ABSTRACT

Title of Thesis: MAKERWEAR: A TANGIBLE CONSTRUCTION KIT FOR YOUNG CHILDREN TO CREATE INTERACTIVE WEARABLES

SeyedMajeed Kazemitabaar, Master of Science, 2017

Thesis directed by: Dr. Jon E. Froehlich
Department of Computer Science

Wearable construction toolkits have shown promise in broadening participation in computing and empowering users to create personally meaningful computational designs. However, these kits present a high barrier of entry for some users, particularly young children (K-6). In this thesis, we introduce MakerWear, a new wearable construction kit for children that uses a tangible, modular approach to wearable creation. We describe our participatory design process, the iterative development of MakerWear, and results from single- and multi-session workshops with 32 children (ages 5-12; M=8.3 years). Our findings reveal how children engage in wearable design, what they make (and want to make), and what challenges they face. As a secondary analysis, we also explore age-related differences.

MAKERWEAR: A TANGIBLE CONSTRUCTION KIT FOR YOUNG CHILDREN TO CREATE INTERACTIVE WEARABLES

by

SeyedMajeed Kazemitabaar

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2017

Advisory Committee:
Dr. Jon E. Froehlich, Chair/Advisor
Dr. Ben B. Bederson
Dr. Mark D. Fuge
Dr. Michelle L. Mazurek

Dedication

To my beloved family
My nephews, and nieces
And all the children in the world

Acknowledgements

Through this long journey that started all the way from *customizable light-up shoes* for kids, I was privileged with the amount of support I received from my advisor, Jon Froehlich over the past three years. I thank him for his passion, creative and visionary mindset, and constructive criticism throughout this work.

Much of the work presented in this thesis was undertaken with the help of a fantastic group of collaborators. I have had the privilege to supervise a group of talented and hard-working undergraduate and high school summer interns, who have contributed to the design, construction and evaluation of MakerWear (and its previous prototypes): Jason McPeak, Alex Jiao, Katie Wang, Tony Cheng, and Chloe Aloimonos.

I am also thankful for having the chance to collaborate with Liang He and Leyla Norooz, two amazing graduate students who have worked closely with me on the research presented in this thesis.

Most of the fabrications done for this project was done in the HCIL HackerSpace and UMD's Rapid Prototyping Center. I had an amazing time tinkering and making using the equipment in these labs and collaborated with amazing undergrads such as Aliaksandr Mamonau and Daniel Bogachek who provided lots of technical assistance.

I would also like to thank several groups and organizations that helped us with our participatory design sessions, evaluation sessions and deployments of our workshops: The University of Maryland's Kidsteam, Maryland's Boys & Girls club, The Bethesda Kid-Museum, and Baltimore's FutureMakers. I also thank all the young participants in our workshops.

I also thank my friends and colleagues at the Makeability Lab, the HCIL, and the department of Computer Science. Thank you all for making grad life fun!

Finally, thanks to my parents and sisters for their unconditional care and love. Thank you for always supporting me.

Table of Contents

List of Figures

Chapter 1: Introduction

Wearable construction kits such as LilyPad [10], Flora [3], and EduWear [36] have shown promise in attracting underrepresented groups to STEM [13], expanding perceptions of computing [31], and empowering users to create self-expressive and personally meaningful computational designs [33]. These kits, however, require programming, an understanding of circuits, and manual craft skills like sewing and soldering. Though this complexity allows users to create diverse and increasingly sophisticated designs—fitting Resnick and Silverman's notion of "wide walls" and "high ceilings" [59]—it also presents significant challenges to young children and can impede playful experimentation and rapid prototyping (echoing [21,32]).

In this thesis, we introduce and examine MakerWear, a new wearable construction kit for young children (K-6) that uses a tangible, 'plug-and-play' approach to wearable creation. MakerWear is comprised of two parts: (i) single-function, electronic modules that, when combined, create complex interactive behaviors, and (ii) a flexible, magnetic socket mesh that is either pre-integrated into clothing or attached post-hoc like a fabric patch. The mesh provides power, a communication infrastructure, and an easy method to attach and remove modules. By manipulating these tangible

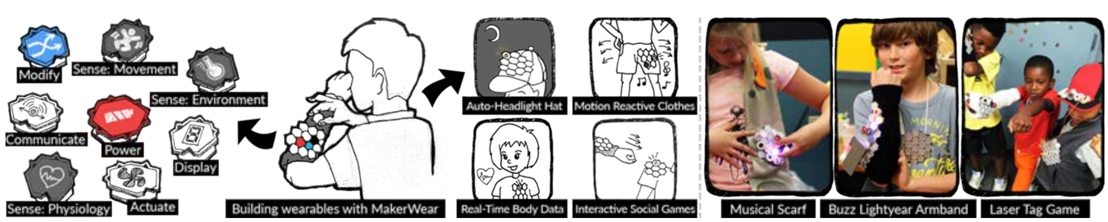


Figure 1.1: ***MakerWear*** **is a new wearable construction kit for young children (K-6). By combining tangible, 'plug-and-play' electronic modules, children can create a wide range of wearable designs. Designs that are reactive to their body and movement, sensitive to changing environments, and that mediate social interactions.**

modules, children can create a wide range of designs, such as: a ‘sound-reactive shirt’ that changes color with music, a ‘fitness tracker’ that automatically counts and displays steps, or a new game of ‘laser tag’ where children interact together through their designs (Figure 1).

MakerWear is informed and inspired by prior work in digital-physical construction kits [70,71] and robotic kits [5,24] that demonstrate how, with appropriately designed tools, young children can develop basic programs and work with electronic sensors and actuators. Wearables, however, present a fundamentally different creative design context. First, constructions are worn and, thus, are inherently social, mobile, and potentially always with the child. Second, the focus of design shifts from electro-mechanical objects to designing for the self—children can create designs that react to their movement, physiology, and changing environment. Third, wearable creation pushes computational design outward from the confines of a room or a screen into the context of a child’s everyday life (e.g., designing electronic wearables to be used in sports and pretend play). Thus, we see MakerWear not just as a platform for creativity and self-expression but as a way for children to augment meaningful experiences in their lives with computation.

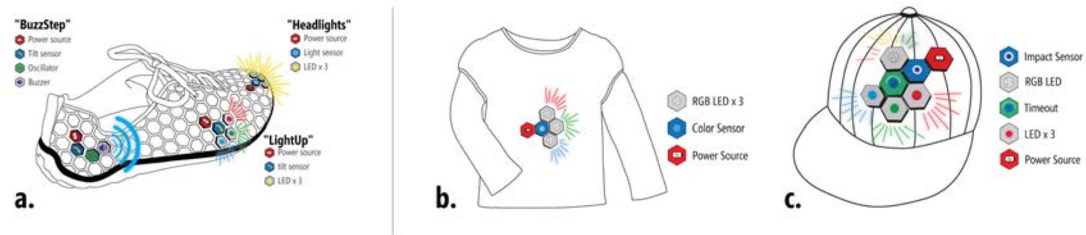


Figure 1.2: Previous generations of our wearable construction kit. (a) MakerShoe: using only a few modules, children can create this MakerShoe that buzzes and lights up during movement and automatically turns on 'headlights' when it's dark. With ReWear, children can retrofit their existing clothes using e-textile modules and build (b) a 'chameleon' shirt that changes color based on surroundings, (c) a hat that flashes with movement.

To build MakerWear, we pursued a two-year iterative design process, beginning with participatory design sessions with children, design probe sessions with STEM educators, and iteratively building and pilot testing two major prototypes with target users: MakerShoe [41] (Figure 1.2a) and ReWear [38] (Figure 1.2b-c). Informed by these experiences, we built MakerWear [40], our final prototype with a focus on enabling children to leverage the richness of wearability—their changing environments, their bodies (e.g., movement, physiology), and social interactions. To examine what and how children make with MakerWear, what challenges arise, and how (and if) children leverage the unique properties of wearability in their designs, we conducted two single-session and three four-session workshops with 32 children (ages 5-12; M=8.3 years). Our findings show how children of all ages were able to build interactive wearable designs and develop understanding of key MakerWear principles (e.g., input/output, sensing, sequencing). The multi-session workshops allowed children to work on their own self-directed projects, which resulted in a broad set of creative wearable designs from fitness trackers to superhero costumes.

Finally, we've been investigating methods to allow children to move from a purely tangible programming approach to a hybrid tangible-graphical approach [25] of wearable design. Thus, to allow more experienced children to build increasingly sophisticated designs using MakerWear, we developed a special 'programmable' MakerWear module and an early prototype for a complementary visual programming tool (similar to Scratch [60]) to program modules via a touchscreen interface. Using this drag-and-drop, block-based programming tool, children can potentially create and upload code to their programmable modules and create their own behaviors.

In summary, our contributions include: (i) the MakerWear system, including the 'plug-and-play' modules and custom socket design, which dramatically lowers barriers to wearable design; (ii) findings from pilot studies and single- and multi-session workshops characterizing how children engage in wearable design, what they make, and the challenges therein; and (iii) an analysis of age-related differences in MakerWear creation and understanding.

Most of the work presented in this thesis has been previously published. Chapter 3 are based on our ACM IDC2015 demo paper [41] and our ACM CHI2016 late-breaking-work paper [38]. Chapter five and six are based on our ACM CHI2017 paper [40].

Chapter 2: Related Work

We draw on two primary sources to inform our designs: (i) developmental psychology and early childhood education (e.g., [15,54,63]) and (ii) programming tools and creative construction kits for younger children (K-6). We also position our contributions within wearable design tools.

2.1. Developmentally Appropriate Design for Young Children

Building developmentally appropriate programming tools and interfaces requires understanding children's unique cognitive, social, and physical development [28]. To inform our work, we draw upon influential developmental and learning theories from psychology and education.

At roughly kindergarten age, there is a well-documented shift in cognitive and physical abilities [68]. Children demonstrate increased attention, self-direction, and logical thinking. Hand and finger control also improves, leading to greater enjoyment of and involvement with fine-motor activities [68]. These children are in what Piaget termed the preoperational stage [54,63]. They begin to think symbolically but struggle with abstract concepts like perspective-taking and mental modeling, so often need to rely on physical representations to help formulate, test, and revise ideas about how the world works [1,54,65]. By around age 7, children enter the concrete operational stage, characterized by the development of logical thought [54,63], but still primarily limited to concrete events or objects rather than abstract ideas. Thus, in our work, we take a tangible approach where wearable designs are built and 'programmed' using physical digital manipulatives [57].

Our work is also rooted in Papert's theory of constructionism, which suggests that the best learning experiences occur when children are actively engaged in designing and creating things [35,52]. Constructionism places a critical focus not just on learning through making but on the social nature of design—that is, that ideas are shaped by the knowledge of an audience and the feedback provided by others [35]. As an outward facing medium, wearables are uniquely social compared to the more insular contexts of other construction kits, presenting opportunities for diverse feedback across social spheres from peers and parents to teachers and coaches. MakerWear also explicitly supports building social interaction through module behaviors (e.g., sending data to other wearers). Finally, Papert stresses that intellectual engagement is heightened when children work on activities and projects that are personally meaningful and interesting [59]. Our work begins to examine how children, enabled by our toolkit, engage with computational design to augment meaningful experiences and objects in their daily life (e.g., sports, pretend play).

2.2. Computational Thinking

There is significant recent interest in engaging children in computational thinking and design [44, 116]. In 2016, the Computer Science Teaching Association (CSTA) and

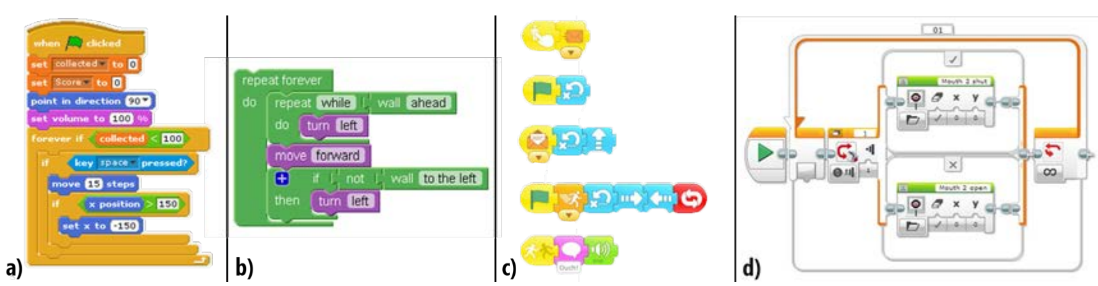


Figure 2.1: Visual programming languages that allow children and novice users to build programs by manipulating graphical blocks: (a) Scratch, (b) Google Blockly, (c) Scratch Jr., and (d) LEGO Mindstorms.

the ACM published draft standards for K-12 education that highlight expectations and learning goals [26]. Most relevant to our target age group for MakerWear, children in grades K-2 are expected to understand basic computing abstractions, develop elementary understanding of sequencing, events, and simple loops, and construct and debug programming statements to accomplish a task (with or without a computer). Children in grades 3-5 are expected to use steps in algorithmic problem solving (e.g., problem statement, design, implementation, testing), develop basic understanding of conditionals, parallelism, and variables, and implement problem solutions using a block-based visual programming language such as Scratch [124] (Figure 2.1a) or Blockly [12] (Figure 2.1b). Despite these emerging standards, most prior work in the area—both in tool support and assessment—has occurred with ages 10 and above (a concern echoed by [64, 66, 87, 115]). Consequently, there is limited knowledge about how younger children (K-5) may engage with computational concepts, how their understanding may develop and deepen over time, what motivations/interests they may have, and how to support those interests through developmentally appropriate tools [64]. As Portelance and Bers recently stated, "there exists a need to create developmentally appropriate tools, pedagogy, and assessments for the learning of computational thinking during early childhood" [115] (p. 271)—this thesis contributes to this call.

2.3. Programming Tools and Construction Kits

A broad set of work exists on building and studying programming tools for children (see reviews [7,43]) though only a small subset is aimed at and evaluated with elementary-age children [55], our target group. For these younger users, two

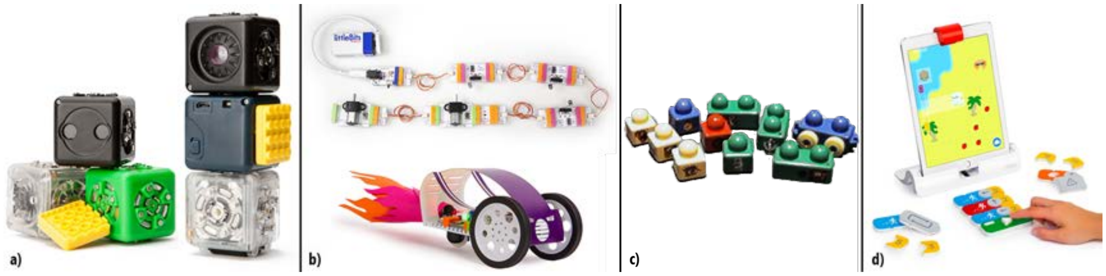


Figure 2.2: Digital-physical construction kits allow children to build interactive behaviors: (a) Cubelets, (b) littleBits, (c) Electronic Blocks, and (d) Strawbies.

approaches are common: (i) simplified graphical, block-based user interfaces like Scratch Jr. [19,66] (Figure 2.1c) and KidSim [56,64] and (ii) tangible approaches that use physical manipulatives such as Tern [26] or Strawbies [29] (Figure 2.2d). For either approach, tool designers attempt to reduce literacy and fine-motor requirements (e.g., typing, mouse input), simplify programming constructs (e.g., eliminate variables), and enforce syntactically correct programming statements through block-shaped constraints—compare Scratch Jr. (ages 5-7) vs. Scratch (ages 8+) [60], for example.

Our work takes a tangible approach. Tangibles can provide sensory engagement for young children [72], an easy entry-point for novices [65], visibility and concreteness of work [6], and opportunities for peer collaboration [26]. Tangibles also support embodied and distributed cognition by allowing children to use their bodies and physical actions to aid thinking and offload some cognitive demands (e.g., mental representations, memory) to the physical world [88]. MakerWear builds on work in digital-physical, tangible construction kits, such as Electronic Blocks [70,71] (Figure 2.2c), roBlocks/Cubelets [48,61] (Figure 2.2a) and littleBits [4,45] (Figure 2.2b), where the entire programming experience—both user input and output—is tangible, without the need for a computer. Despite the popularity of these kits, we could find no empirical examinations with our target age range. One exception is Electronic Blocks (for ages

4-8): two qualitative studies showed that children could build structures with sensor and action blocks but struggled with logic blocks and sequencing. No direct comparisons across ages were made. Though not a purely tangible approach, Marina Bers' extensive work with children (ages 4-7) and robotics also demonstrates that with proper instruction and tools, young children can build and program simple digital-physical constructions, though they struggle with looping, variables, and conditional statements [5,6,37,65,67].

In summary, while not aimed at wearable design, the above studies and tools help demonstrate that even the youngest users in our target range (ages 5-6) are capable of basic programming and working with sensors and actuators. We extend these findings to the context of wearable design.

2.4. Wearable Construction Kits

As noted in the Introduction, wearable construction kits such as LilyPad [10,13] (Figure 2.3) and Flora [2,3] have helped broaden participation in wearable creation but still have high barriers to entry for children. Other wearable toolkits—Quilt Snaps [12], EduWear [36], TeeBoard [49], i*Catch [50,51], and fabrickit [16]—attempt to address some of these issues but are still designed for older children (10+) or adult hobbyists and, consequently, do not provide developmentally appropriate interfaces or architectures for younger ages. For example, EduWear (for ages 10-14) consists of a small set of pre-made fabric modules and a graphical programming interface for Arduino, but is otherwise similar to LilyPad—it requires sewing, creating circuits, understanding analog and digital I/O, and software programming to build even the simplest designs. MakerWear uses higher-order abstractions with a focus on behaviors

Figure 2.3: Wearable construction kits targeted at adults. (a) the Lilypad Arduino wearable kit that includes a number of sewable components: LEDs, Vibration, Buttons, and the main Arduino microcontroller in the middle, (b) using conductive thread to attach Lilypad components to clothes as well as a mechanism for connecting components together. (c) using a mult-imeter to make sure that components are connected correctly.

