# Voxel Invention Kit: Reconfigurable Building Blocks for Prototyping Interactive Electronic Structures

Miana Smith*
Center for Bits and Atoms
MIT
Cambridge, Massachusetts, USA
miana@mit.edu

Jack Forman*
Center for Bits and Atoms
MIT
Cambridge, Massachusetts, USA
MIT Media Lab
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA
jackform@mit.edu

Amira Abdel-Rahman
Center for Bits and Atoms
MIT
Cambridge, Massachusetts, USA
amiraa@mit.edu

Sophia Wang
Aeronautics and Astronautics
MIT
Cambridge, Massachusetts, USA
wangsj@mit.edu

Neil Gershenfeld
Center for Bits and Atoms
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA
neil.gershenfeld@cba.mit.edu

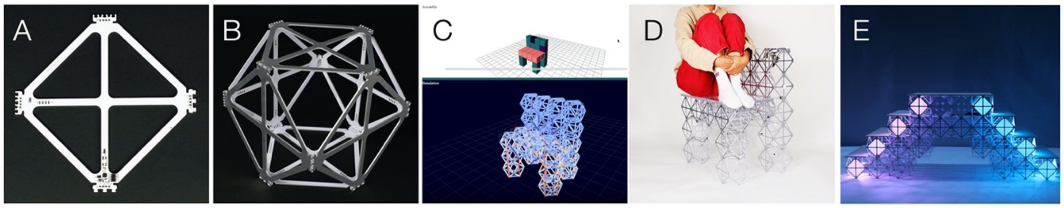


**Figure 1: An overview of VIK. A) a single lattice face produced in a PCB house; B) an assembled functional voxel; C) a screenshot of the design and simulation tool; D) an assembled responsive chair; E) a pressure sensing bridge assembled from VIK.**

## Abstract

Prototyping large, electronically integrated structures is challenging and often results in unwieldy wiring, weak mechanical properties, expensive iterations, or limited reusability. While many electronics prototyping kits exist for small-scale objects, relatively few methods exist to freely iterate large and sturdy structures with integrated electronics. To address this gap, we present the Voxel Invention Kit (VIK), which uses reconfigurable blocks that assemble into high-stiffness, lightweight structures with integrated electronics. We do this by creating cubic blocks composed of PCBs that carry electrical routing and components and can be (re)configured with simple tools into a variety of structures. To ensure structural stability without expertise, we created a tool to configure structures and simulate applied loads, which we validated with mechanical testing data. Using VIK, we produced devices reconfigured from a shared set of voxels: multiple iterations of a customizable AV lounge seat, a dance floor game, and a force-sensing bridge.

## CCS Concepts

• **Human-centered computing**; • **User interface toolkits**;

## Keywords

Personal fabrication, interactive structures, modular electromechanical systems

**ACM Reference Format:**
Miana Smith, Jack Forman, Amira Abdel-Rahman, Sophia Wang, and Neil Gershenfeld. 2025. Voxel Invention Kit: Reconfigurable Building Blocks for Prototyping Interactive Electronic Structures. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3706598.3713948

## 1 Introduction

Functional interactive devices can be difficult to prototype, especially at sizes larger than the desktop scale. To help illustrate the friction in current standard digital fabrication workflows, we consider a hypothetical scenario: Victor is a designer who wants to prototype an interactive chair as part of their course final. Though Victor is hypothetical, this story is informed by the many years we have spent teaching a digital fabrication course, and the common struggles we have observed students encounter while trying to deploy functional objects. Victor begins by using CAD to design the mechanical elements of the chair, intending to eventually

---

*Both authors contributed equally to this work.

*CHI '25, Yokohama, Japan*

ACM ISBN 979-8-4007-1394-1/2025/04
https://doi.org/10.1145/3706598.3713948

CNC cut the chair from wood. Victor first makes a prototype from laser-cut cardboard at their university maker space. After cutting, Victor realizes that the slot-fits are misaligned and has to re-design and re-cut, wasting the previous material and shop-time. Victor then prototypes the electrical system by placing different modules on the chair and connecting them via a mix of jumper cables and copper tape, which have to be soldered as the connections aren't robust enough. Victor then moves on to a final version using ½" plywood, which is thicker than necessary to guarantee load-bearing capacity. They then have to adjust the design, re-cut, re-assemble, and re-integrate the electronics. While carrying the chair to class, a copper tape trace is torn on a door handle, and Victor has to last-minute debug and fix the chair. After the demo, some students have suggestions as to how the chair design and electronics could be improved, but Victor won't follow up on these suggestions as they would require substantially reworking the chair. In the end, Victor moves the chair to their office, before eventually throwing it out once the electronics stop reliably working.

Instead of this laborious process, an ideal combined mechanical and electrical prototyping system would afford cheap, quick iterations with simple tools, robust mechanical and electrical properties, and various I/O capabilities. However, the development of such a system comes with three main challenges: 1) the prototyping of large and robust structures is too time-consuming and/or expensive for rapid iteration; 2) approaches for embedding electronics into 3D forms are difficult or impossible to reconfigure, while the grafting of electronics onto 3D forms is typically fragile and inelegant; and 3) iterative prototyping of electromechanical systems requires continuous access to digital fabrication tools.

While prior work has addressed some issues associated with porting electronics to 3D forms, most of this work is targeted at a relatively small scale and wouldn't help our hypothetical designer. For example, at the desk-top scale, [40] uses curved breadboards for more accurate looks-like prototyping of electronic devices, [35] uses commercial metal 3D printing services to create functional components with embedded sensing elements, and [23] creates a workflow for conforming circuits to curved surfaces. And, while prior works have developed tools for large-scale mechanical prototyping systems, such as large-scale dynamic constructions in [26], or a handheld extruder for 3D-printing prototype furniture-scale objects [3], these leave electronics integration as an afterthought.

We present VIK, the Voxel Invention Kit, as a potential solution. VIK uses aluminum PCBs as 3D lattice building blocks, or voxels, that assemble into structures (see Figure 1 for an overview). The system's basic unit has high mechanical performance with embedded electronics and uses a cheap, fast, and commercially available manufacturing process to increase accessibility to the system.

Our main contributions are:

The creation of an integrated prototyping platform for creating large and robust yet lightweight electromechanical structures that can be built and reconfigured cheaply, quickly, without waste, and using only a soldering iron and pliers.

The development of press-fit functional voxels composed of high-stiffness and high-strength lattice units with integrated sensing, response, and processing abilities for assembling reconfigurable electromechanical structures, as well as electrical and physical accessories to customize IO capabilities and structural shape.

A design tool to enable end-to-end rapid prototyping with VIK: users can use the tool to design structures and simulate the structure's response to mechanical loads to inform their design.

The demonstration of potential interactive use cases with the voxels, specifically, we demonstrate the interactive prototyping process through several iterations of an IO-enabled chair, a dance floor gamepad, and a force-monitoring bridge, which are (re) assembled from a shared set of voxels.

## 2 Related Works

Few works in HCI focus on prototyping large and sturdy structures with embedded electronics. So, to further inform the development of our system, we draw from three topics in HCI that have received more extensive investigation: modular electrical prototyping kits, embedding electronics into 3D forms, and fabricating large, sturdy structures. In the following section, we investigate each of these sub-areas to identify their trends, successful approaches, and overarching limitations. A comparison of some key parameters is shown in Figure 2. These are then synthesized together to construct an intersectional set of principles used to develop VIK.

### 2.1 Electrical Prototyping Kits

When designing modular electronic kits, there are critical design decisions and tradeoffs that affect the audience and agency a kit provides. We intend VIK to be a low-threshold high-ceiling prototyping platform for electromechanical structures that is targeted toward makers and practitioners. To inform the design of our system, we look to review papers that compare many types of prototyping kits to extract meta-themes. We prioritized reviews that discuss the prototyping kits for practitioners and makers, instead of those used for educational purposes. For educational kits, the focus largely lies on the learning that occurs through making, whereas maker-oriented kits emphasize what can be made with that system.

In a survey paper of electrical prototyping systems, Lambrichts et. al [28] identify several key takeaways. First, the top feature that users look for in these kits is that prototypes are easy to iterate—after that, users prioritize easy debugging, durability, reusability, and affordability. With this the authors postulate that "it is more convenient to construct a prototype with a toolkit that (1) does not require connecting individual wires, (2) can be programmed by physically interconnecting blocks, (3) does not require additional tools for connecting modules, such as adhesives or stitches, and (4) can be connected in a bus topology" [28]. Additionally, it is noted that people who employ electrical prototyping kits are generally resistant to adopting new platforms— they are happy with what they use, fearing that a new system will be inconvenient to learn, offer few feature improvements, and could be deprecated in the future. Thus, even if a new system is easier to use, that does not mean that people will actually use it. They also note that for users already familiar with electrical prototyping kits, the electrical and programming expertise required by existing systems is rarely a major barrier and suggest focusing on other features to improve.

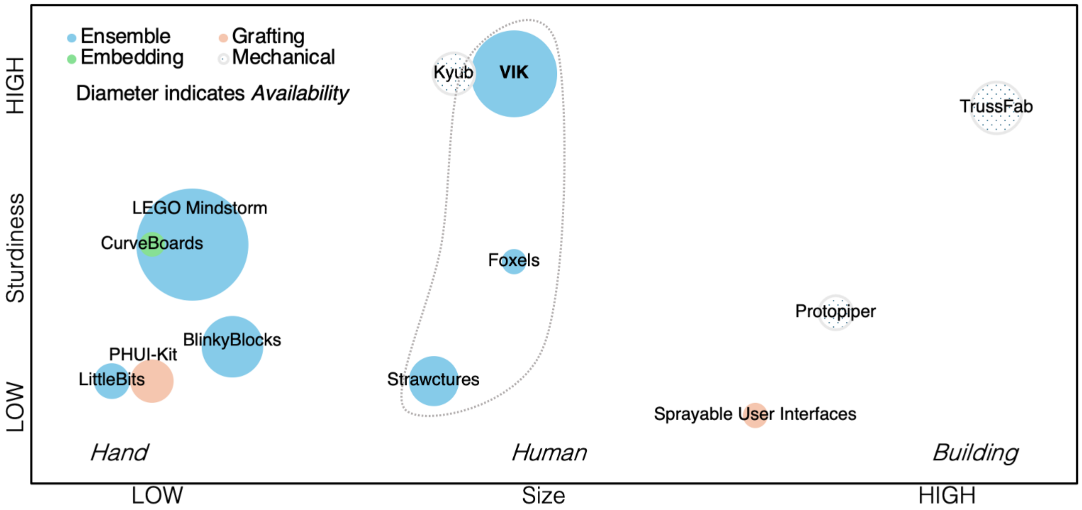


**Figure 2: Comparison of selected electrical and mechanical prototyping kits in terms of sturdiness, size, and availability.**

## 2.2 Fabricating 3D Forms with Integrated Electronics

Since VIK is intended as a combined mechanical and electrical prototyping system, we investigated the different ways that people integrate electronics into physical devices. From this, we identified three main themes: grafting, embedding, and ensemble configuration. In this section, we sift through the considerations and limitations of each approach to inform the approach we take with VIK.

