Assignment–1

Name: Deepak Bharti

Roll No.: 240328

Department: EE

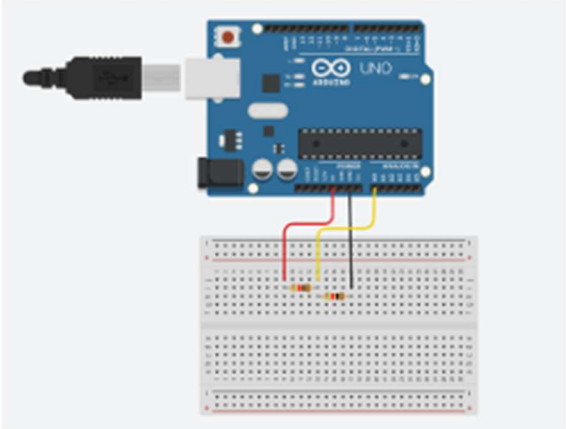
1. Introduction

Measurement of electrical quantities is a fundamental requirement in electrical and electronic systems. In this experiment, a simple voltage divider circuit is designed and interfaced with an Arduino Uno to understand how voltage measurement is performed in a digital multimeter. The task focuses on converting an analog voltage into a digital value using the Arduino's ADC and comparing theoretical and measured results.

2. Objective

The objective of this experiment is to design a voltage divider using two resistors, measure the divided voltage using Arduino analog pin A0, and verify the accuracy of the measurement by comparing theoretical and simulated values.

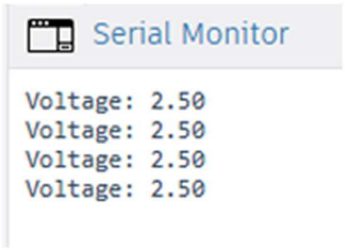
Task A — Voltage Divider Analysis and Measurement Module:



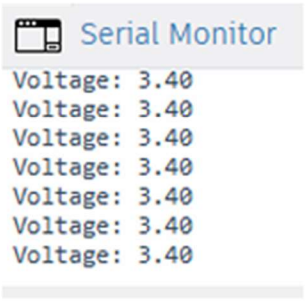
Text

```
// C++ code
//
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int adc=analogRead(A0);
  float voltage=adc*(5.0/1023.0);
  Serial.print("Voltage: ");
  Serial.println(voltage);
  delay(1000);
}
```

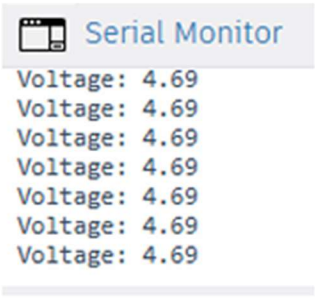
For R1=R2=10 kΩ



For R1=4.7kΩ & R2=10kΩ



For R1=1kΩ &R2=15kΩ

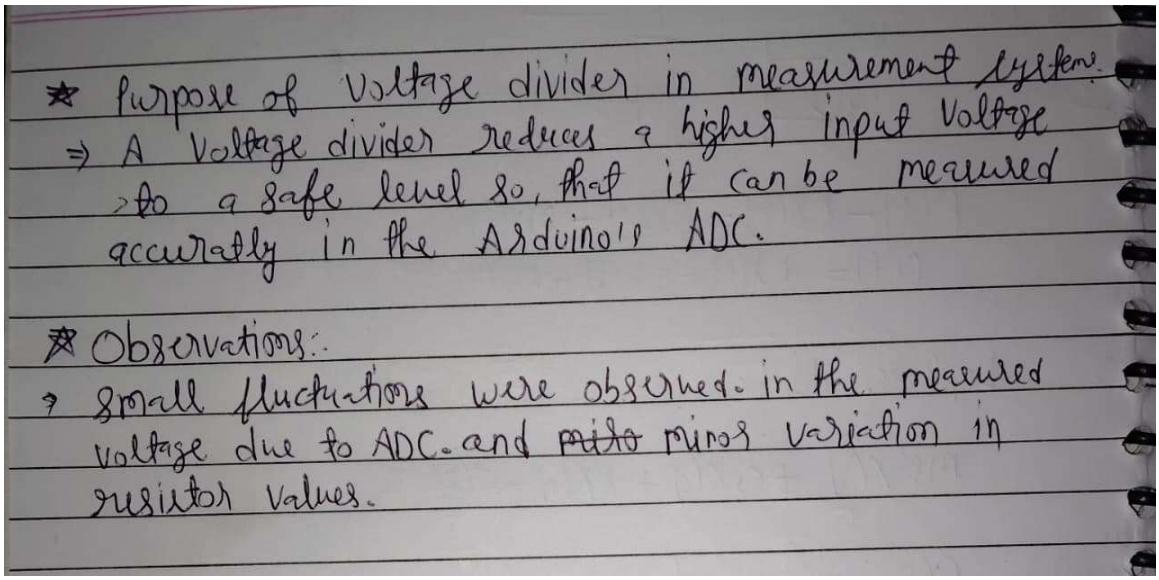


-ADC to Voltage Conversion Formula:(10 bits) V=ADC*(5/1023)

