Development of a Microcontroller Based Capacitive Sensor for Non-Contact Measurement of Level of Conductive Liquid

Thesis Submitted in the Partial Fulfilment of the requirement for the Degree of

MASTER OF ELECTRICAL ENGINEERING

By

Barundeb Mondal

Class Roll No: 002210802019

Examination Roll No: M2ELE23021

Registration No: 163490 of 2022-23

JADAVPUR UNIVERSITY

Under the guidance of

Prof. Biswajit Bhattacharyya

Prof. Sugata Munshi

Electrical Engineering Department

Faculty of Engineering and Technology

JADAVPUR UNIVERSITY

KOLKATA- 700032, W. B. INDIA

2022-2024

JADAVPUR UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

ELECTRICAL ENGINEERING DEPARTMENT

KOLKATA – 700032, INDIA

CERTIFICATE OF RECOMMENDATION

This is to certify that, Mr. Barundeb Mondal (Registration No. 163490 of 2022-2023), has completed and submitted his thesis entitled, "Development of a Microcontroller Based Capacitive Sensor for Non-Contact Measurement of Level of Conductive Liquid", in partial fulfilment of the requirements for the degree of "Master of Electrical Engineering" of Jadavpur University. The thesis work has been carried out by him under our guidance and supervision. The project, in our opinion, is worthy of its acceptance.

SUPERVISORS

Biswajit Bhattacharyya 27-08-2024

Prof. Biswajit Bhattacharyya
Associate Professor
Electrical Engineering Department
Jadavpur University
Associate Professor
Electrical Engg. Deptt.
Jadavpur University
Kolkata - 700 032

S. Munshi 27.8.2024.

Prof. Sugata Munshi
Retired Professor
Electrical Engineering Department
Jadavpur University
Professor
Electrical Engg. Deptt.
Jadavpur University
Kolkata - 700 032

Countersigned

28/08/2024

Prof. (Dr.) Biswanath Roy
Head, Dept. of Electrical Engineering Department
Faculty of Engineering and Technology
Jadavpur University
HEAD
Electrical Engg. Deptt.
JADAVPUR UNIVERSITY
Kolkata-700032

Dipak Laha 28.8.24

Prof. (Dr.) Dipak Laha
Dean
Faculty council of Engg. and Technology
Jadavpur University
DEAN
Faculty of Engineering & Technology
JADAVPUR UNIVERSITY
KOLKATA-700 032

JADAVPUR UNIVERSITY

KOLKATA-700032, INDIA

FACULTY OF ENGINEERING AND TECHNOLOGY

ELECTRICAL ENGINEERING DEPARTMENT

CERTIFICATE OF APPROVAL*

The foregoing thesis is hereby approved as a creditable study **of Master of Electrical Engineering** under Electrical Engineering department and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion therein but approve this thesis only for the purpose for which it is submitted.

Final Examination for Evaluation of the Thesis	Board of Examiners
	………………………………….
	………………………………….
	………………………………….
	………………………………….
	………………………………….
*Only in case thesis is approved	(Signature of Examiners)

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that the thesis entitled **"Development of a Microcontroller Based Capacitive Sensor for Non-Contact Measurement of Level of Conductive Liquid"**, contains literature survey and original research work as part of the course of Master of Engineering under Electrical Engineering department. All the information in this document have been obtained and presented in academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name	:	**BARUNDEB MONDAL**
Class Roll No.	:	**002210802019**
Examination Roll No.	:	**M2ELE23021**
Registration No.	:	**163490 of 2022-2023**
Thesis Name	:	**Development of a Microcontroller Based Capacitive Sensor for Non-Contact Measurement of Level of Conductive Liquid**
Signature with date	:	Barundeb Mondal, 27.08.2024

Dedicated

To

All Of My Respected Teachers

And

My Parents

Acknowledgements

It has been a great privilege for me to have the opportunity of pursuing postgraduate studies at the Department of Electrical Engineering of Jadavpur University. The satisfaction of the successful completion of any task remains incomplete without mention of the people whose constant inspiration, guidance and support made this possible. I would like to take this opportunity to humbly express my gratitude for the innumerable gestures of help, cooperation and inspiration that I have received from my teacher, friends, well-wishers during my post graduate course.

I feel honoured to express my profound regard and deep sense of gratitude to my guides Prof. Sugata Munshi and Prof. Biswajit Bhattacharyya for allowing me to complete my project work under their guidance. Their judicious cooperation, immense support, encouragement, and constructive criticism throughout the tenure of the project work steered me in the right direction.

I would also wish to express my sincere gratitude to my professors in Measurement and Instrumentation Dept. Prof. Dr. Mita Dutta, Prof. Dr. Amitava Chatterjee, Prof. Dr. Gautam Sarkar, Prof. Dr. Palash Kundu and Prof. Dr. Debangshu Dey for their valuable guidance.

I would like to thank Prof. (Dr.) Biswanath Roy, Head of the Department of Electrical Engineering, who was kind enough to provide me with necessary facilities for carrying out this project.

I would also like to thank all the technical staffs in the Measurement and Instrumentation laboratory for assisting me whenever I needed their help during the very project work.

I thank my friends Bisal Sarkar, Susmita Bhattacharya and Saurabh Shit for their support that has helped me to continue this work during my Master of Electrical Engineering course.

The cordial gratitude goes to my parents and relatives for their huge support, patience and sympathetic co-operation, without which I could hardly complete my course of study. They are always there for me.

Thank you very much.

Date: 27.08.2024

Barundeb Mondal

Place: Jadavpur University
Kolkata 32

Barundeb Mondal

CONTENTS

Introduction	**1**
Chapter 1: Capacitive Sensor and Related Signal Conditioning Technique	**3**
1.1 Working Principle of Capacitive Sensor	**4**
1.2 Basic Theory about Capacitance Measurement	**5**
1.2.1 Bridge Method	**6**
1.2.2 RC Time Constant Method	**7**
1.2.3 Measurement of Capacitance in terms of Frequency:	**8**
1.3 Different Types of Capacitive Sensors	**11**
1.4 Different types Signal Conditioning Circuits of the Capacitive Sensor	**13**
1.5 555-Timer as Signal Conditioning Circuit of Capacitive Sensor	**14**
1.5.1 555-Timer Pin Out	**15**
1.5.2 555-Timer Internal Circuit Block Diagram	**15**
1.5.3 555-Timer Based Astable Multivibrator Operation	**16**
1.5.4 Advantages of 555-Timer based Astable Multivibrator as Signal Conditioning Circuits of the Capacitive Sensor	**19**
Chapter 2: Hardware Used in Present Application of Signal Conditioning	**21**
2.1 NE5532 Op-amp	**22**
2.2 1N4733 Zener Diode	**24**
2.3 Arduino Mega 2560 Microcontroller	**26**
2.3.1 The Reason of Choosing a Microcontroller	**27**
2.3.2 Arduino Mega 2560 Microcontroller Pin Out	**28**
2.3.3 Components and features of the Arduino Mega 2560 Microcontroller	**29**
2.3.3.1 ATmega2560 Microcontroller Chip	**31**
2.3.3.2 Digital Input/Output Pins (GPIO)	**32**
2.3.3.3 Analog Input Pins	**33**
2.3.3.4 Serial Communication Ports (UART)	**34**
2.3.3.5 Serial Peripheral Interface (SPI)	**35**
2.3.3.6 Inter-Integrated Circuit (I^2C)	**36**
2.3.3.7 Power Supply Pins	**37**
2.3.3.8 Crystal Oscillator	**38**

2.3.3.9 Reset Circuitry 38
2.3.4 Interrupts of Arduino Mega 2560 Microcontroller 39
2.4 16×2 LCD keypad shield 42

Chapter 3: Present Work 45
3.1 Introduction 46
3.2 Block Diagram of Present Work 46
3.3 Liquid Container with the Capacitive Sensor 46
3.4 Suitable signal conditioning circuit of capacitive sensor 50
3.5 555-timer based astable multivibrator circuit as the signal conditioning circuit 51
3.6 Requirement of the Capacitance Multiplier 53
3.7 Capacitance Multiplier Circuit using Operational Amplifier 54
3.8 555-Timer based Astable Multivibrator Circuit using Capacitance Multiplier 58
3.9 Suitable Microcontroller and its Input Signal range 61
3.10 Establishing Compatibility of Voltage between Microcontroller and Signal Conditioning Circuit 62
3.11 Voltage Regulator using Zener Diode 63
3.12 Arduino Mega 2560 Microcontroller in the Present Work 67
3.12.1 Operations of Arduino mega 2560 microcontroller 67
3.13 16×2 LCD Display 69
3.14 Proposed Capacitive Sensor Circuit 70
3.14.1 Block Diagram of Capacitive Sensor Circuit 70
3.14.2 Schematic Diagram 70
3.14.3 Circuit Arrangement 71
3.14.4 Experimental Setup 72
3.15 Results and Discussions 73
3.15.1 Curve Fitting using Exponential 75
3.15.2 Curve Fitting using Polynomial 78

Conclusions 83

Future scope 85

References 86

Introduction

Introduction

Capacitive sensors are extensively used in the field of industrial, scientific, space and automation for the purpose of measuring pressure, force, humidity, moisture, position, displacement, acceleration, liquid-level, respiration rate, thickness etc.

The basic principle of operation of the capacitive sensors utilizes the effect of either the change in geometrical properties of the materials used or the dielectric property of the same. The geometrical property of the capacitive sensor involves the effective overlapping area of the electrodes, the separation between the electrodes, and thickness of each of the interleaving dielectric materials when multiple dielectric materials are used. Based on geometrical properties of capacitive sensors different structures have been adopted by researchers to suit in different applications. Various capacitive sensors have been designed for the measurement of liquid level [1 – 10]. Some of these sensors employ different structural design, some uses different alignment and positions of electrodes, some of them have non-conventional topology and placement compared to the typically available structures.

On the other hand, capacitive sensors based on the dielectric property of the material used employs the effect of external factors such as humidity, fringing field, type and extent of contamination of the dielectric medium, nature of amalgamation of dielectric materials etc. upon the permittivity of the material. Based on the nature and variety of such dependences, various types of capacitive sensors are in use [11 – 18].

The output of the capacitive sensors, in most cases, are not suitable for direct measurement of the targeted quantity. The output may have small magnitude, may be contaminated with noise and stray interferences, or is suffering from the problem of non-linearity, repeatability or resolution. For such reasons various signal conditioning techniques have been proposed by various authors [9, 10, 19 – 26].

While developing appropriate capacitive sensors to suit various applications, some authors have used on-chip and on-board fabrication methods utilizing appropriate geometrical and dielectric properties of the sensor. The sensors developed in such processes have emerged as a compact package for the required application [1, 13, 19, 27, 28, 29].

Capacitive sensors in present employs the operation of a particular geometric arrangement. The top plate of a glass container is metallic in nature and is used as an electrode. The saline water serves as the other electrode. When the container is partially filled with liquid, air serves as the dielectric. The overall and effective permittivity has been increased by applying a layer of Teflon below the surface of top electrode. Purpose of using such arrangement is to increase sensitivity of the sensor.

In the report of the present work chapter-1 contains the survey of related literatures, broad classifications of various techniques adopted by various authors for implementing targeted tasks and the scope of present work.

Chapter-2 covers the lists and descriptions of components used in the present work.

Chapter-3 contains in detail the proposal and implementation of the targeted work.

Conclusions and references have been included thereafter.

Chapter 1: Capacitive Sensor and Related Signal Conditioning Technique

1.1 Working Principle of Capacitive Sensor

Present work is based on the measurement of level of a certain class of liquid utilizing the working principle of capacitive sensor. Capacitor is a passive electronic component that store and release electrical energy in the form of an electrostatic field. They consist of two conductive plates separated by a dielectric material as shown in Fig: 1.1.

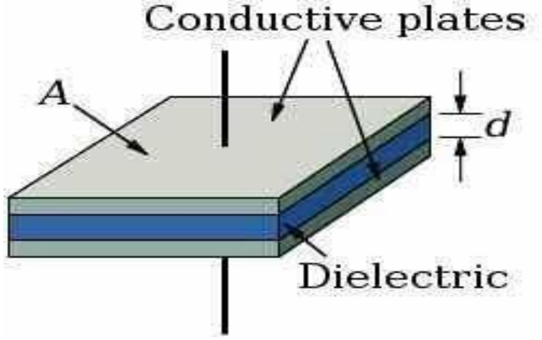


Fig 1.1 Parallel plate capacitor

Construction: Capacitors have two conductive plates typically made of metal, such as aluminium or tantalum. These plates serve as the terminals for the capacitor. The space between the plates is filled with a dielectric material, which is an insulator that prevents direct electrical contact between the plates. Common dielectric materials include ceramic, polyester, polypropylene, and electrolytic materials.

Working Principle: When a voltage is applied across the terminals of a capacitor (as shown in Fig 1.2), it creates an electric field between the plates. Positive charge accumulates on one plate and negative charge accumulates on the other plate which creates a potential difference between the plates. The dielectric material between the plates prevents the flow of current between the plates. The amount of charge that a capacitor can store is depend on value of capacitance (C) of capacitor. Capacitance (C) of capacitor is depends on the surface area (A) of the plates, the distance (d) between the plates (dielectric thickness), and the permittivity (ε) which is depend on properties of the dielectric material. Capacitance is a measure of a capacitor's capability to store electrical charge. It is measured in farads (F).

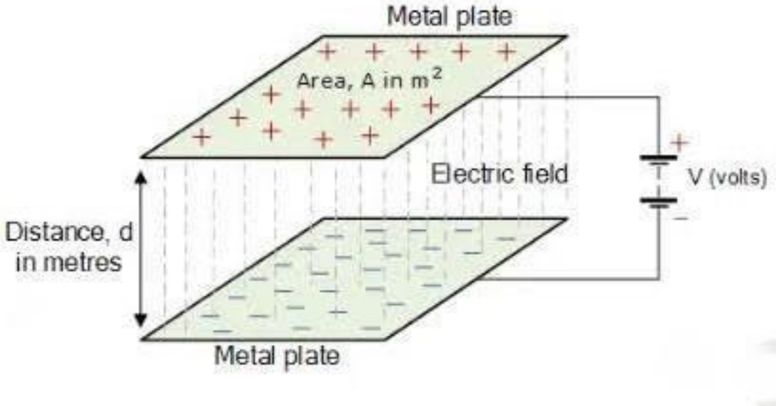


Fig 1.2: Capacitor connected across battery

The capacitance (C) of a capacitor is determined by the formula:

$$C = \varepsilon \frac{A}{d} \tag{1.1}$$

Where ε is permittivity of dielectric material
A is surface area of the plates
d is distance between the plates

The capacitance (C) of a capacitor is also determined by the formula:

$$C = \mathrm{QV} \tag{1.2}$$

Where Q is the charge stored on the plates
V is the voltage across the plates

The larger the capacitance, the more charge a capacitor can store for a given voltage. When a capacitor is connected to a voltage source, it charges up as it accumulates charge on its plates until the voltage across the capacitor reaches the same level as the source voltage. When the voltage source is removed or the circuit is opened, the capacitor can discharge its stored energy through a load. It releases the accumulated charge and generate a current flow to the load until its fully discharged.

There are many types of capacitive sensors. However, generally in a capacitive sensor the parameters can be changed are surface area (A) of the plates, distance between the plates and permittivity of the dielectric material (ε) (as shown in equation 1.1). When any of those parameters changes with certain environmental factor, a capacitive sensor can be made out of that phenomenon.

1.2 Basic Theory about Capacitance Measurement

In the electrical sensor, with the change of a certain environmental factor, the value of specific electrical component also changes. By measuring that change of value of the electrical component, the change of that certain environmental factor can be measured. In this case, the electrical component which value will change with change of environmental factor is capacitor. So, lets learn about basic theory of the capacitor's value measurement or the capacitance measurement to make a capacitive sensor.

Value of capacitance depended on ability of the capacitor to store electrical charge per unit voltage (as shown in equation 1.2). A capacitor consists of two conductive plates separated by a dielectric material. When a voltage is applied across the plates, an electric field is formed, causing the accumulation of positive and negative charges on the plates. There are many methods of capacitance measurement like bridge method, RC time constant method, oscillator method, LC resonance method etc. The suitable method of capacitance measurement depends on factors such as the accuracy required, the range of capacitance being measured and the available equipment etc.

1.2.1 <u>Bridge Method</u>

The bridge method is a common technique for measuring capacitance accurately. It involves using a bridge circuit to compare the unknown capacitance with a known reference capacitance. There are many bridges which are used to measure value of electrical component like resistance, inductance, capacitance. Where Schering bridge and Wien's bridge are bridges used to measure capacitance. Now, let's study about Schering bridge to understand capacitance measurement using bridge method.

Circuit Diagram:

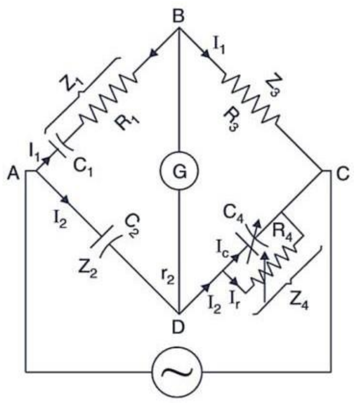


Fig 1.3: Schering bridge

Circuit Configuration:

The Schering bridge, also known as the De Sauty bridge, is a type of AC bridge circuit used for measuring the capacitance of an unknown capacitor. The Schering bridge consists of four arms, similar to a Wheatstone bridge, but with two arms containing capacitors instead of resistors as shown in fig. The arms of the bridge are connected in such a way that the ratio of the impedances in one pair of arms is equal to the ratio of the impedances in the other pair of arms. Components of Schering bridge are C_1, R_1, C_2, R_3, C_4, R_4 and a galvanometer. Where C_1 is the unknown capacitance to be measured, $R_1$is the series resistance representing loss of the capacitor C_1, C_2 is the standard capacitor, R_3 is the non-inductive standard resistance, C_4 is the variable capacitor and R_4 is the variable resistor. The galvanometer is used to balance the bridge.

Expression for Computation of Capacitance:

In this circuit there are two variable components C_4 and R_4. To measure unknown capacitor, the bridge should be in balanced by varying C_4 and R_4. The bridge is balanced when there is no current flow through the galvanometer or null detector connected between the junction of C_1 and R_3 and the junction of C_2 and R_4.

At balance condition ratio of impedance of bridge's branch will be equal to each other:

$$\frac{Z_1}{Z_2} = \frac{Z_3}{Z_4}$$

$$Z_1 Z_4 = Z_2 Z_3 \qquad (1.3)$$

After putting value of impedance of each branch in the equation (1.3) the new equation will be:

$$(R_1 + \frac{1}{jwC_1})(\frac{R_4}{1 + jwC_4 R_4}) = \frac{1}{jwC_2}.R_3 \qquad (1.4)$$

Now after equating the real and imaginary the equation of capacitance will be:

$$C_1 = C_2(\frac{R_4}{R_3}) \qquad (1.5)$$

Using equation (1.5), capacitance of the unknown capacitor (C_1) can be calculated using value of the standard capacitor (C_2), the non-inductive standard resistance (R_3)and the variable resistor (R_4).

Accuracy and Precision:

The accuracy and precision of capacitance measurement using the Schering bridge depend on factors such as the stability of the bridge components, the accuracy of the null detection method (galvanometer or other device), and the calibration of the standard components (Cs and Rs).

1.2.2 RC Time Constant Method:

The RC time constant method is a simple yet effective technique for measuring capacitance by measuring the time it takes for a capacitor to charge or discharge through a known resistor as shown in Fig 1.4.

Circuit Diagram:

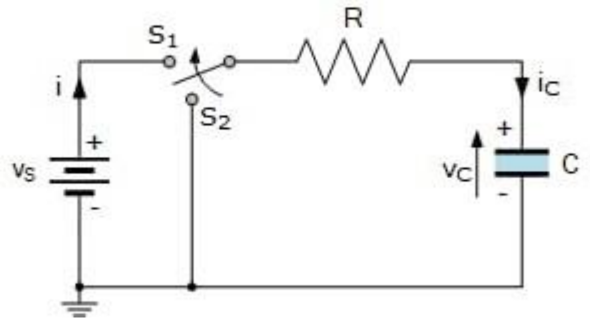


Fig 1.4: RC time constant circuit

Circuit Configuration:

Connect the capacitor (C) in series with a known resistor (R).

Apply a known voltage (Vs) across the series combination of the resistor and the capacitor.

When switch S_1 is closed and S_2 is open then capacitor C start charging from the voltage source Vs and when switch S_2 is closed and S_1 is open then capacitor C start discharging.

So, depending on the configuration, the capacitor can either charge or discharge through the resistor.

Expression for Computation of Capacitance:

The time constant (r) of the RC circuit is the time it takes for the voltage across the capacitor to reach approximately 63.2% of its final value during charging or to decay to the time it takes for the voltage across the capacitor to reach approximately 36.8% of its initial value during discharging.

The time constant (r) of an RC circuit is given by the formula:

$$r = RC \tag{1.6}$$

Where:
C is the capacitance in farads (F),
r is the time constant second (s),
R is the resistance in ohms (Ω).

Once the time constant (τ) is determined, the capacitance (C) of the capacitor can be calculated using the formula:

$$C = \frac{r}{R} \tag{1.7}$$

Accuracy and Precision:

The accuracy and precision of capacitance measurement using the RC time constant method depend on factors such as the accuracy of time measurement, the stability of the resistor value, and the stability of the voltage source. Higher precision can be achieved with more accurate measurement equipment and stable components.

1.2.3 Measurement of Capacitance in terms of Frequency:

Capacitance can be measured in terms of frequency using an oscillator. Oscillator circuit whose output frequency depend on value of capacitor is used for this type of capacitor measurement. Common oscillator circuits include RC oscillators, LC oscillators and 555-timer based astable multivibrator etc are used. The frequency of the oscillator circuit should be relatively high and easily measurable. Measurement the frequency of the oscillator circuit is done using a frequency counter, oscilloscope, or any device capable of accurately measuring frequency. Now, let's study about 555-timer based astable multivibrator.

Circuit Diagram:

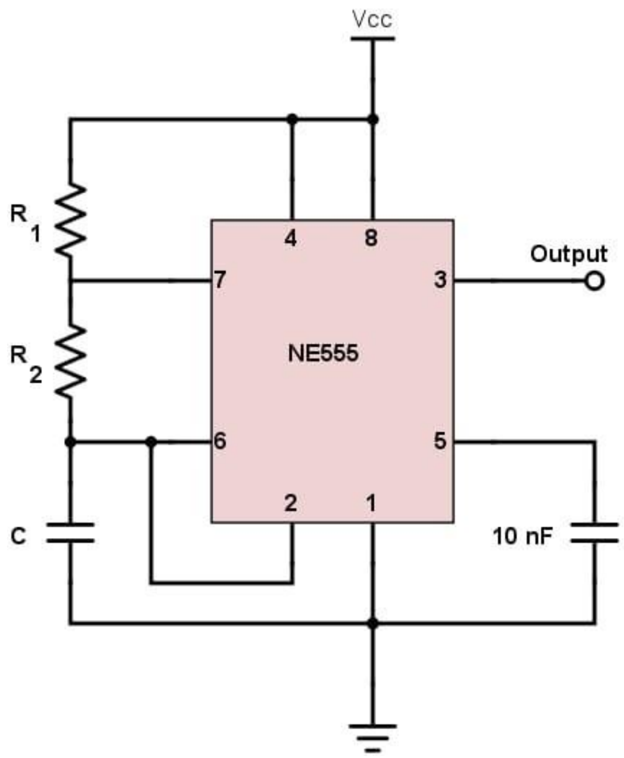


Fig 1.5: 555 astable multivibrator circuit

Circuit Configuration:

Fig 1.5 is a 555 astable multivibrator circuit. 555 IC is a timer, which is used in various timing operation. It generally operates in three modes (monostable, bistable and astable). Astable mode is used for oscillator operation. For astable mode following connections between circuit components are made.

Pin 1 (Ground) of 555 IC Connected to the ground (0V). Pin 2 (Trigger) of 555 IC Connected to pin 6 (threshold) and pin 7 (discharge) through R_2. Pin 3 (Output) of 555 IC Provides the output square wave signal. Pin 4 (Reset) of 555 IC Connected to Vcc to disable reset functionality. Pin 5 (Control Voltage) of 555 IC Connected to capacitor C_2 through the ground. Pin 6 (Threshold) of 555 IC Connected to pin 2 (trigger) and to pin 7 (discharge) through resistor R_2 and capacitor C. Pin 7 (Discharge) of 555 IC Connected to pin 6 (threshold) through resistor R_2 and to pin 8 through resistor R_1. Pin 8 (Vcc) of 555 IC Connected to the positive supply voltage.

Expression for Computation of Capacitance:

The frequency(f) of the square wave output of 555-timer based astable multivibrator circuit can be calculated using the formula:

$$f = \frac{1.44}{(R_1 + 2R_2)C} \tag{1.8}$$

Adjusting the values of resistors R_1 and R_2 and capacitor C allows controlling the frequency and duty cycle of the output waveform. Now, when only capacitor's value is changes then, frequency change depends on only value of capacitor. So, now new formula will be:

$$f = \frac{k}{C} \tag{1.9}$$

Where $\mathrm{k} = \frac{1.44}{(R_1+2R_2)}$ = Constant

Now, if a known standard capacitor C_s is connected in the circuit and measured output frequency of the circuit is f_1. And when an unknown capacitor C_x whose capacitance is to be measured is connected in the circuit and measured output frequency of the circuit is f_2. Then capacitance calculation formula will be:

From the equation (1.9) obtained equations are:

$$f_1 = \frac{k}{C_s} \tag{1.10}$$

$$f_2 = \frac{k}{C_x} \tag{1.11}$$

From the equation (1.10) and equation (1.11) obtained equation is:

$$C_x = C_s \frac{f_1}{f_2} \tag{1.12}$$

Using the equation (1.12) the value of capacitance of the unknown capacitance (C_x) can be calculated.