**Grafting** involves adding electrical circuits that conform to the surface of a 3D object. Grafting is handy for adding low-profile electronics to existing objects. Prior work has explored techniques such as mounting electronic modules [23], attaching flexible circuit boards [40], adding conformable copper traces [41], sewing in conductive threads [7], and patterning with conductive inks [18, 36, 38]. Overall, grafting approaches can afford close coupling of the electronics with the final device form but often lack discretion and durability. Additionally, these grafts are usually bespoke to a specific form, and changing the reform requires producing new grafts.

**Embedding** is when the electrical sensing components are integrated within the volume of the objects. Researchers have explored a variety of techniques, such as multi-material printing with conductive material [16], [4], [12, 13], [19], [30]. A common trait of embedding approaches is the co-fabrication of form in tandem with electronics, which allows a more seamless electrical integration. However, iteration and repair are difficult without damaging the larger structure, and complete refabrication is usually easier. Rapid prototyping with these approaches also requires maintained access to digital fabrication equipment, restricting where and when interaction can occur.

**Ensemble Configuration**, the third and final approach, employs electromechanical building blocks that can be (re)assembled into various forms. Blinky Blocks [24], a foundational example of ensemble assembly of digital materials, are cheap building blocks that can be freely assembled into 3D structures with magnetic interconnects. Commercially available ensemble configuration kits exist today, such as littleBits [6] and LEGO Mindstorm [2], but none support the facile creation of human-scale structures. Since ensemble configuration focuses on the rapid reconfiguration of inexpensive building blocks, these approaches prioritize inter-module connections that are fast and easy to (un)make but rarely robust. This can still support the construction of human-scale assemblies, like those seen in Strawctures [37], but with limited load-bearing capabilities. Overcoming this limitation is a central challenge to producing large and sturdy structures, and doing so would allow material to be reused instead of consumed by expensive, time-intensive fabrication processes that disincentivize iteration.

## 2.3 Rapid Prototyping of Large & Sturdy Structures

While rapid prototyping at the desktop-scale is fast and cheap, as scale increases, it becomes increasingly slow and expensive. As the size of a prototype increases, so do the cost of materials, the time needed to fabricate, and the need for structural integrity. As a result, several works in HCI have proposed tools to ameliorate the misalignment between rapid prototyping and building large structures. *ProtoPiper* [3] approaches this challenge by creating structures out of tape tubes that are lightweight, cheap to iterate, and quick to form self-supporting structures. However, these artifacts have minimal load-bearing support.

In *Forte* [9], the authors develop a generative design tool that offers material-optimized designs based on the expected loads and desired form, but iteration requires complete refabrication. *Kyub* [5] employs closed-box designs to create sturdy objects. This system allows for rapidly designing large objects with a Minecraft-inspired "boxel" based design tool, which allows non-experts to quickly design sturdy laser-cut objects. However, the work does not optimize the usage and reusability of the fabrication material, making iterations expensive due to the consumption of sturdy materials.

*TrussFab* [27] uses plastic bottles and 3D-printed joints to create reconfigurable load-bearing structures. The user defines a target geometry, and an accompanying design tool automatically generates a corresponding truss structure and performs structural analysis to ensure suitable strength for the specific application. *TrussFormer* [25] adds pneumatic elements to this system for kinetic structures. The goal of these works is to empower non-experts to design large, structurally sound objects that can be straightforwardly assembled, and using water bottles helps keep costs low. However, the 3D printed joints are bespoke, making them difficult to re-use for different geometries, and the need for a 3D printer may be limiting (getting parts printed via a service results in part costs on the order of \$10-\$30 per part, which will quickly become prohibitive for large structures). This work is extended to steel tubes in [26], which, though much more structurally sound than bottles, necessitates significant metal-work abilities to recreate. All of these are purely mechanical systems as well— though *TrussFormer* uses actuators, the design tool and system are for designing motion, not for designing electrical routing.

As perhaps the closest related work, *Foxels* [31] creates smart human-scale furniture with reconfigurable modules to introduce the benefits of modularity and flexibility to human-scale design. Still, the work has several areas that could be improved. The first is that Foxels are intended entirely for furniture and prioritize aesthetics over cost. Foxels are also not easily available as the design files are not freely available, and replication would require non-trivial amounts of in-house fabrication. The second is assembling Foxel structures requires 3-6 people, which limits the ability for single users to engage with the system. Third, and most importantly for load-bearing structures, the modules presented are only intended for compressive forces, limiting the achievable geometries as tensile and bending loads are not well supported.

## 2.4 Overview of Related Works & VIK Guiding Principles

With VIK, we introduce a construction system that lowers the barrier of entry for the iterative prototyping of big & sturdy things, both on a financial and skill level. At the same time, VIK preserves all of the freedom and customizability that come with electrical prototyping kits like Arduino but with the ability to be reliably and reversibly integrated into the overall structure. Through the related works, we find three overarching tradeoffs that VIK aims to resolve collectively. Thus, we use these challenges to define VIK's guiding principles: Prototyping vs. Big and Sturdy, Availability, and Electrical Flexibility vs. Structural Coupling.

*2.4.1 Prototyping vs. Big and Sturdy.* We define prototyping as the rapid and cheap iteration of a design. Based on prior work, we assume that an ideal prototyping system affords reconfigurability— the ability to easily make reversible connections with components with no special tools. We define big as human-scale, and sturdy as having load-bearing capabilities similar to furniture, able to support both compressive and tensile forces like those shown in Kyub [5]. Additionally, big & sturdy structures need to be safe. Thus, our system must provide a built-in workflow for non-experts to quickly evaluate the stability and safety factor of a given configuration.

*2.4.2 Availability.* Availability refers to the ability to immediately access the system with minimum labor, cost, fabrication tools, and custom processes required. Availability also relates to the educational overhead needed to use the system. A characteristic example of high availability is LEGO Mindstorm [2, 43], which can be readily purchased, leverages common programming frameworks, and provides extensive education support online.

For a general-purpose prototyping platform, it is important to have both. Few will reproduce a fully remixable system that requires fabricating many components and hundreds of parts, especially if the designs are not openly available. On the other hand, a commercially available system that only supports a small number of proprietary modules will inevitably fall short of some users' needs. While a modular set of discrete parts can never encompass all of the things a user may want to create, it should do its best to encompass all the basic needs so that customization is not essential to reach a working prototype. For a VIK system, we prioritize two areas of customization: form and electronics. For electronics, a variety of I/O devices and microcontrollers must be physically and electrically easy to integrate into the structure and network with other devices. For form, the building system should be able to produce a variety of mechanical skeletons that can then be augmented with curved and sloped profiles that can be quickly produced.

*2.4.3 Electrical Flexibility vs. Structural Coupling.* Electrical flexibility is the ability to, at any point in the prototyping process, support easy configuration and reconfiguration of various electronic modules, sensors, inputs, and custom PCBs. Close structural coupling is the seamless and robust integration of electronics into the structure that avoids single-use adhesives and long-spanning wires that are precarious and messy. Electronics should be embedded into the structure when possible instead of grafted onto the surface to keep them secure and hidden.

The contribution of this work lies in addressing these three guiding principles. Because of the difficulty in reconciling these at times opposing priorities, most of the existing literature focuses on either only one of the guidelines or rather makes a tradeoff between two values (e.g. disregarding electrical flexibility in favor of clean structural coupling. In this work, we aim to synthesize the contributions of prior work to address all three principles and increase options for human-scale electrically integrated mechanical prototyping.

# 3 VIK OVERVIEW AND FABRICATION

VIK is an inexpensive and reconfigurable assembly-based building system that enables the easy construction of large and sturdy structures with integrated electronics: adding each voxel not only builds out the mechanical structure but simultaneously builds out the electrical wiring harness.

