

Teaching Practice

Using Arduino in Physics Teaching: Arduino-based Physics Experiment to Study Temperature Dependence of Electrical Resistance

Uğur SARI *¹ , Talip KIRINDI ² 

¹ Kırıkkale University, Education Faculty, Department of Mathematics and Science Education, Turkey, usari05@yahoo.com

² Kırıkkale University, Education Faculty, Department of Mathematics and Science Education, Turkey, talipkirindi@yahoo.com

* Corresponding Author: usari05@yahoo.com

Article Info

Received: 18 June 2019

Accepted: 23 August 2019

Keywords: Physics education, Arduino platform, Arduino experiment, electrical resistance, temperature coefficient

DOI: 10.18009/jcer.579362

Publication Language: English

Abstract

Nowadays, the rapid developments in science and technology have an impact on education as well as on all gear units. The integration of technology into the education process offers effective opportunities, particularly in the fields of STEM (Science, Technology, Engineering and Mathematics). In this context, Arduino platform has started to be used as a low cost, easy to use microcontroller in physics teaching. In this study, an Arduino-based physics experiment that can be used in physics laboratories was designed. We have used Arduino platform to study temperature dependence of electrical resistance and to define the temperature coefficient for a conductor. Experimental data were collected using the Arduino ohmmeter and a temperature sensor. The resistance-temperature curve obtained is in the expected character and the temperature coefficient is in the expected value. With this experiment, the student can easily observe the temperature change of the conductor's resistance and calculate the temperature coefficient of the resistance. The Arduino-based apparatus is presented as a simple and low cost alternative to physics laboratories.



To cite this article: Sari, U. & Kirindi, T. (2019). Using arduino in physics teaching: arduino-based physics experiment to study temperature dependence of electrical resistance. *Journal of Computer and Education Research*, 7 (14), 698-710. DOI: 10.18009/jcer.579362

Introduction

Laboratory activities in the teaching of physics are important as students are involved in the process of learning and exploring with firsthand experience. Laboratory experiences enable students to understand the functioning of the laws of physics, to recognize, understand and reinforce the concepts of physics and to develop scientific skills (Darrah, Humbert, Finstein, Simon & Hopkins, 2014; Sari, Pektaş, Çelik & Kirindi, 2019). Nowadays, technology is developing rapidly and technological products have taken place in physics laboratories, and experimental physics has reached an impressive point. In particular, microprocessors, mobile data collection tools, sensors and analogue instruments have been integrated into a single system, providing significant convenience in data collection, data processing and visualization processes (Chen et al., 2012; Russell, Lucas & McRobbie 2004).

In this context, Arduino Board, based on a microprocessor, has been preferred in physics experiments with its low cost, flexible and easily applicable structure and quick data collection advantages (Atkin, 2016; Pereira, 2016; Tunyagi, Kandrai, Fülöp, Kapusi & Simon, 2018).

Electrical resistance and its temperature dependence in conductors are included in physics curricula at secondary, high school and university levels. For example, in Turkey, the electrical resistance and factors affecting the electrical resistance (cross-sectional area, length, type of conductor) are included in the 8th grade secondary (MoNEa, 2018). Temperature-dependent change of conductor resistance takes place in the 10th grade high school curriculum (MoNEb, 2018). Similar topics are included in physics courses in many numerically university programs. In addition, the factors affecting the resistance such as length, type and cross-sectional area of conductor are examined and the experiments related to these effects are carried out widely in schools. Thus, students can easily comprehend and apply these factors. As another effect on the resistance of the conductor, the temperature effect in the physics curricula cannot be studied experimentally enough. Therefore, students have difficulty in understanding the effect of temperature on resistance. The reason that the physics experiments related to the temperature-resistance relationship cannot be done very often may be the complex structure of the test equipment such as heating element, current module, resistance measurement module and not being as cheap as each school can provide. In this study, a simple and economical Arduino-based physics experiment is presented, which examines the variation of the resistance of a copper wire with temperature. An Arduino board, an Arduino ohmmeter and a temperature sensor were used to measure the temperature-resistance relationship.

