IncluSim: An Accessible Educational Electronic Circuit Simulator for Blind and Low-Vision Learners

Aya Mouallem
Electrical Engineering
Stanford University
Stanford, California, USA
mouallem@stanford.edu

Mirelys Mendez Pons
Design
Stanford University
Stanford, California, USA
mirelys@stanford.edu

Ali Malik
Computer Science
Stanford University
Stanford, California, USA
malikali@stanford.edu

Trini Rogando
Design
Stanford University
Stanford, California, USA
irogando@stanford.edu

Gene S-H Kim
Symbolic Systems
Stanford University
Stanford, California, USA
gkim248@stanford.edu

Trisha Kulkarni
Computer Science
Stanford University
Stanford, California, USA
trishak8@stanford.edu

Charlene Chong
Office of Accessible Education
Stanford University
Stanford, California, USA
charlene.chong@stanford.edu

Danyang Fan
Mechanical Engineering
Stanford University
Stanford, California, USA
danfan17@stanford.edu

Shloke Nirav Patel
Stanford University
Stanford, California, USA
shlokep8@stanford.edu

Lauren Aquino Shluzas
Mechanical Engineering
Stanford University
Stanford, California, USA
larenino@stanford.edu

Helen L. Chen
Center for Design Research
Stanford University
Stanford, California, USA
hlchen@stanford.edu

Sheri D. Sheppard
Mechanical Engineering
Stanford University
Stanford, California, USA
sheppard@stanford.edu

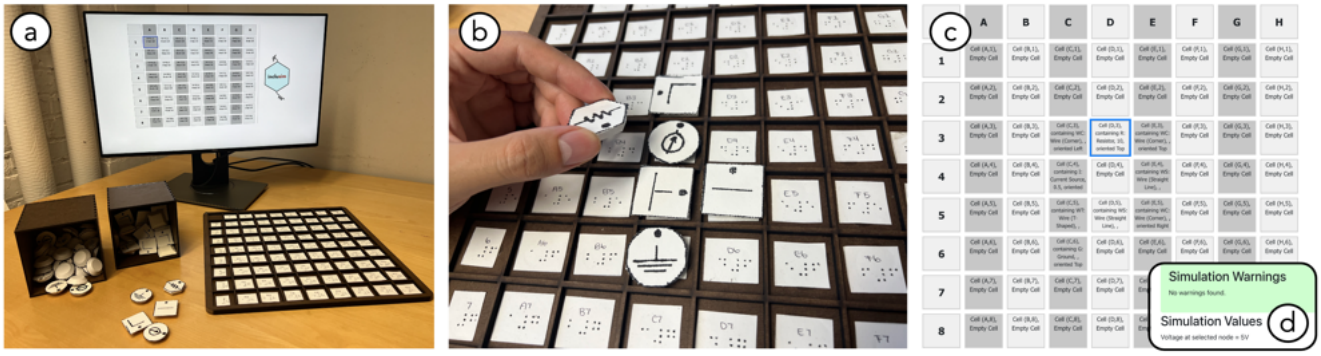


Figure 1: Overview of the IncluSim tool. (a) The hardware tool components (tactile tiles and tactile grid board) and the digital tool component (computer app). (b) A resistor tile being added to a circuit design on the hardware grid board. (c) The circuit design presented in the digital grid. (d) A sample of the simulation outputs, including a simulated voltage value.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CHI '25, Yokohama, Japan

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-1394-1/25/04
https://doi.org/10.1145/3706598.3713437

Abstract

Electronic circuit simulation is a core skill for electronics-based education. However, conventional introductory simulators often rely on visual tasks and features and are inherently inaccessible to learners who are blind or have low vision (BLV). In this work, we present IncluSim, a novel, open-source BLV-accessible circuit simulator tool, incorporating tactile elements and a digital interface. We relied on extensive needfinding to identify barriers faced by BLV learners in electronics-based education. Next, with the larger

BLV community, and as a team of BLV and sighted researchers, we adopted the co-design method over 2.5 years to design and develop the simulator tool. Over two studies, BLV participants completed different circuit design and simulation tasks using the IncluSim tool. Our findings indicate that IncluSim, via its hardware-digital medium, enables BLV learners to successfully design, simulate and debug circuits, overcoming the accessibility barriers of conventional simulators.

CCS Concepts

• **Human-centered computing → Accessibility systems and tools**.

