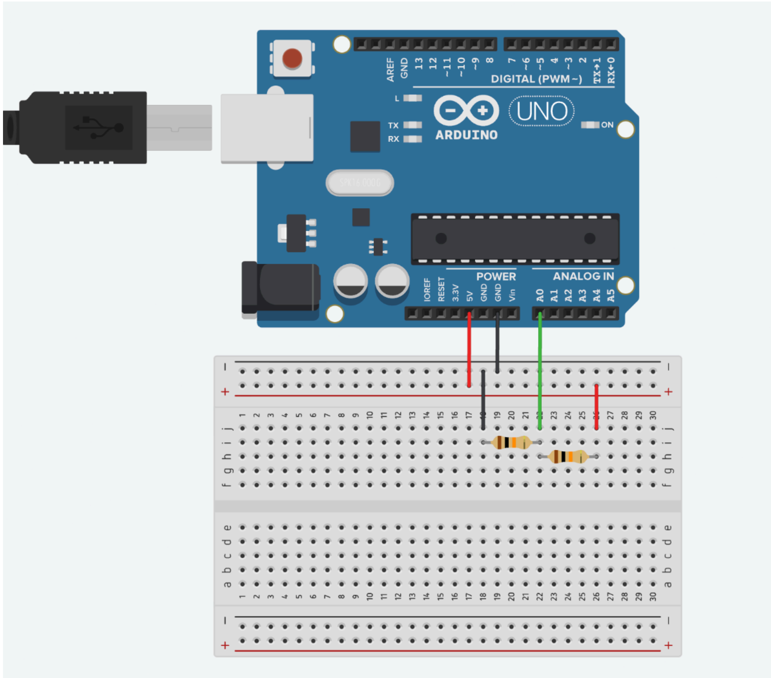
Assignment 1

Adish Kapate(240046)

TASK A -VOLTAGE DIVIDER ANALYSIS FOR ESTIMATING VOLTAGE

1)10kohm-10kohm



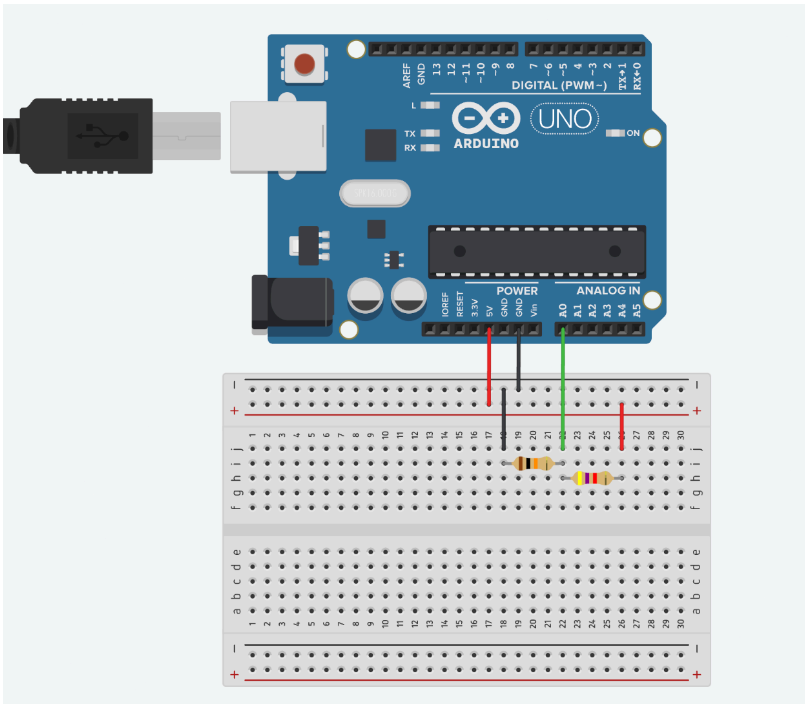
Text | 1 (Arduino Uno R3)

```
const int analogPin = A0;

const float V_ref = 5.0;

const float V_in = 5.0;

const float R1 = 10000.0;   // Resistor connected to 5V (ohms)
const float R2 = 10000.0;   // Resistor connected to GND (ohms)

void setup() {
  Serial.begin(9600);   // Open Serial Monitor
}

void loop() {

  // Read ADC value (0 to 1023)
  int ADCvalue = analogRead(analogPin);

  // Convert ADC value to measured voltage
  float V_out = (ADCvalue / 1023.0) * V_ref;

  // Theoretical voltage using voltage divider formula
  float V_theoretical = V_in * (R2 / (R1 + R2));


  Serial.print("ADC Value: ");
  Serial.print(ADCvalue);
  Serial.print(" | Measured V_out: ");
  Serial.print(V_out);
  Serial.print(" V | Theoretical V_out: ");
  Serial.print(V_theoretical);
  Serial.println(" V");
```