The Model of Temperature Dependent Electrical Resistivity of Metals

When a potential difference (V) is applied between the ends of a metallic conductor, an electric field is generated in relation to the potential difference. Due to this field, negative charges in the conductor move from low electrical potential to high potential. This movement of charges is called electric current (I). The direction of the electric current is considered to be the direction in which the positive charges move with the effect of the electric field. In real life, electric current in many electrical circuits is caused by the movement of negatively charged electrons. The actual direction of movement of the charged

particles is therefore the opposite of the accepted direction of current. Experiments show that there is a linear relationship between current and potential difference for many metals. This relationship known as Ohm's law is given as

$$V=IR \quad (1)$$

Ohm's law is an experimental correlation only for certain conductors and the conductors that comply with this law are called ohmic. Equation 1 can be written as

$$R = V/I \quad (2)$$

Here, R is the resistance of the conductor. The resistance for a conductor with a uniform cross section (for example, a wire or rod) can be measured by

$$R_0 = \rho_0 \frac{l}{A} \quad (3)$$

Here, l is the length of conductor, A is the cross-sectional area, and ρ_0 is the resistivity that takes into account the structure of the material in resistance (Serway, Faughn,&Vuille, 2014, pp. 66).

The main reason for the electrical resistance in conductors is that the electrons acting under the influence of the electric field collide with the atoms (ions) forming the crystal lattice. These collisions prevent the movement of electrons and electrons lose some of their kinetic energy during collision. This lost energy is transformed into heat energy in metal. In addition, foreign atoms that disrupt impurity also contribute to the resistance of metals. If the length of a conductor increases, the number of collisions of the electrons increases and thus resistance increases. When the cross-section area of the conductor (A) is larger, it contributes to carry more electron current per unit field and thus the resistance decreases. When the structure of the conductor changes (the type of atoms forming the conductor), the resistance for different materials becomes different. Another factor affecting resistance in metals is temperature. If the temperature of the metal is increased, the metal atoms vibrate stronger and the transmission electrons make more violent collisions with them, thus the resistance of the metal increases (Meaden, 2013, pp.6). If the temperature range is not too large, the resistivity in metals is a linear function of temperature and it is expressed as

$$\rho = \rho_0[1 + \alpha(T - T_0)] \quad (4)$$

Here, ρ (ohm-meters) is the resistivity at any T temperature ($^{\circ}\text{C}$), and ρ_0 is the resistivity at the reference temperature T_0 (usually 20°C). α is the temperature coefficient of the

resistivity. When equations 3 and 4 are taken into account, the temperature change of the resistance is given as

$$R = R_0[1 + \alpha(T - T_0)] \quad (5)$$

for a conductor with a uniform cross-sectional area (Meaden, 2013, pp.6; Young, Freedman, Sandin, & Ford, 1996, pp. 854). Where R_0 is the resistor of the sample at the reference temperature T_0 , and α is the temperature coefficient (or resistance change per degree). If the reference temperature is chosen as zero degree, the equation 5 will be

$$R = R_0(1 + \alpha T) \quad (6)$$

$$R = R_0\alpha T + R_0 \quad (7)$$

If this equation is compared to a linear equation,

$$y = mx + c \quad (8)$$

it is seen that this equation becomes as following

$$\alpha = \frac{\text{slope}}{R_0} \quad (9)$$

Thus, the temperature coefficient for a conductor with a uniform cross-section can be experimentally defined from the slope of the resistance-temperature graph.

Experimental Set-up

The experimental set-up for temperature dependent analysis of the resistance of a copper wire and determination of the temperature coefficient consists of 0.3 mm in diameter and 30 m in length varnished copper wire, an Arduino Uno card, a USB cable, DS18B20 waterproof temperature sensor, 320 Ω and 4.7 K Ω resistors, 8 male-male and 2 male-female jumper wires, a spiral stove, beaker and some distilled water (Figure 1).

