

DESIGN OF CAPACITIVE ANGLE SENSOR FOR AUTOMOBILE BRAKE SYSTEM

A Thesis

Submitted By

MOHD SUAIB DANISH

for the award of the degree

of

MASTER OF TECHNOLOGY

Under the guidance of

Prof. Jagadeesh Kumar V



**DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS**

May 2014

THESIS CERTIFICATE

This is to certify that the thesis titled “**Design of Capacitive Angle Sensor for Automobile Brake System** ”, submitted by **Mr. Mohd Suaib Danish**, to the Indian Institute of Technology Madras, Chennai for the award of the degree of **Master of Technology**, is a bona fide record of research work done by him under my supervision. The contents of this thesis, in full or a part has not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Jagadeesh Kumar V

Research Guide

Professor (CEC Head)

Dept. of Electrical Engineering

IIT-Madras, Chennai-600036

Place: Chennai

Date: 10 May 2014

ACKNOWLEDGEMENTS

I would sincerely like to thank my guide Dr. Jagadeesh Kumar V, Professor (CEC Head) for his support and for his supervision during the project. I am also indebted to him for his guidance in different fields specially in sensors and instrumentation and for sharing his knowledge and resources.

I am grateful to Dr. Bobby George, Assistant Professor, Department of Electrical Engineering, IIT Madras for his encouragement and support during the whole project. He is one of the fewest teachers that you will never forget his way of dealing with circuits at high and well organized levels.

I would like to thank my parents for their unconditional love, encouragement and support that they have given me throughout the years I have gotten to where I am today. I am deeply grateful for all their hard work and sacrifice.

I want to thank all the teaching and non-teaching staff of the Department especially from Measurements and Instrumentation Lab, for their great help.

Finally I'd like to say thanks to all my friends, batchmates and labmates for supporting me and making my project successful.

Mohd Suaib Danish

EE12M104

ABSTRACT

This thesis focuses on the design, fabrication and hardware testing of the capacitive angle transducer. The designed capacitive angle transducer provides the digital output which is linearly proportional to the angle being sensed. The range of angle for which testing is done is 0° to 180° . The variations in the sensor capacitances are converted to digital employing dual slope digital conversion principle. High accuracy is easily obtained since the output is dependent only on a dc reference voltage and a high frequency clock.

The prototype of the sensor is fabricated using copper clad boards and etchant (ferric chloride). The sensor has given nice variation in differential push-pull type of capacitances. The transducer is first simulated on the LTSpice which gave ideal results. Then hardware testing is done by building signal conditioning circuit on NI ELVIS board. ATmega328 microcontroller is used for proper functioning of the signal conditioning circuit and display of angle sensed.

This type of angle sensor finds application in automobile brake system in order to sense the life of brake shoes. In the brake system as the brake shoes wear-out, they make angular movement. So this sensor can be attached to the brake shoes to sense the amount of angular displacement (in degrees) and hence the life of brake shoes can be sensed. The dimensions of the prototype are set such that it can be applied to the brake system.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	i
ABSTRACT	ii
CONTENTS	iii
LIST OF TABLES	v
LIST OF FIGURES	vi
ABBREVIATIONS	vii

CHAPTER 1: INTRODUCTION

1.1. About Transducer	1
1.2. Capacitive Transducer	2
1.3. Objective and Scope of work	4
1.4. Organization of Work	4

CHAPTER 2: SENSOR DESIGN

2.1. Differential Capacitive-Sensor System	5
2.2. LVDCAT	7
2.3. Sensor Fabrication	10

CHAPTER 3: SIGNAL CONDITIONING CIRCUIT

3.1. Introduction	11
3.2. Circuit	12
3.3. Working	12
3.3.1 Auto-Zero Phase	14
3.3.1 Conversion Cycle.....	15

CHAPTER 4: SIMULATION ANALYSIS

4.1. Simulation Tool-LTSpice IV	19
4.1.1. Main differences between LTSpice and Spice	19
4.1.2. Differences between PSpice and LTSpice	20
4.1.3. Differences between PSpice and LTSpice	20
4.1.4 Simulation Functions, Capabilities and Limitations.....	21
4.1.5 Component Libraries.....	22

4.1.6 Models.....	22
4.1.7 Graphical Viewer.....	22
4.2. Simulation Model.....	23
4.3. Simulation Results.....	26
 CHAPTER 5: EXPERIMENTAL SETUP AND RESULTS	
5.1. Methodology	29
5.2. Arduino Platform	29
5.2.1 Arduino Uno Board	30
5.2.2 ATmega328 Microcontroller	31
5.3 NI ELVIS II.....	33
5.3.1 Applications	36
5.3.2 NI ELVIS II Bench top Workstation	36
5.3.3 NI ELVIS II Series Prototyping Board	36
5.3.4 NI ELVIS Functions.....	37
5.4 Circuit Testing.....	38
5.5 Experimental Results.....	39
 CHAPTER 6: CONCLUSION AND FUTURE WORK	
6.1. Conclusion.....	43
6.2. Future Scope.....	43
 APPENDIX	 44
REFERENCES.....	49

LIST OF TABLES

Table 2.1 Specifications and dimensions of the sensor.....	10
Table.4.1: Pulse signal used in simulations.....	24
Table 4.2: Value of T_2/T_1 at different angular positions(simulation).....	28
Table 5.1: Value of T_2/T_1 at different angular positions(experimental).....	41

LIST OF FIGURES

Figure 1.1: Principle of Sensor/Transducer.....	2
Figure 2.1: Electrical equivalent circuit of a differential capacitive sensor.	6
Figure 2.2: Sensor part of LVDCAT.....	7
Figure 2.3: Top view of the sensor showing middle plate at different positions.....	8
Figure 2.4: Variation of sensor capacitances C_1 and C_2 and linear function with θ	9
Figure 3.1: Signal Conditioning Circuit.....	12
Figure 3.2: Integrator output v_{oi} and the comparator output v_c	13
Figure 3.3: Flowchart of the auto-zero phase	14
Figure 3.4: Flowchart showing the logic of the conversion phase	16
Figure 4.1: The LTSpice ‘Edit Simulation Command’.....	23
Figure 4.2: Simulation circuit in LTspiceIV	24
Figure 4.3: Waveforms for controlling switches during time.....	25
Figure 4.4: Integrator outputs in LTspiceIV.....	27
Figure 4.5: Linear variation of T_2/T_1 w.r.t θ	28
Figure.5.1: Block Diagram.....	29
Figure 5.2: The Arduino Uno Board.....	30
Figure 5.3: ATmega328 Microcontroller Architecture.....	31
Figure 5.4: Block diagram of the AVR CPU Core architecture.....	33
Figure 5.5: NI ELVIS II hardware.....	35
Figure 5.6: NI ELVIS II Soft Panel.....	35
Figure 5.7: Prototyping Board Description.....	37
Figure 5.8: Experimental Setup of the angle transducer.....	39
Figure.5.9: Integrator outputs.....	40
Figure 5.10: Linear variation of T_2/T_1 w.r.t θ	42

ABBREVIATIONS

ADC	Analog-To-Digital Converter
AVR	Advanced Virtual RISC
CDC	Capacitance-To-Digital Converter
CLU	Control and Logic Unit
CMOS	Complementary Metal Oxide Semiconductor
DMM	Digital Multimeter
DSA	Dynamic Signal Analyzer
DSCDC	Dual-Slope CDC
EEPROM	Electrically Erasable Programmable Read-Only Memory
EMI	Electro-Magnetic Interference
FGEN	Function Generator
FR	Flame Resistant
ICSP	In Circuit Serial Programming
LVDCAT	Linearly Variable Differential Capacitive Angular Transducer
NI ELVIS	National Instruments Educational Laboratory Virtual Instrumentation Suite
PCB	Printed Circuit Board
SPDT	Single-Pole Double-Throw
SRAM	Static random-access memory
USB	Universal Serial Bus
VPS	Variable Power Supply

CHAPTER 1

INTRODUCTION

1.1 About Transducer

An electronic instrumentation system consists of a number of components which together are used to perform a measurement and record the result. An instrumentation system generally consists of three major elements: an input device, a signal conditioning device and processing device and an output device.

The input device receives the quantity that is to be measured and delivers a proportional electrical signal to the signal conditioning device. In the signal conditioning device, the signal is amplified, filtered or modified to a format acceptable to the output device. The output device may be a simple indicating meter, an oscilloscope, or a chart recorder for visual display. The kind of system depends on what is to be measured and how the measurement result is to be presented.

The input quantity for most instrumentation system is nonelectrical. In order to use electrical methods and techniques for measurement or control of input non-electrical quantity, it must be converted into an electrical signal. And this conversion is made by a device called “Transducers”.

The sensor or the sensing element is the first element in a measuring system and takes information about the variable being measured and transforms it into a more suitable form to be measured. Figure 1.1 illustrates the difference between sensor and transducer.

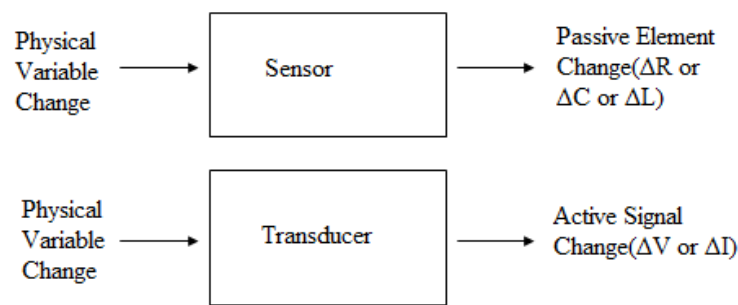


Figure 1.1: Principle of Sensor/Transducer.

Sensor is sometimes called a primary measuring element, it can be found simply as a capacitive sensor to measure the capacitance. It may be embedded in the transducer to perform its function. That means the transducer consists of a primary element (sensor) plus a secondary element (signal conditioning circuit) that transforms the passive change or small voltage signal into active signal range that can be easily used in other chains of the control loop.

Example: In capacitive angular sensor, the capacitance depends on the angular movement of the sensor. It can be inserted into a signal conditioning circuit (secondary element) in order to transform the change in the capacitance value to a change in the voltage output.

Finally, the output voltage from the signal conditioning circuit express about the angular change value. In general, we can say that:

$$\text{Transducer} = \text{Sensor} + \text{Signal conditioning circuit}$$

1.2 Capacitive Transducer

Capacitive transducers are nothing but the capacitors with the variable capacitance. These are mainly used for the measurement of displacement, pressure, angle etc. They are passive part of transducer. The capacitive transducer comprises of two parallel metal plates that are separated by the material such as air, which is called as the dielectric material. In the typical capacitor the distance between the two plates is fixed, but in variable capacitance transducers the distance between the two plates is variable as in case of capacitive pressure sensors. Sometimes in variable capacitor these parallel plates are made to slide over each other in order to change the area of overlap which then leads to change in capacitance. Such type of variable capacitors is used in angular sensors.

The capacitance C between the two plates of capacitive transducers is given by:

$$C = \epsilon_o \epsilon_r \frac{A}{d}$$

Where C is the capacitance of the capacitor or the variable capacitance transducer

ϵ_o is the absolute permittivity

ϵ_r is the relative permittivity

The product of ϵ_o & ϵ_r is also called as the dielectric constant of the capacitive transducer.

A is the area of the plates

D is the distance between the plates

It is clear from the above formula that capacitance of the capacitive transducer depends on the area of the plates and the distance between the plates. The capacitance of the capacitive transducer also changes with the dielectric constant of the dielectric material used in it.

Thus the capacitance of the variable capacitance transducer can change with the change of the dielectric material, change in the area of the plates and the distance between the plates. Depending on the parameter that changes for the capacitive transducers, they are of three types as mentioned below.

- **Changing Dielectric Constant type of Capacitive Transducers**

In these capacitive transducers the dielectric material between the two plates changes, due to which the capacitance of the transducer also changes. When the input quantity to be measured changes the value of the dielectric constant also changes so the capacitance of the instrument changes. This capacitance, calibrated against the input quantity, directly gives the value of the quantity to be measured. This principle is used for measurement of level in the hydrogen container, where the change in level of hydrogen between the two plates results in change of the dielectric constant of the capacitance transducer. Apart from level, this principle can also be used for measurement of humidity and moisture content of the air.

- **Changing Area of the Plates of Capacitive Transducers**

The capacitance of the variable capacitance transducer also changes with the area of the two plates. This principle is used in the torque meter, used for measurement of the torque on the shaft. This comprises of the sleeve that has teeth cut axially and the matching shaft that has similar teeth at its periphery.

- **Changing Distance between the Plates of Capacitive Transducers**

In these capacitive transducers the distance between the plates is variable, while the area of the plates and the dielectric constant remain constant. This is the most commonly used type of variable capacitance transducer. For measurement of the displacement of the object, one plate of

the capacitance transducer is kept fixed, while the other is connected to the object. When the object moves, the plate of the capacitance transducer also moves, this results in change in distance between the two plates and the change in the capacitance. The changed capacitance is measured easily and it calibrated against the input quantity, which is displacement. This principle can also be used to measure pressure, velocity, acceleration etc.

In the instruments using capacitance transducers the value of the capacitance changes due to change in the value of the input quantity that is to be measured. This change in capacitance can be measured easily and it is calibrated against the input quantity, thus the value if the input quantity can be measured directly.

1.3 Objective and scope of the work

The main objective of the work described in this thesis is to design a prototype of angle sensor which will sense the angular movement and display the angle on display unit. This type of angle sensor finds application in automobile brake system to sense the life of brake shoes. As the brake shoes wear-out, they make angular movement. If we connect the middle plate of our angle sensor to these shoes, the angular movement of these shoes can be sensed and hence the life of the shoes. Since the sensor is fabricated using copper clad board so it is an affordable sensor.

1.4 Organization of Work

A brief introduction to transducers and capacitive transducer is presented in Chapter1. Chapter 2 deals with design of the angle sensor and its fabrication technique. Chapter 3 explains the working of signal conditioning circuit. Chapter 4 gave a brief introduction to LTSpice and simulation results are shown here. Chapter 5 shows the experimental results and gave an idea about the hardware used. The conclusion of the work carried out and its future scope is provided in Chapter 6.

CHAPTER 2

SENSOR DESIGN

2.1 Differential Capacitive-Sensor System

Capacitive sensor elements can be applied in many applications to measure many different types of signals such as displacement, proximity, humidity, acceleration, liquid level, gas concentration, etc.. They can be implemented on printed-circuit boards, glass substrates, silicon chips, or other types of material. Because the electrodes of a capacitive sensor element do not need to be in mechanical contact with each other, they are suited for small-range contact-less sensing. The attractive properties of capacitive sensors are that they consume very little power, their cross sensitivity to temperature is very low, and shielding stray electric fields is less complex than shielding, for instance, inductive sensors from magnetic disturbances. The main drawbacks of capacitive sensors concern their sensitivity to contamination and condensation, and their sensitivity to Electro-Magnetic Interference (EMI).

Depending on the application, capacitive sensor can be floating (i.e. sensors in which neither of the electrodes is grounded) or grounded (i.e. sensors in which one of the electrodes is grounded). Based on the properties of the electrode structure and the dielectric material, the electrical properties of capacitive sensors can differ significantly. For instance, they can demonstrate pure capacitive behavior or can have resistive leakage. Their values can range from less than one pF up to hundreds of pF or even to nF. Sometimes their values can change very fast, such as in displacement sensors for servo systems, while in other applications their values can be semi-static. Besides the aforementioned sensor conditions, the effects of parasitic capacitances of the connecting wires should also be taken into account.

For sensing displacement (linear and angular), pressure, and acceleration, capacitance-type sensors are widely employed in the industry as they provide better resolution, sensitivity, and linearity compared with other types of sensors. A differential capacitive sensor has two sensing capacitances C_1 and C_2 whose values change in proportion to the parameter being sensed, and

the changes in C_1 and C_2 are normally equal and opposite. Since the value of one of the capacitances increases and that of the second capacitance decreases in proportion to the parameter being sensed, differential capacitive sensors are also popularly known as push–pull-type capacitive sensors[1]. A simplified electrical equivalent circuit of a typical differential capacitive sensor is shown in Figure 2.1.

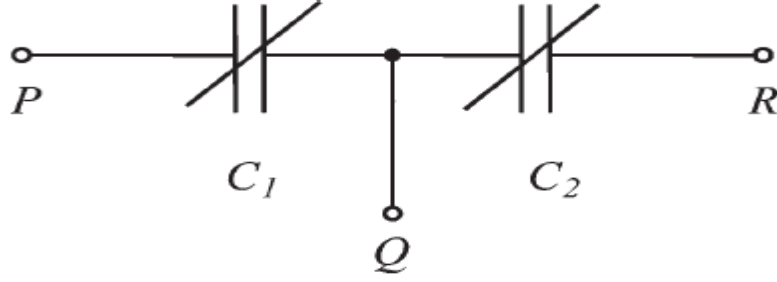


Figure 2.1: Electrical equivalent circuit of a differential capacitive sensor.