(e.g., sensing motion, turning on a light) rather than circuits, low-level I/O, and writing code. Moreover, the mesh sockets eliminate the need for sewing.

Closest to our work is i*Catch (for ages 10+), which uses specially designed e-textile clothing with integrated wiring ("host substructure") that interfaces with electronic modules via conductive snap fasteners. While i*Catch eliminates the need for some craft and engineering skills, the approach is still fundamentally software driven and requires writing code. Indeed, each of the aforementioned kits attempt to simplify aspects of building wearables but all use a conventional embedded systems model: a one-microcontroller-to-many-peripherals approach that requires interfacing with a computer to write, compile, and download code. In contrast, our approach allows children to build a range of wearable designs tangibly by manipulating plug-and-play modules. When a module is placed, creations work instantly to better support tinkering and rapid iteration.

Chapter 3: Design Processes

To design and build MakerWear, we employed an iterative, human-centered design process that included participatory design activities with children and design probe sessions with STEM educators. We then conducted lab-based and museum-based pilot studies with increasingly refined prototypes before deploying and studying our final MakerWear prototype in single- and multi-session workshops. Here, we describe the initial participatory process and resulting design goals.

3.1. Participatory Design with Children

To gather design ideas and solicit critical feedback, we conducted five participatory design sessions with children. We employed a participatory design method called Cooperative Inquiry (CI), where adults and children collaboratively brainstorm, design, develop, and test technology [20]. The sessions spanned the design process, from early ideation and lo-fi prototyping to using and critiquing functional prototypes. In total, 11 children participated (6 female) with an average age of 9.2 (range=6-11). Each session included 6-9 children and 3-5 adult co-designers; 8 children participated in more than one session (M=2.9 sessions/child).

All sessions followed the same format: children and adults sat in a circle to answer the "question of the day" (a method of transition into the session [14]), completed a design activity, presented designs, and, finally, discussed "Big Ideas" (where an adult synthesizes repeated, popular, and surprising ideas with feedback from the group [8]). Ideas were thus analyzed in situ during the CI session. A debrief occurred at the session's end.

In the two initial CI sessions, children used low-tech prototyping materials to design and sketch their own interactive clothing and wearable modules. Though the envisioned MakerWear hardware architecture is clothing-agnostic, we were focused on shoes for the first two CI sessions. We believed that interactive shoes such as light-up shoes are probably the most common and fun-to-use interactive clothing a young kid would have encountered by that age. Young children also often imbue their shoes with whimsical notions and powers (e.g., "*these shoes are really fast*"), making them a compelling platform for design.

3.2. First CI Session: 'Blue Sky' Shoes

For the first CI session, we used 'blue sky' methods to elicit unbounded ideas for interactive shoes. The session began with: "If you had a shoe that you could customize to do whatever you'd like, what would it do?" After a round of responses, children and adult design partners were divided into two groups. Each group received a set of low-tech prototyping materials: shoes, a set of blank adhesive cardboard pieces, a large Post-it easel pad ("Big Paper" [6]), and markers. In the design activity (Figure 3.1), groups designed different interactive behaviors on cardboard pieces and adhered them

Figure 3.1: In the first CI session, adult and child teams co-designed their own MakerShoes using 'blue sky' methods. Materials included paper, markers, shoes, and hexagonal cardboard pieces that could be designed to represent functionality and attached to the shoe. For example, (c) shows a prototype shoe with a brake light that turns on when the wearer stops moving, a music button that teaches you how to dance, a pulse sensor, and an ice skate button that turns shoes into ice skates.

to the shoes (e.g., a pulse sensor, a brake light, a button to share secrets with other 'MakerShoe' wearers). As a blue sky session, we did not limit the scope or feasibility of ideas.

Several recurring themes emerged: (i) reacting to body movement and physical actions; (ii) using the wearer's physiology; (iii) designing custom-built games; (iv) appropriating clothing as a social communication device; and (v) pragmatic designs (e.g., increasing visibility at night for safety). In addition, groups wanted the ability to program their clothing to add new functionality (e.g., using Scratch), combine pieces to create custom inputs, and activate multiple pieces simultaneously.

3.3. Second CI Session: 'littleBits' Shoes

For the second CI session, we were interested in exploring how children would conceptualize and use actual electronic modules to design wearable behaviors. For this, we used an existing modular electronic kit called littleBits [1], which we adapted with Velcro attachments to easily adhere to shoes (Figure 3.2). littleBits is a recent but popular commercial construction kit for prototyping electronics through an expansive library of input and output modules that connect magnetically. Similar to the first CI

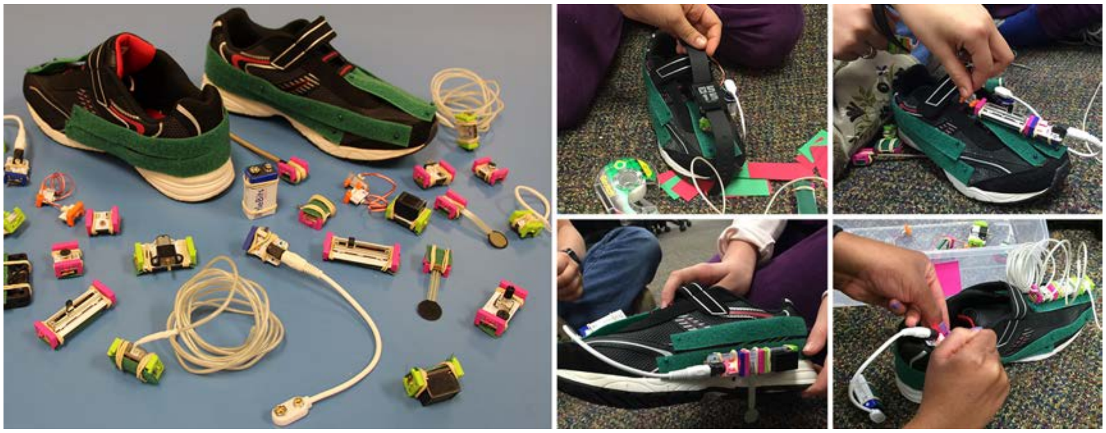


Figure 3.2: In the second CI session, two co-design teams were provided with littleBits and shoes (a) modified with Velcro for attachment. Teams first examined and played with littleBits to build up understanding before designing their own littleBits-based MakerShoes. In the last co-design activity, teams could create new functionality using everyday objects (e.g. a watch) and cardboard (red for input, green for output).

session, we divided the design team into two groups. Each group was provided with a shoe covered in Velcro, a set of littleBits modules including inputs (bend sensor, light sensor, sound trigger, pressure sensor, slide dimmer, oscillator, toggle switch, and roller switch), outputs (vibration motor, dc motor, fan, light wire, led, and buzzer), power (9v battery and cable), and cardboard to describe new functionality (red for input, green for output).

Before breaking into groups, we began with: *"If you had a shoe with a piece of existing technology inside it, what would it be and what would the shoe do with it?"* and provided a brief demonstration of each littleBit module. The design activity began with five minutes of exploration and then ~20 minutes of rapid prototyping with littleBits. Groups then briefly presented their designs before engaging in a second round of rapid prototyping using the cardboard pieces with littleBits. Groups once again presented their ideas and an adult synthesized the results of both prototyping rounds using 'Big Ideas' on a whiteboard.

For the first design activity, each group was able to prototype at least three unique designs. Recurring themes included the use of sound, light, and haptics, which children described with rich narratives that exceeded the behavior of the modules themselves. For example, the sound-generating modules were used to emit humorous noises (e.g., “derp” noises as one walks), communicate with others (e.g., messaging a friend’s phone), and capture others’ attention (e.g., “get-out-of-my-way-horn shoe”). Light-generating and glowing modules were used to beautify shoes (e.g., glowing shoelaces), increase visibility (e.g., “headlight shoes”), and provide notifications (e.g., light-up when your friend is nearby). Haptic outputs and actuators were used to make the shoe more comfortable (e.g., “massaging shoe,” “air-conditioner shoe”).

For the second design activity, groups suggested several new modules for input—a pedometer, motion sensor, temperature sensor—and output—a drawing module, projector module, an LCD display, “bubble blowers”, and “sparkle shooters.” In addition, groups wanted the ability to combine sensor modules to create custom inputs, and activate multiple output modules simultaneously. In summary, while the ideas ranged in feasibility, there was a clear emphasis on customization, personal expression, and, particularly, designing experiences that reacted to the wearer and his/her surrounding environment.

Figure 3.3: adults and children working together and providing feedback on various visual and textual properties of each module (such as their names and icons).

These two sessions helped us synthesize the initial set of design goals for wearable toolkits which set the basis for our first prototype, MakerShoe. In later CI sessions, children helped test initial, semi-functional prototypes, pilot test design activities, and informally use and compare MakerWear to other construction kits Cubelets and littleBits. From these sessions, children suggested new module types, larger socket meshes, and a greater diversity of clothes. Children also helped co-design the look-and-feel of modules (e.g., changing names like *UV Light Sensor* to *Sunlight Detector*).

3.4. Design Probe with STEM Educators

Once we had created initial, semi-functional prototypes, we solicited feedback from two groups of professional STEM educators: staff from an interactive children's museum (N=4) and a STEM education consultancy (N=8). These sessions (Figure 3.4) lasted roughly 90 minutes and included an introduction, a semi-structured interview about educational experiences and making philosophies, a demonstration of the current MakerWear prototype, and, finally, a brainstorm session focused on eliciting design ideas and workshop activities. Both groups of educators were generally positive about our prototypes, particularly the 'plug-and-play' and tinkerability aspects, the use of

Figure 3.4: In the second design probe session with STEM educators, we demonstrated the latest version of MakerWear modules along the previous generation ReWear modules in order to design workshop activities.

wearables as a design platform (e.g., to support movement-based design experiences), and our use of iconography and color to distinguish the module types. They also suggested ideas for new modules, physical designs, and design activities, such as integrating lo-fi materials and trying to design for universal accessibility. Key concerns included: (i) the learnability of modules; (ii) the small size of modules, especially for younger children's (ages 4-5) fine motor abilities; (iii) the robustness of modules, particularly when involved in vigorous activity like running or jumping.

3.5. Wearable Toolkit Design Principles and Goals

Informed by our participatory design sessions, our own experiences building, using, and testing initial prototypes (including early systems [39,42]), and relevant prior work (e.g., [19,59,70]), we synthesized the following key goals for a wearable toolkit aimed at children:

- Leverage wearability. Previous wearable toolkits provide basic components largely undifferentiated from robotic kits (e.g., light sensors, LEDs, speakers). In contrast, we aim to leverage the richness of wearability and mobility—for example, changing environments, children's bodies (e.g., movement, physiology), and social interaction.
- Augment daily experiences. We aim to support designs that are personally meaningful and augment everyday experience, be it socio-dramatic play, soccer practice, or a dance recital.
- Low floors, high ceilings, wide walls. Extending from [59], children-oriented wearable toolkits should be approachable but also support the creation of sophisticated, multi-faceted designs as a child gains experience.

- Tinkerable. Because of a dual reliance on craft skills and programming, previous wearable toolkits limit children's ability to tinker and rapidly prototype—two important aspects of the creative making process [58,69]. Wearable toolkits should allow children to easily try out multiple alternatives, to take things apart, and to create new versions.
- Developmentally appropriate. An overarching principle is to create developmentally appropriate designs informed by the literature and revised through iterative design.

Chapter 4: The MakerShoe and ReWear Systems

Within our iterative design process, we built two major prototypes before MakerWear. Here, we describe the design of MakerShoe and ReWear, and a preliminary evaluation of ReWear with 9 children (ages 6-11; 6 female).

4.1. 1st Prototype: MakerShoe

The MakerShoe prototype consisted of two parts: (i) 3D-printed hexagonal magnetic, plug-and-play modules; (ii) a custom-built e-textile shoe, which supplies power (V_{cc}) and (*GND*) via internal wiring, and has a 3D-printed hexagonal mesh with magnetic sockets for the modules to be attached to the shoe. MakerShoe's modular architecture was strongly inspired by littleBits—we remixed their open-source designs from linear to hexagonal configurations and adapted the modules to a wearable context.

4.1.1. Hexagonal Modules

We developed four types of modules for MakerShoe: *power*, *sensors*, *actions*, and *modifiers*. When placed in the correct orientation in adjacent hexagonal slots on the MakerShoe, an analog connection was formed and new interactive behaviors were created. *Power* modules provide power to all connected modules, *sensors* sense and

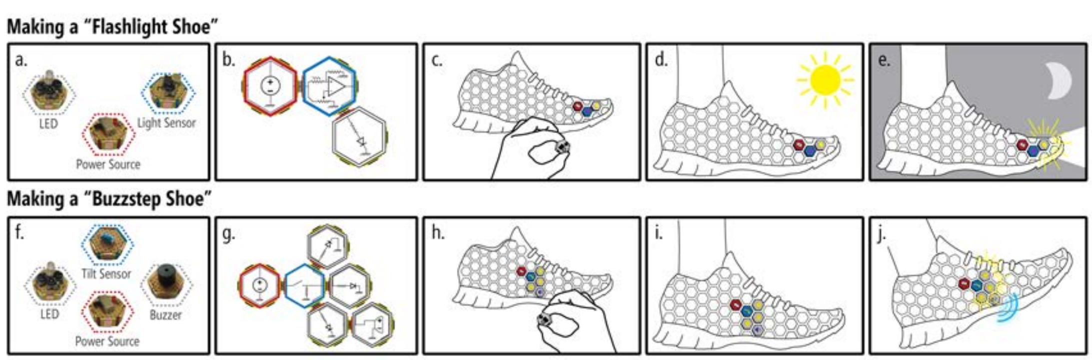


Figure 4.1: Example usage scenarios with MakerShoe showing the hexagonal modules (a, f), the module schematics (b, g), placing the modules (c, h), and wearing the shoes (d-e, i-j).

translate physical phenomena into analog signals (e.g., body movement, light levels), *Actions* translate analog signals into perceptual forms (e.g., sound, light, vibration), and *Modifiers* transform analog signals into other types of analog signals (e.g., oscillators, potentiometers).

Each MakerShoe module had one input signal (S_{in}) and one output signal (S_{out}) duplicated on three sides. The two sides next to (S_{in}) remained open and allowed for tightly packed hexagonal configurations (i.e., to avoid accidental connections between neighboring modules). The exception here was the power module where all six sides were output.

We created 9 prototype modules (Figure 4.2): 2 *sensors* (*Light Sensor* and *Tilt Sensor*), 4 *actions* (*Vibration*, *RGB Adjustable LED*, *Single-Color LED*, and *Buzzer*), and two modifiers (*Oscillator* and *Potentiometer*). Each module had a 3D-printed case made out of unique filament colors that represented the type of that module: *Actions* were white, *sensors* were blue, *modifiers* were green, and *power* was red. For connections between modules, we used copper tape on the input (S_{in}) and output (S_{out}) sides. The copper tapes were internally connected to the electronic circuitry on a small perf-board.

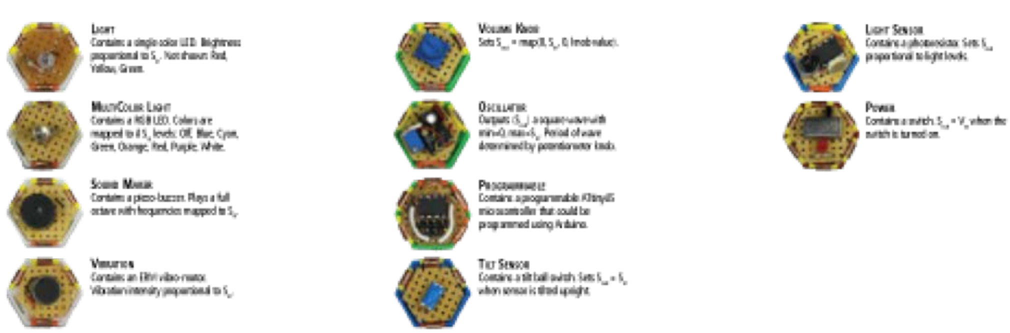


Figure 4.2: Our MakerShoe module design was inspired by and extended littleBits and included 10 modules: 4 actions (white), 3 modifiers (green), 2 sensors (blue, and 1 power (red).

MakerShoe modules, utilized fully analog circuits to create their individual behaviors. For example, a *Light Sensor* used an Op-Amp as a comparator between the output voltage of a photoresistor (V_{photo}) in a voltage divider circuit, and the voltage of a potentiometer (V_{pot}) in order to control the sensitivity of the *Light Sensor*. *Sout* of the *Light Sensor* is V_{cc} when $V_{\text{photo}} > V_{\text{pot}}$ and *GND,* otherwise. Similar to littleBits, all designs began with a power module; however, our approach was unique in that the power module was not actually a direct power source (e.g., a battery) but rather the only module to tap into the shoe's V_{cc}—which was available in each socket and connected to the shoe's internal battery (Figure 4.3b). All subsequent modules received their power from the preceding module's output (the S_{out} connection) and also tapped into the shoe's *GND* rail. Multiple power modules could co-exist on a shoe, which enabled children to design many behaviors together on the same shoe (e.g., Figure 1.2a). This architecture formed the basis of our connected design in MakerShoe, which was substantially changed for the next two prototypes.

4.1.2. The Shoe

The shoe contained an embedded Lithium-Ion Polymer (LiPoly) battery embedded in the sole. Insulated stranded copper wires or conductive thread distributed V_{cc} and *GND* to each hexagonal mesh socket. The 3D-printed hexagonal mesh surrounded the entirety of the shoe and provided secure magnetic slots for attaching modules.

