Notebook Contents

Interaction
- Research and Planning
- Design
- Modelling
- Prototyping

July

Week 1: Research and Planning

Week 2
- Further material testing
  - Material combinations qualitative testing
  - Quantitative analysis of geometries

Week 3
- July 10th Slanted Chip trial
  - Validity of the Tesla Valves

Week 4 : July 21st to July 28th
- Persisting Challenges
- New production protocol
  - Laser Cut optimization
- Fabrication breakthroughs
  - Assembly
- New Geometry layout
  - Meander Design Challenge
  - Johannes Kepler University Collaboration
    - Meander Chip iterations

September

Week 1-Week 3
- Actuation
  - Pushing bending/inclinations principles to the limit
- Air pumps
  - Air pump radius variance test
- Multiplexing and channel shape
- Material Shortcomings

October

Final design and renderings

References

Actuation

Thermocycler

July 7th: Research

July 8th-July 14th Ideation and Discussion

July 15th-July 16th Temperature Sensing
Sensor Selection

July 17th: Peltier Controversy
Peltier Alternatives

July 18th: Alternative non-electric heating

References

Appendix
Code for the Sensor and heater Circuit

Final Thermocycler

Fluorimetry

Interaction

Research and Planning

Our starting points for the design of Volatect were Dieter Rams' principles of good design. These principles are applicable to all modern design, though we believe some of them, such as comprehensibility and honesty, are particularly relevant to Volatect's mission.

Principles of Good Design:

1. is innovative
2. makes a product useful
3. is aesthetic
4. makes a product understandable
5. is unobtrusive
6. is honest
7. is long-lasting
8. is thorough down to the last detail

9. is environmentally friendly
10. has as little design as possible

The interaction between the device and the user was idealized as a straightforward process, one in which the user is informed of the proper use and function of the device. This type of user-device interaction allowed us to avoid elaborate safeguards or workarounds of possible user error.

The design was partly inspired by common devices in biological sample processing, such as centrifuges and RNA bio analyzers.

Making Volatect innovative proved to be a challenge since it involved defining what an innovative appearance would look like while aligning to current design trends in biomedical devices. On the other hand, many modern devices are designed to prominently display their advanced technology and future-oriented vision. We believe we achieved good design by creating a device using cutting-edge technology while still remaining connected to devices people are used to seeing in biomedical procedures. Therefore, the design of the device was made to be a conceptually simple aesthetic, something that could be quickly identified but inconspicuous. Our research on similar biological analysis devices show a clear preference for a friendly prominent forward-facing curve that would attract the attention of the user directly towards the main interface, a design that is perhaps best exemplified by common centrifuges. Simultaneously, we aimed to design a device that would also be environmentally conscious, minimizing waste and maximizing effective lifetime.

Figure 1: Examples of devices used in our characterization of important aesthetic features in already existing biological devices.

Design

Volatect ensures in its design that any function it has is directly involved in reaching its main objective of fast and reliable diagnosis. Volatect is stripped down to its most bare necessities; there are no extraneous features that distract from the final result, maximizing the straightest path towards properly providing our advertised function. In essence, the device

design prioritizes just two elements: the status LEDs and the chip slots. To achieve that, the chip slots are prominently displayed forwards, without any covering. As the device is designed to be in a sparsely occupied and clean room we decided that the clarity and ease of accessibility provided by the open slot superseded the reasons for applying any slot cover.

The expected use of Volatect is delineated by having the device on a table while the user is either standing or sitting next to the device. Due to the movement required in handling a sample, applying it to the chip, and inserting the chip we decided to have a symmetrical device usable from any direction and the LEDs pointing in a direction that could be easily seen regardless of the exact angle the user's perspective would be at. The only other significant design feature is the logo placed squarely on top, reinforcing the quick identification of Volatect we aim for.

This simple design allows for the user to clearly identify all the parts of Volatect while simultaneously understanding what they are there for. With the outer design of Volatect meeting our goals, there was also the inner design. In this area we focused primarily in using as little material as necessary while maximizing the utility life of each component. The chip system allows for the majority of the device to be unlikely to suffer damage or tear and wear, and highly increases the estimated lifespan of the device. While the chip itself cannot be recycled due to biological hazards, using Volatect means being able to analyze a sample using only a fraction of the resources a lab would use in conventional analysis, which cause a reduction in resource consumption.

With this design in mind, we moved on to the next step: bringing the product to life.

Modelling

As we were going through the process of design ideation, several prototypes, either in model or in physical shape were made. A first draft was made in Blender to have an initial idea of what our device could look like.

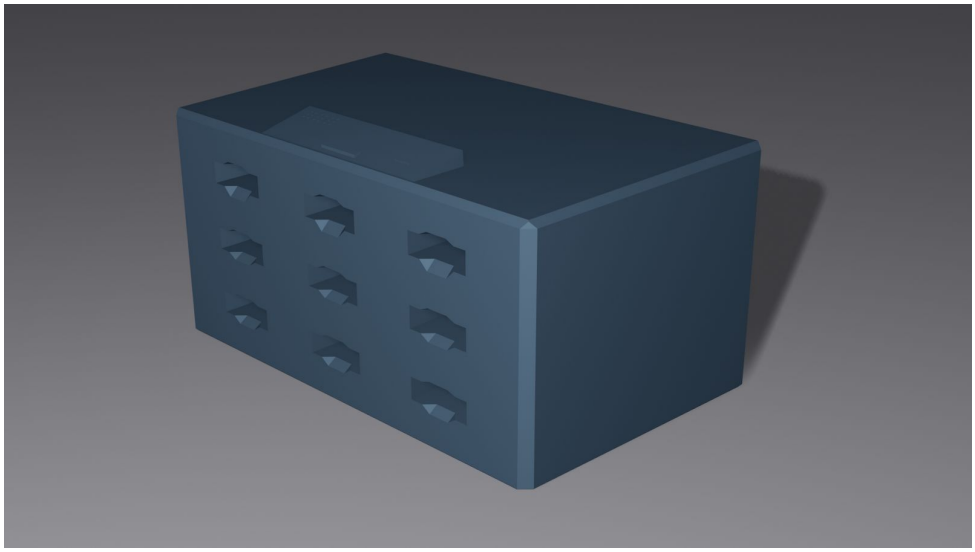


Figure 2: Blender render of early draft.

Looking at the draft, it became apparent that having 9 chip slots for our testing purposes was rather unnecessary, so we reduced the number of slots from 9 to 3. Moreover, the interface was deemed too visually unappealing and we began to look for alternatives. Going through the visual aesthetics of pre-existing devices provided an answer. As previously mentioned, the forward-facing curve provided a friendlier look as well as an optimal location to place the LEDs. With that in mind, we moved on to Solidworks to create a more refined model. It was at this time that we decided that placing the product logo on top would align with our branding ideals and provide another feature to ensure clear recognition of the device.

Figure 3: Later Solidworks model containing more detailed features

Prototyping

At this point we decided it was a good idea to start prototyping physical models of our current design; that way we would be able to do troubleshooting in the small details that, more often than not, can only be noticed when physically handling an object. Our first models using foam helped us visualize the specific measurements to have a device that would be too small for the internal compartment to fit.

Once we had a few foam models we decided to use paper mache and plaster to create a hard exterior that could be sanded and modeled to match our design. The foam would be melted away using acetone. We began by covering the foam with tape to avoid the acetone breaking away the paper later. We quickly learned the possible dangers in using those materials. If the foam was melted too early in the process the plaster would weigh the paper down to the point of deforming the model. The last step of plastering the model was what caused the eventual cancellation of that prototyping line. Plaster proved to be too easy to deform and create extra problems, as removing plaster deformities meant continuous sanding, which itself could cause plaster pellets to pry off, taking part of the underlying paper layer with them. In the end we decided to go back to the Solidworks model and 3D print our parts using PLA. 3D printing allowed for a fast turnover rate of improvement, high fidelity, and easy modifications, including designing the latches that would be used to screw the entire enclosure together. Any printing imperfections were sanded over.

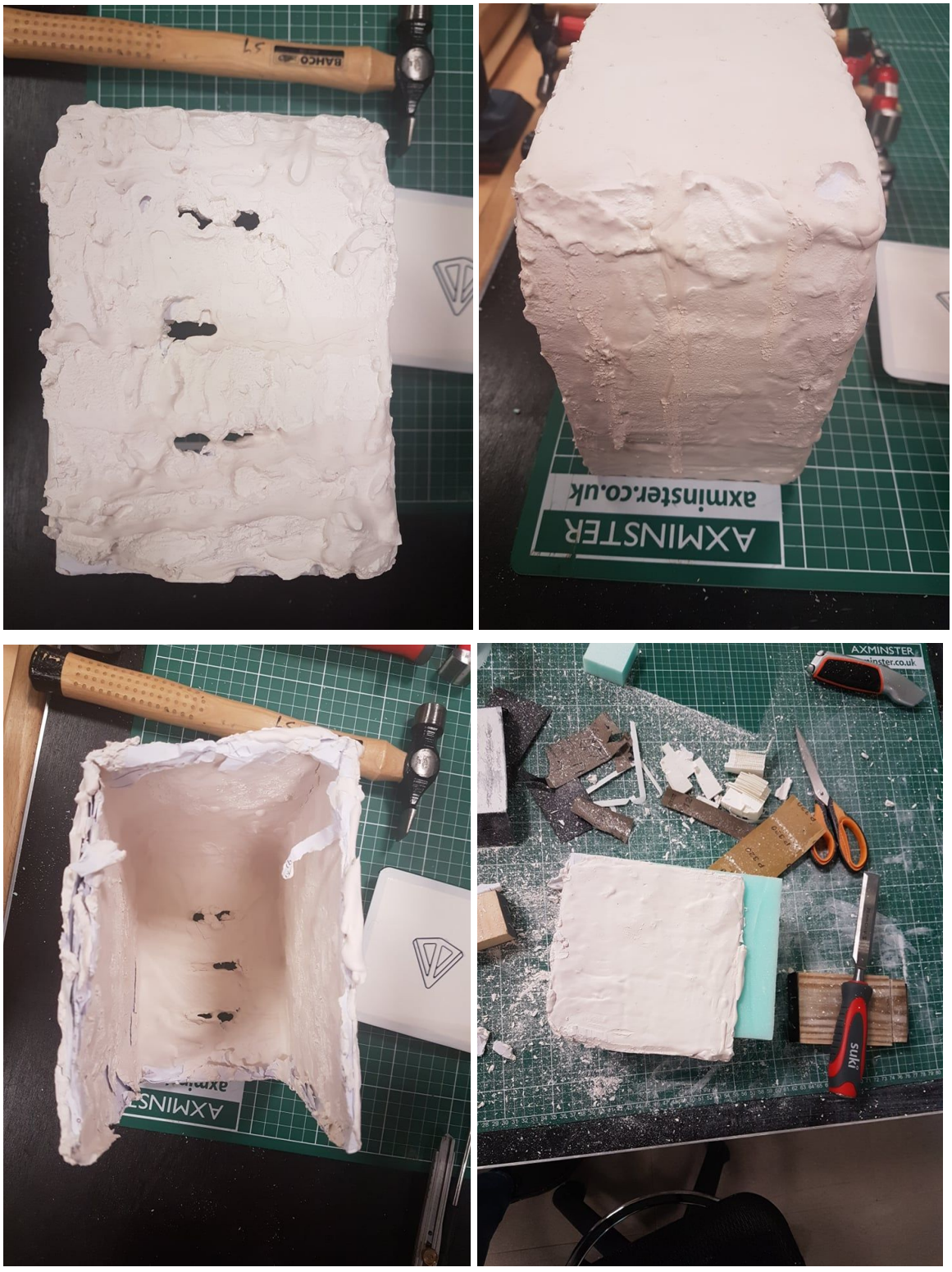


Figure 5,6,7,8:Pictures denoting the problems in using plaster: Deformation of the plaster (top left) and the paper skeleton (bottom left), paper tearout (top right), and excessive sanding (bottom right)

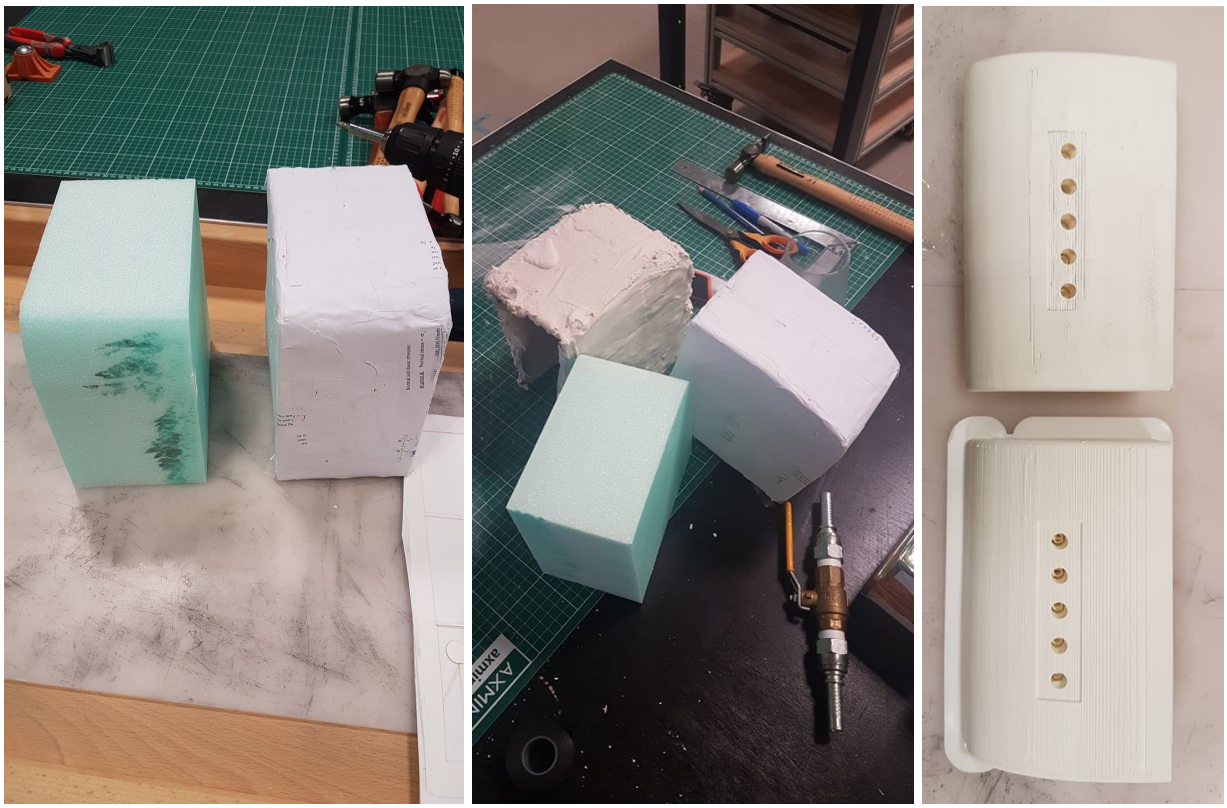


Figure 9,10,11: As part of the process we experimented with a series of materials, but 3D printing and sanding PLA ended up being the best solution.

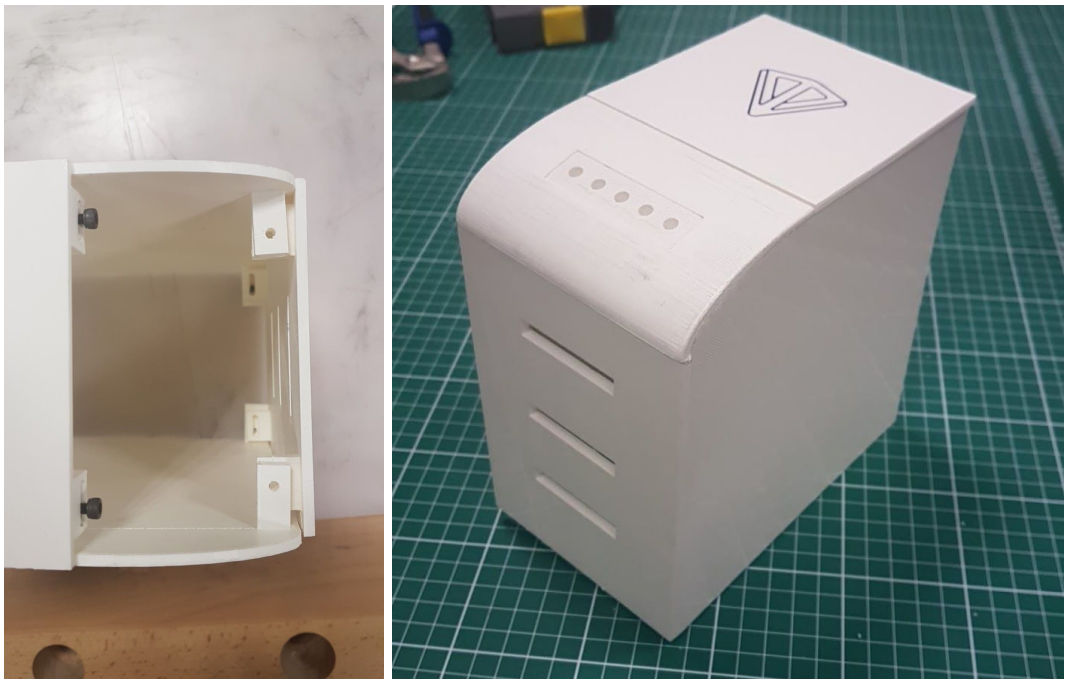


Figure 12,13:3D printing allowed us to iterate quickly and easily, like adding space for screws to connect parts together instead of time intensive gluing or slotting.

Sample Collector

- Considerations based on market research etcccc

July

Week 1: Research and Planning

After preliminary research

The engineering team acknowledged the importance of understanding all past projects before delving into the ideation process. The team also intended on building on previous projects by NYUAD iGEM and the whole iGEM community. As such, the team set out to read all the documentation and replicated the milestone of their design process to understand the shortcomings of the product in terms of materials chosen and overall hindering effects on the final resolution of the device. The NYUAD iGEM 2018 team was used to replicate the NYUAD iGEM 2018's which later served as an ideal starting point.