If the parameter being sensed alters the area between the plates of the sensor, then such a capacitive sensor will possess a linear input–output relationship, as given by the following:

$$\begin{aligned} C_1 &= C_0(1 \pm kx) \\ C_2 &= C_0(1 \mp kx) \end{aligned} \quad (2.1)$$

Here, k is the transformation constant of the sensor, and C_0 is the nominal value of C_1 and C_2 , when x , which is the physical quantity being sensed, is zero. Alternatively, a capacitive sensor that utilizes the distance between the plates as the transduction parameter will possess an inverse relationship, as given by the following:

$$\begin{aligned} C_1 &= \frac{C_0}{(1 \mp kx)} \\ C_2 &= \frac{C_0}{(1 \pm kx)} \end{aligned} \quad (2.2)$$

2.2 Linearly Variable Differential Capacitive Angular Transducer-LVDCAT

The sensor part of the proposed linearly variable differential capacitive angular transducer, shown in Figure 2.2, consists of three circular shaped conducting plates mounted concentrically one above another. While the top and bottom plates (TP and BP) are firmly fixed, the middle plate MP, having a semi-circular shape, freely rotates between the top and bottom plates. The bottom plate is made to divide into two semi-circular plates.

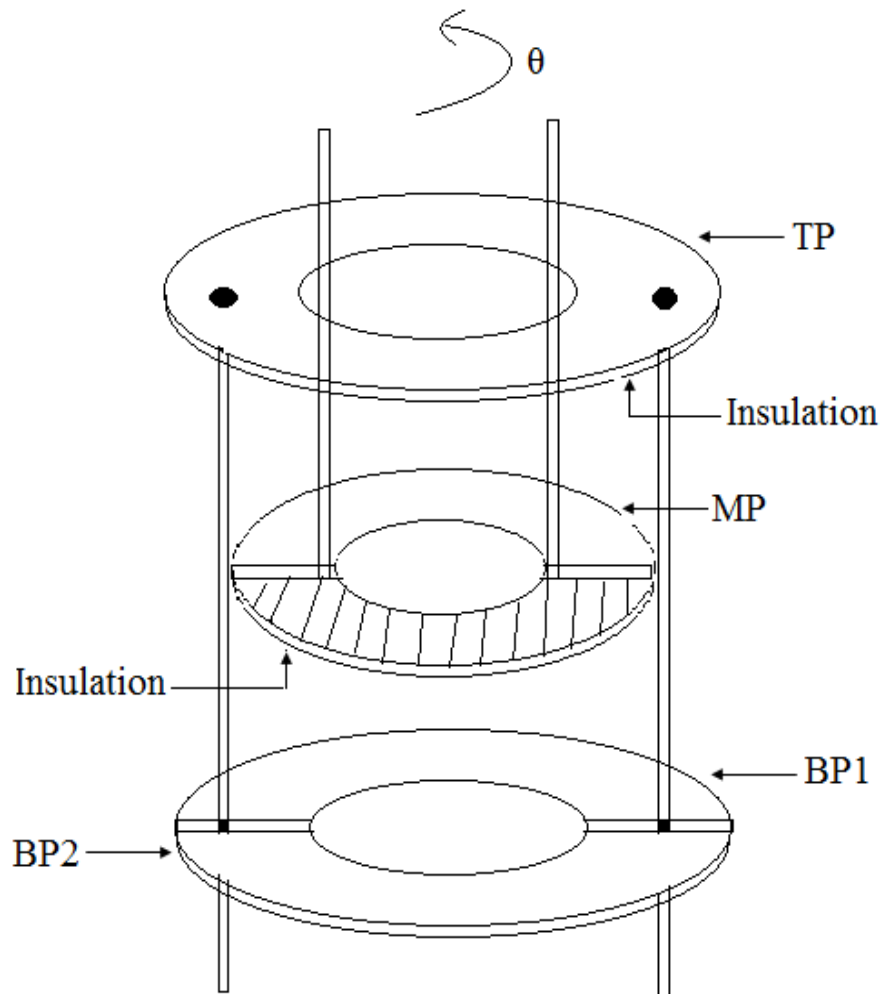


Figure 2.2: Sensor part of LVDCAT.

The top, middle and the bottom plates are electrically insulated from each other. The two, semi circle parts of the bottom plate are identical in dimensions and are insulated from one another.

The circular top plate is positioned such that it is aligned with bottom plate when both of its parts, BP1 and BP2, are placed appropriately forming the full circle. The middle, semi circular plate MP mounted on a spindle and suitably anchored, rotates freely between the top and bottom plates. The spindle attached to the middle plate MP is mechanically linked to the element whose angular position θ is to be measured. The inner and outer radius of the middle plate is made slightly lesser in order to make mechanical connection between middle plate and spindle. If the middle plate MP and the two bottom semi circular plates are all kept at the same potential, it nicely turns out that the resulting two capacitances $C1$ and $C2$ between lead pairs TP-BP1 and TP-BP2 respectively vary as θ of a pair of plates of a capacitance. As θ varies from 0° to 180° , the values of capacitances $C1$ and $C2$ will vary as indicated in Figure 2.3, resulting in a push-pull or differential pair of capacitances.

Mathematically the variation of capacitances $C1$ and $C2$ in terms of θ can be expressed as:

$$\begin{aligned} C1 &= C0 + (CM - C0) \left(\frac{\theta}{180} \right) \\ C2 &= C0 + (CM - C0) \left[1 - \left(\frac{\theta}{180} \right) \right] \end{aligned} \quad (2.3)$$

Figure 2.3 gives the visual idea about the movement of middle plate between top and bottom plates of the sensor. For clarity the top plate, insulating plates and spindles are not shown in the figure.

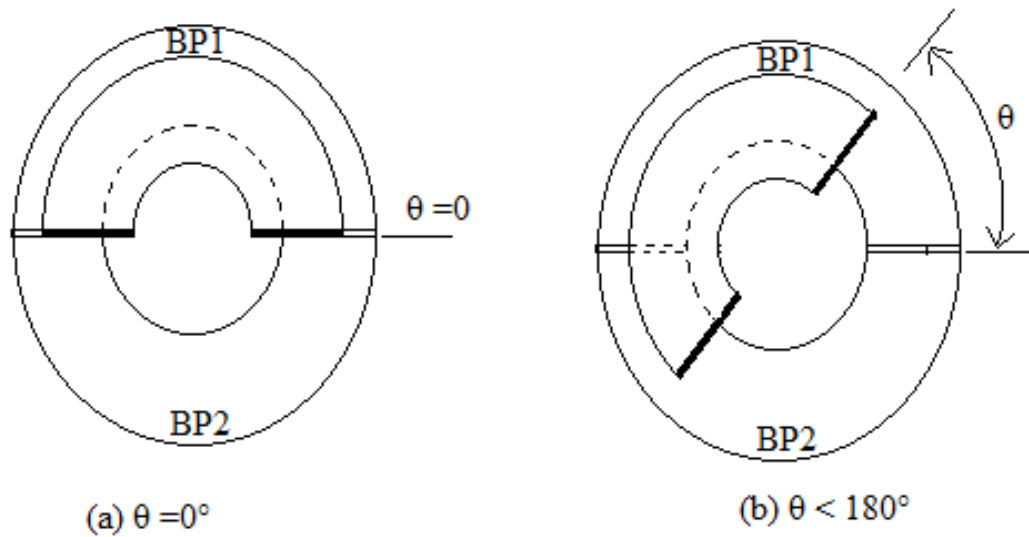


Figure 2.3: Top view of the sensor showing middle plate at different positions.

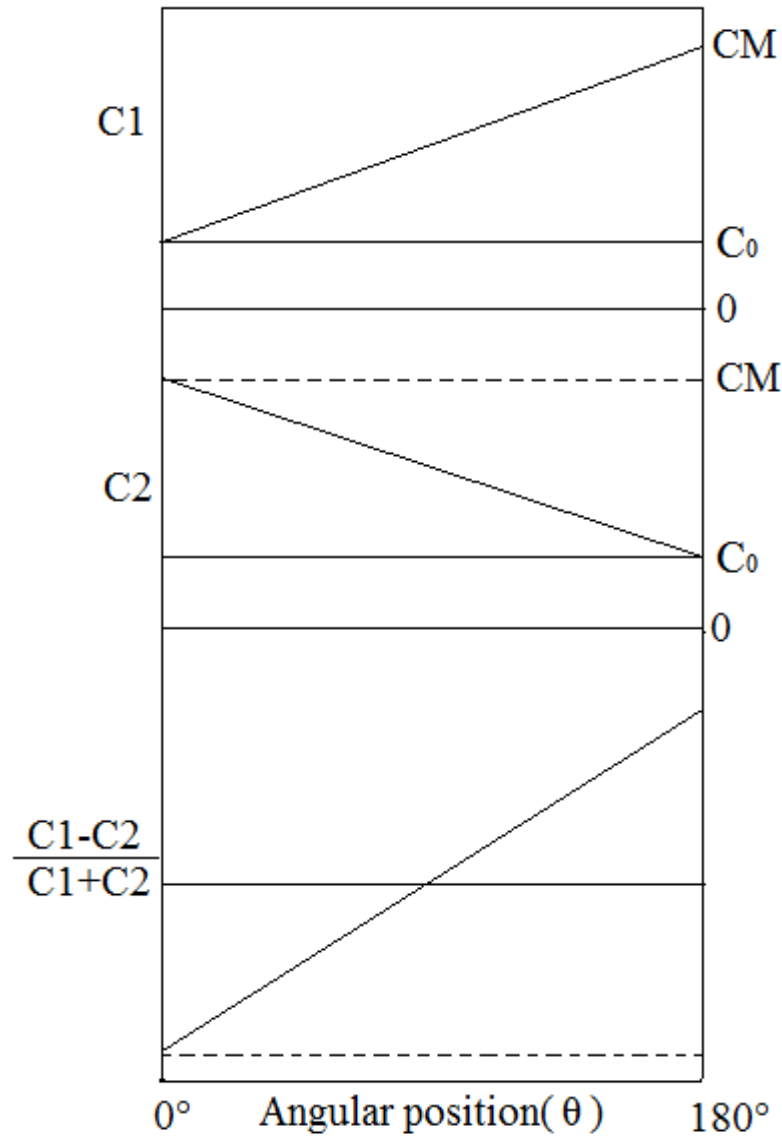


Figure 2.4: Variation of sensor capacitances C_1 and C_2 and linear function with θ .

It is easily seen from the Figure 2.4, that the value of the capacitance C_1 increasing from C_0 to CM in the interval $0^\circ < \theta < 180^\circ$ and simultaneously the value of capacitance C_2 is decreasing from CM to C_0 where CM is the maximum capacitance and C_0 is minimum capacitance. Assuming C_0 to be equal to zero, the value of (C_1-C_2) is continuously increasing from $-CM$ to CM . If (C_1-C_2) is divided by (C_1+C_2) we will get a linear plot ranging from -1 to 1 . Thereafter final plot is deduced by adding 1 to the previous plot which gave linearly varying function with respect to θ .

2.3 Sensor Fabrication

The fabrication process of the sensor is rather simple. The sensor is fabricated using the single sided copper clad boards. The copper of these boards acts as electrodes of the capacitor. The board material here is FR-4(FR stands for flame resistant) which acts as insulator between the electrodes of the capacitive sensor.

Firstly, three copper clad boards were cut by the lathe machine in circular disc shapes of required dimensions. The copper the middle plate is then etched using ferric chloride (etchant) in order to make it semi circular disc like shape. Similarly a strip of copper is removed from the bottom plate in order to divide it into two identical semi circular parts. The specifications of the sensor are depicted in Table 2.1.

Electrode Material	Copper
Insulator	FR-4(Flame Resistant-4)
Electrode thickness	10 μ m
Insulator thickness	1.5mm
Dielectric Constant of FR-4	4.8
Inner radius of top plate	2.5cm
Outer radius of top plate	7.5cm
Inner radius of middle plate	1.8cm
Outer radius of middle plate	6cm
Inner radius of bottom plate	2.5cm
Outer radius of bottom plate	7.5cm

Table 2.1: Specifications and dimensions of the sensor.

CHAPTER 3

SIGNAL CONDITIONING CIRCUIT

3.1 Introduction

To obtain a measurable output relative to the parameter being sensed by a capacitive sensor, a signal-conditioning circuit that converts the variations in the sensor capacitances C_1 and C_2 to a proportional analog voltage or current is required [2].

Digital instrumentation systems are preferred over analog systems as they offer better user interface and excellent processing power. To interface a sensor to a digital instrumentation system, an analog-signal-conditioning circuit cascaded to an analog-to-digital converter (ADC) is required. A typical ADC will possess an analog part and a digital part. It would be advantageous if the analog part of an ADC itself is designed to accept the elements of a sensor and the ADC logic appropriately modified to provide a digital output directly proportional to the parameter being sensed. Such a scheme does not require a separate analog-signal-conditioning unit. A direct digital converter suitable for a differential resistive sensor has been reported [3]. Methods based on a sigma–delta modulator [4], auto-ranging [5], and duty cycle variation [6] proposed earlier are suitable only for a single-element-type capacitive sensor. A method suitable for differential capacitive sensors has been proposed in [7]. However, this method provides an output proportional to the ratio of two capacitances, and hence, the operation of the method is limited to a small range of variations in the capacitances of the sensor. A charge-balancing-type capacitance-to-digital converter (CDC) reported earlier employs a very complicated switching arrangement [8]. The successive approximation-type CDC that uses the well-known successive approximation register structure requires a high-precision digital-to-analog converter [9]. ICs AD7746 and AD7747 marketed by Analog Devices Inc. are CDCs that convert single or differential capacitances to an equivalent digital output utilizing the sigma–delta principle [10]. Since the output of AD7746/AD7747 is dependent on the nominal values of the sensor capacitances, as they do not employ the ratio-metric method, additional processing is required if an output directly proportional to the parameter being sensed is required. Moreover, these ICs

can accept capacitances only up to 21 pF and have an active but limited compensation range over which the effect of stray capacitances on the output can be nullified.

A novel switched-capacitor dual-slope technique that converts the variations in the capacitances C_1 and C_2 of a differential capacitive sensor directly into a proportional digital value based on a ratio-metric approach was discussed [11]. The conversion time of the proposed technique is 33% less than that of the triple slope CDC reported earlier [12]. Apart from the increased speed of conversion, the dual-slope CDC (DSCDC) offers additional advantages, i.e., negligible sensitivity toward various error sources, such as stray capacitances, switch leakage currents, and op-amp offset voltage, in comparison with the triple-slope CDC. Moreover, this method requires only a single dc reference voltage and hence avoids errors arising out of mismatched dc reference voltages (offset voltage) of opposite polarity, which is a major source of error in the triple-slope CDC scheme.

3.2 Circuit

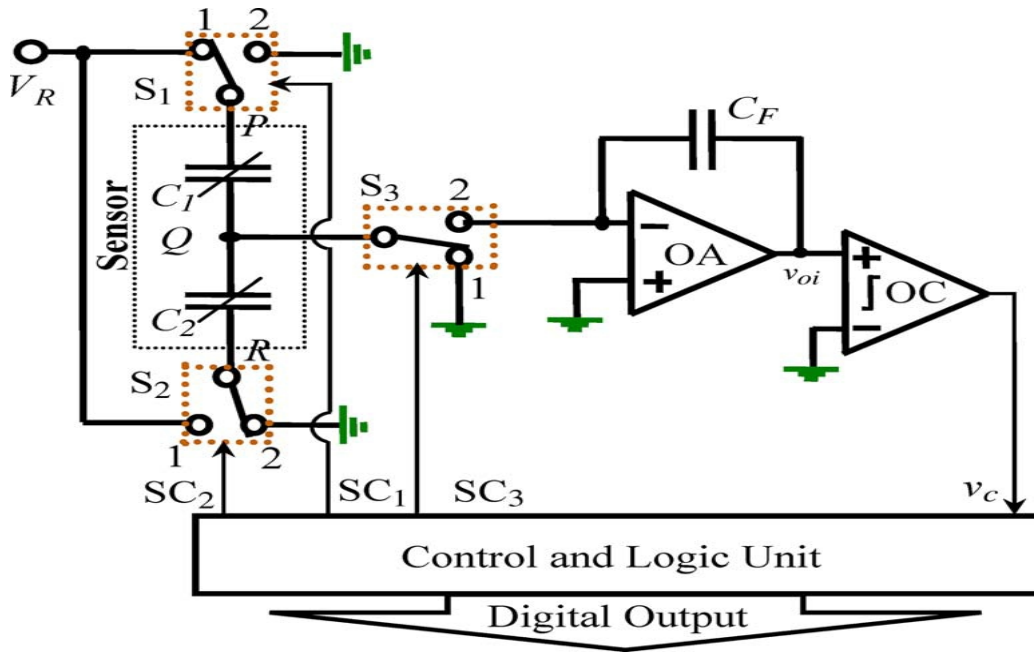


Figure 3.1: Signal Conditioning Circuit [1].