Each 3D-printed hexagonal socket provided connections to all of its six sides using exposed conductive tape. Thus, each socket was a platform that would connect the S_{in} or S_{out} of a module to the S_{in} or S_{out} of another module. In MakerShoe, we did

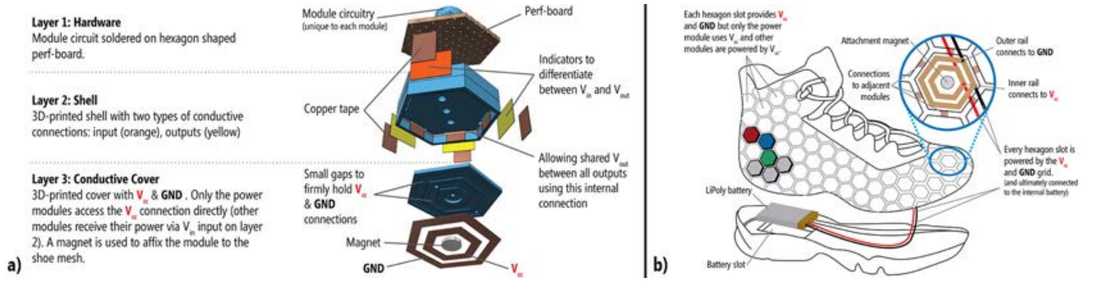


Figure 4.3: Inside the MakerShoe prototype: (a) module's exploded view that demonstrates all three layers including top-layer circuitry, 3D-printed shell with conductive connections to adjacent modules, and the bottom conductive cover for receiving power and ground from the shoe; and (b) the shoe that provides a hexgonal mesh with magnetic slots to attach the modules. V_{cc} and GND are exposed at each mesh slot. Power is provided by an internal LiPoly battery.

not have any mechanisms in order to avoid connections such as S_{in} to S_{in} or S_{out} to S_{out} between two modules.

Finally, we also included inductive charging circuitry to avoid having to plug in the shoe to charge, increasing overall usability. For that, we installed an inductive charging pad inside a doormat and covered the bottom of the shoe with a Universal Qi Wireless Receiver. This enabled the shoe to automatically charge itself when placed on top of the doormat.

4.2. 2nd Prototype: ReWear

Our prior work enumerated five design goals for wearable construction kits for young children. With ReWear, we had an additional design goal: the ability to augment any (existing) garment with computational/electronic behaviors. This required a different approach from MakerShoe, which relied on specialized, custom-built clothing with embedded wiring, power, and pluggable sockets for modules. For ReWear, the main technical challenges were thus: How to power modules? How to connect them without embedded wiring in the clothing? And how to easily attach/detach modules to clothes to promote rapid design iteration? Below, we describe the ReWear architecture, it's module library, and some example designs.

4.2.1. Modular Architecture

Similar to MakerShoe, ReWear used a modular approach comprised of four module types: *power*, *action*, *sensor*, and *modifier*; however, unlike MakerShoe, the underlying architecture used in ReWear was different (e.g., how modules are powered, connected, and how data is transmitted). Similar to MakerShoe, all designs in ReWear started with a power module, however, the power module contained a tiny Li-Poly battery.

Modules were hexagonal in shape with 17.5mm sides with colored fabric enclosures that represented module type (Figure 4.4 and 4.5). The electronic circuitry (hidden within the fabric enclosures) were powered by the *Vcc* and *GND* pins from the previous module, and were all eventually connected to the same power module used at the beginning of the design. Each hexagonal side in ReWear was a potential connection point and connections were formed by snapping the inputs and outputs of two modules together. For example, the simplest possible fully functional design was a power module + an action module such as an LED (which would always be on in this case).

Unlike MakerShoe, we developed two mechanisms in ReWear in order to prevent incorrect connections: (i) Male/Female connections: connections were made by connecting a 3-pin male connector (output) to a 3-pin female connector (input). This

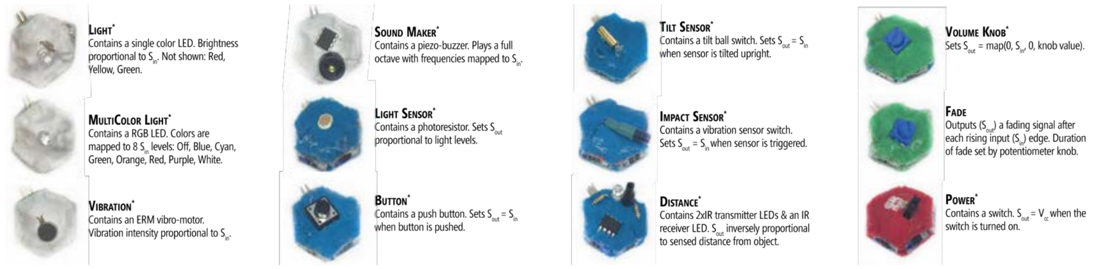


Figure 4.4: ReWear's fabric module library.

automatically created a visual affordance that disabled connecting two output sides together; and (ii) Repelling magnets: small, neodymium magnets embedded on each input/output side helped hold connections and physically repel incorrect links (e.g., if a child tries to put two female sides together).

The three connector pins and their function are similar to littleBits: V_{cc}, *GND* and *Signal*. In ReWear, modules only had one input side but up to five output sides (with the exception of the power module where all six sides were outputs). The hexagonal shape allowed designs to extend and branch into different non-linear forms (e.g., Figure 1.2b-c). While two output sides couldn't be connected, two action modules were able to connect to one another as all action modules forwarded their input V_{cc}, *GND*, and *Signal* to their output pins, enabling cascading designs.

4.2.2. ReWear Module Library

We created 12 different modules (Figure 4.4): 1 *power* module, 5 *sensor* modules (*Light Sensor*, *Distance Sensor*, *Tilt Sensor*, *Impact Sensor*, and *Button*), 2 *modifier* modules (*Timeout*, *Potentiometer*), and 4 *action* modules (*Vibration*, *RGB LED*, *Single-Color LED*, and *Buzzer*). The modules were labeled with child-friendly names

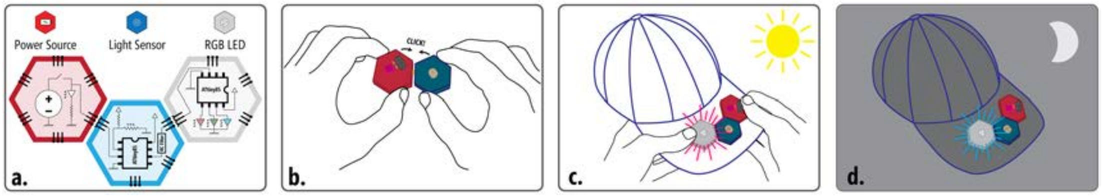


Figure 4.5: Making a hat that changes color based on ambient light levels: (a) electronic schematics, (b) connecting modules, (c, d) testing behavior.

(e.g., the potentiometer was called a 'tuner' module). Six of the 12 modules contained embedded ATtiny85 microcontrollers to read, process, and, optionally, manipulate the Signal. For example, the RGB LED module used the ATtiny's ADC to read the input Signal and translate this into one of eight colors. A simple design with electronic diagram is shown in Figure 4.5. The most complicated module was the *Distance Sensor* that interfaced with an infrared emitter and transmitter through the ADC of the ATtiny in order to output a PWM signal proportional to the module's distance from objects.

4.2.3. Attachment to Clothes

We are still investigating approaches to allow children to attach/detach modules to their clothing. Ideally, the attachment mechanism would be robust, quick, and easy-to-use but non-permanent and non-destructive. We envision children quickly shifting back-and-forth between different work surfaces—a table or floor to prototype different behaviors and shapes and then affixing these designs to their clothes for testing. For

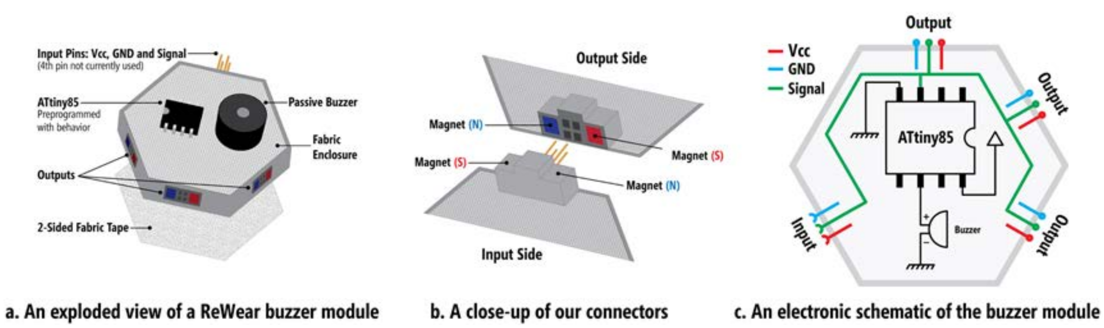


Figure 4.6: ReWear's physical design and electronic architecture.

Figure 4.7: The two other digital-physical kits used in our evaluation: (a,b) Cubelets, and (c,d) logiblocs

attachment, the back of ReWear modules were covered by double-sided fabric adhesive tape (Figure 4.6a), which allowed children to quickly and easily attach/detach modules to their clothing but did not create a strong adhesion (specially to sustain the type of vigorous movement we hope to support) and also lost its adhesive quality over time. We have brainstormed other approaches (e.g hl., safety pins, fusible bonding web) but this remains an important open challenge.

4.3. Preliminary ReWear Evaluation

We performed a preliminary, exploratory evaluation of our initial ReWear prototype in a single two-hour study session with 9 children (ages 6-11; 6 female). To help contextualize our findings, participants used three separate modular digital-physical kits (Figure 4.7), which were placed at different 'maker' stations in our lab: (i) a robotics kit called Cubelets [48], (ii) an electronics kit called Logiblocs [46], and (iii) ReWear. The nine participants were split into three groups; each group had three

Figure 4.8: Some example ReWear designs made during our study session with 9 children.

children and two adult facilitators, which rotated stations every ~25 minutes. The first five minutes included an overview of that station's kit followed by ~20 minutes of open-ended, creative play. For the ReWear station, we reserved the last 10 minutes for two small design challenges: (i) a hat that changes color with movement, and (ii) a bracelet that makes sound when the wearer's hand is raised. Finally, after rotating through the stations, all of the participants reconvened to discuss likes/dislikes, design ideas, and their favorite kit. This discussion was summarized on a whiteboard by a facilitator.

4.3.1. Findings

We collected video recordings and pictures of the making activities and the final all-group discussion. For analysis, we pursued a mixed deductive and inductive approach (informed by Chi's 8-step process [14]). Two researchers independently analyzed video recordings and developed summaries with examples and photos. The researchers then met to discuss and co-interpret common themes. We summarize our findings below.

4.3.2. Making Process

For all three kits, we observed an iterative process of playful experimentation, ideation, creation, testing, and debugging. As children were in groups of three, the making activity was highly social and iterative, with children collectively brainstorming, problem solving, and critiquing designs. Some children would immediately dive into the making activity and experiment with modules. Others were more tentative, watching first before constructing. Sometimes a child would break off from the core group to focus on his/her own design but would soon return with a new design or insight to share.

The Cubelets and ReWear designs seemed considerably easier to iterate on compared with Logiblocs—perhaps due to their magnetic connections and swappable, unified modules. In contrast, Logiblocs come in many different shapes and sizes, with plastic, slotted connections that some found difficult to use (e.g., to pull apart). With ReWear, we also observed a slightly different testing procedure that involved first examining a design's behavior by manually simulating a wearable experience (e.g., holding and shaking a design) before actually placing the design on clothing where it was tested again. Even before the design challenge, seven of the nine children tried to wear their creations (e.g., on a sock, shirt, or pants).

4.3.3. Common Problems

Common problems across all kits included: difficulty comprehending specific module identities and behaviors (e.g., "What does this module do?") and misunderstanding the correct order of modules. With ReWear, we observed children putting action modules before sensor modules, forgetting that power modules needed to come first in a design, and struggling to understand some modifiers and sensors (e.g., tilt sensor, timeout module). Unlike action modules, which could be directly attached to a power module to examine their behavior, modifiers and sensors required more sophisticated designs for testing. Some children brainstormed creations that would require combinational logic (e.g., ANDing sensors), which we do not support. Others suggested a new ReWear module type, a bridge, that would function simply as a 'wire' between modules to allow for longer designs (both Cubelets and Logiblocs have these).

4.3.4. ReWear Creations

Children made a wide range of creations (Figure 4.8) from the simple—a button-activated wearable light—to the more complex—using an impact-sensor module and LED module to light up clothes upon a shoulder tap. Even for the simpler designs, children would provide backstories ("whenever you score a goal, you would press the button and light up" or "this is a back massager"). One child connected all of the action modules together to make her shirt simultaneously light up, vibrate, and buzz. Standing to show her design, she said "Look at me, I'm beautiful." Another hid an RGB LED module beneath her scarf, which caused an interesting diffusion of light. After positive feedback from her group, she added a light sensor so that the scarf also changed color based on light levels. While making, verbal expressions reflected excitement and satisfaction: "Yay, I'm shining!", "You're glowing!", "You are magic!". Overall, children retrofitted socks, shoes, hats, pants, and shirts.

4.3.5. The Design Challenge

Two groups successfully completed both design challenges, while the third group completed only the hat. While some groups struggled initially to determine which modules were necessary to complete a challenge, they eventually arrived at a solution through discussion and experimentation. For example, one group initially used an impact sensor for the second challenge rather than a tilt sensor but discovered their error through testing. The group that did not finish both challenges ran out of time.

4.3.6. Preferred kit

At the end of the design session, we asked children to vote for their favorite kit: five chose ReWear (4 female) and four Cubelets (2 female); no one chose Logiblocs. Children who chose ReWear liked that it was wearable, simple but versatile, and easy to reconfigure. They also appreciated having multiple power modules that could be used to create and wear more than one design. For Cubelets, children liked that their designs could move autonomously and that it was also versatile. Logiblocs was perceived as, comparatively, too complex and the connections did not always work even when properly placed. Children also struggled with knowing what to design with it.

Our preliminary evaluation demonstrated interest in our platform, showed how it could be used with minimal training to create fun, expressive, and interesting wearable designs, and uncovered problems that future designs should address.

Chapter 5: The MakerWear System

Informed by our experiences building and evaluating MakerShoe and ReWear, we built our final prototype, MakerWear, that included (i) visual and physical affordances that makes it easier for children to understand what each module does and how to use them; (ii) more robust connections to adjacent modules and attachment to clothes; (iii) a broader set of modules that leveraged the unique opportunities of wearability.

Similar to MakerShoe, MakerWear is also comprised of two parts: (i) single-function, 'plug-and-play' magnetic modules that can be combined to create complex interactive behaviors (Figure X); (ii) a flexible, magnetic socket mesh that is either pre-integrated into clothing or attached post-hoc like a fabric patch (e.g., via a safety pin or iron-on Velcro). The mesh provides power (V_{cc}) via an internal LiPoly battery, ground (*GND*), a communication wire (*Signal*), and an easy method to attach and remove modules. The modules and mesh are hexagonal, enabling creations to extend and branch into non-linear forms that are visually interesting and can adapt to clothing contours. Our architecture is scalable—allowing for large cascading designs—and responsive (e.g., modules work instantly when placed and react within 10ms to input). The MakerWear system is open source, including hardware (schematics and board layout), microcontroller software, and design files: https://github.com/MakerWear.

5.1. Module design

Modules are 25.5mm across × 9-30mm in height, depending on the embedded electronics. Each module is colored by type with a laser-etched, child-friendly name and icon on the top layer. There are currently five module types: power (red), actions (white), sensors (black), modifiers (blue), and misc (orange). In addition to power,

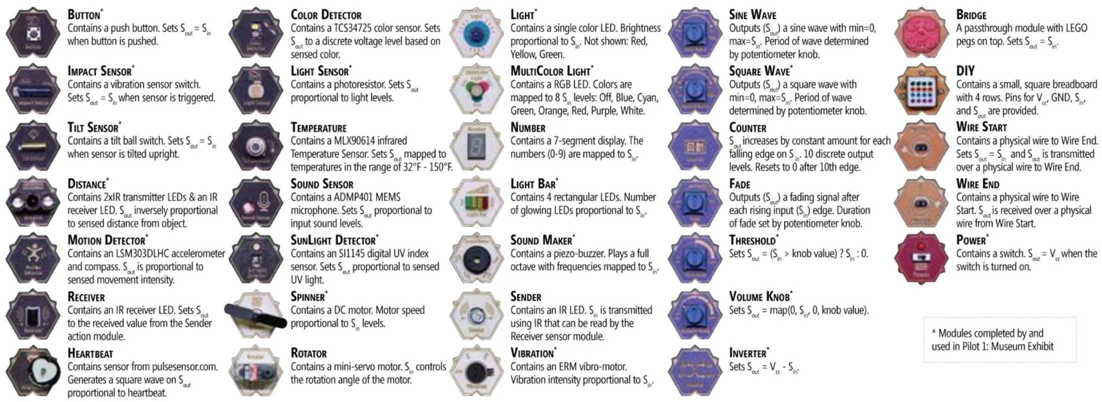


Figure 5.1: The final MakerWear prototype has 32 modules: 12 sensors (black), 9 actions (white), 7 modifiers (bluye), 3 misc (orange), and 1 power (red).

actions, sensors, and modifiers that are the same type of modules used in MakerShoe and ReWear, we built a new type of modules called Miscellaneous (misc) that includes a DIY electronic module for building with raw electronic components and a wire module that allows users to jump across sockets or link multiple socket meshes together.

