Assembling Integrated Electronics

by

Zach Fredin

B.S., Case Western Reserve University (2007)
M.E.M, Case Western Reserve University (2008)

Submitted to the Program in Media Arts and Sciences
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2021

Author . . . .
Program in Media Arts and Sciences
August 20, 2021

Certified by . . . .
Prof. Neil Gershenfeld
Director, MIT Center for Bits and Atoms
Thesis Supervisor

Accepted by . . . .
Prof. Tod Machover
Academic Head, Program in Media Arts and Sciences

Assembling Integrated Electronics

by

Zach Fredin

Submitted to the Program in Media Arts and Sciences
on August 20, 2021, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

Modern high-performance computing (HPC) systems consist of static architectures built from monolithic components. Miniaturization driven by lithographic technology has pushed Moore's Law to its limit after more than half a century, to the point that new chips require multi-billion dollar investments and supercomputer systems are built on a decades-long planning horizon. At the same time, typical HPC workloads like physical simulation have inherent geometry which is not reflected in the compute architecture, leading to a broad range of issues from cache concurrency to programming difficulty. Beyond integrated circuits, adjacent problems exist in electronics generally; printed circuit board assemblies (PCBAs) are similarly static, and the production and recycling of these products is environmentally unsustainable and requires extensive infrastructure.

The solution is to modularize electronics and autonomously assemble 3-dimensional computing structures from asynchronous, reusable elements. Of course, this concept brings with it a host of new questions: how are the devices programmed, how is communication bandwidth conserved, how do the elements physically interact, and how are the structures fabricated and assembled?

This thesis provides insight on module design and assembly automation for 3-dimensional electronics through two distinct prototype iterations. Evaluation of these systems revealed the mechanical limitations of commercial connectors, so an alternative method called digital materials is described which merges electrical interconnect and physical substrate. This method discretizes substrates into the fundamental elements that make up interconnect systems: conductive and insulating parts which are properly arranged to route signals to asynchronous processing nodes. Along the way, a novel method for constraining motion in these discrete assembly systems using modular superelastic flexures is introduced, characterized, and used to rapidly fabricate several machines.

Thesis Supervisor: Prof. Neil Gershenfeld
Title: Director, MIT Center for Bits and Atoms

Acknowledgments

Thank you, CBA colleagues, past, present, and future. Alfonso, Camron, Chris, Jiri, Dave, Jake, Erik, Amira, Ben, Will, Prashant, Sam, Pranam, Sabrina, Jack, Patricia, Filippos, Justin, Alex, Miana, Eyal: You are the smartest, most capable people I have ever had the privilege of working with.

Thank you, Neil, for creating the CBA and letting us explore the limits of digital fabrication in a world-class shop.

Thank you, Joe and Santanu, for your support and feedback as I consolidated my work into this document.

Thank you, CBA staff: Joe, Kara, Tom, John, James, and Sherry. We could not do what we do without you.

Thank you, Magneteers: Salima, Dan, Natalie, Bob. Your work is so interesting, and your machines are so cool (insert liquid helium joke).

Thank you, MAS '21 cohort: we got through this together against sometimes overwhelming odds. If you did not get a hoodie, I have the extras from Nina under my desk.

Thank you, Haystackers: Lauren, James, Amanda, Jera, Paul, Tanya, Andrea, Matt, Adam, Ellen, and many more. You have created a wonderful place.

Thank you, CBA sponsors and granting agencies, for supporting our work and holding us to account.

Thank you, MAS staff, for putting up with late-night panicked emails and helping us navigate MIT's myriad systems and protocols.

Thank you, family: Becca, Mom, Bill, Daniel, and Peter. You mean the world to me, and I can't wait to spend more time together.

Thank you, Dad, for teaching me so much. I wish you could see what I have done.

Thank you, Danica, for joining me on this wild journey and supporting me every step along the way. I love you.

Assembling Integrated Electronics

by

Zach Fredin

The following people served as readers for this thesis:

Professor Neil Gershenfeld . . . .

Director, Center for Bits and Atoms
Professor, Media Arts and Sciences
Massachusetts Institute of Technology

Professor Joseph A. Paradiso . . . .

Director, Responsive Environments Group
Alexander W Dreyfoos (1954) Professor, Media Arts and Sciences
Massachusetts Institute of Technology

Professor Santanu Chaudhuri . . . .

Director, Manufacturing Science and Engineering, Argonne National Laboratory
Professor, Civil, Materials, and Environmental Engineering
University of Illinois Chicago

Contents

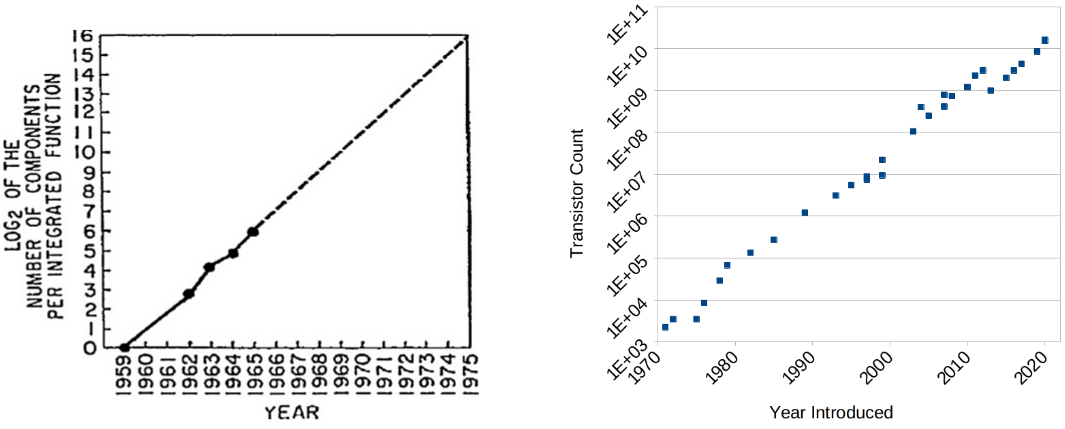
List of Figures

List of Tables

Chapter 1

Introduction

In 1965, only six years after the invention of the planar silicon transistor, Fairchild Semiconductor co-founder and future Intel CEO Gordon Moore plotted the number of circuit elements in four commercial integrated circuits (ICs) versus their release dates on a semi-log scale [58]. The resulting graph was remarkably linear, suggesting that the circuit elements in a given IC would double each year. Moore extrapolated his graph out another decade, which proved accurate if a bit modest; what later became Moore's Law held for another half-century with biennial doubling [54], as shown in Figure 1-1.



(a) Gordon Moore's original decade-long extrapolation of chip complexity [58].

(b) Moore's Law through 2021, adapted from [72].

Figure 1-1: Moore's Law, which suggests periodic doubling of IC transistor counts.

A good deal of ink has been spilled examining the push and pull of Moore's Law. Does the predictable trajectory of the last 60 years suggest that technological progress is inherently exponential? Have the applications for integrated circuits only required transistor density to double on a biennial schedule? Is Moore's Law now the motivating force behind a trillion-dollar industry, as suggested in Intel's latest quarterly earnings release ("Inspired by Moore's Law..." [43])? One trend is clear: the capital costs for building new semiconductor fabrication facilities is growing at an unsustainable rate, with 3 nm fabs costing upwards of $20 billion USD in 2017 [28], as compared to a 90 nm fab costing $2.5 billion USD in 2007 [76]. This represents a compound annual growth rate (CAGR) of 23%.

In order to increase the number of transistors on a single chip from thousands in the 1970s to millions in the 1990s to billions in the 2010s, manufacturers could have taken one of three paths: increase the overall chip area; reduce the size of each transistor; or start building chips upwards into 3-dimensional space. While modern chips are larger than their early ancestors, and some designs use limited 3D features, Moore's Law is mostly the result of the ongoing shrinking of transistors, as seen in Figure 1-2.

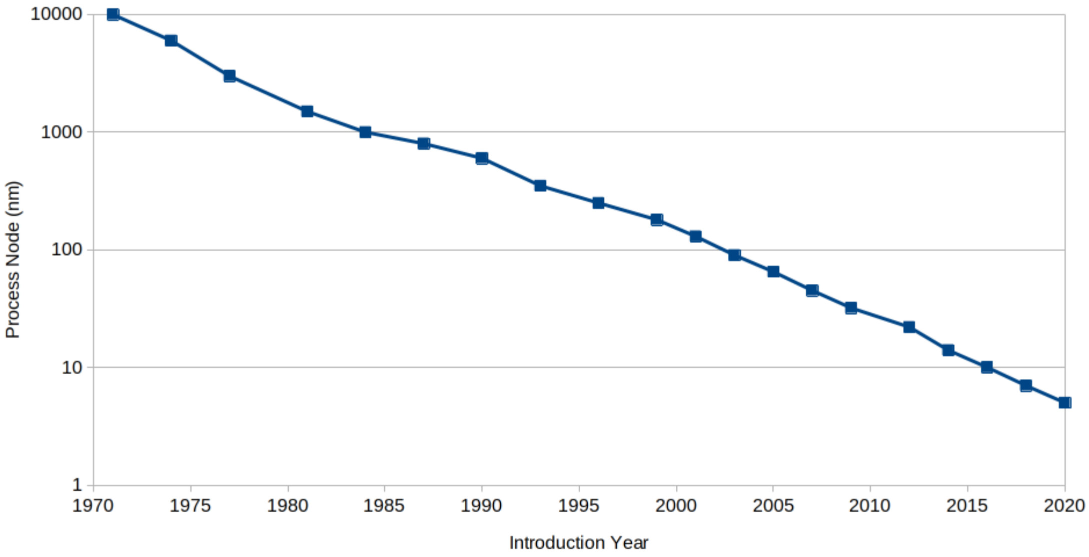


Figure 1-2: Semiconductor process nodes vs introduction year, graphed on a semi-log scale [14].

Transistor miniaturization is quickly reaching its practical limits. Smaller lithographic techniques require shorter wavelengths of light, which means even exotic extreme ultraviolet (EUV) sources [19] are nearing their resolution limit. Structures that may only be a few dozen atoms across are hard to make without defects at scale, one reason for the recent multi-year delays bringing new fabs online[43] [15]. In order to continue the half-century march towards ever-increasing computational density, a novel approach is needed. The work presented here is an early step in this direction, and focuses on reversibly assembling true 3-dimensional computational structures, where the incremental cost to adding layers increases linearly as a simple function of material cost. These structures can then serve a wide variety of uses, from short-run desktop prototyping to large scale physics simulation.

The remainder of this introductory chapter briefly surveys the modern electronics and computing landscape with an eye towards scaling, cost-control, and environmental stewardship, and ends with a summary of this thesis' contribution. Chapter 2 introduces Discrete Integrated Circuit Electronics (DICE), an implementation of reconfigurable asynchronous 3D computing. Chapters 3, 4, and 5 focus on physical DICE module design and assembly, while Chapters 6 and 7 explore novel methods for constraining motion in DICE assembly systems. Chapter 8 concludes, and provides a roadmap for future research.

1.1 Chip Fabrication

Modern semiconductor fabrication is a complex, multi-step lithographic process that takes place in highly controlled clean rooms. The majority of devices are silicon-based, although other semiconductor materials such as gallium arsenide (GaAs) or gallium nitride (GaN) are used for specialized applications such as light-emitting diodes (LEDs). In order to function as a predictable substrate with stable electrical properties, the material must first be grown into a perfectly ordered single crystal called a boule, and then sawed into wafers which are polished flat. Typically, the boules are grown from a pure melt using the Czochralski method [86] or progressively solidified from a polycrystalline precursor in the Bridgman-Stockbarger method [61]. During the growth process, minute impurities such as boron are added to the melt to dope the silicon.

Once wafers are produced and polished, a variety of steps are performed to build up electronic structures. These steps can be additive, where new material is added to the wafer through deposition or oxidation; subtractive, where material is selectively removed using an etchant or mechanical method; electrical, where the substrate's electrical properties are changed using ion implantation to locally dope the material; or process-related, such as patterning a photoresist using UV light to mask a subsequent etching step. Between many of these steps the wafers must be carefully washed with ultra-pure water, requiring modern fabs to use 2-4 million gallons per day of the resource [24]. While the number of steps and layers varies dramatically with chip complexity, a typical application processor may have 10 distinct layers and take 300 steps to fabricate over the course of several months.

After the final nano-scale features are etched, implanted, or oxidized into existence, the wafer consists of many identical rectangular devices on a single round substrate. Using a wafer probing apparatus, each device is electrically tested and the results recorded. Device yields vary dramatically depending on process node, device complexity, test stringency, and fab experience; one manufacturer reported 5 nm SRAM test chips being produced at 80% yield [93], but extrapolating these values suggest device yields of physically larger chips may be much lower [26]. In some cases, testing is used to characterize performance grades, allowing identically designed devices to be binned by quality and sold in tiers.

Finished and tested wafers are singulated into individual chips using a scribe, diamond saw, or laser. Individual dies are then packaged using a method dependent on the end user's requirements and technical capabilities. The largest integrated circuit packages were originally designed for through-hole electronics in which component leads are inserted through printed circuit boards (PCBs) and soldered. These ceramic or plastic packages, called DIPs (for Dual-Inline Pin), have leads at 2.54 mm pitch and are still used for prototyping with spring-terminal breadboards. Smaller surface-mount versions of DIPs, called SOICs (for Small Outline Integrated Circuit, 1.27 mm pitch) and SSOPs (for Shrink Small Outline Package, 0.65 mm pitch), are still seen on modern electronic devices. DFNs (Dual Flat No-Lead, 0.5 mm pitch) and QFNs (Quad Flat No-Lead, 0.5 mm pitch) reduce size by replacing the leads with flush tinned terminals, but still only electrically interface at the package perimeter. As processors have become more sophisticated, this dimensional interconnect limitation was solved by so-called grid arrays, where an entire face of the chip is covered in a regular pattern of through-hole pins (in a PGA, 2.54 mm pitch) or reflowable solder balls (in a BGA, 0.5 mm pitch). The Wafer-Level Chip-Scale Package (WLCSP, 0.4 mm pitch) extends this further by eliminating the ceramic or plastic package entirely, and applying the interconnect solder balls to the minimally-protected die directly.

1.1.1 The Yield Problem

Device yield is fundamental to fab economics. The Murphy model [60] begins by relating the overall yield of a monolithic device Y_N to the individual yields of each component (i.e discrete transistors on a given chip) Y_1 and the number of components N:

$$Y_N = Y_1^N$$

This equation alone quickly points engineers towards minimizing N to maximize yield, so Murphy introduces several other factors that push optimal per-chip component counts higher. These factors include the difficulty in separating and handling extremely small devices made with only a handful of components, and the substantial expense of off-chip wiring that can be integrated with monolithic devices. In the 1960s, fabrication limitations meant the lowest per-component cost can be achieved with an N value of 20 and a physical area of 30 mil square. Later yield models [22] combined both point defects and so-called parametric yield losses which affect entire wafers. And modern processors such as Apple's M1 have more than ten billion transistors [68], demonstrating half a century of steady fab yield improvements. Still, a clear negative correlation between device yield and overall device size exists.

One extreme exception is the fabrication of far larger chips that take up an entire wafer, as is the case with wafer-scale integration [44]) Cerebras' Wafer Scale Engine 1, with 1.2 trillion transistors and 400k cores, is 56 times larger than the largest GPU on the market. It is not known how Cerebras pushes the yield curve far enough to make such devices economical; one can speculate it is a combination of excellent process control, inherently fault tolerant design, and relaxed binning strategies. The Wafer Scale Engine benefits from fast on-chip interconnect and extremely local memory resources available to each core, particularly for some deep-learning applications. But even so, massive dies are delicate and awkward to package, and (like all chips) such devices are architecturally static after fabrication.

1.1.2 Chiplets

Chiplets represent another approach to increasing complexity by discretizing formerly monolithic devices into configurable 2-dimensional blocks [47]. These blocks are then permanently placed on a high-density interconnect called an interposer, which is another fabricated silicon chip that can be fabricated using lower-density design rules. Dividing large single devices into subsystems like this positively impacts yield, since for a given technology node the required defect-free area is smaller. Chiplets also allow physically closer integration of disparate devices, such as FPGAs and microprocessors, which is advantageous for high-bandwidth communication [55].

Another important advantage of this strategy is that chip developers can focus on their core competencies, and use intellectual property (IP) licensing agreements to reuse chiplets from other vendors in their designs. Given standardized interfaces and specifications, one could imagine a marketplace of chiplets performing similar functions, where competitive pressures push costs down and encourage innovation. This type IP reuse and licensing is already prevalent in FPGA development, where vendors often provide developers with standard blocks to add common functionality such as USB or HDMI connectivity to a design [12]; chiplets merely extend the concept into the physical realm.

Chiplets, like System-on-Module (SoM) designs which perform similar functions but use conventional PCBs as interposers, are still architecturally static after assembly, meaning they cannot be reconfigured after packaging to fit novel problems. And chiplets are inherently 2-dimensional systems (technically up to 2.5-dimensional, when several chiplets are layered [87]), where power and data must flow from a single face, down to an interposer, laterally to another chiplet, and up to the device. Compared to a truly 3D system where all devices directly communicate with their nearest neighbors, this means chiplets must maintain communication bandwidth over a longer distance, while at the same time providing higher routing density since some elements may need to communicate with systems further away. These limitations strain interposer technology, requiring through-silicon vias and microwires that ultimately limit the complexity of systems [79].

1.2 Electronics Assembly

Once integrated circuits are designed, fabricated, and packaged, they must be further integrated into systems prior to use. Typically, ICs are soldered to PCBs along with myriad support components. For example, most chips require extremely stable power supplies, since even a brief sag in supply voltage can cause a fault or reset. Thus, most systems include dedicated power supplies that convert noisy external power (whether from a wall socket or a power bus) into one or more carefully regulated supply rails. At minimum, power supplies require a regulating device, such as a low-dropout regulator (LDO) or switching power controller, along with several passive components such as capacitors and inductors. Other ancillary equipment includes RF transceivers (Bluetooth, WiFi, cellular, etc) for communicating with off-board networks; audio circuits for amplifying signals for headphones or speakers; external flash memory for storing configuration data or logs; user-operated controls such as switches and encoders; and electrical connectors for delivering power and data to the device.

1.2.1 PCBs

The vast majority of integrated electronic systems are built on PCB technology. PCBs serve as an intermediate interconnect that allows electrical designers to use monolithic integrated circuits in flexible applications without needing access to semiconductor fabrication facilities. Physically, a PCB is a stack of planar conductors and insulators, glued together into a flat rigid or flexible sheet. The top conductive layer is coated in gold or tin, allowing components to be easily soldered to the surface. Portions of the top layer along with additional conductive layers are selectively etched away during fabrication, leaving isolated conductive nets which allow the designer to electrically connect components at will. The nets are joined in the vertical direction using vias, which are physical holes through the insulating substrate that are plated during processing to allow current to travel between layers.

With these essential characteristics in mind, actual PCBs vary drastically in cost, complexity, and construction. At the lowest end, single-sided PCBs are made from 1.6 mm rigid phenolic FR1 sheet with a layer of 35 um copper foil glued to one side. Nets are milled into the copper sheet using 1/32" and 1/64" end mills on a desktop router. Rather than plating the copper, hand-soldering techniques are used along with flux to secure components to the PCB. This method, while primitive, is readily accessible with minimal equipment and is ideal for rapid circuit prototyping. Commercially, most circuit boards use fiberglass FR4 sheet as a substrate, and the patterns are etched rather than milled into the copper layer using photolithography techniques followed by acid etching. Such boards may have net traces as narrow as 75 um, and as many as 64 distinct insulating and conducting layers (although the vast majority are less than 10 layers).

The economics of modern PCB fabrication technology are interwoven with integrated circuit design. As discussed earlier, increased processor complexity has pushed chip designers to increase the density of off-board electrical interconnect, from DIPs (2.54 mm spacing, half perimeter) to QFNs (0.5 mm spacing, full perimeter) to WLCSPs (0.4 mm spacing, 2D array). This means there must be more signal wires reaching each chip across the 2D surface of a given PCB layer. Increasing trace density by reducing trace width requires higher precision lithography machines and better etchant control. Similarly, smaller vias require tiny drill bits which operate slowly and are subject to breakage. Each additional layer adds an opportunity for failure caused by misalignment, contamination, or handling damage. All of these factors result in electronic assembly cost and complexity tracking increasing chip transistor count.

1.2.2 Picking and Placing

Once PCBs are prepared, components must be affixed to one or both sides. This is done by soldering, in which minute quantities of a conductive alloy with a low melting temperature permanently joints parts to pads. Historically, and in prototyping, this is done by hand with a soldering iron and flux-cored solder wire. In a production environment, solder paste and reflow techniques are used instead.

Solder paste is a spreadable mixture of minute solder balls mixed with gel flux. When the circuit board is fabricated, a matching stencil is also produced which includes apertures over the pads where components will be placed. The stencil is lined up on the PCB and solder paste is forced into the holes, leaving a pattern of flat paste-covered areas after the stencil is removed. Next, an automated machine called a pick-and-place pneumatically lifts components off of reels and precisely places them on the PCB, using an integrated vision system to correct alignment as needed. Once all of the parts are placed, the board is carefully transported to a reflow oven where the ambient temperature is slowly raised following a programmed profile. As the temperature increases, the flux boils off, cleaning any residual contamination off the pads and components. Next, the solder balls melt and coalesce, forming robust connections between the components and the board. The reflow oven holds the peak temperature for a time to ensure complete solder melting before gradually cooling to room temperature. At this point, the PCBA (for -Assembly) can be cleaned, tested, programmed, and further packaged into a complete system.

1.2.3 Breadboards

An interesting corollary to the ongoing miniaturization of IC packages is that fast reversible prototyping methods have not kept up. When DIP ICs and other through-hole components reigned, platforms called breadboards were commonly seen on workbenches for quickly testing circuit designs. Breadboards thick plastic plates with a grid of holes sized to accept DIP IC pins or passive component leads. The holes are spaced identically to ICs (2.54 mm) and have metal spring retainers at their base, such that items inserted into the breadboard are firmly retained but removable with additional force. Each spring retainer spans several holes, effectively creating a regular array of static nets which can be used to electrically connect components. Most breadboarding kits include a quantity of pre-stripped and bent wires of various lengths, so that users can quickly connect non-adjacent components.

Of course, anyone who has recently toured an electrical engineering department, a FabLab, or a widget company workspace has probably spotted plenty of breadboards hard at work testing circuit designs. While modern ICs are one-fifth the size of their DIP ancestors, breadboards have survived through the proliferation of breakout boards, adapter PCBs that convert tiny modern components into pseudo-DIP devices with breadboard-compatible 2.54 mm pin spacing. Some good has come from this ongoing trend; breadboards are cheap, easy to use and understand, and they enable backwards prototyping compatibility with half a century worth of electronic parts. Breakout boards are often open-source with robust community support, one key to the explosive growth of the electronics-focused parts of the maker movement.

The main reason breadboards are still in common use is that they are still the best, and only, method for reversibly prototyping electronic circuits. While some of this is due to the inertia of 2.54 mm pin spacing, the switch from through-hole to surface-mount components is mostly to blame. Surface mount devices have minimal (or no) leads, and as compared to standard DIP spacing the lead pitches vary dramatically between components. Even a given package configuration, such as QFP, is available in many pitches (0.8 mm, 0.5 mm, occasionally 0.4 mm) which would need to be accommodated with mechanical adapters. And surface mount devices are really designed for automated pick-and-place production, so they are often difficult to handle.

Unfortunately, breadboards have a few problems. First, as discussed above, they aren't actually sized for modern electronic components, limiting their ability to fully utilize modern processors. For example, the popular Teensy 4.0 prototyping breakout board uses the NXP MIMXRT1062DVL6A Arm Cortex M7 chip, which is packaged in a 10 x 10 mm BGA with 196 balls, 127 of which can be used as general-purpose input-output (GPIO) lines [77]. However, the breakout board only provides breadboard access to 24 GPIO lines, despite taking up over five times the area of the core processor. Second, the long unshielded connectors which bus signals are unsuitable for high-speed electronics; while the exact limit depends on many factors such as wire length and adjacent signals, general design guidelines suggest breadboards shouldn't be used above 10 MHz. Such speeds are suitable for low-speed sensors and small displays, but fall short when routing closely coupled RAM chips or high-definition displays. Third, breadboards are rather delicate, particularly when connected to numerous off-board components; the plague of makers is the breadboarded circuit that functions properly on the workbench, but fails when transported to Maker Faire. And finally, breadboards hold a good deal of difficult-to-recover state, as seen in Figure 1-3. Even with a detailed photograph, it is hard to determine exactly how a breadboard is wired, so without careful documentation the results of a breadboarded test can be easily lost.

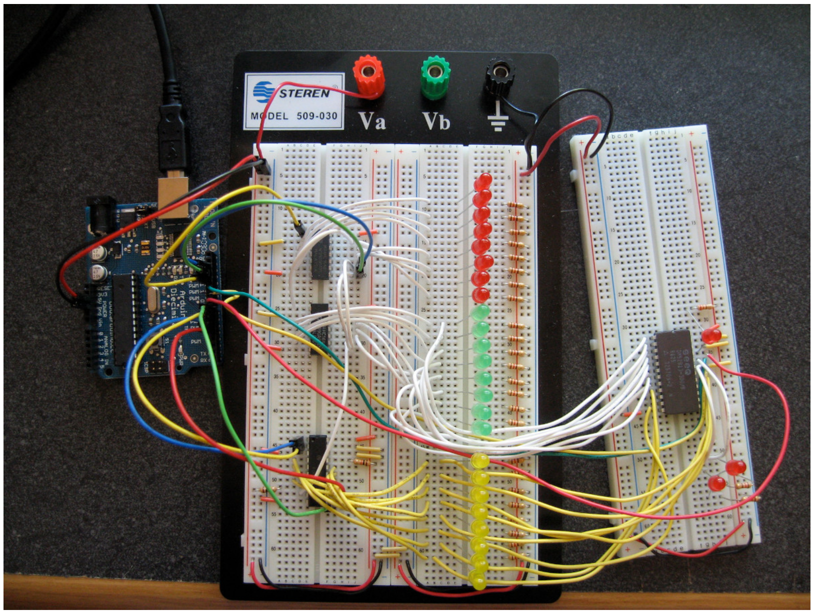


Figure 1-3: A typical breadboard in use, demonstrating the difficulty inherent in reproducing or repairing work prototyped using this method [59].

1.2.4 Environmental Footprint

A troubling side effect of PCB fabrication is water pollution. As discussed above, PCB production uses photolithography, which requires bare boards to be treated with various chemicals to mask and expose traces for subsequent etching. The etch process historically used ferric chloride; other compounds can also be used to selectively remove copper from the PCB substrate. After etching and washing, outer surfaces are plated, usually either with tin or gold (which includes a nickel plating step to improve adhesion to the copper base). Boards are also coated in solder mask, which prevents solder from sticking to exposed traces.

Most of the aforementioned chemicals are aqueous, and in between many process steps (such as etching and plating) the PCBs must be thoroughly washed. Beyond PCB fabrication itself, the component placement process also uses some water to wash residual solder paste off assembled boards. While numbers from industry sources are difficult to find, various studies [39] have suggested that PCB fabrication uses on the order of 1000 L per square meter of board production. In other words, immersing a PCB in a column of water one meter tall shows roughly the amount of water polluted during the production cycle.

Industrial wastewater treatment is one clear answer to PCB production water use. Removing metals from wash water and neutralizing pH prior to discharge (or water reuse) is fairly trivial at industrial scale. However, since the infrastructure and raw materials needed for mitigation are far from free, an economic incentive exists for manufacturers to avoid water treatment. Such incentives can be counteracted by reasonable regulations, such as those outlined by the US EPA's Clean Water Act, but are dependent on local governmental oversight. Unfortunately, production globalization has far outpaced regulation, leading to the current reality where many PCBs are fabricated without proper consideration for environmental effects. A better solution is to find alternatives to aqueous processing entirely.

1.3 High-Performance Computing

Early computer systems used a mainframe and terminal model, where multiple users shared time on a centralized platform. Such architecture was borne of necessity; logic elements (whether they be relays, vacuum tubes, discrete transistors, or ICs) and memory (magnetic cores, punch cards, etc) were simply too large, inefficient, and expensive to provide dedicated systems for each user. As Moore's Law continued, minicomputers from DEC [21] and others enabled small workgroups to share systems, and personal computers (PCs) eventually gave us the 1:1 (or greater) computer:user ratio we enjoy today. A fundamental reason for this shift is that economical computational power now exceeds most user's needs, making it fiscally reasonable for a typical desktop workstation to rarely reach its performance limit (and, in most cases, sit idle when users are not physically present).

High-performance computing (HPC) refers to systems designed around a subset of problems which continue to benefit from increased computational power. While problem types vary dramatically, many are based around modeling the real world. For example, in one study of the ORNL Kraken workload over one year [94], the vast majority of users came from academic disciplines focused on the physical sciences. More than 3/4 of the studied compute jobs were from atmospheric science, molecular bioscience, chemistry, materials research, or physics groups. While the specific problems likely varied dramatically in substance, it is reasonable to assume that many users were simulating real-world phenomena to predict its effects and compare them to observation.

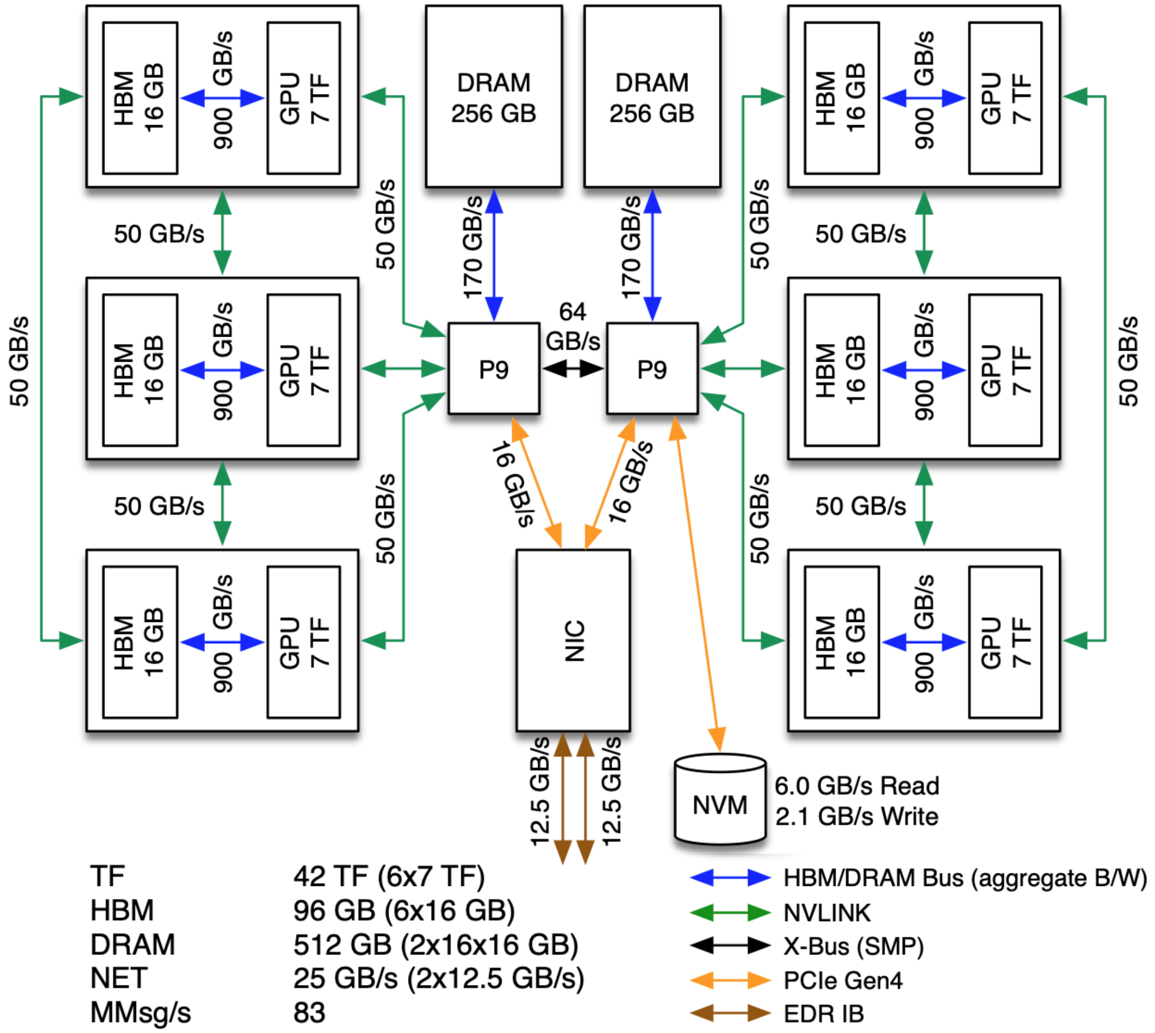
Modern HPC systems include national-scale supercomputers, which are conceptually similar to last century's mainframes. Physically, a large HPC system may take up one or more floors of a building, weigh tens or hundreds of tons, cost many millions of dollars, and take a decade to develop and deploy [56]. Users share time on the system, which is often set up to accommodate several problems simultaneously; and unlike PCs, the system runs constantly, with controls and scheduling systems designed to maximize uptime.

1.3.1 Architecture

One of the key decisions during the HPC design process is how processors and memory are connected to one another. Oak Ridge National Labs' Summit [10] is a recent example of an HPC system that demonstrates the trade-offs system designers must make to accommodate memory and interconnect bandwidth and core processor speed.

Summit consists of roughly 4,600 nearly identical nodes housed in a room-filling grid of 19" server server cabinets. The nodes, officially IBM Power System AC922s [3], are each capable of roughly 40 TF (TeraFlops, or billion floating point operations per second). Nodes are physically distinct 30 kg devices, with redundant 2200-watt power supplies, water and air cooling systems, and physical enclosures allowing them to be removed and serviced or replaced. In addition to power, each AC922 has rear-mounted hose ports to connect to facility-provided cooling water, along with network connections that support 23 GB/s bandwidth via a protocol called EDR InfiniBand. Cabinets include InfiniBand switches that aggregate their installed nodes; multiple cabinets are then linked using higher level Director Switches to form the complete computing cluster.

A pair of IBM POWER9 processors control each Summit node. These 22-core devices are interlinked with a 64 GB/s connection, and they each connect to the InfiniBand controller using a dedicated PCIe Gen4 16 GB/s link. The processors have their own dedicated 256 GB DDR4 memory modules connected at 170 GB/s. Each processor is further connected to three NVIDIA V100 GPU accelerators via 50 GB/s NVLINK; each accelerator then has its own dedicated 16 GB HBM2 memory, connected at 900 GB/s. A node diagram is shown in Figure 1-4.



TF	42 TF (6x7 TF)
HBM	96 GB (6x16 GB)
DRAM	512 GB (2x16x16 GB)
NET	25 GB/s (2x12.5 GB/s)
MMsg/s	83

Figure 1-4: Diagram of one ORNL Summit AC922 HPC node [10].

1.3.2 The Programming Disconnect

HPC systems are architected hierarchical behemoths whose internal structure is inherently static. Importantly, while HPC users and system designers may attend the same conferences, they generally are not the same people; and given the decade-long planning horizon for new HPC installations, it is unlikely that a given user's specific problem informed the overall system architecture. Of course, this disconnect is not unique to supercomputers; single-threaded desktop PCs are usually designed and programmed in different buildings (or companies) as well. And so programmers use compilers and interpreters so they can write abstracted code, like C or Python, and processors use common instruction sets so they can be readily programmed directly in assembly language without system design-specific knowledge.

For national-scale HPC systems like Summit with tens of thousands of cores, the analogs to abstract programming language are MPI and CUDA. MPI, for Message Passing Interface, is a communications protocol that defines how distributed nodes in a parallel computing system exchange information [6]. MPI is based around local memory that is not shared system-wide, and allows programmers to handle synchronization between compute threads in an abstract manner that is decoupled from physical processing node demarcation. CUDA, an NVIDIA application programming interface (API), extends this abstraction to GPUs, such as the six V100s in each Summit node [13]. Both MPI and CUDA are practically used in conjunction with familiar languages like C++. Unfortunately, there is not an "assembly language" for HPC systems. While users could theoretically write machine code for individual cores, the multiple layers of connectivity paired with a typical HPC system's workload of many simultaneous jobs from different users makes this impractical at any substantial scale.

Programming abstraction hides static architecture which inherently conflict with one of the most common HPC workloads, multiphysics simulation, because problem geometry is not reflected in compute geometry. Most physics is local; for example, the molecules that make up a fluid only interact with their neighbors. But the computational fluid dynamics (CFD) [89] model of a fluid system must keep track of all of the particles in a central synchronized database. Clearly, a more efficient model would allow nodes to only track particles to the range of their interaction distance. But this would require the computational geometry to align with the problem geometry, a virtual impossibility given the breadth of problem types, the disparate realms of HPC designers and users, and the fundamentally static nature of computational architecture.

1.4 Contribution

This thesis presents a number of steps towards asynchronous reconfigurable 3D computing systems, outlined in the following chapter. The core work presented in Chapters 3-7 focuses on overcoming the physical challenges associated with reconfigurability, and lays the groundwork for future system iterations. Along the way, a number of novel concepts are explored, including an electronics prototyping method that goes beyond breadboards, and a modular system for building practical motion systems using flexural linkages rather than conventional sliding or rolling elements.

Chapter 2

Discrete Integrated Circuit Electronics (DICE)

Monolithic processors present a fundamental limitation to computation due to the relationship between die size and fabrication yield. This observation is not novel; all high-performance computing systems spread tasks among a multitude of processing elements working simultaneously on different parts of a given problem. Building heterogeneous systems is physically advantageous, since packaging and interconnect can be designed around thermal management and fab capabilities. But such systems are inherently static, meaning their architecture does not reflect the geometry of a given problem set. Discrete Integrated Circuit Electronics [33] [50], or DICE, is a radical re-examination of how computation can reflect geometry by making reconfigurability a foundational part of system architecture.

