Indian Institute of Technology Gandhinagar
ES116 Principles and Applications of Electrical Engineering
Lab - 2
Introduction to Arduino and Design a Voltage divider circuit with Arduino ADC

Lab Report No : 2
Name : Maitreyee Patel
Roll no. : 25110185

Name of teammate : Parth Malani
Roll no. : 25110186

Name of the TA : Khomendra Sahu

1.Introduction

A. Theory

A voltage divider is basically a simple circuit that turns a large voltage into a smaller one. It uses two resistors in series. According to Ohm's law, the voltage is shared between them. The formula we used to find the output voltage (V_{out}) is $V_{out} = V_{in} \times R_2 / (R_1 + R_2)$

Here (V_{in}) is our 5V supply from the Arduino, and R_2 is the resistor where we measure the output.

The Arduino UNO has a built-in analog to digital converter that measures the analog value and maps it to a digital value between 0-1023

The formula for converting ADC value to Analog voltage is;
V = ADC value × (5.0/1023.0)

B. Objectives

Learning Arduino Basics*:* Getting a feel for how the Arduino Uno board works and how to set it up.

Building the Circuit: Learning the practical steps to wire up a basic voltage divider on a breadboard.

ADC Usage: Using the Arduino's built-in Analog to Digital Converter (ADC) to read and measure the output voltage.

Accuracy Check: Comparing the Arduino's readings against a standard multimeter to see how precise the board actually is.

C. Motivation

The reason we are doing this lab is that most real-world sensors like light sensors, flex sensors, or thermistors don't just give a direct digital reading. Instead, they change their resistance based on the environment. To actually read those changes with a microcontroller like an Arduino, we have to turn that resistance into a voltage.
The voltage divider is the most basic way to do this. By mastering this, we can eventually interface all sorts of analog sensors to our projects. In addition to this it is a good way to see how accurate the Arduino's ADC (Analog to Digital Converter) actually is compared to professional tools like a multimeter.

D.Components and Tools

- Arduino Uno Board
- Breadboard
- Resistor: 1 kΩ (R_1)
- Potentiometer: 10 kΩ (R_2, variable)
- Screwdriver
- Jumper wires (male-to-male)
- Digital Multimeter
- USB cable for Arduino-laptop connection
- 5V DC supply
- Arduino IDE software.

2. Experiments and Implementation

A. Experiments Performed

This experiment was performed in three configurations to validate the voltage divider principle;

1. Vout > Vmid; The potentiometer was set to 1.35kΩ, creating a higher output voltage.
2. Vout ≈ Vmid; The potentiometer was set to 1.02kΩ, providing equal Voltage division.
3. Vout < Vmid; The potentiometer was set to 0.39kΩ, creating a lower output voltage.

For every configuration, voltage measurements were performed via three methods; Arduino ADC, digital multimeter and Voltage calculated by theoretical formula.

B. Challenges Encountered

1.Had lots of difficulty in getting the desired resistance using a screwdriver.
2. Taking precise measurements due to fluctuations in ADC readings.

C. Solutions Attempted

1.Made fine measurements by rotating the screwdriver back and forth on the potentiometer.
2. Waiting till the fluctuations in the ADC become relatively small and then taking measurements.

D. Relevant Diagrams, Algorithms, Codes

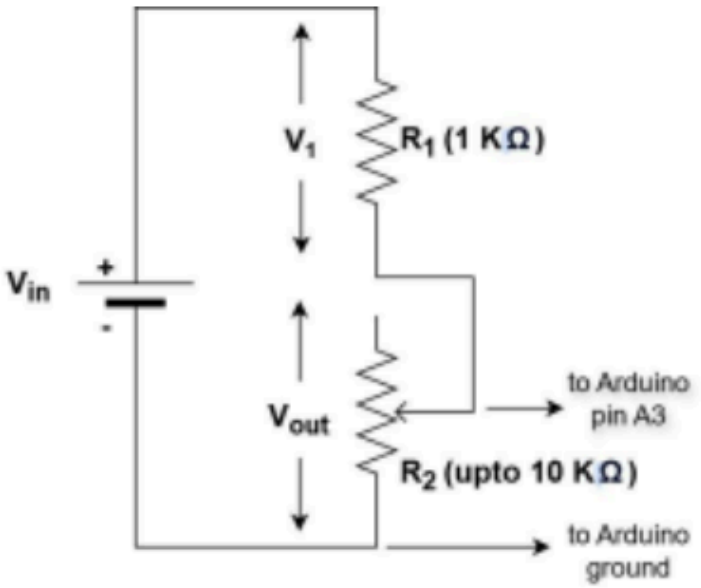


Fig: Circuit Diagram of Voltage divider circuit with Arduino ADC

```
  void setup() {
    // initialize serial communication at 9600 bits per
second:
    Serial.begin(9600);
  }

  // the loop routine runs over and over again forever:
  void loop() {
    // read the input on analog pin 0:
    int sensorValue = analogRead(A3);
    // Convert the analog reading (which goes from 0 - 1023)
to a voltage (0 - 5V):
    float voltage = sensorValue * (5.0 / 1023.0);
    // print out the value you read:
    Serial.println(voltage);
  }
```

Code: Analog Voltage measurement code for Arduino

E. Complete Procedure

Circuit Assembly

1. Connect one end of 1kΩ resistor (R_1) to 5V pin of Arduino UNO board.
2. Connect the other end of R_1 to middle pin of the Potentiometer
3. Connect one fixed terminal (either 1 or 3) of the Potentiometer to Arduino GND
4. Connect the middle pin of the Potentiometer to the A3 Analog pin of Arduino UNO board.
5. Connect the Arduino UNO to your laptop using a USB cable.
6. Verify that all the wires are firmly connected in the breadboard.