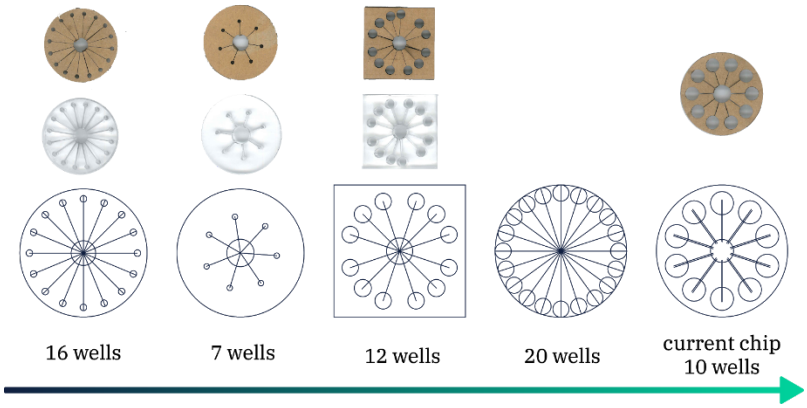


Figure 1: NYUAD iGEM 2018 (Pathogenie) Microfluidic chip design timeline

The entire team then delved into the ideation phase by looking into current problems and needs of synthetic biology and how engineering parts can be used to meet them. The team looked into relevant relevant literature and published papers on the matter. Our Principal investigators guided us through an extensive literature review. Papers on increasing the resolution of the signal through electrokinetic concentration of DNA in a microfluidic chip exposed us to the wide variety of solutions that were already available for us to explore [1].

We were also provided with laser-cut safe materials to test for the new microfluidic chip: 3M 9960 Diagnostic Microfluidic Hydrophilic Film that had not been previously used, thus consisting in a potential innovation since the early stages of the design process.

The brainstorming sessions and the literature review of the research and development of Point-of-Care diagnostics devices [3] [4] demonstrated the versatility of using hydrophilic film. The choice of film and double-sided tape and potential components as opposed to the traditional use of acrylic or polydimethylsiloxanethe [5] was also a direct result of suggestions by our main PI Prof. Song who had distinguished the potential of 3M films in his research work earlier in 2018 [1]. 3M was the company of choice due to their availability in the region, reasonable prices and delivery time and extensive documentation provided by the company to push for certain products to be used in ways similar to ours [6].

We started by doing initial trails on the two materials to obtain a better understanding of their fluid flow behaviours. We tested the vector files of the old team and reprinted the microfluidic chip ensuring that the technical steps of manufacturing the chip were followed. After observing the chip and it's efficiency, we built a new chip with a simple geometry rather than the radially symmetrically distributed well. For gathering pure fluidic flow empirical data, TE Buffer with food coloring to enhance contrast was found to be an ideal to simulate the reagents that would be used eventually in the completed microfluidic chip.

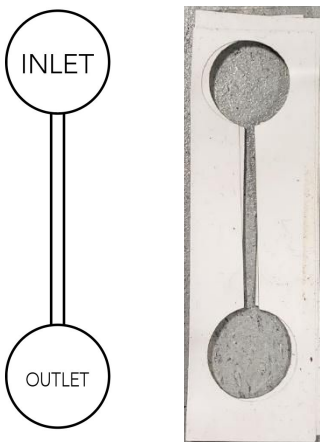


Figure 2: Vector File and Cut version of tape

Week 2

Further material testing

The materials were made from polyester and hence rendered safe for use in a laser cutter. A Roland Epilog Legend 36EXT was used to laser cut and later engrave the materials during the material testing period. A simple geometry was printed to also test the power, frequency and speed - variables entered on the laser cutter - for a satisfactory print on a laser cutter. At first, power of 80%, speed of 75% and frequency of 4700 Hz were used. This value were obtained from iGEM NYUAD 2018's team optimized settings.

It was noticed that when the new double-sided tape (henceforth tape) — which was used as a means of keeping the two layers of hydrophilic film in place and trap the fluid within the confines of the channels—would burn off from the laser printer. A temporary solution was peeling the top layer which lead to a significant and inevitable loss in adhesion. The burn was not toxic, nor did it have a smell, but the burning would form a significant burr that interferes with the dimensions of the cuts - making it even more of an issue considering the channels size are in the range of 500 microns.

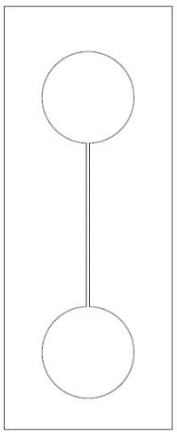


Figure 2: Vector File testing

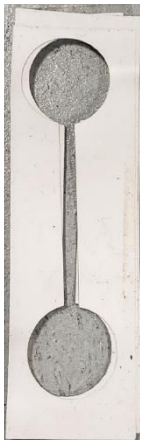


Figure 3: Material

To ensure we tested a variety of options before settling on one, the vector file in figure above was cut on a Poly (methyl methacrylate) (PMMA) sheet of 3 mm thickness as well as double sided-tape. The hydrophobicity of the materials was verified using the principle of contact angle (Figure 4) and wetting of a single pipetted water droplet on the dry acrylic (Figure 5)

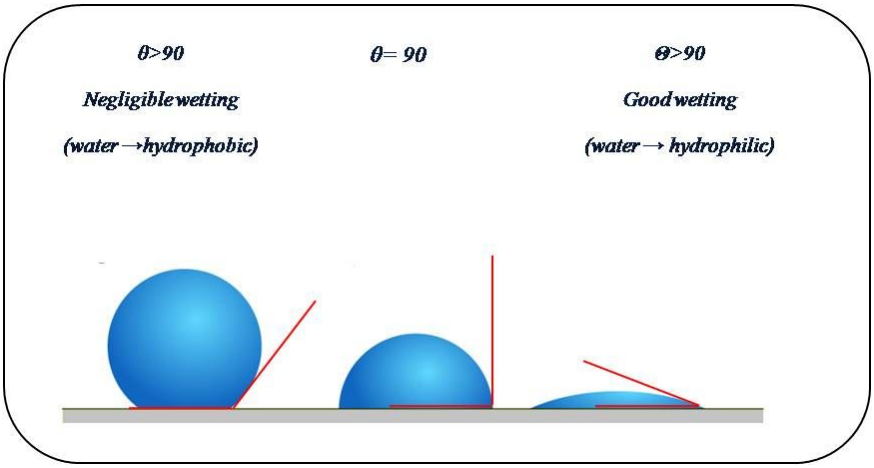


Figure 4 Theory behind the investigation

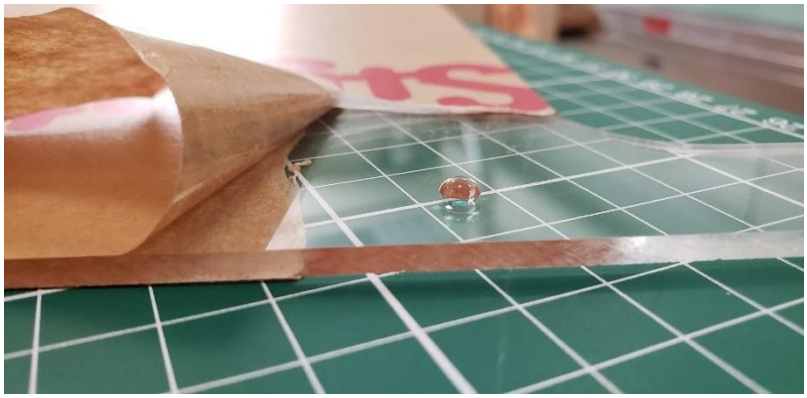


Figure 5 Contact angle observation and match with the theoretical model above

Using the production protocol of the previous day, the vector file was printed on acrylic and super-glued to the hydrophilic film, a procedure that proved to be unreliable as the film would slip and contaminate the 500 μm channel with the liquid super-glue. Figure 6 shows the successful flow test.

Figure 6: Testing acrylic flow on a hydrophilic floor

Flow in the acrylic spacer was comparable to the tape-film combination of the previous day. Based on the principles of chromatography, it was hypothesized that the presence of hydrophobic materials cannot be entirely harmful in attaining an efficiently fast flow.

Material combinations qualitative testing

Since the number of combinations increased, it was decided that the following tests would be made with the combinations listed in Table 1.

Table 1. Table showing the combinations of the available materials

	3M Tape 2x sided tape	3M Hydrophilic film	Acrylic
3M 2x sided Tape	-	4a & 4b	1
3M Hydrophilic film	4a & 4b		2
Acrylic	3	2	-

Figure four below illustrates the chip material combinations that were tried.

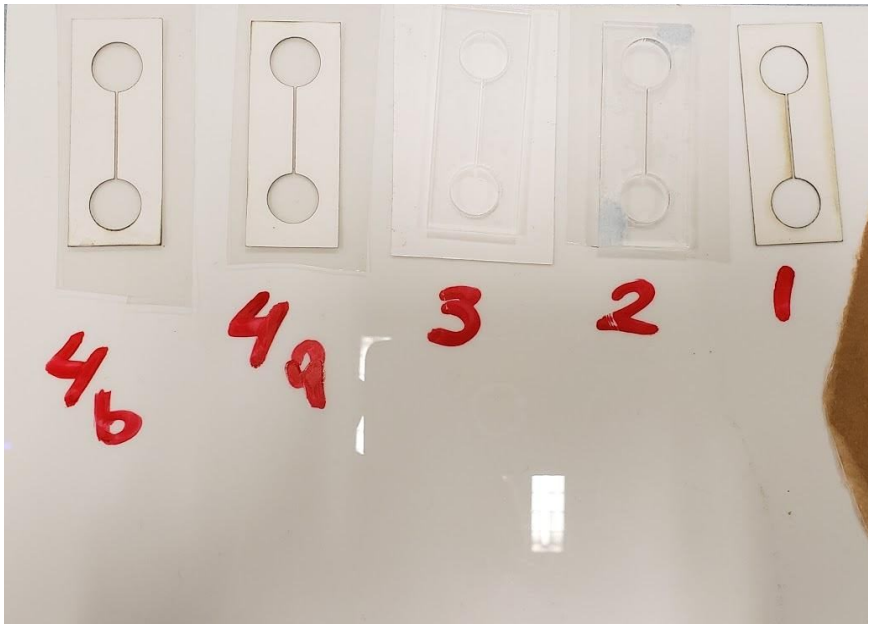


Figure 7 The combinations that were tested.

The test was meant to be qualitative at first but after noticing incoherent results, it was concluded that a more rigorous testing had to be made where the volume of each chip was calculated (featured on the next section). Furthermore, the inlets were redesigned to have smaller radiuses in order to better simulate the real conditions.

The experiment however, helped understand that the acrylic-tape combination (Chip 3) was the slowest regardless of the volume inserted. Thr acrylic-tape combination was dropped, radius scaled down and it was unanimously decided that the acrylic would be properly held to the hydrophilic film only via the double-sided tape.

Another result was that the flat acrylic floor resulted too hydrophobic for a good flow. It was also self-evident that the only way for the acrylic combination to have a decent flow that followed the constraints dictated by the feedback we got as part of our Integrated Human Practices was that either slanting the acrylic chip or adding hydrophilic film.

Quantitative analysis of geometries

Three main material combinations were made in order to make quantitative assessments and calculate the volume. The thickness measurement of each unit of material was obtained from the 3M's official website [6]. Figure 8 shows a brief schematic of how material is stacked. Three combinations were made as a result of the

lessons learned from the previous section's qualitative observations (refer to qualitative testing section).

Combination	Well Radius (mm)	Canal Width (mm)	Height (mm)	Canal Length (mm)	Volume (mm^3)	New Central inlet volume (mm^3)
Tape + film	2	0.5	0.864	35	36.835	45.964
Combined layers	2	0.5	1.0376	35	44.236	55.199
Acrylic+film	2	0.5	3	35	127.899	159.597

Volume formula for each well of Figure 2 geometry:

volume of two cylindrical wells and a cuboidal channel

$$V = l \cdot h \cdot w + 2\pi r^2 h$$

Where:

$h \triangleq$ *number of layers x thickness of a layer from* 3M.com

h is the height of the chip without taking the floor under consideration

$w \triangleq$ *is the channel width,* $0.5\ mm$, *same for all*

$l \triangleq$ *the channel length, which was arbitrarily decided and it is the same for all the chips*

$r \triangleq$ *is the radius of the well*

The new volume was calculated by subtracting the 4m radius of the circle centered at the center of the channel and adding the volume of the central cylinder as appropriate.

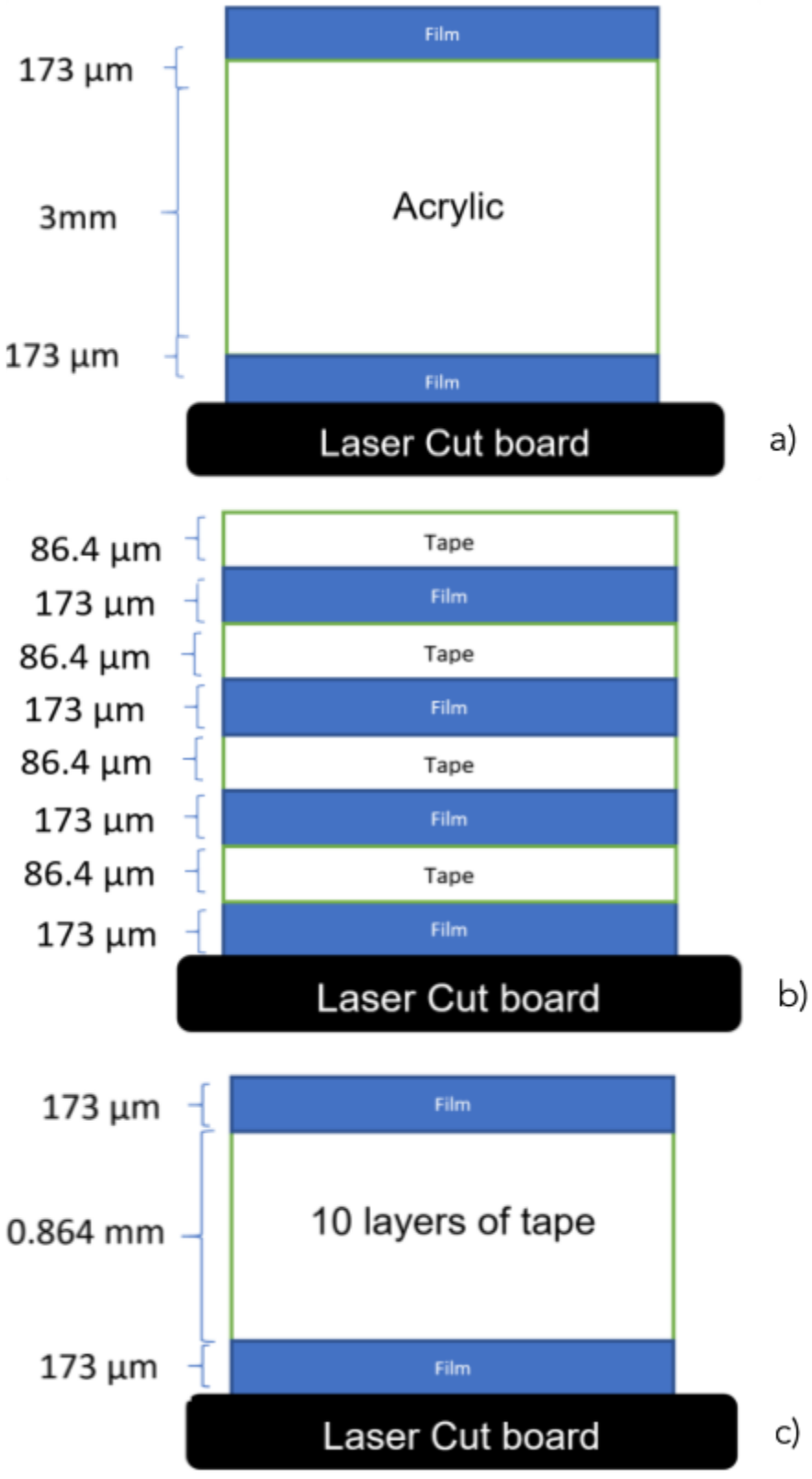


Figure 8. Material stacking

Given the final chip would require meltiplexing of the inlet, a central inlet geometry was also tested as seen in Figure 9. Although the final product did not follow the radial distribution, it was only reasonable that that principle was tested and properly tried.

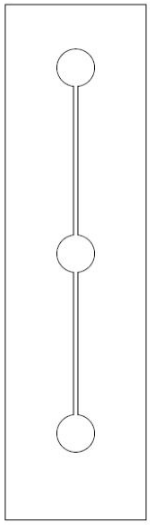


Figure 9: Middle inlet geometry

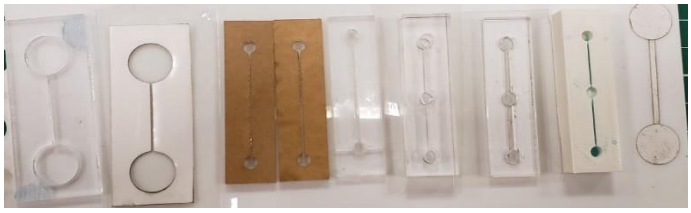


Figure 10: The printed samples by comparison

The observations were made by pipetting the smallest volume (which belonged to the tape + film geometry) to three of the material setups. As measured and seen in the video, acrylic had a significantly better flow than the other materials when the floor was hydrophilic.

Week 3

July 10th Slanted Chip trial

Volumes were recalculated and the idea of a slanted chip was discussed in further details.

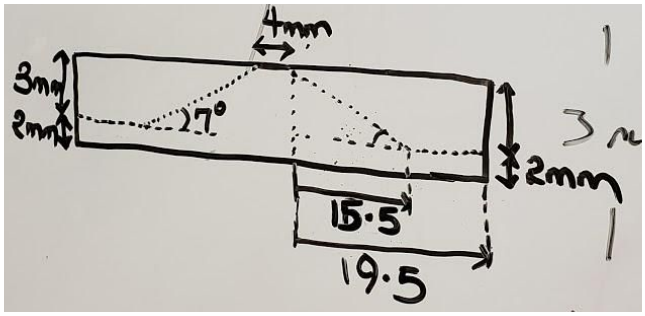


Figure 11 . The design discussion regarding the slanting angle. 7 degrees was thought to be enough for the qualitative test

Another point of discussion was the shape of the outlets and the avoiding of the eddie currents so that there is no backflow or cross-contamination (Figure 12). A Computational Fluid Dynamics simulation showed that there was no significant difference in the shape of the outlets even though the less circular the rims, the less eddies would form.