A DICE system consists of many distinct but identical processing nodes arranged in a 3D lattice. The nodes provide structural support to one another, and share electrical connections that provide power and local communication channels. Like physical particles such as molecules interacting in the real world, DICE nodes are fully asynchronous, and interchange information through data tokens which are produced and consumed by adjacent devices. Crucially, DICE structures can be disassembled and reconfigured to suit new problem sets, and the physical assembly infrastructure is integral to the functionality of the overall system. External data can be sent serially to individual nodes to pass on, or in parallel to a lattice face using a suitably equipped build surface.

2.1 Prior Work

While the combination of asynchronicity, reconfigurability, and three-dimensionality is unique to DICE, all of these concepts have been previously explored in the context of computing. A non-exhaustive summary of several projects that lead up to DICE is presented here.

2.1.1 Project Tinkertoy

Project Tinkertoy [45] was a project started in 1953 by the National Bureau of Standards to automate the production of electronic assemblies. The system consisted of many identical ceramic wafers that each carried a few passive components, such as resistors, capacitors, or inductors. The wafers had etched grooves that were filled with silver ink, which was then sintered into conductive nets analogous to modern PCB traces. After fabrication, wafers were assembled into stacks with vertical interconnecting wires, and often topped with a wafer equipped with a socket to receive a vacuum tube. While some final assembly steps required limited manual intervention, the Project Tinkertoy workflow was designed to be highly automated.

Remarkably, Project Tinkertoy came about before the widespread adoption of semiconductor logic devices such as transistors, and was not a modular computing system. The concept was instead motivated by the long development time for electronic assemblies; in particular, project supporters were concerned that it would take too long for the military to reactively develop new systems during wartime. Project Tinkertoy went through at least two hardware iterations during the late 1950s, including a follow-up effort with RCA in 1957 [46], but ultimately did not keep up with transistor- and IC-driven miniaturization trends.

2.1.2 Neuron Simulators

Neurons are biological cells that humans and animals use to process information. While neurons vary dramatically by function, they all have inputs, called dendrites, and an output at the end of a structure called the axon. Neurons maintain an electrical voltage potential across their membranes using ion channels and pumps. As a neuron's dendrites are excited by other neurons or sensory stimuli, the membrane potential increases according to the relative weighting of the signaling dendrite. If the membrane potential exceeds the neuron's action potential threshold, the neuron fires and sends a signal down its axon to other neurons or output systems like muscles.

Neurons are naturally occurring asynchronous computing systems. Their independence was originally postulated by Santiago Ramón y Cajal near the end of the 19th century, who observed and drew neurons after staining them with Gogli's method [80]. Several decades later, Otto Schmitt built the first neuron simulator as part of his PhD work at Washington University [74]; as part of this work he also invented the now-ubiquitous Schmitt trigger, which unfortunately overshadowed his contribution to neuroscience. Much later, I started a company building electronic neuron simulators for educational purposes that included a number of sensory and motor devices so students could build complex asynchronous systems on their desks [65].

While neuron-like models called perceptrons [73] were used to further neuroscience research, their use as computational building blocks was never practical when compared to boolean transistor-based logic systems.

2.1.3 Electronic Digital Materials

More recently, Langford [49], Hiller [42], Ward [88], Popescu [67], MacCurdy [53], and others introduced the concept of electronic digital materials. These 3D lattices of conductive, insulating, and resistive elements are composed into functional electronic structures using a purpose-built assembly robot. Langford later proposed adding computation [50] nodes for robotic control, and prototyped early physical devices that later evolved into DICE. Notably, Langford's 2019 PhD thesis was the first document to use the term Discrete Integrated Circuit Electronics. As will become apparent in subsequent chapters, discretizing electronic components such as PCBs and connectors into insulating and conductive elements is beneficial beyond simply reducing part inventories to primitives based on physical properties.

2.2 Performance Projections

Asynchronous reconfigurable 3D computational structures like DICE have apparently stark advantages, such as the capacity to scale beyond physical clock phase limits, and clear issues, such as the added complexity and bandwidth loss of intra-node interconnect. These opposing characteristics can be quantified together to reveal optimal system configurations for a given problem space. Specifically, where should the demarcation between inter- and intra-node exist? At one extreme, each node is a simple ALA [16] [36] device, performing single-bit operations on single-bit tokens. At the other extreme, the entire structure consists of a single massive internally synchronous node, analogous to a modern HPC system. It is informative to first consider a computing model in purely abstract terms, then bound it with a specific problem space, then add dimensionality to the configuration (i.e. 1D, 2D, and 3D lattices), and finally add real-world numbers based on currently available devices and near-term fab capabilities. From there, it is possible to project performance of a DICE structure on simple benchmark problems and compare it to current systems.

2.2.1 Model

Consider a computational structure build from asynchronous token-passing modules with a simple single-threaded internal processing architecture. On every clock cycle, each module can perform one of several tasks:

1. Perform a computation.
2. Store a value in local memory.
3. Retrieve a value from local memory.
4. Send a data token to a neighbor.
5. Receive a data token from a neighbor.
6. Do nothing.

To further simplify the model, one can assume memory retrieval is instantaneous; token-passing is bidirectional; and doing nothing is optimized out; thus:

1. Compute
2. Communicate

Given computation and communication cannot happen simultaneously, and only computation furthers completion of the problem (i.e communication is overhead), the computational efficiency E of the system can be described as:

$$E = \frac{T_{COMP}}{T_{COMP} + T_{COMM}}$$

where T_{COMP} is time spent on computational tasks, and T_{COMM} is time spent communicating with adjacent nodes.

It is well known that intra-chip electrical communication bandwidth exceeds inter-chip bandwidth by a substantial factor. This is due to several factors including physical proximity and lower transmission line capacitance. So it makes sense that internal computation can be clocked at a far higher speed than node-to-node communication. That means if the processor must send a bit with every computational cycle, communication time will dominate and the computational efficiency E will plummet. The application being run on the processor is thus important to E; an example problem space, physical simulation, is analyzed later in detail.

2.2.2 Communication Peripherals

First, it is helpful to examine simpler ways to improve E without changing communication speed or compute time, using a commercially available chip used for prototyping in later chapters as an example. The Microchip ATSAMD51J20A microcontroller has an ARM Cortex M4F core that runs at 120 MHz and can be reliably clocked upwards of 160 MHz, while its six built-in UART peripherals are only capable of 3 Mbps communication [1]. In other words, performing a single computational operation is always far faster than sending a single bit of data to an adjacent device.

In order to accommodate the difference in computation and communication speeds, the model can be iterated to include communication peripherals. The aforementioned ATSAMD51J20A microcontroller, along with most microcontrollers generally (and microprocessors, via off-chip devices), have a separate subsystem dedicated to managing intra-device communication. This peripheral mediates the speed differential between the faster core processor and the slower physical communication channel, allowing the processor to work on other tasks while the peripheral manages chatter with off-chip devices. In practice, this means the processor only needs to spend a single computational cycle to fill a local buffer with a data token. On the receiving end, the buffer independently receives data and flags the processor only when the buffer is full, at which point the processor spends another cycle to retrieve the data.

Again taking the ATSAMD51J20A as an example, assume a given problem takes 100 computational cycles to complete, and produces 10 bits of data that must be exported at the end. Without a communication peripheral, the computational portion would take:

$$T_{COMP} = \frac{1}{160MHz} \times 100cycles = 6.3 \times 10^{-7}s$$

while communication would take:

$$T_{COMM} = \frac{1}{3MHz} \times 10cycles = 3.3 \times 10^{-6}s$$

giving a computational efficiency of:

$$E = \frac{6.3 \times 10^{-7}s}{6.3 \times 10^{-7}s + 3.3 \times 10^{-6}s} = 16\%$$

Conversely, if the processor was able to offload the result to the communication peripheral at full speed (but still one bit at a time):

$$T_{COMM} = \frac{1}{160MHz} \times 10cycles = 6.3 \times 10^{-8}s$$

and thus:

$$E = \frac{6.3 \times 10^{-7}s}{6.3 \times 10^{-7}s + 6.3 \times 10^{-8}s} = 91\%$$

Clearly, this abstraction glosses over many details; the ATSAMD51J20A is a 32-bit microcontroller, for example, and there is some overhead involved in starting the communication peripheral and moving data to and from memory. It is also notable that if there were 100 bits of data to transmit and the calculation were performed repeatedly, the communication buffer would overflow since the peripheral wouldn't be able to keep up with the processor. This introduces an important inequality, which relates communication time T_{COMM}, processor speed S_{COMP}, communication peripheral speed S_{COMM}, and compute time T_{COMP}:

$$\frac{T_{COMM}}{T_{COMP}} <= \frac{S_{COMM}}{S_{COMP}}$$

As long as this inequality holds true, the processor will remain saturated and the communication peripheral will idle (or saturate, when the expressions are equal). Normalized to communication speed, devices should spend more time "doing" and less time "talking" to remain at peak efficiency. Put another way, as communication speed deviates from processor speed, each module should increase its computational workload with respect to its communication. Importantly, the physical electrical limitations of intra-module communication (as compared to inter-module) is analogous to the slower rate of a communication peripheral, even if a given DICE node is not so equipped. This fundamental notion can be used to scale DICE nodes to problem sets and vice versa, as will be shown later in this chapter.

2.2.3 Physical Modeling

Most classical physical modeling, be it finite element analysis applied to a structure or computational fluid dynamics applied to a turbine, is concerned only with local interactions. This is because the force laws that govern classical physics are also local; atoms push against one another in a crystal lattice, but they do not directly affect atoms beyond their immediate bonds. Fluids, likewise, consist of particles interacting locally and transmitting forces over distance indirectly. This observation, that physics is local and asynchronous, is central to DICE as applied in a simulation environment [78].

Consider a cube of computational nodes, each internally synchronous but externally asynchronous, used to simulate a 3-dimensional structure using the discrete element method (DEM). The nodes keep track of internal particles on a local clock, tracking velocities and interactions with each time step and subsequently adjusting velocity and position data as interactions occur. These interactions occur over some finite distance; when two particles are within such proximity of each other, the physics model must determine to what degree they affect each other. As particles travel to the edge of a given node, they eventually enter the interaction distance of the adjacent node. At this point, the two nodes must begin exchanging data to determine when the particle should eventually transition from one node to the next. Thus, the interaction distance forms a "shell" around each node; when particles occupy this space, they increase the communication time T_{COMM} of the node.

Say each node keeps track of a space 10 units long on a side, and the interaction distance is 1 unit. The shell volume as a fraction of overall volume is thus $(10 \times 10 \times 2 + 10 \times 8 \times 2 + 8 \times 8 \times 2)/(10 \times 10 \times 10) = 48.8\%$ If the particles are randomly distributed, that means nearly half the particles contribute to T_{COMM}. As discussed above, depending on communication peripheral speed this could be significant enough to bottleneck the processor and reduce its computational efficiency. But simply tripling the side length reduces this value by more than half, to $(30 \times 30 \times 2 + 30 \times 28 \times 2 + 28 \times 28 \times 2)/27000 = 18.7\%$. This classic surface-area-to-volume scaling continues to benefit larger nodes; a node with 100-unit length sides and 1-unit interaction distance, for example, has less than 6% of its as "shell".

For the ATSAMD51J20A used in early DICE prototypes, the communication peripheral runs at $3/160 = 1.9\%$ of the core processor speed. Based on a 1-unit interaction distance, each node would need to be upwards of 300 units per side to avoid bogging down communication. In tests, these devices had sufficient memory to keep track of roughly 1500 particles, or 5.5×10^{-5} particles (on average) per volumetric unit. This suggests that such devices could be useful for 3D gas simulation. If one particle occupies each volumetric unit (if they were, say, simulating a solid cubic lattice of digital materials), each node would represent a volume with a side length of 11 (rounding down). In this case, the communication peripheral would bottleneck the processor unless it ran at 45% of the core processor speed, or 72 MHz. This is clearly out of the question with conventional UART communication systems, but in the realm of practicality for a device equipped with a low-voltage differential signaling (LVDS) peripheral.

2.2.4 The DEM Node

A better method for processor selection is to consider the ideal interaction distance for a particle system and then specify communication peripheral speed, core clock speed, and memory (which is needed to keep track of more particles) to size an ideal device. Since the work in subsequent chapters is focused primarily on physical DICE infrastructure and end-to-end workflow development, the ATSAMD51J20A is still ideal for prototyping as it is relatively low-cost, available in a compact package with six independent communication channels (for cubic tiling), and easy to program using ubiquitous open-source tools.

Looking forward, an application-specific integrated circuit (ASIC) for physical modeling using the DEM summarized above would look substantially different as compared to any commercially available chip. The UART peripherals from the ATSAMD51J20A would be changed for a system explicitly designed at the hardware level for asynchronous token-passing, so that these functions (producing and consuming tokens, checking for the presence of a token) would occur without processor intervention. UART is unclocked, instead depending on relatively matching clock domains across asynchronous components; this method relies on stable clock domains, so switching to a truly asynchronous single-wire method like Manchester encoding [48] would be beneficial.

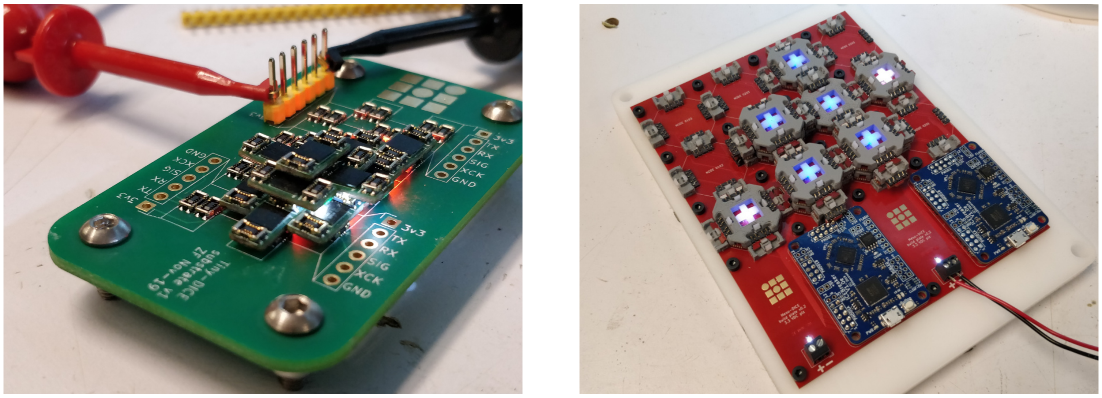
Computationally, a dedicated DEM ASIC would balance memory resources with particle count, shell size, and the previously discussed relationship between compute and communication speed. For modeling solid materials, this likely means a dramatic increase in local memory as compared to the microcontroller-based nodes discussed above, since a close-packed lattice has a far greater particle density as compared to a gas. The DEM ASIC would also include a fast double-precision floating point unit (FPU) rather than the single-point FPU in the ATSAMD51J20A. Finally, the shell ratio (i.e, the ratio of the shell volume to the interior volume) and T_{COMM} would be used to determine a T_{COMP} sufficient to maximize E.

The physical packaging of a dedicated DEM ASIC would reflect the interconnect strategy discussed in Chapter 5. In particular, the bare chip would be designed for ready interfacing with glitter-scale electronic digital materials by increasing the relative size and thickness of the metallized interconnect pads. At the same time, the number of off-board electrical connections could be dramatically reduced, since the system would have no need for the broad applicability requirements that drive COTS microcontrollers to have high pin counts. The die would also include an on-board switching power controller designed for physical close-coupling to discrete inductors and capacitors, allowing the system to maintain full core voltage despite consecutive voltage drops across serially-connected discrete power rails.

Chapter 3

Modules

In order to demonstrate the viability of reconfigurable computational systems, two distinct hardware platforms were designed and fabricated. The first generation, called Tiny-DICE, used the smallest available commercial-off-the-shelf (COTS) components and high-density interconnect (HDI) PCB fabrication technology to maximize computational density. The second generation, called Meso-DICE, was substantially larger and more mechanically reliable, and was intended to support the development of a full end-to-end DICE workflow, from design tools through applications.



(a) Tiny-DICE.

(b) Meso-DICE.

Figure 3-1: Two DICE hardware iterations assembled on their respective build plates.

3.1 DICE Architecture

While the two physical DICE versions differ in some ways, such as mechanical configuration and neighbor count, they also share many characteristics which can be addressed together.

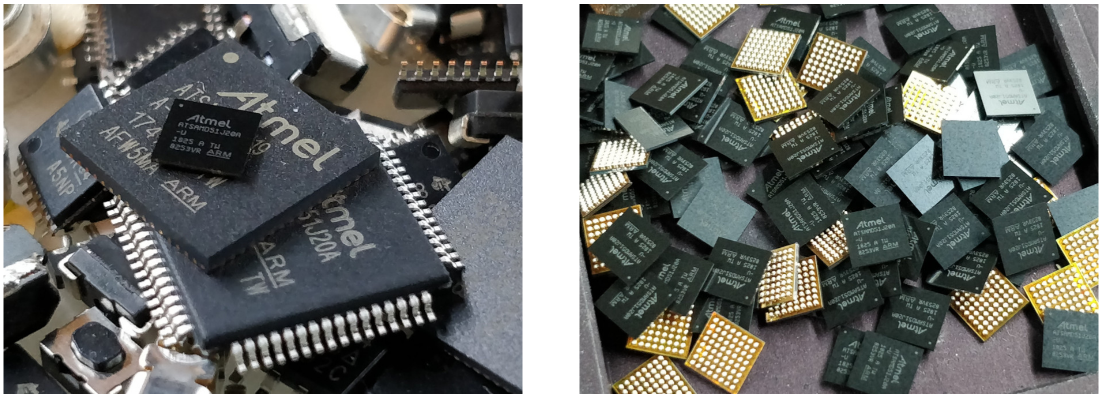
This strategy was deliberate, since there was some overlap between development timelines of each design and sharing features (particularly the microcontroller) dramatically reduced firmware development overhead.

3.1.1 Processor Selection

A multitude of axes exist to evaluate the suitability of a microcontroller for a project. Since the DICE platform is intended to be application-agnostic and performant, a fast 32-bit microcontroller with a large on-board memory bank and a hardware floating-point unit (FPU) is desirable. To minimize inter-node communication overhead, the device needs to have enough independent physical communication peripherals to manage data streams from all adjacent modules; in the case of a cubic lattice, this means six total channels (North, South, East, West, Up, and Down). Physically, the microcontroller needs to be available in a wafer-level chip-scale package (WLCSP), a method of packaging integrated circuits (ICs) that is minimally larger than the silicon chip itself. The device must be able to operate with minimal external components to reduce overall module size, including flash memory chips, power management ICs, and crystal oscillators. Finally, the part must be readily available from domestic commercial distributors in small quantities, and ideally should have well-documented support in the open-source hardware community to ease initial development.

A field-programmable gate array (FPGA) would provide the greatest flexibility since a custom-tailored soft processor core could be implemented based on application requirements. The requirement for dedicated communication peripherals could be satisfied in FPGA fabric; even better, custom protocols designed for token-passing could be used instead of standard serial standards like Universal Asynchronous Receive-Transmit (UART). However, FPGAs come with significant downsides, such as requirements for multiple staged power rails and external flash memory, both of which take up valuable PCB area. Furthermore, with some exceptions such as the Project IceStorm [91], most FPGA development environments are closed-source and difficult to extend to custom workflows.

After considering several ARM Cortex M4F processors, such as the STM32F4 series and the Maxim MAX32660, the Microchip ATSAMD51J20 [1] was selected for DICE. This microcontroller has six independent serial communication channels, allowing for cubic lattice packing, while still being offered in a small 64-contact package in several form factors. The SAMD51 series forms the core of the Adafruit Feather M4 Express and Metro M4 products, open-source development platforms that are extensively documented and used by the electronics community, along with the networked dataflow machine controllers built by Jake Read [71], a CBA colleague.



(a) ATSAMD51J19s and -20s in WLCSP, QFN, and QFP packages from the pick-and-place "dropped part" bin.

(b) A pile of uninstalled 3x3 mm WLCSP ATSAMD51J20s randomly oriented to show 0.4 mm spaced solder ball contacts.

Figure 3-2: Microchip ATSAMD51J20 microcontrollers.

3.1.2 Electrical Design

DICE modules share common power rails, using parallel interconnect pathways and dedicated power and ground planes when possible to minimize node-to-node resistance (see Figure 3-3). This ultimately limits the distance between a given module and a power source; at some point, consecutive voltage drops will cause far-away modules to brown out. DICE build plates feed power into an entire face in parallel, so theoretically the length and width of a DICE structure could be infinite with a fixed maximum height. One method for increasing the maximum height of the computational structure would be to add local regulation to each module and supply a higher bus voltage; however, this adds significant complexity and cost to the design and reduces power efficiency, even if switch-mode buck regulators are used. Since computational efficiency (i.e W/Flops) is a key test for DICE, this strategy was not pursued.

In contrast to the common power rails, DICE modules only communicate with physically adjacent neighbors. To minimize overhead, the ATSAMD51J20's built-in SERCOM peripherals were used in UART mode for this task, so two communication wires for high-speed bulk data transfer connect the appropriate RX and TX port on each device. Two addi-

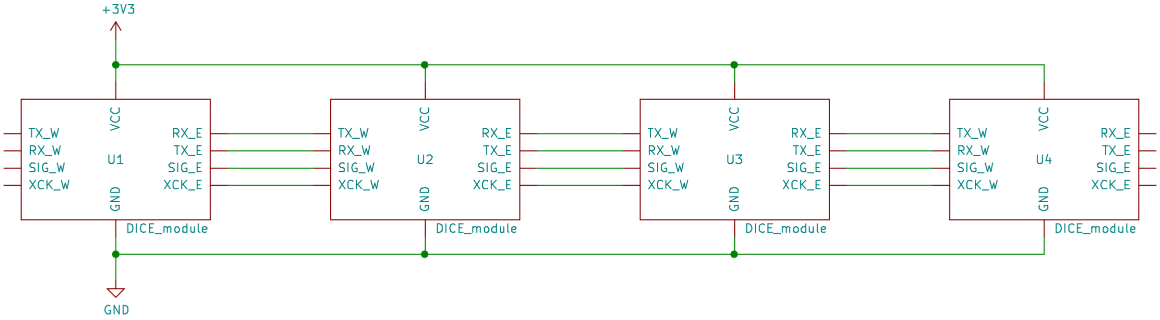


Figure 3-3: A simplified one-dimensional DICE network with four modules. All the devices share a common 3.3 V DC power bus, but only communicate with adjacent neighbors. Practical implementations have 4 or 6 neighbors per module, and may include passive struts to maintain lattice geometry.

tional general purpose input-output (GPIO) connections are included to provide flexibility for communication protocol development.

An important benefit of an intentionally asynchronous system is the lack of need for accurate local clocking. Because of this, the microcontroller support circuits lack the space-intensive crystal oscillator portion often seen on devices that need to manage timing-critical external communication protocols like USB. Thus, the only external components included beyond connectors are bypass and regulation capacitors and a status LED.

3.1.3 Firmware and Programming

DICE firmware is written in C and C++ and compiled using the factory-supported GNU Arm Embedded Toolchain. The build process uses a CMake framework developed by Erik Strand [78], a CBA colleague, which allows test and application code to be quickly ported between different DICE generations and other platforms.

The ATSAMD51J20 offers several options for loading, or flashing, firmware onto the device. The microcontroller includes a built-in USB peripheral, meaning the device can connect directly to the USB port on a computer and self-program if equipped with a suitable bootloader. However, this requires breaking out USB pins or adding a dedicated connector to each board in addition to the standard programming pins, which must still be accessible for initial bootloader installation. Instead, DICE uses the ARM SWD debugging protocol for firmware flashing and debugging, a clocked serial protocol that requires a total of five electrical connections to use (data, clock, reset, power, and ground).

Using SWD means inserting a dedicated programmer between the DICE module and the computer. Programmers are fairly cheap and ubiquitous, but they still represent another piece of hardware that can be misplaced or broken. An excellent alternative is to use the open-source Free-DAP project [82], a firmware image that turns a cheap Microchip ATSAMD21 microcontroller into a generic ARM-CMSIS programmer. For a few dollars and a simple milled PCB, one can then build the programmer into the physical programming rig, leaving the user to supply a common USB cable. In the same vein, one of several open-source command-line flashing utilities can be used to initiate the programming sequence, such as Open-OCD or edbg [81]. The latter, by the same author as Free-DAP, is used here.

3.2 Tiny-DICE

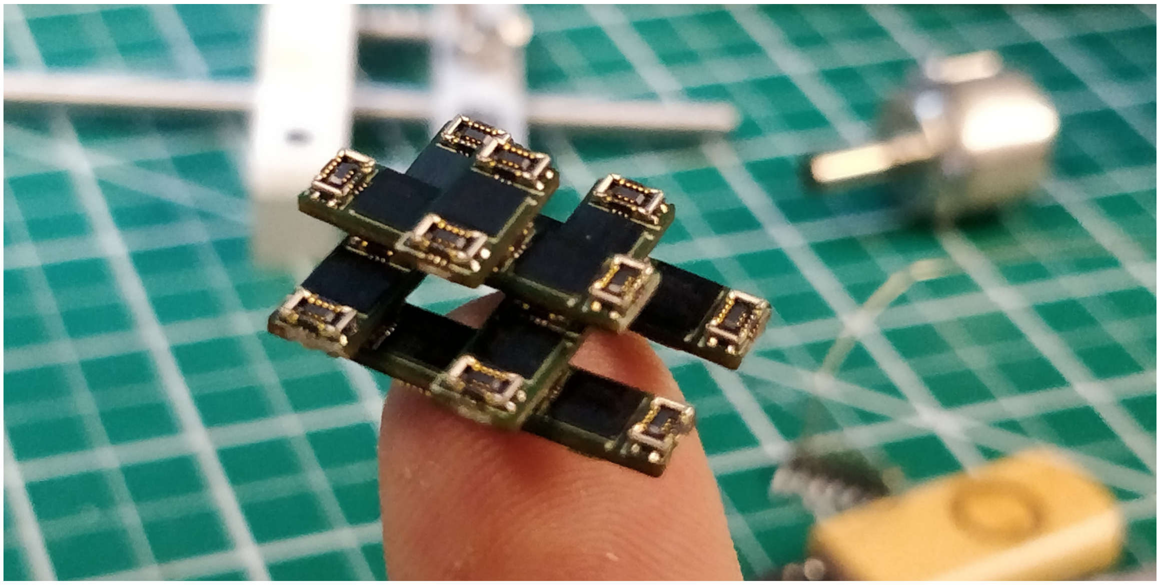
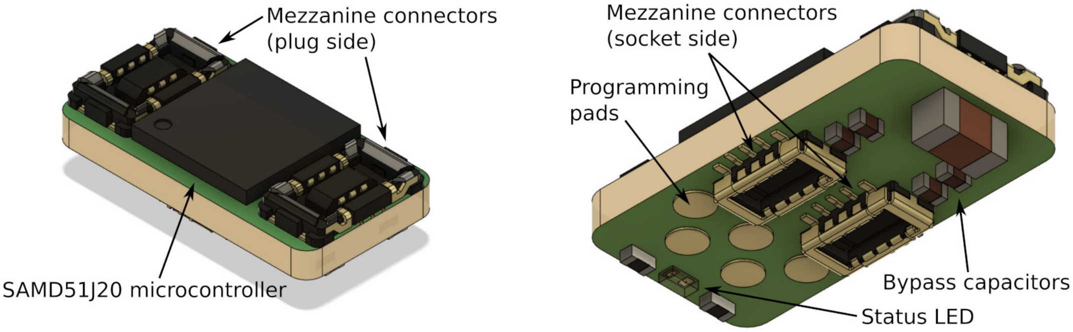


Figure 3-4: Nine Tiny-DICE modules assembled into a tetrahedral computational lattice and perched atop the author's pinkie finger.

3.2.1 Design

Tiny-DICE modules are designed around the smallest available COTS components. In addition to the aforementioned WLCSP ATSAMD51J20 microcontroller, each module includes four Molex SlimStack [7] mezzanine connectors which allow the devices to attach to a purpose-built base-plate and each other. The two bottom socket-style connectors are oriented side-by-side and centered, while the two top plug-style connectors are orthogonal to the bottom connectors and arranged at the edge of the PCB, shown in Figure 3-5. Overall, each module measure 9 mm x 4.5 mm, the minimum dimensions required to accommodate the mezzanine connectors and microcontroller.



(a) Tiny-DICE module, top.

(b) Tiny-DICE module, bottom.

Figure 3-5: Tiny-DICE module renders with labels, top and bottom.

Beyond the microcontroller and connectors, the Tiny-DICE design includes numerous bypassing and power regulation capacitors as recommended by Microchip, along with a tiny 0201 status LED and current-limiting resistor (see schematic, Figure 3-6). The PCB was designed around 4 mil/4 mil space/trace rules for a 6-layer HDI PCB with 0.15 mm drills and 0.25 annular via rings (see layout, Figure 3-7). Due to the 0.4 mm pitch on the WLCSP microcontroller, numerous blind vias were required to fan out some inner balls which substantially increased the cost of the boards.

Each Tiny-DICE module directly connects to up to four neighbors in a flattened tetrahedral configuration. This configuration results in an overall lattice packing density of approximately 50%, and eliminated the need for a second part type to join adjacent modules. One shortcoming of this strategy is that devices at the edge of a given lattice are cantilevered well beyond the edge of the adjacent boards, resulting in a moment force during uniaxial assembly.

3.2.2 Fabrication

Bare Tiny-DICE PCBs were purchased from a commercial vendor; however, the modules were ordered as un-routed 2x8 panels as the vendor edge machining tolerance was greater than required for the project. Singulation was performed on a removable aluminum jig using a 1/16" carbide end mill on a Roland MDX-540 desktop CNC router. Since the same edge routing tolerance applied to the panel perimeter used to locate the boards on the jig's three

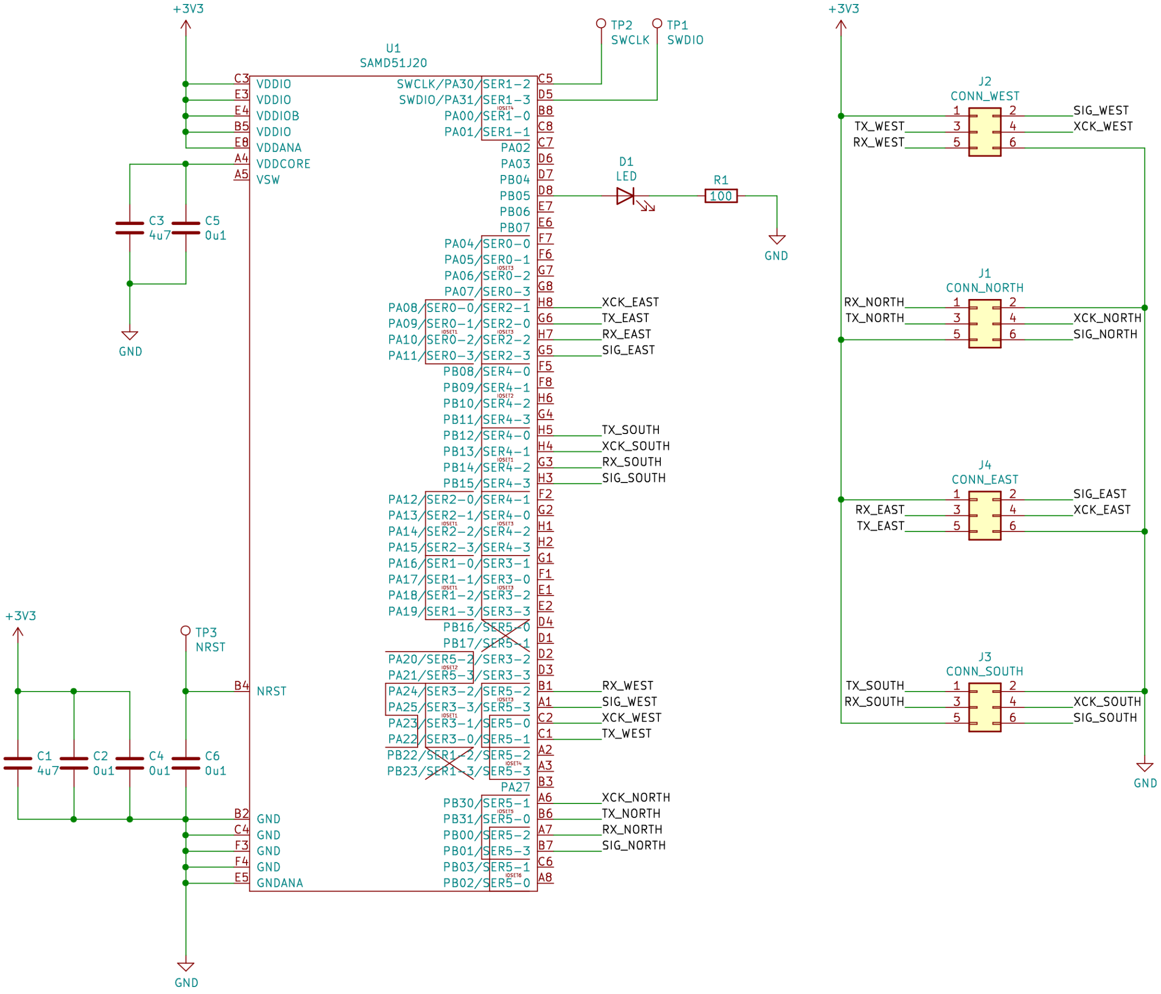


Figure 3-6: Tiny-DICE module schematic, showing ATSAMD51J20 microcontroller, bypass and regulation capacitors, LED with current-limiting resistor, programming pads, and interconnect. Microcontroller symbol includes labels grouping SERCOM peripherals.

ground locating pins, a panel was initially routed and examined under optical microscopy to adjust machining offsets for the singulation operation; it was then determined that these same offsets could apply to subsequent PCBs, so later milling operations yielded all 16 modules.

After edge routing, the panels were mounted on a milled phenolic fixture for solder paste application using a commercial press, and the back side (opposite the microcontroller) populated using a Mechatronika M10V pick-and-place machine. The panels were reflowed using a shop-modified reflow oven, cooled, and flipped so the front side could be similarly populated and soldered. A sample of these steps is shown in Figure 3-8.

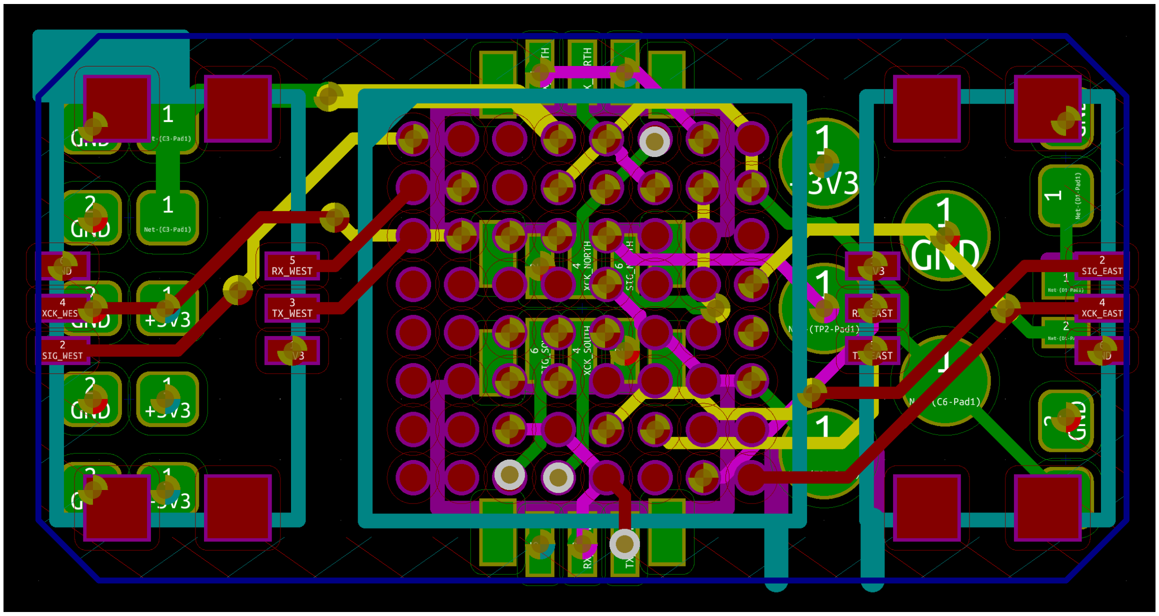
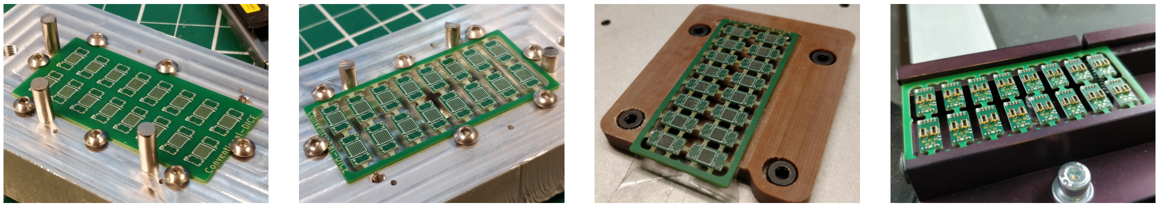


Figure 3-7: Tiny-DICE module 6-layer PCB layout, showing front traces and pads (red), rear traces and pads (green), internal traces (magenta and yellow), silkscreen marks (magenta and cyan), net names (white), PCB outline (blue), blind vias (gold crossed), and thru vias (gold and white). Power and ground pours indicated by hash marks around perimeter.



(a) Unrouted Tiny-DICE PCB panel, on singulation jig.

(b) PCB after singulation with Roland MDX-540.

(c) PCB after solder paste application, on paste jig.