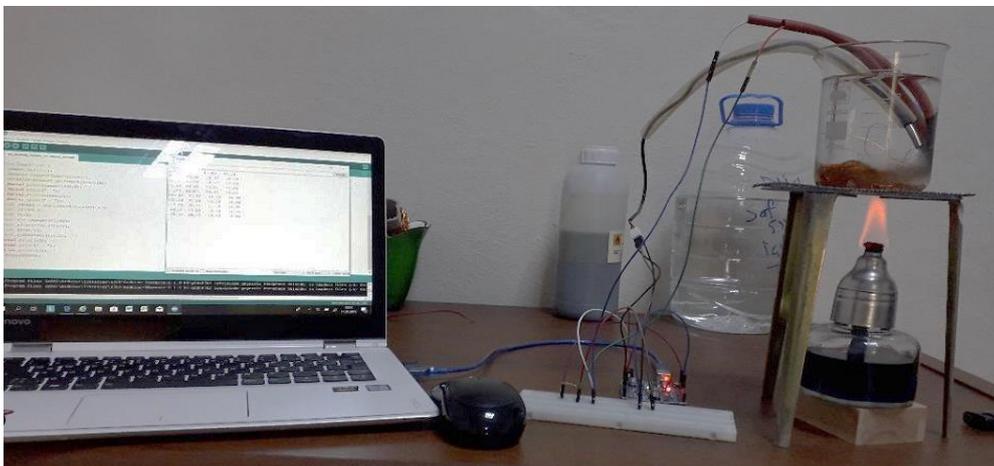


Figure 1. The picture of the experimental set-up

Firstly, an Arduino ohmmeter is prepared on the breadboard for resistance measurement. The structure and operation of the ohmmeter was given in the next part. The copper wire to be used in the experiment is mounted on the unknown resistance section in the ohmmeter apparatus. 30 m long copper wire is connected using jumper wires. Then the temperature sensor is connected to the Arduino board. The tip of the temperature sensor and the copper wire attached to the breadboard with the jumper wires are placed in a beaker with pure water. The pure water at room temperature in the beaker is heated with a spirit stove. Experimental data is collected until the temperature of the pure water is reached from room temperature to the boiling temperature. When measuring the temperature sensor, the change in resistance is observed with the Arduino ohmmeter. The resistance-temperature graph is then generated for the copper wire. The temperature coefficient is calculated by using the slope of this graph and equation 9.



Figure 2. DS18B20 waterproof temperature sensor

The DS18B20 waterproof temperature sensor used in the experiment is a digital sensor and the temperature in deep distances can be measured with the help of a 1m long cable (Figure 2). It communicates with the microprocessor using the 1-Wire terminal and can perform a 9-12 bit reading. It can measure the temperatures between $-55\text{ }^{\circ}\text{C}$ and $+125\text{ }^{\circ}\text{C}$. It is a sensitive sensor that can measure temperatures between $-10\text{ }^{\circ}\text{C}$ and $+85\text{ }^{\circ}\text{C}$ with a margin of error of $\pm 0.5\text{ }^{\circ}\text{C}$. The DS18B20 waterproof cable model allows measurements in wet environments such as water and oil, dry air and hard floors such as soil. In the sensor, the yellow cable is data (DQ) cable, the black cable is GND ground cable, and the red cable is the VCC voltage cable (www.datasheetspdf.com).

Construction of the Arduino Ohmmeter

The simplest way to measure resistance is to create a voltage divider with an output voltage that is connected to the unknown resistance. An Arduino ohmmeter is a very simple and convenient resistance meter created with a voltage divider. Figure 3 shows the structure of the Arduino ohmmeter with the circuit diagrams and simple schematic illustrating.

