Lab Report 5

Name: Vishak Kashyap K
Roll Number: 2023113012
Group: 9

Experiment 5 (Part A) - SR Latch

Objective:

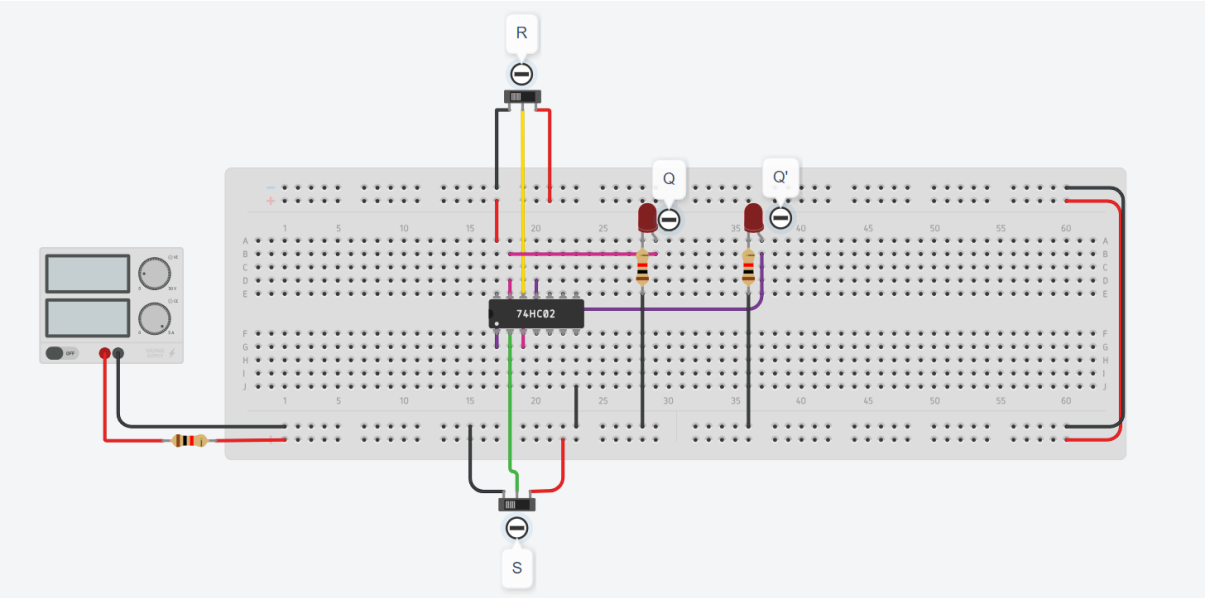
To design, assemble and test an SR Latch.

Electronic Components Required:

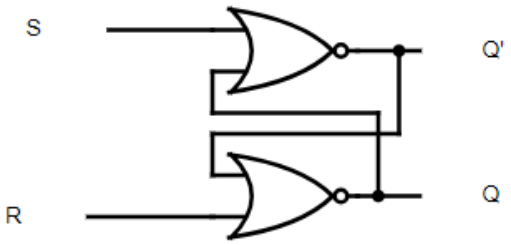
1. Digital Test Kit
2. CD4001 IC NOR Gate
3. Voltage Supply
4. Normal Wires

Reference Circuit:

1. Tinkercad Screenshot



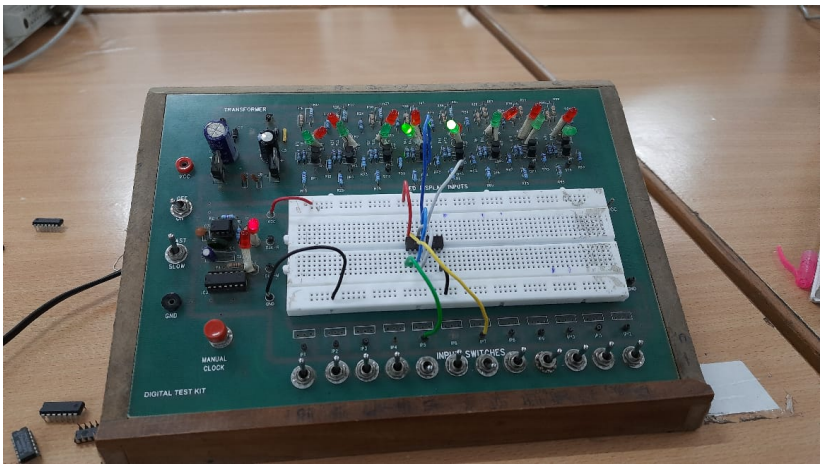
2. Circuit Diagram

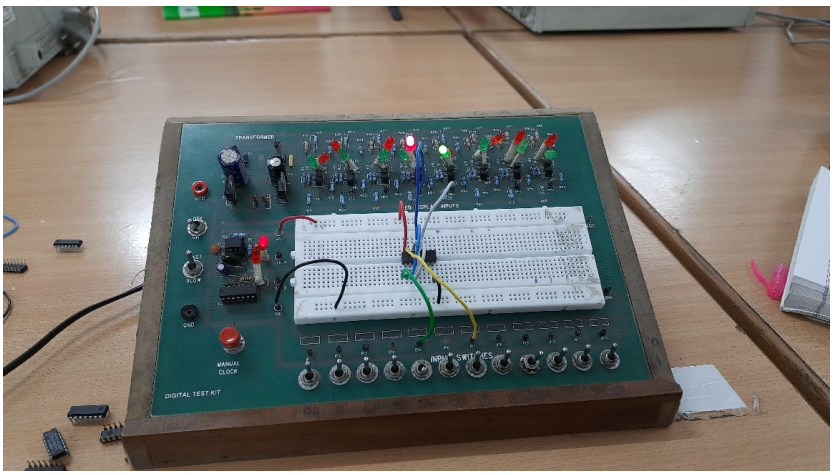


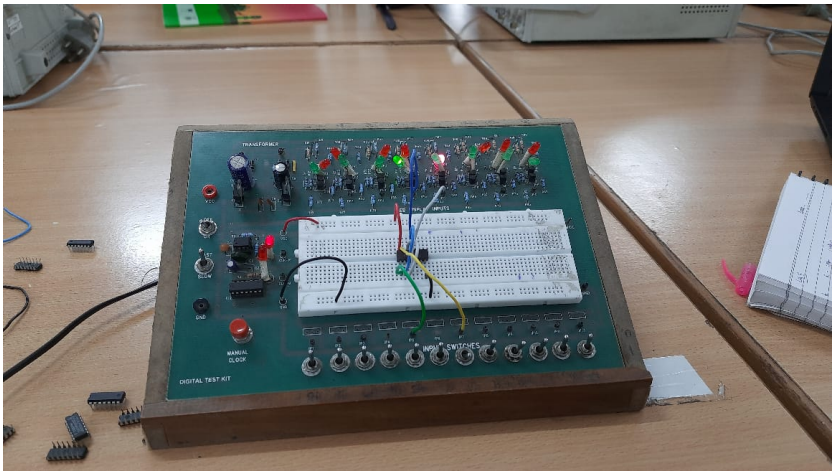
Procedure:

1. Attach the CD4001 IC to the breadboard. Connect the VCC and GND pins of the breadboard to those of the ICs.
2. Using two switches as inputs for two NOR gates and the outputs of each as a second input for the other.
3. Connect the outputs to LEDs.
4. View the outputs for different values of input and tabulate the observations.

Observations:







1. The expected working of the SR Latch is as follows:

S	R	Q	Q’
0	0		
0	1		
1	0		
1	1		

2. For the given inputs, we observe the following outputs.

R	S	Q	Q’
0	1	1	0
0	0	1	0
1	0	0	1
0	0	0	1
0	1	1	0
1	0	0	1
0	1	1	0
0	0	1	0
1	1	0 (Forbidden)	0 (Forbidden)
0	0	Random	
1	0	0	1
1	1	0 (Forbidden)	0 (Forbidden)
0	0	Random	
0	1	1	0
1	1	0 (Forbidden)	0 (Forbidden)
0	0	Random	

Conclusions:

We have created an RS Latch, verified its working and observed its operation for a given set of inputs.

Link for Tinkercad Simulation:

https://www.tinkercad.com/things/3AYyZhgOgeI-5a-sr-latch-/editel?sharecode=1i9ETRpt9OGOupuk_gfb6A8RrgKj2YeraFlHqfymRkI

Experiment 5 (Part B) - JK Master-Slave Flip-Flop

Objective:

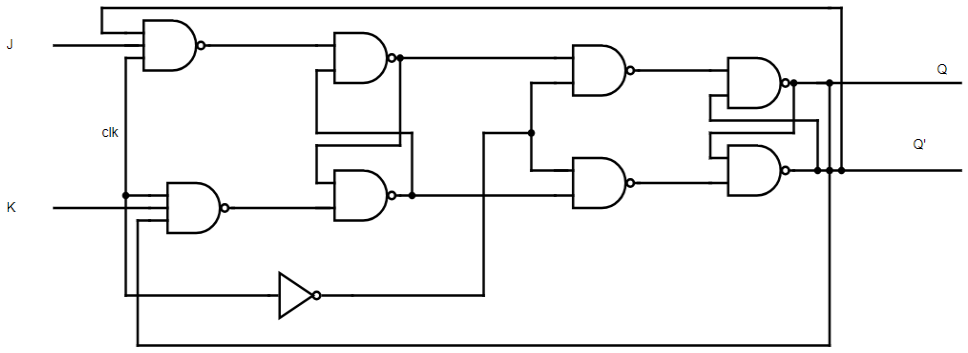
Designing, assembling and testing a JK Flip Flop.

Electronic Components Required:

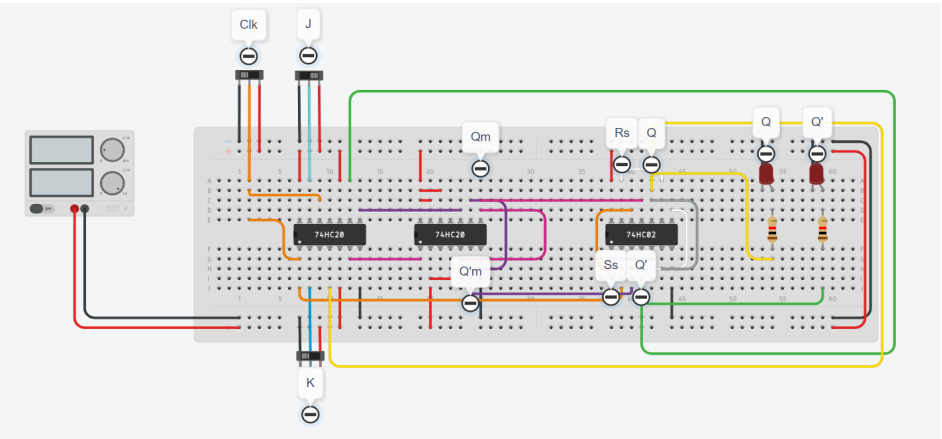
1. CD4012 IC NAND Gate
2. CD4001 IC NOR Gate
3. Normal Wires
4. Digital Test Kit
5. Voltage Supply

Reference Circuit:

1. Circuit Diagram



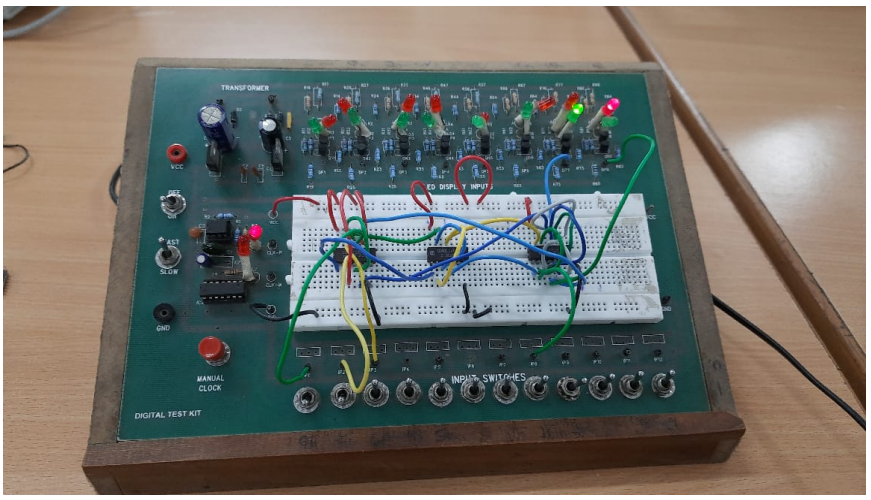
2. Tinkercad Screenshot

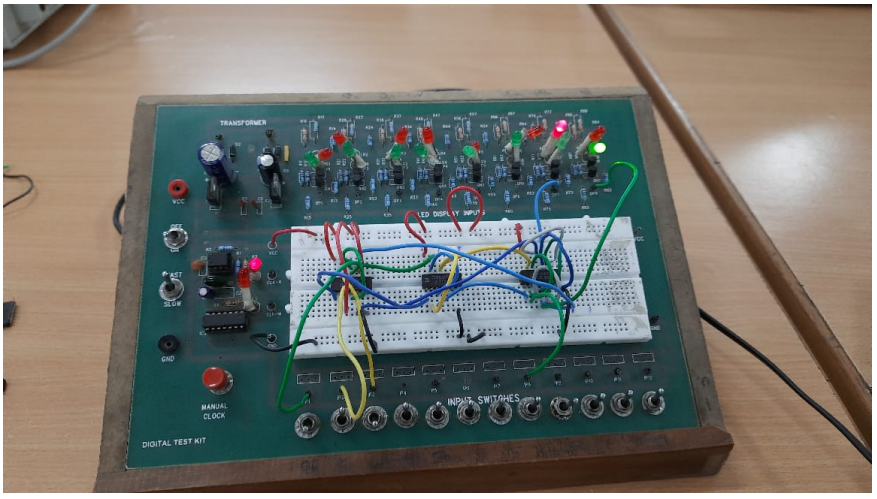


Procedure:

1. Connect the CD4012 and CD4001 ICs to the breadboard. Connect the VCC and GND pins to those of the Breadboard.
2. Make all the connections to the ICs as per the reference diagram.
3. Observe the outputs of the Flip flop for various values of input. Verify its working and tabulate the outputs.

Observations:





We obtain the following truth table:

J	K	Q(t+1)
0	0	Q(t) [Unchanged]
0	1	0 [Reset]
1	0	1 [Set]
1	1	Q'(t) [Complements]

Conclusions:

We have created a JK Flip Flop and verified its working.

Link for Tinkercad Simulation:

https://www.tinkercad.com/things/lY8Xi6O6CUZ-5b-jk-master-slave-flip-flop/editel?sharecode=BeIXjz0Q8fh69ergIv1YuQ9vy7zBdgS2MVwP4BetHg0

Experiment 5 (Part C) - 4-Bit Up-Down Counter

Objective:

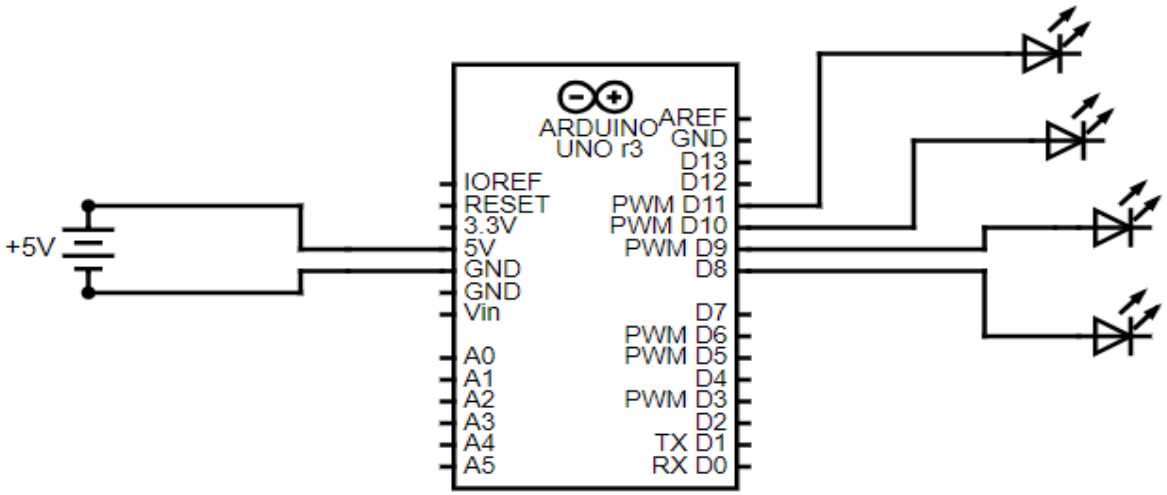
To design a 4-bit Up-Down Counter that first goes UP from 0(0000) to 15(1111), then goes down from 15 to 0, then goes UP, and this cycle repeats until the simulation stops.

Electronic Components Required:

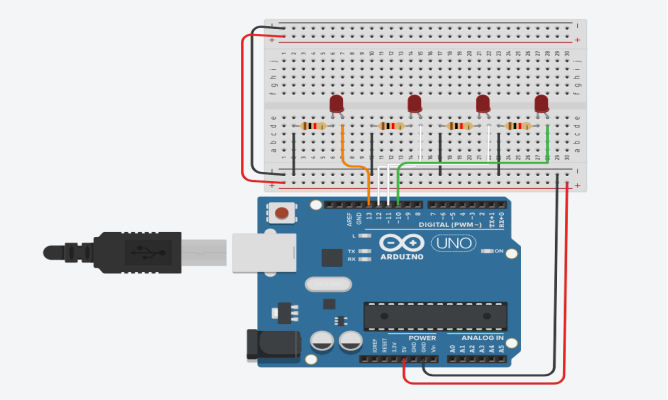
1. Arduino UNO
2. Digital Test Kit
3. Normal Wires
4. Jumper Wires
5. Voltage Supply

