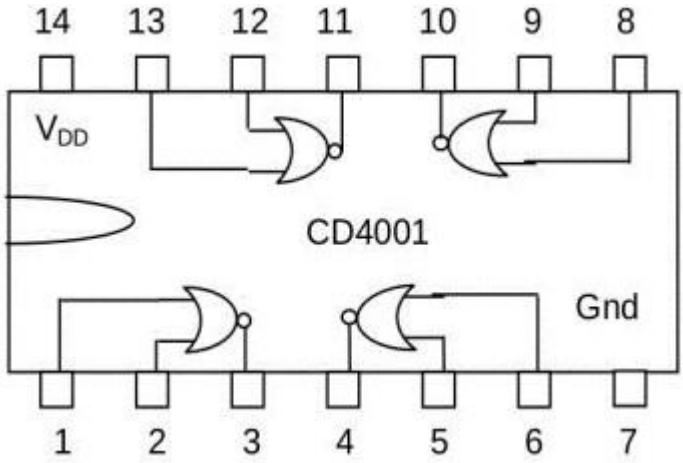
# Lab Report-6

Name: Chanda Akshay Kumar
Roll number: 2024102014
Group number: 7 (Teams-9)

## Experiment-1

## Objective:

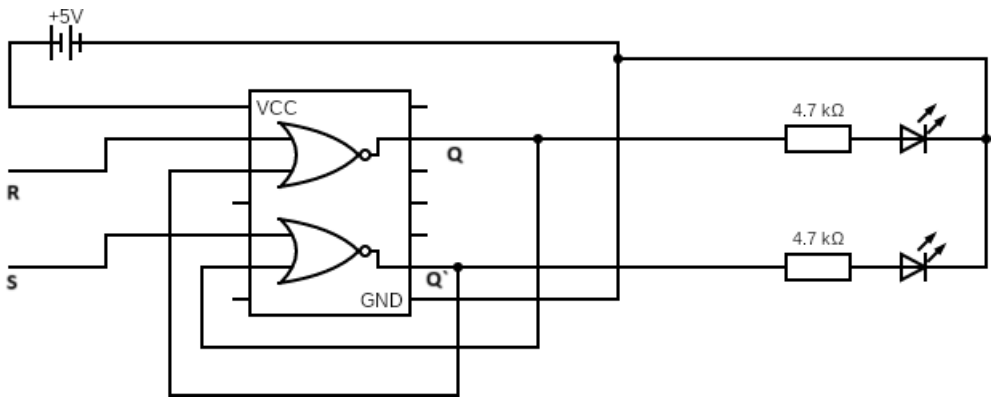
Preparing a RS latch by using NOR gates (CD4001 IC)




## Electronic components used:

1. One NOR IC (CD4001 IC)
2. DSM Kit
3. Connecting Wires
4. Power supply

# The reference circuit:




# Procedure:

1) Take Inputs by using Switch as R and S respectively.

2) Connect R and S to inputs 1A and 2A of the NOR Gate.

3) Connect outputs 1 and 2 of the gate to Inputs 2B and 1B of the same gate.

4) Connect Outputs 1 and 2 of the NOR gate to the resistors and form the resistors to Anode of the LEDs and from the cathode of LEDs to ground of the circuit.

