Low Cost PCR Microfluidic Device

Technical Report

Fall 2018 - Summer 2019

Vinh - Trung Trinh, Kevin Tang, Edward Dickhoff

Table of Contents

Abstract 2

Introduction 2

Design & Schematic 4

Materials Used 4

Source Code & Software Used 5

Design & Setup 5

Dimensions & Measurements 9

Calculations & Equations Used 10

Data 11

Dry Test 16

Wet Test 18

Future Works & Conclusion 21

Acknowledgements 22

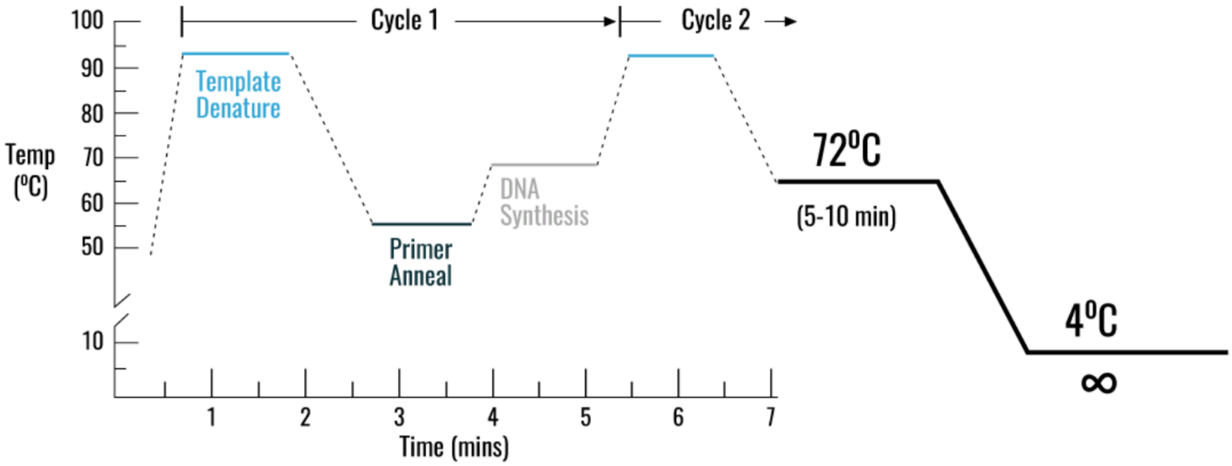
Sources & References 22

Abstract

A Microfluidic device is a small device dealing with the precise control and manipulation of small amounts of fluid. We are creating a microfluidic device in order to stimulate Polymerase Chain Reaction (PCR) on a smaller scale. A Polymerase Chain Reaction is a method of DNA replication of a single segment for scientific purposes and research. During PCR, DNA is heated and broken down to single strands, the strands are then cooled enabling the DNA to attach to the strand. Finally, the temperature is raised, allowing the new strand of DNA to be made. Repeating this process results in thousands of copies of a single DNA strand, which can be used in medical research. Using specially fabricated acrylic chips, Peltier devices, and microprocessors, we can replicate PCR on a smaller scale. DNA goes through thermal cycles done by Peltier devices within the microfluidic device, to replicate PCR. Most PCR devices are very hard to scale down and control. We are creating these Microfluidic devices in order to automate and accurately produce PCR at a smaller scale.

Introduction

Polymerase Chain Reaction, PCR, is a laboratory technique done to make copies of specific segments of DNA. There are three phases of PCR, done through thermocycling. The first phase, Denaturing, is done from 94 °C to 98 °C, to break down the DNA strands. The second phase, Annealing, is done from 50 °C to 65 °C, to rebuild. The third phase, Extension, is ran from 70 °C to 75 °C to reform the DNA strands. All phases are depicted in the image provided below.



The current method used to perform PCR is done with thermocyclers. They tend to be large and specialized, costing an average of 2500 USD. To be done, thermocyclers are used with large amounts of specialized tools and facilities, requiring specific workplaces and skills to use. Furthermore, PCR is naturally a two-step process. In one step, you have to perform PCR to be able to copy DNA strands and get more segments to actually analyze. In the second step, you have to actually analyze the DNA segments that have now been copied. This forces companies to essentially have to have the resources capable of supplying two different environments. As a result, the process of performing PCR is not streamlined and is hard to perform. With this current project, we worked to perform such a process at the microfluidic level, to have precise control and manipulation of fluids at a small-scale level. Performing PCR at a microfluidic level allows for decreased usage and cost, along with the ability to encompass a wide range of experiments.

Accomplishing such a project would allow us to perform PCR in a small scale, giving way for more accessibility with a low-cost device, and the ability to efficiently streamline a process that would be otherwise difficult to perform. Being able to achieve this would pave a path for labs that aren't as funded. Countries or areas around the world that don't have the resources or accessibility would be able to access a new way to make breakthroughs in their respective fields, helping provide research and analysis.

This research project had previously been assigned to past students in previous years to work on, though nothing was ever made of it. With it on the shelf and on hold, us three researchers were onboarded to the project at the start of the Fall 2018 quarter, thus beginning to work on the project throughout the rest of the school year, even giving a presentation at a symposium capping off the year spent working and researching. The presentation will be linked in the Sources portion of the report below. With the work done throughout this past academic year, we plan to continue working on this project throughout the next school year, 2019 - 2020. Any links needed to outside areas, such as materials or parts used will be embedded within the listings themselves.

