Frequency Counter Using Arduino

I. Objective:

The project aims to design a digital frequency counter circuit using Arduino UNO and 555 Timer IC. The counter is used to measure the frequency of a pulse without the need for an oscilloscope.

II. Introduction:

- Frequency Counter is an electronic device or component used to measure the frequency of a signal. In the case of a repetitive electronic signal, a frequency counter measures the number of pulses in that signal.
- Generally an oscilloscope is used to depict the signal, calculate the signal's time period, and finally convert it to calculate the signal's frequency. But oscilloscopes are very expensive, and everyone cannot afford them.
- Hence, a simple Digital Frequency Counter can be built, which might come in handy to measure the frequency of a clock signal, for example.

III. Components:

- Arduino UNO
- 16 X 2 LCD display
- Prototyping board
- Connecting wire
- Power supply
- NE555 timer IC
- 10 KΩ potentiometer
- 100 nF capacitor
- 1 µF / 16V electrolytic capacitor

IV. Component Description:

- *Arduino UNO:* The ATmega328P microcontroller-based Arduino UNO is the central part of the project. It captures the period of the incoming signal and calculates the signal's frequency.
- *NE555 Timer IC:* The 555 Timer IC is used as a pulse generator, i.e., it works in astable mode.
- *16×2 LCD:* The 16×2 LCD module displays critical information like the calculated frequency of the signal or any other additional custom messages.

V. Circuit Design:

The design can be divided into two parts:

- **Arduino**:

  - Here, the processing of the signal's information takes place. The project consists of an Arduino UNO board and a 16×2 LCD. Pins 1 and 2 of the LCD (Vss and Vdd) are connected to the ground and 5V supply. Pin 3

(Vee), which is used to adjust the display's contrast, is connected to a potentiometer.

- Pins 4 and 6 (RS and E) of the LCD are connected to digital I/O pins 2 and 3 of the Arduino. Pin 5 (RW) of the LCD is connected to the ground.

- Pins 11 to 14 (D4 to D7), i.e., the data pins of the LCD are connected to the digital I/O pins 4 to 7 of Arduino. Pins 15 and 16 of the LCD are connected to the ground and 5V (Pin 16 to 5V through a 1KΩ resistor).

- **Signal Generator:**

  - Here, the signal whose frequency is to be measured is generated.

  - A 555 Timer is used to generate a pulse in this project. Hence, it is operated in astable mode. The connections for this model are as follows.

    - Pins 4 and 8 (Reset and Vcc) are connected to a 5V Supply. Pin 1 (GND) is connected to the ground. Pins 2 and 6 (Trigger and Threshold) are shorted.

    - A 10KΩ Potentiometer is connected between the power supply and Pin 7 (Discharge). The wiper terminal of the potentiometer is connected to Pin 7.

    - Similarly, another 10KΩ Potentiometer is connected between pins 7 and 6. This time, the wiper of the potentiometer is connected to Pin 6.

    - A 100nF capacitor is connected between pin 6 and the ground. By selecting this resistor, the frequency of the output signal, which can be taken from Pin 3 (output), would be in the range of 480 Hz to 48 kHz.

    - Optionally, a 1 μF capacitor can be connected, using which the frequency of the generated signal would be in the range of 50 Hz to 4.8 kHz.

