***C-Stem Summer 2018***

**Hair Strength Measurement System**

***RPI Class of 2019***

*--Nicolas Cheng*

*Instructor: James Davis*

## Introduction

Nowadays, most salons in the United States provide only one type of hair treatment. However, because different people have different hairs, applying the same treatment to everyone is likely to bring damage to most people. Therefore, the development of specialized treatments that focuses on different kinds of hair is a crucial step for salons in the future. The project that we have been working on is designed to create an inexpensive suite of tools to help the whole community engage in scientific inquiry, improve hair care, and debunk phony advertising. More importantly, by conducting this research project, it is beneficial to attract more students to know the STEM education.

One of the most crucial characteristics of hair is its tensile strength, and by designing an accurate and easy-operating system to detect hair’s tensile strength, it would be more convenient to tell the difference that different hair products make to hair health. This report focuses on the design of a high-accuracy hair tensile strength measuring system.

## Background information

Hair characteristics differ according to age, race and treatment. However, the composition of hair is always the same. A hair is made up from three parts: medulla, cortex and cuticle. [3]

Cuticle is the outer layer of a hair. It is made of deal cells, overlapping in layers and serves for two main functions: block moisture from leaving the hair and protects the cortex. The cortex, made of braided fibers, determines most of characteristics of hair – color, width, style and strength. The medulla is at the center of hair. It contains air only and does not decide a hair’s property. When different hair products are used on hair, it would open up the cuticle first and then apply changes on cortex properties, by which the hair characteristics can be altered. Relative treatments include hair perm, hair dye and hair bleach. [2]

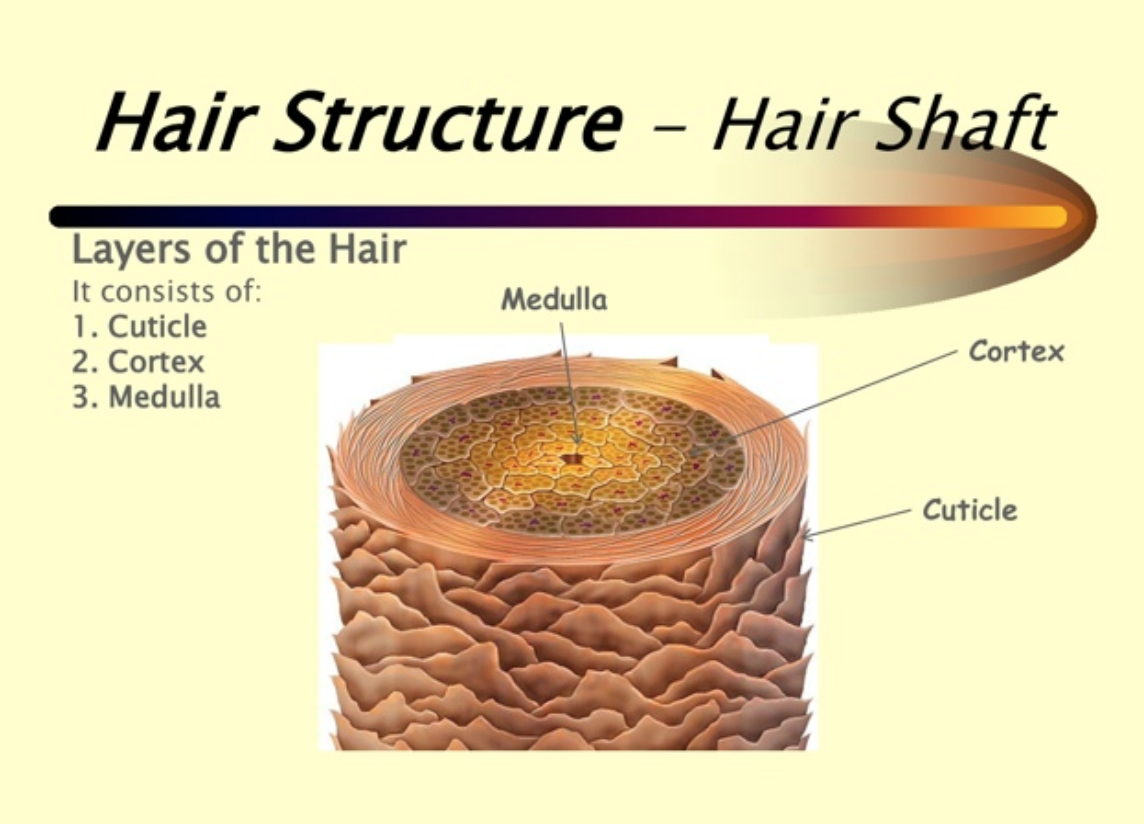


Figure 1 - Hair Structure Breakdown[1]