Each module contains a small embedded microcontroller (either an Atmel ATtiny85 or ATmega328), a custom printed circuit board (PCB), electronic components, and a neodymium magnet (Figure 5.2). The bottom of the module has small, spring-based conductive pins to robustly connect with the mesh. Most modules have one input signal side (S_{in}) and three output sides (S_{out}), indicated by triangular slots and tabs, which fit together like puzzle pieces. In our current prototype, S_{out} is shared (equivalent) on all three sides, but this is not intrinsic to our architecture and future modules could have multiple inputs and outputs. The two remaining sides remain open to prevent accidental connections in tightly packed configurations. While we experimented with both fabric and flexible PCB designs, current modules use a traditional rigid PCB with a laser-cut top made of matboard.

Figure 5.2: Different types of MakerWear socket meshes. (a-c) pre-sewn socket mesh on a scarf, vest, and hat. (d) self-contained socket mesh for attaching to clothes using safety pins.

5.1.1. Module library

Selecting appropriate abstractions and providing a diverse catalog of modules is crucial to any construction kit. In addition to providing standard electronic modules (e.g., LEDs, vibro-motors), we focused on building modules that leveraged the unique opportunities of wearability, particularly: body movement and physiology, social interaction, and the changing environment. Modules range from low-level behavioral abstractions (e.g., an LED module) to higher-level abstractions (e.g., an accelerometer-based motion detector that outputs values corresponding to movement intensity). We have currently designed and built 32 modules (Figure 5.1). While a large number of modules can be overwhelming, the tradeoff is that too few modules could constrain creativity, especially as a user gains experience. As a comparison, Scratch Jr. has 25 programming blocks. In our studies, we introduce blocks incrementally, or exclude some more complicated ones altogether depending on age group.

5.2. Socket mesh design

The socket mesh serves two primary functions: (i) it provides power, GND, and the communication wires, and (ii) an easy, robust mechanism to attach/detach modules to clothes. Each hexagonal socket is made of a PCB base encased by 0.8mm 3D-printed

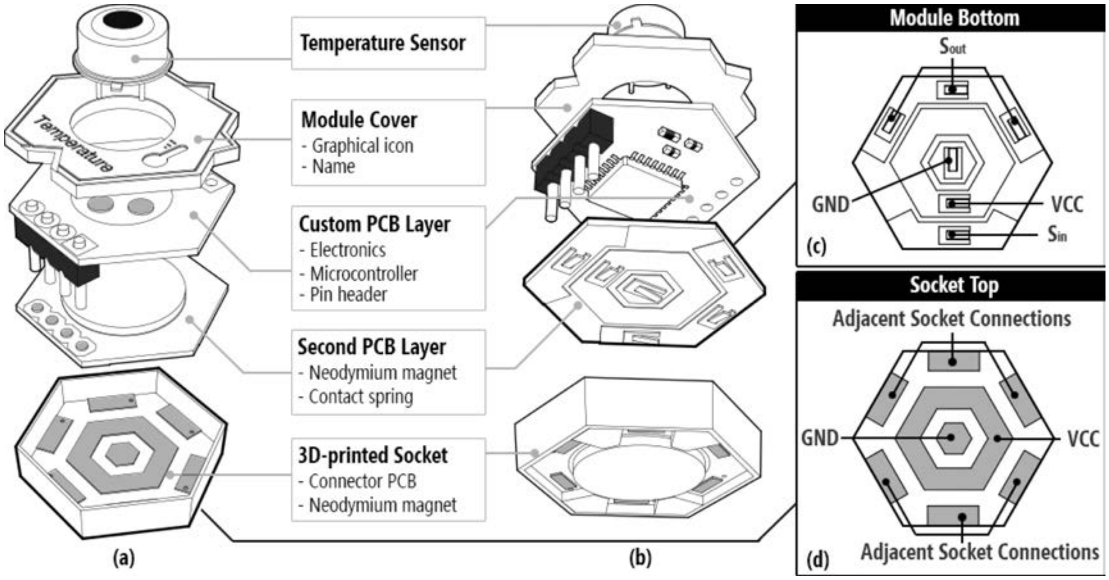


Figure 5.3: An exploded (a) top-down and (b) bottom-up view of an example MakerWear module and socket as well as overhead views of (c) module and (d) socket connector points. The contact springs ensure a robust connection.

walls. Similar to a wooden puzzle with precut slots, the sockets provide a strong visual affordance about how and where to place modules.

In contrast to MakerShoe that used copper tape on the 3D-printed walls of each socket to provide connections between adjacent modules, the base PCB in each MakerWear socket is designed to receive the S_{in}, S_{out}, *GND*, and V_{cc} connections from above. This allowed modules to snap in much easier and with less friction into the sockets compared to sliding inside sockets in MakerShoe. The back of each socket has 6 exposed copper pads that are used for connecting wires to adjacent sockets. The magnet is also glued on the back of each socket.

We have created two types of socket meshes: those integrated directly into clothes (e.g., hats, scarves, vests in Figure 5.2a-c) and a set of self-contained mesh patches (Figure 5.2d) that can be attached to clothes or other artifacts (such as backpacks) via safety pins or worn as jewelry. The meshes are individually wired and contain an integrated, rechargeable LiPoly battery. A small recharging cable is hidden in the fabric material. For the purposes of our research, we focused on clothing that could be easily taken on and off, such as hats, sleeves, and vests. Socket counts range from 14 sockets on a sleeve to 23 on a vest.

5.3. Creating with MakerWear

Wearable creations are built by placing correctly oriented modules in adjoining sockets on a mesh and adjusting on-module knobs, when available. All programs start with a Power module, which has six outputs, all of which set $S_{out} = V_{cc}$. The simplest functional program is thus *power* → *action*. In this design, if the action module is a Blue Light, it would always be on. By adding a sensor, the design becomes interactive. For example, *Power* → *Tilt Sensor* → *Blue Light* would turn on the light when in the proper tilt position and *Power* → *Light Sensor* → *Inverter* → *Blue Light* would turn on the light proportional to darkness level. Finally, because each module has three S_{out} connections, creating non-linear designs is straightforward. A single sensor module, for example, can be directly connected to up to three action modules, activating each simultaneously.

5.4. How does this work?

All module behavior is contingent on its S_{in}, which itself is a function of all preceding modules in the input chain. Action modules forward their S_{in} to their S_{out} ($S_{out} = S_{in}$), while sensor and modifier outputs are a function of two factors: for sensors, $S_{out} = f(S_{in}$, *sensing value*) and for modifiers, $S_{out} = f(S_{in}$, [*on-module knob value*]). We use a hybrid analog-digital design: the analog S_{in} is read in, digitized, and processed by a module's microcontroller and then converted back to analog for S_{out} using pulse width modulation with an RC low-pass filter for smoothing. To ensure that modules with different current consumption (e.g., *Spinner* vs. *Light*) would not cause brownouts, we isolate each module's S_{in} from the previous module by using an op-amp voltage follower or the microcontroller's ADC pin, which uses a high impedance input (100 MΩ).

Depending on the module, custom code on the microcontroller interfaces with its embedded electronics (e.g., via I2C) smooths out S_{in} using a small sliding window, and/or performs some signal processing. For example, *Counter* increases S_{out} by a set amount every time a falling edge on S_{in} is detected. Actions modules either map S_{in} into discrete actions (e.g., the *MultiColor Light* maps S_{in} into 8 different colors) or react proportionally to S_{in}'s voltage (e.g., the *Spinner* or *Vibration*). Similarly, sensors and modifiers either discretize their outputs—e.g., *Color Detector* has 8 and *Counter* has 10 voltage levels—or outputs an analog value between 0-V_{cc}. See Figure 5.1 for complete descriptions. A brief explanation about the circuitry and the software running on their microcontrollers is provided in Appendix A.

Chapter 6: MakerWear Evaluation

To gain preliminary understanding of how and what children could build with MakerWear and to uncover usability issues, we conducted two pilot studies: an interactive museum exhibit and a 1.5-hour workshop. Our findings were used to refine our final prototype as well as our workshop approach.

6.1. Pilot 1: Museum Exhibit

We hosted a 3-hour interactive MakerWear exhibit at a local children's museum. Though open to all attendees, the exhibit was in a small private room to ensure informed consent by a parent or guardian. We set up three MakerWear stations, two of which had a sleeve (14 sockets each) and one of which had a scarf (9 sockets). In total, 17 children participated (ages 4-16; 5 female) and spent an average of 20 minutes (SD=13 mins) using MakerWear. Unlike our later studies, no detailed demographic or questionnaire data was obtained. Two research assistants provided introductory demos to newcomers, answered questions, and facilitated making. At the time of this study, MakerWear had a total of 18 modules, which are marked with a * in Figure 5.1.

6.1.1. Results

Though time-limited and with minimal training, we observed an iterative process of playful experimentation, creation, and testing. Children made a wide range of designs from a simple, button-activated, light-up scarf to a go-away, sneak-up alarm system. Some children reappropriated lo-fi materials from the museum into their designs. Common challenges included: difficulty comprehending module behaviors (especially modifiers), sequencing issues (placing actions before sensors), and some small

Figure 6.1: Some of the designs children made during the museum exhibit pilot study.

technical issues (e.g., faulty socket wiring). Younger children tended to create simpler designs, often only using action modules, and had a tendency to fill up every available socket; however, they also seemed to enjoy themselves. For example, a mother commented about her 4-year old son: “he hasn’t been captivated like that for any other activity in this museum.”

6.1.2. Outcomes

Informed by this experience, we made a few key improvements: (i) we provided more explicit support for lo-fi integration by adding LEGO pegs and Velcro to the Rotator and Bridge and brought lo-fi materials to our sessions; (ii) we increased the number of sockets on MakerWear clothes and introduced a wire module to connect multiple meshes; (iii) we created 12 additional modules to inspire a greater diversity in designs, including the Rotator, Number, Heartbeat, Temperature, and Sender and Receiver. These revisions were made before our next pilot evaluation.

6.2. Pilot 2: Single-Session Workshop

To help test the revised MakerWear platform and trial our workshop plan, we conducted a 1.5-hour pilot with 6 children (4 female) ages 5-9 (M=6.8; SD=1.3). At this point, we had 30 modules (everything except Sound Sensor and Bridge); however,

Figure 6.2: Some of the designs that children made during the single-session pilot workshop.

only 20 were used here due to time constraints. Participants could choose from nine pieces of clothing: two vests (23 sockets), two sleeves (14), two hats (15 and 19), one scarf (19), and two fabric patches (19). The session began with a questionnaire and demonstration, then alternated between introducing new modules and playtime.

6.2.1. Results

While more structured than the museum exhibit, our results were, surprisingly, more mixed. Though five of the six participants were able to build basic designs using power and action modules as well as a simple modifier, the Volume Knob, only a few were able to confidently build more sophisticated designs (e.g., with sensors). Moreover, sequencing continued to be a challenge, especially for younger participants. Despite these issues, we observed children making connections to their everyday life, being able to accurately describe their designs, and using modules to help problem solve. For example, Zara (girl, age 7) made Power → Impact Sensor → Light and said, “This is exactly how light-up shoes work. When you stomp your shoe, it would light up.” Later, she used a Light Bar to visualize the signal between a Volume Knob and Sound Maker and stated “I made something to show how much power is going up.”

6.2.2. Outcomes

We identified three key areas of improvement: (i) we appeared to overwhelm the children with content and new modules, which caused confusion and frustration; (ii), unsurprisingly, we found that the two younger children needed more time to play with and understand each module; (iii) finally, some of the new sockets and modules malfunctioned, further inducing frustration and confusion. To address these issues, we rewrote our workshop plans to reduce content, split the workshops into age groups, and implemented a more comprehensive testing procedure to find and fix errors before our deployments.

6.3. Study 1: Single-Session Workshops

With a refined MakerWear platform and workshop protocol, we ran two single-session workshops at a local children's museum (N=13; ages 5-12). The goals were similar to our pilot studies—to examine the approachability of MakerWear and how and what children make. A secondary goal was to help inform the design of our multi-session workshops.

6.3.1. Method

Participants were recruited via the children's museum. Sign-ups occurred online with one session for ages 5-7 and one for ages 8+. The workshop was free apart from museum admission ($12). We had five participants (all female) in the younger session and eight (3 female) in the old. See Table 1. Four parents also attended (3 in younger, 1 in older) who provided constructive prompts, helped facilitate making, and offered emotional security, especially to younger children.

N	Age	Gender	Computer Use	Graphical Programming Experience	Electronics Experience
5	*M*=6 *Range*=5-7 *SD*=1	5 girls	1 Multiple times a day 4 A few times a week	1 A few times a week 1 A few times a month 3 Never	1 A few times a month
8	*M*=9.9 *Range*=8-12 *SD*=1.5	3 girls 5 boys	6 Multiple times a day 1 A few times a week 1 Never	1 Once a day 2 A few times a month 5 Almost never/never	2 A few times a week 2 A few times a month 4 Almost never/never

Table 1: Single-session workshop group sizes and demographics.

6.3.2. Procedure

Sessions lasted just over 1.5 hours and included: a pre-study questionnaire (10 mins), an introduction to MakerWear (5 mins), building/playing with MakerWear (70 mins), and a post-study questionnaire (10 mins). A team of three researchers facilitated each workshop. Based on our pilot studies, we prepared slightly different workshop plans for the two age groups (Figure 6.3). The older group had a faster pace, which allowed us to introduce additional modules and design challenges. To reduce confusion and provide time for playful exploration, we used only a subset of our module library—10 modules for the younger group and 16 for the older group (Table 2). New concepts and modules were introduced incrementally, starting with Power then simple actions. When a module was first introduced, we would either explain and quickly demo the module or ask the children to experiment and figure it out themselves. Children had ~5 minutes of playtime to explore each new module. We used the same clothing as in Pilot 2 but

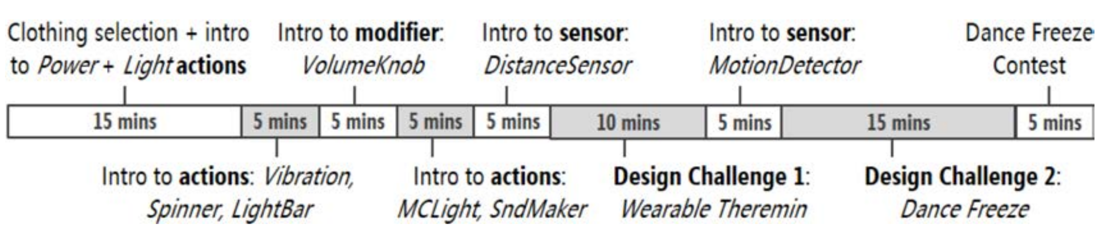


Figure 6.3: The single-session workshop plan for ages 5-7. After receiving a new module, children had ~5 minutes of playtime and experimentation. The workshop for 8+ was similar but had 16 modules and 3 design challenges.

with two new larger sleeves (20 sockets each). Participants selected their clothing at the beginning of the workshop but could switch anytime.

To help assess understanding as well as computational and problem-solving skills, we conducted two design challenges in the ages 5-7 workshop (Wearable Instrument and Dance Freeze) and three in the 8+ workshop (Auto-Flashlight Clothes, Buzz Lightyear, and Dance Freeze)—see Figure 6.4. The workshop ended with a "Dance Freeze Game" where children danced wearing their Dance Freeze designs- in a game similar to musical chairs. When the music stopped, the children had to stop dancing. They were eliminated if their designs were still flashing lights and making sounds, which indicated that they were still moving. During the session, researchers intermittently performed artifact-based interviews [9] to assess understanding and design motivation. Three long mirrors allowed children to see their creations while wearing them.

Ages	Actions	Sensors	Modifiers	Total
5-7	*Light, Vibration, Spinner, Light Bar, MultiColor Light, Sound Maker*	*Distance Sensor, Motion Detector*	*Volume Knob*	10
8+	*+Rotator*	*+Light Sensor, +Tilt Sensor, +Impact Sensor, +Button*	*+Inverter*	16

Table 2 Modules used in our single-session workshops.

6.3.3. Data and Analysis

We used a mixed-methods approach to assess understanding, computational thinking, subjective factors (e.g., enjoyment), as well as to analyze how children made with MakerWear, what they made, and challenges therein. We analyzed session video, design challenge performance, artifact-based interviews, and the pre- and post-study questionnaires. The pre-questionnaire collected demographic data and prior relevant experience. The post-questionnaire asked about how participants felt about MakerWear and their designs as well as questions assessing understanding of module behavior, sequencing, and other computational thinking principles. Questionnaires used a mixture of closed-form, age-appropriate Likert-scale questions [22] as well as free response. For some younger children, questions were individually read by a researcher and responses transcribed.

Multiple video cameras captured how children used MakerWear and their facial expressions, physical movement, and social interactions. To analyze session video, we used a thematic coding approach with a mixture of inductive and deductive codes [8]. Two researchers created an initial codebook based on study goals and pilot study experiences, including for engagement, use of modules, troubleshooting behavior. One researcher coded sample video from the 8+ group, concurrently updating the codebook to support new themes. Two researchers then coded the remaining videos, discussed their findings, and co-interpreted the data.

6.3.4. Findings

We describe key themes and common patterns related to making with MakerWear. For the Likert-scale questions, we report means (M) and standard deviations (SD)—a score of 5 is best. All names are pseudonyms with (age, gender).

6.3.5. Making with MakerWear

Children across both age groups were engaged in making throughout the workshop. Because of the workshop structure and pace, children did not have time to create their own designs like we observed in the museum exhibit or later in the multi-session workshops. Instead, their focus was on understanding and building with modules and completing the design challenges. In terms of MakerWear clothing: six used sleeves, four used the fabric patches, two used hats, and one used a scarf. Interestingly, rather than affixing the fabric patch to clothes, a child in the younger group adapted it with string to wear it as a necklace.

Two styles of making emerged, which was at least partially influenced by clothing type: seven children spent most of their time iteratively building and testing their designs while wearing their clothing (6 sleeves, 1 vest). The other six children primarily built on a table and then infrequently switched to wearing for testing. Of course, some clothing like the two hats made it difficult to build a design while wearing it. Across both groups, children took advantage of the three module outputs and made branching, non-linear designs with interesting visual patterns. While designs with multiple actions were common or even sequences such as action → sensor → action, very few children cascaded two sensors together. One exception was: Dmitry (12, boy) who combined a Button and a Tilt Sensor so that a button press would only work when his design was tilted.

