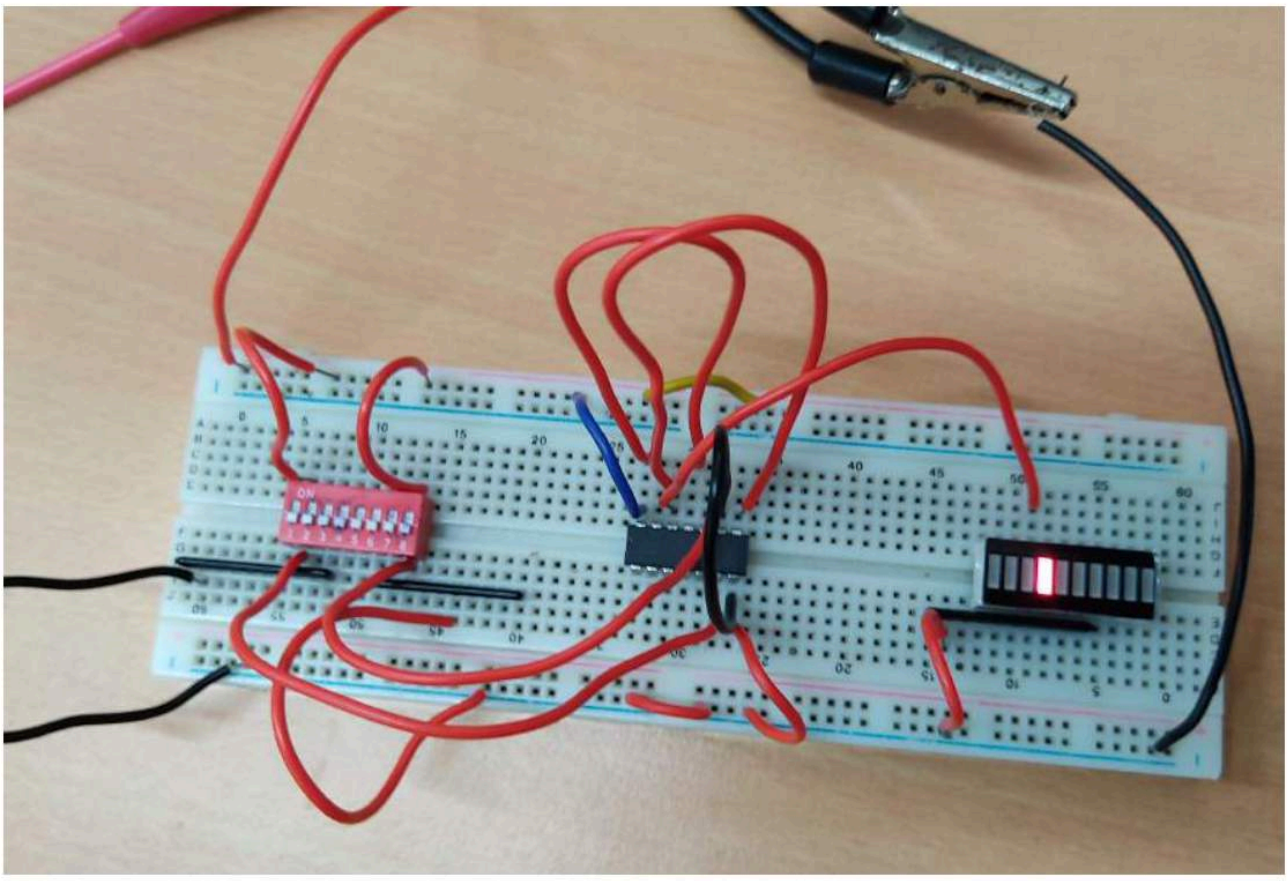


1_1 —> 0

