MAR BASELIOS INSTITUTE OF TECHNOLOGY AND SCIENCE

NELLIMATTOM P.O, KOTHAMANGALAM, KERALA

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

THIRUVANANTHAPURAM

COURSE PROJECT REPORT

EEEL331 MICROPROCESSOR & MICROCONTROLER LAB

DIGITAL MEASURING TAPE

Submitted by

ARFAN ERSHAD-MBI23EE005
ATHIF SHAFI-MBI23EE006
BASIL MATHEW ELDHO-MBI23EE007

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

ELECTRICAL & ELECTRONICS ENGINEERING

MAR BASELIOS INSTITUTE OF TECHNOLOGY AND SCIENCE
NELLIMATTOM P.O, KOTHAMANGALAM, KERALA

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that the mini project on **"DIGITAL MEASURING TAPE "** is a bonafide record of the course project presented by **ARFAN ERSHAD- MBI23EE005** in partial fulfillment for the award of the degree of **Bachelor of Technology** in **Electrical & Electronics Engineering** of **APJ Abdul Kalam Technological University**, Thiruvananthapuram during the academic year 2023-2027.

Project Guide:
Thomas George
Assistant Professor
Department of EEE

Head of the Department
Dr. Arun Eldho Alias
Associate Professor
Department of EEE

ACKNOWLEDGEMENT

First and the foremost, I thank **ALMIGHTY GOD** who gave us the inner strength, resource and ability to complete our work successfully, without which all our efforts would have been in vain.

We express our sincere gratitude to the principal**, Dr. Thomas George** for providing us the provision to do the mini-project in the successful way.

We stand grateful to **Dr. Arun Eldho Alias**, Head of the Department of Electrical and Electronics Engineering, for her valuable advice and motivation.

We would like to extend our gratitude to the mini-project coordinators**, Asst. Prof Thomas George** and **Asst. Prof Basil Kummar NK**, Electrical and Electronics Engineering Department for their guidance and support. We would like to express our sincere gratitude for our mini-project guide **Asst. Prof Thomas George**, Electrical and Electronics Engineering Department for the encouragement and valuable guidance throughout the project.

We use this opportunity to thank all faculties in the department and the friends who have helped us to complete the mini-project, with their inspiration and co-operation.

ABSTRACT

DIGITAL MEASURING TAPE

This project presents a digital measuring tape built using an Arduino Nano, rotary encoder, and 16x2 LCD display. The system measures linear distance by detecting wheel rotation and converting it into digital output. As the wheel moves, the rotary encoder generates pulses that are counted by the Arduino Nano. These pulses are used to calculate the distance based on the wheel's radius and encoder resolution.

The measured distance is displayed in real-time on the LCD screen, and a potentiometer is used to adjust screen contrast. The circuit is compact, easy to build, and ideal for learning microcontroller programming and sensor interfacing. The project demonstrates a practical application of embedded systems in measurement and automation.

TABLE OF CONTENTS **PAGE.NO**

LIST OF FIGURES

LIST OF TABLE

CHAPTER 1

INTRODUCTION

This project is a digital measuring tape designed using an Arduino NANO, rotary encoder, and LCD display. The goal is to create a compact and accurate tool that can measure distance digitally, replacing traditional measuring tapes with a smarter alternative.

The system works by detecting the rotation of a wheel using a rotary encoder. As the wheel moves, the encoder generates electrical pulses. These pulses are counted by the Arduino, which then calculates the distance traveled based on the wheel's radius and the number of pulses per revolution. The calculated distance is displayed in real-time on a 16x2 LCD screen.

A 10k potentiometer is used to adjust the contrast of the LCD, and a 221Ω resistor helps protect the backlight circuit. The setup is simple, cost-effective, and ideal for learning embedded systems, microcontroller programming, and sensor interfacing.

This project is especially useful for students and hobbyists who want to understand how sensors and microcontrollers can be combined to solve practical problems. It also opens up possibilities for future upgrades, such as wireless data logging or mobile app integration.

.