Serial Monitor

```
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50 V
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50 V
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50 V
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50 V
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50 V
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50 V
ADC Value: 511 | Measured V_out: 2.50 V | Theoretical V_out: 2.50
```

2)4.7kohm-10kohm



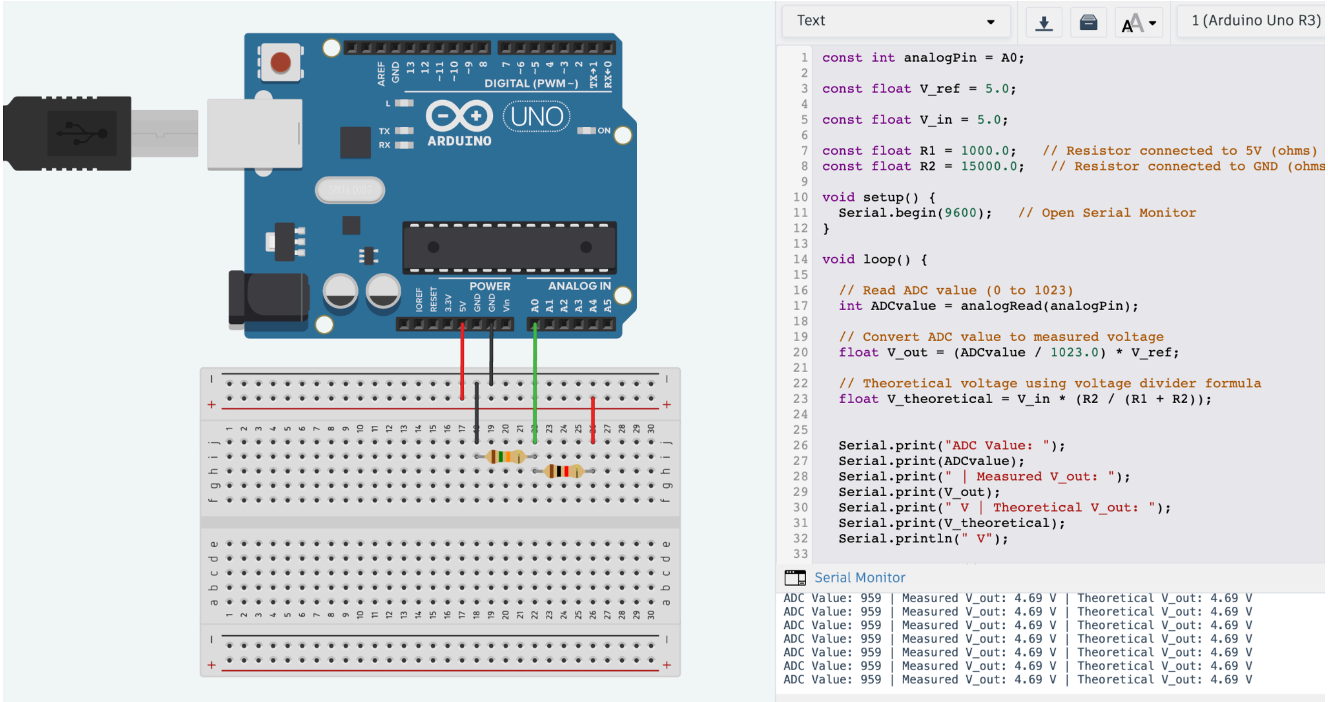
Text | 1 (Arduino Un

```
const int analogPin = A0;

const float V_ref = 5.0;

const float V_in = 5.0;

const float R1 = 4700.0;    // Resistor connected to 5V (o
const float R2 = 10000.0;    // Resistor connected to GND

void setup() {
  Serial.begin(9600);   // Open Serial Monitor
}

void loop() {

  // Read ADC value (0 to 1023)
  int ADCvalue = analogRead(analogPin);

  // Convert ADC value to measured voltage
  float V_out = (ADCvalue / 1023.0) * V_ref;

  // Theoretical voltage using voltage divider formula
  float V_theoretical = V_in * (R2 / (R1 + R2));


  Serial.print("ADC Value: ");
  Serial.print(ADCvalue);
  Serial.print(" | Measured V_out: ");
  Serial.print(V_out);
  Serial.print(" V | Theoretical V_out: ");
  Serial.print(V_theoretical);
  Serial.println(" V");
```

