Exploring Fabrication Principles for Making Freeform BreadBoards

by Lotta-Gili Blumberg

S.B., EECS, Massachusetts Institute of Technology (2018)
S.B., Physics, Massachusetts Institute of Technology (2018)

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Computer Science and Engineering

at the

Massachusetts Institute of Technology

June 2019

Author: ____
Department of Electrical Engineering and Computer Science
May 21, 2019

Certified by: ____
Stefanie Mueller, Professor of Computer Science and Engineering
Thesis Supervisor
May 21, 2019

Accepted by: ____
Katrina LaCurts, Chair, Masters of Engineering Thesis Committee

Exploring Fabrication Principles for Making Freeform BreadBoards

by Lotta-Gili Blumberg

Thesis Supervisor: Stefanie Mueller, Professor of Computer Science and Engineering

Submitted to the Department of Electrical Engineering and Computer Science

June 7, 2019

In Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Computer Science and Engineering

ABSTRACT

CurveBoards are 3D breadboards integrated into the surface of physical prototypes. CurveBoards offer both the flexibility of breadboards, i.e., the ability to add or remove components, while also integrating well with the shape of the prototype. Thus, designers can use CurveBoards to test function directly in context of the actual physical form.

We demonstrate our method for fabricating the CurveBoard in which designers only have to 3D print the housing and then fill the wire channels with conductive silicone. We also discuss limitations to this approach and alternative fabrication methods. We display a range of different application scenarios and report on a user study with six participants that showed that prototypes created as CurveBoards looked and felt closer to the final design.

ACKNOWLEDGEMENTS

First, I would like to extend a huge thank you to my advisor, Professor Stefanie Mueller. She has been incredibly supportive and has offered me much guidance and instruction along the way. I would also like to express my sincere gratitude to my collaborators Junyi Zhu, Martin Nisser, Yunyi Zhu, Xin Wen, and Kevin Shum. Junyi in particular has been a great role model and friend. Together, Junyi, Yunyi, and Martin were the masterminds for the software portion of this project. Junyi and Yunyi also helped me test various fabrication methods, and Junyi and Martin helped run the user study. Xin and Kevin helped design the prototypes for this project, with Kevin on the bracelet and headphones and Xin on the helmet and frisbee. Over all, Professor Mueller, Junyi, Martin, Yunyi, Xin, and Kevin have been fantastic to work with and I could not have asked for a better group. This project would not have been possible with this amazing team.

I would also like to thank all of my mentors that helped me brainstorm various fabrication methods and designs, including Isabel Qamar our expert on carbon fibers and silicone, Michael Wessely our expert on alternative methods for fabricating circuits, Chris Haynes at the International Design Center, Joe Steinmeyer, and Ron Wiken at the CSAIL machine shop.

Lastly, I would like to thank everyone else in the Human Computer Interactions Engineering Group for creating such a positive and creative environment. The group has been incredibly supportive and welcoming, and the people have provided an energy that made me excited to go to lab.

PUBLICATIONS

This MEng thesis contributed the hardware and material development to the following publication currently under submission:

Junyi Zhu, Lotta-Gili Blumberg, Martin Nisser, Yunyi Zhu, Xin Wen, Kevin Shum, Stefanie Mueller.
CurveBoards: 3D Breadboards for Prototyping Function in the Context of Physical Form.
In submission to ACM UIST 2019.

LANGUAGE AND FIGURE NOTE

Throughout this thesis, I use the word “we” to signify that this work has been a group effort. However, the work that I provide in detail, unless stated otherwise, is my own. Photographs were taken jointly by the project team.

Table of Contents

List of Figures

1. Introduction

Breadboarding on solderless breadboards is a popular approach to prototyping because it allows users to easily plug and unplug electronic components for testing. A large part of the appeal of breadboards is how easy it is to experiment with various circuits layouts and components [17].

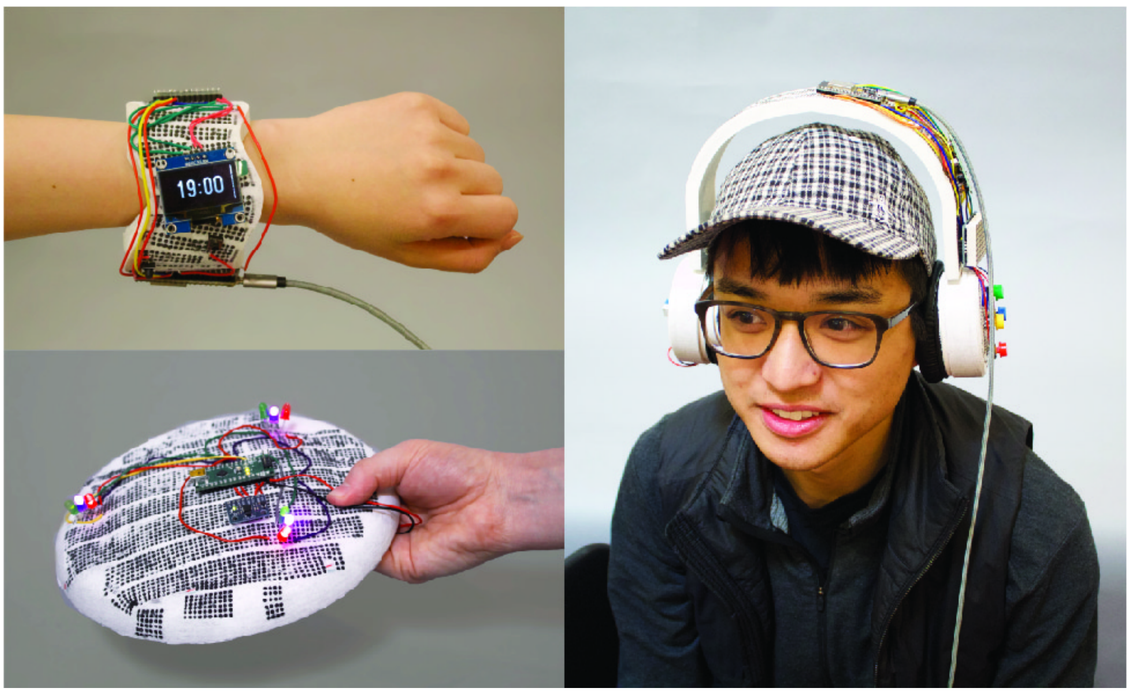


Figure 1. CurveBoards are 3D breadboards directly integrated into the surface of a physical prototype. Since CurveBoards offer both the flexibility of breadboards, i.e. the ability to plug and unplug components, while also integrating well with the shape of the prototype, they enable designers to test functionality in the context of the actual physical form.

However, while breadboarding is useful for testing and iterating on the functionality of circuits such as making sure a button or screen work as it should, breadboards are not ideal in the next stage of prototyping; it is difficult to integrate a circuit on a breadboard into the actual physical form of a prototype. The easiest method is to simply stick the breadboards onto the prototype, but often the blocky standardized shapes of breadboards will not integrate well onto a physical prototype (Figure 2).

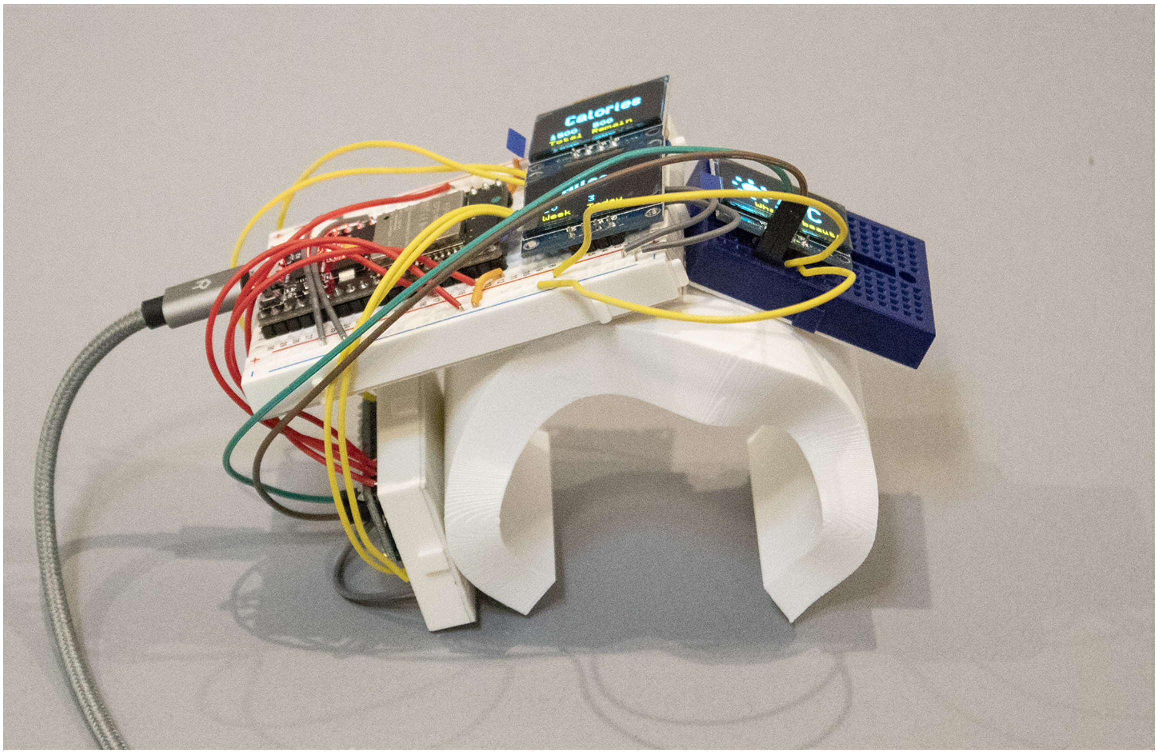


Figure 2. Attempt at integrating breadboards with the physical prototype of an interactive bracelet. Result by P1 from our user study.

For more expert designers, another approach is to move the circuit onto either a protoboard, which requires soldering, or custom PCB (printed circuit board). However, this process can be somewhat time-consuming making iterating on the design a slow process. Additionally, flaws in a design might not reveal themselves to a designer until after integrating the electronics with the prototype itself. For example, they might notice that a button is hard to reach due to moving parts or that a certain display is difficult to see when it is close to a light source. The designers would want a quick way to swap or move components. If they were using a protoboard or custom PCB at this stage in the design process, the designers would have to spend a lot of time either resoldering or redesigning a board in order to iterate on their design.

1.1 Contribution

Our solution is a CurveBoard: a 3D breadboard that is directly integrated into the physical form of a prototype (Figure 1). CurveBoards offer the flexibility of a breadboard, allowing a designer to easily add, remove, and swap components through the iterative design process, while also giving the designer the context they need to see how the electronics will integrate into their final product or prototype.

In this thesis, we will propose a simple design process using off-the-shelf materials for fabricating a CurveBoard. Our process, overviewed in Section 3 and described in more detail in Section 4, uses a regular 3D printer for printing a frame and conductive silicone for creating conductive circuit connections. The process only requires a few fabrication steps: 3D printing and 15-30 minutes for filling the channels. We will also discuss alternative approaches for building a CurveBoard such as pure silicone fabrication, generating 2D conductive traces for transfer onto a 3D object, and pure 3D printing in Section 5.

In Section 6, we will provide additional detailed scenarios where a CurveBoard is useful during the prototyping process. These examples include a variety of use-cases including a helmet, a frisbee, and a pair of headphones. Of course, these are only a few examples while our pipeline should be usable for many more types of prototypes.

Lastly, we will provide data on our user study in Section 7. Our user study showed that CurveBoards are intuitive and easy to use, and that they can be readily used in today's prototyping workflow.

2. Related Work

Our work is related to research projects that (1) integrate form and function in early stage prototyping, (2) extend standard breadboard use, (3) have unusual breadboard shapes, and (4) investigate new approaches to fabricating with conductive materials.

2.1 Integrating Form & Function in Early Stage Design

For electronics testing, breadboards are the go-to tool for prototyping because of their fast iteration cycle [13]. It is easy to quickly change designs and test them, making them an integral part of the design process for designing interactive prototypes [20]. However, today's breadboards are also quite limiting because of their fixed size, so they are not always easy to integrate into a physical prototype.