3.3 Working

The functional block diagram of the signal conditioning circuit is shown in Figure 3.1. As in any dual-slope technique, the circuit is made of an integrator and a control and logic unit (CLU)

incorporating an n -bit or N -digit timer counter. As indicated in Figure 3.1, the sensor capacitances C_1 and C_2 in combination with three single-pole double-throw (SPDT) switches S_1 , S_2 , and S_3 , op-amp OA and the feedback capacitor C_F form a switched-capacitor integrator. The status of the output of the integrator is sensed by the comparator OC. If the output of the integrator $v_{oi} \geq 0$, then the output v_c of the comparator will be high; otherwise, the comparator output will be low. A high-to-low or low-to-high transition on v_c indicates that the output v_{oi} is crossing through zero. The CLU senses output v_c of the comparator and controls the switches through control lines SC1, SC2, and SC3, and performs two integrations for a complete conversion. All the operational logic of the CLU is timed by a clock signal possessing a period T_C and 50% duty cycle. As in a conventional dual-slope ADC, the CLU of the DSCDC performs an auto-zero function to force the integrator output to zero to ensure an appropriate initial condition for initiating a conversion.

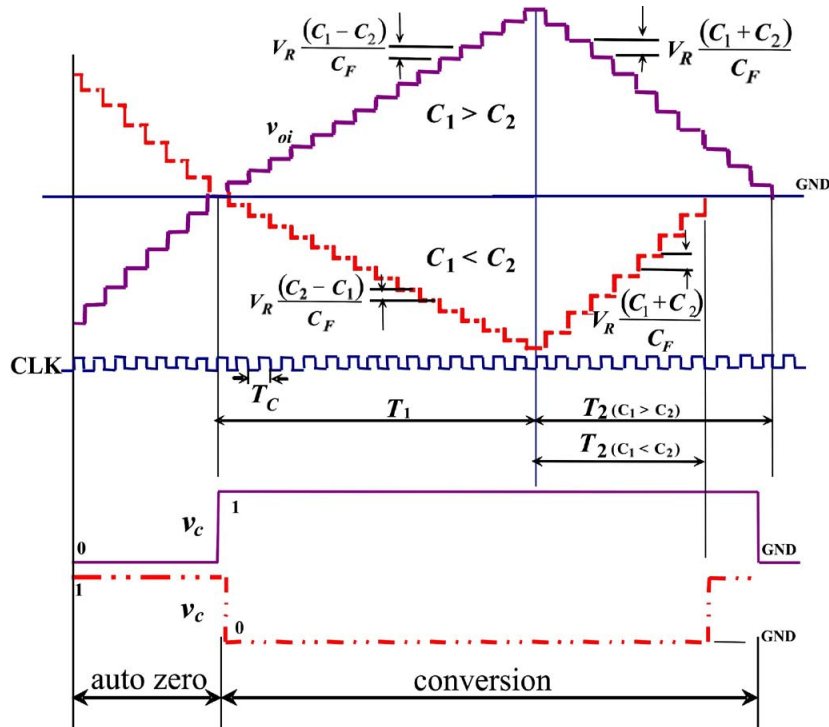


Figure 3.2: Integrator output v_{oi} and the comparator output v_c for (solid line) $C_1 > C_2$ and (dotted line) $C_1 < C_2$. In the auto-zero phase, the integrator voltage is assumed to be positive for $C_1 < C_2$, and vice versa [1].

3.3.1 Auto-Zero Phase

The auto-zero ensures that the integrator output voltage v_{oi} is initialized to zero prior to the start of the first integration period T_1 . Forcing the integrator output to become zero can also easily be accomplished by shorting C_F (discharging the feedback capacitor) with a switch placed across it. However, such an arrangement will then introduce an error due to the offset voltage of the comparator and is hence not preferred. The duration of the auto-zero cycle can be considerable at the start of the first conversion cycle, but it can be as low as one clock cycle from the second conversion cycle onward when the CDC is operated in a continuous conversion mode. In the auto-zero phase, the CLU senses the comparator output v_c . If v_c is high (integrator output $v_{oi} \geq 0$), then the CLU logic is designed to set S_1 and S_2 to be in position 1 and S_3 to be in position 2 for a period $T_{C/2}$ (whenever the clock is high) by suitably controlling lines SC_1 , SC_2 , and SC_3 . When the clock turns low (at the end of $T_{C/2}$), all three switches are toggled and kept in that position for the next $T_{C/2}$.

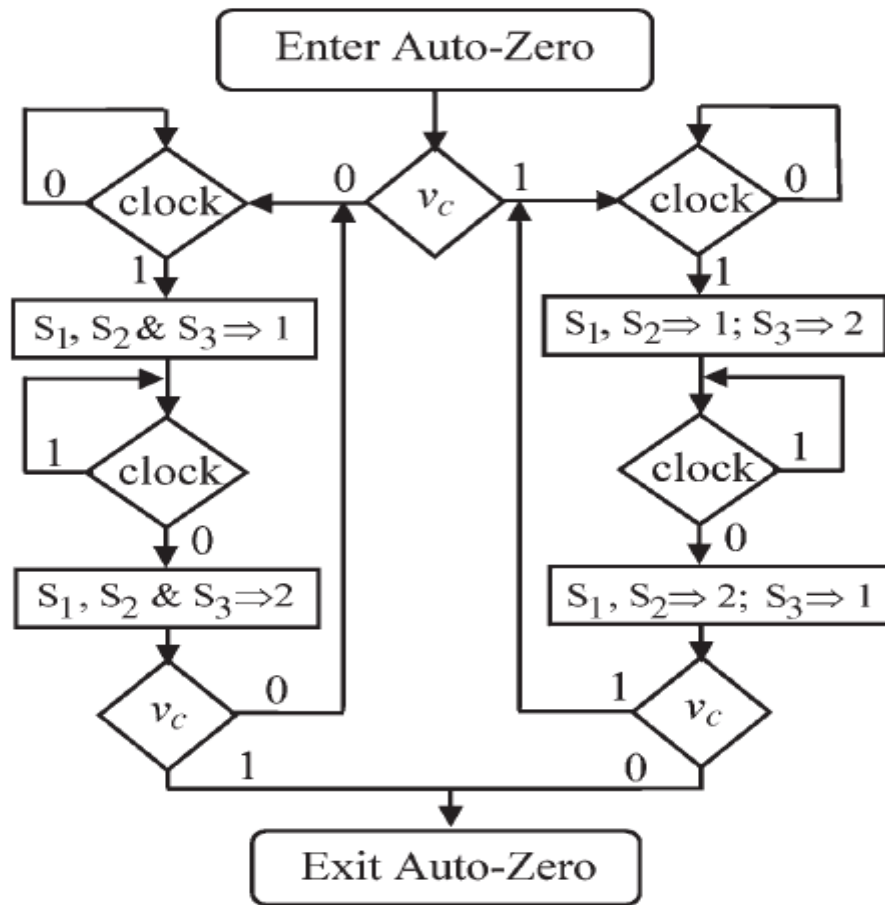


Figure 3.3: Flowchart of the auto-zero phase [1].

Thus, during the first $TC/2$, the capacitors $C1$ and $C2$ get charged to VR with the charging current discharges CF . In the succeeding $TC/2$, $C1$ and $C2$ are discharged and hence made ready for the charging again in the first half of the next clock cycle. Hence, for every clock cycle, the output of the integrator voi ramps down with a step of $\Delta v = VR(C1 + C2)/CF$ and reaches zero. Figure 3.2 shows the pertinent waveforms at cardinal points of the scheme. When voi reaches zero, the comparator output vc will transit from high to low, signaling to the CLU that voi has reached zero and the CLU enters the conversion phase.

If, in the beginning, voi is negative, then vc will be low. In such a case, the CLU logic sets $S1$, $S2$, and $S3$ to be in position 1 for a period $TC/2$, toggles all the switches, and keeps them in the toggled position for the next $TC/2$. Hence, during the first half of the clock, $C1$ and $C2$ get charged to VR , and during the second half, they discharge to ground, drawing a charge of $VR(C1 + C2)$ from CF . Hence, for every clock cycle, the output of the integrator voi ramps up with a step of $\Delta v = VR(C1 + C2)/CF$ and reaches zero. Once voi becomes zero, the comparator output vc will transit from low to high, indicating to the CLU the end of the auto-zero phase.

3.3.2 Conversion Cycle

A typical conversion cycle consists of two time periods, i.e., $T1$ and $T2$. While the period $T1$ is a preset value, $T2$ is measured. During $T1$, when the clock is high ($TC/2$), switch $S2$ is set at position 2, and switches $S1$ and $S3$ are set to position 1. When the clock turns low, switch $S2$ is changed to position 1, and $S1$ and $S3$ are set to position 2. Thus, whenever the clock is high, $C1$ will charge to VR , and $C2$ will get discharged to ground. As soon as the clock goes low, the charge in $C1$ will be transferred to CF , and at the same time, the charging current of $C2$ is also sent into CF . Hence, the differential charge between $VRC1$ and $VRC2$, i.e., $VR(C1 - C2)$, will get transferred to CF for every clock cycle. If $C1 > C2$, then voi ramps in the positive direction in steps of value $VR(C1 - C2)/CF$ for every clock period TC , as indicated by the solid line in Figure 3.2. On the other hand, if $C2 > C1$, then voi will ramp in the negative direction in steps of value $VR(C2 - C1)/CF$ for every clock, as indicated by the dotted line in Figure 3.2. If $C1 = C2$, then the differential charge transferred to CF for every clock is zero; hence, at the end of $T1$ ($= N1TC$, with $N1$ being a preset integer), the output of integrator voi will remain zero. The second integration period $T2$ commences as soon as the period $T1$ is completed. The CLU senses the

output of the comparator v_c at the end of T_1 and also starts an internal counter which is clocked by TC to measure T_2 . If v_c is high at the end of T_1 (i.e., $C_1 > C_2$), then the CLU switches S_1 and S_2 to position 1 and S_3 to position 2 whenever the clock is high and toggles all three switches whenever the clock turns low. Thus, when the clock is high, both C_1 and C_2 will get charged to VR, and their charging currents will discharge CF by VR ($C_1 + C_2$).

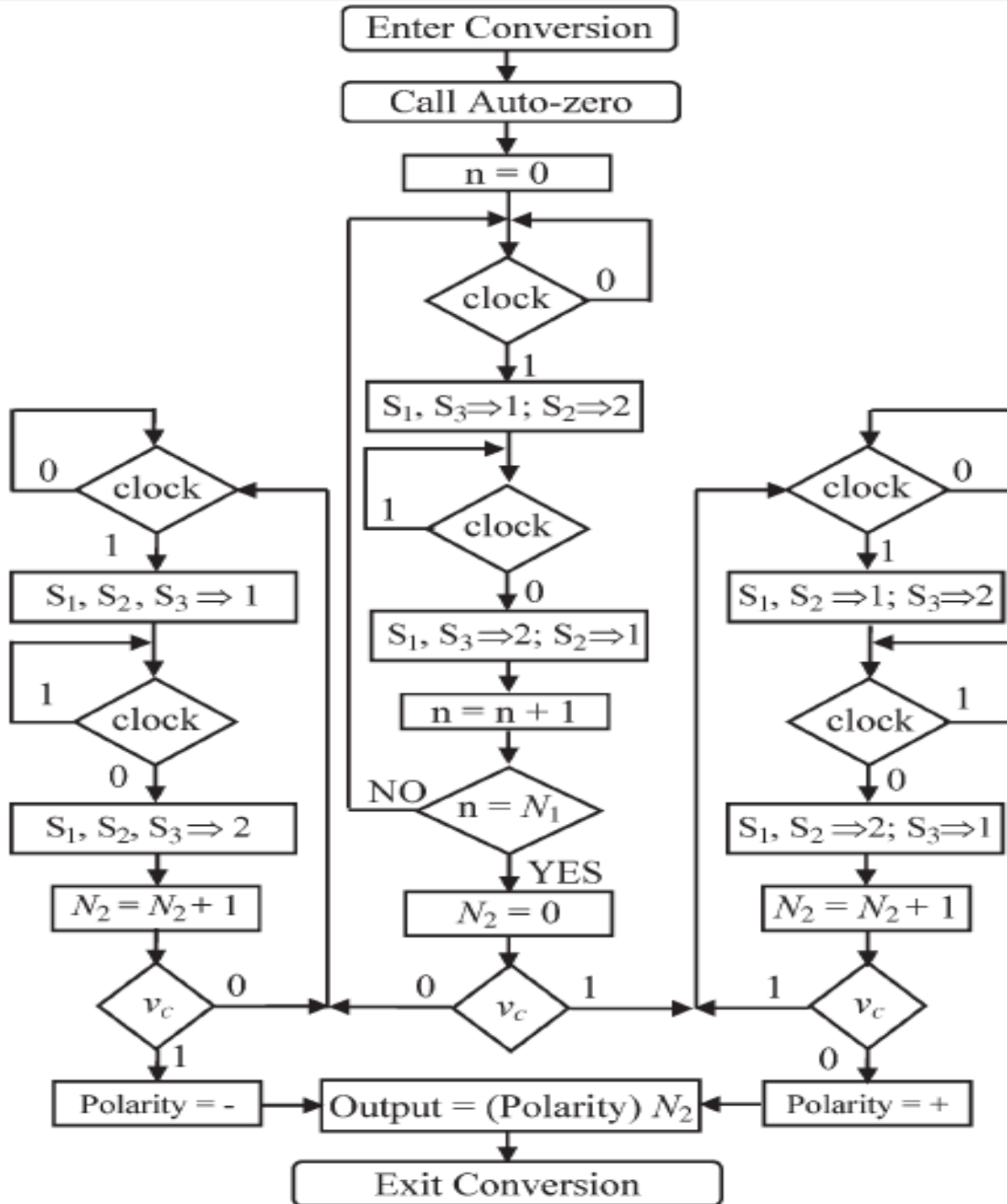


Figure 3.4: Flowchart showing the logic of the conversion phase [1].

When the clock goes low, the sensor capacitances are discharged to ground. This process is repeated for every clock cycle. Hence, the output of the integrator gradually decreases in steps of $VR (C1 + C2)/CF$ and reaches zero. voi reaching zero is indicated to the CLU by a high-to-low transition on vc .

On the other hand, if vc is low at the end of T1 (i.e., $C2 > C1$), then the CLU switches S1, S2, and S3 to position 1 whenever the clock is high and changes all three to position 2 whenever the clock turns low. Thus, when the clock is high, both C1 and C2 will get charged to VR. In addition, when the clock goes low, their net charge $VR (C1 + C2)$ is transferred to CF, discharging it.

Under this condition, voi increases in steps of $VR (C1 + C2)/CF$ and reaches zero, as indicated by the dotted line in Figure 3.2. Here, voi reaching zero is indicated to the CLU by a low-to-high transition on vc . As soon as the transition is sensed, the CLU stops the counter, and the count value, i.e., N2, is taken as the output. A flowchart showing the sequence of operations and switch positions set by the CLU during the periods T1 and T2, depending on the comparator output voltage vc , is shown in Figure 3.4. In either case, the time taken for voi to reach zero, i.e., T2 (vide Figure 3.2), is measured by the CLU and is provided as the final digital output. In all the cases, at the end of period T2, the net charge in CF is zero; hence, we get

$$VR \left(\frac{C1 - C2}{CF} \right) \frac{T1}{TC} = VR \left(\frac{C1 + C2}{CF} \right) \frac{T2}{TC} \quad (3.1)$$

Rearranging the terms, we get

$$T2 = \left(\frac{C1 - C2}{C1 + C2} \right) T1 \quad (3.2)$$

If the number of clock cycles in T1 and the number of clock cycles measured during T2 are denoted as N1 and N2, respectively, then

$$N2 = \left(\frac{C1 - C2}{C1 + C2} \right) N1 \quad (3.3)$$

Substitution of the values for C1 and C2, as given in (2.1), into (3.3) results in the following:

$$kx = \frac{N2}{N1} \quad (3.4)$$

It nicely turns out that substitution of the values for $C1$ and $C2$ for a sensor possessing inverse relationship, as given in (2.2), into (3.3) also results in (3.4). Hence, even if $C1$ and $C2$ are governed by the inverse relationship, as given in (2.2), the digital output $N2$ is linearly related to x , as given in (3.4). The polarity of x is taken as positive if $v_c = 1$ at the end of $T1$; otherwise, the polarity is treated as negative, as shown in Figure 3.4. Since $N1$ is a preset value and k is the sensor's transformation constant, the digital value $N2$ directly represents the quantity x being sensed by the sensor. Thus, the technique presented here implements a linear direct CDC applicable for not only differential capacitive sensors possessing linear characteristics but also sensors possessing inverse characteristics without any change.

