Arduino Ohmmeter

A project by Carlos Ricoveri

Equipment

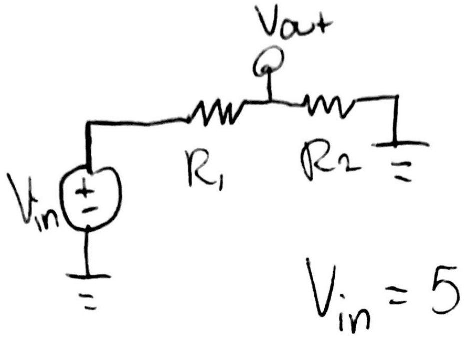
- Arduino Mega2560
- Macbook Pro running the Arduino IDE
- Breadboard
- A 10k ohm reference resistor
- Jumper cables

Introduction

I was in the middle of a personal project where I encountered the annoyance of having to read the resistor bands to determine their resistance. Over the years as a Computer Engineering student, I've added quite a bounty of different resistors to my stock. As expected, it's very hard to keep these organized and while a regular multimeter could fix my dilemma, I found it extremely difficult to go out and shop without being somewhat exposed to the pandemic. I didn't want to wait for it to come through the mail either. As a result, I did what any engineer would do in this situation: I decided to build one.

Analysis

Before I began building the circuit for the ohmmeter, I decided to make sure all of the math behind it checked out. The circuit for the ohmmeter was a very simple one that only required voltage divider.



$$V_{in} = 5\ V$$
$$R_1 = 10K\ \Omega$$

$$\frac{V_{in} - V_{out}}{R_1} = \frac{V_{out}}{R_2}$$

$$V_{out} = \frac{V_{in} R_2}{R_1 + R_2}$$ → Voltage Divider

R1 in this case is our reference resistor. I decided to go with a 10k ohm resistor. This value, theoretically, could have any resistance as long as it is updated in the code accordingly. By shifting the variables around in the Vout equation, the equation for R2 can be obtained. R2 would essentially be the resistor that we would want to measure on the board.

Code

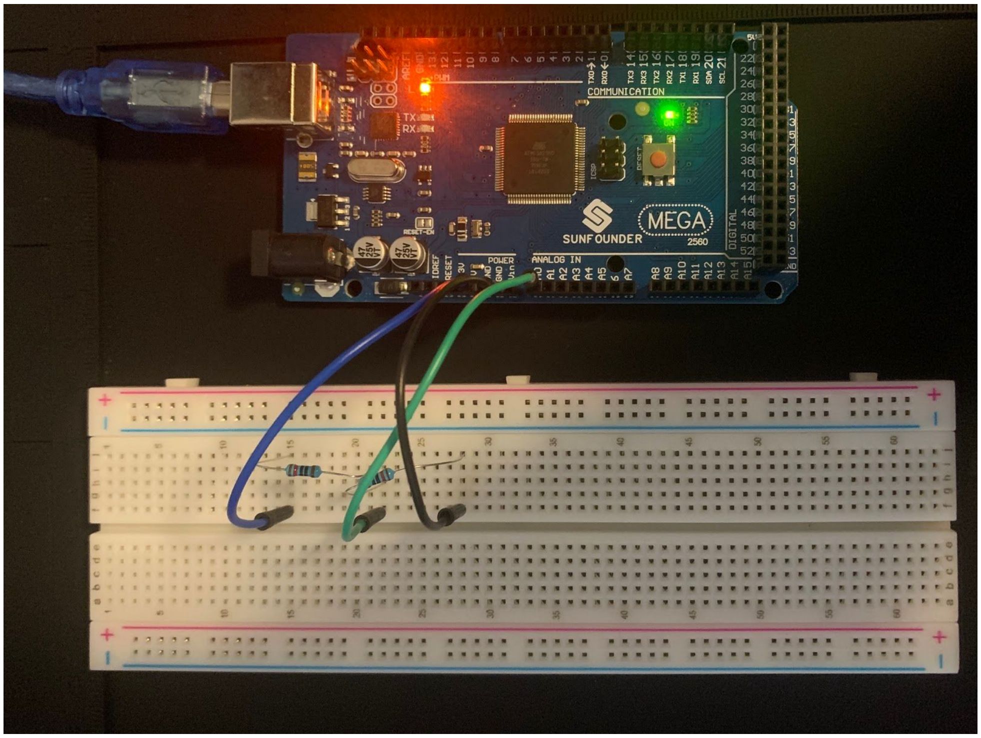
Within the code, I set it so that the resistance would be printed to the screen via Board-PC serial communication.

```
// Carlos R.
// ATMega2560 Ohmmeter.

const float vin = 5.0;          // Input voltage used from the Arduino.
const float resRef = 10000.0;   // Reference resistor of 10k ohms.
float r = 0.0;                  // Measured resistance.
float vout = 0;                 // Measured voltage from Analog In pin.
int readPin = A0;

void setup()
{
  // Use a standard 9600 baud rate.
  Serial.begin(9600);
}

void loop()
{
  // Analog read reads a percentage value (x / 1024).
  // To get the reading you have to multiply by Vin.
  vout = analogRead(readPin)/1023.0 * vin;

  /* For debugging purpose:
  String inp = String(vout * vin);
  Serial.println("Measured voltage: " + inp + " volts.");
  */

  // Using voltage divider:
  // Vout = (Vin * r) / (refRes + r)
  r = (resRef * vout)/(vin - vout);

  Serial.print("Resistance is: ");
  Serial.print(r);
  Serial.println(" ohms.");

  // 2.5 sec delay to allow measurements to be read.
  delay(2500);
}
```

Procedure

1. Connected board's ground to one of the rails on the breadboard.
2. Connected the 5V from the board to the breadboard.
3. Connected one of the Analog In pins (A0) to the breadboard.
4. Connected the 10k ohm reference with one end on the same node as the 5V input and the other end to the node as the A0 pin.

At this stage the ohmmeter should be set up to read any resistor you connect. To test a resistor, just connect one end of the resistor to the same node as the A0 pin, and the other end to ground.



Ohmmeter % Error

I wanted to identify the working range for my ohmmeter. I tested the device four different resistance values: 1k, 10k, 100k, and 1M ohms. Each resistance value was tested with 6 different resistors of the same resistance. The tables below show the observed resistances. The percent error was then calculated using the calculated averages.

1K:

Resistor	Resistance (Ohms)
1	964.63
2	964.63
3	964.63
4	964.63
5	964.63
6	964.63
Average	964.63

Percent Error: 3.5%

10K:

Resistor	Resistance (Ohms)
1	9980.47
2	9941.52
3	9941.52
4	9902.72
5	9902.72
6	9980.47
Average	9941.57

Percent Error: 0.58%

100K:

Resistor	Resistance (Ohms)

1	101195.64
2	102417.57
3	101195.64
4	102417.57
5	101195.64
6	101195.64
Average	101602.95

Percent Error: 1.6%

1M:

Resistor	Resistance (Ohms)
1	1126667.75
2	1126667.75
3	1126667.75
4	1126667.75
5	1126667.75
6	1126667.75
Average	1126667.75

Percent Error: 12.7%

By analysing the % error of the ohmmeter, it can be seen that the optimal working range with the least % error is [10k ohms, 100k ohms]. Extremely high or low resistances will give off wrong readings. Anything under 1k ohms was pretty much unreadable.

The observed minimum percent error was 0.13% for a resistance of 30k ohms. This was done by connecting three 10k ohm resistors in series.