Many HCI papers such as DTools emphasize the importance of integrated prototypes as they allow for a faster design-test-redesign cycle. DTools [8] is an example of a tool that helps close the loop and improve the design workflow in an integration test environment with hardware interactions. It does this through software which enables you to easily visualize the hardware data during testing to better gauge an understanding of how to iterate on a design. DTools is able to improve the design workflow by integrating the test environment used for hardware testing with the prototyping capabilities.

Similar to DTools, our goal with CurveBoards is to integrate to stages in the early design process, however, our aim is to integrate a completely different set of steps: the steps of testing electronics on a breadboard and placing components onto the prototype.

2.2 Extensions to Standard Breadboard Use

In HCI there has been a recent upsurge in papers focusing on extending the capabilities of breadboards. Visible Breadboard [15] allows a user to visualize the flow of current in a breadboard. However, it is not cohesively integrated as it requires a separate touch interface. Toastboard [6] and CurrentViz [28] integrated the visuals and breadboard together by displaying voltage and current information on the breadboard, and Circuit-Sense [29] added a feature for recognizing the component plugged into a breadboard. A different functional extension to breadboards, CircuitStack [26] allows users to plug in their components wirelessly. It works by having the wired connections internal to the breadboard, eliminating the potential for false wiring so long as the component is placed in the correct location. While the main contribution of this thesis is the fabrication of novel form factors for breadboards, it is worth noting that as future work these features could potentially be integrated with our CurveBoards.

2.3 Unusual Breadboard Shapes

Not many projects have focused on changing the shape of a breadboard to better approximate what it is prototyping. BitBlox [4] come somewhat close. They are breadboards shaped like 2D puzzle pieces. Some microcontrollers also take unusual form factors in order to allow for better design and testing, however, these are largely one-offs for one particular use-case and do not offer much flexibility. For example, CARduino [23] is designed specifically for a car interior. LilyPad Arduino [11] is designed for wearables/textiles. While these do allow for better prototyping in the context of a physical form, these microcontrollers are made by experts for only one specific use.

There are somewhat more examples of protoboards of various shapes. Many of them are simply 2D shapes such as Sparkfun's penta board [21] or hex board [22] or the circular Lilypad protoboard [12]. The Dodecahedral Protoboard [5] is the only 3D protoboard we found, and it is built out of the pentaboards. There are also some flexible protoboards which could potentially wrap a surface. However, all these protoboards have the limitation that they require soldering, making it much more difficult to adapt the prototypes and make quick changes.

Using neither a breadboard nor a protoboard, VoodooIO [24] is a fabric substrate that allows users to rearrange components such as buttons freely. However, this interface is largely restrictive as it uses an active communication network. The system only works with specialized components as opposed to general off-the-shelf components, so it too is a very restrictive system for general prototyping.

Our CurveBoard, for comparison, is able to take on the shape of many different objects and can use many of the same common electrical components used in today in breadboarding.

2.4 Fabricating 3D Objects with Conductive Materials

To find the best fabrication method for CurveBoards, we investigated a range of different approaches since fabricating 3D circuits is still difficult today [1].

One approach is to 3D print the circuit using conductive filament as illustrated in PrintPut [2] and Flexibles [19]. However, the high resistance of the filament prevents the resulting objects from going beyond simple LED circuits and capacitive touch sensors. Low-resistance solutions that 3D print with silver ink are still not at the level of reliable fabrication (i.e., the Voxel 8 [25], the most promising platform in this area, has been discontinued).

To fabricate interactive objects that require higher conductivity, HCI researchers have investigated a variety of methods including drawing with silver ink (Un-Toolkit [14]), using silver inkjet printing (Instant Inkjet Circuits [10]), conductive tape (Midas [18]), screen printing (PrintScreen [16]) as well as hydrographic baths [7].

However, these methods worked less well for fabricating CurveBoards since the resulting conductive patches did not make good electric contact with the components that are being plugged and unplugged (more details in Section 4, "Fabrication"). Inspired by a DIY project on rubber breadboards [3], our method instead uses conductive silicone that is low resistance, elastic (allows to plug and unplug components repetitively while maintaining steady electrical connection), and can be made using readily available off-the-shelf materials and fabrication equipment.

3. Our System: CurveBoard Pipeline

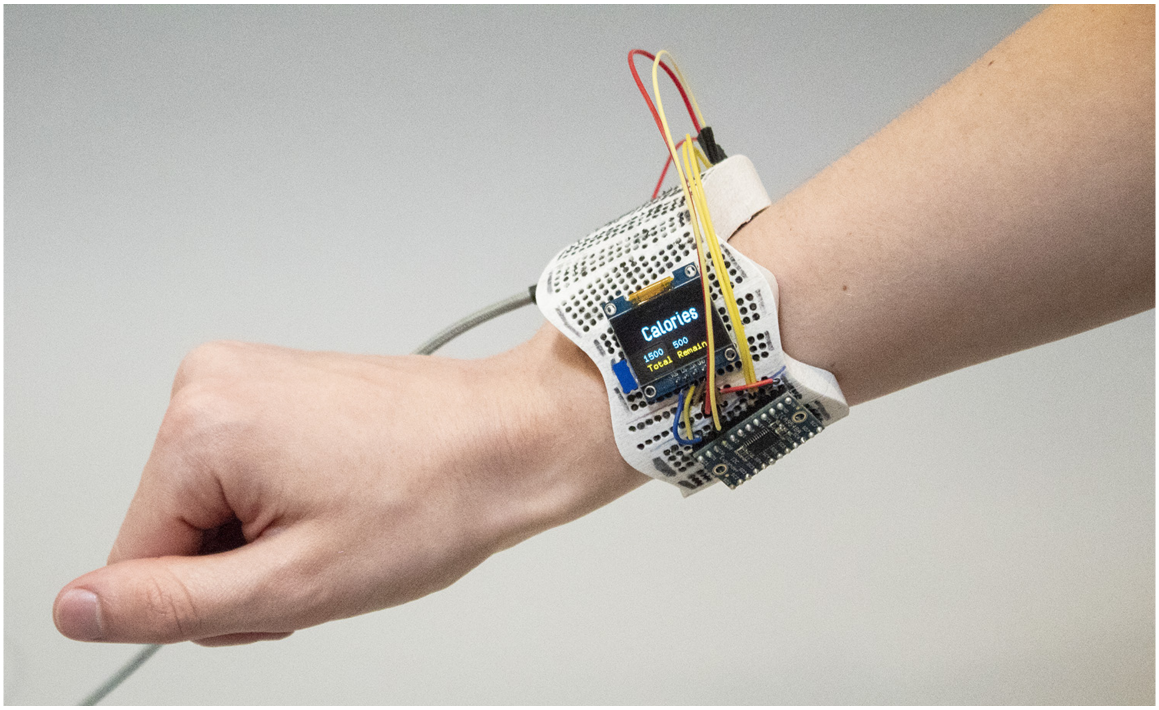


Figure 3. CurveBoard of an interactive health tracking bracelet with display, buttons, and microcontroller.

In this section, we give an overview of how we go from an object design to a CurveBoard that we use for prototyping such as the one in figure 3. While we will describe the overall pipeline for context, for the purpose of this thesis, we will be focusing on the hardware part of the pipeline, and the software is covered in another thesis.

3.1 Step 1: Converting the 3D Model into a CurveBoard

We designed software to allow a user to generate their own CurveBoard. The user is able to upload their own 3D design into the software. It automatically generates a set of points for pinholes (holes where eventually wires can get plugged into). The user can then add their channel layout (Figure 4). Each channel represents what will eventually be electrically connected. The software makes it easy to design a layout similar to that of a standard breadboard with many short channels and two long channels as power rails (one for power and one for ground).

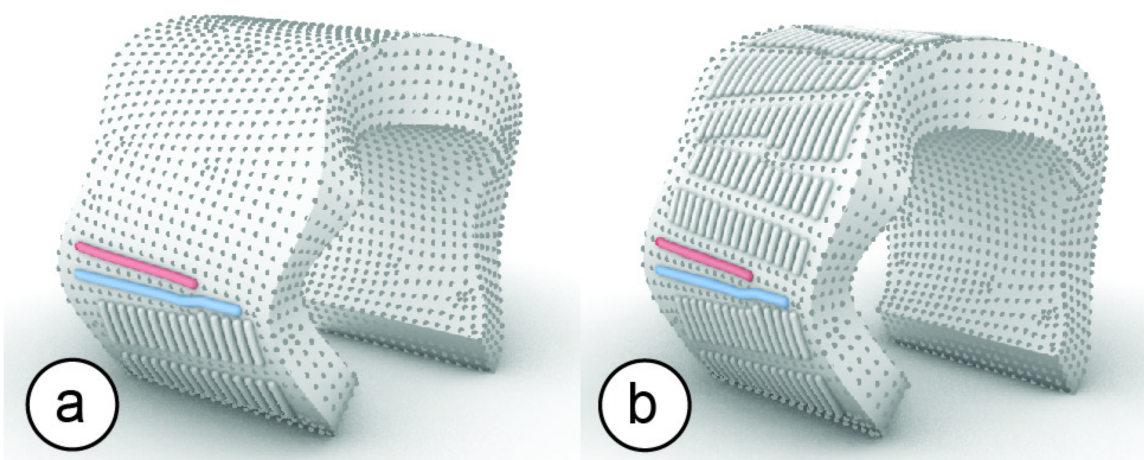


Figure 4. Laying out channels in the CurveBoard software

3.2 Step 2: Export & Fabrication

Next, we can export our design to generate an STL file that includes our pinholes and channels under the CurveBoard's surface (Figure 5).

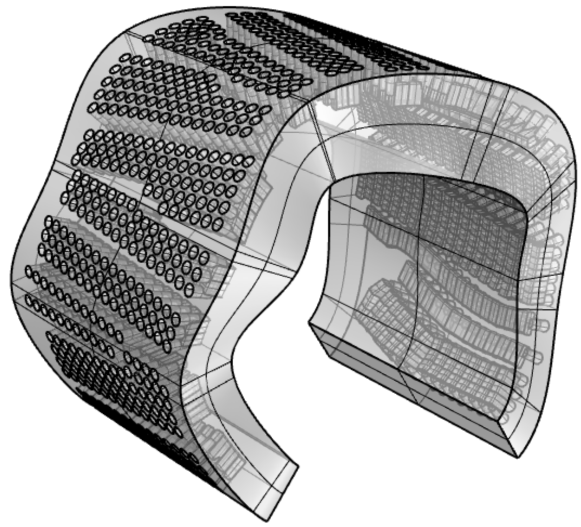


Figure 5. Generated CurveBoard geometry containing the pinholes and channels underneath the surface.

Then, we use our 3D printer (Ultimaker 3) to fabricate the housing of the CurveBoard (Figure 6).

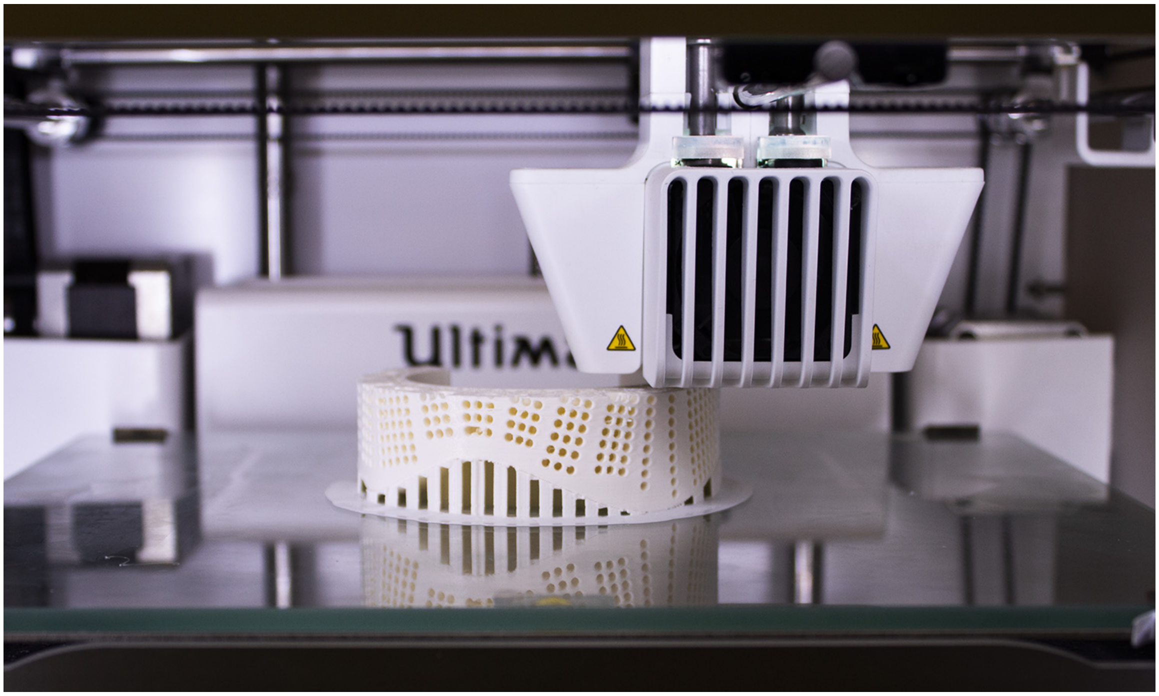


Figure 6. 3D printing the CurveBoard housing.