VI. <u>Circuit Diagram:</u>

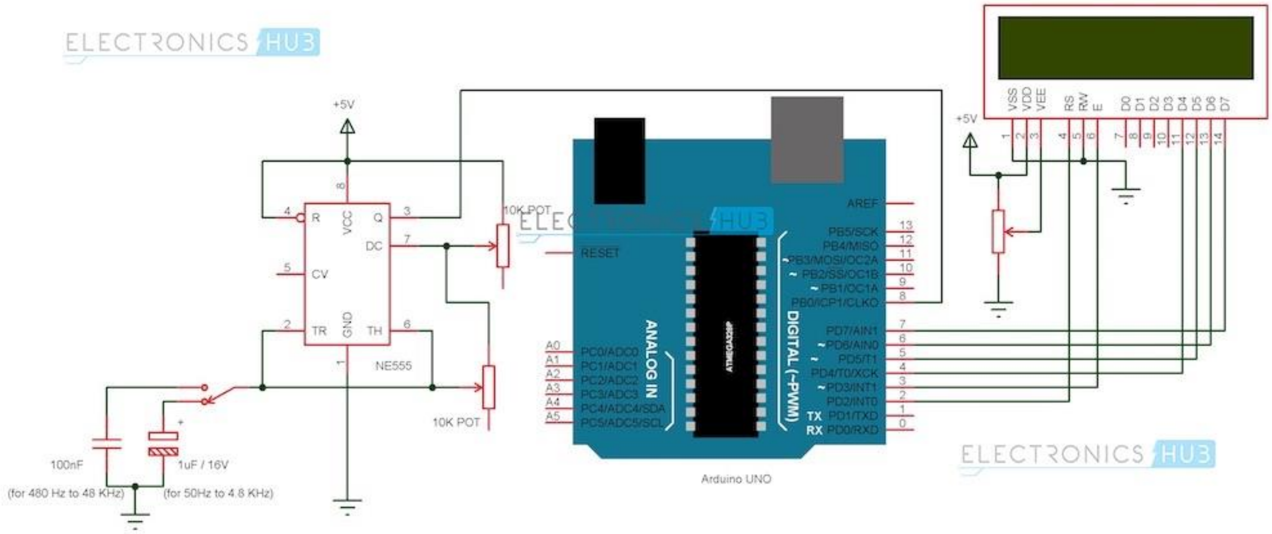


Fig. 1: Circuit diagram of the frequency counter

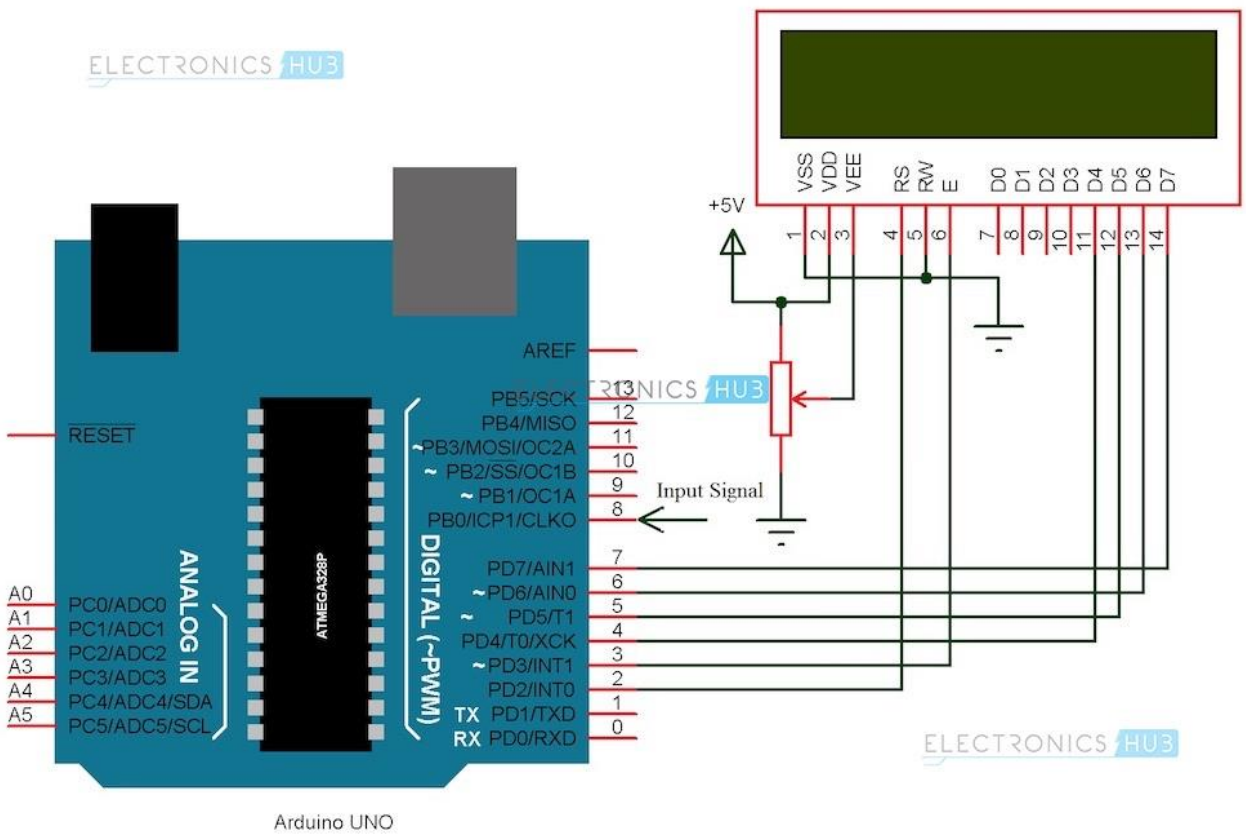


Fig.2: Circuit diagram of the Arduino part

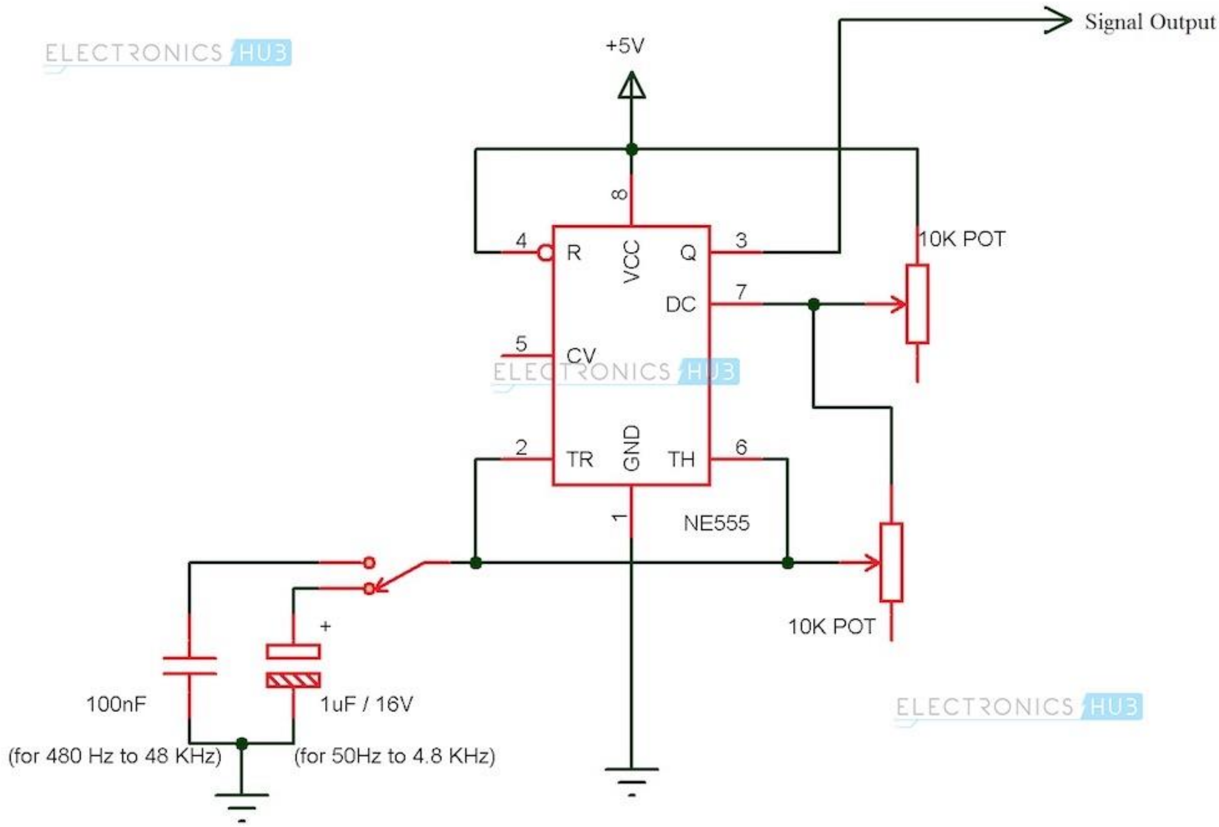


Fig.3: Circuit diagram of the signal generator

