# ChainFORM: A Linear Integrated Modular Hardware System for Shape Changing Interfaces

**Ken Nakagaki**[1], **Artem Dementyev**[1], **Sean Follmer**[2], **Joseph A. Paradiso**[1], **Hiroshi Ishii**[1]

[1]MIT Media Lab
Cambridge, MA
{ken_n, artemd, joep, ishii}@media.mit.edu

[2]Stanford University
Stanford, CA
sfollmer@stanford.edu

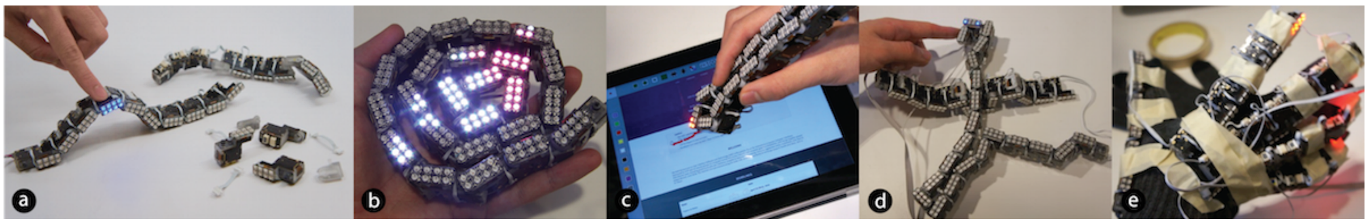


**Figure 1. Device and application examples of the ChainFORM hardware system. (a: ChainFORM hardware configurations and modules, b: reconfigurable display, c: shape changing stylus, d: animated character, e: haptic glove.)**

### ABSTRACT

This paper presents ChainFORM: a linear, modular, actuated hardware system as a novel type of shape changing interface. Using rich sensing and actuation capability, this modular hardware system allows users to construct and customize a wide range of interactive applications. Inspired by modular and serpentine robotics, our prototype comprises identical modules that connect in a chain. Modules are equipped with rich input and output capability: touch detection on multiple surfaces, angular detection, visual output, and motor actuation. Each module includes a servo motor wrapped with a flexible circuit board with an embedded microcontroller.

Leveraging the modular functionality, we introduce novel interaction capability with shape changing interfaces, such as rearranging the shape/configuration and attaching to passive objects and bodies. To demonstrate the capability and interaction design space of ChainFORM, we implemented a variety of applications for both computer interfaces and hands-on prototyping tools.

### Author Keywords

Actuated Curve Interfaces; Shape Changing Interfaces; Modular Robotics

### ACM Classification Keywords

H.5.2. User Interfaces: Input Devices and Strategies, Haptic I/O.


*UIST '16*, October 16 - 19, 2016, Tokyo, Japan.

ACM 978-1-4503-4189-9/16/10$15.00.
DOI: http://dx.doi.org/10.1145/2984511.2984587

### INTRODUCTION

As shape changing interfaces being an emerging field in HCI, a lot of actuation techniques have been introduced to provide physical shapes to represent digital data and to embody spatial interactions [9, 4]. Researchers are continually seeking techniques that have a variety of transformational capabilities in different geometries and scales [14, 24]. To extend the sensing and display capability of such shape changing interfaces, extra sensors or cameras and projectors have been installed for detecting human input and displaying information on the active surfaces. However, this strategy poses a challenge for scaling the system, which presents a problem, especially for mobile applications. To push the boundaries of shape-changing interface research, another approach calls for self-contained systems that integrate sensing, actuation and display across different scales, geometries, and transformations.

We present ChainFORM: a modular integrated hardware system that has a chained, linear form factor (Figure 1). The hardware comprises identical actuated modules connected in series, which allows the user to customize the length and the configuration of devices they construct. The form-factor of line and the modularity expands the possibility of transformation for both shapes and scales. In addition, each module integrates sensing, actuation, and display, enabling a wide array of applications and interactions to be developed with a uniform, easily scalable hardware infrastructure. Our approach is a step toward a general platform for custom shape-changing interfaces. Building on the idea and implementation of modular and serpentine robotics[17, 33, 35, 34, 30], we intend to extend their knowledge and technique to enrich interactions with shape-changing interfaces.

In this paper, we describe the technical implementation of our prototype and present a wide range of applications to demonstrate the modularity and rich input and output capability of our hardware design. We conducted technical evaluations and discuss the capabilities and limitations of our current prototype, with points for future improvement.

The contributions of the paper are as follows:

1. We developed a modular shape changing interface system which has a linear configuration. The module connect together to form an arbitrary linear shape using mechanical joints and electrical communication architecture.
2. We designed each module of the hardware system to have self-contained sensing, actuation and display system together for rich interaction capability.
3. We performed technical evaluations of the system.
4. We implemented a variety of application scenarios for shape changing computer interfaces and actuated prototyping tools.

## RELATED WORK

### Modular and Serpentine Robots

Many modular robotic systems and concepts have been proposed in the robotics field [17, 33, 35]. Our design is similar to that of modular snake robots which use identical modules based on motors with linear communication architecture [34, 30]. Although the focus of such systems is mainly on locomotion, we focus on how the hardware can interact with users to represent information and detect human input. Most robotics systems use modular systems because of robustness and maintenance-ease purposes, in contrast we take this capability for users to customize their own interfaces in different scales and configurations.

### Robot Prototyping System

There are commercial products available for users to construct robots with customizable joints, blocks and brackets [5, 19]. Dynamixel servomotor system enables users to link the servomotor modules together with a dedicated bracket system [7]. They have daisy chain communication architecture where every motor can be controlled by a shared communication bus. ChainFORM has a similar communication system, but our hardware system can detect tangible interactions and display visual feedback with LEDs specialized for developing an interactive system. Also, our module is relatively small and light weight that users can construct hand held sized interfaces and easily attach to other materials. Furthermore, Dynamixel system requires manual address assignment for each module, in contrast our system automatically assigns the address to each module, which allows for quicker prototyping.

