

**Development of a Low-Cost Potentiometer Using Arduino Technology**

By

R . M . P . M . Rajapaksha

(SC/2020/11577)

B . K . R . I . Wanshanatha

(SC/2020/11523)

An report submitted to the Faculty of Science, University of Ruhuna, in partial fulfilment of the requirement for the B.Sc (General) Degree

Matara

Sri Lanka

October, 2024

## Declaration

I hereby declare that this report or part of this report has not been previously submitted to a university or institute for awarding a degree/diploma/certificate and all the materials (data, theoretical analysis, text, etc) referred from other sources are properly cited in the text of the essay.

|  |  |
| --- | --- |
| ……………….. | ………………………… |
| Date | Signature of the student  R . M . P . M . Rajapaksha  SC/2020/11577 |
| ……………….. | ………………………… |
| Date | Signature of the student  B . K . R . I . Wanshanatha  SC/2020/11523 |

I certify that the above particulars are correct.

|  |  |
| --- | --- |
| ……………….. | ………………………… |
| Date | Dr. Chinthaka Nadun Ratnaweera  Senior Lecturer Gr.II |
| …………………..  Date | ……………………………...  Prof. Nishantha K. Kalutharage |
|  |  |
| Supervisor’s Remarks (If Any)  ……………………………………………………………………………………….  ………………………………………………………………………………………. | |

**ACKNOWLEDGMENT**

Firstly, we are extremely grateful to our supervisors **Dr. Chinthaka Nadun Ratnaweera** for giving me this valuable opportunity to join this project "Development of Low-Cost Potentiometer for Physical chemistry practical" and for his invaluable advice and Continuous support for my project. Also, we give our special thanks to **Mr. K. A. S. Lakshan (Probationary Lecturer)** who gave invaluable advice and maximum support to make our project successful.

Furthermore, we would also like to acknowledge with much appreciation the crucial role of staff in the laboratory department of chemistry and physics, who gave us permission to use lab equipment and also support. Finally, we humbly appreciate our family and all of our friends who helped with this project.

**ABSTRACT**

Potentiometry is one of the most useful electroanalytical techniques which has a wide range of industrial applications. Also, potentiometers are frequently used in physical and analytical chemistry practicals. However, the cost of electrodes and potentiometer devices limits its usage of this instrument in practical laboratories. Therefore, this study attempts to develop a potentiometer using the Arduino Uno board coded by C language. Another intended advantage of this instrument is all the potential measurements are directly fed to the Android mobile application through a Blue-tooth connection and perform the necessary calculations. The development process of Low-Cost potentiometer was done according to the design of an electronic circuit for the potentiometer, calibration of the voltage sensor and AD620 instrumental amplifier, implement source code for Arduino potentiometer and design and implement android application. High accuracy with four decimal points and very much closer readings were obtained by using the new device compared with respect to the laboratory potentiometer. Finally, as obtained results students were able to use Low-cost potentiometer for the potentiometric titrations without any issues.

**INTRODUCTION**

**Literature Review**

Recently, there has been an interest in the development of low-cost scientific instrumentation for both teaching and research, with the aim of enhancing student learning by hands-on experience. In this vein, equipment based on Arduino is one popular option; it allows cost reduction and inter-disciplinary learning. Potentiometers are important equipment in analytical chemistry; they measure the potential differences, but commercial versions are expensive, prone to damage, and require inconvenient manual collection of data

**Research Problem**

The fiscal constraints of universities and the ever-growing needs of practical learning tools have raised the need for low-cost, easy-to-operate, efficient laboratory equipment. Among such apparatus, the potentiometer-a must in electroanalytical chemistry-is a perfect candidate for low-cost redesign. Traditional potentiometers involve manual data logging and are often prone to damage and costly replacement or repair costs. Therefore, the problem statement would be:

What is the development of a low-cost Arduino-based potentiometer, which, for educational and research purposes, increases efficiency while reducing the costs in a physical chemistry laboratory?