Design & Schematic

Materials Used

- Thermal Grease x1 (Halnziye HY710 10g)
- Acrylic Blanks (sheets cut into chips according to needed measurements)
- Water Pump x1 (Harvard Apparatus, PicoPlus; Purple)
- Thermocouple Amplifier x2 (MAX31855)
  - Datasheet
- Peltier x2 (Single-Stage Thermoelectric Generator TG12-4, Marlow Industries)
- Arduino Uno x1 (ATMEGA328P - PU)

- Bantam Tools Desktop PCB Milling Machine x1
- FLIR C2 Compact Thermal Camera x1
- Clamps x2 (1 Dewalt, 1 unbranded)
- Solder
  - Soldering Iron
  - Soldering Stand
  - Wire Cutters
- Solderless Breadboard WB-106 x1 (CircuitSpecialists.com)
- Power Supply (Hewlett Packard E3630A, Triple Output DC Power Supply)
- Mosfet x2 (IRF840S Y31K)
- Digital Multimeter x1 (DT830B)
- Carbon Fiber Composites Digital Caliper x1

Source Code & Software Used

Software:

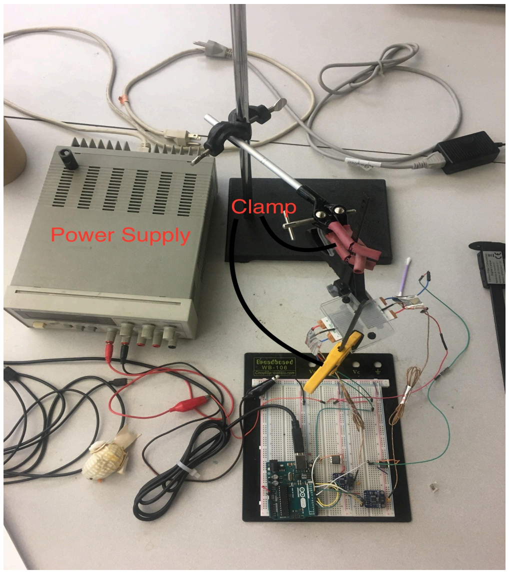
- Bantam Tools
- Inkscape (Tutorial)
- COMSOL (Tutorial)
- Arduino IDE

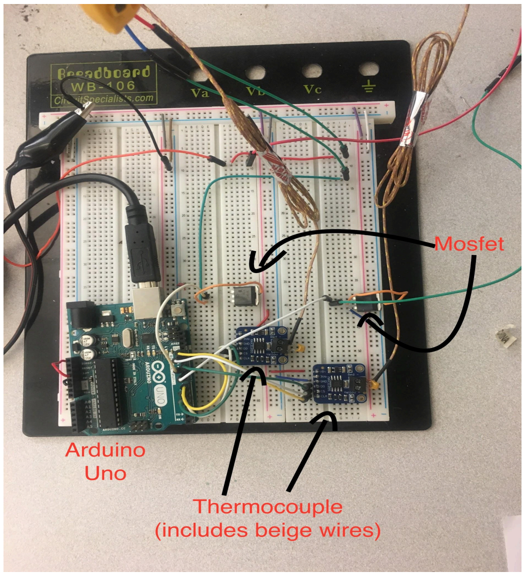
Source Code:

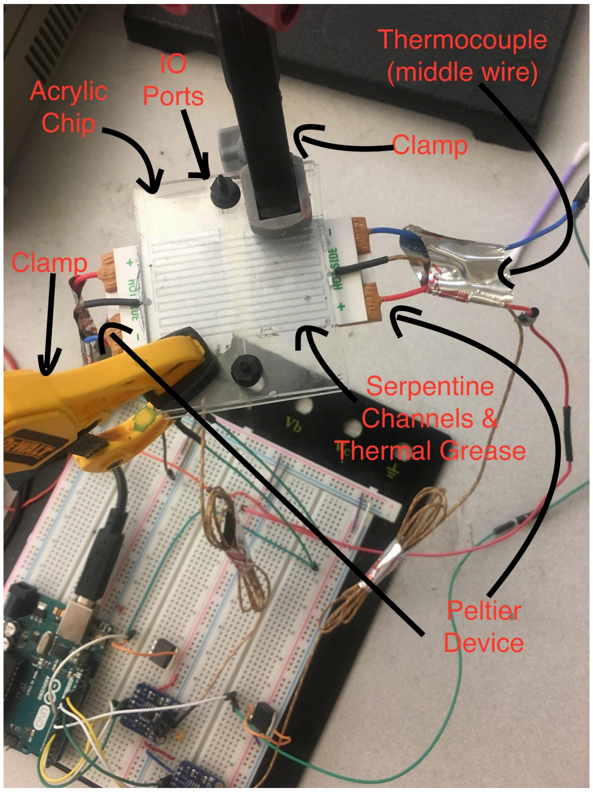
All code used was ran and developed in Arduino's IDE, and is given via GitHub below.