To troubleshoot, children employed three common strategies: (i) they removed and re-added a module either to the same socket or to a neighboring socket that created a functionally equivalent design; (ii) they re-ordered modules around until their design worked as expected—this was the most common solution to solving sequencing issues; and/or (iii) they asked a researcher, parent, or another child for help.

6.3.6. Understanding MakerWear.

We analyzed how children understood MakerWear-specific concepts (e.g., actions vs. sensors, individual modules) as well as higher-level principles related to computational thinking: sequencing, branching, and logic. When first introduced to MakerWear, children quickly understood basic concepts: a Power module is always needed to start a design and that modules had to be in neighboring sockets to be connected. Some children struggled initially with orienting inputs and outputs, which was corrected by facilitators using a puzzle analogy and focusing attention on the I/O triangles. All 13 participants struggled initially with sequencing (e.g., modifiers and sensors must come before actions), particularly in the younger group. This was mostly resolved by the session's end, as evidenced by their design challenge performance and post-study questionnaire responses. For example, on the questionnaire, 12 children (92.3%) correctly fixed a design that had an ordering problem. The one participant who got it wrong (Sayuri, 5, girl) left it blank.

On the post-study questionnaire, all children correctly described action modules—e.g., "they do something like light or move" (Brody, 8, boy). In contrast, for the Volume Knob, which is a modifier module and therefore potentially more complicated, all of the older children described it properly —e.g., "it allows you to control how much power gets through to power the other modules' (Brian, 12, boy) but only two younger children did: "it controls volume and controls actions' (Angel, 6, girl). For those who got this wrong, we observed proper use in our video analysis (e.g., to change light color, sound), so the problem may have been in articulating this knowledge on a written questionnaire. Children also exhibited understanding through their artifact-based interviews. For example, when asked to describe her design, which contained three branches, Angel (6, girl) stated "the power comes from here [points at the Power module] and then it kinda travels here, here and also travels here."

Figure 6.4: The design challenge descriptions and example solutions used in the single-session workshops.

In summary, children seemed to understand differences between module types, module behaviors, and higher-level principles like sequencing and how a signal traverses through a design; however, they had little exposure to modifiers, multiple sensors, and few opportunities to demonstrate understanding of more complex concepts like conditionals.

6.3.7. Design challenges

Generally, most children were able to complete the design challenges (Figure 6.4). For the younger group, all five children successfully created their wearable instruments; however, Elise's (5, girl) father helped her determine which modules to use though she placed them correctly herself. For the older group, all eight children were able to make their auto-flashlight clothes; however, one child required some prompting (e.g., "remember which module allows your design to do the opposite thing"). For the two harder challenges, Buzz Lightyear (only for older group) and Dance Freeze (both groups), performance was more mixed but still largely positive. For Buzz Lightyear (Figure 6.4c), seven children built the tilt-based rotating shield but only five were able to create a design that correctly alternated between attack and shield mode. While we used this challenge to examine understanding of the inverter and control structures, two children created unexpected designs that, instead, used two parallel 'threads' extending from Power. For example, Ellie (9, girl) created two independent branches: (Power →

Figure 6.5: The design challenges that children built during the single-session workshops.

Tilt → Rotator) and (Power → Button → Light); however, this is only partially correct because both the shield and attack modes could be activated at the same time.

Finally, for the Dance Freeze challenge (Figure 6.4d), seven children (1 from younger; 6 from older) confidently built and tested a successful design with no assistance. Three children in the younger group and all children in the older group were eventually able to create working designs with minimal prompting (e.g., "remember, your design should also make sound when you are dancing"). Two younger children received significant assistance from their parents, so we did not count these in our assessment. Children enjoyed playing the Dance Freeze Game with their designs, particularly the older children who asked to play three full rounds.

6.3.8. Overall reactions

In their post-study questionnaires, all 13 children reported wanting to use MakerWear again (M=4.8; SD=0.4) and to bring their design home (M=4.9; SD=0.3). All but one child reported having fun (M=4.4; SD=0.9). The exception was Leiko (7, girl) who marked a ‘2’, but in video analysis was engaged and smiling, and quickly and successfully completed her design challenges. Finally, all but one child reported being proud of their creations (M=4.3; SD=1.1). The exception was Angel (6, girl) who selected 5’s on all other Likert questions. When asked to select a favorite module, action modules were selected most frequently, including the Sound Maker (N=5 votes) because ‘you can change the sound’ (Hiroka, 7, girl), the MultiColor Light (N=3) because ‘it changes color’ (Ellie, 9, girl), and the Distance Sensor (N=2) because ‘it’s like magic’ (Jay, 9, boy). When asked to describe the coolest thing about the workshop, 10 of the 13 children mentioned something that they made such as Buzz Lightyear or Dance Freeze.

6.3.9. Summary

In summary, our findings show that children across age groups were able to understand basic principles such as power, I/O orientation, and sequencing and apply these to build with MakerWear. However, the workshops were time-constrained and focused primarily on basic modules (with the exception of the Inverter in the 8+ group).

6.4. Study 2: Multi-Session Workshops

To gain deeper insight into how children use and understand MakeWear over longer periods of time with a wider variety of modules, we conducted three four-day workshops in after-school programs at two local community centers.

6.4.1. Method

Participants were recruited through the after-school programs and informed consent was obtained before the first workshop. We had 19 participants in total, who were split into different sessions by age: youngest (M=6.3 years), middle (M=8.8), and oldest (M=10.2)—see Table 3.

N	Age	Gender	Computer Use	Graphical Programming Experience	Electronics Experience
7	*M*=6.3 *Rnge*=5-7 *SD*=0.8	3 girls 4 boys	6 At least once per day 1 A few times a week	4 Indicated some exp. 3 Never	4 Indicated some exp. 3 Never
6	*M*=8.8 *Rnge*=8-9 *SD*=0.4	1 girl 5 boys	4 At least once per day 1 A few times a week 1 A few times a month	1 A few times a week 2 A few times a month 3 Almost never/never	2 A few times a week 2 A few times a month 2 Almost never/never
6	*M*=10.2 *Rnge*=8-12 *SD*=1.3	4 girls 2 boys	3 At least once per day 2 A few times a week 1 A few times a month	3 A few times a week 2 A few times a month 1 Almost never	3 A few times a week 1 A few times a month 2 Almost never/never

Table 3: Multi-session workshop group sizes and demographics.

6.4.2. Procedure

Sessions lasted ~1.5 hours and roughly followed the single-session format, intermixing the introduction of new modules with design challenges; however, the multi-sessions covered more content, allowed for more open playtime, and, following Bers et al.'s TangibleK robotics curriculum [5,67], included a final design project (Days 3 and 4). Children completed a pre-study questionnaire at the beginning of the first day. All subsequent sessions started with two 'fix-it' design challenges where children were given pre-made designs with a problem and told to fix it. Each day ended with a ~10-minute post-study questionnaire.

The final project design process was open-ended. Children brainstormed, sketched, and implemented their ideas with intermittent feedback from peers and workshop facilitators. Lo-fi materials like fabric, pipe cleaners, ping pong balls, and LEGOs were provided. On Day 4, projects were presented to family members and peers at an informal exhibition. Overall, we introduced 21 modules in the youngest group—most of the actions and sensors but only two modifiers (Volume Knob and Inverter)—and 31 in the middle and oldest groups (no DIY module because covering circuits was

Figure 6.6: Some of the final projects in our multi-session workshop. (a) LeShawn's light-up sock that vibrated and made colorful lights on each step, (b) Brett's fitness armband that would light-up and make beep sounds with each heartbeat, (c) Dan's ninja armband that vibrated and flashed lights when he performed an uppercut, and (d) Justin's flying armband controlled by a number of knobs and thresholds.

beyond the workshop scope). At least two researchers and one program staff helped facilitate sessions. We used the same mixed-methods approach as Study 1 to analyze Study 2 data.

6.4.3. Findings

We present findings uniquely afforded by the multi-day evaluation: (i) what children designed and built for themselves with their final projects; (ii) age-related differences; and (iii) how children progressed in their understanding and use of MakerWear.

6.4.4. Final projects

Unlike the built artifacts from the single-session workshops, the final projects (Figure 6) allow us to understand what children can and choose to create after gaining experience with MakerWear. For analysis, we focused on project themes, how children used modules in their designs, and the complexity of the artifacts themselves (e.g., number of modifiers). The most common theme was sports/fitness (6 designs), followed by role-play characters like superheroes (5), socio-dramatic play (2), and decoration (2). Two children brought materials from home to use in their designs: a

Figure 6.7: Some of the final projects in our multi-session workshop. (a) Omar's wrecking-ball superhero armband, (b) Amelia's jogging clothes, (c) her clothes while running in the dark, (d) Jake's fitness tracker, and (e) Sarah's sneak attack lacrosse alarm system.

lacrosse stick and a Pokémon doll. In terms of sensing, children most commonly sensed: movement (7 designs), physical actions like pressing or twisting (5), the environment (4), physiology (3), or social interactions (2). Seven designs used at least one modifier, four used more complex control structures with Inverters and Thresholds, and all but one design integrated lo-fi materials.

For the designs themselves, the sports projects included both equipment and clothing augmentation. For example, Sarah (9, girl) added a Distance Sensor to her lacrosse stick along with MultiColor Lights and a Sound Maker to warn her when someone was about to take the ball. Amelia (10, girl) created intricate jogging clothes that included four meshes: a left sleeve that tracked and beeped on every heart beat by using a Heartbeat Detector, a Counter, a Number display, and a Sound Maker, a right

Figure 6.8: Some of the final projects in our multi-class workshops. (a) Kevin's superhero armband, (b-c) Austin's wireless Pokémon doll that was controlled from his vest.

sleeve with two Spinner fans controlled by a Volume Knob to cool her down, and a safety vest and hat that lit up in the dark using a Light Sensor and Inverter. Finally, Jake (11, boy) made a fitness tracker to count steps and, using a Threshold, reward the wearer with flashing lights and sounds if they reached 900 steps.

Role-play characters were more fantastical. For example, Omar (6, boy) made a wrecking-ball superhero armband that lit-up, made sound, and moved a ping-pong ball and pipe cleaners via a Rotator when a button was pressed, while Dan (7, boy) made a ninja armband that vibrated and flashed lights when he performed an uppercut. Finally, as an example of socio-dramatic play, Sean (10, boy) made a Harry Potter Sorting Hat that could tell if someone belonged to Gryffindor or Slytherin. This hat had a social element: one child held the hat while another wore a scarf. The scarf detected what color the person wore (an "evil" or "good" color) and transmitted the data to the hat, which in turn lit up to indicate Slytherin or Gryffindor. These examples illustrate the range of creative possibilities that MakerWear was able to support.

Group	# Modules	# Sensors	# Modifiers	# Control Structures	# Branches	# Lo-Fi
5-7	5.8 (3.7)	0.8 (0.4)	0 (0)	0 (0)	0.6 (0.9)	2.0 (1.3)
8-9	9.3 (4.7)	1.2 (0.8)	1.0 (0.5)	0.5 (1.2)	1.3 (1.0)	2.3 (2.1)
8-12	18.1 (14.2)	2.7 (1.5)	3.7 (1.5)	1.0 (0)	2.0 (3.1)	0.7 (1.2)

Table 4: A quantitative analysis of final projects with format. Cells show average/design and SD. Control structures refers to designs that used thresholds or inverters to setup conditionals. Branches refers to the number of times more than one output of a module was used.

6.4.5. Age-related differences

We analyzed differences related to how children built with MakerWear, the modules they used, the sophistication of their projects, and their understanding of key principles. A quantitative analysis of final projects is shown in Table 4. Unsurprisingly, as age increases, children used not only more modules in their final projects but also more complex modules like modifiers with sophisticated structures (e.g., using Inverters or Thresholds). Interestingly, the opposite appears true with integrating lo-fi materials—the two younger groups, on average, used about two pieces of lo-fi material each while the older group used less than one piece. In the most extreme case, one child (Kayla, girl, 6) made her final project, a puppet, entirely out of lo-fi materials without any modules. Younger children were also more likely to use simple sensors like Buttons to create interaction compared with older children.

Using fix-it challenge and questionnaire data, we analyzed sequencing, branching, and complex cascading (e.g., sensor → sensor or modifier → sensor). We had four fix-it 'sequencing' challenges on Days 2 and 4. The youngest group successfully solved 81% of the challenges while the two older groups solved 100%. Interestingly, while we asked similar questions on end-of-day questionnaires—e.g., by showing a picture of a design with an ordering problem—all children performed worse here: the youngest group scored 44% and the middle and oldest groups ~90%. For branching, we showed two functionally equivalent designs on Day 4, one using linear branching and one using non-linear branching. All of the older children correctly stated that the two designs do not behave differently but only two (33%) in the youngest group did. Finally, we observed only one child in the youngest group use complex cascading—LeShawn (7, boy) who used Tilt Sensor → Inverter to complete a design challenge; these structures were far more common in the two older groups.

In summary, all children were able to create designs with MakerWear and generally understood basic concepts (i.e., I/O orientation, sequencing). The younger children, however, created simpler designs, had difficulty with more complex concepts, and a few struggled even on Day 4 with some basic-to-intermediate principles like branching.

6.4.6. Progressions

We also examined how children's use and understanding of MakerWear changed over the four days as they acquired more experience and more modules were available. Children demonstrated learning both through how they made designs and their questionnaire responses. For example, on Day 1 only 47% of children answered sequencing questions correctly; this jumped to 77% on Day 4. Keisha (6, girl) said, "I remembered that if you put the lights with power, it's just gonna stay on but if you put it after the distance, it will change." For the Threshold module, which was only used in the older group, 20% of children correctly answered questions on Day 3, rising to 78% on Day 4. Two children used a Threshold in their final projects. Finally, introducing new modules opened new opportunities for learning and design. For example, children used the Wire module not just to skip sockets, as expected, but also to connect across meshes and to spread out their designs (e.g., to isolate an Impact Sensor from reacting to Vibration).

6.4.7. Overall reactions

When asked what they thought of MakerWear on the end-of-study questionnaire, all but two participants selected '5' (M=4.9; SD=0.4). When asked to describe their favorite activity, the most common response was the final project (N=7): "My lacrosse stick, because it represented me and what I like to do" (Sarah, 9, girl), "my own super hero thing" (Mike, 9, boy), and "my final project because it was hard to use and fun to make" (Keisha, 6, girl).

Chapter 7: A Preliminary Tangible-Graphical System

Although MakerWear allows children to build a wide range of designs, however, the sole reliance on a tangible, modular approach limits designs to available modules in contrast to completely open kits (e.g., [53]). To address this concern, we are exploring a hybrid tangible-graphical approach [24] that will allow older children or more experienced users to program modules via a touchscreen interface. For that, we've built a preliminary graphical programming and debugging tool that complements the tangible plug-and-play module library.

We had two primary design goals for MakerWear Blockly: first, to enable children in programming new behaviors using a visual programming tool similar to Scratch, and second, to increase the system's transparency for children in both hardware (by visualizing signals) and the software running on a module (using step-by-step debugging).

7.1. The MakerWear Blockly System:

The system which is an add-on to the tangible MakerWear system, consists of two parts: (i) a new programmable module that includes a microcontroller, a small OLED display, and three inputs and outputs; and (ii) a tablet application for visually programming and debugging new behaviors using a block-based programming tool.

7.1.1. Special programmable module

The new programmable module contains an ATmega328 microcontroller and has three inputs and three outputs that are exposed to the user in the visual programming tool for communicating with other modules. Similar to other Sensors and Modifiers, *Inputs* are

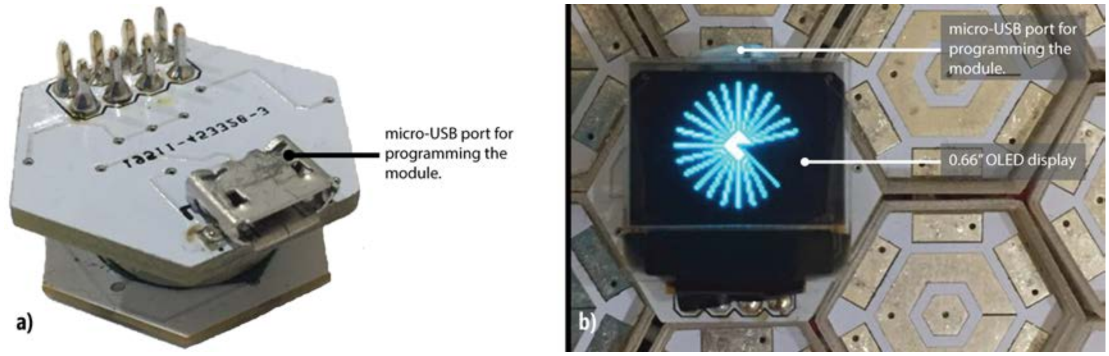


Figure 7.1: modules that are compatible with the MakerWear Blockly visual programming tool: (a) the programmable module, and (b) the programmable+display module.

pulled down using separate 1MΩ resistors and are connected to the microcontroller's ADC ports and *outputs* are connected to separate low-pass RC filters that smooths the output signal from the microcontroller's PWM ports. The microcontroller is also preprogrammed with an Arduino firmware that allows the module to be easily flashed using the RX and TX pins (which is exposed to the user as a micro-USB port on top of the module). Additionally, the module contains a small 0.66 inch (64x48 pixel) OLED display on its top that provides on-module information (such as active I/O ports that are being used in the program that is running on the microcontroller) as well as various programmable UI *Screen Views* such as *Text View, Bargraph View*, and *Signal View*.