The triple-slope method [12] takes a total time period of $3N1TC$ for a full-scale conversion. Compared with that method, in the present method, the total conversion time is $(T1 + T2)$, which works out to be $2N1TC$ for full scale, thus providing 33% improvement in conversion speed. In the triple-slope CDC, two separate integration periods are employed to get information proportional to $(C1 - C2)$ and a third integration period to obtain an output proportional to $(C1 + C2)$. In the DSCDC, the information proportional to $(C1 - C2)$ is achieved in a single integration period, reducing the overall number of integration periods from three to two. As the technique used for developing the CDC is based on the well-known dual-slope integrating type ADC principle [13], the DSCDC possesses features such as high repeatability in the output, even in the presence of high-frequency noise, very small nonlinearity errors, and very good rejection of interfering frequencies with periods of integral multiples of the measurement period. Two opposing criteria, i.e., fast conversion and noise suppression, dictate the selection of $T1$. $T1$ should be as small as possible for fast conversion, whereas for suppression of noise and interference, it should be as large as possible [14]. Typically, with this method, a conversion speed of a few conversions to a few hundred conversions per second is obtainable.

CHAPTER 4

SIMULATION ANALYSIS

4.1 Simulation Tool-LTSpice IV

LTSpice IV is a schematic-driven circuit simulation program. The LTSpice simulator was originally based years ago on Berkeley SPICE 3F4/5. The simulator has gone through a complete re-write in order to improve the performance of the simulator, fix bugs, and extend the simulator so that it can run industry standard semiconductor and behavioral models. A digital simulation capability, including co-simulation, has been added. Extensive enhancements have been made to the analog SPICE simulator, such as parallel processing and dynamic assembly and object code generation in the SPARSE matrix solver to make LTSpice IV the industry superlative analog simulator [15].

Many Linear Technology products are modeled with proprietary building blocks and/or proprietary hardware description languages that accurately encapsulate realistic behavior with custom macro-models. This allows a SMPS to be prototyped rapidly via simulation.

LTSpice can be used as a general-purpose SPICE simulator. New circuits can be drafted with the built-in schematic capture. Simulation commands and parameters are placed as text on the schematic using established SPICE syntax. Waveforms of circuit nodes and device currents can be plotted by clicking the mouse on the nodes in the schematic during or after simulation.

4.1.1 Main differences between LTSpice and Spice

The company Linear Technology extended Spice 3F4/5 with algorithms to simulate advanced analogue functions and released this under the name LTSpice. These algorithms are beneficially used as intended to simulate switching power supply in comparison with Spice [16]. The difficulty when simulating switching power supply with Spice is where there are combinations between high frequency square waves and low frequency overall loop response [16].

LTSpice uses integrated logic primitives to simulate the high frequency square waves that Spice also does not handle. When LTSpice uses logic in the simulations the result becomes detailed even if the response time is short. LTSpice has by this way reduced the number of calculations which resulted in a faster response time [16].

LTSpice also contains a mixed-mode compiler which does not exist in Spice. The mixed-mode compiler gives realistic models of circuits when a multiple mode device is modelled. LTSpice handles, in a realistic way, to model the devices in switch mode, burst-mode and cycle skipping mode [16].

4.1.2 Differences between PSpice and LTSpice

The main difference between PSpice and LTSpice is that Cadence has extended the Spice 2G.6 to PSpice where as LTSpice is an extended Spice 3F4/5 tool. In contrast to LTSpice, which is free of charge, the PSpice tool is license charged and in addition the Advanced Analysis extension tool cost more money.

PSpice has 11,300 models compared with LTSpice which has only 1,500 models. Often are the device manufacturers releasing models of their devices in PSpice format and not in LTSpice format.

PSpice has extended the advanced analyses, as well as LTSpice has, compared to Spice. But PSpice has extended the LTSpice's analyses as well. The PSpice Advanced Analysis tool helps the engineers to improve yield and reliability of their designs. PSpice Advanced Analysis is integrated with MathWorks Matlab Simulink and a co-simulation is provided.

4.1.3 Installing the LTSpice tool

To install LTSpice the installer needs to be an administrator on the local computer. The LTSpice tool files are stored in the folder “%programfiles%\LTC\LTspiceIV\”LTC/LTspiceIV.

LTSpice uses an internal updater to update the tool files into the latest revision level [16]. The updater only updates the tool files. The component libraries are merged because of the ability for the user to do updates or add components into the library files. The LTSpice license is free of charge and can be distributed as it is [16].

4.1.4 Simulation Functions, Capabilities and Limitations

The schematic tool included in LTSpice is converting the schematics into a text Spice netlist. The text netlist is used during the analysis. LTSpice saves the netlist in .asc format and reads schematics from .asc, .cir, .net and .sp formats. The netlist can also be exported to an ASCII file where formats as Accel, Algorex, Allegro, Applicon Bravo, Applicon Leap, Cadnetix, Calay, Calay90, CBDS, Computervision, EE Designer, ExpressPCB, Intergraph, Mentor, Multiwire, PADS, Scicards, Tango, Telesis, Vectron and Wire List [16] can be used. The limitation of exporting the netlist is when the tools do not use the same symbols. To avoid scrambled connections where for example the diodes are connected in the wrong direction synchronization between the tools need to be done. The symbols are saved and read in .asy format.

LTSpice uses the advantage of hierarchical schematic resulting in that a large circuit fits one sheet [16]. Another advantage of hierarchy is when the tool can reuse parts of the design within the circuits. A disadvantage with LTSpice is that the tool does not simulate effects from the PCB without the engineer adding the wanted parasitic elements. Another disadvantage is that LTSpice does not simulate other domains than the electronic and therefore cannot be used when a mechatronic system needs to be simulated.

LTSpice uses Spice models and does not simulate the models in different ways. Therefore such a function cannot help the engineer make sure that the models are simulating the devices correct. As a result of this the simulating engineer needs, when using LTSpice, critical examine, by using his or hers experience, if the result of the simulation can be trusted or not.

LTSpice has a feature to simulate wave files in the designs. For example if an amplifier is to be simulated the simulation tool can use an audio file and the audio file can be used when the device is to be tuned. This is an advantage compared with Spice.

Disadvantages with LTSpice are that in the schematic creator the standard windows keys are not used making the tool impractical to use. It is also not possible to copy symbols between two sheets making it time demanding.

4.1.5 Component Libraries

The only way to use component library files in LTSpice is to copy the library files (.asy) into the LTSpice folder. Therefore it is not easy to share library files among the computers which are used to simulate hardware. The advantage of all users having the same library is also missed as well as that all libraries are up to date.

4.1.6 Models

The installation of LTSpice includes 1,500 models. There are also possibilities to add two types of models. The first type of model is the basic type such as transistors, diodes and MOSFETs. They are described as ordinary Spice models. The basic models are either imported into the library or extended as text in the schematic.

The second type of model is subcircuit models. The subcircuit model is used when more than the ordinary model is needed. Examples are additional lead inductors and capacitors or more complex subcircuits like operational amplifier models.

LTSpice can simulate hardware with several different types of models as it is based on Spice and simulates all Spice models used by Spice 3F.5. LTSpice has included integrated logic primitives with Spice to perform faster response and still have the resolution when simulating switching power supply simulations [16].

PScope is a device to analyze and evaluate the components and translate signals into parameters of a model. PScope is a USB-based board equipped with several analog to digital converters. PScope collects data from the analog to digital converters and analyses it both in the time and frequency domains.

4.1.7 Graphical Viewer

To run a simulation in LTSpice, either enter the wanted analysis as text called dot commands in LTSpice or let the program place the analysis text by the command 'Simulate/Edit Simulate Command'. The last option is illustrated by Figure 4.1. The text reads '.tran .03' and it means

that the tool will perform a Transient analysis. The analysis will last 30 ms.

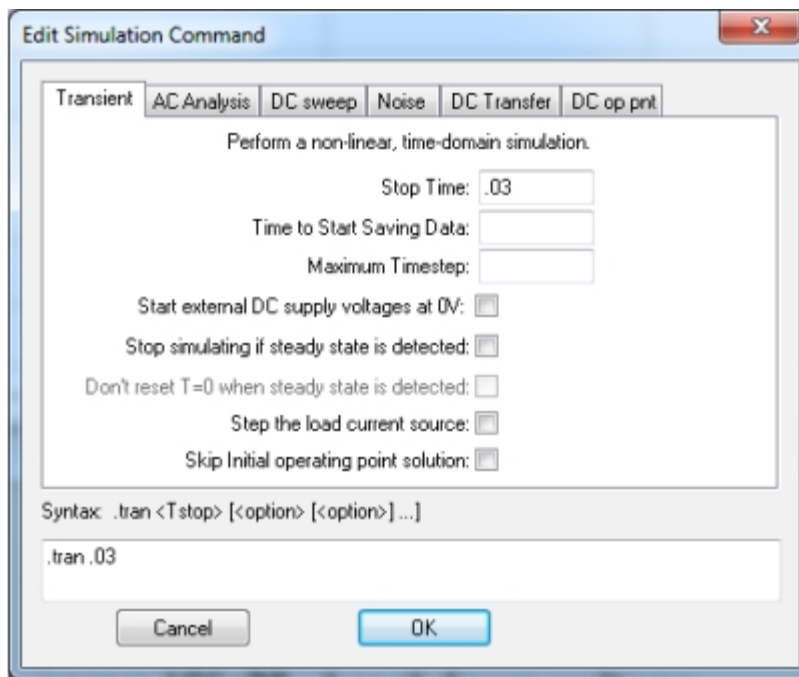


Figure 4.1: The LTSpice 'Edit Simulation Command'.

4.2 Simulation Model

The signal conditioning circuit, seen in Figure 3.1, is simulated in LTSpice. Figure 4.2 portrays a screenshot of the circuit in LTSpice. On the right side under the headline Control Unit, are circuits used for controlling the switches according to the logic explained in Figure 3.4 and on the right side under the headline Parameter are the capacitor values of the sensor which are varying in step of 10° angular movement of middle plate of the sensor from 1° to 179° .

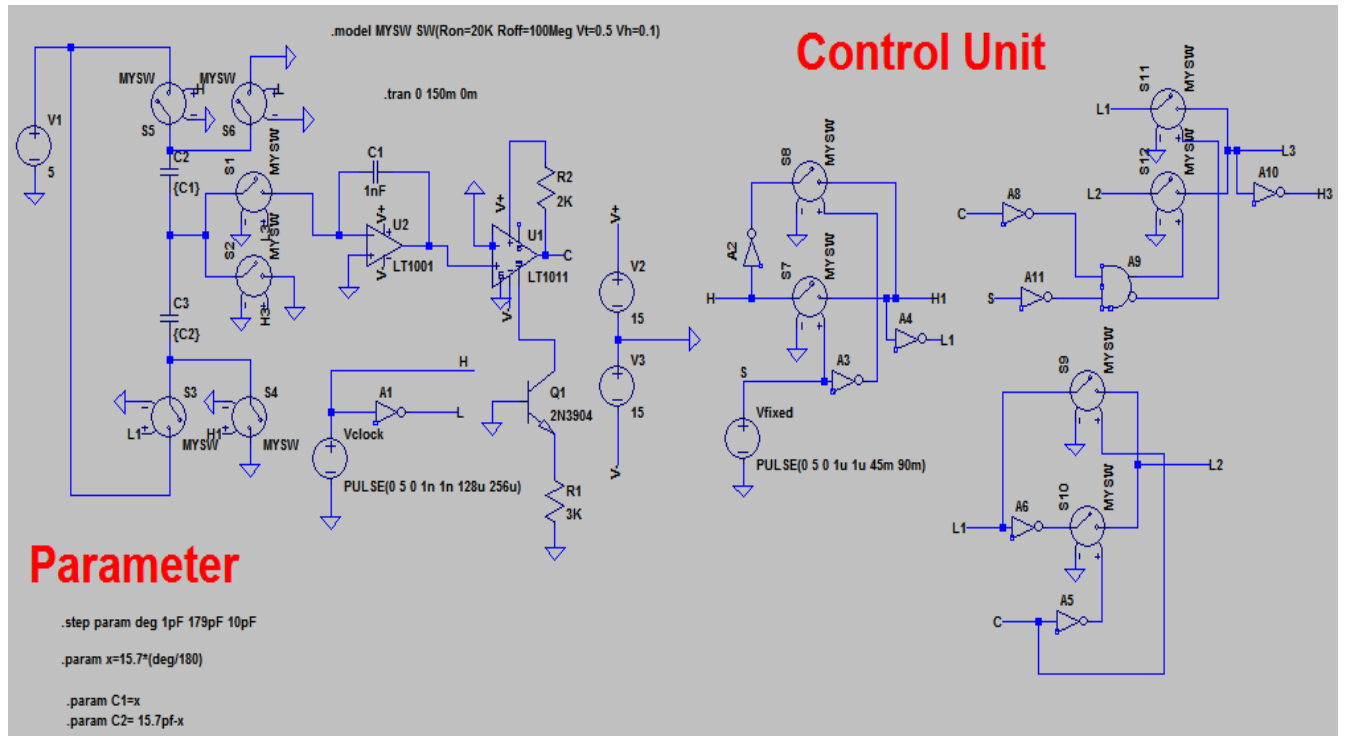


Figure 4.2: Simulation circuit in LTSpiceIV.

The VR used in simulation model is 5V. Two pulse signals which are used to control the switches have the following properties:

	Function	Vinitial[V]	Von[V]	Tdelay[s]	Trise[s]	Tfall[s]	Ton[s]	Tperiod[s]
Vclock	PULSE	0	5	0	1n	1n	128u	256u
Vfixed	PULSE	0	5	0	1u	1u	45m	45m

Table.4.1: Pulse signals used in simulations.

Vclock is used for controlling the switches. When Vclock is high all three SPDT switches are set to a particular switching position and when this pulse turns low all the switch positions will be toggled. Vfixed will remain high for T1 time which is fixed for all the simulations and for this time switches will toggle within the set of particular switching positions. When Vfixed turns low control unit will sense the output of the comparator and set the switching position and toggle them accordingly for time T2 which depends on the difference between C1 and C2.

For the working of the switches, the model that I had used is:

.model MYSW SW(Ron=20K Roff=100Meg Vt=0.5 Vh=0.1)

Figure 4.3 shows the screenshots of switch control signals generated using the control unit according to the logic explained in Figure 3.4. The green waveform is control signal for switch1, blue waveform is control signal for switch2 and red waveform is control signal for switch3.

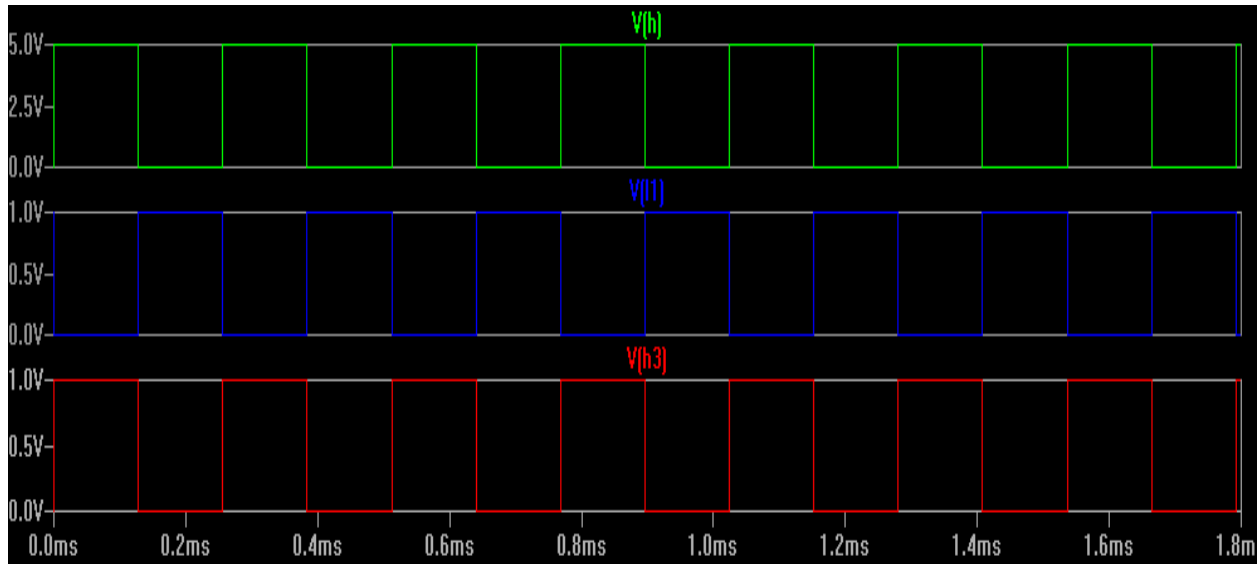


Figure 4.3(a): Waveform for controlling switches during time T1.