**Objectives**

The primary goal of this project is to develop a low-cost potentiometer to overcome the limitations of traditional laboratory instruments. The key objectives include:

1. Develop a Low-Cost Potentiometer

2. Enable Wireless Data Transmission

3. Enhance Student Learning Experience

4. Improve Laboratory Efficiency

By achieving these objectives, the project aims to offer an affordable and educationally valuable alternative to expensive potentiometers, benefiting both students and educational institutions.

**MATERIALS AND METHOD**

**Materials**

Power supplies, Multimeters, Arduino Uno board, 16 x 2 LCD Display, HM 10 Blue-tooth module, AD620 Instrumental Amplifier, Voltage sensor(0-25V), DC to DC Voltage Regulator,12V AC to DC power adapter, Jumping Wires, Bread Board, Connecting wires, soldering wire, Crocodile clips, Hg/HgCl Electrode, Platinum Electrode.

**Glass Ware:** Beakers, Pipettes, Burettes, Funnel

**Chemical:** 25 ml of 0.1MHCl Solution, 50ml of 0.1M NaOH Solution, Distil water

Constructing Arduino Potentiometer and Android Application

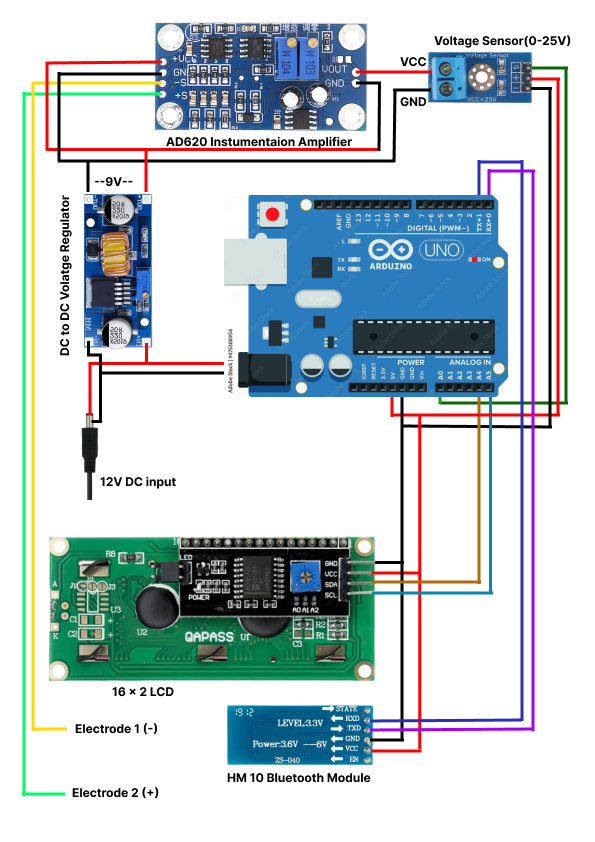
1. Step – Design an electronic circuit for the potentiometer

Figure 1 – Arduino base circuit

When the circuit was powered, the voltage level was captured by the voltage sensor, which was amplified by the AD620. Here 9 voltage was suppled to the AD620 instrument output GND terminal because to get the negative reading of voltage. The processed signal was read by the Arduino, and the value was then displayed on the LCD. Simultaneously, the data was transmitted wirelessly via the Bluetooth module by Arduino, enabling real-time monitoring on a mobile device.