7.1.2. Visual programming tool

The visual programming tool is a Google Blockly-based [73] application that has been modified to include MakerWear specific blocks and to generate, compile, and upload an Arduino sketch to the Programmable module (using avr-gcc for compiling the .ino file into a .hex file, and avrdude for flashing the microcontroller with the .hex file).

Using MakerWear Blockly, children can create block-based programs for their special Programmable modules using six categories of drag-and-droppable blocks:

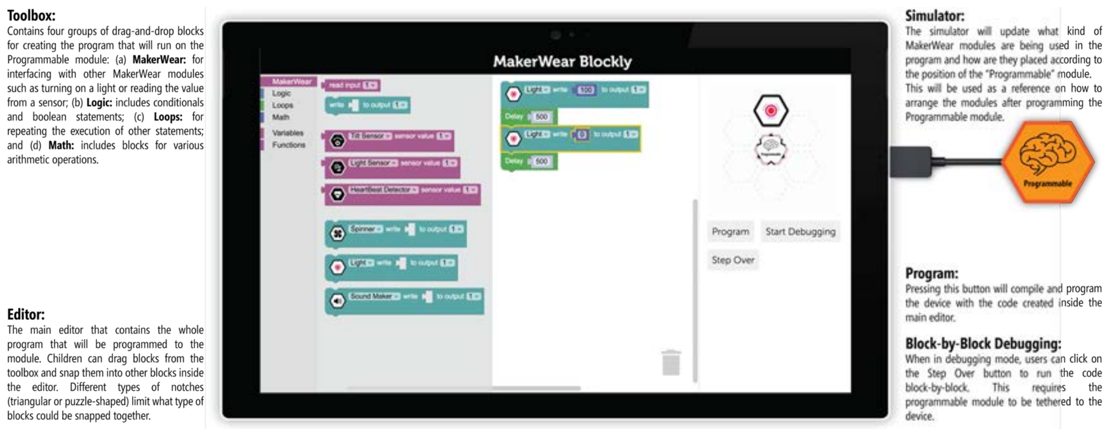


Figure 7.2: MakerWear Blockly system overview.

MakerWear, *Logic*, *Loops*, *Math*, *Variables*, and *Functions*. The *MakerWear* category, contains blocks for interfacing with the module's inputs and outputs through reading and writing signals. Signal values (both input and output) are mapped into 0-100 to represent low-high Voltages. Logic, provides blocks for creating Boolean statements and if-else conditional statements. Loops, provides blocks for repeating the execution of other blocks for a fixed number of times, or based on Boolean conditions. Math, provides blocks for basic arithmetic operations (such as addition, subtraction, multiplication, and division) and more advanced operations (such as generating a random number). Variables, provides blocks for creating, and accessing variables. And similarly, Functions, provides blocks for creating, and calling functions.

7.1.3. Programming

Programs are created by dragging blocks from the toolbox to the main canvas and snapping them together. Currently, the created programs will automatically run in a forever loop and components that require initialization are setup in the background without requiring the user to modify them. When children use *MakerWear* blocks in their program, the top-right simulator will instantly get updated to visualize how and where the new module should be connected to the Programmable module. Programs are flashed to the Programmable module simply by clicking on the “Program” button and using a micro-USB cable. The program will run on the device in a standalone mode unlike Scratch4Arduino which requires the device to be tethered to the computer at all times for running. Finally, the current system is constrained to a limited set of blocks and combinations to avoid syntax errors, however, the tool could be improved in the future by providing warnings for created programs and giving feedback when flashing the module.

7.1.4. Debugging

To increase the transparency of software execution on the device, and to allow children to debug their programs at reduced speeds or block-by-block, we have added a preliminary debugging mode to the tool. When this mode is activated, the system will pre-process the generated Arduino code with various instrumentations such as serial printing the block id that is being executed between each block the user has placed in their program. Pressing the “Start Debugging” button while the module is connected to the computer, will start the debugging mode by highlighting the first block that will be executed. Pressing the “Step over” button will execute that block on the Programmable module and cause the system to move forward by exactly one block.

Finally, although the instrumentations are currently limited to highlighting the block that will be executed next, but the system could be expanded to visualize more information such as the value of any variable over time, or the voltage of any input/output ports as time-series plots to increase the system transparency and allow for more detailed debugging.

7.1.5. Example Program: Fitness Tracker

To create a 2-in-1 smartwatch and fitness tracker we need the following modules: *Programmable+Display*, *Motion Detector*, *Button*, *Light Bar*, and *Sound Maker*. The watch logic and interface is created in *MakerWear Blockly* using a *Timer* block, three variables to keep time (*second*, *minute*, and *hour*), three conditional blocks–to update the time variables on threshold points (e.g. set *second* and *minute* to 0 when they reach 60, or *hour* to 0 when it reaches 24)–and a *Text View* element that allows displaying text on the *Programmable+Display* module.

Similarly, the fitness tracker part of the design is created using a *Read Input* block, an accumulator variable–to keep track of all movements, a *Constrain* function block, and a *Bargraph View* to visualize the user's activity compared to the goal they've set. The program also reads the value of a *Button* module to switch between the two views when it's pressed. Finally, when the user reaches her fitness goal, a Sound Maker will start making sounds in all pitches for three times using a *Count Loop* block.

7.1.6. Example Program 2: The Infection Game

To build this multi-player game using MakerWear Blockly, we need the following modules: *Programmable*, *Sender, Receiver, Button, Single Light, MultiColor Light*. There are two groups of players in this game: clean players that have to survive for a certain amount of time in the game (implemented using a *Timer* block), and infected players that have to infect all other players in order to win (using variables to keep track of the status of other players). Infected players turn red, and clean players turn green which is indicated by a *MultiColor Light*. Each player has two abilities: heal and attack which are activated using two *Buttons* connected to the *Programmable* module. The heal or attack signals are then transmitted to other players using the *Sender* and *Receiver* modules. Abilities have a 5 second cooldown at the beginning of the game which is indicated by a *Single Light*. Clean players need to be hit 3 times before becoming infected. And every time they're hit, they lose a portion of their abilities (*e.g.* increasing the cooldown of their abilities).

The programmable module is programmed to send different types of codes based on each players ability and their infection level. The three inputs of the *programmable* module are connected to the two buttons and the receive, and the three outputs of the *programmable* module are connected to a *MultiColor Light* (displaying infection level), a *Single Light* (indicating the cooldown on their abilities), and a *Sender*.

Chapter 8: Discussion and Conclusion

This thesis contributes a new tangible, modular approach to wearable design called MakerWear. Our findings show that children across our target range (K-6) were able to successfully create a wide variety of wearable designs, actively apply computational thinking (e.g., I/O, sequencing, logics), and create artifacts of which they were proud. In the multi-session workshops, children designed and built final projects that both leveraged the unique properties of wearability and augmented meaningful experiences and objects in their lives (e.g., sports, fictional characters).

8.1. Limitations

In this section, we discuss some of the technical limitations in our current work.

Not Preventing Syntax Errors. We used two approaches to prevent wrong connections between modules in our designs. In ReWear, we used magnets to repel incorrect connections, and to attract correct connections. While, in MakerWear we designed the physical shape of the top cover with inwards input triangles and outwards output triangles to create a puzzle affordance that would prevent incorrect orientations. Although in our multi-session workshop, almost all children were ccorrectly onnecting modules together by the end of the first session, however, there were a number of incidents that children accidentally connected modules in the wrong orientation. On the other hand, construction kits such as littleBits use a mixed approach of both physical affordances as well as magnetic repulsion/attraction to almost completely eliminate incorrect connections between two modules. Coming up with a module design that could completely prevent wrong connections, makes the wearable design process require less cognitive load.

Similarly, using these two mechanisms we were only able to prevent connecting modules with the wrong orientation. Unfortunately, the current system does not prevent or provide any feedback on having an incorrect sequence of modules (such as having an *action* before a *sensor*).

Confusing Module Behaviors. The way we designed the *action* modules required us to directly connect the input and output of these modules. Because of this, incorrect orientations on *action* modules would actually make them work properly. Although this did cause some confusions at the beginning of our workshops, but it seemed that the physical shape of modules and the tiny triangles had a stronger affordance for them to match them correctly, and thus became less of an issue.

Similarly, our current analog architecture limits the knowledge of each module about its surround modules. Because of this, if there are no modules connected before an inverter, they would act as a power module, as with the current architecture, the S_{in} of the inverter is 0 and thus it will output V_{cc} on it's S_{out}. Similarly, the counter module would continue to work and output it's S_{out} even if the power module before it is switched off.

8.2. Robustness and Power Analysis

At this stage in our research, we focused on usability, engagement, and enabling creative design rather than addressing pragmatic wearable concerns such as weight, robustness, and power. While our STEM educators raised concerns about the robustness of an early MakerWear prototype, we did not see a single module fall off during evaluations; however, a few modules did break with use (primarily magnets becoming detached). In terms of power analysis, individual modules draw between 9-

96mA (M=24.8mA) for sensors to 5-80mA (M=46.5mA) for actions. A 14-socket mesh completely filled with the highest current-drawing action modules in their fully-on states draws ~644mA, while a more average design draws half that. Most children's creations, however, include interactivity so are not always fully on, reducing power consumption.

8.3. Directions for Future Research

In this section, we enumerate a number of opportunities for future work that would address some of the limitations in our current research. We first discuss improvements in the current tangible approach of wearable design such as moving from analog to a purely digital architecture, better integration with craft material, and a robust attachment mechanism without requiring socket meshes. Second, we discuss how does introducing a graphical interface (similar to MakerWear Blockly in Chapter 7) opens new avenues such as allowing children to program complex behaviors by demonstration and creating gesture recognizers. Third, we enumerate various evaluation studies to fill the current gap in the literature such as comparing MakerWear to purely graphical methods of wearable design, examining more longitudinal uses of MakerWear and looking for opportunities in collaborative learning.

8.3.1. Improved Wearable Construction Kit

Moving to a Purely Digital Architecture. The current analog communication mechanism between modules causes a number of limitations in the system. First, the RC filter used in most modules has a trade-off between the smoothness of its output and the responsiveness to its input. In order to have a reasonable responsiveness, we had to accept a minimum amount of noise in the S_{out} of each module (around ~5% of V_{max}). This has dramatically reduced the resolution of the information transmitted between modules (to around ~3 bits of data). Second, more electronic components are used due to converting from analog to digital and back to analog in each module: a 1MΩ pull-down resistor and an RC filter (4.7KΩ resistor along with a 0.22uF capacitor). Third, modules do not pass information about the source of the signals they generate to their connected modules. Therefore, these two designs work exactly the same: (1) *Inverter* → *Single Light* and *Power* → *Single Light*.

A purely digital architecture, would not only address these issues, but it would also open up new opportunities for improving MakerWear such as: (i) providing warnings on designs such as having an action module *before* a sensor; and (ii) send multiple lines of data on the same single-output bus (e.g. have the *Motion Detector* to output raw accelerometer data for a *Programmable Module*).

A unidirectional implementation of the digital architecture would require a single serial port on each module. One TX pin that is connected directly to all three outputs of the module, and one RX pin that is connected directly to the module's input.

Better Integration with Craft Material. The role of craft and aesthetics—which is not just a key part of fashion but also touted as one reason why previous wearable toolkits have been successful in broadening participation in CS [31,34,62]—is deemphasized in MakerWear. Future work should explore how to better integrate craft opportunities by: (i) providing more modules, like the Rotator, that easily interface with craft materials; or (ii) Adding a set of new *activator* modules that could be used to control existing wearable electronic pieces such as off-the-shelf fabric NeoPixel

LED Strips, Sewable LED Ribbons, and RGB LED Matrices (e.g. the Adafruit NeoMatrix). The flexibility of these e-textile components allows users to modify and customize their clothes while using MakerWear modules for controlling them.

Retrofitting existing clothes without socket-meshes. While the textile-integrated socket mesh enables our tangible, plug-and-play approach, the mesh itself is not fabric, is relatively heavy, and requires expertise to build. The socket mesh patch (visible in Figure 6f) may offer a nice compromise; it provides the benefits of tangible wearable design but can be attached to any material. A flexible kit should provide the option of specialized clothing or patches to allow for retrofitting of existing clothes.

Allow Expanding MakerWear Modules. During our participatory design sessions with STEM educators, we received many suggestions on allowing workshop organizers or more advanced users to be able to build their own custom modules. For that, we are interested in providing tutorials for expanding our module library with new Actions, Sensors, Modifiers, and other types of modules. Currently, the DIY module, which we did not use in our evaluation sessions, provides a similar opportunity by allowing children to create their own modules from raw electronic components. Modules like DIY may serve as introductory pathways to more complex kits like the LilyPad Arduino.

8.3.2. User Study Evaluations

Comparing MakerWear with Graphical Approaches. While there is limited work in comparing graphical vs. tangible programming, Horn *et al.* [27] found that tangibles were used more by children ("16 and under") and better supported collaborative use compared with a functionally equivalent graphical interface;

however, no significant differences were found for time-of-use, programming comprehension, or the resulting created artifacts. And from a learning perspective, it is currently unclear which elements of using tangible interfaces are critical in supporting learning and which ones are incidental [47].

Although in our work, we've shown that young children were able to create personally meaningful and interactive wearable designs using our tangible construction kit, however, a detailed comparison between MakerWear (as a tangible approach to wearable design) and purely graphical tools to wearable design such as the Adafruit Circuit Playground with MakeCode would highlight the benefits and downfalls of using one approach over another. This would eventually enable better design choices for combining these two approaches (e.g. for our proposed hybrid graphical-tangible visual programming tool discussed in Chapter 7).

Longitudinal Studies of MakerWear. We evaluated MakerWear using a workshop-based study methodology that, while common for construction kit research [11,30,36,44], makes it difficult to assess the effect of curriculum and adult facilitation on outcomes. More research is needed to examine how children would use MakerWear in less-structured environments and for longer periods. Moreover, while we provided both a qualitative and quantitative assessment of age-related differences in using MakerWear, the latter was limited by the content and modules introduced in each workshop, which was imbalanced (the youngest group used 20 modules, the two older groups 30). We have recently partnered with two organizations to examine more longitudinal uses of MakerWear—specifically, in the context of children creating wearables for sports programs and community theatre.

Opportunities for Collaborative Learning. During the MakerWear workshops we observed numerous times where two or more children collaboratively worked together on a single design. Sometimes they built on each other's clothes, sometimes they concurrently used the same socket mesh to develop different parts of a single design. As enumerated by Marshall in [47], tangible interfaces have a strong potential for collaborative learning in which the design space and it's control is shared between learners. This provides increased visibility of other members and the current state of their work, and easier person-to-person interactions. More in-depth evaluations are required to explain what (if any) learning opportunities might exist when using MakerWear in collaborative settings.

8.3.3. Graphical User Interfaces

Programming by demonstration. In our participatory design sessions, children described and created low-fidelity prototypes of an emergent class of behaviors that require classifying sensor streams to triger some action–*e.g.* recognizing physical activities such as throwing, jumping, and twirling. Building on top of our preliminary work in hybrid graphical-tangible approach to wearable design in Chapter 7, we propose an important direction for future work on enabling children to author their own complex sensor-based interactions through a PBD interface that uses interactive machine learning (IML) [17].

Although IML tools such as the Wekinator [18] and Examplar [23] have been successful in enabling adults to build their own gesture recognizers, however, designing developmentally appropriate tools for children to program ML classifiers raises several research questions: (i) what is an understandable way to visualize sensor data, particularly before algebra is taught? (ii) how should the UI provide feedback on the quality of recorded samples (traditionally used confusion matrices)? (iii) how should the tool support deubbing the ML model, for example by updating data or tweaking parameters? (iv) how to create models that are re-usable and sharable?

We envision a new *gesture* module that would include an Accelerometer and a wireless communication mechanism such as BLE and a complementary touchscreen graphical interface for an iterative process of (i) visualizing sensor data, (ii) recording/modifying gestures, and testing the gesture recognizers.

Appendices

Appendix A: Electronics and Software

Includes:

- The open-source repository of MakerWear:

  https://github.com/MakerWear/MakerWear

- The open source repository of MakerWear Blockly:

  https://github.com/MakerWear/MakerWearBlockly

Appendix B: Workshop Study Materials

Includes:

- Pre- and Post-study questionnaires for the single-session workshops
- Pre- and Post-study questionnaires for the multi-session workshops
- Interview questions

Single-session and multi-session workshops pre-study questionnaire:

makeability lab | MakerWear Pre-Activity Worksheet | HCIL Human-Computer Interaction Lab

MakerWear

Pre-Activity Sheet

Name:

Age:

Grade:

Gender (circle one): Boy/Girl

Race/ethnic group:

- ☐ American Indian/Native American
- ☐ Asian
- ☐ Black/Afro-Caribbean/African American
- ☐ Hispanic/Latino
- ☐ White/Caucasian
- ☐ Pacific Islander
- ☐ Other:___________________________
- ☐ I don't know / I don't want to answer

1. How **often** do you use a **smartphone**, **tablet** (like an iPad), or a **computer**?

   ○ Multiple times a day ○ Once a day ○ A few times a week ○ A few times a month ○ Almost never ○ Never

2. A **construction kit** is a toy that **helps you build things**. Some examples include LEGO, SnapCircuits, wooden blocks, MagFormers, etc. How **often** do you play with these kind of toys?

   ○ Multiple times a day ○ Once a day ○ A few times a week ○ A few times a month ○ Almost never ○ Never

3. Have you **ever used** Scratch, Blockly, or some other programming tool that enables you to program a computer, tablet (like an iPad), or a robot?

Yes	No	I don't know
○	○	○

4. If you said **yes** above, how often do you **make programs**?

Multiple times a day	Once a day	A few times a week	A few times a month	Almost never	Never
○	○	○	○	○	○

5. Have you ever **built electronic creations** using things like wires, lights, and batteries—perhaps using a kit like SnapCircuits, littleBits,or LEGO robotics?