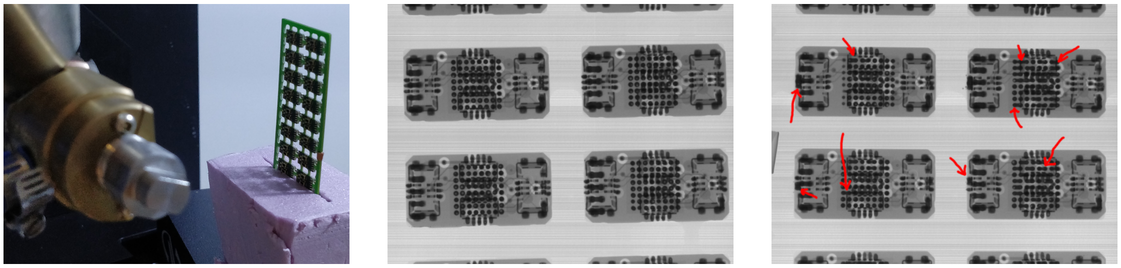
(d) PCB after population of back side components.

Figure 3-8: Tiny-DICE manufacturing steps.

3.2.3 Testing

The majority of Tiny-DICE modules were assembled in one multi-day run, with each step being performed on multiple panels prior to moving on to the next. This saved a great deal of setup time but hid significant problems until after a large quantity of microcontrollers were used, an unfortunately wasteful practice that could have been avoided with a bit more patience.

After the final reflow operation, one PCBA was visually examined on both sides using a Lynx Evo projection stereo microscope. Several modules appeared to have solder bridges between mezzanine connector pins; given the likelihood of similar but hidden defects under the microcontroller, all of the PCBAs were X-rayed using a Nikon Metrology XTH160 Micro CT scanner in single-image mode. Careful examination of the resulting micrographs, shown in Figure 3-9, showed frequent solder bridges, some of which were also in undetectable areas under the mezzanine connectors.



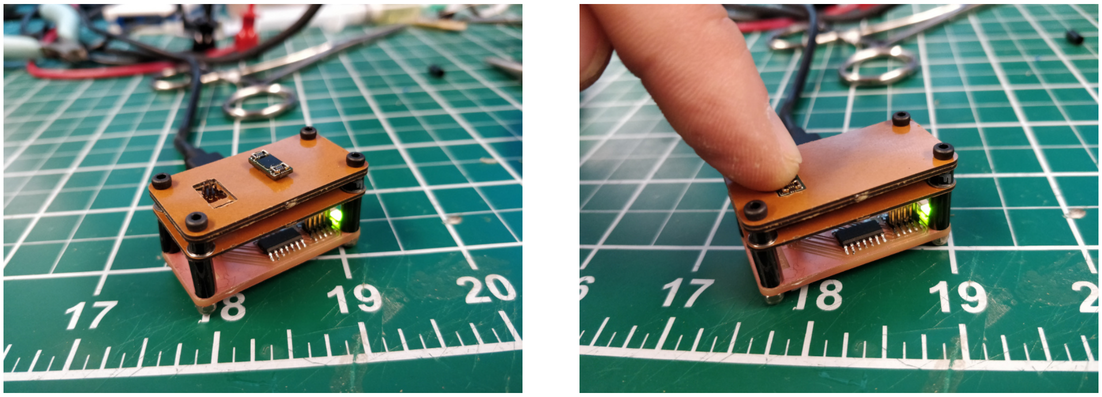
(a) X-ray setup, showing source in the foreground to the left. Note copper tape on PCB to identify board.

(b) X-ray micrograph of four good Tiny-DICE modules. Tabs are invisible due to lack of copper core.

(c) X-ray micrograph of four faulty Tiny-DICE modules. Red arrows point to solder bridges.

Figure 3-9: Tiny-DICE post-assembly X-ray examination.

A custom programming jig, shown in Figure 3-10, was designed and fabricated using sheets of phenolic to constrain the pogo pins and provide a recess for a single module. The in-house-routed PCB which forms the programmer's base includes a ATSAMD21 loaded with Free-DAP firmware, a linear regulator, and a status LED.



(a) Tiny-DICE programmer with module.

(b) Programmer in use, testing a module.

Figure 3-10: Tiny-DICE programmer with Free-DAP ARM-CMSIS-compatible firmware loaded on a built-in ATSAMD21 microcontroller.

PCBAs that passed X-ray inspection were singulated by clipping the tabs holding the modules together and sanding the rough edges smooth with a small Proxxon disk sander. They were then flashed with a simple test program that blinked the on-board LED. Ulti-

Step	Input Qty	Output Qty	Yield	Comments
Singulate	188	156	83.0%	fixture misalignment, test singulation cuts
Reflow	156	58	37.2%	solder paste quality issues, stencil too thick
Rework	58	15	25.9%	abandoned, remaining qty 43 pcs viable
Overall	188	15	8.0%	yield up to 30.1% with finished rework

Table 3.1: Tiny-DICE manufacturing run yields.

mately, of 188 modules attempted, only 15 functional modules were produced, for a final yield of 8.0%; an additional 43 modules required reasonably achievable but time-consuming rework that was not completed, so the best possible yield for the run would have been 30.1%. A breakdown of scrap sources can be seen in Table 3.1.

The vast majority of Tiny-DICE scrap was caused by unrecoverable solder bridges. Later tests showed that the Chip Quik SAC305 Thermally Stabilized paste, combined with a relatively thick stainless steel stencil, resulted in too much solder remaining on the PCB prior to component installation. Yields would likely improve dramatically with a finer sized, less viscous solder paste, such as Kester T5.

Tiny-DICE boards were loaded with a simple program [37] which uses a series expansion to calculate pi using a 5-step iterative loop:

$$\pi(N) = \sum_{i=1}^{N} \frac{0.5}{(i - 0.75)(i - 0.25)}$$

N, or the number of program iterations, was set to a large value such as 1,000,000, and the program was set to toggle a GPIO pin at the start and stop of the calculation. A digital storage oscilloscope was connected to the pin and set to trigger on a rising edge in single-shot mode, which allowed the total calculation time to be accurately calculated. Since each iteration involves exactly five floating point operations, calculating the processor speed in Flops was simply a matter of dividing 5,000,000 operations by the oscilloscope-recorded pin toggle delay. This yielded a value of 16.8 MFlops, roughly equivalent to an Intel Pentium Pro processor [35].

A number of Tiny-DICE nodes were programmed identically to iterate the pi calculation for a much longer period, so that they would run continuously as the lattice reached thermal equilibrium. The onboard LED was also deactivated to reduce unnecessary current consumption. Eleven such nodes were assembled in a tetrahedral lattice and examined using a FLIR i5 infrared imaging system, as seen in Figure 3-11.

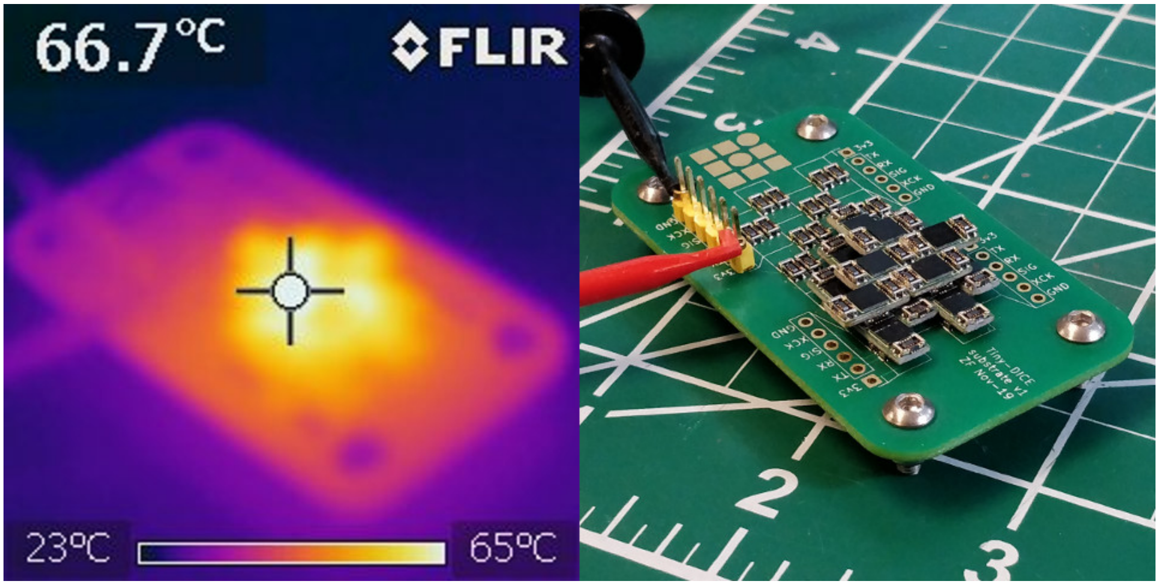


Figure 3-11: Testing tiny-DICE modules using a 5-step pi series expansion, thermally imaged after reaching equilibrium. Optical image to right shows 11-module lattice.

The thermal imaging revealed that the center of the lattice approached 70 C, near the ATSAMD51J20's 85 C operating limit. As such, larger lattice structures would likely benefit from increased sparsity or active cooling, either through forced air or fluid immersion. During the test, the lattice consumed 0.384 A at 3.28 V, both measured using an HP 34401A 5.5 digit bench multimeter. The structure thus required 1.26 W of power, which equates to 0.20 GFlops/W. This is on the order of an Intel i7-8700 at 0.34 GFlops/W [33], but well below GPUs like the NVIDIA V100 at 50.4 GFlops/W.

3.3 Meso-DICE

3.3.1 Design

In contrast to Tiny-DICE, the Meso-DICE iteration was less concerned with minimizing size as it was with maximizing utility, reliability, and manufacturing yield. As such, no physical parts were carried over between the two designs. Even though the processor remained the same, the packaging was changed to the Quad Flat No-Lead (QFN) version to lessen the

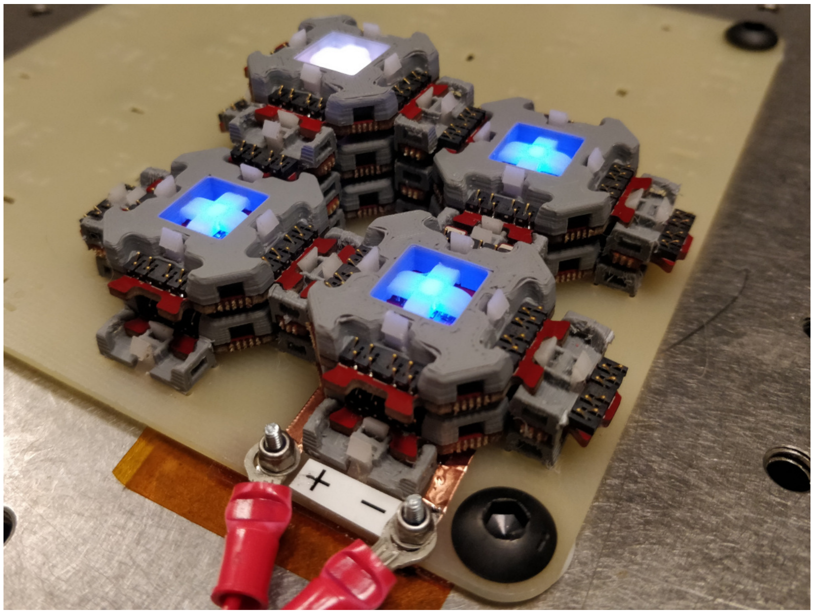


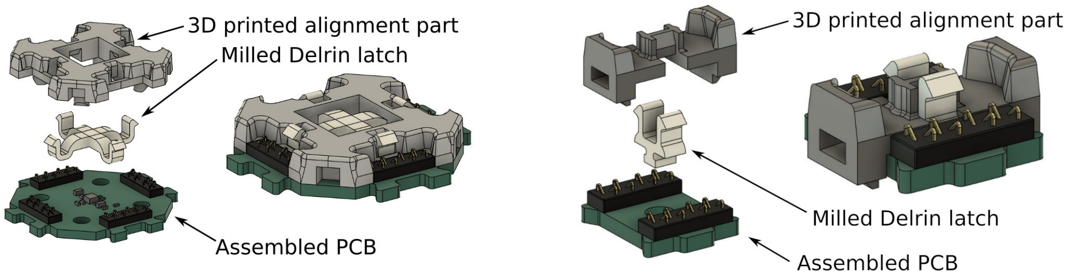
Figure 3-12: Nine Meso-DICE nodes and seven struts assembled onto an early version of the build plate.

chance for soldering defects and allow for more conservative PCB design rules.

An important change for Meso-DICE is the addition of two neighbors to each node to form a cubic lattice. This configuration is advantageous for many reasons; in particular, a Cartesian layout is much simpler to map to a physical volume defined in X/Y/Z coordinates such as in a simulation environment. However, the flat planes that make up each layer of the lattice require modules to be joined edge-to-edge, which is difficult particularly if uniaxial assembly from the Z direction is used. One strategy is to use asymmetric modules that overlap edge-to-edge, but this requires electrical components at multiple heights which increases fabrication complexity and cost. Instead, the DICE modules were decomposed into active nodes and passive struts which join uniaxially and form a cubic lattice, as shown in Figure 3-12.

A number of integrated commercial connectors were considered for Meso-DICE; that is, connectors that both mechanically constrain and electrically interconnect two printed circuit boards. These types of connectors are generally designed without parallel insertion in mind, meaning they are not intended to be used in pairs or greater quantities on the same substrate. In some cases, part datasheets specifically call out this limitation, or specify that flexural elements should be added to resolve the mechanical overconstraint caused by the parallel configuration. While this limitation also exists for the mezzanine connectors used in Tiny-DICE, the tetrahedral lattice configuration meant there were always at least six connection points between two connectors on the same board. Even when fully mated, these extra connectors provided sufficient flexibility to avoid putting too much stress on the solder joints or components. With Meso-DICE, vertically stacked nodes are separated by four connectors via struts; a brief test using a milled board showed that this configuration inadequately relieved the connector overconstraint and resulted in PCB and solder joint failure.

Instead of depending on integrated COTS connectors that manage mechanical alignment, latching, and electrical connectivity, these three requirements were separated into discrete parts, shown in Figure 3-13. The electrical interconnect uses surface mounted spring terminals that press against plated PCB pads. After PCBs are fabricated and populated, Meso-DICE struts and nodes are assembled with a 3D printed part which aligns and horizontally constrains mating parts, along with milled Delrin latches that vertically secure the parts together. The parts are secured using posts integrated into the 3D printed parts which are heat-staked to the PCBs, forming a tight and compact connection.

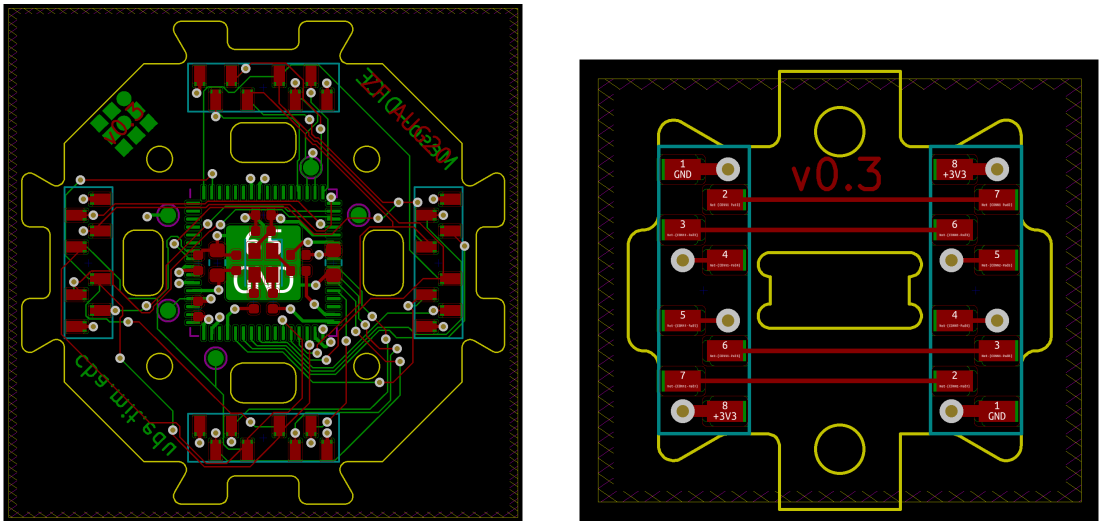


(a) A Meso-DICE node.

(b) A Meso-DICE strut.

Figure 3-13: Meso-DICE node and strut, exploded to show 3D printed alignment part, milled Delrin latch, and assembled PCB.

Electrically, Meso-DICE nodes were nearly identical to Tiny-DICE but substituted an integrated red/green/blue (RGB) LED to increase local indication options. Each connector included eight electrical contacts: two for power and ground, four for horizontally adjacent nodes, and two for vertically adjacent nodes. In order to pass signals up or down, the struts simply pass these lines through; to form a complete connection, a pair of adjacent struts must be used to route all four data lines. The Meso-DICE PCB routing for the strut and node are shown in Figure 3-14. Both boards are 4-layer designs with internal power and ground pours; the struts required 4/4 design rules due to the extensive cutouts for Delrin latch clearance.



(a) Meso-DICE node PCB layout.

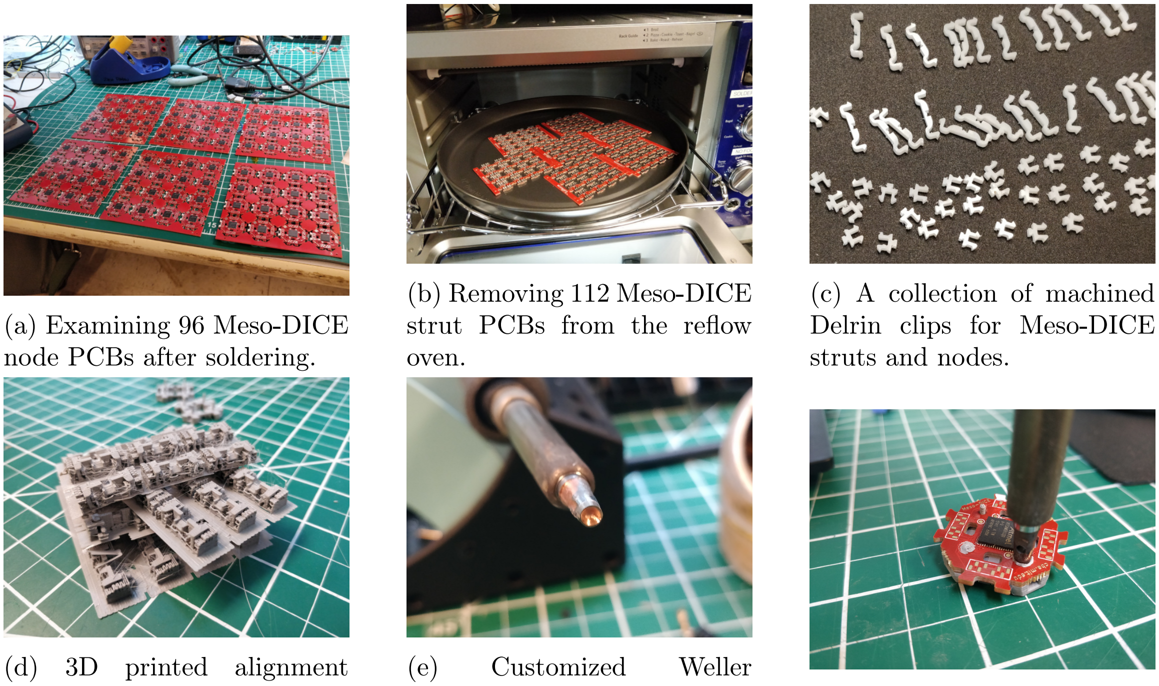
(b) Meso-DICE strut PCB layout.

Figure 3-14: Meso-DICE node and strut 4-layer PCB layout showing front traces and pads (red), rear traces and pads (green), net names (white), PCB outline (yellow), and vias (gold and white). Power and ground pours indicated by hash marks around perimeter.

3.3.2 Fabrication

As with Tiny-DICE, bare Meso-DICE PCBs were purchased from a commercial vendor. The boards arrive tab-routed in 4x4 panels; since edge clearances were not critical, this work was left to the vendor. Due to the larger component position tolerances, a solder paste jig was not needed either, simplifying paste application setup. After paste application, the boards were populated using the Mechatronika M10V pick-and-place machine and reflowed as before. This process was repeated on the opposite side for nodes since they include parts on both sides. After cooling, the boards were visually examined for soldering defects and bridges were manually fixed.

Delrin latches were milled on a Roland MDX-540 desktop router from 2.5 mm sheet stock using a 1/32" carbide end mill. For nodes, the top and bottom pieces were snapped together prior to insertion into the PCB. The plastic alignment parts were 3D printed using Matter Hackers Tough PLA on a Prusa MK3S printer and manually cleaned after removal from the print bed. A spare extra-large chisel tip for a Weller WES51 soldering iron was milled into a concave hemisphere and used to heat stake the struts and nodes together at 260 C. This process worked sufficiently but required careful clamping to ensure no gaps formed between



(a) Examining 96 Meso-DICE node PCBs after soldering.

(b) Removing 112 Meso-DICE strut PCBs from the reflow oven.

(c) A collection of machined Delrin clips for Meso-DICE struts and nodes.

(d) 3D printed alignment pieces before removing support material.

(e) Customized Weller WES51 iron tip for heat-staking 3D printed parts.

(f) Heat-staking a Meso-DICE 3D printed part onto a PCB.

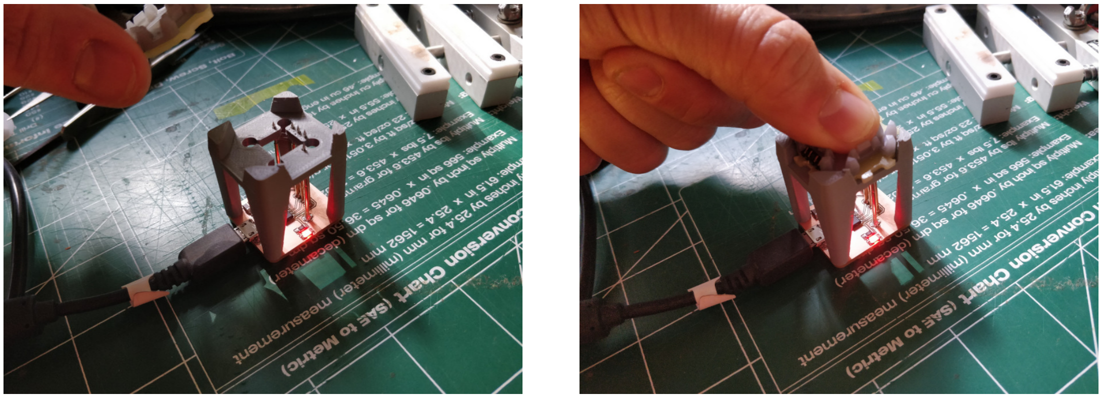
Figure 3-15: Meso-DICE fabrication process.

the 3D printed parts and the PCBs.

3.3.3 Testing

After reflow soldering, Meso-DICE PCBAs were visually inspected using a Lynx Evo stereo projection microscope. Solder bridges on the QFN microcontroller were somewhat common but quickly remedied using a bit of paste flux and a clean hot soldering iron tip. Yields were not tracked but were subjectively much higher than Tiny-DICE; occasionally a solder bridge was not recoverable, but nearly all boards with soldering defects were fixed. A notable exception occurred when the shop-modified reflow oven proved unable to adequately hold soak temperatures long enough to fully reflow the connector pads; after this, a commercial convection toaster oven was used manually to solder the remaining struts and nodes.

A programming jig, shown in Figure 3-16 was fabricated to quickly flash test firmware onto Meso-DICE nodes. The design is similar to the Tiny-DICE programmer in that it uses an integrated ATSAMD21 running Free-DAP. The top of the programmer has a 3D printed alignment jig that horizontally constrains the Meso-DICE nodes during flashing and keeps the pogo pins oriented correctly. Prior to final assembly with the Delrin latches and 3D printed alignment jigs, all nodes were flashed with the a simple test program that flashed the onboard RGB LED white and blue.



(a) Programmer with 3D printed alignment jig, pogo pins, and Free-DAP PCB.

(b) Programmer in use with an early non-functional mock up node.

Figure 3-16: Meso-DICE programmer.

Chapter 4

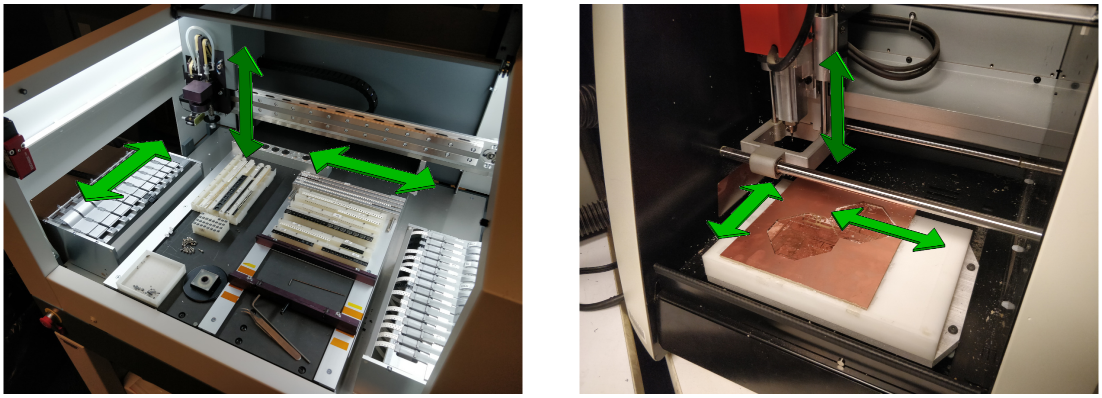
Assembly Systems

An important step in discrete modular computation is assembly. While hand-assembly is useful during testing, the goal of this project is to integrate automated assembly into the design workflow, just as compilation integrates into the traditional software development process. Building on the two module designs discussed previously (Tiny-DICE and Meso-DICE), this chapter presents two strategies for automated assembly of computational systems.

4.1 Cartesian Assembly

Most commercial computer numeric control (CNC) machines use some form of Cartesian position system. In other words, these systems have orthogonal X, Y, and Z axes which allow the user to address a rectangular work volume using simple coordinates. Two examples of commercially available Cartesian machines are shown in Figure 4-1.

Cartesian motion systems benefit from fully decoupled axes, meaning each axis has its own actuation system which is not affected by the others. Moving in a straight line along one of these axes simply means rotating a single motor a set amount; coordinated movement requires path planning between several motors to account for acceleration. This stands in contrast to serial or parallel manipulators, where any straight line motion must be coordinated between multiple actuators and actuator rotation is related to position through joint kinematics.



(a) Mechatronika M20V pick-and-place machine, used for electronics assembly.

(b) Roland SRM-20 desktop mill, used for rapid PCB prototyping.

Figure 4-1: Two commercially available Cartesian machines with principle linear axes labeled using green arrows.

4.1.1 Design

An existing Cartesian machine originally fabricated by Will Langford was adapted for Tiny-DICE assembly. This work was carried out by Jiri Zemanek and is included here for the sake of completeness, and because some of the lessons learned during testing informed the Meso-DICE design.

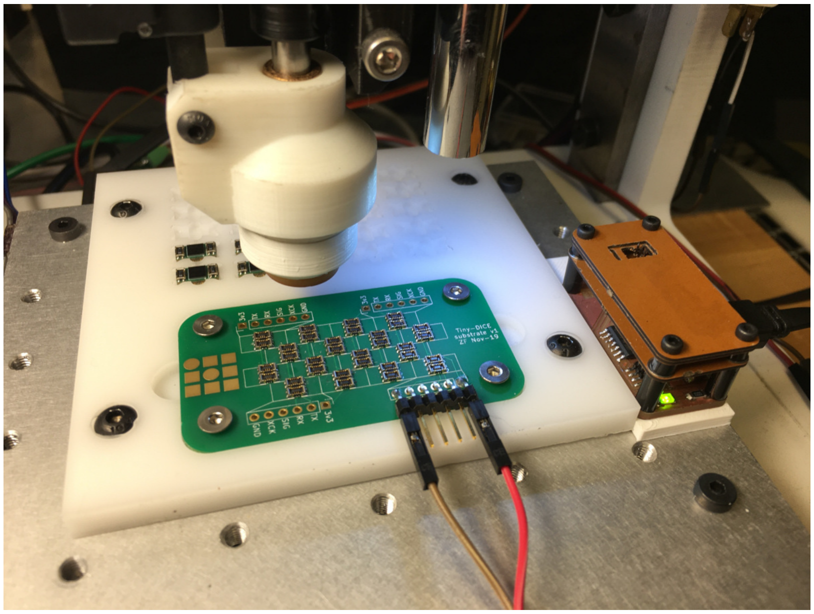
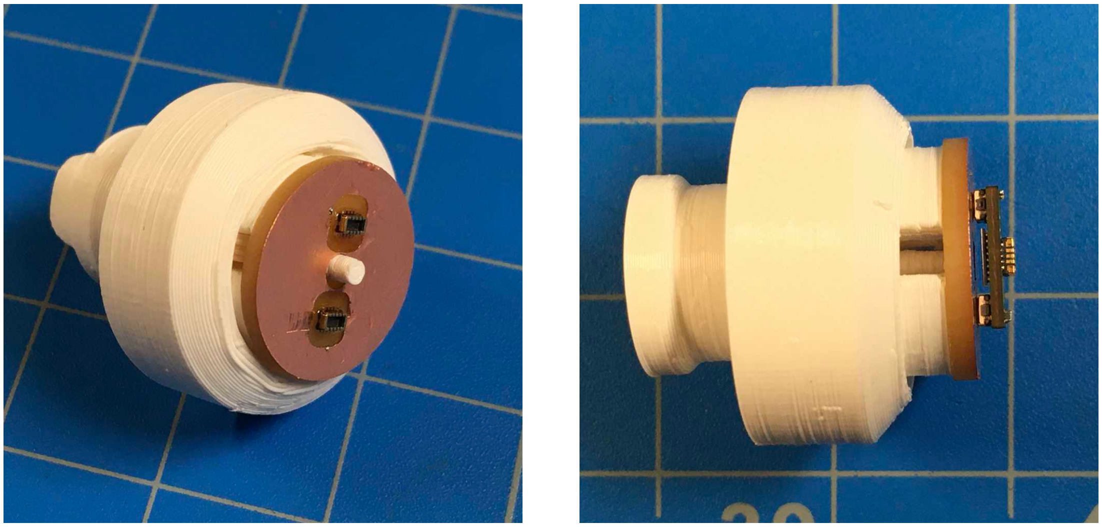


Figure 4-2: An overview of the Tiny-DICE assembly machine, showing green PCB build plate, white 3D printed end effector, planar part storage plate, and integrated programming station to the right. Photo courtesy of Jiri Zemanek.

With Tiny-DICE, assembly automation was a secondary consideration that was not taken into account during the module design process detailed previously. As such, the modules lack features to assist with handling operations such as pick-up, alignment, or placement. Because of this, the 3D printed end effector simply uses a pair of Molex SlimStack connectors soldered to a milled FR1 PCB, as seen in Figure 4-3. A 3D printed ejector pin runs through a hole in the PCB and is actuated by the motion platform's existing linear servo.



(a) End effector isometric view showing mezzanine connectors and ejector pin.

(b) Side view with Tiny-DICE module installed.

Figure 4-3: Tiny-DICE assembly machine end effector detail. Photos courtesy of Jiri Zemanek.

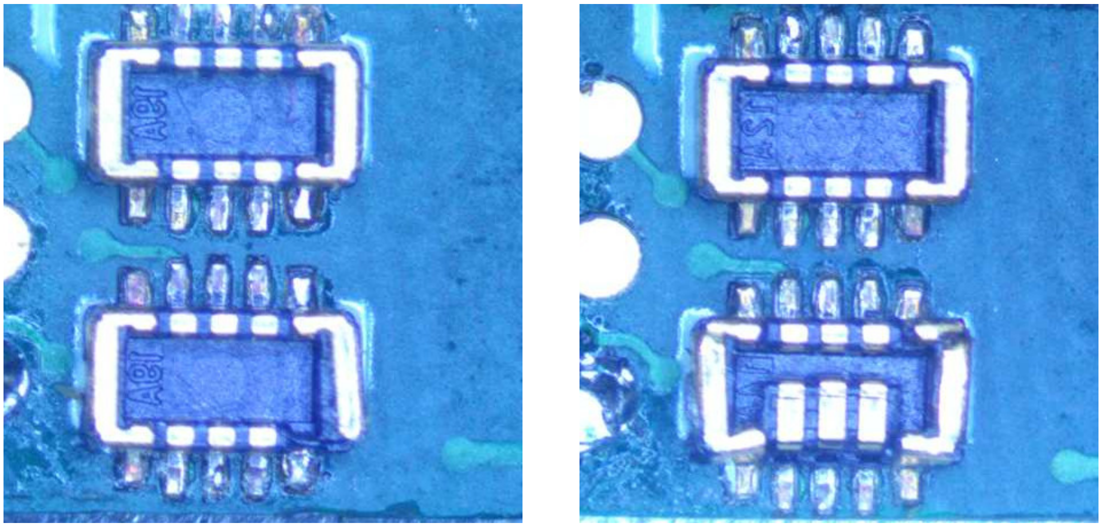
The Tiny-DICE assembler is controlled by a web-based Javascript interface and a Tiny-G controller. More information about this implementation can be found in [50]. Minor modifications were made to the code to allow quick pick-up of nodes based on integer grid location rather than raw distance, along with rotation during placement since alternating layers are orthogonal.

4.1.2 Evaluation

The Cartesian Tiny-DICE assembler is able to build several layers of computational volume without direct user intervention. In order to quickly evaluate electrical connectivity, each node was programmed with a simple script to continuously blink the on-board LED as soon as it powered up. Each node took roughly 12 seconds to pick up, relocate, and assemble; this rate varied somewhat depending on the location of the node in the pickup grid and its distance to the placement point. As discussed previously, the cantilevered end of each node meant that higher levels could not be assembled as reliably as the first layer, which is built upon a rigid PCB.

During automated assembly tests, inconsistent but alarmingly loud noises were observed during insertion of the nodes. Sometimes, the Tiny-DICE devices would seat quietly; however, in other cases the linear servo driving the insertion pin would audibly strain, and the entire end effector would visibly snap into position when the node seated. This usually occurred during node placement on the build plate, suggesting connector overconstraint due to the shorter kinematic chain connecting the two mezzanine connectors.

After a few automated assembly cycles, several of the nodes stopped working reliably. These failures were identified by observing the onboard LED; failed modules ceased blinking. In some cases, this was observed on the second assembly run; in others, a given node continued to work without any issues. The failed nodes were examined under a Lnyx Evo projection microscope and the connectors were imaged (see Figure 4-4).



(a) Tiny-DICE module with broken mezzanine connector.

(b) Tiny-DICE module with crushed mezzanine connector.

Figure 4-4: Close-up detail of Tiny-DICE modules after several assembly cycles showing broken mezzanine connectors caused by assembly over-constraint and misalignment. Photos courtesy of Jiri Zemanek.

The connector failures noted above are predicted in the SlimStack connector datasheet, which states:

> When mounting several board to board connectors on a same PWB [Printed

Wire Board, synonymous with PCB in this context], ensure to mount the each mating connector on a separate PWB.

Prior to design of the Tiny-DICE modules, it was observed that such language exists in nearly all commercially available subcompact mezzanine connectors. The decision to move forward despite this concern reflected a disinterest in compromising miniaturization in favor of durability. However, the result directly informed the Meso-DICE design decision to separate connector alignment, latching, and connectivity into distinct sub-components.

4.2 6-DOF Assembly

An alternative to Cartesian locating mechanisms is a serial manipulator, where a string of rotary or linear axes are linked end-to-end. This configuration improves the flexibility of the motion system, since rotating joints can allow end effectors to reach under or around objects that would block an X/Y gantry. Because of this flexibility, serial manipulators in the form of 6-degree-of-freedom (6-DOF) robotic arms, such as those shown in Figure 4-5 are popular for factory automation applications where systems must be adapted for different jobs without structural changes.

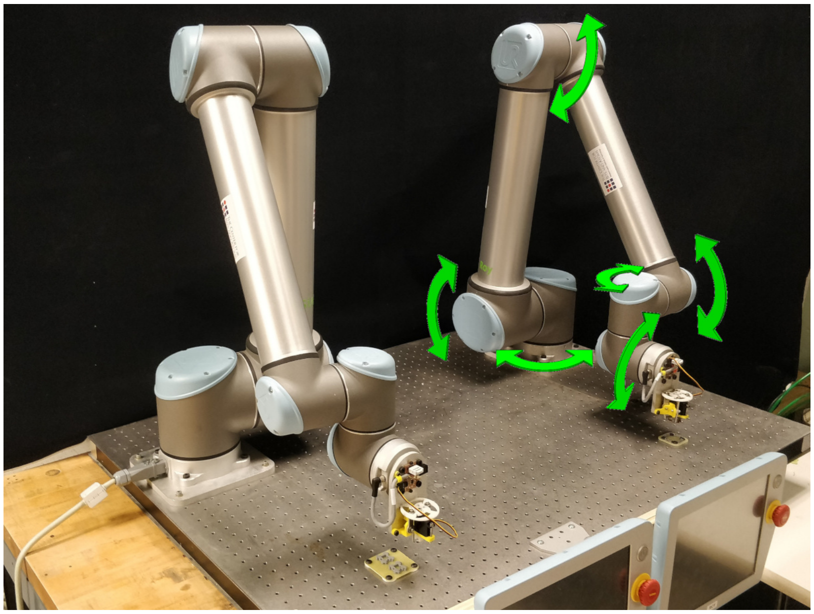


Figure 4-5: The two Universal Robots UR10 6-DOF arms used for Meso-DICE assembly. The strut placement arm is on the left, while the node placement arm is on the right and includes green arrows indicating the six principle rotary axes.

Serial manipulators are substantially more complicated than Cartesian machines to control, since the entire kinematic chain must be considered together to calculate the final location of the end effector. In particular, path planning is difficult since the degrees of freedom are inherently linked; it is relatively straightforward to get a tool from one point to another, but programming a specific path requires substantially more calculation. Fortunately, commercial robotic arms such as the Universal Robots UR10s used in this project include sophisticated inverse kinematic functions that allow users to simply input Cartesian or joint-space coordinates, and the control system translates between the two coordinate systems as needed.