### Modular Interfaces

In the field of HCI, modular hardware systems have been explored both for sensing and actuation to enable users to customize their own interfaces. As for input systems, Jacobson et al. presented a modular sensing device for CAD tools that can be customized to adapt to various character models to tangibly design postures and motions [15]. SensorTape has the form factor of tape that user can cut and arrange the length and shape of the sensor array using flexible circuit boards with a chained communication system design [6]. Another work demonstrates block-like interface, that automatically detected changes in shape. [32]. Phigets[11] and LittleBits[2] are toolkit which enable users to create interactive objects with various input and output modules.

Various concepts and methods of shape changing interfaces have been presented recently in the field of HCI to render physical shape of digital data and provide dynamic affordance for physical interactions [4, 24, 9]. Because the scalability of these systems is often limited to the size of the hardware designs, there have been some modular systems proposed to create custom actuated interfaces. Topobo and Bosu are toolkits for designers and kids to create kinetic crafts and motion with tangible interactions[23, 22]. ShapeClip is a prototyping tool that is composed of linear actuated modules for constructing customized shape displays [13].

### Line-based Interfaces

Within the HCI field, there have also been proposed various types of tangible interfaces which have form factor of lines. ShapeTape was introduced as a passive 3D modeling tool that leveraging the flexibility and affordance of a spline[12]. FlexiBend is a tape-like input interface that can be attached to deformable materials for fabricating multi-input interfaces [3]. Katsumoto et al. presented a bi-stable geometries for a controller composed with chained mechanical hinges and proposed applications that provide different digital functions according to shapes [16].

As a novel form of shape changing interfaces, LineFORM introduced the concept of actuated curve interface leveraging the dynamic transformation capability and tangible interaction of lines [20]. Although their implementation was not able for users to change the length of the hardware, in our system, in addition to modular design, we intend to enrich the concept and interaction for an actuated curved interface with rich sensing capabilities and display technologies.

## CHAINFORM

### Function and Interaction Overview

ChainFORM is a modular hardware system which is composed of identical modules. Each module has rich sensing and actuation capability (Figure 2).

The module can detect tangible interactions by users. Specifically, the angular deformation and the way users touch the modules can be detected. The ChainFORM module can provide several outputs. One is motor actuation, it can change the angle of the joint and also torque can be controlled to lock the joint or loose so that user can deform by their hands and also feel different stiffness as tactile feedback. The color of the surface of each module can be displayed with LEDs to provide visual feedback.

The core feature of ChainFORM is its modularity. This feature enables users to customize the length and configuration of the chained hardware. We were strongly inspired by linear craft materials such as strings, wires or tapes that can be cut to separate and knotted to connect, and our hardware design enables the user to customize the interface similar to these

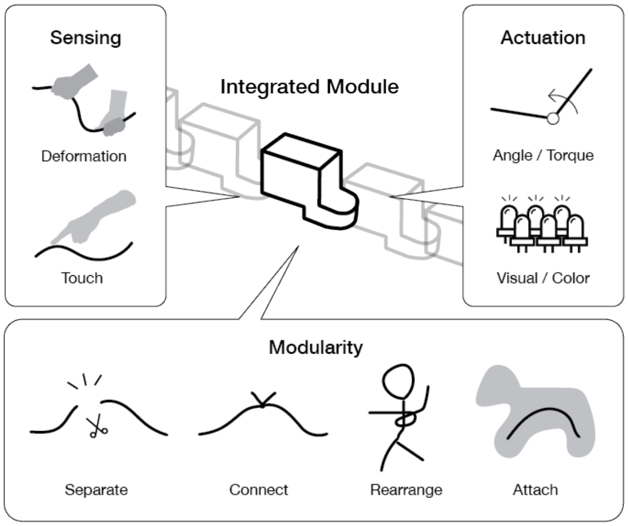


**Figure 2. The integrated module with three functionalities; Sensing, Actuation and Modularity.**

materials. Leveraging the capability, users can rearrange the configuration of the device to create desired shape and transformation. Because of its relatively small size, it can also be attached to passive objects or bodies to detect input and actuate accordingly.

## System Design

The overall system is composed of a chain of integrated modules, a master board and a software on a computer. The master module contains a Teensy 3.2 with an ARM Cortex-M4 (MK20DX256VLH7, Freescale). The master coordinates the communications between the modules and shuttles data to the computer over USB. The software was developed using Processing on OSX computer.

### *Module Design*

The module of our system is composed of three components as shown in Figure 3; a circuit board, a 3D printed bracket and a servo motor. The motor is placed inside the 3D printed bracket and the bracket provides a mechanical connection for joining to another module. Some parts of the circuit board are designed to have flexible hinges (Figure 4a), so that it can be wrapped around the 3D printed bracket which incorporates the motor. Using the joint of this 3D printed bracket, only 2D planar transformation can be created. Therefore we also developed a joint that translate this configuration into 3D to expand the transformation capability (Figure 5). For the servo motor, we used HS-5035HD (Hitec) which is the smallest servo motor we could find on the market.