Keywords

Accessibility, Circuit simulation, Co-Design, Electronics, Engineering Education, Tangible user interface

ACM Reference Format:

Aya Mouallem, Mirelys Mendez Pons, Ali Malik, Trini Rogando, Gene S-H Kim, Trisha Kulkarni, Charlene Chong, Danyang Fan, Shloke Nirav Patel, Lauren Aquino Shluzas, Helen L. Chen, and Sheri D. Sheppard. 2025. IncluSim: An Accessible Educational Electronic Circuit Simulator for Blind and Low-Vision Learners. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3706598.3713437

1 Introduction

Electronics are becoming increasingly ubiquitous in Science, Technology, Engineering, and Math (STEM) education, particularly within movements focused on hands-on making, computing education, and do-it-yourself (DIY) technology [29][48][24]. Learners from K-12 to higher education are being introduced to electronics in the form of e-textiles, paper electronics, and hardware prototyping platforms such as Arduino and Raspberry Pi [31][63][54]. Performing simulations, especially in higher education, is a core skill pertaining to learning electronic circuit design. It allows users to explore the behavior of a circuit design theoretically and rapidly, without any manual assembly of electronic components, before committing to a hardware implementation [67]. Most introductory electronic circuit simulators for educational purposes rely extensively on visual content, such as drag-and-drop cursor functions to place electronic components, drawing freeform digital wires to connect circuit elements, and presenting simulation outputs and results as numerical labels at different nodes in the spatially laid-out digital circuit schematic. As a result, such simulators inherently exclude students who are blind or have low-vision (BLV) from learning and adopting simulation practices in their STEM learning experiences.

In this work, we present IncluSim, a novel BLV-accessible circuit simulator tool which incorporates tactile hardware elements and a digital interface. IncluSim enables a grid-based, modular circuit design process, using tactile hardware tiles that are braille-labeled, in addition to a similar grid-based, digital interface where a learner using the tool can input their hardware design, define component values, set up simulation profiles, and explore accessible simulation results. IncluSim handles content generation and conversion to run simulations in the backend using LTSpice [11], a well-known, widely adopted, and free circuit simulator in engineering education, while maintaining accessible data presentation via the IncluSim interface to the BLV user.

Needfinding: IncluSim is an open-source software tool, designed to be used with low-cost hardware components that can be replicated by educators. Our tool design was informed by needfinding studies to identify barriers faced in electronics education by BLV learners. This included an autoethnography recently conducted with the first blind learner to complete an introductory electrical engineering course at a large academic institution in the United States.

Preliminary Prototype Development and RAD 1.0 Study (Co-Design): Based on learnings from the needfinding efforts, our research team, consisting of sighted and BLV researchers, ideated and built an accessible, preliminary prototype of the simulator tool. This preliminary prototype was then used in the first of two design and evaluation sessions with BLV community members, who participated as co-designers; this first session was called Research on Accessible Design (RAD) 1.0 and was held over two weekend sessions in December 2023. During RAD 1.0, the co-designers redirected the research agenda to add and modify tool features, which they believed were core to an accessible simulator tool.

RAD 2.0 Study (User Evaluations and Co-Design Reflection): The research team implemented the requested features, and BLV co-designers were invited back to a follow-up study, RAD 2.0. During this study, the co-designers evaluated the implementation of their design decisions and recommendations, and they further refined the design agenda.

For both studies, RAD 1.0 and RAD 2.0, the research team created multiple tasks that BLV participants had to complete using IncluSim. These tasks, covered later in this work, aimed to gauge the effectiveness of IncluSim in supporting users in achieving certain circuit design tasks using specific tool features.

Research Questions: This research was guided by the three following questions.

(1) Does IncluSim enable the BLV users to design and simulate a circuit according to given requirements?
(2) Does the tool provide BLV users with an appropriate medium to successfully complete the debugging process?
(3) What impact does the hardware-digital interface of IncluSim have on the tool experience of the BLV users?

Based on the results of the RAD studies, BLV participants were able to successfully design and simulate circuits using IncluSim for a variety of activities. They also debugged different errors with their designs, and they leveraged the hardware-digital tool interface to build intuition and verify electrical engineering theories. Additionally, through co-design, the participants refined the tool hardware design to become more inclusive and accessible.

Consequently, our main contributions in this paper are: (1) an understanding of the barriers faced by BLV learners in electronics education, and particularly pertaining to electronic circuit simulation, (2) a demonstrated approach for addressing these issues by co-designing an open-source, accessible educational circuit simulator tool for BLV learners that is low-cost and replicable by educators, and (3) analyses of the results of both studies, demonstrating the tool's ability to afford an accessible experience for the BLV users

in successfully achieving intended circuit design tasks, addressing our aforementioned research questions.

2 Related Work

2.1 Makers, Learners, and Students Developing Electronics and Simulation Skill Sets

Electronics knowledge is core to expanding technical skill sets among many audiences. Some electronics kits prioritize democratizing access to hardware, giving everyone the opportunity to design and tinker with electronics to build personal devices, autonomous gadgets, and enable further STEM exploration, such as science experiments [42][59][2]. K-12 learners can use LittleBits to prototype devices that beep, move, and flash in classrooms and at home [49]. Making hobbyists with an interest in wearables and fashion may incorporate the Lilypad Arduino for more engaging e-textile design [5]. As for higher education students, electronics are often introduced across STEM majors, from computer science to biomedical engineering, as they form the basis of many technological systems that enable such fields [21][16].