The voxel design in VIK abstracts both the structural and electrical engineering knowledge so that the end user can focus on developing interactive high-performance structures without needing significant expertise in either of these fields. Similarly, we designed the electrical system— and how it interfaces with the mechanical system— to be straightforward and error-proof. In contrast

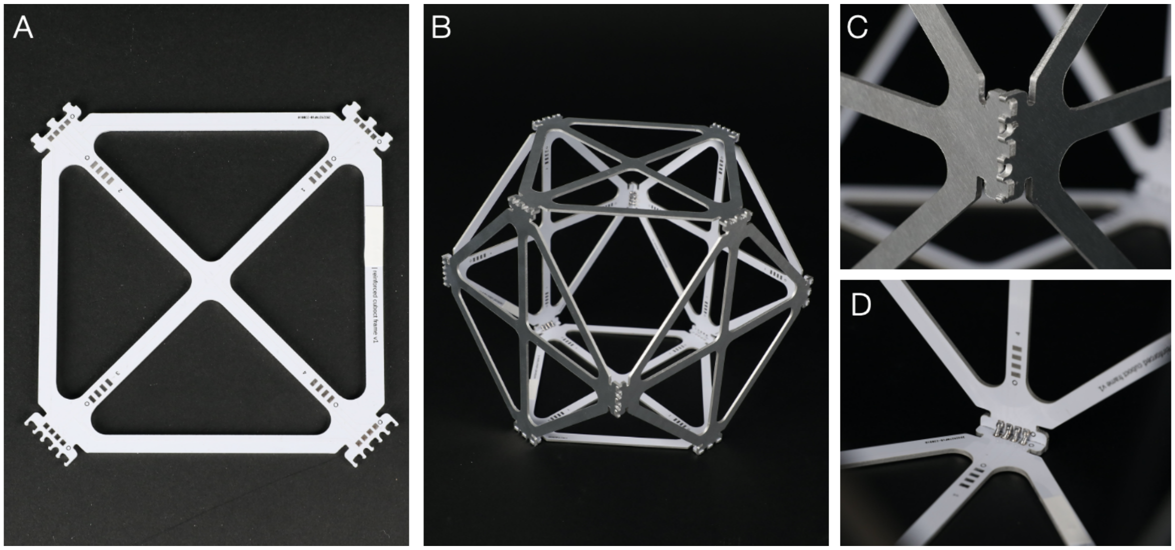


**Figure 3: A) Top view of a PCB voxel face. B) An assembled voxel. The voxel is a reinforced cuboctahedron unit cell made from six aluminum PCB faces. These are made with small snap-fit features at the corners, as shown in C), to mechanically connect faces. Faces are then soldered, as shown in D), to bridge the electrical connections.**

to many other mechanical and electrical platforms, VIK minimizes the necessary digital fabrication expertise and access. VIK primarily uses cheap, recyclable components ordered from a PCB vendor. Because VIK is mechanically and electrically reconfigurable, iterating large-scale structures does not result in wasted material. And, even if a user *does not require* the electronic functionality, VIK is a cheap and accessible tool for building lightweight, high-strength structures.

We provide all the files necessary for replication in the Supplementary Materials.

## 3.1 Voxel Mechanical Structure

VIK uses reinforced cuboctahedra, or cuboct, as the base unit cell, as shown in Figure 3B. The cuboct geometry is chosen for its optimal stiffness-to-weight properties [10], while the face-connected geometry is used for easy assembly [22]. Because the cuboct lattice only achieves its optimal mechanical properties when many unit cells are tiled together [17], we add cross-bracing to each face to make the resulting voxel stiffer and stronger.

A single voxel under compression supported a maximum load of 2243.02 N, corresponding to the force exerted by 228 kg weight at the Earth's surface, e.g., the weight of an average upright piano, 3.5 adults, or 2000 times the voxel's own weight. Under tension, it supports a maximum load of 1198 N, or approximately one thousand times its own weight. The voxel has a modulus of 22.58 MPa at a lattice density of 30.756 kg/m^3. This is competitive with prior research on truss-based architected materials, such as the high-performance micro-lattices described in [39], and outperforms the nearest mass-manufactured discrete approaches [22, 33] by an order of magnitude. Similarly, a three-voxel beam (including clips) under bending supports a maximum load of 1037 N. Taken together, this means that one voxel could support a ~910-voxel long beam in any orientation, and under compression only, could support up to 2000 voxels. Complete details of the mechanical testing setup and results are available in the Supplementary Materials.

We assemble the voxels from 1.6 mm thick aluminum PCB faces. The assembled voxels fill in a 150 x 150 x 150 mm bounding box—this size is chosen as appropriate for human hands to fit into. Aluminum is chosen as the PCB substrate as it provides good mechanical properties at a cheap cost with minimal design and fabrication effort required from the end user.

Currently, a user could order our design directly from a board house at a low cost (we used JLCPCB, which priced the basic frame at approx. $0.70 per face at an order quantity of 250 PCBs), requiring no fabrication equipment aside from a soldering iron. It is worth noting that it is orders of magnitude cheaper to purchase these faces as aluminum PCBs than as custom CNC profiles or as outsourced 3D-printed parts. Even without the added benefit of included PCB routing, the aluminum PCB voxels represent a cost-effective strategy for achieving large-scale, high-performance structures.

## 3.2 Basic Voxel Frames and Connector Systems

VIK uses two types of structural PCB face: a basic frame that routes power and signal but carries no electrical components, as shown in Figure 3A, and a microcontroller frame, which integrates a microcontroller directly into the lattice. VIK additionally provides a small library of traditional PCB attachment boards, which mount directly to the basic frame (further discussed in the following section). A snap-fit joint makes the mechanical connections between faces within a voxel. Figure 3C shows a connected voxel. We make the inner-voxel electrical connections by bridging exposed pads at the corner of each voxel with solder, as shown in Figure 3D, similar to the system used in [8] or [34]. Although soldering every corner

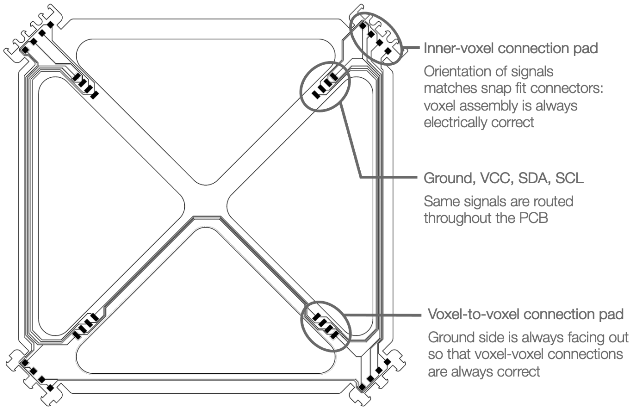


**Figure 4: Diagram of the PCB routing. Four signals are routed throughout the structure: Ground, power, and two data lines. These signals are exposed at the corners for inner-voxel connections and on the beams for between-voxel connections.**

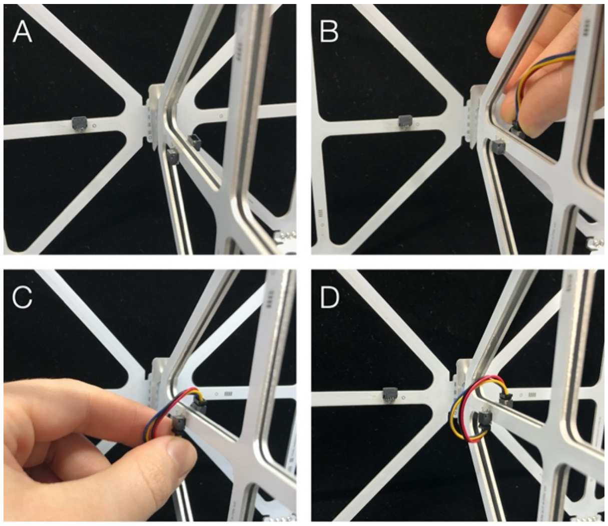


**Figure 5: Installation of the Qwiic connector. 1) shows the two voxel faces to be connected, with the connector pre-soldered in the same position on each face (the connector we have chosen is polarized, so the connectors must be installed consistently for the cable to plug in). 2-3) the cable is plugged in, and 4) the resultant connection.**

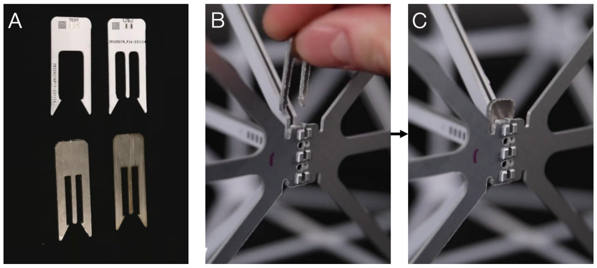


**Figure 6: A) Two-prong clip and three-prong clip examples. B and C illustrate the installation process of the clips.**

in a voxel is unnecessary, it reinforces joints and provides failure-avoidant electrical redundancy. Each voxel has twelve solderable joints; it is only necessary to solder five to achieve full electrical connectivity within a voxel.

In this implementation, VIK routes four signals: power, ground, and two data lines, which we use as the data and clock lines for I2C [42], a common and simple communications protocol for microcontrollers widely used in, e.g., SparkFun and Adafruit sensor boards. Basic frames are designed with multiple breakout pads with these four lines to enable easy interfacing with custom boards (such as the microcontroller attachment boards in this project) or with commercial sensors or development boards, which often include I2C as an interface option. Figure 4 shows the routing of the basic frame, showing the joining of all the connection pads within a voxel frame. The snap-fit connectors orient the four inner-voxel connection pads at the corners of the voxels— this means that it is impossible to connect two voxel faces with the wrong electrical polarity. The voxel-voxel connection pads on the inner beams of the voxels are all oriented with the same polarity, which means we can make voxel-to-voxel connections from any side or orientation of the voxel face.

We make voxel-to-voxel connections using off-the-shelf connectors (see Figure 5). We specifically use Qwiic connectors due to their compatibility with many popular electronics prototyping systems (Sparkfun, for example, carries 320 products tagged with "Qwiic"), though any other 1 mm pitch 4-pin connector may be substituted. The connectors have a wider range of deformation than the lattice, so under loading, they will only fail after the voxels have already mechanically failed. The separation of the electrical connector and mechanical connector also enables more advanced users to route multiple I2C networks in a single structure or to more easily adapt the existing physical traces for their preferred network.

Voxel-to-voxel mechanical connections are made using a snap-fit clip that latches into indexing features on the voxel faces, as shown in Figure 6. The clips are also made from aluminum PCB via a board house, at <\$0.10 per part. We explored two clip configurations: a 2-prong and a 3-prong version. We recommend users use the 3-prong version for added stability, though we include an image of the 2-prong version for reference as we use it throughout this paper.

### 3.3 Integrating Microcontrollers

A structure made using VIK will likely consist of basic frames with microcontrollers placed into the structure as needed. We currently implement multiple microcontroller types so that users can choose what they are most comfortable with. The first set is built around the ATtiny1626 microcontroller, which we chose as it is low-cost, robust, and compatible with the Arduino programming environment, making it easier to use. This is provided as both a frame-integrated board and as a separate attachment PCB (see Figure 7).