Yes	No	I don't know
○	○	○


6. How **often** do you **build things** with **electronics**?

Multiple times a day	Once a day	A few times a week	A few times a month	Almost never	Never
○	○	○	○	○	○

7. Have you **ever created anything** with **fabric** and **sewing**—for example, to make a new stuffed animal, a pillow, or new clothes?

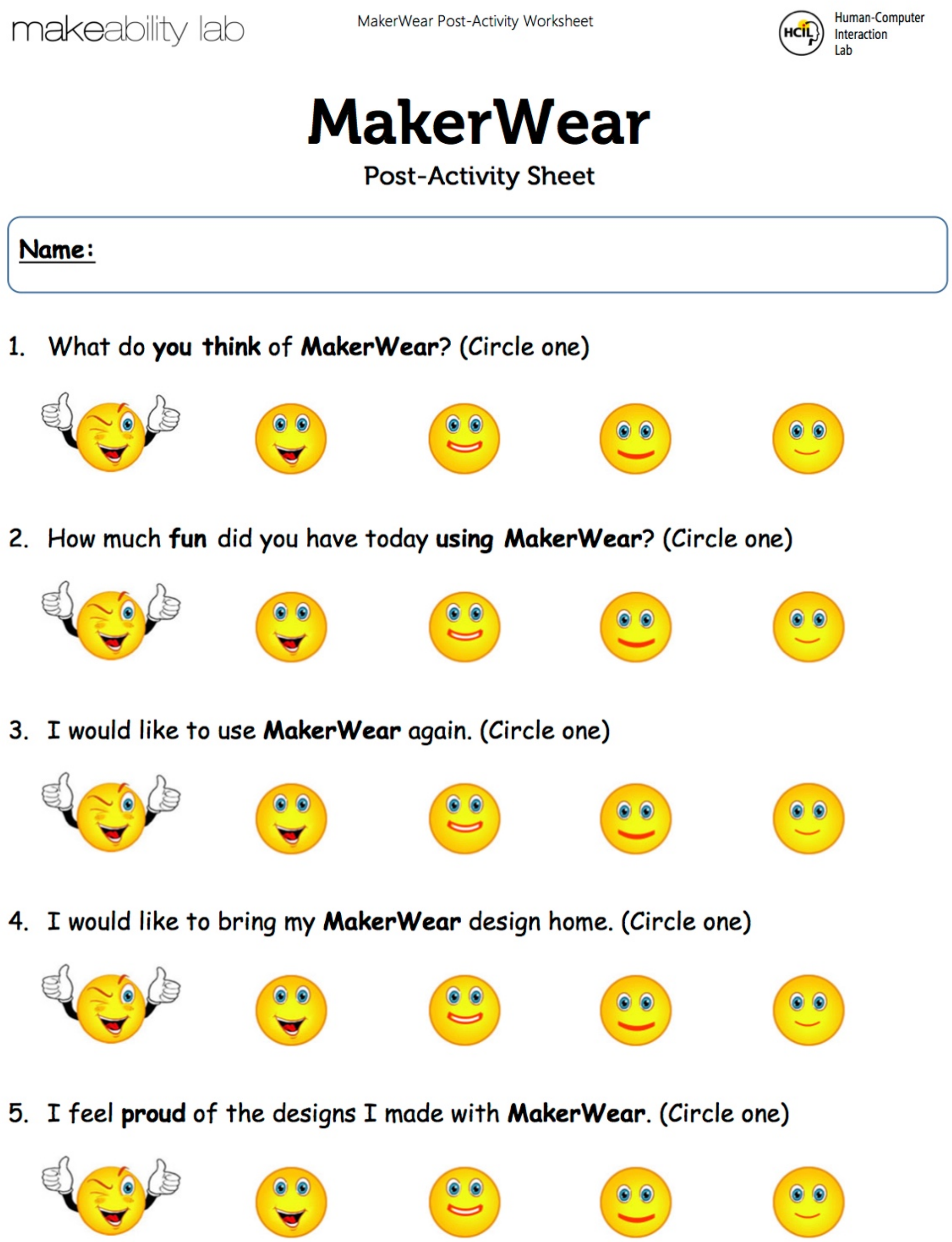
Yes No I don't know

8. Have you **ever** worn an electronic wearable like light-up shoes, a fitbit, a smartwatch?

Yes No I don't know

9. We are **all makers**. Making can include almost any creative activity like coloring or painting, working with crafts, sewing, baking, and programming. **Describe** something **that you made** that **you're proud of**. Feel free to draw/sketch in your response (you can even use the back of this page!).

Single-session and the 1^{st} day of multi-session workshops post-study questionnaire:



makeability lab

MakerWear Post-Activity Worksheet

Human-Computer Interaction Lab

MakerWear

Post-Activity Sheet

Name:

1. What do **you think** of **MakerWear**? (Circle one)

2. How much **fun** did you have today **using MakerWear**? (Circle one)

3. I would like to use **MakerWear** again. (Circle one)

4. I would like to bring my **MakerWear** design home. (Circle one)

5. I feel **proud** of the designs I made with **MakerWear**. (Circle one)

6. What was your **favorite MakerWear module?** Why?

7. What was your **least favorite MakerWear module?** Why?

8. What was the **coolest thing** that you did today?

9. What were some of the **hardest** parts about using MakerWear?

10. What do **Action** (white) modules **do?**

11. What does the **VolumeKnob** do?

12. **Imagine** that your **friend Tom** is trying to build a MakerWear design that **measures the distance to objects. Will his design work?** If not, why not and what would he do to fix it?

13. **Imagine** that your **friend Jenny** is trying to create a MakerWear design that **changes color based on light levels. Will her design work?** If not, why not and what should she do to fix it?

Multi-session workshop post-study questionnaire (Day 2):

makeability lab

MakerWear Post-Activity Worksheet

Human-Computer Interaction Lab

Name:

1. How much **fun** did you have today **using MakerWear**? (Circle one)

1. Do you think **MakerWear** is hard to use? (Circle one)

Yes ○ No ○ I don't know ○

2. What was your **favorite MakerWear module** today? **And why was it your favorite?!**

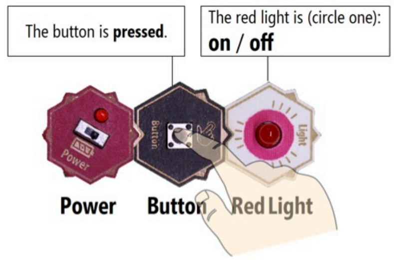
3. What was the **coolest thing** that you did today?

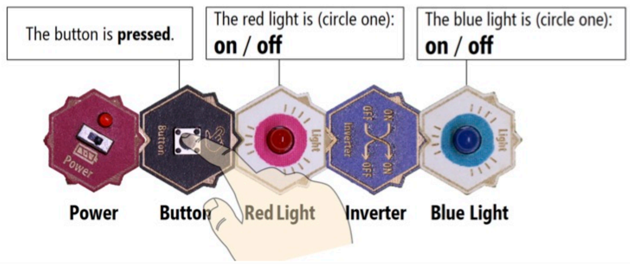
4. What were some of the **hardest** parts about using MakerWear today?

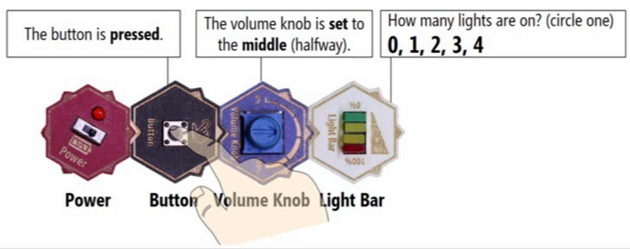
5. What does the **Inverter** do?

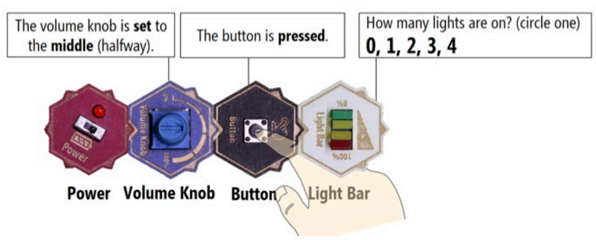
Inverter

6. For each design below, circle the correct answer or answers.









Multi-session workshop post-study questionnaire (Day 3 - Beginner):

makeability lab | MakerWear Post-Activity Worksheet | HCIL Human-Computer Interaction Lab

Name:

1. How much **fun** did you have today **using MakerWear**? (Circle one)

2. Do you think **MakerWear** is hard to understand (Circle one)

Yes — No — I don't know

3. What was your **favorite MakerWear module** today? And why?!

4. What was the **coolest thing** that you did today? And why?!

5. What were some of the **hardest** parts about using MakerWear today?

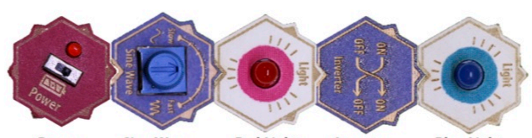
6. If you could invent any **module**, what would your module do?

MakerWear_MultiSession_Day3_PostQuestionnaire_v1_Beginner.docxs 8/16/2016 6:54:58 PM

7. Beyoncé wants her **Light Bar** to light up based on measured **distance**, **would her design work**? If not, what's the correct solution?

8. Beyoncé **tried again** but with a different configuration, would this design work? If not, what's the correct solution?

9. Majeed is going to be a **policeman** for **Halloween** and he's using **MakerWear** to help **design a new costume, what does his design do?**



Multi-session workshop post-study questionnaire (Day 3 - Advanced):

makeability lab

MakerWear Post-Activity Worksheet

Human-Computer Interaction Lab

Name:

1. How much **fun** did you have today **using MakerWear**? (Circle one)

2. Do you think **MakerWear** is hard to understand (Circle one)

Yes — No — I don't know

3. What was your **favorite MakerWear module** today? **And why?!**

4. What was the **coolest thing** that you did today? **And why?!**

5. What were some of the **hardest** parts about using MakerWear today?

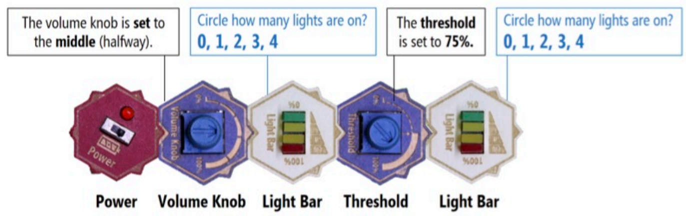
6. If you could invent any **module**, what would your module do?

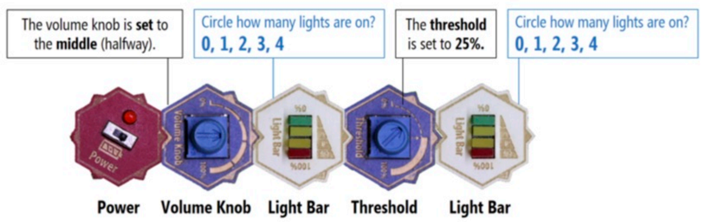
MakerWear_MultiSession_Day3_PostQuestionnaire_v1_Advanced.docxs

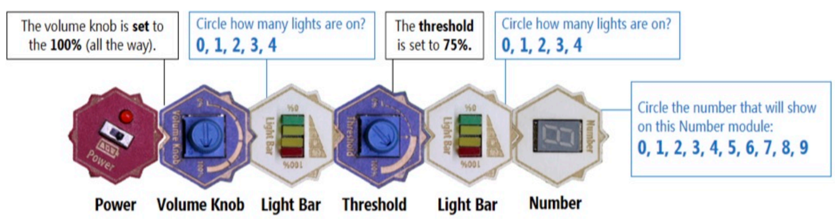
8/16/2016 6:54:44 PM

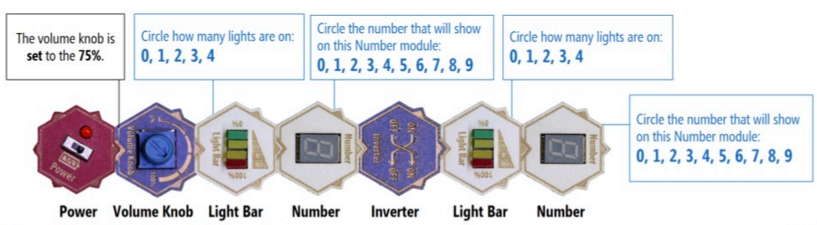
7. What does the **Threshold** do? 

8. For each design below, circle the correct answers or answer.









Multi-session workshop post-study questionnaire (Day 4 - Beginner):

makeability lab

MakerWear Post-Activity Worksheet

Human-Computer Interaction Lab

Name:

1. How much **fun** did you have today **using MakerWear**? (Circle one)

2. Do you think **MakerWear** is hard to understand (Circle one)

Yes — No — I don't know

3. Overall, what do **you think** of **MakerWear**? (Circle one)

4. Overall, what were your **top three favorite** modules?

   i. Favorite Module:____

Why?

   ii. 2nd Favorite Module:____________________________________

Why?

   iii. 3rd Favorite Module:____________________________________

Why?

MakerWear_MultiSession_Day4_PostQuestionnaire_Beginner_v3_JEF.docxs — 8/18/2016 11:05:12 PM

5. What do you **like** about **MakerWear**?

6. What do you **dislike** about **MakerWear**?

7. What was your **favorite thing** that you made **this week** with **MakerWear**? <u>And why was this your favorite?</u>

8. If **MakerWear** could do one new thing, what would it be?

9. What would you like to be able to use with **MakerWear?** Think about different kinds of **clothes** (*e.g.,* jacket, shoes), **things you carry** (*e.g.,* backpack, lacrosse stick), or, really, anything. ☺

10. Jason wants the **Light Bar** to light up based on sensed **Distance**. Is his design correct? If not, what's wrong?

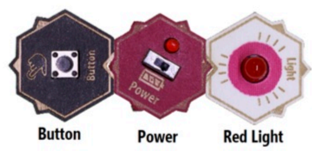
11. Is the **Light Bar** connected to the **Distance** sensor? Mark one.

- O **Yes**, the Light Bar **is connected** to the Distance Sensor
- O **No**, the Light Bar **is not connected** to the Distance Sensor
- O I don't know

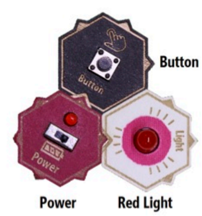
12. Jon wants the **MultiColor Light** to change color based on the **Volume Knob**. Is his **design correct?** If not, what's wrong?

13. Jon **tried again**. Will this **design work?** If not, what's wrong?

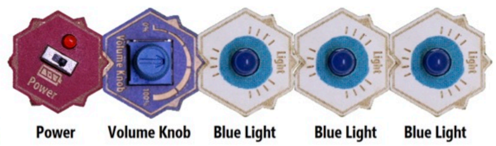
14. Tommy wants the **Red Light** to turn on with the **Button**. Is his **design correct?** If not, what's wrong?

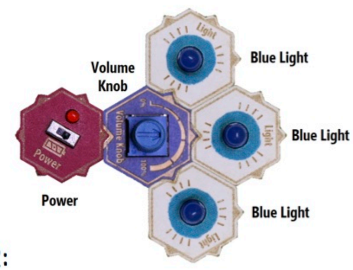


15. Tommy **tried again**. Will this **design work?** If not, what's wrong?



16. What's the **difference** between these two designs? Do they **behave differently**? Circle one: Yes / No

Design 1:


Design 2:


Multi-session workshop post-study questionnaire (Day 4 - Advanced):

makeability lab | MakerWear Post-Activity Worksheet | HCIL Human-Computer Interaction Lab

Name:

1. How much **fun** did you have today **using MakerWear**? (Circle one)

2. Do you think **MakerWear** is hard to understand (Circle one)

Yes — No — I don't know

3. Overall, what do **you think** of **MakerWear**? (Circle one)

4. Overall, what were your **top three favorite** modules?

    i. Favorite Module:____

Why?

    ii. 2nd Favorite Module:____________________________________

Why?

    iii. 3rd Favorite Module:____________________________________

Why?

MakerWear_MultiSession_Day4_PostQuestionnaire_Advanced_v2_JEF.docxs 8/18/2016 11:05:02 PM

5. What do you **like** about **MakerWear**?

6. What do you **dislike** about **MakerWear**?

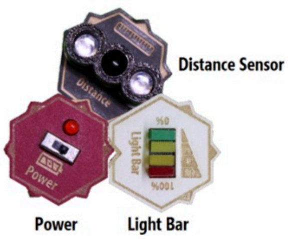
7. What was your **favorite thing** that you made **this week** with **MakerWear**? And why was this your favorite?

8. If **MakerWear** could do one new thing, what would it be?

9. What would you like to be able to use with **MakerWear?** Think about different kinds of **clothes** (*e.g.,* jacket, shoes), **things you carry** (*e.g.,* backpack, lacrosse stick), or, really, anything. ☺

10. Jason wants the **Light Bar** to light up based on sensed **Distance**. Is his design correct? If not, what's wrong?

11. Is the **Light Bar** connected to the **Distance** sensor? Mark one.

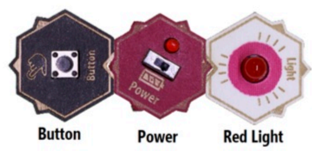


- O **Yes**, the Light Bar **is connected** to the Distance Sensor
- O **No**, the Light Bar **is not connected** to the Distance Sensor
- O I don't know

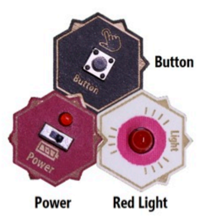
12. Jon wants the **MultiColor Light** to change color based on the **Volume Knob**. Is his **design correct**? If not, what's wrong?