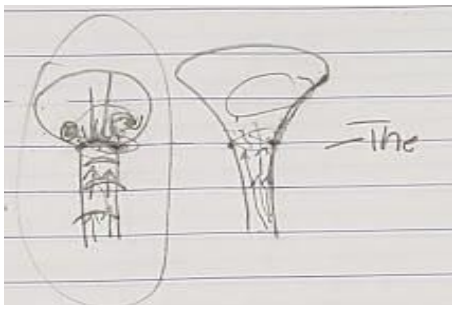
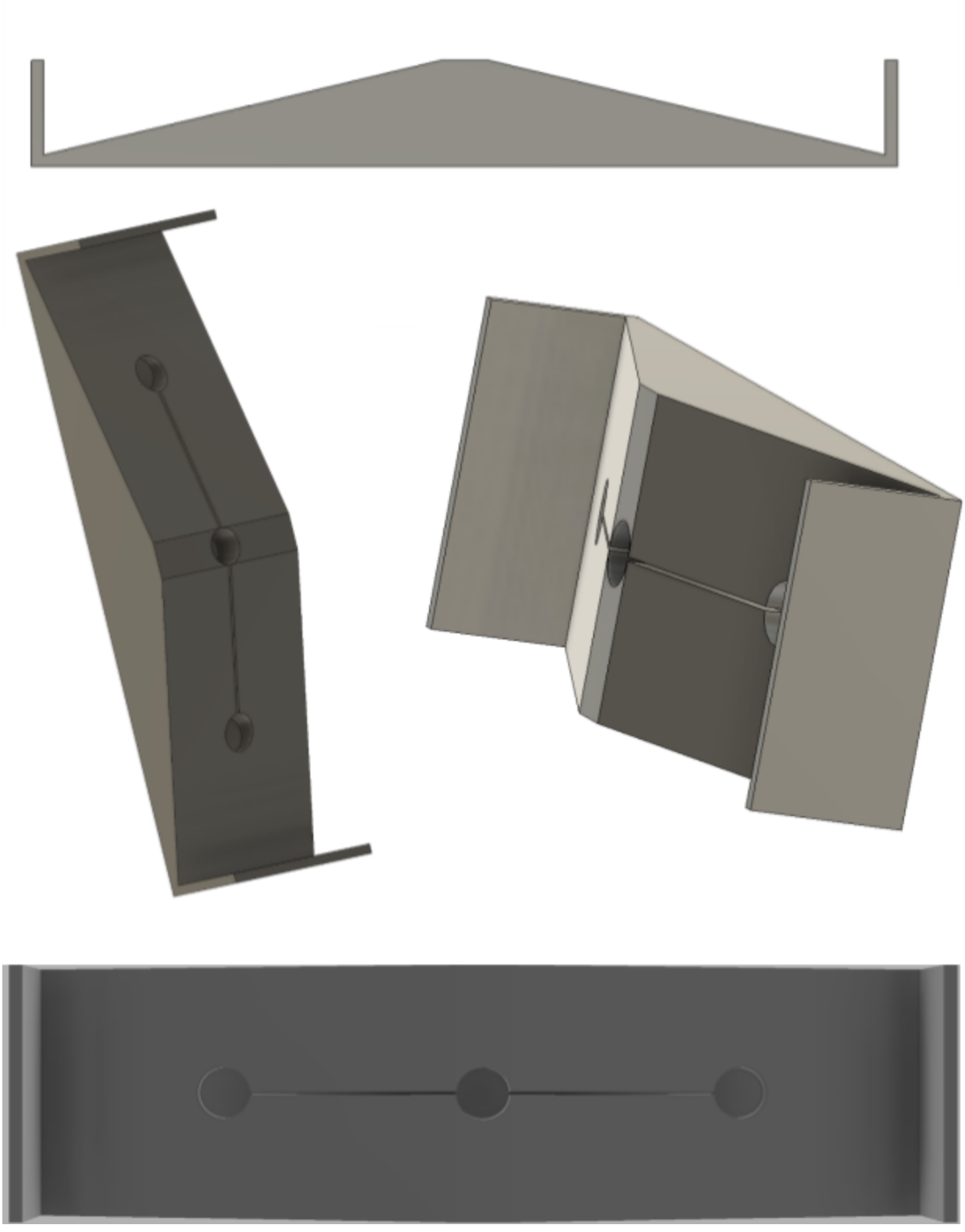


Figure 12. Eddie Current Discussion[1]

[1] Trivial to the final design but featured for the sake of documentation and discussion weight it had

Figure 13 shows the first prototype of the slanted chip as rendered in Fusion 360. The chip was later manufactured on Roland Modela MDX-540 milling machine with a 0.5 mm drill bit.



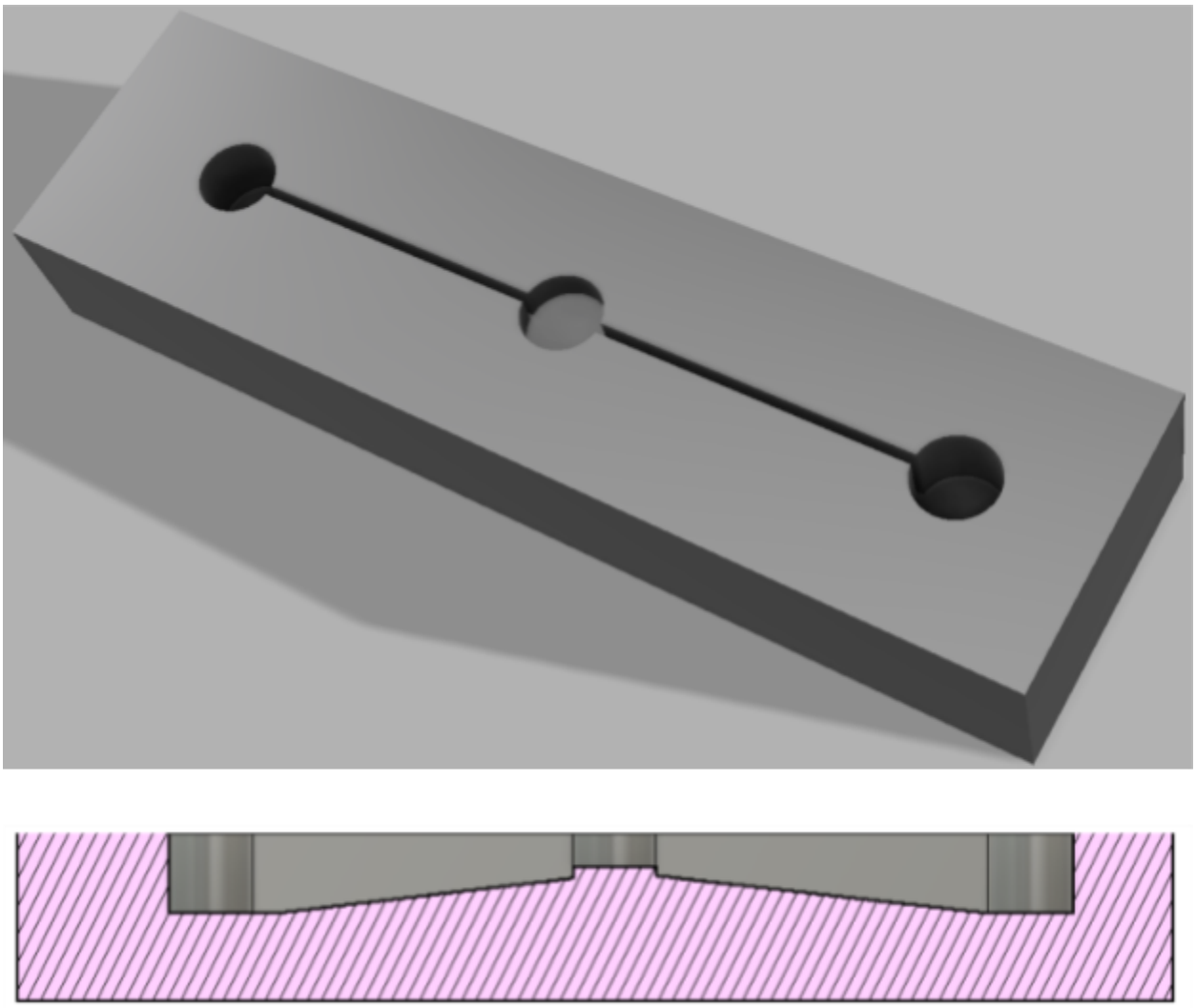


*Figure SEQ Figure * ARABIC 14.View of the final slanted chip that was milled. Top is side view and bottom is the cross section that shows the slant.*

The chip proved to be successful after being cleaned up after milling (washed or air-cleaned). However, the printing with the 0.5 mm drill was too big given the bit would eat away material while milling. It also proved to be a slow process given acrylic melts easily when exposed to high temperatures, and has to be milled at slow speeds. Based on the outcome of this trial, the slanted alternative was eliminated as it left fewer options for actuation and posed optic gradient problems for fluorescence test planned by the biology team.

Validity of the Tesla Valves

While communicating with the biology team, It was made clear to the engineering team that there would be a need to have liquid from reagents while keeping in mind that contamination ***was a significant area that required improvement.*** With the slanted chip being discarded as a solution to the present setup of biology protocols, it was upon us to come up with a geometry that would prevent backflow and premature mixing of the reagents (specifically RPA with Sample and CRISPR before the amplification at 37 °C happened.

Research on multiplexed microfluidics led the team to a video released by The Thought Emporium (https://www.youtube.com/watch?v=eNBg_1GPuH0) where the uploader had used a Tesla Valve to prevent backflow by increasing the required pressure to take the liquid in the opposite direction. Refer to Figure 15 for a 3D Printed model.

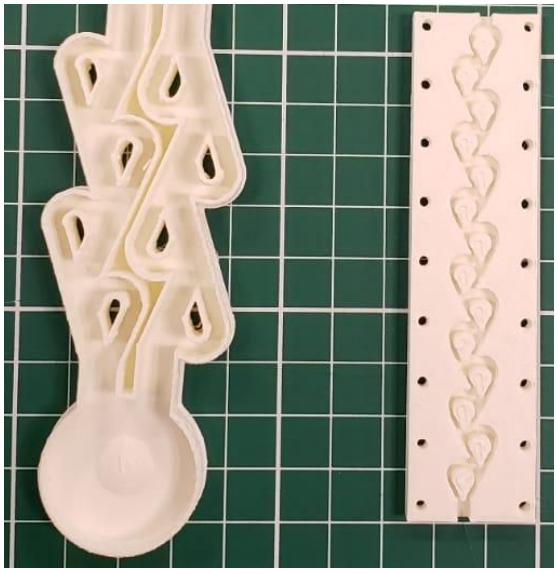


Figure 15. STL files from Thingiverse.com used for modelling, under MIT License

The main purpose of the valve is to control/ block the flow in one direction. A geometric valve does this without any moving mechanical parts. In the right-hand model from the figure below, water can flow down easily and is supposed to take longer to flow in the other direction (when horizontal). This however, did not prove to be as effective when tested with the usual TE buffer+food coloring mix that was used for testing. The acrylic lower scale models proved to be inefficient as well and did not meet the expectations regarding the prevention of cross-contamination.

Having these in mind, the team revisited the hydrophilic film. Accuracy and the right materials are paramount as one designs and manufactures microfluidic technologies. Armed with the knowledge of the precision, quality and ease of manufacturing that would come with using the hydrophilic team, the team understood a synthetic biological microfluidic part would be of great use to the field.

To accommodate the volumes provided by the biology team a meander microfluidic chip was constructed.

Week 4 : July 21st to July 28th

Persisting Challenges

Having created an idea of what materials not to use, the team moved on to solving challenges like the temperature controlling circuit and actuation that ways to control the flow of microliter scale amounts of liquid into the chip.

The challenges for the chip at this points were very clear:

(1) The movement of small amounts of liquid at a given time so that CRISPR acted after amplification
(2) harnessing the potential of 3M's hydrophilic film
(3) prevention of contamination and air bubbles

The 3M products that were available to us were 3M™ 9960 Diagnostic Microfluidic Hydrophilic Film and 3M™ Microfluidic Diagnostic Tape 9965. All the models that were entirely made of the 3M combinations exhibited outstanding flow performance but presented unstable and imprecise results in manufacturing. Some geometries would burn out or the removal of the adhesive cover compromised the insulation of the chip and the food coloring used for testing would flow outside the designated reaction area and cause hazardous contamination.

New production protocol

In order to use the 3M film to the full potential, with the suggestion of Prof.Song, the team set out to optimize the laser cut parameters for the products mentioned above. Now, as stated in the first section of Volatect's microfluidics documentation, the willingness to improve last year's

product led to the team taking the parameters for the tape and PDMS material for granted. The laser cut parameters were the obvious bottleneck as they were the only things that had been left unchanged along the entire first weeks of materials testing.

Laser Cut optimization

As the figure 16 shows, meticulous tests were planned. Figure 17 shows the progression of burr reducing. It can bee seen from left to right in Figure 18 that the burn mark gets less and less dark as a result of proper optimization.

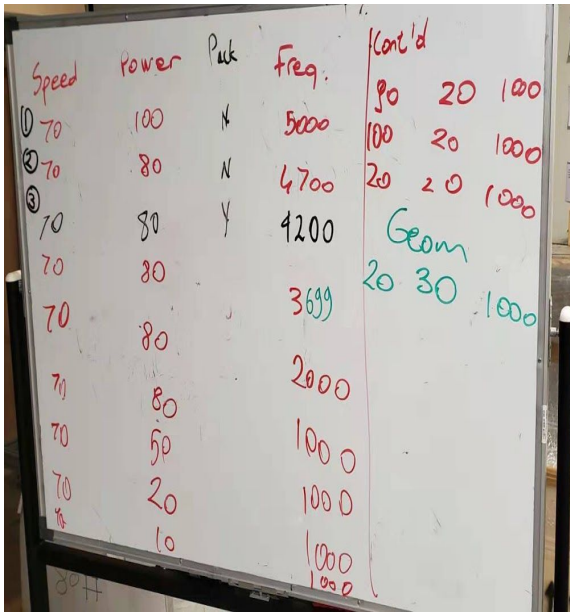


Figure 16. Optimization planning, parameter testing. Increments were made numerically through small increments so that the best solution was attained.

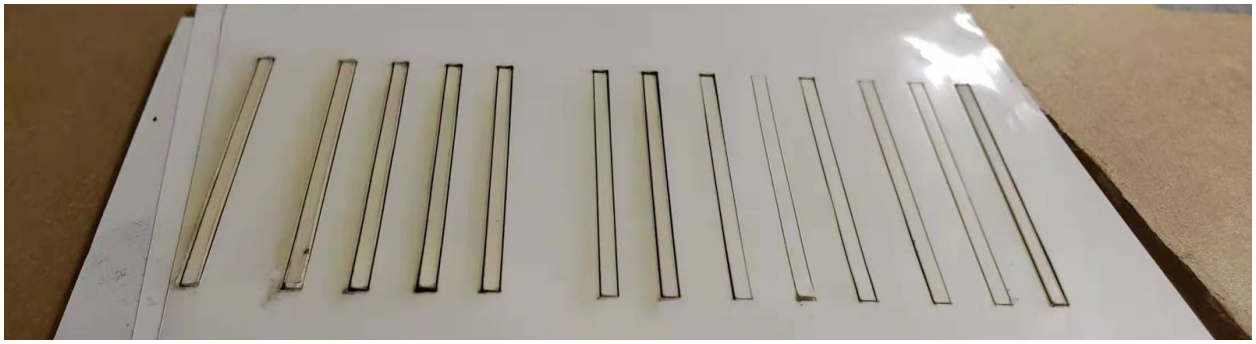


Figure 17. The reduction of burr from left to right hand side

Table 2. Tested values

Speed percent	Power percent	Frequency in Hz
70	100	5000
70	80	4700
70	80	4200
70	80	3699
70	80	2000
70	80	1000
70	50	1000
70	20	1000
70	10	1000
90	20	1000
100	20	1000
20	20	1000
100	30	1000

Table 2 features the values tested and the last row shows the final combination for the geometry.

-

Fabrication breakthroughs

The observed loss of adhesion and the burning of the double tape during laser cutting was a major obstacle when trying the double-sided tape and hydrophilic film combination. Residues from cutting would get in the channel space and corrupt the integrity of the flow, thus getting in the way of the reaction sequence. The optimization paved the way for the assembly of better flow channels.

Assembly

With the successful optimization, the team gained valuable experience on how to improve the production on any layout on any type of microfluidics material. Most importantly, the optimization was so extensive and detailed that the laser now could only cut through four layers of 3M diagnostic tape, but not through the bottom cover layer that was taped to the acrylic board for cutting.

4 layers of double sided tape were stacked and the optimized laser would cut just right before the other protective player of the double sided tape began (orange line in figure 18).

The 3M tape comes with two layers of plastic that protect the adhesive properties. The laser would cut until the bottom plastic layer of the fourth 3M tape. This would make possible the removal of the laser cut geometry with tweezers and its placement on the hydrophilic film. The top plastic layer would be removed for the 3M tape to be sandwiched in between the 2 layers of hydrophilic film. The adhesion would be maximal as the air exposure would be minimized. (Before the optimization, the tape had to be laser cut with the top plastic payer removed which would hinder the adhesive properties but would minimize burn.

Since the laser was optimized and the burns were not as visibly significant, the team decided to keep the top plastic layer (orange line on top in Figure 18) until the final film assembly, thus improving the insulation for the fluid flow and preventing possible backflow and contamination.

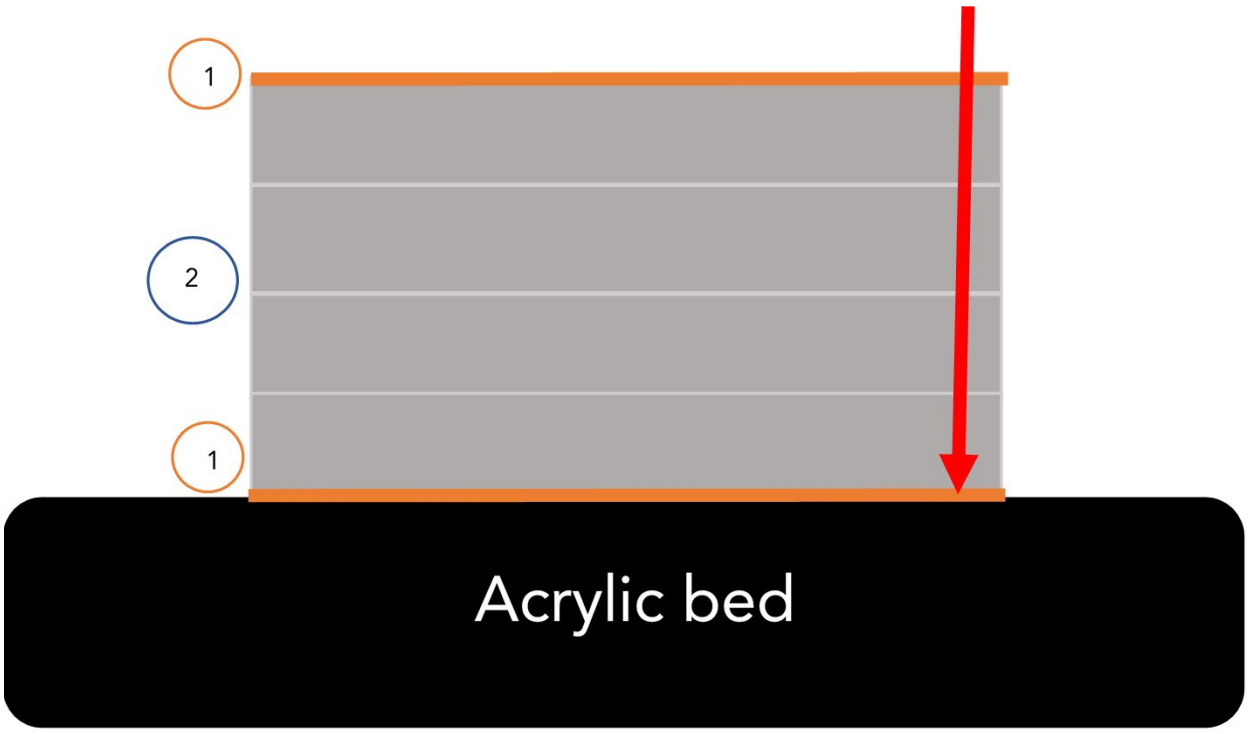


Figure 18. The red arrow shows the laser stopping right before the protective layer that is stuck to the acrylic bed. The number 1 in the orange lines shown the plastic layers that were kept.. The other ones were removed for the four layers (number 2) to be stacked.