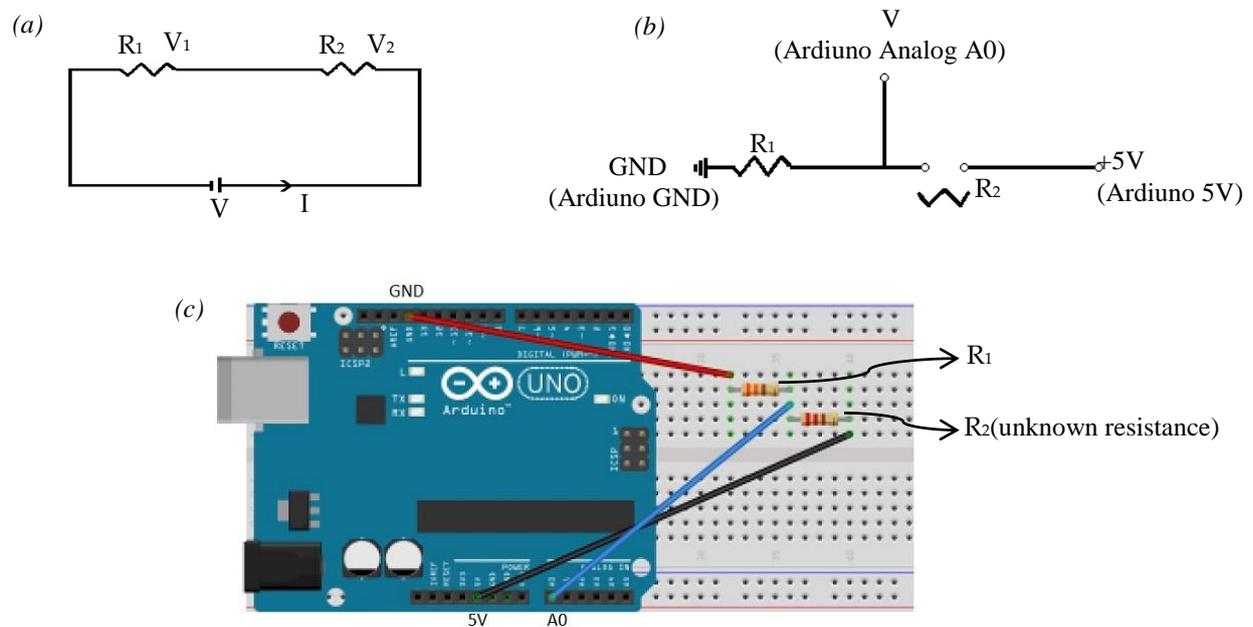


Figure 3. (a) Circuit consisting of resistors in series, (b) Arduino ohmmeter circuit diagram, (c) simple schematic illustrating how Arduino ohmmeter developed by the students is set up.

In Figure 3, the correlations can be written for closed circuit consisting of resistors in series as

$$V = I(R_1 + R_2) \quad (10)$$

$$I = V/(R_1 + R_2) \quad (11)$$

$$V_1 = IR_1 \quad (12)$$

$$V_2 = IR_2 \quad (13)$$

If the current value in equation 11 is written in equation 12 and if R_2 is left alone, the equation becomes as

$$R_2 = \frac{R_1(V - V_1)}{V_1} \quad (14)$$

It is $V = 5V$ in this equation (Figure 3). The potential value of V_1 is obtained from the value read from pin A0. The analog input A0 is a 10-bit analog to digital converter (A/D converter, ADC). This means that the input voltages between $2^{10} = 1024$ and 0-5 Volt will correspond to

the integer values between 0 and 1023. In this situation, the analog value read from A0 pin can be transformed into volt by the help of,

$$V_1 = \frac{\text{readingValue} \times 5V}{1023} \quad (15)$$

R_1 is the known resistor and 320Ω -resistor is used in this study. R_2 is the resistor unknown and to be measured with the Arduino ohmmeter.

Arduino program

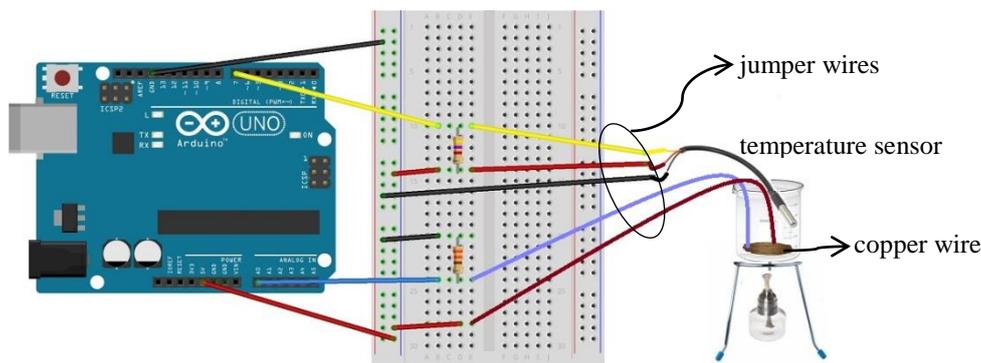


Figure 4. Schematic illustrating how the experimental apparatus is set up.