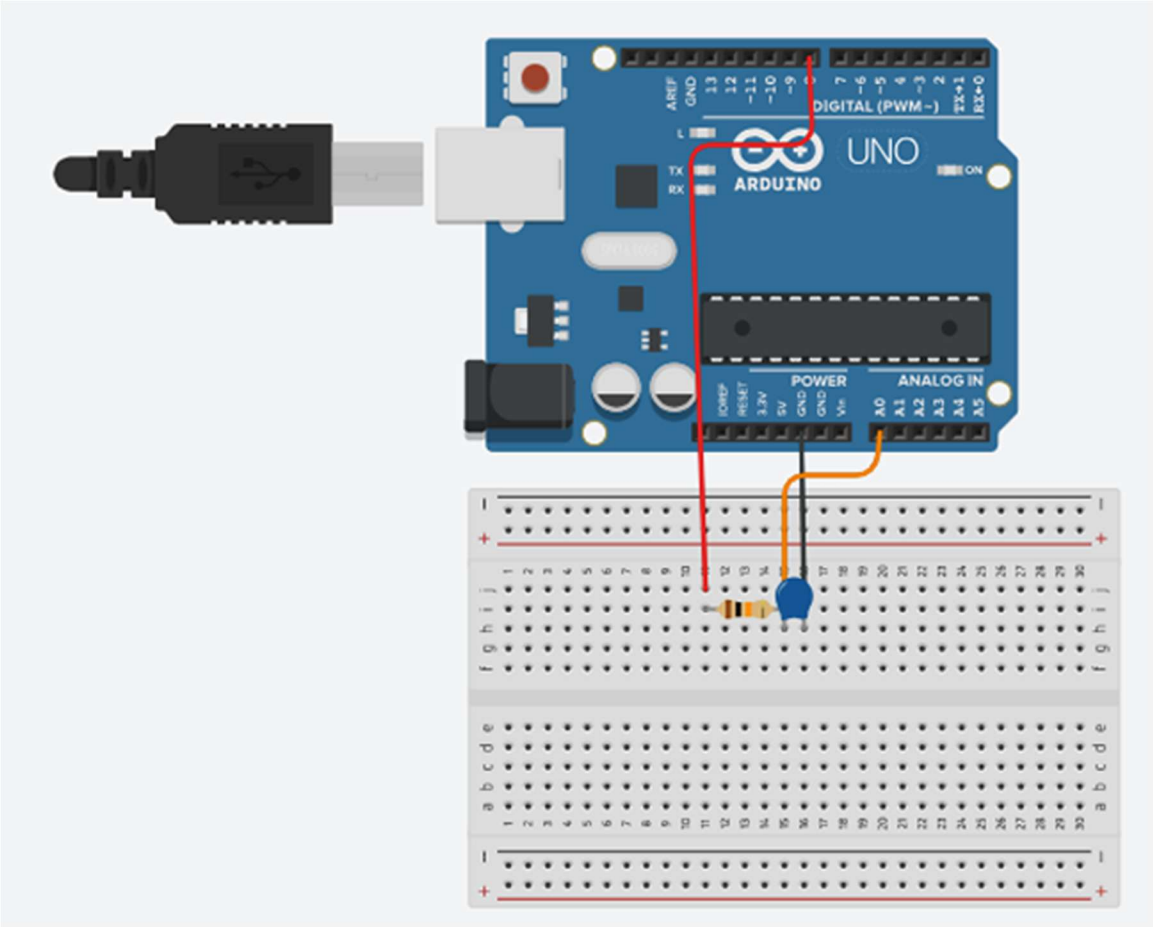
- Voltage Divider Formula: Vout = Vin*[R2/(R1+R2)]

Resistance values	Theoretical	Tinker cad Measured
R1=10kΩ R2=10kΩ	2.5 V	2.5 V
R1=4.7kΩ R2=10kΩ	3.4013 V	3.4 V
R1=1kΩ R2=15kΩ	4.6875 V	4.69 V

Measured output voltages closely match the theoretical values calculated using the voltage-divider formula. Small deviations are observed due to the Arduino's 10-bit ADC resolution, electrical noise, and resistor tolerances etc.