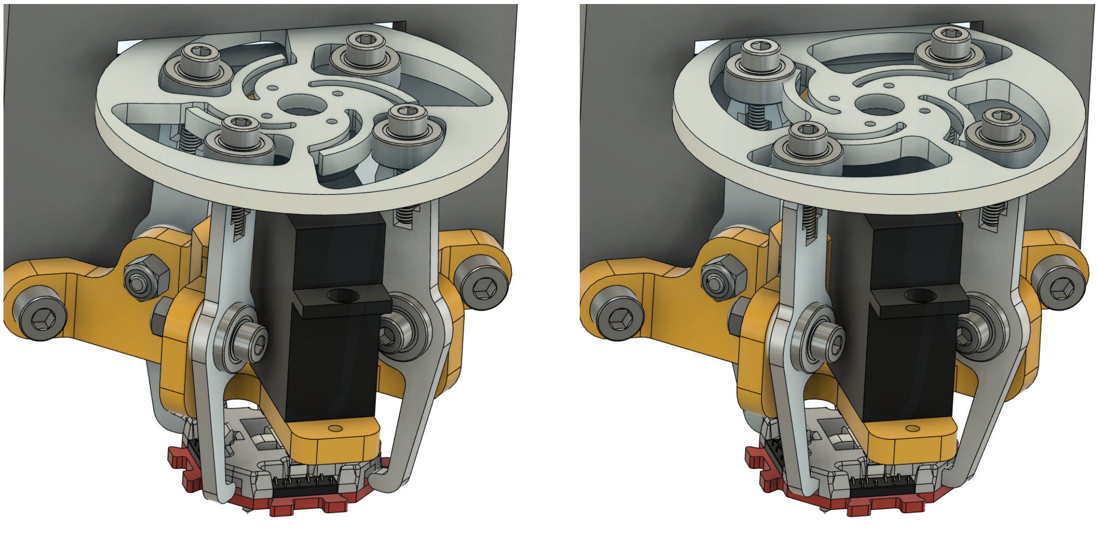
4.2.1 End Effector

As seen in Figure 3-13, Meso-DICE struts and nodes include a 3D printed part that holds the Delrin flexural latches in place, aligns struts and nodes to each other, and serves as a gripping point for the assembly machine. The gripping point features consist of tapered rectangular blind cavities, with two such features on each strut and four on each node. The gripping points are located to avoid interfering with adjacent struts or nodes in a partially constructed computational lattice.

Since the 6-DOF assembly infrastructure uses a pair of UR10 arms and the Meso-DICE lattice consists of two unique parts, an arm and matching end effector was dedicated to placing each type of module. The end effectors use 2.5 mm aluminum arms with tapered rounded barbs to squeeze the gripping points on the modules. The tapered nature of these components accommodates a millimeter of misalignment during module pickup, dramatically simplifying the manual part feeder tray design. The aluminum arms were fabricated on the OMAX waterjet cutter, and press-fit onto ball bearings which are then connected to a 3D printed frame. In order to minimize slop in the arm pivots, stiff Teflon washers fit between the bearings and the frame, and the joint is preloaded and secured with a bolt and locknut. Once adjusted for tension, this arrangement proved to be simple, compact, reliable, and remarkably easy to actuate due to the Teflon's low friction against the bearings.

To minimize size and complexity, the two (struts) or four (nodes) aluminum jaws are actuated with a single Hitec micro hobby servo using a Delrin cam orthogonal to the arm pivot axes. Each arm has a second ball bearing mounted in line with the cam and opposite the barbs which ride along the cam grooves. Notably, the inner working surface of the cam includes flexural beams which grant a small amount of compliance to the cam during pickup, which further helps the system tolerate minor misalignment. Once the servo fully actuates, these flexural elements bottom out and lock the part precisely in place for transport and placement in the computational lattice. This design strategy enabled rapid iteration of the pickup mechanism, as multiple cam profiles could be quickly fabricated and tested without redesigning and fabricating the other end effector components. Figure 4-6 shows a CAD render of the node end effector in both its open and closed state as it picks up a module.

During early testing, it was observed that the struts tended to rotate during placement when not supported on one side, such as at the edge of the lattice structure. Since the strut end effector only grips the modules at two points, this sometimes resulted in incomplete assembly of the parts which could then cause damage when subsequent nodes were added. To address this issue, a milled Delrin "foot" was bolted to the bottom of the strut end effector to provide uniform uniaxial assembly pressure across the entire module. This solved the problem and resulted in reliable assembly of struts at any point around the lattice.



(a) Open.

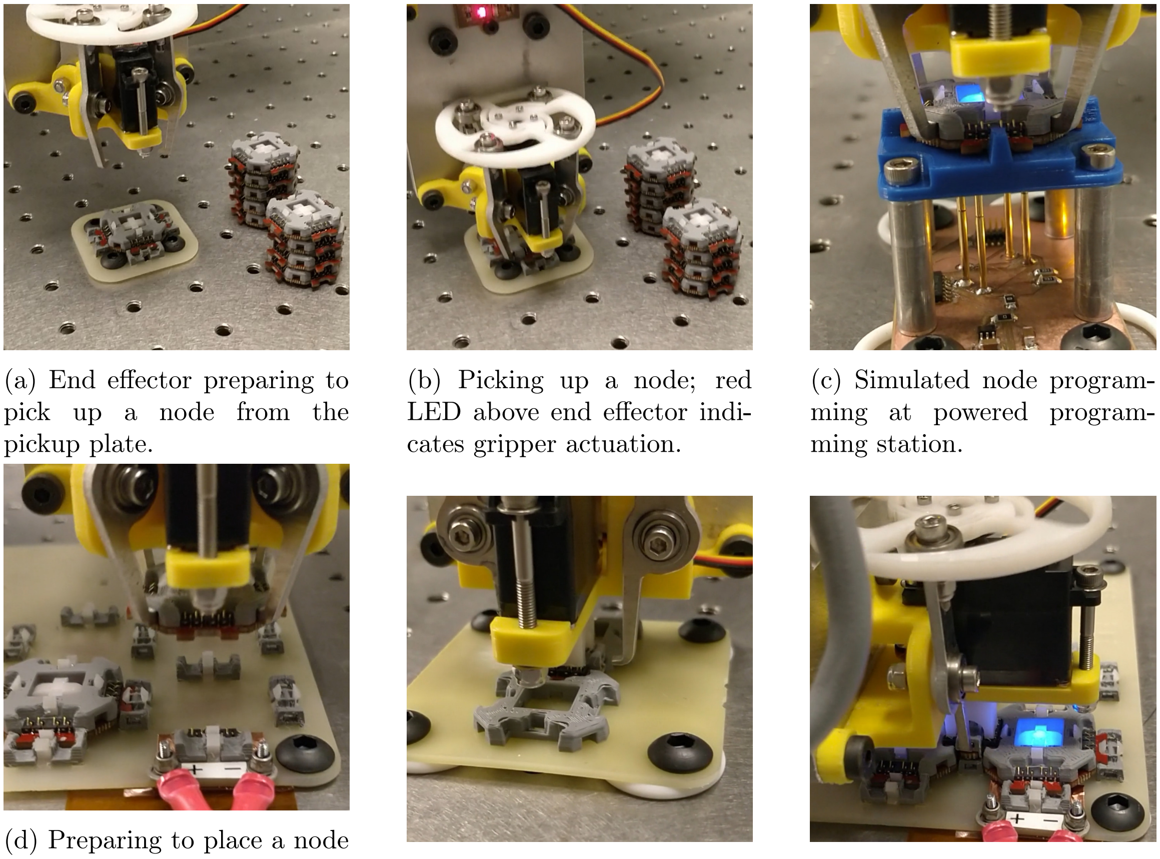
(b) Closed.

Figure 4-6: CAD renders of Meso-DICE node end effector, showing cam and arm action as gripper opens and closes around a module. Note that the compliant inner working surface of the Delrin cam does not distort in the closed render; on the fabricated end effector, these beams bend and bottom out, increasing the rigidity of the gripped part after pick-up.

4.2.2 Infrastructure

The two UR10s are rigidly mounted to a large stainless steel optical table with 1/4-20 tapped holes on 1" centers. This mounting scheme dramatically simplified system development, as it provided a known flat reference surface with convenient bolt holes for attaching various Meso-DICE accessories, including feeder trays, the node programmer (shown in 3-16), and the lattice build plate.

In a future DICE incarnation, a vibratory feeder or other sorting mechanism would collate modules and prepare them for pickup by the placement system, fully automating the mechanical assembly and leaving the user to simply pour a container of loose modules into a waiting hopper to "recharge" the system. For the Meso-DICE iteration, a simpler strategy was used in which the experimenter manually places a strut or node on a pickup plate while the UR10 is placing the previous module on the lattice. The pickup plate, shown in Figure 4-7, consists of several 3D printed alignment features from the opposite module type (struts or nodes) secured to a milled fiberglass plate without the associated Delrin latch components. In use, the experimenter simply must remain vigilant and replace parts as they are consumed, since the system has no feedback mechanism to determine whether a strut or node is actually picked up from the pickup point.



(a) End effector preparing to pick up a node from the pickup plate.

(b) Picking up a node; red LED above end effector indicates gripper actuation.

(c) Simulated node programming at powered programming station.

(d) Preparing to place a node on an early version of the build plate.

(e) Picking up a strut; notice milled Delrin placement foot.

(f) Placing a strut after rotating 90 degrees from pickup.

Figure 4-7: Video stills of the Meso-DICE automated assembly process, showing strut and node pickup, programming simulation, and placement.

Chapter 5

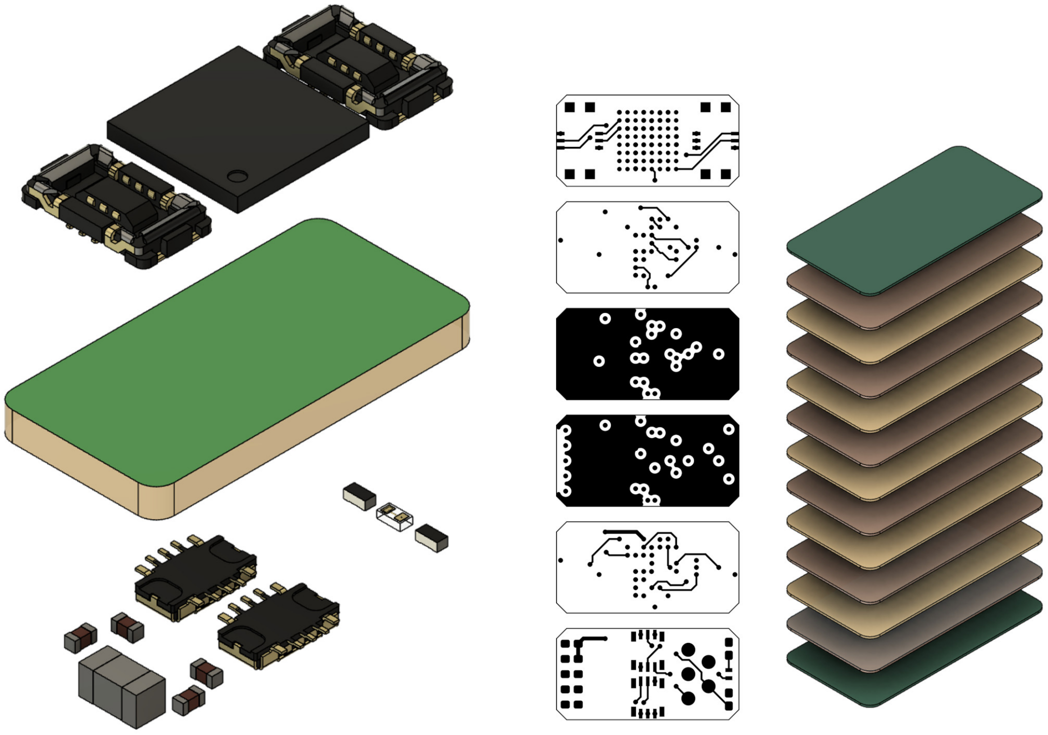
Electronic Glitter Lattices

The preceding chapters showed a relatively conventional approach to module design and assembly for DICE, using COTS components and commercially fabricated PCBs. However, the shortcomings of this approach became clear, particularly when pushing the limits of miniaturization with Tiny-DICE. Commercial interconnect systems are application-specific; in the case of the mezzanine connectors used in Tiny-DICE, this application is single-use connection of mobile phone parts during factory assembly. As demonstrated by the frequent failures during automated Tiny-DICE assembly, such connectors do not self-align and cannot withstand the substantial forces caused by overconstraint from parallel insertion.

Clearly, one approach to solving this problem is to design a new connector to fit the novel DICE application. Such a connector would be designed around existing electronics assembly techniques, such as pick-and-place component placement and reflow soldering. The part would use stamped and plated metal features for electrical connectivity, and the structure would consist of overmolded mineral-filled liquid crystal polymer (LCP), a high-temperature engineered plastic commonly used for electronic parts. The LCP features could be designed with large approach chamfers to facilitate connector self-alignment, and the metal contacts could have sufficient compliance to relieve the overconstraint caused by parallel insertion. But a conventional approach to connector design requires a conventional approach to project management. Designing a novel miniature connector from scratch would require a multi-year effort and substantial investment in tooling and pilot-scale fabrication runs, an inflexible method that would risk decoupling the connector design from the unique requirements of the DICE project.

5.1 Electronic Digital Materials

Another approach starts by reducing connectors to their fundamental building blocks, which are simply conductive and insulating materials cleverly composed to pipe electrons in specific directions. Interestingly, this same definition could be applied to PCBs, where copper, sheet adhesive, and fiberglass-reinforced polymer (FRP) are marked, etched, drilled, and plated to isolate electrical nets from one another, as seen in Figure 5-1.



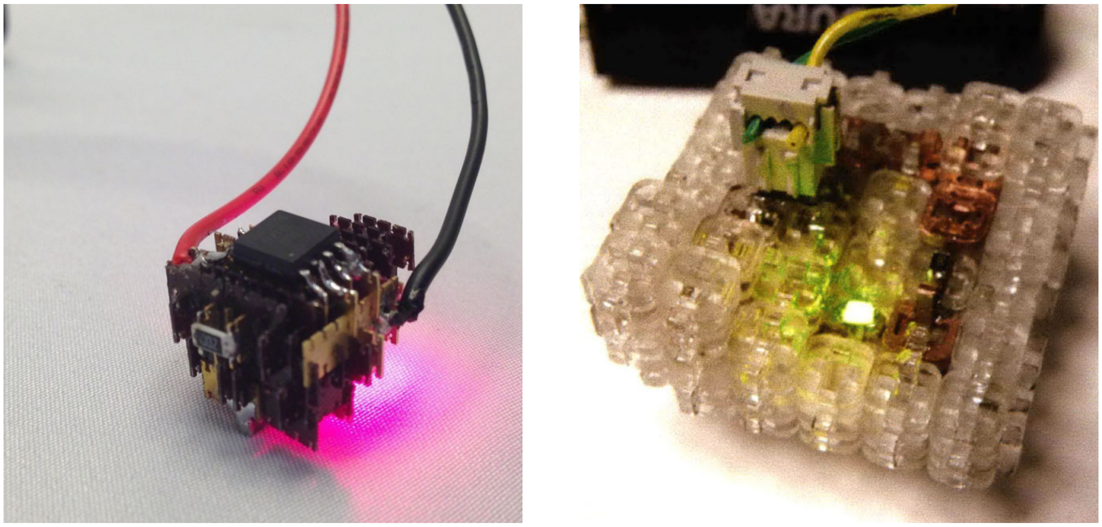
(a) Tiny-DICE PCB separated from its components.

(b) Tiny-DICE PCB blown up, showing six conductive copper layers interspersed with insulating fiberglass and solder mask.

Figure 5-1: Tiny-DICE renders at various levels of deconstruction.

In a dramatic re-imagining of PCBs and connectors, conductive and insulating materials are discretized into two dimensional plates, which are then friction-fit together to build routing structures. This approach, called electronic digital materials, has been demonstrated several times at the millimeter to centimeter scale at CBA, as seen in Figure 5-2.

Using digital materials with friction fit joints means heterogeneous materials can be utilized, and structures can be deconstructed and reused after assembly. Two dimensional planar parts also benefit from simple uniaxial assembly, so automating the construction of

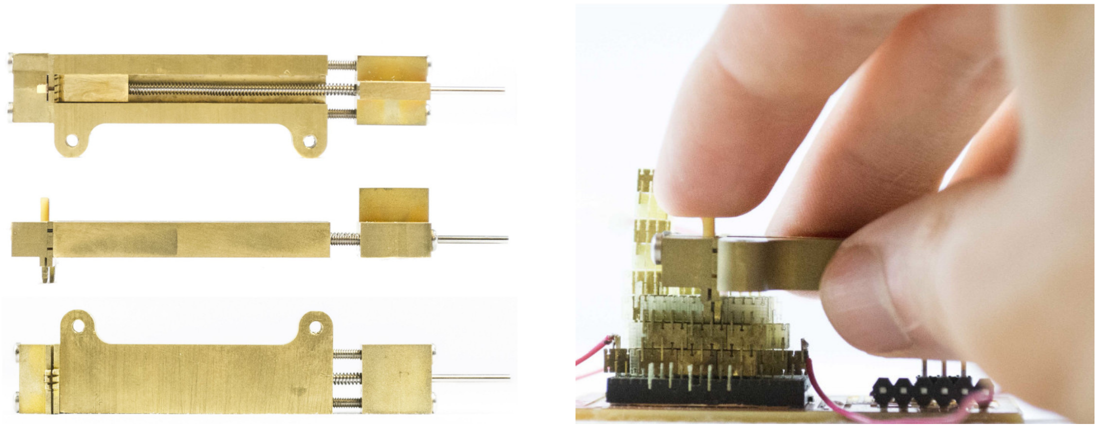


(a) Circuit by Will Langford, 2014 [49].

(b) Circuit by Jonathan Ward, 2010 [88].

Figure 5-2: Two prior CBA projects to build electronic circuits from digital materials.

the lattices can be achieved with a relatively straightforward three axis machine. Planar materials can be conveniently dispensed from a magazine, dramatically simplifying part handling. Langford used this strategy to design and fabricate a digital materials "stapler", shown in Figure 5-3.



(a) Manual stapler design.

(b) Stapler in use.

Figure 5-3: Will Langford's electronic digital materials "stapler" [49].

Langford and Ward's work on electronic digital materials was ultimately too large to effectively route complex electronic circuits. While both researchers were able to create simple demonstration circuits, the individual routing elements were too long to freely run circuit nets in any direction. This limitation stemmed from the lack of prototype-scale fabrication methods; Langford suggested deep reactive ion etching (DRIE) as a potential method for scaling electronic digital materials down, but such a process requires lithographic techniques relegated to a clean room and results in unsorted parts which must be handled and prepared for assembly. He also showed that direct-write laser machining was simply too slow to practically produce sufficient parts for a demonstration circuit.

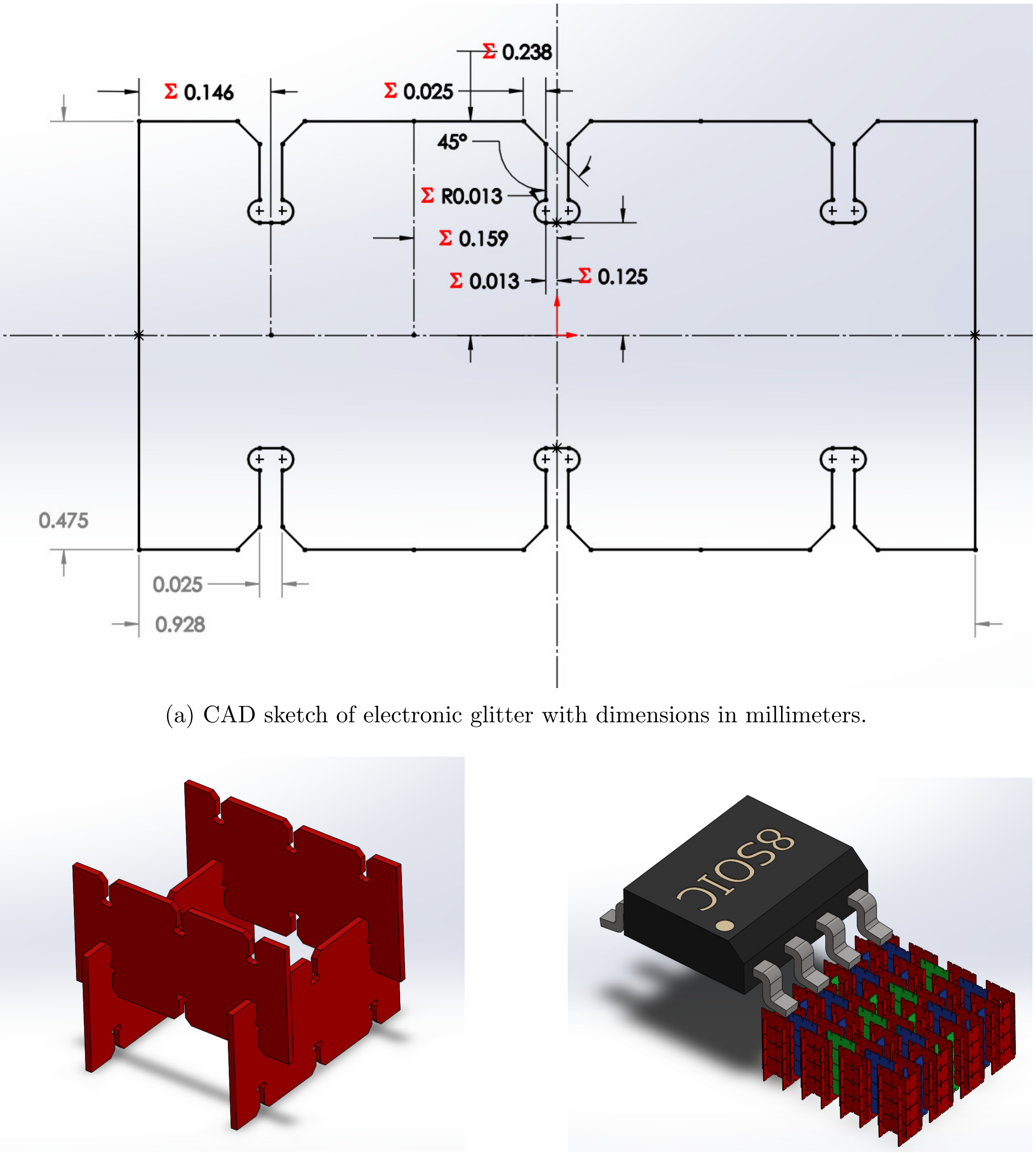
5.2 Glitter

The largest commonly used surface-mount integrated circuits are provided in Small-Outline Integrated Circuit (SOIC) packages with a lead spacing of 1.27 mm. Denser components include Quad Flatpack (QFP) components with 0.8 mm or 0.5 mm lead spacing, along with Thin Shrink Small Outline Package (TSSOP) with 0.65 mm or 0.5 mm lead spacing. In order to freely access all of the pins on such a device, a square lattice with a pitch in the range of the lead spacing can be used, provided adjacent elements can be electrically isolated from one another. If four plates are joined near their corners to form such a routing element, each plate is around a millimeter along its longest dimension, roughly the size of fine glitter and a five-fold decrease in size compared to Langford's work. One design for such a material is shown in Figure 5-4.

Once isolated conductive nets can be routed at the same pitch as the smallest components, any arbitrary circuit can be constructed to connect components to one another. In conventional PCB fabrication, as complexity grows and more nets need to cross one another, additional layers must be added to the board during fabrication; adding such layers increases fabrication cost and turnaround time substantially, limiting real-world design complexity. The approach described here, in contrast, can simply grow vertically to accommodate additional routing with a corresponding linear increase in part count, cost, and fabrication time.

5.2.1 Fabrication

Conductive glitter is prepared by stacking multiple sheets of stock, such as 25 micron phosphor bronze foil, and sandwiching the stack between two thicker pieces of aluminum plate. The entire assembly is then cut using a micro wire-EDM [20] that uses 10 or 20 um tungsten wire, shown in Figure 5-5. Unfortunately, delays related to the Covid-19 pandemic pushed



(a) CAD sketch of electronic glitter with dimensions in millimeters.

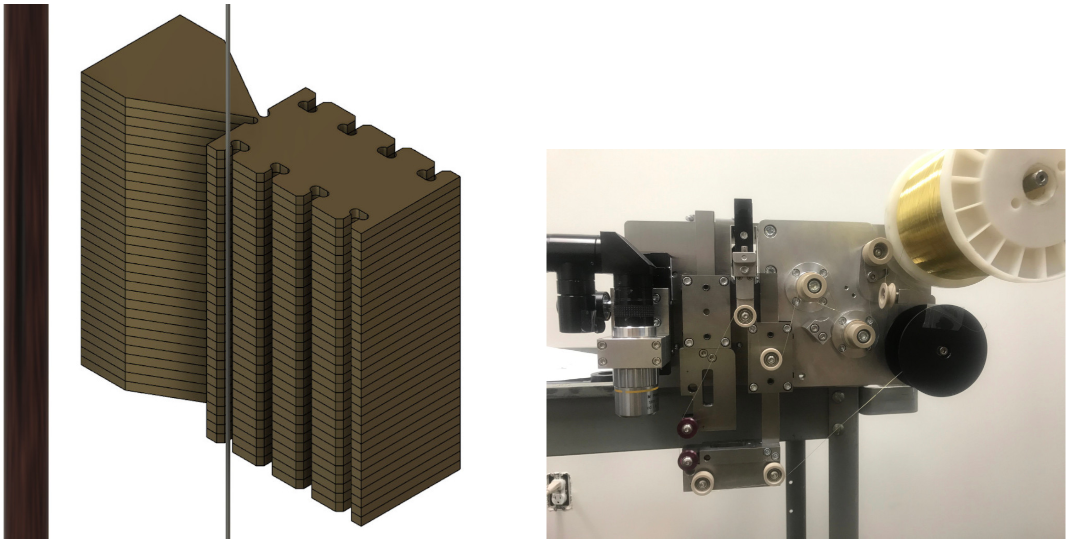
(b) Four glitter plates fit together to form a routing node.

(c) Multiple nodes (red) joined with struts (blue, green) to route an SOIC-packaged IC.

Figure 5-4: Electronic glitter element design and lattice structure.

back micro-wire EDM system fabrication and delivery, so reducing this concept to practice falls into the "future work" category.

Insulating glitter cannot be cut using the same techniques as wire-EDM fundamentally requires a conductive substrate. While insulating elements could be prepared using laser micro-machining or another direct-write process, a better option is to use the micro-wire



(a) Fabricating multiple conductive elements from 25 um phosphor bronze foil. Human hair shown for scale.

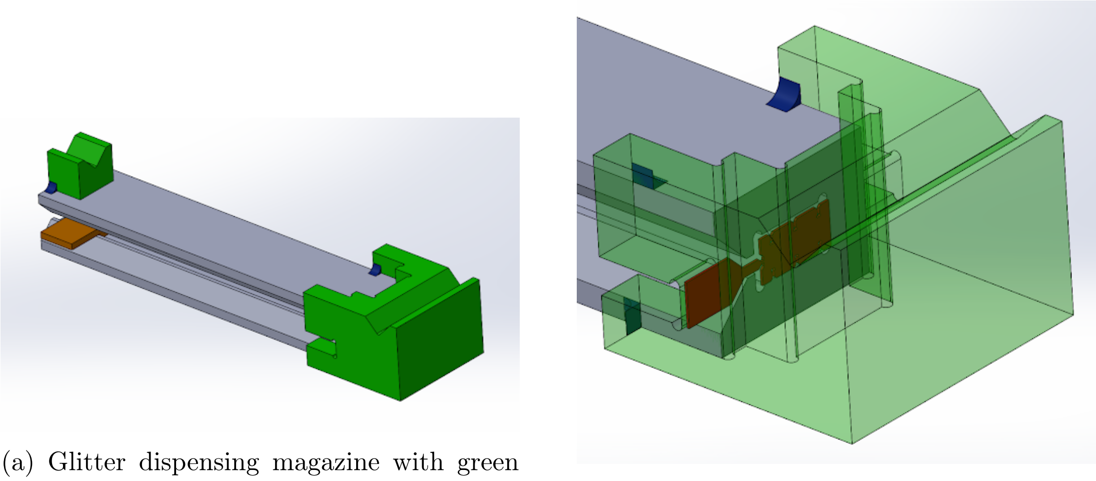
(b) Custom-built micro-wire EDM tool head as of August 2021. Photo courtesy of Viteris Technologies [20].

Figure 5-5: Conductive glitter fabrication process using micro-wire EDM.

EDM to machine precision stamping tooling out of hardened steel or tungsten carbide, and then rapidly stamp the material out of a strip of material. There are several polymeric materials that could be suitable and are available as thin 25 um films, but an interesting low-cost alternative is to use muscovite mica, a naturally occurring mineral with a long history of use in the electronics industry as a low-cost structural insulator. For example, most household pop-up toasters use sheets of mica as heating element supports [84], since the material is low-cost, heat resistant, electrically insulating, and reasonably strong. Mica has a distinct advantage in that its crystal structure is extremely anisotropic with a reliable cleavage plane, so the material can be readily split into precise thin sheets. In fact, one of the scientific uses for high-grade mica is as a sample surface for atomic force microscopy [29], since freshly split sheets are atomically flat. Once prepared, the material is then stamped using the aforementioned precision dies.

5.2.2 Assembly

As discussed previously, a compelling advantage to assembling 3D structures from press-fit 2D components is uniaxial assembly. This means parts can be dispensed from a magazine rather than picked up from a tray or tape as is done with electronics pick-and-place machines that are not of the high-speed "chip-shooter" variant. By including an additional break-away tab on each conductive part, a stack of plates can be removed from the micro-wire EDM and inserted in bulk into a magazine; once secured with a follower, the tab can be snapped off and the parts are ready to dispense. This is shown in Figure 5-6.



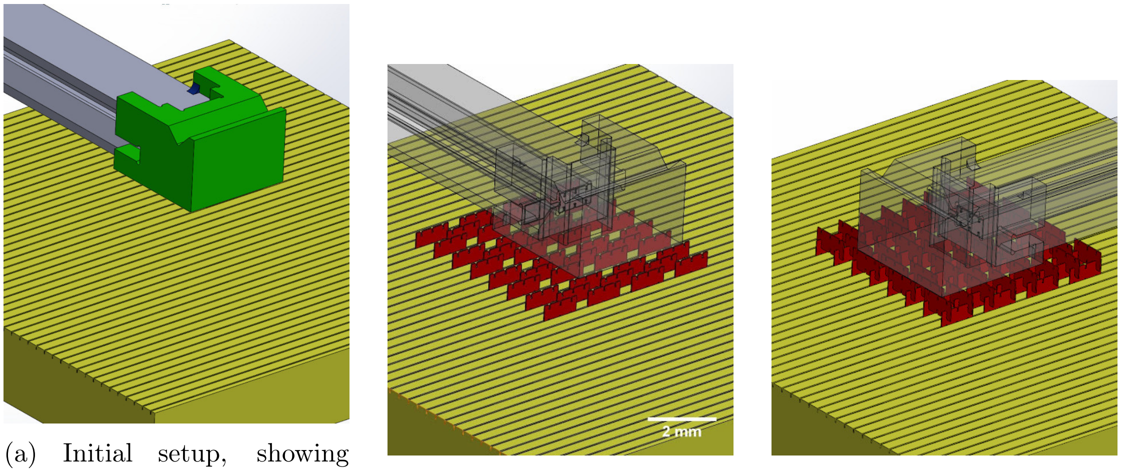
(a) Glitter dispensing magazine with green kinematic locating features at top and orange magazine follower at left.

(b) Close-up view of stapler head, showing glitter with loading tab still attached in red.

Figure 5-6: Conductive glitter magazine with integrated stapler head.

The magazine mounts kinematically using a Maxwell-type coupling, in which the three V-grooves on the magazine interface with three matching spheres on the assembly machine, constraining all six degrees of motion in a repeatable manner. The magazine is held in place with a powerful rare-earth magnet, allowing it to be quickly swapped to change between conductive and insulating parts. The assembly machine consists of the magazine and coupling, which translate vertically to add layers to the build lattice, and the build surface itself, which is mounted on an X/Y platform below the magazine. The build surface is a laminated assembly which has a series of grooves in one direction that match the lattice pitch and stock thickness, and are sized to provide a light friction fit for the assembled structure. During assembly, the stapler head moves horizontally relative to the build plate to assemble a complete layer, and then translates vertically to begin the next orthogonal layer. This process is depicted in Figure 5-7.

The maximum placement rate of the assembly machine will be limited by the stiffness of the motion control system and the speed of the actuators controlling the location of the placement head, along with the cycle time of the stapler mechanism itself.



(a) Initial setup, showing build plate in yellow and magazine in gray and green.

(b) Placing the first row of glitter (red).

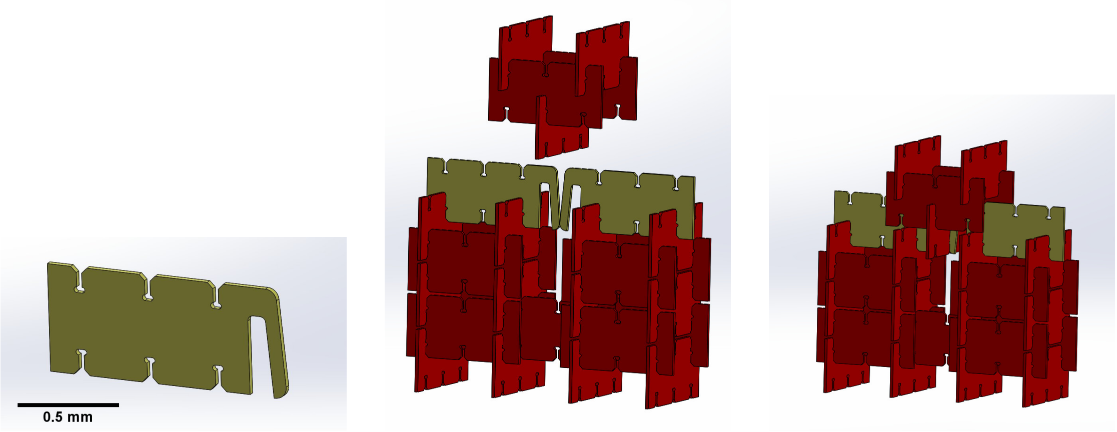
(c) Placing the second row of glitter at 90 degrees (red).

Figure 5-7: Automated electronic glitter lattice assembly using stapler-type plate dispenser.

5.3 Interconnect

Adding new building blocks to the glitter assembly ecosystem must be approached cautiously. Each new part type requires a dedicated magazine for placement; even if these assemblies can be automatically swapped during the build process, they add significant complexity to the assembly machine. A reasonable approach is to only add parts that provide unique and interesting inherent qualities tied to their construction material; for example, the conductive and insulating parts described earlier could be supplemented by actuation and flexure to provide motion, as shown by Langford in 2019 [50], and by resistive and semiconducting parts to provide passive and active circuit elements, as shown by Langford in 2014 [49]. Other parts could include environmental sensing elements or output devices whose function depend on construction material. A comparison can be made here to biology, in which 20 amino acids form the basis for life and differ in fundamental physical property, such as affinity for water or pH [38].

A worthy exception to this guideline is a dedicated conductive part which enables reversible module-to-module interconnect using flexural extensions. On the surface, such a part is similar to the standard conductive part since it is manufactured similarly out of identical material. However, as shown in Figure 5-8, the interconnect part interfaces with the flat edge of normal conductive parts using a flexural sliding interface rather than an interlocking sliding joint, so the joint characteristics can be tuned separately from the interlocking parts. This means we can get around the LEGO problem: it is difficult to assemble a LEGO structure using tools made from LEGO blocks if the tools pick up blocks in the same manner they are held together. Mechanically distinguishing inter- and intra-DICE module assembly is fundamental to enabling computational reconfiguration without risking module deconstruction.



(a) Interconnect part detail, showing scale and integrated flexural beam.

(b) Pre-assembly view showing two interconnect parts installed facing each other.

(c) Post-assembly view showing reversibly assembled structure.

Figure 5-8: Flexural interconnect glitter part used to join adjacent DICE nodes.

5.4 Scaling

A typical circuit fabricated in a FabLab is 50 x 50 mm, and consists of only one routing layer. With square millimeter routing nodes and connecting struts, this means an equivalent substrate fabricated from electronic glitter would require $50 \times 50 \times 5 = 12,500$ discrete conductive and insulating elements. Assuming a 1 Hz assembly rate, such a circuit could be prototyped in a few hours. Clearly, this will require at least some consideration for magazine changes; with 25 um parts, a reasonably sized 25 mm magazine would only hold 1000 elements, so this process will also require automation. A more complex circuit such as Meso-DICE, which is a 4-layer design which is roughly 25 x 25 mm would take up to twice as long, since each layer would be separated by a full layer of insulating elements. Of course, this assumes the components are mounted conventionally on a planar top layer; in reality, the structure would only be as tall as it needs to be, so the actual quantity of parts might be half or fewer than that estimate.

Beyond prototyping, simply adding additional assemblers would only increase throughput linearly; even 100 machines would only assemble 400 circuits in a typical 8-hour shift. A far more compelling approach mirrors the biological ribosome by recursing the assembly machines themselves: in other words, assemblers building assemblers, which could then scale exponentially given materials, energy, and space. In this arrangement, assembly machines would be built out of the same conductive and insulating parts they use to build circuits, taking advantage of two additional fundamental elements: flexural, to constrain motion; and mechanically active, to create motion. Such a system would require attention to a great many details, including part handling (since magazines would be fabricated out of the parts they hold) and large-scale motion logistics (since the machines would need to leave the build platform eventually). But a clear next step is to build a glitter assembly system using flexural elements, which is the motivation for the following chapters.

