DSM LAB REPORT – 5

Macharla Harish

2020102062

Objective:

To build an RS latch, a JK Master-Slave Flip-Flop, and a 4-bit Up-Down Counter in Tinker cad.

Part A: SR Latch

Objective:

To Assemble a NOR latch using two NOR gates

Components Required:

*****Bread Board

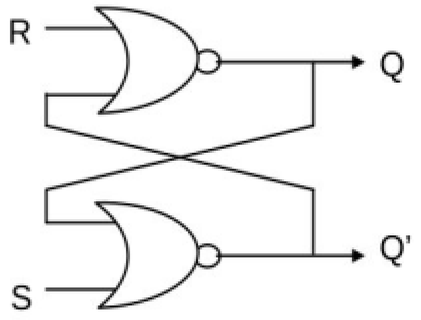
*Arduino

*Logic Nor gates

*Push Button

*Led's, Resistors and connecting wires.

Reference Circuit:



Procedure:

1) Construct the circuit with the components mentioned above on Bread Board as shown in the figure.

2) Give the inputs R and S through Arduino.

3)Connect the outputs Q and Q' to the Led's with resistors using connecting wires.

4)Note down the readings.

Observations:

S	R	Q	Q'
0	1	0	1
0	0	0	0
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	1
1	0	1	0
1	1	0	0
0	0	0	1
0	1	0	1
1	1	0	0
0	0	0	1

- From my observations latch works properly until both the inputs were given 1 (S=R=1) because it contradicts outputs that is the number and the complement of the number should be opposite but in this case both are equal.
- Hence it leads to forbidden state where it loses it previous states.

Link for Tinker cad:

https://www.tinkercad.com/things/4Mij0z8XqtK-2020102062lab5part1/editel?sharecode=a2N433xRKdCb8q0q37iN1cLZkN51qa3Vxiid9V08Lqg

Conclusion:

Truth Table for SR Latch:

Boolean exp:

$$Q(n+1) = [\, R + (S + Q(n))' \,]'$$

$$= R' . (Q(n)+s)$$

S	R	Q(n+1) state
0	**0**	**Q(n) --> No change**
1	**0**	**1 --> Set**
0	**1**	**0 --> Reset**
1	**1**	**0 --> Forbidden**

Part B: JK Master-Slave Flip-Flop

Objective:

To Assemble a JK Master-Slave Flip-Flop.

Components Required:

*Bread Board

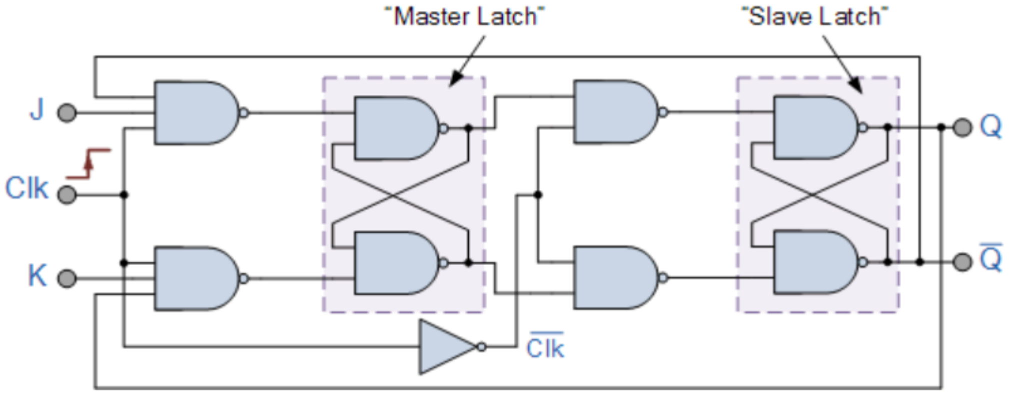
*Arduino

*Logic Triple Nand gate, Two Quad Nand gates, Hex converter.

*Led’s, Resistors and connecting wires.

*Push Buttons

Reference circuit:



master slave JK flip flop

Procedure:

1) Construct the circuit with the components mentioned above on Bread Board as shown in the figure.