Accuracy and Precision:

The accuracy and precision of capacitance measurement using the oscillator method depend on factors such as the stability of the oscillator circuit, the accuracy of frequency measurement equipment, and the precision of component values. Higher precision can be achieved with stable oscillator circuits and accurate frequency measurement devices.

1.3 Different Types of Capacitive Sensors

Our present work is with capacitive sensor, so some study about commons types of capacitive sensors which are available in the market, will widen our understanding about capacitive sensor. There are several types of capacitive sensors depend on applications, including:

Level Sensor: Detect the level of liquids or solids in containers or tanks, commonly used in industrial and commercial applications for inventory management and process control. One of the simplest examples is shown in Fig 1.6.

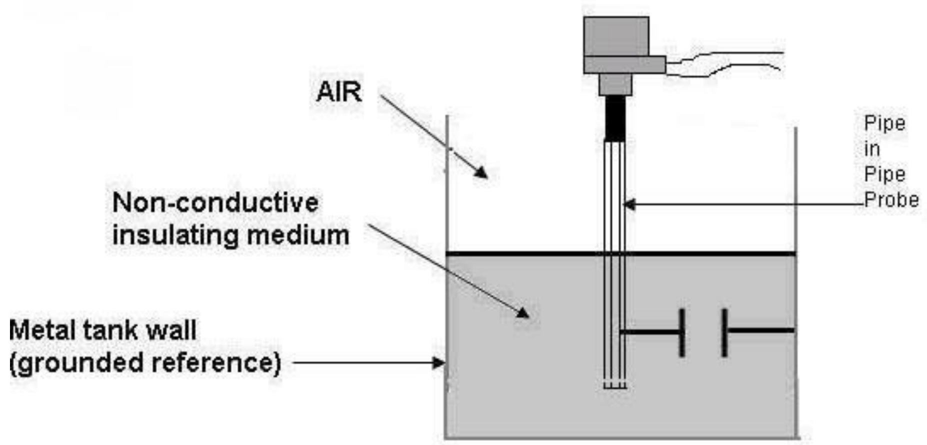


Fig 1.6: Capacitive level sensor

Touch Sensor: These sensors detect touch or proximity of a conductive object, such as a finger or stylus, and are commonly used in touchscreens, touch-sensitive buttons, and interactive display (as shown in Fig 1.7).

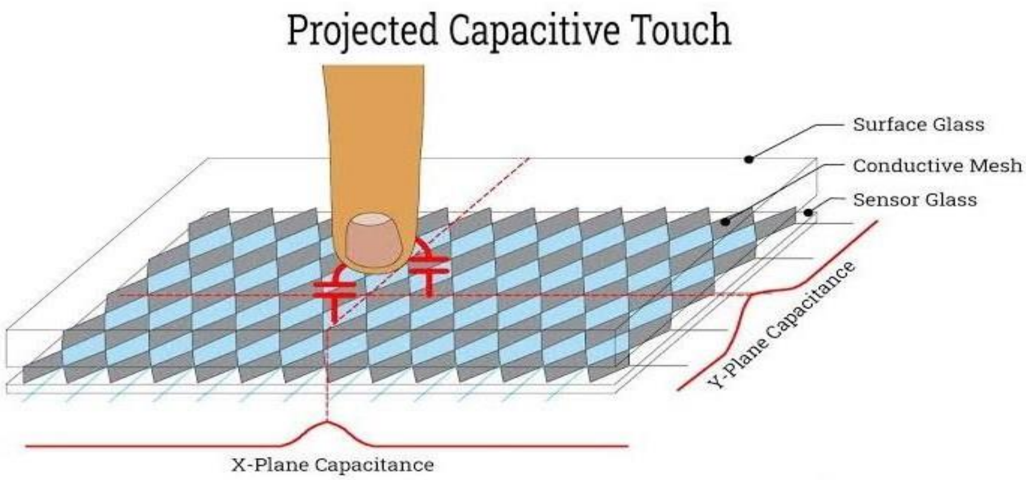


Fig 1.7: Capacitive touch sensor

Proximity Sensor: Detect the presence or proximity of objects without physical contact, commonly used in applications such as object detection, liquid level sensing, and gesture recognition (as shown in Fig 1.8).

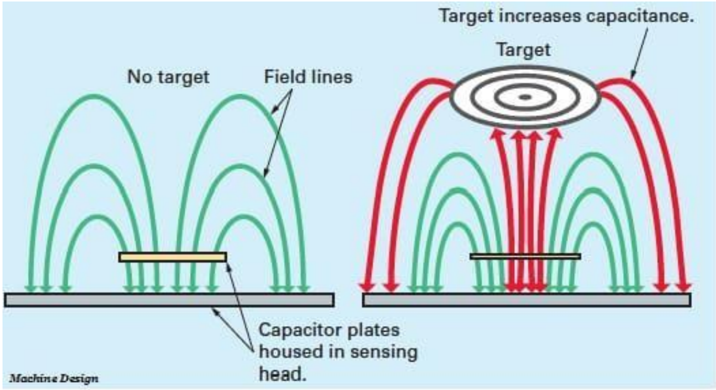


Fig 1.8: Capacitive proximity sensor

Position Sensor: Measure the position or displacement of an object based on changes in capacitance, often used in industrial automation and robotics (as shown in Fig 1.9).

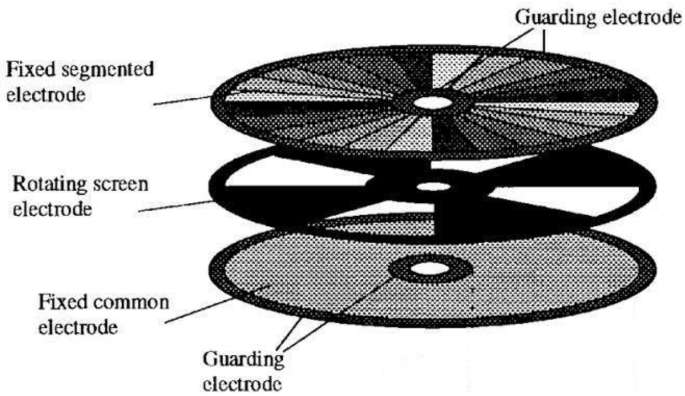


Fig 1.9: Capacitive position sensor

Pressure Sensor: Capable of measuring pressure or force based on changes in capacitance (as shown in Fig 1.10). They are utilized in various industries, including automotive, aerospace, and medical, for applications such as tire pressure monitoring and medical devices.

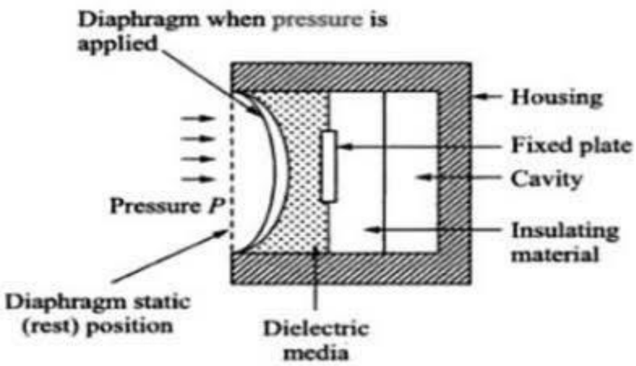


Fig 1.10: Capacitive pressure sensor

Humidity Sensor: Measure humidity levels based on changes in capacitance caused by moisture absorption (as shown in Fig 1.11). They are used in weather stations, HVAC systems, and consumer electronics.

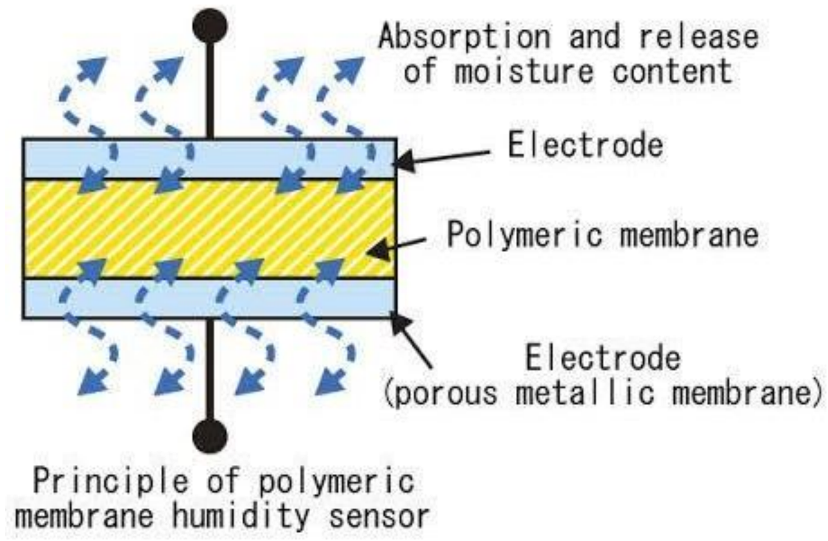


Fig 1.11: Capacitive humidity sensor

1.4 Different types Signal Conditioning Circuits of the Capacitive Sensor

There are various types of signal conditioning circuits are used for capacitive sensors, depending on the specific application and requirements. Here are some common types:

555-timer based astable multivibrator circuit: This circuit generates a square wave signal whose frequency is changed by the capacitance change of the sensor. The frequency and duty cycle of the square wave can be adjusted based on the sensing requirements.

RC Oscillator Circuit: This circuit generates an oscillating sinusoidal signal whose frequency is determined by the capacitance of the sensor. When the capacitance changes due to the presence of an object, the oscillation frequency also changes, allowing for detection.

Charge-Transfer Circuit: This circuit measures the time it takes to charge or discharge a capacitor through the sensor. Changes in capacitance result in changes in the charging or discharging time, which can be detected and used for sensing.

Voltage Divider Circuit: In this circuit, the capacitive sensor is part of a voltage divider network. Changes in capacitance alter the voltage across the sensor, which can be measured using an Analog to Digital converter (ADC) or other sensing circuitry.

Bridge Circuit: Capacitive sensors can be incorporated into a bridge circuit configuration, where changes in capacitance cause an imbalance in the bridge, resulting in a measurable output signal. This circuit is often used for precise measurements and compensation of the environmental factors.

Impedance Measurement Circuit: By measuring the impedance of the capacitive sensor at a certain frequency, changes in the capacitance can be detected. This method is commonly used in proximity sensors and touchscreens.

These are just a few examples of circuits used for capacitive sensors. The choice of circuit depends on factors such as sensor type, application requirements, desired sensitivity, and integration with other electronic systems.

1.5 555-Timer as Signal Conditioning Circuit of Capacitive Sensor

The 555-timer based astable multivibrator circuit is used as the signal conditioning circuit for our present work. The 555-timer is a versatile integrated circuit that can be configured in various modes, including astable, monostable, and bistable modes.

For capacitive sensing, the 555-timer can be used in astable mode to generate a continuous square wave signal. This signal can then be used to charge and discharge a capacitor through a capacitive sensor. The time it takes for the capacitor to charge and discharge, influenced by the capacitance of the sensor. Which can be measured to detect changes in capacitance, such as the presence or absence of an object. Here's a basic overview of how a 555-timer can be used for capacitive sensing:

Astable Mode Configuration: Set up the 555-timer in astable mode by connecting the required resistors and capacitors to generate a square wave output signal. The frequency and duty cycle of the square wave can be adjusted based on the sensing requirements.

Capacitive Sensor Integration: Connect the capacitive sensor to the threshold pin of the 555-timer circuit. The sensor capacitance influences the charging and discharging of the capacitor in the timer circuit.

Signal Processing: Measure the characteristics of the square wave signal, such as its frequency or duty cycle, to determine changes in capacitance caused by the presence or absence of an object near the sensor.

Calibration and Adjustment: Calibrate the circuit to account for environmental factors and optimize sensitivity. This is involve adjusting the values of resistors and capacitors in the 555-timer circuit.

1.5.1 555-Timer Pin Out

The 555-timer typically comes in an 8-pin Dual In-line Package (DIP) or surface-mount package. Here's the pinout of the 555-timer:

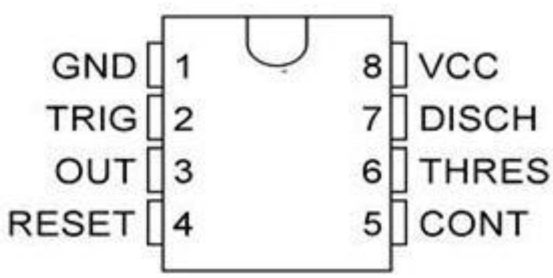


Fig 1.12: 555-timer pin out

GND (Pin 1): Ground pin, connected to the ground reference of the circuit.

TRIG (Pin 2): Trigger pin, used to trigger the timer's internal flip-flop when voltage drops below 1/3 Vcc.

OUT (Pin 3): Output pin, provides the output waveform (e.g., square wave) generated by the 555-timer.

RESET (Pin 4): Reset pin, used to reset the timer's internal flip-flop when voltage exceeds 2/3 Vcc (active low).

CTRL (Pin 5): Control voltage pin, used for overriding the internal voltage divider and setting the upper reference voltage for the comparator.

THRESH (Pin 6): Threshold pin, connected to the inverting input of the internal comparator.

DISCH (Pin 7): Discharge pin, connected to an open collector output transistor used to discharge the timing capacitor.

VCC (Pin 8): Positive supply voltage pin, connected to the positive supply voltage of the circuit.

1.5.2 555-Timer Internal Circuit Block Diagram

The internal circuit of the 555-timer consists of several functional blocks interconnected to perform various timing and oscillation functions. The block diagram of internal circuit of the 555-timer is shown in Fig 1.13.

Voltage Divider: The internal voltage divider consists of three resistors connected in series between Vcc and GND. The voltage at the junction between the second and third resistors (2/3 Vcc) is used as a reference voltage for the internal comparators.

Comparator: The 555-timer contains two voltage comparators. One comparator compares the voltage at the THRESH pin (Pin 6) with 2/3 Vcc, while the other compares the voltage at the TRIG pin (Pin 2) with 1/3 Vcc.

Flip-Flop: The output of the comparators is connected to a set-reset flip-flop. The flip-flop toggles its output state based on the comparator inputs, generating the square wave output at Pin 3.

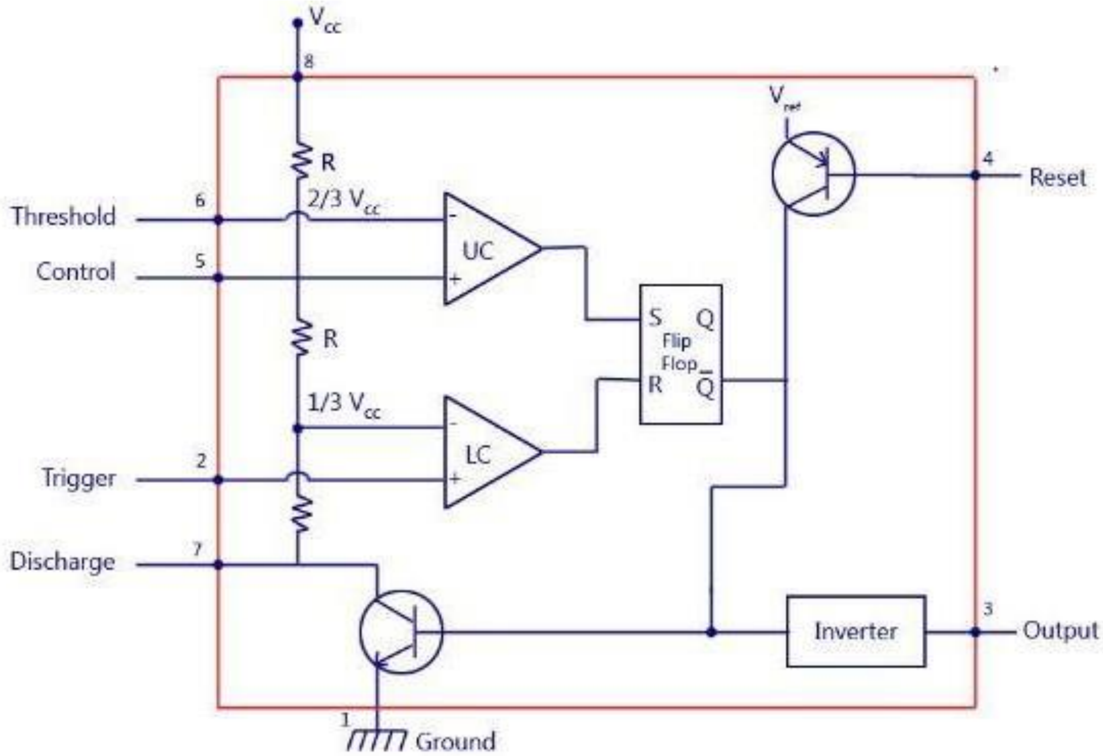


Fig 1.13: 555-timer Internal Circuit Block Diagram

Discharge Transistor: The DISCH pin (Pin 7) is connected to an open collector output transistor used to discharge the timing capacitor during the discharge phase of the 555-timer operation.

Control Voltage Input: The CTRL pin (Pin 5) allows an external voltage to override the internal voltage divider, providing flexibility in setting the upper reference voltage for the comparators.

Overall, the 555-timer's internal circuitry enables it to perform various timing and oscillation functions, making it a versatile and widely used component in electronic circuits.

1.5.3 555-Timer Based Astable Multivibrator Operation

The 555-timer can be configured as an astable multivibrator, also known as an oscillator, to generate a continuous square wave output. Here's a basic schematic and explanation of how to set up a 555-timer in astable mode:

Component Requirements: 555-timer IC**,** Resistors (R_1 and R_2)**,** Capacitor (C and C_2)**,** Power supply (Vcc)

Circuit Diagram:

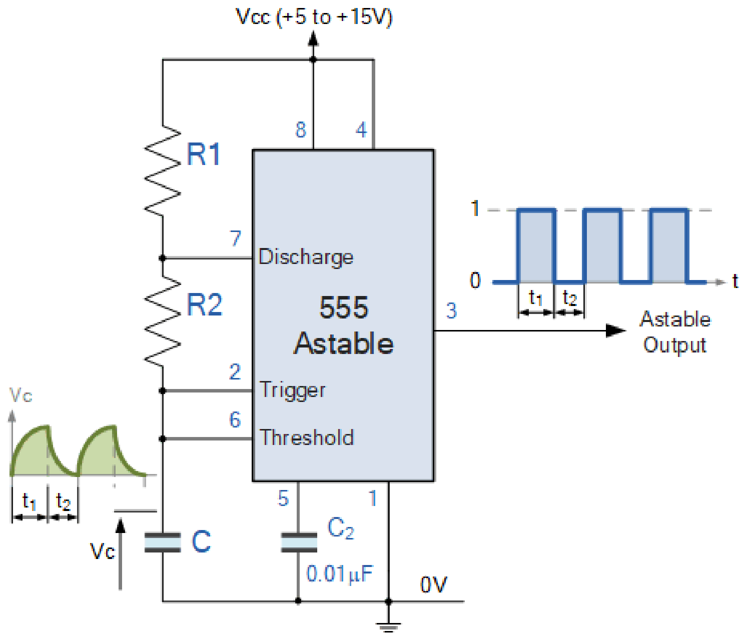


Fig 1.14: 555-timer based astable multivibrator circuit

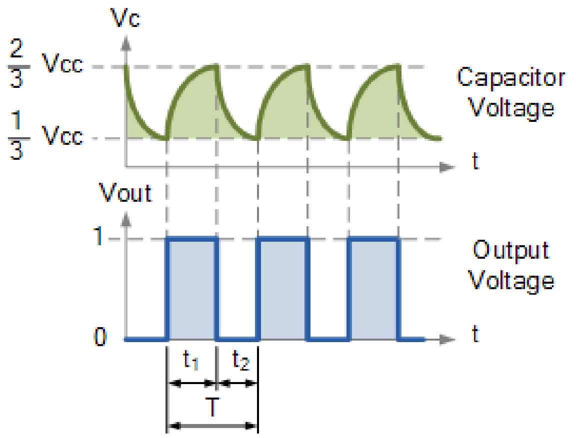


Fig 1.15: Output voltage waveform and capacitor voltage waveform of 555-timer based astable multivibrator circuit

Specification of NE555 Timer IC:

Supply Voltage (Vcc): 4V to 16V
Supply Current (Icc): 200mA maximum
Output Current: 200mA maximum
Threshold Voltage: 2/3 Vcc
Trigger Voltage: 1/3 Vcc
Reset Voltage: 0.7V (active low)
Control Voltage (Vcontrol): 2/3 Vcc
Timing Capacitor: Typical range from 100 pF to 1000 µF

Circuit Configuration:

Pin 1 (Ground): Connected to the ground (0V).

Pin 2 (Trigger): Connected to pin 6 (threshold) and pin 7 (discharge) through R_2.

Pin 3 (Output): Provides the output square wave signal.

Pin 4 (Reset): Connected to Vcc to disable reset functionality.

Pin 5 (Control Voltage): Connected to capacitor C_2 through the ground.

Pin 6 (Threshold): Connected to pin 2 (trigger) and to pin 7 (discharge) through resistor R_2 and capacitor C.

Pin 7 (Discharge): Connected to pin 6 (threshold) through resistor R_2 and to pin 8 through resistor R_1.

Pin 8 (Vcc): Connected to the positive supply voltage.

Operation:

Initially, capacitor C is discharged, and the output at pin 3 is low.

When power is applied, capacitor C starts charging through resistors R_1 and R_2.

Once the voltage across capacitor C exceeds 2/3 of Vcc (threshold voltage), the internal comparator resets the flip-flop, causing the output at pin 3 to go low.

Capacitor C then discharges through resistor R_2 and pin 7 (discharge).

Once the voltage across capacitor C falls below 1/3 of Vcc (trigger voltage), the flip-flop is set again, and the output at pin 3 goes high.

This cycle repeats, resulting in a continuous square wave output at pin 3 with a frequency determined by the values of resistors R_1 and R_2 and capacitor C.

Frequency Calculation:

Time period of output waveform of 555-timer based astable multivibrator will be:

On time of the output waveform is:

$$t_1 = 0.693(R_1 + R_2)C \tag{1.13}$$

Off time of the output waveform is:

$$t_2 = 0.693R_2C \tag{1.14}$$

Time period of the output waveform is:

$$T = t_1 + t_2 = 0.693(R_1 + 2R_2)C \tag{1.15}$$

The frequency(f) of the square wave output can be calculated using the formula:

$$f = \frac{1}{T} = \frac{1.44}{(R_1 + 2R_2)C} \tag{1.16}$$

Adjusting the values of resistors R1 and R2 and capacitor C allows controlling the frequency and duty cycle of the output waveform.

Duty cycle:

The duty cycle (D) of the square wave signal is ratio of on time (t_1) and time period (T) of the signal. Then duty cycle can be calculated using the formula:

$$\mathrm{D} = \frac{t_1}{T} = \frac{R_1 + R_2}{R_1 + 2R_2} \tag{1.17}$$

The duty cycle can be adjusted by changing the values of resistors R_1 and R_2 while keeping their ratio constant.

This configuration of the 555-timer as an astable multivibrator provides a simple and versatile way to generate square wave signals for various electronic applications, including clock generation, pulse generation, and waveform generation.

1.5.4 Advantages of 555-Timer based Astable Multivibrator as Signal Conditioning Circuits of the Capacitive Sensor

The 555-timer configured as an astable multivibrator offers several advantages over different types of oscillators for certain applications like capacitive sensor:

Simplicity: The 555-timer is relatively simple to configure and use compared to other oscillator circuits, making it suitable for beginners or the applications where simplicity is desired.

Versatility: The output of the astable multivbrator is a pulse train and the time-period/ frequency of the pulse train will remain unaltered even if the train is transmitted over a long distance. This ensures that the measured capacitance value and hence the measured physical variable remains unaffected although the waveshape may get distorted. Furthermore, the output can be taken to a processor based system without requirement of an ADC.

Low Cost: The 555-timer is a low-cost integrated circuit, making it an economical choice for many applications. Its affordability makes it suitable for mass-produced products and projects with budget constraints.

Wide Operating Voltage Range: The 555-timer can operate over a wide range of supply voltages, typically from 4.5V to 15V or higher, depending on the specific variant. This flexibility in voltage requirements allows it to be used in diverse electronic systems.

Stability and Reliability: When properly configured, the 555-timer can provide stable and reliable oscillation over a wide temperature range. This stability is important for applications requiring consistent timing or frequency generation.

Ease of Integration: The 555-timer can easily be integrated into larger electronic circuits due to its standard pinout and compatibility with common electronic components. It can be used alongside other components to create more complex systems.

Availability: The 555-timer is widely available and has been in production for many years, making it easy to obtain from multiple suppliers. Its popularity and ubiquity ensure continued support and availability in the electronics market.