We built the second family of microcontroller boards around ESP32 microcontrollers for added WiFi and Bluetooth capabilities (see Figure 7). The current VIK system architecture typically uses

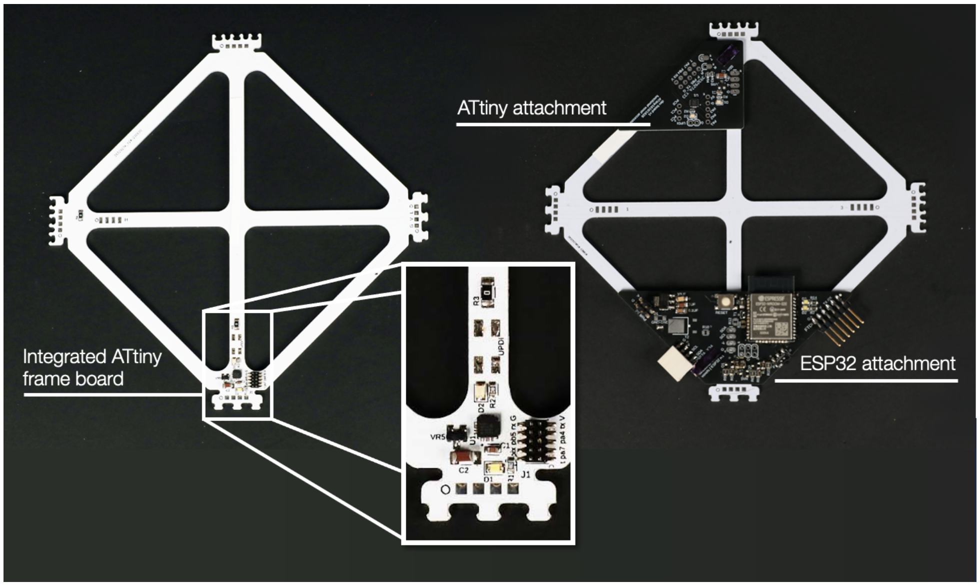


**Figure 7: (Left) ATtiny microcontroller frame. The inset provides a zoomed-in view of the integrated microcontroller on the frame. (Right) A basic frame with an ATtiny attachment PCB installed on top and an ESP32 attachment installed on the bottom.**

one ESP32 microcontroller as the primary controller of the I2C bus, with ATtiny boards as targets (though the current hardware is compatible with some alternate networking architectures). ATtinys would act as interfaces to input and output devices, with the ESP32 determining overall behavior. This architecture enables easier re-programming, as users can change the system behavior by modifying only the ESP32’s code instead of all of the ATtinys. This enables easy interfacing with external applications: the ESP32 could be controlled via a smartphone over Bluetooth or accessed via a website, given its WiFi capabilities. In the Supplementary Materials, we provide the schematic and board files for all the custom development microcontroller boards used in VIK so that users can easily order them from a board house and assemble the PCBs.

For users who are less comfortable with microcontrollers, we additionally implement boards based on the Seeed Studio XIAO development boards, which are small-form-factor and require minimal intervention to interface with VIK. Development boards with built-in Qwiic connector interfaces, such as the Sparkfun Artemis Nano, may also be directly integrated into VIK with no soldering, which we show in Section 6.

## 3.4 Electronic Modules

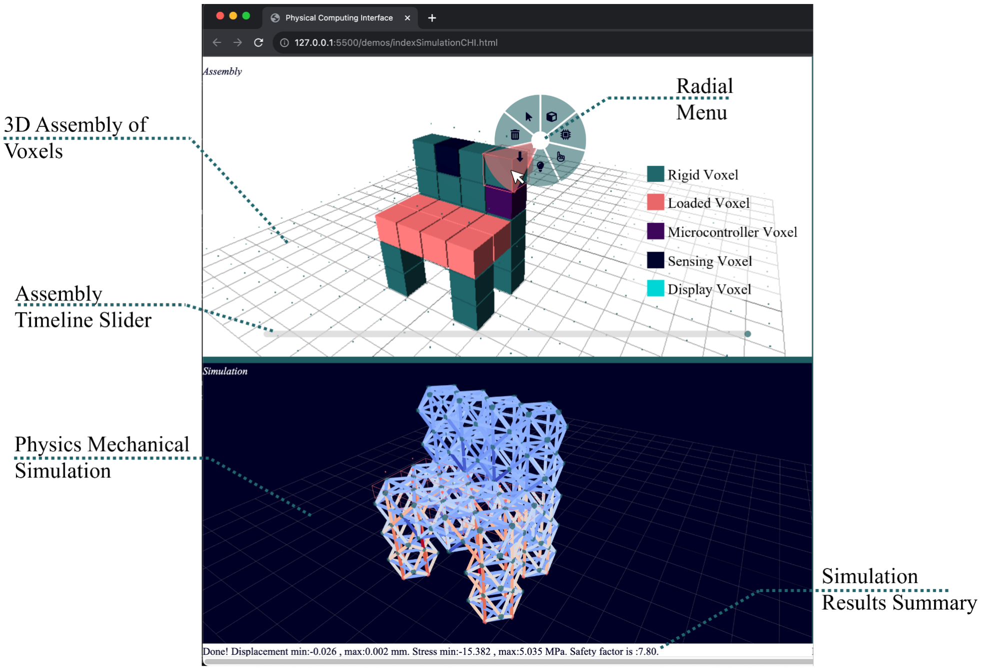
Input and output devices are added to the VIK system to build out electronic modules and customize the system’s operation to the end user’s desire. We designed the microcontroller frames and the basic frame to facilitate the inclusion of almost any input or output device an end user may require. Sensors, actuators, or microcontroller development boards with I2C interfaces directly integrate into the basic frame of VIK. For devices or systems without these interfaces, we have developed microcontroller board designs to operate de facto as an I2C device in the lattice.

We have explored a set of both self-configured and commercially available I/O devices. Primarily, we have focused on implementing our own version of capacitive force, touch, and proximity sensing as inputs to our system and used LEDs, speakers, and vibration motors as outputs, while additionally demonstrating a selection of Sparkfun and Adafruit Qwiic modules on VIK For capacitive sensing, we use either a two-plate system for force sensing or a single-electrode version of a proximity and touch system, as in [44]. We make the electrodes out of copper tape and focus on this one input style as it is simple and cheap to implement yet enables a wide range of tactile and non-contact interactions. For our outputs, we tested LEDs, NeoPixel LEDs, speakers, and vibration motors, though we can also trivially drive other similar systems (e.g., DC motors, buzzers, hobby servos). The Applications section implements several examples of these systems.

# 4 DESIGN TOOL OVERVIEW

## 4.1 Design and Assembly

We developed a physical computing design tool: an online integrated tool where the users design a structure and simulate the loads/deflection of the structure simultaneously, based on the tool introduced in [15]. The goal of this is to enable users without engineering backgrounds to rapidly validate the structural stability of their designs, as well as to provide a fast way to design in a novice-friendly Minecraft-like environment. The tool has two main windows (see Figure 8). In the first “Assembly” window, the user places the voxels on a predefined grid the same size as the voxel. The cursor automatically snaps so voxels can only be placed on the base grid or attached to previously placed voxels. In the background, the assembly sequence is saved, and one can later replay this assembly sequence to help inform the assembly process.



```
Done! Displacement min:-0.026 , max:0.002 mm. Stress min:-15.382 , max:5.035 MPa. Safety factor is :7.80.
```

**Figure 8: Design tool overview. The design tool has two windows: Assembly and Simulation. Voxels can be assembled in the Assembly window, and mechanical response to applied loads can be simulated in the Simulation window along with the summarized results, including safety factor.**

When the user right-clicks on any of the voxels in the "Assembly" window, a radial menu appears, and one can select, delete, or change the type of the voxel placed. There are five types: rigid/base voxels, microcontroller voxels, sensing voxels, display voxels, and load voxels (where loads/forces are applied). When right-clicking on the background of the "Assembly" window, three buttons appear: one can either save, upload the design, or start a static simulation to calculate the deflection of the structure. To save and upload the design, we use a .json file format to capture all the voxels added in the design session, including their properties and placement order. We additionally export a suggested BOM, in terms of PCB face count and type to facilitate easy part ordering.

## 4.2 Structural Mechanics Simulation

Our goal for the structural mechanic simulation is to provide fast and reliable feedback to the designer— unlike fully meshed solid FEA models, which require expertise to set up and frequently fail for complex structures like voxels, our system returns results in seconds. We developed a static linear finite element analysis (FEA) implementation to perform reduced-order beam model simulation using Bernoulli-Euler beam elements for the simulation engine used to model the deflection of the structure under self-load and applied external load. The simulation discretizes the voxel structure into node and beam elements. Note that the clips between voxels are modeled as beams, with their constitutive properties set based on measured data (see the Supplementary Materials for more details), but because of their short length, they are not easily visible. Each node has 6 degrees of freedom: three rotational degrees of freedom and three traditional translational degrees. We use the Bernoulli-Euler beam theory to prescribe the biaxial bending, transverse shear, and axial stretching that each beam experiences.

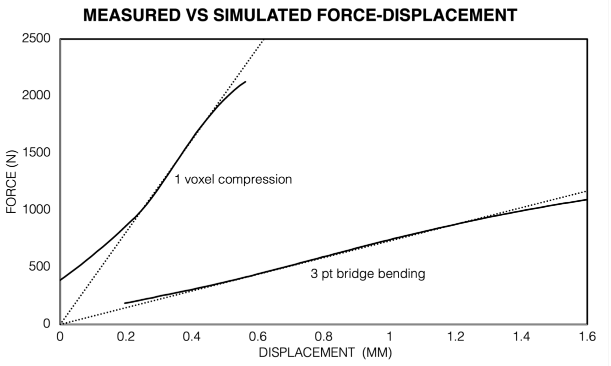


**Figure 9: Measured vs. simulated results for force-displacement in the elastic range, showing good agreement with the toe region removed.**

In [22], a study compares the results of reduced-order beam FEA simulations in comparison to fully detailed meshed voxels simulated using solid FEA models, showing that the simulation results are in good agreement with each other and the experimental data. However, the fully meshed FEA model required approximately three orders of magnitude more elements than the corresponding beam model. Since the purpose of the design tool is to give fast and instant feedback on design decisions, we chose to use the beam model as it achieves the best balance between simulation speed and accuracy. We also compare our simulated elastic response against the measured response for a few voxel geometries, showing good agreement for both single voxel compression and 5-voxel 3-point bridge bending (see Figure 9 and the Supplementary Materials for mechanical testing details).