The experimental set-up shown in Figure 4 is installed. The DS18B20 sensor is mounted on the Arduino uno board. 5 male-male jumper cables and $4.7\text{ K}\Omega$ resistors are used for this set-up. On the breadboard, the $4.7\text{ K}\Omega$ pullup resistor is connected between the sensor's DQ end (yellow end) and the VCC end (black end). The VCC, DQ and GND ends of the sensor are connected to the Arduino uno card +5V, 7 and GND pins respectively (Figure 4). The end of the sensor to be measured is left inside the beaker filled with water. When using the DS18B20 temperature sensor, two libraries are needed in general. These libraries include 1-Wire (one wire) (<http://www.pjrc.com>) and Dallas Temperature (<https://github.com/milesburton>). For the operation of the sensor, these libraries must be added to the Arduino code. Otherwise, the program code will give an error.

The Arduino ohmmeter and resistance measuring device are equipped with $320\ \Omega$ resistor, two male female jumper wires, 3 male-male jumper wires and copper wire to be measured. One end of the $320\ \Omega$ resistor is connected (R_1 resistor in Figure 3) to the A0 analog pin of the board and the other end to the GND pin with the help of the two male-male jumper wires. One end of the copper wire is connected to the pin on the breadboard to connect the end of the $320\ \Omega$ analog resistor to the A0 analog pin. The other end is connected

to the + 5V pin of the Arduino card via the breadboard (R_2 resistor in Figure 3). In order to avoid damage to the temperature of the test device, first the male-female jumper wire and then the male-male jumper wires are connected to the ends of the copper wire at these connections. Thus, the spirit stove is removed slightly from the set-up in this way. In addition, the jumper wires connected to the ends of the copper wire are covered with heat-resistant waterproof material to prevent damage in hot water.

The end of the temperature sensor connected to the Arduino board and the copper wire is placed inside the beaker with some water. Thus, the experimental setup is ready. Arduino code is written to the Arduino interface to collect the data as shown in Figure 5. The compiler compiles this program in the computer, and sends it to the Arduino uno card with the USB cable. Data is collected by burning the spirit stove. Arduino card sends repeating experimental measurements (time, temperature and resistance values) to the computer at every 20 seconds by running the code. Data collection is continued until the water is boiled. The Arduino serial monitor will display a three-column table with experimental data. This data is easily copied and pasted to an Excel file.

```
//Temperature dependence of electrical resistance
#include<OneWire.h>           // add OneWire library
#include<DallasTemperature.h> // add DallasTemperature library
#define sensorPin 7           // declare as sensorPin to pin 7
#define VoltagePin A0         //declare as VoltagePin to AnalogPin A0
OneWire oneWire(sensorPin);   //start DS18B20 device
DallasTemperature sensors(&oneWire);
//transfer to the DallasTemperature library for mathematical calculations after DS18B20 device
starts

float temperature;           //declare float variable for temperature
unsigned long time =0;       //start with zero value and declare unsigned long variable for time
void setup(void) {
  Serial.begin(9600);        //initialize Serial Monitor
  sensors.begin();           //start libraries
}
void loop(void) {
  time= millis();            //start time as ms
  sensors.requestTemperatures();
  //send the command to the sensor to measure the temperature value
  temperature=sensors.getTempCByIndex(0);
  //assign the temperature reading from the sensor to temperature variable
```

```

Serial.print(time/1000.0);    //print time in seconds
Serial.print(" ");          //print space for separator
Serial.print(temperature);   //print temperature
Serial.print(" ");          //print space for separator

float readingValue = analogRead(VoltagePin);
                          //assign the value read from the analog A0 pin to the variable

float R1=320.0;             //declare float variable for R1 and assign value
float V=5;                  //declare float variable for V and assign 5 value
float V1= readingValue*V/1023; //convert the ADC reading to voltage (volt)
float R2=(R1*(V-V1))/V1;    //calculate the value of R2 (ohm)
Serial.println(R2);         //print R2
delay(20000);               //wait for 20 seconds
}

```

Figure 5. Arduino programme code.