When designing circuits, especially physical ones, identifying errors or testing functionalities in real-time may become laborious and resource-consuming, given the many iterations that debugging the circuit (i.e., finding and resolving its errors) may take. As a result, electronic circuit simulation significantly facilitates the circuit design process, and as such, is an important pillar of electronics education. Strasnick et al. addressed the importance of simulation in aiding circuit designers at different stages of the design process [67]. First, simulation allows designers to theoretically and efficiently test their planned designs before implementing them with real hardware. Simulation results can aid in verification, serving as a correct reference of what the circuit behavior should actually look like if implemented correctly. Additionally, it allows for an efficient exploration of design alternatives. The designer could vary specific values or parameters digitally and simulate for their results.

From an educational perspective, the use of simulation in these different scenarios can help the learner build intuition about certain electronic theories, and the former serves as a low-cost testing process. Moreover, simulators can be used to analyze the behavior of circuit designs that incorporate non-linear characteristics or are modeled by complex formulas, as those would take significantly longer for students to solve via manual calculations [12]. Today, educators and learners can access a suite of paid and open-source simulation and debugging tools to aid students in their circuit design learning process, including platforms such as Toastboard, Fritzing, Tinkercad, and SPICE [14][34][28][12], in addition to open-source research projects such as VirtualWire, SchemaBoard, and VirtualComponent [32][33][39].

2.2 Common Student Experiences with Debugging and Troubleshooting Using Simulators

Electronic circuit simulation is a medium of interchange between theory and practice, and simulation educational tools allow users to design circuits from scratch, build intuition, verify designs, analyze complex circuit states, and troubleshoot or debug [50]. That being said, students still face universal challenges when attempting to become proficient in simulation. Fritz et al. studied students' troubleshooting processes to understand how they structured and made their debugging decisions [18]. Among roughly 100 undergraduate students who participated in the study using an online circuit simulation tool, 30 students were able to correctly diagnose a circuit with a "difficult error" as not working. However, only 10 students were able to trace back the error and isolate its source on the given circuit schematic. In fact, this error isolation skill in introductory simulation activities is commonly challenging to students. From an affective learning perspective, Mouallem et al. noted that roughly half of the 42 participants in their undergraduate electrical engineering course assessment using the CARE methodology associated debugging using a simulator with terms such as "frustrating," "struggled with patience," and "too time consuming" [44]. Some students found it "rewarding" to avoid debugging altogether, and others were "proud" to design a correctly functional circuit on the first try. The authors recommended introducing the simulator earlier in the course to provide students with ample opportunities to practice debugging using simulator feedback and to understand the importance of the process.

2.3 (In)accessibility of Electronics Education to BLV Learners

While gaining increasing importance, electronics-based education remains inherently inaccessible to BLV learners. There is limited documentation of BLV community experiences in electronics-based education. In his autoethnographic work in 2019, Seo highlighted inaccessible instructions for electronics toolkits, less tangible designs of breadboards, and the lack of multi-sensory modules as barriers to the BLV community with regards to accessible making using electronics [64]. Kulkarni et al. later detailed the reliance of electrical and computer engineering education on visual processes, such as graphical representations of circuit schematics, spatial labeling of circuits, and soldering in the lab [37]. This created inequitable difficulties for Kulkarni as a blind learner, in comparison to her sighted peers in the same electrical engineering course, in the areas of conceptual learning and tooling, translating electronic theories into practice, and learning methods and approaches.

Several research efforts have attempted to address and overcome some of these barriers, and we categorize these efforts into workshops, tools, community knowledge, and academic practices.

2.3.1 Workshops: Bennett et al. led a series of workshops guiding BLV participants through assembling an accessible voltmeter, and they reflected on good practices pertaining to the use of fragile hardware and audio outputs in accessible making using electronics [3]. Race et al. organized and led a three-day nonvisual soldering workshop for BLV participants, during which the latter learned to operate hot irons and succeeded at implementing basic soldering techniques [57]. Tactile and textual instruction helped supplement the learning process, in addition to instruction pacing based on the participants' progress.

2.3.2 Tools: He et al. proposed A11yBits, a toolkit for BLV users that is composed of "plug-and-play" electronic modules that incorporate sensing and multimodal feedback circuitry. These modules

could be assembled to create functional, personalized technology circuits [22]. Davis et al. developed TangibleCircuits, an interactive tool that 3D-prints circuit models and integrates audio feedback. TangibleCircuits enabled better geometric, spatial, and structural circuit understanding among BLV users [10]. Chang et al. designed AccessibleCircuits, which consists of 3D-printable, low-cost components, integrating braille labels and audio feedback, that add onto existing breadboards and circuit elements [8]. Pender and Healy introduced NetlistToText, a software that generates textual description of the connections of circuit components to BLV learners [51]. Finally, Race et al. created, via iterative design activities with tactile graphics experts and BLV participants, an improved set of tactile schematic symbols to represent electronic components [55]. Common features across these tools include tangible user interfaces, multimodal feedback (such as audio outputs), and text-based or tactile description when possible.

2.3.3 Community Knowledge: Three communities, from in-person to blog-based, have sourced tips and lived experiences from BLV makers and electronics-hobbyists to aggregate resources on accessible circuit prototyping and electronics toolkits. The Blind Arduino project was founded by Joshua Miele and aimed to share tutorials and resources in person and online, via a blog, with BLV makers using the Arduino hardware [27][43][56]. The Raspberry VI blog similarly provided resources on setting up assistive technology for accessible development using the Raspberry Pi hardware [58]. As for Blind Electronics, this blog spanned different platforms, such as Snap Circuits and LittleBits, and offered relevant advice and tutorials for accessible making using such hardware [52]. These three blogs have been inactive for the past few years but continue to serve as online references for BLV-accessible electronics and making practices.