While the housing is printing, we prepare the conductive silicone for filling the channels (see Section 4.1 for more details). As can be seen in Figure 7, after the housing finished printing, we fill the channels with the silicone using a syringe (ca. 15 min for this CurveBoard with 208 channels). After the silicone cures (between 1-4 hours depending on the type of silicone), we clean up the residue from the surface and the CurveBoard is ready to be used.

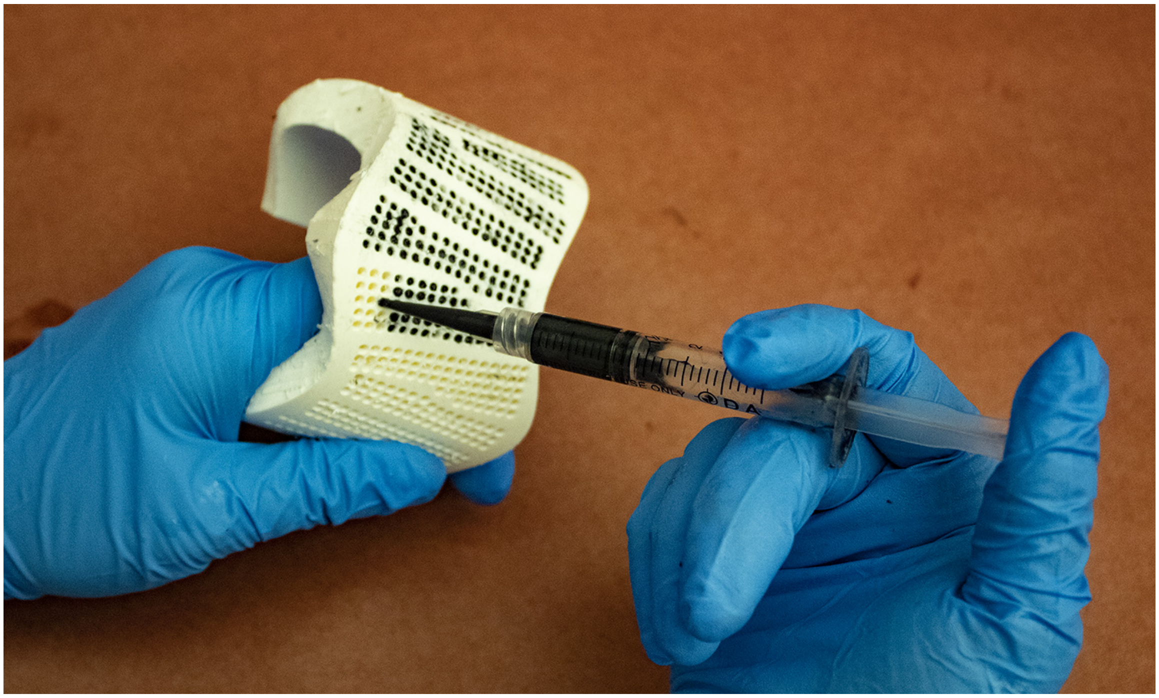


Figure 7. Filling the hollow channels of the CurveBoard with conductive silicone.

3.3 Step 3: Exploring the Placement of Components

Now we walk through a sample scenario where we use the CurveBoard we just made to explore the layout of an interactive bracelet (Figure 8). This bracelet uses an ESP32 microcontroller, an OLED display, two buttons, and a handful of wires.

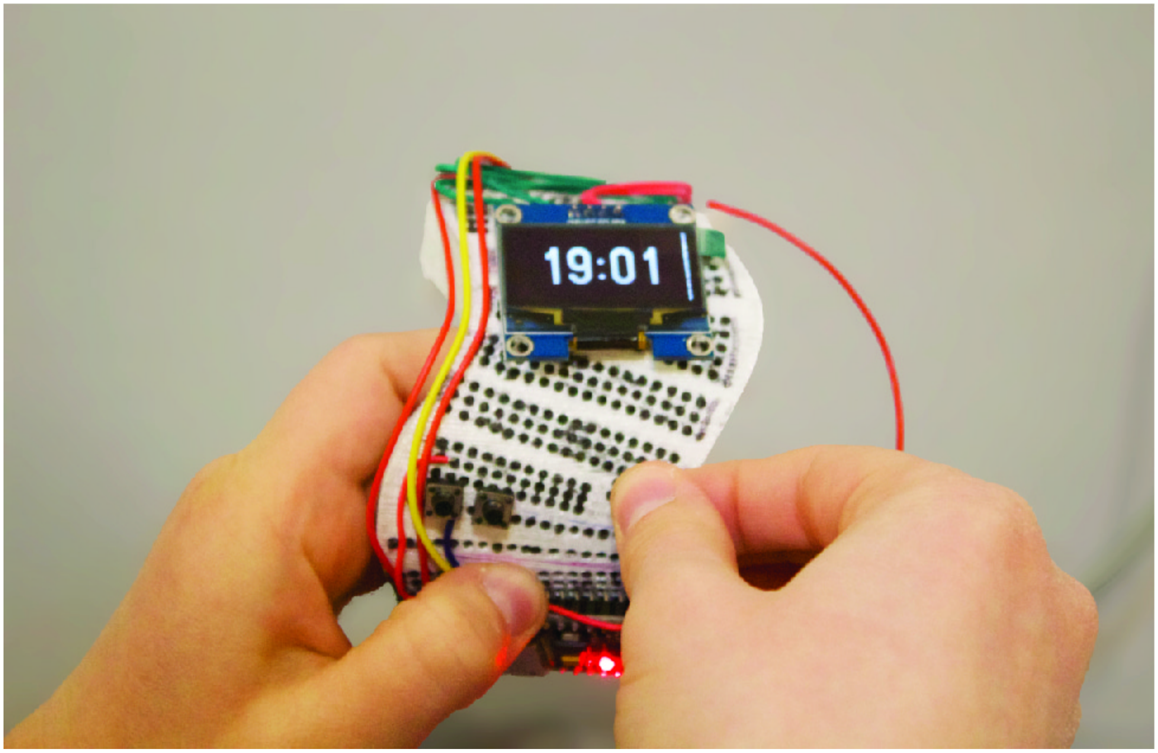


Figure 8. Wiring up the initial version of our bracelet.

We first place the microcontroller on the outer end of the CurveBoard since we want to save the middle as our interaction space, the area with our inputs (buttons) and outputs (display). Next, we add a display and after exploring different sizes of displays and different locations on the bracelet, decide to position it in the lower half. Finally, we add two buttons for controlling what the display shows.

After testing the prototype, we notice that moving the button closer to the display would improve usability. We therefore change the position of the button and the corresponding wiring, i.e., we unplug the button and plug it below the display (Figure 9).

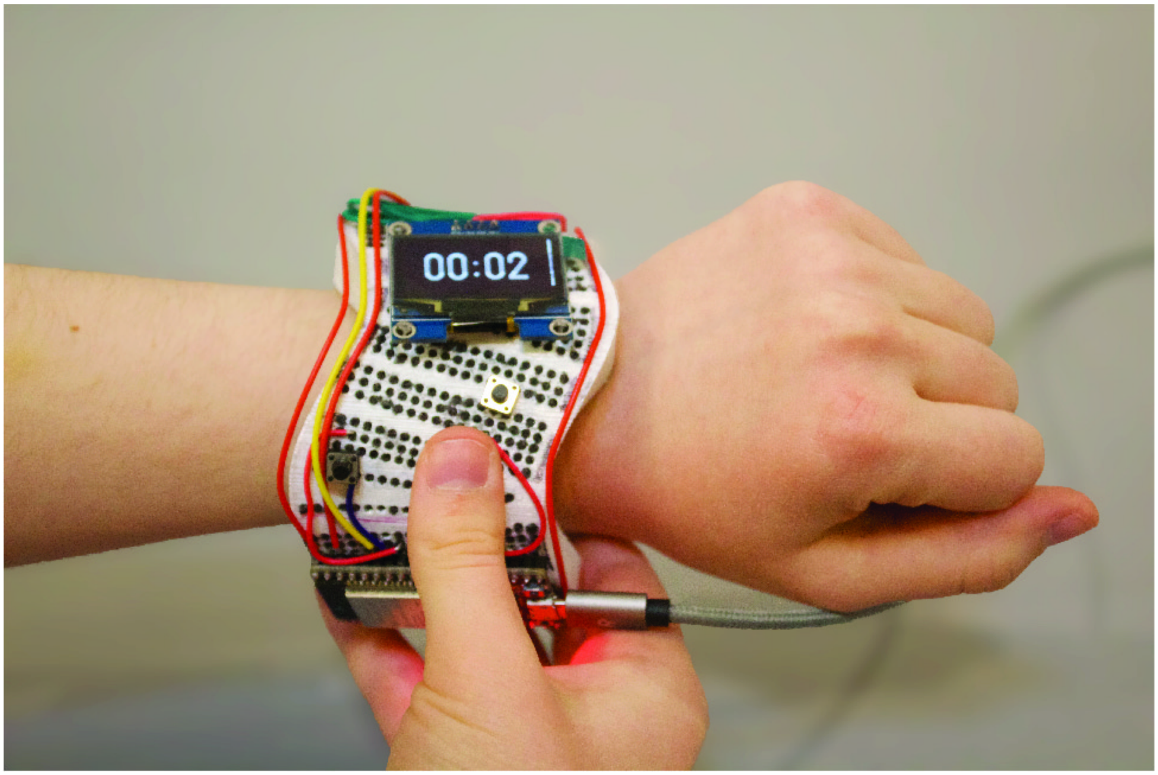


Figure 9. Testing different button-display configurations for our interactive bracelet.

4. Fabrication Details

Inspired by a DIY project on rubber breadboards [3], we developed a fabrication method that enables both high conductivity and fast fabrication with reasonably low manual effort. While the rubber breadboards [3] are made from both non-conductive silicone (housing) and conductive silicone (channels) through silicone casting, we decided to instead 3D print the housing and then fill its hollow channels with conductive silicone after the print had finished. This enabled us to have a fabrication method that is faster than the rubber breadboards, which required multiple rounds of curing, while also being higher in fidelity.

4.1 Mixing the Conductive Silicone

Next, we provide more information on the mixing procedure for the conductive silicone.

Step-by-Step Walkthrough

Preparing the Conductive Silicone (Carbon Fibre + Silicone Part A): To create the conductive silicone, we first mix a small spoon of chopped carbon fiber (0.7mm long, 7 μm diameter from Procotex) with a splash of isopropyl alcohol. After stirring and dispersing the fiber hairs, we mix them into part A of a regular two-component silicone (type: Smooth-On SORTA-Clear 37 Clear) and stir for 5 minutes (Figure 10a/b). Without part B, the conductive silicone will not start curing, so we can keep the mix of part A + carbon fiber on the shelf to use over the course of several days if stored covered.