Findings

The experimental set-up in Figure 4 was established, and the spirit stove was burned to collect experimental data. The copper wire was heated from room temperature to the boiling temperature of the water in the beaker (97.31 °C) and the data are collected in this range. The resistance-temperature graph of this data is given in Figure 6. The data are fitted on a function of the type of equation 7 through the linear regression algorithm in the Excel Office application. The data obtained, including the correlation coefficients, R^2 and the correlation curves drawn by solid line according to equation 7, are shown in Figure 6. The fact that the correlation coefficients are close to 1 indicates that the first order linear model exceptionally explains experimental data. In this graph, students can easily see that the resistance changes linearly with temperature. The slope of the graph was calculated as 0.0319. For the reference resistance value (R_0) of the copper wire, the resistance was measured with the Arduino ohmmeter at 20 °C using the experimental setup and it was measured as $R_0 = 7.69 \Omega$. The coefficient of resistance temperature was calculated as,

$$\alpha = \frac{0,0319}{7,69} = 0,0041$$

by using this value and the slope of the graph in equation 9. Thus, the temperature coefficient for the copper wire was determined experimentally. This value is consistent with the values given in the literature for copper (Giancoli, 2009, pp. 658).

The resistance-temperature relationship for copper wire was theoretically investigated, too. R_0 reference resistance was calculated using equation 3 in the theoretical evaluation. Here, the resistivity value for copper was taken at 20 °C, $\rho_0 = 1.724 \times 10^{-8}$ ohm.m (Poker & Klabunde, 1982). The length of the copper wire used in the experiment is $l = 30$ m and $r = 0.015$ mm. The cross-sectional area of the wire was calculated from $A = \pi r^2$. The temperature coefficient for copper was $\alpha = 3.93 \times 10^{-3}$ (1/°C) (Eargle, 2012). For theoretical calculations, T temperature values were taken as the same values from experimental values. Then in equation 5, resistance-temperature data were obtained by using these values. The graph drawn by these data is given in Figure 6. It can be seen in Figure 6 that the experimental and theoretically obtained resistance-temperature change graphs have the same characteristic. This result proves the reliability of the experiment performed with the Arduino board.