The circuit board contains ATmega328p (Atmel) microcontroller, which is the “brain” of the module. MTCH6102 (Microchip) was used to perform capacitive sensing. A small solid-state relay, G3VM-21UR11 (OMRON) was used to turn on/off the torque of servo motor. An array of 8 Neopixels Mini (Adafruit) was used for the LED display covering a single surface of a module. The circuit board is designed to integrate six capacitive sensing surfaces, as shown in Figure 4B,

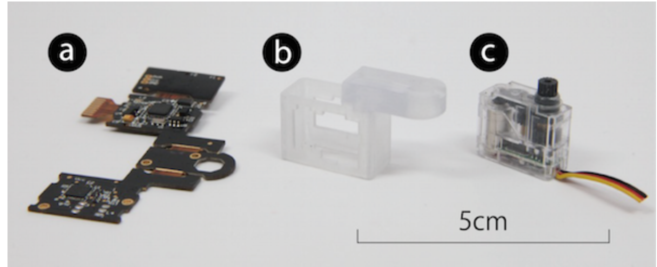


**Figure 3. The components of each module (a: circuit board with flexible hinges, b: 3D printed bracket, c: HS-5035HD Servo Motor.**

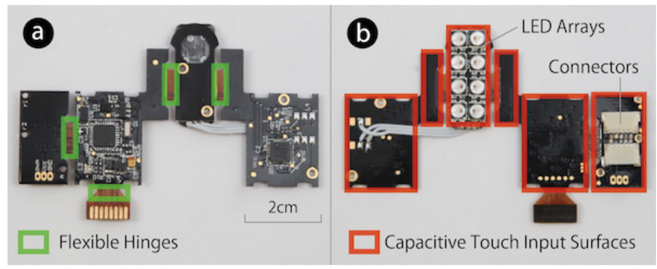


**Figure 4. Design of the circuit board (A: inner side, B: external side).**

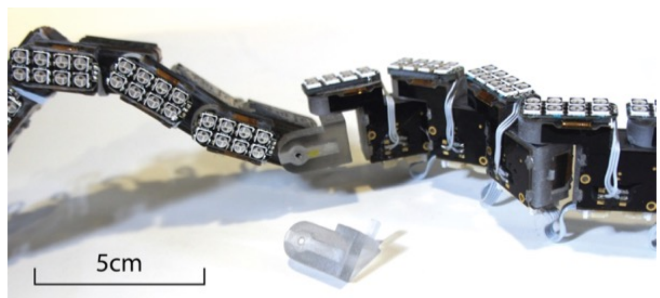


**Figure 5. A joint that translates the motion into 3D configuration.**

so that every surface of the module is capable of touch sensing.

Figure 6 shows the system configuration of each module. We modified each servo motor to read the input of internal potentiometer values and to control the connection of the wire to the motor so that the motor axis can either be flexible for manual control or actuated(stiff) for transforming and locking shape (Figure 6). Two connectors are used to attach the modules in series using a 5-wire cable for communication and power.

### *Chained Communication Architecture*

We wanted the users to have an ability to quickly add and disconnect modules, without the need to reprogram the network. To do so we developed a communication architecture that automatically determines how many modules there are and how they are connected. Furthermore, we wanted to have real-time I/O, even with a large number of modules. There are no ready-made communications protocols that do so. We created a custom network based on the I2C protocol and inspired by the SensorTape Project [6]. We believe that the developed

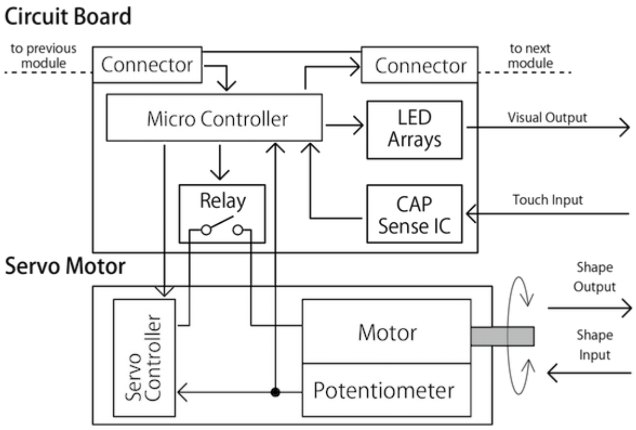


**Figure 6. System configuration of each module.**

network architecture can be applied to other types of modular systems, therefore we explain it in detail.

The network has two parts; the serial peer-to-peer connections between the modules and the global Inter-Integrated Circuit (I2C) bus. The peer-to-peer network only ran once after start-up. It was used to assign a unique address to each module based on how far they are from the master. Peer-to-peer scheme functions similarly to a shift register. The master sends a "0" to the first module, which assigns its address to "0". The first module sends a "1" to the next module. This cascade repeats until the last module is reached. At the end, each module has a unique address and is individually addressable by the I2C bus.

During normal operation, the I2C master polls the modules sequentially, and immediately sends the data to the PC over 12 Mbit/sec USB connection. After polling all the modules, the master checks if there is any data from the PC (e.g, motor or LED commands). If data is available, it is instantly transmitted to the addressed module. Then, the polling resumes.

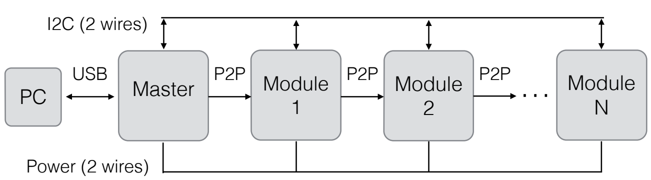


**Figure 7. Network architecture of ChainFORM. All the modules and the master share the same power, ground and I2C bus. Also, each module is connected to it's neighbor for peer-to-peer communications. In total, there are 5 connections between all the modules.**

### *Software Control*

Our Processing-based software is designed to communicate with the chained hardware system through Teensy. Every input data including potentiometer and touch sensing data are received every frame and output data can be sent once every frame (60 Hz framerate). Our GUI system enables users to control angles and torque of each motor and visualize raw input data. Visualization can be switched between 2D and 3D. We developed some mid-level software application for specific function and interactions described below.

*Color Mapping Function for LEDs* - Mainly for reconfigurable display applications, we developed a function that the color of each LED can be defined through default sketch function on Processing. As the software receives potentiometer values, it can estimate the whole shape of the device and location of each LED. Using this function on the software, users only need to write codes on visualization such as shapes, texts or loaded images which are default functions for Processing. Then, according to the visual sketch and location of each LED, the software automatically allocates a color for each LED 8. In our implementation, we built this application only for 2D transformation, but we believe this can be applied for 3D transformation with extra joints and 3D sketching functions.