- Source Code

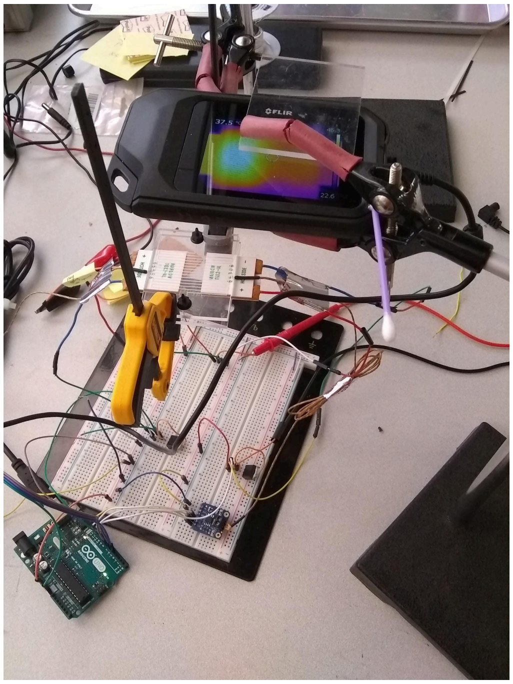
Design & Setup

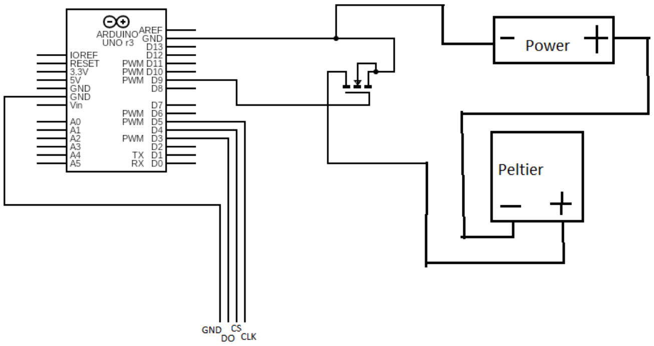


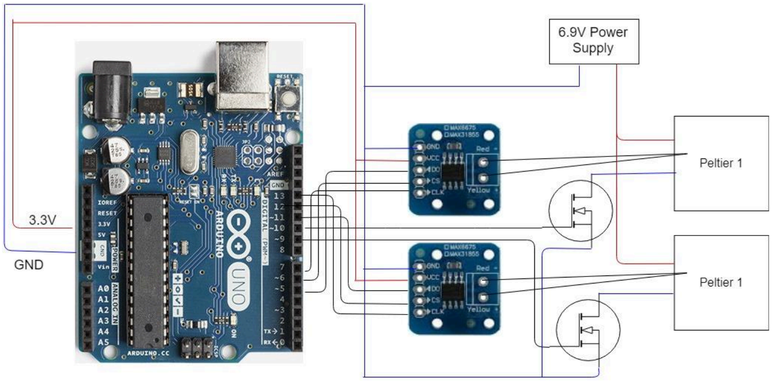



Thermocouple (middle wire)
IO Ports
Acrylic Chip
Clamp
Clamp
Serpentine Channels & Thermal Grease
Peltier Device

The images above help to show how the materials were used, and where such materials were located in our workstation. Below is a picture depicting where the thermal camera went to allow for thermal imaging.



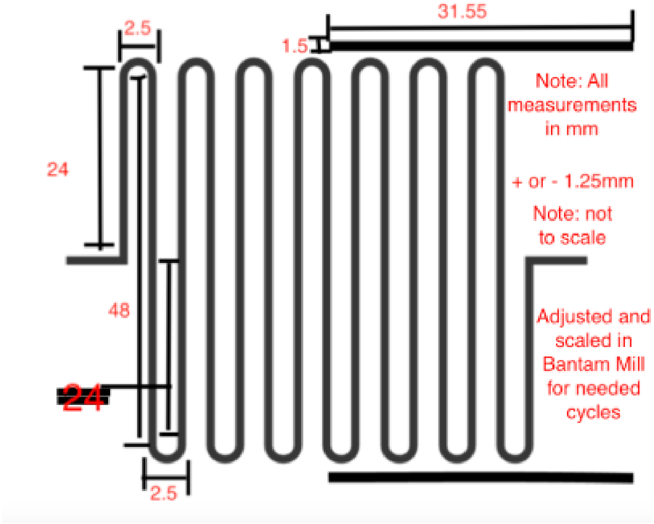




The shape and length of the channels in the device helps to dictate the time spent in the region by the liquid, thus determining the cycling and timing itself. The channels and flow rate are both also adjustable to account for different timings, allowing freedom for the user to analyze for specific needs through various cycling needs. The Arduino thus controls the

MOSFETS to run the peltier and constantly reads the thermocouples to dynamically maintain temperature.

To design the acrylic chip, we ran Inkscape and COMSOL. We used Inkscape to create and design the microfluidic channels themselves, saving such designs as SVG files. With such designs uploaded to COMSOL, we worked to analyze the flow rate and test liquid flow before milling physical copies onto acrylic chips. An example of our design on Inkscape is shown below. The bars on the bottom and top right were used to slide in the thermocouples to measure the temperature on both the hot side and cold side. The designs, created to be SVG files, are designed from Inkscape to set the channels and then uploaded to Bantam to mill the design onto an acrylic chip.


2.5
1.5
31.55
Note: All measurements in mm
24
+ or - 1.25mm
Note: not to scale
48
Adjusted and scaled in Bantam Mill for needed cycles
24
2.5

Note: Not drawn to scale
Include for error 0r + or -
1.25mm

Using such designs for the SVG files, we will most likely redesign the channels and the overall acrylic chip to run actual PCR, with reference to desired cycles and calculations. The chip used was made to simply conduct base tests and build a platform to run our initial conditions.

