PROJECT REPORT

For the Course

EE-383 Instrumentation and Measurements

B.E. Electrical Engineering

Multi Range Digital Ammeter

Name	Registration Number	Syndicate
SHAFI ULLAH	423652	B
Muhammad Umer Sajid	413946	B
Adeeba Tahir	412229	B

Instructor: Rabeea Basir

Lab Engineer: LE Haris

Department of Electrical Engineering

College of Electrical and Mechanical Engineering (CEME), NUST-Pakistan

Contents

1 Objective

To design and implement a multifunctional measurement device capable of measuring current, voltage, and rotational speed (RPM).

2 Abstract

This project presents the design and implementation of a multifunctional measurement device integrating three key functionalities: current measurement using a custom multirange ammeter circuit, voltage measurement with a high-precision sensor, and RPM measurement using a tachometer. The device incorporates a 3D-printed casing for compactness and user-friendliness. It ensures accuracy and reliability in measurements, making it ideal for various applications in instrumentation and measurements.

3 Introduction

Instrumentation plays a crucial role in modern electrical engineering by providing accurate and reliable data for analysis. This project aims to create a multifunctional measurement device that can measure current, voltage, and RPM. The custom current measurement circuit (a multirange ammeter) is designed to operate across multiple current ranges with high accuracy. The voltage measurement uses a sensor module connected to an ADC, while the RPM measurement utilizes a tachometer sensor. A 3D-printed casing ensures the device's portability and durability.

4 Analysis

Roadmap of the Project

The project development process was structured into the following stages:

1. **Conceptualization**: Identifying the core functionalities (current, voltage, and RPM measurements) and the design goals for accuracy, portability, and ease of use.
2. **Research and Component Selection**: Exploring potential components to meet performance and cost requirements.
3. **Design and Simulation**: Creating circuit designs and simulating them to validate the functionality.
4. **Implementation**: Building the physical device, programming the Arduino Nano, and integrating hardware with software.
5. **Casing and Packaging**: Designing a durable and compact 3D-printed casing.
6. **Testing and Optimization**: Conducting rigorous testing under various conditions to ensure reliable operation.
7. **Documentation**: Compiling project details and analysis for review and further development.

Component Options and Selection

Each component of the device was chosen after careful consideration of several alternatives:

1. **Current Measurement**
   - **Options Considered**: Shunt resistors of varying resistances, current transformers, and Hall effect sensors.
   - **Selected**: Shunt resistors (0.5 Ω, 0.25 Ω, and 0.11 Ω) paired with an operational amplifier (LM358).
   - **Justification**: Shunt resistors provide a cost-effective and accurate means of current measurement. The LM358 operational amplifier ensures the voltage drop across the resistors is amplified for precise ADC readings.
2. **Voltage Measurement**
   - **Options Considered**: Dedicated voltage sensors, resistive voltage dividers, and integrated circuit modules.
   - **Selected**: A voltage sensor module with a built-in voltage divider.
   - **Justification**: The module simplifies the interface with the Arduino Nano and ensures safety while maintaining accuracy.
3. **RPM Measurement**
   - **Options Considered**: Optical encoders, Hall effect sensors, and tachometer sensors.
   - **Selected**: A tachometer sensor.
   - **Justification**: The tachometer sensor offers reliable performance and seamless integration for RPM detection without requiring extensive circuitry.
4. **Microcontroller**
   - **Options Considered**: Arduino Nano, Arduino Uno, and ESP32.
   - **Selected**: Arduino Nano.
   - **Justification**: Compact form factor and sufficient processing capability for this application, while being cost-effective.
5. **Display**
   - **Options Considered**: OLED, LCD, and seven-segment displays.
   - **Selected**: OLED display.
   - **Justification**: High readability, low power consumption, and ability to display multiple parameters simultaneously.
6. **Casing**
   - **Options Considered**: ABS plastic, metal enclosures, and PLA 3D-printed casings.
   - **Selected**: 3D-printed casing made of PLA material.
   - **Justification**: Customizability, cost-effectiveness, and compatibility with the project’s compact design goals.

5 Theory

5.1 Current Measurement Circuit

The current measurement circuit utilizes shunt resistors to produce a proportional voltage drop based on the current passing through. An operational amplifier amplifies this voltage, which is then fed into an ADC for digital processing. Multiple ranges are achieved by switching between shunt resistors.

5.2 Voltage Measurement

The voltage sensor module measures the potential difference and ensures safety and precision when connected to high or low voltage sources.

5.3 RPM Measurement

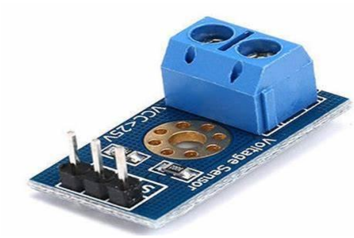
The tachometer sensor captures the rotational speed of a motor or shaft and converts the data into RPM, enabling accurate monitoring of mechanical systems.