Chapter 6

Modular Superelastic Flexures

Most motion systems, such as the Cartesian and 6-DOF assembly systems presented previously, use rolling or sliding elements to constrain movement to desired rotary or linear axes. For macro- and meso-scale work such systems are ideal, since bearings and slides are ubiquitous, cheap, and sufficiently performant. In fact, the proliferation of low-cost 3D printers over the past ten years has meant a wide variety of linear rails, ball screws, and precision bearings are available in single-lot quantities through a multitude of international distributors. The DICE designs shown previously existed on the right scale for such machines; Meso-DICE in particular included self-alignment features large enough to tolerate millimeters of misalignment during assembly provided the tools included compliance. While that iteration used commercially available robotic arms for assembly, it would have been reasonable to quickly build a Cartesian assembler without too much effort.

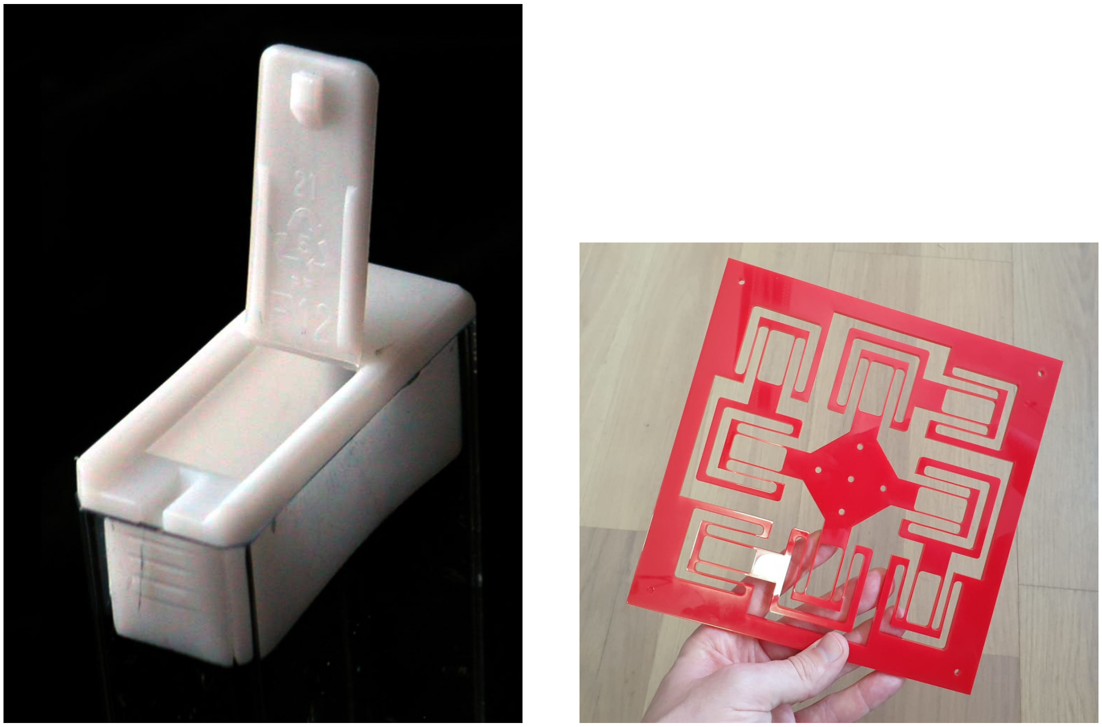
Reducing resolvable size below around 10 microns, however, becomes difficult with low-cost commercial components. While rotary actuators can be geared down to increase resolution, this strategy starts to introduce backlash and complexity into the system. Bearing and lead screw quality starts to play a larger role, and simple roller-on-extrusion motion constraints reach their limits due to inconsistent surface finish. Of course, many of these issues can be ameliorated through grinding, lapping, and other precision fabrication techniques. However, these methods are highly operator dependent and can require expensive specialized machines to accomplish.

Notably, the electronic glitter lattices presented previously are near this complexity inflection point. The discussed part geometry uses 25 um feedstock with matching slots, each of which has a 25 um chamfer to pull mating parts into alignment. Ideally, the assembly machine will have a positioning resolution several times smaller than that so fine adjustments can be made to minimize insertion force and error rate.

6.1 Flexures

Flexures are a method for constraining motion that provides an appealing alternative to conventional rotating or sliding joints, particularly at micron (or smaller) resolution [18]. These deliberately compliant features rely on reversible elastic deformation rather than sliding or rolling elements. The advantages of such a design are clear; flexures are highly repeatable, have practically zero backlash, require no lubrication, and can be modeled using simple beam equations. On the other hand, flexures cannot perfectly emulate rotary joint kinematics; even with exotic beam shapes, the virtual center of rotation of a flexure does not remain stationary through its whole range of motion [51]. While this motion is repeatable and predictable, the traced path of a flexural joint is not perfectly circular. Flexures are also non-trivial to fabricate; most are either bolted on pieces of flexible shim stock, or are integrated into complex monolithic assemblies fabricated using wire-EDM, milling, laser cutting, or injection molding, as seen in Figure 6-1.

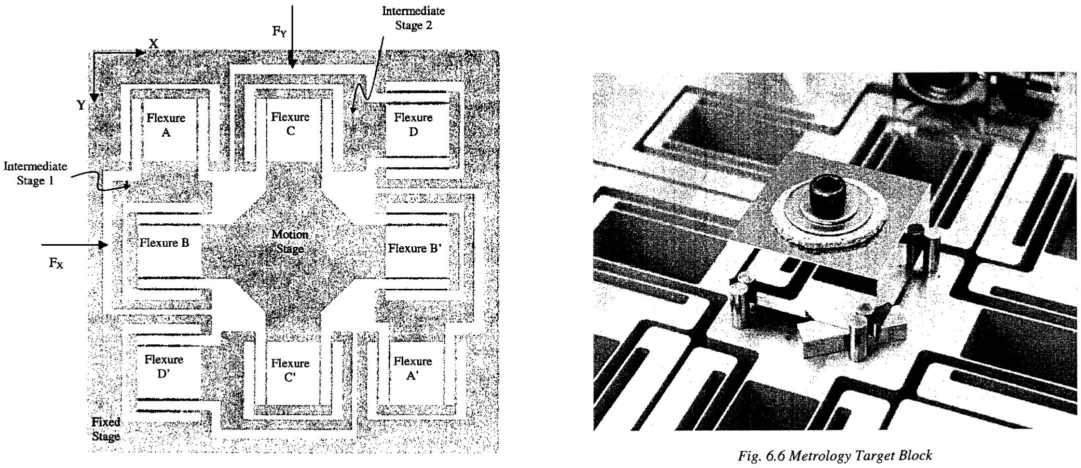
A far more limiting aspect of flexures is mechanical range, particularly for elements fabricated from stiff metal. Typical stainless steel alloys yield at less than 1% strain; beyond this, unrecoverable plastic deformation takes place which adversely affects the repeatability of the system. Worse, fatigue concerns usually push designers to restrict mechanism displacement to one-third of the material's yield strain. Taken together, these limitations dramatically reduce the addressable workspace for a given mechanism. For example, Shorya Awtar's two-axis parallel XY stage [17], shown in Figure 6-2, measures 300 x 300 mm, but only has an addressable range of 5 x 5 mm. Notably, the aluminum beams that make up the flexural mechanisms have a length to width ratio of 76:1 (47.5 mm long and 0.625 mm wide). Decreasing this ratio would increase the stiffness of the machine and reduces overall size, but would also further restrict addressable range.



(a) The injection molded plastic lid from a Tic-Tac candy dispenser, showing an integrated flexural hinge [66].

(b) A 2-degree-of-freedom flexural stage from the Urumbu circuit mill project, laser-cut from acrylic [83].

Figure 6-1: Examples of monolithic flexure construction techniques.



(a) Stage diagram showing actuator locations (Fx and Fy), flexures, and intermediate stages.

(b) Image of actual test flexure, fabricated using wire-EDM from a monolithic plate of 6061-T651 aluminum.

Figure 6-2: 2-axis flexural motion stage by Shorya Awtar [17]. Used with permission.

6.2 Superelastic Materials

The traditional flexural elements discussed above constrain motion by averaging many molecular deformations that occur throughout the material when loaded. As strain increases beyond the yield point, irreversible plastic deformation starts to complement this elastic behavior; when the load is removed, the material only springs back partway. An annotated stress-strain curve for such a material is shown in Figure 6-3.

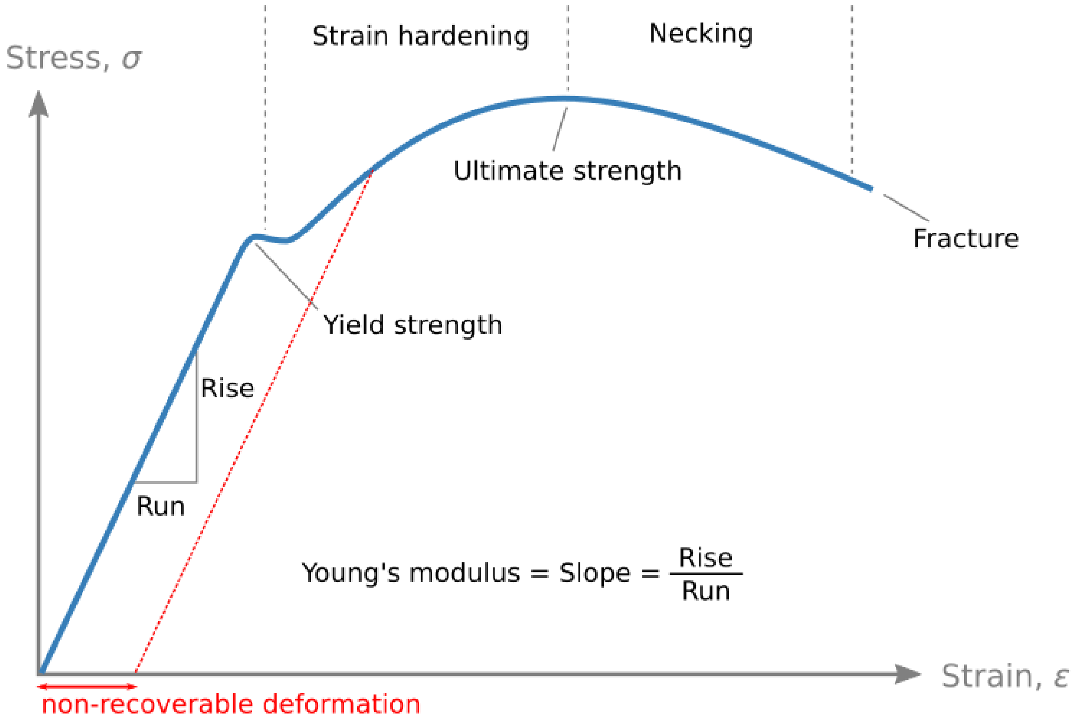


Figure 6-3: Stress-strain curve of a typical metal, annotated to show recoverable and non-recoverable displacement after load is removed. From [62], with added annotations in red.

A specialized type of materials called shape memory alloys (SMAs) exist which are capable of sustaining far greater reversible strain than conventional metals. This property, called superelasticity, is the result of a stress-induced crystallographic phase transformation. In the popular nickel-titanium SMA called nitinol (for Nickel Titanium Naval Ordinance Laboratory, where it was discovered), the stable austenitic phase phase reversibly changes to martensite, allowing for 6-8% strain recovery without permanent deformation. This behavior is illustrated in Figure 6-4.

Clearly, fabricating flexural elements out of superelastic materials would be beneficial, conceivably allowing far smaller flexural mechanisms to be designed. However, nitinol is difficult to manufacture and generally used in wire or sheet form; thicker stock is expensive and hard to find, with one vendor selling a 25 x 150 x 10 mm bar for $299. This is roughly 200 times more than extruded aluminum bar stock. In other words, Dr. Awtar's 2-axis stage

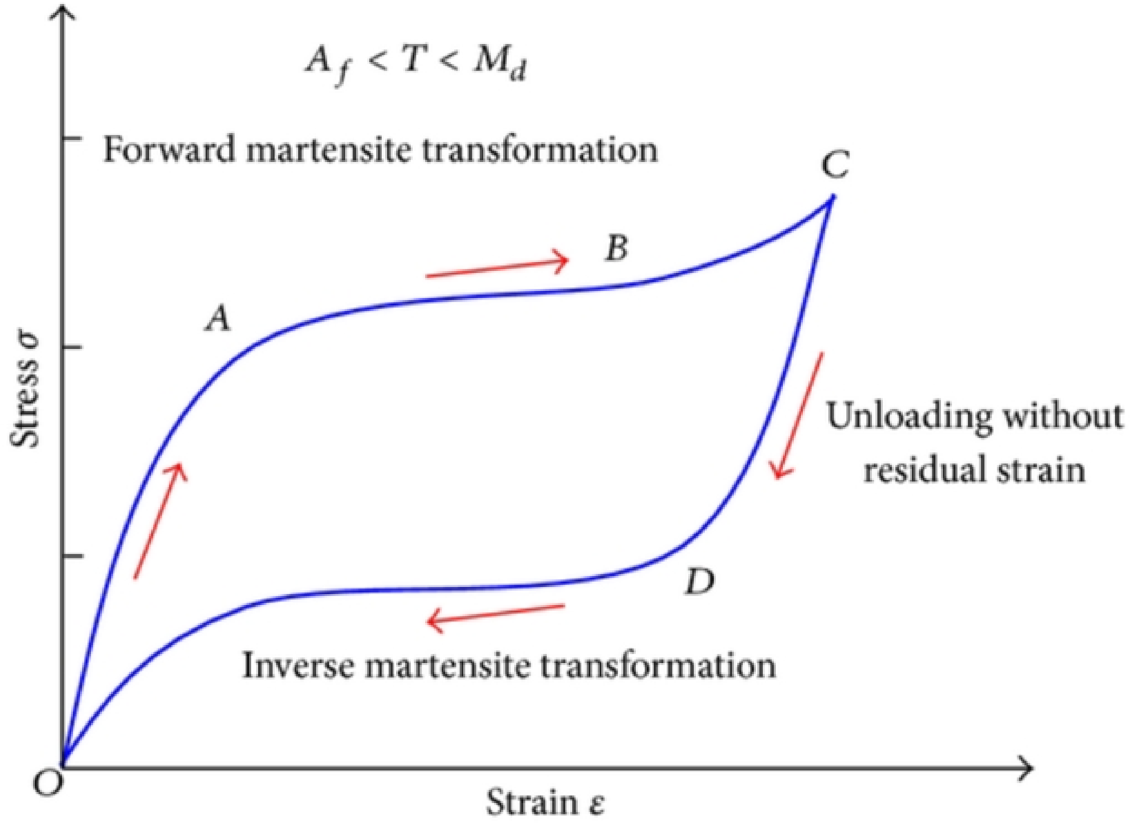


Figure 6-4: Stress-strain curve of a superelastic alloy, showing recoverable strain beyond the typical yield point of a conventional metal [69].

would increase in cost from $90 (if aluminum) to $18k (if nitinol), assuming 25 mm nitinol plate could be procured at the same volumetric cost. Calculations for nitinol, aluminum, and a few other metals are shown in Table 6.1.

6.3 Modularity

One approach to controlling high material costs is to introduce heterogeneity into the design. In other words, if the structural elements of a mechanism could be fabricated out of a cheaper material such as aluminum, the overall cost of the system could be dramatically reduced. However, this introduces the significant complication of joinery; the various materials must be connected in a sufficiently rigid manner so as to avoid introducing any unintentional flexibility into the system. Welding is easy to disqualify, due to the significant difficulty encountered in working with titanium alloys, along with the importance of precise heat treatment for superelastic materials (which a welding-induced heat-affected zone

Alloy	Cost per cubic centimeter	Cost of Dr. Awtar's stage
Al 6061-T6	$0.040	$90.60
304 Stainless	$0.084	$190
Invar 36	$0.704	$1590
Nitinol	$7.97	$17,900
Gold	$1,130	$2,530,000

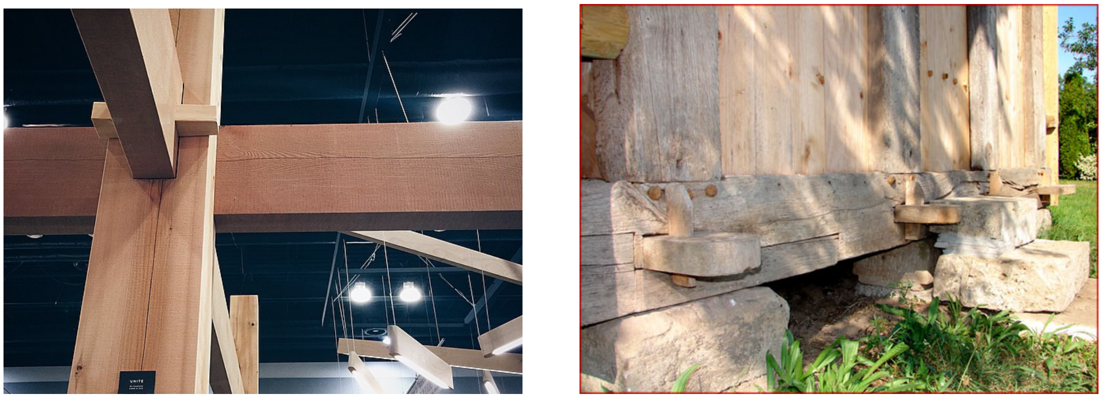
Table 6.1: Nitinol cost as compared to other metals and alloys.

would damage). Bolts are frequently used to assemble blade-type flexures with spring steel, but require substantial design considerations and can suffer from micro-slip if not torqued properly. Adhesives can be highly performant, but require careful surface preparation and can exhibit viscoelasticity under sustained load.

6.3.1 Orthogonal Taper Pin Joints

An alternative to the conventional metal-to-metal connection techniques outlined above draws inspiration from two unrelated sources: the aerospace industry and traditional wood joinery. Since the dawn of the jet age, one of the most important joints on an airplane has been the interface between a turbine blade and a turbine disk. While fabrication methods vary, blades are generally investment cast from superalloys capable of withstanding high-temperature creep; disks, on the other hand, are forged and machined titanium. The blades are replaceable, but once installed must withstand massive centrifugal forces caused by the disk rotating about the turbine's axis. To minimize rotating mass, the joint must be as small as possible, but (particularly for compressor blades) must be able to accommodate the thermal expansion of the blade during service. To address all of these demands, blades are secured in machined slots around the disk perimeter, using a bulb, a dovetail, or most recently, a "fir-tree" arrangement.

Wood joinery also makes frequent use of dovetail joints, most clearly seen to attach drawer faces to sides. Angled dovetails are relatively easy to fabricate by hand (via chisel) or machine (via router), and result in far stronger joints than simple finger joints which do not mechanically lock together. In some cases, such as complex 3- or 4-way wood beam joints, internal dovetails are supplemented by tapered wedges which are tapped in place after assembly to pull the joint together. Two examples of wedge-secured wood joints are shown in Figure 6-5.



(a) Reproduced traditional Japanese joinery, showing wedge securing method [41].

(b) Tenon joints from a French granary (public domain image).

Figure 6-5: Examples of wood joinery which makes use of simple wedges to secure pieces together.

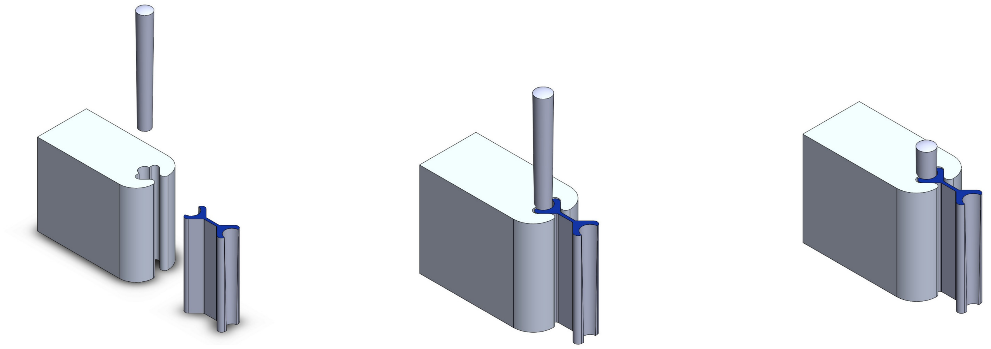
The wedge is an incredibly powerful simple machine. Discounting energy dissipation (via friction-induced heating) or storage (via elastic deformation of the wedge itself), a wedge is a linear force multiplier, where the mechanical advantage is simply the ratio of the width of the wedge versus its length:

$$MA = \frac{length}{width}$$

This phenomenon is used extensively in the wood examples shown above, where a wedge is inserted orthogonal to the joint's principle axis and tapped home. This imparts a higher magnitude axial force along the joint, pulling the wood beams together. Friction between the wedge and the substrate secure the joint until the wedge is tapped out.

The metal joint design demonstrated here uses commercially sourced round taper pins, which are typically ground from steel and have a 48:1 taper, providing a far greater mechanical advantage as compared to wood wedges. Conventionally, such taper pins are used to precisely and repeatably align parallel sheets of material during mechanical assembly, or to secure rotating elements onto shafts. Here, taper pins are used like wood wedges, providing orthogonal force amplification to the dovetail and relying on friction to stay in place. A diagram of the orthogonal taper pin joint is shown in Figure 6-6, and a dimensioned drawing of the flexural element is shown in Figure 6-7.

Importantly, the taper pins themselves must be hard relative to the structural elements and flexures. Commercially, hardened stainless steel and carbon steel taper pins are readily



(a) Taper pin, flexure, and structural piece awaiting assembly.

(b) Flexure inserted into dovetail and pin loose-fitted.

(c) Taper pin driven home, either via hammer taps or a tool.

Figure 6-6: Orthogonal taper pin joints at three stages of assembly.

available which fit this requirement. It is tempting to leverage the clock industry, which makes extensive use of soft brass taper pins for assembly tasks, as these pins are substantially cheaper than their ferrous counterparts. However, in testing the brass pins tended to plasticly deform and loosen over time, rather than elastically strain the dovetail elements and pin the assembly together with sustained force. Furthermore, brass pins are far easier to accidentally bend or break during assembly and disassembly, resulting in awkward and time-consuming pin extraction operations.

6.4 Fabrication

6.4.1 Flexural Elements

Flexural elements are prepared using a type of electrical discharge machining (EDM) that uses a hair-thin length of wire as the electrode. This method, called wire-EDM, can cut through virtually any conductive material with a high degree of precision. In use, the wire-EDM system maintains a voltage potential between the wire and the workpiece. As the wire is brought closer to the workpiece, the dielectric gap between the two conductors breaks down and allows a pulse of current to pass across the gap. This minute spark vaporizes a tiny section of the workpiece, which quickly condenses and solidifies in the dielectric fluid surrounding the cut. A high pressure jet of dielectric flushes the particle out of the gap to avoid shorting against the wire, and the process repeats thousands of times every second. Wire is continuously refreshed from a spool and collected in a waste bin for later recycling.

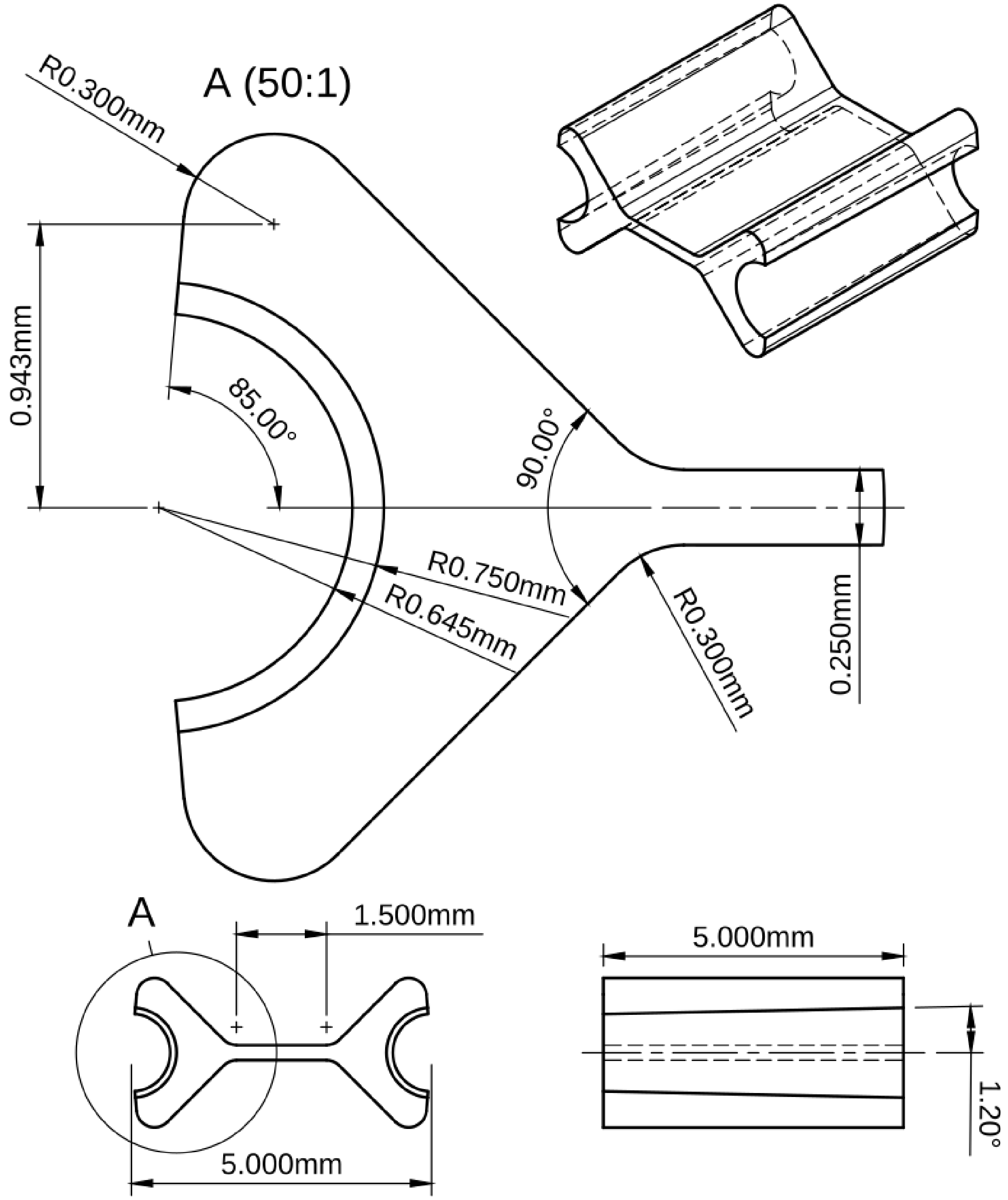


Figure 6-7: Dimensioned drawing of a modular superelastic flexure intended for orthogonal taper-pin installation [31].

As the gap between the wire and the workpiece grows, the voltage required to create the electrical discharge increases; a servoing circuit on the wire-EDM machine detects this change and advances the wire according to a predefined program.

Wire-EDM is good fit for prototyping nitinol flexures because it is a non-mechanical method, so it avoids the normal pitfalls associated with milling titanium alloys: tool wear, breakage, and overheating. The wire feed system on most machines, including the Sodick SL400G at the CBA, can be selectively tilted to create tapers on parts; in this way, both the straight edges of the flexural area and the precise 48:1 taper of the joint can be fabricated in a single operation. Using 150 um brass wire and typical machining settings, it takes roughly 15 minutes to fabricate a single flexure. Fortunately, many flexures can be linked together with tabs and fabricated together, and the machine can be left to run overnight. An image of many nitinol flexures machined in this manner is shown in Figure 6-8.

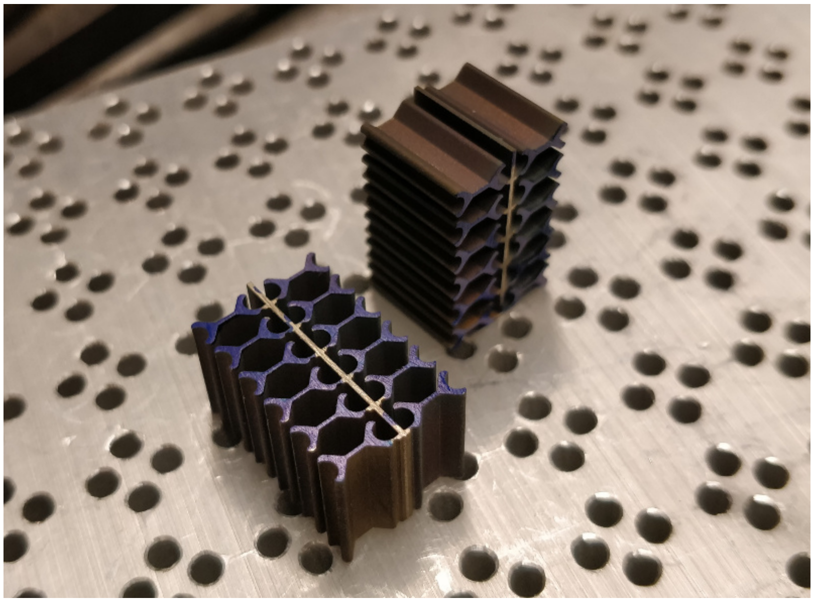


Figure 6-8: Twenty-four modular superelastic flexures machined in two batches. Note blue paint marks, which indicate the larger side of the tapered hole [31].

At scales beyond prototyping, other methods of fabrication would need to be explored. An obvious choice is investment casting, where a replica of the desired part is first fabricated from wax, which is then joined to a common "tree" with other identical parts. The assembly is then dipped in ceramic slurry and coated in sand; once this dries, the wax is burned out and the ceramic sintered in a furnace. Molten metal is then poured into the resulting cavity and allowed to cool, at which point the ceramic is broken away and the parts separated from the tree. One problem with this scheme is that molten titanium is highly reactive, so the melting and casting operation must take place in a vacuum furnace. While such facilities are commonly used to cast titanium elements for the aerospace industry, the capital costs of the equipment and the substantial pre-casting process makes investment cast parts extremely expensive.

A far better fabrication solution is to use metal injection molding (MIM). In this process, a powdered precursor alloy is compounded with a polymeric binder and pelletized, and then fed through a conventional injection molding system to produce "green" parts, which are then washed in a recirculating solvent bath to remove most of the binder. These delicate and porous "brown" parts are then sintered in a vacuum furnace, which burns away the remaining binder and fuses the metal into a densified structure that usually requires minimal post-secondary operations to meet final dimensional specifications. This method has been used to fabricate nitinol parts [75], but mechanical tests showed lower-than-expected elongation at failure suggesting further process development would be needed. If these limitations could be overcome, the per-unit price for the flexures would likely plummet as compared to investment cast or wire-EDMed parts.

6.4.2 Supporting Structures

One advantage to modularizing flexural systems is that the rigid supporting sections can be made from a variety of convenient stiff materials. In this case, extruded aluminum plate is ideal, as it is relatively cheap and easy to machine. As with the flexures themselves, the structural parts of the flexure system can be wire-EDMed, which provides outstanding dimensional control and includes the 48:1 taper to match the assembly pins. However, the structural elements are generally far larger than the flexures themselves, so the machining time is unacceptably long.

Another option is to cut the structural parts using a precision waterjet cutter. This machine uses an ultra-high pressure jet of water, normally around 50,000 psig, to accelerate a stream of 40-grit sharp garnet grains to roughly Mach 2. The jet is controlled by a gantry, which directs it to follow a cutting path defined by a part file. As with other cutting methods using lasers or plasma jets, the kerf tends to expand slightly throughout the cut; to counter this, higher-end waterjet cutters (like the one used here, an OMAX 5555) can be equipped with a pair of tilting axes near the nozzle to counteract taper. This produces high quality parts out of thick stock; importantly, the cutting rate exceeds wire-EDM by two orders of magnitude or more, turning overnight jobs into coffee-break-scale tasks.

Waterjet cutting is fundamentally a 2D process; while kerf angle can be reduced to negligible values using the tilting nozzle scheme outlined above, the mechanism is unsuitable for producing the precision 48:1 taper needed by the pins. Fortunately, the proliferation of taper pins for other purposes has resulted in the availability of matching 48:1 precision reams, designed to add a subtle angle to a drilled hole. Even better, spiral-pattern taper reams can be procured which are suitable for use in discontinuous holes that may cause an ordinary parallel ream to jam. In use, an extra superelastic flexure is inserted into a freshly water-jetted part; this element acts as a guide for the ream, which is then inserted and gently turned until the matte finish in the bore is completely shiny. This process is shown in Figure 6-9.

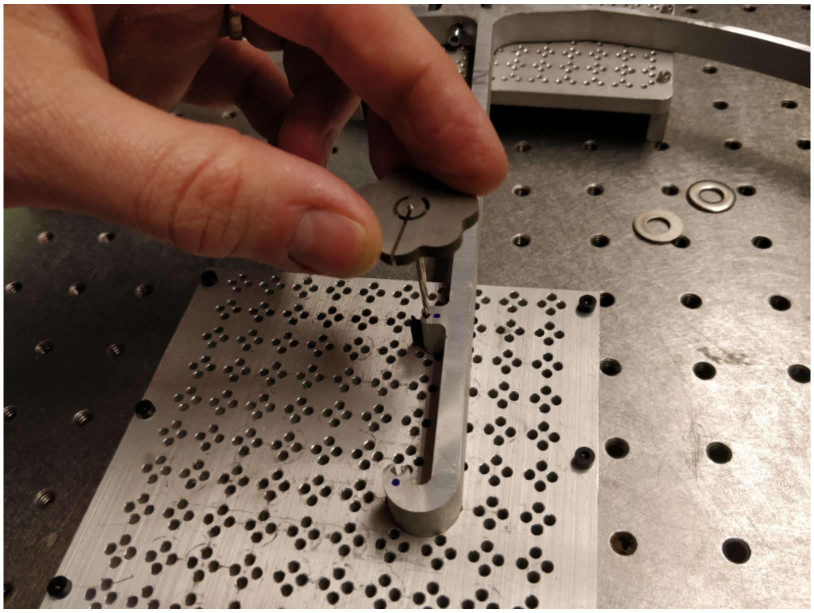


Figure 6-9: Hand-reaming a waterjet-cut aluminum frame to produce a taper suitable for orthogonal pinning. Note custom wire-EDMed handle on taper ream [31].

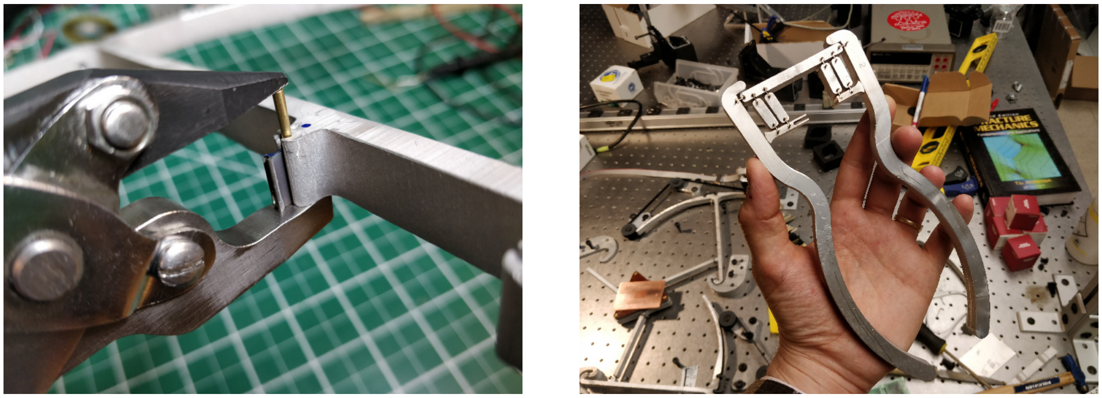
The fabrication methods described above are more suitable for mass production than wire-EDM, but water-jetting still suffers from two shortcomings: it requires a significant supply of garnet, which is relatively expensive, and it inevitably wastes raw material unless adjacent parts are perfectly tessellated on the stock sheet. To scale production economically, the clear choice here would be die casting, or for high performance applications, a more sophisticated but superficially similar process such as squeeze casting or thixoforming. Die casting and its relatives are ideal because dimensional control is more than sufficient for taper pin secured joinery, meaning parts would likely require little post-processing beyond aesthetic operations such as flash removal and light sanding. Another option would be to machine parts using conventional subtractive methods and then broach the dovetail features with a custom tool; this could be feasible if ferrous alloys (which cannot be die-cast) are required. However, this would still require a post-fabrication taper reaming step which would add cost and complexity.

6.4.3 Assembly

Building assemblies from modular superelastic flexures is simply a matter of driving the taper pins far enough into the joints that they lock in place. The simplest method is to use a ball peen hammer and, optionally, a tapered punch. First, the structural part and the flexure are assembled on a flat surface that has holes which allow the taper pins to protrude from the bottom. Next, a taper pin is inserted and pushed into place to temporarily hold the structure together and verify final alignment. Finally, the top of the pin is gently tapped until the assembler deems the pin secure enough for their application. If necessary, the punch can be used to drive the pin sub-flush; however, this is only advisable if the structural parts are thicker than the flexures (for example, using 12.7 mm aluminum plate and 10 mm nitinol). Removing taper pins is simply a matter of flipping the assembly over and tapping the pin free, again using the punch if necessary to free the flexure. Depending on the length of taper pin used, the dis-assembler must exercise caution to avoid bending the pin and accidentally locking it in place.

Assembling precision mechanisms with a hammer is not always favorable; in particular, the aforementioned method requires the use of a flat backup plate which may not be convenient. Another method is to use hand tools to apply precise force which gently pushes the pins home. Most "pinch-action" hand tools, such as channel-lock pliers, do not maintain parallel jaws throughout their range of motion, which quickly results in bent taper pins. However, the craft metalworking industry makes extensive use of parallel-jaw pliers that use sliding mechanisms to apply predictable parallel force across a 10 mm working distance. Using wire-EDM, a replacement jaw can be fabricated from 17-4 stainless steel which accommodates taper pins via a forked end, as seen in Figure 6-10. This arrangement can be conveniently used for both assembly and disassembly, provided the taper pins are long enough to remain proud of the structural elements.