Dimensions & Measurements

The acrylic chips used were measured to be 2 inches by 2.5 inches, with about 3 millimeters worth of thickness. The channels had a depth of 0.15 millimeters, and a width of 0.35 millimeters. The maximum for the channels possible was calculated to be 0.26 millimeters by 0.5 millimeters.

The length of the thermocouple did not matter, as we had it inserted into both sides of the acrylic chip to measure both the hot and cold sides. The width of the thermocouple with a sleeve was measured to be 1.29 millimeters, and 0.50 millimeters without a sleeve (radius 0.25 millimeters). Two peltiers were used on each side of the acrylic chip, to essentially set both a hot and cold side. The holes drilled for the I/O ports ended up having a diameter of 0.159 inches per hole.

Calculations & Equations Used

The overall length travelled by the liquid through the acrylic chip was measured to be 101 millimeters, with a cycle time calculated to be 3.98 minutes, or 3 minutes and 58.8 seconds per cycle. For example, the serpentine design we used to run tests included 7 cycles, calculating to be 27.86 minutes, or 27 minutes and 51.6 seconds. Based off of these values, it is possible to adjust the channel width and length, or slow the pump, to change the cycle times or PCR reaction. Each cycle was defined to be whenever the liquid would pass through both the hot and cold side, with our design used essentially having 7 hot and 7 cold zones for the liquid to pass through to achieve PCR.

In terms of equations used, we show that:

- Q = <v> * w * h
    - Q = flow rate (mm^3/s or µL/s)
    - <v> is average velocity (mm/s)
    - W = width of channel (mm)
    - H = height of channel

Plugging in the actual numbers into the calculations, we get the following values for our specifications, along with for the chip design used.

- To calculate the length travelled by the liquid, we simply added the length of the channels designed in Inkscape. Per 1 cycle: 24 + 2.5 + 48 + 2.5 + 24 = 101 millimeters
- Per cycle time: 3.98 minutes, 7 cycles used in design.

- Sample serpentine designed used, with 7 cycles: 7 * 3.98 = 27.86 minutes, or 27 minutes and 51.6 seconds.

The slowest possible flow rate to achieve PCR was 30 microliters per hour, with a recommendation to make designs based off of 100 microliters per hour. However, for the tests, we ran it at 80 microliters per hour with a pump to inject the liquid at such a rate.

The Fluid Velocity was calculated to be 4.23 * 10^{-4} meters per second, given by the equation:

- Fluid Velocity = Flow rate / (W*D) = 80 microliters per hr / [0.35 millimeters * 0.15 millimeters] = 4.23*10^{-4} meters per second

To calculate the per cycle time, we took the overall length of the channels, divided by the fluid velocity.

- Per Cycle Time = Length / Fluid Velocity = 101 millimeters / (4.23*10^{-4} meters per second) = 27.86 minutes

Data

Before beginning our tests, we ran initial trials to check the PWM and voltage control of both the Arduino and MOSFET. This worked to essentially ensure that the power supply would work in junction with our setup to actually heat and cool as needed, given the temperature that is sensed. Both tables depicted below work to validate and show the functionality of the supplied voltage from the power supply, along with the measured values at the MOSFET. They further show the temperature read at different measurements, ensuring that PCR would work and going as far as to validate our dynamic PWM voltage control and allow us to control the peltiers with our Arduino. We then furthered testing to check different temperature measurements for both the hot and cold sides of the acrylic chip through two different peltiers, measuring the change of temperature with respect to time. The tables below depict the voltage and amps with respect to PWM, given the voltage supplied by the power supply. They further depict the temperature given the PWM in one trial, along with the temperatures of both the hot and cold sides given a set time in another trial. The main thing we wanted to validate via the peltier device was simply maintaining a constant temperature, and ensuring that we could vary and control the temperature base of voltage. Reference code relating to the peltier device is provided in the Sources section of this report.

Power Supply: 6.5 V		
PWM	Voltage	Amp
0	0.07	0
20	0.55	0.08
40	1.02	0.16
60	1.48	0.23
80	1.91	0.3
100	2.31	0.37
120	2.7	0.42
140	3.07	0.47
160	3.42	0.52
180	3.75	0.57
200	4.07	0.61
220	4.37	0.64
240	4.66	0.68
255	4.87	0.71

Temperature Readings (Power Supply: 6.90V)			
Temp (C)	PWM	Voltage	Amp
72	150	3.56	0.48

92	255	5.31	0.67

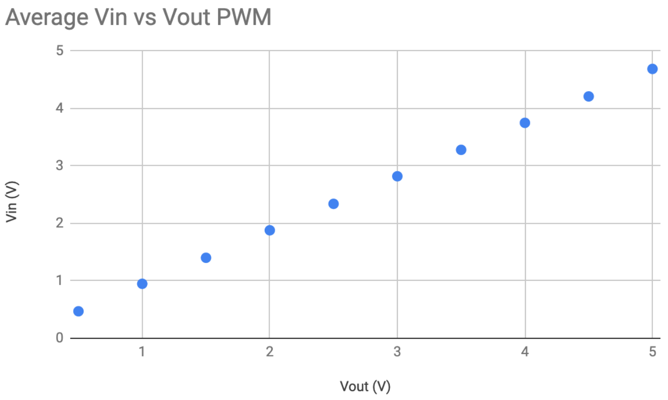
Hot Side						
Peltier with no heat sink, with back plate, temp probe behind 3mm acrylic						
#	Volt	Amps	Baseline Temp	Temperature (C)	Difference (C)	Time(mins)
0	0	0	22.5	22.5	0	0
1	0.5	0.09	22.5	24.75	2.25	5
2	1	0.17	24.75	27.25	2.5	10
3	1.5	0.25	27.25	31.25	4	15
4	2	0.32	31.25	36.25	5	20
5	2.5	0.39	36.25	42.25	6	25
6	3	0.45	42.25	49.75	7.5	30
7	3.5	0.5	49.75	57.25	7.5	35
8	4	0.55	57.25	65.25	8	40
9	4.5	0.59	65.25	73.25	8	45
10	5	0.64	73.25	82	8.75	50