After the user runs the simulation, an animation of the structure's deflection (exaggerated 10x) appears, and the beams are colored based on their stress value to identify potential points of failure. Finally, a summary of the results (minimum and maximum displacement, minimum and maximum stress values, and safety factor) is calculated and displayed at the bottom footer.

The user can make informed design decisions by having instant feedback and visualization of the structural assembly and simulation in a simple integrated design tool without going back and forth between different interfaces. Importantly, the user does not need to understand any of what this section just described. Users assemble voxels together like in Minecraft, and the simulation window helps users understand the flow of mechanical forces in their system without experience in this area.

## 5 SLOPES AND CURVES

Many of the human-scale structures we interact with include curved forms and soft padding that clad an object's 'mechanical skeleton'. Since the forms made from the voxels on their own are rigid and cubic, we designed a fast, inexpensive, and machine free way to make curved profile modules of varying stiffness using PCBs to make molds for expanding foams. The goal of this is to provide users without access to digital fabrication equipment or CAD background a fast, easy, and cheap way of producing a small library of curved/interpolating shapes to augment VIK with.

The exterior of the mold is made from slot-fit rigid PCBs with a templated flex-PCB, which can be cut and folded into preset shapes. The total cost of the PCB/flex-mold is ~$14 and can be ordered along with the VIK faces (see Figure 10). This approach may seem circuitous, but importantly, it does not require digital fabrication equipment, is still cheaper than ordering custom 3DP or laser cut parts, and is relatively fast: the two foams we used, Smooth-on FlexFoam for soft parts and FoamIt for rigid parts, solidify and can be handled after 30 minutes and fully cure in 2 hours. This means with ten molds, we could produce 20 shapes in one hour and would be fully deployable in 2.5 hours. The system is also reconfigurable, much like the voxels (see Section 6.4 for reference). We provide a detailed instruction walkthrough for this in the Supplementary Materials. For more advanced users, we additionally provide the base 3D models in the Supplementary Materials, which may be augmented to provide more customized molding surfaces.

Though there are limitations to this approach, it is still a fast and easy way to generate more accurate looks-like and feels-like prototypes with VIK, which is our priority. We additionally provide the designs for the other surfaces we used, so users with fabrication equipment access or CAD experience can build off these methods as well.

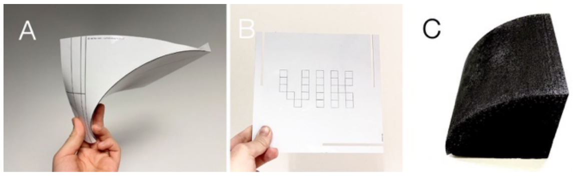


**Figure 10: A) A curved-crease folded flex PCB molding insert. B) One side of the rigid PCB molding base. C) A resultant soft-foam curved block.**

## 6 APPLICATIONS

In this section, we demonstrate how VIK enables the reconfiguration of a standard set of voxels to create various interactive devices. Here, we revisit our hypothetical designer, Victor, who has a library of VIK parts and wants to build some interactive structures. Each demo object is a large, electronically integrated system that would be difficult to produce via standard digital fabrication tools. VIK enables Victor to (almost) trivially assemble these human-scale, load-bearing applications and place electronics into the structure without needing to think about wiring harnesses, cable management, or structural validation, as VIK handles all of this.

Further, VIK enables the complete re-use of the voxels. Starting from a dance floor gamepad, each demo is assembled (and disassembled) from the same set of voxels. This is illustrated in Figure 11, where the dance floor becomes a chair and mobile robot, which then combine, are disassembled, and added into a bridge, which is then transformed into a lounge chair.

### 6.1 Party Dance Floor

In the first example, Victor is hosting a party and wants a fun activity for the guests— a Dance Dance Revolution-style game. But, they're daunted by the task of ensuring that what they build is safe— a 70 kg person jumping on the structure repeatedly exerts forces in the range of 1700 N— and that the electronics system can survive these loads, and that their friend who's really into Dance Dance approves of the physical feel of the dance floor. Additionally, they don't have access to any shop tools, let alone digital fabrication machines, which puts most existing solutions such as [20, 21] out of reach. With VIK, they can quickly implement a capacitive sensor-based dance pad using a 6x6 bed of voxels, using only a soldering iron, scissors, and pliers (see Figure 12).

Victor first assembles the 6x6 voxel grid with a few friends before outfitting it with electronics. After doing a little research online, Victor determines the easiest solution will be capacitive sensing arrows and USB emulation so that they can play the game with an existing online dance floor game— this is all functionality VIK supports. Four active dance direction sensors using capacitive sensing are implemented by plugging in VIK attachment ATtiny boards to the voxel grid. The top sensor surface is formed from acrylic panels that sandwich copper tape to create durable pressure sensors. The sensing boards, which additionally control NeoPixel LEDs, are networked with a VIK ESP32 board, which emulates a USB keyboard for compatibility with popular open-source dance games (in our case, we adapted a JavaScript web app from [11]). The dance floor logs approximately two person-hours of total use

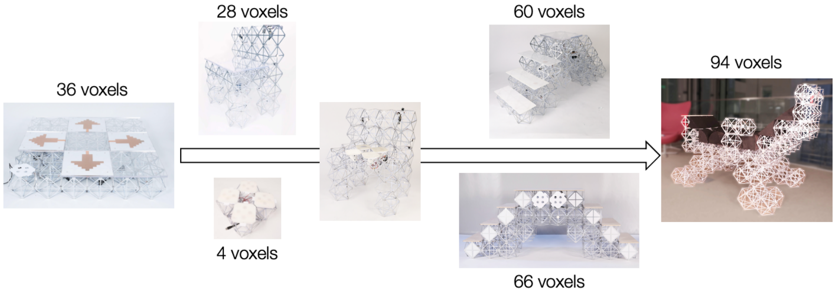


**Figure 11: Flow of voxels used in the example applications.**

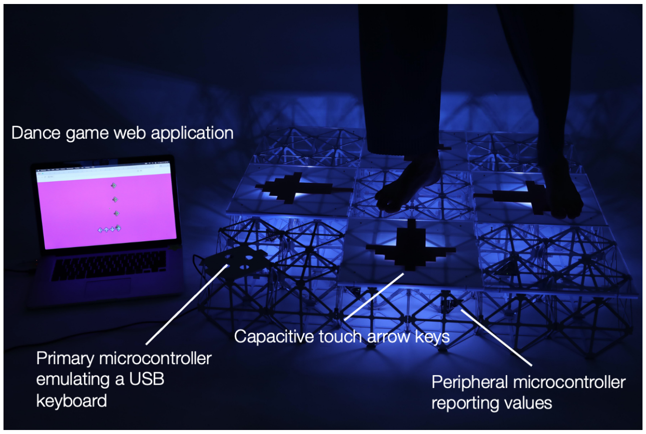


**Figure 12: Dance floor with subcomponents labeled.**

and shows no signs of mechanical or electrical damage, indicating good fatigue performance. If we assume a person dances at 1 Hz and exerts 1.5kN with each move, evenly distributed over one of the plates, then each of the loaded voxels goes through ~480 load cycles of 375 N. As this is only ~17% of a voxel's total loading capacity, it follows that we would not expect fatigue failure.

## 6.2 Interactive Chair Version 1

Once the party is over, the dance floor is no longer needed. Instead of having to forever store an awkwardly large piece of equipment (or throw it out, wasting the materials, cost, and effort), Victor disassembles the dance floor without compromising mechanical stability or electrical functionality.

Victor then gets a new hobby: experimental music instruments, and so reconfigures the voxels into a chair that is also a theremin: a Chairemin. Wanting to reuse as many components from the dance floor as possible, Victor first runs through a few designs of the chair in the design tool to help visualize potential sensor locations, as well as to validate the structural performance (see Figure 8). Victor finds that for their weight, approximately 80 kg, the chair has a safety factor of 7.8, which is plenty, and sets off to build it. The Chairemin uses two capacitive proximity sensors to detect the user's body while sitting on the chair (see Figure 13 for labelled components of the Chairemin). The sensor on the back of the chair controls the volume of a tone sent to a speaker module, and the sensor on the side of the chair controls the frequency. The Chairemin is played by using their hand to control the frequency, which offers finer-grained control, while the volume is controlled by how far back they lean into the back of the chair.

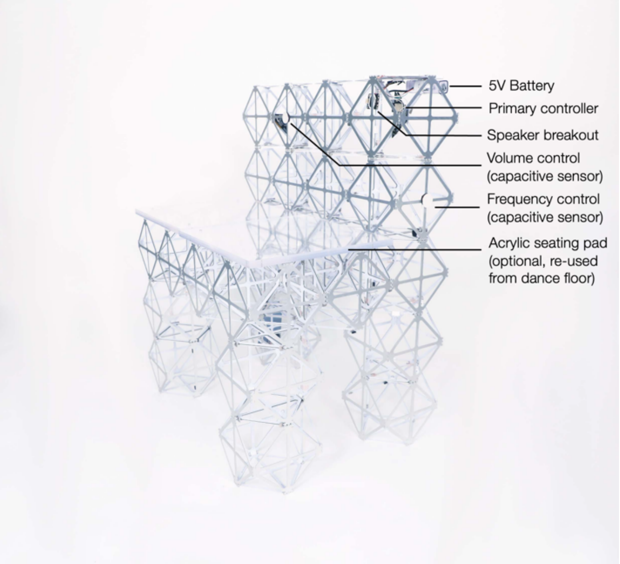


**Figure 13: Chairemin with labeled components. The primary controller, speaker, and rechargeable battery are all placed within the same voxel in the upper corner of the chair. The capacitive sensing modules are placed one and two voxels away, with the sensing pads placed on the exterior frame of the chair.**

The Chairemin reuses two clear panels from the dance floor to form the seat and reuses two capacitive touch microcontrollers for the audio controls. Note that the Chairemin does not require the acrylic panels to support loads, as shown in Figure 14, and is there for comfort.