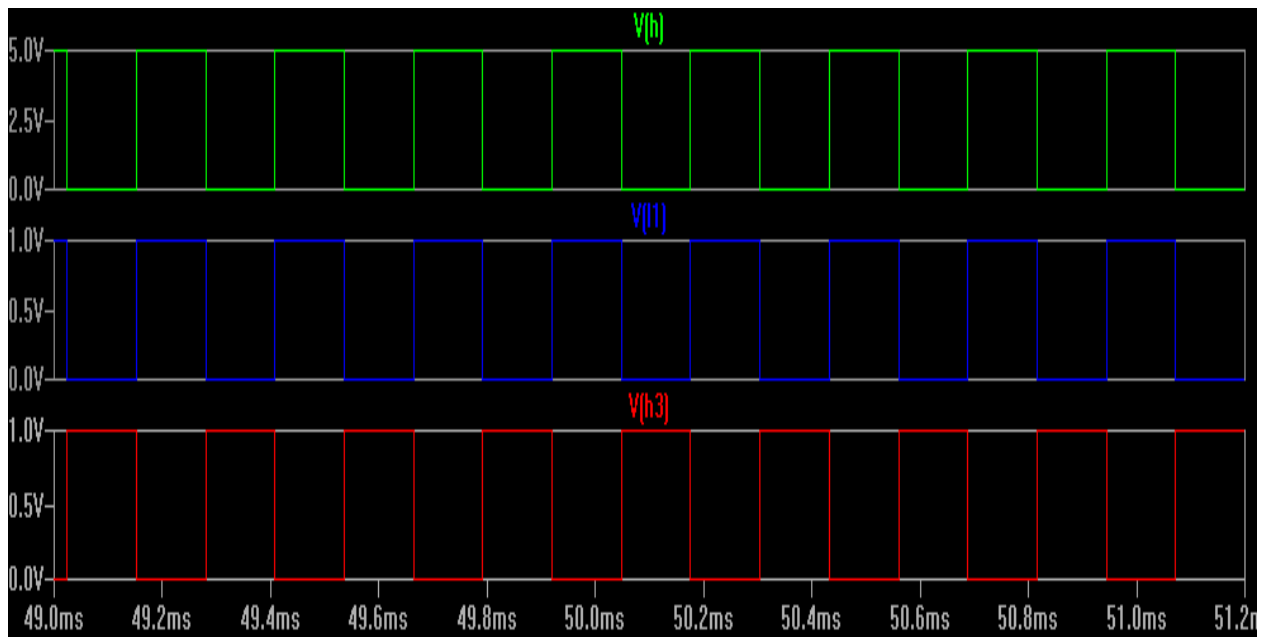


Figure 4.3(b): Waveform for controlling switches during time T2($C1 > C2$).

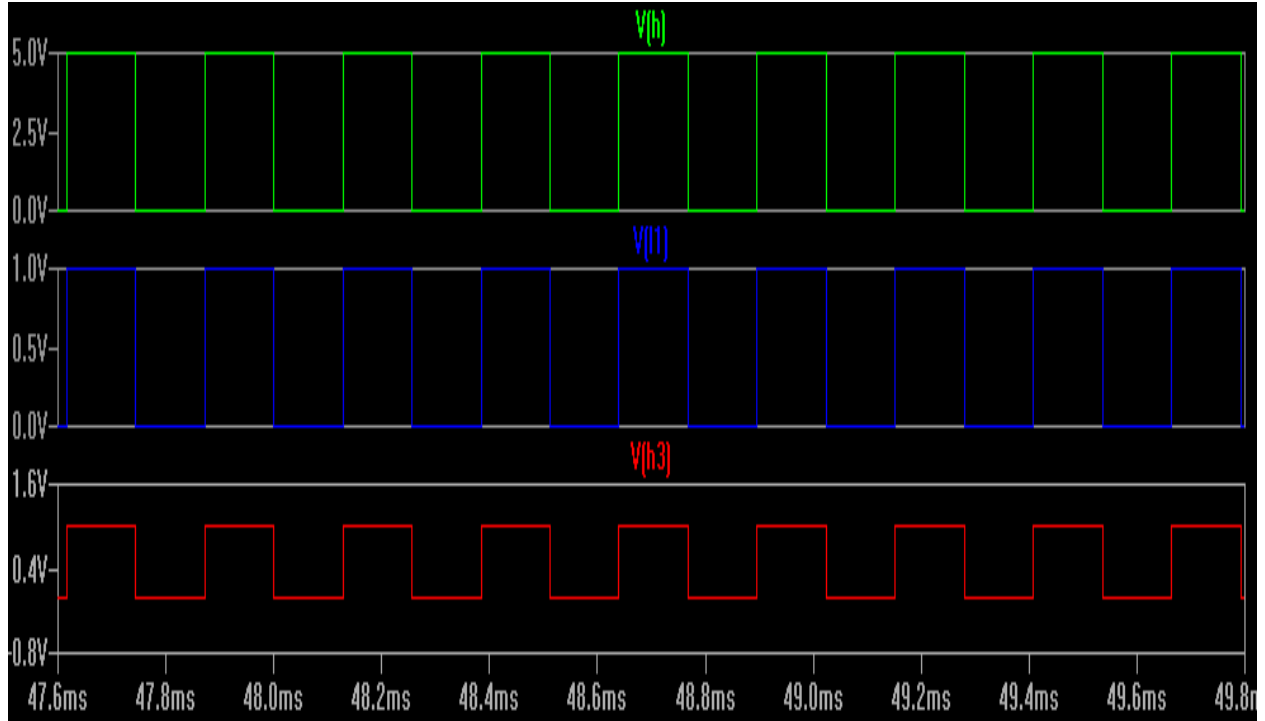


Figure 4.3(c): Waveform for controlling switches during time T_2 ($C_1 < C_2$).

Figure 4.3(a) shows the switch control signals for all three switches during the fixed time T_1 . Figure 4.3(b) shows the switch control signals for all three switches during time T_2 when $C_1 > C_2$. Figure 4.3(c) shows the switch control signals for all three switches during time T_2 when $C_1 < C_2$.

4.3 Simulation Results

In order to find the angle of the angle sensor the output of the integrator is sensed which have dual slope characteristics as shown in Figure 4.4. The figure shows the integrator output for different values of C_1 and C_2 which are varying in step of 10 degrees. There are total 19 simulations in Figure 4.4 with angle varying from 1° to 179° in steps of 10° . Transient analysis was done from time period 0 to 90 ms.

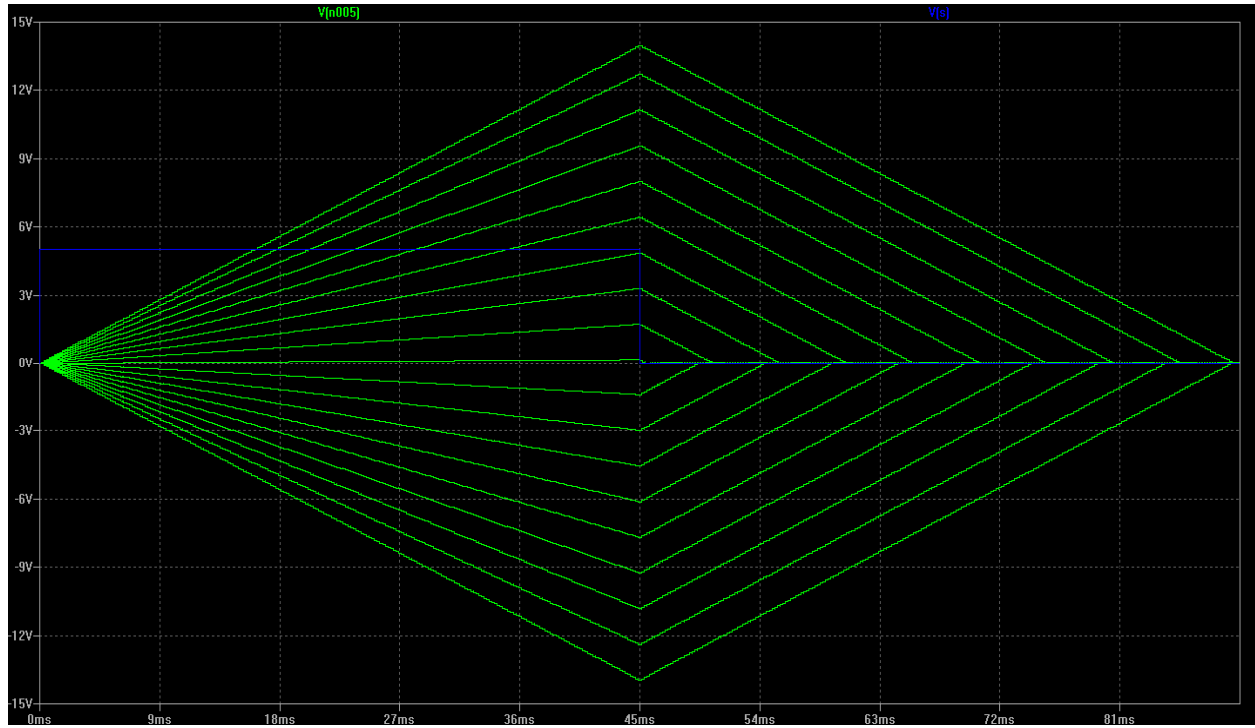


Figure 4.4: Integrator outputs in LTspiceIV.

The time period for first slope of the integrator output, T_1 , is fixed whereas the time period for the second slope of integrator output, T_2 , is varying with angular movement of the sensor which is to be measured. Table 4.2 shows the values of T_2/T_1 for corresponding angular movement of the sensor.

S.No.	Degree(θ)	$\left(\frac{C1 - C2}{C1 + C2}\right) = \frac{T2}{T1}$
1	1	-.99
2	11	-.88
3	21	-.77
4	31	-.66
5	41	-.54
6	51	-.43
7	61	-.32
8	71	-.21

9	81	-.10
10	91	.01
11	101	.12
12	111	.23
13	121	.35
14	131	.46
15	141	.57
16	151	.68
17	161	.79
18	171	.90
19	179	.99

Table 4.2: Value of T2/T1 at different angular positions.

The plot shown in Figure 4.5 represents the linear characteristics of T2/T1 w.r.t. angular position (θ) of the sensor. The plot is deduced from Table 4.2.

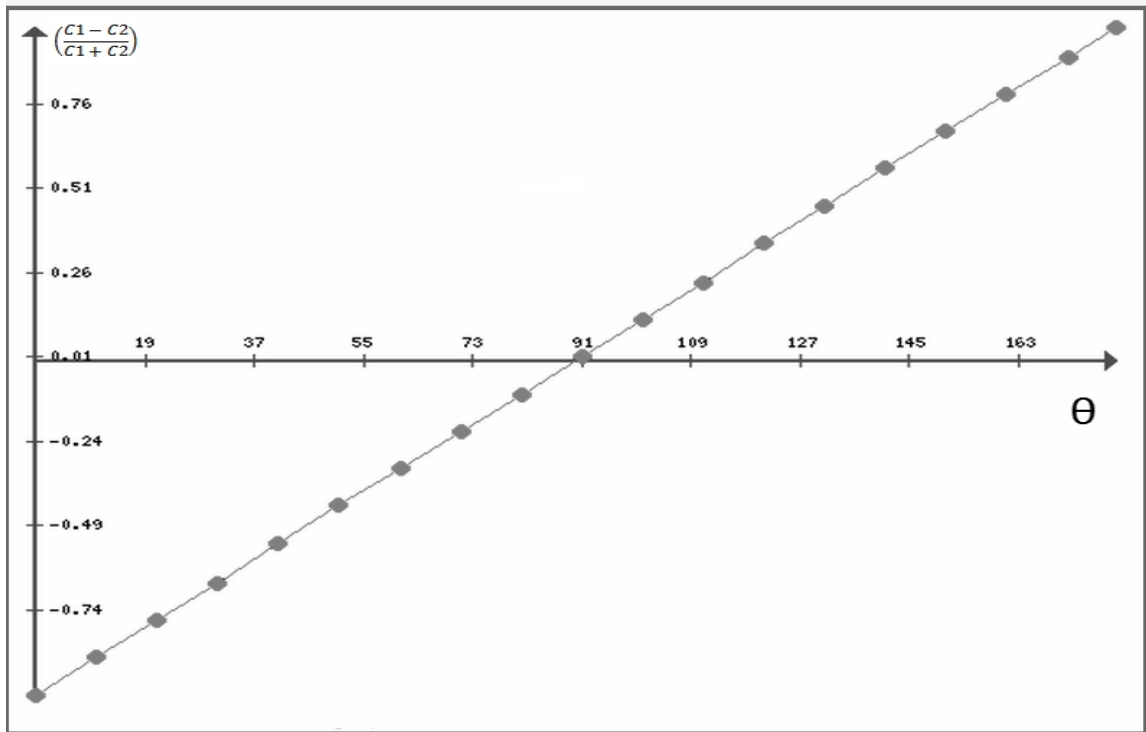


Figure 4.5: Linear variation of T2/T1 w.r.t θ .

CHAPTER 5

EXPERIMENTAL SETUP AND RESULTS

5.1 Methodology

The methodology used for the angle sensing is represented in Figure 5.1. The angle sensor in the first block was fabricated as explained in section 2.3. ATmega 328 microcontroller along with Arduino UNO was used as the control unit for controlling the switches. The signal conditioning circuit was built on NI Elvis II.

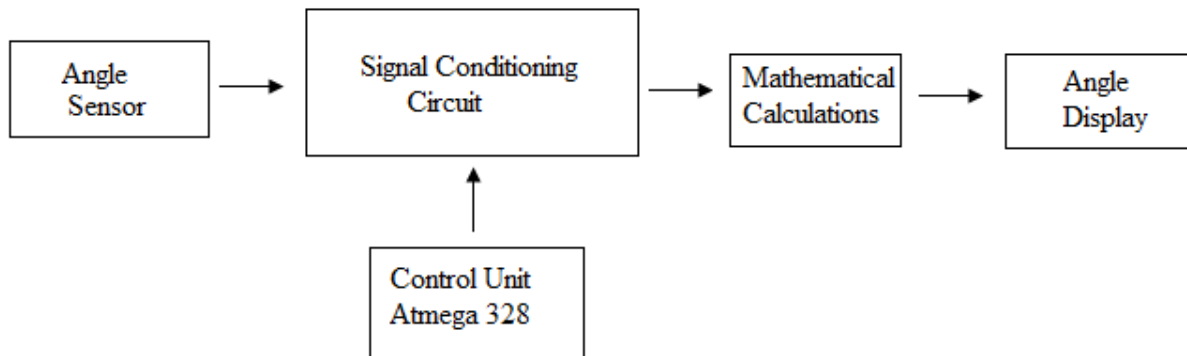


Figure.5.1: Block Diagram.

5.2 Arduino Platform

Arduino is an open source electronics prototyping platform based on flexible hardware and software. The Arduino is a simple yet sophisticated device which is based on Atmel's ATmega microcontrollers. The Arduino software is supported by Windows, Macin-tosh OSX and Linux operating systems despite the fact that most microcontrollers are limited to Windows operating system. The software language is based on AVR C programming language and can be expanded through C++ libraries. There are various types of Arduino microcontroller board available in the market including the Arduino kits and Arduino shields [17].

5.2.1 Arduino Uno Board

Arduino Uno is one of the microcontroller boards manufactured by the Arduino and it is a microcontroller board based on the Atmel's ATmega328 microcontroller. "Uno" means one in Italian and the uno board is the latest in a series of USB (Universal Serial Bus) Arduino boards which is the reference model for the Arduino platform. The Arduino Uno board has a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, a reset button, 6 analog inputs and 14 digital input/output pins (of which 6 can be used as PWM outputs). It uses the Atmega16U2 programmed as a USB-to-serial converter instead of FTDI USB-to-serial driver chip which was used in all the pre-ceding boards. The board has 32 KB flash memory of which 0.5 KB is used by boot-loader, 2 KB of SRAM, 1 KB of EEPROM and 16 MHz clock speed [18].