# Conclusion:

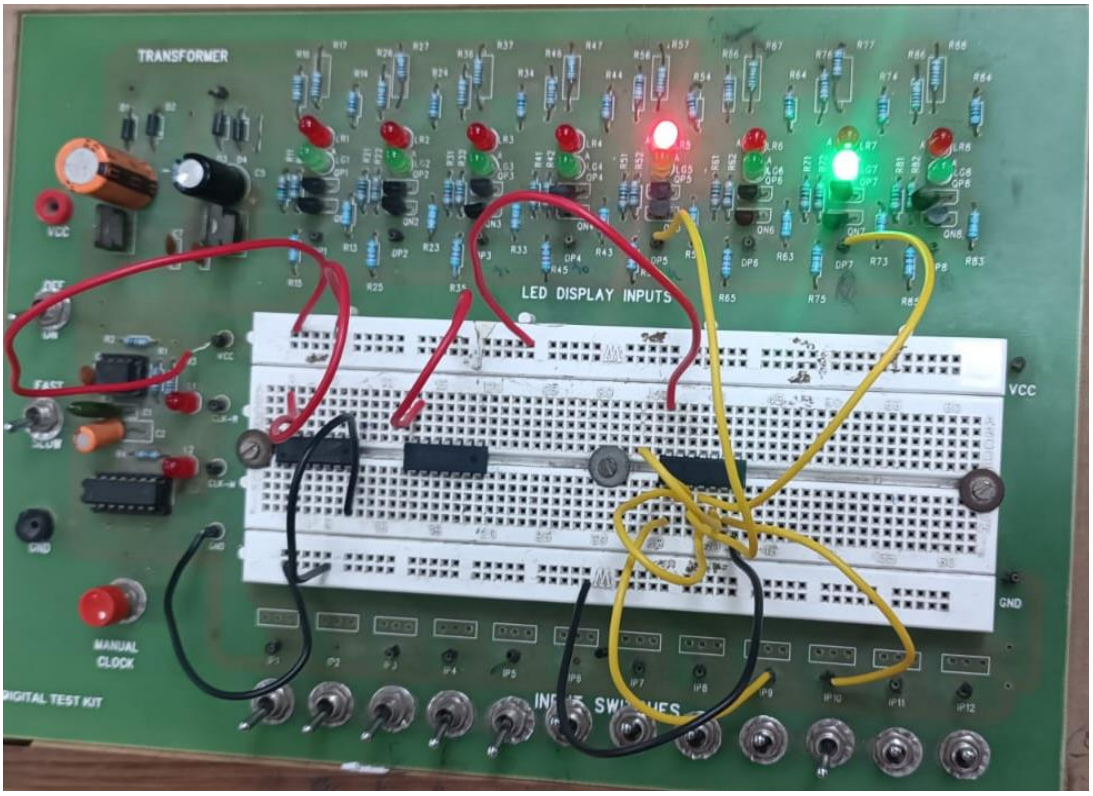
## Truth table:

| S | R | Q | Q’ |
|---|---|---|---|
| 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 |

We built an RS latch using the CD4001 chip and saw how it reliably stored and changed states based on the inputs. The experiment showed how latches handle memory in circuits, especially when inputs are toggled.latch gives us a invalid output (It’s value can’t be fixed).

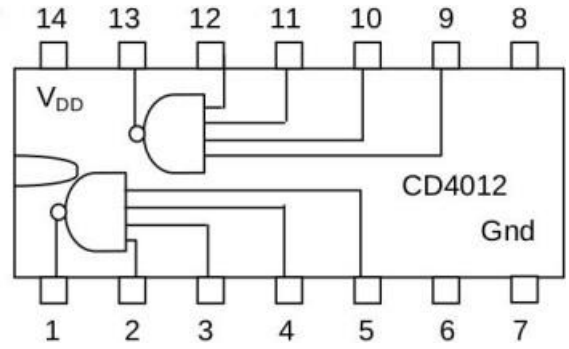
# Link for the Tinkercad simulation:



# Experiment-2

## Objective:

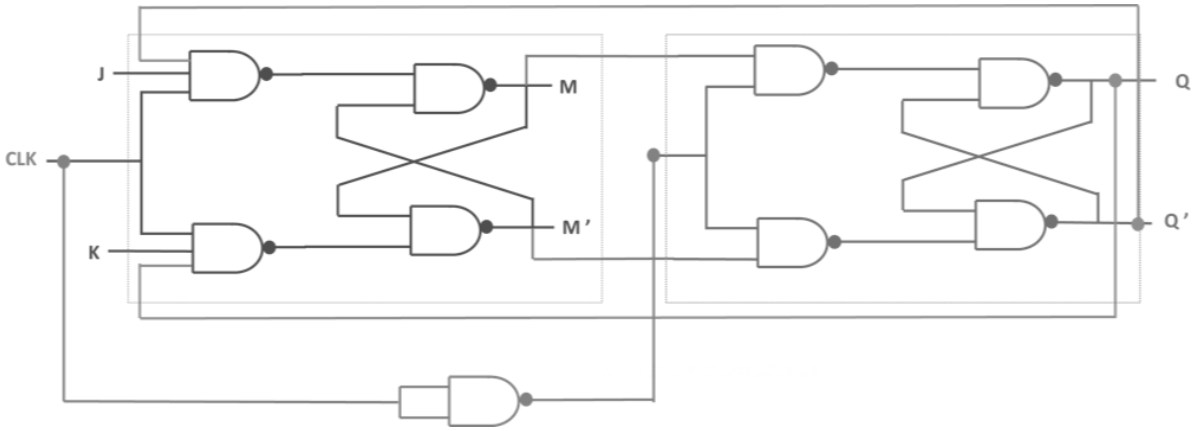
Producing JK Master-Slave Flip-Flop using logic gates.




