Lab Report: Experiment 7

EE24BTECH11003 : Akshara Sarma Chennubhatla
EE24BTECH11005 : Arjun Pavanje

March 27, 2025

Experiment:
Designing an asynchronous
mod 7 counter using
T Flip-Flops

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Bachelor of Technology

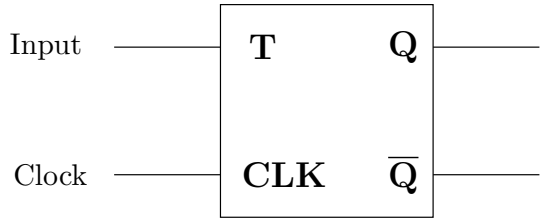
Department of Electrical Engineering

1 Introduction

This experiment focuses on designing and implementing a Mod-7 asynchronous counter using T flip-flops, using clock from an Arduino.

2 Theory

2.1 T Flip-Flop



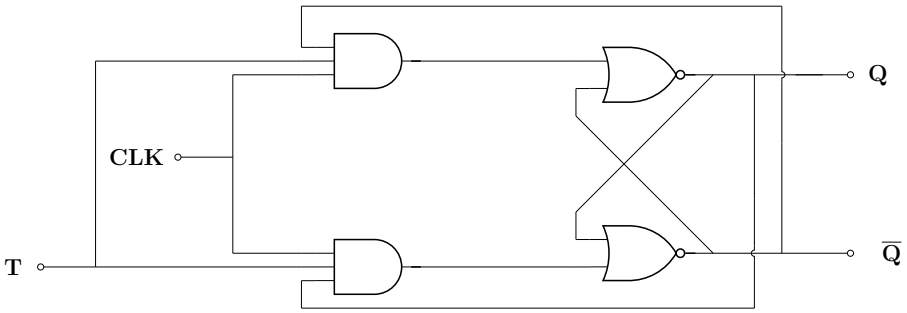
The Toggle Flip-Flop also known as T Flip-Flop is a type of Flip-Flop in which the output takes in the input value if the state of T is high (T = 0) and takes the value of the complement of the input if T is low (T = 1). This happens when the clock is high. When the clock is low, the output retains its previous state.

$$Q_{n+1} = \begin{cases} Q_n & , T = 0 \\ \\ \overline{Q_n} & , T = 1 \end{cases}$$

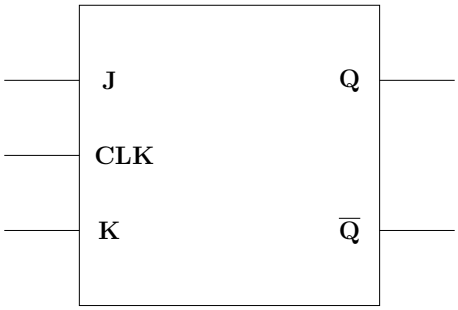
The Truth table for the Flip-Flop is,

T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

The internal Circuit diagram of the T Flip-Flop is given below,



2.2 JK Flip-Flop



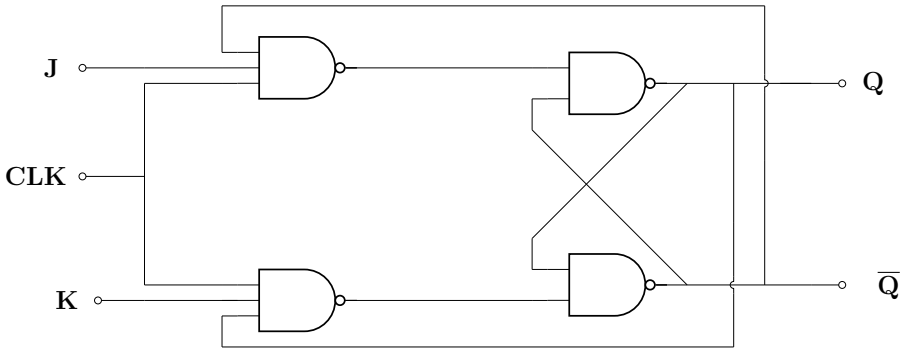
The JK Flip-Flop works on the principle that the output is equal to the input when both J and K are low (J = K = 0). The output takes the value of the complement of the input if both J and K are high (J = K = 1). In any other case, the output takes the value of J regardless of K. This happens when the clock is high. When the clock is low, the output retains its previous state.

$$Q_{n+1} = \begin{cases} Q_n & , J = 0, K = 0 \\ \overline{Q_n} & , J = 1, K = 1 \\ 0 & , J = 0, K = 1 \\ 1 & , J = 1, K = 0 \end{cases}$$

The truth table of the JK Flip-Flop is given below,

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

The internal circuit diagram of the JK Flip-Flop is as below,



2.3 JK to T Flip-Flop Conversion

Since JK flip-flops were provided instead of T flip-flops, conversion was necessary. Converting a JK flip-flop to a T flip-flop can be done by simply shorting J and K ports of JK flip-flops.

T	Q_n	Q_{n+1}	J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	X
1	1	0	X	1

Writing Karnaugh-map for J

T \ Q_n	0	1
0	0	X
1	1	X

We get,

$$J = T$$

Now writing karnaugh-map for K,

T \ Q_n	0	1
0	X	0
1	X	1

We get,

$$K = T$$

2.4 Asynchronous Counter

In an asynchronous counter, the clock input of each flip-flop (except the first) is driven by the output of the previous flip-flop. This creates a ripple effect as the signal propagates through the counter, hence the alternative name "ripple counter".

3 Materials Required

1. T flip-flops (or JK flip-flops LM7476 (or similar))
2. NAND gate LM7410 (or similar)
3. LED
4. Arduino (for clock)

5. Oscilloscope
6. Wires
7. Breadboard

4 Procedure

4.1 Converting JK Flip-Flops to T Flip-Flops

Short the J and K pins of JK flip-flops as mentioned above to convert it into a T flip-flop

4.2 Counter Design

1. The PRESET pin (J) of all flip-flops was connected to *HIGH*.
2. The output of the first flip-flop was connected to the clock input of the second flip-flop.
3. The output of the second flip-flop was connected to the clock input of the third flip-flop.
4. The outputs of all three flip-flops were connected to LEDs through resistors indication. This gives a counter that goes from 0 to 7 (for the 0 to 7 counter to work, set $\overline{\mathbf{CLR}}$ to *HIGH*)

4.3 Reset Mechanism

1. The outputs of all three flip-flops were connected to the inputs (1A, 1B, 1C) of the LM7410 NAND gate.
2. The output of the NAND gate (1Y) was connected to the $\overline{\mathbf{CLR}}$ pins of all flip-flops.

This configuration ensures that when the counter reaches the state 111 (decimal 7), the NAND gate outputs a LOW signal, which clears all flip-flops, resetting the counter to 000" (decimal 0).

4.4 Logic

The output of one of the first Flip-Flop is connected to the clock of the second flip-flop. Since both J and K are 1, i.e., T = 1, output of first flip-flop will take the value of complement of its previous output at every falling edge. Similarly, for the second and the third flip-flop. But after 6 cycles of the clock, on the seventh cycle when all the 3 outputs are 1, the NAND gate to which these 3 are given as inputs takes the value 0 which is then given to the $\overline{\mathbf{CLR}}$ of the flip-flops which then sets all the outputs to 0 regardless of the inputs effectively

looping it back to 0. So here, all the output signals suddenly drop to 0 from their previous state. So the graph of the outputs looks like the following plot. The period of every output signal is 7 times the time period of the clock.

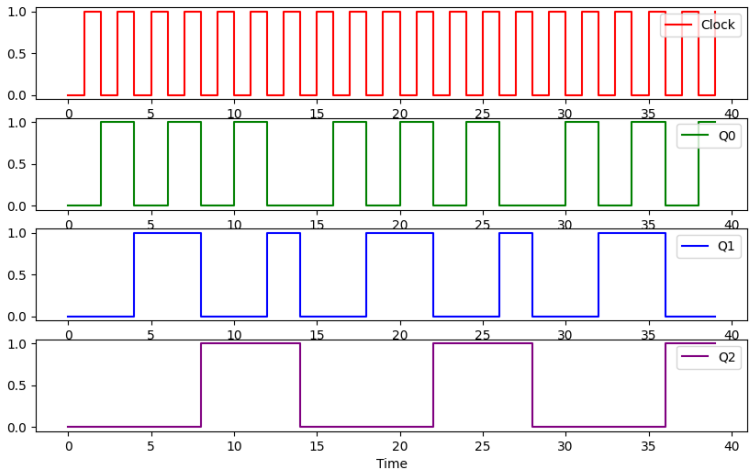


Figure 1: Clock and Output Signals

4.5 Clock Signal Generation

The Arduino code used for generating a clock signal of period $1s$ is as follows:

```
void setup () {
    pinMode(13, OUTPUT);
}

void loop () {
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);
}
```

4.6 Circuit Image

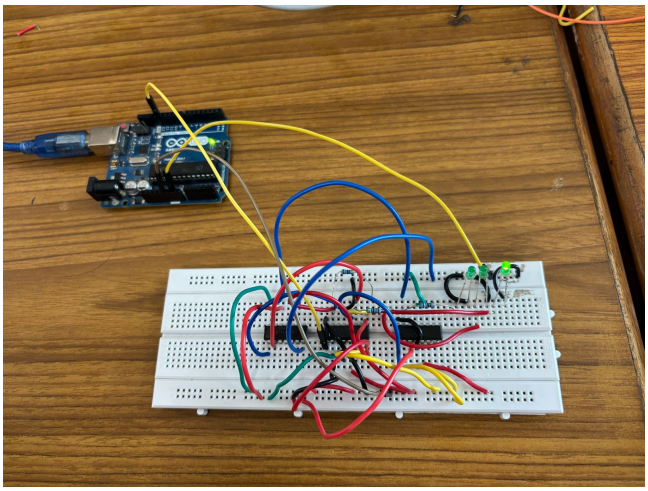
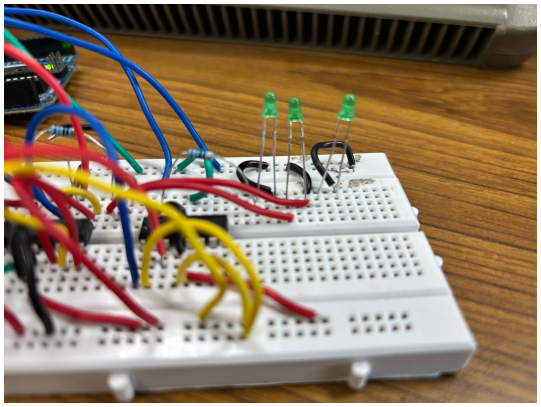


Figure 2: Circuit

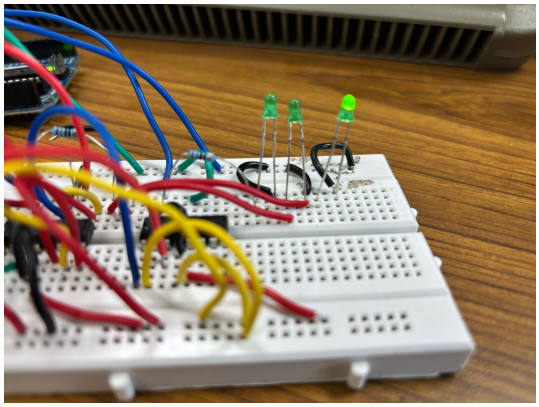
5 Results and Observations

5.1 Counter

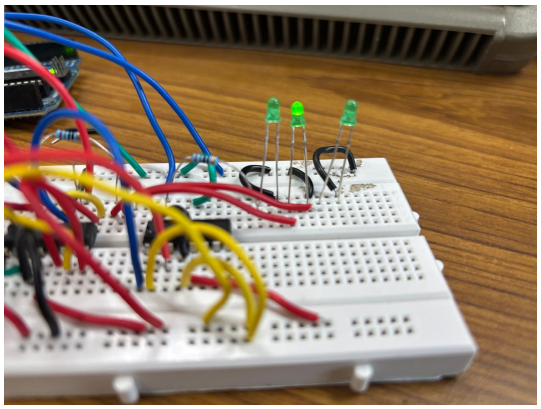
(a) 0



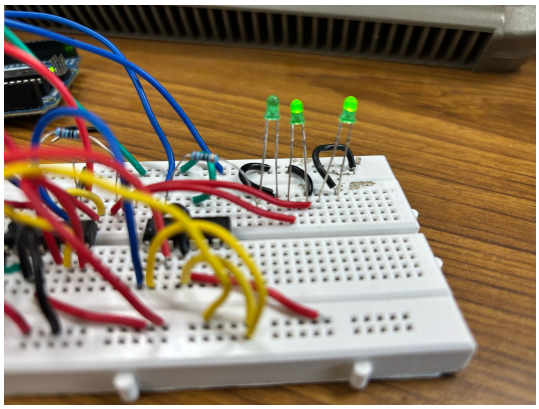
(b) 1



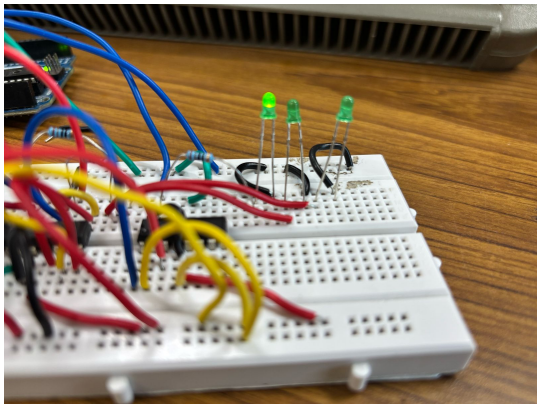
(a) 2



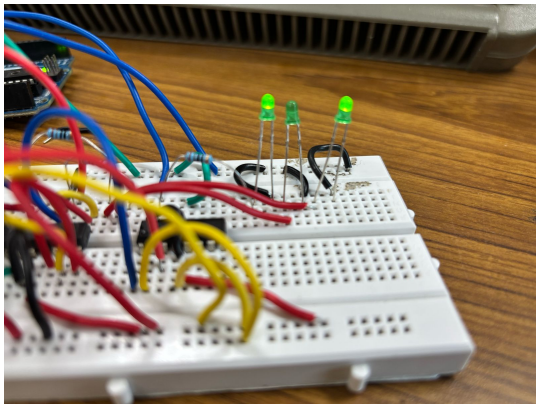
(b) 3



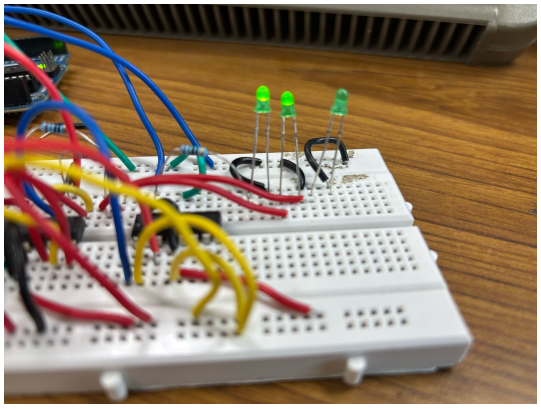
(c) 4



(d) 5



(e) 6



Count	Q_3	Q_2	Q_1	Decimal
0	0	0	0	0
1	0	0	1	1
2	0	1	0	2
3	0	1	1	3
4	1	0	0	4
5	1	0	1	5
6	1	1	0	6
7	0	0	0	0

Table 1: Counting sequence of the Mod-7 counter

5.2 Oscilloscope Readings

Figure 5: Oscilloscope Reading-1

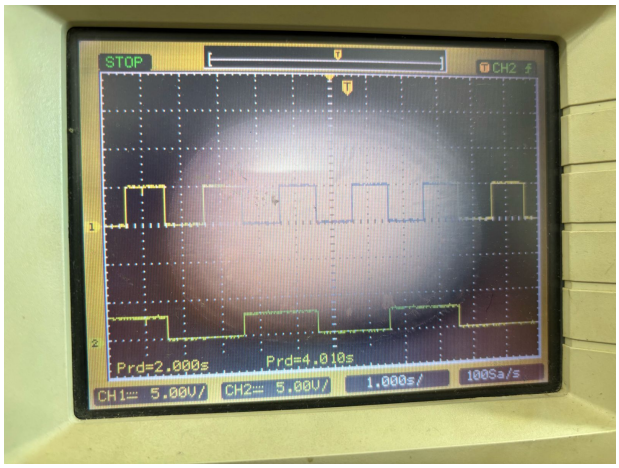


Figure 6: Circuit-1

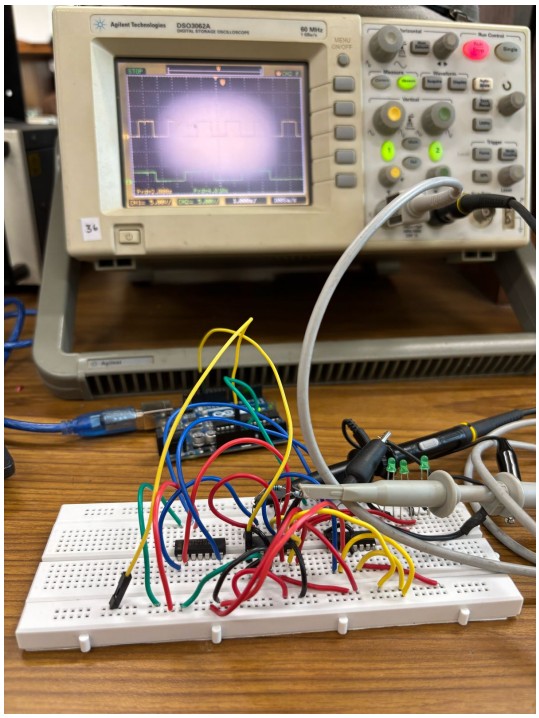


Figure 7: Oscilloscope Reading-2

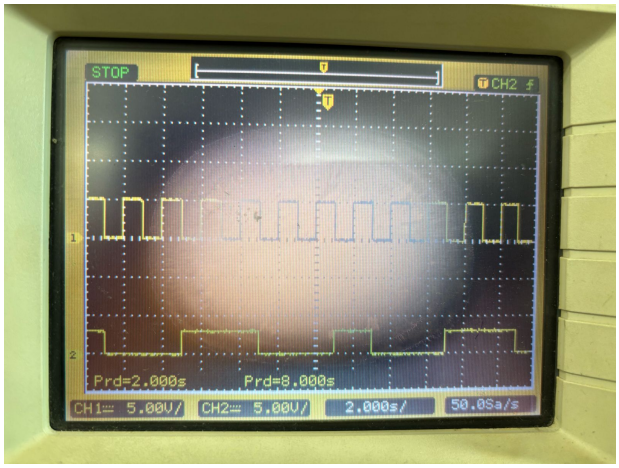


Figure 8: Circuit-2

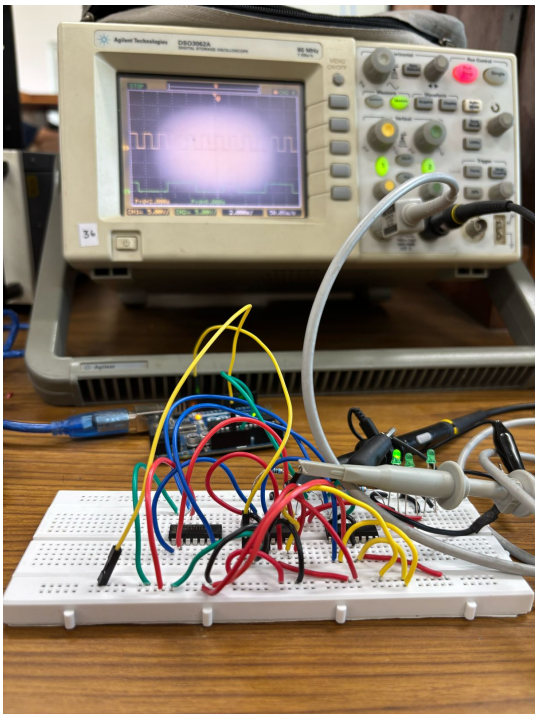


Figure 9: Oscilloscope Reading-3

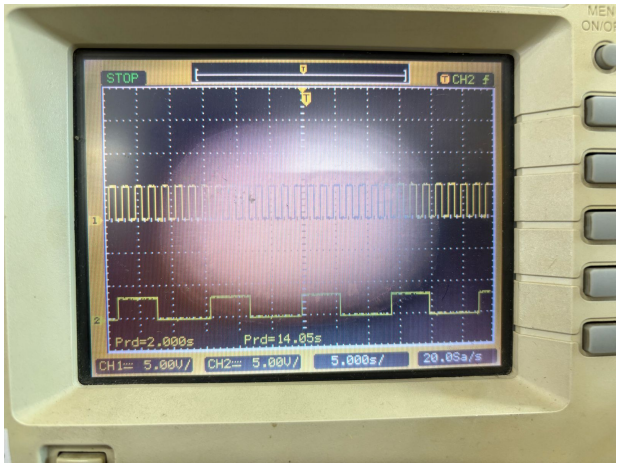
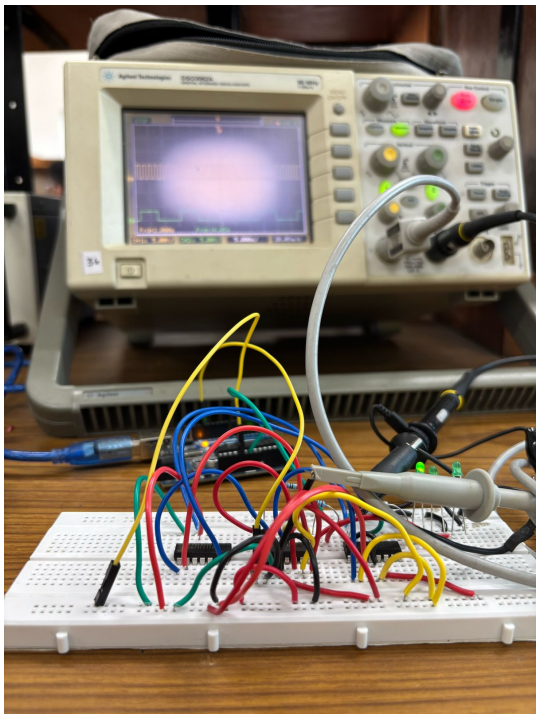


Figure 10: Circuit-3



6 Conclusion

The experiment successfully demonstrated the design and implementation of a Mod-7 asynchronous counter using T flip-flops converted from JK flip-flops, using Arduino as a clock.