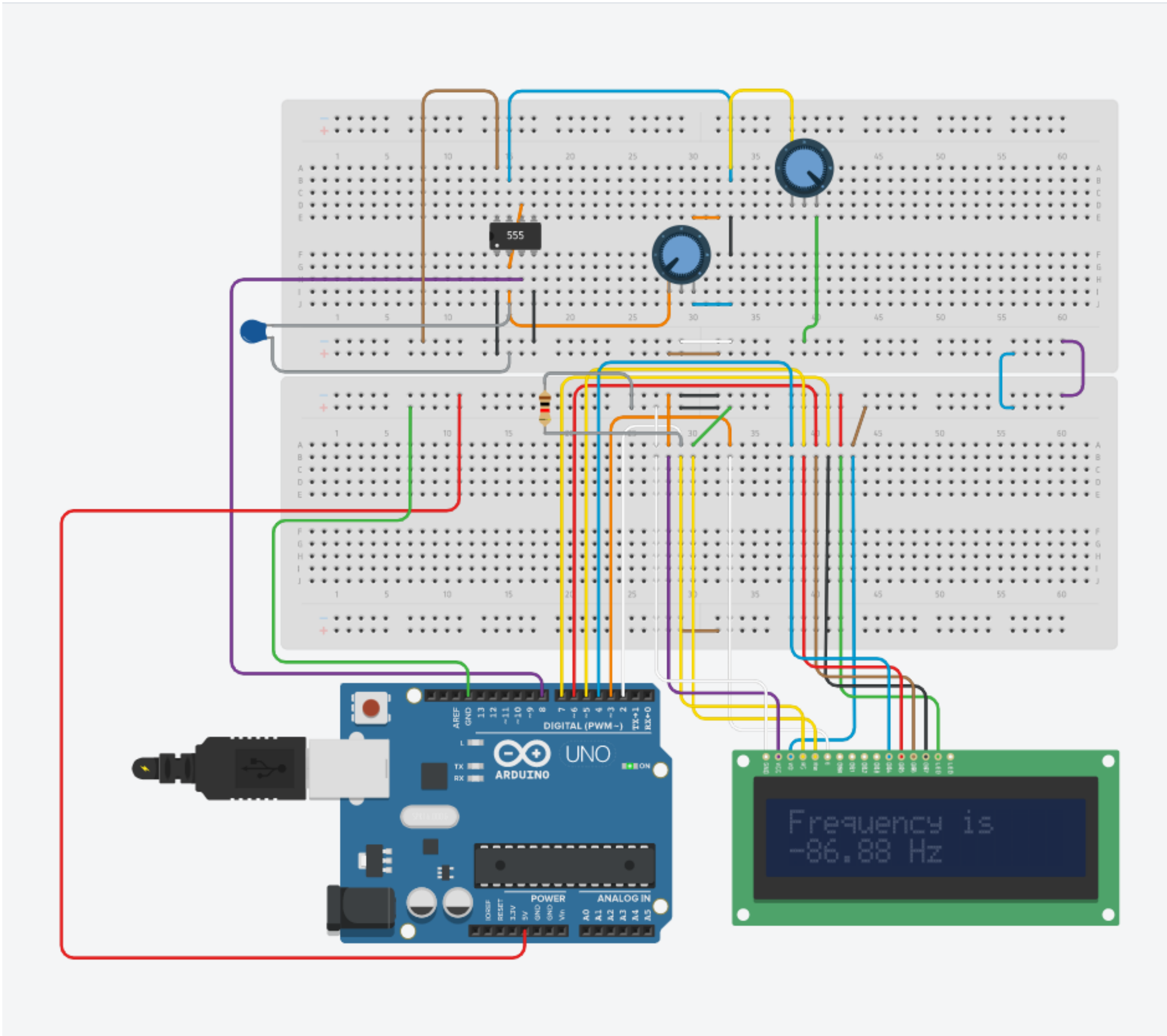


Fig 4: Simulation of the frequency counter on Tinkercad

VII. Working Principle:

- As mentioned earlier, the 555 Timer IC is configured to operate in astable mode. Hence, the 555 Timer IC output (or the signal generator circuit) is a pulse with variable frequency (varied using a potentiometer). This pulse is given as an input signal to the Arduino UNO at one of its digital I/O pins.

- In the Arduino, a function called "pulseIn ();" is implemented. The function is used to read either LOW or HIGH pulse on a digital I/O pin and return the length of the pulse in microseconds.