After 3D Print Finished (Adding Silicone Part B): Once the 3D printed housing is finished, we can add part B of the silicone to the mix (Figure 10c), which will initiate the curing process. After stirring for 5 minutes, we transfer the conductive silicone to a syringe (3ml, with 16-gauge blunt tip tapered dispensing needle) as can be seen in figure 10d. The syringe can then be used to fill the channels in the CurveBoard.

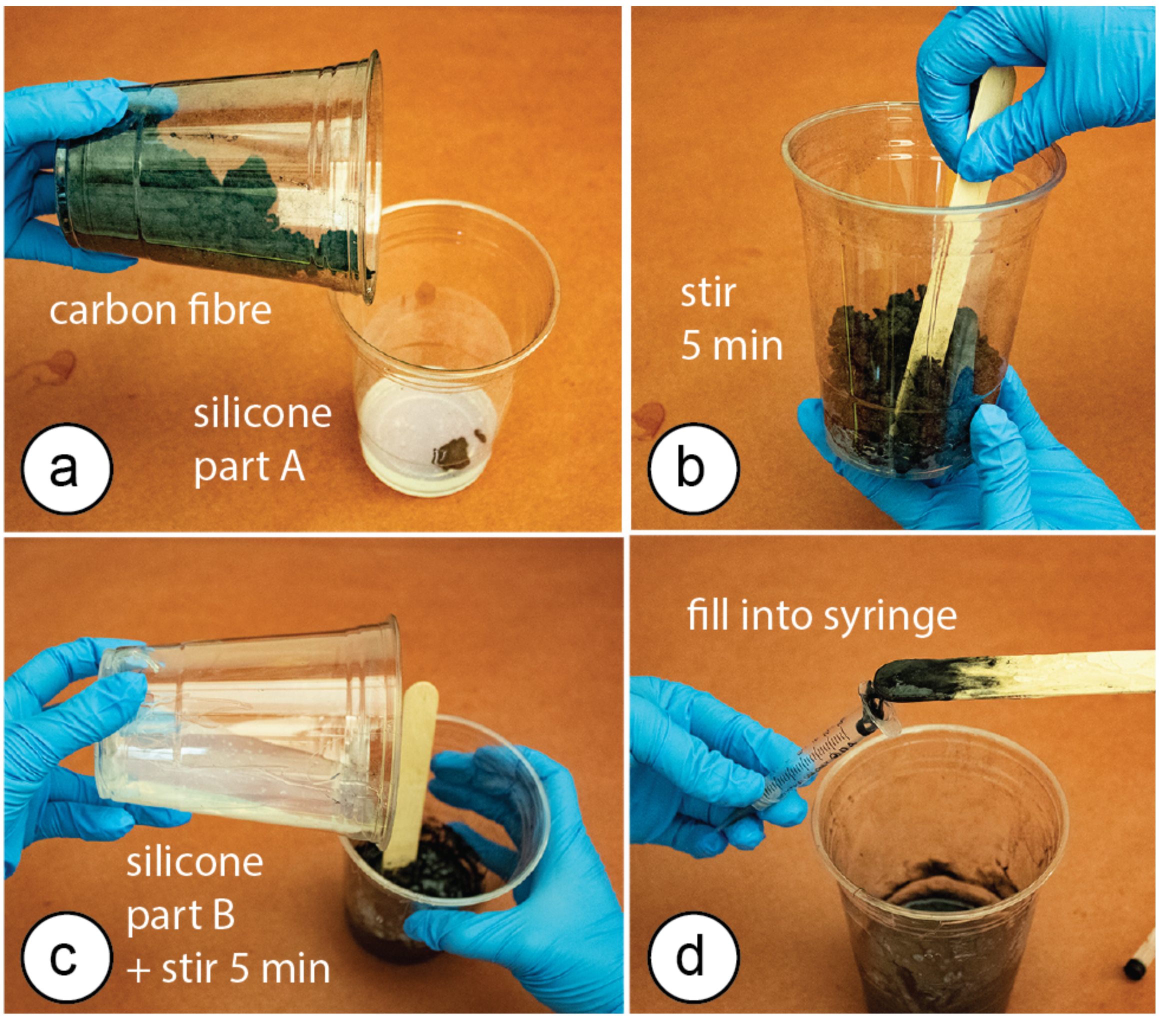


Figure 10. Mixing the silicone.

Waiting for Curing: We experimented with different silicone types. Our fastest curing silicone (Smooth-On OOMOO 25) has a curing time of 75 minutes—however, its light blue color did not look as visually appealing. In general, we found that when mixing carbon fibers with silicone, the clear silicone produced the most aesthetically-pleasing results. For the pictures in this thesis, we thus used the slower curing silicone, which takes 4 hours to cure.

Effects of Different Lengths of Carbon Fibers

Throughout the testing process, we experimented with a variety of lengths of carbon fibers. Our original experimentation was done with 3mm length fibers. While these performed well in out initial testing, we found that when we moved to using a syringe, the fibers would often get jammed in nozzle. For that reason we started using shorter fibers that were .7 mm in length. However, because the fibers are shorter, they require a bit more thorough mixing to guarantee that they are touching each other enough for good conductivity.

Effects of Carbon Fiber to Silicone Ratio

We also experimented with several different ratios between the carbon fiber and the silicone. Unsurprisingly, we found that in general, more carbon fiber meant higher conductivity. However, we also found that adding too much carbon fiber made the mixture difficult to work with. When there is too much fiber, the resulting silicone loses its flexible and rubbery nature. This can cause it to tear or crumble. Furthermore, the mixture is much more difficult to work with. Extruding a syringe filled with a high-density mixture becomes much more difficult, and even just mixing the silicone can be arduous. We therefore settled on a middleground between 5:100 and 10:100 ratios between fibers and silicone, by weight (the 100 includes both the A and B parts). A trick for eye-balling approximately this amount is to mix in as much carbon fiber into the silicone A as possible while still making sure that all of the fibers are well-coated. Then, adding the silicone B substantially thins out the mixture, making it easier to stir and extrude.

4.2 Technical Evaluation

To measure the conductivity and durability of the conductive silicone, we ran two experiments.

Conductivity

To measure the resistance across different channel lengths we fabricate a 16-hole long channel and measured by iteratively increasing the pinhole distance between the measurement points. Figure 11 shows the result. When a line is fitted, we can see that for every extra pin hole, resistance goes up by on average 8.5 ohms.

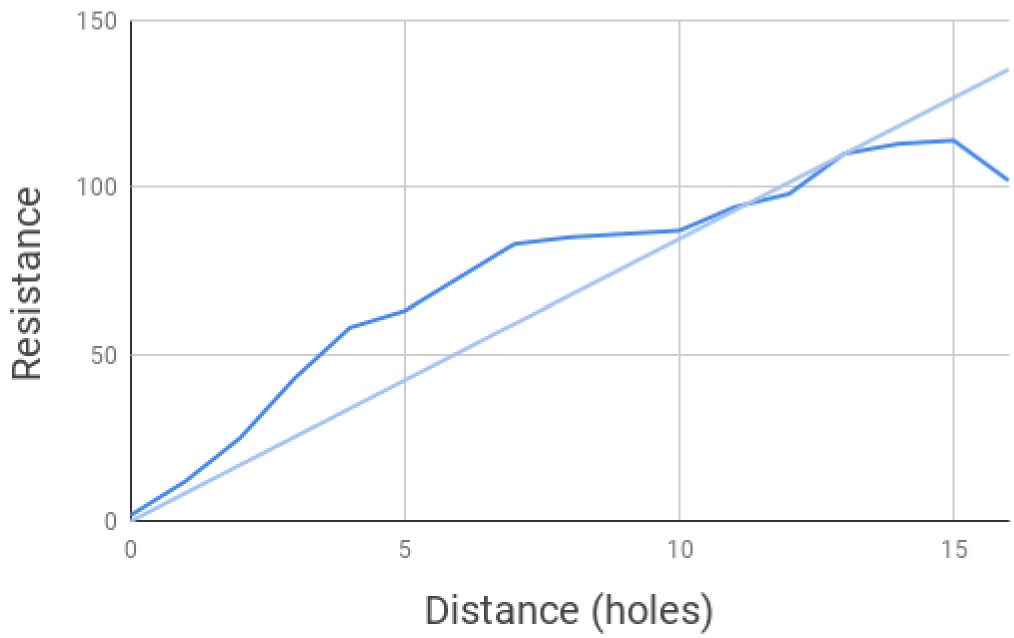


Figure 11. Resistance (ohms) for different channel lengths

This conductivity is good enough to run many prototypes using a microcontroller. Digital signals work well, including PWM (pulse width modulation) signals. Analog components such as potentiometers also work, provided they have a large enough internal resistance. Some analog sensors may or may not provide good readings depending on the internal impedance of those components.

Since the resistance is a function of the distance and the cross sectional area of the channel (R = k*L / A), we can decrease the resistance further by allowing wider channels. However, this would restrict the density of channels on the breadboard, and would prevent the use of some standard components with header pins. Alternatively, we could also make the channels deeper, but that would restrict the types of objects we can create.

Durability

To evaluate the durability of the silicone against repeated plugging and unplugging of components, we selected a CurveBoard channel of three pinhole length and repeatedly inserted a wire into it. We measured the resistance of the channel after every 10 plugs. We found that, surprisingly, over repeated use the resistance decreases (Figure 12). Our best guess is that poking the pinhole over and over actually packs the carbon fibers tighter so that the wire can make a better connection.

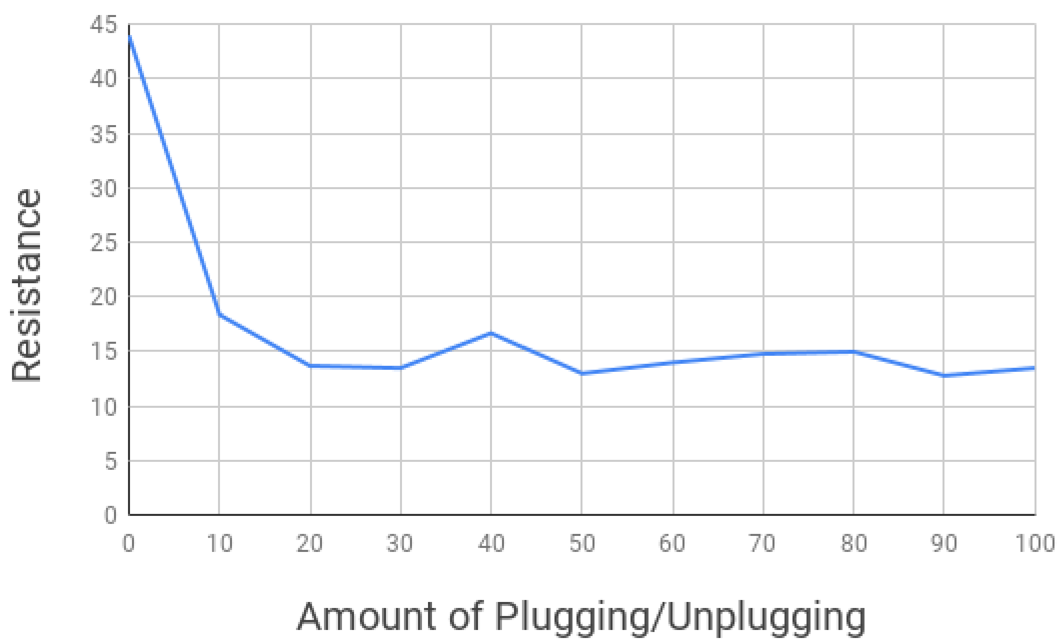


Figure 12. Resistance (ohms) over 100 plugs.

In conclusion, by using a combination of 3D printing (housing) and conductive silicone (channels), we are able to create CurveBoards that can be readily used with today's technology.

5. Alternative Fabrication Methods of CurveBoards

5.1 Requirements for CurveBoard Fabrication

When investigating which fabrication method to use, we had two main requirements:

#1 High conductivity: We wanted to ensure that CurveBoards work with a wide variety of electronic components in use today including common I/O components, such as buttons, knobs, sliders, and displays available from platforms, such as Sparkfun and Adafruit.

#2 Fast Fabrication Using Off-the-Shelf Hardware: We wanted to ensure that CurveBoards can be fabricated in a reasonable amount of time (< 30 min of manual work) while requiring only off-the-shelf materials and standard equipment available to makers.

