***On my honor, I pledge that I have not violated the provisions of the NJIT Student Honor Code***

STUDENT TEAM N**o** \_\_2\_\_

Name Signature

…Arsh Bhamla………… **ARSH BHAMLA~~\_\_\_~~**\_ …Pranay KC………….. \_**\_PRANAY RAJ KC\_\_\_\_\_** ..Shreedhar Rajendran. **\_\_Shreedhar Rajendran—**

..Christopher Karkenny. \_\_**Christopher Karkenny**\_

Experiment performed on date \_02/ 7 / 2025\_\_

Report submitted on date \_\_\_\_\_\_\_\_\_

Returned for corrections on date \_\_\_\_\_\_\_\_\_ Grade \_\_\_\_\_\_

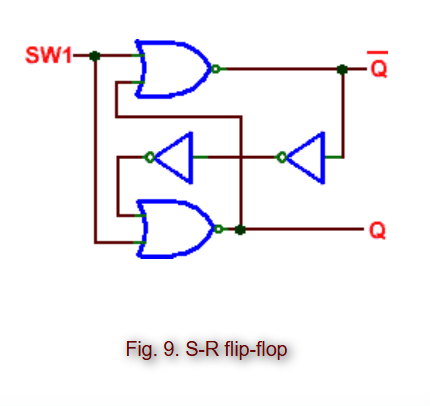
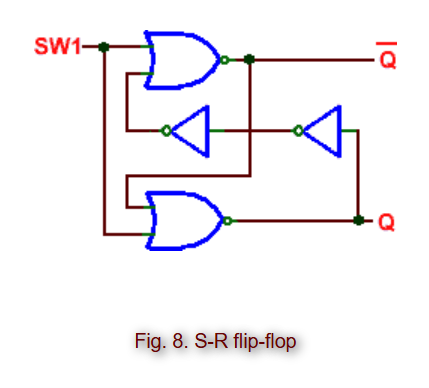
Returned after corrections on date \_\_\_\_\_\_\_\_\_ Grade \_\_\_\_\_\_

**Introduction**

In this lab, we will further practice sequential logic circuits, checking out the basic components of memory in digital systems. Unlike combinational circuits that only depend on current inputs, sequential logic circuits have memory cells so they can store and use past inputs to figure out current outputs. We will focus on building and analyzing different types of flip-flops, like S-R, D, and J-K flip-flops which are important in these circuits. By looking at how they behave under different conditions, we will understand key concepts like race conditions and setup times. We also will design a sequence detector to show how sequential logic can be used in finite state machines. This lab will give valuable insights into how memory elements work and why they are crucial in digital circuit design.

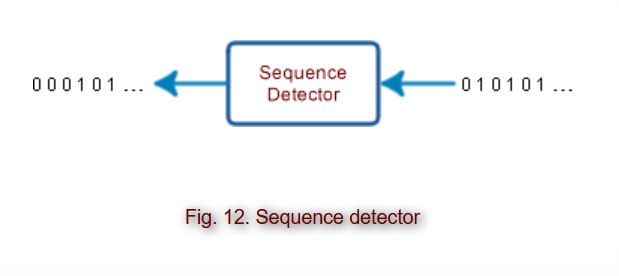
**Procedure**

**3.1 Flip-Flops**



Figures shown provided from [here](https://ecelabs.njit.edu/ece294/lab2.php).