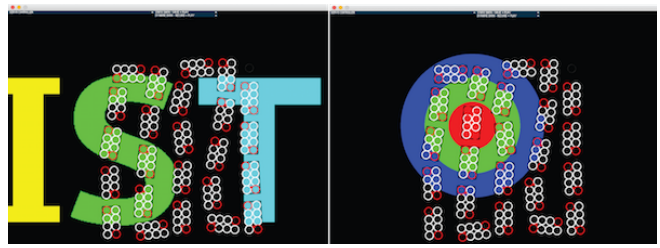

**Figure 8. Software on Processing to allocate LED color according to the visual sketch on screen (left: texts, right: shapes).**

*Record and Replay with Hands* - We developed a function for shape and motion design. For chain-based hardware design, it is complicated to generate target shape and motion by coding because users need to define an angle for each module to get the whole shape. We developed manual record and replay function to overcome this problem. Similar to Topobo [23], the shape and motion can be recorded based on the potentiometer values and replayed back anytime. Non-programmers can easily design motion without programming, and programmers can develop interaction system to replay these shapes and motions according to specific input data from sensors.

## APPLICATIONS

Here, we present applications that demonstrate the capability of proposed system. We mainly proposed two categories of applications. The first category is computer interfaces that can adapt to user's needs and digital functions with transforming and modular capability. This category includes three applications; adaptive input interface, shape changing modular display, and shape changing stylus. The second category is prototyping tools that leverage the customizability of our hardware system to make creation process of actuated interactive system easier. We propose two application under this category; animated craft and body augmentation tool.

### Computer Interfaces

#### *Adaptive Input Interface*

This application presents input interfaces which can transform into various form factor that can adapt to a different function on computers and provide dynamic physical affordances[9] to users. Specifically, this system could be used to applications which require various input. For example, for

CAD software, the chained interface can let users manipulate vector data tangibly as a line and form a touch surface as a color picker(Figure 9a, b). The modularity of the system enables users to have multiple controls on demand such as rotation sensors or sliders for CAD as shown in 9c.

By turning off the motor torque on specific joints, it can produce mechanical hinge input interface. Figure9d, e shows examples of a gun controller that can detect when the trigger is depressed, and 2 DoF joystick that is be constructed using joints for 3D transformation. This input could be used for gaming applications that require different controller for different modes. This sensing interface can be combined with motor actuation to provide haptic feedback to users.

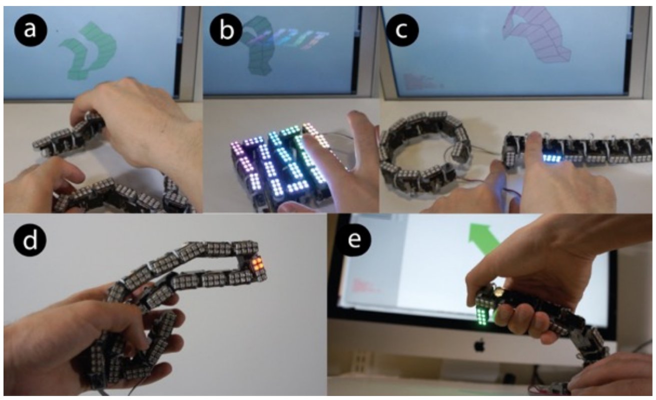

**Figure 9. Examples of dynamic and modular input interfaces (a: tangible curve manipulation, b: touch surface for color picker, c: rotary sensor and slider for rotation and scaling created from single line, d: gun controller with mechanical toggle input, e: 2 DoF joystick interface.)**

### *Shape Changing Modular Display*

Utilizing the LED arrays on each module, ChainFORM can construct displays in various shapes. Although a lot of systems for shape changing flexible displays have been proposed, they mostly consist of rectangular surfaces that can create slight curves [10, 26]. In contrast, our chained hardware system has more dynamic transformation capability to create shapes either in 2D or 3D. Using our display technology, we can imagine a future smartphone that can change shape from rectangular shapes to present texts or pass cord lock information, to circle shapes for navigating users as a compass (Figure 10a,b,c). Similar to smart watches that can change the visual skin, our system can inform time in various shapes and appearance according to users' preferences (Figure 10d). Utilizing the form factor of a line, this display can wrap around objects to change any surface into displays. Especially when attaching on our body, it can be wearable displays for fashions (Figure 10e).

The modular functionality of ChainFORM enables users to customize the size of the display as well. Figure 11 shows the example scenario utilizing the modularity. When multiple users have personal mobile displays composed with ChainFORM(a), they can combine their displays(b) to create a large display that can provide multi-user contents such as movies or gaming(c). Conversely, a large screen can be split into small personal displays for multiple users. Comparing to

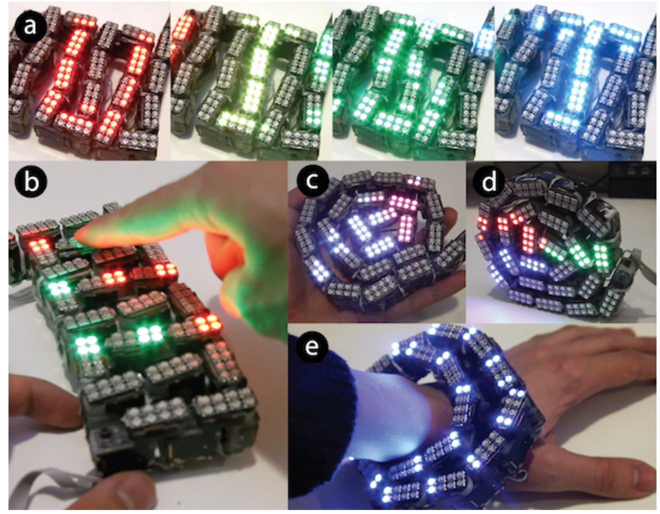

**Figure 10. Examples of reconfigurable displays (a: square for showing texts "U, I, S, T", b: rectangle for pass code, c: circle for compass, d:circle for analog clock, e: wrapping around a wrist as fashionable wearable device)**

other modular display systems [1, 18, 25], our system can physically actuate and it is scalable from few centimeter small unit to large displays together with rich tangible sensing capability. Our approach of display with a chain of LED arrays has more freedom and capability of transformation than other modular display techniques.