2.3.4 Academic Practices: Mouallem et al. adopted the Challenging And Rewarding Experiences (CARE) assessment methodology to evaluate BLV and sighted student experiences in an introductory electrical engineering course [44][45]. They proposed pedagogical and institutional practices to better support BLV learners in introductory electrical engineering contexts, such as incorporating alternative presentation formats, using universal labels, testing lab tools for accessibility ahead of time, and proactive coordination among different stakeholders involved in the BLV student's learning experience. The AccessEngineering project at the University of Washington offered guidance on creating more inclusive making and engineering education spaces, including electronics education, for learners with disabilities, among whom are BLV learners [65].

Despite the growing and promising efforts to improve the accessibility of electronics-based education, there remains a major gap in the literature, as no work has addressed the inaccessibility of electronic circuit simulation to BLV learners. As a result, our current work explores this gap by developing and evaluating the first BLV-accessible electronic circuit simulator tool.

3 Understanding the Challenges Faced by BLV Learners in Electronic Circuit Simulation

3.1 A Heavily Visual Electronic Circuit Simulation Process

The following illustrates a typical step-by-step electronic simulation process used by a sighted student in an introductory electrical engineering course at our academic institution and observed by researchers on our team. To emphasize the reliance of simulation on vision, we use asterisks to note phrases representing visual characteristics of the simulation process in the following steps.

(1) The student launches the simulator application. A blank 2D design space is displayed.
(2) The student uses their cursor to **drag and drop*** components from the component menu onto the design space. The circuit design can **spread spatially*** in all directions.
(3) The student uses the cursor to draw wires by **tracing paths spatially*** between the terminals of different components to set up the circuit connections.
(4) The student **moves their cursor*** to select a component (e.g., can press Enter or right click on a component) to modify its characteristics. For example, the student can set a resistance value for a resistor.
(5) The student uses their cursor to place a "probe" pin on the circuit schematic, **at the exact location*** where they want to simulate for a value. For example, they can place a probe at a specific point on a wire to measure the current flowing through it.
(6) Depending on the simulation type, the simulation results will be displayed differently. A DC operating point simulation will result in numerical values **printed at specific points*** on the circuit schematic, where the probe was placed. An AC analysis simulation would result in a **graphical display*** of color-coded waveforms, representing frequency-varying results.

3.2 The Inaccessibility of Electronic Circuit Simulators to BLV Learners

There is very limited literature that explores the experiences of BLV learners with electronic circuit simulation. Kulkarni, reflecting in her autoethnography on being the first blind student in her introductory electrical engineering course, simply could not use the simulation software adopted in the course [37]. The software was fully inaccessible to her, as it relied on drawing circuits digitally using a cursor, as outlined in **Section 3.1**. Additionally, when her debugging experiences were compared to those of sighted learners in the same course, her experiences were significantly impacted by inequity given the inaccessibility of the debugging process. On the other hand, experiences of sighted learners were described as simultaneously challenging and rewarding [45]. Moreover, Kulkarni often had to wait for significant time periods for upcoming office hours to receive visual feedback from the teaching team on the outputs of her physical circuit designs, given that digital simulation was not an option [37].

Exploring the step-by-step simulation process described previously, multiple barriers to BLV learners could be identified. The use

of the cursor for placing components, drawing wires, connecting elements, and placing simulation probes is inherently not accessible to a BLV learner using a keyboard and a digital screen reader. The expansion of a circuit design spatially in different directions makes it harder to explore it systematically, the way a screen reader navigates items on a page based in a defined order. Sighted learners often use the 2D digital design space in the simulator interface for cognitive offloading, as it is convenient for them to review their whole design efficiently. A BLV learner would have to go through the circuit again, with no defined component order, to remember the circuit composition. This may create a mental burden, especially when the circuit includes tens of elements connected in different parallel and series configurations. As for the simulation results, numbers may be printed on the screen at different locations, and a screen reader may not be able to identify any characteristics of the location that could explain the numerical value. Graphical outputs are currently color-coded and do not incorporate multimodal feedback (such as sonification). Thus, a screen reader may not be able to describe such outputs either.

Consequently, a BLV learner attempting to use a conventional electronic simulator would experience inequitable barriers that make such a tool inaccessible. These challenges are purely accessibility-centric and supersede any conceptual or learning challenges that sighted students face when learning the simulation process. Therefore, as a team of BLV and sighted researchers, we decided to co-design with the larger BLV community an accessible electronic simulator to address these barriers. Our work focuses on designing a simulator for introductory electronics education. While more advanced simulation practices involve netlisting, which is text-based description of circuit designs, such processes are often introduced in more advanced electronics courses, not in introductory ones. Kulkarni et al. argue that netlisting, while accessible, is an inequitable learning burden to introductory electronics BLV learners [37].