**3.2 Sequence Detector**



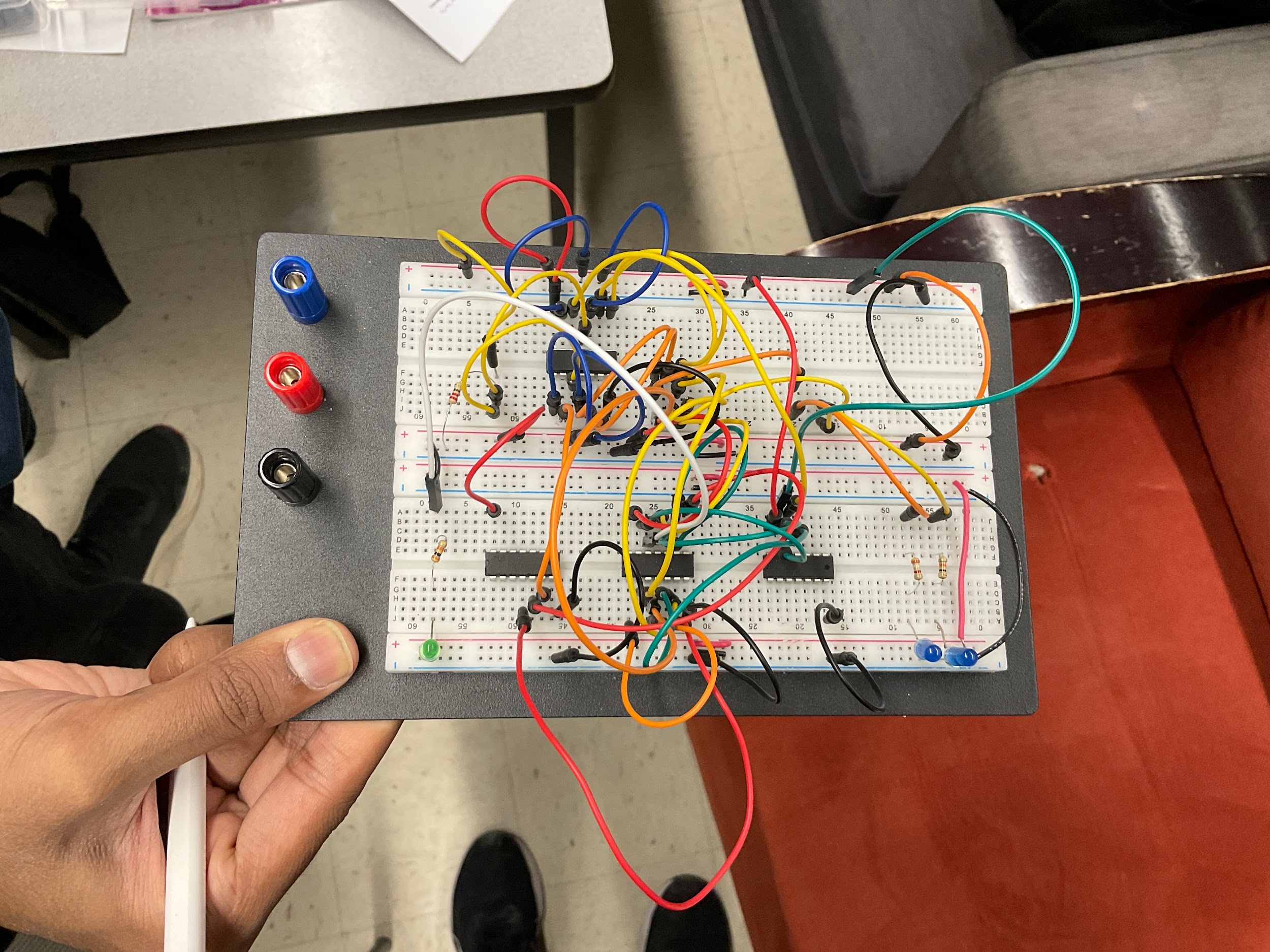
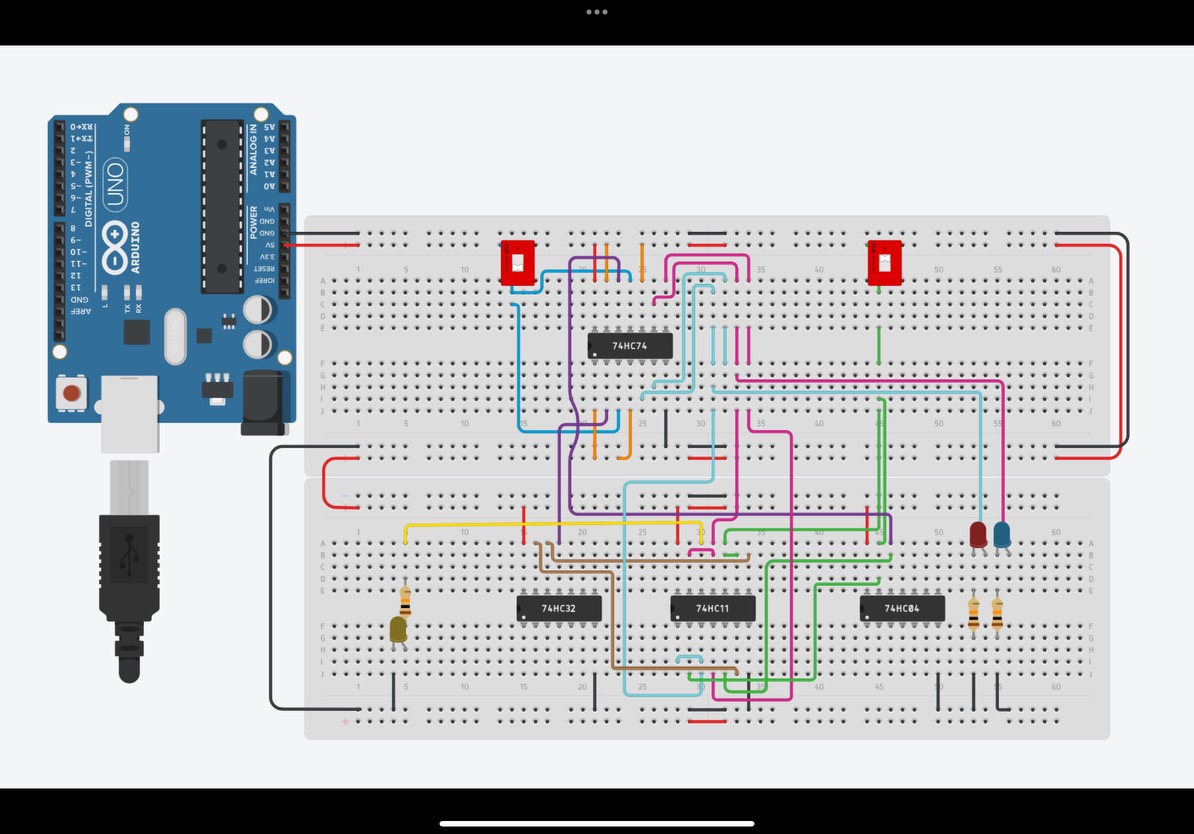
Figures shown provided from [here](https://ecelabs.njit.edu/ece294/lab2.php).