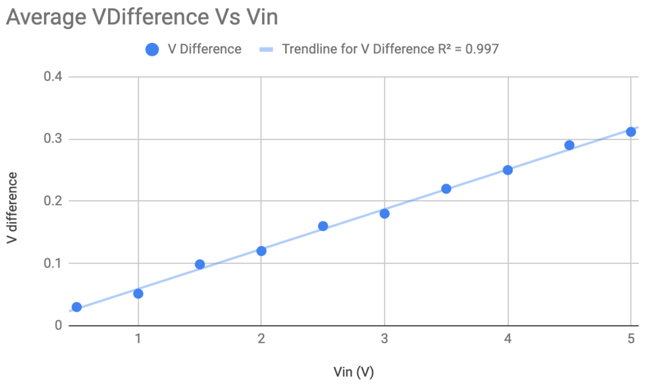
Cold Side						
Peltier with no heat sink, with back plate, temp probe behind 3mm acrylic						
#	Volt	Amps	Baseline Temp	Temperature (C)	Difference (C)	Time (mins)
0	0	0	25	25	0	0
1	0.5	0.09	25	23.25	-2.75	5
2	1	0.17	23.25	22.75	-0.5	10
3	1.5	0.25	22.75	23.5	0.75	15
4	2	0.32	23.5	25.75	2.25	20
5	2.5	0.39	25.75	29	3.25	25
6	3	0.45	29	34	5	30
7	3.5	0.51	34	38.25	4.25	35
8	4	0.55	38.25	43.5	5.25	40
9	4.5	0.59	43.5	49.25	5.75	45
10	5	0.64	49.25	55.25	6	50

To further testing and ensure the validation of everything working, we also compared the voltages coming out and the voltage coming in with the Arduino PWM. We ran seven trials, averaged the data, and created a set to represent the averages experienced during out trials.

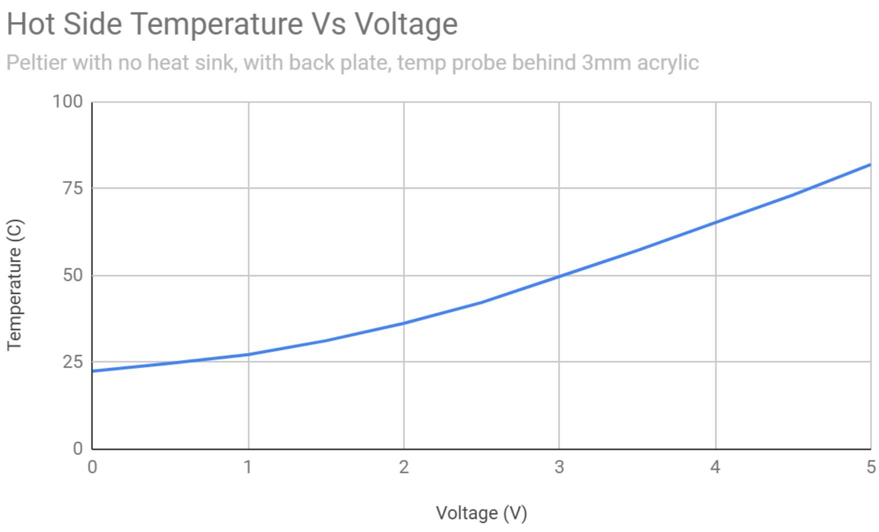
These trials simply help to show the voltage flow in the circuit while using PWM in Arduino. The tables and graphs below help to depict our results.

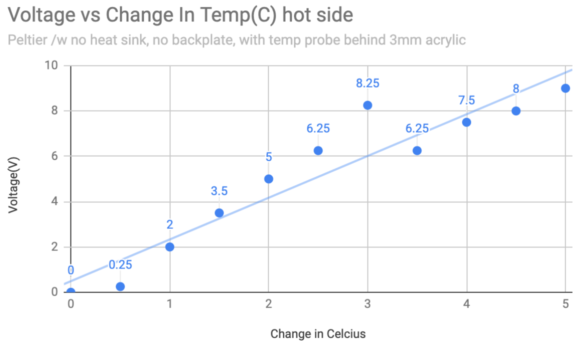
	Vin	Vout	PWM	Percent	V Difference
Avg data	0.5	0.47	25.5	9.9609375	0.03
	1	0.9485714286	51	19.921875	0.05142857143
	1.5	1.401428571	76.5	29.8828125	0.09857142857
	2	1.88	102	39.84375	0.12
	2.5	2.34	127.5	49.8046875	0.16
	3	2.82	153	59.765625	0.18
	3.5	3.28	178	69.53125	0.22
	4	3.75	204	79.6875	0.25
	4.5	4.21	229.5	89.6484375	0.29
	5	4.688571429	255	99.609375	0.3114285714