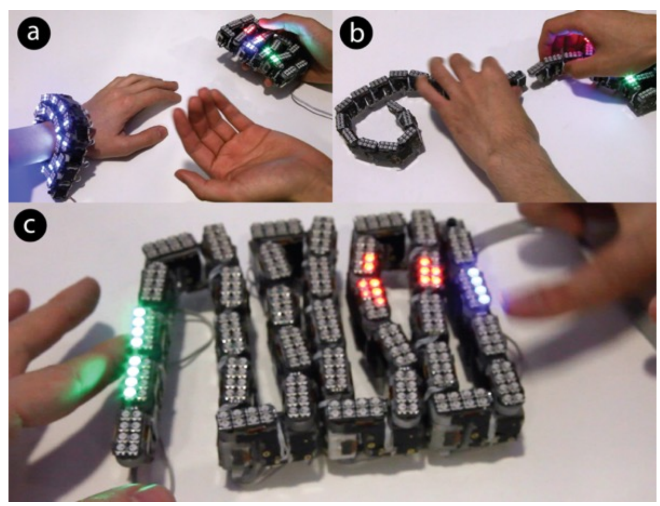

**Figure 11. User scenario with modular display for multi-user interaction (a: two separate ChainFORM displays for personal use, b: combining two displays, c: connected into a single chain, the device reconfigure to one large display for multi-user applications, in this case, the Pong game.)**

### *Shape Changing Stylus*

The relatively small size of the modular system allows building of transforming hand-held tools. Using our prototype, we developed a stylus device which transforms according to the function in digital CAD software. Similar to the multi-touch pen system presented in [28], the CAD function can be defined by the way users hold the grip utilizing the multiple touch sensing surfaces (Figure 12a). Our proposed stylus interface transforms the physical shape of tool tip instantly once users change gripping. The tip becomes pen for thin stroke, brush for wide stroke and magnifying lens for zooming (Figure 12b, c, d). LED can help users visualize what kind of color they are picking now.

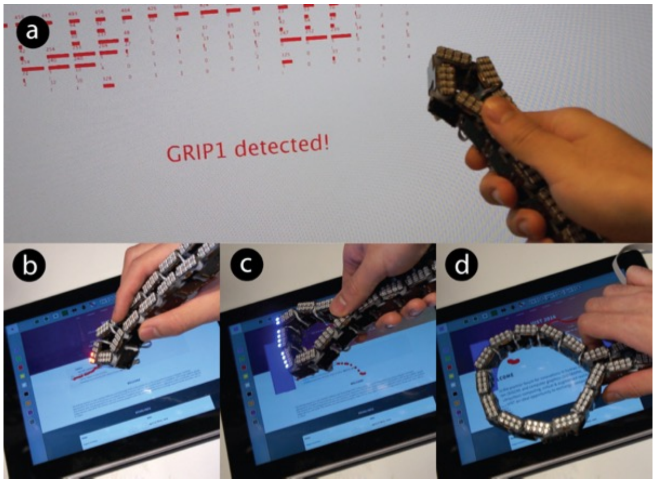


Figure 12. Stylus interface that can transform according to the grip (a: detecting grip with multiple touch sensing feedback, b: pen mode for thin stroke, c: brush mode for wide stroke, d: magnifying glass for zooming).

### Prototyping Tool for Actuation and Interaction

*Animated Craft*

As a prototyping toolkit, modularity and rich interactivity of the ChainFORM system allow crafting animated objects. Children can learn to build an active mechanical structure through prototyping. Artists may quickly prototype storytelling medium. Just like using traditional linear craft material such as wires or tapes, the user can deform, cut, connect and attach to other materials to construct their own shapes and motions. For example, as Figure 13a shows, the user can make a stick figure character that reacts to touch by changing his facial expression with LEDs and by moving his body.

Users can attach the hardware with customized length and configuration to paper crafts or hand puppets as an actuated skeleton to create expressive motion (Figure 13 b, d). The user can use LEDs for additive expressions, such as facial emotion as Figure 13c shows. Comparing to previous works of attachable actuators for animated crafts [21, 29], our hardware design has capabilities for a wider variety of motion, sensing tangible interactions, and visual representation. It is enabled by a high density of actuators, sensors, and LEDs. Moreover, the actuation enables materials not only for expression but also for actual kinetic locomotion. Figure 13e shows an example of turning a plain box into a walking robot, using ChainFORM as legs.

To design the motion of the chain, users can either write a code on Processing or, to make it more accessible to non-programmers, tangibly program the motion using record and replay function, similar to Topobo[23]. Not only dynamic motion but also passive stiffness can be defined by users to generate joints and hinges just like character rigging in Computer Graphics.

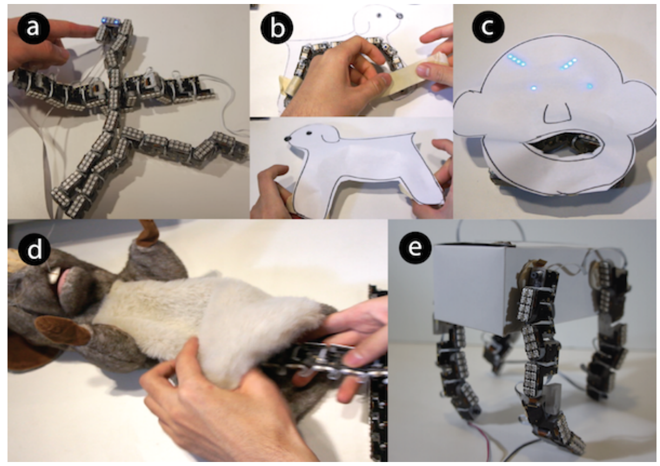


Figure 13. Examples of animated craft (a: interactive stick man, b: attaching to paper craft and recording motion by hand, c: facial expression represented with transformation and LEDs, d: installing in hand puppet, e: attaching to a plain box as a leg for locomotion).