Chapter 2: Hardware Used in Present Application of Signal Conditioning

2.1 NE5532 Op-amp

In our present work NE5532 Op-amp is used to construct a capacitance multiplier circuit. We also attempted to use an Op-amp to construct Schmitt trigger circuit as a voltage regulator. So, let's learn about Op-amp. An operational amplifier, commonly known as an Op-amp, is a versatile and widely used electronic component in analog circuit design. It is a high-gain voltage amplifier with differential inputs and, usually, a single-ended output. Op-amps are fundamental building blocks in a wide range of applications, from simple amplifiers to complex signal processing circuits.

Key Characteristics of an Op-amp:

High Gain: Op-amps typically have very high open-loop gain, which can range from 10,000 to over 1,000,000. This high gain allows the Op-amp to amplify even very small differences between its input terminals.

Differential Inputs: An Op-amp has two input terminals (Inverting Input, Non-Inverting Input). Applying a signal to Inverting Input (-) inverts the phase of the output. Applying a signal to Non-Inverting Input (+) results in an output that is in phase with the input.

Single-Ended Output: The output of an Op-amp is typically referenced to ground and provides a voltage that is a function of the difference between the inputs.

High Input Impedance: The input impedance of an Op-amp is usually very high, meaning it draws negligible current from the input source.

Low Output Impedance: The output impedance of an Op-amp is typically low, allowing it to drive loads without significant loss of signal strength.

Wide Bandwidth: Depending on the design, Op-amps can offer a wide bandwidth, allowing them to amplify signals over a broad range of frequencies.

Offset Voltage: There is usually a small difference in the input voltage that can cause a non-zero output even when the inputs are equal. This is called the offset voltage.

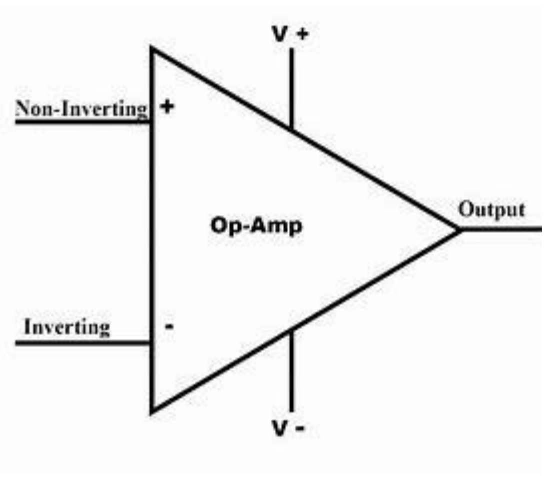


Fig 2.1: Op-amp circuit diagram symbol

Key Parameters of an Op-amp:

Gain-Bandwidth Product (GBP): The product of the Op-amp's gain and bandwidth, indicating how much gain can be maintained at a given frequency.

Slew Rate: The maximum rate at which the output voltage can change, affecting how quickly the Op-amp can respond to rapid changes in the input signal.

Input Offset Voltage: The voltage difference that must be applied between the inputs to make the output zero when it should be zero.

Common-Mode Rejection Ratio (CMRR): The ability of the Op-amp to reject input signals that are common to both input terminals.

Power Supply Rejection Ratio (PSRR): The ability of the Op-amp to maintain a stable output despite changes in the power supply voltage.

Input Impedance: The impedance seen by the input signals, ideally infinite to prevent loading the source.

Output Impedance: The impedance of the output, ideally very low to efficiently drive loads.

Specification of NE5532 Op-amp:

Electrical Characteristics (at Vcc = ±15V, Ta = 25°C)
Input Offset Voltage: Typically, 0.5 mV, maximum 4 mV
Input Offset Current: Typically, 20 nA, maximum 200 nA
Input Bias Current: Typically, 200 nA, maximum 800 nA
Input Resistance: Typically, 300 kΩ, minimum 30 kΩ
Large Signal Voltage Gain: Typically, 100 dB, minimum 86 dB
Common Mode Rejection Ratio (CMRR): Typically, 100 dB, minimum 70 dB
Power Supply Rejection Ratio (PSRR): Typically, 100 dB, minimum 80 dB
Slew Rate: Typically, 9 V/µs
Gain Bandwidth Product: Typically, 10 MHz
Total Harmonic Distortion (THD): Typically, 0.002% at 1 kHz
Output Voltage Swing: Typically, ±13 V at RL = 600Ω
Supply Voltage Range: ±3V to ±20V
Quiescent Current: Typically, 8 mA, maximum 16 mA (per amplifier)
Output Current: Typically, 38 mA, maximum 60 Ma

NE5532 Op-amp Pin Out:

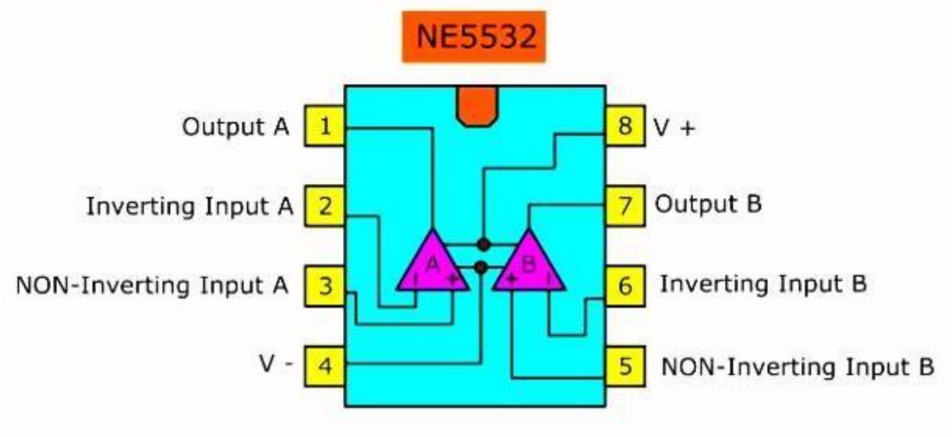


Fig 2.2: NE5532 Op-amp pin out

Applications of Op-amps:

Signal Amplification: Op-amps are commonly used to amplify weak signals in audio devices, sensors, and communication systems.

Filters: By combining resistors, capacitors, and Op-amps, various types of filters (low-pass, high-pass, band-pass, band-stop) can be created to manipulate signal frequencies.

Oscillators: Op-amps can be used in oscillator circuits to generate waveforms like sine, square, and triangle waves.

Comparators: An Op-amp can be used as a comparator to compare two voltages and output a digital signal indicating which is larger.

Voltage Regulators: Op-amps are used in voltage regulation circuits to maintain a constant output voltage despite changes in input voltage or load conditions.

Analog Calculators: In complex analog circuits, Op-amps can perform mathematical operations like addition, subtraction, integration, and differentiation.

2.2 1N4733 Zener Diode

In our present work Zener diode is used to construct a voltage regulator circuit. So, let's learn about Zener diode. Zener diodes are essential components in many electronics for voltage regulation, protection, and reference. A Zener diode is a type of diode designed to allow current to flow in the reverse direction when the voltage exceeds a certain value, known as the Zener breakdown voltage. Unlike regular diodes, which are meant to block reverse current, Zener diodes are used primarily for voltage regulation and protection purposes. To use Zener diode in a circuit, it should be careful chosen as per required parameters.

Key parameters of a Zener Diode:

Zener Breakdown Voltage (V_z): The voltage at which the diode allows reverse current to flow. This voltage is carefully controlled during manufacturing, making Zener diodes useful for maintaining a stable reference voltage.

Reverse Current (I_z): The current that flows when the Zener diode is in reverse breakdown mode. The Zener diode is designed to handle this current without damage as long as it does not exceed its maximum power dissipation.

Forward Voltage (V_f): Like a regular diode, a Zener diode will conduct in the forward direction when the forward voltage (typically around 0.7V for silicon diodes) is applied.

Reverse Leakage Current (I_r): The leakage current when a reverse voltage slightly below the Zener voltage is applied.

Power Dissipation (P_z): The maximum power the Zener diode can dissipate without overheating. This is the product of the Zener voltage and the Zener current.

Also, Zener diodes are designed to maintain a stable Zener voltage across a range of temperatures, although this stability can vary depending on the specific diode.

Specifications of 1N4733 Zener Diode:

Zener Voltage (Vz): 5.1V
Zener Current (Iz): 5mA
Maximum Zener Impedance (Zz): 2 Ω
Reverse Leakage Current (Ir): 5µA
Forward Voltage (V_f): 0.7V
Power Dissipation (Pz): 1W

1N4733 Zener Diode Pin Out:

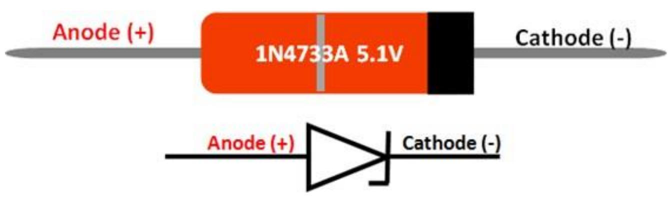


Fig 2.3: 1N4733 Zener diode pin out

Operations in Zener Diode:

Forward Bias: When a Zener diode is forward-biased (anode to positive, cathode to negative), it behaves like a regular diode, allowing current to flow once the forward voltage is reached (typically 0.7V for silicon diodes).

Reverse Bias (Zener Region): When reverse-biased (anode to negative, cathode to positive), the diode blocks current flow until the reverse voltage exceeds the Zener breakdown voltage. At this point, the diode "breaks down" and conducts current in the reverse direction, stabilizing the voltage across its terminals at the Zener voltage.

Applications of Zener Diodes:

Voltage Regulation: Zener diodes are commonly used in power supply circuits to maintain a stable output voltage. They are placed in reverse bias across the load, where they stabilize the voltage by conducting excess current when the voltage exceeds the Zener voltage.

Voltage Clamping: Zener diodes can protect circuits from voltage spikes by clamping the voltage to a safe level.

Reference Voltage: Zener diodes provide a stable reference voltage for precision measurement and control circuits.

Overvoltage Protection: Zener diodes can protect sensitive components from overvoltage conditions by shunting excess voltage away.

2.3 Arduino Mega 2560 Microcontroller

The Arduino Mega 2560 microcontroller is used for signal processing in our present work. So, now let's learn about The Arduino Mega 2560 microcontroller. The Arduino Mega 2560 microcontroller is a versatile microcontroller board that offers more input/output pins and memory compared to the Arduino Uno, making it suitable for larger and more complex projects. Here are some key features of the Arduino Mega 2560:

Microcontroller: The Arduino Mega 2560 is based on the ATmega2560 microcontroller, which has 256 KB of flash memory for storing code, 8 KB of SRAM for variables, and 4 KB of EEPROM for data storage.

Digital I/O Pins: It has a total of 54 digital input/output pins, of which 15 can be used as PWM (Pulse Width Modulation) outputs. These pins allow interface with various sensors, actuators, and other digital devices.

Analog Inputs: The Arduino Mega has 16 analog input pins, which can be used to read analog sensors and signals. These pins have a resolution of 10 bits, providing 1024 possible values (0-1023).

Serial Communication: It supports multiple serial communication interfaces, including UART, SPI (Serial Peripheral Interface), and I2C (Inter-Integrated Circuit). This allows the Arduino Mega to communicate with other microcontrollers, sensors, displays, and other peripherals.

USB Interface: Like other Arduino boards, the Mega has a built-in USB interface for programming and serial communication with a computer. It uses a standard USB Type-B connector.

Power Supply: The board can be powered via USB or an external power supply. It has a voltage regulator that can accept input voltages from 7V to 12V DC.

Memory Expansion: The Arduino Mega has more memory compared to the Uno, making it suitable for larger projects with more complex code. It also has additional memory expansion options, such as external EEPROM or SD card modules.

Compatibility: The Arduino Mega is compatible with the Arduino software development environment (IDE) and supports the same programming language and libraries as other Arduino boards. This makes it easy to get started with programming and prototyping.

Overall, the Arduino Mega is an excellent choice for projects that require a large number of I/O pins, extensive memory, and advanced features. It's commonly used in robotics, automation, data logging, and other applications.

2.3.1 The Reason of Choosing a Microcontroller

In our present work an Arduino Mega 2560 microcontroller is used for signal processing. The Arduino Mega offers several advantages over other microcontroller boards. Here are some key advantages:

More I/O Pins: The Arduino Mega has a significantly larger number of digital and analog input/output pins compared to boards like the Uno. This makes it suitable for projects that require interfacing with a large number of sensors, actuators, and other peripherals without the need for additional multiplexing or external hardware.

Increased Memory: With 256 KB of flash memory, 8 KB of SRAM, and 4 KB of EEPROM, the Arduino Mega provides more resources for storing code and data compared to smaller Arduino boards. This extra memory allows for more complex and feature-rich programs, making it suitable for larger and more demanding projects.

Multiple Serial Ports: The Mega features multiple hardware serial ports, which can be used for serial communication with other devices such as GPS modules, Bluetooth modules, or other microcontrollers. This allows for more flexible communication options without sacrificing performance.

Multiple Timers: The Mega has six hardware timer/counter. Because of that, six tasks can execute parallelly along with main program. Which is useful in future extension of present work.

Multiple Interrupts: It has enriched feature of interrupt handing (external interrupts, software interrupts, timer interrupts). Which was useful in implementing the present task.

Higher Processing Power: The Mega is based on a more powerful microcontroller compared to entry-level Arduino boards, featuring an ATmega2560 with a higher clock speed (16MHz) and more advanced features. This increased processing power allows for faster execution of code and handling of complex tasks.

Compatibility with Existing Libraries: Since the Mega is part of the Arduino family, it is fully compatible with the Arduino development environment (IDE) and the vast ecosystem of libraries and resources available for Arduino. This makes it easy to leverage existing code and libraries when developing projects.

Expandability: The Mega offers more options for expansion and connectivity compared to smaller Arduino boards. It has additional headers and connectors for interfacing with external modules, sensors, displays, and other peripherals, allowing for greater customization and versatility in project design.

Overall, the Arduino Mega is well-suited for projects that require a high number of I/O pins, extensive memory, multiple serial communication channels, and increased processing power. It's an excellent choice for advanced hobbyists, educators, and professionals working on complex electronics and robotics projects.

2.3.2 <u>Arduino Mega 2560 Microcontroller Pin Out</u>

The Arduino Mega has a total of 54 digital input/output pins, 16 analog inputs, and various other pins for power, communication, and programming (as shown in fig2.1). Here's a basic overview of the pinout:

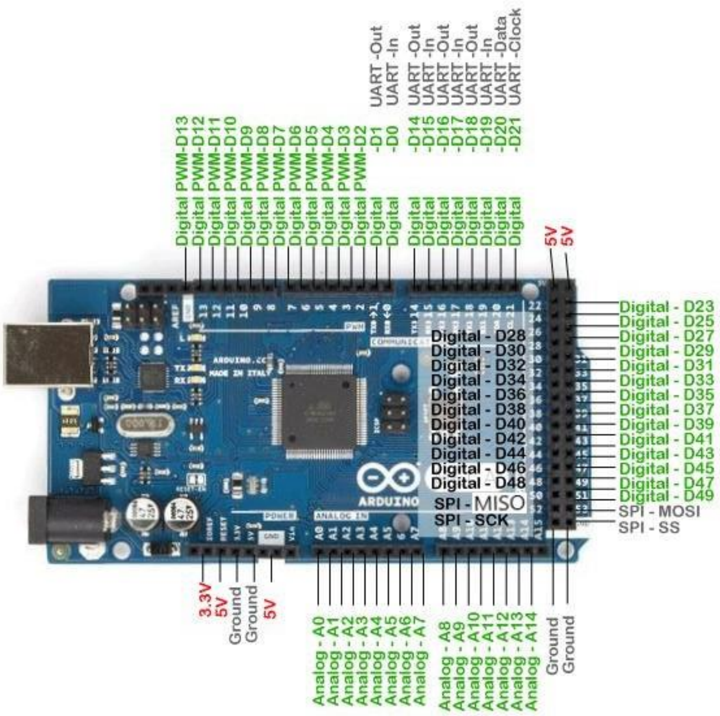


Fig 2.4: Arduino mega 2560 microcontroller pin out

Digital Pins (0-53): These are digital input/output pins. Pins 0 and 1 are also used for serial communication (RX and TX).

Analog Pins (A0-A15): These are analog input pins. They can also be used as digital input/output pins (pins 54-69).

Power Pins:

Vin: Input voltage to the board when using an external power supply (7V - 12V).

5V: Provides 5V output from the voltage regulator.

3.3V: Provides 3.3V output from the voltage regulator.

GND: Ground pins.

Communication Pins:

Serial Ports: RX0/TX0 (pins 0 and 1), RX1/TX1 (pins 19 and 18), RX2/TX2 (pins 17 and 16), RX3/TX3 (pins 15 and 14).

SPI: MOSI (51), MISO (50), SCK (52), SS (53).

I2C: SDA (20), SCL (21).

Other Pins:

Reset: Reset pin (connected to the reset button).

AREF: Analog reference voltage pin.

ICSP Header: Used for in-circuit programming of the microcontroller.

External Interrupts: Pins 2, 3, 18, 19, 20, and 21 can be used as external interrupt pins.

PWM Pins: Pins 2 to 13 and 44 to 46 support PWM (Pulse Width Modulation).

LEDs: There are several onboard LEDs for power, communication, and debugging purposes.

This is a basic overview, and the pin functions can vary depending on how they are configured in our code. Always refer to the official Arduino Mega pinout diagram and documentation for detailed information and usage guidelines.

2.3.3 Components and features of Arduino Mega 2560 Microcontroller

Now let's learn about the different parts of the Arduino Mega 2560 microcontroller:

ATmega2560 Microcontroller Chip: This is the brain of the Arduino Mega 2560. It's an 8-bit AVR microcontroller manufactured by Atmel (now part of Microchip Technology). The ATmega2560 features a high-performance CPU, flash memory for program storage, SRAM for data storage, and EEPROM for non-volatile storage.

Digital Input/Output Pins (GPIO): The ATmega2560 microcontroller has a total of 54 digital I/O pins, labelled from 0 to 53. These pins can be individually configured as digital inputs or outputs, allowing the Arduino to interface with external digital devices such as sensors, LEDs, motors, and more.

Analog Input Pins: The Arduino Mega 2560 has 16 analog input pins, labelled A0 through A15. These pins can read analog voltage levels from sensors or other analog devices. Each analog input pin has a 10-bit analog-to-digital converter (ADC), allowing it to convert analog signals into digital values.

Serial Communication Ports (UART): The ATmega2560 microcontroller has multiple hardware UART (Universal Asynchronous Receiver-Transmitter) serial communication ports. These ports allow the Arduino to communicate with other serial devices, such as computers, GPS modules, Bluetooth modules, and more. The Mega 2560 has four UART ports: Serial (pins 0 and 1), Serial1 (pins 19 and 18), Serial2 (pins 17 and 16), and Serial3 (pins 15 and 14).

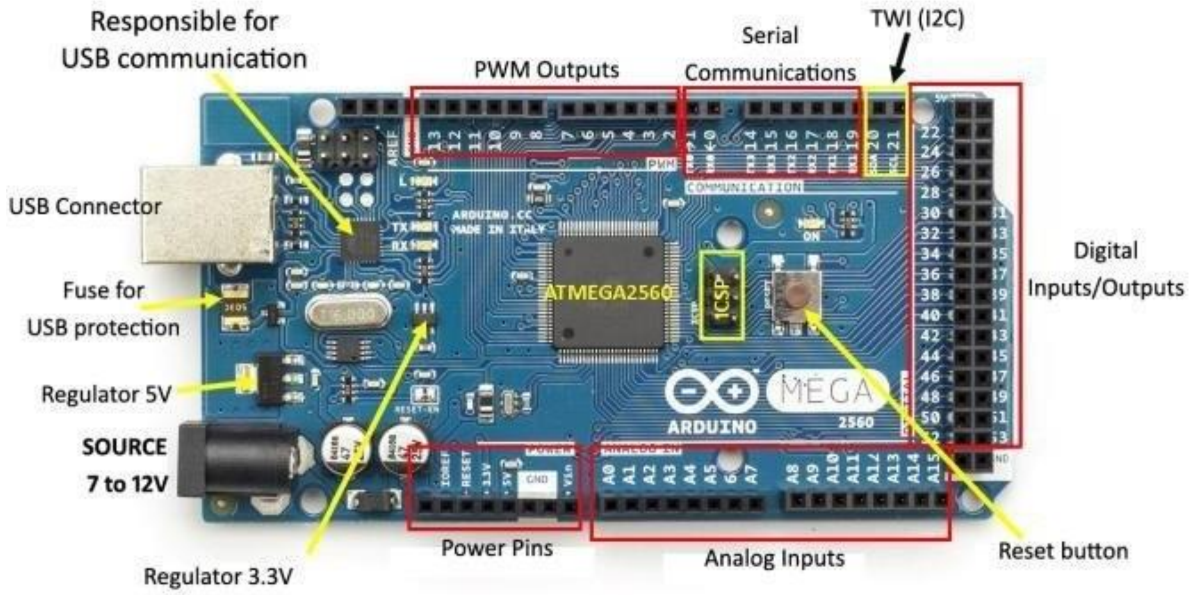


Fig 2.5: Components and features of the Arduino Mega 2560 microcontroller

SPI (Serial Peripheral Interface): The ATmega2560 supports SPI communication, which allows for high-speed serial communication with external SPI devices such as displays, SD cards, and sensors. The SPI pins on the Mega 2560 are MOSI (Master Out Slave In), MISO (Master In Slave Out), SCK (Serial Clock), and SS (Slave Select), located on pins 51, 50, 52, and 53 respectively.

I2C (Inter-Integrated Circuit): The ATmega2560 supports I2C communication, which allows multiple devices to communicate with each other using a two-wire serial interface. The I2C pins on the Mega 2560 are SDA (Serial Data) and SCL (Serial Clock), located on pins 20 and 21 respectively.

Power Supply Pins: The Arduino Mega 2560 has various power supply pins for providing power to external components. These include 5V, 3.3V, and GND (Ground) pins. Additionally, there is a Vin pin for providing an external power supply voltage (typically 7V to 12V) and a reset pin for resetting the microcontroller.

Crystal Oscillator: The ATmega2560 microcontroller uses an external crystal oscillator or resonator to generate a precise clock signal for timing its operations.

Reset Circuitry: The Arduino Mega 2560 includes circuitry for managing the reset functionality of the microcontroller. This allows the microcontroller to be reset manually using a reset button or through software.

These are the main components and features of the Arduino Mega 2560 microcontroller, which enable it to interact with the outside world and execute programmed tasks in various projects.

2.3.3.1 ATmega2560 Microcontroller Chip

The ATmega2560 microcontroller chip is the heart of the Arduino Mega 2560 board. Here's a breakdown of its key features:

Manufacturer: The ATmega2560 is manufactured by Atmel Corporation, which is now part of Microchip Technology after the acquisition in 2016.

Architecture: It belongs to the AVR family of microcontrollers, which are based on the modified Harvard architecture. AVR microcontrollers are known for their low power consumption, high performance, and ease of use.

CPU: The ATmega2560 features an 8-bit AVR RISC (Reduced Instruction Set Computer) CPU running at a clock speed of up to 16 MHz. It executes instructions in a single clock cycle, providing efficient operation.

Memory:

Flash Memory: The ATmega2560 has 256 KB of flash memory for storing the user's program code. This allows for complex programs to be stored directly on the microcontroller.

SRAM (Static Random Access Memory): It has 8 KB of SRAM for storing variables and other data during program execution.

EEPROM (Electrically Erasable Programmable Read-Only Memory): The ATmega2560 includes 4 KB of EEPROM for storing non-volatile data that needs to be retained even when power is removed.

Peripheral Features:

Digital I/O: The ATmega2560 provides a large number of general-purpose digital I/O pins, which can be configured as inputs or outputs.

Analog Inputs: It has 16 analog input channels with a 10-bit resolution ADC (Analog-to-Digital Converter), allowing it to read analog sensor values.

Timers/Counters: The microcontroller includes multiple timers/counters for generating PWM (Pulse Width Modulation) signals, generating interrupts, and other timing-related tasks.

UART, SPI, and I2C: The ATmega2560 supports multiple serial communication interfaces, including UART (Universal Asynchronous Receiver-Transmitter), SPI (Serial Peripheral Interface), and I2C (Inter-Integrated Circuit), allowing it to communicate with other devices.

Power Management:

Low Power Modes: The ATmega2560 supports various low-power modes to reduce power consumption when the microcontroller is idle or not actively performing tasks.

Power Supply: It operates at a wide voltage range, typically from 1.8V to 5.5V, making it compatible with a variety of power sources.

Package: The ATmega2560 is available in a 100-pin TQFP (Thin Quad Flat Package) or QFP (Quad Flat Package) package, making it suitable for surface-mount assembly.

Overall, the ATmega2560 microcontroller chip provides the processing power, memory, and peripheral features necessary for controlling various electronic devices and executing programs in Arduino Mega 2560-based projects.

2.3.3.2 Digital Input/Output Pins (GPIO)

Digital Input/Output (I/O) pins, also known as GPIO (General Purpose Input/Output) pins, are essential components of microcontrollers like the ATmega2560 found on the Arduino Mega 2560 board. Here's a detailed explanation:

Functionality: GPIO pins can be dynamically configured by the user to serve as either digital inputs or digital outputs. Their versatile nature allows them to interface with a wide range of digital devices and sensors or control digital components like LEDs, relays, and motors.