## System Design

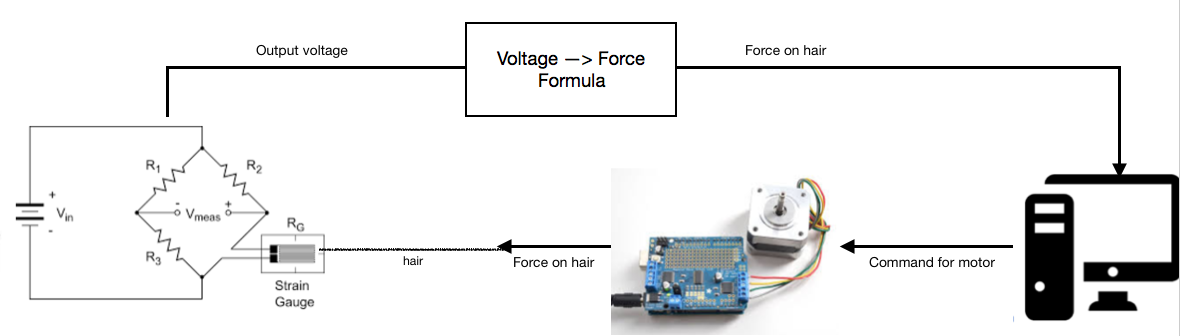


Figure 2 - System Summary [5][8]

The whole system consists of four main parts: Motor, Arduino (with motor shield), electrical circuit and strain gauge. The function of each part is summarized in the table below:

|  |  |
| --- | --- |
| Part | Function |
| Motor | Provide pulling force on hair |
| Arduino | Provide measuring and motor operational platform |
| Circuit | Transfer strain gauge resistance change to voltage |
| Strain Gauge | Provide output according to force applied on hair |

Table 1 - Functions of different parts

With the rotation of motor, an increasing force will pull the hair until its breakage. During this process, the force that applied on the strain gauge will increase as well, which alters its resistance. The circuit contains a wheatstone bridge to accurately capture changes in strain gauge’s resistance and op-amp, which offsets the output to 0-5V for the Arduino to read. The voltage to force formula is obtained by getting enough voltage-force data points of the system, and building a second order polynomial fit model on Matlab. In this way, it is convenient to see how much force is applied on the force by finding the matching voltage on the graph.