Reference Circuit:

1. Circuit Diagram



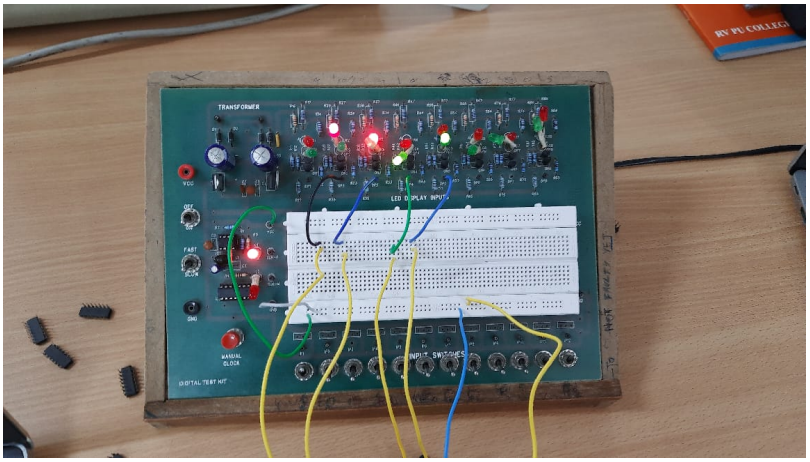
2. Tinkercad Screenshot



Procedure:

1. Using the Timer library, implement a 4-bit counter. The bit outputs of the 4-bit ripple counter will be represented by LEDs. (On LED for each bit)
2. Initialise a Timer t and use the t.oscillate function to toggle the pin values in a pre-defined time period (each pin will have a different time period)
3. Once the ripple counter reaches 15(1111), make it go down to 0. Stop existing timers using t.stop.
4. Use t.every to fire a function after a set-interval of time that will do two tasks: stop all timers and restart them in the opposite direction.

Observation:



Code:

Intro_to_Arduino_Workshop.ino

```
};
#endif

//// YOUR CODE STARTS HERE

Timer t;
int pin0 = 3;
int pin1 = 6;
int pin2 = 9;
int pin3 = 13;

bool forward = true;
int t0,t1,t2,t3;
int d = 1000;

void setup()
{
    Serial.begin(9600);
    pinMode(pin0, OUTPUT);
    pinMode(pin1, OUTPUT);
    pinMode(pin2, OUTPUT);
    pinMode(pin3, OUTPUT);
    t0 = t.oscillate(pin0, d, LOW);
    t1 = t.oscillate(pin1, d << 1, LOW);
    t2 = t.oscillate(pin2, d << 2, LOW);
    t3 = t.oscillate(pin3, d << 3, LOW);

    t.every(d << 4, stopAllTimers);
}

// 1 unit of your timer = 500ms in real time

void loop() {
    t.update();
}

// "every" X milliseconds
void stopAllTimers()
{
    t.stop(t0);
    t.stop(t1);
    t.stop(t2);
    t.stop(t3);
    forward = !forward;
    if (forward)
    {
        t0 = t.oscillate(pin0, d, LOW);
        t1 = t.oscillate(pin1, d << 1, LOW);
        t2 = t.oscillate(pin2, d << 2, LOW);
        t3 = t.oscillate(pin3, d << 3, LOW);
    }
    else
    {
        t0 = t.oscillate(pin0, d, HIGH);
        t1 = t.oscillate(pin1, d << 1, HIGH);
        t2 = t.oscillate(pin2, d << 2, HIGH);
        t3 = t.oscillate(pin3, d << 3, HIGH);
    }
}

//// YOUR CODE ENDS HERE

// For Arduino 1.0 and earlier
// #if defined(ARDUINO) && ARDUINO >= 100
// #include "Arduino.h"
```

Conclusions:

We have successfully created a 4 bit Up-Down Counter that first goes UP from 0(0000) to 15(1111), then goes down from 15 to 0, then goes UP , and this cycle repeats until the simulation stops.

Link for Tinkercad Simulation:

https://www.tinkercad.com/things/cmxmvYNdGEq-5c-up-down-counter/editel?sharecode=9MDBlm3a5C2l6kvz_uuWUzAbCoA-5kuxHecXMqexdB0