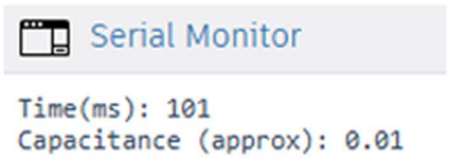


Task 2— Capacitance Measurement Using RC Time Constant:

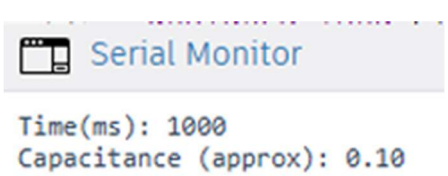


```
const int chargePin=8;
const int analogPin=A0;
const float R=10000.0;

void setup(){
  Serial.begin(9600);
  pinMode(chargePin,OUTPUT);

  digitalWrite(chargePin,LOW);
  delay(2000);

  digitalWrite(chargePin,HIGH);
  unsigned long startTime = millis();

  while(analogRead(A0)<0.63*1023){
    // wait
  }

  unsigned long elapsed=millis()-startTime;
  float C=elapsed/R;    // in millisecond/ohm
   Serial.print("Time(ms): ");
  Serial.println(elapsed);
  Serial.print("Capacitance (approx): ");
  Serial.println(C);
}

void loop() {}
```

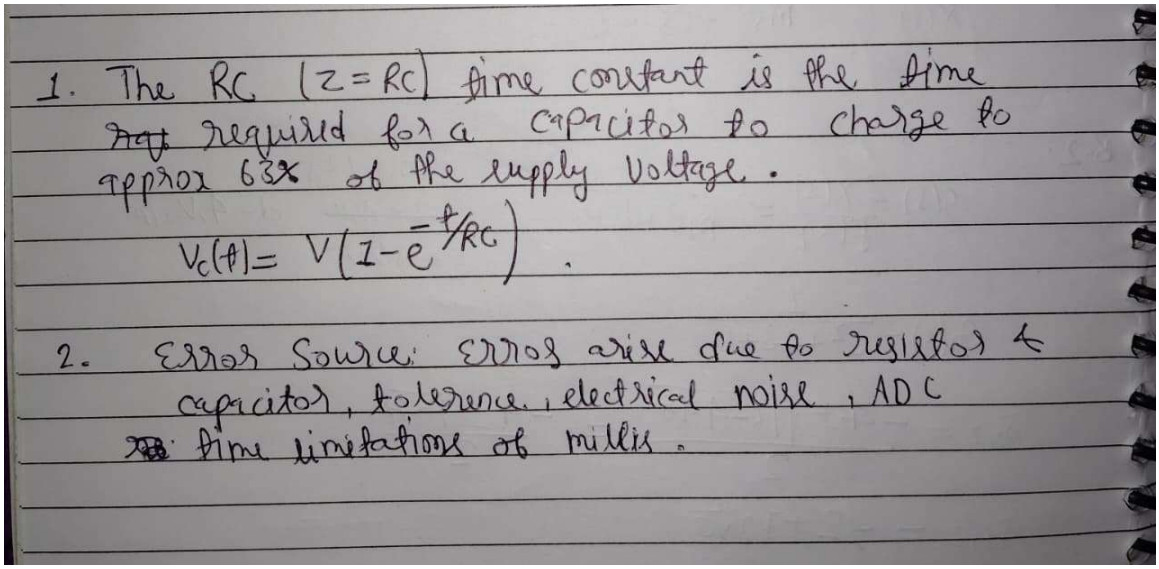
For C=10 µF = 0.01 ms/Ω



For C=100 µF = 0.1 ms/Ω



Capacitance Value	Measured Time	Expected Time (RC)
10 µF	101 ms	100 ms
100 µF	1000 ms	1000 ms