## System building

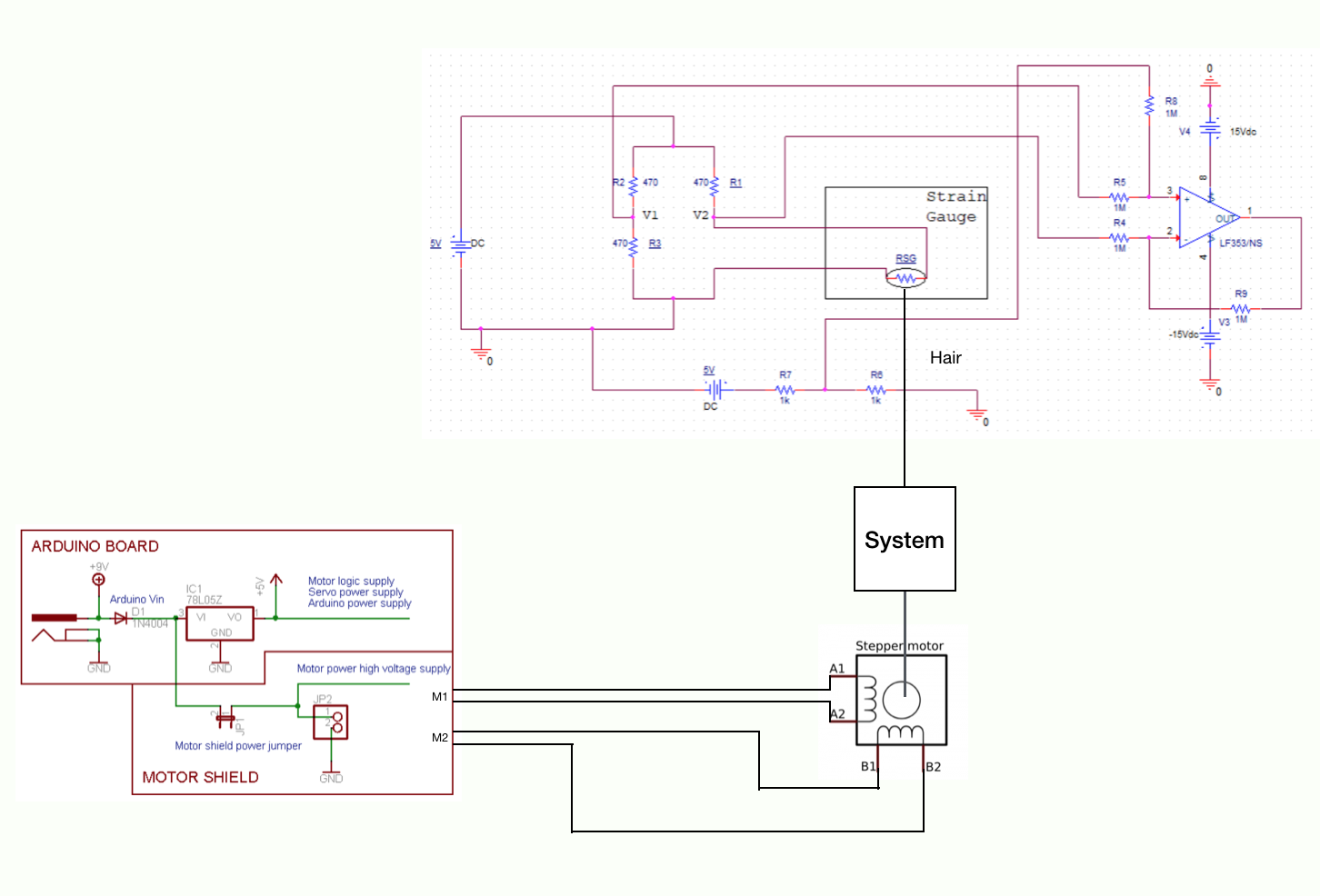


Figure 3 - Syetem breakdown

The strain gauge is fixed on the board by using tape to cover the end of it. However the tae itself is not enough to hold the strain gauge from falling off. Therefore, glue is used between strain gauge and tape to increase the maximum friction force. The hair is connected to the strain gauge by going through a tube connected to the feet of it. On the other end of the hair, the spinning axis of motor is also attached a tube for the hair to go through. Therefore, when the motor starts spinning, its rotation will lead the hair to twine around the rotor and exert force on the tube on strain gauge. As the rotation of rotor continues, force on the hair will keep increasing until its breakage. At its breakage, the resistance of the strain gauge is also the largest, which minimizes the voltage output. This voltage number can then be used to determine the maximum force the hair can handle.