New Geometry layout

The new assembly protocol increased the precision of geometries and thus the team was exposed to a new wider array of solutions. Now that the adhesion was effective, the flow was fast and there was no contamination, the control of 5.6 microliters as dictated by the biology team at that time was still unsolved and quite crucial to attaining a better fluorescence with CRISPR and guide RNAs.

In the last PI meeting for July, the idea of lyophilizing the reagents was discussed [7]. The biology team pointed out that there was a chance the reagents could not be lyophilized. This meant we had to account for liquid CRISPR being on the chip, waiting for the sample to go through the inlet, react with the RPA factors and then causing post-amplification fluorescence.

Inspired by the work of Boston U HW and helped by their extensive wiki documentation, the engineering team considered meanders as a way of keeping the liquid moving in a contrainted surface through a meander with a small lateral distance for long enough so that the RPA amplification could happen. Figure 19 shows a helpful depiction of a curved mixer in the Boston U HW 2017 wiki.

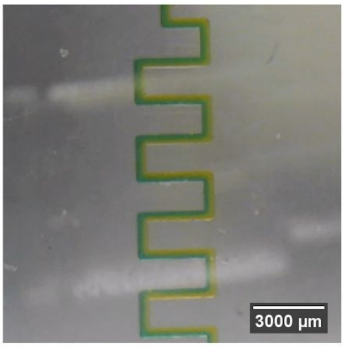
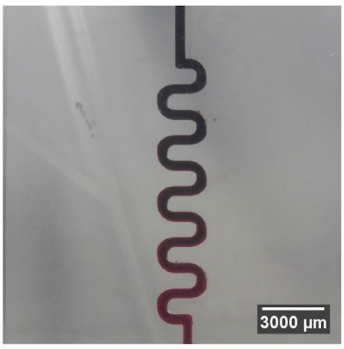
Mixer Mixes two or more liquids together **Design Parameters:** Bend Spacing Number of Bends Bend Length Channel Width		
Curved Mixer An iteration on the previous mixer. Mixes two or more liquids together but has improved fluid flow due to its curved design **Design Parameters:** Bend Spacing Number of Bends Bend Length Channel Width		

Figure 19. Volatect's inspiration for the meanders as mixing chambers.

Meander Design Challenge

Even though the iGEM wiki of Boston U served as an inspiration, the Volatect constraints were vastly different. In less than 5 cm overall lateral distance (distance from beginning of curve to end, the team had to fit a volume of 56 microliters. The height of the four layers was about 3.2 mm (check first section on material testing).

Through an open source Adobe script [9], area of the meandering shape was calculated and multiple iterations were made.

Johannes Kepler University Collaboration

The optimization of the area the meanders took was important since there would be at least 4 paths that would need amplification (one for each pathogen). While researching meandering design principles and ways to optimize the area to volume ratio of the meander, the work by Grimmer, A., Frank, P., Ebner, P., Häfner, S., Richter, A., & Wille of Johannes Kepler University stood out.[10]

Their paper [10] featured a link to an online tool developed by them. Their tool (Figure 20) allowed for the user to enter desired resistance, desired length, bent radius etc and an svg of the file would be generated.

The creation of the meandering path was through heuristic analysis and evaluating functions of A* algorithms.

Input Values		Output Values	
Board Ratio	3	Target Resistance (Corrected)	3 mbar / (µl/min)
Desired Resistance	3 mbar / (µl/min)	Channel Length	12894.87 µm
Tolerance	0.1 %	Channel Volume	0.064 µl
Dynamic Viscosity	1 mPa s	Actual Channel Resistance	3.0011 mbar / (µl/min)
Channel Width	100 µm	Board Width	3172.8 µm
Channel Height	50 µm	Board Height	1057.6 µm
Lateral Channel Distance	100 µm	Board Ratio	3
Bend Radius	50 µm	Performance	329 ms
Starting Point X-Position	0.5 Inlet: top		
Target Point X-Position	0.5 Outlet: bottom		
Correction Factor	None		
C_0	0		
C_1	1		

Figure 20. Input Example for the meander design tool

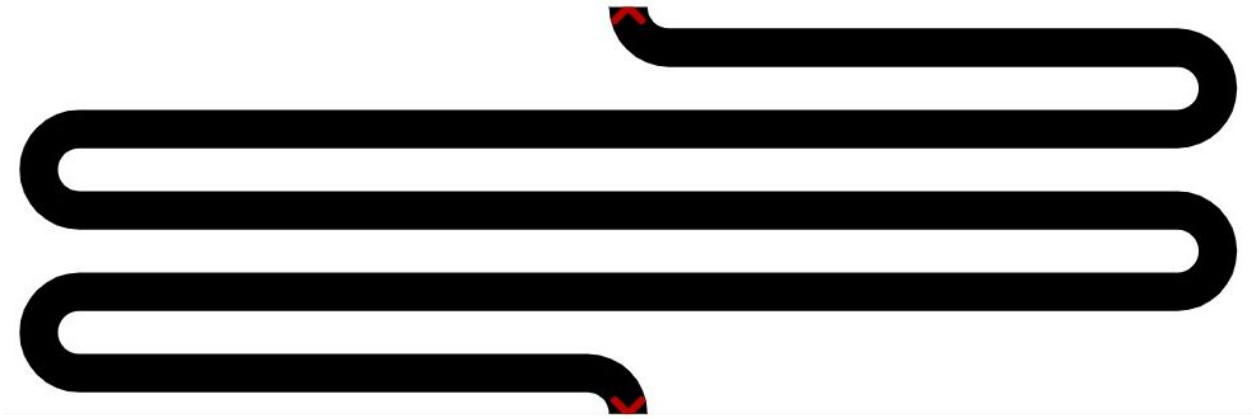


Figure 21.Output example for the meander design tool.

Through some help by the authors of the paper and the beta version of their volumetric designer in addition to the resistive one, adequate meanders were designed that fit our geometric and volumetric constraints.

Meander Chip iterations

Figure 22 shows the final result for the meander-featuring chip. Figure 23 shows the meanders only.

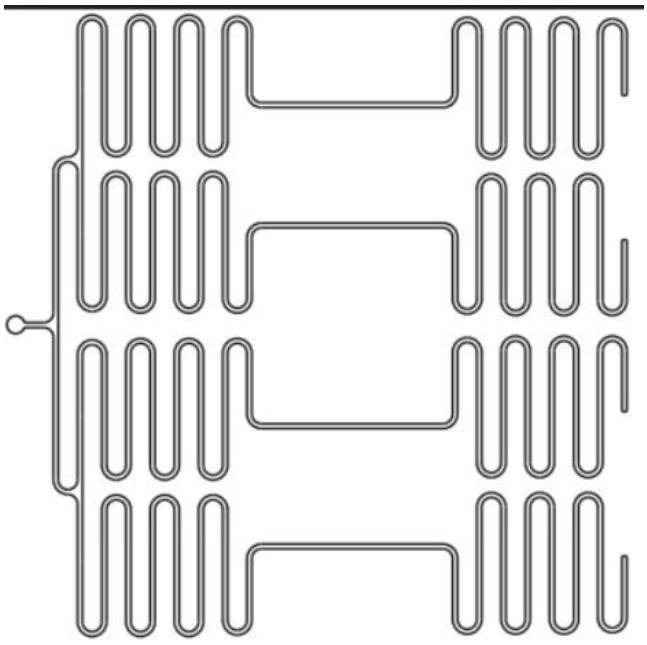


Figure 22. Chips with meanders

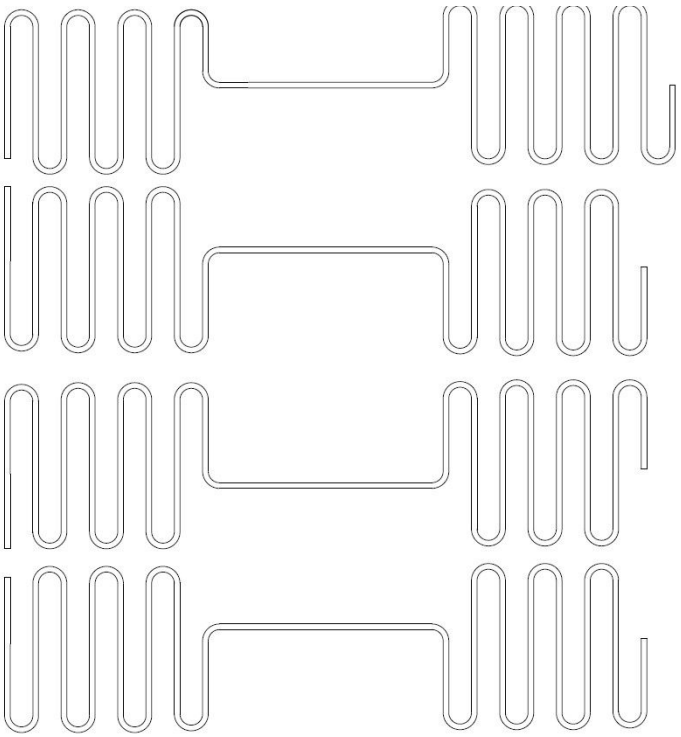


Figure 23. Meanders created and modified

Video of the final meander chip on Figure 22.

https://2019.igem.org/wiki/images/c/c6/T--NYU_Abu_Dhabi--chipassemblyvideo.mp4

September

Week 1-Week 3

Actuation

A working microfluidic chip made entirely out of 3M film is a breakthrough. However, a fully bendable chip made entirely out of flexible material meant that use of solid state actuator valves was tricky or at least not as straightforward. The existing documentation by previous iGEM teams and microfluidics research in general only provided Volatect with valve solutions that employed traditional traditional materials.

Bending the chip to prevent the flow seemed like the most viable solution. However as shown in the video above, bending did not produce consistent results. Figure 24 shows an iteration of

the meander format that used channel width variation to counteract the effects of bending to prevent overflow.

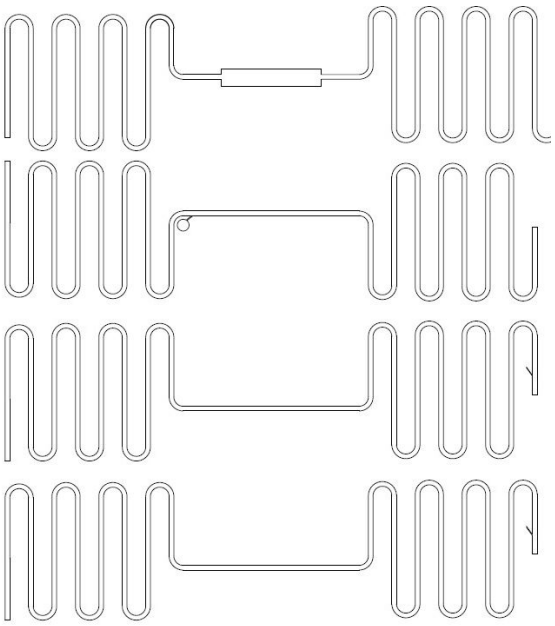


Figure 24. First channel uses width variation, second channel was used to test storage chambers in the side of the channel.

Testing Bending and Inclining as alternatives to valves

The bending principle as a valve replacement should.

(1) Keep the fluid at one place
(2) Make sure the fluid mixes properly when a change in curvature is made

A mechanism was required to add the crispr to the RPA mix only after a certain time interval in order to allow the sample and RPA to react. Thus, a mechanism was needed to introduce the CRISPR at a later time interval. This figure shows the prototype which was made to test out the concept of loading the crispr into an area on the chip which can then be made to incline. This would allow the CRISPR to flow down into the RPA mix at the appropriate time. The edges of the circle are flexible such that they can be bent by hand, or by an actuator. This concept was tested by manually bending the edges with our hand.

CRISPR did not actually flow as expected. Also, at the point of bending, the fluid was unpredictably bouncing back and forth between the inclined and non inclined surface.

Figure 25 shows the chip used to test inclination and bending

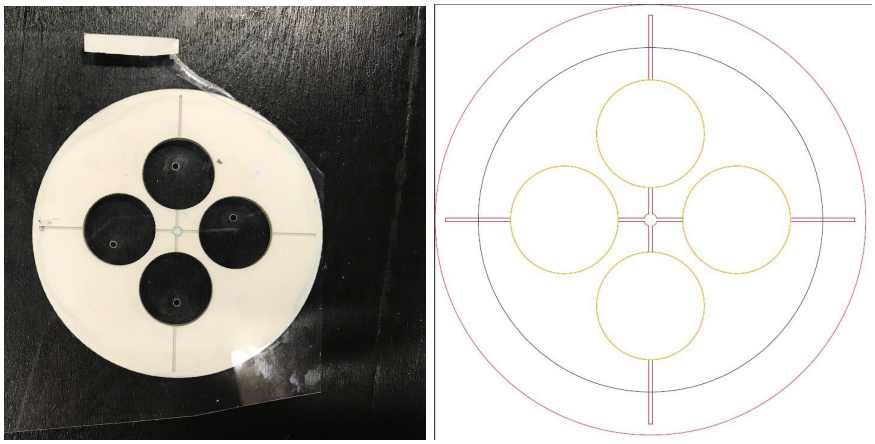


Figure 25. Left: the Chip tested, Right: Laser cut vector file

It was clear that bending had some inherent limits that did not accommodate for the small volumes of liquid that were going to be used.

Air pumps

Air pumps were suggested as a way to move the liquid that did not require bending or inclining. Everything would be kept at the same plane at all time.

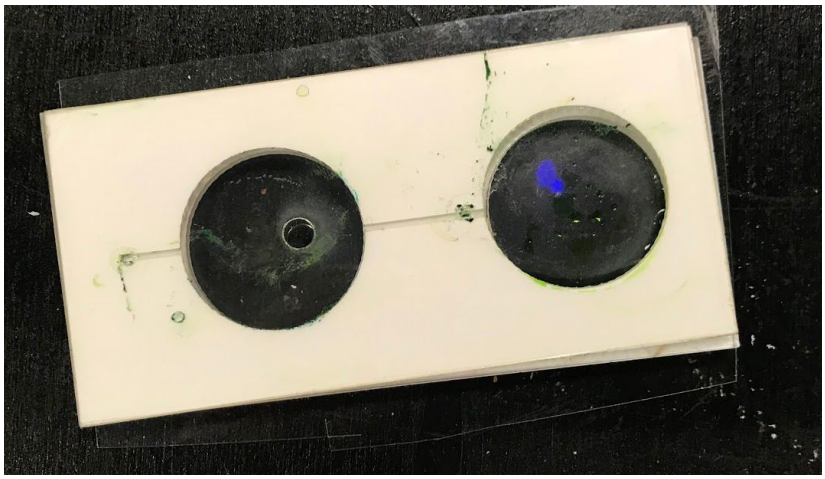


Figure 26 shows a mechanism for the air pump

Figure 26 shows a mechanism for pushing the crispr into a reservoir containing an RPA mix. The RPA was pre-loaded into the chamber on the left side of the figure. The air pump is the chamber on the right. The crisper was loaded into the channel in between the two chambers through a little hole in the film. The hole was then sealed with tape. The air pump chamber was pushed down which someone just did with their finger, and the crispr sitting in the channel was

pushed into the RPA reservoir. When pushed and released quickly, the crispr would jump back and forth into the channel. However, this can be reduced by applying a force on the air chamber slowly and steadily, without a sudden release.

Air pump radius variance test

The efficiency of the air pump resulted in the team trying to optimize the radius for the full potential of this solution to be studied.

The radius of the air pump was varied to experiment with how the size would affect the pushing mechanism. The radius sizes which were tested are 10 and 8 um. The experiment was terminated when it was discovered that a radius of 8um needed too much force to push the crispr to the reservoir which we estimated the motor would not be able to provide.

Multiplexing and channel shape

Multiplexing was decidedly controlled by the air pumps. Furthermore, the inlet would distribute the sample to the four channels. This is where the channel shape was a factor. Zografo's CFD work on simulating various channel shapes and calculating the shear stress in the walls of bifurcating networks [12] helped the team choose the channel shape in the right hand side of figure 27.

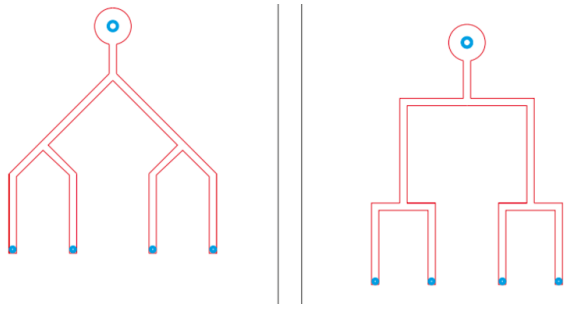


Figure 27. Two candidates for the bifurcating network.