2) Give the inputs J, K and clock through Arduino.

3)Connect the outputs Q and Q' to the Led's with resistors using connecting wires.

4)Note down the readings.

Observations:

J	K	Q	Q'
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	1	0
0	0	0	1
1	0	1	0
1	1	0	1
0	0	0	1
0	1	0	1
1	1	1	0

0	**0**	**1**	**0**

Link for Tinker Cad:

https://www.tinkercad.com/things/dxKsmNH09wj-2020102062lab5partb/editel?sharecode=9RSBbateS7_eFFb2QpOmmgnnfRt1AdkbV-rj1JOFuvI

Conclusion:

- We can theoretically understand with the behavior of the JK Master-Slave Flipflop with help of Circuit made on Tinker cad.

JK master slave flip flop function table:

J	**K**	**ACTION**	Q_{n+1}
0	**0**	**Hold**	Q_n
0	**1**	**Clear**	**0**
1	**0**	**Set**	**1**
1	**1**	**Toggles**	Q'_n

Part C: 4-bit Up-Down Counter

Objective:

To study the working of a counter.

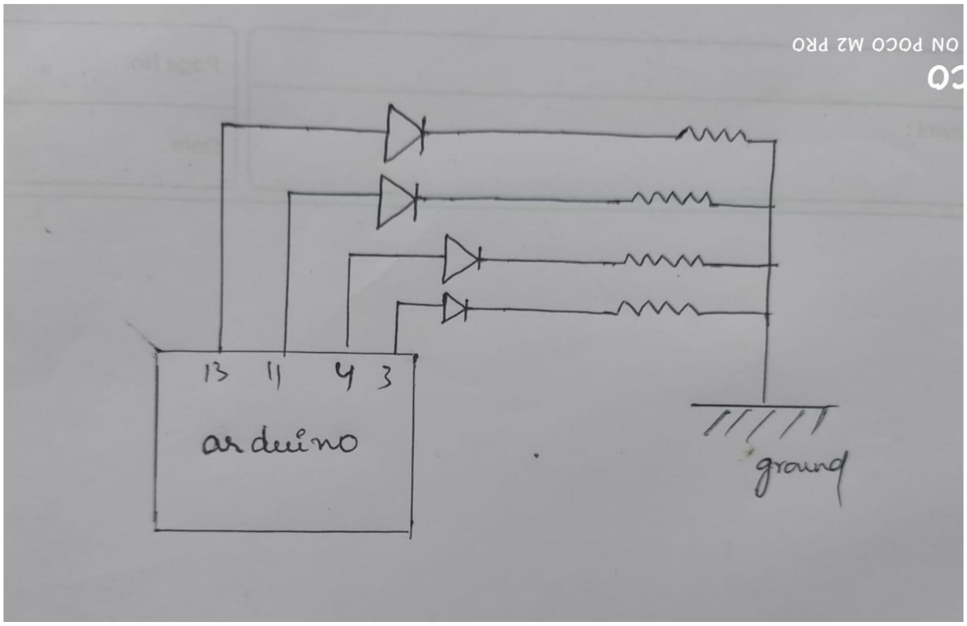
Components Required:

*Bread Board

*Connecting Wires, Led’s and Resistors

*Arduino

Reference Circuit:



Procedure:

- Using the Timer library, implement a 4-bit counter. The bit outputs of the 4-bit ripple counter will be represented by LEDs. (One LED for each bit)
- Initialize a Timer t and use t. oscillate function to toggle the pin values in a pre-defined time period (each pin will have a different time period.
- Down counter: once the ripple counter reaches 15 (1111), make it go down to 0. You will need to stop existing timers using t.stop. You can 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 opposite direction.
- Simulate the tinker Cad project and observe the LEDs.

Observations:

The ripple counter 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 is stopped.

Link for TinkerCad:

https://www.tinkercad.com/things/enbX0r9sZm2-copy-of-timer-starter-code/editel?sharecode=_aNC047fTY87zytC8uGbLNPUKrlciTg0w4ky16kYmpQ

Conclusion:

Thus, we have implemented a 4-bit UP DOWN counter using tinkercad.