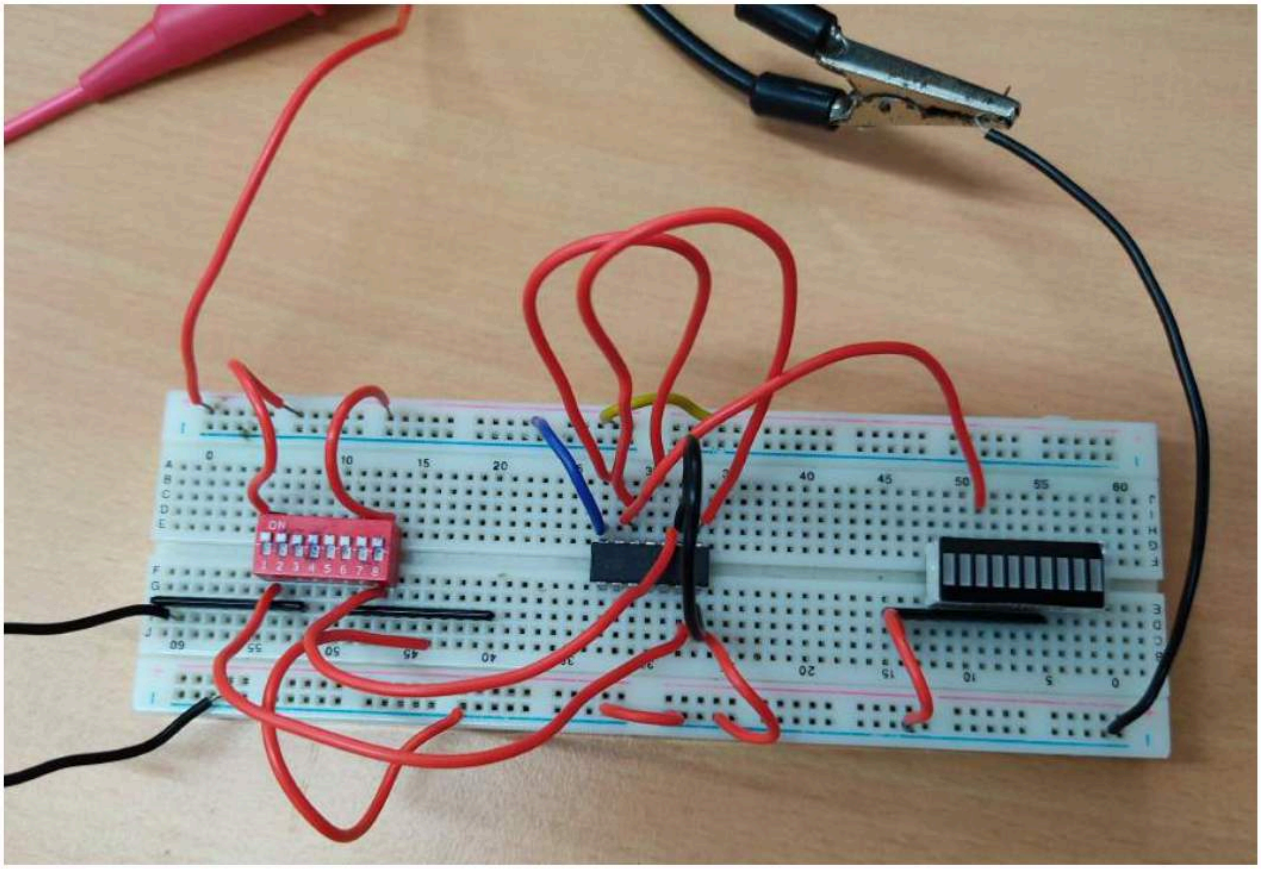
NOR:



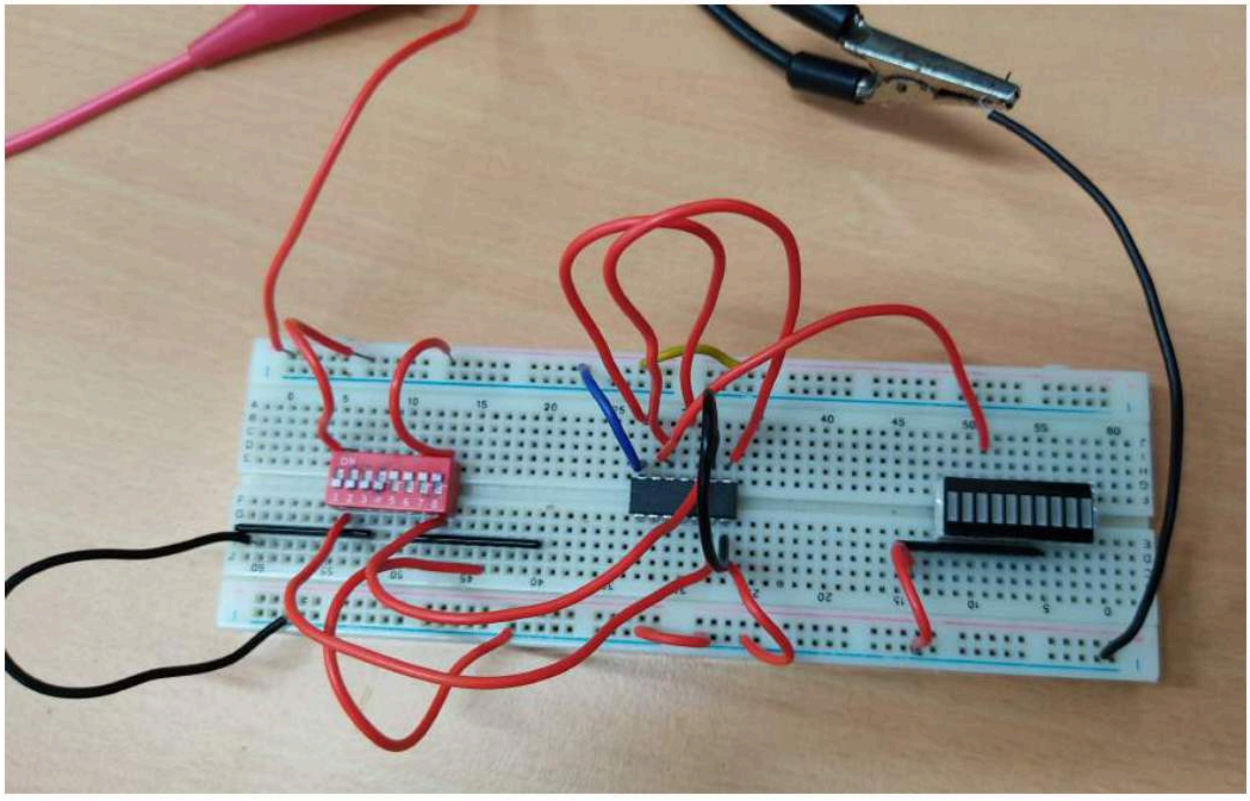
1_0 —> 0



0_0 —> 1