4 The Research on Accessible Design (RAD) Program

The Research on Accessible Design (RAD) program was launched at our academic institution to promote synergistic collaborations between researchers, designers, and BLV community members, and exchange expertise and lived experiences pertaining to inclusive electrical engineering education, technology design, and accessibility. Our core research team consisted of blind and sighted researchers and an accessibility manager at the disability services office at our academic institution. Our research focus areas spanned accessibility, engineering education, human-centered design, and human-computer interaction. Through a community partnership, we engaged participants from the greater BLV community as co-designers throughout the design process of our project. The community participants had different levels of prior exposure to electronics, simulation, and design thinking and vastly diverse lived experiences with disability. The RAD program had three goals: (1) co-design iteratively, with the BLV community, an accessible hardware-digital electronic circuit simulator tool, prioritizing emerging inclusive features and preferences, and supporting features in prior literature with further evidence, (2) evaluate the effectiveness of the co-designed simulation tool in enabling the successful completion of circuit design tasks, and (3) seek evidence to answer the research questions presented in **Section 1**.

4.1 Co-Designing through a Community Partnership

The RAD program adopts co-design as a primary method. Co-design elevates the role of participants (in our case, BLV participants) to active co-designers and co-researchers, with the power to influence the research directions, agenda, and priorities [66][73][6]. This allows the tool design and features to emerge directly from and be continuously shaped by the lived experiences of the BLV community. As a result, the co-design process brings together researchers and participants as collaborators, having equally critical research influence. We established a community partnership with LightHouse for the Blind and Visually Impaired, which is a leading non-profit organization serving the BLV community in the San Francisco Bay area. Through a series of meetings over six months with the non-profit team leads, we discussed shared values, goals, and expectations, culminating in a written memorandum of understanding.

4.2 Design Frameworks

We relied on specific frameworks in guiding the development and implementation of specific tool features. These frameworks include ability-based design, Universal Design for Learning (UDL), the Web Content Accessibility Guidelines (WCAG), and design criteria for accessible electrical engineering education technology tools for BLV learners, proposed recently by Kulkarni et al [72][62][7][37]. We point out specific principles adopted from each framework in **Section 7.3**.

4.3 Design Timeline

Throughout RAD, two tool prototypes were designed: the preliminary simulator prototype and the IncluSim tool. We discuss each further in later sections. Our entire design timeline spanned 2.5 years, as follows. *January 2022 – November 2022:* We conducted and published needfinding efforts, including observations, brief surveys, and informal conversations with BLV students in STEM fields, their educators, and the staff at the disability services office at our academic institution.

November 2022 – May 2023: Two blind students joined our group as undergraduate researchers, ideating and brainstorming in relation to the identified needs from the needfinding stage.

May 2023 – November 2023: Blind and sighted researchers prototyped low-fidelity designs with preliminary material. Two sighted researchers then implemented the remaining design features, relying on makerspace tools such as laser cutters and 3D printers. This stage resulted in the development of the preliminary simulator prototype.

November 2023 – March 2024: BLV participants joined our efforts, via the community partnership, as co-designers. The research team facilitated RAD 1.0, which we discuss further in **Section 5**. The participants explored the preliminary simulator prototype design and made decisions regarding modifying and adding features.

March 2024 – Aug 2024: A part of the team worked on data analysis after RAD 1.0, developing findings to inform the second

study and the tool design. The other part implemented further modifications to the tool design, resulting in the IncluSim tool.

Aug 2024 – September 2024: The research team led RAD 2.0, inviting the co-designers for virtual and in-person programming focusing on evaluation and reflection. The takeaways are discussed in **Section 6**.

5 The RAD Studies

The RAD program consisted of two studies: RAD 1.0 and RAD 2.0. The studies were approved by the Institutional Review Board (IRB) Committee on Human Subjects Research at our academic institution, and participants were fairly compensated for their involvement. A summary of the two studies is presented in **Table 1**. Of note in this summary are areas of overlap and similarity in the planned tasks, procedures, and methods adopted for both studies. As a result, for brevity and clarity, we discuss RAD 1.0 and RAD 2.0 in conjunction in the following sections, noting any distinctive elements when available.

The RAD 1.0 study ran for 3.5 hours twice, on consecutive weekends, with four participants joining each iteration in December 2023. Most participants had no prior experience with electronics and design thinking. Seven researchers organized and co-led the program, and four BLV guest panelists were invited to share their experiences as students, educators, and STEM professionals. RAD 1.0 consisted of individual simulation activities, an introduction to design thinking, a panel discussion over a meal, and a collective co-design reflection session. During RAD 1.0 programming, BLV participants explored a preliminary simulator prototype and made several design decisions and changes to influence upcoming iterations, further discussed in **Section 6.5**.

Almost a year after RAD 1.0, we organized RAD 2.0 in August 2024, inviting back the co-designers for another design iteration, resulting in an improved simulator tool, called IncluSim. We ran 2-hour virtual onboarding sessions, using the new digital tool tutorial, during the week leading up to the in-person RAD 2.0 programming. We hosted the latter on campus for 3.5 hours during the weekend. Four researchers co-led the program, which consisted of individual and collaborative simulation activities. Over lunch, the researchers facilitated a co-design reflection workshop.