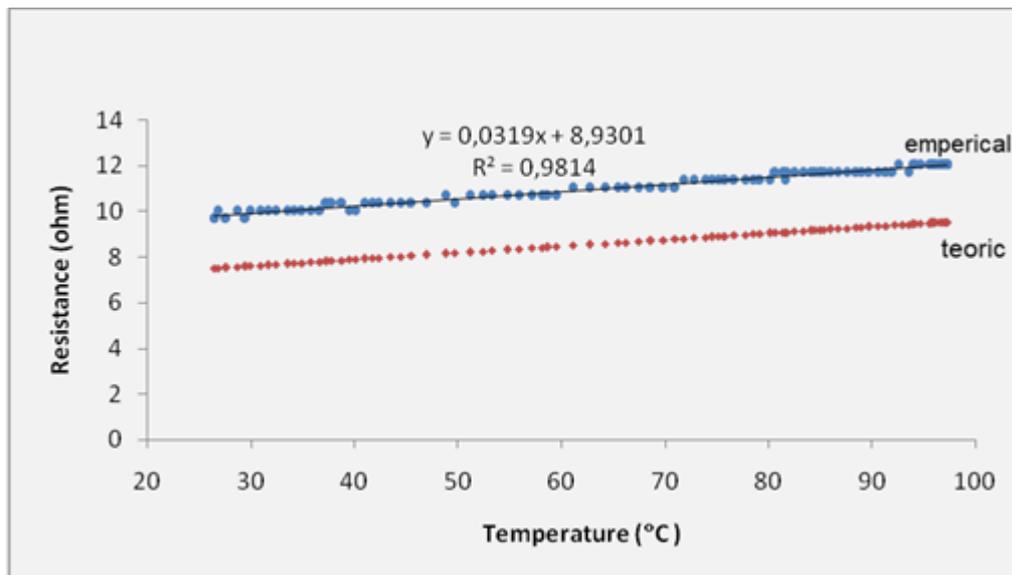


Figure 6. Graphs of experimental data and theoretical calculations

Discussion and Conclusions

The Arduino platform has been widely used recently in physics education to perform a number of different measurements with low cost. There are many examples of applications including various Arduino-based physics experiments developed to measure physical quantities such as voltage, temperature and pressure (Gingl et al., 2019). Bezerra et al. (2019) used an Arduino platform to demonstrate Faraday's law. Sari (2019) designed an Arduino experiment to examine the movement of the object on an inclined plane and to define the kinetic friction coefficient. Hahn, Oliveira Cruz, and Carvalho (2019) determined the speed of

sound as a function of temperature using Arduino. In this study, an Arduino-based physics experiment that can be used in physics laboratories was designed. An Arduino platform was used to examine the temperature variation of the electrical resistance and to calculate the temperature coefficient. A harmonious match can be observed between the resistance-temperature data obtained in the experiment and the theoretical model. According to the result, the experiment can be performed reliably in physics laboratories. Teachers and students can build their own instruments using simple equipment with the Arduino board. They can code to calculate and display the result. Thus, the Arduino-based physics experiment can be done in a cheap and practical way with the system described above. We used an Arduino ohmmeter to measure the resistance change in the experiment. In the Arduino ohmmeter, resistance is measured by building a voltage divider with an output voltage that depends on the unknown resistance. This method is quite simple and useful. Therefore, Arduino ohmmeter can be valuable tool in physics labs and it can be used for different physics experiments (Makan, Mingesz & Gingl, 2019).

Experimental systems examining the relationship between resistance and temperature have a very complex structure (Yolkin, 2002). However, the Arduino microprocessor can be programmed with a very simple programming language (URL-1). Students can collect and process experimental data quickly and easily with sensors connected to the microprocessor. Thus, they can focus most of their time and attention on interpreting data rather than collecting data and graphing (Russell, Lucas & McRobbie 2004; Sarı, 2019). Additionally, this activity can provide students the opportunity to work interdisciplinary with enabling them to use the disciplines of mathematics, engineering and technology to study electrical resistance in physics. Students can be asked to develop algorithms for solving problems by creating problem situations and coding according to this algorithm. Then, they can collect the data quickly and convert them into graphs in the computer environment by establishing the experimental setup. Therefore, we expect that this experiment effects students' algorithmic thinking skills, data collection skills, and problem solving skills etc. along with the gains in physics (Hsu & Wang, 2018; Jaipal-Jamani & Angeli, 2018). However, these effects should be tested in future studies.