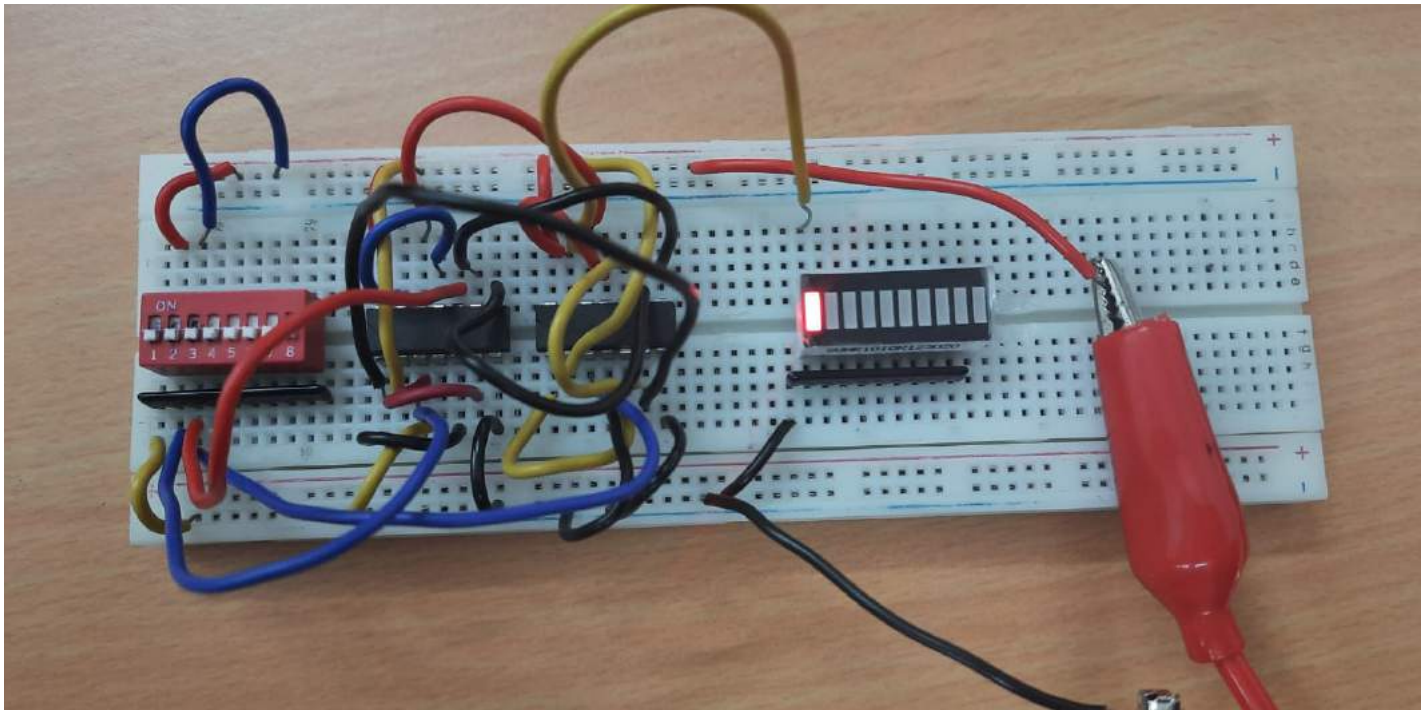


1_1 —> 0

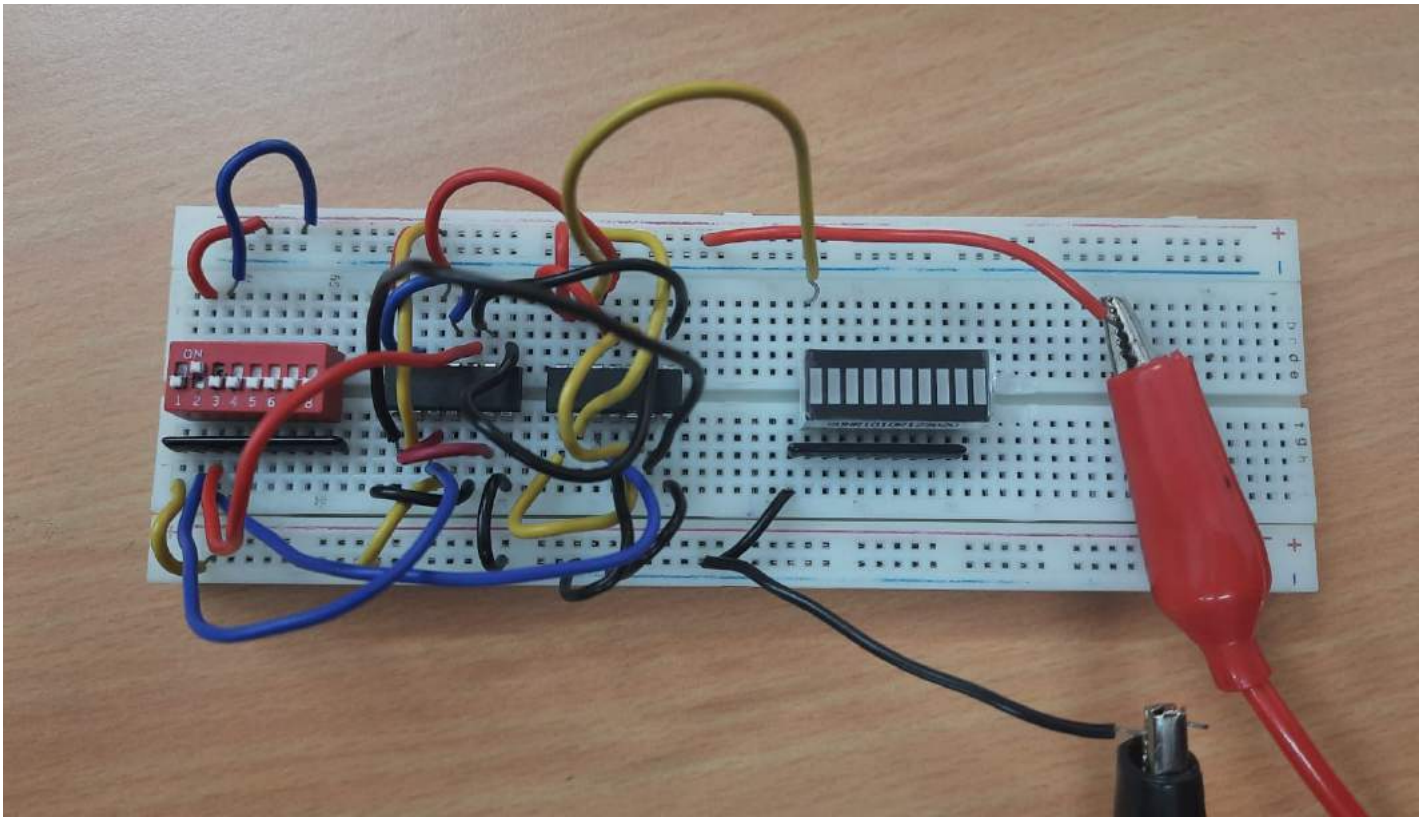