Digital Input: When configured as digital inputs, these pins can detect the presence or absence of a voltage level (typically 0V for LOW and 5V for HIGH in the case of Arduino). They can sense the state of external digital signals, such as switches, pushbuttons, or sensors, and pass this information to the microcontroller for processing.

Digital Output: In digital output mode, GPIO pins can drive external devices by toggling their voltage levels. They can produce either a HIGH (5V) or LOW (0V) output voltage, allowing them to turn on or off LEDs, relays, motors, or other digital components.

Configuration: The configuration of GPIO pins is controlled through software using functions provided by the Arduino IDE or other development environments. Users can set pins to input or output mode and manipulate their states (HIGH or LOW) programmatically.

Numbering: On the Arduino Mega 2560, there are a total of 54 digital I/O pins, labelled from 0 to 53. These pins are accessible through the header connectors on the board and can be easily connected to external circuits using jumper wires, breadboards, or connectors.

Voltage Levels: When configured as inputs, GPIO pins typically have high-impedance inputs, meaning they draw minimal current from external sources. When configured as outputs, they can sink or source a limited amount of current, depending on the specific microcontroller's specifications.

Usage: GPIO pins are fundamental building blocks for almost every Arduino project, as they provide the means to interact with the external world. Whether reading sensor data, controlling actuators, or responding to user input, GPIO pins play a crucial role in facilitating communication between the microcontroller and its environment.

Understanding and effectively utilizing GPIO pins is essential for developing Arduino projects of varying complexity, from simple blinking LED demonstrations to advanced robotics and automation applications.

2.3.3.3 Analog Input Pins

Analog input pins are essential components of microcontrollers like the ATmega2560 found on the Arduino Mega 2560 board. Here's an explanation of analog input pins and their role:

Functionality: Analog input pins allow the microcontroller to read analog voltage levels from external sensors, potentiometers, or other analog devices. Unlike digital input pins, which can only detect two states (HIGH or LOW), analog input pins can measure a range of voltage levels and convert them into digital values for processing.

Analog-to-Digital Conversion (ADC): The analog input pins are connected to the microcontroller's built-in ADC (Analog-to-Digital Converter), which samples the analog voltage applied to the pin and converts it into a digital value. The ADC resolution determines the number of discrete steps the analog voltage is divided into, typically ranging from 8 to 12 bits in microcontrollers like the ATmega2560.

Range: The analog input pins on the Arduino Mega 2560 are labelled A0 through A15, indicating the 16 available analog input channels. Each analog input pin can measure voltage levels within a specific range, usually 0 to 5 volts (corresponding to 0 to 1023 digital values for a 10-bit ADC).

Resolution: The resolution of the ADC determines the granularity of the analog-to-digital conversion. For example, a 10-bit ADC can represent analog voltages using 2^10 (1024) discrete digital values, providing higher precision compared to lower-resolution ADCs.

Usage: Analog input pins are commonly used in Arduino projects to interface with analog sensors such as temperature sensors, light sensors, humidity sensors, and potentiometers. By reading the analog voltage levels from these sensors, the microcontroller can gather real-world data and respond accordingly based on the programmed logic.

Configuration: Analog input pins are configured in the Arduino IDE using functions such as “analogRead()” to read the analog voltage levels applied to the pin. The returned value represents the digital equivalent of the analog voltage and can be used for further processing or decision-making in the Arduino sketch.

Understanding and effectively utilizing analog input pins is essential for interfacing with the analog world in Arduino projects. They enable the microcontroller to sense and respond to changes in physical quantities such as light, temperature, and pressure, expanding the range of applications that can be implemented with the Arduino platform.

2.3.3.4 Serial Communication Ports (UART)

Serial communication ports, also known as UART (Universal Asynchronous Receiver-Transmitter) ports, are crucial features of microcontrollers like the ATmega2560 found on the Arduino Mega 2560 board. Here's an explanation of serial communication ports and their role:

Functionality: Serial communication ports allow the microcontroller to exchange data with other serial devices, such as computers, sensors, GPS modules, Bluetooth modules, and other microcontrollers. They provide a bidirectional communication interface for transmitting and receiving serial data streams.

Asynchronous Communication: UART serial communication is asynchronous, meaning that data is transmitted without a clock signal. Instead, data is transmitted one bit at a time, along with start and stop bits to delineate the data frame. This simplicity makes UART communication suitable for many applications and allows devices with different clock speeds to communicate effectively.

Serial Ports on Arduino Mega 2560:

Serial: The Arduino Mega 2560 has a hardware UART serial port, commonly referred to as Serial, which is connected to pins 0 (RX) and 1 (TX) on the board. This serial port is used for programming the board and for serial communication with the computer through the USB-to-serial converter.

Serial1, Serial2, Serial3: In addition to the primary Serial port, the Arduino Mega 2560 has three additional hardware UART serial ports labelled Serial1, Serial2, and Serial3. These ports provide additional options for serial communication with external devices, sensors, or other microcontrollers. They are available on pins 19/18, 17/16, and 15/14, respectively.

Baud Rate: Serial communication requires both the transmitting and receiving devices to agree on a common baud rate, which specifies the transmission speed in bits per second (bps).

Common baud rates include 9600, 19200, 38400, 57600, and 115200 bps, among others. The baud rate must be set the same on both communicating devices for successful data exchange.

Data Format: UART communication typically uses a data frame consisting of a start bit, followed by 8 data bits, an optional parity bit (for error checking), and one or more stop bits. The data format, including the number of data bits, parity, and stop bits, must be configured identically on both the transmitting and receiving devices.

Usage: Serial communication ports are widely used in Arduino projects for interfacing with various external devices and sensors, such as GPS modules, RFID readers, Bluetooth modules, and more. They provide a convenient and efficient means of exchanging data between the microcontroller and external peripherals.

Understanding and effectively utilizing serial communication ports is essential for developing Arduino projects that require communication with external devices or systems. By leveraging the UART capabilities of the microcontroller, Arduino enthusiasts can create a wide range of interactive and interconnected projects.

2.3.3.5 Serial Peripheral Interface (SPI)

SPI (Serial Peripheral Interface) is a synchronous serial communication protocol commonly used for communicating between microcontrollers and peripheral devices such as sensors, displays, SD cards, and other microcontrollers. Here's an explanation of SPI and its key features:

Synchronous Communication: SPI is a synchronous communication protocol, meaning that data is transmitted synchronously with a clock signal. This allows for high-speed data transfer between devices with precise timing.

Master-Slave Architecture: SPI typically operates in a master-slave architecture, where one device (the master) controls the communication and timing, while one or more devices (the slaves) respond to commands and transmit data.

Communication Lines: SPI uses four communication lines:

MOSI (Master Out Slave In): The master sends data to the slave(s) on this line.

MISO (Master In Slave Out): The slave(s) send data back to the master on this line.

SCK (Serial Clock): The clock signal generated by the master to synchronize data transfer.

SS (Slave Select): This line is used by the master to select which slave device it wants to communicate with. Each slave has its own SS line, allowing the master to communicate with multiple slaves on the same bus.

Full-Duplex Communication: SPI supports full-duplex communication, meaning that data can be transmitted simultaneously in both directions. The master can send data to the slave(s) while receiving data from them during the same transaction.

Data Format: SPI data is typically transmitted in a serial stream of bytes, with each byte transmitted sequentially. The data format (such as the number of bits per byte, the order of transmission, and clock polarity/phase) can be configured based on the requirements of the devices being interfaced.

Speed: SPI supports high-speed communication, with data rates typically ranging from a few hundred kilobits per second (Kbps) to several megabits per second (Mbps), depending on the microcontroller and peripheral devices.

Usage: SPI is commonly used in Arduino projects for interfacing with a wide range of peripheral devices that support SPI communication, such as TFT displays, Ethernet controllers, temperature sensors, and more. Many Arduino boards, including the Arduino Mega 2560, have dedicated SPI pins for easy connection to SPI devices.

Understanding SPI and its capabilities is essential for effectively interfacing with SPI-compatible devices in Arduino projects. By leveraging SPI communication, Arduino enthusiasts can create projects that interface with a variety of sensors, displays, and other peripherals, enabling rich and interactive applications.

2.3.3.6 Inter-Integrated Circuit (I^2C)

I^2C (Inter-Integrated Circuit) is a serial communication protocol commonly used for communication between microcontrollers and peripheral devices such as sensors, displays, EEPROMs, and other integrated circuits. Here's an explanation of I^2C and its key features:

Two-Wire Communication: I^2C uses only two communication lines:

SDA (Serial Data Line): This bidirectional line is used for transferring data between the master (microcontroller) and slave (peripheral) devices.

SCL (Serial Clock Line): This line carries the clock signal generated by the master to synchronize data transfer between devices.

Master-Slave Architecture: I^2C operates in a master-slave architecture, where one device (the master) initiates and controls communication with one or more slave devices.

Addressing: Each slave device on the I^2C bus has a unique 7-bit or 10-bit address assigned to it. The master device uses these addresses to select the specific slave device it wants to communicate with.

Multi-Master Support: I^2C supports multi-master operation, allowing multiple master devices to share the same bus. Arbitration mechanisms ensure that only one master can control the bus at a time to prevent data collisions.

Data Format: Data transmission on the I^2C bus typically occurs in packets consisting of a start condition, followed by the device address (including read/write bit), data bytes, and a stop condition. Each byte of data is transmitted sequentially, with acknowledgment from the receiving device.

Clock Speed: The clock speed on the I^2C bus is typically configurable and can range from a few kilohertz to several hundred kilohertz, depending on the devices involved and the bus capacitance.

Usage: I^2C is commonly used in Arduino projects for interfacing with a variety of peripheral devices that support I^2C communication, such as temperature sensors, accelerometers, gyros, OLED displays, EEPROMs, and more. Many Arduino boards, including the Arduino Mega 2560, have dedicated I^2C pins (SDA and SCL) for easy connection to I^2C devices.

Understanding I^2C and its capabilities is essential for effectively interfacing with I^2C compatible devices in Arduino projects. By leveraging I^2C communication, Arduino enthusiasts can create projects that interface with a wide range of sensors, displays, and other peripherals, enabling rich and interactive applications.

2.3.3.7 Power Supply Pins

Power supply pins on the Arduino Mega 2560 provide voltage and ground connections for powering the board and external components. Here's an overview of the power supply pins on the Arduino Mega 2560:

Vin (Voltage Input): The Vin pin allows to supply an external voltage to power the Arduino Mega 2560. This voltage typically ranges from 7V to 12V DC. The onboard voltage regulator regulates this input voltage to provide a stable 5V supply to the microcontroller and other components on the board.

5V Pin: The 5V pin provides a regulated 5V output from the onboard voltage regulator. This pin can be used to power external sensors, modules, or other components that require a 5V supply.

3.3V Pin: The 3.3V pin provides a regulated 3.3V output from the onboard voltage regulator. This lower voltage level is suitable for powering certain sensors, modules, or other components that require a 3.3V supply.

GND (Ground) Pins: There are several ground (GND) pins distributed throughout the Arduino Mega 2560 board. These pins provide the reference ground connection for the board and external components. They should be connected to the ground terminals of external power supplies, sensors, and other devices to complete the circuit.

Usage: The power supply pins on the Arduino Mega 2560 allow to power the board and connected components from various sources, such as USB, external power supplies, or batteries. Depending on the project requirements, the appropriate power source can be chosen and connected to the Vin pin or the 5V and GND pins to provide power to the board and peripherals.

Understanding the power supply pins is essential for properly powering the Arduino Mega 2560 board and ensuring reliable operation of connected components in our projects. Always refer to the Arduino Mega 2560 documentation for specific details and guidelines regarding power supply connections and requirements.

2.3.3.8 Crystal Oscillator

The crystal oscillator is a critical component of many microcontrollers, including the ATmega2560 found on the Arduino Mega 2560 board. Here's an explanation of the crystal oscillator and its role:

Functionality: The crystal oscillator provides the clock signal that synchronizes the operation of the microcontroller. It generates a precise and stable oscillating signal, typically at a specific frequency specified by the crystal used.

Clock Signal: The clock signal generated by the crystal oscillator serves as the timing reference for all operations performed by the microcontroller. It determines the rate at which instructions are executed, data is transferred, and peripherals are synchronized.

External Crystal or Resonator: The crystal oscillator circuit typically consists of an external crystal or ceramic resonator connected to the microcontroller. The crystal/resonator oscillates at a specific frequency when voltage is applied, generating the clock signal for the microcontroller.

Accuracy: Crystals are chosen for their high accuracy and stability, which is essential for precise timing in microcontroller applications. The frequency of the crystal is specified with high precision, allowing for reliable operation of the microcontroller over a wide range of operating conditions.

Configuration: The frequency of the crystal oscillator is determined by the selection of the crystal or resonator component. Common frequencies used in microcontroller applications include 8 MHz, 16 MHz, and other values depending on the specific requirements of the application.

Usage: The crystal oscillator is a fundamental component of the microcontroller circuitry and is essential for proper operation of the Arduino Mega 2560 board. It ensures that all internal operations are synchronized and executed at the correct rate, enabling the microcontroller to perform tasks accurately and reliably.

Understanding the crystal oscillator and its role in providing timing and synchronization is crucial for designing and working with microcontroller-based systems like the Arduino Mega 2560. It ensures that our projects operate correctly and reliably under various conditions.

2.3.3.9 Reset Circuitry

The reset circuitry in a microcontroller, including the ATmega2560 found on the Arduino Mega 2560 board, manages the reset functionality of the device. Here's an explanation of the reset circuitry and its role:

Reset Button: The most visible component of the reset circuitry is the reset button on the Arduino board. Pressing this button triggers a reset of the microcontroller, restarting its operation from the beginning.

External Reset: In addition to the reset button, the microcontroller can also be reset externally by applying a low voltage pulse to the reset pin (typically labelled RESET). This allows for reset control from external circuits or devices.

Power-On Reset (POR): When power is initially applied to the microcontroller, the reset circuitry ensures that the device starts in a known state by performing a power-on reset. This resets the microcontroller's internal registers and initializes its peripherals to their default states.

Brown-Out Detection (BOD): Some microcontrollers, including the ATmega2560, feature brown-out detection circuitry. This monitors the voltage level of the power supply and triggers a reset if the voltage falls below a certain threshold. This helps prevent erratic behaviour or data corruption that can occur when the supply voltage drops too low.

Low Voltage Programming (LVP): In some microcontrollers, including AVR microcontrollers like the ATmega2560, there is a feature called Low Voltage Programming. It allows the microcontroller to be reprogrammed even if the voltage level is lower than the normal operating voltage. This feature is often integrated into the reset circuitry.

Fuse Settings: The reset circuitry is also responsible for managing fuse settings in the microcontroller. Fuses are special bits of memory that control various aspects of the microcontroller's behaviour, such as clock source, brown-out detection level, and bootloader protection. These settings can sometimes affect the behaviour of the reset circuitry.

Overall, the reset circuitry ensures that the microcontroller operates reliably and consistently by providing mechanisms for resetting the device under various conditions, including power-up, manual intervention (reset button), and external control. Understanding the reset circuitry is important for troubleshooting and designing robust embedded systems using microcontrollers like the ATmega2560.

2.3.4 Interrupts of Arduino Mega 2560 Microcontroller

The interrupts are used to measure frequency in our present work so now let's about Interrupts of Arduino mega 2560 microcontroller. Interrupts are a mechanism that allows the microcontroller to respond to an event and execute a specific block of code immediately, interrupting the normal flow of the program.

The Arduino Mega 2560, which uses the ATmega2560 microcontroller, supports several types of interrupts. Those are External Interrupts, Pin Change Interrupts, Timer Interrupts, Serial Communication Interrupts (USART), ADC Conversion Complete Interrupts, SPI and I2C Interrupts, EEPROM Ready Interrupts.

External Interrupts:

The Arduino Mega 2560 has 6 external interrupt pins: 2, 3, 18, 19, 20, and 21. These interrupts can be triggered in several ways:

LOW: Trigger the interrupt whenever the pin is low.

CHANGE: Trigger the interrupt whenever the pin changes value (rising or falling).

RISING: Trigger the interrupt when the pin transitions from low to high.

FALLING: Trigger the interrupt when the pin transitions from high to low.

Pin Change Interrupts:

Pin change interrupts on the Arduino Mega 2560 allow to monitor multiple pins for any changes in their state. Unlike external interrupts, which are limited to specific pins, pin change interrupts can be used on almost any pin.

The ATmega2560 microcontroller has three pin change interrupt control registers: PCICR, PCMSK0, PCMSK1, and PCMSK2.

PCICR (Pin Change Interrupt Control Register): This register enables pin change interrupts.

PCMSK0, PCMSK1, PCMSK2 (Pin Change Mask Registers): These registers select which pins can trigger the interrupt.

Timer Interrupts:

Timer interrupts on the Arduino Mega 2560 are used to perform tasks at specific intervals, independently of the main program flow. The ATmega2560 microcontroller has six hardware timers: Timer0, Timer1, Timer2, Timer3, Timer4, and Timer5. These timers can be configured to generate interrupts at regular intervals. Where:

Timer0: 8-bit timer (used for functions like “delay()” and “millis()”).

Timer1: 16-bit timer.

Timer2: 8-bit timer.

Timer3, Timer4, Timer5: 16-bit timers.

Different Timer Registers are used for different purposes like

TCCRnA and TCCRnB Timer/Counter Control Registers used for configuring the mode and prescaler.

TCNTn Timer/Counter Register holds the current count value.

OCRnA/B/C Output Compare Registers determine the count value at which an interrupt is triggered.

TIMSKn Timer/Counter Interrupt Mask Register, which enables the interrupt.

TIFRn Timer/Counter Interrupt Flag Register, which indicates if an interrupt has occurred.

Serial Communication Interrupts (USART):

Serial communication interrupts in the Arduino Mega 2560 are a way to handle data reception and transmission efficiently without constantly polling the serial port. The ATmega2560 microcontroller has four hardware serial ports (Serial, Serial1, Serial2, and Serial3), each capable of generating interrupts for various events. Those are:

USART_RX_vect: This interrupt is triggered when a byte of data is received.

USART_UDRE_vect: This interrupt is triggered when the data register is empty, indicating that the USART is ready to receive more data to transmit.

USART_TX_vect: This interrupt is triggered when the entire data has been transmitted.

ADC Conversion Complete Interrupts:

Analog-to-Digital Conversion (ADC) interrupts in the Arduino Mega 2560 allow to execute a block of code when an ADC conversion is complete. This is useful for applications that require precise timing or when the analog data should be processed immediately after conversion. The ATmega2560 microcontroller has an ADC that can be configured to generate an interrupt when an analog-to-digital conversion is complete. This interrupt is called ADC_vect.

SPI and I2C Interrupts:

Both SPI (Serial Peripheral Interface) and I2C (Inter-Integrated Circuit) are communication protocols used to transfer data between microcontrollers and peripherals. The Arduino Mega 2560 supports interrupts for both SPI and I2C, allowing efficient handling of data transfer without constantly polling the communication status. SPI interrupts on the Arduino Mega 2560 can be used to handle the completion of data transmission or reception. The SPI module in the ATmega2560 microcontroller generates an interrupt when a data transmission is complete.

EEPROM Ready Interrupts:

The ATmega2560 microcontroller on the Arduino Mega 2560 supports EEPROM Ready interrupts, which are triggered when the EEPROM write operation is complete. This allows to perform other tasks while waiting for the EEPROM to finish writing, rather than blocking the program. This action happened in two steps:

Enable EEPROM Ready Interrupt: the EEPROM Ready interrupt should be enabled.

Interrupt Service Routine (ISR)**:** Define the ISR to handle the EEPROM interrupt.

2.4 16×2 LCD keypad shield

The 16x2 LCD keypad module is used in our present project to display the final result from Arduino mega 2560 microcontroller. So, now let's learn about 16x2 LCD keypad module. The 16×2 LCD Keypad Shield is a versatile component that integrates a 16x2 character LCD display and a set of push buttons on a single board. This shield makes it easy to add a user interface to the Arduino projects, allowing display information and capture user input without needing separate components for the LCD and buttons.

LCD Panel:

The display area is divided into 32 cells (16 per line) where each cell can display one character. The backlight of LCD panel provides illumination for the display, making it readable in various lighting conditions.

Push Buttons:

The 16×2 LCD Keypad Shield typically comes with five push buttons (SELECT, RIGHT, UP, DOWN, LEFT) and an optional reset button. These buttons are connected to the Arduino via a single analog pin (usually A0) through a resistor network. The buttons are connected in a voltage divider network. When a button is pressed, a specific voltage is sent to the analog pin A0. Each button produces a different voltage level on A0 when pressed.

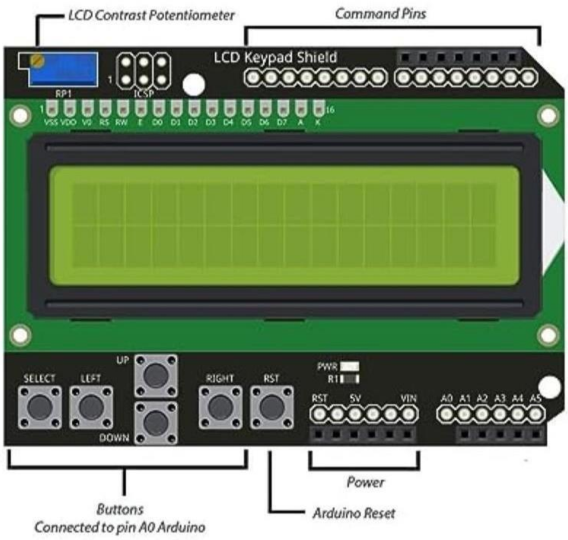


Fig 2.6: 16×2 LCD Keypad Shield

Power Supply:

VSS: Ground pin.

VDD: +5V power supply pin.

A (Anode): Connected to +5V through a current-limiting resistor (typically 220 ohms) to power the LCD backlight.

K (Cathode): Connected to ground.

Control Pins:

RS (Register Select): Determines whether the data sent to the LCD is interpreted as command or character data. RS=0 selects command register, RS=1 selects data register.

RW (Read/Write): Selects the LCD mode. RW=0 for writing data to the LCD, RW=1 for reading data from the LCD. Typically grounded (RW=0) for write-only operations.

E (Enable): Activates data writing to the LCD. A high-to-low transition on this pin latches the data into the display.

Data Pins:

D0-D3: Lower 4 data bits, used in 8-bit mode.

D4-D7: Higher 4 data bits, always used. In 4-bit mode, only these pins are used for data transmission.

Contrast Adjustment:

V0 pin is connected to a potentiometer to adjust the LCD contrast. Adjusting the potentiometer changes the voltage applied to this pin, which affects the contrast of the displayed characters.

Backlight:

A (Anode): Positive terminal of the backlight LED, connected to +5V through a resistor.

K (Cathode): Negative terminal of the backlight LED, connected to ground.

Internal Components

LCD Driver IC: A specialized integrated circuit that controls the individual pixels of the LCD based on the data and commands received.

Segment and Common Electrodes: Internal electrodes that create electric fields to manipulate the liquid crystals and form characters on the display.

Specifications of 16×2 LCD Keypad Shield:

Display Size: 16 characters × 2 lines
Character Size: 5×8 dots
Display Type: Alphanumeric LCD
Operating Voltage: 4.7V to 5.3V (commonly powered by 5V)
Supply Current: 1-2 mA without backlight, up to 20 mA with backlight
Logic Level: TTL compatible

Pin Configuration and Connections:

LCD Pin	Function	Arduino Pin
1	VSS (Ground)	GND
2	VDD (+5V)	5V
3	V0 (Contrast)	Potentiometer wiper (middle pin)
4	RS (Register Select)	Digital Pin 8
5	RW (Read/Write)	GND
6	E (Enable)	Digital Pin 9
11	D4 (Data)	Digital Pin 4
12	D5 (Data)	Digital Pin 5
13	D6 (Data)	Digital Pin 6
14	D7 (Data)	Digital Pin 7
15	A (Anode - Backlight)	5V through 220-ohm resistor
16	K (Cathode - Backlight)	GND
Analog	Button inputs	Analog Pin A0

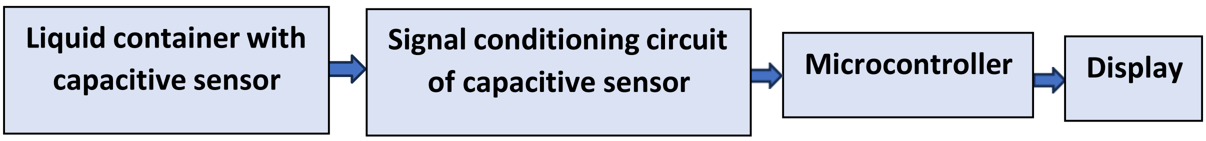
Chapter 3: Present work

3.1 Introduction

The objective of our present work is to measure the level of conductive liquid of the container using a capacitive sensor. The present work can be applicable for any conductive liquid. Those liquids which are corrosive and as well as conductive may be measured using this scheme. Therefore, the level of the liquid should be measured using non-contacting method (without contacting the liquid).