An interesting extension of the custom jaw is a completely fabricated recursive assembly tool; in other words, a tool that can be used to assemble copies of itself. Such a design is also shown in Figure 6-10 and is made using the techniques outlined in this chapter:



(a) Pinsetter tool made by fabricating a custom lower jaw for a pair of common parallel-jaw pliers.

(b) Recursive pinsetter made with ten modular superelastic flexures and an aluminum frame.

Figure 6-10: Two "pinsetter" tools, used to quickly place and remove taper pins for securing modular superelastic flexures [31].

water-jetted aluminum plate for the handles, taper-reamed dovetail joints, wire-EDMed superelastic flexures, and taper pins (brass, in this case). The jaws themselves are also wire-EDMed out of 17-4 stainless steel, and take advantage of the material-agnostic nature of taper pin joints to join with the flexures in the same manner as the aluminum parts. This design is not particularly ergonomic (a bit of Plasti-Dip on the handles would go a long way), but it does work for assembly and disassembly of taper pin joints.

6.5 Fatigue

When designing around superelastic materials, the proverbial elephant in the room is fatigue life. By far the most common application for superelastic alloys is in the medical device industry, where they are used to fabricate highly collapsible stents that minimize incision size while maximizing arterial support [23]. In some cases, the effective fatigue cycle rate is the patient's heartbeat, since the stented artery expands with each blood pressure pulse and deforms the device. Such devices must reliably survive for the remaining life of the patient, which may end up being many millions of heartbeats. Unfortunately, the medical device industry is highly proprietary in nature and the methods used to guarantee high-cycle fatigue life in superelastic devices is a closely held trade secret. While academic literature points in the general direction of methods to increase fatigue performance, such as improving surface finish to reduce initial crack growth and carefully controlled heat treatment methods, specific guidance is noticeably lacking [92].

During early testing it was noticed that modular superelastic flexures do occasionally fracture after repeated use. In some cases the reason is obvious; they are accidentally overstrained or twisted off-axis when a mechanism is relocated. However, in other cases the failure was clearly fatigue, so further study was required.

6.5.1 Testing

A simple automatic testing apparatus was fabricated to quantify flexure fatigue failures. The machine consists of a large hobby servo which rotates a lever arm, which then connects via bearings to a second and third arm, forming a 3-bar parallel linkage. The first and third bars are the same length, so the final linkage angle mirrors the servo rotation angle exactly. This final linkage is replaced with the superelastic flexure under test. To facilitate swapping in new flexures, the base and final linkage are quickly removable so the taper pins can be easily accessed as seen in Figure 6-11

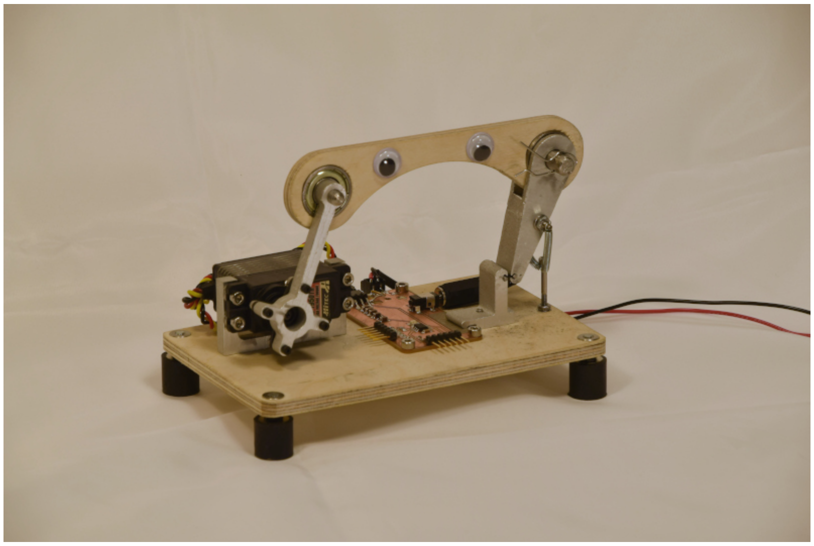


Figure 6-11: Automated fatigue tester used to evaluate modular superelastic flexures [31].

A Microchip 8-bit ATtiny412 microcontroller controls the servo motor, adjusting a pulse-width modulation (PWM) signal to rotate the actuator 20 degrees. Each time the servo cycles, the microcontroller sends the current count to a computer via a UART port and serial adapter. An automatic end-of-test function takes advantage of the conductivity of nitinol by monitoring electrical continuity across the flexure under test, and immediately stopping the servo when it detects an open circuit. Small tension springs attached to the third bar near the flexure ensure that it pulls away cleanly upon breakage.

Prior to building the automated testing apparatus, a conventional Instron 4411 material testing machine was used for the same measurements and produced limited qualitative results. However, the screw-driven nature of the Instron meant that the testing speed was quite slow; moving the test to the shop-built machine increased the testing speed twentyfold to roughly 4 Hz, and added automatic end-of-test detection. Both of these changes dramatically improved experiment productivity and quality.

6.5.2 Experiments

As discussed above, the most commonly cited method for improving nitinol fatigue life is to improve surface finish. While raw wire-EDMed parts are dimensionally accurate to within a few microns and visually smooth, they are microscopically pitted and rough due to the periodic nature of the spark erosion process. A scanning electron microscope (SEM) micrograph of an as-machined flexure surface is shown in Figure 6-12.

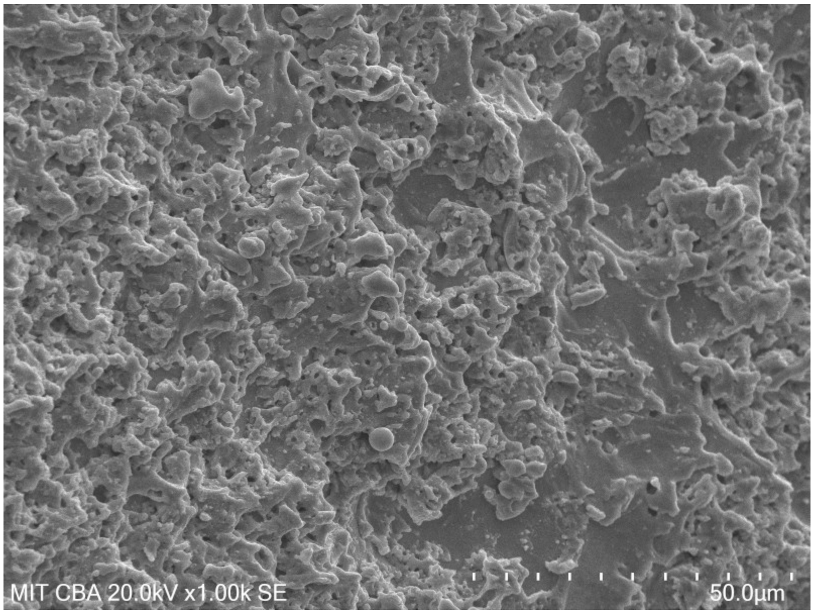


Figure 6-12: SEM micrograph of modular superelastic flexure as-machined surface, showing pitting and re-solidified debris from the wire-EDM process [31].

The first experiment using the Instron involved repeatedly pushing down on one side of a flexure with a polished plate, and using image analysis of a video recording to estimate the bend angle of the beam. During these tests, peak force was recorded and used to estimate the fracture point of the flexure, but determining the exact moment of failure was

Sample Type	Qty	Length	Cycles
Nitinol, 1 wire-EDM pass	7	5 mm	8223 ± 2510
Nitinol, 1 wire-EDM pass, annealed	2	5 mm	9814 ± 280
Nitinol, 2 wire-EDM passes	5	5 mm	9943 ± 3357
Nitinol, 3 wire-EDM passes	4	5 mm	7881 ± 796
Nitinol, 1 wire-EDM pass	4	6.5 mm	16,373 ± 4579
Nitinol, 1 wire-EDM pass	2	10 mm	17,875 ± 2181
6061 Aluminum, 1 wire-EDM pass	1	6.5 mm	1504
17-4 Stainless, 1 wire-EDM pass	1	6.5 mm	1970

Table 6.2: Fatigue testing flexures with various characteristics.

highly subjective. Generally, the force curves decreased quickly by 10-15% and held steady for some time, and then gradually fell off to zero. The first test examined three samples deflected to various angles, while the second compared a raw flexure to one that has been hand-polished with a 400-grit stone. In both cases, drawing conclusive results is difficult; but it does appear that no order-of-magnitude changes in fatigue life occur, with all of the flexures failing between 1000 and 3000 cycles.

A second round of experiments were performed once the aforementioned testing apparatus was fabricated. These tests surveyed a variety of strategies for improving the fatigue life of the flexures, including lengthening the beams while not changing the bend angle; adding wire-EDM finishing passes, a common method for reducing surface roughness with the fabrication method; and annealing the machined flexures in an atmospheric tube furnace at 550 C for 30 minutes, which are typical values for such work [25]. For comparison's sake, flexures were also fabricated out of 6061 aluminum and 17-4 stainless steel. A complete table of results is shown in Table 6.2.

Clearly, additional wire-EDM passes and annealing had no statistically significant effect on fatigue life. Increasing the flexure length appears to have improved performance to a degree; however, given the jump from 5 to 6.5 mm, it is reasonable to hypothesize that this trend should continue to longer lengths which is not reflected in the data.

6.5.3 Next Steps

Fortunately, while 10^4 cycles to failure is inadequate for a commercial product such as a medical stent, it is more than sufficient for building and testing flexural machines in a limited capacity, as seen in the following chapter. As such, continued work to improve the fatigue life of modular superelastic flexures can occur in parallel with mechanism development and characterization.

A clear first step is to quantify the surface roughness of current wire-EDMed flexures using a micron-scale technique such as confocal microscopy. While the Sodick SL400G used here has a built-in tool to predict roughness for a given number of wire-EDM passes, the actual surface quality of multi-pass cuts should also be examined. If the value does not markedly improve (for reference, the machine estimates a 4-fold improvement in roughness average for three passes), machining parameters should be adjusted to improve that result prior to fatigue testing.

One technique for improving nitinol surface roughness and fatigue life cited in literature [57] is electropolishing. This method was previously used at the CBA by Prashant Patil [64] for deburring laser-machined MEMS structures, and a similar regime exists with wire-EDMed surfaces: micron-scale debris and sharp edges that must be smoothed down. This specific process is also called anodic leveling, and occurs because current densities are increased at sharp corners and peaks. Unfortunately, electropolishing is generally only useful for surfaces that are already quite smooth, so this technique will likely need to be combined with a mechanical polishing step if wire-EDM parameter adjustments are not able to sufficiently reduce surface roughness.

Due to limited equipment availability, annealing was only explored briefly and did not produce a statistically significant shift in fatigue life. But nitinol can be annealed across a broad range of temperatures [25], so a wider sweep of post-machining annealing conditions should be performed. Another note is that annealing can effectively reset lingering defects from the phase transformation that enables superelastic behavior. As such, a worthwhile experiment will be to cycle a flexure below its expected breaking point (i.e 2-5000 cycles), then perform an annealing cycle, and then cycle the flexure for the same number of times again. If this process can be repeated indefinitely, an interesting option could be to electrically anneal flexures in situ: since nitinol is electrically conductive, part of routine maintenance of a mechanism could be to simply run current through the joints to heat them up for a period of time to reset their fatigue life.

Chapter 7

Compliant Machines

Compliant machines are assemblies of mechanisms built from flexural elements. This chapter presents three such machines of varying complexity which were used to characterize the performance of the previously described modular superelastic flexures, and to demonstrate their potential for use both in the electronics micro-assembly system proposed in Chapter 5 and in a community FabLab environment.

7.1 Single-Axis Flexure Test Machine

7.1.1 Description

The first machine is a four-bar flexural linkage driven by a stepper motor. This initial development iteration was used to validate the modular superelastic flexure concept generally, and to test for backlash (or lack thereof) and mechanical repeatability. Flexures are ideal for mechanically dividing using simple lever arms; in this case, the reduction ratio of the lever arm is 5:1, with the longer portion of the arm measuring 200 mm in length. A diagram the machine is shown in Figure 7-1, and an assembled image is shown in Figure 7-2.

A NEMA14 stepper motor drives a GT2 timing belt which provides the motive force to move the linkage. The far end of the long lever arm is an arc that sweeps about the flexural pivot through 90 degrees, onto which both ends of the timing belt are secured with an adjustable 3D-printed clip. Two idlers are arranged such that throughout a ±20 degree swing the timing belt stays tangent to the arc; in this way, the curvature functions as a wedge sliced from a circle, providing an additional mechanical reduction from the stepper's

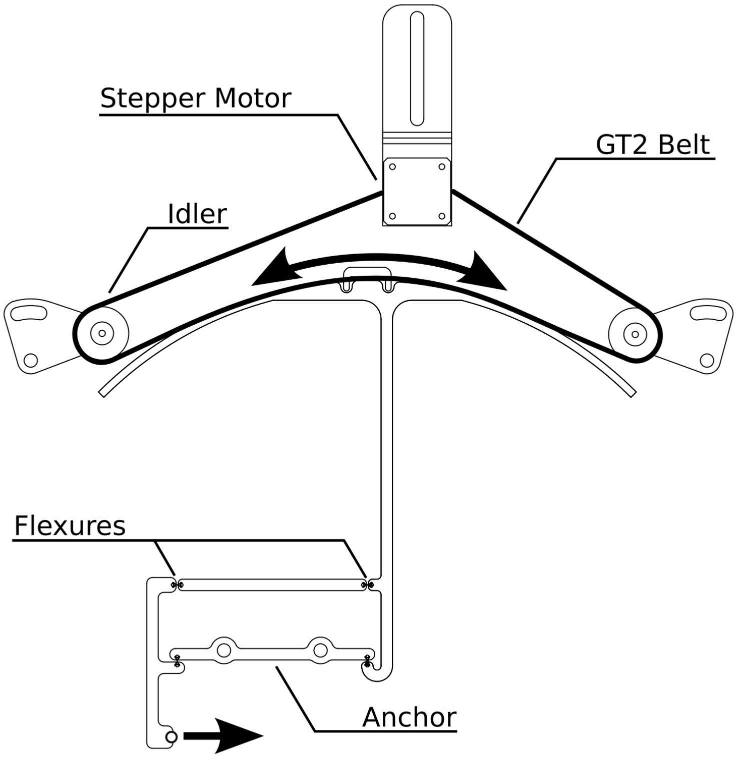


Figure 7-1: Actuator test machine diagram showing flexures, motor, idlers, belt, and anchor. Motion is indicated with arrows.

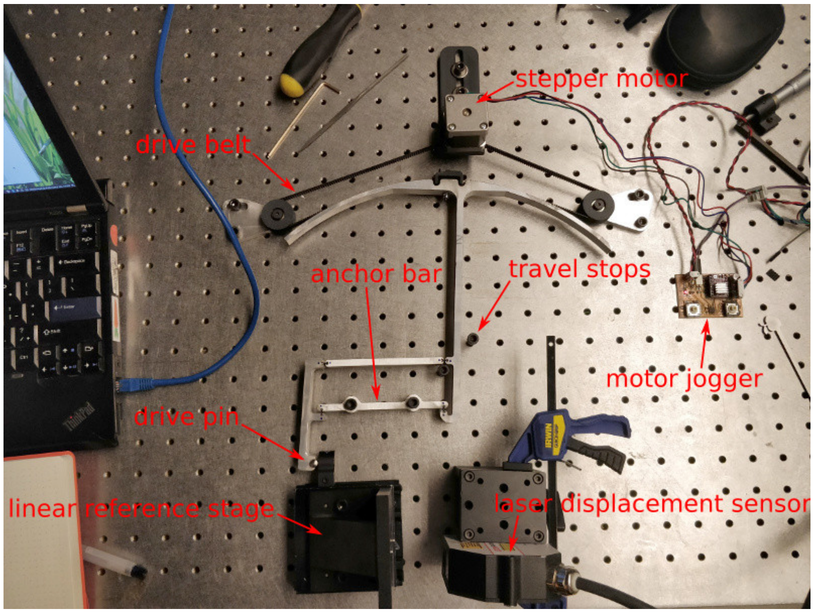


Figure 7-2: Annotated image of actuator test machine [31].

rotation. The constant-radius arrangement means the overall linkage reduction ratio does not change through the mechanism's range. It also means the required belt length is fixed, so tensioning is simply a matter of rotating and securing the idlers.

A commercially sourced precision linear stage is positioned near the short lever arm with its motion axis parallel to the anchor linkage. The short lever arm applies force to the stage using a press-fit hardened dowel pin; this joint rotates freely to avoid over-constraining the flexural mechanism against the linear stage. The linear stage holds a steel target block with is used as a target for a Micro-Epsilon NCDT2300 laser displacement sensor, a non-contact measurement instrument with 150 nm resolution and 2 um linearity across its 10 mm working range. A laptop connects to the sensor via Ethernet to provide a real-time data readout and data logging for offline analysis.

Stepper motors are more complicated to drive than DC brushed motors, which simply require a constant voltage supply to spin. In this case, the stepper motor connects to a simple PCB controlled by an ATtiny412 microcontroller. Two switches mounted to the board allow the experimenter to jog the motor in either direction by a fixed number of steps defined in the microcontroller's code, or sweep continuously in single-step increments interspersed with quarter-second delays. The motor itself receives drive pulses from a dedicated stepper driver module made by Pololu which uses a Texas Instruments (TI) integrated circuit called the DRV8825.

The structural elements, such as linkages and anchors, were waterjet cut out of 12.7 mm aluminum plate, taper-reamed, and assembled with 10 mm wire-EDMed modular superelastic flexures as described in the previous chapter. The anchor linkage and linear stage were bolted to a flat optical table, and the laser displacement sensor secured using a magnetic mount.

7.1.2 Analysis

The stepper uses a 16-tooth pinion whose pitch diameter can be calculated as:

$$D_{pitch} = \frac{N_{teeth} \times P}{\pi}$$

where N_{pitch} is the number of teeth on the sprocket and P is the belt pitch. For a 2 mm pitch GT2 belt, the stepper's pinion thus has a pitch diameter of 10.19 mm. The resulting reduction ratio can be calculated as:

$$X = \frac{R_{arm}}{R_{pinion}}$$

which solves to 39.25. Stepper motor rotation is broken into discrete steps based on the internal arrangement of permanent magnets and armature coils. The low-cost hobby motors used here have 200 steps per rotation, or:

$$\frac{360}{200} = 1.8deg/step$$

Because of the mechanical reduction provided by the belt, the long lever arm thus rotates about its flexural pivot at:

$$A_{rotation} = \frac{1.8}{39.25} = 0.04586deg/step$$

The 4-bar linkage uses equal-length opposing bars, so the mechanism is a simple parallelogram where diagonally opposed pairs of included angles are identical. Thus, the angle of the short lever arm which pushes on the linear stage is the same as the long lever arm's rotation. Since the end of the short lever arm rotates around a pivot, its path through space is not linear; as such, the stage does not move the same amount with each step.

When the flexures are straight and the long lever arm is exactly perpendicular to the linear stage, all four included angles of the linkage are 90 degrees. In this case, the short lever arm effectively only moves parallel to the linear stage. The predicted displacement of the stage can be calculated as:

$$Y = R_{arm} \times \sin A_{rotation}$$

In this case, $R_{arm} = 40mm$, so $Y = 32.02um$. The displacement of the stage per step will drop as the flexure angle becomes more extreme and cosine error is introduced. At 5 degrees, for example, the expected displacement per step is:

$$R_{arm} \times (\sin 5.04586 - \sin 5.00000) = 31.89um$$

so the error introduced by the rotating arm is 0.13 um. This error is on the order of the resolution of the laser displacement sensor.

7.1.3 Evaluation

First, the mechanism was evaluated for linearity by single-stepping the motor across the laser displacement sensor's 10 mm working range. A zoomed in detail of distance data from the sensor is shown in Figure 7-3. Notably, the mechanism shows significant "ringing" at the beginning of each step, later traced back to the spring loaded stage skipping off the dowel pin and slapping back down with each stop-start cycle. A simple Python script was used to convert this raw data into distinct step values. The script iterated through the data set and flagged any jump greater than 10 um in a single time step, and then threw out the following ten data points to eliminate ringing effects. The remaining values at each step were then averaged and the change with each step recorded. A histogram of the data is shown in Figure 7-4.

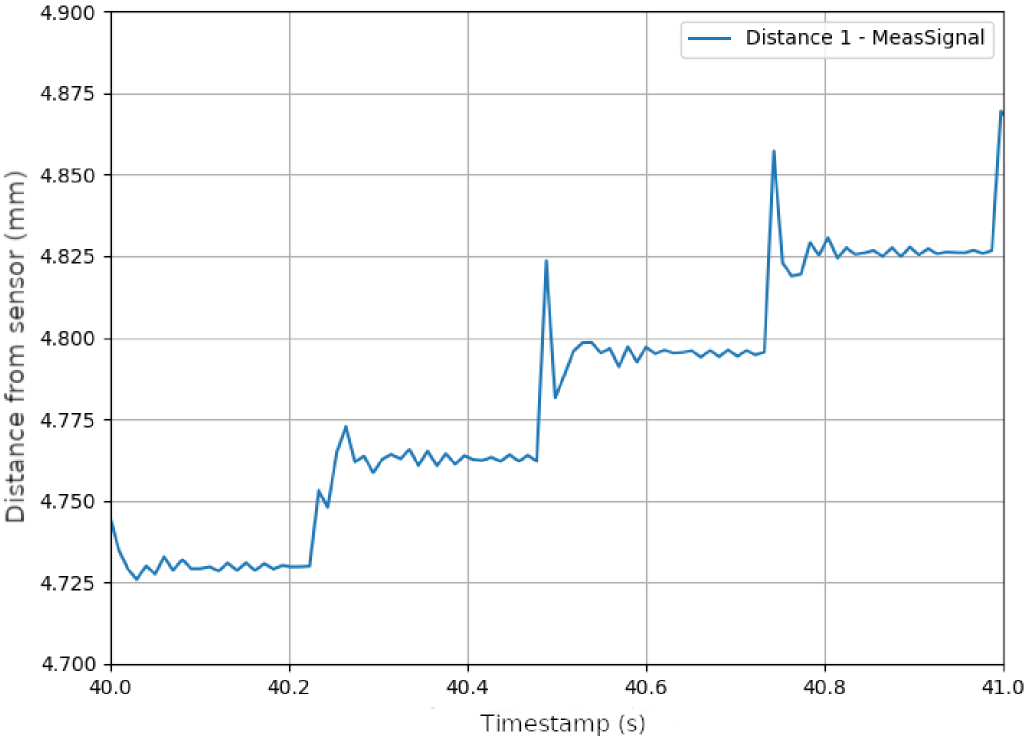


Figure 7-3: Several 250 ms steps of the linear actuator as measured with the laser displacement sensor, showing substantial ringing [31].

Clearly, measurement noise overwhelms the expected sub-micron error described above. Averaging the entire range of data results in a per-step displacement of $31.8 \pm 1.3 um$ which

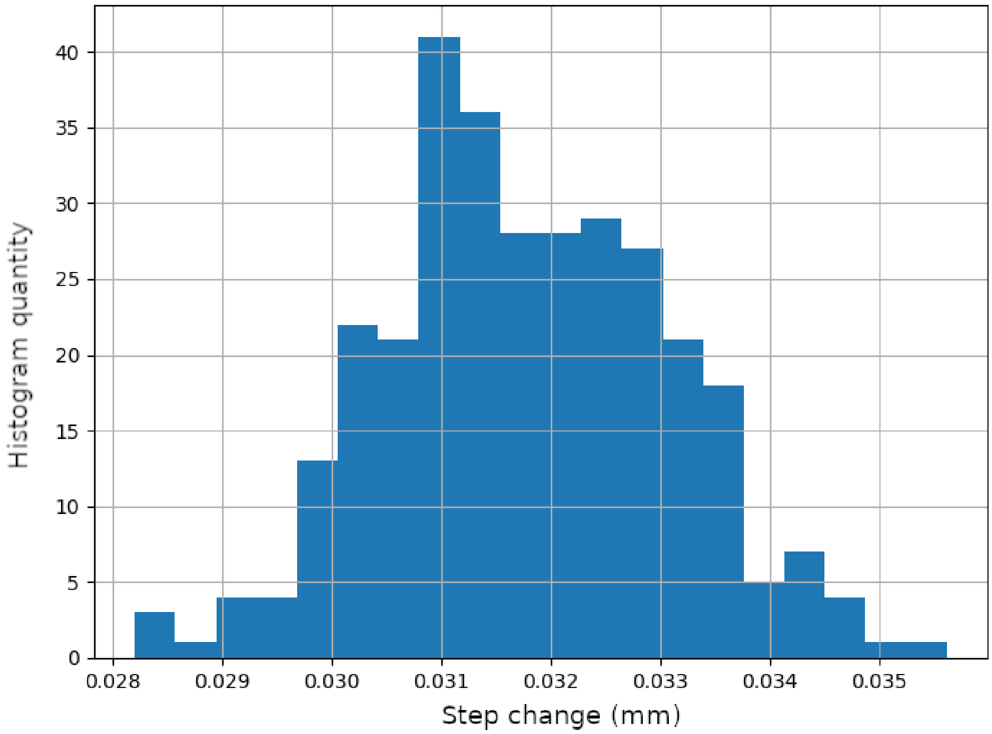


Figure 7-4: A histogram of the actuator displacement step size across the laser displacement sensor's 10 mm range, showing a narrow normal distribution centered at 31.8 um [31].

agrees closely with the predicted values. The relatively large standard deviation of the data could be the result of inadequate settling time for the ringing; alternatively, this could be a measurement-related error due to an imperfect target surface for the laser displacement sensor. Validation of the metrology strategy using a better measurement scheme such as a precision LVDT or an interferometer should be performed prior to assuming this result is due to unpredictable flexure behavior.

Subsequent tests were run in which the machine was cycled several steps from either direction towards a reference value to detect backlash. As a result of signal averaging, the variability of the measurement fell well short of the 150 nm device resolution again suggesting that backlash, if present, would require more precise tools to resolve.

7.2 3-RRR CPM

The second machine is a prototype of the planar motion system that will be used to assemble the electronic glitter lattices discussed previously. The desired work area is in the range of a centimeter or two; this roughly lines up with the maximum displacement of the first machine, so several parts of its design was reused in triplicate to avoid repeated work. In addition to further demonstrating the feasibility of modular superelastic flexures as a machine-building component, this experiment aims to characterize the repeatable distortion caused by the non-ideal nature of flexural pivots; that is, the imperfect kinematics caused by non-static virtual rotational centers. The machine is also tested for stiffness in several directions, benchmark values that can be used to compare future iterations.

7.2.1 Description

This machine falls into a category of mechanisms called compliant parallel manipulators, or CPMs [40]. Its architecture draws on previous work [92], changing the modular flexure design and replacing the prismatic actuator with a rotary joint. The system is compliant, because it uses flexures as opposed to sliding or rolling contacts; and it is parallel, because the three actuators act simultaneously on the same output stage, versus robotic arms which are serially oriented strings of rotary joints. Each linkage uses three rotary linkages, including the drive linkage; and there are three independent linkages total, hence 3-RRR. The flexures constrain motion in the Z-axis or rotation around the X- or Y-axis, but the final stage is able to translate along the XY plane and rotate around the Z-axis. The three actuators fully constrain these three degrees of freedom. A diagram of the machine is shown in 7-5, and a photograph of the system assembled on a flat optical bench is shown in 7-6.

As with the first machine, structural components were waterjet cut from 12.7 and 6.3 mm aluminum plate, and the nine modular superelastic flexures were wire-EDMed out of 10 mm nitinol. The build plate itself was milled out of 7 mm phenolic, and mounts to the machine's stage using three 250 um/turn micro-adjustment screws from Kozak Micro [4]. The hardened ball tips of the three micro-adjusters interface with three D-shaped inserts press-fit into the aluminum stage. These inserts have 90 degree grooves cut in the top face which are oriented at 120 degree angles with respect to each other, forming a Kelvin-type kinematic coupling. The inserts were wire-EDMed in two operations out of pre-hardened

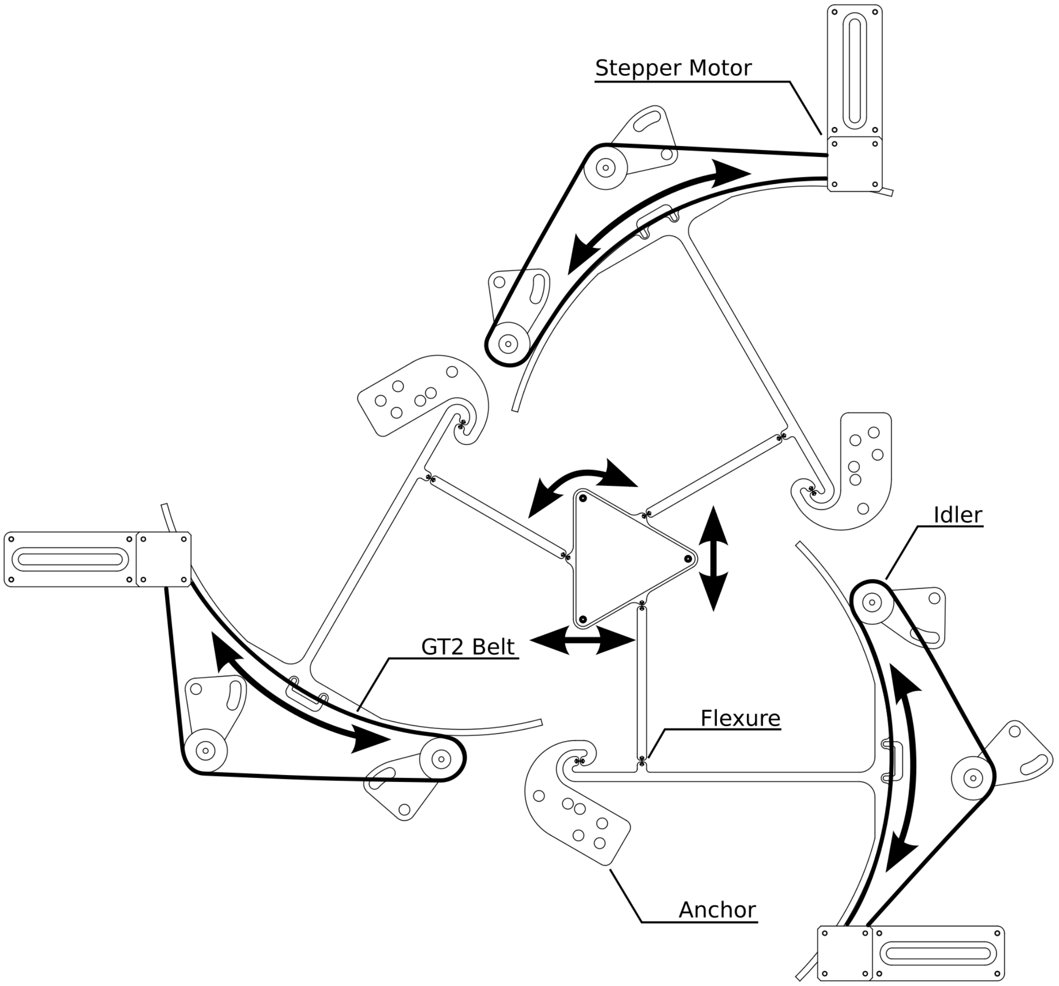


Figure 7-5: 3RRR CPM diagram showing flexures, motors, idlers, belts, and anchors. Motion is indicated with arrows.

4140 steel. The stage is kept in place by means of a pair of opposing 12.7 mm neodymium magnets, one press-fit into the stage and the other epoxied into a hole in the phenolic build plate. This arrangement allows the build plate to be quickly removed so samples can be mounted, and then replaced in a repeatable manner and leveled as needed using the micro-adjusters. A detailed view of the build plate and kinematic mount is shown in Figure 7-7.

7.2.2 Inverse Kinematics

Determining where the three actuators should go to produce a desired stage position (and angle) in Cartesian space means solving the inverse kinematic equations that describe the

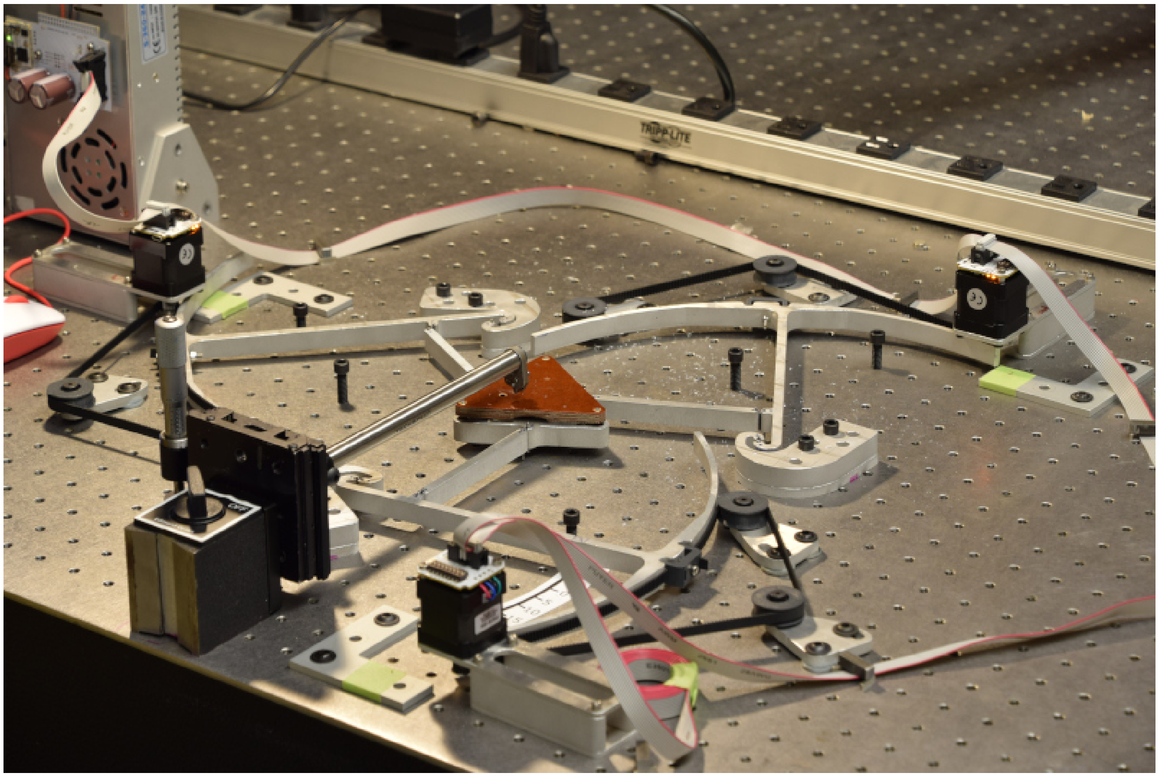
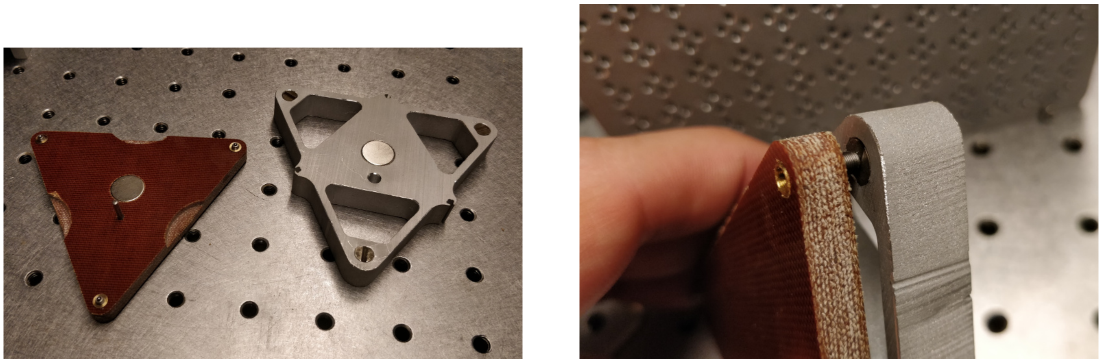


Figure 7-6: Image of installed 3RRR CPM, shown with control circuitry, build plate, and grating tool [31].



(a) Build plate (left) and stage (right) disassembled, showing kinematic inserts, magnet, and micro-adjusters.

(b) Close-up view of build plate mounted to stage, showing interface between micro-adjusters and kinematic inserts.

Figure 7-7: 3RRR CPM kinematic stage detail views. [31].

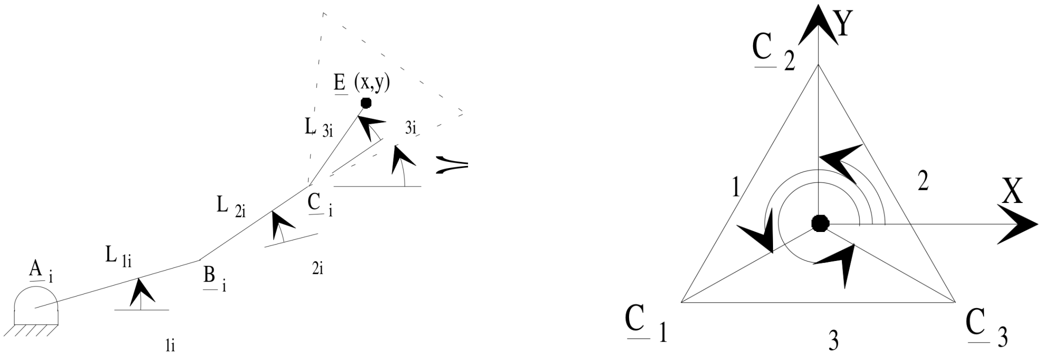
link system. The straightforward way to determine these values is via algebraic methods, discussed in [90] and used here.