0_1 —> 0

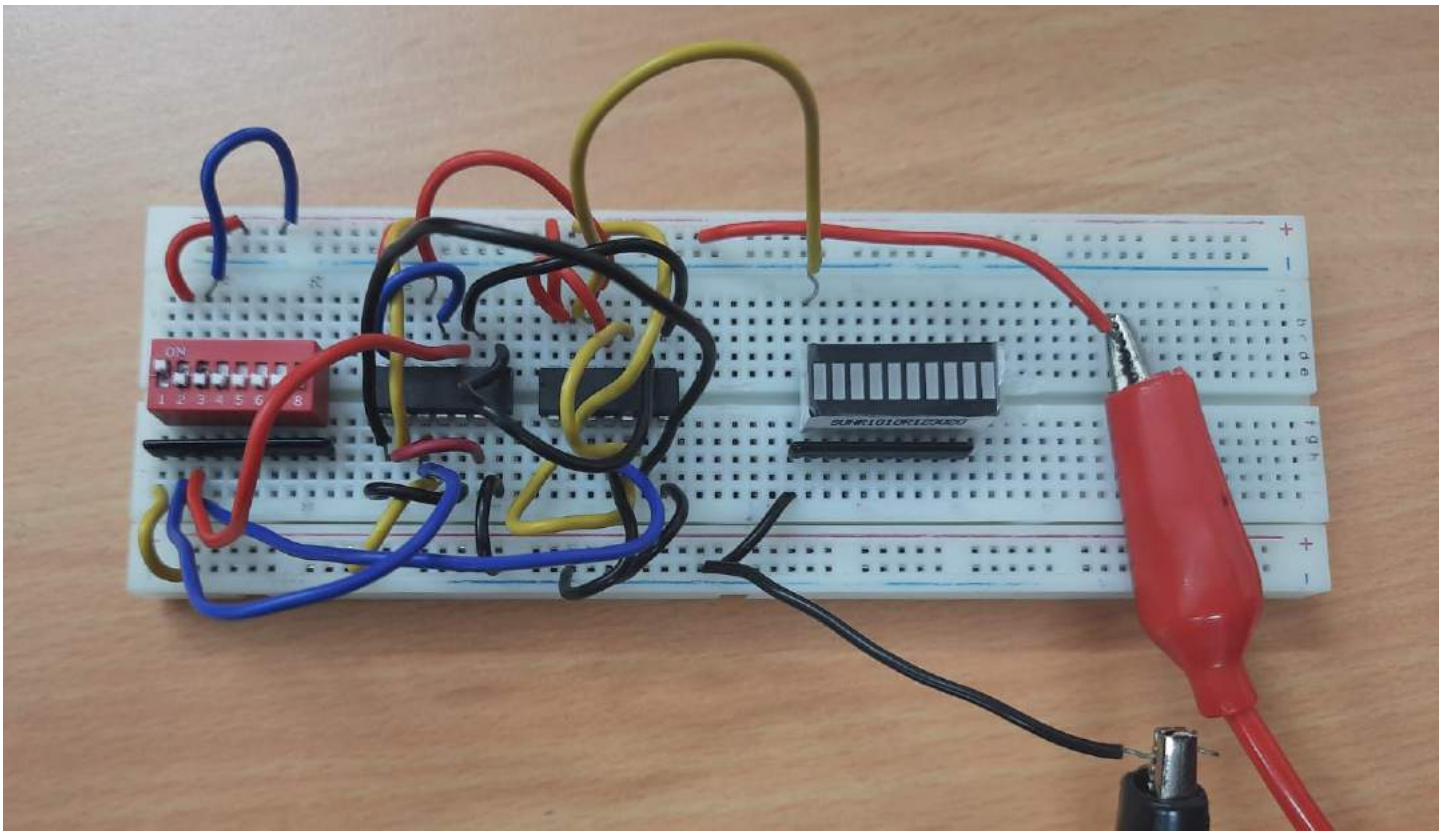
XNOR:



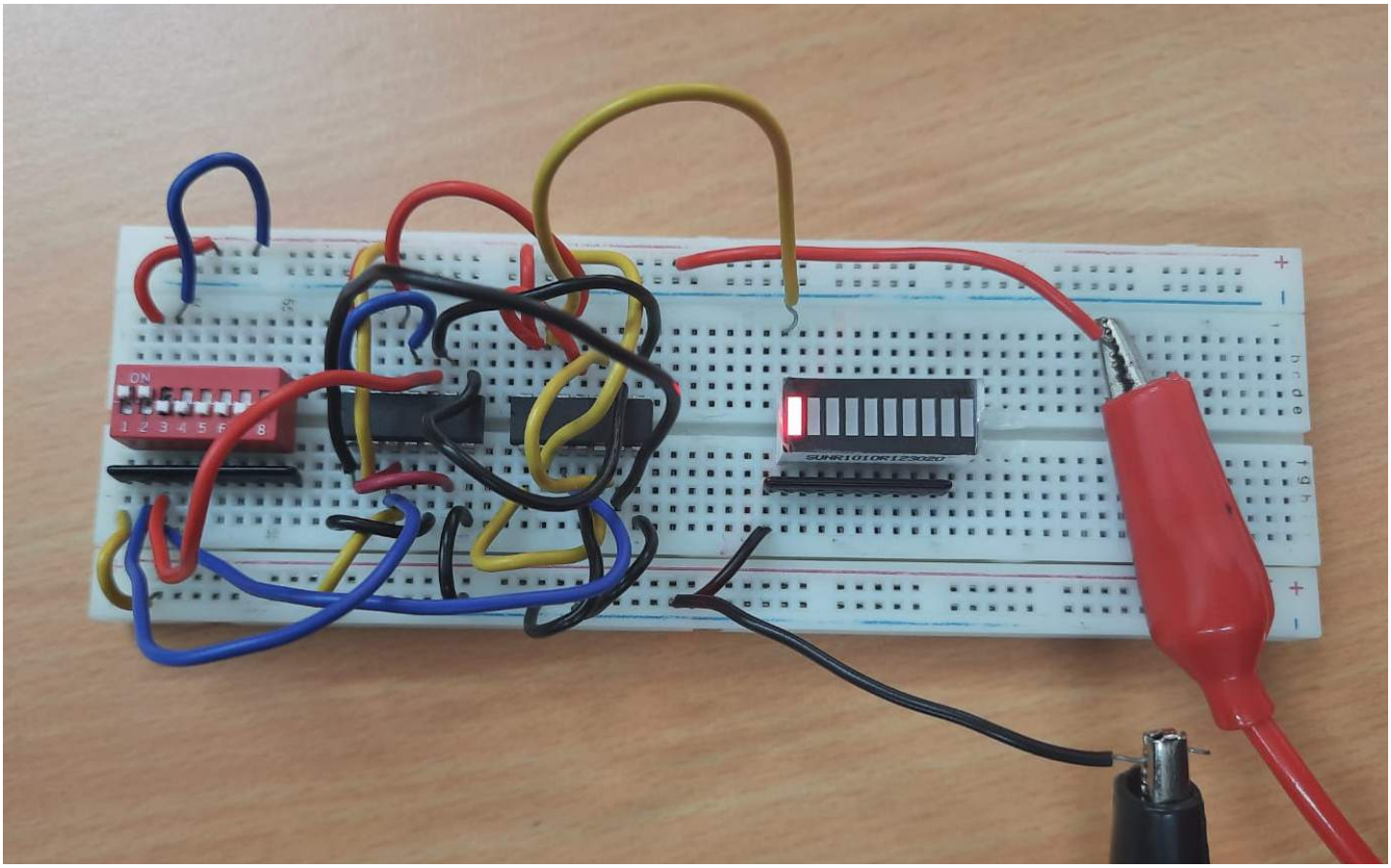
0_0 —> 1



0_1 —> 0



1_0 —> 0



1_1 —> 1