5.2 Silicone Only Method

One of the first CurveBoards that we fabricated was made entirely out of silicone, some conductive and some non-conductive (Figure 13).

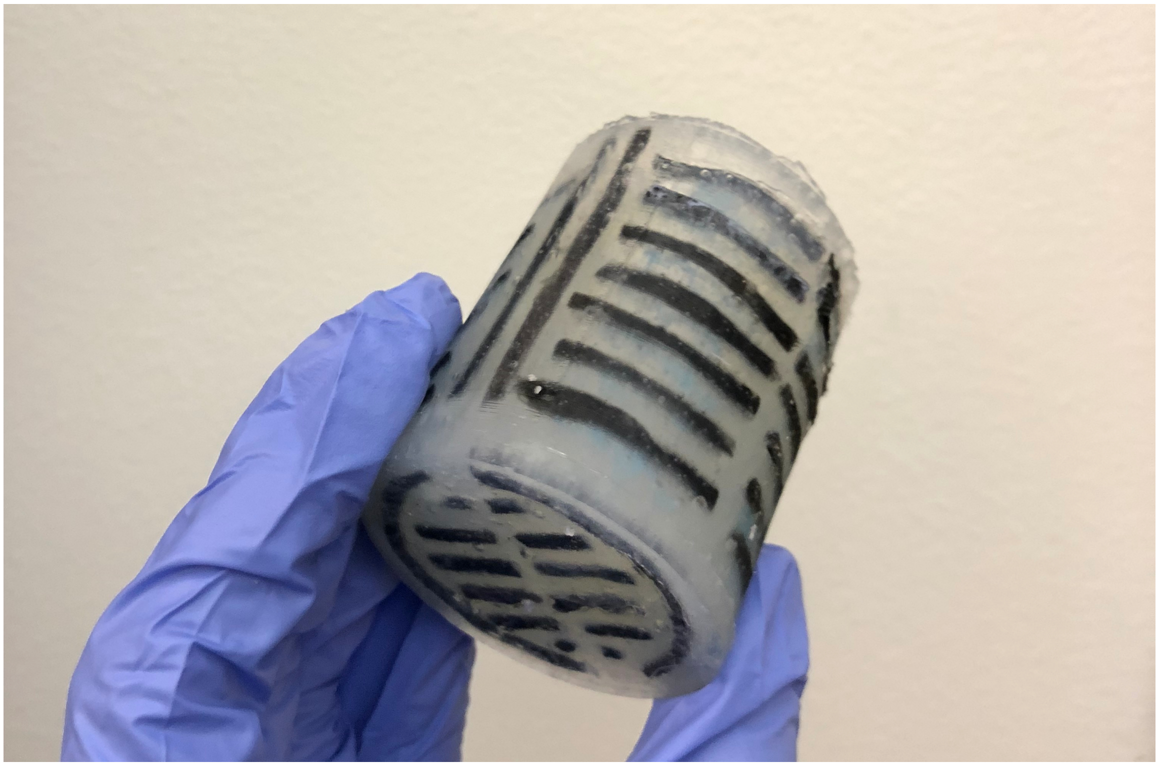


Figure 13. A Curveboard made with silicone

This type of CurveBoard has many great properties. The fact that it was purely made of silicone gives it self-healing properties, meaning that one could plug and unplug a wire many times over without causing the object to break or fall apart. It is also rather flexible and even water-proof. Additionally, since there are no predefined holes, electronics maybe be plugged in with many more options than with a standard breadboard. These properties could allow to be used for many more applications such as wearables or remote-controlled boats.

However, fabrication of this CurveBoard takes a long time and lots of manual work. This is because in order to get the interior conductive channels in place, it requires 2 iterations of curing. The fabrications steps are as follows:

1. 3D print two sets of molds (Figure 14)
2. Fill the first mold and wait for it to cure (Figure 14 on the left). This likely takes at least an hour depending on the type of silicone.
3. Add the conductive silicone channels (Figure 15). We assume that these are either bought pre-fabricated or they are made in parallel with step 2.
4. Add to the first cast plus channels using the second mold and wait for it to cure (Figure 14 on the right)

Figure 14. Examples of molds used for fabrication. A single rectangular mold is sufficient for fabricating a flat CurveBoard, but the cylindrical cup requires the two sets of two molds in the back. Additionally, the blue mold needed to be printed in TPU, a flexible material, in order to extract the cured product.

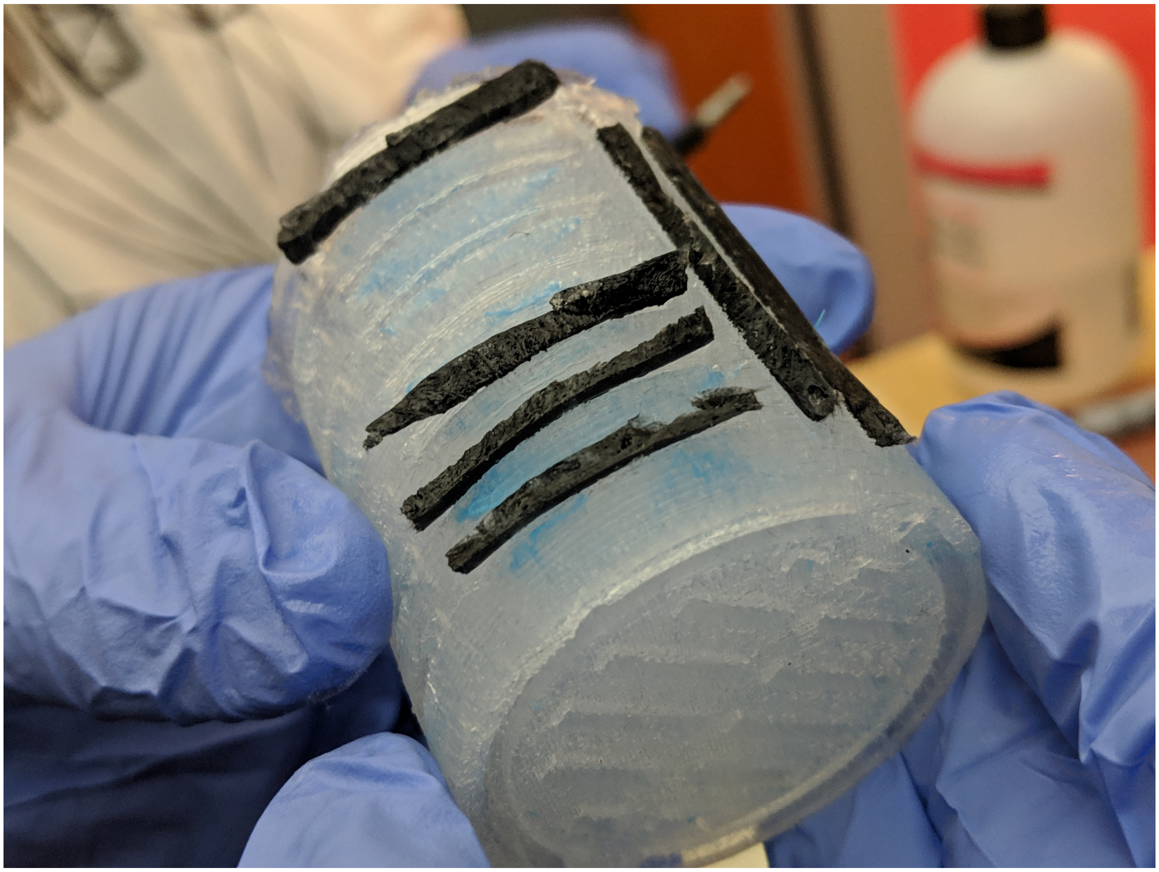


Figure 15. Adding the conductive silicone strips to the first cure of the CurveBoards

Overall, this fabrication method will at a minimum take the amount of time that it takes to 3D print the first mold plus two cure times. The fabrication process can also be somewhat messy and is prone to some error. It also becomes tedious to produce boards with very thin channels since they need to be manually placed.

5.3 Surface Methods

Another approach to building a CurveBoard is to make the conductive connections on the surface of the object as opposed to inside. These methods have a lot of potential since they include methods based on silver ink [10] or conductive tape [18, 30], which are highly conductive.

While these approaches are tempting because they are highly conductive, they have various drawbacks. Attaching these conductive traces can be a laborious task, and it is not easily automated. They require a lot of manual work to either draw [14] or transfer [27] the conductive patches onto the 3D geometry. Using hydrographics [7], the transfer process could be sped up by automation, however, the setup is complex and requires pre-computation of the distorted texture. In addition, it can be difficult with the surface method approaches to get a strong electrical connection between the plugged components and the traces. This is because there is very little surface touching between the wires and the surface trace. The problem can be mitigated by adding rivets (Figure 16), but this further increases the amount of work it takes to assemble a CurveBoard.

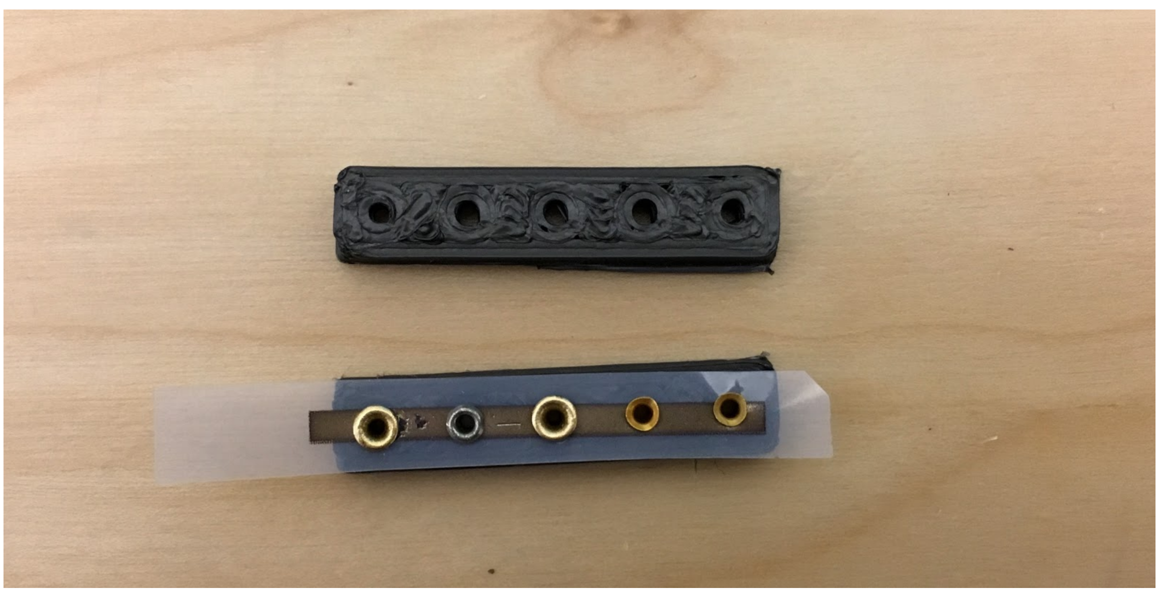


Figure 16. A small example of how one could connect an inkjet-printed trace to a 3D print via rivets

All of these approaches also involve wrapping a two-dimensional object around a three-dimensional object. This means that it is difficult to make them look good when wrapped around any doubly-curved surface, such as a sphere, largely limiting the types of objects that can be supported by this method.

5.4 3D Printing

The ideal fabrication method would not require any manual labor. 3D printing with conductive filament [2, 19] on a dual-material 3D printer uses only standard equipment and requires no manual effort, making it seem like viable solution. With a dual-extrusion printer with one conductive filament and one insulating (non conductive) filament, one could simply print a CurveBoard with minimal effort.

Although we envision that one day, CurveBoards will be built in this way, the technology is unfortunately not quite there yet. While there are various conductive filaments on the market, many of them are not all that conductive, with a resistance so high that you cannot do much more than light an LED. Furthermore, the print settings on these filaments are difficult to tune, and prints using conductive filaments tend to fail very often.

We also considered building a silicone 3D printer. While these currently exist, they are generally in the early stages of development. 3D printing silicone via extrusion is a difficult problem that requires fine tuning, and this technology also isn't quite there to build CurveBoards that need close to .1 mm precision.