Serial Monitor

```
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
ADC Value: 696 | Measured V_out: 3.40 V | Theoretical V_out: 3.40 V
```

3)1kohm-15kohm



Results:

	Thereotical	Tinkercad
1)	2.5V	2.50V
2)	3.4013 V	3.40V
3)	4.6875 V	4.69V

1. Purpose of a Voltage Divider in Measurement Systems

- A voltage divider is used to reduce an input voltage to a level that can be safely measured.
- It enables measurement of higher voltages using low-voltage ADCs.
- Voltage dividers are commonly used in digital multimeters for range scaling.

2. ADC Reading-to-Voltage Conversion Formula

- The Arduino Uno uses a 10-bit ADC with output values from 0 to 1023.
- For this experiment, the reference voltage VrefV_{ref}Vref is 5 V.

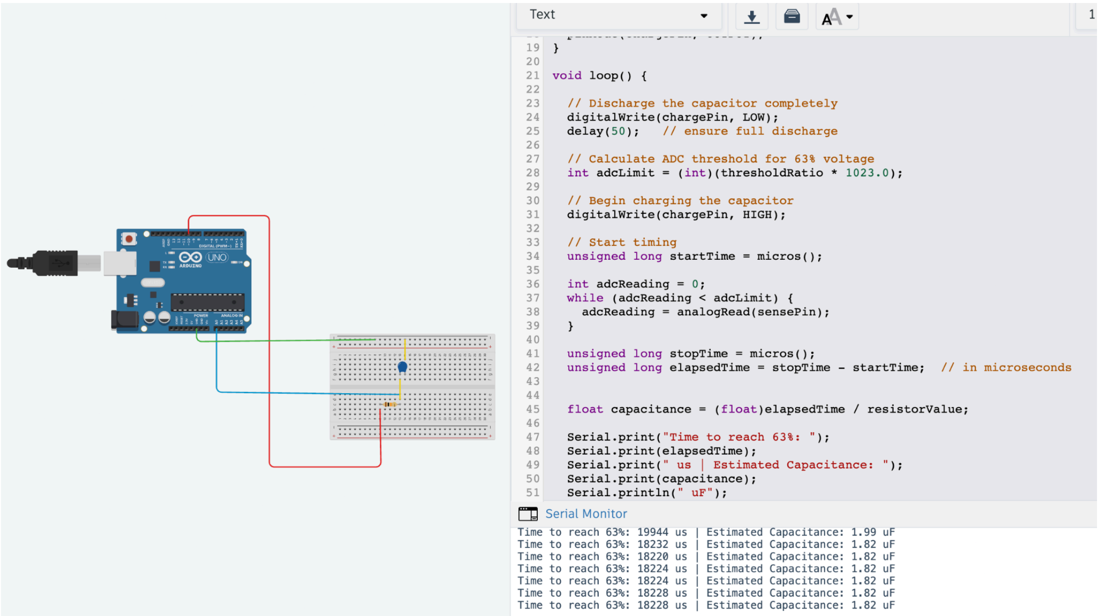
- $$V_{out} = \frac{\text{ADC Reading} \times V_{ref}}{1023}$$