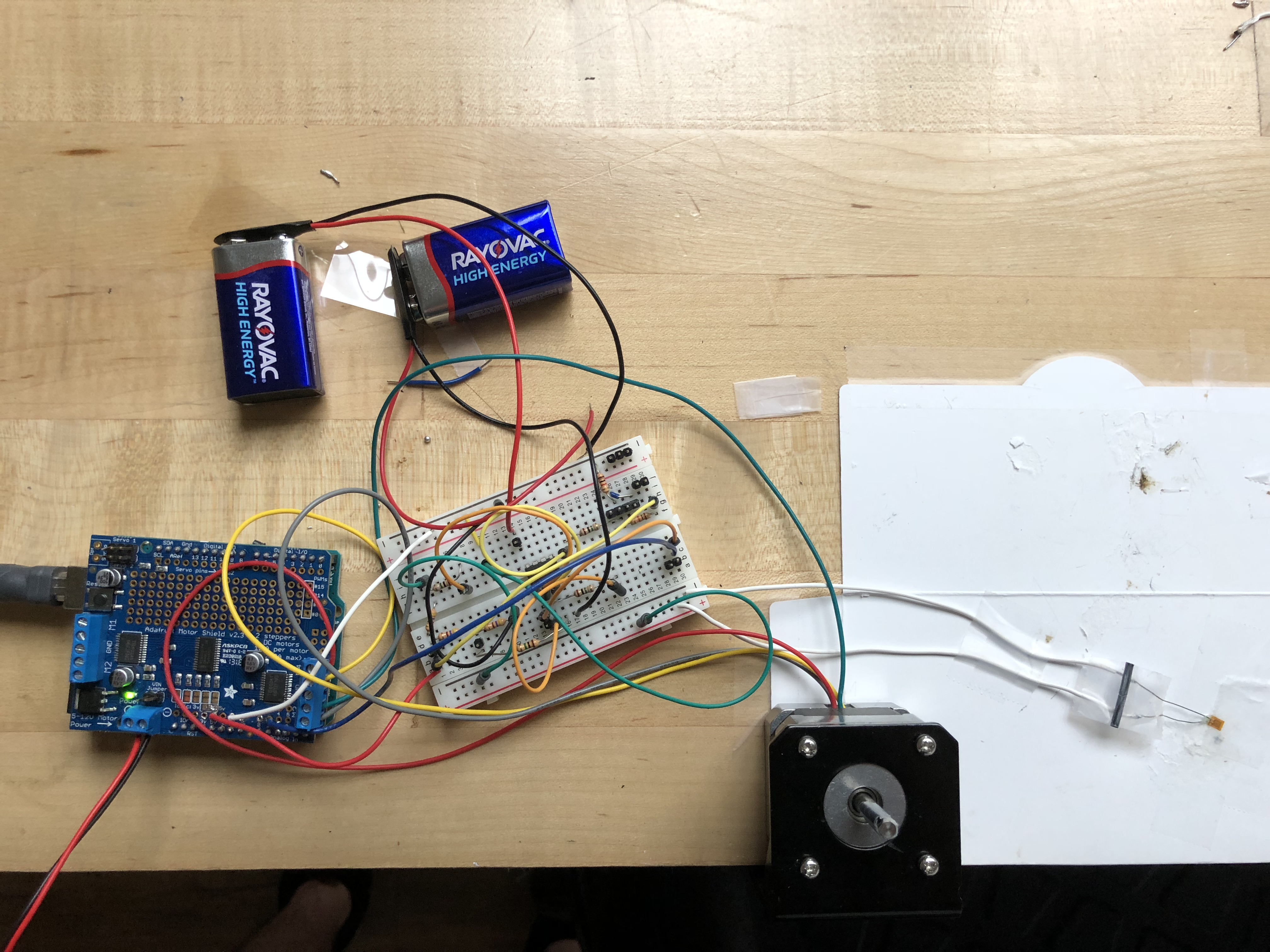


Figure 4 - Actual system buildup

## Data points obtaining

Before doing the actual hair testing, some data points need to be obtained to build a voltage-force curve of the specific strain gauge. The data points obtained is shown below:

|  |  |
| --- | --- |
| Force (N) | Voltage Output |
| 0 | 2.79723 |
| 0.25 | 2.78178 |
| 0.5 | 2.75993 |
| 0.75 | 2.70889 |
| 1 | 2.62578 |
| 1.25 | 2.54302 |
| 1.5 | 2.48983 |
| 1.6 | 2.45532 |
| 1.75 | 0 |

Table - Voltage Force relation

The reason that at voltage jumps to 0 when force equals 1.75N is because that there is a limit range of how much force the train gauge can measure, and 1.75N has exceeded that range. Therefore, this pair of data will simply be abandoned.

By using Matlab, a fit curve is drawn with the least error range. Based on the data above, the 2nd polynomial equation is:

f =

Linear model Poly2:

f(x) = p1\*x^2 + p2\*x + p3

Coefficients (with 95% confidence bounds):

p1 = -6.935 (-16.5, 2.634)

p2 = 32.29 (-18.05, 82.64)

p3 = -35.93 (-102, 30.16)

In addition, the fir graph(in blue) is:

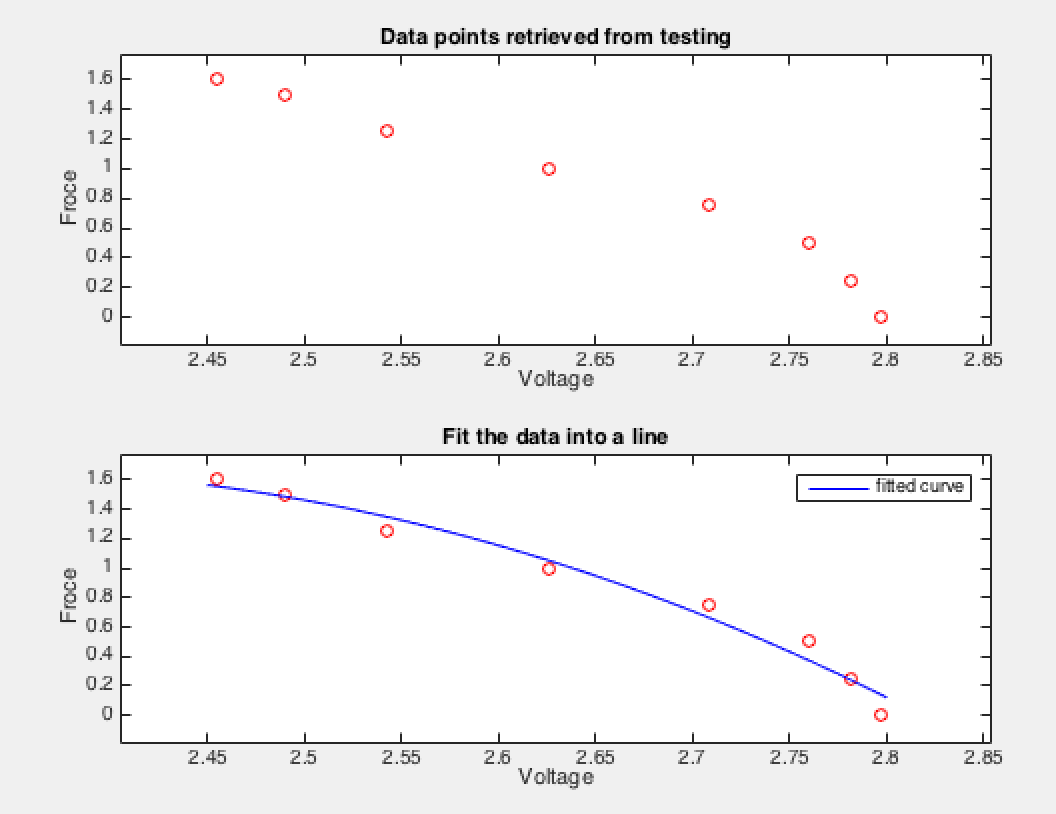


Figure 5 - Fit curve for data points

The code of how this graph is acquired in in appendix B.