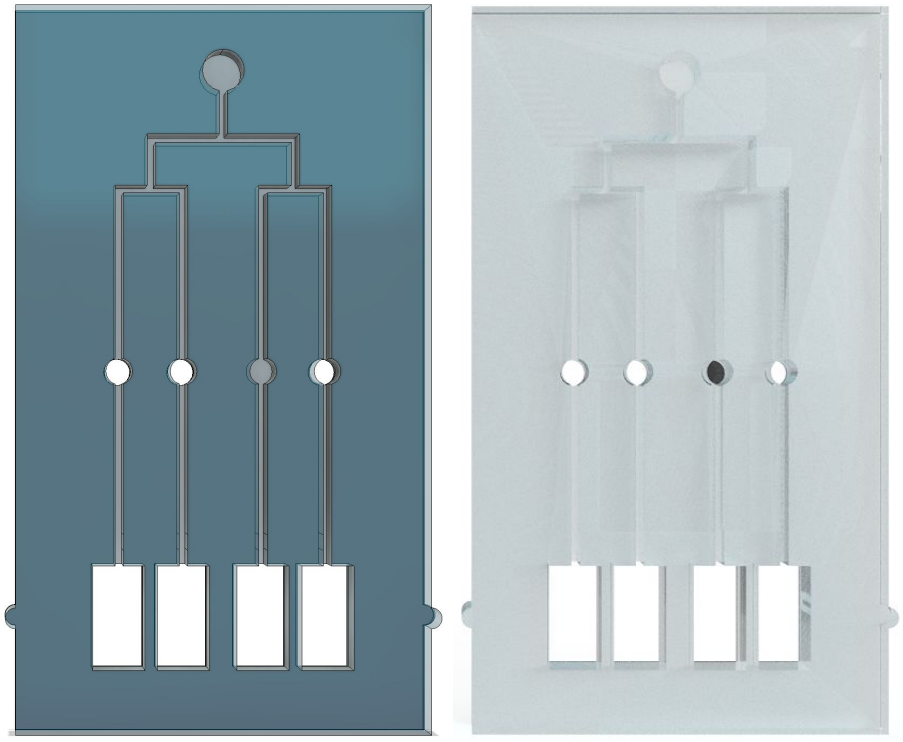


Figure 28. Chip CAD and Rendering

Material Shortcomings

The sandwich assembly (film + tape) failed to keep the air bubbles from entering the system at various instances.

October

Final design and renderings

Having a proper way of actuation, good flow using cost-efficient eco-friendly materials, the tam set out to assemble the chip.

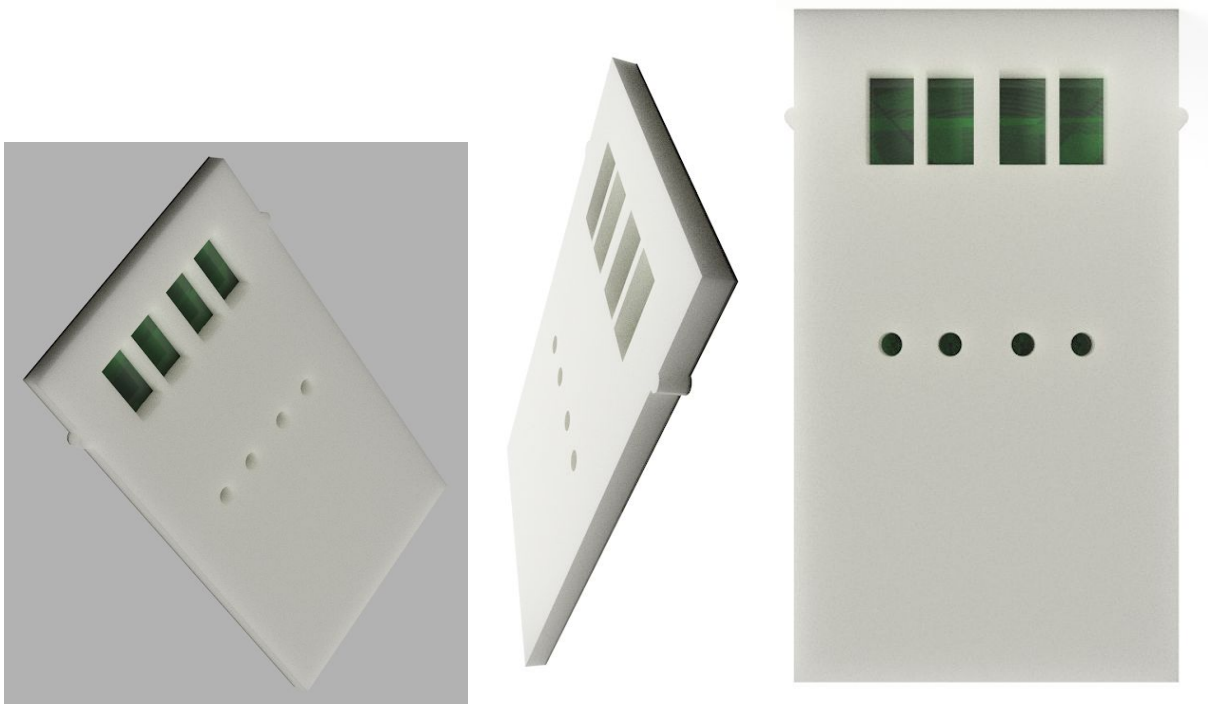


Figure 29. Renderings for the final chip

https://drive.google.com/file/d/1cc8eQY0XOzPdjvJitSs0JFO1K5eYrsnX/view?usp=sharing
Assembly of the final chip

References

[1] Martins, D., Levicky, R., & Song, Y. A. (2015). Enhancing the speed of morpholino-DNA biosensor by electrokinetic concentration of DNA in a microfluidic chip. *Biosensors and Bioelectronics*, *72*, 87-94.

[2] Herold, Keith E., and Avraham Rasooly, eds. Lab on a Chip Technology: Fabrication and microfluidics. Vol. 1. horizon scientific press, 2009.

[3] Mohammad Zarei, Portable biosensing devices for point-of-care diagnostics: Recent developments and applications, TrAC Trends in Analytical Chemistry, Volume 91,2017, 26-41, https://doi.org/10.1016/j.trac.2017.04.001

[4] Mohammad Zarei, Portable biosensing devices for point-of-care diagnostics: Recent developments and applications, TrAC Trends in Analytical Chemistry, Volume 91,2017, 26-41, https://doi.org/10.1016/j.trac.2017.04.001

[5] 3m.com. (2019). *3M™ Microfluidic Diagnostic Film 9960, Thick Double-Sided Hydrophilic Film, CONFIGURABLE | 3M United States*. [online] Available at: https://www.3m.com/3M/en_US/company-us/all-3m-products/~/3M-9960-Diagnostic-Microfluidic-Hydrophilic-Film/?N=5002385+3294605502&rt=rud [Accessed 15 Oct. 2019].

[6] Thought Emporium, Youtube Channel https://www.youtube.com/watch?v=eNBg_1GPuH0

[7] Caliendo, A. M., & Hodinka, R. L. (2017). A CRISPR way to diagnose infectious diseases. New England Journal of Medicine, 377(17), 1685-1687.

[8] http://2017.igem.org/Team:BostonU_HW BostonU HW 2017 wiki

[9] https://gist.github.com/bryanbuchanan/11387501 Script to calculate area on Illustrato

[10] Grimmer, A., Frank, P., Ebner, P., Häfner, S., Richter, A., & Wille, R. (2018). Meander designer: Automatically generating meander channel designs. Micromachines, 9(12), 625.

[11] Obahiagbon, Uwadiae, Joseph T. Smith, Meilin Zhu, Benjamin A. Katchman, Hany Arafa, Karen S. Anderson, and Jennifer M. Blain Christen. "A Compact, Low-Cost, Quantitative and Multiplexed Fluorescence Detection Platform for Point-of-Care Applications." *Biosensors and Bioelectronics* 117 (2018): 153–60. https://doi.org/10.1016/j.bios.2018.04.002.

[12] Zografos, Konstantinos, et al. "A design rule for constant depth microfluidic networks for power-law fluids." Microfluidics and Nanofluidics 19.3 (2015): 737-749.

Actuation

Components:
- NEMA 23 570 bi-polar stepper motor
- DRV-8825 Pololu bi-polar stepper motor driver
- 100 micro-farad capacitor (35V max)
- Raspberry Pi
- 12V power source
- Connection wires

We started out by testing different motors and comparing their power ratings, turn precision, and torque produced.

The stepper motor was chosen for its precision and the torque it could produce to provide proper actuation of the reservoirs on our chip. Several actuation designs were attempted before the final design (see figures below).

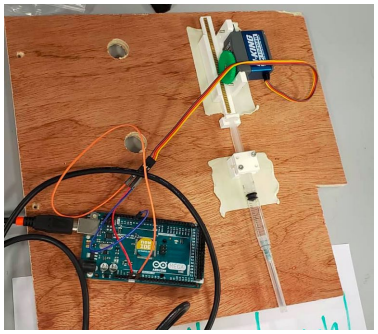


Figure 1: proposed actuation mechanism using servo motor.

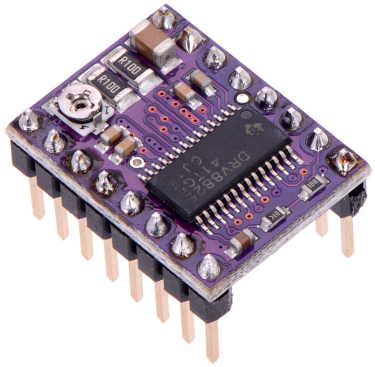


Figure 2: DRV-8825 Pololu bi-polar stepper motor driver

Accurate control of the stepper motor was achieved by micro-stepping. This also prevented jerky movement of the motor.

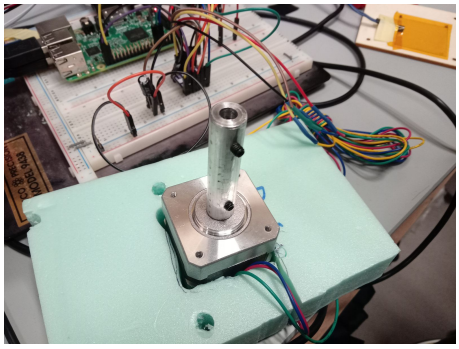


Figure 3: Stepper motor

The motor was observed to heat up after some time as a result of the current being supplied to it by the driver board. The current output on the stepper driver was then carefully changed to a more appropriate value using the reference voltage (V_{ref}) formula:

Current = Reference voltage x 2

Since the current required by the stepper motor, as determined from the datasheet, was 0.33A, the reference voltage was changed to 0.165V using a screwdriver and a multimeter.

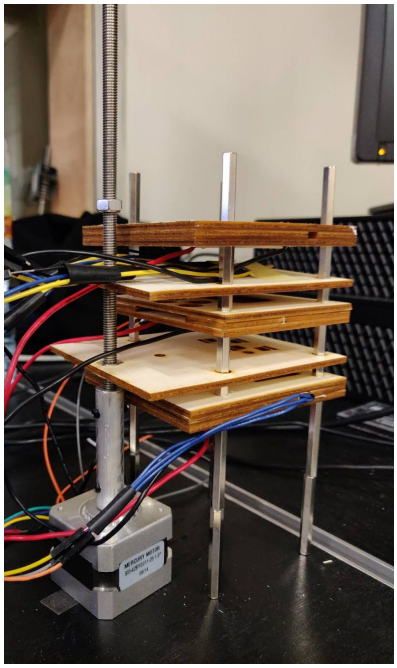


Figure 4: Actuation prototype manufactured out of plywood

Thermocycler

Contents

July 7th: Research 2

July 8th-July 14th Ideation and Discussion 4

July 15th-July 16th Temperature Sensing 5

Sensor Selection 5

July 17th: Peltier Controversy 6

Peltier Alternatives 7

July 18th: Alternative non-electric heating 8

References 9

Appendix 10

Code for the Sensor and heater Circuit 10

Heating Device

July 7th: Research

The team's research work started by looking at last year's research notebook. The old documentation was used as a mean of update, catching up with literature review and understanding the shortcomings of last year's products and documentation methodology. Figure 1 and Figure 2 show some of the design research.

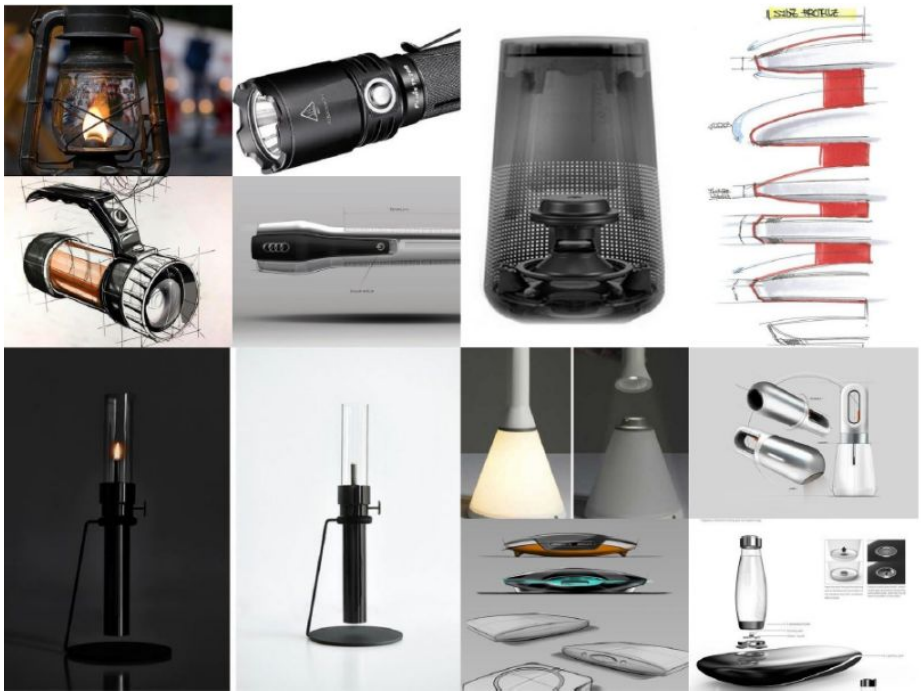


Figure 1 Last year's design exploration

*Figure SEQ Figure * ARABIC 2 Portable heating devices*

Other portable Point-of-Care diagnostic devices were studied as inspiration for possible design solutions. Figure 3-4.

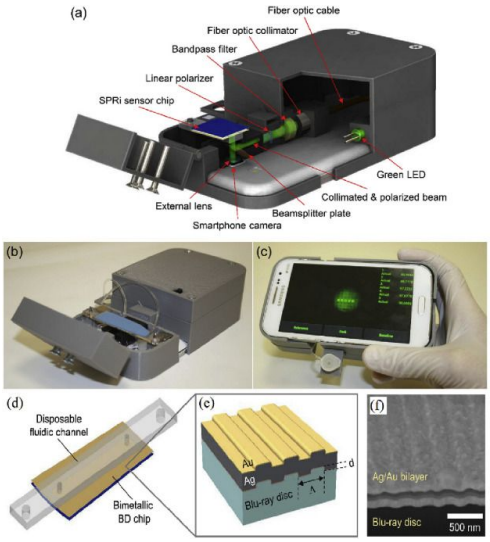


Figure 3. A smartphone-based surface plasmon resonance imaging (SPRi) platform for on-site biodetection by Guner et al. [1]

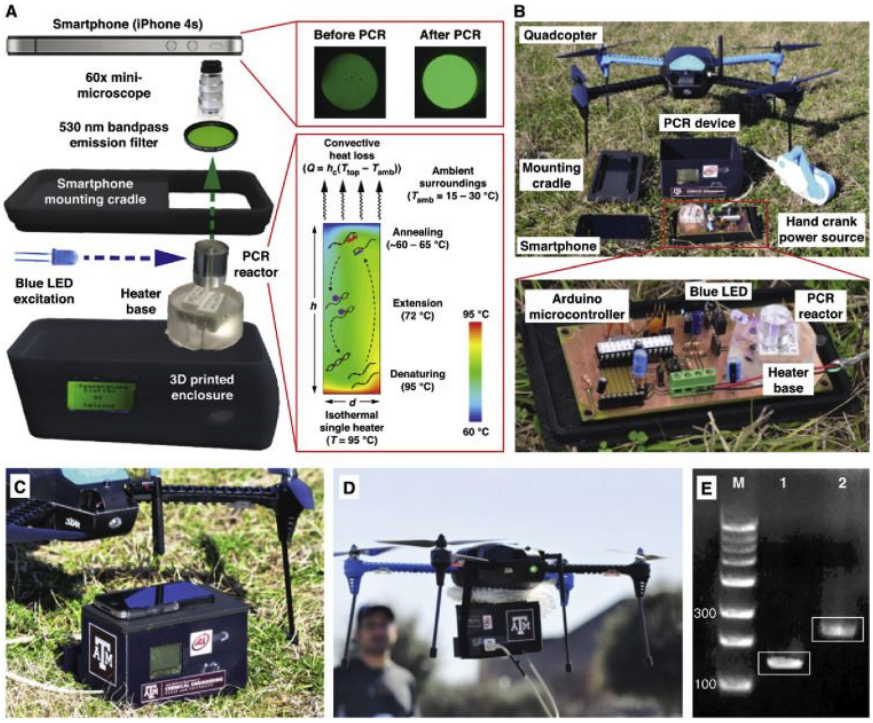


Figure 4 Paper from Zarei Portable biosensing devices for point-of-care diagnostics: Recent developments and applications*. A typical Lab-on-a-drone systems. (A) Isothermally actuated PCR using a single heater. (B) Components of the device. (C) Smartphone ordinary camera was used for fluorescence detection. (D) Lightweight assembly enabled deployment on consumer-class quadcopter drones. (E) Successful in-flight PCR of two different DNA targets [2].*

Reading reviews and opinion pieces in order to get a good cross-sectional understanding of the market solutions was of the essence considering that the heating device is one of the key deciding factors for the shape of the device. (The chip will be inside, and the inserting syringes are not form factors as much as the internal circuitry whose main purpose is the actuation of the heating device to maintain an optimal temperature for the Recombinase Polymerase Amplification).

July 8th-July 14th Ideation and Discussion

The discussion that had started in the early stages of iGEM this year was formalized in a brainstorming that kick-started our prototyping endeavors.

Starting from last year's device were aware of the potential issues that had arisen. Starting from the energy inefficient Peltier element to the inability to assure for a constant temperature.

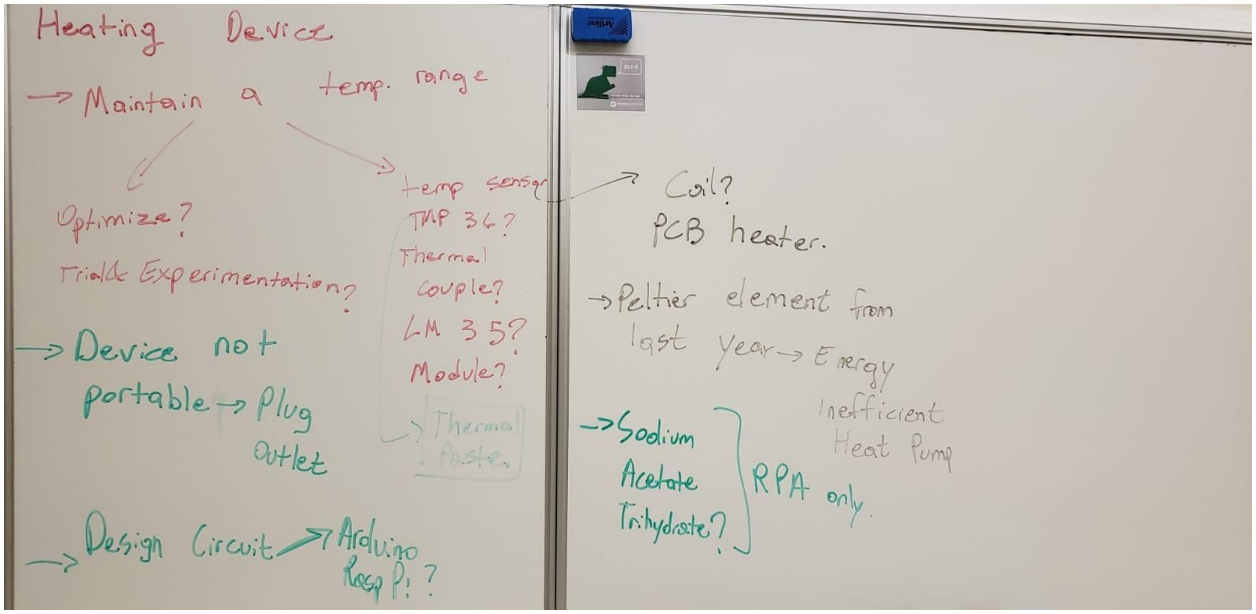


Figure 5. Snipper from the discussion, changing the heater was a main part of the discussion based on last year's feedback

The use of PCB heaters and other non-electronic chemical options were discussed. Maintaining a temperature within a reasonable range was very important. In addition to that, we sought to minimize the energy requirement.