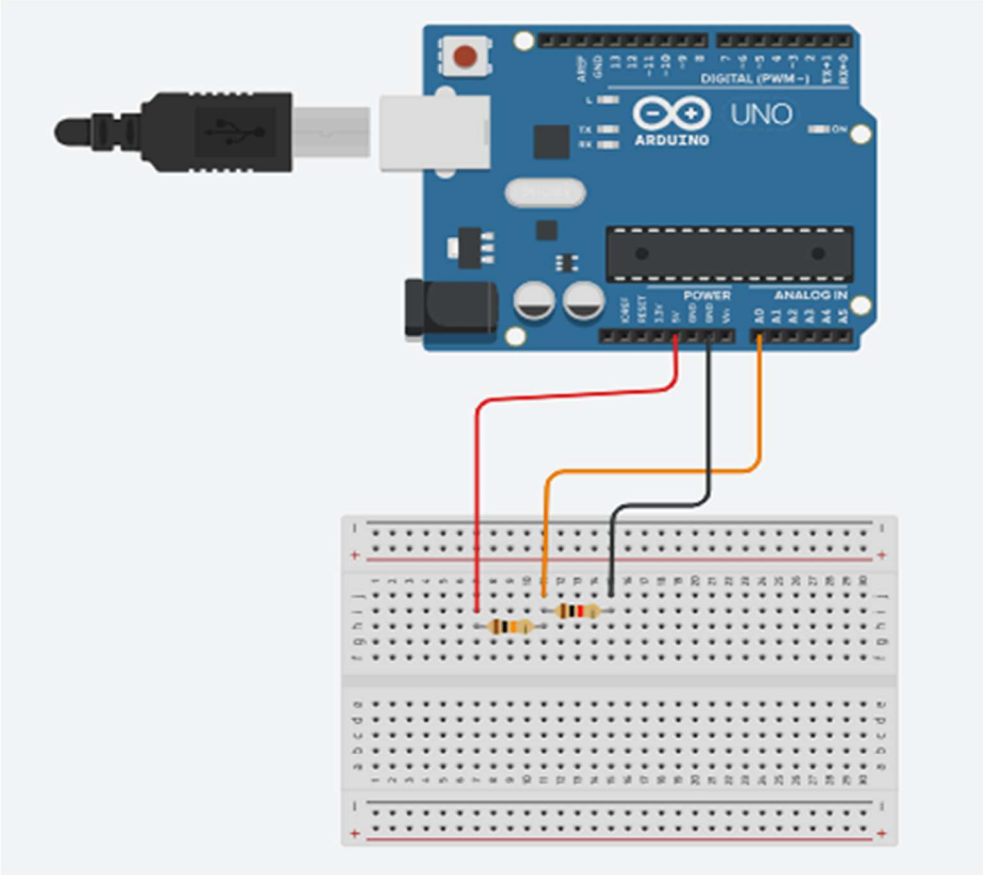


1. The RC ($\tau = RC$) time constant is the time required for a capacitor to charge to approx 63% of the supply voltage.

$$V_c(t) = V\left(1 - e^{-t/RC}\right).$$

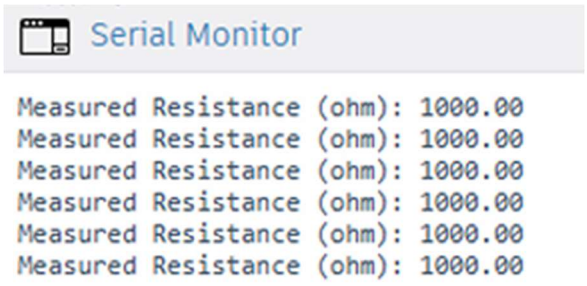
2. Error Source: Errors arise due to resistor & capacitor, tolerance, electrical noise, ADC time limitations of millis.

Task C — Beginner Ohmmeter Prototype:

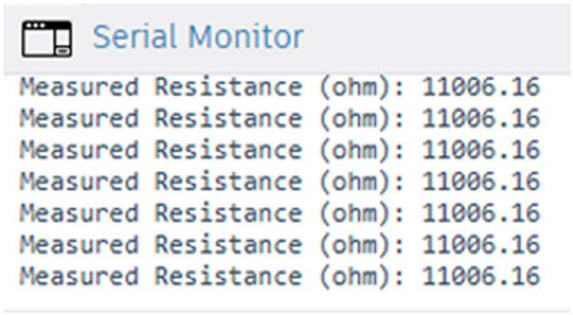



```
// C++ code
const int analogPin = A0;
const float R1 = 10000.0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int adc = analogRead(analogPin);
  float Vout = adc * (5.0 / 1023.0);

  float Rx = R1 * (Vout / (5.0 - Vout));

  Serial.print("Measured Resistance (ohm): ");
  Serial.println(Rx);

  delay(1000);
}
```

For R1=10kΩ,Rx=Unknown:



For R1=10kΩ,Rx=Unknown:



For R1=10kΩ,Rx=Unknown:

Known R1	Unknown Rx (Actual)	Unknown Rx (Measured)
10kΩ	1kΩ	1kΩ
10kΩ	11kΩ	11.00616 kΩ
10kΩ	4.2kΩ	4.20833 kΩ

1. Calculation

The Arduino first converts the ADC reading into voltage using ADC formula

ADC to Voltage conversion formula:-

$$V_{out} = ADC \times \left(\frac{5}{1023}\right)$$

The unknown resistance is calculated by ~~rearranging~~ rearranging the voltage divider.

$$V_{out} = V_{in}\left(\frac{R_2}{R_1 + R_2}\right)$$

2. Measurement Uncertainity

small errors, occur due to resistor tolerance, ADC resolution, electrical noise.