13. Jon **tried again**. Will this **design work**? If not, what's wrong?

14. Tommy wants the **Red Light** to turn on with the **Button**. Is his **design correct**? If not, what's wrong?

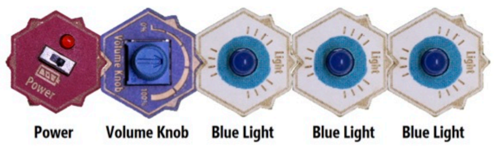


15. Tommy **tried again**. Will this **design work**? If not, what's wrong?

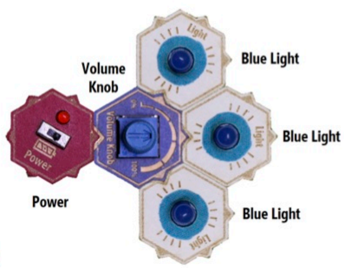


16. What's the **difference** between these two designs? Do they **behave differently**? Circle one: Yes / No

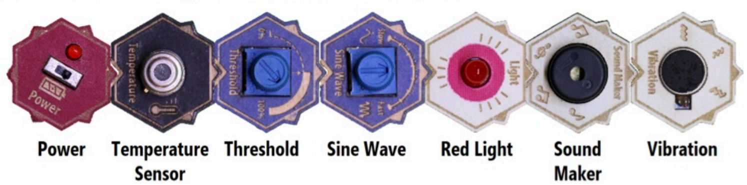
Design 1:



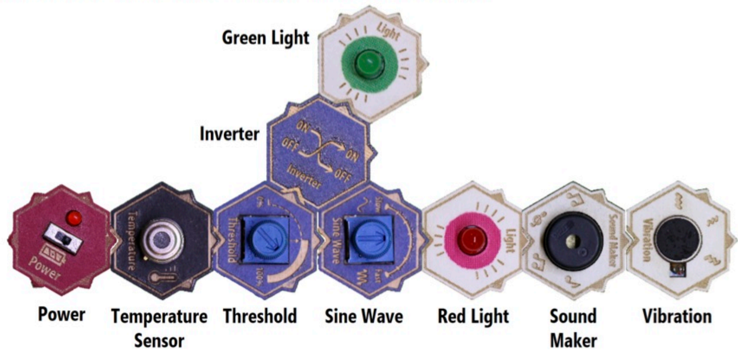
Design 2:



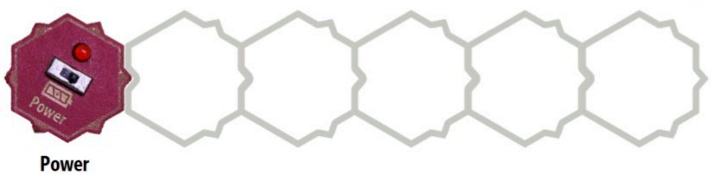
17. You are designing new **MakerWear clothes** for **firefighters**. This is one **design** that you came up with that attaches to the firefighter's helmet or jacket, what does it do?



18. After **testing** your design with some **firefighters**, you decided to make a new prototype. What does this design do **differently**?



19. You and your friends are **designing a new game** to use with MakerWear called "*Never Stop Moving.*" If you **stop moving**, an **alarm sounds** and **lights flash** and you are eliminated from the game. Using MakerWear modules, **build this design**. Write in the module names below. Feel free to draw in more modules as necessary.



Multi-session workshop post-study interview:

makeability lab | MakerWear Post-Study Interview Questions | HCIL Human-Computer Interaction Lab

Interviewee Name:

Facilitators: First, review the interviewee's post study-questionnaire and use their responses as the basis for initial questions. Probe them on their answers a bit. Try to better understand their reasoning and get more in-depth responses. Be conversational and informal. Then, please ask the following. Feel free to add questions but do not subtract (that is, at least ask the following).

1. **What** did you **make** for your **final project**?

2. How did you **come up** with the **idea** for your final project?

3. What was the **coolest thing** that you saw or did with MakerWear this week?

MakerWear_MultiSession_Day4_Interviews_v2_JEF.docxs 8/18/2016 11:04:52 PM

4. What is the **hardest thing** (or things) about using MakerWear?

5. **How can** we make MakerWear **better**?

6. How do you think we could use **MakerWear** for things like sports (e.g., soccer, basketball), dance (e.g., dance competitions), music, and social games like tag?

Bibliography

[1] Ackermann, E. Perspective-taking and object construction. In Y.B. Kafai and M. Resnick, eds., *Constructionism in practice: Designing, thinking, and learning in a digital world*. Lawrence Erlbaum Associates, Mahwah, NJ, 1996, 107–123.

[2] Adafruit Industries, L. FLORA: A wearable Arduino-compatible electronics platform. .

[3] Adafruit Industries, L. Announcing the FLORA, Adafruit's wearable electronics platform and accessories. 2012.

[4] Bdeir, A. and Ullrich, T. Electronics as material: littleBits. *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction - TEI '11*, ACM Press (2011), 341–344.

[5] Bers, M.U., Flannery, L., Kazakoff, E.R., and Sullivan, A. Computational Thinking and Tinkering: Exploration of an Early Childhood Robotics Curriculum. *Comput. Educ. 72*, (2014), 145–157.

[6] Bers, M.U. and Horn, M.S. Tangible programming in early childhood: Revisiting developmental assumptions through new technologies. In *High-tech tots: Childhood in a digital world*. Information Age Publishing, Greenwich, CT, 2010, 49–72.

[7] Blikstein, P. Gears of Our Childhood: Constructionist Toolkits, Robotics, and Physical Computing, Past and Future. *Proceedings of the 12th International Conference on Interaction Design and Children*, ACM (2013), 173–182.

[8] Braun, V. and Clarke, V. Using thematic analysis in psychology. *Qualitative*

Research in Psychology 3, 2 (2006), 77–101.

[9] Brennan, K. and Resnick, M. New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada*, (2012).

[10] Buechley, L. and Eisenberg, M. The LilyPad Arduino: Toward Wearable Engineering for Everyone. *IEEE Pervasive Computing 7*, 2 (2008), 12–15.

[11] Buechley, L., Eisenberg, M., Catchen, J., and Crockett, A. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, ACM Press (2008), 423–432.

[12] Buechley, L., Elumeze, N., Dodson, C., and Eisenberg, M. Quilt Snaps: A Fabric Based Computational Construction Kit. *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education*, IEEE Computer Society (2005), 219–221.

[13] Buechley, L. and Hill, B.M. LilyPad in the Wild: How Hardware's Long Tail is Supporting New Engineering and Design Communities. *Proceedings of the 8th ACM Conference on Designing Interactive Systems - DIS '10*, ACM Press (2010), 199–207.

[14] Chi, M.T.H. Quantifying Qualitative Analyses of Verbal Data: A Practical Guide. *Journal of the Learning Sciences 6*, 3 (1997), 271–315.

[15] Copple, C. and Bredekamp, S., eds. *Developmentally Appropriate Practice in*

Early Childhood Programs Serving Children from Birth through Age 8. National Association for the Education of Young Children, Washington, DC, USA, 2009.

[16] fabrickit. Fabrick Kit. .

[17] Fails, J.A. and Olsen, D.R. Interactive machine learning. *Proceedings of the 8th international conference on Intelligent user interfaces - IUI '03*, (2003), 39.

[18] Fiebrink, R., Trueman, D., and Cook, P.R. A metainstrument for interactive, on-the-fly machine learning. *Proc. NIME 2*, (2009), 3.

[19] Flannery, L.P., Silverman, B., Kazakoff, E.R., Bers, M.U., Bontá, P., and Resnick, M. Designing ScratchJr: Support for Early Childhood Learning Through Computer Programming. *Proceedings of the 12th International Conference on Interaction Design and Children*, ACM (2013), 1–10.

[20] Guha, M.L., Druin, A., and Fails, J.A. Cooperative Inquiry revisited: Reflections of the past and guidelines for the future of intergenerational co-design. *International Journal of Child-Computer Interaction 1*, 1 (2013), 14–23.

[21] Guler, S.D. and Rule, M.E. Invent-abling: Enabling Inventiveness Through Craft. *Proceedings of the 12th International Conference on Interaction Design and Children*, ACM (2013), 368–371.

[22] Hall, L., Hume, C., and Tazzyman, S. Five Degrees of Happiness: Effective Smiley Faced Likert Scales for Evaluating with Children. *Proceedings of the ACM SIGCHI Interaction Design and Children Conference*, (2016).

[23] Hartmann, B., Abdulla, L., Mittal, M., and Klemmer, S.R. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '07)*, (2007), 145–154.

[24] Horn, M.S., Crouser, R.J., and Bers, M.U. Tangible Interaction and Learning: The Case for a Hybrid Approach. *Personal Ubiquitous Comput. 16*, 4 (2012), 379–389.

[25] Horn, M.S., Crouser, R.J., and Bers, M.U. Tangible interaction and learning: the case for a hybrid approach. *Personal and Ubiquitous Computing 16*, 4 (2012), 379–389.

[26] Horn, M.S. and Jacob, R.J.K. Designing Tangible Programming Languages for Classroom Use. *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, ACM (2007), 159–162.

[27] Horn, M.S., Solovey, E.T., Crouser, R.J., and Jacob, R.J.K. Comparing the use of tangible and graphical programming languages for informal science education. *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*, ACM Press (2009), 975.

[28] Hourcade, J.P. Interaction Design and Children. *Foundations and Trends® in Human-Computer Interaction 1*, 4 (2007), 277–392.

[29] Hu, F., Zekelman, A., Horn, M., and Judd, F. Strawbies: Explorations in Tangible Programming. *Proceedings of the 14th International Conference on Interaction Design and Children*, ACM (2015), 410–413.

[30] Jacobs, J., Resnick, M., and Buechley, L. Dresscode: Supporting Youth in

Computational Design and Making. *Constructionism and Creativity*, (2014).

[31] Kafai, Y., Fields, D., and Searle, K. Electronic Textiles as Disruptive Designs: Supporting and Challenging Maker Activities in Schools. *Harvard Educational Review 84*, 4 (2014), 532–556.

[32] Kafai, Y.B., Fields, D.A., and Searle, K.A. Making Technology Visible: Connecting the Learning of Crafts, Circuitry and Coding in Youth e-Textile Designs. *Proceedings of the 10th International Conference of the Learning Sciences (ICLS)*, (2012), 188–195.

[33] Kafai, Y.B., Lee, E., Searle, K., Fields, D., Kaplan, E., and Lui, D. A Crafts-Oriented Approach to Computing in High School. *ACM Transactions on Computing Education 14*, 1 (2014), 1–20.

[34] Kafai, Y.B. and Peppler, K.A. Transparency Reconsidered: Creative, Critical, and Connected Making with E-textiles. In M. Ratto, M. Boler and R. Deibert, eds., *DIY Citizenship: Critical Making and Social Media*. MIT Press, 2014, 179–188.

[35] Kafai, Y.B. and Resnick, M. Introduction. In Y.B. Kafai and M. Resnick, eds., *Constructionism in Practice: Designing, Thinking, and Learning in A Digital World*. Routledge, 1996, 1–8.

[36] Katterfeldt, E.-S., Dittert, N., and Schelhowe, H. EduWear: Smart Textiles As Ways of Relating Computing Technology to Everyday Life. *Proceedings of the 8th International Conference on Interaction Design and Children*, ACM (2009), 9–17.

[37] Kazakoff, E.R. and Bers, M.U. Put Your Robot in, Put Your Robot out:

Sequencing through Programming Robots in Early Childhood. *Journal of Educational Computing Research 50*, 4 (2014), 553–573.

[38] Kazemitabaar, M., He, L., Wang, K., Aloimonos, C., Cheng, T., and Froehlich, J.E. ReWear. *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '16*, ACM Press (2016), 2072–2080.

[39] Kazemitabaar, M., He, L., Wang, K., Aloimonos, C., Cheng, T., and Froehlich, J.E. ReWear: Early Explorations of a Modular Wearable Construction Kit for Young Children. *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, ACM (2016), 2072–2080.

[40] Kazemitabaar, M., McPeak, J., Jiao, A., He, L., Outing, T., and Froehlich, J.E. MakerWear: A Tangible Approach to Interactive Wearable Creation for Children. *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*, (2017), 133–145.

[41] Kazemitabaar, M., Norooz, L., Guha, M.L., and Froehlich, J.E. MakerShoe. *Proceedings of the 14th International Conference on Interaction Design and Children - IDC '15*, ACM Press (2015), 449–452.

[42] Kazemitabaar, M., Norooz, L., Guha, M.L., and Froehlich, J.E. MakerShoe: Towards a Wearable e-Textile Construction Kit to Support Creativity, Playful Making, and Self-expression. *Demo Proceedings of the 14th International Conference on Interaction Design and Children*, ACM (2015), 449–452.

[43] Kelleher, C. and Pausch, R. Lowering the Barriers to Programming: A Taxonomy of Programming Environments and Languages for Novice

Programmers. *ACM Comput. Surv. 37*, 2 (2005), 83–137.

[44] Lau, W.W.Y., Ngai, G., Chan, S.C.F., and Cheung, J.C.Y. Learning Programming Through Fashion and Design: A Pilot Summer Course in Wearable Computing for Middle School Students. *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, ACM (2009), 504–508.

[45] littleBits Electronics Inc. littleBits. .

[46] Logiblocs Ltd. Logiblocs Electronic Discovery System. .

[47] Marshall, P. and Paul. Do tangible interfaces enhance learning? *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07*, ACM Press (2007), 163.

[48] Modular Robotics. Cubelets. .

[49] Ngai, G., Chan, S.C.F., Cheung, J.C.Y., and Lau, W.W.Y. The TeeBoard: an education-friendly construction platform for e-textiles and wearable computing. *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*, ACM Press (2009), 249.

[50] Ngai, G., Chan, S.C.F., Leong, H.V., and Ng, V.T.Y. Designing i*CATch: A multipurpose, education-friendly construction kit for physical and wearable computing. *ACM Transactions on Computing Education 13*, 2 (2013), 1–30.

[51] Ngai, G., Chan, S.C.F., Ng, V.T.Y., et al. i*CATch: a scalable plug-n-play wearable computing framework for novices and children. *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*, ACM Press (2010), 443–452.

[52] Papert, S. *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, Inc., New York, New York, 1993.

[53] Perner-Wilson, H., Buechley, L., and Satomi, M. Handcrafting textile interfaces from a kit-of-no-parts. *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction - TEI '11*, ACM Press (2011), 61.

[54] Piaget, J. *The origins of intelligence in children.* International Universities Press, New York, 1952.

[55] Portelance, D.J. and Bers, M.U. Code and Tell: Assessing Young Children's Learning of Computational Thinking Using Peer Video Interviews with ScratchJr. *Proceedings of the 14th International Conference on Interaction Design and Children*, ACM (2015), 271–274.

[56] Rader, C., Brand, C., and Lewis, C. Degrees of Comprehension: Children's Understanding of a Visual Programming Environment. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, ACM (1997), 351–358.

[57] Resnick, M., Martin, F., Berg, R., et al. Digital manipulatives: new toys to think with. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '98*, ACM Press (1998), 281–287.

[58] Resnick, M. and Rosenbaum, E. Designing for Tinkerability. In M. Honey and D.E. Kanter, eds., *Design, Make, Play : Growing the Next Generation of STEM Innovators*. Routledge, 2013, 182–199.

[59] Resnick, M. and Silverman, B. Some reflections on designing construction kits

for kids. *Proceeding of the 2005 conference on Interaction design and children - IDC '05*, ACM Press (2005), 117–122.

[60] Resnick, M., Silverman, B., Kafai, Y., et al. Scratch: Programming for All. *Communications of the ACM 52*, 11 (2009), 60.

[61] Schweikardt, E. and Gross, M.D. roBlocks: A Robotic Construction Kit for Mathematics and Science Education. *Proceedings of the 8th International Conference on Multimodal Interfaces*, ACM (2006), 72–75.

[62] Searle, K.A., Fields, D.A., Lui, D.A., and Kafai, Y.B. Diversifying high school students' views about computing with electronic textiles. *Proceedings of the tenth annual conference on International computing education research - ICER '14*, ACM Press (2014), 75–82.

[63] Siegler, R.S., DeLoache, J.S., and Eisenberg, N. *How Children Develop*. Worth Publishers, 2010.

[64] Smith, D.C., Cypher, A., and Spohrer, J. KidSim: Programming Agents Without a Programming Language. *Commun. ACM 37*, 7 (1994), 54–67.

[65] Strawhacker, A. and Bers, M. "I want my robot to look for food": Comparing Kindergartner's programming comprehension using tangible, graphic, and hybrid user interfaces. *International Journal of Technology and Design Education*, (2014), 1–27.

[66] Strawhacker, A., Lee, M., Caine, C., and Bers, M. ScratchJr Demo: A Coding Language for Kindergarten. *Proceedings of the 14th International Conference on Interaction Design and Children*, ACM (2015), 414–417.

[67] Sullivan, A. and Bers, M.U. Robotics in the early childhood classroom:

learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade. *International Journal of Technology and Design Education 26*, 1 (2016), 3–20.

[68] Tomlinson, H.B. Developmentally Appropriate Practice in the Kindergarten Year—Ages 5-6. In C. Copple and S. Bredekamp, eds., *Developmentally Appropriate Practice in Early Childhood Programs Serving Children from Birth through Age 8*. National Association for the Education of Young Children, Washington, DC, USA, 2009, 187–216.

[69] Vossoughi, S. and Bevan, B. *Making and Tinkering: A Review of the Literature*. 2014.

[70] Wyeth, P. How Young Children Learn to Program With Sensor, Action, and Logic Blocks. *Journal of the Learning Sciences 17*, 4 (2008), 517–550.

[71] Wyeth, P. and Wyeth, G.F. Electronic blocks: Tangible programming elements for preschoolers. *IFIP TC. 13 International Conference on Human-Computer Interaction*, (2001), 496–503.

[72] Zuckerman, O., Arida, S., and Resnick, M. Extending Tangible Interfaces for Education: Digital Montessori-inspired Manipulatives. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '05*, ACM Press (2005), 859–868.

[73] Google Blockly - A library for building visual programming editors. https://developers.google.com/blockly/.