*Body Augmentation Tool*

Leveraging the customizing and attaching capability, the ChainFORM system enables users to construct customized body augmentation devices, as our bodies have different size and shapes. Figure 14a shows an example of attaching to gloves to augment our fingers, that can be used either for a haptic glove in VR contents or for rehabilitation tool to help grasping objects for elders. Figure 14b shows attaching a long piece of ChainFORM to the back of an ordinary shirt as an external smart spine that can sense posture and correct by actuating. Our modular system would make personal fabrication and prototyping of body augmentation tool easy and accessible for users.

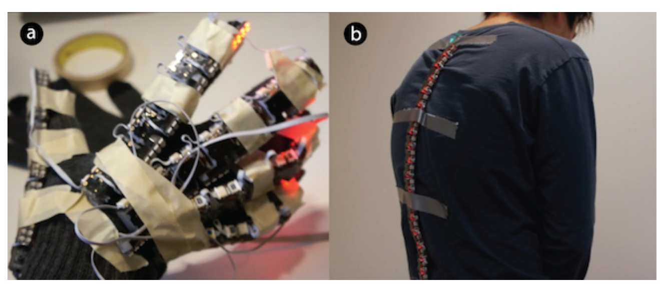


Figure 14. Examples of body augmentation using ChainFORM (a: haptic glove, b: external smart spine for posture correction.)

## EVALUATION

### Technical Evaluation

**Speed:** To have seamless interactions, it is important to have high update rate, even if there are many ChainFORM modules. The download data rate from the computer to the modules is 170 kbits/sec. The upload data rate from the modules to the computer is also 170 kbits/sec. Download and upload rates share the same I2C bus, running at 400kHz. Theoretically, the data rate can be up to 400 kbits/sec. Due to overhead, we only utilized 43% of maximum capacity of the bus, even with much timing optimization.

The I2C bus is the bottleneck of the current speed, as bus capacitance makes higher speeds unreliable. A different signaling method, such as CAN (Controlled Area Network) bus with differential pair can have bandwidth as high as 2Mbit/sec [8].

The download and upload rate is independent of the number of modules, and is shared between modules. For example, one module will have download data rate of 170 kbits/sec but with 10 modules, each one will be allocated 17 kbits/sec. The raw USB data rate from the master to the PC was 12 MBit/sec. The actual data rate depends on the computer's USB driver. Windows 7 driver could not deal well with frequent small packets. MAC OSX (10.9.4) USB driver worked more effectively.

In practice, the raw data rate translates to about 2ms of latency for each module. The latency is shown in Figure 15. Ideally, modules would be updated at a frame rate of 30Hz, which is especially important for animations on LED displays. The data shows that up to 17 modules can be updated in one frame.

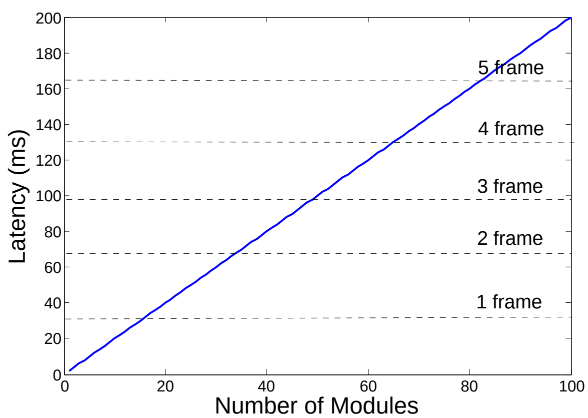


**Figure 15. Latency with different number of modulus. The latency is also shown in terms of frame rate (at 30Hz). For example, updating 50 modules will take 3 frames.**

**Force:** The force of the system relies on the servomotor we use for each module. The servomotor used for ChainFORM has a torque of 0.8kg.cm. As for haptic joystick or gloves, this torque can be perceived by users when the force is applied to a finger according to Walmsley et al[31]. Accordingly, the torque can also be perceived with the back of user as the external spine application, while it is not enough to directly 'move' their back to correct posture. In our prototyping process of locomotive craft application shown in Figure 13e, we observed that the system can hold and move the weight of box up to 350g.

**Power Consumption:** The power consumption is important as it influences how many modules can be connected together. The average power consumption of each module is 83.0 mA, when the LED shield and motors are on. The maximum peak current is 125 mA. When the motor and LED shield are off the power consumption decreases to 13.3 mA. The motors consume the most energy: 51.5 mA.

**Maximum Length:** An important consideration is how many modules can be theoretically connected together. There are three main considerations: communications reliability, power delivery, and address space.

First, the address space is limited to 7 bits or 127 modules. The address space is defined programmatically, so it's not a fundamental limitation.

Second, the power delivery. Each module needs to have enough power to power its parts. With high currents, the resistance of the power wires can create a large voltage drop at modules that are further away from the power supply. Each module needs at least 3.6 V to function and the maximum supply voltage is 9 V. The resistance of 3 cm 28 AWG cables that connect the modules together is 0.0062 Ohm. The maximum resistance of each connector is 0.040 Ohm. The peak current consumption of each module is 125 mA. We calculated the voltage drop with a different number of connected modules, as shown in Figure 16. We found that the maximum number of nodes that can be powered is 32. The main source of resistance are connectors between modules, as each module contains two of them. Without the connectors, up to 112 modules can be powered. It is likely there are more resistance sources, such as the power supply cable, and the PCB traces. A larger number of nodes can be connected if using thicker wires with less resistance and using a different type of connectors. Also, the modules can be powered by two power supplies from two sides of the chain, as we had to do when powering 35 modules.