3. Observations (Error, Noise, Unexpected Results)

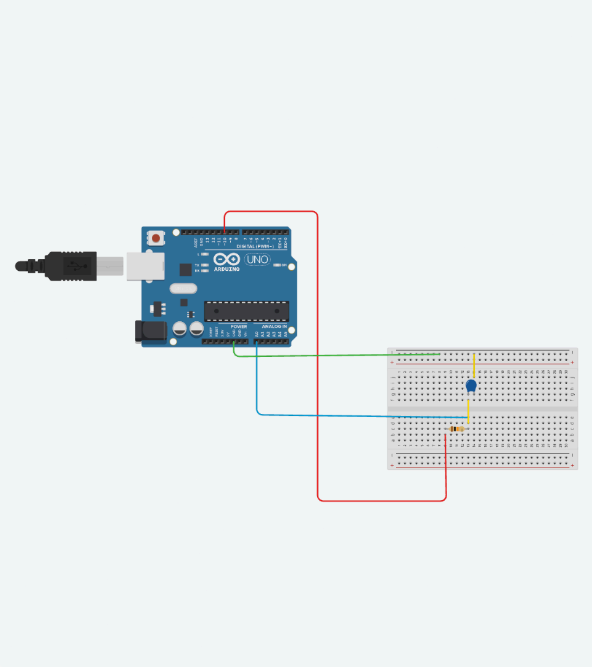
- Measured voltages were close to theoretical values for all resistor combinations.
- Small errors occurred due to ADC resolution and resistor tolerances.
- Minor noise and fluctuations were observed in the simulated readings.

TASK B – CAPACITANCE MEASUREMENT FROM RC TIME CONSTANT

1)For C=2uf



2)For C=1uf



```
}

void loop() {

  // Discharge the capacitor completely
  digitalWrite(chargePin, LOW);
  delay(50);   // ensure full discharge

  // Calculate ADC threshold for 63% voltage
  int adcLimit = (int)(thresholdRatio * 1023.0);

  // Begin charging the capacitor
  digitalWrite(chargePin, HIGH);

  // Start timing
  unsigned long startTime = micros();

  int adcReading = 0;
  while (adcReading < adcLimit) {
    adcReading = analogRead(sensePin);
  }

  unsigned long stopTime = micros();
  unsigned long elapsedTime = stopTime - startTime;  // in microseconds


  float capacitance = (float)elapsedTime / resistorValue;

  Serial.print("Time to reach 63%: ");
  Serial.print(elapsedTime);
  Serial.print(" us | Estimated Capacitance: ");
  Serial.print(capacitance);
  Serial.println(" uF");
```

Serial Monitor

```
Time to reach 63%: 9972 us | Estimated Capacitance: 1.00 uF
Time to reach 63%: 9908 us | Estimated Capacitance: 0.99 uF
Time to reach 63%: 9908 us | Estimated Capacitance: 0.99 uF
Time to reach 63%: 9900 us | Estimated Capacitance: 0.99 uF
Time to reach 63%: 9920 us | Estimated Capacitance: 0.99 uF
Time to reach 63%: 9904 us | Estimated Capacitance: 0.99 uF
Time to reach 63%: 9916 us | Estimated Capacitance: 0.99 uF
```

Code:

```
// Pin used to charge the capacitor
const int chargePin = 10;

// Analog pin used to sense capacitor voltage
const int sensePin = A0;

// Known resistor value (Ohms)
const long resistorValue = 10000;   // 10kΩ

// Arduino reference voltage
const float referenceVoltage = 5.0;

// Fraction of final voltage for time constant (63%)
const float thresholdRatio = 0.63;

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

void loop() {

  // Discharge the capacitor completely
  digitalWrite(chargePin, LOW);
  delay(50);   // ensure full discharge

  // Calculate ADC threshold for 63% voltage
  int adcLimit = (int)(thresholdRatio * 1023.0);

  // Begin charging the capacitor
  digitalWrite(chargePin, HIGH);

  //| Start timing
  unsigned long startTime = micros();

  int adcReading = 0;
  while (adcReading < adcLimit) {
    adcReading = analogRead(sensePin);
  }

  unsigned long stopTime = micros();
  unsigned long elapsedTime = stopTime - startTime;  // in microseconds
```

```
  float capacitance = (float)elapsedTime / resistorValue;

  Serial.print("Time to reach 63%: ");
  Serial.print(elapsedTime);
  Serial.print(" us | Estimated Capacitance: ");
  Serial.print(capacitance);
  Serial.println(" uF");

  delay(1000);
}
```