## Electronic components used:

1. Six 2-input NAND Gate (7400 IC)
2. Two 4-Input NAND Gates (CD4012 IC)
3. Connecting wires
4. DSM Kit
5. Power supply

## The reference circuit:




## Procedure:

1) Take inputs such as Clock, J, and K respectively from the switches.

2) Take the left LED as a result of Q and the right LED as a result of Q' respectively.

3) Connect the unused inputs of the 4-input NAND gates to the HIGH (VCC)

level.

4) Build the Flip-flop as shown in the above reference circuit.

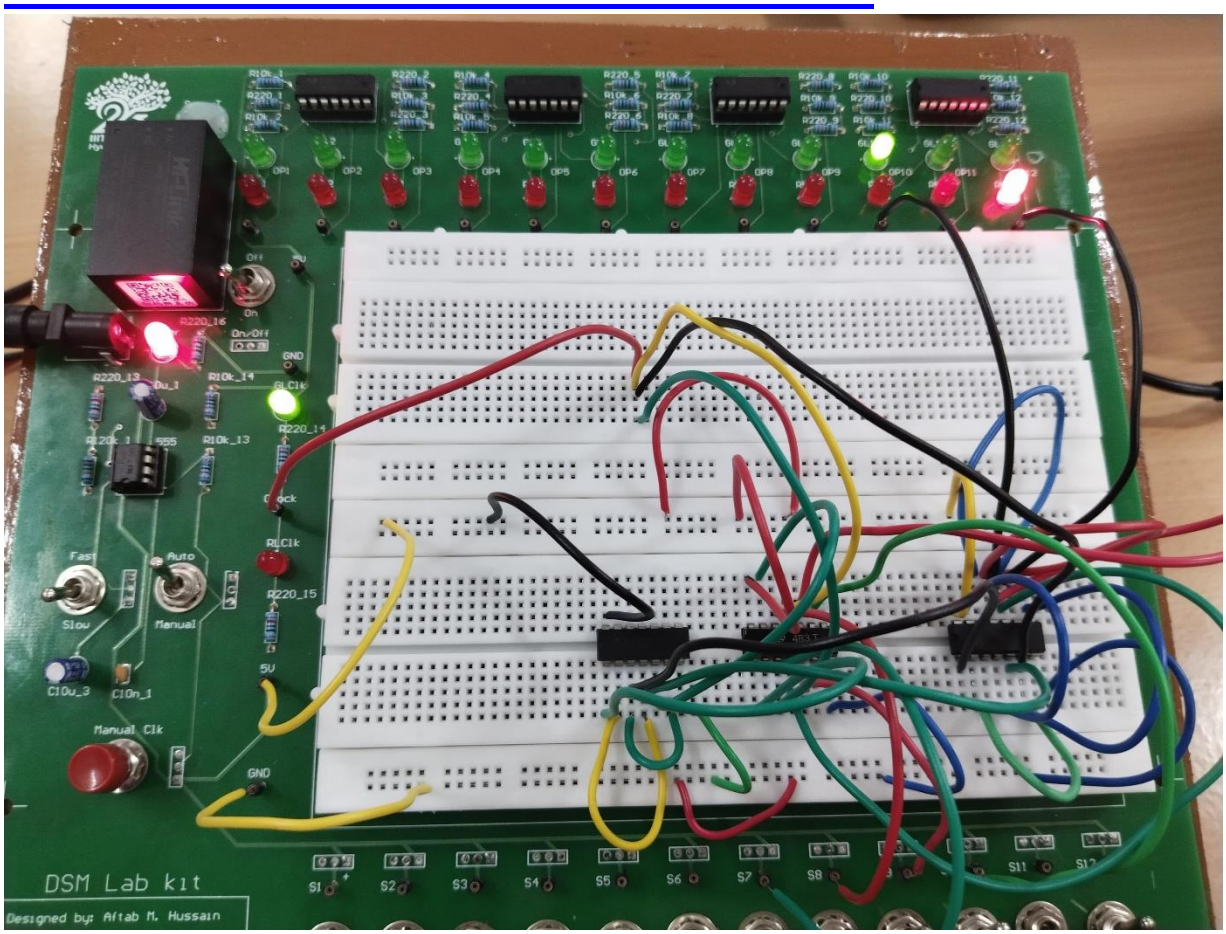
# Conclusion:

## Truth table:

| J | K | Action | Q(n+1) |
|---|---|---|---|
| 0 | 0 | Hold | Q |
| 0 | 1 | Clear | 0 |
| 1 | 0 | Set | 1 |
| 1 | 1 | Toggle | Q' |

By assembling a JK master-slave flip-flop with CD4012, we saw how it synced with the clock to control data flow. This helped us understand how flip-flops work in managing stable data in sequential circuits.

# Link for the Tinkercad simulation:



# Experiment-3

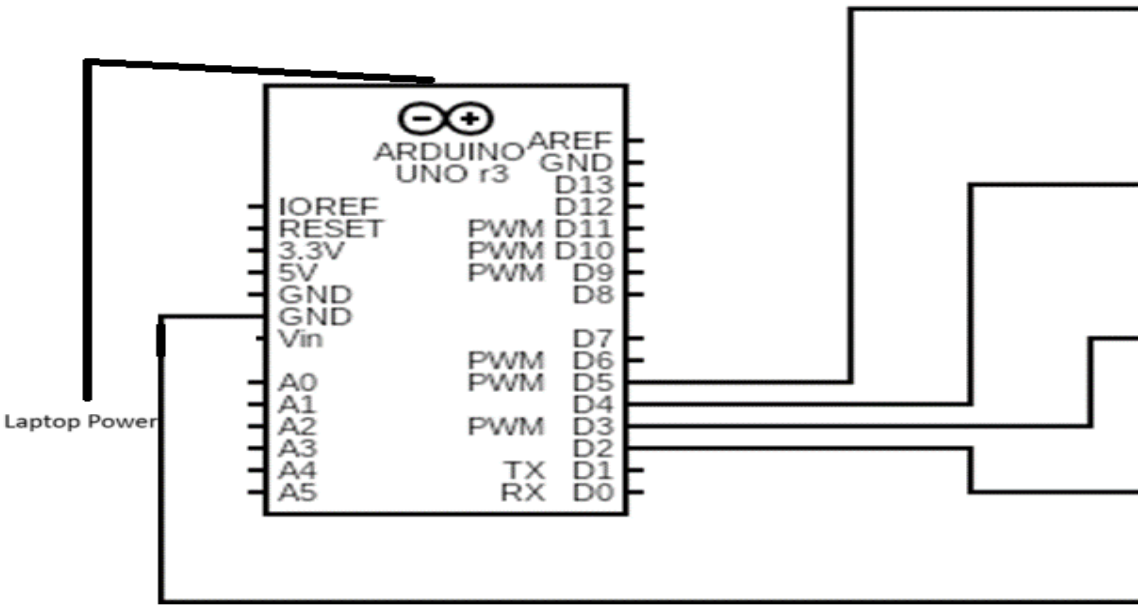
## Objective:

Preparing a 4-bit Up-Down Counter using Arduino by using code.