The capacitive sensor is used as a liquid level sensor in the present work. So, the capacitance of the capacitive sensor will change with change of the level of the liquid in the container. Then the capacitive sensor is connected to the signal conditioning circuit to generate a voltage signal which changes with capacitance change of capacitive sensor. Then signal conditioning circuit send the generated voltage signal as an input signal to the microcontroller. The microcontroller decodes the liquid level information from the input signal. The Curve Fitting method is used to decode the liquid level information from the input signal. Then the microcontroller sends the final result or liquid level data to the display. And lastly, the liquid level data is displayed in the display.

3.2 Block Diagram of Present Work



3.3 Liquid Container with the Capacitive Sensor

The present work includes a capacitive sensor. Whose capacitance changes with the liquid level. But in this case, the liquids which is corrosive as well as conductive are also used. So, a capacitive sensor should be made as, it's can't come to contact with the liquid. Otherwise, if corrosive liquid is used then it will damage the capacitive sensor. If capacitive sensor gets damaged it's won't give proper data about capacitance change and it won't survive a long time.

At first, a jar made of glass was taken as shown in Fig 3.1. It will contain the conductive liquid whose level will be measured. The glass jar is used because glass is corrosive resistant materials so, it will also contain corrosive liquid without getting damaged. Also, the liquid level can be noted from outside of the container as glass is a transparent material.

To build a capacitor, two conductive surfaces are needed and dielectric material between them. Above the glass jar it has a metal cap which will act as a conductive surface of capacitor. Another conductive surface will form above conductive liquid because conductive liquid is used.

Between two conductive surfaces there are two layers of dielectric material, they are air and Teflon. After metal cap, the Teflon is used as a dielectric material as shown in Fig 3.2. A 1.5 cm thick Teflon sheet is taken. Then its measurement was taken using jar cap and mark the part which is need to cut out from the sheet. Then using Hacksaw blade, a Teflon disk was cut out from its sheet which is fit into the metal cap. Teflon was used as a dielectric material because it has some important advantages. Teflon is a corrosive resistant compound so; it doesn't get damaged and protect metal cap from contact of corrosive liquid. Teflon also has high relative permittivity that is 2.05. As value of capacitor or capacitance is directly proportional to relative permittivity. So, it increases capacitance of the capacitive sensor respectively, so that sensitivity of the capacitive sensor will also increase. After Teflon next layer of dielectric is air. Air has very low value of relative permittivity that is 1.0006. Which is very close to relative permittivity of vacuum that is 1. When the conductive liquid is taken out from tap of the jar or level of conductive liquid decrease then thickness of layer of air between conductive surfaces increases. But thickness of layer of Teflon always remains constant.

Fig 3.1 liquid container with capacitive sensor

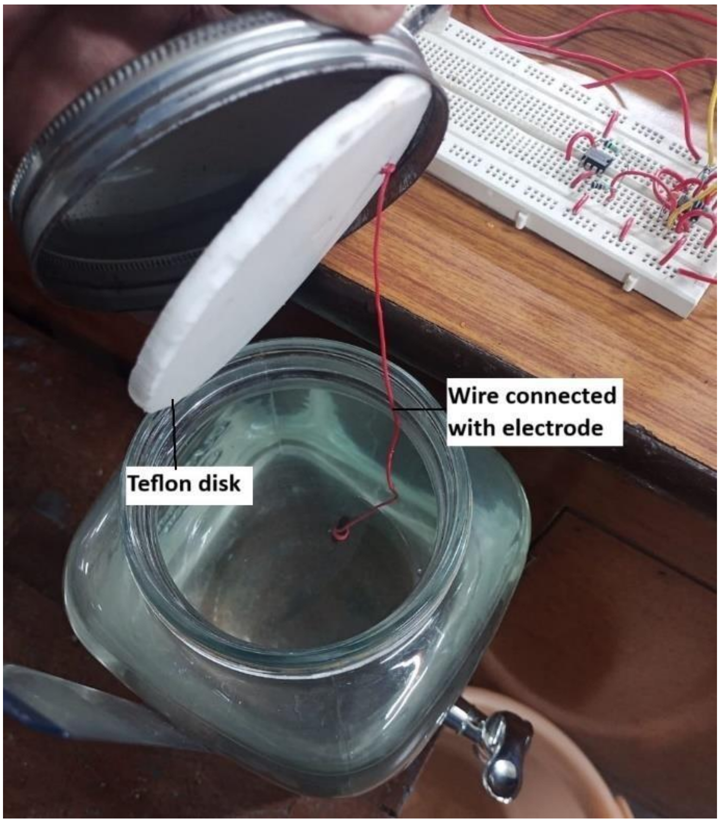


Fig 3.2: Capacitive sensor inside view

Then a wire was connected with each conductive surface to build a capacitor. The wire was connected with metal cap of the jar using solder and soldering iron. And another wire was connected to an electrode made out of iron, which is submerged into the conductive liquid. Presently a wire was submerged into the liquid through metal cap and Teflon disk to bottom of the jar. To do that a hole was drilled into the metal cap and Teflon disk then pass a wire between them. And at the end of the wire there is the metal electrode near bottom of the jar. This structure is satisfactory for the experimenting. Because presently dilute saline water is used which is a conductive but non-corrosive liquid. So, presently there will be no damage to the electrode and the wire connected with the electrode. But in case of highly corrosive liquid, then there should be an electrode through glass at the bottom of the jar and wire should be connected with the electrode from outside of the jar. Also, electrode should be made from corrosion resistant material. There are some metal electrodes made of platinum or gold or titanium are very resistant to corrosive liquid but these electrodes are costly. So, graphite or carbon electrodes can be used. They are corrosion resistant as well chipper than other corrosion resistant electrodes. And the ground should be connected with the electrode of this capacitor because current through ground is very less in a capacitor, which will further reduce the damage in the electrode.

A scale was attached with the glass jar to measure the conductive liquid level manually. Compering this data, the microcontroller was trained to calculate level of the conductive liquid from voltage signal of signal conditioning circuit.

Equations of capacitance of capacitive sensor:

For a capacitor which has two layers of different dielectric, it will act as two capacitors connected in series. If value of those capacitors is C_1 and C_2 then equation for the capacitance is given below.

$$C_1 = \frac{\varepsilon_0 \varepsilon_r A}{D} \tag{3.1}$$

Where C_1 is capacitance of layer with Teflon
ε_0 is permittivity of vacuum
ε_r is relative permittivity of Teflon
A is cross section of capacitor
D is thickness of Teflon layer

$$C_2 = \frac{\varepsilon_0 A}{l} \tag{3.2}$$

Where C_2 is capacitance of layer with Teflon
ε_0 is permittivity of vacuum
A is cross section of capacitor
l is thickness of air layer

Now equation overall capacitance (C) of capacitive sensor will be:

$$C = \frac{C_1 C_2}{C_1 + C_2} = \frac{\frac{\varepsilon_0 \varepsilon_r A}{D} \cdot \frac{\varepsilon_0 A}{l}}{\frac{\varepsilon_0 \varepsilon_r A}{D} + \frac{\varepsilon_0 A}{l}} = \frac{\varepsilon_0 \varepsilon_r A}{\varepsilon_r l + D} \tag{3.3}$$

Equation for thickness of air layer (l) will be:

$$l = \frac{\varepsilon_0 A}{C} - \frac{D}{\varepsilon_r} \tag{3.4}$$

Now when thickness of air layer is l_1 then equation capacitance is:

$$C_1 = \frac{\varepsilon_0 \varepsilon_r A}{\varepsilon_r l_1 + D} \tag{3.5}$$

And when thickness of air layer is l_2 then equation capacitance is:

$$C_2 = \frac{\varepsilon_0 \varepsilon_r A}{\varepsilon_r l_2 + D} \tag{3.6}$$

Equation of capacitance change (ΔC) with change in thickness of air layer is:

$$\Delta C = C_1 - C_2 = \frac{\varepsilon_0 \varepsilon_r A}{\varepsilon_r l_1 + D} - \frac{\varepsilon_0 \varepsilon_r A}{\varepsilon_r l_2 + D}$$

$$\Delta C = \varepsilon_0 \varepsilon_r A \, . \, \frac{\varepsilon_r l_2 + D - \varepsilon_r l_1 - D}{(\varepsilon_r l_1 + D)(\varepsilon_r l_2 + D)}$$

$$\Delta C = \varepsilon_0 \varepsilon_r^2 A \cdot \frac{l_2 - l_1}{(\varepsilon_r l_1 + D)(\varepsilon_r l_2 + D)} \qquad (3.7)$$

Now, if the height of the glass jar is h and the thickness of air layer is l, so level of the conductive liquid (W_l) will be:

$$W_l = h - l \qquad (3.8)$$

$$l = h - W_l \qquad (3.9)$$

$$l_1 = h - W_{l1} \qquad (3.10)$$

$$l_2 = h - W_{l2} \qquad (3.11)$$

The equation of capacitance (C) of the capacitance sensor is obtained from equation (3.3) and equation (3.8):

$$C = \frac{\varepsilon_0 \varepsilon_r A}{\varepsilon_r (h - W_l) + D} \qquad (3.12)$$

The equation of capacitance change (ΔC) is obtained from equation (3.7), equation (3.10) and equation (3.11):

$$\Delta C = \varepsilon_0 \varepsilon_r^2 A \cdot \frac{W_{l1} - W_{l2}}{(\varepsilon_r (h - W_{l1})) + D)(\varepsilon_r (h - W_{l2}) + D)} \qquad (3.13)$$

3.4 Suitable signal conditioning circuit of capacitive sensor

The capacitive sensor is connected to the signal conditioning circuit to generate a voltage signal which changes with capacitance change of capacitive sensor. The generated voltage signal is further taken as input signal in the microcontroller. There are many types of signal conditioning circuit like bridge circuit (Schering bridge), RC oscillator circuit, 555-timer based astable multivibrator circuit etc.

The Schering bridge circuit can be used as the signal conditioning circuit of the capacitive sensor. Where changes in capacitance of the capacitive sensor cause an imbalance in the bridge, resulting in a measurable output signal (analog voltage signal). But in this case the size of the capacitive sensor is pretty large. So, there are high possibility that the capacitive sensor will catch noise from the environment. Now, if Schering bridge was used circuit as the signal conditioning circuit, then the output signal will get distorted and that signal can't be used for further processing. So, a signal conditioning circuit which will generate the frequency signal (frequency of output signal will change with capacitance change of the capacitive sensor) will be most suitable for present operation.

The RC oscillator circuit and 555-timer based astable multivibrator circuit can be used as the signal conditioning circuit of the capacitive sensor which will generate the frequency signal. Between those two circuits a 555-timer based astable multivibrator circuit is used as the signal conditioning circuit of the capacitive sensor. Because, 555-timer based astable multivibrator circuit is relatively simple to configure and use compared to RC oscillator circuit. Also, 555-timer based astable multivibrator circuit is low-cost integrated circuit and can generate a wide range of frequencies and duty cycles by adjusting the values of external resistors and capacitors. The frequency signal is used as the input signal of microcontroller then signal should be digital signal where, RC oscillator circuit generates sinusoidal wave signal and 555-timer based astable multivibrator circuit square wave signal (which can be used as digital signal). Because of those advantages 555-timer based astable multivibrator circuit as the signal conditioning circuit will be most suitable for present operation.

3.5 555-timer based astable multivibrator circuit as the signal conditioning circuit

NE555 IC (555-timer) is used for our signal conditioning circuit. The 555-timer is configured in astable mode, which means it will oscillate between high and low states without any external trigger. The circuit includes two resistors (denoted as R_1 and R_2) and a capacitor of the capacitive sensor (denoted as C), which determine the timing intervals for the oscillations. Initially, capacitor C is discharged, and the output at pin 3 is low. When power is applied, capacitor C starts charging through resistors R_1 and R_2. Once the voltage across capacitor C exceeds 2/3 of Vcc (threshold voltage), the internal comparator resets the flip-flop, causing the output at pin 3 to go low. Capacitor C then discharges through resistor R_2 to pin 7 (discharge). Once the voltage across capacitor C falls below 1/3 of Vcc (trigger voltage), the flip-flop is set again, and the output at pin 3 goes high. This cycle repeats, resulting in a continuous square wave output at pin 3 with a frequency determined by the values of resistors R_1, $R_2$and capacitor C.

Circuit Diagram of Astable Multivibrator Circuit using NE555 Timer IC:

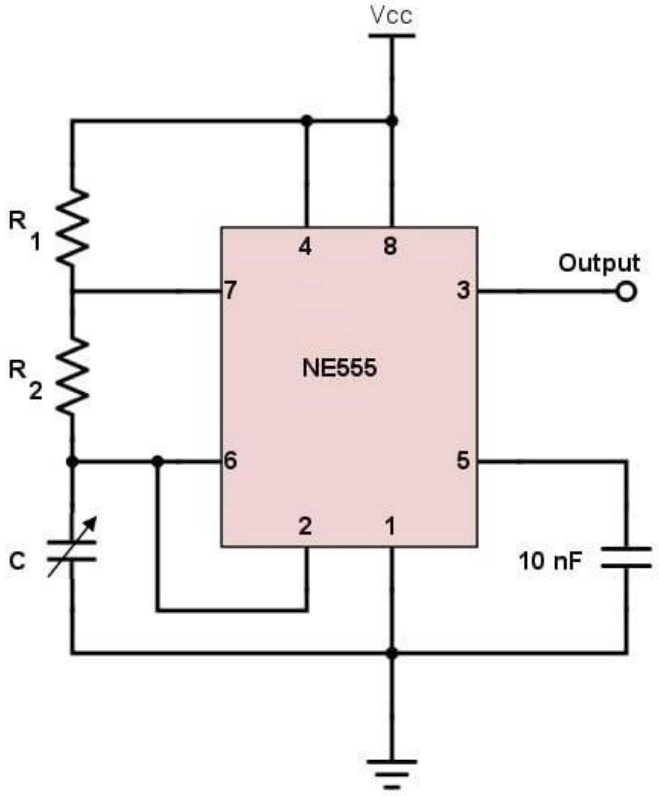


Fig 3.3 Astable multivibrator circuit using NE555 timer IC

Operation and Results of the 555-Timer Based Astable Multivibrator:

At first, experimentation was started with different value of standard resistors with the capacitive sensor to get the suitable value of resistor for signal conditioning circuit. The full-scale frequency change (approximately 1.2kHz) was obtained from output signal after subtracting frequency at maximum liquid level and the frequency at minimum liquid level. Then the sensitivity of the system was calculated, which was pretty low. The sensitivity of the system low because range of capacitance of the capacitive sensor is pretty low (within 11pF to 14pF).

Also, frequency of the output signal of the 555-timer based astable multivibrator is fluctuating (as shown in the Fig 3.4) because the capacitive sensor is prone to noise of the environment. To solve those two problems, a capacitance multiplier between capacitive sensor and 555-timer based astable multivibrator circuit will be most suitable for present operation.

Fig 3.4: output signal of 555-timer based astable multivibrator circuit

3.6 Requirement of the Capacitance Multiplier

A capacitance multiplier is a circuit configuration used to amplify the capacitance of the capacitor. As the value of our capacitive sensor is very less (11pF to 14pF) so, sensitivity of the system was pretty low. Also, the frequency of the output signal of the 555-timer based astable multivibrator is fluctuating (as shown in the Fig 3.4) because of the environmental noise. Those problems can be eliminated after using capacitance multiplier. The advantages offered by capacitance multiplier in our present work are:

Increased sensitivity: By amplifying the capacitance change detected by the sensor, capacitive multipliers enhance the sensitivity of the sensor system. This allows for more precise measurement or detection of small changes in capacitance, improving overall performance.

Extended detection range: Capacitance multiplier enable the detection of a wider range of capacitance values, extending the operational range of the capacitive sensor. This is particularly useful in applications where the liquid level or other parameters being measured may vary widely.

Improved signal-to-noise ratio: By amplifying the signal produced by the capacitive sensor, capacitance multiplier helps to improve the signal-to-noise ratio of the sensor system. This results in cleaner, more reliable output signals, especially in noisy environments.

Flexibility: Capacitance multiplier can be designed with adjustable gain or amplification factors, providing flexibility to optimize the sensor system for different applications or environments. This allows for fine-tuning of the sensitivity and range to meet specific requirements.

Reduced susceptibility to interference: Amplifying the capacitance change at the sensor helps to reduce the effects of external interference or noise on the sensor signal. This improves the robustness and reliability of the sensor system, especially in electrically noisy environments.

3.7 Capacitance Multiplier Circuit using Operational Amplifier

Component Requirements: Op-amp, resistors (R_1 and R_2), capacitor (C_1).

Circuit Diagram:

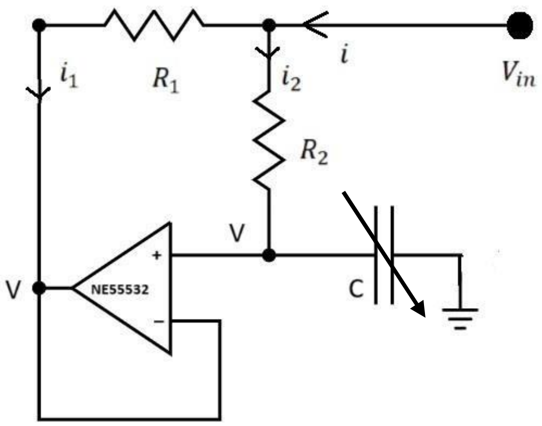


Fig 3.5: Capacitance multiplier circuit using Op-amp

Circuit Explanation:

Op-amp (+ve) pin: Connected to input through resistor R_1 and connected to ground through capacitor C_1.

Op-amp (-ve) pin: Connected to output pin of Op-amp.

Op-amp output pin: Connected to input through resistor R_2 and to (-ve) pin of Op-amp

Equations for Capacitance Multiplier:

Output of Op-amp is connected with (-ve) of Op-amp. So, Op-amp will act like a buffer, where voltage of (+ve) and output of Op-amp will be same. Now (+ve) of Op-amp is connected with capacitive sensor so, voltage of capacitive sensor and output of Op-amp will be same. Now equations for present circuit are given below:

Equation of charge (Q) in capacitive sensor is:

$$Q = CV = i_2 t \tag{3.14}$$

Where Q is charge in capacitive sensor

C is capacitance of capacitive sensor

i_2 is current flowing in capacitive sensor

t is time of current flowing in capacitive sensor

Equation of voltage different between input voltage (V_{in}) of capacitance multiplier and capacitive sensor voltage (V) is:

$$V_{in} - V = i_1 R_1 = i_2 R_2 \tag{3.15}$$

$$or, \qquad i_1 = i_2 \frac{R_2}{R_1} \tag{3.16}$$

Where $i_1$is current flowing in to Op-amp

R_1, R_2 is resistance used in capacitance multiplier

Now, capacitance multiplier forms the synthesized capacitor (C'). The equation of charge (Q') in synthesized capacitor of capacitance multiplier is:

$$Q' = C'V = it \tag{3.17}$$

Where C' is capacitance of synthesized capacitor of capacitance multiplier

i is current in synthesized capacitor or total current of capacitance multiplier

Equation of current (i) in synthesized capacitor or total current of capacitance multiplier is:

$$i = i_1 + i_2 \tag{3.18}$$

After putting value of i from equation (3.18) in equation (3.17), obtained equation will be:

$$C'V = (i_1 + i_2)t \tag{3.19}$$

After putting value of i_1 from equation (3.16) in equation (3.19), obtained equation will be:

$$C'V = (i_2 \frac{R_2}{R_1} + i_2)\, t$$

$$or, \qquad C'V = i_2 t\, (1 + \frac{R_2}{R_1}) \tag{3.20}$$

After putting value of $i_2 t$ from equation (3.15) in equation (3.20), obtained equation will be:

$$C'V = CV\, (1 + \frac{R_2}{R_1})$$

$$or, \qquad C' = C\, (1 + \frac{R_2}{R_1}) \tag{3.21}$$

Equation of capacitance of synthesized capacitor of capacitance multiplier is:

$$C' = C\, (1 + \frac{R_2}{R_1}) \tag{3.22}$$

$$or, \qquad C' = CK \tag{3.23}$$

Where $K = (1 + \frac{R_2}{R_1})$ is known as gain or amplification factor of capacitance multiplier.

The terminal voltage of resistors R_1 and R_2 is same so, those two resistors will act like two parallel connected resistors connected in series with synthesized capacitor of capacitance multiplier. Then equation of resistance of synthesized resistor connected in series with synthesized capacitor of capacitance multiplier will be:

$$R = (\frac{R_1 R_2}{R_1 + R_2}) \quad (3.24)$$

Equivalent circuit of present capacitance multiplier circuit:

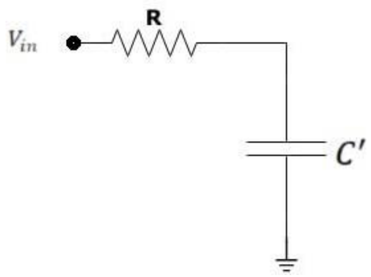


Fig 3.6: equivalent circuit of capacitance multiplier circuit

The capacitive sensor is represented by the capacitor (C). The resistors R_1 and R_2 sets the gain or amplification factor of the circuit. The output voltage is proportional to the capacitance change detected by the sensor, amplified by the Op-amp.

The synthesized capacitor also brings a series resistance equal to R and a leakage current appears across the capacitance because of the input offsets of Op-amp. Also, only one terminal of the synthesized capacitor can be accessed. Another terminal will be always connected with ground. There are another capacitance multiplier circuit with Op-amp which does not have those disadvantages, but its gain can't be increase so much because of some other limitation this circuit is most suitable present operation.

The disadvantages of present capacitance multiplier circuit are insignificant for our present overall circuits. Because the synthesized capacitor is used in frequent charging and discharging so leakage current because of the input offsets of Op-amp does not create any problem. And in astable multivibrator circuit one terminal of capacitor is connected with ground so one terminal accessibility of capacitance multiplier doesn't create any problem. Series resistance of synthesized capacitor affect with pulse width of output signal of astable multivibrator but it is very less. Also, frequency change data of signal conditioning circuit will be used for further processing. So, series resistance of synthesized capacitor doesn't create much problem. That's why, this capacitance multiplier circuits will be most suitable for our present operation.

Op-amp selection for the Capacitance Multiplier Circuit:

There are many Op-amp with different specification in the market like OP07, LM741, NE5532 etc. a specific Op-amp is chosen for our circuit requirement and market availability.

At first, OP07 Op-amp is used for capacitance multiplier circuit. But after some days of experimenting, it was understood that output frequency of astable multivibrator isn't changing at high frequency even after change in capacitance of capacitive sensor. After some other experimenting, it was understood that the problem is with bandwidth of Op-amp. Synthesized capacitor of capacitance multiplier is connected with astable multivibrator so, capacitor gets charged and discharged frequently. Now the bandwidth of Op-amp can't keep up with the frequency of astable multivibrator. So, the frequency of astable multivibrator doesn't increase even after change in capacitance or other parameters and it became constant at a specific frequency. This phenomenon happened because, each Op-amp has specific slew rate. And the rise time and fall time of signal depend on slew rate of Op-amp. So, an Op-amp with high bandwidth or slew rate should be used. Within higher bandwidth of Op-amps NE5532 was available in the market. So, NE5532 Op-amp is used for capacitance multiplier and get desired output signal at astable multivibrator.

When capacitance multiplier and astable multivibrator are used together then, there will be a specific maximum frequency with specific Op-amp because of its specification. With our present set up maximum frequency of output signal of the astable multivibrator using Op-amp OP07 for capacitance multiplier is 22kHz, using Op-amp LM741 for capacitance multiplier is 25kHz, using Op-amp NE5532 for capacitance multiplier is 110kHz. And the maximum frequency of output signal of the astable multivibrator without using capacitance multiplier is 400kHz.

Proposed Capacitance Multiplier Circuit:

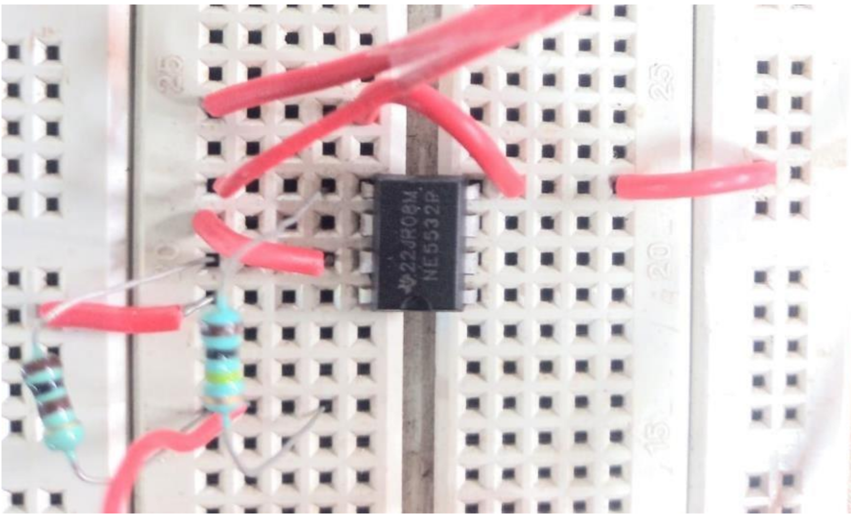


Fig 3.7: Proposed capacitance multiplier circuit

Proposed capacitance multiplier circuit is shown in Fig 3.7. After experimentation with many different combinations of resistors R_1 and R_2. And lastly, the suitable combination of resistors is chosen R_1 and R_2 for our overall present circuit. In the present capacitance multiplier circuit value of R_1 is 100Ω and value of R_2 is 100kΩ.