1. RC Time Constant and Significance of 63%

- The RC time constant (τ=RC) represents the time taken by a capacitor to charge to approximately 63% of its final voltage.
- The 63% level is significant because it simplifies the exponential charging equation and allows direct calculation of capacitance.

2. Measured Time vs. Expected Time

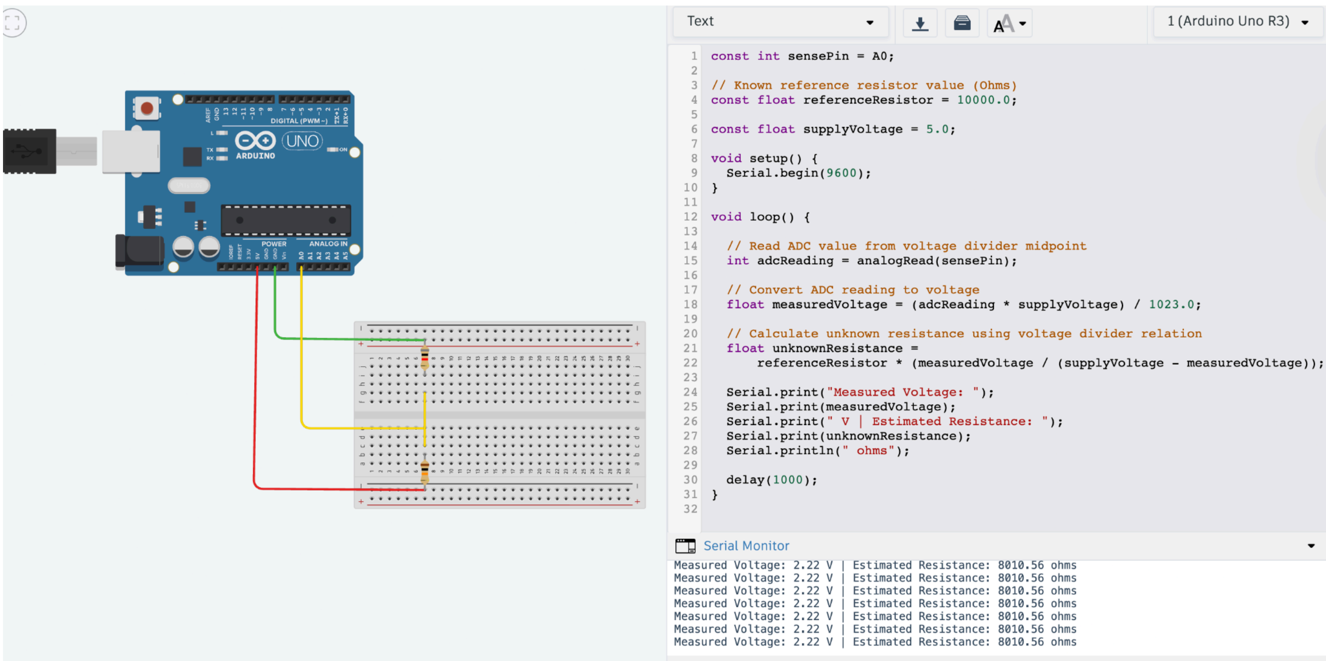
- The charging time to reach the threshold voltage was measured using the Arduino timer.
- Measured times were compared with expected RC time constants using tabulated results.