July 15th-July 16th Temperature Sensing

Sensor Selection

TMP 36 was chosen for prototyping and as a potential final sensor to be used (Figure 6). The datasheet for the sensor gives a 2 degrees Celsius error which our research proved to be enough for the RPA temperature range of 30-40 degrees.

TMP 36 -type sensors have a very robust way of determining temperature. They do not use a Thermistor (resistance changing from temperature) or bimetallic strips. They work under the principle that the increase of temperature decreases the voltage between the base and the emitter. By amplifying that difference as a transistor, a signal directly proportional to the temperature can be generated.

Using the respective formula for each sensor, the temperature in Celsius can be expressed as a function of voltage.
Temp in °C = [(V_{out} in mV) - 500] / 10

Figure 6. TMP36 sensor which can be used to measure the temperature using a 5V reference.

However, without the use of a thermal paste to make sure a good contact between the device and the sensor, the sensor provides temperature values way higher than the 2 degrees range

of uncertainty. Later hardware iterations featured better contraption that rendered thermal paste obsolete. However, the principle of insulation helped the team design a good replacement and the experimentation with the paste helped the team understand a new design constraint.

Different electrical engineering forums recommended using DS18B20 which comes in a waterproof version as well. (Figure 7).

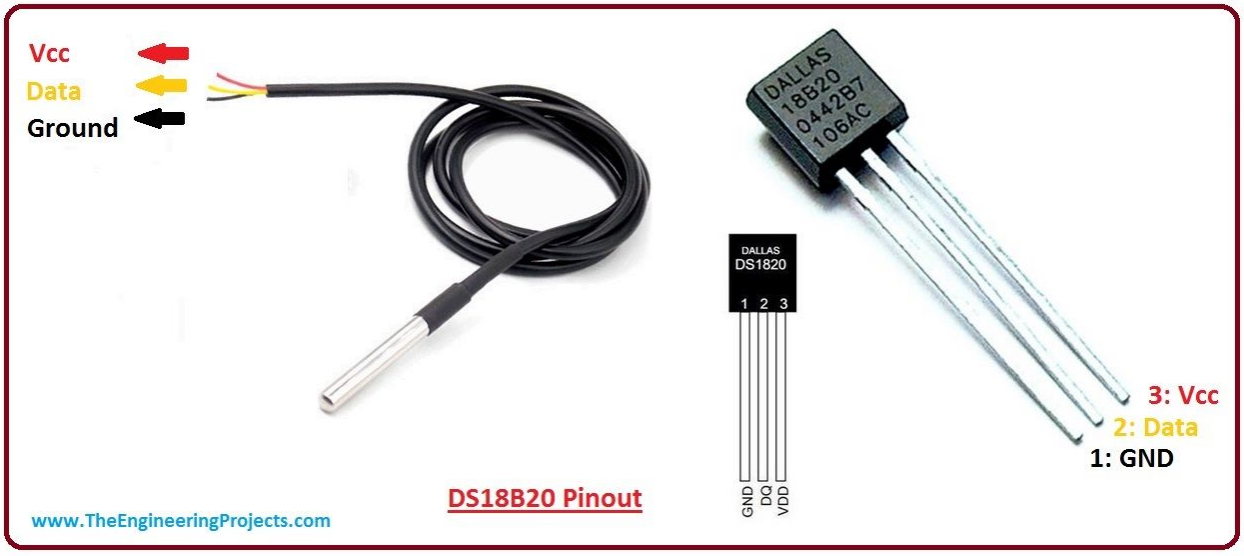


Figure 7. DS18B20

Regardless, the circuit was made with TMP36 as the setup for the DS18B20 would be the same. Further testing with the biological reactions when the microfluidics sufficed in dictating the accuracy of TMP36, which is the sensor of choice for the final design as well.

July 17th: Peltier Controversy

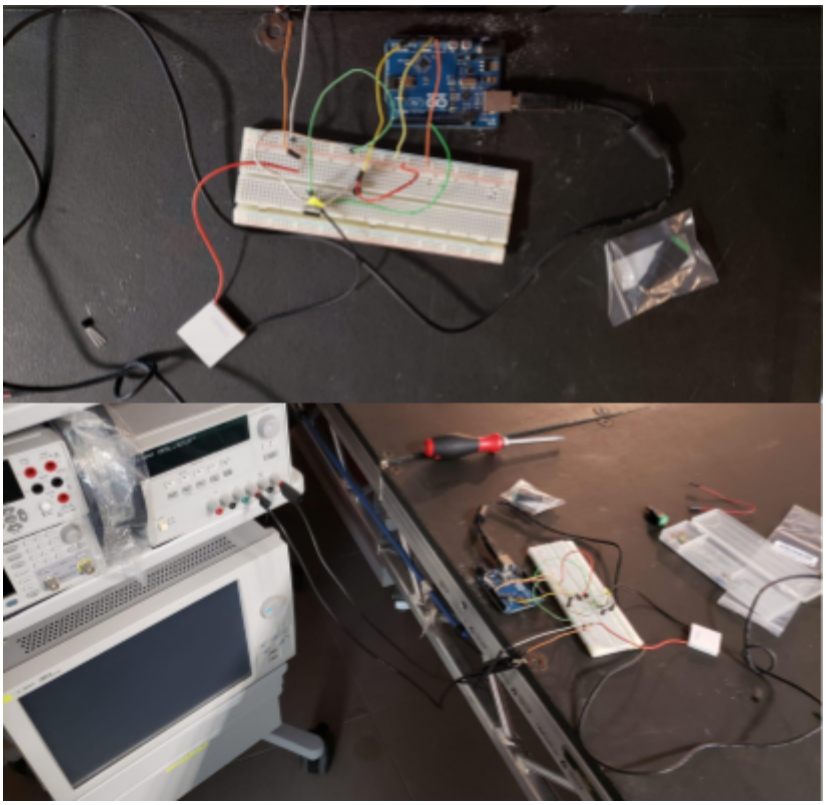
We were told by the previous iGEM team and various instructors that the Pletier element was not the best choice to provide heat due to its energy consumption. A 9V LiPo battery would hold the Peltier running for 8 mins.

However, since the team was using a sensor and this year's device did not have to be portable, the energy requirement was not a problem. Hence, the circuit was built with a Peltier Element in such a way that it could be replaced by more energy efficient parts like a coil or a PCB heater.

Figure 8 shows the circuit which turns on the Peltier when the temperature is below a certain threshold. The experiment was made with a Power Supply before the actual energy

requirement were discussed as that would also depend on the stepper motors for the pumps or the servos for the rotating parts.

The new code for the circuit in Figure 8 made by the 2019 iGEM team can be found in the Appendix. The code from last year was not initially used but was later requested for the sole purpose of understanding the design that we set out to improve. The last year's code was provided by its author Kai-Wen Karen Yang. Other codes and iterations can be found on Volatect's Github. This Appendix features code that is useful to test temperature sensors with any type of heater.



Peltier Alternatives

In the spirit of iGEM, the team took the feedback from the previous year and consulted with Electrical Engineers on the NYUAD campus (refer to Attributions page for more).

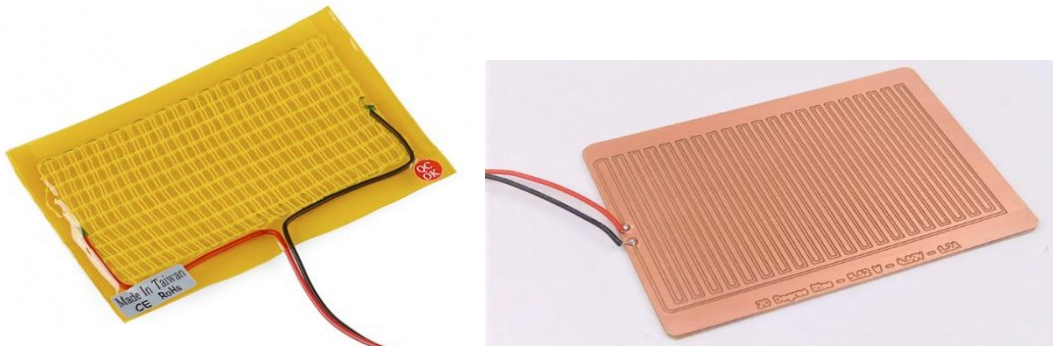


Figure 9 Heating Pad on the left and a PCB heater on the right, alternatives that may replace the Peltier heater.

July 18th: Alternative non-electric heating

Another heating alternative that was brought to our attention by the biology team was the use of Sodium Acetate Trihydrate (SAT). Inspired by Lilli's et al. 's paper "Non-Instrumented Incubation of a Recombinase Polymerase Amplification Assay for the Rapid and Sensitive Detection of Proviral HIV-1 DNA" [3], As can be seen in figure 10, this heating contraption does not require electricity. Commonly used in hand warmers, the 70-75% percent concentration in water was found to be adequate for an optimal RPA temperature. (The lower or higher the concentration, the more necessary would time adjustments be). The authors of the paper provided the illustration shown in Fig. 10 which is still a prototype. The initial design is anticipated to need a kinetic switch to be initiated. The device, as the authors say, can then be easily placed in a cup of boiled water to resolubilize the SAT before it is ready to be used again.

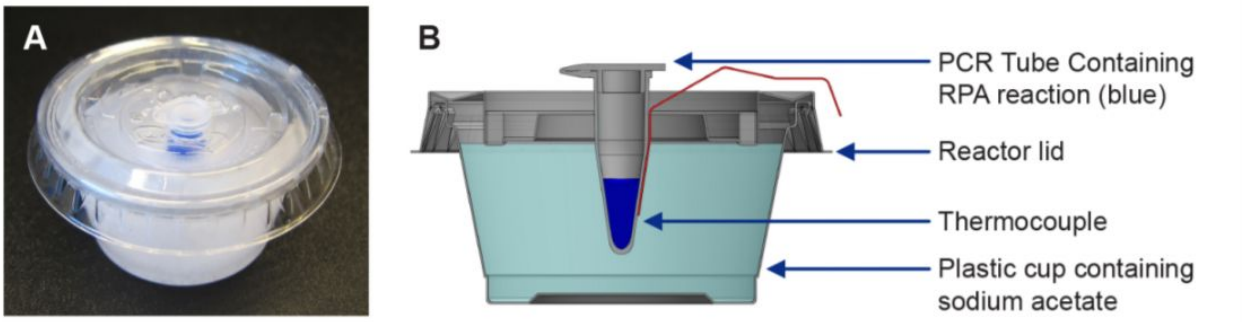


Figure 10. The prototype SAT heater used to incubate RPA reactions. (A) A photograph of an activated SAT heater, the reaction tube is filledwith blue dye to improve definition. (B). A cross sectional diagram depicting the SAT heater and other test components. doi:10.1371/journal.pone.0108189.g001

References

[1] Guner H., et al. , "A smartphone based surface plasmon resonance imaging (SPRi) platform for on-site biodetection," Sens. Actuators, B 239, 571–577 (2017).https://doi.org/10.1016/j.snb.2016.08.061 [Google Scholar]

[2] Mohammad Zarei, Portable biosensing devices for point-of-care diagnostics: Recent developments and applications, TrAC Trends in Analytical Chemistry, Volume 91,2017, 26-41, https://doi.org/10.1016/j.trac.2017.04.001

[3] Lillis L, Lehman D, Singhal MC, Cantera J, Singleton J, Labarre P, et al. (2014) Non-Instrumented Incubation of a Recombinase Polymerase Amplification Assay for the Rapid and Sensitive Detection of Proviral HIV-1 DNA. PLoS ONE 9(9): e108189. https://doi.org/10.1371/journal.pone.0108189

Appendix

Code for the Sensor and heater Circuit

Code written to test the working principles of the TMP 36 and Peltier circuit.

```
//TMP36 Pin Variables
int sensorPin = A0; //the analog pin the TMP36's Vout (sense) pin is connected to
const int transistorPin = 8;

void setup()
{
  pinMode(8, OUTPUT);
  Serial.begin(9600);  //Start the serial connection with the computer
                       //to view the result open the serial monitor
}

void loop()                     // run over and over again
{
 //getting the voltage reading from the temperature sensor
 int reading = analogRead(sensorPin);

 // converting that reading to voltage, for 3.3v arduino use 3.3
 float voltage = reading * 5.0;
 voltage /= 1024.0;

 // print out the voltage
 Serial.print(voltage); Serial.println(" volts");

 // now print out the temperature
```

```
 float temperatureC = (voltage - 0.5) * 100 ;  //converting from 10 mv per degree wit 500 mV
offset
                                        //to degrees ((voltage - 500mV) times 100)
 Serial.print(temperatureC); Serial.println(" degrees C");


 if (temperatureC<19)
 {


      digitalWrite(8, HIGH);


 }


 else{
  digitalWrite (8, LOW);
 }
 delay(1000);                                //waiting a second
}
```

Final Thermocycler

Components:
The materials used for building the heating device:
- Thermoelectric Peltier Heater/cooler (TEC1-12706)
- Temperature Sensor (TMP36)
- N-channel Transistor
- 12V power adapter
- Thermal paste
- Arduino nano
- Schottky Diode (specs)
- Connection wires

For our device, a thermoelectric cooler (TEC) was used to ensure rapid heating and cooling. Since TECs make use of the Peltier effect, a temperature difference can easily be achieved between both sides of the device. Another option which we considered for heating purposes is the thermal coil. However, thermal coils take longer to heat and cool as compared to TECs. With speed being of such high importance to our device, we chose the Peltier over the heating coil. Another reason we had for not using a thermal coil was the fact that it requires placement on an additional board where the chip would sit. This increases the heating period as the board-and-chip combination would require more time to achieve thermal equilibrium with the coil.

Figure 1: Heating coil

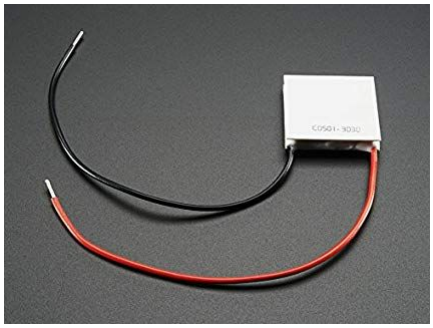


Figure 2: Thermoelectric cooler (peltier)

The TEC was powered using an external power adapter rated 12V and 1.5A. With this input voltage, the Peltier heats up to the required temperature (39℃) in less than five seconds.

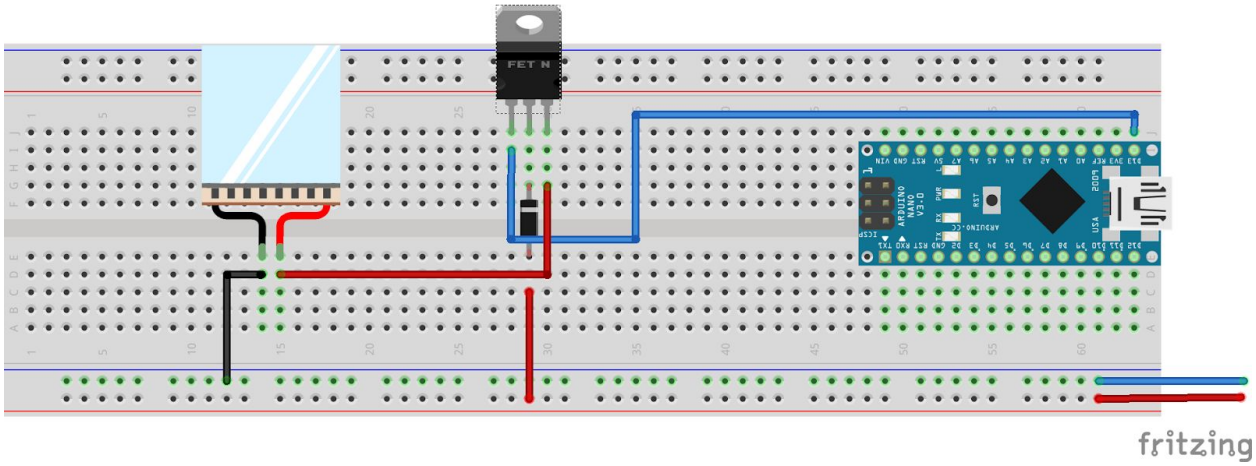


Figure 3: TEC circuitry

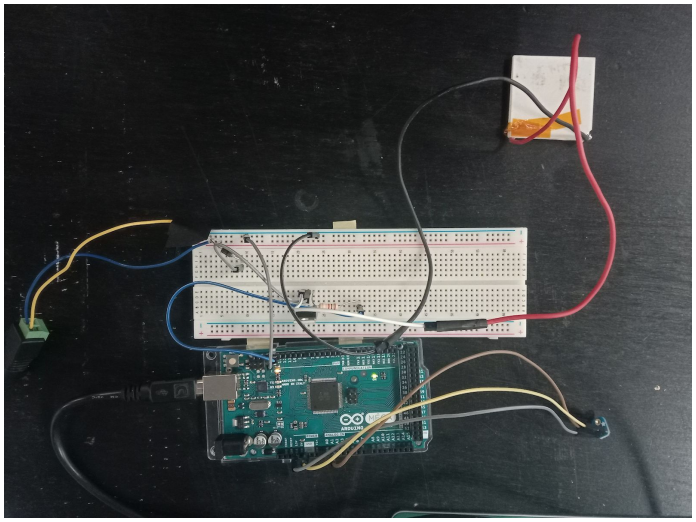


Figure 4: Physical Heater Circuitry

Temperature control was achieved using a high precision TMP36 temperature sensor. Other TMP sensors were tested, including the TMP102 sensor which proved to be very sensitive as well. However, the TMP102 sensor breakout board was too large to fit properly into our device. Making it fit would lead to other unnecessary trade offs as the TMP36 already displayed good performance.