The 3-RRR CPM can be divided into three identical linkages, anchored on one side and connected on the other to the common stage. A diagram of each arm, along with parameter naming conventions and the location of the origin, is shown in Figure 7-8. The values for

Parameter	Value	Description
θ_{max}	15 degrees	maximum allowable joint angle
L_1	40 mm	first link length
L_2	100 mm	second link length
L_3	33.6953 mm	distance from stage pivot to origin
A_{1x}	124.9124 mm	x position of first actuator pivot
A_{1y}	43.6453 mm	y position of first actuator pivot
A_{2x}	-24.6582 mm	x position of second actuator pivot
A_{2y}	130.0003 mm	y position of second actuator pivot
A_{3x}	-100.2545 mm	x position of third actuator pivot
A_{3y}	86.3546 mm	y position of third actuator pivot
ψ_1	57.0849 degrees	angle between x-axis and first link stage pivot
ψ_2	297.0849 degrees	angle between x-axis and second link stage pivot
ψ_3	177.0849 degrees	angle between x-axis and third link stage pivot
θ_{1_1}	120 degrees	angle between x-axis and actuator 1
θ_{1_2}	0 degrees	angle between x-axis and actuator 2
θ_{1_3}	240 degrees	angle between x-axis and actuator 3

Table 7.1: Parameters used in the 3-RRR CPM inverse kinematics model [31].

each parameter were pulled from measurements in the CAD model, and are shown in Table 7.1. Note that the length values were all calculated to the nominal center of rotation for each flexure.



(a) Figure 3 from [90], showing the kinematic diagram for a single arm of the 3RRR CPM.

(b) Figure 4 from [90], showing the rotation values ϕ for the three arms.

Figure 7-8: 3RRR CPM kinematic diagram from [90]. Used with permission.

Starting from an initial desired location in Cartesian space $\{x, y, \theta\}$, the stage corner locations C_{ix} and C_{iy} are calculated with respect to the origin for each link i:

$$C_{ix} = x + L_{3i}\cos(\phi + \psi_i)$$
$$C_{iy} = y + L_{3i}\sin(\phi + \psi_i)$$

Next, the two possible angle solutions ("elbow-up" and "elbow-down") are calculated using the half-angle substitution method [52]:

$$\theta_{1i_{1,2}} = 2\tan^{-1}\left(\frac{-F \pm \sqrt{E^2 + F^2 - G^2}}{G - E}\right)$$

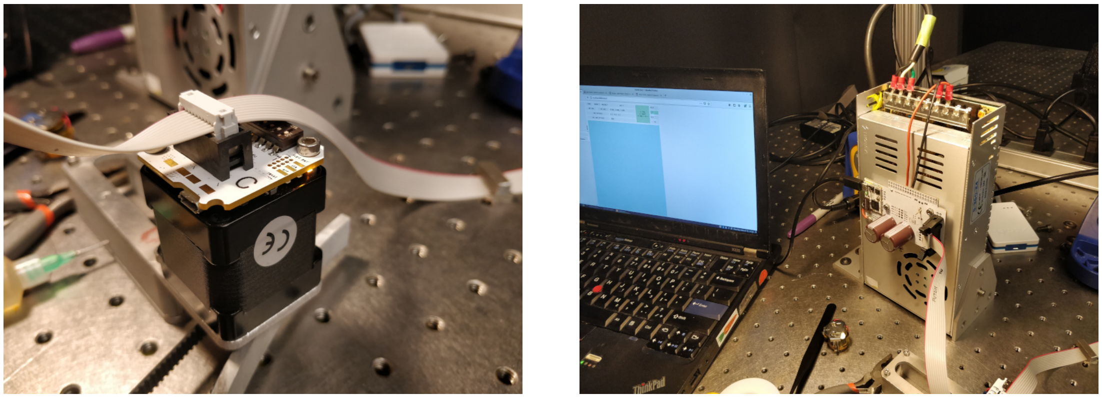
where the intermediate values E, F, and G are:

$$E = 2(C_{ix} - A_{ix})L_{1i}$$
$$F = 2(C_{iy} - A_{iy})L_{1i}$$
$$G = L_{2i}^2 - L_{1i}^2 - (C_{ix} - A_{ix})^2 - (C_{iy} - A_{iy})^2$$

A Python script is used to iteratively call these functions on a given pandas dataframe of desired coordinates. Before storing the resulting angle values, the two possible solutions are compared to the maximum joint angle and the appropriate one is chosen. If the desired coordinates over-rotate a joint in both solutions, the script throws an error and stops.

7.2.3 Control

This machine makes extensive use of Jake Read's Clank [70] platform, which he developed for the remote pandemic version of the class How to Make (Almost) Anything. Clank is a desktop 3-axis Cartesian CNC router designed for fabricating small precise things like circuit boards and wax molds. The machine control architecture is unique, and builds on the work from [71]. The stepper motors are connected in a stateless dataflow network controlled by a web browser. Electrically, this takes the form of modules wired to each stepper that are then connected with a long piece of ribbon cable using insulation-displacement connectors (IDCs). This chain of control "nodes" is then connected to a "head" PCB, which connects via USB to the host computer (see Figure 7-9). A large 24 V DC power supply provides motor power to the nodes, while the USB connection provides logic power. Optionally, the motors can be equipped with shaft magnets that are picked up by a PCB-mounted high resolution magnetic encoder; with an added control loop, this allows them to function as closed-loop servo motors.



(a) A close-up of one NEMA14 stepper motor with control PCB and IDC cable connection.

(b) Head PCB mounted to power supply with control computer to the left.

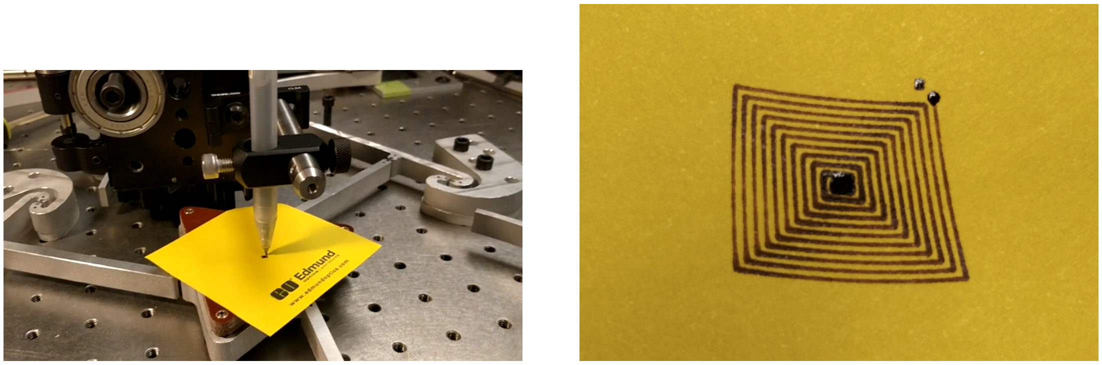
Figure 7-9: 3RRR CPM control system images [31].

In this adaptation of Clank, the three motors corresponding to the three compliant linkages are connected and reprogrammed to identify as the X, Y, and Z axes. The Clank web user interface (UI) is used as-is; it includes buttons for quickly jogging the machine along an axis, and a window for loading Gcode files for fabrication.

7.2.4 Evaluation

The machine was first tested with a pen writing on a sticky note adhered to the build plate. The pen was held vertically using a right-angle optical post adapter with a thumb screw, which was then screwed into a commercial linear stage. The stage was then held vertically with a magnetic clamp, such that the pen was directly over the sticky note. The stage includes a bias spring to hold it against a micro-adjuster, which was installed such that the spring tended to pull the pen off the paper. A series of heavy bearings were then hung on the stage, allowing for precise force control.

An additional function added to the Python script produced a simple square spiral pattern, which was then run through the inverse kinematics equations to produce stepper angles. These values were then written to a text file with the appropriate Gcode syntax characters and saved as a text file. A scaling value was also added to the inverse kinematics function so the same test could be executed at different scales. Figure 7-10 shows the setup and resulting drawn pattern from the pen. As expected, the spiral is clearly visible but has some distortion caused by the non-ideal rotational characteristics of the flexure joints.

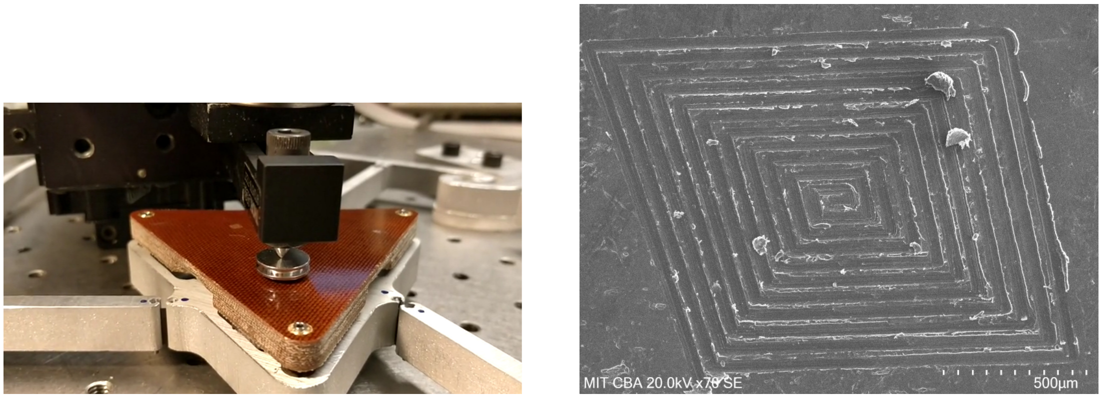


(a) Pen spiral test setup showing stage, pen holder, and bearings for weight.

(b) Drawn square spiral showing some distortion. Spiral is roughly 15 mm across.

Figure 7-10: Testing the 3RRR CPM with a pen [31].

Next, the same spiral pattern was scaled down by a factor of ten, resulting in an overall width of roughly 1.5 mm. The pen was replaced with a sharpened bolt, and a small hole was drilled in the build plate to accommodate a press-fit aluminum SEM sample holder. This setup and resulting pattern micrograph are shown in Figure 7-11. Again, there is some expected distortion present from the non-ideal joints. The distortion does appear to line up with the pen drawing, suggesting the distortion may scale up and require correction only at the macro level.



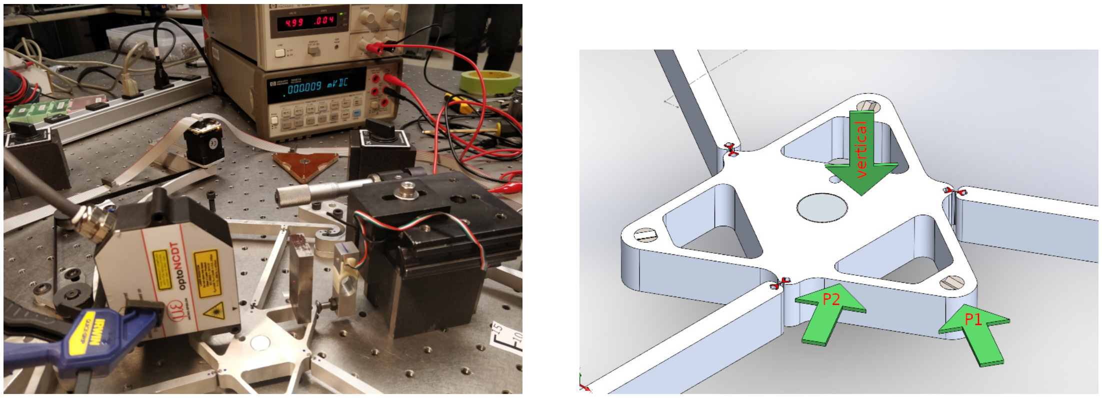
(a) Bolt test setup showing stage and bolt holder.

(b) Scribed square spiral showing similar distortion to the drawn version.

Figure 7-11: Testing the 3RRR CPM with a sharpened bolt [31].

The machine was also characterized for static stiffness as a benchmark for future iterations. To do this, the control system was first powered on to lock the motors. Next, a 5 kg capacity load cell was attached to a linear stage with a micrometer adjustment mechanism, which was then bolted to a magnetic anchor. A bolt was attached to the other side of the load cell such that force could be applied to objects from various directions depending on the orientation of the magnetic base. Before use, the load cell was calibrated by applying a 5 V DC bias voltage to two terminals and measuring the output voltage with no load, and again with 500, 1000, and 1500 g calibration weights hanging from the module. Fitting a linear regression to the data returned a load cell sensitivity of 1.0081 V/g.

To measure the displacement of the stage under various loads, the laser displacement sensor was affixed to another magnetic base and aimed at a steel target temporarily secured to the magnet at the center of the stage. The load cell apparatus was then oriented to apply force to the stage in two directions, one parallel to a side of the stage and one normal to the face. Vertical stiffness was characterized by turning the laser displacement sensor on end and adding weights to the stage. An image of the setup and the test directions is shown in Figure 7-12.



(a) Stiffness characterization setup showing load cell, laser displacement sensor, bias voltage supply, and readout voltmeter.

(b) Diagram showing three stiffness test axes. Planar tests were performed with the load cell, while vertical tests simply used weights.

Figure 7-12: 3RRR CPM stiffness characterization setup [31].

The horizontal stiffness tests were recorded in roughly 100 mV increments, corresponding to approximately 1 N of force. The vertical tests used the same 500 and 1000 g calibration weights as were used to calibrate the load cell. Resulting plots with fitted linear regressions are shown in Figures 7-13, 7-14, and 7-15.

The regressions visually fit the data well. Taking their slopes, the stiffness in direction P1 is 143 N/mm; P2 is 237 N/mm; and vertical is 110 N/mm.

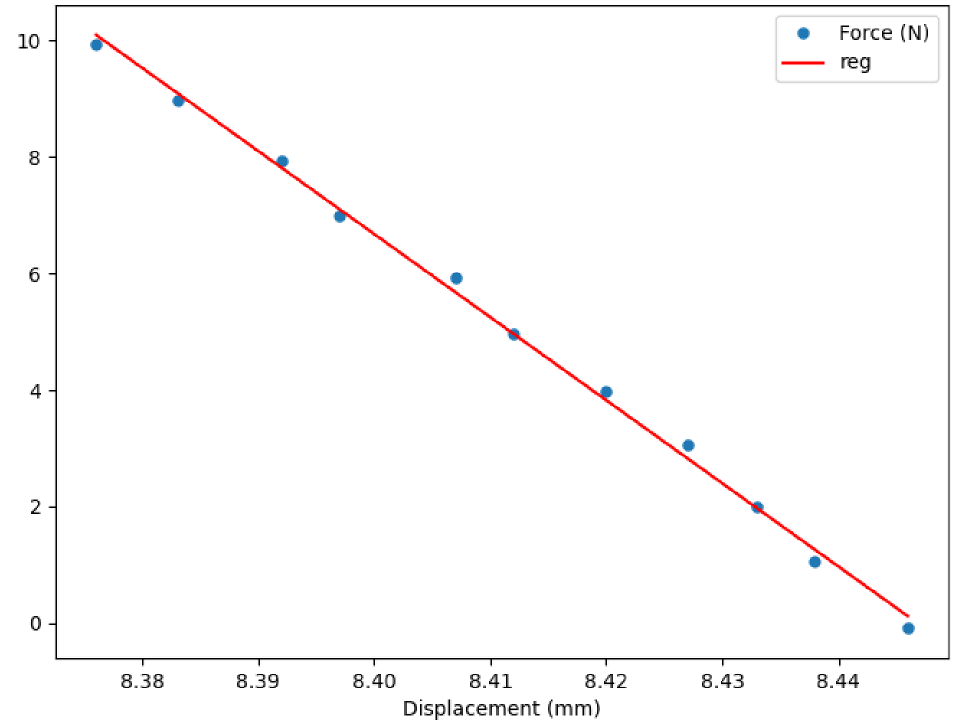


Figure 7-13: 3RRR CPM planar stiffness test results in direction P1 [31].

7.2.5 Computer Vision System

A computer vision system was installed in order to characterize and correct the distortion caused by non-ideal flexural pivots. The system consists of a Raspberry Pi 4 with 8 GB of RAM running Ubuntu 21.04; a Raspberry Pi High-Quality Camera with 12.3 megapixel resolution [30]; and a 50 mm CS-mount lens. The camera and computer were mounted to a rigid frame fabricated from modular extrusion.

The angular field of view $AFOV$ of a lens is:

$$AFOV = 2 \times \arctan(\frac{H}{2f})$$

where H is the sensor size and f is the focal length of the lens. The Sony IMX477 in the Raspberry Pi High Quality Camera measures 7.9 mm diagonally; thus, the lens and camera system has an angular field of view of 9.0 degrees. The field of view FOV can then

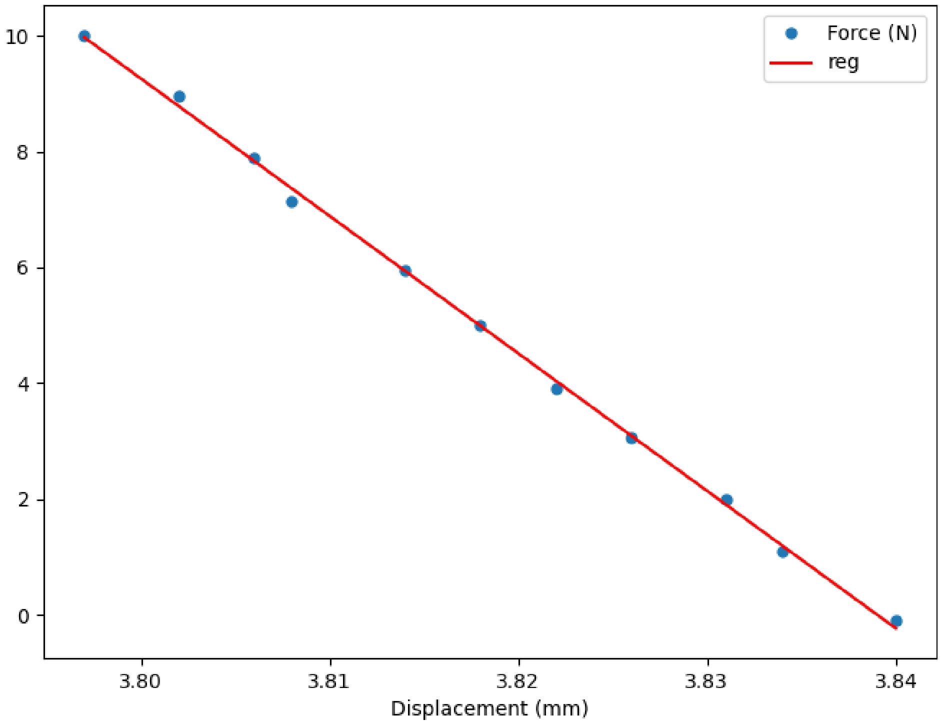


Figure 7-14: 3RRR CPM planar stiffness test results in direction P2 [31].

be calculated as:

$$FOV = 2 \times D_{working} \times \tan(\frac{AFOV}{2})$$

where $D_{working}$ is the working distance between the lens and the stage. For this setup, $D_{working}$ is approximately 400 mm, giving a diagonal field of view of 63.2 mm. Similarly, the effective resolution can be calculated from the IMX477's pixel size of 1.55 um square; in this case, the system should be able to resolve features down to 12.4 um.

In order to automatically detect the precise stage location and orientation, fiducials called ArUcO markers were used [34]. These markers are easy to scale and are often implemented for multi-object tracking; in this case, only a single marker was needed. A Python script was used to generate a valid marker pattern which was then raster-engraved on a polished piece of copper-clad FR1 circuit board material using a 532 nm nanosecond pulsed laser micro-

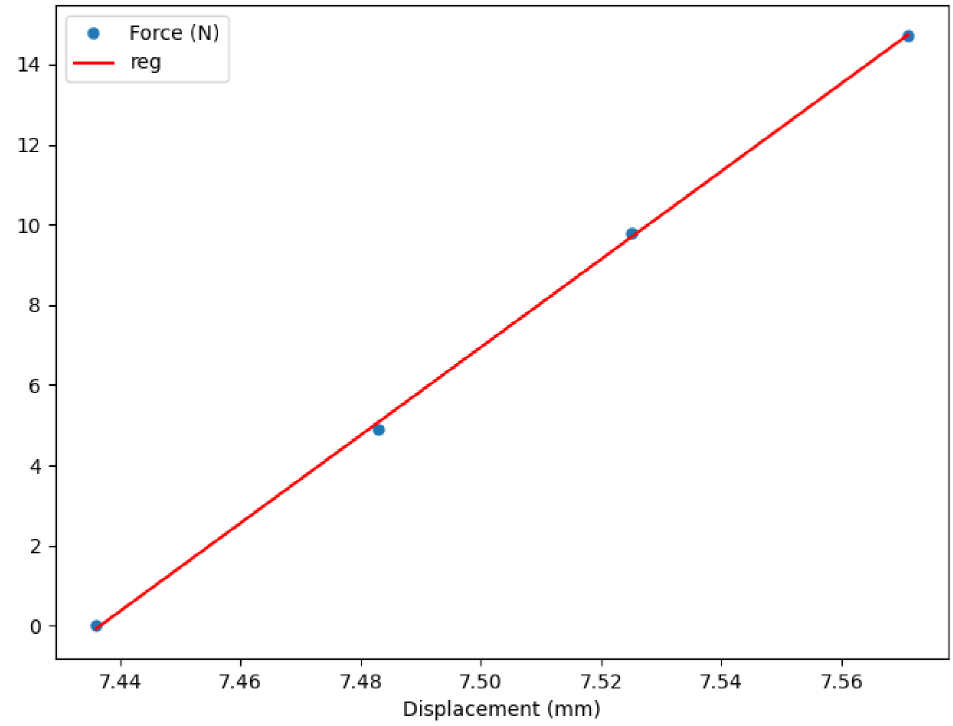


Figure 7-15: 3RRR CPM vertical stiffness test results [31].

machining system. OpenCV [8], an open-source computer vision application, was then used to identify the center location of the marker. An image from the computer vision system with an identification box around the ArUcO marker is shown in Figure 7-16.

To check the performance of the computer vision system, the 3RRR CPM was temporarily removed and replaced with a commercial linear stage, driven by a micrometer. The previously discussed laser displacement sensor was aimed at the stage to precisely measure its position. This setup is illustrated in Figure 7-17.

The linear stage was advanced in roughly 1 mm increments, and the laser displacement sensor data manually recorded and a corresponding image acquired. Plotting sensor data against ArUcO data yielded the expected straight line, as shown in Figure 7-18.

This process was repeated four additional times, covering most of the working area of the stage. The results are shown in Table 7.2, and suggest that the computer vision system is functioning at the level of performance predicted by the resolution calculations above.

Unfortunately, due to shortness of time the computer vision system was not utilized for

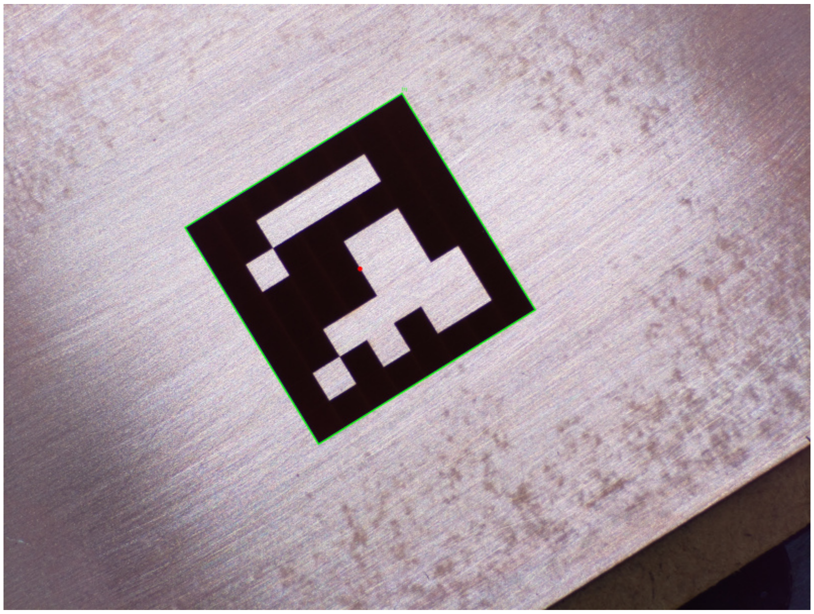


Figure 7-16: ArUcO marker laser-engraved on copper, as imaged and identified by the computer vision system [31].

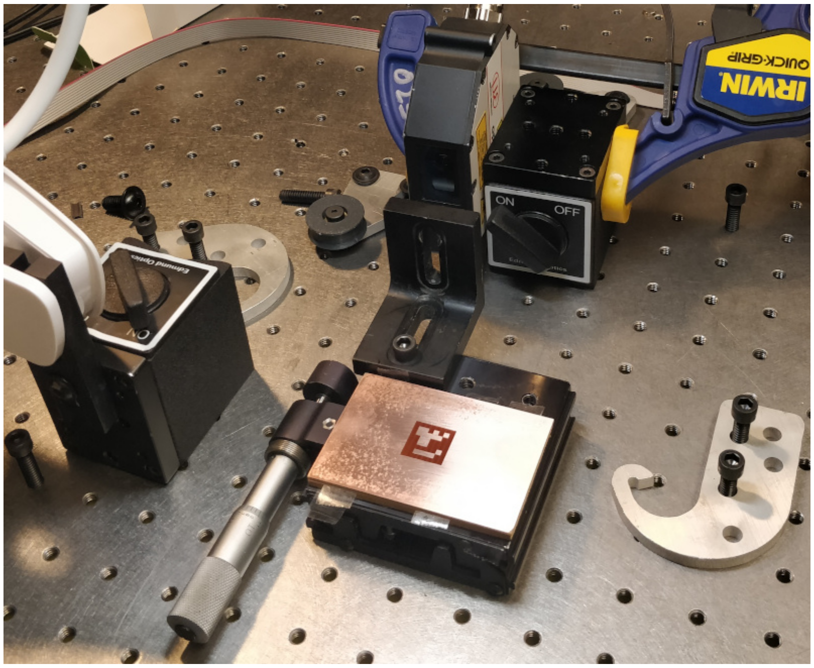


Figure 7-17: ArUcO calibration setup with linear stage and laser displacement sensor [31].

removing distortion from the motion system.

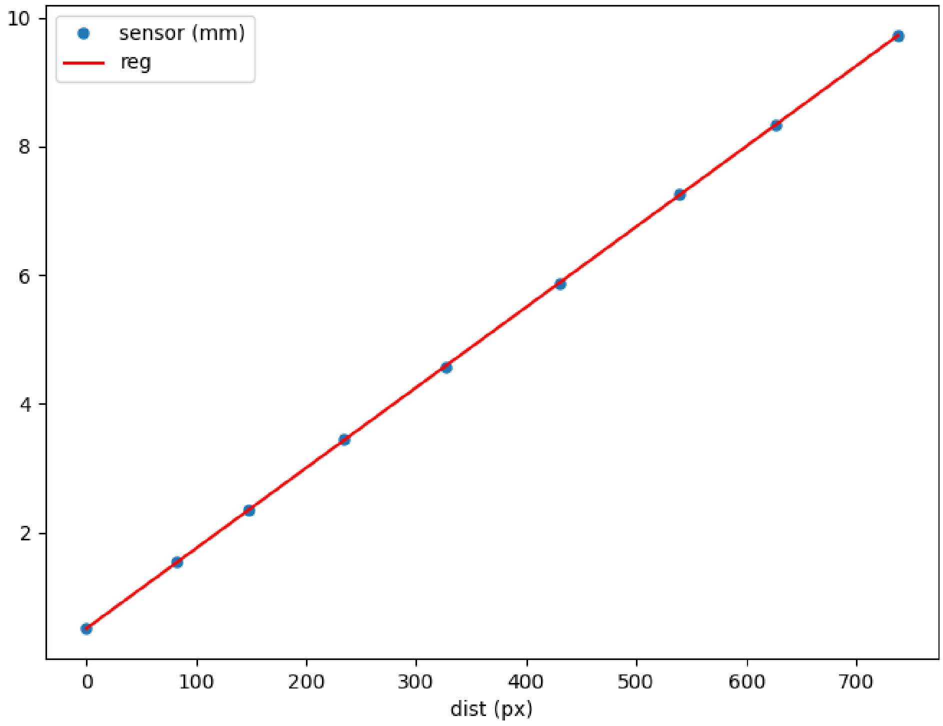


Figure 7-18: ArUcO calibration plot comparing laser displacement sensor values to computer vision measurements [31].

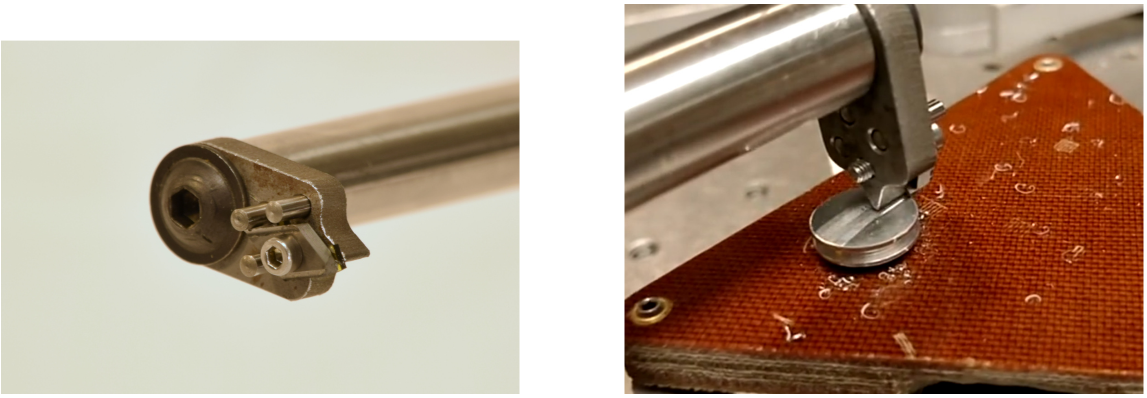
7.2.6 Ruling Diffraction Gratings

The 3RRR CPM was also evaluated qualitatively, to see if was sufficiently precise to mechanically rule a low-resolution diffraction grating capable of breaking white light into its constituent parts. In total, nine attempts were made with a variety of line spacing parameters and tool choices. Best results were obtained using a scrap of CVD diamond held in a purpose-built tool holder. This tool and a freeze-frame of the "ruling engine" in action is shown in Figure 7-19.

The fifth grating, with a groove spacing of roughly 40 um (or 25 lines/mm), proved functional if not particularly efficient or useful. Several images and SEM micrographs of the grating are shown in Figure 7-20. Notably, the head-on micrograph shows a pattern of micron-scale "waves", likely a result of individual motor steps at the resolution limit of the machine.

Test	Location	Direction	Slope (μm/px)	Error (stdev, μm)
1	top left	horizontal	12.50	8.34
2	bottom right	horizontal	12.50	6.15
3	bottom left	vertical	12.52	6.95
4	top right	vertical	12.57	10.53
5	center	diagonal	12.49	7.48

Table 7.2: Results from 3RRR CPM stage calibration tests [31].



(a) Grating ruling tool, initially a sharp carbide insert, later the piece of CVD diamond scrap shown here.

(b) Freeze-frame from grating ruling video showing tool scratching fine grooves into an aluminum SEM sample holder.

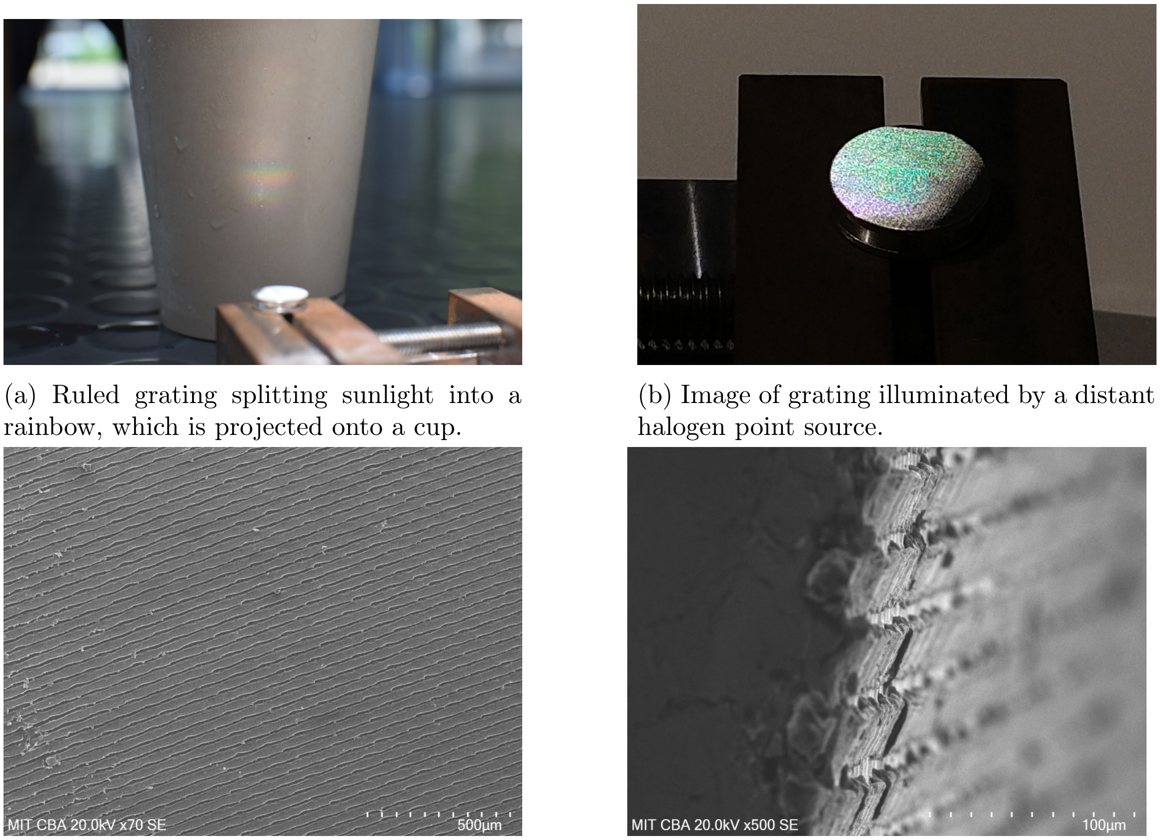
Figure 7-19: Ruling a primitive diffraction grating using the 3RRR CPM and a diamond tool [31].

7.3 MicroPanto

The third machine is a 50:1 pantograph reducer, designed shrink the working area of any desktop-scale CNC router with a corresponding decrease in resolution. MicroPanto [32] was designed and fabricated in roughly ten days, with half that time spent at the CBA preparing flexures and structural components and the other half spent at Haystack Mountain Craft School assembling and using the machine. This iteration was intended to examine the validity of the flexural pantograph concept as a micron-scale reduction system, and to further test modular superelastic flexures as a rapid machine prototyping platform.

7.3.1 Description

MicroPanto uses the same 12.7 mm aluminum structural stock as the previous two machines; however, to save weight, the pantograph arms are 8 mm pulltruded carbon fiber reinforced polymer (CFRP) tubes, which are epoxied into holes drilled in the aluminum parts. To



(a) Ruled grating splitting sunlight into a rainbow, which is projected onto a cup.

(b) Image of grating illuminated by a distant halogen point source.

(c) SEM micrograph of grating as viewed head-on, showing wavy distortion at the resolution limit of the machine.

(d) SEM micrograph of side of grating, showing consistent blaze angle and poor surface finish.

Figure 7-20: Ruled diffraction grating images and micrographs. [31].

achieve a 50:1 reduction ratio, the long arms of the pantograph measure 1000 mm from flexural pivot to flexural pivot, while the short arms measure 20 mm. On the drive ("large") side of the mechanism, another aluminum piece is fitted with a 6.3 mm ground shaft that can be mounted in a rotary tool equipped with a suitable collet. For the Haystack installation, the driving tool was a Handibot portable CNC router [9]. A picture of the overall setup is shown in Figure 7-21.