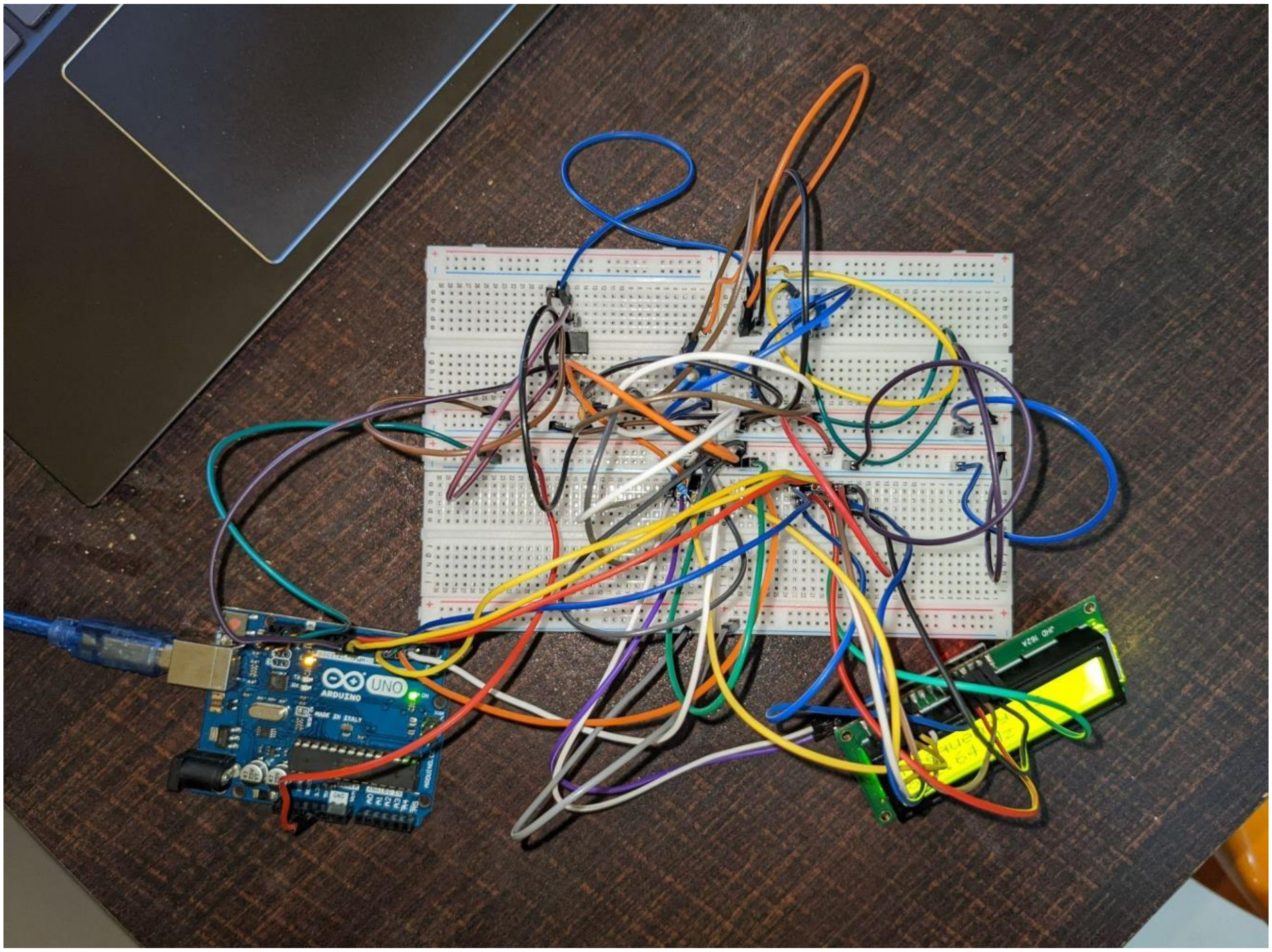
- For example, if the pulseIn function reads a HIGH pulse on a pin, it waits for the pin to go HIGH. Once the pin goes HIGH, it starts the timer and runs until the pin goes LOW. The duration (in microseconds) of this HIGH pulse is then returned.

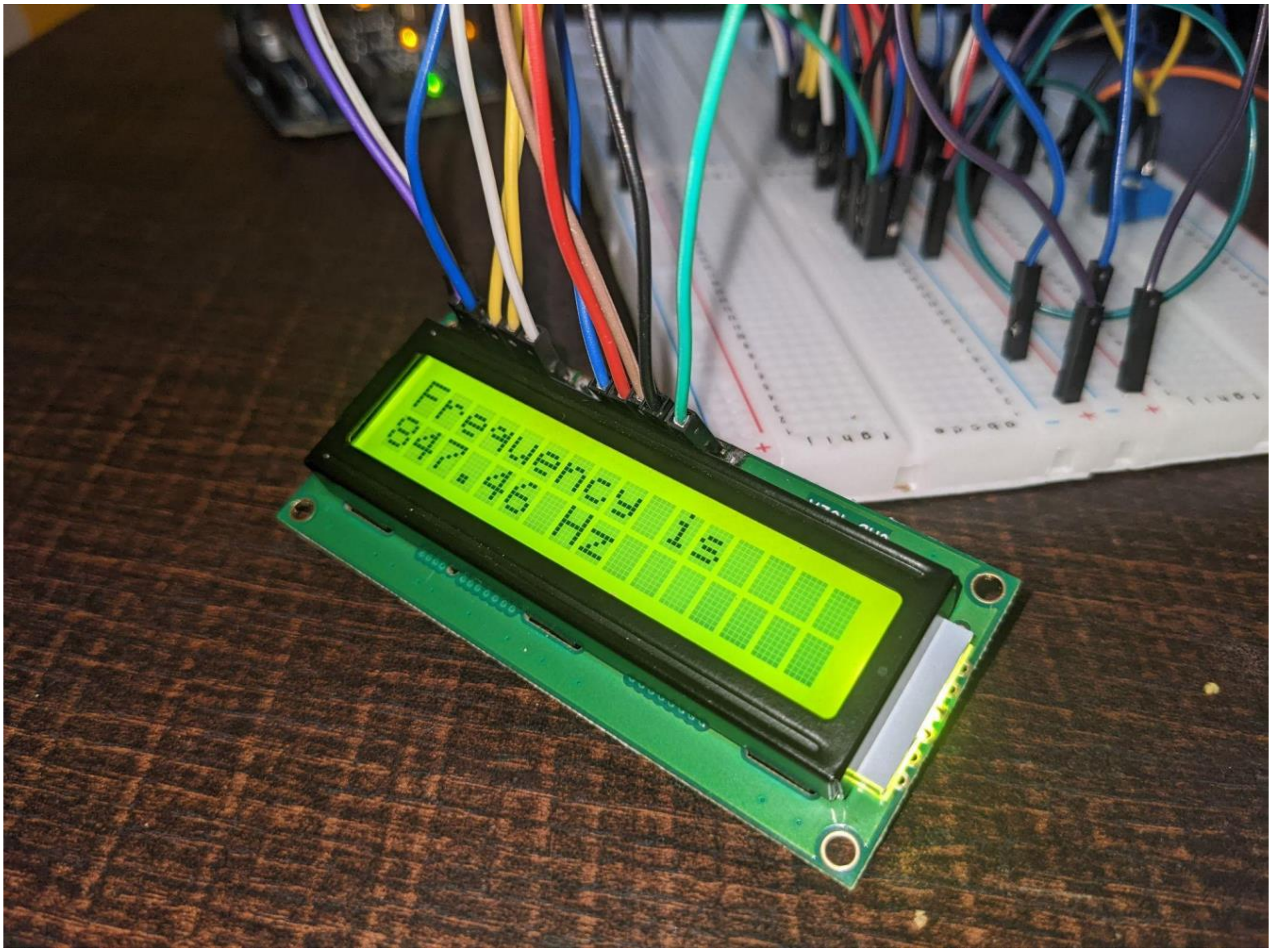
- The duration of the HIGH pulse and LOW pulse has been calculated, and by adding them together, the input signal period is obtained. The inverse of this value gives the signal frequency displayed on the LCD.

VIII. **Code**:

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

const int pulsePin = 8;

int pulseHigh;
int pulseLow;
float pulseTotal;
float frequency;

void setup()
{
  pinMode(pulsePin,INPUT);
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print("FrequencyCounter");
  lcd.setCursor(0,1);
  lcd.print("Group6 ");
  delay(5000);
}
void loop()
{
  lcd.setCursor(0,0);
  lcd.print("Frequency is    ");
  lcd.setCursor(0,1);
  lcd.print("                ");

  pulseHigh = pulseIn(pulsePin,HIGH);
  pulseLow = pulseIn(pulsePin,LOW);

  pulseTotal = pulseHigh + pulseLow;
  frequency=1000/pulseTotal;

  lcd.setCursor(0,1);
  lcd.print(frequency);
  lcd.print(" kHz");
  delay(500);
}
```

IX. **Output**:





X. Future Scope:

- Multiple ranges of frequencies can be measured by selecting suitable components.
- Frequencies of all types of test signals can be calculated by adding a Schmitt Trigger between the generated signal and Arduino.

XI. Conclusion:

In this project, the NE555 timer IC is used as a digital pulse generator and the output pin of the timer is connected to a potentiometer so that the range of signals generated by the timer can be varied. This output signal is fed into ArduinoUNO to count the number of pulses and calculate the frequency of this signal.

XII. References:

[1] Szplet, Ryszard & Jachna, Zbigniew & Rozyc, K & Kalisz, J. (2010). High precision time and frequency counter for mobile applications. WSEAS Transactions on Circuits and Systems. 9.

[2] Peng, Jin-Long & Liu, Tze-An & Shu, R.-H. (2008). Optical frequency counter based on two mode-locked fiber laser combs. Appl. Phys. B92. 513-518. 10.1007/s00340-008-3111-6.

[3] Szplet R, Kwiatkowski P, Różyc K, Jachna Z, Sondej T. Picosecond-precision multichannel autonomous time and frequency counter. Rev Sci Instrum. 2017 Dec;88(12):125101. DOI: 10.1063/1.4997244. PMID: 29289186.

[4] Xiulian Yin, Xuejuan Zhang, Miaomiao Wan, Xiuli Duan, Qinghong You, Jinfeng Zhang, Songlin Li, Intensification of biodiesel production using dual-frequency counter-current pulsed ultrasound, Ultrasonics Sonochemistry, Volume 37, 2017, Pages 136-143, ISSN 1350-4177, https://doi.org/10.1016/j.ultsonch.2016.12.036.(https://www.sciencedirect.com/science/article/pii/S1350417716304758)

[5] The PIC-based Digital Frequency Counter Wai Phyo Aung Manuscript was implemented on August 15, 2008. This work was supported by the Ministry of Science and Technology, Union of Myanmar. Wai Phyo Aung is with the Mandalay Technological University, Mandalay, Myanmar (phone: 095-2-88704(Electronic Engineering Department), fax: 095-2-88702 (Office, MTU), e-mail: aungwp@gmail.com).

[6] Yaghoubi, Forough & Catovic, Armin & Gusmao, Arthur & Pieczkowski, Jan & Boros, Peter. (2021). Traffic Flow Estimation using LTE Radio Frequency Counters and Machine Learning.

[7] M. C. McCoy, C. Isert, D. Jackson and J. Naber, "A frequency counter-based analog-to-digital converter for an RFID telemetry system," 2007 50th Midwest Symposium on Circuits and Systems, 2007, pp. 1384-1387, DOI: 10.1109/MWSCAS.2007.4488806.

[8] B. El-Asir and E. H. Said, "A low-frequency counter using time-discriminant connectionist systems," in IEEE Transactions on Instrumentation and Measurement, vol. 53, no. 2, pp. 493-497, April 2004, DOI: 10.1109/TIM.2004.823300.

[9] Wijayanto VR, Sakti SP. Design of Dual Edge 0.5 Hz Precision Frequency Counter for QCM Sensor. AMM 2015;771:29–32. https://doi.org/10.4028/www.scientific.net/amm.771.29.

[10] M. A. Syahbana, D. J. H. D. Santjojo, and S. P. Sakti, "High-resolution multiple channel frequency counters using spartan-3E FPGA," 2016 International Seminar on Sensors, Instrumentation, Measurement and Metrology (ISSIMM), 2016, pp. 111-114, DOI: 10.1109/ISSIMM.2016.7803734.