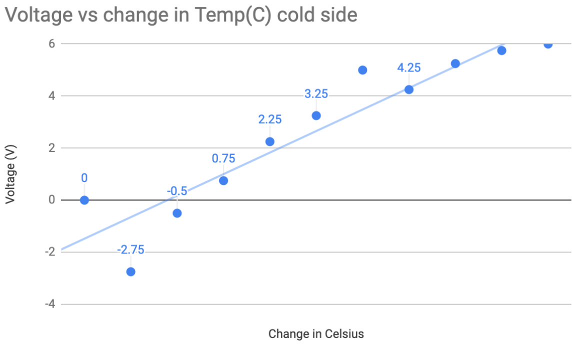




After our initial tests validated needed base conditions to actually work as needed, we began our main tests to observe different temperatures on both the hot and cool side, given different conditions and settings. We ran an overall test to observe the heat with respect to voltage, a dry test, and a wet test with water. The graph shown below is for the test to observe the heat with respect to voltage, simply showing our accuracy and capabilities.



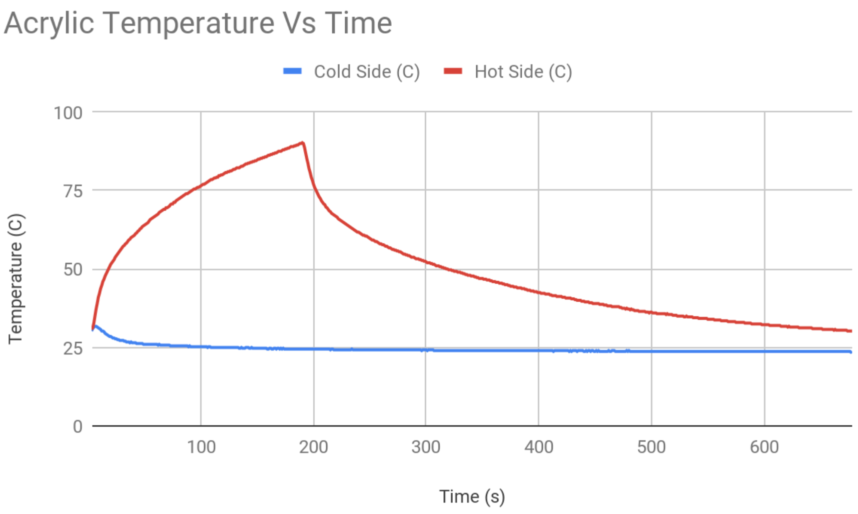


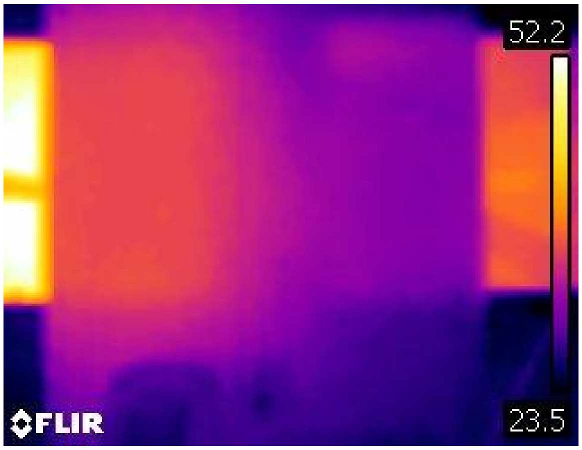


We were able to observe 82 °C at 5V. At 6.9V, the peltier reached 110.5 °C, retaining an accuracy of 1 °C and maintaining a constant temperature.