The "small" side of the mechanism is built on a 6.3 mm aluminum plate, which supports the pivot flexure through a stack of adjustable riser blocks. The plate also supports a miniature vise which is used to hold the workpiece. The working tool is a laser-turned CVD diamond mounted to a stainless steel flexure, which is actuated using a servo motor controlled by a simple ATtiny412-based circuit. The servo is activated by a microswitch temporarily mounted with magnets to the Handibot's Z-axis, allowing conventional tool-

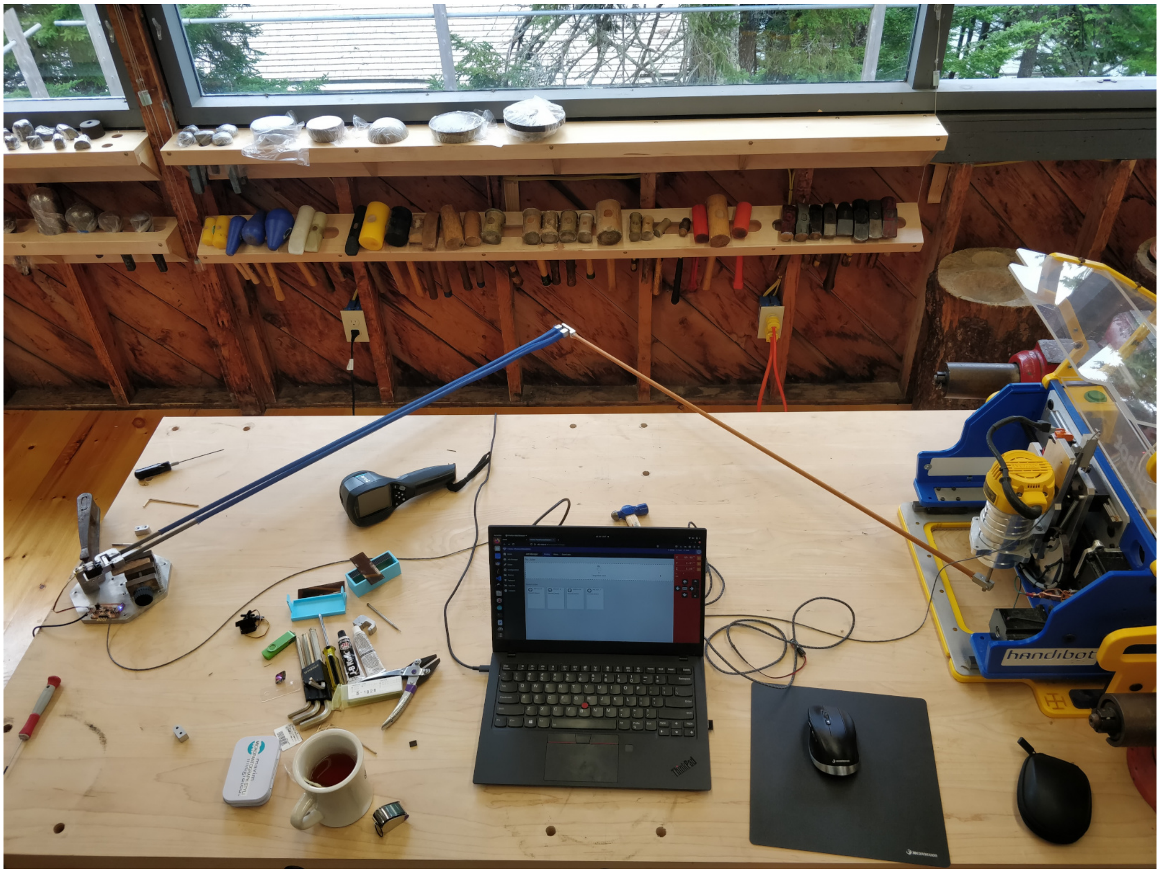
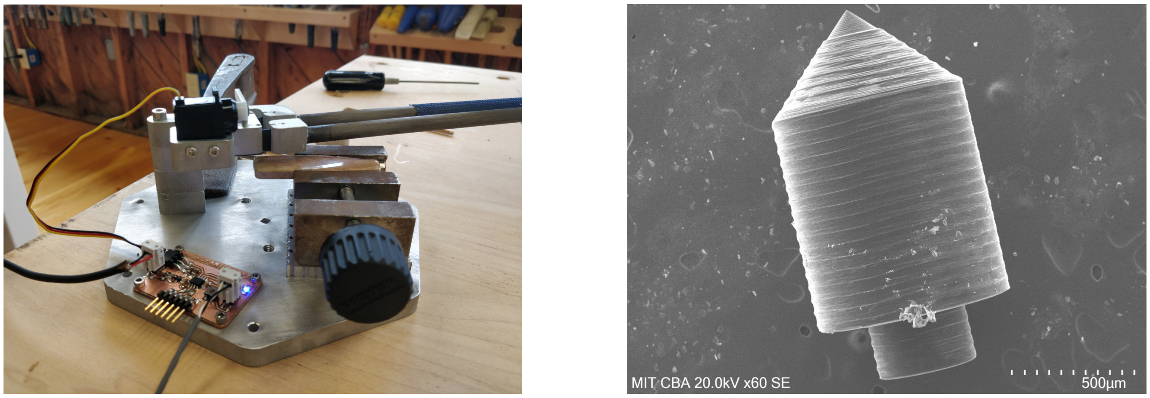


Figure 7-21: MicroPanto overview, showing Handibot CNC router on the right, control laptop at center, and engraving mechanism at left. Blue and orange rods are pulltruded CFRP tubes covered in cable loom to avoid splinters.

pathing techniques to be used without modification. A picture of the engraving head and an SEM micrograph of the diamond stylus are shown in Figure 7-22.

7.3.2 Micro-Engraving

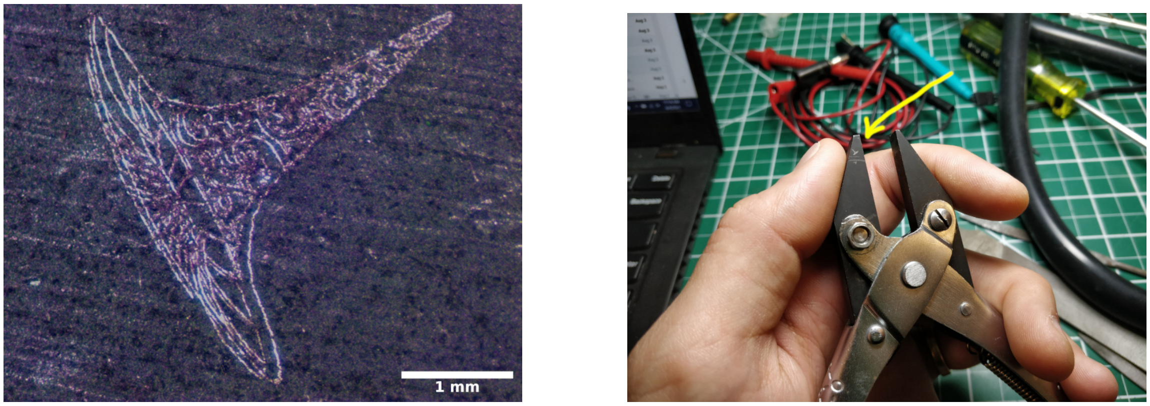
MicroPanto was not rigorously characterized because that wasn't the point of the project. Instead, the machine was used to engrave various designs on a variety of brought and found substrates in collaboration with several of the artists that joined on the trip to Haystack. An example micro-engraving on black oxide coated steel is shown in Figure 7-23, and several engraved and inked pieces of dry rice are shown in Figure 7-24. Generally, the machine performed quite well when properly adjusted, but would benefit from a gentler stylus flexure and more precise height control.



(a) Engraving head detail, showing base plate, control circuit, stylus servo, and workpiece.

(b) SEM micrograph of laser-turned diamond stylus. Note finishing pass on tip, providing 2 um radius.

Figure 7-22: MicroPanto construction details [32].



(a) Close-up image of engraved black hole, showing good detail but some evidence of work shifting during the operation.

(b) Zoomed out view of the same engraving, demonstrating scale. Workpiece is a pair of ever-useful parallel jaw pliers.

Figure 7-23: MicroPanto example engraving, using a design by Lauren Fensterstock [5] [32].

7.3.3 Implications

While MicroPanto is a fairly limited tool (it has a binary Z-axis, for example), it does extend the mechanically addressable resolution available to FabLabs. Other than document scanners and paper printers, FabLab machines are generally limited to tens- or hundreds of microns in minimal feature size and addressable resolution. With more sophisticated tooling and perhaps a larger reduction ratio, the MicroPanto concept is a clear path toward bringing FabLab capabilities firmly into the single-micron realm.

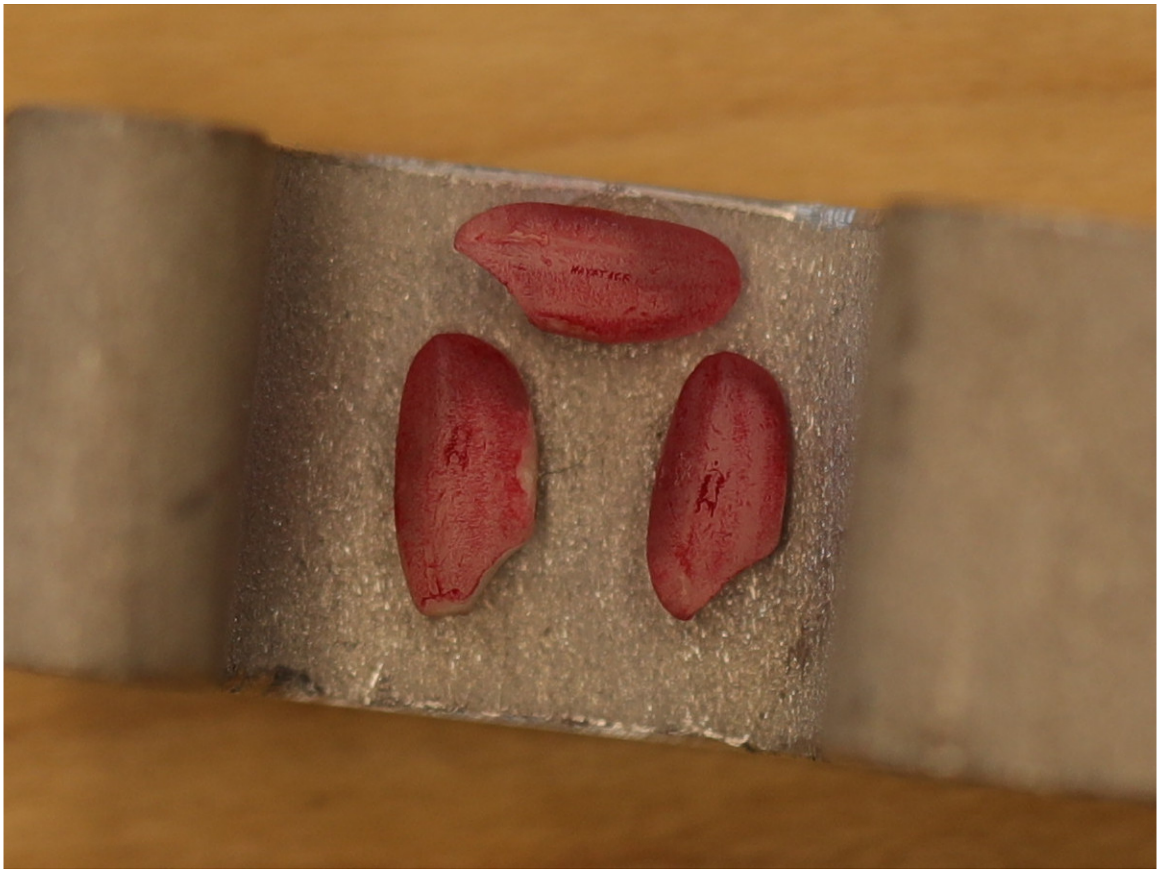


Figure 7-24: Three pieces of dry rice micro-engraved and inked to reveal detail. Two show Andrea Dezsö's [27] Forest Beings, while the third shows a hand-drawn "HAYSTACK" sign.

Chapter 8

Future Work

8.1 Glitter Fabrication

A clear next step is to commence fabrication and evaluation of the electronic glitter parts discussed in Chapter 5. As mentioned there, the covid-19 pandemic delayed delivery of a unique micro wire-EDM capable of cutting parts with 10 um tungsten wire. On paper, this plan seems sound; it's a straightforward scaling-down of previous electronic digital materials work [49]. On the other hand, scaling down is never that simple; the proposed feature sizes, for example, will be much closer to the grain size of the feedstock, which may lead to mechanical failures or unreliable results. Another concern is the current handling of the wire itself, which drops as the square of the diameter; this will reduce the cutting rate through the material and may require a tricky part transfer between the larger wire-EDM to minimize time spent cutting with tiny wire.

The proposed method for fabricating insulating parts is to micro-stamp them out of mica or a polymeric sheet stock. While early tests suggested that constraining tooling using flexural mechanisms may be feasible, the alignment requirements for parts at this scale are daunting and may require extensive mechanical development. Stamped parts also don't leave the machine nicely collated as with bulk wire-EDMed parts, so handling will need to be addressed. Again, the smaller scale of the parts means this will be complicated, since surface forces start to play a larger role in physical behavior.

Once a reasonable set of parts are produced, it will be exciting to explore automated assembly methods using the dispenser and assembly platform design outlined earlier. Here, perhaps the greater role of surface effects may be beneficial, if an interstitial material such as solder can be used to thermally re-align the parts after assembly through capillary forces.

8.2 Micro-DICE

As a reliable workflow for mass-producing and assembling electronic glitter lattices comes online, it will make sense to look forward to the next DICE iteration. Given sufficient resources, taping out a DEM ASIC as discussed in Chapter 2 would allow DICE to move beyond a demonstration and workflow development platform and into the realm of real HPC work. For a first order approximation, one can scale the NVIDIA V100's computational performance (8 TFlops), power consumption (300 W), die size (800 mm^2), and transistor count (21 billion) to a more modest technology node, such as Intel's 22 nm node from 2011 [63]. Assuming a linear reduction and up-sizing from the V100's 12 nm FinFET node, a 10 GFlops DEM ASIC would include 30 million transistors, require a 2 mm^2 die, and consume 700 mW. The DICE carrier would use roughly 10^3 conductive and insulating parts, or 250 mm^2 each of 25 μm-thick conductive and insulating material, worth $5 if purchased from research-grade sources [2] and taking 20 minutes to assemble at 1 Hz. A render of a proposed Micro-DICE node is shown in Figure 8-1.

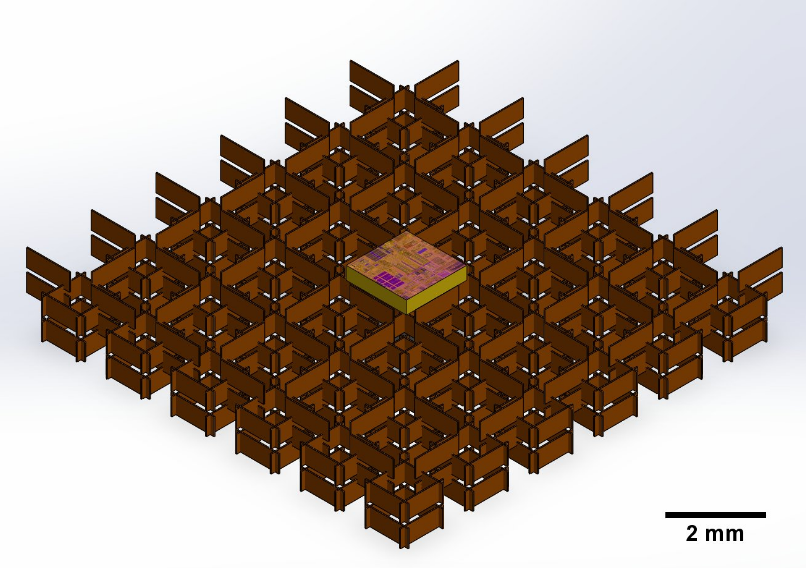


Figure 8-1: Render of a single Micro-DICE node, including a custom 10 GFlop DEM ASIC and electronic glitter lattice substrate/interconnect system.

It would take 800 Micro-DICE nodes to then equal the computational power of a single V100. Such a structure would measure 65 mm square and 36 mm tall, or roughly 13% of the volume of a dual-width PCIe V100 (which measures 40 mm x 111 mm x 267 mm [11]). While some additional space would be needed for a support plate and power input connection, it is notable that the V100's case does not include active cooling systems, so the comparison is surprisingly fair. Since the electronic glitter lattice has minimal cross sectional area in the vertical direction, forced liquid cooling through the lattice would be efficient and feasible. With the same economics as outlined above, the 800 Micro-DICE nodes would require 11 machine-days to assemble at a raw material cost of $4000. But notably, the scrap rate of the substrate material is many orders of magnitude cheaper than the research grade materials discussed here, and as discussed in Chapter 5, assembly machine recursion represents a clear potential path to an eventual reduction in module assembly time. A render of a V100-scale lattice of Micro-DICE nodes is shown in Figure 8-2.

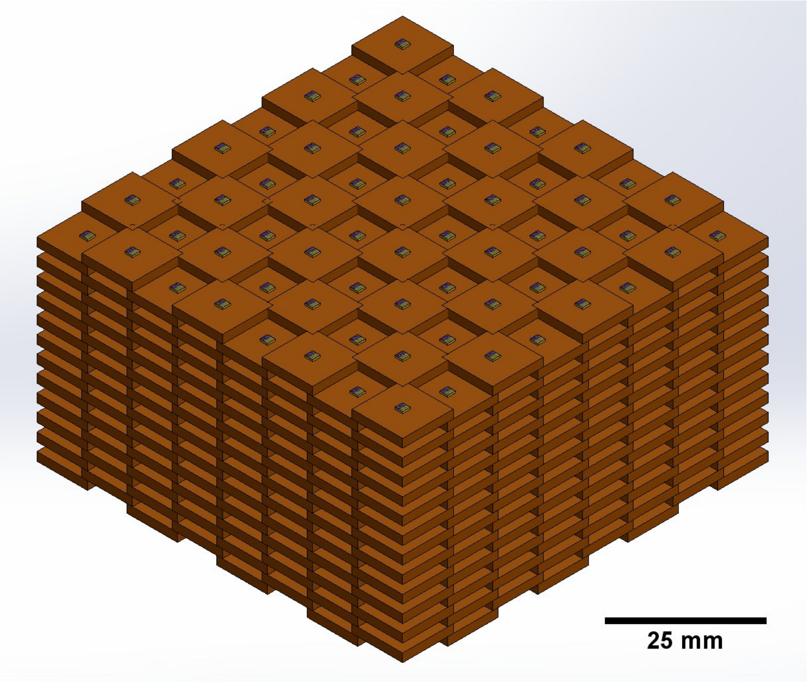


Figure 8-2: Render of 768 Micro-DICE nodes in a corner-connected cubic lattice, roughly equivalent in overall computational power to a single V100.

A rigorous examination of the economics of Micro-DICE, particularly as a V100 replacement, would consider many factors beyond the scope of this chapter. In particular, currently available process node capacity and design capabilities are clearly paramount; for example, given fab availability it may make sense to tape out a 45 nm chip. Furthermore, the chip designer's ability to integrate sufficient memory for DEM simulation may push the nodes to be larger or smaller in terms of transistor count and resulting GFlops. Of course, increasing the process node size affects current consumption and efficiency. But notably, all of these trade-offs must be considered in light of Micro-DICE's stark advantages: the decreased development cost associated with a 30 million versus a 21 billion transistor ASIC, and the increased yield from a 2 mm^2 versus an 800 mm^2 die.

8.3 Super-DICE

Perhaps the most exciting result of the DICE project is the emergence of Super-DICE, which my colleague Camron Blackburn covers extensively in her concurrent Master's thesis. Briefly, superconducting electronics can be up to 10^5 times more power efficient than CMOS devices when performing computational tasks, even including the roughly 10^3 watt/watt cooling overhead needed to keep cryostats below 4 K. However, the latest superconducting fabrication technology has only recently resulted in putting a million junctions (analogous to CMOS transistors) on a single chip [85], meaning their fabrication technology is more than two decades behind the current state-of-the-art on a transistor-count basis [72].

Merging superconducting technology with DICE provides a clear path towards real relevance for superconducting electronics, since the concept would allow massive numbers of discrete asynchronous superconducting chips to work together on computational tasks. Even better, the inherently 3D nature of DICE is physically beneficial, since cryostats are three-dimensional spaces and heat transfer occurs as a function of surface area. We plan to continue our collaboration with MIT Lincoln Labs to further explore this concept.

8.4 Flexural Mechanisms

The frustrating fatigue problems with modular superelastic flexures will be solved, given enough time and (likely) electrochemistry. As this proceeds, a reasonable parallel effort will be to continue scaling down the working range of the machines, perhaps adding a stage to the MicroPanto to address nanometer-scale parts. Rather than attempting to fabricate devices at this scale, a good early step will be to build an atomic force microscope (AFM), likely using commercially available cantilevers and a fabricated feedback circuit.

Bibliography

[1] ATSAMD51J20A | Microchip Technology. https://www.microchip.com/en-us/product/ATSAMD51J20A.

[2] Goodfellow USA. http://www.goodfellowusa.com/. Accessed 8/18/2021.

[3] IBM Power System AC922: Technical Overview and Introduction. https://www.ibm.com/products/power-systems-ac922.

[4] Kozak Micro M3-0.25 Microadjuster Sets. https://kozakmicro.com/collections/m3-0-25-sets. Accessed 6/2/2021.

[5] LAUREN FENSTERSTOCK. https://laurenfensterstock.com/home.html. Accessed 8/10/2021.

[6] The Message Passing Interface (MPI) standard. https://www.mcs.anl.gov/research/projects/mpi/. Accessed 8/4/2021.

[7] Molex SlimStack Board-to-Board/Board-to-FPC Connectors. https://www.molex.com/molex/products/family/slimstack_fine_pitch_smt_board_to_board_connectors. Accessed 8/19/2021.

[8] OpenCV. https://opencv.org/. Accessed 8/10/2021.

[9] ShopBot HandiBot. https://www.shopbottools.com/products/handibot. Accessed 8/10/2021.

[10] Summit User Guide - OLCF User Documentation. https://docs.olcf.ornl.gov/systems/summit_user_guide.html. Accessed 8/4/2021.

[11] Tesla V100 PCIe Product Brief. https://images.nvidia.com/content/tesla/pdf/Tesla-V100-PCIe-Product-Brief.pdf. Accessed 8/18/2021.

[12] Xilinx Intellectual Property. https://www.xilinx.com/products/intellectual-property.html. Accessed 8/18/2021.

[13] CUDA Toolkit. https://developer.nvidia.com/cuda-toolkit, July 2013.

[14] List of semiconductor scale examples. https://en.wikipedia.org/w/index.php?title=List_of_semiconductor July 2021. Accessed 7/28/2021.

[15] btarunr . Samsung 3 nm GAAFET Node Delayed to 2024. https://www.techpowerup.com/283983/samsung-3-nm-gaafet-node-delayed-to-2024. Publication Title: TechPowerUp.

[16] Andrew Adamatzky, R. Alonso-Sanz, and A. Lawniczak. *Automata-2008: Theory and Applications of Cellular Automata*. Luniver Press, 2008.

[17] Shorya Awtar. *Synthesis and Analysis of Parallel Kinematic XY Flexure Mechanisms*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2004.

[18] Shorya Awtar, Alexander H. Slocum, and Edip Sevincer. Characteristics of Beam-Based Flexure Modules. *Journal of Mechanical Design*, 129(6):625–639, May 2006.

[19] Vivek Bakshi. *EUV Sources for Lithography*. SPIE Press, 2006.

[20] Eberhard Bamberg. Viteris MW250 Micro-EDM. http://www.viteris.com/products/MW250.html. Accessed 6/2/2021.

[21] C. G. Bell, A. Kotok, T. N. Hastings, and R. Hill. The evolution of the DECsystem 10. *Communications of the ACM*, 21(1):44–63, January 1978.

[22] C.N. Berglund. A unified yield model incorporating both defect and parametric effects. *IEEE Transactions on Semiconductor Manufacturing*, 9(3):447–454, August 1996.

[23] Arani Bose, Marius Hartmann, Hans Henkes, Hon Man Liu, Michael M.H. Teng, Istvan Szikora, Ansgar Berlis, Jurgen Reul, Simon C.H. Yu, Michael Forsting, Matt Lui, Winston Lim, and Siu Po Sit. A Novel, Self-Expanding, Nitinol Stent in Medically Refractory Intracranial Atherosclerotic Stenoses. *Stroke*, 2007(38):1531–1537, 2007.

[24] Arthur Chuang. Water Management Innovation is Central to the Future of ICs, July 2021.

[25] William B Cross, Anthony H Kariotis, and Frederick J Stimler. *Nitinol characterization study*. NASA, Langley Research Center, 1969.

[26] Ian Cutress. Early TSMC 5nm Test Chip Yields 80%, HVM Coming in H1 2020. https://www.anandtech.com/show/15219/early-tsmc-5nm-test-chip-yields-80-hvm-coming-in-h1-2020.

[27] Andrea Dezsö. The home site of visual artist Andrea Dezsö. http://andreadezso.com. Accessed 8/11/2021.

[28] Samson Ellis, Yuan Gao, and Cindy Wang. TSMC Ready to Spend $20 Billion on its Most Advanced Chip Plant - Bloomberg. *Bloomberg*, October 2017.

[29] Sylvain Ferrero, Agnès Piednoir, and Claude R. Henry. Atomic Scale Imaging by UHV-AFM of Nanosized Gold Particles on Mica. *Nano Letters*, 1(5):227–230, May 2001. Publisher: American Chemical Society.

[30] The Raspberry Pi Foundation. Buy a Raspberry Pi High Quality Camera. https://www.raspberrypi.org/products/raspberry-pi-high-quality-camera/.

[31] Zach Fredin. Building machines with Zach. https://fab.cba.mit.edu/classes/865.21/people/zach/index.htm Accessed 6/10/2021.

[32] Zach Fredin. MicroPanto: a 50:1 flexural pantograph for micron-futzing with a ShopBot. https://gitlab.cba.mit.edu/zfredin/micropanto. Accessed 8/10/2021.

[33] Zach Fredin, Jiri Zemanek, Camron Blackburn, Erik Strand, Amira Abdel-Rahman, Premila Rowles, and Neil Gershenfeld. Discrete Integrated Circuit Electronics (DICE). In *2020 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–8, Waltham, MA, USA, September 2020. IEEE.

[34] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, June 2014.

[35] Neil Gershenfeld. CBA pi benchmark database. https://gitlab.cba.mit.edu/pub/pi. Accessed 8/20/2021.

[36] Neil Gershenfeld, David Dalrymple, Kailiang Chen, Ara Knaian, Forrest Green, Erik D. Demaine, Scott Greenwald, and Peter Schmidt-Nielsen. Reconfigurable asynchronous logic automata: (RALA). *ACM SIGPLAN Notices*, 45(1):1–6, January 2010.

[37] Neil A. Gershenfeld and Neil Gershenfeld. *The Nature of Mathematical Modeling*. Cambridge University Press, 1999. Google-Books-ID: zYAcGbp17nYC.

[38] Amanda Paige Ghassaei. Rapid Design and Simulation of Functional Digital Materials. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2016.

[39] Martin Goosey. Water use in the printed circuit board manufacturing process and approaches for reducing consumption. *Circuit World*, 31(2):22–25, January 2005. Publisher: Emerald Group Publishing Limited.

[40] C. Gosselin and J. Angeles. The Optimum Kinematic Design of a Planar Three-Degree-of-Freedom Parallel Manipulator. *Journal of Mechanisms, Transmissions, and Automation in Design*, 110(1):35–41, March 1988.

[41] Gaile Guevara. Traditional Japanese joinery techniques. https://www.flickr.com/photos/gaileguevara/17173017438. Accessed 8/1/2021.

[42] Jonathan D. Hiller, Joseph Miller, and Hod Lipson. Microbricks for Three-Dimensional Reconfigurable Modular Microsystems. *Journal of Microelectromechanical Systems*, 20(5):1089–1097, October 2011. Conference Name: Journal of Microelectromechanical Systems.

[43] Intel. Intel Reports Second-Quarter 2021 Financial Results. Technical report.

[44] Michael James, Marvin Tom, Patrick Groeneveld, and Vladimir Kibardin. ISPD 2020 Physical Mapping of Neural Networks on a Wafer-Scale Deep Learning Accelerator. In *Proceedings of the 2020 International Symposium on Physical Design*, ISPD '20, pages 145–149, New York, NY, USA, March 2020. Association for Computing Machinery.

[45] W. James. Characteristics of Modular Electronics Components. *IRE Transactions on Component Parts*, 3(2):69–72, September 1956. Conference Name: IRE Transactions on Component Parts.

[46] Ronald K. Jurgen. Whatever happened to project tinkertoy? *IEEE Spectrum*, 24(5):20–21, May 1987. Conference Name: IEEE Spectrum.

[47] Gordon Keeler. Common Heterogeneous Integration and IP Reuse Strategies. https://www.darpa.mil/program/common-heterogeneous-integration-and-ip-reuse-strategies.

[48] V Lalitha and S Kathiravan. A Review of Manchester, Miller, and FM0 Encoding Techniques. *The Smart Computing Review*, 4(6), December 2014.

[49] William Kai Langford. Electronic Digital Materials. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2014.

[50] William Kai Langford. *Discrete Robotic Construction.* PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2019.

[51] Nicolae Lobontiu, Jeffrey S. N Paine, Edward O'Malley, and Marc Samuelson. Parabolic and hyperbolic flexure hinges: flexibility, motion precision and stress characterization based on compliance closed-form equations. *Precision Engineering*, 26(2):183–192, April 2002.

[52] Hamilton H. Mabie and Charles F. Reinholtz. *Mechanisms and Dynamics of Machinery.* John Wiley & Sons, January 1987.

[53] Robert MacCurdy, Anthony McNicoll, and Hod Lipson. Bitblox: Printable digital materials for electromechanical machines. *The International Journal of Robotics Research*, 33(10), 2014.

[54] Chris A. Mack. Fifty Years of Moore's Law. *IEEE Transactions on Semiconductor Manufacturing*, 24(2):202–207, May 2011.

[55] P. Y. Martinez, Y. Beilliard, M. Godard, D. Danovitch, D. Drouin, J. Charbonnier, P. Coudrain, A. Garnier, D. Lattard, P. Vivet, S. Cheramy, E. Guthmuller, C. Fuguet Tortolero, V. Mengue, J. Durupt, A. Philippe, and D. Dutoit. ExaNoDe: Combined Integration of Chiplets on Active Interposer with Bare Dice in a Multi-Chip-Module for Heterogeneous and Scalable High Performance Compute Nodes. In *2020 IEEE Symposium on VLSI Technology*, pages 1–2, Honolulu, HI, 2020. IEEE. ISSN: 2158-9682.

[56] Morgan McCorkle. ORNL Launches Summit Supercomputer | ORNL. https://www.ornl.gov/news/ornl-launches-summit-supercomputer. Accessed 8/18/2021.

[57] Wei-dong Miao, Xu-jun Mi, Xin-lu Wang, and Hua-chu Li. Electropolishing parameters of NiTi alloy. *Transactions of Nonferrous Metals Society of China*, 16:s130–s132, June 2006.

[58] Gordon E Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8), April 1965.

[59] Seth Morabito. Messy. https://www.flickr.com/photos/twylo/2556770440/, June 2008. Accessed 8/18/2021.

[60] B.T. Murphy. Cost-size optima of monolithic integrated circuits. *Proceedings of the IEEE*, 52(12):1537–1545, December 1964.

[61] D. Nicoară and I. Nicoară. An improved Bridgman-Stockbarger crystal-growth system. *Materials Science and Engineering: A*, 102(2):L1–L4, July 1988.

[62] Nicoguaro. Stress strain ductile curve. https://commons.wikimedia.org/w/index.php?curid=89891144, May 2020. Accessed 8/20/2021.

[63] Nilay Patel. Intel announces 22nm chips for 2011. https://www.engadget.com/2009-09-22-intel-announces-22nm-chips-for-2011.html, September 2009.

[64] Prashant Patil. *Laser Direct-Write Fabrication of MEMS*. PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2019.

[65] Andrew Petto, Zachary Fredin, and Joseph Burdo. The Use of Modular, Electronic Neuron Simulators for Neural Circuit Construction Produces Learning Gains in an Undergraduate Anatomy and Physiology Course. *Journal of Undergraduate Neuroscience Education*, 15(2):A151–A156, June 2017.

[66] Polyparadig. Empty Tic Tacs box. https://commons.wikimedia.org/wiki/File:Mint_box_polypropylene_l
Accessed 5/2/2021.

[67] George A Popescu. Digital Materials for Digital Fabrication. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2007.

[68] Jon Porter. Apple says new Arm-based M1 chip offers the 'longest battery life ever in a Mac'. https://www.theverge.com/2020/11/10/21558095/apple-silicon-m1-chip-arm-macs-soc-charge-power-efficiency-mobile-processor, November 2020.

[69] Hui Qian, Hongnan Li, Gangbing Song, and Wei Guo. Recentering Shape Memory Alloy Passive Damper for Structural Vibration Control. *Mathematical Problems in Engineering*, 2013:e963530, November 2013. Publisher: Hindawi.

[70] Jake Read. clank-lz. https://gitlab.cba.mit.edu/jakeread/clank-lz. Accessed 5/2/2021.

[71] Jake Robert Read. Distributed Dataflow Machine Controllers. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2020.

[72] Max Ritchie and Hannah Roser. A logarithmic graph showing the timeline of how transistor counts in mircochips are almost doubling every two years from 1970 to 2020; Moore's Law. https://commons.wikimedia.org/wiki/File:Moore%27s_Law_Transistor_Count_1970-2020.png, November 2020. Accessed 7/28/2021.

[73] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[74] Otto Schmitt. *An electrical theory of nerve impulse propagation.* PhD Thesis, Washington University, 1937.

[75] E. Schöller, L. Krone, M. Bram, Hans Buchkremer, and D. Ståaver. Metal injection molding of shape memory alloys using prealloyed NiTi powders. *Journal of Materials Science*, 40:4231–4238, August 2005.

[76] Anand Lal Shimpi. AMD's Fab 36 Grand Opening - 90nm and 300mm in Germany. https://www.anandtech.com/show/1821. Accessed 7/28/2021.

[77] Paul Stoffregen. Teensy® 4.0 Technical Information. https://www.pjrc.com/store/teensy40.html#tech. Accessed 7/31/2021.

[78] Erik Steven Strand. Inverse Methods for Design and Simulation with Particle Systems. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2020.

[79] Masahiro Sunohara, Takayuki Tokunaga, Takashi Kurihara, and Mitsutoshi Higashi. Silicon interposer with TSVs (Through Silicon Vias) and fine multilayer wiring. In *2008 58th Electronic Components and Technology Conference*, pages 847–852, May 2008. ISSN: 2377-5726.

[80] Larry W. Swanson, Eric Newman, Alfonso Araque, and Janet M. Dubinsky. *The Beautiful Brain: The Drawings of Santiago Ramón y Cajal.* Abrams, 2017.

[81] Alex Taradov. edbg: Simple utility for programming Atmel MCUs though CMSIS-DAP protocol. https://github.com/ataradov/edbg. Accessed 5/2/2021.

[82] Alex Taradov. Free-DAP. https://github.com/ataradov/free-dap, August 2021. Accessed 8/19/2021.

[83] Denis Terwagne. Urumbu flexible X/Y stage. https://gitlab.fabcloud.org/academany/fabacademy/2020/bootcamp/spicy/-/tree/master/Urumbu/flexible-XYstage. Accessed 7/1/2021.

[84] Thomas Thwaites. *The Toaster Project: Or a Heroic Attempt to Build a Simple Electric Appliance from Scratch.* Chronicle Books, September 2011.

[85] Sergey K. Tolpygo, Vladimir Bolkhovsky, Daniel E. Oates, Ravi Rastogi, Scott Zarr, Alexandra L. Day, Tarence J. Weir, Alex Wynn, and Leonard M. Johnson. Superconductor Electronics Fabrication Process with MoNx Kinetic Inductors and Self-Shunted Josephson Junctions. *IEEE Transactions on Applied Superconductivity*, 28(4):1–12, June 2018. Conference Name: IEEE Transactions on Applied Superconductivity.

[86] Reinhard Uecker. The historical development of the Czochralski method | Elsevier Enhanced Reader. *Journal of Crystal Growth*, 401(1):7–24, September 2014.

[87] Pascal Vivet, Eric Guthmuller, Yvain Thonnart, Gael Pillonnet, Guillaume Moritz, Ivan Miro-Panadès, Cesar Fuguet, Jean Durupt, Christian Bernard, Didier Varreau, Julian Pontes, Sebastien Thuries, David Coriat, Michel Harrand, Denis Dutoit, Didier Lattard, Lucile Arnaud, Jean Charbonnier, Perceval Coudrain, Arnaud Garnier, Frederic Berger, Alain Gueugnot, Alain Greiner, Quentin Meunier, Alexis Farcy, Alexandre Arriordaz, Severine Cheramy, and Fabien Clermidy. A 220GOPS 96-Core Processor with 6 Chiplets 3D-Stacked on an Active Interposer Offering 0.6ns/mm Latency, 3Tb/s/mm2 Inter-Chiplet Interconnects and 156mW/mm2@ 82%-Peak-Efficiency DC-DC Converters. In *2020 IEEE International Solid- State Circuits Conference - (ISSCC)*, pages 46–48, San Francisco, CA, 2020. IEEE. ISSN: 2376-8606.

[88] Jonathan Ward. Additive Assembly of Digital Materials. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2010.

[89] John F. Wendt, John D. Anderson, and Von Karman Institute for Fluid Dynamics, editors. *Computational fluid dynamics: an introduction.* Springer, Berlin ; [London], 3rd ed edition, 2008. OCLC: ocn288984495.

[90] Robert L. Williams and Brett H. Shelley. Inverse Kinematics for Planar Parallel Manipulators. American Society of Mechanical Engineers Digital Collection, February 2021.

[91] Claire Wolf and Mathias Lasser. Project IceStorm. http://bygone.clairexen.net/icestorm/. Accessed 8/19/2021.

[92] Miao Yang, Chi Zhang, Guilin Yang, and Wei Dong. Optimal Design and Tracking Control of a Superelastic Flexure Hinge Based 3-PRR Compliant Parallel Manipulator. *IEEE Access*, 7:174236–174247, 2019. Conference Name: IEEE Access.

[93] Geoffrey Yeap, S. S. Lin, Y. M. Chen, H. L. Shang, P. W. Wang, H. C. Lin, Y. C. Peng, J. Y. Sheu, M. Wang, X. Chen, B. R. Yang, C. P. Lin, F. C. Yang, Y. K. Leung, D. W. Lin, C. P. Chen, K. F. Yu, D. H. Chen, C. Y. Chang, H. K. Chen, P. Hung, C. S. Hou, Y. K. Cheng, J. Chang, L. Yuan, C. K. Lin, C. C. Chen, Y. C. Yeo, M. H. Tsai, H. T. Lin, C. O. Chui, K. B. Huang, W. Chang, H. J. Lin, K. W. Chen, R. Chen, S. H. Sun, Q. Fu, H. T. Yang, H. T. Chiang, C. C. Yeh, T. L. Lee, C. H. Wang, S. L. Shue, C. W. Wu, R. Lu, W. R. Lin, J. Wu, F. Lai, Y. H. Wu, B. Z. Tien, Y. C. Huang, L. C. Lu, Jun He, Y. Ku, J. Lin, M. Cao, T. S. Chang, and S. M. Jang. 5nm CMOS Production Technology Platform featuring full-fledged EUV, and High Mobility Channel FinFETs with densest 0.021µm2 SRAM cells for Mobile SoC and High Performance Computing Applications. In *2019 IEEE International Electron Devices Meeting (IEDM)*, pages 36.7.1–36.7.4, December 2019.

[94] Hao Zhang and Haihang You. Comprehensive Workload Analysis and Modeling of a Petascale Supercomputer. In *Workshop on Job Scheduling Strategies for Parallel Processing*, pages 253–271, Berlin, May 2012. Springer.