Third, high-speed I2C bus is very susceptible to parasitic capacitance, that increases with the number of modules. By measuring the RC-constant, we calculate that the base parasitic capacitance is 83 pF and each node adds 3.3 pF of capacitance. This leads us to believe that up to 96 modules can be safely supported until reaching the maximum of 400 pF, as specified in I2S standard [27]. This limitation of I2C can be avoided by using different signaling method, such as differential signaling or by using wireless communications between modules.

The above analysis indicates that the power delivery limits the number of modules to 32. Largely, it is caused by the resistance of the connectors between the modules.

## DISCUSSION AND FUTURE WORK

### Technical Limitation and Improvements

Various technical aspects of the implementation can be improved. The resolution of LED arrays is one obvious spec that can be improved that our current system requires large screen to show only single alphabet as shown in Figure 10a. The resolution can be improved by using a small pixelated display instead of discrete LEDs. Also, there was only single surface of LED arrays, but having LEDs on every surface of the module would expand the display capability especially in 3D configurations. Adding pixelated displays or more displays will require faster communication speeds and more memory

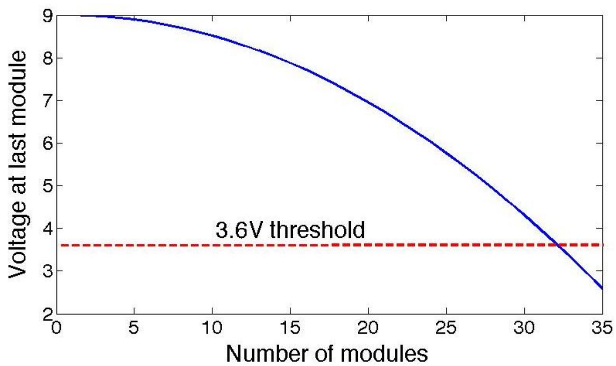


**Figure 16. Simulation of voltage drop at the last module, with different number of modules connected. The module can not function if voltage is below 3.6V. The estimated maximum number of modules is 32**

to maintain real-time interactions. In the future, we plan to use a differential communication scheme to increase the raw data rate to 2 Mbit/sec, which will allow higher resolution displays.

### Improving the Joint for Easy Manual Assembly
The mechanical connection between each module can be designed to include electrical connection as well so that it is easier for users to assemble the device, similar to littleBits [2]. We could design a joint that can be connected in multiple configurations without additional joint, and the configuration could be recognized by sensors or electrical connections automatically. Because our system requires restarting the software when the configuration of the module is changed, we plan to introduce hot plug communication system which can recognize the updated configuration in real time.

### Applying the Architecture to Other Motors and Actuators
Although the size is relatively small to be attached to different materials and objects, the torque is weak for some applications. Utilizing the general design of our implementation technique which uses circuit board design that turns ordinary servo motors into daisy chained controlled system, we could develop a board which can adapt to various servo motors so that users or designers can choose the motors according to their size and torque for appropriate applications which require scalable and modular system. We even foresee that the chained architecture system could be applied to other actuation techniques such as pneumatics for lighter and softer purposes, such as actuating clothing.

### Adding Modules for Further Interaction and Configuration
We also consider one next step of ChainFORM to have other types of modules which provide different sensing and actuation functions. For example, accelerometers, cameras, speakers, mics, or pulse sensing modules could be added for various applications such as input interfaces, mobile devices, and wearables. In addition, we are interested in building a branch modules which can split the module into two or more branches of a chain. Advanced communication architecture is required for such implementation, as they will need to detect the branching.

### Self Assembly Functions
While our system design requires manual assembly for changing the length and configuration, implementing self-assembling function may expand the display and interaction possibility as well. This will allow multiple ChainFORMs to autonomously connect together. Minimizing the size of the module can be a challenge because this function requires adding extra actuators which can clutch other modules as previous reconfigurable robotics system do [35, 17, 33]. Also, it is challenging to develop the algorithms and sensors required for autonomous assembly.

### Conducting Workshops for User Study
Regarding the strong customizability and prototyping aspect of ChainFORM, we are very interested in conducting workshop-based user studies. We plan to observe the usage of ChainFORM and how it would stimulate the creativity of children, designers, and artists. Towards this user study, improvement of the software is required especially for a non-programmer to prototype interactions easily. Also, this would require making the modules easy to manufacture at larger scale.

### Further Applications for Explorative Learning
We would like to explore further applications which demonstrate the novel interaction of ChainFORM, leveraging the customization capability. One domain is learning tools. This kind of application lets users customize length or shape of the device that represents abstract information, then observe behavior through their transformation to learn algorithms or underlying abstract ideas. For example, architecture simulation for learning physics and structure : A user can construct a different structure of buildings or bridges to see how they behave when an external/virtual force (e.g. earthquake) is provided through motion and visual feedback. The physical exploration aspect could be applied to learning molecular structure or protein folding as well.

## CONCLUSION
In summary, we introduced chained modular hardware system that can transform, display visual information, and detect tangible interactions. We presented the broad set of applications that shows the potential interaction capability of the hardware. We believe that the modularity and integrated sensing, actuation and display capabilities of ChainFORM will enrich our interaction for computer interfaces and prototyping tools.

In addition, we envision a future that the ChainFORM system becomes an accessible roll of material just like tapes or strings, so that user can easily cut and connect the material to create customized interfaces, tools, wearables or even furniture that can sense our intention, transform to adapt to our hands and bodies, and inform us through display.

## ACKNOWLEDGMENTS

We thank an undergraduate researcher Kyle Joba-Woodruff for help with implementation, Lining Yao for constructive feedback on this paper and Penny Webb for proof reading.