CHAPTER 2

OBJECTIVES

- To design and build a digital measuring tape using Arduino NANO and a rotary encoder
- To convert rotational movement into accurate linear distance using pulse counting
- To display the measured distance in real-time on a 16x2 LCD screen
- To create a compact and user-friendly tool for distance measurement
- To understand and apply basic principles of microcontroller programming & sensor interfacing
- To explore practical applications of embedded systems in measurement and automation

CHAPTER 3

MATERIALS AND COMPONENTS

SL NO	COMPONENTS USED	QUANTITY
1	Arduino NANO	1
2	Rotary Encoder	2
3	16x2 LCD Display	1
4	10k Variable Resistor	1
5	221Ω Resistor	1
6	Breadboard	1
7	Power Supply 9V	1
8	Jumper Wires	

TABLE : 3.1 Material Estimation

3.1. Arduino NANO:

Acts as the brain of the system. It reads signals from the rotary encoder and calculates the distance

Fig:3.1

3.2. Rotary Encoder:

Detect the rotation of the measuring wheel. As the wheel moves, the encoder generates electrical pulses that are sent to the Arduino. These pulses are counted and converted into distance using a mathematical formula..

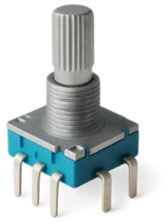

Fig:3.2

3.3. 16x2 LCD Display:

Shows the measured distance in centimeters in real-time..

Fig:3.3

3.4. 10k Variable Resistor:

Controls the contrast of the LCD screen for better visibility

Fig:3.4

3.5. 221Ω Resistor :

Provide the required voltage for logic operation and actuator drive

Fig:3.5

3.6. Breadboard:

A platform for building and testing the circuit without soldering.

Fig:3.6

3.7. Power supply

Provide the required voltage for logic operation and actuator drive.

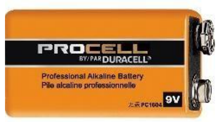

Fig:3.7

CHAPTER 4

CIRCUIT DESIGN AND WORKING PRINCIPLE

4.1. WORKING PRINCIPLE

The rotary encoder detects the rotation of the measuring wheel and generates electrical pulses. These pulses are read by the Arduino Nano through its interrupt-capable pins. Each pulse corresponds to a small movement of the wheel.

Using the formula**:**

$$\text{Distance} = (2\pi R/N) \times \text{Pulse Count}$$

- R is the radius of the wheel
- N is the number of pulses per full rotation
- Pulse Count is tracked by the Nano

The Nano calculates the distance and updates the LCD display in real-time. The potentiometer allows the user to adjust the contrast of the display for better readability.