1. Step – Calibration of the Voltage sensor and AD620 Instrumental Amplifier

|  |  |
| --- | --- |
| Voltage Sensor: |  |
| AD620 Instrumental Amplifier: |  |
|  |  |

In this step power supplies and multimeters were used to calibrate the above mentioned as main apparatus and reading were obtained with respect to the input characteristics and output characteristics. Then as obtained results, graphs were plotted and by using these graphs were obtained following Equations for instrument calibration. The data sheets were added to the appendices. (Appendix 1)

1. Step - Implement source code for Arduino potentiometer

According to the above circuit diagram and our expected outcome were implemented most suitable and accurate C language code by using equations mentioned in calibrate part (step 2). Arduino source code was implemented by using Arduino IDE and some important Arduino libraries such as LiquidCrystal\_I2C.h, Wire.h and SoftwareSerial.h, Arduino source code were attached to the appendices. (Appendix 2)

1. Step – Design and Implement Android application

A graph plotting functionality in an Android application was designed and implemented using React Native. The developed application provides students with the ability to plot some data in graphical format directly from their mobile phones in line graphs. This has been developed in a structured manner, reaping benefits from modern JavaScript libraries and tools. Application interfaces were attached to the appendices. (Appendix 3)

**READINGS AND DISCUSSION**

Results validation of the Low-Cost potentiometer

Table 1: EMF Values of Low-Cost potentiometer and Laboratory Potentiometer with NaOH volumes Pt as working Electrode and Hg/HgCl as Reference. (at 30 c)

|  |  |  |
| --- | --- | --- |
| Volume of NaOH  (V±0.05 ) / ml | EMF Value of Laboratory Potentiometer (E±0.0001 ) / V | EMF Value of Arduino Potentiometer (E±0.0001 ) / V |
| 0 | 0.3810 | 0.3819 |
| 1 | 0.3796 | 0.3792 |
| 2 | 0.3772 | 0.3778 |
| 3 | 0.3731 | 0.3737 |
| 4 | 0.3684 | 0.3689 |
| 5 | 0.3661 | 0.3667 |
| 6 | 0.3615 | 0.3611 |
| 7 | 0.3584 | 0.3583 |
| 8 | 0.3561 | 0.3566 |
| 9 | 0.3538 | 0.3545 |
| 10 | 0.3504 | 0.3508 |
| 11 | 0.3493 | 0.3491 |
| 12 | 0.3477 | 0.3474 |
| 13 | 0.3428 | 0.3421 |
| 14 | 0.3385 | 0.3389 |
| 15 | 0.3369 | 0.3362 |
| 16 | 0.3303 | 0.3307 |
| 17 | 0.3281 | 0.3284 |
| 18 | 0.3254 | 0.3259 |
| 19 | 0.3101 | 0.3106 |
| 20 | 0.3073 | 0.3078 |
| 21 | 0.3052 | 0.3055 |
| 22 | 0.3016 | 0.3019 |
| 23 | 0.2998 | 0.2992 |
| 24 | 0.2951 | 0.2957 |

**Discussion**

Initially we observed certain fluctuations in the reading. To overcome this issue, we have taken 50 readings. Then calculated an average value and sent it to the display and via Blue-tooth to the Android application. When connecting the connecting wires, it is possible to observe very small frequent variation errors in the readings, mainly due to loosen connections. The reading of the newly developed Low-Cost potentiometer is given to 4 decimal places, which means that the accuracy of the measurement is much closer to the values obtained with 4 decimal places of a Laboratory potentiometer.

**CONCLUSIONS**

A low-cost Potentiometer was developed according to the steps mentioned above. It can successfully obtain the closest and most accurate readings obtained by a normal potentiometer in the laboratory. So, with this Arduino potentiometer, our aim has been successful. We have shown that such expensive devices can be manufactured at very low cost, and we developed an Android application to plot the graph with chemical potentials and volume. This will save time for the students and money for our university and country.

Finally, Low-cost potentiometer was able to get very much closer and accurate potentials compared with normal laboratory potentiometer and obtained readings were able to transmit to android application via Blue-tooth technology. As a result, students can use Low-cost potentiometer for the potentiometric titrations without any issues.