**Data and Calculations**

**3.1 Flip-Flops**

| D Input | Q |
| --- | --- |
| 0 | 0 |
| 1 | 1 |

**3.2 Sequence Detector**



**Discussion**

**3.1 Flip-Flops**

In the first part of the lab, we built an S-R flip-flop using NOR gates and checked its truth table against Table 1. The S-R flip flop has two inputs (Set and Reset) and two outputs, Q and Q’ where Set makes Q equal to 0 and both inputs at 0 hold the previous state. We tweaked the flip flop according to Figure 8 and connected the data switch SW1 to a pulse generator so we could watch the output change when the input switched from 1 to 0. Then we made more changes following Figure 9, and this showed a race condition which happens when almost simultaneous input changes lead to instability due to interval propagation delays. Race conditions occur when outputs rely on exact timing of events, causing unpredictable results in feedback loops. Next, we connected a 7474 edge-triggered D flip-flop as shown in FIgure 10 and set SW1 to logic 0 and 1 and observed that data transfer to the output only happened on the positive lock edge. We added four 7404 inverters in series to the CLK line following Figure 11 which made sure that the clock signal reached the flip-flop after the data input had settled. By doing so, the flip-flop was able to function correctly, capturing the data present at D on the clock’s rising edge.

**3.2 Sequence Detector**

In the second experiment of the lab, we designed and constructed a one input, one output sequence detector to detect the sequence 0101 using D flip-flops. We began by designed the state diagram with states S0 (initial state), S1(0 detected), S2(01 detected), S3(010 detected), and S4 (0101 detected) with state transitions based on input X and output Z that is 1 only when the sequence 0101 is detected. The state transition table had the assigned states: S0=00, S1=01, S2=10, S3=11.

**Conclusions**

In this lab we learned how sequential logic circuits using flip-flops work and how to design them. For experiment 3.1 we gained experience with the basics of S-R flip-flops and edge-triggered D- flip-flops worked and understood the timing requirements as well as how race conditions affect circuit stability. By making the appropriate changes we learned how setup time issues can affect flip-flop behavior. For experiment 3.2 we took it further by designing and building a sequence detector for the sequence 0101 using D flip-flops. We created a state diagram, derived state transition and output equations, and implemented our design. By making and testing these circuits, we gained hands-on experience in designing efficient sequential circuits, which is essential for more advanced digital design applications.