## Testing result

There are 16 hairs in total in the testing procedure. Each of them follows the same procedure and 5 of the testing results are re-validated by using the old clip and dynamometer measurement method. For these 5 hairs, the errors are rather small and differences are all smaller than 0.1 Newton. Then all 16 hairs are treated with the same concentration hair bleacher for 15 minutes, before they get dried out completely and the same testing process happens again.

The data set is presented below:

|  |  |  |  |
| --- | --- | --- | --- |
| Before Treating | Force (By arduino system) | Force (By strength tester) | After Treating |
| 1 | 0.7 |  | 0.5 |
| 2 | 1.25 | 1.2 | 0.8 |
| 3 | 0.85 |  | 0.55 |
| 4 | 1.4 |  | 0.6 |
| 5 | 1.3 | 1.3 | 0.7 |
| 6 | 1.1 |  | 0.45 |
| 7 | 1.1 |  | 0.6 |
| 8 | 1.75 |  | 0.95 |
| 9 | 1.05 | 1.1 | 0.65 |
| 10 | 1.4 |  | 0.75 |
| 11 | 1.5 |  | 0.7 |
| 12 | 1.15 |  | 0.5 |
| 13 | 0.9 | 0.9 | 0.6 |
| 14 | 0.75 |  | 0.25 |
| 15 | 1.3 |  | 0.6 |
| 16 | 1.2 | 1.1 | 0.55 |

Table 3 - Hair data before & after treatment

By entering these data into matrices and using Matlab, three graphs can be drawn – One distribution graph each for before and after treatment, and one for distribution curve comparison.

Before treatment:

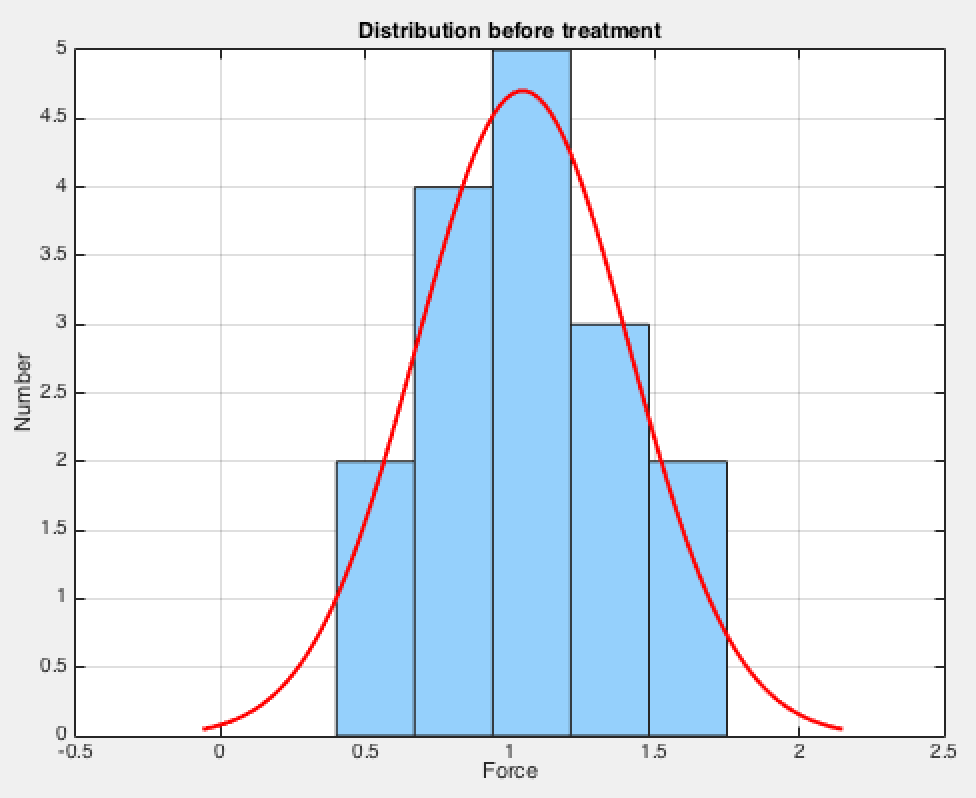


Figure 6 - Distribution before tratment

After Treatment:

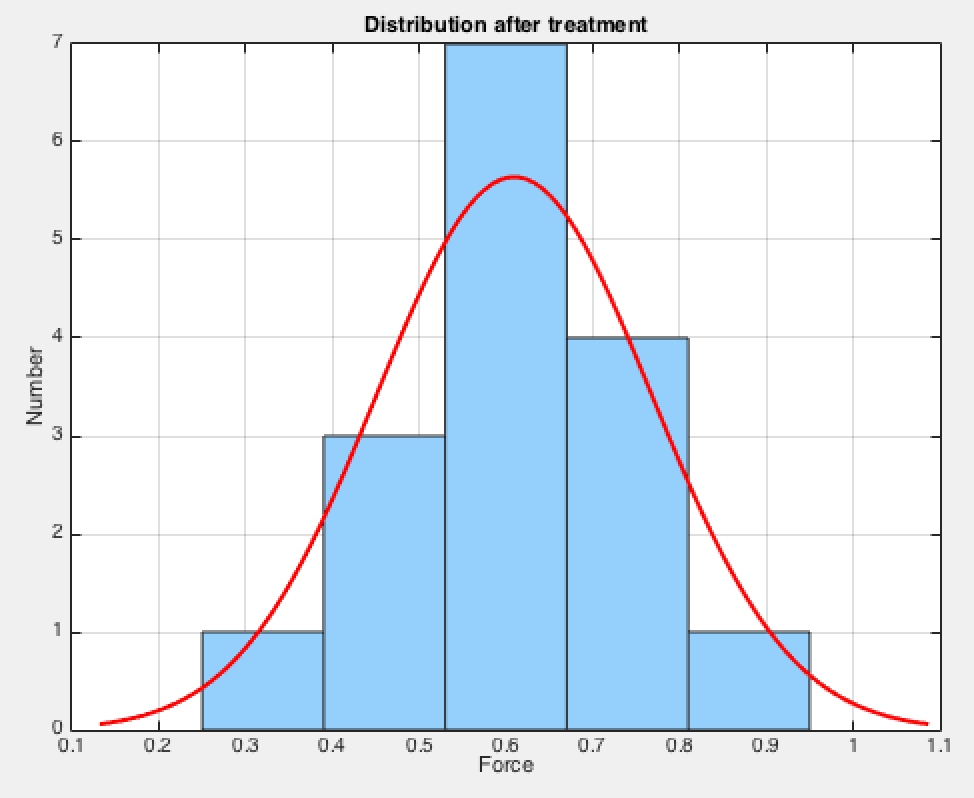


Figure 7 - Distribution after treatment

Comparison of distribution:

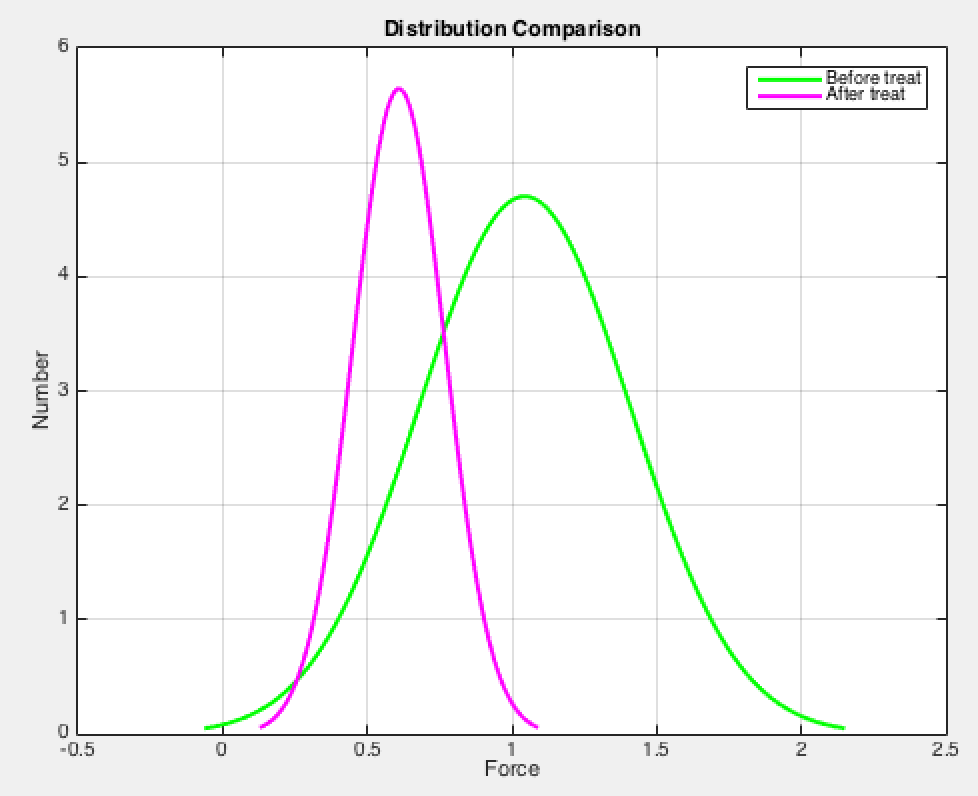


Figure 8 -Distribution curve comparison

## Conclusion:

The treatment does have a substantial effect on the hair, as the average tensile strength drops from 1.1N to 0.6N, which is almost half of the strength. Although some hair does not follow the general rules, such as No.1 hair only drops from 0.7 to 0.5 (28%) and No.14 drops from 0.75 to 0.25 (67%), the distribution curve does represent the major characteristics of strength change. For this bleacher, we can draw the conclusion that it damages the hair strength to half of its original value.