5.1 Evolution of the Apparatus Design

5.1.1 Preliminary Simulator Tool Prototype: An initial simulator tool prototype was brainstormed and designed by blind and sighted researchers on our team, prior to the RAD 1.0 study. Factoring in the barriers faced by BLV learners with simulation tools, as identified earlier in this work, we designed the preliminary tool in line with the design frameworks highlighted in **Section 4.2**. This prototype was meant to serve as a "scrappy, rough foundation" for the BLV participants. It was an initial apparatus or starting point, which BLV participants could build upon in the subsequent RAD studies. It was made clear to the participants that this tool was not in any adoptable shape yet and that it needed their co-design insight for significant improvements.

A blind researcher and a sighted researcher commenced the prototyping process by manipulating physical prototypes, relying on cardboard, acrylic, and 3D printing to model brainstormed ideas



Figure 2: Overview of the preliminary prototype hardware components. (a) The reference guide. (b) The tactile blocks with braille embossing. (c) The hardware grid board.

for the hardware components of the simulator tool. Two sighted researchers led the hardware manufacturing efforts at a makerspace on campus, utilizing 3D printing, laser cutting, and soldering irons to put together the hardware resources needed. These include the 3D printed blocks, the hardware grid, and the electronic circuits to support audio feedback via buttons. One researcher worked closely with the accessibility manager at the disability services office to create, design, validate, and emboss braille files. In parallel, two researchers built the accessible digital simulator interface as a ReactJS app, applying WCAG principles. Two blind researchers on the team evaluated the app at different stages of its development, in addition to sighted researchers completing routine accessibility checks. This prototyping stage took seven months to complete, with a total material cost of $247.

This preliminary simulator tool was provided to participants at the start of the RAD 1.0 study. The prototype relied on hardware and digital components to enable a complete circuit simulation process. We describe each of these components below, and the full tool set-up is displayed in **Figure 2**.

A set of tactile blocks: Each block represents an electronic component and has a tactile representation of its schematic symbol (a drawn symbol to represent the electronic component), letter representation in braille, and a tactile dot serving as an orientation marker, to identify the orientation the component is placed with. An example is shown in **Figure 3**. There were blocks representing 15 different component types, including power sources, wires of different shapes, and resistors.



Figure 3: A block representing the resistor component, with its tactile schematic symbol, braille label, and an orientation marker.

Table 1: Description of the RAD studies, including participant groups, study plans, and research methods used.

Overview	RAD 1.0 (2023)	RAD 2.0 (2024)
Participants	• Eight participants, all older than 18, five men, three women • Six participants were blind, two had low vision • Participants had diverse experiences with disability (born blind, progressive vision loss)	• Three participants, all older than 18, two men, one woman • Two participants had low vision, one was blind • Participants were invited back after RAD 1.0 and were available to join
Study Duration	• 3.5 hours of in-person activities • 30 minutes of break time	• 2 hours of virtual activities • 3.5 hours of in-person activities • 30 minutes of break time
Apparatus Used	• Preliminary prototype designed by the BLV and sighted researchers • Served as foundations for the co-design process with the BLV community	• IncluSim, the improved tool design based on design decisions made by the BLV community • Design changes implemented by the research team
Pre-Study Assignments	• Review the electronics conceptual refresher • Install the digital simulator app on personal laptops • Complete the simulator tutorial (only for RAD 2.0)	
Tasks During Study	• Design a circuit according to given requirements • Compare theoretical calculations and simulation results • Reflect on the co-design process • Debug a given circuit design • Vary parameter values in a given circuit design and build intuition about trends (mainly during RAD 2.0)	
Data Collection Methods	• Surveys • Multimedia recordings (video, photos, audio) • Artifacts (digital activity workbooks) • Supplementary notes by researchers	
Data Analysis Methods	• Grounded theory building approach – for survey data analysis • Visual-Verbal Video Analysis (VVVA) method [15] – adopted fully with RAD 1.0 data, and partially with RAD 2.0 data	

A reference guide: This resource acts as a component dictionary, providing additional information on each component type, such as the full name of the component in braille, its tactile schematic symbol, its braille letter symbol, and an audio button that plays the recording of the name of the component.

Hardware grid board: This is a square grid with raised walls and 64 cells. The user can place the blocks conveniently in the cells of this grid to set up a specific circuit design. The grid has braille labels and audio buttons reading out loud the labels of the columns and rows.

Digital simulator interface: This is a packaged desktop app that displays an identical digital replica of the hardware grid. After setting up their circuit design using the tactile blocks on the hardware grid, the user can input their design into the digital grid by

navigating the grid and defining the content of the cells (component type, orientation, and value, if applicable). The user then completes the simulation profile, selecting the coordinates of the cell(s) across which they would like to find specific simulation values. The simulator interface then translates this grid-based circuit set-up into a netlist circuit file that can be simulated by LTSpice. After LTSpice simulates the netlist file, the simulator interface displays the simulation results textually in an accessible format to the BLV user. The interface also collects and displays warning and error messages, if any. The user can navigate the app using their screen reader, via keyboard Arrow keys, the Enter key, or the Space bar, in addition to a shortcut S key to call the simulation profile.