3. Sources of Error

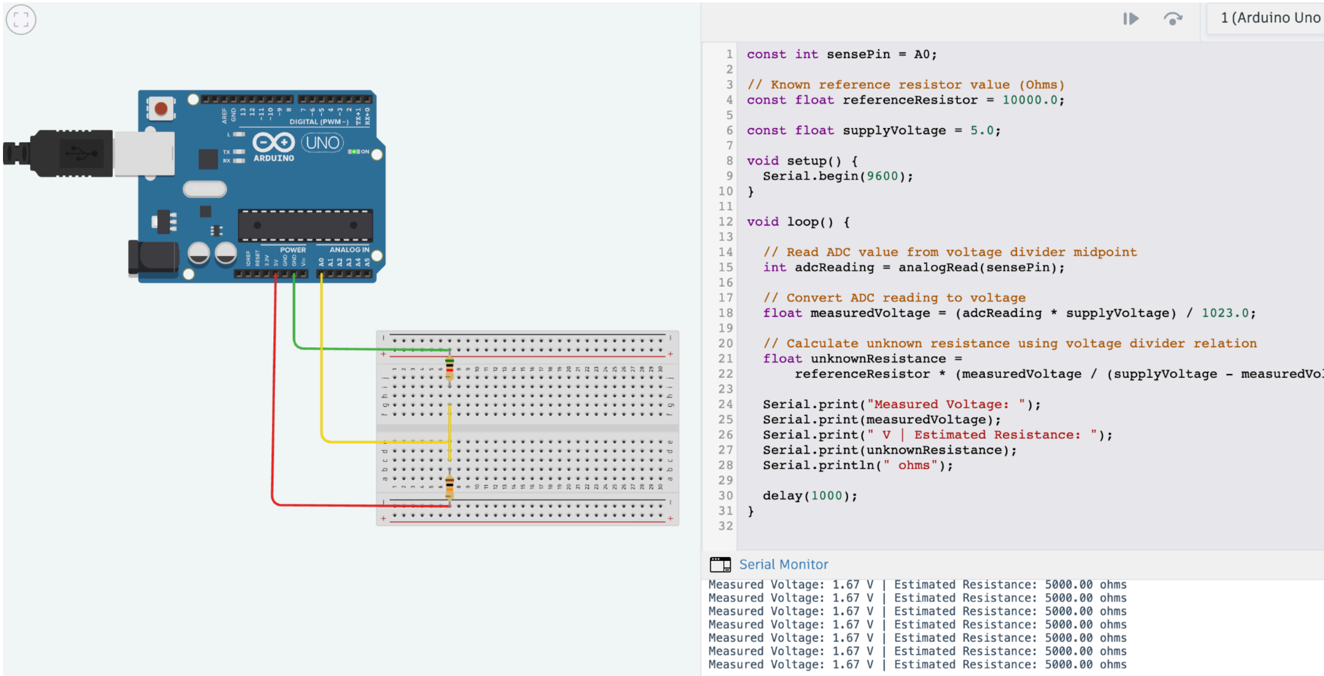
- Capacitor tolerance leads to deviations from nominal capacitance values.
- ADC resolution limits the accuracy of voltage threshold detection.
- Timing inaccuracy and simulation noise introduce minor measurement errors.