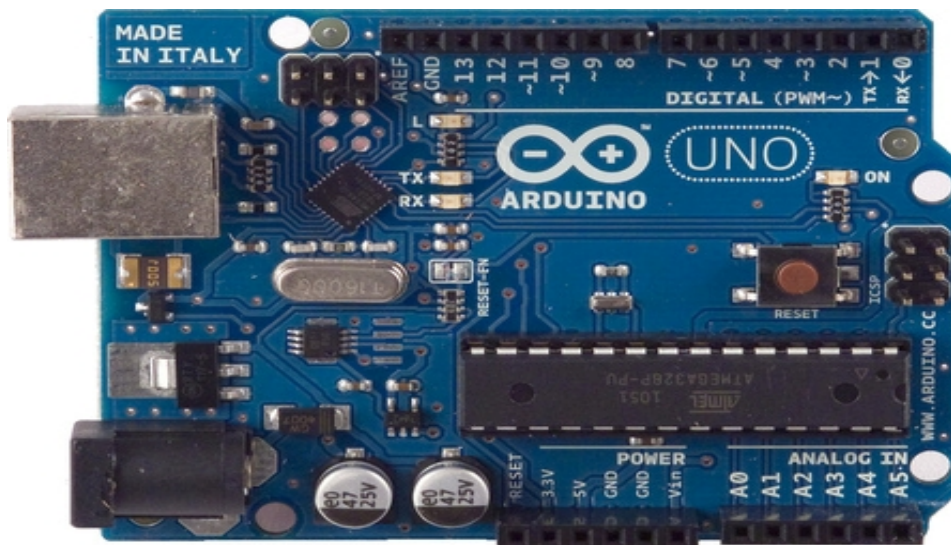


Figure 5.2: The Arduino Uno Board [18].

Figure 5.2 shows the Arduino Uno Board manufactured by the Arduino in Italy. It can be powered via a USB connection or with an external power supply. As can be seen in Figure 5.2, pins A0 to A5 are the analog input pins, pins 0 to 13 are 14 digital input/output pins and the pins with a "~" sign can be used as the PWM output pins. The digital pins can be used as input or output pins by selecting the mode by using the function `pin-mode()` and then using the function `digitalRead()` or `digitalWrite()` according to the necessity. Pins 0(RX) and 1(TX) are used for serial communication while pins 10(SS), 11(MOSI), 12(MISO) and 13(SCK) are used for SPI

(Serial Peripheral Interface) communication. In addition to pin 0 and 1, a Software Serial library allows serial communication on any of the Uno's digital pins [18].

5.2.2 ATmega328 Microcontroller

The microcontroller is a low-power CMOS (Complementary Metal Oxide Semiconductor) 8-bit microcontroller based on the AVR enhanced RISC (Reduced Instruction Set Computer) architecture. The powerful execution of instructions in a single clock cycle leads to the achievement of 1 MIPS per MHz throughputs allowing the designer to optimize power consumption versus processing speed [19].

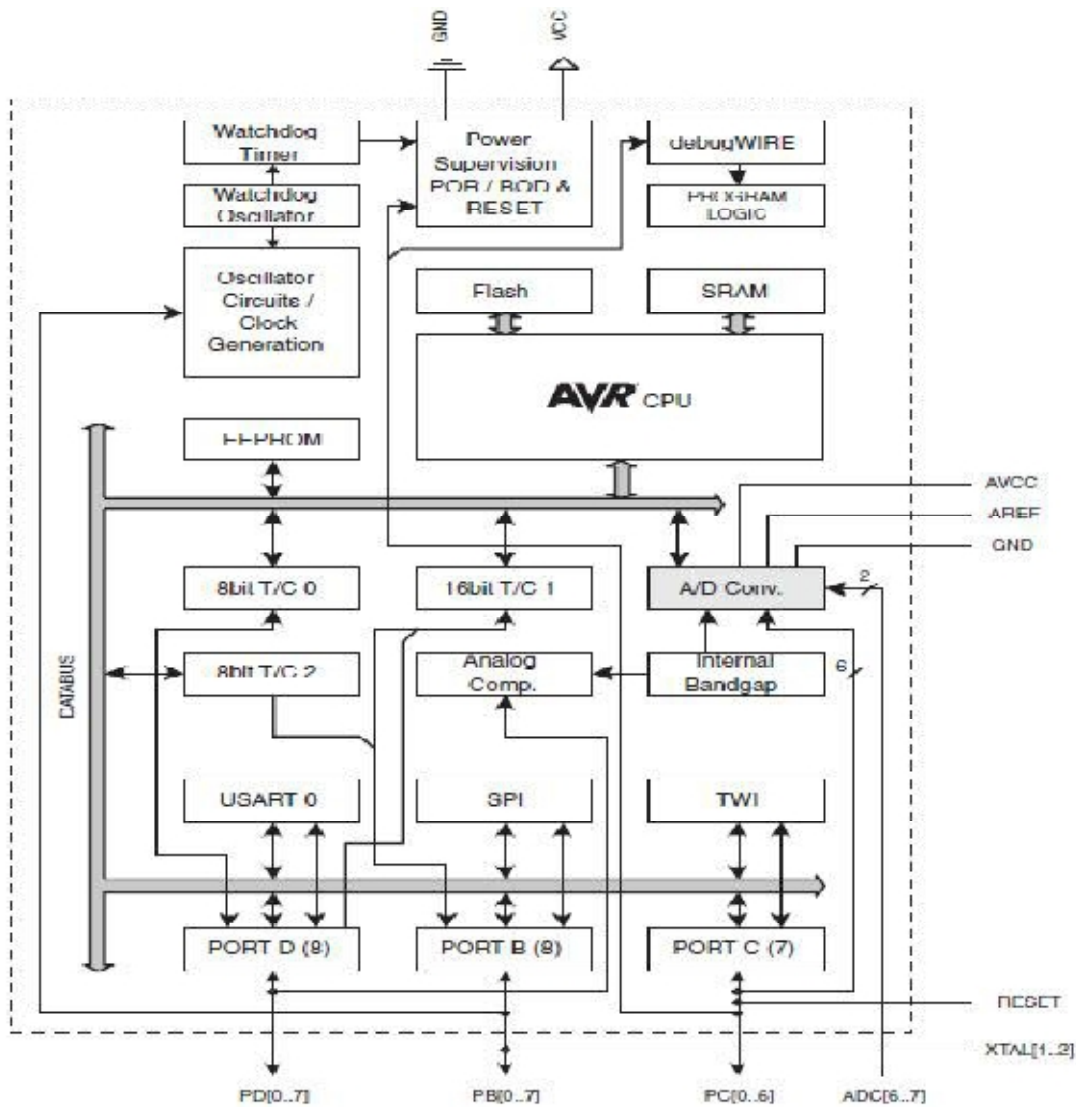


Figure 5.3: ATmega328 Microcontroller Architecture [19].

The internal architecture of the microcontroller is shown in Figure 5.3. The central processing unit (CPU) is the brain of the microcontroller which controls the execution of the program. The MCU (Microcontroller unit) consists of 4K/8K bytes of in-system programmable flash with read-while-write capabilities, 256/412/1K bytes EEPROM along with the 512/1K/2K bytes of SRAM. Along with this, the MCU consists of many other features [19]:

- 23 general purpose I/O lines and 32 general purpose working registers
- 3 flexible timer/counters with compare modes, internal and external interrupts and a serial programmable USART
- A byte-oriented 2-wire serial interface, an SPI serial port ,a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable watch-dog timer with an internal oscillator and 5 software-selectable power saving modes [19].

The five, software selectable, power saving modes are Idle mode, Power-down mode, Power-save mode, ADC Noise Reduction mode and the Standby mode. As mentioned in earlier, the CPU is the brain of the microcontroller which controls the execution of the program. Therefore the CPU is able to access the memories, perform calculations, control peripherals and handle interrupts. The AVR uses the Harvard architecture with separate memories and buses for program and data to maximize the performance as well as the parallelism. The principle of execution of instructions in the program memory is the single-level pipelining. The concept of pre-fetching the next instruction while executing one instruction enables the instructions to be executed in every clock cycle and the program memory is in the System Reprogrammable Flash memory [19].

The block diagram of AVR CPU Core architecture is shown in Figure 5.4. The fast-access Register File contains 32 x 8 bit general-purpose working registers with a single cycle access time which results in a single-cycle ALU operation. The arithmetic and logical operations between the registers or between the constant and a register are supported by the ALU. The status register is updated to reflect information about the result of the operation after an arithmetic operation [19].

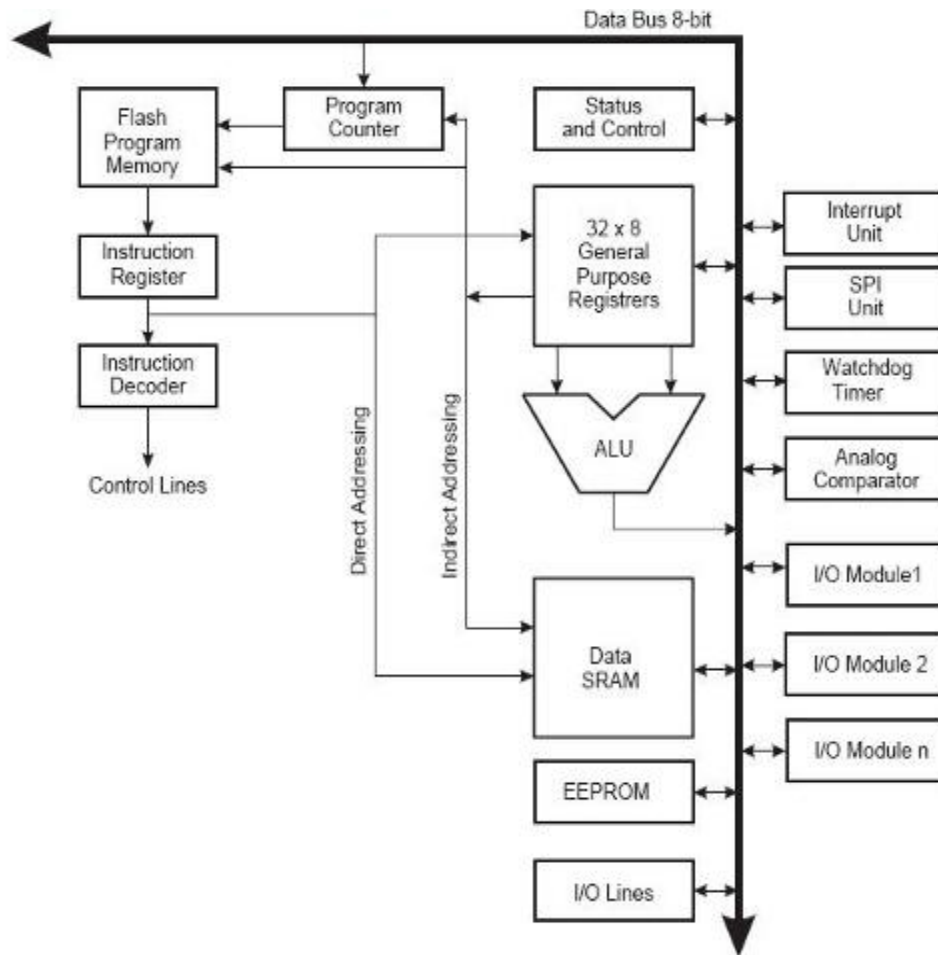


Figure 5.4: Block diagram of the AVR CPU Core architecture [19].

The boot program section and the application program section are the two main sections of the program flash memory. Stack stores the return address of the program counter during the interrupts and subroutine calls which is allocated in the general data SRAM. The size of the stack is limited by the total size and usage of the SRAM. The data SRAM is accessible through five different addressing modes supported in the AVR architecture while the stack pointer is read/write accessible in the I/O space. The memory spaces in the AVR architecture are all linear and regular memory maps [19].

5.3 NI ELVIS II

The National Instruments Educational Laboratory Virtual Instrumentation Suite II (NI ELVIS II) is a LabVIEW and computer based design and prototyping environment. NI ELVIS II consists of

accustom-designed bench top workstation, a prototyping board, a multifunction data acquisition device, and LabVIEW based virtual instruments [20]. This combination provides an integrated, modular instrumentation platform that has comparable functionality to the DMM, Oscilloscope, Function Generator, and power Supply found on the laboratory workbench.

The NI ELVIS II Workstation can be controlled either vi manual dials on the stations front or through software virtual instruments. The NI ELVIS II software suite contains virtual instruments that enable the NI ELVIS II workstation to perform functions similar to a number of much more expensive instruments.

One can use NI ELVIS II in engineering, physical sciences, and biological sciences laboratories. The suite offers full testing, measurement, and data logging capabilities .The environment consists of the following two components [20]:

1. Bench top hardware workspace for building circuits, shown in Figure 5.5.
2. NI Elvis software interface consisting of twelve soft front panels (SFP) instrument, Figure 5.6.

The soft panels are:

- Digital Multimeter (DMM)
- Oscilloscope (Scope)
- Function Generator (FGEN)
- Variable Power Supply (VPS)
- Bode Analyzer
- Dynamic Signal Analyzer (DSA)
- Arbitrary Waveform Generator (ARB)
- Digital Reader (DigIn)
- Digital Writer (DigOut)
- Impedance Analyzer

- Two –wire Current-Voltage Analyzer
- Three –wire Current-Voltage Analyzer

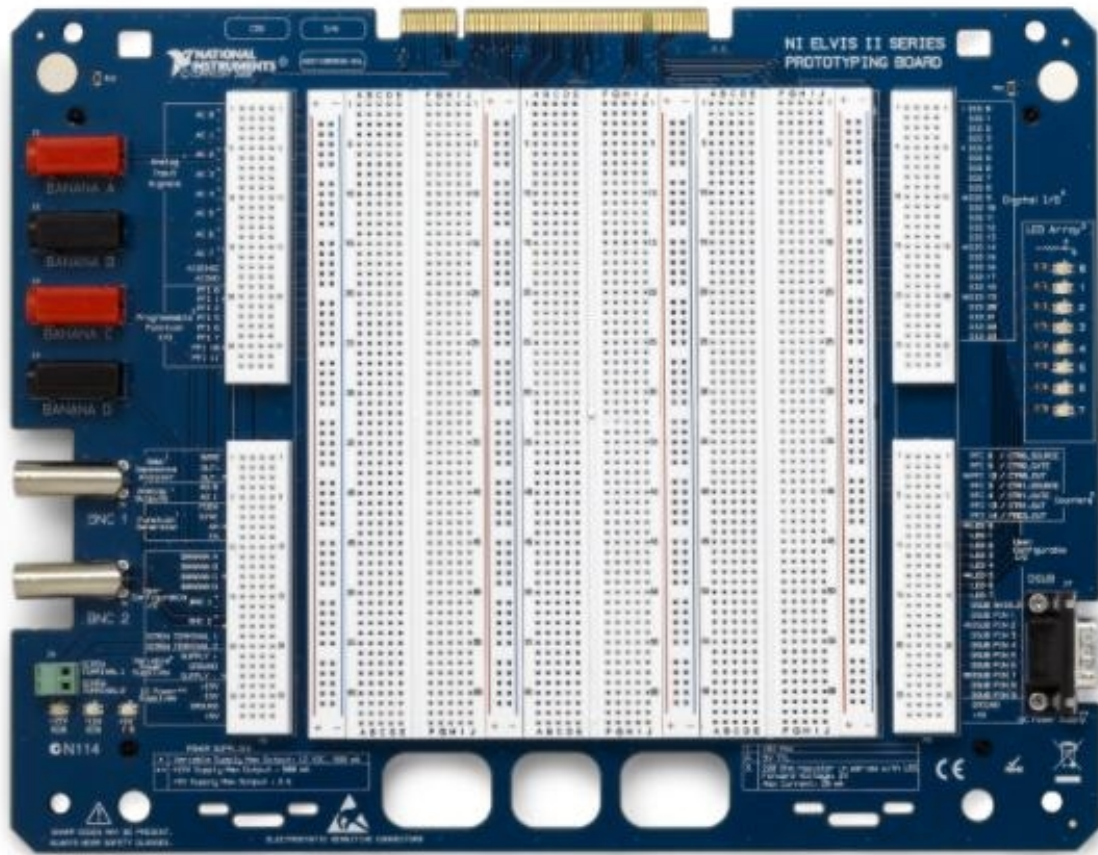


Figure 5.5: NI ELVIS II hardware [21].