So, gain of capacitance multiplier will be (from equation 3.22):

$$K = (1 + \frac{R_2}{R_1}) = (1 + \frac{100\text{k}}{100}) = 1001$$

And value of series resistor connected with synthesize capacitor will be (from equation 3.24):

$$R = \left(\frac{R_1 R_2}{R_1 + R_2}\right) = \left(\frac{100k \times 100}{100k + 100}\right) \Omega = 99.9\Omega$$

The Op-amp used for capacitance multiplier circuit is NE5532, because of its high bandwidth.

Voltage Waveform of Synthesize Capacitor of the Capacitance Multiplier:

Voltage waveform of synthesize capacitor of the capacitance multiplier is shown in Fig 3.8. Which is slightly different than actual capacitor (voltage waveform using actual capacitor is shown in Fig 1.15). But it's doesn't create any problem with another operation of overall circuit.

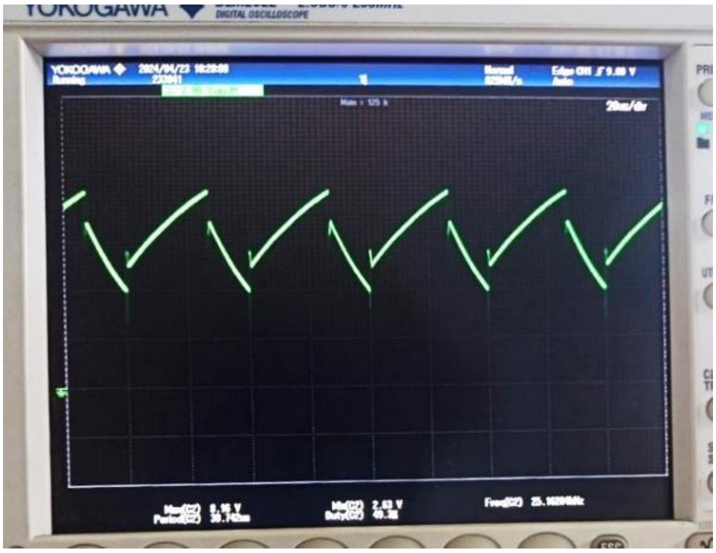


Fig 3.8: Charging discharging waveform of synthesize capacitor from capacitance multiplier

3.8 555-Timer based Astable Multivibrator Circuit using Capacitance Multiplier

In the present circuit capacitive sensor is connected with astable multivibrator circuit through capacitance multiplier. And output frequency of the astable multivibrator circuit change with change of value of capacitance of capacitive sensor. But synthesize capacitor formed by capacitance multiplier came with a series resistor (R). So, operation of astable multivibrator will be slightly different because of series resistance (R) formed by capacitance multiplier. The operation of the astable multivibrator circuit using NE555 IC with capacitance multiplier is stated below:

Initially, synthesize capacitor (C') is discharged, and the output at pin 3 is low. When power is applied, synthesize capacitor (C') starts charging through resistors R_1, R_2 and R. Once the voltage across terminal of capacitance multiplier exceeds 2/3 of Vcc (threshold voltage), the internal comparator resets the flip-flop, causing the output at pin 3 to go low. Then synthesize capacitor (C') discharges through resistor R_2 and R to pin 7 (discharge). Once the voltage across capacitance multiplier falls below 1/3 of Vcc (trigger voltage), the flip-flop is set again, and the output at pin 3 goes high. This cycle repeats, resulting in a continuous square wave output at pin 3 with a frequency determined by the values of resistors (R_1, R_2 and R) and synthesize capacitor C').

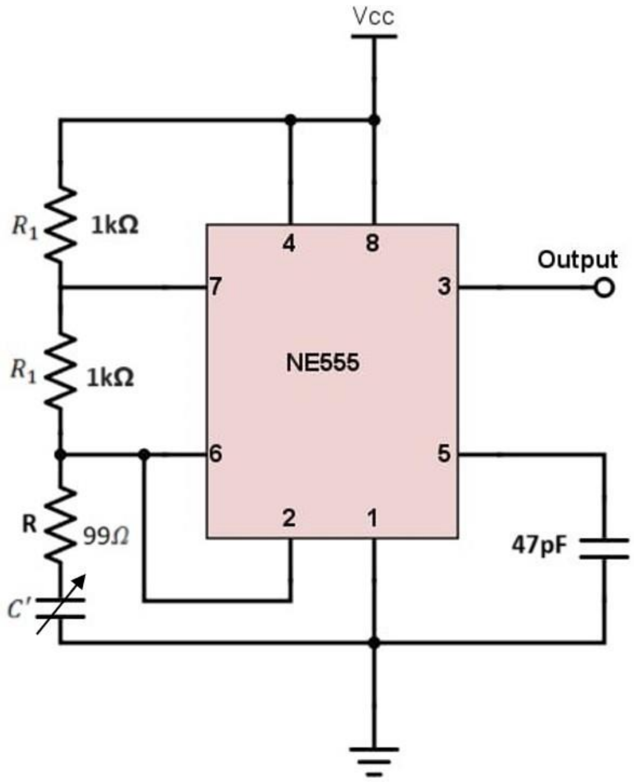


Fig 3.9 Proposed astable multivibrator circuit using NE555 timer IC

Frequency Calculation:

On time of the output waveform is:

$$t_1 = 0.693(R_1 + R_2 + R)C' \tag{3.25}$$

Off time of the output waveform is:

$$t_2 = 0.693(R_2 + R)C' \tag{3.26}$$

Time period of the output waveform is:

$$T = t_1 + t_2 = 0.693(R_1 + 2R_2 + 2R)C' \tag{3.27}$$

Now the frequency of the output waveform can be calculated using the formula:

$$f = \frac{1}{T} = \frac{1.44}{(R_1 + 2R_2 + 2R)C'} \tag{3.28}$$

After putting value of C' from equation (3.23) in equation (3.28), obtained equation will be:

$$f = \frac{1.44}{(R_1 + 2R_2 + 2R)KC} \tag{3.29}$$

After putting value of C from equation (3.12) in equation (3.29), obtained equation will be:

$$f = \frac{1.44(\varepsilon_r(h - W_l) + D)}{\varepsilon_0 \varepsilon_r AK(R_1 + 2R_2 + 2R)} \tag{3.30}$$

Where f is frequency of the output signal
W_l is liquid level in the glass jar
K is gain of capacitance multiplier
h is height of glass jar
ε_0 is permittivity of vacuum
ε_r is relative permittivity of Teflon
A is cross section of capacitor
D is thickness of Teflon layer

The relation between frequency of the output signal of 555-timer based astable multivibrator and liquid level in the glass jar is shown in the equation (3.30).

Adjusting the values of resistors R1 and R2 and capacitor C allows controlling the frequency and duty cycle of the output waveform.

The output waveform is a square wave with a duty cycle (D) determined by the ratio of on time and time period:

$$D = \frac{t_1}{T} = \frac{R_1 + R_2 + R}{R_1 + 2R_2 + 2R} \tag{3.31}$$

Operation and Results of 555-Timer based Astable Multivibrator after using Capacitance Multiplier:

At first, a suitable gain for the capacitance multiplier was set up, then experimenting was started with different combination of resistors R_1 and R_2, as the frequency of astable multivibrator is inversely proportional with $(R_1 + 2R_2)$. Earlier without using capacitance multiplier, output signal of astable multivibrator was noisy as shown in Fig 3.4. But, after using capacitance multiplier, output signal of astable multivibrator become noiseless squire wave as shown in Fig 3.10. Also, sensitivity of the system increase. Because, frequency change of signal conditioning circuit was 4.5kHz at full scale of liquid level change.

Suitable Output Frequency Selection:

Synthesize capacitor formed by capacitance multiplier is connected with pin no 6 of NE555 timer IC. To reduce noise and increase sensitivity, gain of the capacitance multiplier should be increased. As gain of the capacitance multiplier increased value of synthesize capacitor also increase but frequency of astable multivibrator decrease. Because frequency of astable multivibrator is inversely proportional with the value of synthesize capacitor (C') from capacitance multiplier. So, after choosing suitable gain for capacitance multiplier, where noise is minimum and sensitivity is maximum. Then suitable combination of connected resistors (R_1 and R_2) was chosen to get desired frequency range from astable multivibrator. The values of those selected resistors are: R_1=1kΩ and R_2=1kΩ.

Voltage Waveform of 555-Time Astable Multivibrator after using Capacitance Multiplier:

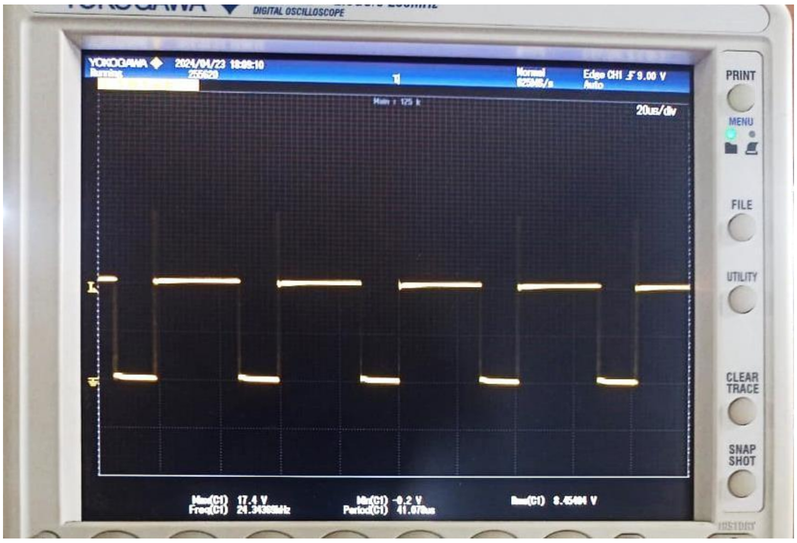


Fig 3.10: Output signal of astable multivibrator using capacitance multiplier

Our capacitive sensor is catching noise from the environment because of its large size. Because of that frequency of output signal of astable multivibrator was changing rapidly (as shown in Fig 3.4). But after usage of capacitance multiplier effect of noise is reduced (as shown in Fig 3.10). Now this signal will be received at the microcontroller, which generally supports 0-3V or 0-5V at its input terminal. But the output signal of the signal conditioning circuit is 12V pulse signal with 17.4V peak voltage (as shown in Fig 3.10). So, a voltage regulator circuit is used to convert 12V pulse signal into suitable pulse signal for the microcontroller.

3.9 Suitable Microcontroller and its Input Signal range

An Arduino microcontroller is used for present operation, because Arduino microcontroller is a cost effective, user friendly and widely available microcontroller. The Arduino Integrated Development Environment (IDE) is easy to write and upload code to the board. Arduino offers various boards for different needs, from simple Arduino Uno to more powerful board like Arduino Mega. Within Arduino boards Arduino Mega board is used because, it offers more input output pins and it has larger memory. Also, it is compatible with most Arduino shields designed for other Arduino boards.

Now, the output signal from the signal conditioning circuit (555-timer based astable multivibrator circuit) of the capacitor is taken as input signal in the microcontroller. The output signal of 555-timer based astable multivibrator circuit is 12V pulse signal with 17.4V peak voltage (ringing effect at 555-timer cause voltage peak) as shown in Fig 3.10. But input signal range of the Arduino Mega microcontroller is 0-5V. So, a voltage regulator is used which will convert the output signal of 555-timer based astable multivibrator into a suitable signal for Arduino Mega microcontroller.

3.10 Establishing Compatibility of Voltage between Microcontroller and Signal Conditioning Circuit

For the measurement purpose, the output voltage signal of the astable multivibrator is send into the Arduino mega 2560 microcontroller as digital input. But output voltage signal of astable multivibrator isn't suitable for Arduino mega 2560 microcontroller as digital input. Because, output voltage signal of astable multivibrator is 12-volt square wave but, it also has maximum voltage peak of 17.4 volt and minimum voltage peak of -0.2 volt because of ringing effect of 555-timer as shown in Fig 3.10. But a 5-volt digital signal is needed for Arduino mega 2560 microcontroller. So, a voltage regulator circuit is used between signal conditioning circuit and microcontroller.

Causes of Ringing in Astable Multivibrator:

Fig 3.11: Astable multivibrator output waveform using long connecting wire

There are certain causes of ringing in astable multivibrator circuit using NE555 timer IC like present of parasitic inductance and capacitance. There are unintended inductances and capacitances that are present in the circuit due to the physical layout of the components and connecting wire. Long connections can add inductance and capacitance, exacerbating ringing problems. They can cause the circuit to oscillate briefly at high frequencies during transitions.

Another cause of ringing in NE555 IC is high speed switching. The 555-timer switches relatively quickly between high and low states. If the load connected to the 555-timer output has significant inductance or capacitance, it can interact with the switching, leading to ringing.

So, respectively shorter wire was used which reduce ringing but it couldn't erase effect of ringing completely. Because in our present circuit, the parasitic inductance and capacitance of the circuit can't be eliminated completely. At first voltage peaks were 19.5 volts and -2.8 volts (as shown in Fig 3.11) but after using shorter connecting wire its voltage peaks become 17.4 volts and -0.2 volts (as shown in Fig 3.10).

Schmitt Trigger as a Voltage Regulator for Present Circuit:

At first, a Schmitt trigger was used for voltage regulation, which is a type of electronic circuit that converts an analog input signal to a digital output signal. It is characterized by hysteresis, which means it has two different threshold voltage levels for transitioning from high to low (and vice versa). It has some advantage like, its hysteresis helps to eliminate noise and provide a clean, stable digital output even when the input signal is noisy or slow-changing. Also, it can give more power to the load than using voltage regulator made of Zener diode. So, a Schmitt trigger was build using Op-amp, but it wasn't suitable for our frequency requirement. If was giving a triangular waveform from its output at high frequency because of low slew rate of Op-amp. The different Op-amp like OP07, LM741, NE5532 was used, but even with using high slew rate Op-amp NE5532 Op-amp the frequency range was up to 10kHz in a Schmitt trigger using Op-amp. After realising voltage regulator using Schmitt trigger isn't compatible with our present circuit so, started experimenting with the voltage regulator using Zener diode.

3.11 Voltage Regulator using Zener Diode

After failed attempt at making the voltage regulator using Schmitt trigger for our present operation, experimenting with the voltage regulator using Zener diode was started. A Zener diode is a special type of semiconductor diode designed to allow current to flow in the reverse direction when a specific, predetermined voltage (known as the Zener breakdown voltage) is reached. This property makes Zener diodes very useful for voltage regulation in electronic circuits. A Zener diode is heavily doped to reduce the breakdown voltage. When the reverse voltage applied to the diode reaches this Zener breakdown voltage, it starts conducting current in the reverse direction without damaging the diode and it maintains a constant voltage across its terminals. This is because the Zener breakdown mechanism allows it to conduct reverse current to maintain the voltage drop. To use a Zener diode as a voltage regulator, it is typically connected in parallel with the load. A resistor is placed in series with the Zener diode to limit the current flowing through the diode and the load. Zener diode is placed in parallel with the load to maintain a constant output voltage. The circuit diagram of voltage regulator using Zener diode is shown in Fig 3.12.

Circuit Diagram of the Voltage Regulator using Zener Diode:

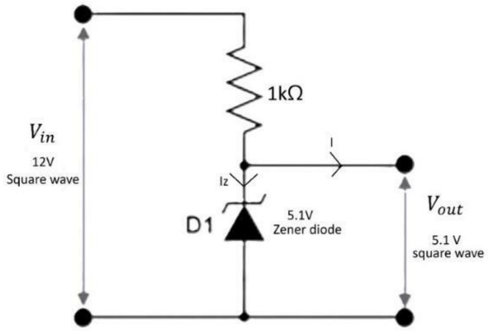


Fig 3.12: Proposed voltage regulator using Zener diode

Load Current of Voltage Regulator:

The output of voltage regulator will be connected with input pin of Arduino Mega 2560 microcontroller. When an Arduino Mega 2560 digital pin is configured as an input and used for digital read operations, the current it draws is extremely minimal. When a digital pin on the Arduino Mega 2560 is set to input mode, it enters a high-impedance state. This means it draws very little current, effectively isolating the pin from the circuit to accurately read the input voltage without influencing it. If no external resistor is used, the pin can be left floating. However, for stable readings, internal pull-up resistors can be enabled. These resistors typically have a value of around 20kΩ to 50kΩ. Arduino Mega 2560 digital pin will get 5.1V as an input from voltage regulator. So, using Ohm's law load current can be calculated to Arduino Mega 2560 digital pin, which is around 0.25mA.

Selection of Series Resistor of Voltage Regulator:

At first, a Zener diode with a breakdown voltage equal to the desired output voltage is selected, which is 5.1V in this case. Then the series resistor was determined. The value of the series resistor is critical. It is calculated based on the input voltage (V_{in}), Zener voltage (V_z), Zener current (I_z), and the maximum load current (I). The maximum series resistor value (R_z) can be calculated using the formula:

$$R_{zmx} = \frac{V_{in} - V_z}{I + I_z} \tag{3.32}$$

Where (I_z) is the minimum Zener current required to keep the Zener diode in breakdown. For 1N4733 Zener diode Zener current (I_z) is 5mA, Zener voltage (V_z) is 5.1V, input voltage (V_{in}) is 12V. The output of voltage regulator is connected to the Arduino mega 2560 microcontroller's digital pin. Which will draw very little current let's consider maximum 1mA. So, maximum load current (I) is 1mA.

Then, maximum series resistor value (R_{zmx}) will be:

$$R_{zmx} = \frac{(12 - 5.1) \times 1000}{(5 + 1)} = 1150\Omega$$

Minimum series resistor value (R_z) can be calculated using the formula:

$$R_{zmn} = \frac{V_{in} - V_z}{I_{zmx}} \tag{3.33}$$

Where I_{zmx} is maximum Zener current can be flow through Zener diode without damaging the diode. Which is equal to:

$$I_{zmx} = \frac{Zener\ diode\ power\ ratting}{Zener\ voltage} \tag{3.34}$$

$$I_{zmx} = \frac{1}{5.1} = 196mA$$

Then, minimum series resistor value (R_{zmn}) will be:

$$R_{zmn} = \frac{(12 - 5.1) \times 1000}{196} = 35.2\Omega$$

So, theoretically series resistance value can be chosen from 35.2Ω to 1150Ω. But to reduce power loss in voltage regulator its best to choose the value of series resistor close to maximum series resistor value. That's why the series resistor value of the voltage regulator in our circuit is $1k\Omega$.

Advantage of Voltage Regulator using Zener Diode:

Advantage of voltage regulator using Zener diode is that, it is a simple and inexpensive circuit which provides stable voltage for low current applications. But it is not efficient for high current loads due to power dissipation in the series resistor. Also, regulation can be affected by changes in input voltage and load current. In this case output of voltage regulator is connected with the Arduino mega 2560 microcontroller which required very little current. So, voltage regulator using Zener diode is best choice for our circuit requirement.

Output waveform of voltage regulator:

Input voltage signal of voltage regulator coming from astable multivibrator. It is 12-volt square wave with ringing effect which is shown in Fig 3.13 as yellow colour waveform. Output of voltage regulator is a 5-volt square waveform which is shown in Fig 3.13 as green colour waveform.

Fig 3.13: Input and output waveform of voltage regulator

Experimental Circuit of Capacitance Multiplier, Astable Multivibrator and Voltage Regulator:

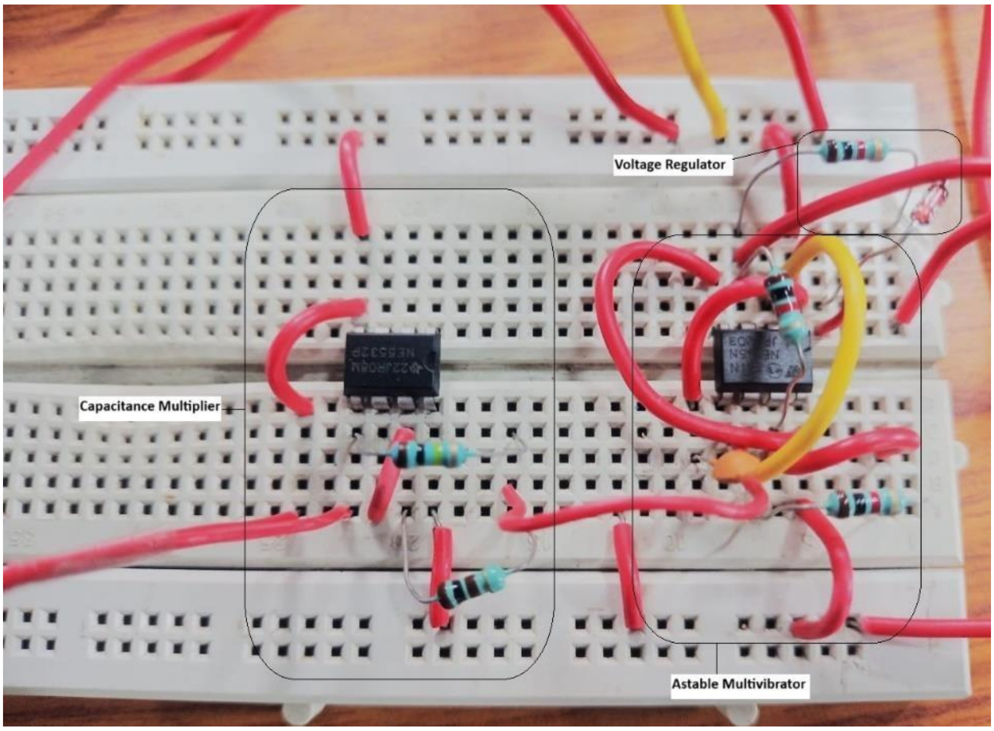
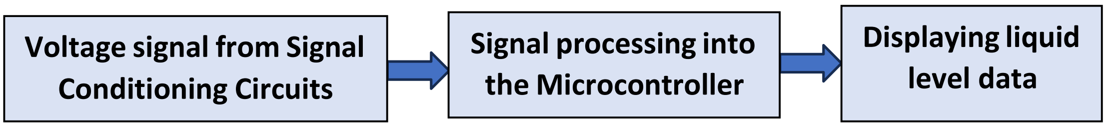


Fig 3.14: Experimental circuit of Capacitance Multiplier, Astable Multivibrator and Voltage Regulator

3.12 Arduino Mega 2560 Microcontroller in the Present Work

The output signal of the signal conditioning circuit converted into a suitable signal for the selected microcontroller (Arduino Mega 2560 microcontroller) using the voltage regulator. Then the output signal of the voltage regulator connected with specific digital pin of the Arduino mega 2560 microcontroller. The input signal of Arduino mega 2560 microcontroller has only high state of 5V and low state of 0V, so it can be used for digital purposes.

Presently the Arduino mega 2560 microcontroller has three objectives, those are measuring frequency of the input signal, analyse input signal's frequency data to get the information about level of the conductive liquid and send the liquid level information to the display.



Arduino Mega 2560 Microcontroller Board:

Fig 3.15: Arduino mega 2560 microcontroller board

3.12.1 <u>Operations of Arduino mega 2560 microcontroller</u>

Measuring Frequency of Input Signal:

There are two ways of frequency measurement using Arduino mega 2560 microcontroller. These are pulse width measuring method of input voltage signal and interrupt method of frequency measurement.

The 'pulseIn' command in Arduino is used to measure the duration of a pulse on a specific pin. It measures how long a pulse (either high or low) lasts. Syntax for measuring the duration of a pulse is pulseIn(pin, value). Where ‘pin’ is the number of the pin on which the pulse is read. This pin should be configured as an input. And ‘value’ is the type of pulse to measure (either HIGH or LOW). If high time and low time of input signal is measured. Then time period of the signal will be addition of high time and low time. Inverse of time period is frequency so, using those steps the frequency of the input signal can be measured. But there are some limitations using this method. Using a standard function generator the programming of ‘pulseIn’ command is tested and find out that measuring frequency range using this method is up to 20kHz. Which isn’t suitable for input signals frequency range. So, interrupt method of frequency measurement is most suitable for present operation.

Arduino has external interrupts, pin change interrupts, software interrupts and timer interrupts, however for the external interrupts is used for the present operation (frequency measurement). External interrupts are triggered by external events, such as a signal change on a specific pin. The Arduino Mega 2560 has external interrupts on pins 2, 3, 18, 19, 20, and 21. External interrupts has different triggering modes like LOW (Triggered when the pin is low), CHANGE (Triggered when the pin changes value from high to low or low to high), RISING (Triggered when the pin goes from low to high) and FALLING (Triggered when the pin goes from high to low). 21 no digital pin is used as external interrupts controls input pin and the triggering mode is RISING. This method of frequency measurement is very accurate. The programming is tested using standard function generator and find out that measuring frequency range using this method is up to 120kHz. Which is very suitable for our input signal’s frequency range.

In the microcontroller programming using ‘attachInterrupt’ command, the interrupt is attached with 21 no digital pin and then introduced 1 second of delay using ‘delay’ command. After that the number of pulses between that 1 second of delay (by counting number of RISING in the signal) is counted. After counting using ‘detachInterrupt’ command, interrupt is detached from 21 no digital pin. Now the number of pulses counted is the frequency of the signal.

Input Signal’s Frequency Data Analysis:

After getting value of frequency of the input signal, using the Exponential or Polynomial equation the liquid level data is obtained. The Exponential or Polynomial equation is obtained from Curve Fitting tools of MATLAB.