**REFERENCES**

1. Arduino. *Arduino Docs | Arduino Documentation*. docs.arduino.cc. https://docs.arduino.cc/.
2. *Google Books*. Google.lk. https://www.google.lk/books/edition/Arduino\_Cookbook/nxE245VgtsUC?hl=en&gbpv=1&pg=PP2&printsec=frontcover (accessed 2024-10-15).
3. *Introduction · React Native*. reactnative.dev. <https://reactnative.dev/docs/getting-started>.
4. dotintent. *GitHub - dotintent/react-native-ble-plx: React Native BLE library*. GitHub. https://github.com/dotintent/react-native-ble-plx (accessed 2024-10-15).
5. reactorman. *Send text data via Bluetooth using React-Native App*. Stack Overflow. https://stackoverflow.com/questions/60106721/send-text-data-via-bluetooth-using-react-native-app (accessed 2024-10-15).

**‌**

**APPENDIX 1 – Calibration Part in step 2**

Voltage Sensor Calibration

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | chart 1 |  | | (X±0) /no unit | (Y±0.0001)/V | | 96 | 2 | | 184 | 4 | | 267 | 6 | | 354 | 8 | | 440 | 10 | | 524 | 12 | | 608 | 14 | | 693 | 16 | | 781 | 18 | | 864 | 20 | | 950 | 22 | | chart 2 |  | | 99 | 2 | | 184 | 4 | | 270 | 6 | | 353 | 8 | | 439 | 10 | | 526 | 12 | | 608 | 14 | | 695 | 16 | | 779 | 18 | | 863 | 20 | | 950 | 22 | | chart 3 |  | | 100 | 2 | | 182 | 4 | | 267 | 6 | | 355 | 8 | | 439 | 10 | | 525 | 12 | | 610 | 14 | | 695 | 16 | | 781 | 18 | | 864 | 20 | | 951 | 22 | | |  |  | | --- | --- | | chart 4 |  | | 102 | 2 | | 187 | 4 | | 271 | 6 | | 357 | 8 | | 444 | 10 | | 528 | 12 | | 615 | 14 | | 699 | 16 | | 782 | 18 | | 871 | 20 | | 950 | 22 | | chart 5 |  | | 102 | 2 | | 187 | 4 | | 273 | 6 | | 356 | 8 | | 444 | 10 | | 528 | 12 | | 614 | 14 | | 699 | 16 | | 784 | 18 | | 870 | 20 | | 955 | 22 | | chart 6 |  | | 100 | 2 | | 186 | 4 | | 270 | 6 | | 357 | 8 | | 444 | 10 | | 528 | 12 | | 613 | 14 | | 695 | 16 | | 781 | 18 | | 866 | 20 | | 951 | 22 | |
| Table 2: Vilatge with Analog Read | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  | Gradient | y-intercept | | chart1 | 0.0235 | 0.286 | | chart2 | 0.0235 | 0.3322 | | chart3 | 0.0235 | 0.3057 | | chart4 | 0.0235 | 0.4002 | | chart5 | 0.0234 | 0.3843 | | chart6 | 0.0235 | 0.3829 | | Summation | 0.1409 | 2.0913 | | Average | 0.023483333 | 0.34855 | |
| Table 2: Calculate Gradient and y-intercept |

|  |  |
| --- | --- |
|  | **A graph with a line  Description automatically generated** |
| Figure 2 – Vilatge vs Analog Read Chart 1 | Figure 3 – Vilatge vs Analog Read Chart 2 |
| **A graph with a line  Description automatically generated** | **A graph with a line  Description automatically generated** |
| Figure 4 – Vilatge vs Analog Read Chart 3 | Figure 5 – Vilatge vs Analog Read Chart 4 |
|  | **A graph with a line  Description automatically generated** |
| Figure 6 – Vilatge vs Analog Read Chart 5 | Figure 7 – Vilatge vs Analog Read Chart 6 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Voltage Error (voltage ± 0.001) / V | | | Input Voltage  (S+ & S-) | Output Voltage  (Vout & GND) | | 0.092 | 2.09 | | 0.094 | 2.13 | | 0.096 | 2.185 | | 0.098 | 2.213 | | 0.1 | 2.271 | | 0.102 | 2.319 | | 0.104 | 2.352 | | 0.106 | 2.4 | | 0.108 | 2.46 | | 0.11 | 2.496 | | 0.112 | 2.533 |   Table 3 – AD620 instrumental amplifier input and output readings |
| **A graph with blue dots  Description automatically generated** |
| Figure 8 – Output Volatge vs Input Voltage Chart |