5.5 Deciding on a Fabrication Method

Overall, there are many ways that one could fabricate a CurveBoard and many factors to consider in choosing the best one. We believe that in the ideal case, the fabrication would be fast, easy, and require minimal work. To that end, 3D printing is likely the best fit, but while technology isn't quite there to 3D print very conductive CurveBoards, we propose an alternate approach in which we 3D print a casing with channels and fill the channels with conductive silicone. While this approach still takes a bit of manual labor, it is reasonably fast, easy, and has a low failure rate.

6. Additional Example Scenarios

We fabricated several additional examples to explore the benefits of our approach and verify that our fabrication method works on a variety of geometries. Overall, we were pleased that with these various shapes and sizes, we were still able to fabricate CurveBoards as expected.

6.1 Navigation Helmet

We designed a navigation helmet that would allow its user, a bike-rider, to receive hands-free navigation from a GPS-enabled IoT system (Figure 17). In this scenario, the CurveBoard takes on the shape of a helmet. This CurveBoard took 33 hours to 3D print and 30 minutes to fill the channels.

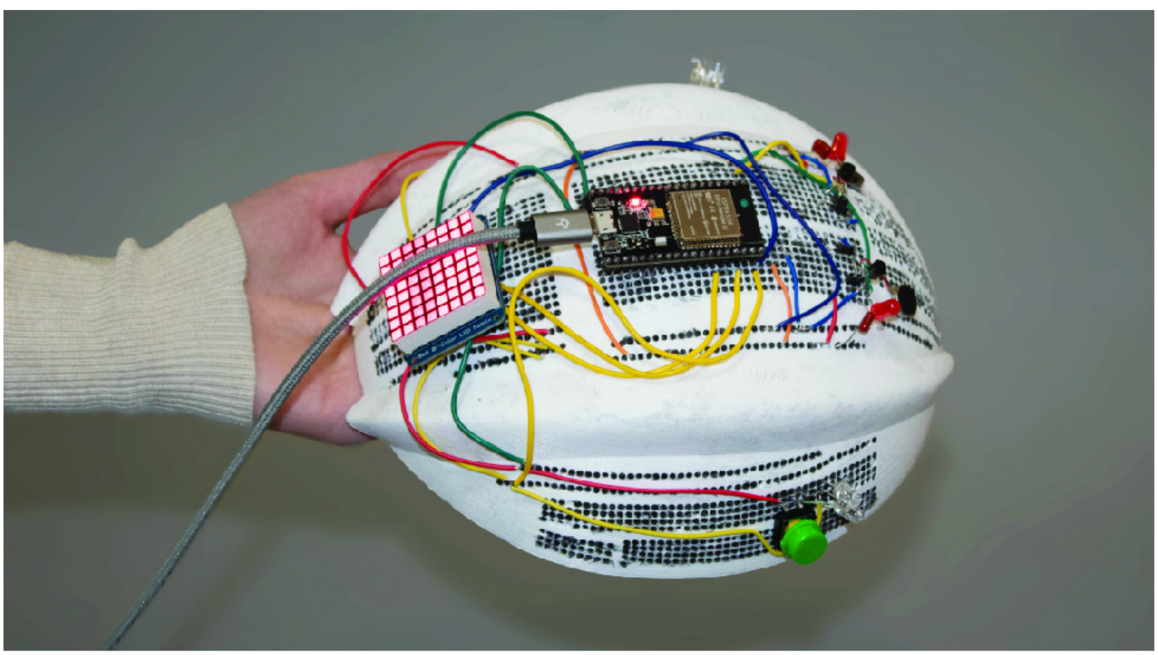


Figure 17. CurveBoard prototype of a navigation helmet

After fabricating the helmet CurveBoard, we used an ESP32 microcontroller and a GPS module to prototype the device. We originally wanted to use speakers close to the ears to give instructions such as “turn left” or “turn right;” however, we found that while wearing the helmet on the street, it was difficult to hear over the sounds of cars and construction on the roads. Since our CurveBoard allowed us to easily swap out our components, we were able to replace our speakers with vibrating buzzers that would indicate which way to turn next. We were also able to test various placements of these buzzers.

We later added lights for signalling to cars and other bikers. An LED array blinks red for better visibility in the night. Each side of the helmet also has a button and a blinking LED that serves as a turn signal. The turn signal lights can automatically turn on when the navigation system believes it is time to turn, but the user can also manually turn them on by pushing the button in case they want to pull over and stop somewhere on the way.

6.2 Frisbee Light Pattern

We created an interactive Frisbee that can display colorful patterns (Figure 18). The CurveBoard took 30 hours to 3D print and 30 minutes to fill the channels. The electrical components used are and Arduino Nano, an IMU, and LEDs of various colors.

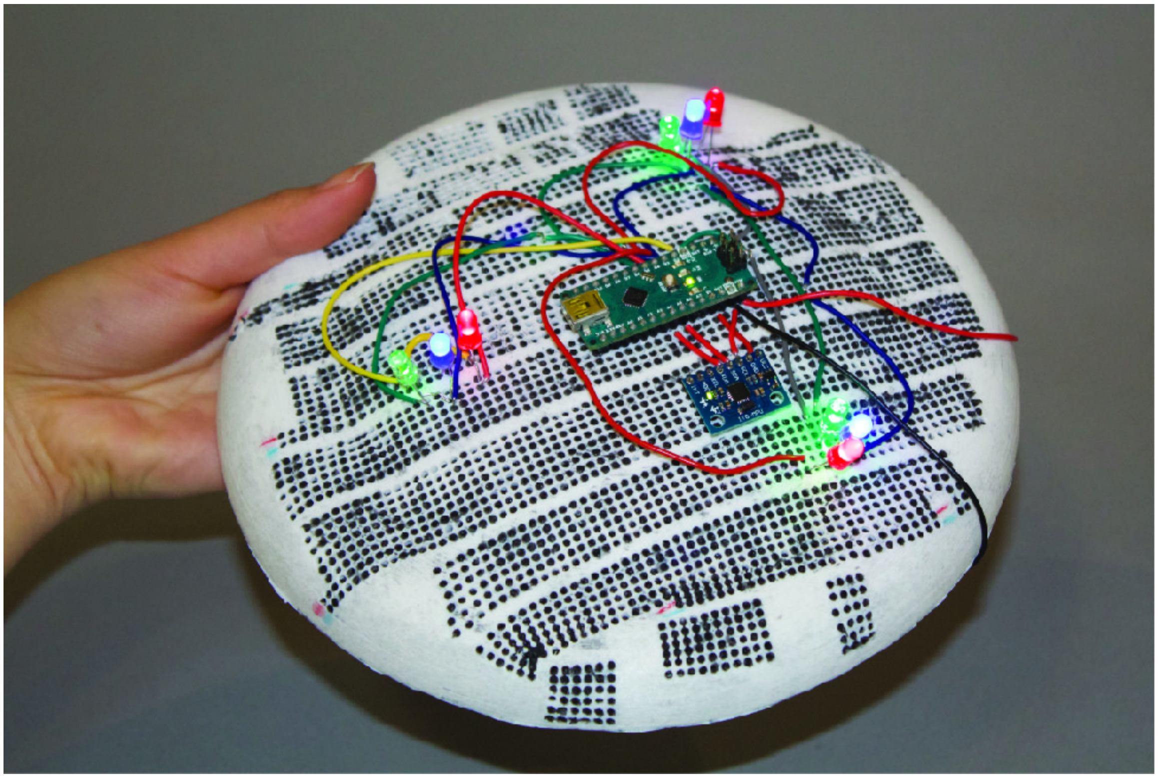


Figure 18. The final LED layout on our frisbee that was created through multiple iterations using this frisbee CurveBoard

The frisbee allowed us to iteratively design different light patterns that can be displayed when the frisbee is thrown. As the frisbee spins in the air, the LEDs can blink at different rates creating unique patterns. The IMU allows us to also sense if the frisbee is in hand, thrown, in mid-air, or caught, so that we can vary what is displayed based on the frisbee.

Using the CurveBoard as a prototyping tool, we can easily move, add, or remove the lights in order to change what pattern is shown, allowing us to quickly prototype the design we want.

6.3 Headphones

We also designed a pair of CurveBoard headphones (Figure 19). This device was actually built out of three smaller CurveBoards that got connected together: one piece for each ear, and one for the headband connecting them together. Speakers were built into each ear component. Overall, this CurveBoard took 18 hours to print (saving time by printing the three parts in parallel) and 20 minutes to fill the channels.

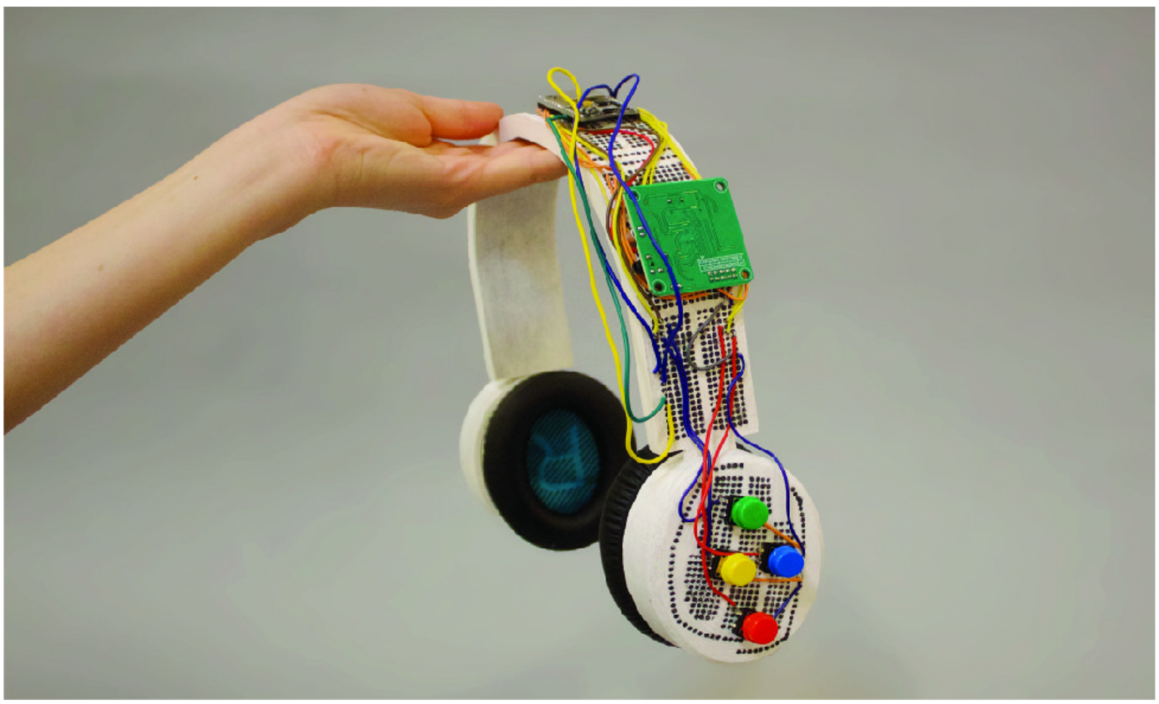


Figure 19. CurveBoard of a pair of headphones that allowed us to find the best placement for the buttons and the LED array.

The headphones use a wifi module built in to the ESP32 microcontroller to connect to internet radio. An mp3 decoder and amplifier allow us to play the radio through the speakers. We also added buttons and knobs that allow the user to change radio channels and control the volume. The CurveBoard allows us to explore the placement of the various buttons, switches, and lights in the context of the user to get a better idea of what parts of the headphones are natural and most comfortable to go to for button presses.

7. Evaluation: User Study

To compare the prototyping experience of CurveBoards to 2D breadboards, we ran a user study with 6 participants.

7.1 Conditions

Our study had two conditions:

Condition #1 (2D Breadboard): In the 2D breadboard condition, participants were given a regular 3D printed bracelet and a selection of differently-sized 2D breadboards for prototyping (Figure 20a).

Condition #2 (CurveBoard): In the CurveBoard condition, participants were given a 3D printed bracelet with integrated CurveBoard for prototyping (Figure 20b).

We ran a within-subjects experiment and randomly assigned participants a condition order.