## Electronic components used:

1. Arduino UNO
2. Laptop with Arduino IDE installed
3. Connecting Wires
4. DSM Kit

## The reference circuit:



# Procedure:

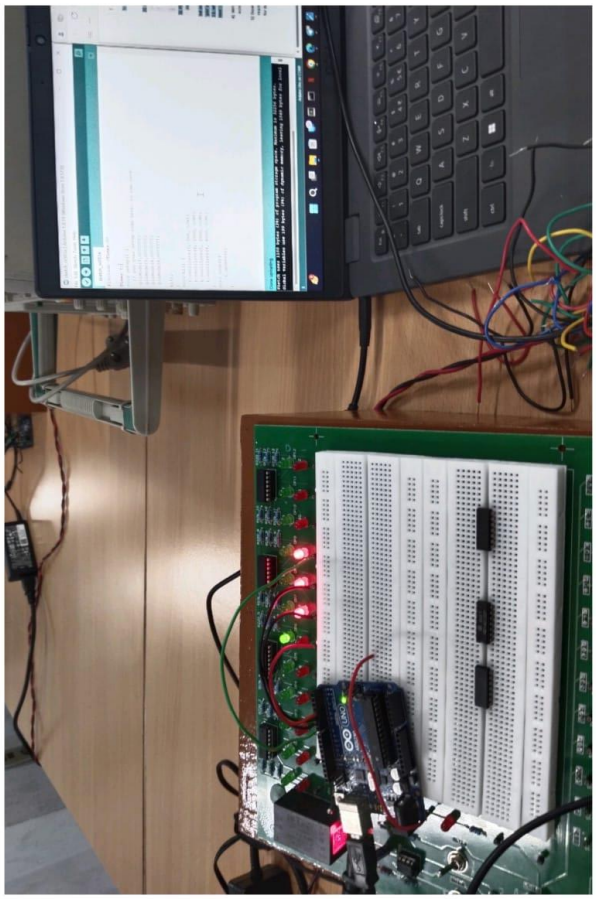
1. Connect the 9,8,7,6 pins of the Arduino to LEDs.
2. Connect the ground of the Arduino.
3. The execute the following code in the Arduino IDE:

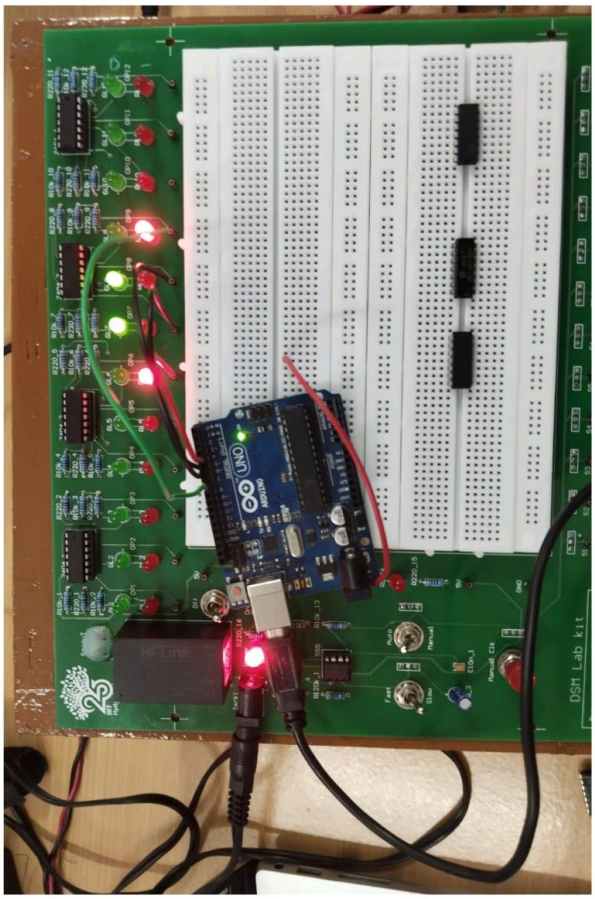
# Code:


```
#include <Timer.h>

Timer t;
void setup() {
  // put your setup code here, to run once:
 pinMode(8,OUTPUT);
 pinMode(9,OUTPUT);
 pinMode(6,OUTPUT);
 pinMode(7,OUTPUT);

hi();
}
void hi() {
  t.oscillate(9, 500, LOW);
  t.oscillate(8, 1000, LOW);
  t.oscillate(7, 2000, LOW);
  t.oscillate(6, 4000, LOW);
}
void loop(){
    t.update();
}
```





## Conclusion:

I have observed that when the code is executed the LEDs starting glowing in a manner such that they represent numbers 1 to 15 in their binary form, then after reaching 15 then they represent 15 to 0. This shows that it is a 4 bit up down counter.

By this experiment we have created 4 bit Up-Down counter using Arduino.

## Link for the Tinkercad simulation: