Development of a low-cost Arduino-based potentiostat

João R. Crespo,[1,2] Stephen R. Elliott,[2] Tanya Hutter,[3,4] Hugo Águas[1] *

[1] *i3N/CENIMAT, Department of Materials Science, Faculty of Science and Technology, Universidade NOVA de Lisboa and CEMOP/UNINOVA, Campus de Caparica, 2829-516 Caparica, Portugal.*

[2] *Department of Chemistry, University of Cambridge, Cambridge CB2 1EW, United Kingdom*

[3] *Department of Mechanical Engineering, The University of Texas at Austin, 204 E Dean Keeton St, Austin, TX 78712, USA*

[4] *Materials Science and Engineering Program and Texas Materials Institute, The University of Texas at Austin, 204 E Dean Keeton St, Austin, TX 78712, USA*

*corresponding author: hma@fct.unl.pt

Abstract

A simple Arduino-based potentiostat has been developed. This potentiostat design is cost effective for low budget applications and educational purposes. The resolution and linear response of the developed potentiosat was evaluated. A cyclic voltammetry of potassium ferricyanide $K_3[Fe(CN)_6]$ was performed to compare the potentiostat electrochemical performance with a commercial one. To show the portable potentiostat capabilities, cyclic voltammograms were also performed at different scan rates. Finally, the diffusion coefficient of potassium ferricyanide was calculated in a solution containing a known concentration of the salt using the Randles-Sevcik equation.

Keywords: electronics, potentiostat circuit, electrochemistry, electrodes, voltammetry, Arduino.

Table of Contents

Abbreviations

RE – Reference Electrode

CE – Counter Electrode

WE – Working Electrode

Ag – Silver

Ag/AgCl – Silver/Silver Chloride saturated electrode

Pt – Platinum

TIA - Transimpedance Amplifier

VF – Voltage Follower

CA – Control Amplifier

PWM – Pulse Width Modulation

Op-amp – Operational Amplifier

V_{pp} – Peak to Peak Voltage

DA – Differential amplifier

μA – Micro amperes

CV – Cyclic Voltammograms

1 Introduction

The Open-source development microcontrolled electronics boards such as Arduino and Raspberry Pi are becoming more popular in the Research and Development field because they offer a great versatility and ease of use for different applications. [1] One of the applications explored here is the potentiostat, a widely used instrument in electrochemistry field.

The potentiostat is an electronic circuit commonly used in electrochemistry to study the electrochemical events taking place at one specific electrode. The electrochemical studies often require specific tuning of the applied conditions to perform certain studies. [2] which open-source software platforms such as Arduino IDE are capable of.

The aim of this work is to present the full development of a simple potentiostat prototype integrating an Arduino Uno board and the detailed explanation of the electronic circuit combined with the Arduino software functions to perform different electrochemical experiments.

1.1 The potentiostat circuit

A simple potentiostat circuit is shown in the Figure 1, the Control Amplifier (CA), is a servo amplifier. It compares the measured Cell voltage with the desire voltage and drives current into the cell to force the voltages to be the same in an inverting configuration to provide the negative feedback. The Voltage follower (VF), sometimes also referred as Electrometer, measures the voltage of the RE and the output signal is introduced in the feedback loop and it is the signal that is measured whenever the cell voltage is needed. An ideal VF has zero input current and an infinite input impedance. Current flow through the reference electrode can change its potential. In practice, all modern VF amplifiers have input currents close enough to zero that this effect can usually be ignored. The transimpedance amplifier (TIA), converts the measured current in the WE into a voltage through the resistance R_f. The cell current in some experiments do not change much. In other experiments, when electrochemical reactions occur, the current can often vary by as much as orders of magnitude, so it is of interest to use different values of the R_f to allow the measurement of widely varying currents. Other important feature of this amplifier is forcing ground voltage in the WE by connecting the non-inverting input to ground. This way the voltages values are measured relative to grounded WE.

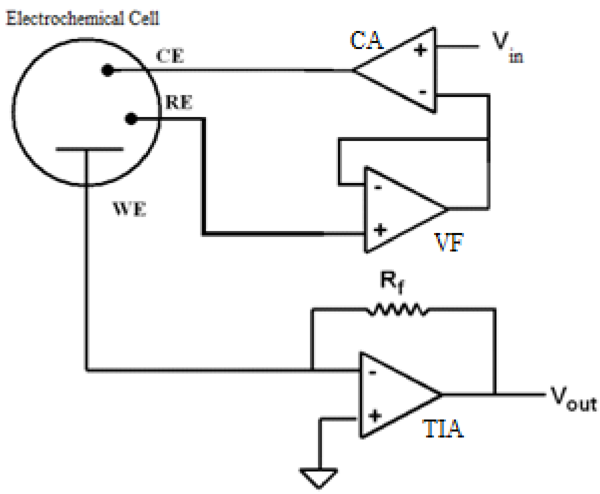

Figure 1 – Basic potentiostat circuit

The Signal (V_{in}) is a computer-controlled voltage source. It is generally an output of a Digital-to-Analog (D/A) converter that converts a digital signal into an analog signal.

Different types of electrochemical experiments depend mostly on the type of the signal applied to the cell.[3] If the signal output is a linear voltage ramp is called Linear Sweep voltammetry, if the signal is a linear voltage ramp that returns to the initial value, (triangular shape), it is called cyclic voltammetry. Other very often used electrochemical experiment is called chronoamperometry where a constant voltage signal is applied, and the current is measured over time. [4]

2 Methodology

Arduino microcontroller boards based on the ATMega microcontroller family stand out because of their outstanding capabilities, affordable price and portability. Also, this boards have digital outputs which go up to 5V, with a range of ± 2.5 V which make this boards capable of producing the required output voltage range for Electrochemical experiments with the help of a voltage shifter circuit.

2.1 Potentiostat circuit design

The Potentiostat Circuit developed, present in the Figure 2, was inspired by the potentiostat circuit proposed by [5].The major potentiostat circuit differences are the use of a single op-amp U1 for signal supply and shifting, with the aim of reducing the total number of op-amps used. Also, instead of using a ladder circuit to supply the signal from the I/O microcontroller ports to the potentiostat circuit a simple RC filter was introduced.

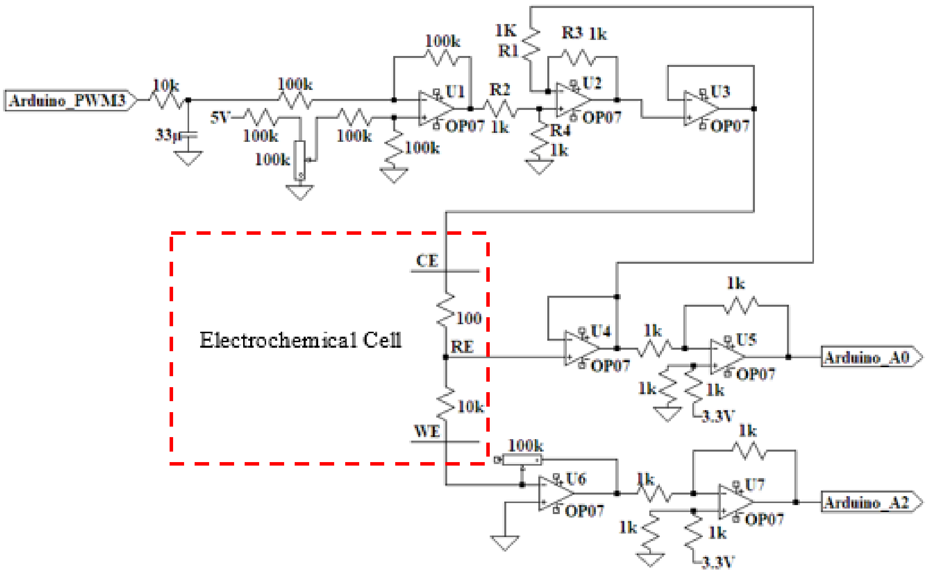

Figure 2 – Potentiostat analog circuit with the Electrochemical Cell highlighted.

2.2 Operation of the potentiostat

Pulse Width Modulation, or PWM, is a technique for getting analog results with digital means. Digital control is used to create a square wave, a signal switched between on and off. This on-off pattern can simulate voltages in between full on (5 Volts) and off (0 Volts) by changing the portion of the time the signal spends on versus the time that the signal spends off. However, in electrochemical measurements it is desired to have a DC like signal where the high frequencies are removed. To do this a RC filter was introduced which is compose by a resistance and a capacitor, this is shown in the Figure 3.

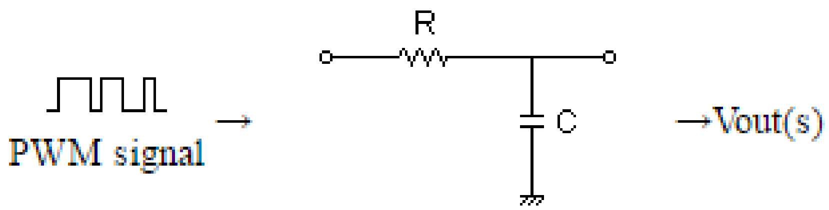

Figure 3 -Basic RC Low Pass Filter Circuit

The RC filter cutoff frequency was calculated to be approximately 0.5Hz which offers a -60dB of attenuation at 1kHz frequency. which is the PWM signal frequency used.

The Op-amp U1 is a Differential Amplifier (DA) and it has the function of driving the input signal from the Arduino PWM output and add offset voltage value, from the voltage divider composed by the 100K resistors and the 100k potentiometer in the non-inverting input, to shift the applied voltage to the cell in the desired range. The potentiometer is used so the user can tune manually the applied voltage range.

The Op-amp U2, as aforementioned, has the roll of the Control Amplifier. It compares the measured Cell voltage with the desire voltage and drives current into the cell to force the voltages to be the same in an inverting configuration to provide the negative feedback. This can be described by the following equations:

At the summing point,

$$V^- = V^+ \qquad (1)$$

$$V^- = V^+ = V_{in}\left(\frac{R_4}{R_2 + R_4}\right) \qquad (2)$$

If $V_{in} = 0$,

$$V'_{out} = -V_{RE}\frac{R_3}{R_1} \qquad (3)$$

If $V_{RE} = 0$,

$$V''_{out} = V_{in}\left(\frac{R_4}{R_2 + R_4}\right)\left(\frac{R_1 + R_3}{R_1}\right) \qquad (4)$$

Using the superposition theorem,

$$V_{out} = V'_{out} + V''_{out} \quad (5)$$

If all the resistors are of the same value, that is R1 = R2 = R3 = R4, then,

$$V_{out} = V_{in} - V_{RE} \qquad (6)$$

the op-amp U2 becomes a unity gain differential amplifier.

The Op-amp U3 is a voltage Follower that isolates the input from the output to prevent loading of the input signal. The voltage output from op-amp U3 was connected to the CE. The Op-amp U4 is a voltage follower and it has the function described previously for the Electrometer. Both the Op amp U5 and U7 have the function of voltage shifters and they sum 3.3 V from the Arduino to the output signals of the potentiostat, i.e. the voltage and the current measured. They are used because Arduino only can read voltage values from 0 to 5000 mV.

Thus, one can determine the maximum applied voltage ranges that can be read by the potentiostat which is from -3300 mV to +1700 mV. Even though is it not centered in 0 V it still matches most of the electrochemical applications ranges. The Op-amp U6 is a transimpedance amplifier and it has the function described previously for the TIA. The 10-100 kΩ gain potentiometer was used to read currents in the ranges from µA to mA where most of the electrochemical reactions take place.

2.3 Simulation

To simulate the circuit shown in the Figure 2 the LTspice simulator was used. A sine wave source was introduced in the circuit input called Arduino_PWM to represent a triangular shape function performed in cyclic voltammetry. All the op amps used were OP07, and they were supplied with +9V and -9V in the rails. The 10 seconds transient response of the circuit is shown in the Figure 4. The input function is represented in a green positive sine wave with an amplitude of 2 V_{pp}.

To shift the input function (shown in green) to be centered in 0 V, 1 V was added by the voltage divider made of the 100 kΩ resistors and the 100 kΩ potentiometer connected in the positive input of the op amp U1. The generated wave applied to the cell through the CE (V(ce), shown in blue) is inverted comparing to the input signal due to the inversion of the negative input signal caused by the Differential Amplifier (U1).

The current sensed by the WE is converted into an inverted voltage value through the TIA (op amp U6), then the signal is again inverted and 3.3 V are added by the Differential Amplifier U7, which sends the signal to the Arduino Analog input (A2), the resulting wave is shown in purple and it can be observed that is in phase with the input signal in blue.

Finally, the voltage sensed by the RE is sent by the VF U4, to the Differential Amplifier U5, which converts it into an inverted value and add 3.3 V to be written in the Arduino Analog input (A0), the resulting wave is shown in red and it can be observed that is in 180° out of phase with the input signal in blue, as expected.

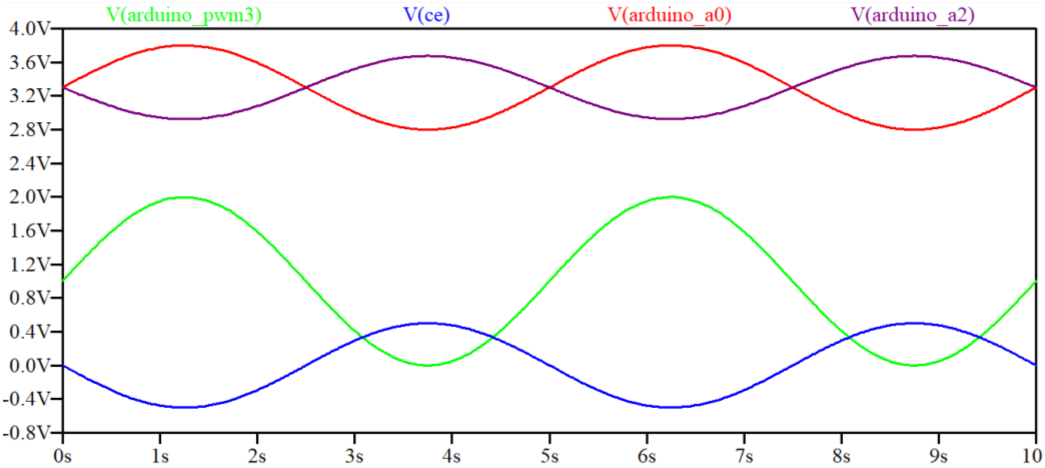

Figure 4 – Waveforms generated in LTspice of the potentiostat analog circuit.

2.4 Arduino software

To Perform Cyclic voltammetry, a triangular-shaped output voltage curve must be applied to the Counter electrode. To perform such output voltage scan, a step function was implemented using two *for* loops, one that increments (forward direction of the cycle) and other that decrements (reverse direction of the cycle) the value in the *analogwrite()* function in each cycle, as shown in the Figure 5. The *analogwrite()* function will increase and decrease, respectively, the duty cycle of the PWM output pin. The signal is then filtered, and an offset is added by the potentiostat circuit before it is applied to the Counter Electrode. The result output function is shown in the

Figure 6.

By changing the *maxvoltage* value one can define the range of the applied voltage.

```
//Cyclic Voltametry function
void loop() {
  int maxvoltage = 120;//voltage range
  int voltage_scan = 1;
  delay(2000);//waits to stabilize the voltage values
  for (int i=0; i <maxvoltage; i=i+voltage_scan){
      analogWrite(3, i);
      findAverage();
  }
  for (int i=maxvoltage; i>0; i=i-voltage_scan){
      analogWrite(3,i);
      findAverage();
  }
  delay(2000);//waits to stabilize the voltage values
  analogWrite(2,LOW);
  Serial.flush();
 while(1);//leaves arduino in a invisible loop
 }
```

Figure 5- Cyclic voltammetry function code.

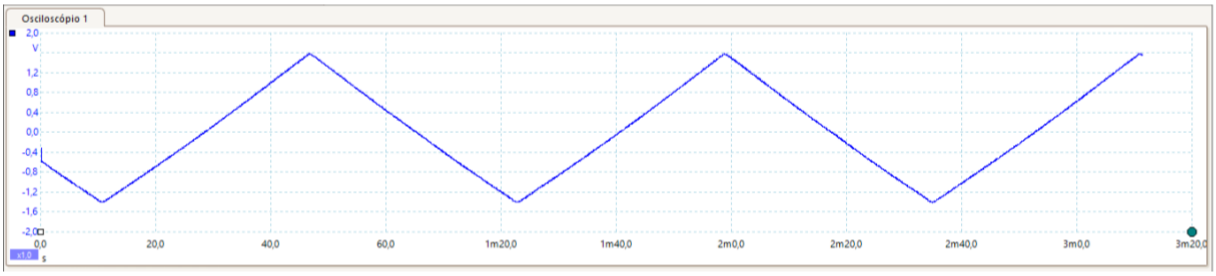

Figure 6 – Output voltage applied to the Counter Electrode when cyclic voltammetry function was performed, measured with an oscilloscope (picoscope 2000 series).

The *findAverage()* function, shown in the Figure 7, reads 9 values in each step and calculates the average voltage and the average current values. The 3.3V are subtracted to the current averaged value to remove the offset imposed by the voltage shifter op amp U7.

Due to the inversion of the average voltage signal read imposed the op amp U5 the value is then multiplied by -1 and 3.3V are added to obtain the real voltage value.

In fact, the calculated current value by the Arduino is a voltage value which then needs to be divided by the resistance of the TIA gain potentiometer to obtain the real current measured in the working electrode.

By changing the *steptime* constant one can change the duration of each step and this means to choose the scan rate of the cyclic voltammogram.

```
void findAverage(){
  float voltage_ave=0;
  float current_ave =0;
  int steptime = 85; // Step time in ms
  int y = 0;
  for(y=0;y<8;y++){
      delay(steptime/8);
      float voltage = analogRead(0);
      float current = analogRead(2);
      voltage_ave = (voltage_ave + voltage)/2;
      current_ave = (current_ave + current)/2;
  }
  voltage_ave = (voltage_ave*5/1023)*-1+3.3;
  current_ave = (current_ave*5/1023)-3.3;
  Serial.print(voltage_ave);
  Serial.print(" ");
  Serial.println(current_ave);
}
```

Figure 7 – FindAverage () function code.

To perform certain electrochemical measurements, it might be useful to have a function where the user can easily control the time applied of each step in the potential scan range. To achieve this goal, a constant potential over time function was built, as shown in the Figure 8.

This function applies an initial constant potential and then the user can change the applied potential by rotating the potentiometer connected to the U1 introducing an offset. Also, this function prints the time of the experiment for EC-SERS measurements over time.

```
//constant potential over time function
void loop() {
   for(int i=0;i<1200;i++){
       analogWrite(3,150);//applied potential value
       float times = millis()/100; //prints time since program started
       Serial.print(times);
       Serial.print(" ");
       findAverage();
       delay(250);// wait a moment to not send massive amounts of data
  }
  analogWrite(2,LOW);
  Serial.flush();
  while(1);//leaves arduino in a invisible loop
 }
```

Figure 8 - Constant potential over time function

2.5 Processing software

All the values measured by the Arduino Board when cyclic voltammetry is performed are sent to the Serial port which can then be seen in the serial monitor. However, is it of interest to save de values in a text file for further analysis. Also, it is very handy to see the cyclic voltammogram in real time for the user to have a better perception of what is happing when the electrochemical experiment is running.

Processing is an open-source programming software sketchbook which has a Serial library that can read serial values from a USB port, it also has the Print Writer library that can create or open text files and print values in it. Also has the function that creates windows for drawing geometric shapes on it.

By this means, *Processing* software was used. The function developed draws a graph where the x axis is the voltage and the y axis is the current. Then it reads in cycle the Serial value from a specified USB port and converts it from a string to a float number. Then this float number is sized to the axis using the map function and finally it is printed on a graph in the drawing window and it is saved in a text file. This function works until the last value of the Serial port is read. The function code is in Appendix.

2.6 Potentiostat circuit on a breadboard

The potentiostat circuit was first mounted and tested on a breadboard, as shown in the Figure 9.

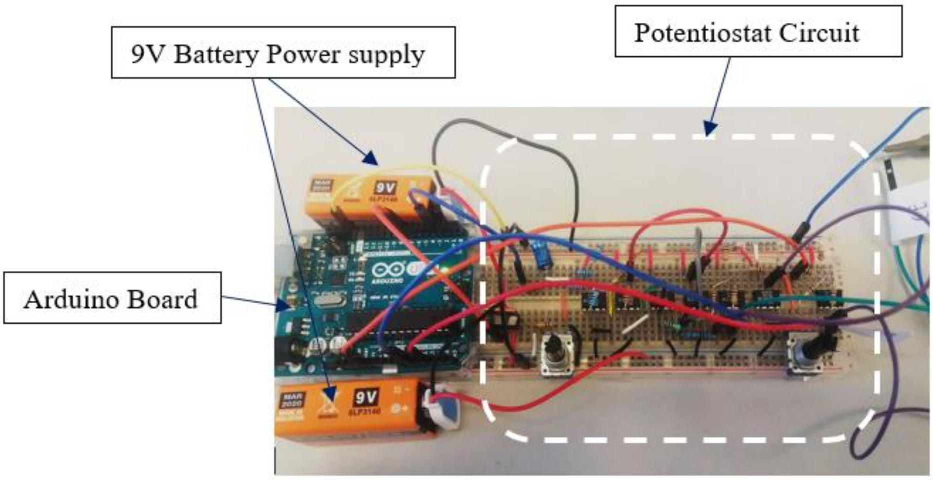

Figure 9 - Potentiostat circuit mounted on the bread board with highlighted the main parts.

The breadboard is known to be very versatile in terms of circuit development and testing. By this means, the potentiostat circuit was implemented and adjusted several times before it was transferred to a final copper strip board.

2.7 Transfer to an Arduino shield board format

To make the potentiostat more compact and robust the circuit was transferred to a copper strip Board. The components were soldered on the top side and the connections were made by soldering wires on the back side of the Board. This can be visualized in the Figure 10.

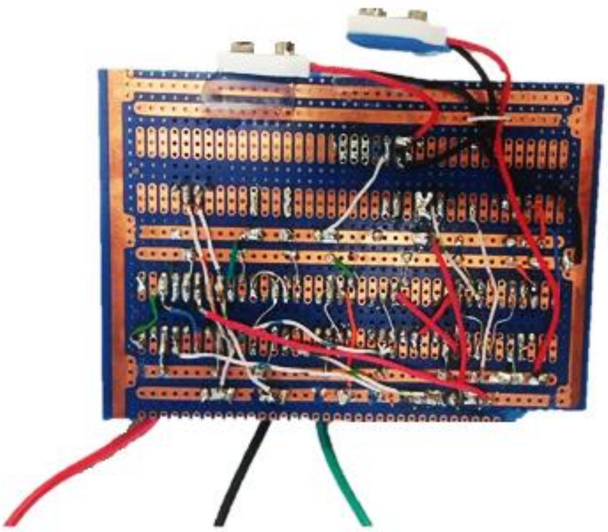

Figure 10 - Back side of the Potentiostat board showing the connections of wires and solder.

The Potentiostat board fits on the top of the Arduino and the connections were made using soldered pins, as shown in the Figure 11.

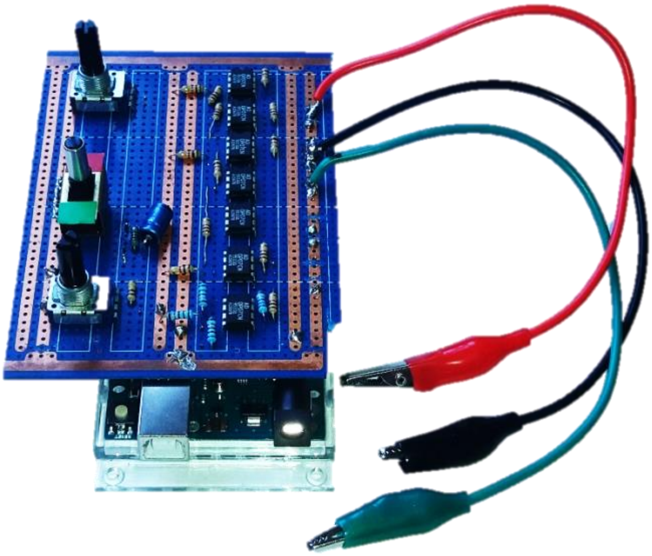

Figure 11 - Potentiostat Arduino Shield Board

2.8 Potentiostat testing and current resolution

To evaluate the fabricated potentiostat and to verify if it was working as expected 10 a 20 kΩ resistors were used. The well-known response of a resistor also makes it a good component to evaluate the performance since when the voltage is swapped the measured current responds linearly with a slope value of (1/R). The resistors were connected between the working electrode and reference electrode (simulating the cell resistance) and a small resistor (100 Ω) was connected between the reference and the counter electrode to have the control of the potential applied. A CV of the resistors was recorded it is shown in Figure 12.

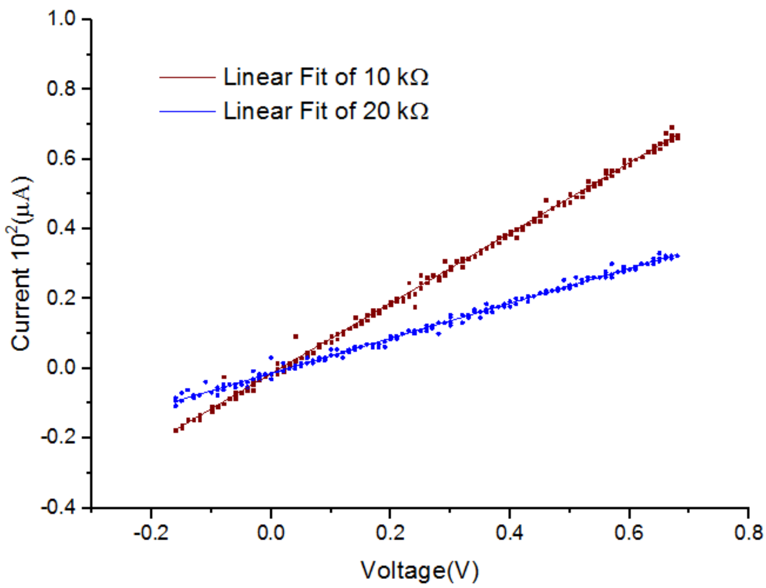

Figure 12 - IV curve of the 10 and 20 kΩ resistors.

From the Table 1, the resistance values were calculated to be 9.992 ± 0,033 kΩ and 19.948 ± 0,098 kΩ for 10 and 20 kΩ, respectively. These values are in good agreement with 5% tolerance of the resistor used.

Table 1 - Values obtained of the linear fits in the Figure 12.

	Slope (10^{-4} Ω^{-1})	Slope $\sigma_{\bar{x}}$	R^2
10 kΩ	1.0078	0.00339	0.997
20 kΩ	0.5013	0.00248	0.995

The reference electrode voltage resolution was calculated to be the lowest resolution of the measurement circuit which is calculated in the equation 7.

$$10bit\ ADC\ resolution = \frac{5000\ mV}{2^{10}} = 4.8\ mV \approx 5mV \quad (7)$$

The current is measured through the transimpedance amplifier and the resistance R_f. The voltage drop across R_f was connected to one of the microcontroller's ADC channels. The current was determined by the implementation of Ohm's law in the control software program based on the measured voltage and the value of Rf.

The current measurement resolution depends on the voltage ADC resolution of the microcontroller's which is 5 mV, calculated in equation 7, and the value of Rf. The calculated minimum current detectable by the potentiostat circuit was thus determined using Ohm's law as show in equation 8.

$$\text{Resolution} = \frac{5\ mV}{100\ K\Omega} = 0.05\ uA \quad (8)$$

To assess experimentally the current resolution of the potentiostat, it was measured the maximum cell resistance that the potentiostat can read without introducing significant error. The gain potentiometer in the TIA was rotate to the maximum value of 100 kΩ, and the IV curves of 100, 200 and 400 kΩ resistor were measured, represented in the Figure 13.

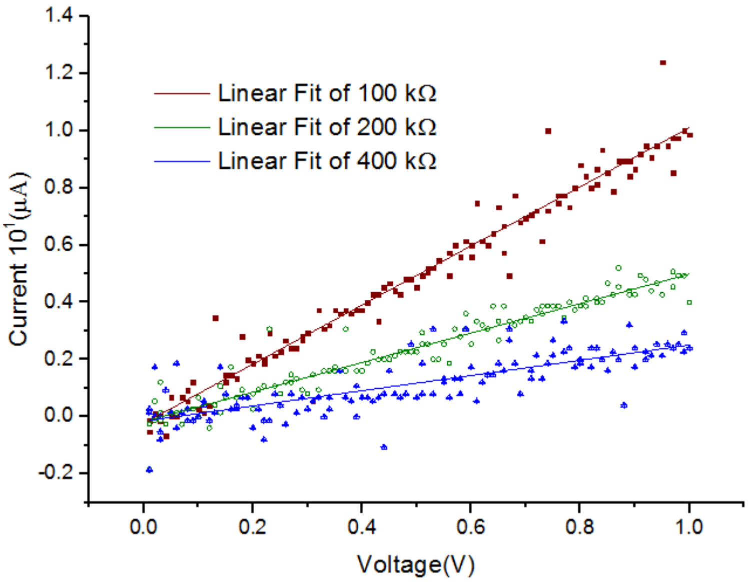

Figure 13 – IV curves of 100 KΩ, 200 KΩ and 400 KΩ resistors.

In the **Error! Not a valid bookmark self-reference.**, it can be observed that the higher the resistance the lower the R^2, and with 200 KΩ cell resistance the value of R^2 of the linear fit was higher than 0.9.

Table 2 - Values obtained of the linear fits in the Figure 13.

	Slope (10^{-5} Ω^{-1})	Slope $\sigma_{\bar{x}}$	R^2
100 kΩ	1.0343	0.01943	0.959
200 kΩ	0.5158	0.01339	0.925
400 kΩ	0.2662	0.02006	0.595

By using the 200 KΩ resistor, the current was measured over time under different applied potentials, see Figure 14, and as can be observed the current values can be discriminated from each other although the signals show significant noise

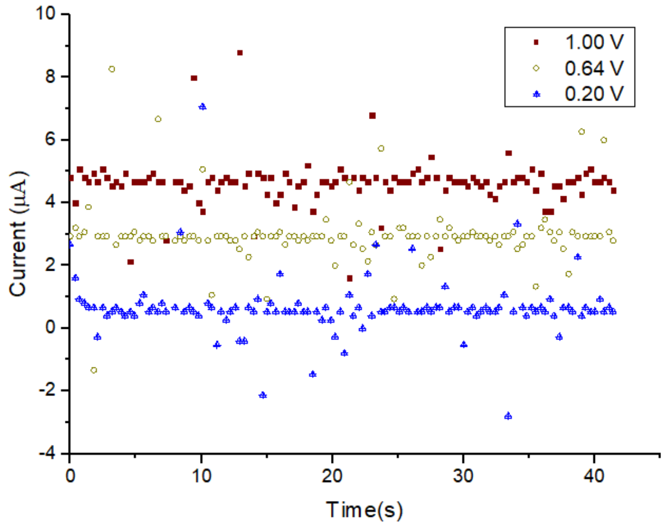

Figure 14 – Current measured over time using a 200 KΩ resistor for different applied voltages.

However, the calculated current resolution value was 0.05 µA, experimentally it was determined to be in the range of 1 µA due to the noise.

3 Electrochemical performance

3.1 Design of an electrochemical cell

A simple Electrochemical Cell was developed to perform electrochemical measurements, Figure 15. The support structure is formed by two rectangular shaped slides with the dimensions of a standard glass slide and some screws to attached both parts together. The cell is the round centered hole in the upper slide. All the pieces are made of non-conductive polymer and to prevent solution leakage it was inserted an o-ring with the same diameter of the cell between the slides.

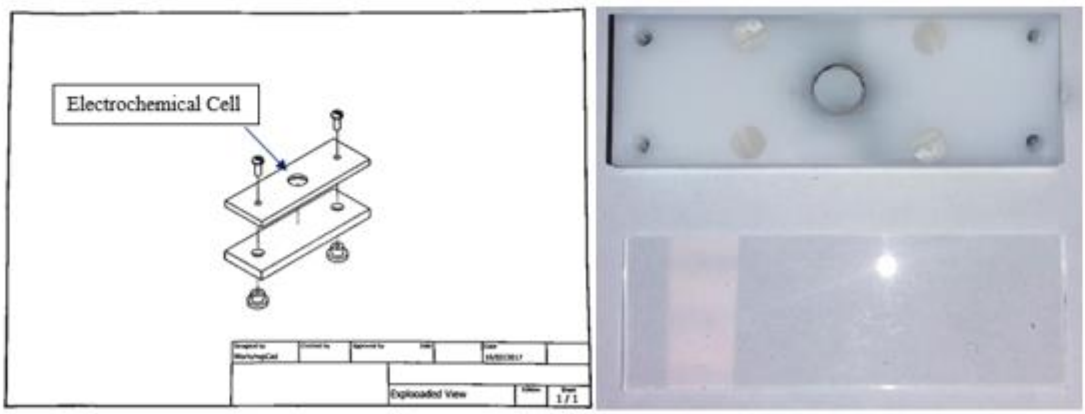

Figure 15- (left) Electrochemical Cell CAD drawing. (Right) Photo of the electrochemical Cell and a glass slide to compare the size.

3.2 Electrodes

To perform cyclic voltammetry measurements in the developed electrochemical cell Platinum and Carbon electrodes (from Zimmer and Peacock) were used and they are shown in the Figure 16. To make the contact between the wires and the screen-printed electrodes silver paste was deposit and left dry during night. Then UV curable glue was deposit on top of the contacts to provide a stronger support and help isolate one contact from each other.

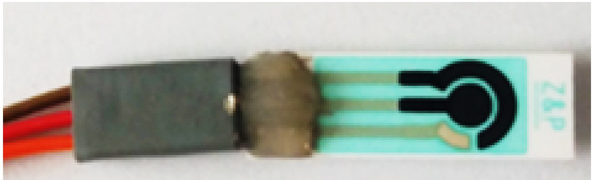

Figure 16 – Screen printed electrode, CE and WE are made of carbon and the reference electrode is Ag/AgCl.

3.3 Comparison with a commercial potentiostat

To evaluate the performance of the potentiostat, cyclic voltammetry was performed. The standard reversible Ferricyanide-Ferrocyanide redox couple was used.

A cyclic voltammetry comparison between the developed and a commercial potentiostat (Ana Pot from Zimmer and Peacock company) is shown in the Figure 17. It is possible to see the oxidation and reduction peaks occurring at the same potentials in both potentiostats. The same peak current is visualized in both cyclic voltammograms but there is a constant offset of the current measured between the cyclic voltammograms. This offset can be explained by a difference of the nominal resistance value from the 1 kΩ resistor where 3.3 V are added in the TIA. In the code 3.3 V are assumed to be added but by a slight change is the resistor values the offset value added can be different than 3.3 V.

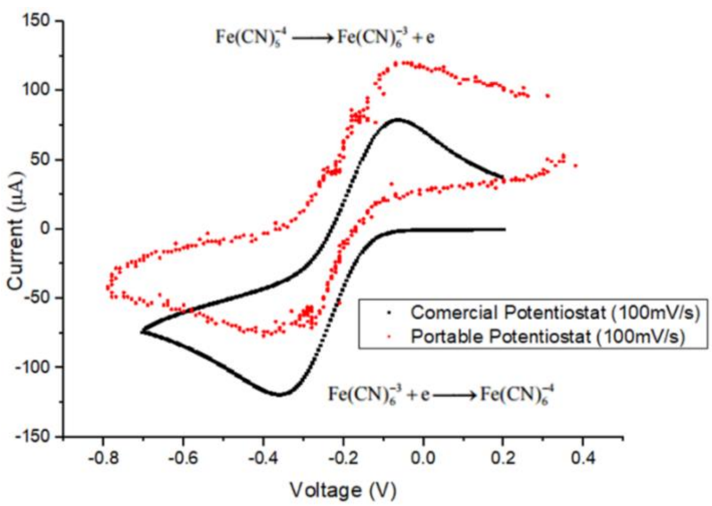

Figure 17 - Cyclic Voltammetry of 10mM of K3[Fe(CN)6] in 0.1M of NaCl for comparison of the developed and a commercial potentiostat. Scan rate of 100 mV/s.

3.4 Diffusion coefficient determination

To show the potentiostat capabilities cyclic voltammograms were also performed at different scan rates and by doing this the diffusion coefficient of potassium ferricyanide was calculated in a solution containing a known concentration of the salt using the Randles-Sevcik equation at 25 °C:

$$i_p = 2.69 \times 10^5 \times n^{\frac{2}{3}} \times A \times D^{\frac{1}{2}} \times C_0 \times v^{\frac{1}{2}} \quad (9)$$

The n is number of electrons transferred, the A is the working electrode area in cm^2, D is diffusion coefficient in cm^2/s, C_0 is concentration in mol/cm^3 and v is the scan rate in V/s.

The Carbon working electrode with a diameter of 4 mm. The electrochemical cell was filled with 5 mM potassium ferricyanide (from Sigma Aldrich) in 0.1 M NaCl (from Breckland Scientific Supplies Ltd) solution. The potential was swapped between 0.35 V and −0.85 V with scan rates of 0.8, 0.11, 0.14, 0.25, and 0.35 $V.s^{-1}$. The recorded data can be seen in Figure 18.

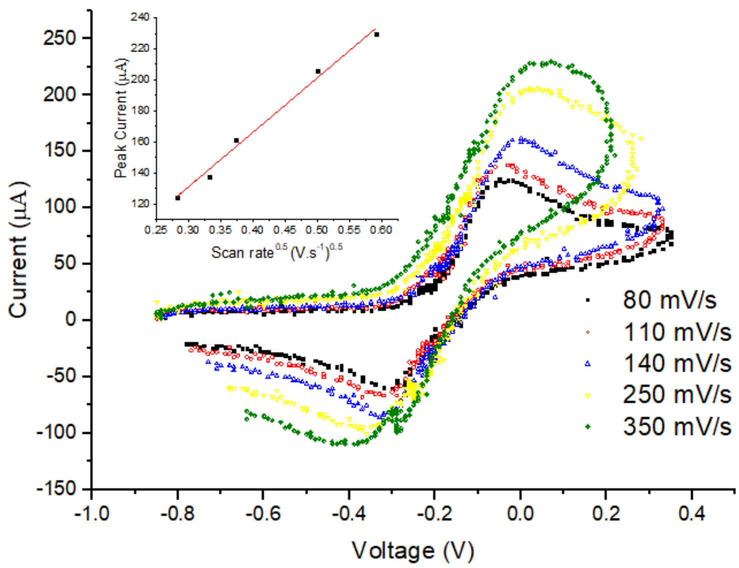

Figure 18 - Cyclic Voltammetry of 5mM of K3[Fe(CN)6] in 0.1M of NaCl at different scan rates. (Inset) Linear fit of the anodic peak currents against the square root of the scan rate.

As expected, Figure 18 shows an increase in the recorded electrochemical current with an increase in the scan rate. The inset shows the plot of the anodic peak current vs the square root of the scan rate. As expected, the plot is linear with a R^2 of 0.9993. From the calculated slope of the linear fit, using the Randles-Sevcik equation, the diffusion coefficient of potassium ferricyanide is calculated to be 4.3×10^{-6} $cm^2.s^{-1}$ which is in good agreement with the value in the literature [6] The difference between the values can be caused by the previous calibration of the scan rates before CV's were performed.

4 Conclusions and perspectives

The main objective of this work was to develop and test a simple, cost effective Arduino-based potentiostat. It was demonstrated that it can perform electrochemical measurements, such has cyclic voltammetry or potential step voltammetry. Also, through software, the scan rate of cyclic voltammetry measurements can be tuned. The data acquisition through the serial USB interface can be displayed in real time using the open-source processing software environment as a serial terminal visual interface, which can also save the data in text file format for analysis.

References

[1] K. S. Kaswan, S. P. Singh, and S. Sagar, "Role of Arduino in real world applications," *Int. J. Sci. Technol. Res.*, vol. 9, no. 1, pp. 1113–1116, 2020.

[2] A. J. B. and L. R. Faulkner, *Electrochemical methods: fundamentals and applications*, 2nd ed. New York, 2001.

[3] A. Scozzari, "Electrochemical Sensing Methods: A Brief Review," in *NATO Science for Peace and Security Series A: Chemistry and Biology*, 2008, pp. 335–351.

[4] R. K. Franklin, S. M. Martin, T. D. Strong, and R. B. Brown, "Chemical and Biological Systems: Chemical Sensing Systems for Liquids," in *Reference Module in Materials Science and Materials Engineering*, Elsevier, 2016.

[5] B. Aremo, M. Oyebamiji Adeoye, I. B. Obioh, and O. A. Adeboye, "A Simplified Microcontroller Based Potentiostat for Low-Resource Applications," *Open J. Met.*, vol. 05, no. 04, pp. 37–46, 2015, doi: 10.4236/ojmetal.2015.54005.

[6] G. N. Meloni, "Building a microcontroller based potentiostat: A inexpensive and versatile platform for teaching electrochemistry and instrumentation," *J. Chem. Educ.*, vol. 93, no. 7, pp. 1320–1322, 2016, doi: 10.1021/acs.jchemed.5b00961.

Appendix

Processing code

```
import processing.serial.*;
Serial mySerial;
PrintWriter output;
void setup() {
  mySerial = new Serial( this, "COM3", 9600 );//choose the USB port for serial
imput
  output = createWriter( "data.txt" );//creates a text file in the same
directory of the Sketch
  size(600, 400); // set the window size:
  background(255);// set initial background
  strokeWeight(2);  // Default
  line(40, 360, 580, 360);//x axis
  line(40, 40, 40, 360);//y axis
  line(40, 360, 40, 365);//x line lower limit
  line(580, 360, 580, 365);//x line upper limit
  line(302, 360, 302, 365);//0 line
  textSize(15);
  fill(0, 102, 153);
  text("Voltage (V)", 285,390);//X axis label
  fill(0, 102, 153);
  text("-1.2", 30, 380);
  fill(0, 102, 153);
  text("0", 300, 380);
  fill(0, 102, 153);
  text("1.2", 575, 380);
  fill(0, 102, 153);
  rotate(-PI/2);
  text("Current", -235, 30); // y axis label

}
//function that reads the serial values as strings and converts them in num
bers then it plots them and prints them on text file
void draw() {
  if (mySerial.available() > 0 ) {
    String value = mySerial.readStringUntil('\n');
    if ( value != null ) {
      println(value);
      int p1 = value.indexOf(" ");
      int l=p1-1;
      String ss = value.substring(0, l+1);
      String ss1 = value.substring(p1);
      float voltage = float(ss);// convert to a number.
      float current = float(ss1);// convert to a number.
      println(voltage + " " + current);
      output.println(voltage + " " + current);
      float volt = map(voltage, -1.2, 1.2, 40, 580);//map to the screen width
      float curr = map(current, -1, 1, 360, 40); //map to the screen height
      ellipse(volt, curr, 3, 3);
    }
  }
}
//funtion to close the plot and save the file when any key of the keyboard
is pressed
void keyPressed() {
```

```
  output.flush();  // Writes the remaining data to the file
  output.close();  // Finishes the file
  exit();  // Stops the program
}
```