Arduino Programming

1. Open Arduino IDE on your laptop
2. Use the code preset (files ~ examples ~ ReadAnalogSignal) and modify the code to read analog value from pin A3.
3. Select appropriate COM port
4. Upload the compiled code to Arduino UNO
5. Open the serial monitor to take the digital readings.

Voltage Measurement

1. Open serial monitor in Arduino IDE
2. Record Voltage readings displayed by arduino ADC
3. Place the Red probe on the middle pin of the Potentiometer and black probe on GND of the Arduino UNO.
4. Set multimeter to record DC voltage (V_{dc})
5. Record multimeter voltage readings

Resistance Measurement

1. Connect the red probe to the middle pin of the Potentiometer and black probe to the side pin of the Potentiometer.
2. Turn off the Arduino UNO board (Unplug the USB from laptop)
3. Set the multimeter to record Resistance
4. Record the resistance

3.Observations and Results

A. Observations/ Learning Outcomes.

1.We successfully checked the voltage divider circuit and it worked just like we expected.
2.As the value of R_2 increased, V_{out} also increased and vice-versa as implied by the formula:

$$V_{out} = V_{in} \times R_2 / (R_1 + R_2)$$

3. The readings we got from the Arduino ADC were much close with the multimeter readings, showing the setup was reliable enough for basic measurements
4.There were some slight differences in the numbers, but that's mostly due to things like the multimeter’s internal resistance, some loose jumper wire connections, and general measurement errors while we were turning the knob.

B. Plots and Measurements

Formulas used to calculate:

$$V_{out} = \mathbf{V_{in} \times R_2/(R_1 + R_2)}$$

$$\% \text{ Error} = \mathbf{|V_{measured} - V_{calculated}| / V_{calculated} \times 100}$$

	V_{in}	R_1 (kΩ)	R_2 (kΩ)	V_{out} (Arduino)	V_{out} (Actual)	Error(%)
1	5	1	1.02	2.96	2.92	1.35
2	5	1	0.39	1.63	1.49	8.58
3	5	1	1.35	3.48	3.33	4.31

C. Summary of Results

1.The lab showed us how to use an Arduino to act like a voltmeter by building a simple voltage divider with a 1kΩ resistor.
2.We proved that you can control output voltage just by changing the resistance ratio.
3.The experimental results closely followed the expected theoretical trend, validating the Voltage divider circuit.
4.Errors in measurement ranged from 2.07% to 2.58% primarily due to Loose connections in breadboard circuit , multimeter reading errors and Arduino ADC fluctuations.

D. Key Findings

1.We successfully proved the voltage divider rule. The output voltage went up or down directly with the potentiometer's resistance (R_2), which is exactly what we expected.
2.The standard formula $V_{out} = V_{in} \times R_2 / (R_1 + R_2)$ worked really well to predict how the circuit would behave in real life.
3.We noticed that using a breadboard is super easy for setup, but it’s not perfect. The loose connections definitely messed with our accuracy and probably caused most of our percentage error.

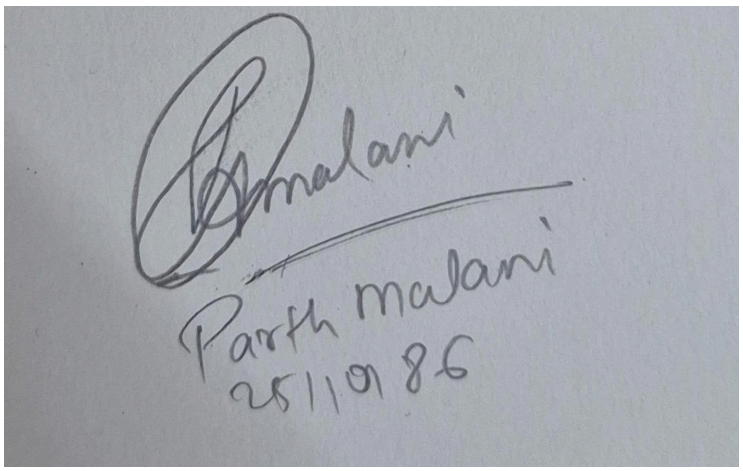
4.Even though there was fluctuation in the numbers, the Arduino stayed consistent with the multimeter readings throughout the whole test.
5.Arduino UNO’s built-in ADC was able to measure analog voltages close enough to multimeter readings, making it suitable for basic voltage sensing (0-5V) applications.

4. Bibliography

[Wikipedia Voltage Divider]

ES 116 Lab manual

Name and Signature with Date



Name and Signature with Date