Figure 5: TMP36 temperature sensor

Regardless of the sensitivity of the heating device, thermal equilibrium could still not be established between the TEC and the temperature sensor. As a result, the TEC was always at a higher temperature than that read by the TMP36. To bypass this problem, we made us of thermal tape which significantly increased the heat conduction between the TEC and the TMP36 and further reduced the time it took the TMP36 sensor to give a good approximate of the temperature of the TEC. Despite the thermal tape used, we still noticed some ±2 degrees difference in the temperature of the TEC and the value read by the TMP36 sensor. Temperature calibration was then implemented by measuring the temperature of the TEC using a high precision IR temperature gun and then setting the appropriate value (in the Arduino code) to achieve the required temperature.

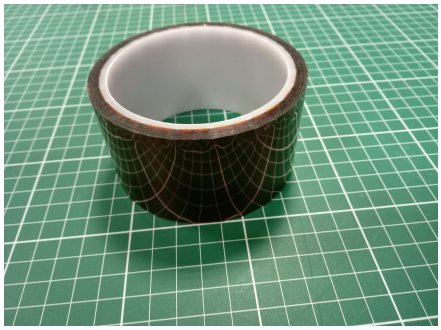


Figure 6: Thermal tape

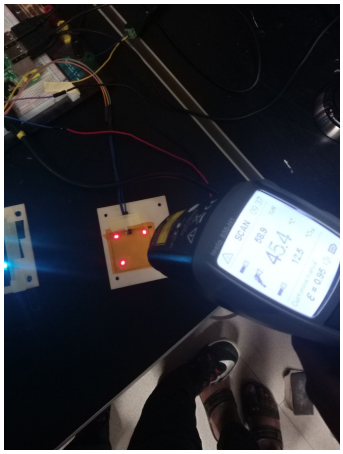


Figure 7: Temperature calibration using IR Thermometer

Data for the TMP36 sensor was used to control the TEC using an N-channel Metal Oxide Semiconductor Field Effect Transistor (MOSFET). The MOSFET allowed for easy flipping between the on and off state for the TEC, thereby regulating its temperature. Reverse current generated by the thermoelectric effect when the TEC is switched off was prevented by connecting a reverse biased Schottky diode in series with the TEC. Schottky diodes were chosen because they have a low forward voltage and hence ensure that only a very small fraction of the applied voltage is lost to the diode.

Figure 8: N-channel MOSFET

Figure 9: Schottky diode for reverse current prevention

Though using a single transistor control system is not recommended in high temperature applications, the single transistor design worked fine for our application since our device's temperature does not exceed 50℃.

Additionally, although the TEC initially cools down one surface while heating up the opposite side, over time, temperature from the hot surface is conducted to the cold one. When the hot surface was at about 40 degrees, the cold surface would get up to about 23 degrees after about five minutes. To solve this problem, we included a 40mm x 40mm x 0.6 mm aluminium plate (as a heat sink) to the cold side of the TECs to dissipate the heat transferred from the hot side.

Figure 10: Aluminium plate used for heat sink

This would however be unnecessary if the TEC was controlled using an H-bridge setup of two N-channel and two P-channel MOSFETs. An H-bridge connection would enable us to reverse the direction of the current going into the TEC, therefore making it possible to switch the hot and cool sides. This was not used, however, because a reversal of hot/cold side was not necessary to cool it down back to normal temperature at the maximum temperature achieved by the TEC (approximately 45℃).

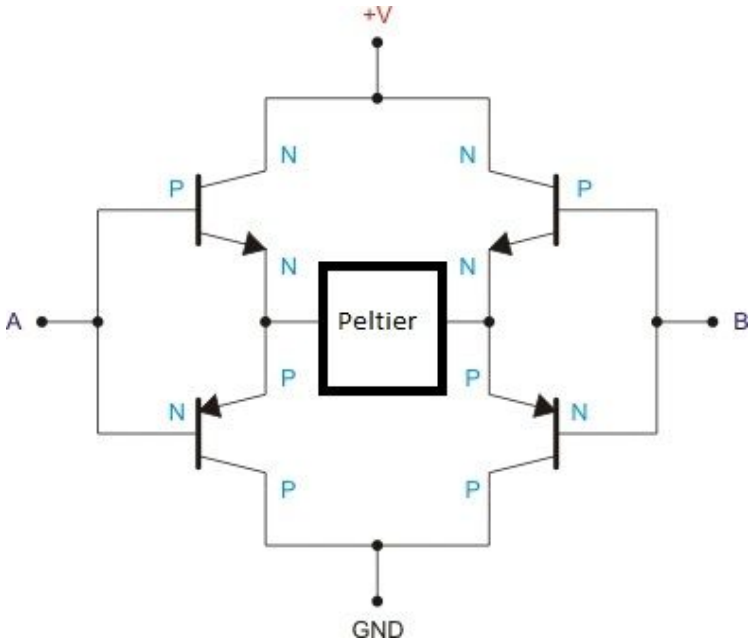


Figure 11: H-bridge circuit for TEC cooler control

Fluorimetry

Components:
- TEMT6000 sensor
- LED465E (465nm LED)
- Arduino nano
- Connecting wires

For fluorescence detection, several sensors were tested. These includes:
- TEMT6000 sensor
- Photocell (LDR)
- Photodiode
- RGB color sensor (TCS34725)
- Raspberry Pi camera
- ALS PT19 light sensor

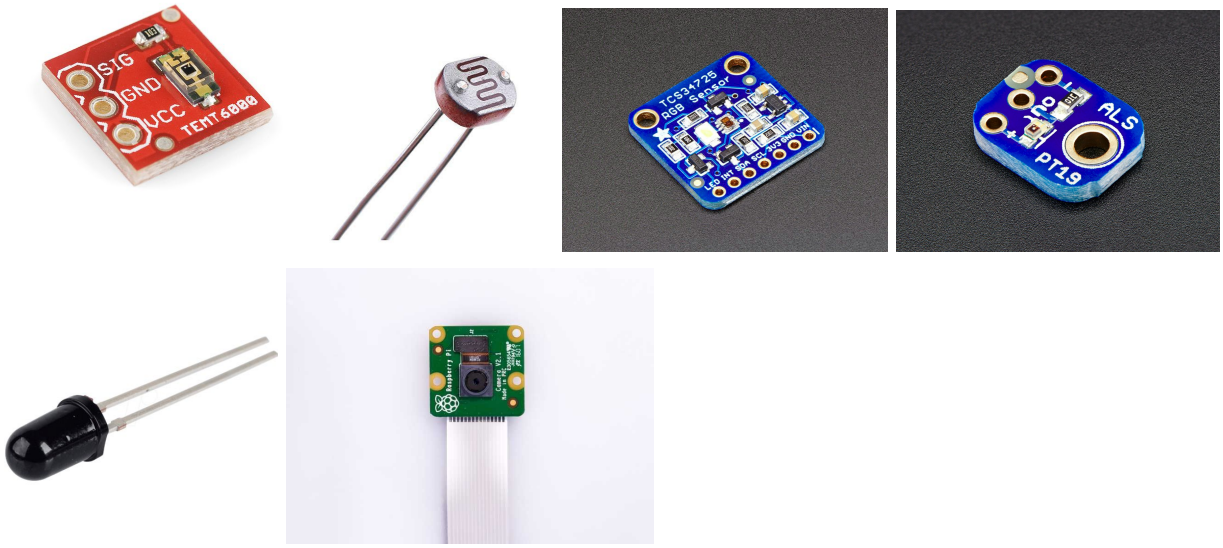


Figure 1: Some of the many sensors that we tested.

Our choice of sensor was based on the following criteria:
- high sensitivity
- manageable size (to fit into device)
- data output (voltage vs current) – we chose to work with devices that gave a voltage output over one that gave a current output in order to simplify the circuitry i.e. since an Arduino can only read voltage, we need not build a current-to-voltage circuit in the case of a voltage output.

Some of the other Sensors tested:

The TCS34725, with its integration of both clear and filter-covered photodiodes, was an appropriate and tempting choice of light sensor for us. However, its use was limited by its dimensions which were simply too big for 4 of them to fit in our device. Photoresistors (Light Dependent Resistors(LDR)) could not generate good readings of fluorescence due to their low sensitivity as compared to Photodiodes and Phototransistors. We also attempted to use a Raspberry Pi camera. Though this was one of the coolest ideas we came up with, it just simply posed a lot of restraints in terms of processing power, processing speed and also the sensitivity of the camera. In order to use a Raspberry Pi camera for detection, we have to take images for both the control and the final samples, and then run them through OpenCV image processing platform. However, OpenCV takes time to process the images and quickly consumes the Raspberry Pi memory. One other disadvantage of using the camera was its low sensitivity; it was sometimes unable to pick up signals from samples with weaker fluorescence.

Our Sensor:

The TEMT6000 sensor is an ambient light sensor that has a very broad range of applications. It consists of a phototransistor placed on a break-out board. The phototransistor can detect light in the visible range with a peak sensitivity at 570 nm which is close to the peak emissions of 556 nm for the quencher used in our CRISPR reaction. An alternative to using phototransistors would be to use photodiodes which offer faster response time than their phototransistor counterparts. However, since the reaction time of our sensor poses no constraint on our detection, phototransistors were selected for their superior sensitivity.

Noise reduction:

We took several steps in our design process to minimize, if not eliminate, noise. Some of these steps include:

- We isolated the reaction chamber from ambient light by enclosing the reaction chamber with a completely opaque covering.
- We attempted to isolate the sensor from the light generated by the excitation blue lights by putting filters on the sensors. However, the bandpass filters allowed some stray wavelengths from the blue lights to be detected by the sensor. We overcame this problem by getting custom-made LEDs with smaller emission spectra (±10nm) from ThorLabs.
- Another method that we tried is what we call the "time-dependent detection". What this means is that we turn on the LEDs for a short period of time (about 1s) and then switch them off. At the moment the LEDs are turned off, the light sensors then read the remaining fluorescence values from the reaction wells. A problem we encountered with this method is that the fluorescence stopped almost immediately after the LEDs were switched off (within microseconds making it difficult for the sensor and the Arduino to pick up).

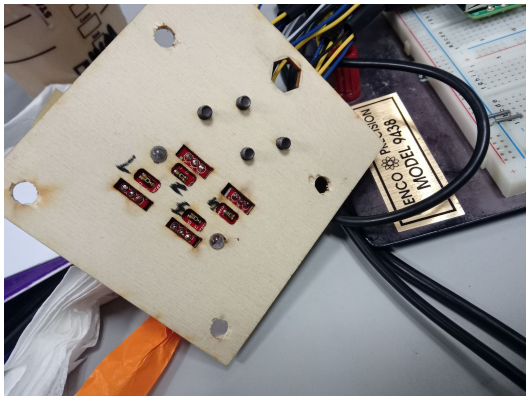


Figure 2: TEMT6000 placement

We also tested several light sources for excitation of fluorescein in the sample. Fluorescence generated from excitation using each of these light sources is shown below:

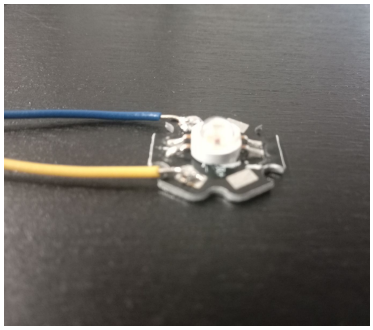


Figure 3: LED 1

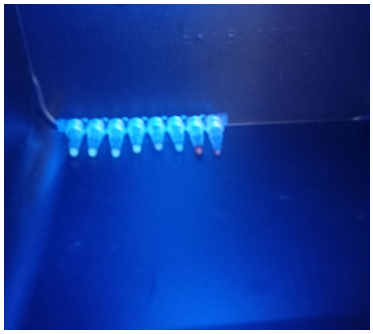


Figure 4: LED 1 fluorescence test

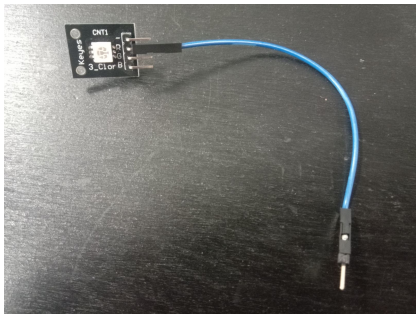


Figure 5: LED 2

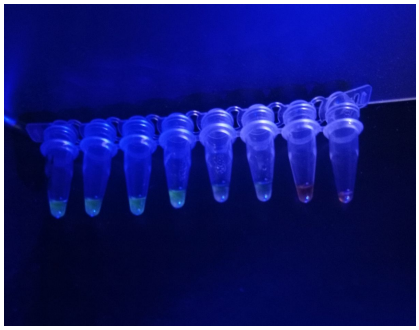


Figure 6: LED 2 fluorescence test

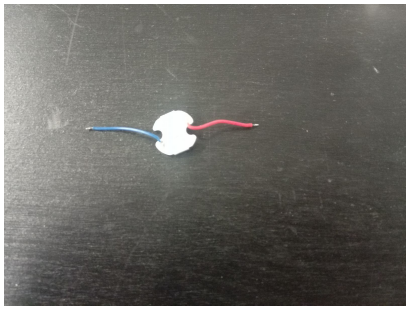


Figure 7: LED 3

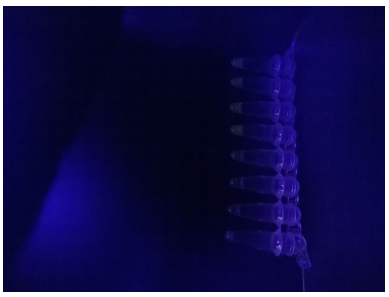


Figure 8: LED 3 fluorescence test

Figure 9: LED 4 (chosen)

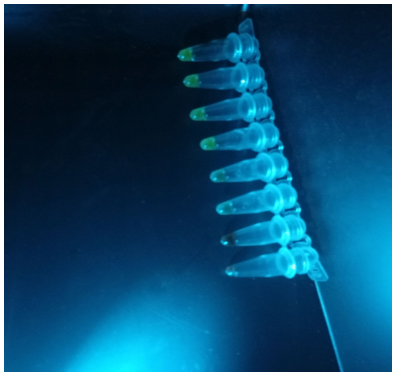


Figure 10: LED 4 fluorescence test

Our final choice of excitation light source, the THORLABS LED465E (figure 9), was based on the following criteria:

- amount of fluorescence generated by the light source
- light intensity
- spectral range of emission (the smaller the spectral range the better)
- size
- power consumption and heat dissipation

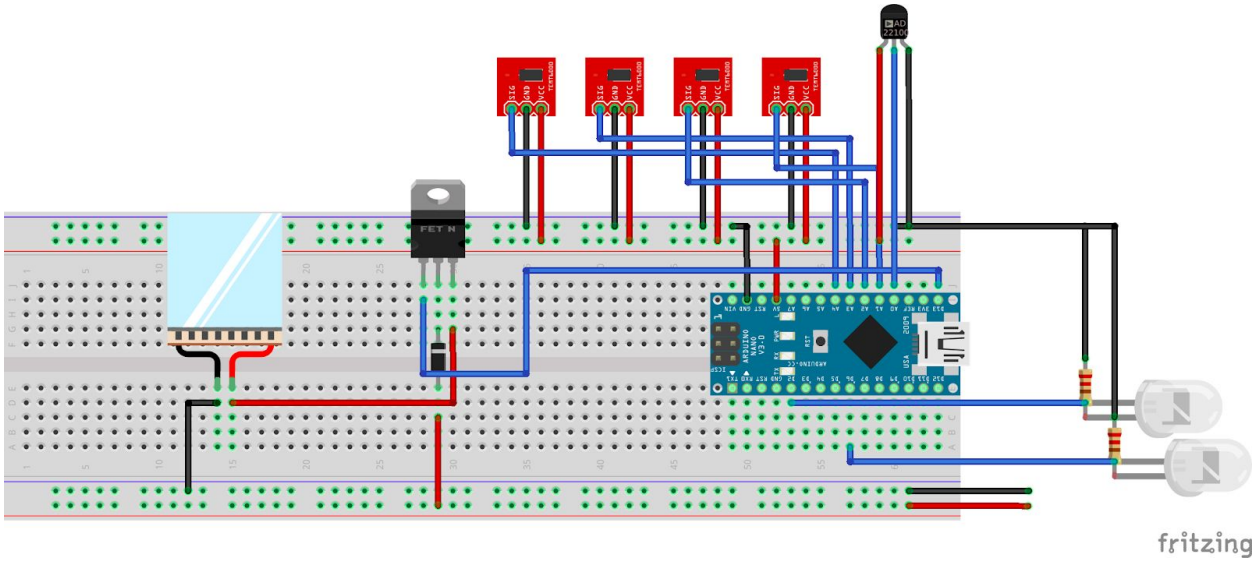


Figure 11: TEC, TEMT6000, TMP36, AND LEDs circuit

INDICATOR LIGHTING

Components:

- 2 Shift registers
- Arduino mini
- 5 RGB LEDs
- 15 330 ohm resistors
- Connecting wires

RGB LEDs were used for the indicator lights. Since the indicator lights make use of 5 RGB LEDs, 15 I/O pins are required. However, an Arduino Nano contains only 13 digital I/O pins. An easy solution to this would have been to use an Arduino MEGA which contains 52 digital I/O pins. However, in order to present a more elegant and less power consuming (from the Raspberry Pi) solution, we opted to use Shift-out registers. With shift-out registers, we could control all 5 RGB LEDs with just three pins from the Arduino and have other pins for all other input/output functions.

Figure: Shift Register

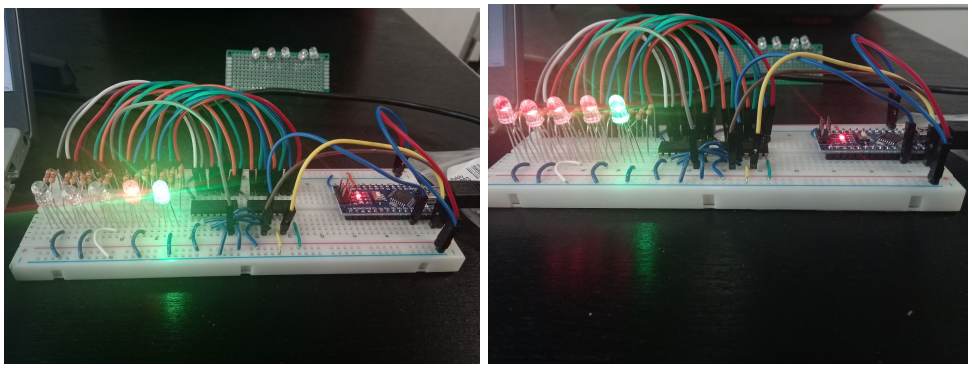


Figure: Indicator lights (in action)

Below is the schematics for the Indicator light circuit.