## REFERENCES


1. Alonso-Mora, J., Breitenmoser, A., Rufli, M., Siegwart, R., and Beardsley, P. Multi-robot system for artistic pattern formation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, IEEE (2011), 4512–4517.
2. Bdeir, A. Electronics as material: littlebits. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, ACM (2009), 397–400.
3. Chien, C.-y., Liang, R.-H., Lin, L.-F., Chan, L., and Chen, B.-Y. Flexibend: Enabling interactivity of multi-part, deformable fabrications using single shape-sensing strip. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, ACM (2015), 659–663.
4. Coelho, M., and Zigelbaum, J. Shape-changing interfaces. *Personal and Ubiquitous Computing 15*, 2 (2011), 161–173.
5. Cubelets. http://www.modrobotics.com/cubelets/, Accessed: 2016-04-13.
6. Dementyev, A., Kao, H.-L. C., and Paradiso, J. A. Sensortape: Modular and programmable 3d-aware dense sensor network on a tape. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, ACM (2015), 649–658.
7. Dynamixel. http://www.robotis.us/dynamixel/, Accessed: 2016-04-13.
8. Farsi, M., Ratcliff, K., and Barbosa, M. An overview of controller area network. *Computing & Control Engineering Journal 10*, 3 (1999), 113–120.
9. Follmer, S., Leithinger, D., Olwal, A., Hogge, A., and Ishii, H. inform: dynamic physical affordances and constraints through shape and object actuation. In *UIST*, vol. 13 (2013), 417–426.
10. Gomes, A., Nesbitt, A., and Vertegaal, R. Morephone: a study of actuated shape deformations for flexible thin-film smartphone notifications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2013), 583–592.
11. Greenberg, S., and Fitchett, C. Phidgets: easy development of physical interfaces through physical widgets. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*, ACM (2001), 209–218.
12. Grossman, T., Balakrishnan, R., and Singh, K. An interface for creating and manipulating curves using a high degree-of-freedom curve input device. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM (2003), 185–192.
13. Hardy, J., Weichel, C., Taher, F., Vidler, J., and Alexander, J. Shapeclip: Towards rapid prototyping with shape-changing displays for designers. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ACM (2015), 19–28.
14. Ishii, H., Lakatos, D., Bonanni, L., and Labrune, J.-B. Radical atoms: beyond tangible bits, toward transformable materials. *interactions 19*, 1 (2012), 38–51.
15. Jacobson, A., Panozzo, D., Glauser, O., Pradalier, C., Hilliges, O., and Sorkine-Hornung, O. Tangible and modular input device for character articulation. In *Proceedings of the adjunct publication of the 27th annual ACM symposium on User interface software and technology*, ACM (2014), 45–46.
16. Katsumoto, Y., Tokuhisa, S., and Inakage, M. Ninja track: design of electronic toy variable in shape and flexibility. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction*, ACM (2013), 17–24.
17. Kurokawa, H., Tomita, K., Kamimura, A., Kokaji, S., Hasuo, T., and Murata, S. Distributed self-reconfiguration of m-tran iii modular robotic system. *The International Journal of Robotics Research 27*, 3-4 (2008), 373–386.
18. Merrill, D., Kalanithi, J., and Maes, P. Siftables: towards sensor network user interfaces. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, ACM (2007), 75–78.
19. Moss. http://www.modrobotics.com/moss/, Accessed: 2016-04-13.
20. Nakagaki, K., Follmer, S., and Ishii, H. Lineform: Actuated curve interfaces for display, interaction, and constraint. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, ACM (2015), 333–339.
21. Niiyama, R., Sun, X., Yao, L., Ishii, H., Rus, D., and Kim, S. Sticky actuator: Free-form planar actuators for animated objects. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction*, ACM (2015), 77–84.
22. Parkes, A., and Ishii, H. Bosu: a physical programmable design tool for transformability with soft mechanics. In *Proceedings of the 8th ACM Conference on Designing Interactive Systems*, ACM (2010), 189–198.
23. Raffle, H. S., Parkes, A. J., and Ishii, H. Topobo: a constructive assembly system with kinetic memory. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM (2004), 647–654.
24. Rasmussen, M. K., Pedersen, E. W., Petersen, M. G., and Hornbæk, K. Shape-changing interfaces: a review of the design space and open research questions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2012), 735–744.
25. Rekimoto, J., Ullmer, B., and Oba, H. Datatiles: a modular platform for mixed physical and graphical interactions. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM (2001), 269–276.

26. Roudaut, A., Karnik, A., Löchtefeld, M., and Subramanian, S. Morphees: toward high shape resolution in self-actuated flexible mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2013), 593–602.
27. Semiconductors, P. The i2c-bus specification. *Philips Semiconductors 9397*, 750 (2000), 00954.
28. Song, H., Benko, H., Guimbretiere, F., Izadi, S., Cao, X., and Hinckley, K. Grips and gestures on a multi-touch pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2011), 1323–1332.
29. Sugiura, Y., Lee, C., Ogata, M., Withana, A., Makino, Y., Sakamoto, D., Inami, M., and Igarashi, T. Pinoky: a ring that animates your plush toys. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2012), 725–734.
30. Thakker, R., Kamat, A., Bharambe, S., Chiddarwar, S., and Bhurchandi, K. M. Rebis-reconfigurable bipedal snake robot. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, IEEE (2014), 309–314.
31. Walmsley, A., and Williams, L. The perception of torque pulses. *Perceptual and motor skills 72*, 3 suppl (1991), 1223–1227.
32. Watanabe, R., Itoh, Y., Asai, M., Kitamura, Y., Kishino, F., and Kikuchi, H. The soul of activecube: implementing a flexible, multimodal, three-dimensional spatial tangible interface. *Computers in Entertainment (CIE) 2*, 4 (2004), 15–15.
33. Wei, H., Cai, Y., Li, H., Li, D., and Wang, T. Sambot: A self-assembly modular robot for swarm robot. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, IEEE (2010), 66–71.
34. Wright, C., Buchan, A., Brown, B., Geist, J., Schwerin, M., Rollinson, D., Tesch, M., and Choset, H. Design and architecture of the unified modular snake robot. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, IEEE (2012), 4347–4354.
35. Yim, M., Zhang, Y., and Duff, D. Modular robots. *Spectrum, IEEE 39*, 2 (2002), 30–34.