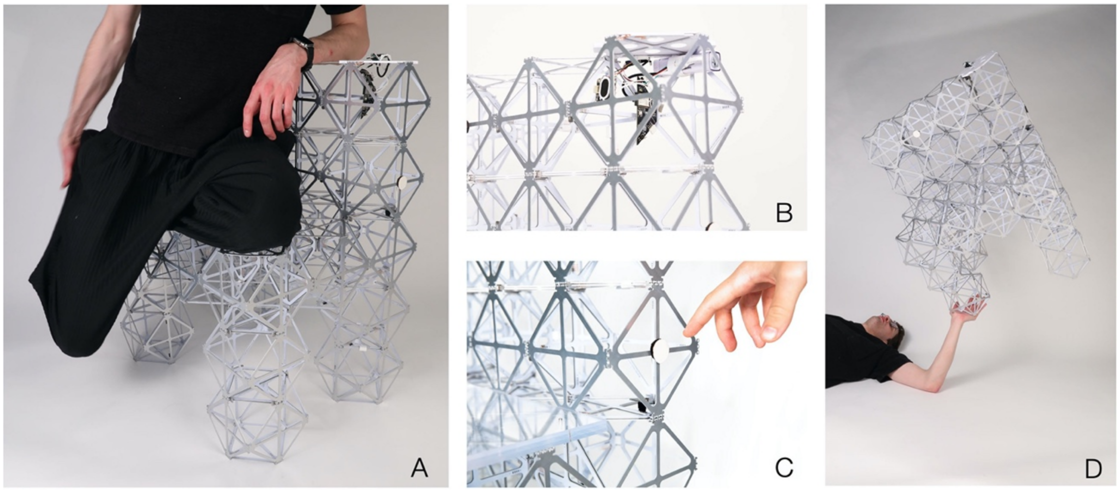


Figure 14: A) A person fully supported by the Chairemin. B) Close-up view of the primary module, containing the ESP32 controller, Adafruit speaker board, and 5V battery. C) View of the frequency control capacitive sensor electrode. D) The Chairemin is lightweight.

The combined electrical and mechanical system means that it is trivial to try different configurations for sensor and audio placement so the experience can be tailored for a specific user. To try new sensor locations, we can simply plug the capacitive sensors into any of the voxels in the structure (without needing to cut new wires, de-solder, re-solder, etc., as would be necessary with a standard setup). This makes it easy to adapt the Chairemin for use by people with different body types and easy to quickly iterate different interaction modes.

## 6.3 Mobile Robot to Massage Chair

After transforming the dance floor into the Chairemin, Victor uses the remaining four voxels to create the Slowbot, a vibrating robot for passively sweeping floors at speeds of up to 3 cm/s. Victor already has the electronics for the system made and is just looking for a convenient and lightweight mechanical harness, and so uses the leftover voxels.

The slow bot is assembled using a 3D-printed directional fur (PolyBrush, OPT Industries). The directional fur means that the slow-bot's ground-facing surface has lower friction in the direction of the fur and higher friction opposite to it. This anisotropy biases the movement caused by the vibrating motors to move the robot in one direction.

Ultimately, Victor decides that the Slowbot is indeed too slow and an uncompelling use of their voxel kit. So, they pop the seat of the Chairemin out and replace it with the Slowbot (see Figure 15). A quick code change switches the capacitive touch sensors to trigger the vibrational motors instead of the speaker output, taking about 10 minutes total to reconfigure and reprogram the chair.

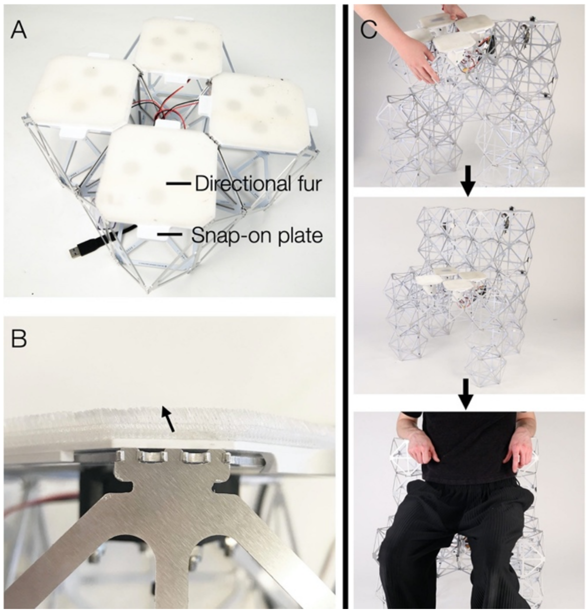


Figure 15: Overview of the Slowbot. A) Bottom view of the Slowbot, showing the directional fur modules. B) Close-up view of directional fur, with an arrow indicating its orientation. C) Installing the slowbot into the Chairemin.

## 6.4 Bridging Play Structure

Victor then wants to build a play structure for their child and decides to assemble a miniature bridge (termed "abridged bridge") with embedded load sensing, which will allow them to track and test what structure configurations result in the most interactions. Load sensing is implemented via two-plate capacitive distance measurement using microcontroller frames, as shown in Figure 16. Voxels are additionally embedded with NeoPixels, and all systems are coordinated via a WiFi-enabled microcontroller and powered via USB 5V battery packs.

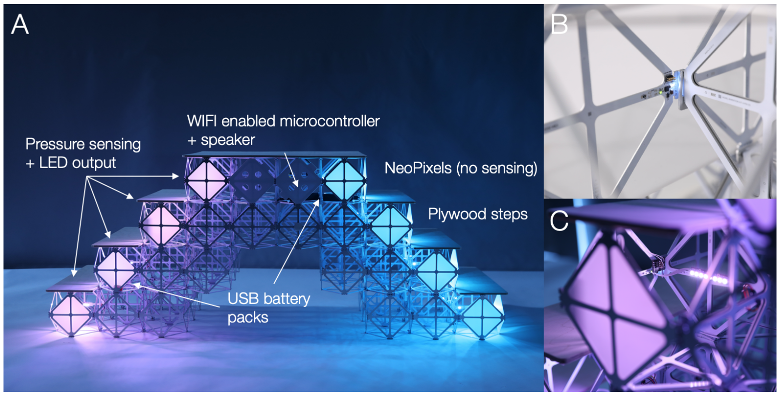


**Figure 16: A) The assembled bridge with components labeled. B) A close-up view of one of the embedded microcontroller frames. C) The microcontroller frame with a capacitive force sensor and NeoPixels.**

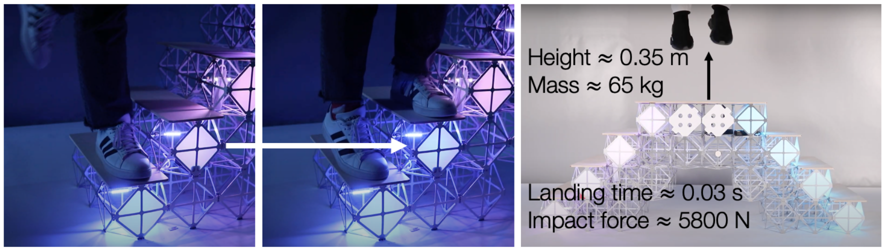


**Figure 17: The stair lights' step-response behavior and the approximate jump impact force calculation.**

The bridge is initially assembled with a 3-voxel length span, as shown in Figure 11. In this configuration, Victor tests out different materials for the stepping surfaces and finds, based on the steps logged, that wood panels result in the highest usage. The user also tests the addition of speakers and proximity sensors (as in the Chairemin) to the bridge and finds that this further increases engagement. The user then additionally notes that if the bridge length is increased by one voxel length, as shown in Figure 16, then the children can more easily crawl underneath it as well.

VIK enables these systems to be almost trivially tested and added. The reconfiguration of the bridge to span a 3-voxel gap to a 4-voxel gap is a process that only requires pliers and no additional new electrical work. In contrast, via standard fabrication and digital fabrication processes, the increase in bridge size would have necessitated a likely re-design of the system, re-fabrication of the parts, and rewiring of the structure, all of which is a time-consuming and likely wasteful process based on the scale of the structure— with VIK, the process took thirty minutes.

Even though the bridge is made from previously used voxels (from the dance floor and chair), VIK is sturdy enough— mechanically and electrically— to be used in high-load applications, taken apart, and reused for new structures. To demonstrate the load-bearing ability of the bridge, we had a person jump up and down on the top surface four times, and we had previously had a person jump on the 3-voxel gap version of the bridge three times. For one of the jumps, we then calculated the average impact force to be 5800 N based on the jump duration, landing duration, and the person's mass, as shown in Figure 17. Mechanically and electrically, the bridge withstood multiple jumps, indicating that its ultimate load-bearing capacity is much higher, which aligns with our expectations. We then removed three voxels from the middle of the structure and tested them under compression (see the Supplementary Materials for reference), recording a maximum average load of 2216 ± 29 N and modulus 24.2 ± 0.7 MPa, showing no significant difference in strength or stiffness for voxels which at this point had been used in the dance floor, chair, and bridge (details of the mechanical testing are available in the Supplementary Materials).

## 6.5 Interactive Chair Version 2

As time goes on, Victor finds themself reflecting on the Chairemin and wishing they could push the concept of an interactive chair further. The simple design of the Chairemin is overly demure compared to the silliness of the chair's function, inconsistent with the other furniture around, and not soft to sit on. In the second version, Victor sets out to correct this.

Victor first spends twenty minutes iterating chair designs in the design tool so that they can quickly explore a few very different chair concepts and then verify that their preferred cantilevered shape is also sufficiently strong to support their weight. After this, Victor prototypes the chair in situ so they can evaluate how the chair looks and functions where they intend to install it. Victor begins by building out the base and legs and then builds out the backrest and leg rest as separate modules so that they can physically evaluate what looks best and works best. Once they're satisfied

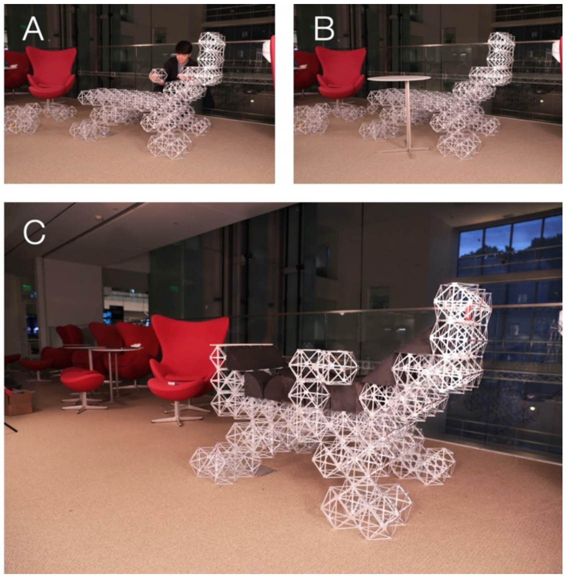


**Figure 18: A) Assembling the basic frame of the lounge chair. B) Testing chair height and comfort in situ against other furniture. C) Iterating based on that to add a table for a computer and armrest before outfitting the structure with soft foam for comfort.**

with its shape, they outfit it with the modular soft foam shapes (see Section 5 for reference) so that the lounge chair is comfortable for lounging. Because these shapes are also modular, Victor can quickly try a variety of arrangements of curves, slopes, and blocks until they find a configuration they find comfortable (see Figure 18).