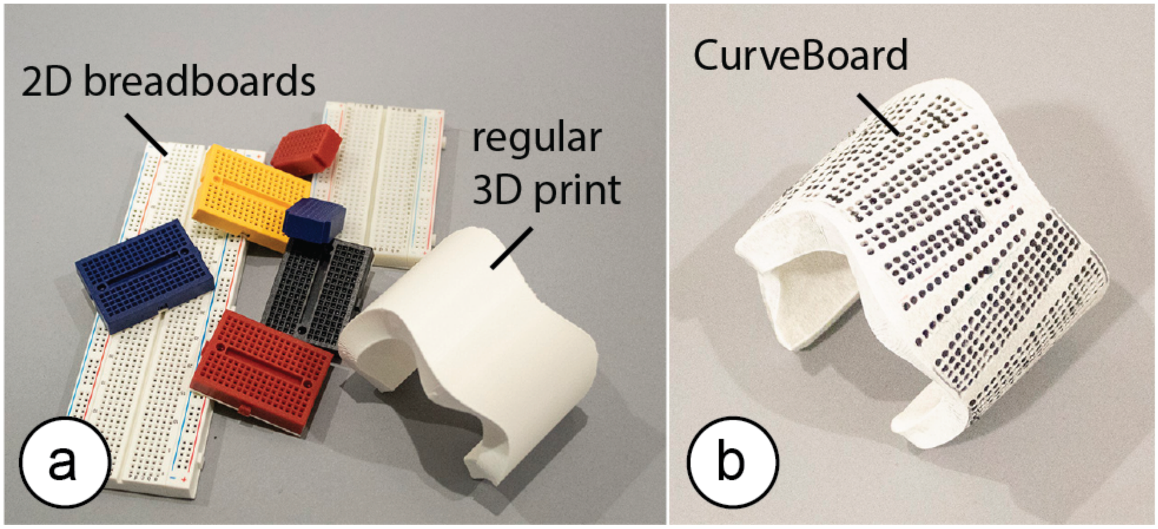


Figure 20. Materials provided to participants in the (a) 2D condition vs. (b) 3D condition

7.2 Task

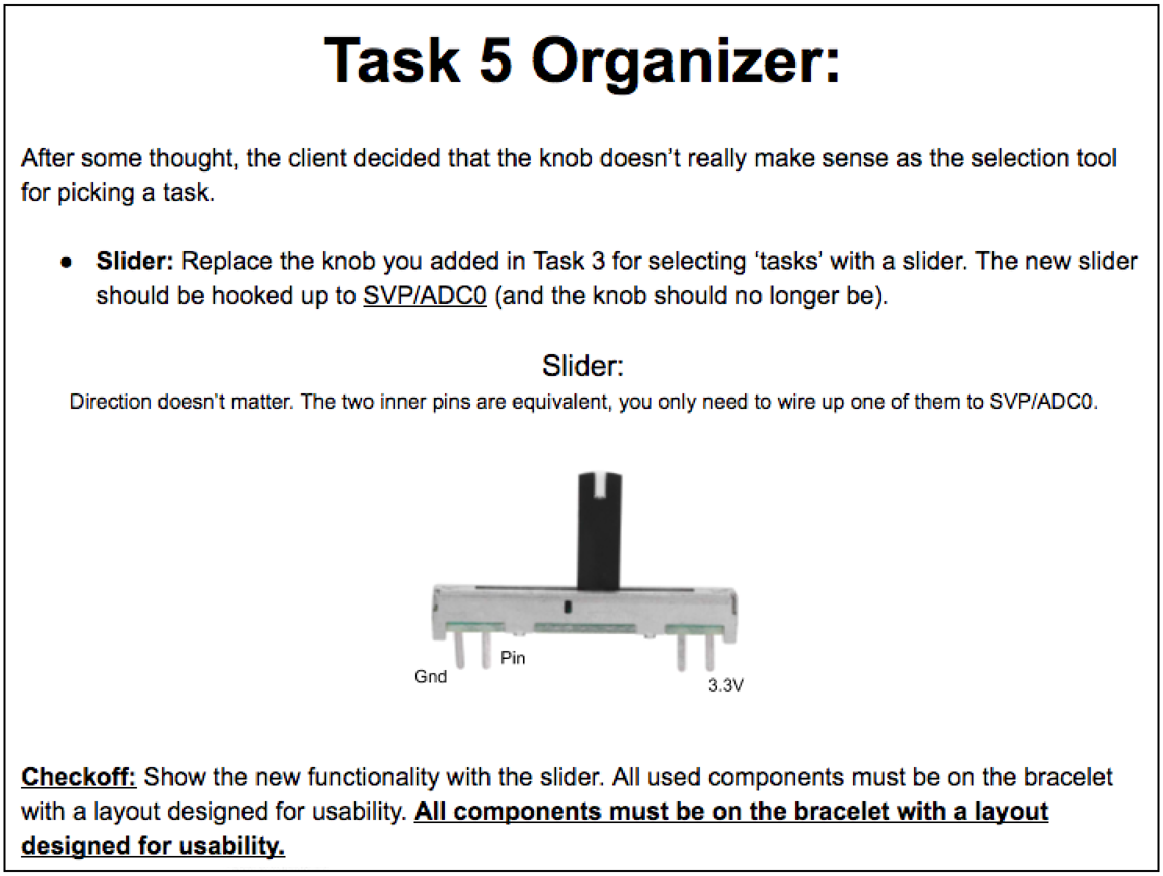
In the study, participants created electronic prototypes for an interactive bracelet. We staged a scenario in which we acted as the clients asking the participants (the designers) to create the bracelet according to our specifications.

The specifications defined which electronic components participants had to add or remove from the prototype but did not tell participants where to place components (e.g. 'add a display to your prototype by connecting it to the microcontroller as shown below in the schematic'; 'add a push button to pin19 to allow users to check off tasks from the todo-list shown on the display').

After participants finished a task, they were asked to show their completed work to the client for the checkoff and then were given a new task to complete since the client had additional feature requests.

This scenario required participants to frequently re-layout their components on the bracelet as new components were added or components were replaced.

Since the wiring for each task was specific (e.g. “connect the button to pin 18”), we were able to upload all the code for running the bracelet onto the microcontroller prior to the experiment. Each task was also clear in including instructions, a schematic of what to wire where exactly, and a simple yes/no checkoff that can easily be tested to see that the functionality works as expected (Figure 21). Thus, participants were able to focus on the wiring in the respective condition and did not have to make any code changes, which was not the focus of the study.



Task 5 Organizer:

After some thought, the client decided that the knob doesn't really make sense as the selection tool for picking a task.

- **Slider:** Replace the knob you added in Task 3 for selecting 'tasks' with a slider. The new slider should be hooked up to SVP/ADC0 (and the knob should no longer be).

Slider:

Direction doesn't matter. The two inner pins are equivalent, you only need to wire up one of them to SVP/ADC0.

Checkoff: Show the new functionality with the slider. All used components must be on the bracelet with a layout designed for usability. **All components must be on the bracelet with a layout designed for usability.**

Figure 21. example task for an organizer bracelet

For each task, participants were instructed to work as fast as possible while making sure the layout had a good usability.

Since we did not want to expose participants twice to the same task sequence, we created two different sequences (Table 1). Both used the same form factor, i.e., the 3D model of the bracelet, but used different electronic components, connected the components to different pins, and displayed different content. One task sequence focused on an organizer bracelet (for setting and displaying alarm, todo list, time) and the other one focused on a health tracking bracelet (showing calories burned, steps walked etc.). Again, participants were randomly assigned one of the task sequences to start with.

Table 1. task summaries for each bracelet

Task Number	Organizer Bracelet	Health Bracelet
1	Add Display 1	Add Display 1
2	Add button to cycle display	Add Display 2
3	Add button and knob to select todos	Add Display 3
4	Add button and knob to set and start countdown timer	Add Display 4
5	Replace todo knob with slider	Add Switchboard to power displays
6	Replace timer button with a toggle button	Replace switchboard with toggle buttons
7	Add Display 2	Replace Display 1 with thinner display
8		Add button (to start/stop timer)

7.3 Study Parameters

Duration of Study

Participants had 60 minutes per condition for finishing as many tasks off the task sequence as possible. We stopped each participants' clock for the duration of the client checkoff after each task.

In total, including the pre-study and post-study questionnaires as well as a break, the study took 4.5 hours. Participants were paid $20 per hour for their time.

Recorded Data

We were mainly interested in qualitative comments from the participants about their experience with CurveBoard vs. traditional 2D breadboard as well as the insights we could obtain from observing participants prototyping approach in each condition. However, for completeness, we also recorded the number of tasks participants finished in the 60 minutes time per condition.

Participants

We recruited six participants (3 females) from our institution. Participants' ages ranged between 18-28 years (mean=20.5, s=3.44).

7.4 Results

Overview of Results

From our pre-test questionnaire, we found that all participants considered themselves experienced (5/6) or experts (1/6) in electronics. 4/6 participants had applied user centered design before in one of their projects, while 2/6 had at least read about the term.

In our post study questionnaire, 5/6 participants stated that they preferred CurveBoard over the 2D breadboard. When comparing the number of tasks participants finished in 2D (avg. 4.5, std. 0.87) vs. 3D (avg. 5, std. 1.5), we did not find a significant difference. Thus, concerning task time, CurveBoards performed equally well to their 2D counterparts.

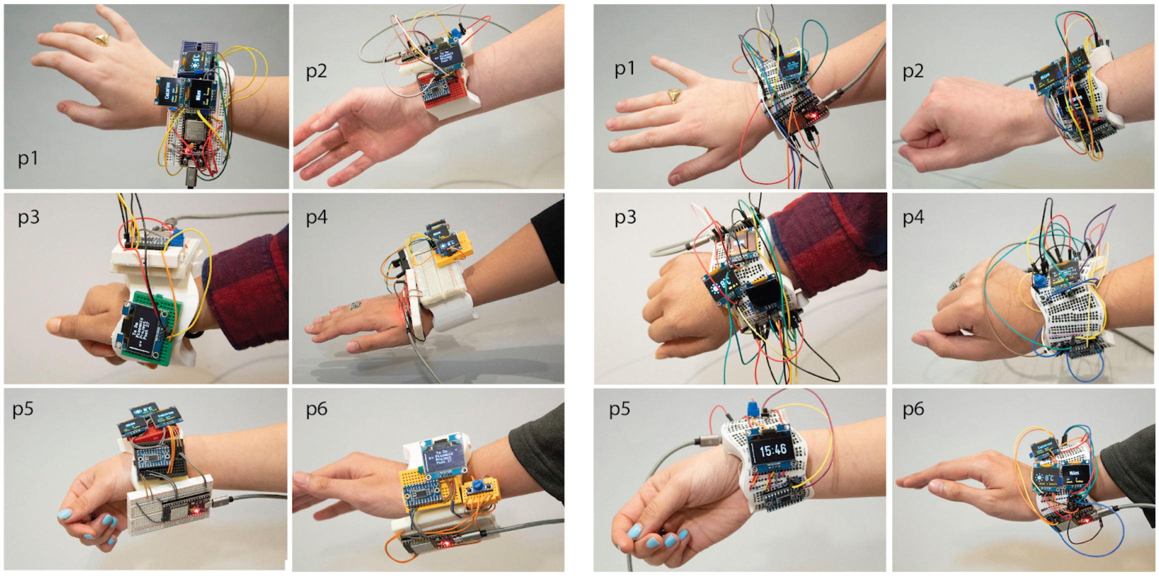


Figure 22. shows the interactive prototypes created in both the 2D breadboard (left) and the CurveBoard (right) condition. We next report on the observations we made during the study and summarize the qualitative comments from the participants.

CurveBoard Advantages

Attaching the Breadboard to the Prototype: In the 2D breadboard condition, participants used a wide variety of methods to attach the 2D breadboard to the physical prototype. Most participants peeled off the backside of the breadboards to expose the sticky surface and then attached it to the prototype. One participant, in contrast, first applied tape across the entire bracelet and then placed the breadboards on it. Another participant used double-sided tape to attach their microcontroller upside-down. Once placed, however, the breadboards were hard to move around in case additional boards had to be added to provide more wiring space for additional components. In the post study questionnaires, participants stated: ‘I spent a lot of time with the 2D breadboard figuring out the best way to situate and attach the 2D breadboards to the bracelet, which I didn’t have to do in the

3D breadboard use.’ (p2), ‘2D was hard to put together, i.e. difficult to attach pieces’ (p4).