TASK C –BASIC OHMETER USING VOLTAGE DIVIDER METHOD

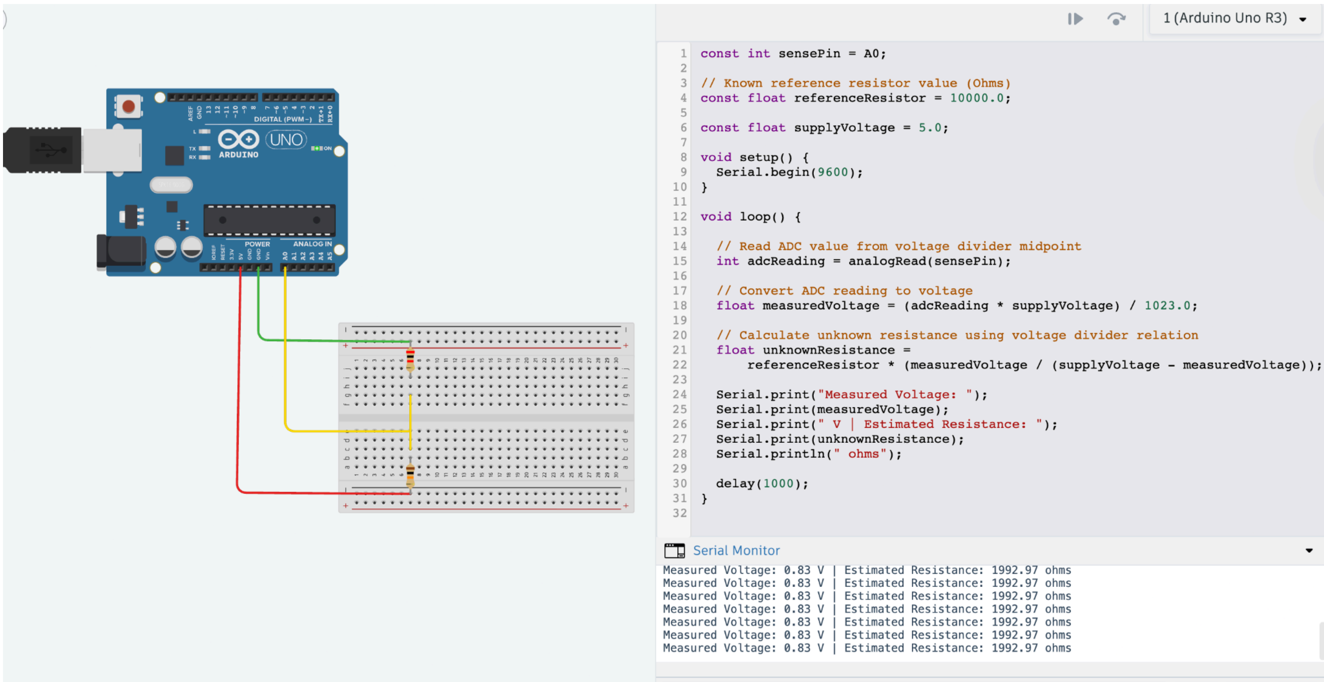
1)R=8kohm



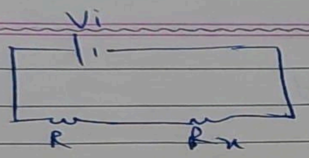
2)R=5kohm



3)R=2kohm



Calculation:



$$V_{out} = \frac{V_i \times R_x}{(R + R_x)}$$

$$V_{out}(R + R_x) = V_i \times R_x$$

$$R_x = \frac{V_{out} \times R}{V_i - V_{out}}$$

for $R = 10\,k\Omega$, $V_i = 5V$

i) $R_x = 8\,k\Omega$

$$V_{out} = \frac{5 \times 8}{10+8} \cong 2.22V.$$

ii) $R_x = 5\,k\Omega$

$$V_{out} = \frac{5 \times 5}{10+5} \cong 1.66V$$

iii) $R_x \cong 2\,k\Omega$

$$V_{out} = \frac{5 \times 2}{10+2} \cong 0.833V$$

Table:

R expected (kΩ)	R measured (kΩ)
8 kΩ	8.01056 kΩ
5 kΩ	5 kΩ
2 kΩ	1.99297 kΩ

V theoretical	V Tinker
2.22	2.22
1.66	1.67
0.83	0.83

Reflection on measurement uncertainty

The ADC has 10-bit resolution, so each step is about 4.9 mV at a 5 V reference, which limits how precisely voltage and resistance can be measured. The actual 5 V supply and 4.7 kQ reference resistor may deviate from their nominal values, introducing additional error. Breadboard contacts, wiring resistance, and electrical noise can slightly change the voltage seen at AO. Real resistors have tolerance (for example +,-(5% or 1%)), so their true value can naturally differ from the printed value even with perfect measurement.
Overall, the measured resistances are extremely close to the nominal values, so any tiny differences fall within normal tolerance and expected measurement uncertainty.