AD620 Instrumental Amplifier Calibration

**APPENDIX 2 – Arduino Source Code in step 3**

**#include <SoftwareSerial.h>**

**#include <LiquidCrystal\_I2C.h>**

**#include <Wire.h>**

**LiquidCrystal\_I2C lcd(0x27,16,2);  // Uncomment if using an I2C LCD**

**// RX from HM-10 -> Pin 10, TX from HM-10 -> Pin 11**

**SoftwareSerial BTSerial(10, 11); // RX, TX**

**int aRead;**

**float avgVoltage, voltage, sumVoltage;**

**void setup() {**

**Serial.begin(9600);      // Start Serial Monitor communication**

**BTSerial.begin(9600);    // Initialize HM-10 communication**

**delay(1000);  // Wait for the HM-10 to initialize**

**// Optional: Set a new name for the HM-10 (if not already configured)**

**sendATCommand("AT+NAMEBT05");**

**// Wait for any setup or connection delays**

**delay(3000);**

**Serial.println("Bluetooth ready. Sending data...");**

**// Uncomment if using an LCD display**

**lcd.init();**

**lcd.backlight();**

**// Set A0 as input (for reading analog values)**

**pinMode(A2, INPUT);**

**}**

**void loop() {**

**// Calculate voltage based on analog read value**

**for (int i = 1; i <= 5; i++) {**

**//  Read the analog input**

**aRead = analogRead(A2);**

**Serial.print("Analog Read: ");**

**Serial.println(aRead);**

**Serial.println("..............");**

**voltage = (aRead \* 0.02348) - 0.34855;**

**voltage = (voltage - 0.0144) / 22.5410;**

**voltage += 0.0228+0.0780;**

**sumVoltage += voltage;**

**voltage =0;**

**delay(10);**

**}**

**// Average the calculated voltagevv**

**avgVoltage = sumVoltage / 5;**

**//avgVoltage += 0.0228+0.0690;**

**sendBluetoothMessage(avgVoltage);**

**// Uncomment if using an LCD display**

**lcd.setCursor(0, 0);**

**lcd.print("Voltage:");**

**lcd.print(avgVoltage,4);**

**sumVoltage = 0;  // Reset sum for the next loop**

**avgVoltage=0;**

**delay(200);  // Delay between readings**

**//delay(5000);  // Send every 5 seconds (adjust as needed)**

**}**

**// Function to send AT commands (optional configuration)**

**void sendATCommand(const char \*command) {**

**Serial.print("Sending AT Command: ");**

**Serial.println(command);**

**BTSerial.println(command);**

**delay(1000);  // Wait for the response**

**}**

**// Function to send a string message over Bluetooth**

**void sendBluetoothMessage(float message) {**

**BTSerial.println(message,4);  // Send the message to the connected device**

**}**

**APPENDIX 3 – Android Application GUI in step 4**

|  |  |
| --- | --- |
| **A logo on a brown background  Description automatically generated**  Figure 9 – Welcome Interface | **A screen shot of a computer screen  Description automatically generated**  Figure 10 – User Login Interface |
| **A brown rectangle with yellow lines  Description automatically generated**  Figure 11 – User Home Interface | |

**APPENDIX 3 – Some Pictures of Potentiometer**

|  |
| --- |
|  |
| Figure 12 – Calibrating time of used apparatus |
| **A computer and electrical equipment on a table  Description automatically generated** |
| Figure 13 – Testing time of new device |