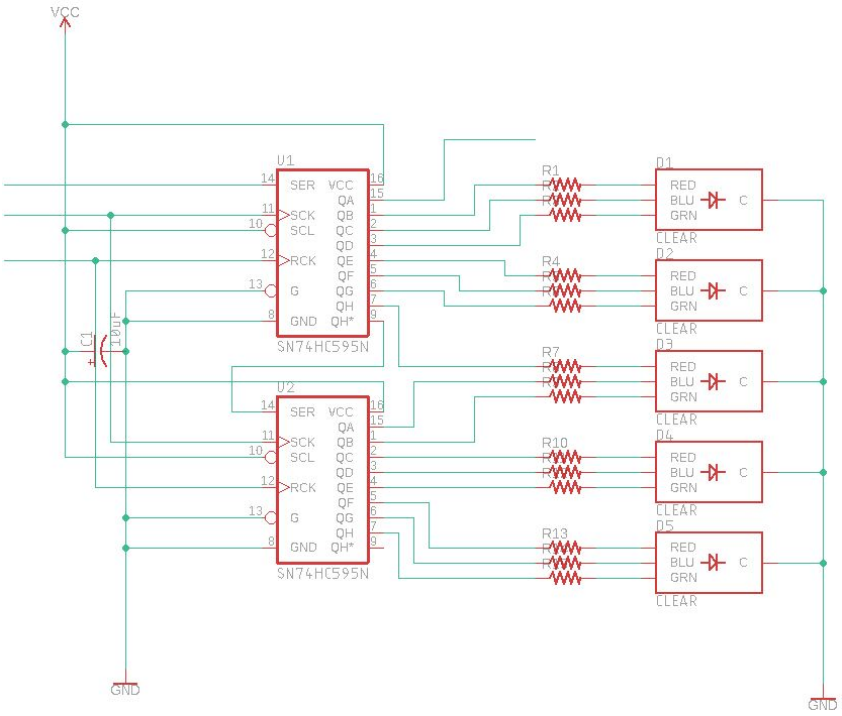


Figure: Circuit diagram for indicator lighting

The functions used to control the shift register were built from scratch to enable for an easy and less abstracted control of the LEDs.

Arduino code for indicator light control:

Working of indicator light:
The indication light consist of 5 RGB LEDs:
LED 1:

When the device starts up and begin receiving information via serial communication from the Raspberry Pi, the first LED turns green. If a serial communication cannot be established, the first LED turns red, and the reaction won't start until a serial communication can be established.

LED 2:

The second LED indicates whether the Raspberry Pi is connected to the internet or not. If the Raspberry Pi is connected, the LED emits green light. However, if the Raspberry Pi is not connected, the LED turns red. The Arduino keeps checking for the internet connection state of the Raspberry Pi, and turns green at any point in the reaction when the Raspberry Pi becomes connected to the internet.

LED 3,4,5:

The other 3 LEDs are in-charge of indicating the progress of the reaction going on in the well. When the reaction is ongoing, the 3 LEDs blinks blue. When the reaction is done, the LEDs blinks green. Note that the LEDs are time controlled based on calibration with test experiments.

POWER

The Raspberry pi, stepper motor, and Thermoelectric cooler were powered from a 12 V source. The Arduino board was powered through the serial USB interface to the Raspberry Pi. All other sensors were powered from the Arduino board I/O pins. The power supply used could generate 3A which is sufficient for the whole device to run.

Figure: Power adapter

DATA ACQUISITION, TRANSMISSION, AND PROCESSING

All data acquisition is done by the Arduino nano. The four TEMT6000 sensors connected to the analog pins 1,2,3 and 4 receive data from the wells on the chip. This data is stored in variables and are transmitted to the raspberry pi via serial communication for processing. Also temperature data from the TMP36 is stored in a variable on the Arduino and is transmitted to the raspberry pi for control. The two excitation LEDs are also controlled by signals from the Raspberry pi. This means that the excitation LEDs hooked up to the Arduino can be turned on and off from the raspberry pi appropriately depending on the situation.

Data from the 4 light sensors and the TMP36 are also stored in a csv file as they are received by the Raspberry Pi. This data is then plotted to generate a graph of lux vs time over the period of the reaction.

Detection of fluorescence is achieved by measuring the value of the lux (read by the TEMT6000) of the control measurement and comparing it to the value of the lux after the reaction. Note that control measurements are taken from each well after five minutes, and this value is compared to the lux value at the end of the reaction.

For fluorescence to be confirmed as positive, the difference between the control lux measurement and the lux measurement taken at the end of the reaction must be greater than or equal to 25 lux units (candela).

Raspberry pi code for data transmission and receiving (Python):

```
port.write(str(internet))   #internet status (sent to arduino via the serial monitor)
        port.write('1')           #LED1 control
        port.write('1')           #LED2 control

        TIME = port.readline().strip()
        light_sensor1 = port.readline().strip()
        light_sensor2 = port.readline().strip()
        light_sensor3 = port.readline().strip()
        light_sensor4 = port.readline().strip()
        temperature = port.readline().strip()
```

Arduino function for data transmitting and receiving (cpp)
:

```
void transmit_and_receive_data()
{
 if (Serial.available() > 0)
 {

  if (starter == 0)
  {
   starting_time = millis();
   working = 1;
   starter = 1;
  }
```

```
  present_time = millis();
  internett = Serial.read();
  led1_data = Serial.read();
  led2_data = Serial.read();
  time_lapsed = present_time-starting_time;
  Serial.println(time_lapsed);
  Serial.println(sensor1);
  Serial.println(sensor2);
  Serial.println(sensor3);
  Serial.println(sensor4);
  Serial.println(temperatureC);

  int led1 = led1_data-'0';
  int led2 = led2_data - '0';
  digitalWrite(LED1,led1);
  digitalWrite(LED2,led2);
 }

}
```

DEVICE STARTUP

The python script responsible for the control of the device is run on startup. This is done by executing the python file in the /etc/rc.local file in the raspberry pi with superuser access.

Code: sudo python /etc/rc.local/receive_transmit.py &

This ensures that the device can be shutdown, and on start up, the device begins running again without any user attention needed.

Note that the device can be run as a standalone, and can also be connected to a screen via the HDMI port at the back of the device. With a screen, a user has access to the data in the csv files generated by the device for data processing. Also device functionality can be configured by the user. However, this requires super user access which is not by default given to the user.

CODE

RASPBERRY PI:

```
#!/usr/bin/python
'''
code by Jimoh Yusuf Omotayo
'''
import subprocess   #for internet connection check
import requests     #for sending data to API
import serial       #for serial communication
import time         #for time management

myfile = open("loggedData.csv",'w')
myfile.write("Time(s),Sensor1(cd),Sensor2(cd),Sensor3(cd),Sensor4(cd),temperature(deg.C)\n")

port=serial.Serial('/dev/ttyUSB1',115200)   #establish serial communication with arduino's port
time.sleep(5)                               #wait time to ensure serial communication setup
TIME = 0
fluorescence_threshold = 25.0      #minimum difference between control and final value
light_sensor1 = ""
light_sensor2 = ""
light_sensor3 = ""
light_sensor4 = ""
temperature = ""

control1 = "0.0"     #control measurements
control2 = "0.0"
control3 = "0.0"
control4 = "0.0"
internet = 0

try:

    for i in range(2700000): #repeat until end of reaction (45 minutes)
        #check for internet connection
        ps = subprocess.Popen(['iwgetid'], stdout=subprocess.PIPE, stderr=subprocess.STDOUT)
        try:
            output = subprocess.check_output(('grep', 'ESSID'), stdin=ps.stdout)
            print(output)
```

```
            internet = 1
        except subprocess.CalledProcessError:
            # grep did not match any lines
            print("No wireless networks connected")
            internet = 0
        port.write(str(internet))   #internet status (sent to arduino via the serial monitor)
        port.write('1')             #LED1 control
        port.write('1')             #LED2 control

        TIME = port.readline().strip()
        light_sensor1 = port.readline().strip()
        light_sensor2 = port.readline().strip()
        light_sensor3 = port.readline().strip()
        light_sensor4 = port.readline().strip()
        temperature = port.readline().strip()

        print(TIME)
        print(light_sensor1)
        print(light_sensor2)
        print(light_sensor3)
        print(light_sensor4)
        print(temperature)

            dataline = str((int(TIME)/60000)) + "," + light_sensor1 + "," + light_sensor2 + "," +
light_sensor3 + "," + light_sensor4 + "," + temperature +"\n"
        myfile.write(dataline)
        time.sleep(1)

        if TIME > 300000 and TIME < 360000:  #read control values between the 5th and 6th minute
            control1 = light_sensor1
            control2 = light_sensor2
            control3 = light_sensor3
            control4 = light_sensor4

except:
    print("closing port...")
    port.close()
```

```
plague_present = float(light_sensor1) - float(control1)  #get difference between control and final fluorescence values
hepatitis_present = float(light_sensor2) - float(control2)
whoopingCough_present = float(light_sensor3) - float(control3)
tuberculosis_present = float(light_sensor4) - float(control4)

if plague_present > fluorescence_threshold:
    plague_present = 1
else:
    plague_present = 0
###
if hepatitis_present > fluorescence_threshold:
    hepatitis_present = 1
else:
    hepatitis_present = 0
###
if whoopingCough_present > fluorescence_threshold:
    whoopingCough_present = 1
else:
    whoopingCough_present = 0
###
if tuberculosis_present > fluorescence_threshold:
    tuberculosis_present = 1
else:
    tuberculosis_present = 0



apikey = '1234'
identifier = 'N19398567'
airportCode = 'MCT'
plague = str(plague_present)
hepatitisB = str(hepatitis_present)
whoopingCough = str(whoopingCough_present)
tuberculosis = str(tuberculosis_present)
malaria = '0'
nationality = 'NI'
```

```
fromAirportCode = 'JFK'
toAirportCode = 'MCT'


data = {
'apikey': apikey,
'malaria': malaria,
'identifier': identifier,
'airportCode': airportCode,
'hepatitisB': hepatitisB,
'whoopingCough': whoopingCough,
'tuberculosis': tuberculosis,
'plague': plague,
'nationality': nationality,
'fromAirportCode': fromAirportCode,
'toAirportCode': toAirportCode
}
response = requests.post('https://igem-nyuad-api.herokuapp.com/request', data=data) #send data
to database through WEB API

myfile.close()
```

ARDUINO:

```
/*
 * code by Jimoh Yusuf Omotayo
 */

//##############################################################################
##################################################
const int TEMP_SENSOR_PIN = A0;                  //the analog pin the TMP36's Vout (sense) pin is
connected to A0
const int TRANSISTOR_PIN = 13;
const int LED1 = 2;
const int LED2 = 3;
```

```
const int latchPin = 8;                    //Shift-out register connection pins
const int ClockPin = 12;
const int DataPin = 11;

#define LIGHTSENSOR_PIN1 A1                    //Ambient light sensors pins
#define LIGHTSENSOR_PIN2 A2
#define LIGHTSENSOR_PIN3 A3
#define LIGHTSENSOR_PIN4 A4

String check_start;             // Variable to check for start of data transmission
char internett;                 //variable to receive internet connection status from Raspberry Pi
char led1_data;
char led2_data;
unsigned long starting_time = 0;  //starting time of reaction
unsigned long present_time = 0;   //current time of reaction
unsigned long time_lapsed = 0;    //time elapsed since the beginning of reaction
int starter = 0;
float temperatureC;             //chip temperature
int working = 1;
int internet = 0;

float sensor1;
float sensor2;
float sensor3;
float sensor4;


//################################################################################
############################################################
void setup()
{
  pinMode(TRANSISTOR_PIN, OUTPUT);
  pinMode(LED1,OUTPUT);
  pinMode(LED2,OUTPUT);
  pinMode(TEMP_SENSOR_PIN,INPUT);
  pinMode(LIGHTSENSOR_PIN1,INPUT);
```

```
 pinMode(LIGHTSENSOR_PIN2,INPUT);
 pinMode(LIGHTSENSOR_PIN3,INPUT);
 pinMode(LIGHTSENSOR_PIN4,INPUT);
 pinMode(latchPin, OUTPUT);
 pinMode(ClockPin, OUTPUT);
 pinMode(DataPin, OUTPUT);
 Serial.begin(115200);                          //Start the serial connection with the computer

}


void loop()
{
 //INNER LED
 //digitalWrite(LED1,HIGH);
 //digitalWrite(LED2,HIGH);

 //TEMPERATURE SENSOR
  int temp = analogRead(TEMP_SENSOR_PIN);               //getting the voltage reading from the
temperature sensor
  float voltage = (temp * 5.0)/1024.0;                  // converting that reading to voltage, for 3.3v
arduino use 3.3
  temperatureC = (voltage - 0.5) * 100 ;          //converting from 10 mv per degree with 500 mV
offset

 if (temperatureC > 36)
 {
  digitalWrite(TRANSISTOR_PIN, LOW);
 }
 else if (temperatureC < 36)
 {
  digitalWrite(TRANSISTOR_PIN, HIGH);
 }

 //LIGHT SENSORS
 sensor1 = analogRead(LIGHTSENSOR_PIN1); //Read light level
 sensor2 = analogRead(LIGHTSENSOR_PIN2);
```

```
 sensor3 = analogRead(LIGHTSENSOR_PIN3);
 sensor4 = analogRead(LIGHTSENSOR_PIN4);

 float square_ratio1 = sensor1 / 1023.0;      //Get percent of maximum value (1023)
 float square_ratio2 = sensor2 / 1023.0;      //Get percent of maximum value (1023)
 float square_ratio3 = sensor3 / 1023.0;      //Get percent of maximum value (1023)
 float square_ratio4 = sensor4 / 1023.0;      //Get percent of maximum value (1023)

 square_ratio1 = pow(square_ratio1, 2.0);     //Square to make response more obvious
 square_ratio2 = pow(square_ratio2, 2.0);
 square_ratio3 = pow(square_ratio3, 2.0);
 square_ratio4 = pow(square_ratio4, 2.0);

 //Transmit to and receive data from Raspberry pi
 transmit_and_receive_data();

 //Control Indicator Lights
 lights();

 delay(1000);                            //waiting a second
}
//############################################################################
############################################################

//############################################################################
############################################################
//FUNCTIONS
/*
 * this function receives and transmits data to the Raspberry Pi using serial connection
 */
void transmit_and_receive_data()
{
 if (Serial.available() > 0)
 {

   if (starter == 0)
   {
```

```
    starting_time = millis();
    working = 1;
    starter = 1;
   }

   present_time = millis();
   internett = Serial.read();
   led1_data = Serial.read();
   led2_data = Serial.read();
   time_lapsed = present_time-starting_time;
   Serial.println(time_lapsed);
   Serial.println(sensor1);
   Serial.println(sensor2);
   Serial.println(sensor3);
   Serial.println(sensor4);
   Serial.println(temperatureC);

   int led1 = led1_data-'0';
   int led2 = led2_data - '0';
   digitalWrite(LED1,led1);
   digitalWrite(LED2,led2);
  }

}

/*
 * This function controls the indication lights of the device
 */
void lights()
{
 startup();
 if (time_lapsed < 36000)
 {
  reaction_running();
 }
 else
 {
```

```
    reaction_done();
  }
}

/*
 * This function latches a high(1) to the current pin of the Shift-out register
 */
 void latch_data_on()
{
  digitalWrite(ClockPin, 0);

  //Sets the pin to HIGH or LOW depending on pinState
  digitalWrite(DataPin, 1);
  //register shifts bits on upstroke of clock pin
  digitalWrite(ClockPin, 1);
  //zero the data pin after shift to prevent bleed through
  digitalWrite(DataPin, 0);
}

/*
 * This function latches a low(0) to the current pin of the Shift-out register
 */
void latch_data_off()
{
  digitalWrite(ClockPin, 0);

  //Sets the pin to HIGH or LOW depending on pinState
  digitalWrite(DataPin, 0);
  //register shifts bits on upstroke of clock pin
  digitalWrite(ClockPin, 1);
  //zero the data pin after shift to prevent bleed through
  digitalWrite(DataPin, 0);
}

/*
 * This function latches the stored in the latch register
 * into the output pins of the shift-out register
```

```
 */
void clock_and_latch()
{
  digitalWrite(ClockPin, 0);
  digitalWrite(latchPin, 1);
}

/*
 * This function controls the state of each pin
 * i.e it takes in pins (of the shift-out register) that
 * should be turned high as argument and turns them high
 */
void on(int a=100, int b=100, int c=100, int d=100, int e = 100 )
{
  //ground latchPin and hold low for as long as you are transmitting
  digitalWrite(latchPin, 0);
  for (int i=1; i<=16; i++)
  {
     if (i == a or i == b or i == c or i == d or i == e)
     {
       latch_data_on();
     }
     else
     {
       latch_data_off();
     }
   }
   clock_and_latch();
}

/*
 * This function turns all the leds off
 * i.e it turns all the shift-out registers pins low
 */
void off()
{
  //ground latchPin and hold low for as long as you are transmitting
```

```
  digitalWrite(latchPin, 0);
  for (int i=1; i<=16; i++)
  {
     latch_data_off();

   }
   clock_and_latch();
}

/*
 * THis function controls the start up blink of the device
 */
void start_blink()
{
  for(int i=0;i<3;i++)
  {
    on(2,5,8,11,14);
    delay(500);
    off();
    delay(500);
  }
}

/*
 * This function controls the LED that shows when the device is available
 */
void device_working()
{
  if (working == 1)
  {
    on(13);
  }
  else
  {
    on(15);
  }
}
```

```
/*
 * This function controls the LED that indicates whether the
 * Raspberry pi is connected to the internet or not...
 */
void is_internet()
{
  if (internet == 1)
  {
    on(13,10);
  }
  else
  {
    on(13,12);
  }
}

void startup()
{
  if (working == 1)
  {
    start_blink();
    device_working();
    delay(1000);
    is_internet();
    delay(1000);
    working = 0;
  }
}

/*
 * This function controls the colors of the indication lights when
 * the reaction is going on...
 */
void reaction_running()
{
  if (internet == 1)
```

```
 {
   on(13,10,2,5,8);
   delay(1000);
   on(13,10);
  // delay(1000);
 }
 else
 {
   on(13,12,2,5,8);
   delay(1000);
   on(13,12);
   //delay(1000);
 }
}

/*
 * This function controls the colors of the indication lights when
 * the reaction is done...
 */
void reaction_done()
{
 if (internet == 1)
 {
   on(13,10,1,4,7);
   delay(1000);
   on(13,10);
   //delay(1000);
 }
 else
 {
   on(13,12,1,4,7);
   delay(1000);
   on(13,12);
   //delay(1000);
 }
}
```

```
//#########################################################################################
#############################################################
```