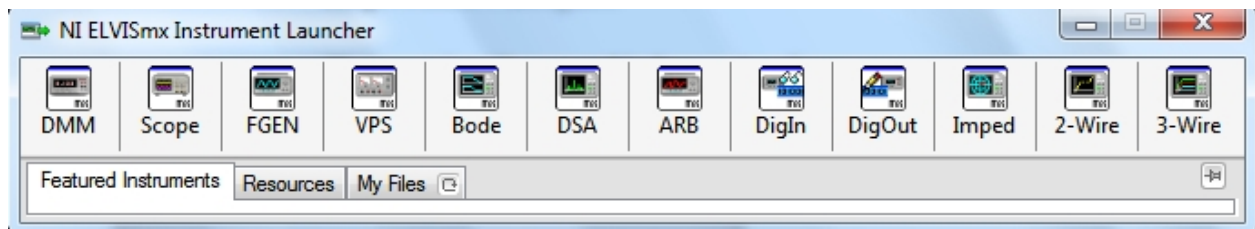


Figure 5.6: NI ELVIS II Soft Panel.

5.3.1 Applications

NI ELVIS II SFP instruments, such as the Bode Analyzer and Dynamic Signal Analyzer, offer instructors an opportunity to teach advanced courses in signal analysis and processing.

Students can learn sensor and transducer measurements, in addition to basic circuit design by building custom signal conditioning. They can install custom sensor adapters on the prototyping board. For example, installing a thermocouple jack on the prototyping board allows robust thermocouple connections. The programmable power supply can provide excitation for strain gauges use in strain measurement.

Physics students typically learn electronics and circuit design theory. NI ELVIS II provides these students with the opportunity to implement these concepts. For example, physics students can use NI ELVIS II to build signal conditioning circuits for common sensors such as photoelectric multipliers or light detector sensors.

5.3.2 NI ELVIS II Bench top Workstation

NI ELVIS II hardware contains Bench top Workstation and Series Prototyping Board. The workstation control panel provides easy-to-operate knobs for the variable power supplies and function generator, Figure 5.5, and offers convenient connectivity and functionality in the form of BNC and banana-style connectors, shown in Figure 5.5, to the function generator, scope, and DMM instruments at the right side of the bench top.

5.3.3 NI ELVIS II Series Prototyping Board

This section describes the NI ELVIS II Series Prototyping Board and how to use it to connect circuits to NI ELVIS II. The NI ELVIS II Series Prototyping Board connects to the bench top workstation. The prototyping board provides an area for building electronic circuitry and has the necessary connections to access signals for common applications. Figure 5.7 shows the prototyping board with a brief description. You can use multiple prototyping boards interchangeably with the NI ELVIS II Bench top workstation, removing it from the bench top workstation. You can use the prototyping board connector to install custom prototype boards you develop. This connector is mechanically the same as a standard PCI connector.

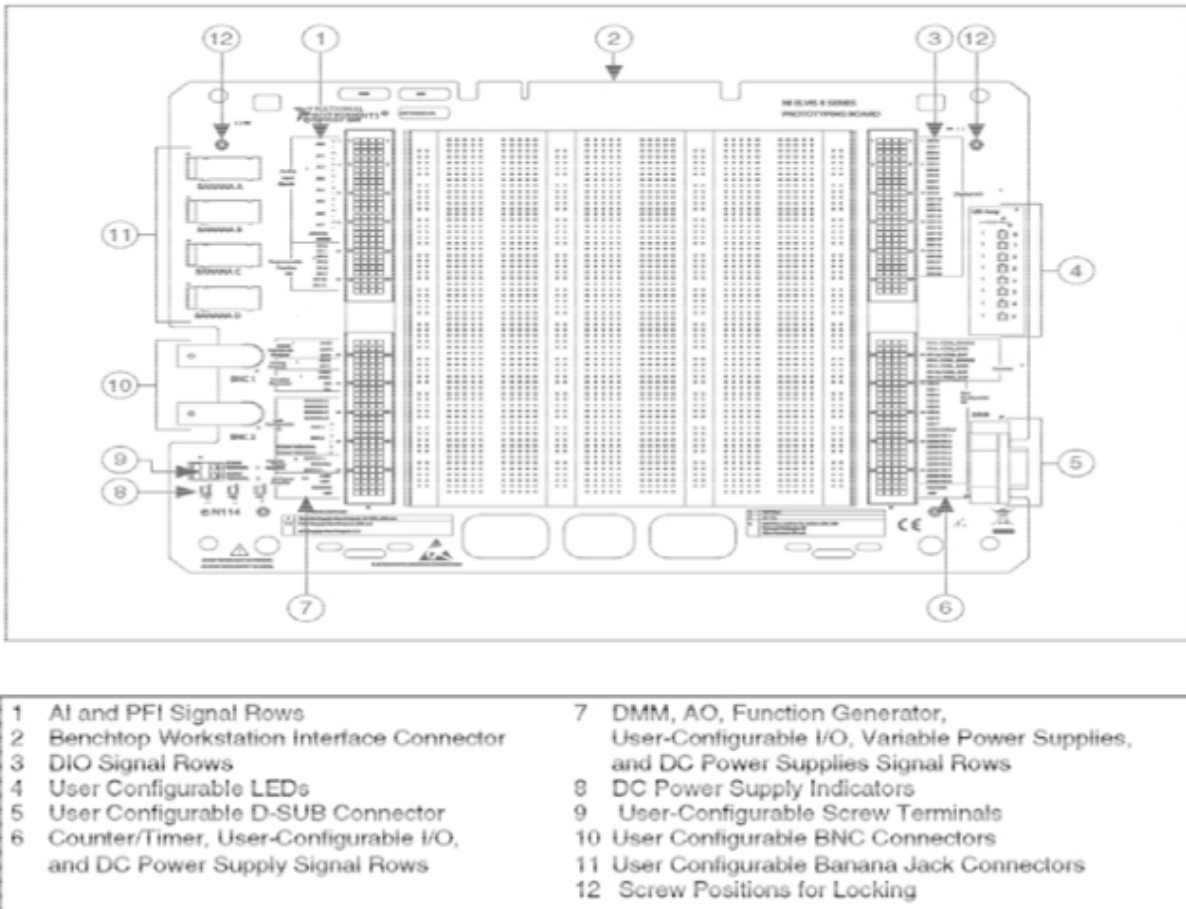


Figure 5.7: Prototyping Board Description [21].

5.3.4 NI ELVIS Functions

NI ELVIS II performs functions similar to a number of real instruments, which are used in common labs. ELVIS hardware and software integrated to gather to serve multi function as described below.

DMM

The primary DMM instrument on NI ELVIS II is isolated and its terminals are the three banana jacks on the side of the bench top workstation. For DC Voltage, AC and COM voltage, Resistance, Diode, and Continuity Test modes, use the V connectors. For DC Current and AC Current modes, use the A and COM connectors [20]. For easy access to circuits on the

prototyping board, you can use banana-to-banana cables to wrap the signals from the user-configurable banana jacks to the DMM connectors on the bench top workstation

Oscilloscope

The two oscilloscope channels are available at BNC connectors on the side of the input impedance and can bench top workstation. These channels have robust 1 M be used with 1X / 10X attenuated probes. You can also use high-impedance Analog Input channels AI to AI7 available on the prototyping board.

Function Generator (FGEN)

The function generator output can be routed to either the FGEN/TRIG BNC connector or the FGEN terminal on the prototyping board. A +5 V digital signal is available at the SYNC terminal. The AM and FM terminals provide analog inputs for the amplitude and frequency modulation of the function generator output [20].

Power Supplies

The DC power supplies provide fixed output of +15 V, -15 V, and +5 V. The variable power supplies provide adjustable output voltages from 0 to +12 V on the supply+ terminal, and 0 to -12 V on the supply- terminal. All power supplies on NI ELVIS II are referenced to ground [20].

Bode Analyzer

The Bode Analyzer uses the Function Generator to output a stimulus and then uses analog input channels AI 0 and AI 1 to measure the response and stimulus respectively.

5.4 Circuit Testing

To check the practicality of the proposed transducer, a prototype unit was built on the NI Elvis board and tested. Precision reference voltage V_R was used from 5V DC power supply of the board. HEF4053 IC was used to realize switches S_1 – S_3 . LF347 served as the opamp OA and the comparator OC. The value for CF was chosen to be 5nF. The CLU was realized with an ATmega328 microcontroller [19]. A suitable program was written and burnt into the

microcontroller to realize the logic of the CLU to control the analog switches, generate time period T_1 , count time period T_2 , and do some mathematical computations followed by displaying the output as angle sensed(in degrees). The clock frequency was chosen to be 7.8 kHz. Figure 5.8 shows the snapshot of the transducer under along with dual slope integrator output.

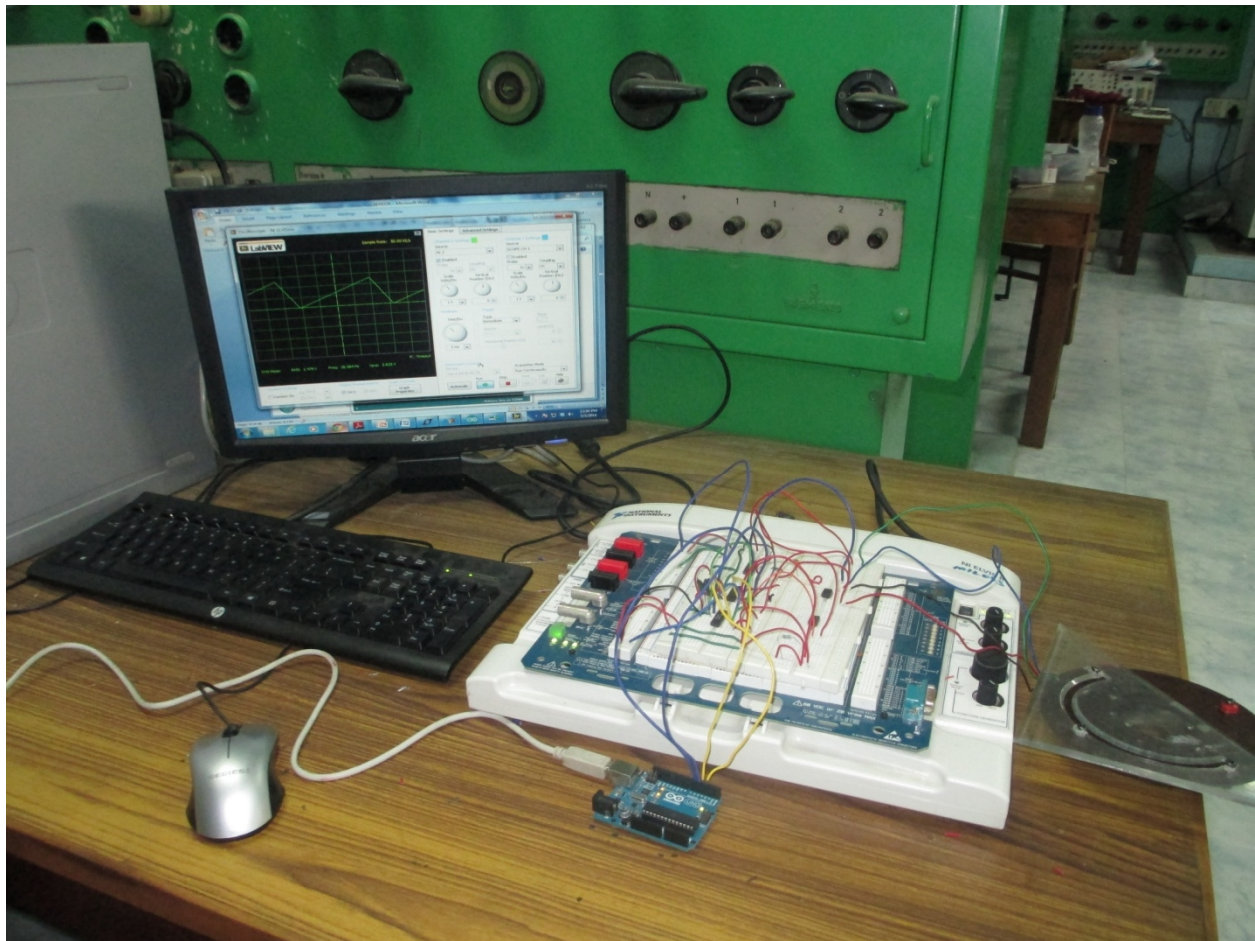


Figure 5.8: Experimental Setup of the angle transducer.

5.5 Experimental Results

Figure 5.9 portrays the screenshots of the waveform of integrator output on oscilloscope of NI ELVIS II board.

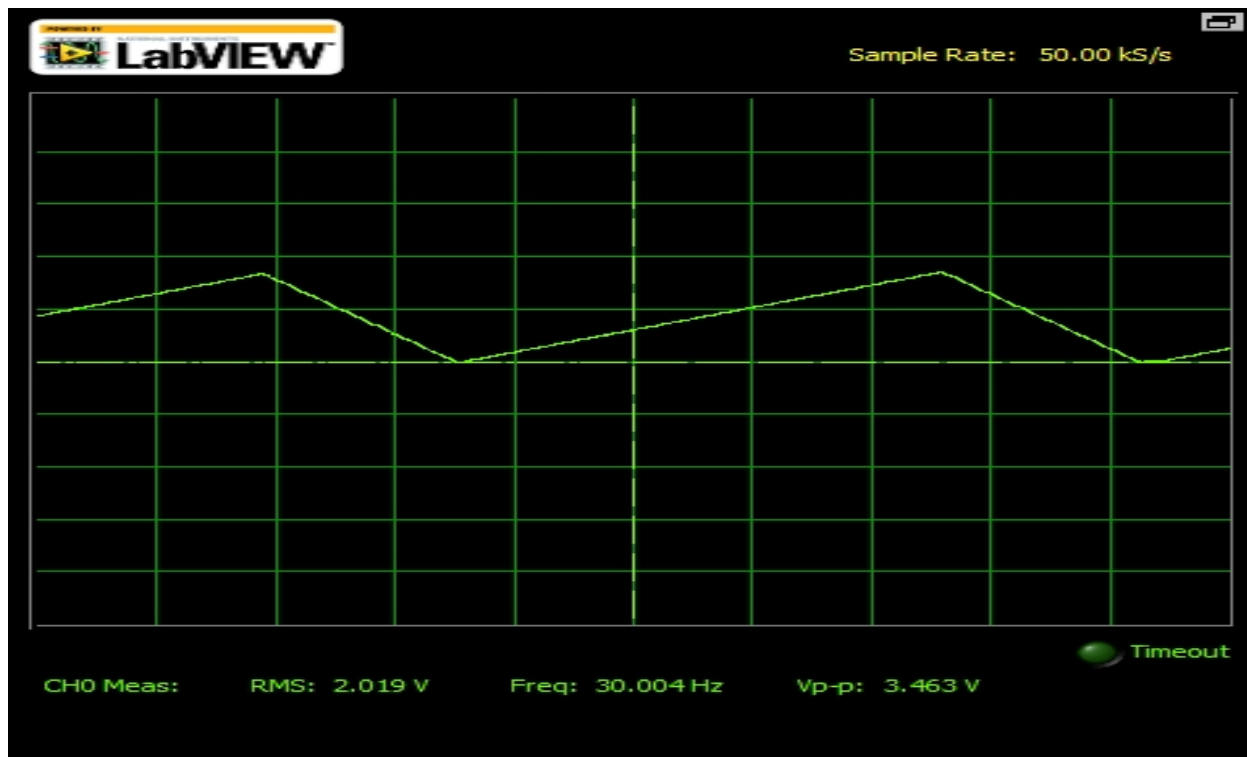


Figure 5.9(a): Integrator output for $C1 > C2$.