4.2. CIRCUIT DIAGRAM

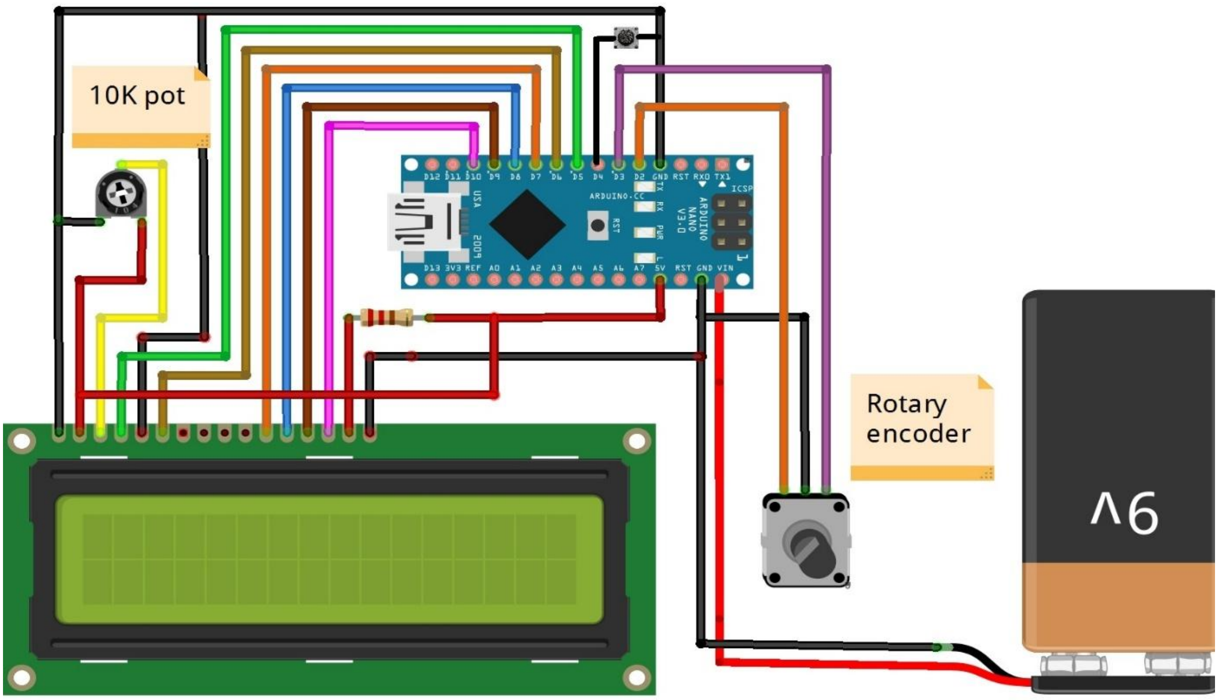

Fig:4.1

The circuit is built around an **Arduino Nano**, a compact microcontroller ideal for breadboard-based projects. A **rotary encoder** is connected to digital pins D2 and D3 of the Nano to detect wheel rotation. As the wheel turns, the encoder generates pulses that the Nano counts to calculate distance.

A **16x2 LCD display** is connected to digital pins D5 through D10 to show the measured distance in centimeters. A **10k potentiometer** is connected to the LCD's contrast pin (V0) to adjust screen visibility. A **221Ω resistor** is placed in series with the LCD backlight to protect it from excess current.

All components are connected using jumper wires on a breadboard, and the Nano is powered an external battery. The compact size of the Nano makes the entire setup neat and portable