After this, they outfit the lounge chair entirely with commercially available development boards and modules. Because they are unsure what functionality they ultimately want with the chair, they iterate through different IO modules and physical voxel placements: initially, they install a Sparkfun Qwiic joystick module into a low armrest, which they then later switch out for a Grove Gesture sensor. After using the sensor, they then rebuild the armrest to be more comfortable based on their actual physically tested experience, all of which takes about 10 minutes. Ultimately, they install a Grove Gesture sensor, Sparkfun PC fan controller, Qwiic speakers, and Qwiic LED strips, which are then controlled via a Sparkfun Artemis Nano, which has a Qwiic connector as well, and so can be plugged into the structure (See Figure 19). Power is supplied by a variable-voltage USB battery pack.

The lounge chair operates by inputting different gestures over the armrest, which toggle the lights in different settings, as well as control audio over a Bluetooth connection. This is designed so that one can lounge in the chair and place their laptop or phone on the stand opposite them and control it without needing to lean forward for peak comfort (see Figure 20).

In these demonstrations, hypothetical Victor can build and rebuild a series of load-bearing human-scale interactive devices using very simple tools, with confidence that the building system handles all of the tricky details that usually come with building big things. The goal of VIK isn't to be the end-all-be-all of building human/furniture-scale objects— instead, it's to enable the rapid prototyping and iteration of these structures. Currently, prototyping at this scale is much more limited than at the desktop scale, and VIK (hopefully) represents a step toward broadening access to human-scale electronics-integrated prototyping.

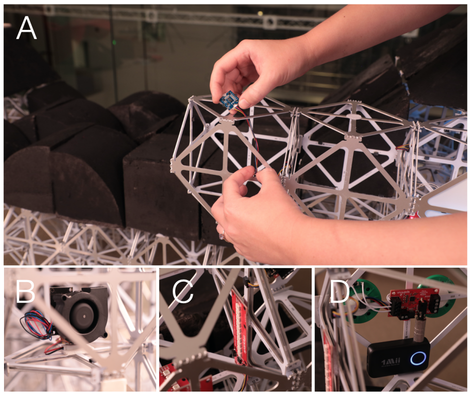


**Figure 19: A) Installing a Grove Gesture sensor via the Qwiic connectors. B) A Qwiic-controlled PC fan for ventilation, with its own dedicated battery. C) Networked LED strips for a controllable reading light. D) A Qwiic and Bluetooth-enabled speaker module for hands-free audio control.**

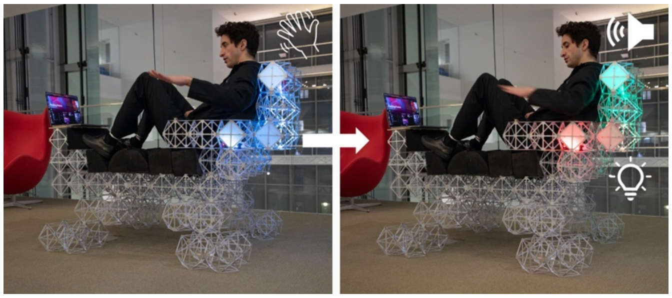


**Figure 20: Inputting gestures to change light and audio settings.**

## 7 DISCUSSION, LIMITATIONS, AND FUTURE WORK

### 7.1 Sensing and Actuating on the Lattice

The lattice structure offers potential opportunities for direct sensing integration, such as adding strain gauges or capacitive sensing into either the voxel frame of the clip system, especially with the exploration of other voxel geometries, as in [22]. Future work can open new dynamic interactive modes by adding actuators such as servos or liquid crystal elastomers [14]. Integration of such actuators would also extend the system's usability for modular robotics systems, similar to [1], or toward self-assembly systems as in [33]. Actuation that takes advantage of the lattice system could also be explored, such as in [29].

## 7.2 Optimizing Mechanical Performance and Scalability

The current mechanical performance of VIK is sufficient for it to be of interest for larger-scale applications beyond what we explore in this work. For example, given the durability of aluminum as a building material, VIK could provide an interesting basis for outdoor electronics prototyping, integrating solar panels for power. In the future, we will work on improving aspects of the system so that it is usable for commercial or industrial applications beyond a prototyping tool. Specifically, we can expand the types of networking supported by VIK beyond an I2C bus, which, while convenient for prototyping, has limited functionality at larger scales. In practice, we find that a single I2C bus on VIK supports total distances of approximately 1 m before path resistance and capacitance become limiting, which could be addressed by switching to a different network, such as CAN bus. For structures with hundreds or more devices on them, it may be beneficial to switch to a communications strategy that doesn't require addresses, such as hop-count, similar to what NeoPixels implement, or a self-discoverable network such as in [32]. Additionally, the joints and faces may be further optimized via both topology optimization approaches and user-study-informed approaches to push the performance of the system even further.

## 7.3 Increasing Accessibility

One of the goals of this work is to open the use of high-performance mechanical systems to people without engineering backgrounds or access to engineering tools. By designing the system to use aluminum PCBs, users can access VIK cheaply and quickly via commoditized PCB manufacturing processes. For this work, we produced approximately ~250 voxels, which is already a high unit count, enabled by the ease of this manufacturing approach.

Our current approach still requires users to be comfortable with Arduino-level microcontroller programming, which leaves out many people. Future work will address improving the usability of this system for electronics novices, such as by leveraging [32], a hardware virtualization system to enable easy microcontroller networking for machines.

# 8 CONCLUSION

We introduced VIK, a kit for creating large-scale, structural, reconfigurable 3D electronics. Our system uses aluminum PCBs, produced by an accessible commercial process, that assemble into lightweight, high-stiffness voxels to create a myriad of interactive devices. Because of these features, VIK has the potential to eventually scale to much larger systems than what we have so far demonstrated. We developed a design tool to help users create and simulate VIK structures, reducing the engineering expertise a user needs to deploy a high-performance mechanical system with integrated electronics quickly. The reconfigurability of the system enables users to take apart a structure and reuse the constituent parts if they no longer need the original structure without sacrificing mechanical performance. Combining the electrical and mechanical elements minimizes the amount of material that goes into VIK constructions. Though even without electronic functionality, VIK still stands as a robust and lightweight building block in its own right as a fully reusable/reconfigurable building system.

# Acknowledgments

We would like to thank Hiroshi Ishii for his support, Mengying (Cathy) Fang, Cedric Honnet, and Mustafa Doga Dogan for their feedback on the paper, and Sophie Fan for fabrication assistance. This work was supported by CBA Consortia and Media Lab Consortia funding.

# References

[1] Amira Abdel-Rahman, Christopher Cameron, Benjamin Jenett, Miana Smith, and Neil Gershenfeld. 2022. Self-replicating hierarchical modular robotic swarms. Commun Eng 1, 1 (November 2022), 1–10. https://doi.org/10.1038/s44172-022-00034-3

[2] Ernest Afari and Myint Khine. 2017. Robotics as an Educational Tool: Impact of Lego Mindstorms. International Journal of Information and Education Technology 7, (January 2017), 437–442. https://doi.org/10.18178/ijiet.2017.7.6.908

[3] Harshit Agrawal, Udayan Umapathi, Robert Kovacs, Johannes Frohnhofen, Hsiang-Ting Chen, Stefanie Mueller, and Patrick Baudisch. 2015. Protopiper: Physically Sketching Room-Sized Objects at Actual Scale. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15), 427–436. https://doi.org/10.1145/2807442.2807505

[4] Marwa Alalawi, Noah Pacik-Nelson, Junyi Zhu, Ben Greenspan, Andrew Doan, Brandon M Wong, Benjamin Owen-Block, Shanti Kaylene Mickens, Wilhelm Jacobus Schoeman, Michael Wessely, Andreea Danielescu, and Stefanie Mueller. 2023. MechSense: A Design and Fabrication Pipeline for Integrating Rotary Encoders into 3D Printed Mechanisms. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), 1–14. https://doi.org/10.1145/3544548.3581361

[5] Patrick Baudisch, Arthur Silber, Yannis Kommana, Milan Gruner, Ludwig Wall, Kevin Reuss, Lukas Heilman, Robert Kovacs, Daniel Rechlitz, and Thijs Roumen. 2019. Kyub: A 3D Editor for Modeling Sturdy Laser-Cut Objects. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, 1–12. https://doi.org/10.1145/3290605.3300796

[6] Ayah Bdeir. 2009. Electronics as material: littleBits. In Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09), 397–400. https://doi.org/10.1145/1517664.1517743

[7] Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08), 423–432. https://doi.org/10.1145/1357054.1357123

[8] Christopher G. Cameron, Zach Fredin, and Neil Gershenfeld. 2022. Discrete Assembly of Unmanned Aerial Systems. In 2022 International Conference on Unmanned Aircraft Systems (ICUAS), 339–344. https://doi.org/10.1109/ICUAS54217.2022.9836082

[9] Xiang "Anthony" Chen, Ye Tao, Guanyun Wang, Runchang Kang, Tovi Grossman, Stelian Coros, and Scott E. Hudson. 2018. Forte: User-Driven Generative Design. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 1–12. https://doi.org/10.1145/3173574.3174070