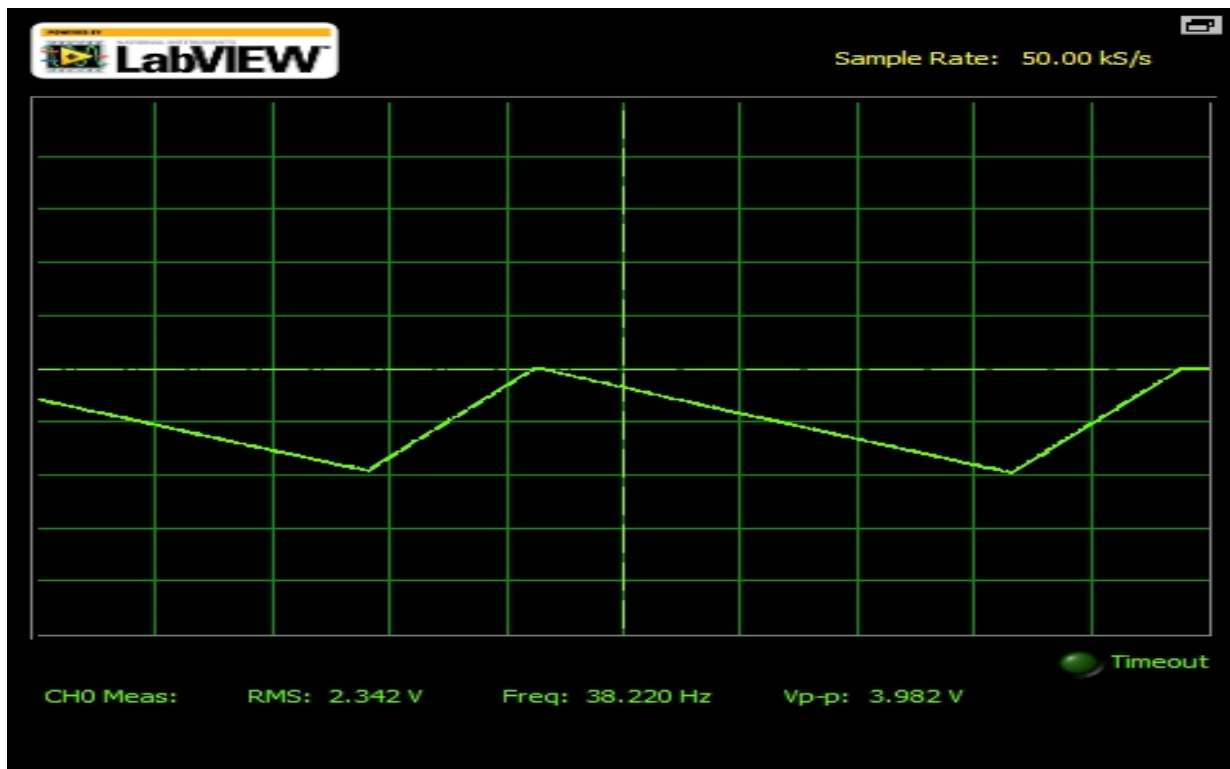


Figure 5.9(b): Integrator output for $C1 < C2$.

As can be seen from Figure 5.9(a) that output of the integrator ramp in the positive direction for $C1 > C2$, and the same ramp in positive direction for $C1 < C2$, Figure 5.9(b). As discussed earlier the time period $T2$ varies as we make angular movement of the sensor. The experimental values of $T2/T1$ are tabulated in the Table 5.1

S.No.	Angle(degrees)	$\left(\frac{C1 - C2}{C1 + C2}\right) = \frac{T2}{T1}$
1	0	-0.4200
2	10	-0.3875
3	20	-0.3300
4	30	-0.2775
5	40	-0.2200
6	50	-0.1575
7	60	-0.1025
8	70	-0.0525
9	80	-0.0025
10	90	0.0800
11	100	0.1825
12	110	0.2850
13	120	0.3850
14	130	0.4575
15	140	0.5525
16	150	0.6325
17	160	0.7325
18	170	0.8000

Table 5.1: Value of $T2/T1$ at different angular positions.

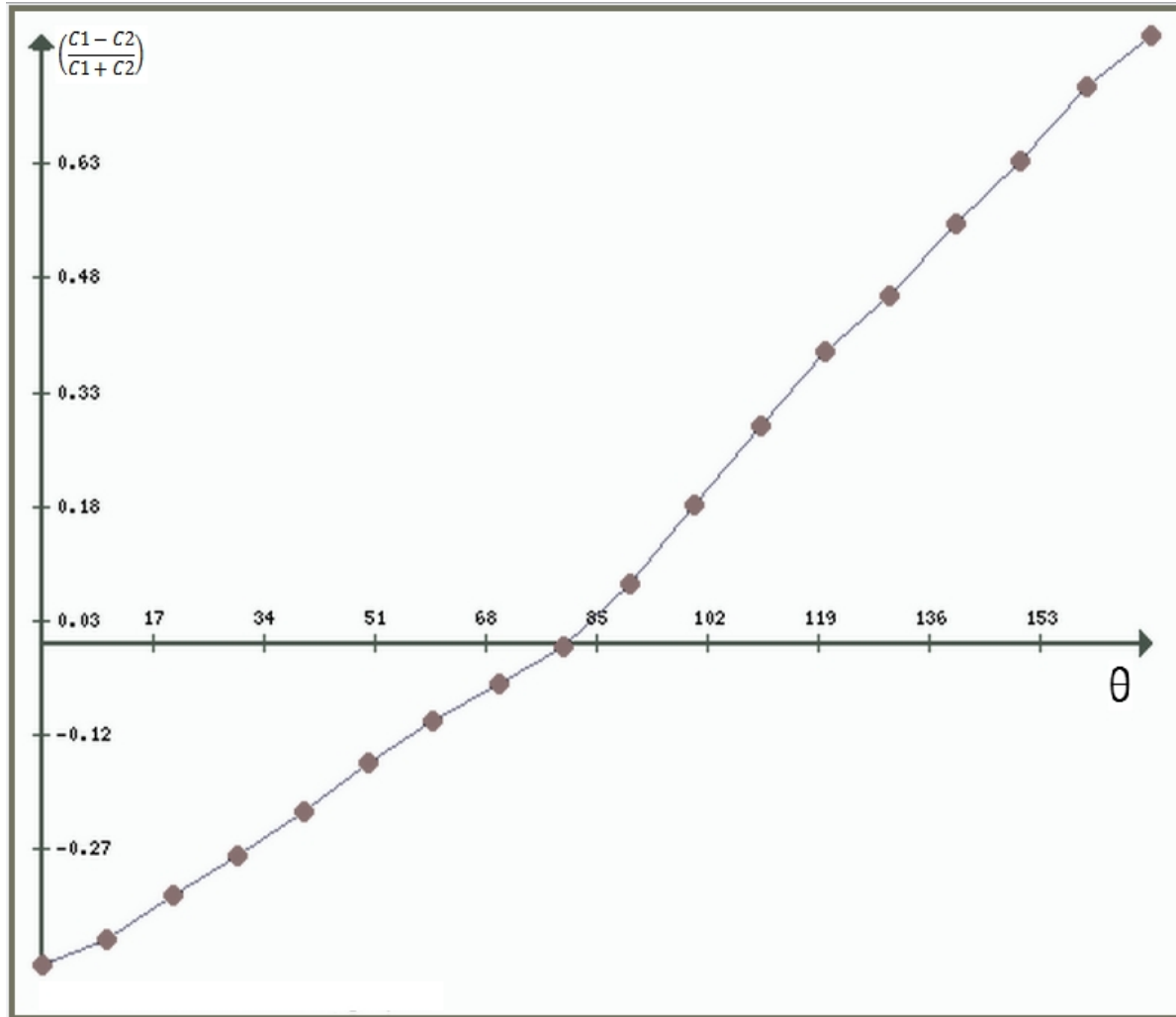


Figure 5.10: Linear variation of T_2/T_1 w.r.t θ .

From the values of Table 5.1 a graph was plotted showing the linear characteristics(Figure 5.10) The values of T_2/T_1 was computed using microcontroller whereas protractor was used to measure the corresponding angle.

The plot shown in Figure 5.10 resembles the ideal plot shown in Figure 4.5, but there is slight change in the slope as the curve crosses x-axis. This is due to some imperfection occurred in one of the bottom plate during etching process.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

An angular sensor was successfully designed and fabricated for the purpose of applying in automobile brake system. The fabrication process was very inexpensive which used only copper clad board and ferric chloride as an etchant.

The signal conditioning circuit along with push-pull capacitance combination was successfully tested on LTSpice. Then the hardware testing was carried out in which signal conditioning circuit was built on NI ELVIS board. An appropriate code was burnt in ATmega328 microcontroller for controlling switches used in the circuit. The angle sensed by the transducer was displayed on the serial monitor of Arduino Software.

6.2 Future Scope

Some additional improvements can be done in the work described in this thesis. Few of them are listed below:

- The prototype of the sensor designed can be made more portable by reducing the dimensions.
- The thickness of copper clad board can also be reduced in order to compensate the decrease in capacitance which was occurred by reducing the dimensions.
- The transducer can be made independent of the computer by using LCD display for displaying the angle sensed.

APPENDIX

C Code used in ATmega328 Microcontroller

```
#include <avr/io.h>
#include <avr/interrupt.h>

// global variable to count the number of overflows
volatile uint8_t tot_overflow0,tot_overflow1 ;
volatile unsigned int count=0;
int ICCmp =3;
int pbIn =1;
void statechange1()
{
    count=TCNT1;
    float temp=(count/373+82);
    Serial.println(temp);
    count=0;
    TCNT1 = 0;           // reset counter
    tot_overflow1 = 0;
    tot_overflow0 = 0;
}

void statechange2()
{
    count=TCNT1;
    float temp=(-count+16800)/210;
    Serial.println(temp);
    count=0;
```

```

TCNT1 = 0;           // reset counter
    tot_overflow1 = 0;
    tot_overflow0 = 0;
}
void timer0_init()
{
    OCR0A= 128;
    TCCR0B |= (1 << CS01);           // set up timer with prescaler = 8

    TCCR0A |= (1 << WGM01);

    TCNT0 = 0;                       // initialize counter
    TIMSK0 |= (1 << OCIE0A);         // enable compare interrupt
    sei();                           // enable global interrupts

}

void timer1_init()
{
    OCR1A = 39999;

    TCCR1B |= (1 << CS11) | (1 << WGM12); // set up timer with prescaler = 8
    TCNT1 = 0;                           // initialize counter

    TIMSK1 |= (1 << OCIE1A);           // enable compare interrupt
    sei();                             // enable global interrupts
    tot_overflow1 = 0;

}

// TIMER1 compare interrupt service routine
// called whenever TCNT1 is equal to OCR1A

```

```

ISR(TIMER1_COMPA_vect)
{
  tot_overflow1=tot_overflow1^1;
  // count++;
}
// TIMER0 compare interrupt service routine
// called whenever TCNT0 overflows
ISR(TIMER0_COMPA_vect)
{
  tot_overflow0=tot_overflow0^1;
}
int main(void)
{
  Serial.begin(9600);
  DDRD = 0b01110000;           // set input and output pins
  // initialize both timers
  timer1_init();
  timer0_init();
  autozero();
  // loop forever
  while(1)
  {
    if ((tot_overflow1 == 0)&(tot_overflow0 == 0))
    {
      PORTD =0b01010000;
    }

    else if ((tot_overflow1 == 0)&(tot_overflow0 == 1)){
      PORTD = 0b00100000;
    }
  }
}

```

```

else if ((tot_overflow1 == 1)&(tot_overflow0 == 0)&(digitalRead(ICCmp)==HIGH)){
    PORTD = 0b01100000;
    attachInterrupt(pbIn, statechange1, FALLING);
}

else if ((tot_overflow1 == 1)&(tot_overflow0 == 1)&(digitalRead(ICCmp)==HIGH)){
    PORTD = 0b00010000;
attachInterrupt(pbIn, statechange1, FALLING);
}

else if ((tot_overflow1 == 1)&(tot_overflow0 == 0)&(digitalRead(ICCmp)==LOW)){
PORTD = 0b01110000;
attachInterrupt(pbIn, statechange2, RISING);
}

else if ((tot_overflow1 == 1)&(tot_overflow0 == 1)&(digitalRead(ICCmp)==LOW)){
    PORTD = 0b00000000;
    attachInterrupt(pbIn, statechange2, RISING);
}
}

}

void autozero()
{
    if (PIND & _BV(3))
    {
        do
        {if (tot_overflow0 == 0)
            PORTD=0b01100000;
            if (tot_overflow0 == 1)

```



```

PORTD=0b00010000;
}while(PIND & _BV(3));
}
else
{
do
{
if (tot_overflow0 == 0)
PORTD=0b01110000;
if (tot_overflow0 == 1)
PORTD=0b00000000;
}while (digitalRead(ICCmp)==LOW);
}
// tot_overflow1 = 0;
//tot_overflow0 = 0;
}

```

REFERENCES

- [1] George B. and Kumar V. J. “Analysis of the Switched-Capacitor Dual Slope Capacitance-to-Digital Converter”, *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 5, pp. 997-1006, May 2010.
- [2] H. K. P. Neubert, *Instrument Transducers—An Introduction to Their Performance and Design*, 2nd ed. London, U.K.: Oxford Univ. Press, 2003.
- [3] E. W. Owen, “An integrating analog-to-digital converter for differential transducers,” *IEEE Trans. Instrum. Meas.*, vol. IM-28, no. 3, pp. 216–220, Sep. 1979.
- [4] J.M. G. Cama, S. A. Bota, E. Montane, and J. Samitier, “AMOSFET-only second order Delta-Sigma modulator for capacitive sensors interfaces,” in *Proc. IEEE ICECS*, Pafos, Cyprus, Sep. 1999, pp. 1689–1692.
- [5] K. Kazuyuki and K. Watanabe, “An auto ranging switched-capacitor analog-to-digital converter,” *IEEE Trans. Instrum. Meas.*, vol. IM-36, no. 4, pp. 879–881, Dec. 1987.
- [6] Z. Ignjatovic and M. F. Bocko, “An interface circuit for measuring capacitance changes based upon capacitance-to-duty cycle (CDC) converter,” *IEEE Sensors J.*, vol. 5, no. 3, pp. 403–410, Jun. 2005.
- [7] B. Wang, T. Kajita, T. Sun, and G. Temes, “High accuracy circuits for onchip capacitance ratio testing and sensor readout,” *IEEE Trans. Instrum. Meas.*, vol. 47, no. 1, pp. 16–20, Feb. 1998.
- [8] H. Matsumoto, H. Shimizu, and K. Watanabe, “Switched-capacitor charge-balancing analog-to-digital converter and its application to capacitance measurement,” *IEEE Trans. Instrum. Meas.*, vol. IM-36, no. 4, pp. 873–877, Dec. 1987.
- [9] P. D. Dimitropoulos, D. P. Karampatzakis, G. D. Panagopoulos, and G. I. Stamoulis, “A low-power/low-noise readout circuit for integrated capacitive sensors,” *IEEE Sensors J.*, vol. 6, no. 3, pp. 755–769, Jun. 2006.

- [10] Data Sheet, AD7745/AD7746, 24-Bit Capacitance-to-Digital Converter With Temperature Sensor, Norwood, MA: Analog Devices, Inc. [Online]. Available: <http://www.analog.com>.
- [11] B. George and V. J. Kumar, "Novel switched-capacitor dual slope capacitance to digital converter for differential capacitive sensors," in *Proc. IEEE I2MTC*, Singapore, May 2009, pp. 1–4.
- [12] B. George and V. J. Kumar, "Switched-capacitor triple slope capacitance to digital converter," *Proc. Inst. Elect. Eng.—Circuits, Devices Syst.*, vol. 153, no. 2, pp. 148–152, Apr. 2006.
- [13] E. R. Hnatek, *A User's Handbook of A/D and D/A Converters*. Melbourne, FL: Krieger, 1988.
- [14] Application note, 1999, AN017, The Integrating A/D Converter (ICL7135), Milpitas, CA: Intersil Americas Inc., 2002. [Online]. Available: <http://www.intersil.com/data/an/an017.pdf>.
- [15] *Semiconductor Device Modeling with SPICE*, by Giuseppe Massobrio and Paolo Antognetti, McGraw Hill, 1993 2nd Edition.
- [16] LTSpice manual (2011) <http://ltspice.linear.com/software/scad3.pdf>.
- [17] Arduino. Introduction to Arduino [Online]. Italy: Arduino URL: <http://arduino.cc/en/Guide/Introduction>.
- [18] Arduino. Arduino board Uno [Online]. Italy: Arduino URL: <http://arduino.cc/en/Main/ArduinoBoardUno>.
- [19] Atmel Corporation. ATmega328 datasheet [Online]. USA: Atmel Corporation; 10/2009 URL: <http://www.atmel.com/Images/doc8161.pdf>.
- [20] NI Educational Laboratory Virtual Instrumentation Suite II (NI ELVIS II) User Manual, National Instrument Corporation, April 2008; 33: 374629A-01.
- [21] <http://www.ni.com/white-paper/8433/en/>.

CURRICULUM VITAE

Name : Mohd Suaib Danish

Date of Birth : 10 October 1989

Place of Birth : New Delhi

Education : B.Tech from JSS Academy of Technical Education, Noida
2012 Batch.

E-mail : suaibdanish@gmail.com