4.3. WORKING MODEL

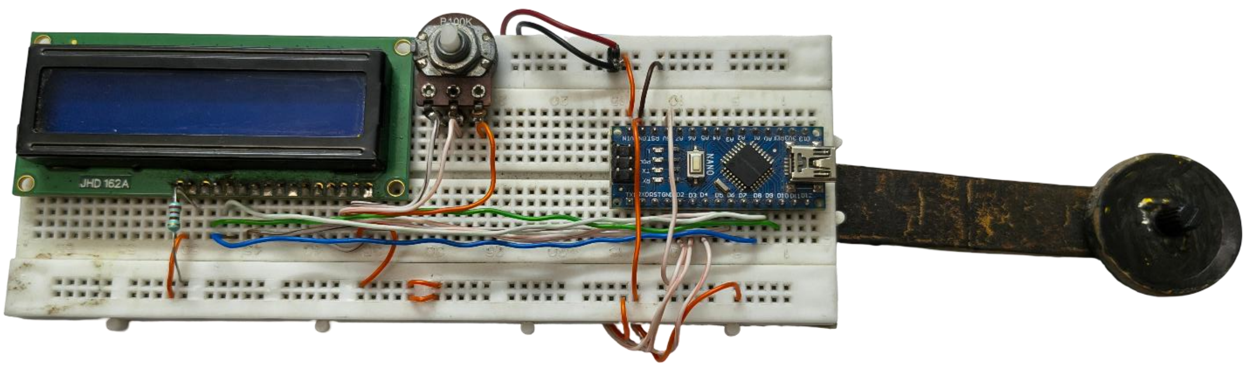

Fig:4.2

CHAPTER 5

BLOCK DIAGRAM

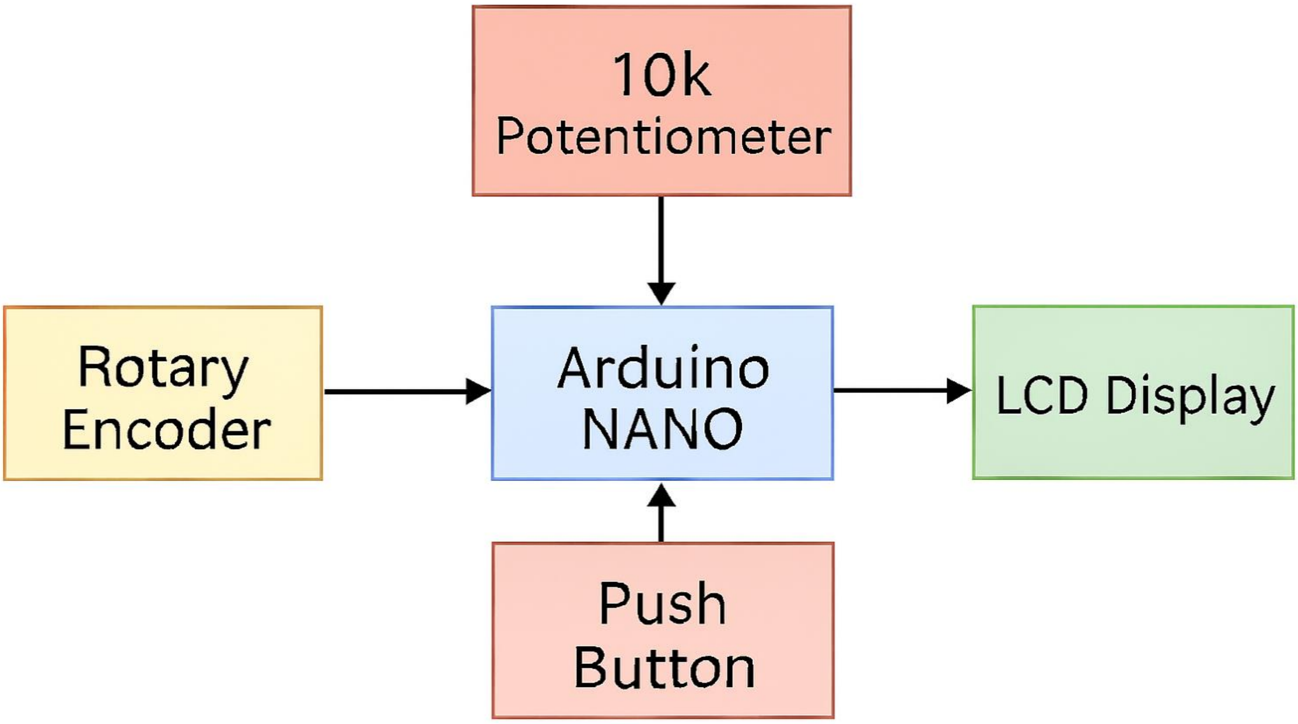

Fig:5.1

CHAPTER 6

OPERATION

1. Place the measuring wheel at the starting point of the surface.
2. As the wheel rolls forward, it rotates the rotary encoder.
3. The encoder sends electrical pulses to the Arduino Nano.
4. The Nano counts these pulses and calculates the distance using a preset formula.
5. The calculated distance is displayed in real-time on the 16x2 LCD screen.
6. The 10k potentiometer allows you to adjust the LCD contrast for better visibility.
7. The system continues updating the display as long as the wheel is moving.
8. The push button is used to reset the reading to zero that shown on the LCD display.
9. Power is supplied through USB or an external battery connected to the Nano.

CHAPTER 7

IMPLEMENTATION STEPS

1. **Component Setup**
    - Gather all required components: Arduino Nano, rotary encoder, 16x2 LCD, 10k potentiometer, 221Ω resistor, jumper wires, and breadboard.

2. **Mount the Rotary Encoder**
    - Attach the rotary encoder to a wheel so it rotates as the wheel moves.
    - Connect the encoder's output pins to digital pins D2 and D3 on the Arduino Nano.

3. **Connect the LCD Display**
    - Wire the 16x2 LCD to digital pins D5–D10 of the Nano.
    - Connect the LCD's V0 pin to the middle pin of the 10k potentiometer for contrast control.
    - Add a 221Ω resistor in series with the LCD backlight pin to protect it.

4. **Power Supply**
    - Power the Arduino Nano using a USB cable or external battery.
    - Ensure all ground and VCC connections are properly made.

5. **Upload the Code**
    - Write or upload the Arduino sketch that:
    - Reads pulses from the rotary encoder
    - Calculates distance using the formula

$$\text{Distance}=(2\pi R/N)\times\text{Pulse Count}$$

    - Displays the distance on the LCD

6. **Testing**
    - Place the wheel at a starting point and roll it forward.
    - Observe the LCD display updating the distance in real-time.
    - Adjust the potentiometer if the screen contrast is too low or too high.