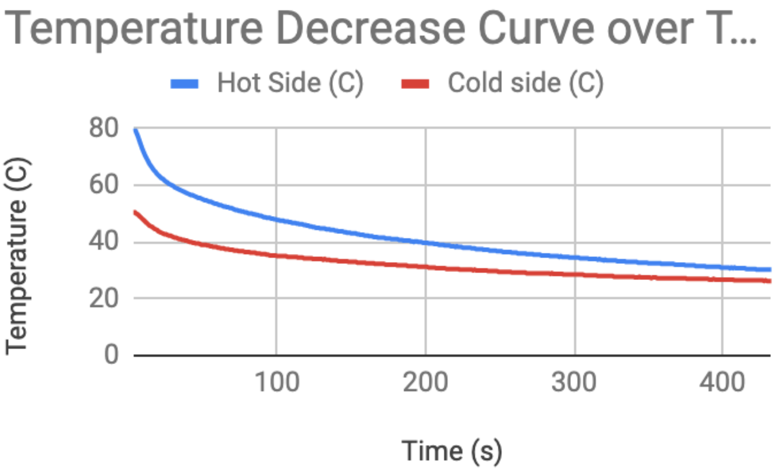
Dry Test

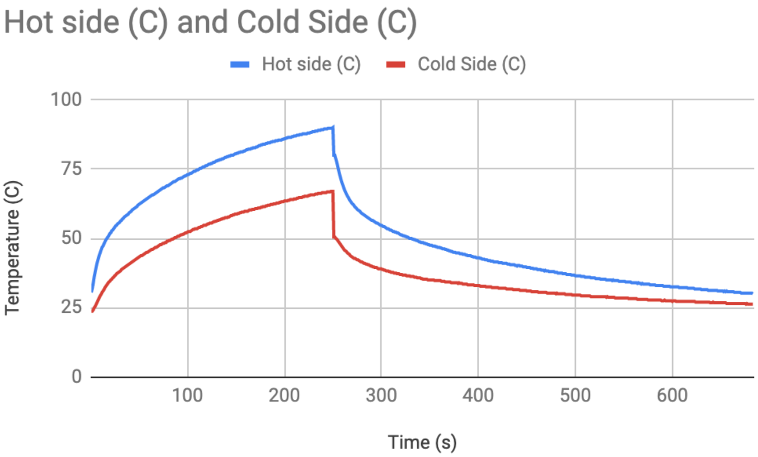
With the graph shown below, we were able to observe a generally quick increase in heat, capped off by a gradual decrease in order to cool the chip back down towards a room temperature environment. The cool side maintained accuracy and stayed very close to its desired temperature. With the thermal image of the dry test below, you are able to see the differences in heat between the hot and cold sides, with the chip maintaining an overall heat of 30 °C. The hot side in the thermal images seemed to be able to get up to about 37.85 °C, although the central, purple area where the liquid would travel through never quite reached that heat. We were able to dynamically use PWM for voltage control to maintain constant temperatures. The thermal image helps to show that the hot and cold side do not affect the other side, essentially staying in their respective zones. The graph and thermal image below depict the data for the dry test.





To gather more data before putting in any liquids, we ran two more dry tests. The first test measured the temperature decrease over time after heating up, and the second measured the overall temperature while both heating up and cooling down. Graphs for both tests are depicted below.

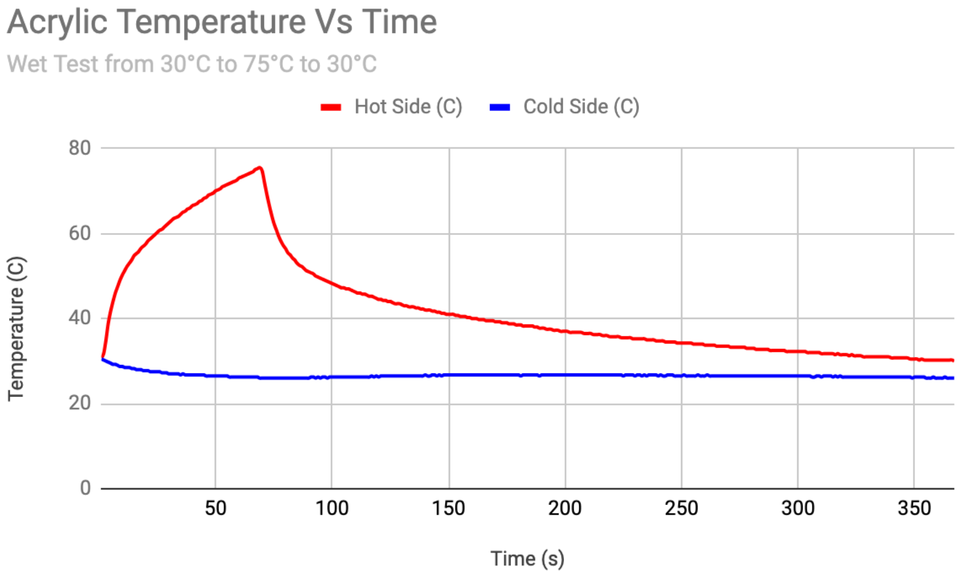


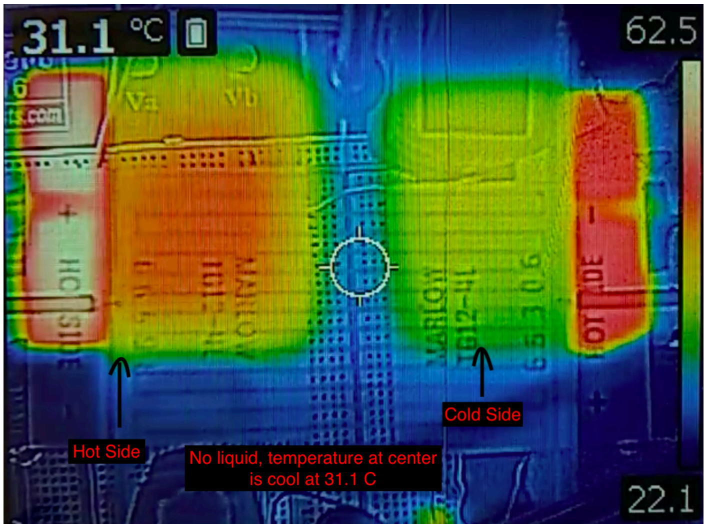


Wet Test

For the wet test, with water, we ran it only to about 80 °C. An issue that we encountered in running the tests was that the plastic that we used to vacuum seal the chip shut had melted due to the heat, blocking the water from travelling through the channels and forcing us to run it at 80 °C instead of 100 °C. However, the data still showed everything that we had expected, just at a lower temperature. Though at a lower temperature, the tests still helped proved the validity and possibility of PCR being performed, simply at a different set of cycling due to a lower heat.