Pre-defined Breadboard Size & Component Density: Participants also found it challenging to place electronic components close to each other: often the already attached 2D breadboard did not have enough space available for the extra components and there was not enough room to add a second breadboard close by. To bypass this issue, participants tried to either stack multiple breadboards (e.g. a small one onto a big one), rotated breadboards, or in general tried to use only the smallest breadboards available as can be seen in Figure 22. Participants stated: ‘The 2D breadboard size made it very hard when I was trying to get the OLEDs near to each other.’ (p1) ‘I tried to use as many smaller breadboards as I could to cut down on excess space.’ (p6) ‘I rotated the breadboards to have two breadboards next to each other so that 4 screens are near each other’ (p1). In contrast, when asked about their experience with CurveBoards, participants said: ‘Having space to put components all over the object was pretty useful.’ (p4) ‘The 3D Breadboard had all the wire holes already there so I didn’t need to think about where to physically place breadboards.’ (p1) ‘It was easier to move things and redesign the layout.’ (p2)

Closeness to Final Prototype Design: Participants highlighted that ‘thick 2D bread boards add a lot of volume/weight.’ (p4) to the prototype and were thus harder to place and carry on the arm. Participants also stated: ‘[CurveBoard] lent itself well to designing things that would be easier for a client to wear and use’ (p2), ‘The curvature helped simulate better what the product will be like'. (p4), 'CurveBoard was more realistic (better representation of object/shape/surface area' (p4), and 'it looked more presentable' (p3).

Transfer of 2D breadboard concepts to 3D breadboard use: We also found that participants were able to quickly transfer the concepts they knew from 2D breadboards to CurveBoards: 'The markings (horizontal and vertical lines) made it clear which were the power rails and normal ones. Going from a regular breadboard to this made intuitive sense.' (p2)

Plugging of Wires & Connections: The main criticism participants had with CurveBoard was that the wires were difficult to plug in ('3D was hard to insert wires' (p4)) and that the connections could have been stronger: 'The only things I preferred in the 2D breadboard were the firmer connections' (p2). However, p3 also mentioned: '[the material] in the holes mimics the feeling of putting the wire in the 2D breadboard' (p3)

Features for Improving CurveBoards

Participants also made a list of features they would like to have added moving forward.

Customizing the Channel Connections: Several participants expressed that they would have liked to customize the CurveBoard layout: '[it] would be nice if the CurveBoard was customizable.' (p4) 'I would like to see more 4 holes in a row instead of 3 holes.' (p3) While our CurveBoard editor enables this, we did not have time in the user study to allow each participant to make a CurveBoard from scratch.

Having Custom Electronic Components: Finally, participants also expressed that they wished the electronic components would not be as rigid. ‘It would be so cool if there were curved components (vs. flat ones)’ (p1), ‘The 3D breadboard could have components built for its shape.’ (p6)

User Study Conclusions

In summary, participants’ appreciated CurveBoard’s ability to merge electronic prototyping directly with the form factor of the prototype and provided useful suggestions how to further improve the prototyping experience concerning better wire connections and a potential combination with flexible electronics.

8. Discussion

In this section, we reflect on the idea of integrating breadboards into physical forms. We also discuss some limitations of the CurveBoard design.

8.1 Pre-Defined Routing of Breadboard

On a normal 2D breadboard, components can only be plugged into predefined holes. The channel layout is also pre-defined. This puts various restrictions on how components can be plugged in. For one, there is limited resolution to the adjustment one can make for the position of an object. On a standard breadboard, pin-holes are a tenth of an inch apart, so it is impossible to plug a component into a position more precise than the nearest tenth of an inch. Furthermore, there is a rotational restriction on the orientation a component can be plugged in with. While small wired components such as LEDs and resistors can more-or-less get plugged in any which way, components with header rows can only fit four ways at 90 degree angles. Two of those are usually electrically invalid because the pre-defined electrical channels underneath would create a short.

A CurveBoard carries many of the same limitations. Once one is fabricated, the placement of the pinholes and the channel layout will put some restrictions on the layout of components. This can in some be mitigated by the fact that the software for generating a CurveBoard can offer some flexibility in how the channels are laid out, but designers would have to think in advance about what channel layout best suits their needs.

8.2 Plugging Rigid Components

Currently, much of the hardware designed for breadboarding come as rigid components. With CruveBoards, these become restrictive, as it is not possible to plug a circuit board with a rigid set of header pins into a very curved surface while still maintaining the electrical connections. We have found that this is not a problem for large and slightly curved geometries such as our example geometries, but this is restrive for small and very curved shapes. We hope that in the future, flexible electronics will help mitigate this problem for CurveBoards and allow a wider range of components to be plugged into more differently shaped CurveBoards.

8.3 Types of 3D Model Geometries

Currently there are some restrictions on the type of 3D geometries we can convert into a CurveBoard. For one, any area with a channel needs to be at least 6mm thick in order for the 3D printer to be able to print it, even at high resolution. In addition, channels cannot exist close to each other on a very curved surface without intersecting each other. Large geometries with gentle curves tend to work best for these reasons. While it is unfortunate that we cannot create CurveBoards of arbitrary shapes and sizes, these limitations are not overly restrictive, and we can still design many prototyping CurveBoards shaped as useful everyday objects.

9. Conclusion

We presented CurveBoards, 3D breadboards directly integrated into the surface of physical prototypes. We discussed different fabrication methods to fabricate CurveBoards and demonstrated a new fabrication approach with 3D printing and conductive silicone that allows us to create CurveBoards using off-the-shelf hardware and materials while being conductive enough to allow designers to use electronic input/output components, such as buttons, sliders, and displays, on their CurveBoards. In our user study with six participants, we demonstrated how CurveBoards allow us to create interactive prototypes that are closer in look and feel to the final version.

References

1. Patrick Baudisch, and Stefanie Mueller. Personal Fabrication. Foundations and Trends® in Human–Computer Interaction 10, 3–4, 165-293, 2017.

2. Jesse Burstyn, Nicholas Fellion, Paul Strohmeier, Roel Vertegaal. PrintPut: Resistive and Capacitive Input Widgets for Interactive 3D Prints. In Proceedings of INTERACT 2015, 332-339.

3. Conductive Rubber Breadboard: https://www.instructables.com/id/Rubber-Breadboard-Silicone-Circuit/

4. Kayla DesPortes, Aditya Anupam, Neeti Pathak, and Betsy DiSalvo. BitBlox: A Redesign of the Breadboard. In Proceedings of the The 15th International Conference on Interaction Design and Children (IDC '16), 255-261.

5. Dodecahedral Protoboard. https://www.thingiverse.com/thing:5750/comments

6. Daniel Drew, Julie L. Newcomb, William McGrath, Filip Maksimovic, David Mellis, and Björn Hartmann. The Toastboard: Ubiquitous Instrumentation and Automated Checking of Breadboarded Circuits. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16), 677-686.

7. Daniel Groeger and Jürgen Steimle. ObjectSkin: Augmenting Everyday Objects with Hydroprinted Touch Sensors and Displays. Proceedings of ACM Interact. Mob. Wearable Ubiquitous Technol. 1, 4, Article 134, 2018.

8. Björn Hartmann, Scott R. Klemmer, Michael Bernstein, Leith Abdulla, Brandon Burr, Avi Robinson-Mosher, and Jennifer Gee. Reflective physical prototyping through integrated design, test, and analysis. In Proceedings of the 19th annual ACM symposium on User interface software and technology (UIST '06), 299-308.

9. Wenzel Jakob, Marco Tarini, Daniele Panozzo, and Olga Sorkine-Hornung. Instant field-aligned meshes. In ACM Transactions on Graphics 34, 6, Article 189, 2015.

10. Yoshihiro Kawahara, Steve Hodges, Benjamin S. Cook, Cheng Zhang, and Gregory D. Abowd. Instant inkjet circuits: lab-based inkjet printing to support rapid prototyping of UbiComp devices. In Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '13), 363-372.

11. LilyPad Arduino. https://www.sparkfun.com/products/13342

12. LilyPad Protoboard. https://www.sparkfun.com/products/9102

13. André Knörig, Reto Wettach, and Jonathan Cohen. Fritzing: A Tool for Advancing Electronic Prototyping for Designers. In Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09), 351-358.

14. David A. Mellis, Sam Jacoby, Leah Buechley, Hannah Perner-Wilson, and Jie Qi. Microcontrollers as Material: Crafting Circuits with Paper, Conductive Ink, Electronic Components, and an "Untoolkit". In Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (TEI '13), 83-90.

15. Yoichi Ochiai. The Visible Electricity Device: Visible Breadboard. In ACM SIGGRAPH 2010 Posters (SIGGRAPH ’10), Article 98.

16. Simon Olberding, Michael Wessely, and Jürgen Steimle. PrintScreen: Fabricating Highly Customizable Thin-Film Touch-Displays. In Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14), 281-290.

17. Ronald J. Portugal. Breadboard for Electronic Components or the Like. US Patent D228, 136, 1973.

18. Valkyrie Savage, Xiaohan Zhang, and Björn Hartmann. Midas: Fabricating Custom Capacitive Touch Sensors to Prototype Interactive Objects. In Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12), 579-588.

19. Martin Schmitz, Jürgen Steimle, Jochen Huber, Niloofar Dezfuli, and Max Mühlhäuser. Flexibles: Deformation-Aware 3D-Printed Tangibles for Capacitive Touchscreens. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 1001-1014.

20. Donald A. Schön. The Reflective Practitioner: How Professionals Think in Action. New York: Basic Books, 1983.

21. Sparkfun ProtoBoard Penta Shape. https://www.sparkfun.com/products/retired/8847

22. Sparkfun ProtoBoard Hex Shape. https://www.sparkfun.com/products/retired/8885

23. Koji Tsukada, Maho Oki, Takafumi Yamamoto, and Tomohiro Imaizumi. CARduino: Device Toolkit Suitable for Use in Automobiles. In Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and

Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers (UbiComp/ISWC'15 Adjunct), 407-410.

24. Nicolas Villar, Kiel Mark Gilleade, Devina Ramduny-Ellis, and Hans Gellersen. The VoodooIO Gaming Kit: A Real-Time Adaptable Gaming Controller. In Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE '06), Article 1.

25. Voxel8. https://www.voxel8.com/

26. Chiuan Wang, Hsuan-Ming Yeh, Bryan Wang, Te-Yen Wu, Hsin-Ruey Tsai, Rong-Hao Liang, Yi-Ping Hung, and Mike Y. Chen. CircuitStack: Supporting Rapid Prototyping and Evolution of Electronic Circuits. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16), 687-695.

27. Martin Weigel, Aditya Shekhar Nittala, Alex Olwal, and Jürgen Steimle. SkinMarks: Enabling Interactions on Body Landmarks Using Conformal Skin Electronics. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 3095-3105.

28. Te-Yen Wu, Hao-Ping Shen, Yu-Chian Wu, Yu-An Chen, Pin-Sung Ku, Ming-Wei Hsu, Jun-You Liu, Yu-Chih Lin, and Mike Y. Chen. CurrentViz: Sensing and Visualizing Electric Current Flows of Breadboarded Circuits. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17), 343-349.

29. Te-Yen Wu, Bryan Wang, Jiun-Yu Lee, Hao-Ping Shen, Yu-Chian Wu, Yu-An Chen, Pin-Sung Ku, Ming-Wei Hsu, Yu-Chih Lin, and Mike Y. Chen. CircuitSense: Automatic Sensing of Physical Circuits and Generation of Virtual Circuits to Support Software Tools. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17), 311-319.

30. Junichi Yamaoka, Mustafa Doga Dogan, Katarina Bulovic, Kazuya Saito, Yoshihiro Kawahara, Yasuaki Kakehi, Stefanie Mueller. FoldTronics: Creating 3D Objects with Integrated Electronics Using Foldable Honeycomb Structures. In Proceedings of CHI ’19 (to appear).