7. **Final Adjustments**
    - Check for loose connections or incorrect readings.
    - Fine-tune the code or wheel calibration if needed.
    - Optionally, mount the setup on a small chassis or enclosure for portability.

CHAPTER 8

TESTING AND DISCUSSION

8.1 TESTING PROCEDURES

- The system was powered using a USB cable connected to the Arduino Nano.
- The wheel was placed at a known starting point and rolled along a flat surface.
- As the wheel rotated, the rotary encoder generated pulses which were read by the Nano.
- The LCD display updated the measured distance in real-time.
- Multiple trials were conducted over different lengths to check accuracy and consistency.

8.2 OBSERVATIONS

- The distance displayed on the LCD matched closely with manual measurements.
- The system responded quickly to wheel movement, with minimal delay.
- Adjusting the potentiometer improved LCD visibility under different lighting conditions.
- The encoder produced stable pulse signals without noticeable jitter or noise.

8.3 DISCUSSION

- The project clearly showed that a rotary encoder can measure distance effectively.
- Using interrupt pins on the Arduino Nano helped count pulses accurately, even when the wheel moved fast.
- The distance formula worked well, but setting the correct wheel size was important for accurate results.
- The setup is small, easy to build, and great for learning or DIY use.
- Future upgrades could include a reset button, saving measurements, or adding wireless features.

CHAPTER 9

CONCLUSION

The digital measuring tape project successfully demonstrates how a rotary encoder and Arduino Nano can be used to measure distance accurately. By converting wheel rotation into electrical pulses and displaying the result on an LCD screen, the system offers a practical and user-friendly solution for real-time distance tracking.

The circuit is easy to build, compact, and reliable. It serves as a great learning tool for understanding microcontroller programming, sensor interfacing, and embedded system design. With further improvements, such as adding memory or wireless features, this project could be expanded into more advanced applications.

CHAPTER 10

APPENDIX

```
#include <LiquidCrystal.h>
 LiquidCrystal lcd(5, 6, 7, 8, 9, 10);
 const int clkPin = 2;
 const int dtPin = 3;
 const int swPin = 4;
 volatile int position = 0;
 float distance = 0;
 const float pi = 3.14;
 const float diameter = 3.4;
 const int pulsesPerRev = 40;
 unsigned long lastInterruptTime = 0;
 const unsigned long debounceDelay = 2;

void setup()
 {
  pinMode(clkPin, INPUT_PULLUP);
  pinMode(dtPin, INPUT_PULLUP);
  pinMode(swPin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(clkPin), readEncoder, CHANGE);
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("MEASURING WHEEL");
  delay(2000);
  lcd.clear();
}

void loop()
 {
  if (digitalRead(swPin) == LOW)
{
    position = 0;
    distance = 0;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("RESETTING...");
    delay(500);
    lcd.clear();
 }
  distance = ((pi * diameter) / pulsesPerRev) * position;
```

```
 if (distance < 0) distance = 0;
 lcd.setCursor(0, 0);
 lcd.print("Distance:");
 lcd.setCursor(0, 1);
 lcd.print(distance, 2);
 lcd.print(" cm   ");
 delay(100);
}

void readEncoder()
 {
 unsigned long currentTime = millis();
 if (currentTime - lastInterruptTime > debounceDelay)
 {
   if (digitalRead(dtPin) != digitalRead(clkPin))
  {
     position++;
   }
else
 {
     position--;
   }
   lastInterruptTime = currentTime;
  }
}
```

CHAPTER 11

REFERENCES

[1] S. P. Sabarinath, “Digital Measuring Tape,” ElectronicWings, Dec. 13, 2021. [Online]. Available:

[2] J. Craftsman, “How to Make a Digital Tape Measure: Step-by-Step Guide / Arduino + 3D Print,” YouTube, Apr. 2023. [Online].

[3] Arduino Forum, “Digital Tape Measure Decoding Serial Data,” Arduino.cc, Dec. 14, 2024. [Online].