5.1.2 IncluSim: The Improved Simulator Tool: Shortly after RAD 1.0 wrapped up and until we launched RAD 2.0, our research team committed to implementing the design changes brought up by the BLV co-designers during RAD 1.0, resulting in an improved design of the accessible simulator tool, which BLV co-designers and researchers collaboratively named IncluSim. **Figure 1** provides an overview of the IncluSim hardware and digital tool design. While we provide relevant evidence and discuss the design changes prioritized by BLV co-designers in **Section 6.5**, we highlight new and improved features that were integrated in IncluSim below.

A comprehensive tutorial: We designed and integrated an in-depth tutorial that covers a step-by-step activity during which the learner designs a circuit, debugs an error, and verifies a basic electronics theory with the simulated results. Throughout the activity, the tutorial highlights the main tool features, including simulation functions and keyboard shortcuts. It also offers detailed insight on how the learner can use the digital simulator interface with a screen reader. **Figure 4** displays a sample tutorial message guiding the learner to define a component.

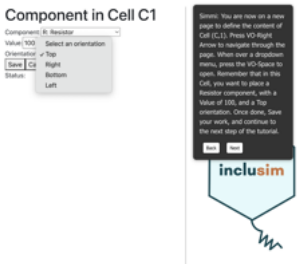


Figure 4: The IncluSim tutorial guiding the learner to place a resistor component in cell C1 of the grid.

Improved software for efficiency and accessibility: The research team integrated several software improvements to address minor issues that came up in the BLV co-designers' experiences during RAD 1.0. For instance, the digital simulator now has many shortcuts to save time and effort in quitting pages, simulating, and toggling between the tutorial and the main grid page. We added advice and hints on how simulation results can be interpreted. We also implemented a queue-based cell update system, so that a change that the user makes to a cell in the grid-based circuit design only impacts neighboring, connected components.

Reduced hardware size: We re-designed the hardware grid to a quarter of the initial grid size. Additionally, we significantly modified the hardware blocks. We replaced the bulky blocks with thumb-sized 3D-printed tiles. We introduced different tile shapes for functional groups of components, such as square tiles for wires and round tiles for power sources and ground nodes. We also removed the pocket fingers that were meant to facilitate carrying the blocks, as some participants noted that they interrupted their tracing of a hardware circuit path.

Color contrast: Finally, we added color contrast to all our hardware resources. **Figure 5** compares the new tile design with the old block design.



Figure 5: A user holds the old block design (2 inches x 2 inches x 2 inches) to the left and the new tile design (1.25 inches x 1.25 inches x 0.3 inches) to the right, both representing the resistor component.

Similar to our prior prototyping efforts, we relied on laser cutting to fabricate our hardware grid and on 3D printing to create our tiles at a makerspace on campus. We continued to collaborate with the accessibility manager to modify our tactile designs and braille-emboss them. **Appendix E.1** includes a pointer to the open-source, do-it-yourself design guide for the IncluSim hardware design, including the design processes and files used. This prototyping stage took three months to complete, with a total material cost of $8.50, a 29 times decrease in cost from that of the prior design. This change was mostly due to eliminating the electronics used for audio feedback for the hardware components and relying on PC-generated audio feedback instead. The IncluSim digital interface is available for free use and download, and more details are included in **Appendix E.2**, including access to the open-source code.

5.2 Tasks Assigned to Participants

5.2.1 Circuit Design and Verification: For this task area, we first presented participants with their hardware tool components (grid board, tactile blocks, tactile reference guide) and the digital simulator interface. During RAD 1.0, the participants spent some time exploring the tool components then began the planned activities in their workbooks. For RAD 2.0, the participants completed a digital tutorial prior to the in-person program. During both studies, the in-person simulation activities included tasks, listed in **Appendix A**, that touched upon the following objectives: designing a circuit according to given requirements, linking theory and practice, debugging, and verification. Additional RAD 2.0 tasks, listed in the same Appendix, aimed to support participants in building intuition

and in applying their use of the simulator tool to a real-life debugging scenario. A researcher was present with each participant during these individual activities. Researchers answered conceptual questions (e.g., "Can you remind me of whether a resistor is polarized?"), but they refrained from giving out next steps. In addition, researchers asked participants to think aloud to better trace their decision processes.

5.2.2 Design Thinking and Co-Design: During RAD 1.0, participants completed a short session led by blind researchers on the design thinking principles, connecting the latter to everyday life experiences. During both studies, participants joined a one-hour collective co-design reflection session, during which they worked with guided prompts, facilitated by researchers, while drawing on design thinking principles. The prompts are listed in **Appendix B**, and they provided the participants with structure to evaluate existing design features, propose new improvements, and reflect on their design experiences, without impacting their power and influence as co-designers.