Some interaction between C-Stem and local students & teachers happened as well during the project, and although with some difficulties, most students and teachers were able to build their own PH-Sensor system to help determine the characteristics of bleacher. Meanwhile, more than 50% of the participants gave positive feedbacks about this experience and suggest that they would love to do more relative things in the future. Therefore, the attempt to attract more students to the STEM relative area tends to have a prospective result if similar activities can be held in the future.

## Future development

There are several more things to be accomplished in the future. First is that so far only one treatment solution is applied so only one comparison curve can be drawn. In the future, more solutions shall be used to treat the hair and draw graphs to see different effects that different hair products have on the hair.

In addition, a simple C++ program is designed to let equations be inserted and output the force with an input voltage ( In appendix C). However, the fit function outputted by Matlab gives a large error range when the same equation is applied in C++, so the accuracy of program is affected severely. In the future, the accuracy of this program should be improved to make it fit for different strain gauges.

## Reference

1. M.L.Tate, Qualification and prevention of hair damage (Dec, 1993) <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.471.2274&rep=rep1&type=pdf>
2. Ladyno Green, Hair Structure (May, 2013) <http://ladynohair.blogspot.com/2013/05/hair-structure.html>
3. Ashish, Why some people have curly hair ( July, 2016) <https://www.scienceabc.com/humans/why-do-some-people-have-curly-hair-while-others-have-straight-hair.html>
4. Hyung Jin Ahn, Ultrastudy of hair damage and restoration (Apr, 2002) <https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1365-4362.2002.01375.x>
5. [Will Delaney](https://www.researchgate.net/scientific-contributions/2144631948_Will_Delaney?_sg%5B_sg%5D=3z3lJk_bJsBWCOXGE41oW0CxQJx-IUOdoTwyto4rSUVoN_0Xd-SXdAQn7jWrJmy_wcvNozRPTqruBQ), Design of a Cross Curriculum Lab Experiment (Jun, 2018)

<https://www.researchgate.net/figure/Wheatstone-Bridge-with-Strain-Gage-Lab-Test-Setup_fig1_326211964>

1. Texas Instruments, µA741 General-Purpose Operational Amplifiers (Jan 2018)

<http://www.ti.com/lit/ds/symlink/ua741.pdf>

1. Texas Instruments, LF353-N Wide Bandwidth Dual JFET Input Operational Amplifier (March, 2013)

<http://www.ti.com/lit/ds/symlink/lf353-n.pdf>

1. Lady Ada, Using Stepper Motors (July 2013) <https://learn.adafruit.com/adafruit-motor-shield-v2-for-arduino/using-stepper-motors>

### Appendix A – Circuit building component choosing

In the circuit that contains Wheatstone bridge and offset op-amp, originally UA741 was used as op-amp. However, due to the fact that ripple voltage goes too high which affects accuracy of circuit, some comparisons between different op-amps took place. The comparison data illustrates why LF353 is chosen for the final project design.

|  |  |  |
| --- | --- | --- |
|  | UA741 | LF353 |
| Input Offset Current (nA) | 10 | 0.025 |
| Input Bias Current (nA) | 80 | 0.05 |
| Input Impedance (Megohm) | 2 | 1.00E+06 |
| Output Impedance (ohm) | 75 | 100 |

Table 4 - Op-amp comparison[6][7]

The testing for circuit took place after the reinstallation of LF353, in which different values of resistor are chosen in the place of strain gauge to simulate resistance change. The error range less than 5% suggests that the circuit is qualified for further operations.

Testing data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resistor | Expected Output (V) | Actual Output (V) | Error (mV) | Error (%) |
| 300 | 3.052 | 3.069 | 17.000 | 0.557 |
| 347 | 2.876 | 2.895 | 19.000 | 0.661 |
| 470 | 2.500 | 2.525 | 25.000 | 1.000 |
| 670 | 2.061 | 2.091 | 30.000 | 1.456 |
| 900 | 1.715 | 1.733 | 18.000 | 1.050 |
| 1000 | 1.598 | 1.631 | 33.000 | 2.065 |
| 1800 | 1.035 | 1.059 | 24.000 | 2.319 |
| 2800 | 0.721 | 0.754 | 33.000 | 4.577 |

Table 5 - Circuit testing result

### Appendix B – Matlab code

For data point obtainment:

%Data for untreated & treated hair obtained from testing

force=[0,0.25,0.5,0.75,1,1.25,1.5,1.6]';

voltage=[2.79723,2.78178,2.75993,2.70889,2.62578,2.54302,2.48983,2.45532]';