6 Components

- Shunt resistors: 0.5 Ω, 0.25 Ω, 0.11Ω (Power Resistors)
- Operational amplifier: LM358
- 3D-printed casing
- dc jack
- Rotary switch

- Mechanical switch

- Voltage sensor module

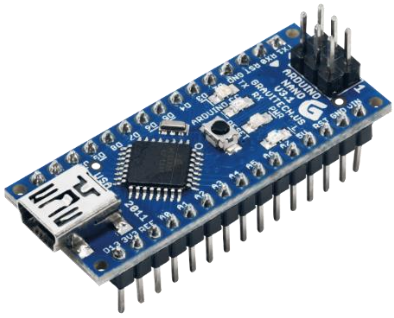


- Power Resistor

- OLED Display

- Tachometer sensor

- ARDUINO NANO



7 Software

- Circuit simulation (Protious)
- Programming environment for microcontroller (Arduino IDE)
- SOLID WORKS for 3D modeling

8 Calculation

➢ **Ammeter:**

The shunt resistor values and their corresponding voltage drops are calculated as follows:

The formula used for the voltage drop across the shunt resistor is given by:

$$V = I \times R$$

- For 2 A: $R = 0.5\ \Omega$, Voltage drop: $V = 2 \times 0.5 = 1.0$ V
- For 4 A: $R = 0.25\ \Omega$, Voltage drop: $V = 4 \times 0.25 = 1.0$ V
- For 9 A: $R = 0.11\ \Omega$, Voltage drop: $V = 9 \times 0.11 \approx 0.99$ V

These calculations ensure the circuit operates within the ADC's input range, providing accurate current measurements for each range.

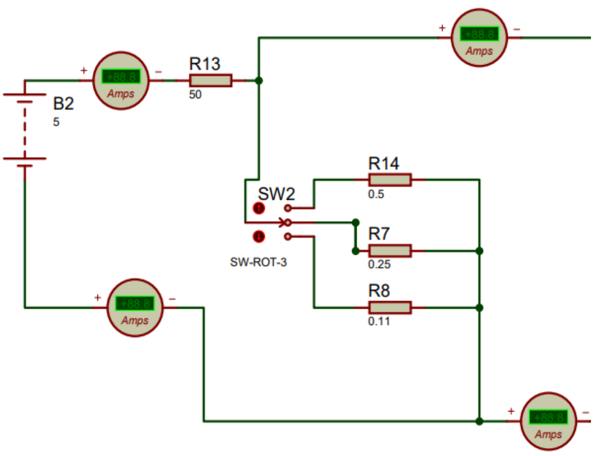
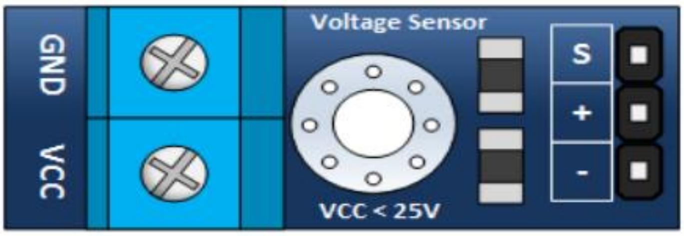


Figure 1: Multirange Ammeter Circuit for Current Measurement

- **Voltmeter**:

1. **Pinout:**



2. **Steps for interfacing**

- **Wiring:**

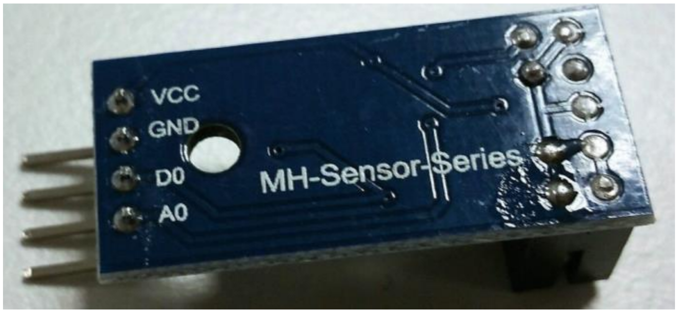
    - **VCC**: Connect to Arduino Nano's 5V pin.
    - **GND**: Connect to Arduino Nano's GND pin.
    - **OUT**: Connect to an analog pin (e.g., A0).

- **Code Logic:**

    - Use `analogRead()` to measure the voltage.
    - Convert the analog value to actual voltage using the formula: V=analogValue×(5.0/1023)×VoltageDividerFactorV = \text{analogValue} \times (5.0 / 1023) \times \text{VoltageDividerFactor}V=analogValue×(5.0/1023)×VoltageDividerFactor
    - Voltage divider factor is typically 5 for a 0-25V module.

- **Tachometer:**

1. **Pinout:**



2. **Steps for interfacing**

- **Wiring**

    - **VCC**: Connect to Arduino Nano's 5V pin.
    - **GND**: Connect to Arduino Nano's GND pin.
    - **OUT**: Connect to a digital pin (e.g., D2).

Code Logic:

- Use `digitalRead()` to detect high/low signal on the output pin.
- Process the signal to determine object presence or calculate RPM (for tachometer use).

9 Simulation

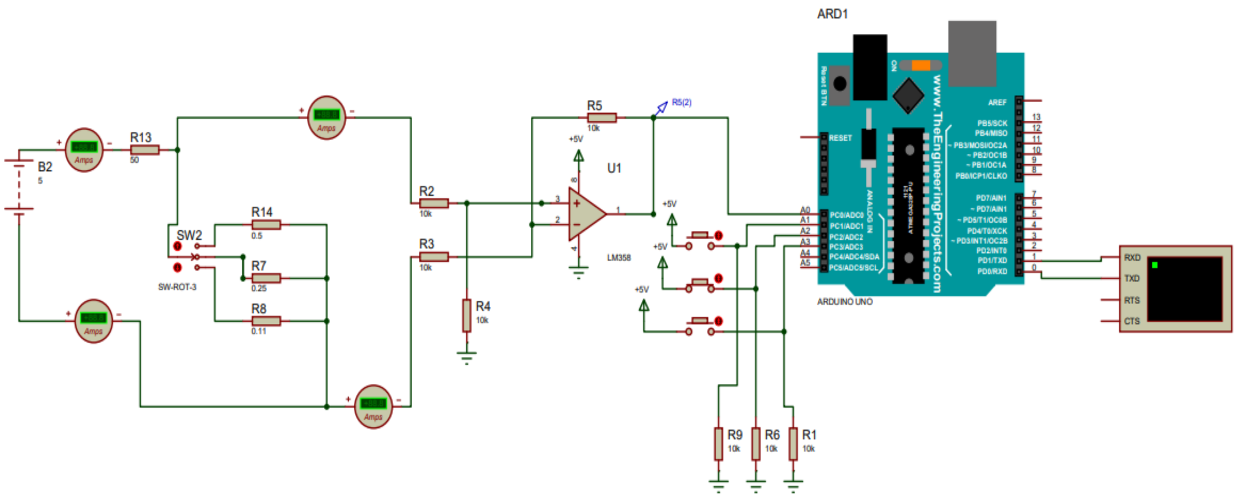


Figure 2: Simulation Results for the Current Measurement Circuit

10 Code

```
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
Adafruit_SSD1306 display(SCREEN_
WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);
// Sensor Pins
#define VOLTAGE_PIN A3
#define CURRENT_PIN A7
#define SPEED_SENSOR_PIN 2
// Rotary Switch Pins
#define ROTARY_SWITCH_VOLTAGE 3

#define ROTARY_SWITCH_CURRENT 4
#define ROTARY_SWITCH_SPEED 5


// Constants for Voltage Sensor


const float VOLTAGE_FACTOR = 5.128;
```

```
const float VCC = 5.00;
// Tachometer Variables
volatile int holes = 0;
volatile unsigned long lastTime = 0;
float rpm = 0;

void setup() {


 pinMode(VOLTAGE_PIN, INPUT);

  pinMode(CURRENT_PIN, INPUT);
  pinMode(SPEED_SENSOR_PIN,
INPUT_PULLUP);

  pinMode(ROTARY_SWITCH_VOLTAGE,
INPUT_PULLUP);
  pinMode(ROTARY_SWITCH_CURRENT,
INPUT_PULLUP);

  pinMode(ROTARY_SWITCH_SPEED,
INPUT_PULLUP);
  // Tachometer Interrupt
```

```
attachInterrupt(digitalPinToInterrupt(SPEED_SENSOR_PIN), countHoles, FALLING);
 // Initialize OLED Display
 if (!display.begin(SSD1306_I2C_ADDRESS, 0x3C)) {
   Serial.println(F("SSD1306 allocation failed"));
   for (;;);
 }
 display.clearDisplay();
 display.setTextColor(SSD1306_WHITE);
 // Power-on Animation
 powerOnAnimation();
}
void loop() {
 if (digitalRead(ROTARY_SWITCH_VOLTAGE) == LOW) {
   displayVoltage();
 } else if (digitalRead(ROTARY_SWITCH_CURRENT) == LOW) {
   displayCurrent();
 } else if (digitalRead(ROTARY_SWITCH_SPEED) == LOW) {
   displaySpeed();
 }
}
// Power-On Animation
void powerOnAnimation() {
 display.clearDisplay();
 display.setTextSize(2);
 display.setCursor(20, 20);
 display.print("PowerMate");
 display.display();
 delay(1000);
 display.clearDisplay();
 display.setTextSize(1);
 display.setCursor(20, 40);
 display.print("Powered by MUS");
 display.display();
 delay(2000);
 display.clearDisplay();
 display.setTextSize(2);
 display.setCursor(40, 25);
 display.print("MUS");
 display.display();
 delay(1000);
 display.clearDisplay();
}
// Display Voltage
void displayVoltage() {
 float voltageSensorVal = analogRead(VOLTAGE_PIN);
 float vOut = (voltageSensorVal / 1024) * VCC;
 float vIn = vOut * VOLTAGE_FACTOR;
 display.clearDisplay();
 display.setTextSize(1);
 display.setCursor(0, 10);
 display.print("Voltage:");
 display.setTextSize(2);
 display.setCursor(0, 30);
 display.print(vIn, 2);
 display.print(" V");
 display.display();
 delay(500);
}
// Display Current
void displayCurrent() {
 float average = 0;
 for (int i = 0; i < 1000; i++) {
  average += (0.19 * analogRead(CURRENT_PIN) - 25) / 1000.0;
  delay(1);
 }
 display.clearDisplay();
 display.setTextSize(1);
 display.setCursor(0, 10);
 display.print("Current:");
 display.setTextSize(2);
 display.setCursor(0, 30);
 display.print(average, 2);
 display.print(" A");
 display.display();
 delay(500);
}
// Display Speed (RPM)
void displaySpeed() {
 noInterrupts();
 rpm = (holes * 60.0) / (millis() - lastTime) * 1000.0;
 holes = 0;
 lastTime = millis();
 interrupts();
 display.clearDisplay();
 display.setTextSize(1);
 display.setCursor(0, 10);
 display.print("Speed:");
 display.setTextSize(2);
 display.setCursor(0, 30);
 display.print(rpm, 2);
 display.print(" RPM");
 display.display();
 delay(500);
}
// Tachometer Interrupt Handler
void countHoles() {
 holes++;
}
```