At first, the input signal’s frequency value is used into the equation from Curve Fitting but after data analysis calculated liquid level value isn’t sensitive enough. So, exact liquid level information isn’t obtained, because the respective frequency change of input voltage signal is very less. Then, after some experimenting, 23000 is subtracted from input signal’s frequency value (as our frequency range of input signal is 24kHz to 29kHz). Then the new scaled data is put into the equation from Curve Fitting, and finally obtained exact liquid level information from the Arduino Mega 2560.

Displaying the Final Result into the Display:

At first, the liquid level data was obtained into the computer on the serial monitor of Arduino IDE. To do that, serial monitor is initialize using ‘SerialBegin’ command. Then the liquid level data is printed into the serial monitor of Arduino IDE using ‘Serial.print’ print command. Using this method, liquid level data is obtained. However, if a computer is used each time to get liquid level data, then it would be very inconvenience. So, a display is used to display the final result or liquid level information.

To display the results, a suitable display is chosen for our present work. Two-line information about frequency of the signal conditioning circuit and level of the conductive liquid (saline water) will be displayed. So, 16×2 LCD display (more specifically 16×2 LCD keypad shield) is chosen as our display. Because, it has 16 characters × 2 lines, which enough for our requirement. Also 16×2 LCD keypad shield is compatible with Arduino Mega 2560 microcontroller.

So, a 16×2 LCD display is used with LCD keypad shield and get the liquid level data. To do that, 'LiquidCrystal' library is included in microcontroller programming. Then LCD display is initialized using 'lcd.begin' command. After that, the liquid level data is printed into the LCD display of connected with Arduino mega 2560 microcontroller using 'lcd.print' print command.

3.13 16×2 LCD Display

The final result (liquid level data) coming from Arduino Mega 2560 microcontroller is displayed in 16×2 LCD display. At first, 16×2 LCD display is connected with Arduino Mega 2560 microcontroller. A 16×2 LCD Keypad Shield is used so, connection with Arduino Mega 2560 microcontroller was pretty easy. But contrast of LCD display was controlled using a screw driver by rotating screw of LCD Contrast Potentiometer.

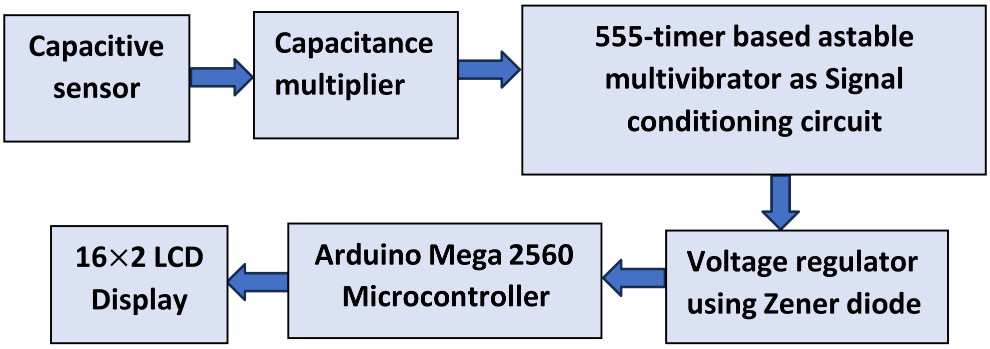
Output in 16×2 LCD Display:

Fig 3.16: Output in 16×2 LCD display

The conductive liquid level data and frequency data of input signal of Arduino Mega 2560 microcontroller is displayed in the LCD display as shown in Fig 3.16. In the LCD display on the first line its show "Water Lv 174 mm" and on the second line its shown "Freq 24690 Hz". Its means, when frequency of input signal of Arduino Mega 2560 microcontroller is 24690 Hz then the conductive liquid (saline water) level is 174 mm. Presently, saline water is used as the measured liquid that's why "Water Lv" displaying on the first line of LCD display.

3.14 Proposed Capacitive Sensor Circuit

3.14.1 Block Diagram of Capacitive Sensor Circuit



3.14.2 Schematic Diagram

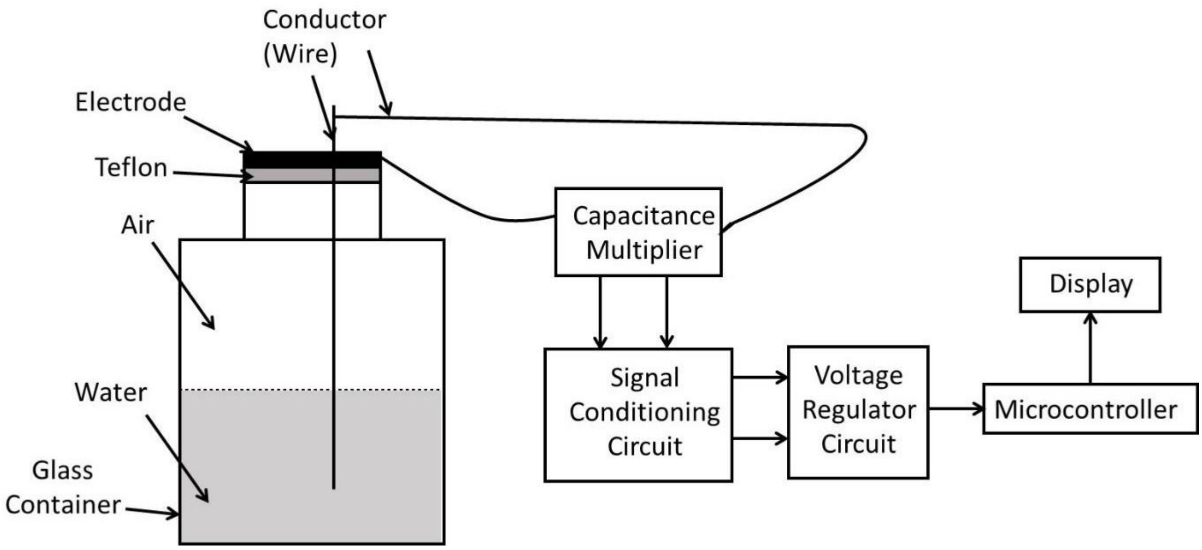


Fig 3.17: Proposed capacitive sensor circuit: circuit arrangement

3.14.3 Circuit Arrangement

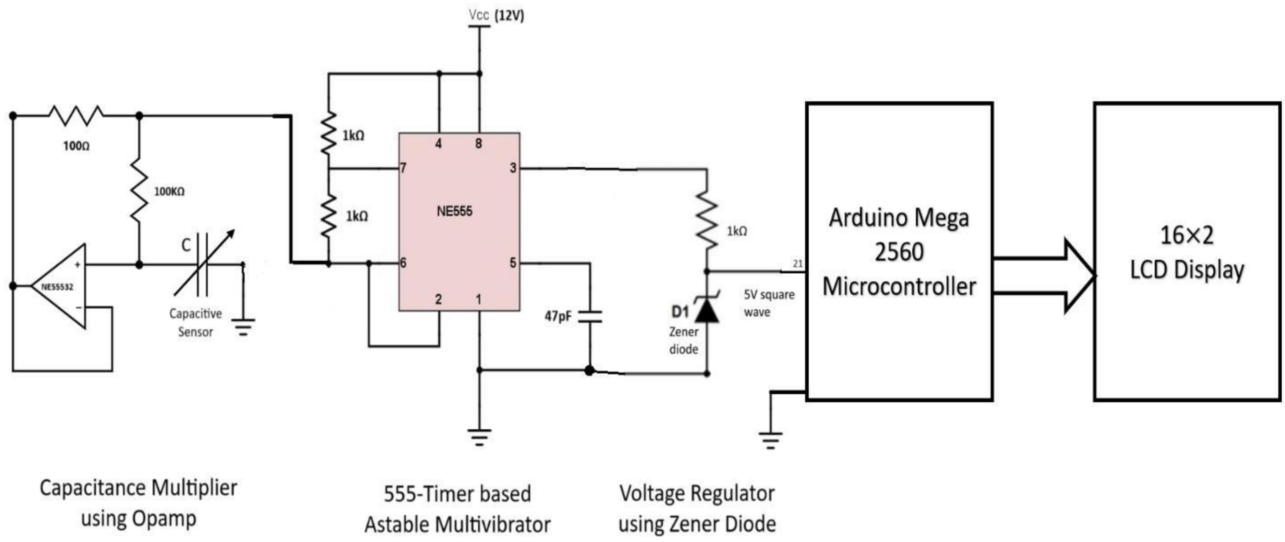


Fig 3.18: Proposed capacitive sensor circuit: circuit arrangement

3.14.4 Experimental Setup

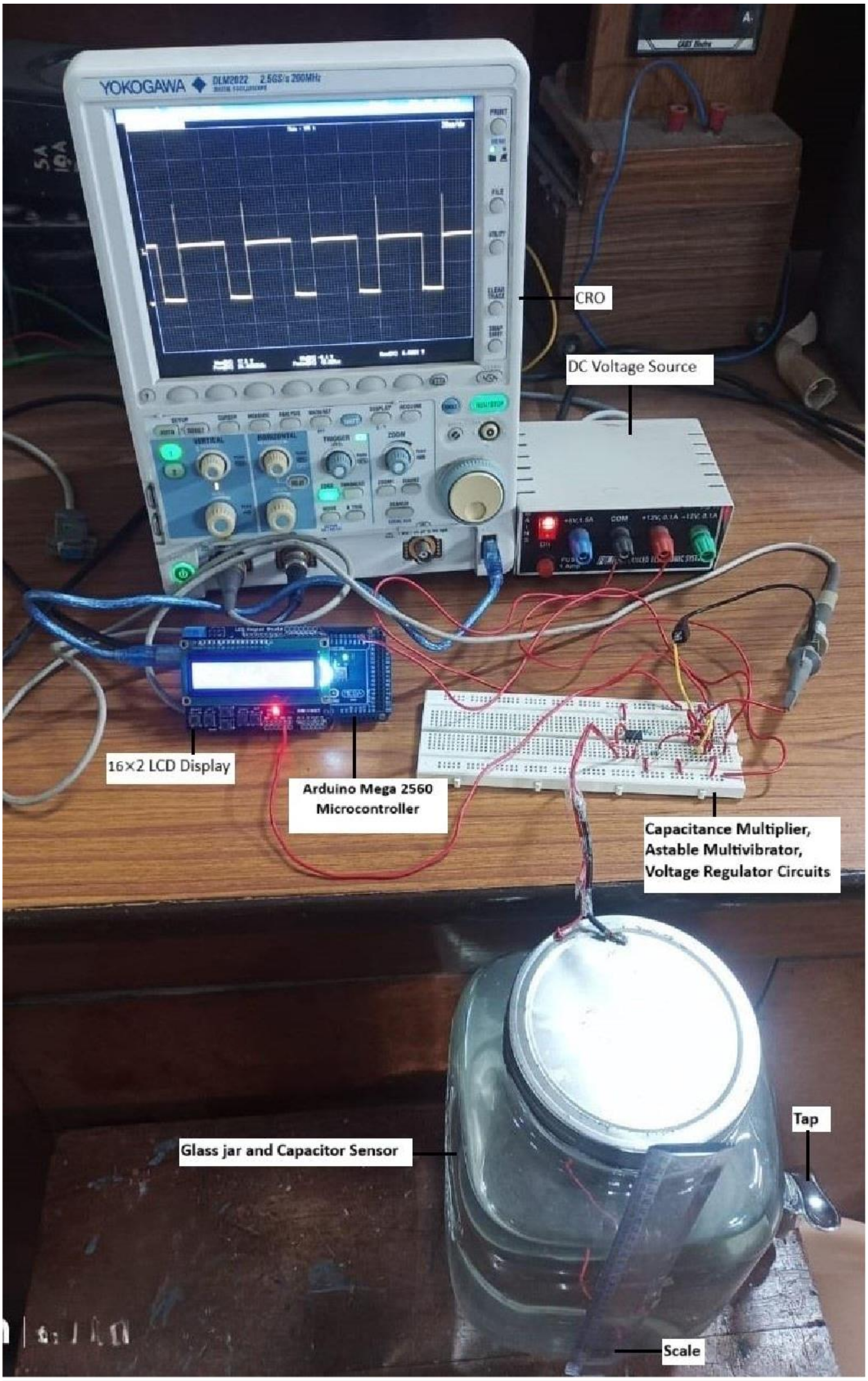


Fig 3.19: Proposed capacitive sensor circuit: Experimental Setup

3.15 Results and Discussions

In the proposed scheme the capacitive sensor is changing its capacitance with change in the saline water level of the glass jar. Then capacitive sensor is connected with the capacitance multiplier circuit, whose amplification factor or gain is kept constant that is 1001. Then capacitor multiplier is connected with astable multivibrator using 555-timer IC. Its value of resistors is kept constant. The frequency of astable multivibrator is depended on connected resistors and capacitors as shown in equation (3.20). Connected resistors value is constant so, frequency change is depended on change of value of capacitance of the capacitive sensor or change of saline water level.

Frequency of astable multivibrator is measured using microcontroller. So, a data table is made with two data set saline water level and frequency. Then an equation is obtained from those two data using Curve Fitting tools of MATLAB. Then that equation is implemented into the microcontroller, and the saline water level is calculated from frequency of astable multivibrator, using the equation obtained from Curve Fitting method. At first, sensitivity was very low because respective frequency change (24.38kHz to 28.12kHz) is very low. So, the data were scaled as

Frequency change (Hz/10) = (Actual frequency in Hz – 23000 Hz) /10.

After using scaled data into MATLAB Curve Fitting tools, a better sensitivity was obtained.

Table 3.1: Data for MATLAB Curve Fitting:

S/L No.	Frequency (Hz)	Frequency change (Frequency - 23000)/10	Actual saline water level (mm)
1	24380	138	180
2	24680	168	175
3	24960	196	170
4	25160	216	165
5	25300	230	160
6	25470	247	155
7	25620	262	150
8	25780	278	145
9	25920	292	140
10	26110	311	135
11	26250	325	130
12	26400	340	125
13	26510	351	120
14	26640	364	115
15	26800	380	110
16	26900	390	105
17	27050	405	100

S/L No.	Frequency (Hz)	Frequency change (Frequency - 23000)/10	Actual saline water level (mm)
18	27230	423	95
19	27270	427	90
20	27400	440	80
21	27540	454	70
22	27600	460	60
23	27720	472	50
24	27880	488	40
25	28040	504	30
26	28120	512	20

Actual water level Vs Frequency graph:

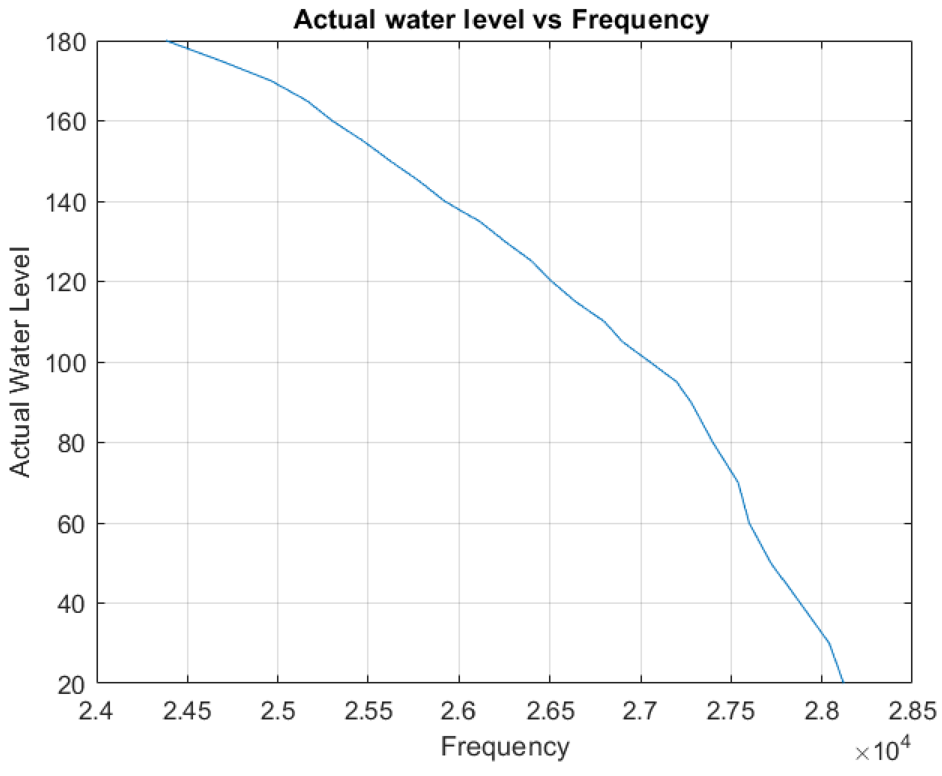


Fig 3.20: Actual water level Vs Frequency graph

3.15.1 Curve Fitting using Exponential

There are many types of equation for Curve Fitting but for our data two types equation are suitable, those are Exponential equation and Polynomial equation. At first the Exponential equation is used for Curve Fitting. The Curve Fitting is done using two data set (Frequency change and Actual saline water level) from table 3.1.

Result from Curve Fitting Tools:

General model Exp2: $f(x) = ae^{bx} + ce^{dx}$

Coefficients (with 95% confidence bounds): a = -1.025, b = 0.008995, c = 217, d = -0.001132

Goodness of fit: SSE 147.5, R-square 0.9973, Adjusted R-square 0.9969, RMSE 2.589

So, the equation obtained from Curve Fitting:

$$f(x) = -1.025e^{0.00895x} + 271e^{-0.001132\mathrm{x}} \qquad (3.35)$$

Where $f(x) =$ Measured saline water level

$x =$ Frequency change

Water Level vs Frequency Change Graph of Curve Fitting using Exponential:

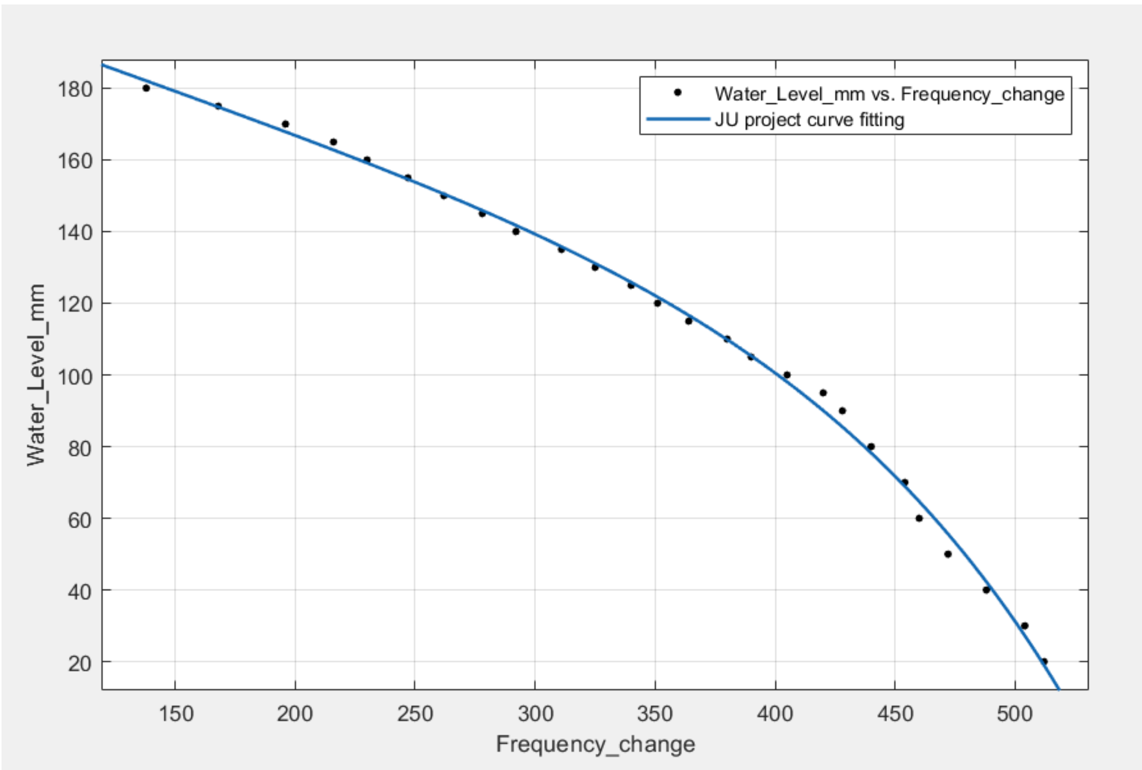


Fig 3.21: Water Level vs Frequency change graph of Curve Fitting

Table 3.2: Performance of Curve Fitting using Exponential

S/L No.	Frequency (kHz)	Actual saline water level (mm)	Measured saline water level (mm)	Error (mm)	% Error (mm)
1	24380	180	182	2	1.111111
2	24680	175	174	-1	-0.57143
3	24960	170	167	-3	-1.76471
4	25160	165	162	-3	-1.81818
5	25300	160	159	-1	-0.625
6	25470	155	154	-1	-0.64516
7	25620	150	150	0	0
8	25780	145	145	0	0
9	25920	140	141	1	0.714286
10	26110	135	135	0	0
11	26250	130	131	1	0.769231
12	26400	125	125	0	0
13	26510	120	121	1	0.833333
14	26640	115	116	1	0.869565
15	26800	110	109	-1	-0.90909
16	26900	105	105	0	0
17	27050	100	98	-2	-2
18	27230	95	88	-7	-7.36842
19	27270	90	86	-4	-4.44444
20	27400	80	78	-2	-2.5
21	27540	70	68	-2	-2.85714
22	27600	60	64	4	6.666667
23	27720	50	55	5	10
24	27880	40	42	2	5
25	28040	30	27	-3	-10
26	28120	20	19	-1	-5
Mean-Squared-Error (MSE)				**6.230769**	**15.3489**
Root Mean-Squared-Error (RMSE)				**2.496151**	**3.917767**

Actual and Measured water level vs Frequency graph:

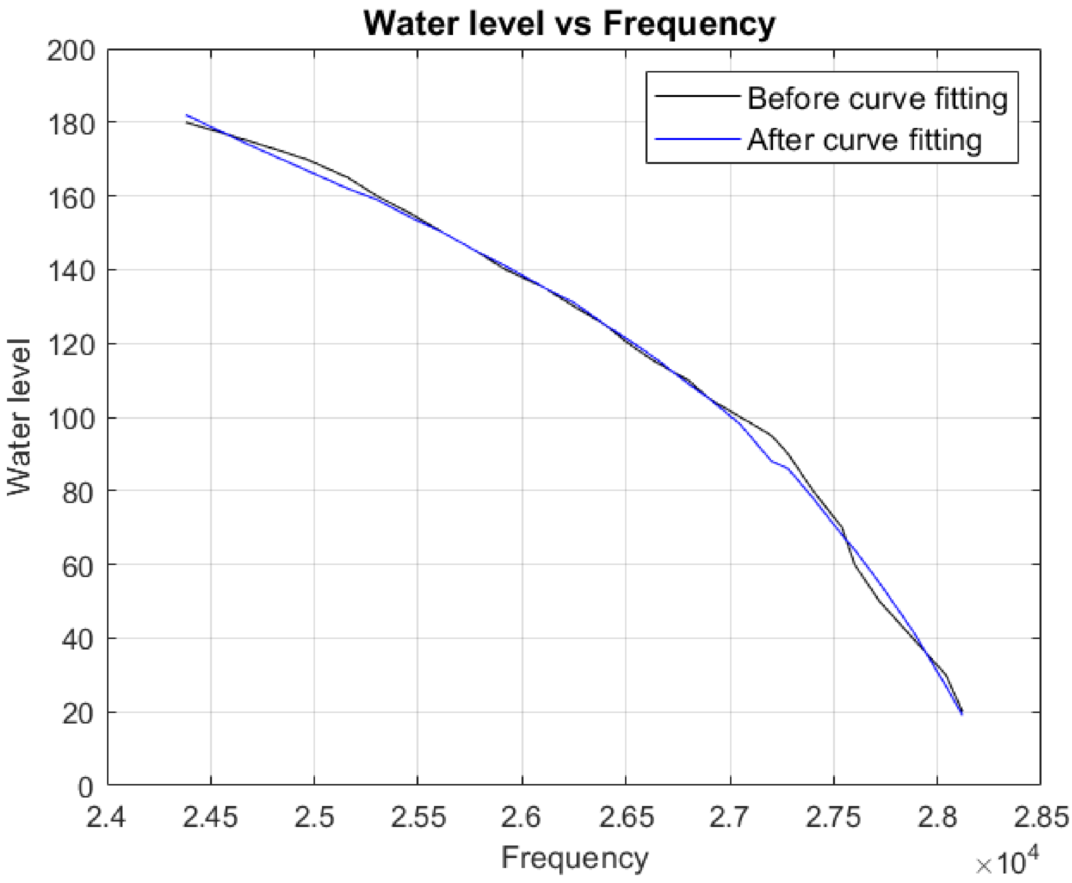


Fig 3.22: Actual and Measured water level vs Frequency graph

In the above figure (Fig 3.22) the plot in black colour is Actual water level vs Frequency graph and the plot in blue colour is Measured water level vs Frequency graph, after curve fitting using Exponential equation.