References

- Arduino temperature control library. Available at: <https://github.com/milesburton/Arduino-Temperature-Control-Library> (accessed 11 May 2019).
- Atkin, K. (2016). Using the arduino with makerplot software for the display of resonance curves characteristic of a series LCR circuit. *Physics Education*, 51(6), 065006.
- Bezerra, A. Z. L. N., Cabreira, F. M., Freitas, W. P. S., Cena, C. R., Alves, D. C. B., Reis, D. D., & Goncalves, A. M. B. (2019). Using an arduino to demonstrate faraday's law. *Physics Education*, 54(4), 043011.
- Chen, S., Lo, H. C., Lin, J.W., Liang, J.C., Chang, H.Y., Hwang, F.K. & Wang, C.Y. (2012). Development and implications of technology in reform-based physics laboratories. *Physical Review Special Topics-Physics Education Research*, 8(2), 020113.
- Darrah, M., Humbert, R., Finstein, J., Simon, M. & Hopkins, J. (2014). Are virtual labs as effective as hands-on labs for undergraduate physics? A comparative study at two major universities. *Journal of Science Education and Technology*, 23(6), 803-814.
- DS18B20 waterproof temperature sensor datasheet. Programmable Resolution 1-Wire Digital Thermometer. Available at: <https://datasheetspdf.com> (accessed 10 May 2019).
- Eargle, J. (Ed.). (2012). *Electroacoustical reference data*. Springer Science & Business Media.
- Giancoli, D. (2009) 25. *Electric currents and resistance in jocelyn phillips physics for scientists and engineers with modern physics (4th ed.)*. Upper Saddle River, New Jersey: Prentice Hall.
- Gingl, Z., Mellar, J., Szepe, T., Mekan, G., Mingesz, R., Vadai, G., & Kopasz, K. (2019). Universal Arduino-based experimenting system to support teaching of natural sciences. *arXiv preprint arXiv:1901.03810*.
- Hahn, M. D., de Oliveira Cruz, F. A., & Carvalho, P. S. (2019). Determining the speed of sound as a function of temperature using arduino. *The Physics Teacher*, 57(2), 114-115.
- Hsu, C.C. & Wang, T.I. (2018). Applying game mechanics and student-generated questions to an online puzzle-based game learning system to promote algorithmic thinking skills. *Computers & Education*, 121, 73-88.
- Jaipal-Jamani, K. & Angeli, C. (2018). Developing teacher self-efficacy to teach science and computational thinking with educational robotics: using scaffolded programming scripts. In *Self-Efficacy in Instructional Technology Contexts* (pp. 183-203). Springer, Cham.
- Mekan, G., Mingesz, R., & Gingl, Z. (2019). How accurate is an arduino ohmmeter?. *Physics Education*, 54(3), 033001.
- Meaden, G.T. (2013). *Electrical resistance of metals*. Springer.
- MoNEa. (2018). *Fen bilimleri dersi (ilkokullar ve ortaokul 3, 4, 5, 6, 7 ve 8. sınıflar) öğretim programı* [Science courses curriculum (primary and secondary school 3, 4, 5, 6, 7 and 8 grades)]. Ankara: Talim ve Terbiye Kurulu Başkanlığı.
- MoNEb. (2018). *Ortaöğretim fen lisesi fizik dersi öğretim programı* [Secondary science high school physics course curriculum]. Ankara: Milli Eğitim Bakanlığı Yayınları.

- OneWire Arduino Library, connecting 1-wire devices-PJRC. Available at: http://www.pjrc.com/teensy/arduino_libraries/OneWire.zip (accessed 11 May 2019).
- Pereira, N.S.A. (2016). Measuring the RC time constant with arduino. *Physics Education*, 51(6), 065007.
- Poker, D.B. & Klabunde, C.E. (1982). Temperature dependence of electrical resistivity of vanadium, platinum, and copper. *Physical Review B*, 26(12), 7012.
- Russell, D.W., Lucas, K.B. & McRobbie, C.J. (2004). Role of the microcomputer-based laboratory display in supporting the construction of new understandings in thermal physics. *Journal of Research in Science Teaching: The Official Journal of the National Association for Research in Science Teaching*, 41(2), 165-185.
- URL-1. Arduino website. Available at: <https://www.arduino.cc/> (accessed 11 May 2019).
- Sarı, U. (2019). Using the Arduino for the experimental determination of a friction coefficient by movement on an inclined plane. *Physics Education*, 54(3), 035010.
- Sarı, U., Pektaş, H.M., Çelik, H. & Kirindi, T. (2019). The effects of virtual and computer based real laboratory applications on the attitude, motivation and graphic interpretation skills of university students. *International Journal of Innovation in Science and Mathematics Education (formerly CAL-laborate International)*, 27(1).
- Serway, R.A., Faughn, J.S. & Vuille, C. (2014). *College physics brooks* Cole Pub Co., Boston p 606.
- Tunyagi, A., Kandrai, K., Fülöp, Z., Kapusi, Z., & Simon, A. (2018). Friction coefficient determination by electrical resistance measurements *Physics Education*, 53(3), 1-9.
- Yolkin, B. (2002). Lab Report–An Example. *PHYS 340: Modern Physics Lab*, 34. Available at: <http://www.physics.purdue.edu/~fqwang/teaching/Phys340-Manual.pdf> (accessed 11 May 2019).
- Young, H.D., Freedman, R.A., Sandin, T.R. & Ford, A.L., (1996). *University physics 2 Reading* MA: Addison-Wesley.