%Plot both graphs in one

figure(1)

subplot(2,1,1)

plot(voltage,force,'ro');

xlim([min(voltage)\*0.98,max(voltage)\*1.02]); %Make the data points away from the edge of graph

ylim([min(force-.2)\*0.9,max(force)\*1.1]);

xlabel('Voltage');

ylabel('Froce');

title('Data points retrieved from testing')

subplot(2,1,2)

plot(voltage,force,'ro');

hold on

%Obtain curve based on data points

f=fit(voltage,force,'poly2')

plot(f,'b-');

xlim([min(voltage)\*0.98,max(voltage)\*1.02]);

ylim([min(force-.2)\*0.9,max(force)\*1.1]);

title('Fit the data into a line')

hold off

xlabel('Voltage');

ylabel('Force');

For distribution plot:

%Data for untreated & treated hair obtained from testing

untreated = [0.7,1.25,.85,.4,1.3,1.1,1.1,1.75,1.05,.4,1.5,1.15,.9,.75,1.3,1.2];

treated = [.5 .8 .55 .6 .7 .45 .6 .95 .65 .75 .7 .5 .6 .25 .6 .55]';

%Untreated data distribution graph

figure(1)

h1=histfit(untreated,5);

set(h1(1),'facecolor','[0.5843 0.8157 0.9882]');set(h1(2),'color','r');

grid on

xlabel('Force');ylabel('Number');title('Distribution before treatment');

%Treated data distribution graph

figure(2)

h2=histfit(treated,5);

set(h2(1),'facecolor','[0.5843 0.8157 0.9882]');set(h2(2),'color','r');

grid on

xlabel('Force');ylabel('Number');title('Distribution after treatment');

%Comparison graph

figure(3)

h3=histfit(untreated,5);

hold on

h4=histfit(treated,5);

delete(h3(1));

set(h3(2),'color','g');

delete(h4(1));

set(h4(2),'color','m');

hold off

legend('Before treat','After treat');

grid on

xlabel('Force');ylabel('Number');title('Distribution Comparison');

### Appendix C – C++ Program for force calculation

H file:

#ifndef strain\_gauge\_h\_

#define strain\_gauge\_h\_

#include <cmath>

#include <vector>

#include <iostream>

using namespace std;

class strain\_gauge{

public:

strain\_gauge():sg\_factor(0){};

strain\_gauge(float sg\_factor\_temp):sg\_factor(sg\_factor\_temp){};

float get\_sg\_factor(){return sg\_factor;}

float get\_resistance(){return r\_original;}

protected:

float sg\_factor;

int r\_original;

};

vector<float> obtain\_polymals(){

cout << "Enter the polynomial number of equation: " << endl;

int length;

cin >> length;

vector<float> poly\_num(length+1);

for (int i=poly\_num.size()-1;i>=0;i--){

cout << "Please enter coefficients for the voltage's " << i << "th exponential." << endl;

float temp;

cin >> temp;

poly\_num[i] = temp;

}

cout << "Equation set for force calculation, to reset it, please enter 1, to continue please enter 2." << endl;

int command;

cin >> command;

if (command == 1){

poly\_num = obtain\_polymals();

}

return poly\_num;

}

float obtain\_force(vector<float> poly\_num){

cout << "To measure force, please enter the output voltage of circuit:" << endl;

int out\_vol;

cin >> out\_vol;

float force=0;

for (int i=0;i<poly\_num.size();i++){

float change;

change = poly\_num[i]\* pow(out\_vol,i);

force += change;

}

return force;

}

#endif

Testing file:

#include "strain\_gauge.h"

#include <cmath>

#include <vector>

#include <iostream>

using namespace std;

int main(){

vector<float> poly\_func = obtain\_polymals();

float force = obtain\_force(poly\_func);

cout<< "\n \*\*\*The force this hair can handle before breaking is: " << force << " Newton.\*\*\*" << endl;

cout <<"\nFor the next step:" << endl;

cout <<"\nTo keep using the same testing system, please enter 1.\nTo reset the system, please press 2, to exit, please enter any other number." << endl;

bool decide = true;

while (decide==true){

int input;

cin >> input;

if (input == 2){

vector<float> poly\_func = obtain\_polymals();

force = obtain\_force(poly\_func);

cout<< "\n\*\*\*The force this hair can handle before breaking is: " << force << " Newton.\*\*\*" << endl;

cout <<"\nFor the next step:" << endl;

cout <<"\nTo keep using the same testing system, please enter 1\nTo reset the system, please press 2, to exit, please enter any other number." << endl;

}

else if (input == 1){

force = obtain\_force(poly\_func);

cout<< "\n\*\*\*The force this hair can handle before breaking is: " << force << " Newton.\*\*\*" << endl;

cout <<"\nFor the next step:" << endl;

cout <<"\nTo keep using the same testing system, please enter 1\nTo reset the system, please press 2, to exit, please enter any other number." << endl;

}

else if (input != 1 && input != 2){

cout << "Thank you for using C-Stem hair force measuring system." << endl;

break;

}

}

return 0;

}