Actual water level vs Error and Actual water level vs percentage of Error graph:

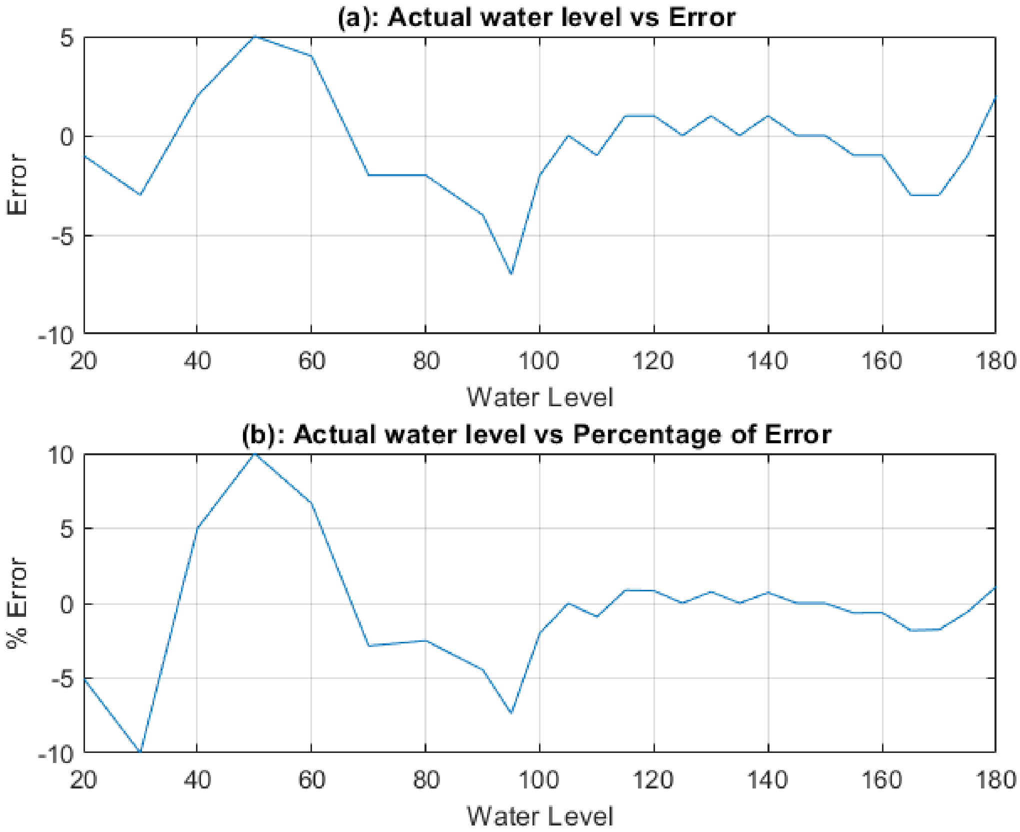


Fig 3.23 (a): Actual water level vs Error

(b): Actual water level vs Percentage of Error

3.15.2 Curve Fitting using Polynomial

After Curve Fitting using Exponential equation, Polynomial equation is used for a comparison. The Curve Fitting is done using two data set (Frequency change and Actual saline water level) from table 3.1.

Result from Curve Fitting using Polynomial:

Linear model Poly3: $f(x) = p_1x^3 + p_2x^3 + p_3x + p_4$

Coefficients (with 95% confidence bounds): $p_1 = -2.766 \times 10^{-6}$, $p_2 = 0.001837$, $p_3 = -0.6556$, $p_4 = 245.8$

Goodness of fit: SSE 170.2, R-square 0.9968, Adjusted R-square 0.9964, RMSE 2.781

So, the equation obtained from Curve Fitting method:

$$f(x) = -0.000002766x^3 + 0.001837x^2 - 0.6556x + 245.8 \qquad (3.36)$$

Where $f(x)$ = Measured saline water level

x = Frequency change

Water Level vs Frequency Change Graph of Curve Fitting using Polynomial:

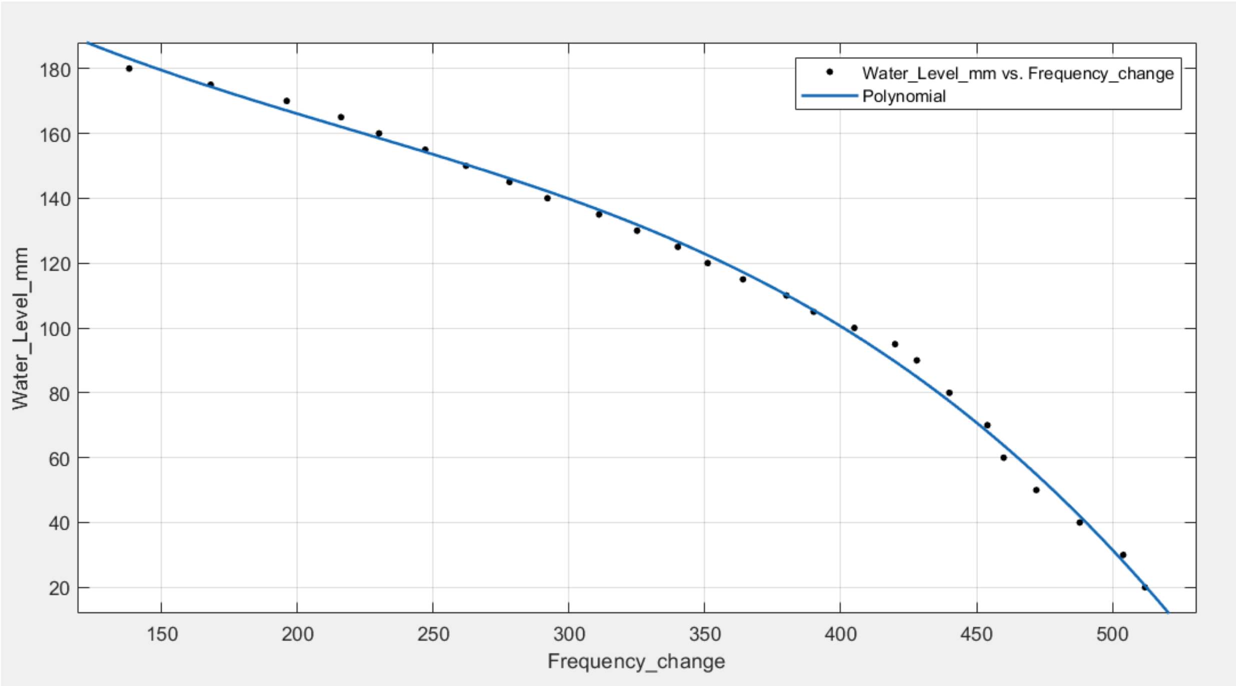


Fig 3.24: Water Level vs Frequency change graph of Curve Fitting

Table 3.3: Performance of Curve Fitting using Polynomial

S/L No.	Frequency (kHz)	Actual saline water level (mm)	Measured saline water level (mm)	Error (mm)	% Error (mm)
1	24380	180	183	3	1.666667
2	24680	175	174	-1	-0.57143
3	24960	170	167	-3	-1.76471
4	25160	165	162	-3	-1.81818
5	25300	160	158	-2	-1.25
6	25470	155	154	-1	-0.64516
7	25620	150	150	0	0
8	25780	145	146	1	0.689655
9	25920	140	142	2	1.428571
10	26110	135	136	1	0.740741
11	26250	130	131	1	0.769231
12	26400	125	126	1	0.8
13	26510	120	122	2	1.666667
14	26640	115	117	2	1.73913
15	26800	110	110	0	0
16	26900	105	105	0	0
17	27050	100	97	-3	-3
18	27230	95	87	-8	-8.42105
19	27270	90	85	-5	-5.55556
20	27400	80	77	-3	-3.75
21	27540	70	67	-3	-4.28571
22	27600	60	63	3	5
23	27720	50	54	4	8
24	27880	40	41	1	2.5
25	28040	30	27	-3	-10
26	28120	20	20	0	0
Mean-Squared-Error (MSE)				**7.692308**	**13.84848**
Root Mean-Squared-Error (RMSE)				**2.773501**	**3.721354**

Actual and Measured water level vs Frequency graph:

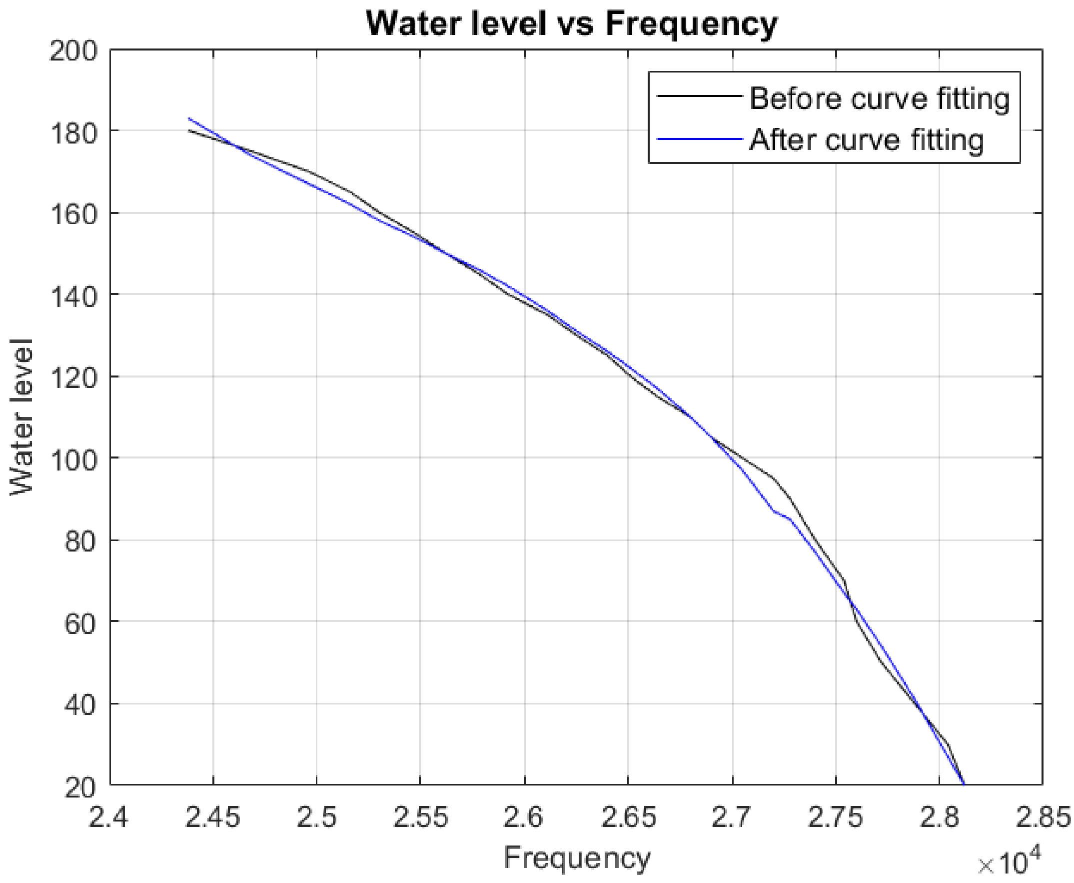


Fig 3.25: Actual and Measured water level vs Frequency graph

In the above figure (Fig 3.25) the plot in black colour is Actual water level vs Frequency graph and the plot in blue colour is Measured water level vs Frequency graph, after curve fitting using Polynomial equation.

Actual water level vs Error and Actual water level vs percentage of Error graph:

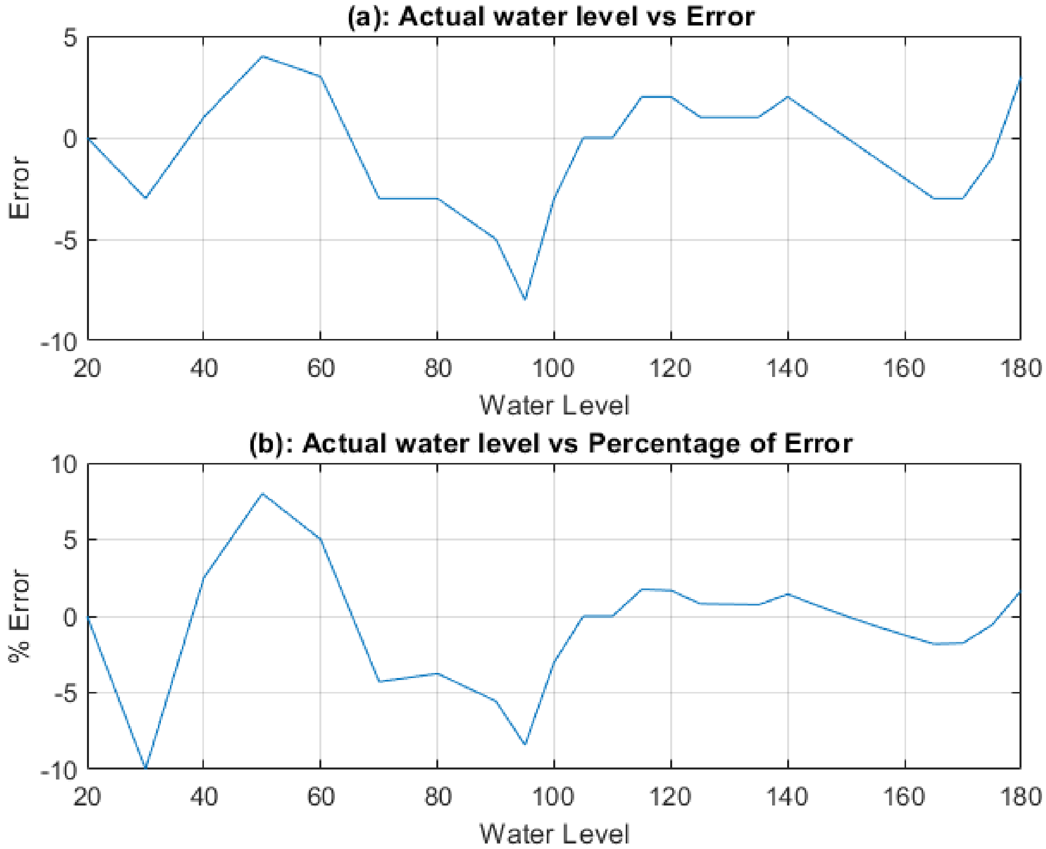


Fig 3.26 (a): Actual water level vs Error

(b) : Actual water level vs Percentage of Error

After a comparison between Curve Fitting using Exponential equation and Curve Fitting using Polynomial equation. So, finally Curve Fitting using Exponential equation is used for our present operation. Because Root Mean-Squared-Error (RMSE) of Curve Fitting using Exponential equation is less than Root Mean-Squared-Error (RMSE) of Curve Fitting using Polynomial equation.

Conclusions

Conclusions

The proposed scheme will be used to measure the level of the conductive liquid by a non-contact method. The arrangement is expected to be particularly useful if the liquid is also corrosive in nature. It includes a capacitive sensor whose capacitance changes with the conductive liquid level, a capacitance multiplier using an op amp which increases the value of connected capacitance, and an astable multivibrator using 555-timer. The frequency of the multivibrator output depends on the effective value of the connected capacitance. A voltage regulator using Zener diode converts the voltage signal of astable multivibrator into certain voltage level which is suitable for the Arduino mega 2560 microcontroller, that has been deployed to process voltage signals and yields the measured liquid level, and an LCD display presents the liquid level value.

Between two conductive surfaces of the capacitive sensor there are two layers of dielectric material, they are air and Teflon. Teflon is used because, it has high relative permittivity, which increase capacitance of the capacitive sensor. And its corrosive resistant properties protect metal cap from contact of corrosive liquid vapour (in case the conductive liquid is also corrosive).

The 555-timer based astable multivibrator is used as a signal conditioning circuit. The output of the multiplier is a periodic train of rectangular pulses whose frequency is related to the timing capacitance (the sensor) value, which in turn is related with the liquid level. The output of the multivibrator can be transmitted over a long distance, with frequency information undisturbed, even if there is a distortion in the pulse shape. Moreover, the pulse train can be directly taken to a processor based system without any ADC , thereby yielding a quasi-digital sensing system. At first, the sensitivity of the system was low because range of capacitance of the capacitive sensor is pretty low (within 11pF to 14pF). Also, frequency of the output signal of the 555-timer based astable multivibrator was fluctuating because the capacitive sensor was picking up too much noise from the environment. After incorporating a capacitance multiplier between capacitive sensor and 555-timer based astable multivibrator circuit those two problems were solved.

Capacitance multiplier constructed using op-amp, amplifies the capacitance of the capacitive sensor. The synthesized capacitor using capacitance multiplier also brings a series resistance, and a small leakage current which appears across the capacitance because of the input offsets of Op-amp. Also, only one terminal of the synthesized capacitor can be accessed. Another terminal will be always connected to the ground. But those disadvantages of present capacitance multiplier circuit are insignificant for our present overall circuits.

The Arduino mega 2560 microcontroller is used to analyse voltage signals of signal conditioning circuit. The range of input signals in Arduino mega 2560 microcontroller is between 0-5V but output of the signal conditioning circuit is 12V pulse signal with 17.4V peak voltage. Because of that, a voltage regulator is used of 5V using Zener diode which converted 12V pulse signal into 5V pulse signal.

The relation between the conductive liquid level and frequency of the signal conditioning circuit is nonlinear. Using Curve Fitting tools of MATLAB an equation relation between the liquid level and frequency was generated and the level of the conductive liquid information have been calculated accordingly from the measured frequency of the multivibrator output. Finally, the conductive liquid level information has been displayed on a 16×2 LCD display from Arduino mega 2560 microcontroller.

Future scope

The present work will contribute to implementation of a cost-effective capacitive level sensor for conductive liquid. However certain aspects of the present work can be further modified to improve the applicability and performance of the system.

- The present circuits can be used for any other application with capacitive sensor.
- Any value of capacitive sensor can be used, because its value can be adjusted using the capacitance multiplier.
- If value of the capacitive sensor is very small then there should be some arrangements which resist change of parasitic capacitance and electromagnetic interference in the system.

References

[1] Cihun-Siyong Alex Gong, Huan Ke Chiu, Li Ren Huang, Cheng Hsun Lin, Zen Dar Hsu, Po-Hsun Tu, "Low-Cost Comb-Electrode Capacitive Sensing Device for Liquid-Level measurment", *IEEE Journals*, vol. 16, No. 9, pp. 2896-2897, Feb 2016.

[2] Daniel Paczesny, Grzegorz Tarapata, Marzecki Michal, Ryszard Jachowicz, "The Capacitance Sensor for Liquid Level Measurement made with Ink-Jet Printing Technology", *Proc. Eurosensors*, UK, 2015.

[3] Ferran Reverter, Xiujun Li, Gerard C M Meijer, 'Liquid-level Measurement System Based on a Remote Grounded Capacitive Sensor', Journal on *Sensors and Actuators*, vol. 138, pp. 1-8, 2007.

[4] B. Hou, B. Zhou, M. Song, Z. Lin, R. Zhang, "A novel single excitation capacitive angular position sensor design," *Sensors*, vol. 16, no. 8, pp. 1196, 2016.

[5] P. Hu, J. Guo, J. Tan, "An annular planar-capacitive tilt sensor with a 360 measurement range," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2469–2476, Apr. 2016.

[6] N. Anandan, B. George, "A wide-range capacitive sensor for linear and angular displacement measurement," *IEEE Trans. Ind. Electron.*, vol. 64, no. 7, pp. 5728–5737, Jul. 2017.

[7] Y. Ye, C. Zhang, C. He, X. Wang, J. Huang, J. Deng," A review on applications of capacitive displacement sensing for capacitive proximity sensor," *IEEE Access*, vol. 8, pp. 45325-45342, 2020.

[8] H. Canbolat, "A Novel Level Measurement Technique Using Three Capacitive Sensors for Liquids", *IEEE Trans. Instrumentation and Measurement*, Vol.58, no.10, pp.3762-3768, 2009.

[9] Cheng- Ta Chiang, Yu-Chung Huang, "A Semicylindrical Capacitive Sensor With Interface Circuit Used for Flow rate Measurement", IEEE Sensors Journal, Vol.6, no.6, pp. 1564-1570, December 2006.

[10] Thomas Mohr, Uwe Ehrenberg, Hermann Uhlmann. "A new method for a Self–calibrating capacitive sensor", *IEEE Instrumentation and Measurement Technology Conference*, Budapest, Hungary, Vol.1, pp. 454-459, May 21 – 23, 2001.

[11] S. R. A. Ruth, V. R. Feig, M. Kim, Y. Khan, J. K. Phong, Z. Bao, "Flexible Fringe Effect Capacitive Sensors with Simultaneous High-Performance Contact and Non-Contact Sensing Capabilities," *Small Struct*, vol. 2, no. 2, p. 2000079, Feb. 2021.

[12] D. Wobschall, D. Lakshmanan, "Wireless soil moisture sensor based on fringing capacitance," in 2005 *SENSORS IEEE*, pp. 4, Oct 2005.

[13] R. N. Dean, A. K. Rane, M. E. Baginski, J. Richard, Z. Hartzog, D. J. Elton, "A capacitive fringing field sensor design for moisture measurement based on printed circuit board

technology," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 4, pp. 1105–1112, April 2012.

[14] R. B. McIntosh, M. E. Casada, "Fringing field capacitance sensor for measuring the moisture content of agricultural commodities," *IEEE Sensors Journal*, vol. 8, no. 3, pp. 240–247, March 2008.

[15] J. Mizuguchi, J. C. Piai, J. A. de Franc¸a, M. B. de Morais Franc¸a, K. Yamashita, and L. C. Mathias, "Fringing field capacitive sensor for measuring soil water content: Design, manufacture, and testing," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 1, pp. 212–220, Jan 2015.

[16] S. Malik, M. Ahmad, M. Punjiya, A. Sadeqi, M. S. Baghini, S. Sonkusale, "Respiration monitoring using a flexible paper-based capacitive sensor," in 2018 *IEEE SENSORS*, pp. 1–4, Oct 2018.

[17] F.N.Toth, G.C.M. Meijer, M. Van der Lee, "A planar capacitive precision gauge for liquid-level and leakage detection", *IEEE Trans. Instrumentation and Measurement*, Vol.46, no.2, pp. 644-646, 1997.

[18] T. N. Toth, G. C. M. Meijer, "A low-cost smart capacitive position sensor," *IEEE Trans. Instrumentation Measurements*, Dec. Vol.41, no.6, pp. 1041–1044, 1992.

[19] S. Malik, L. Somappa, M. Ahmad, S. Sonkusale, M. S. Baghini, "A Fringing Field Based Screen-Printed Flexible Capacitive Moisture and Water Level Sensor," *2020 IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS)*, *Manchester*, pp. 1-4, UK, 2020.

[20] P. Holmberg, "Automatic balancing of linear AC bridge circuits for capacitive sensor elements," in *IEEE Transactions on Instrumentation and Measurement*, vol. 44, no. 3, pp. 803-805, June 1995.

[21] E. Alnasser, "A Novel Fully Analog Signal Conditioning Circuit for Loss-Less Capacitive Sensor Estimation," in *IEEE Sensors Journal*, vol. 19, no. 18, pp. 8019-8026, 15 Sept.15, 2019.

[22] S. Sarath, R. Komaragiri, "Feasibility study of AC bridges in MEMS pressure sensors," *2012 IEEE Students' Conference on Electrical, Electronics and Computer Science*, pp. 1-4, Bhopal, India, 2012.

[23] P. Mantenuto, A. De Marcellis, G. Ferri, "Novel Modified De-Sauty Autobalancing Bridge-Based Analog Interfaces for Wide-Range Capacitive Sensor Applications," in *IEEE Sensors Journal*, vol. 14, no. 5, pp. 1664-1672, May 2014.

[24] N. F. Abdullah, T. N. Tuan Yaakub, M. H. Mamat, Y. Husaini, "Characterization of Heterostructure Capacitive Humidity Sensor Using Indirect Capacitance to Voltage Converter Circuit," *2023 IEEE Symposium on Industrial Electronics & Applications (ISIEA)*, pp. 1-5, Kuala Lumpur, Malaysia, 2023.

[25] A. Arshad, R. Tasnim, A. H. M. Z. Alam, S. Khan, "Capacitance-to-voltage converter design to measure small change in capacitance produced by human body movement," *2015 IEEE*

International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), pp. 114-117, Dhaka, Bangladesh, 2015.

[26] M. A. Jani, W. Ahmad, "Capacitance to Time-Period Conversion Using modified Astable Multivibrator," in *IEEE Transactions on Industrial Electronics and Control Instrumentation*, vol. IECI-22, no. 3, pp. 430-432, Aug. 1975.

[27] T. Islam, O. P. Maurya, "Design and fabrication of non-contact fringing field capacitive sensor for liquid level measurement," *2019 IEEE 16th India Council International Conference (INDICON)*, pp. 1-4, Rajkot, India, 2019.

[28] C. Hong, L. Chu, W. Lai, A. -S. Chiang, W. Fang, "Implementation of a New Capacitive Touch Sensor Using the Nanoporous Anodic Aluminum Oxide (np-AAO) Structure," in *IEEE Sensors Journal*, vol. 11, no. 12, pp. 3409-3416, Dec. 2011.

[29] J. I. Sohn, Y. S. Kim, C. Nam, B. K. Cho, T. Y. Seong, "Fabrication of high density arrays of individually isolated nanocapacitors using anodic aluminum oxide templates and carbon nanotubes," *Appl. Phys.Lett.*, vol. 87, pp. 123115, 2005.