[10] Kenneth C. Cheung and Neil Gershenfeld. 2013. Reversibly Assembled Cellular Composite Materials. Science 341, 6151 (September 2013), 1219–1221. https://doi.org/10.1126/science.1240889

[11] Serena Do. 2021. serenado/proofOfDance. https://github.com/serenado/proofOfDance

[12] Kristen L. Dorsey, Sonia F. Roberts, Jack Forman, and Hiroshi Ishii. 2022. Analysis of DefeXtiles: a 3D printed textile towards garments and accessories. J. Micromech. Microeng. 32, 3 (February 2022), 034005. https://doi.org/10.1088/1361-6439/ac4fad

[13] Jack Forman, Mustafa Doga Dogan, Hamilton Forsythe, and Hiroshi Ishii. 2020. DefeXtiles: 3D Printing Quasi-Woven Fabric via Under-Extrusion. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20), 1222–1233. https://doi.org/10.1145/3379337.3415876

[14] Jack Forman, Ozgun Kilic Afsar, Sarah Nicita, Rosalie Hsin-Ju Lin, Liu Yang, Megan Hofmann, Akshay Kothakonda, Zachary Gordon, Cedric Honnet, Kristen Dorsey, Neil Gershenfeld, and Hiroshi Ishii. 2023. FibeRobo: Fabricating 4D Fiber Interfaces by Continuous Drawing of Temperature Tunable Liquid Crystal Elastomers. In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23), 1–17. https://doi.org/10.1145/3586183.3606732

[15] Zach Fredin, Jiri Zemanek, Camron Blackburn, Erik Strand, Amira Abdel-Rahman, Premila Rowles, and Neil Gershenfeld. 2020. Discrete Integrated Circuit Electronics (DICE). In 2020 IEEE High Performance Extreme Computing Conference (HPEC), 1–8. https://doi.org/10.1109/HPEC43674.2020.9286236

[16] Jun Gong, Olivia Seow, Cedric Honnet, Jack Forman, and Stefanie Mueller. 2021. MetaSense: Integrating Sensing Capabilities into Mechanical Metamaterial. In The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21), 1063–1073. https://doi.org/10.1145/3472749.3474806

[17] Christine E. Gregg, Joseph H. Kim, and Kenneth C. Cheung. 2018. Ultra-Light and Scalable Composite Lattice Materials. Advanced Engineering Materials 20, 9 (2018), 1800213. https://doi.org/10.1002/adem.201800213

[18] Ollie Hanton, Michael Wessely, Stefanie Mueller, Mike Fraser, and Anne Roudaut. 2020. ProtoSpray: Combining 3D Printing and Spraying to Create Interactive Displays with Arbitrary Shapes. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20), 1–13. https://doi.org/10.1145/3313831.3376543

[19] Freddie Hong, Connor Myant, and David E Boyle. 2021. Thermoformed Circuit Boards: Fabrication of highly conductive freeform 3D printed circuit boards with heat bending. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21), 1–10. https://doi.org/10.1145/3411764.3445469

[20] Melanie Huang. 2019. Building a DIY Dance Dance Revolution. Medium. https://medium.com/@melhuang_/building-a-diy-dance-dance-revolution-e136265bbbfc

[21] JarrodB1. Inexpensive Hard DDR/Stepmania Dance Pad Controller. Instructables. https://www.instructables.com/Inexpensive-Hard-DDRStepmania-Dance-Pad-Controller/

[22] Benjamin Jenett, Christopher Cameron, Filippos Tourlomousis, Alfonso Parra Rubio, Megan Ochalek, and Neil Gershenfeld. 2020. Discretely assembled mechanical metamaterials. Science Advances 6, 47 (November 2020), eabc9943. https://doi.org/10.1126/sciadv.abc9943

[23] Michael D. Jones, Zann Anderson, Casey Walker, and Kevin Seppi. 2018. PHUI-kit: Interface Layout and Fabrication on Curved 3D Printed Objects. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 1–11. https://doi.org/10.1145/3173574.3173684

[24] Brian T. Kirby, Michael Ashley-Rollman, and Seth Copen Goldstein. 2011. Blinky blocks: a physical ensemble programming platform. In CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11), 1111–1116. https://doi.org/10.1145/1979742.1979712

[25] Robert Kovacs, Alexandra Ion, Pedro Lopes, Tim Oesterreich, Johannes Filter, Philipp Otto, Tobias Arndt, Nico Ring, Melvin Witte, Anton Synytsia, and Patrick Baudisch. 2018. TrussFormer: 3D Printing Large Kinetic Structures. In Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18), 113–125. https://doi.org/10.1145/3242587.3242607

[26] Robert Kovacs, Lukas Rambold, Lukas Fritzsche, Dominik Meier, Jotaro Shigeyama, Shohei Katakura, Ran Zhang, and Patrick Baudisch. 2021. Trusscillator: a System for Fabricating Human-Scale Human-Powered Oscillating Devices. In The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21), 1074–1088. https://doi.org/10.1145/3472749.3474807

[27] Robert Kovacs, Anna Seufert, Ludwig Wall, Hsiang-Ting Chen, Florian Meinel, Willi Müller, Sijing You, Maximilian Brehm, Jonathan Striebel, Yannis Kommana, Alexander Popiak, Thomas Bläsius, and Patrick Baudisch. 2017. TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 2606–2616. https://doi.org/10.1145/3025453.3026016

[28] Mannu Lambrichts, Raf Ramakers, Steve Hodges, Sven Coppers, and James Devine. 2021. A Survey and Taxonomy of Electronics Toolkits for Interactive and Ubiquitous Device Prototyping. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 5, 2 (June 2021), 70:1-70:24. https://doi.org/10.1145/3463523

[29] Martin Nisser, Leon Cheng, Yashaswini Makaram, Ryo Suzuki, and Stefanie Mueller. 2022. ElectroVoxel: Electromagnetically Actuated Pivoting for Scalable Modular Self-Reconfigurable Robots. In 2022 International Conference on Robotics and Automation (ICRA), 4254–4260. https://doi.org/10.1109/ICRA46639.2022.9811746

[30] Martin Nisser, Christina Chen Liao, Yuchen Chai, Aradhana Adhikari, Steve Hodges, and Stefanie Mueller. 2021. LaserFactory: A Laser Cutter-based Electromechanical Assembly and Fabrication Platform to Make Functional Devices & Robots. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21), 1–15. https://doi.org/10.1145/3411764.3445692

[31] Florian Perteneder, Kathrin Probst, Joanne Leong, Sebastian Gassler, Christian Rendl, Patrick Parzer, Katharina Fluch, Sophie Gahleitner, Sean Follmer, Hideki Koike, and Michael Haller. 2020. Foxels: Build Your Own Smart Furniture. In Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction, 111–122. https://doi.org/10.1145/3374920.3374935

[32] Jake Robert Read, Leo Mcelroy, Quentin Bolsee, B Smith, and Neil Gershenfeld. 2023. Modular-Things: Plug-and-Play with Virtualized Hardware. In Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23), 1–6. https://doi.org/10.1145/3544549.3585642

[33] Miana Smith, Amira Abdel-Rahman, and Neil Gershenfeld. 2024. Self-Reconfigurable Robots for Collaborative Discrete Lattice Assembly. In 2024 IEEE International Conference on Robotics and Automation (ICRA), 3624–3631. https://doi.org/10.1109/ICRA57147.2024.10609866

[34] Miana M. Smith. 2023. Recursive Robotic Assemblers. Massachusetts Institute of Technology. https://dspace.mit.edu/handle/1721.1/152015

[35] Tatyana Vasilevitsky and Amit Zoran. 2016. Steel-Sense: Integrating Machine Elements with Sensors by Additive Manufacturing. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), 5731–5742. https://doi.org/10.1145/2858036.2858309

[36] Michael Wessely, Ticha Sethapakdi, Carlos Castillo, Jackson C. Snowden, Ollie Hanton, Isabel P. S. Qamar, Mike Fraser, Anne Roudaut, and Stefanie Mueller. 2020. Sprayable User Interfaces: Prototyping Large-Scale Interactive Surfaces with Sensors and Displays. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20), 1–12. https://doi.org/10.1145/3313831.3376249

[37] Jin Yu, Prabodh Sakhardande, Ruchita Parmar, and HyunJoo Oh. 2022. Strawctures: A Modular Electronic Construction Kit for Human-Scale Interactive Structures. In Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '22), 1–10. https://doi.org/10.1145/3490149.3501322

[38] Clement Zheng, Bo Han, Xin Liu, Laura Devendorf, Hans Tan, and Ching Chiuan Yen. 2023. Crafting Interactive Circuits on Glazed Ceramic Ware. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), 1–18. https://doi.org/10.1145/3544548.3580836

[39] Xiaoyu Zheng, Howon Lee, Todd H. Weisgraber, Maxim Shusteff, Joshua DeOtte, Eric B. Duoss, Joshua D. Kuntz, Monika M. Biener, Qi Ge, Julie A. Jackson, Sergei O. Kucheyev, Nicholas X. Fang, and Christopher M. Spadaccini. 2014. Ultralight, ultrastiff mechanical metamaterials. Science 344, 6190 (June 2014), 1373–1377. https://doi.org/10.1126/science.1252291

[40] Junyi Zhu, Lotta-Gili Blumberg, Yunyi Zhu, Martin Nisser, Ethan Levi Carlson, Xin Wen, Kevin Shum, Jessica Ayeley Quaye, and Stefanie Mueller. 2020. CurveBoards: Integrating Breadboards into Physical Objects to Prototype Function in the Context of Form. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20), 1–13. https://doi.org/10.1145/3313831.3376617

[41] Junyi Zhu, Yunyi Zhu, Jiaming Cui, Leon Cheng, Jackson Snowden, Mark Chounlakone, Michael Wessely, and Stefanie Mueller. 2020. MorphSensor: A 3D Electronic Design Tool for Reforming Sensor Modules. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20), 541–553. https://doi.org/10.1145/3379337.3415898

[42] 2021. I2C-bus specification and user manual. 2021, (2021).

[43] LEGO®MINDSTORMS®| Invent a Robot | Official LEGO®Shop US. https://www.lego.com/en-us/themes/mindstorms

[44] Arduino Playground - CapacitiveSensor. https://playground.arduino.cc/Main/CapacitiveSensor/