11 Casing Design

The device casing was designed using CAD software and printed using PLA material. The casing provides durability and portability.

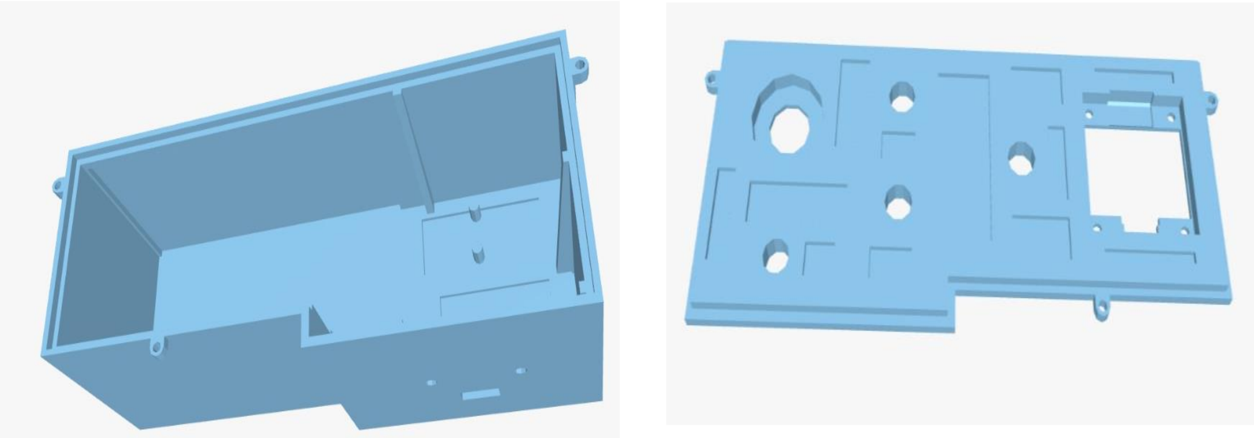


Figure 3: 3D-Printed Casing Design

12 Testing

The device was tested under various conditions to ensure accuracy and reliability. Test results indicate:

- Consistent current measurement across all ranges.
- Accurate voltage readings matching theoretical values.
- Reliable RPM measurements under different speeds.

13 Project

Figure 4: Final Product

13 Conclusion

This project successfully achieved the design and implementation of a multifunctional measurement device. The integration of current, voltage, and RPM measurement systems with a custom 3D-printed casing resulted in a practical and efficient device. Future enhancements could include wireless data transmission and a more compact design.

14 References

1. Instrumentation and Measurements Lab Manual.
2. Principles of Electronic Instrumentation and Measurement, Berlin Getz,1988.
3. Electronic Instrumentation and Measurement, David A. Bell, 199