The temperature measured proved to still be constant and accurate, though with a more drastic dropoff when the chip started to cool down. The reason why for the more drastic drop off during cooldown was simply due to the fact that the liquid would carry out the heat through channels as it would enter and leave the chips through the I/O ports. Essentially, the heat that was shown to cross between the hot and cold sides had been carried by the liquid passing through the serpentine channels. As the liquid left, the heat thus followed along through the I/O ports. There is a thermal timelapse to show the liquid test which unfortunately cannot be embedded in a satisfactory format. The thermal timelapse can be seen in the presentation, embedded under Sources. The images below depict the data and thermal images for the wet test. Given the thermal images, it is possible to see the liquid carrying the heat from the hot side over to the cold side and out of the serpentine channels, which is to be expected.

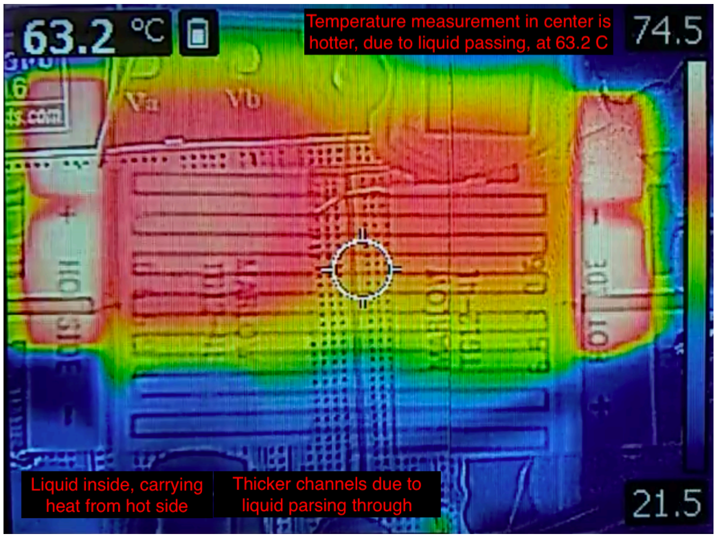




Hot Side

Cold Side

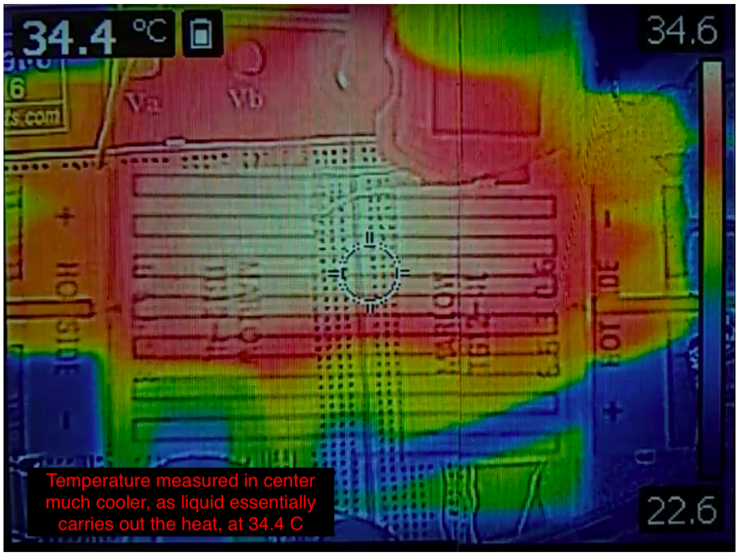
No liquid, temperature at center is cool at 31.1 C



Temperature measurement in center is hotter, due to liquid passing, at 63.2 C

Liquid inside, carrying heat from hot side

Thicker channels due to liquid parsing through



Temperature measured in center much cooler, as liquid essentially carries out the heat, at 34.4 C

Future Works & Conclusion

Something that we were unable to cover during the academic year was the testing and analysis of PBS, Phosphate-buffered-saline, a buffer solution that is water - based. We only ran base tests, with multiple dry and water tests to prove the validity of PCR being able to work at both a microfluidic and cost-efficient level. In the near future, we plan to create a fully complete prototype device capable of performing full PCR. We also plan to create a low cost and easily accessible desktop PCR device. The desktop prototype would enable the ability to have disposable microfluidic devices, allowing the user to be able to simply plug in the cartridge, or the prototype device, and analyze as needed. This would allow for easy accessibility, lower cost, and a wider range of tests with more cartridges at the users' disposal. With a fully functioning desktop device, the user would essentially be able to change settings and have the device run per the users' specific needs. The desktop prototype would essentially create a system to automate different channel geometries on an acrylic chip depending on the given settings, with different reusable and disposable devices. The work done this past year essentially ensures and helps to validify the fact that this project can and will work in the near future, with the next steps being taken towards creating the environments to apply DNA and test in a real-world setting.

Acknowledgements

For allowing the research project to be conducted throughout the school year, we would like to acknowledge a few others and show our gratitude for allowing us to work on this project and for giving us the necessary resources to support us in completing this work. With that being said, we would like to thank Professor Philip Brisk, Dr. Brian Crites, and Heran Bhakta. Without the supervision, collaboration, funding, and equipment provided, we would not have been able to come thus far in this project, and would not be able to further the project to complete the planned future work. We would like to further our acknowledgements for the Undergraduate Mini-Grant Committee here at the University of California, Riverside, for providing the necessary funding towards accomplishing the work that has been put in thus far.

Sources & References

1. Research Presentation
2. Thermal Cycle Graph
3. Reference PCR Microfluidics Information
4. Reference Report on Dynamic Temperature control of Peltier
5. Reference Code on Peltier Device