5.3 Study Procedure

Participants were not expected to have extensive background knowledge in electronics or design thinking. Therefore, we sent out a refresher document prior to RAD 1.0 and RAD 2.0, covering electrical engineering concepts and technical terminology. Throughout both studies, participants used their digital activity workbooks to read activity steps and reflection prompts, and to record answers and reflections. Additionally, we utilized final surveys after both studies to collect data on the participants' experiences. Below, we note minor differences in the scheduled programming for each study.

5.3.1 RAD 1.0 Procedure. In the RAD 1.0 participation confirmation survey, we asked participants basic questions about the refresher document content to verify that they had reviewed it. **Appendix C** includes some of these questions. The program started with an orientation tour of the space and a round of introductions. Then, we split up the participants into two groups, alternating their completion of the simulation activities and the introductory design thinking activities. After that, we welcomed guest panelists over lunch. Finally, we led the co-design reflection session.

5.3.2 RAD 2.0 Procedure. During the virtual onboarding sessions of RAD 2.0, the researcher(s) presented the integrated tool design updates to the participant, refreshed their memory on electrical engineering concepts, and discussed dissemination strategies for the resulting tool design. One hour was dedicated to completing the new digital tutorial, with participant completion time ranging suitably from 38 minutes to an hour. The virtual onboarding session served as a checkpoint to ensure that participants were joining the RAD 2.0 in-person programming with a solid understanding of the needed material. During in-person programming, participants individually completed one hour and a quarter of circuit design and debugging activities. After that, they completed a one-hour collaborative activity, debugging an electric heater circuit to fix its problems and reduce the electric bill. Finally, the research team facilitated a co-design reflection session over lunch.

5.4 Methods

5.4.1 Data Collection Methods.

Surveys: Prior to RAD 1.0, the participants completed the onboarding survey and the consent form, read through the needed electronics material, and installed the digital simulator. After RAD 1.0 and RAD 2.0, we deployed post-program final surveys to collect quantitative and qualitative data focusing on the following aspects of the users' experiences: interactions with the tool, design thinking and co-design processes, learning outcomes, and overall programming feedback. A sample of the final survey questions of RAD 1.0 and RAD 2.0 are provided in **Appendix D**.

Multimedia: During RAD 1.0 and RAD 2.0, we recorded videos of the participants' interactions with the hardware and digital components of the simulator using GoPro cameras and tripods. We also captured photos of the users' final circuit designs. We audio-recorded the reflection sessions, during which the participants engaged in co-design activities to provide feedback and define the next research agenda.

Supplementary Data: During RAD 1.0 and RAD 2.0, we asked participants to submit their digital activity workbooks as artifacts of their participation. In addition, BLV and sighted researchers took and aggregated notes as memos from different sessions of the studies that they led or attended.

5.4.2 Data Analysis. We adopted two main analysis methods to elicit deep and extensive insight from the participants' experiences and interactions with the tool.

Survey Analysis: After collecting data from each study (RAD 1.0 and RAD 2.0), we adopted the grounded theory building approach, relying on inductive open, axial, and selective coding in analyzing the survey data [9]. The grounded theory building approach gave rise to themes and findings directly from the raw data, amplifying the importance and weight of the co-designers' ideas and contributions in defining design features and decisions [68][46]. The identified codes were clustered under main high-level themes, including: accessibility, learning curve, hardware-digital interaction, and recommendations.

Multimedia Analysis: We relied on the Visual-Verbal Video Analysis (VVVA) method to analyze in depth the experiences of participants with the simulator tool via the video recordings [15]. The VVVA method is a systemic coding approach for verbal, visual, and textual content, based on the visual grounded theory and multimodality theory, and it entails six steps to organize and evaluate video material across different dimensions [35][36]. In our analysis, these dimensions offered extensive insight on the multimodal types of user interaction with the hardware and software components of the tool, communication and discourses with researchers and collaborating participants, in addition to compositional characteristics of the user experience with the accessibility features and circuit design tasks.

5.4.3 Improving the Research Validity. As we relied primarily on qualitative data to inform our analysis, we adopted several practices to bolster our research validity and quality [30][69]. We incorporated *investigator triangulation* by involving a team of researchers

of different areas of expertise and abilities in the study design, implementation, and data analysis. Additionally, we engaged the BLV community members as co-designers who influenced and evaluated the research agenda and outcomes. To adopt *data triangulation*, we sought feedback from multiple BLV community members at a time, at different stages, to draw a more thorough understanding of the barriers faced and the features prioritized by the BLV community. In addition, we utilized *methods triangulation* to diversify our data collection methods, relying on surveys with quantitative and qualitative questions, video and audio recordings, photos, and researchers' notes. Moreover, we relied on *theory or perspective triangulation* to evaluate the tool against different design frameworks. This evaluation is presented in **Section 7.3**.

6 Findings

This section presents findings that emerged from the survey and video analyses of RAD 1.0 and RAD 2.0 data. **Table 2** presents insight on participants' Likert-scale evaluation of the tool's accessibility, effectiveness of hardware-digital media, and debugging experiences. Participants' evaluations improved on all three fronts from RAD 1.0 to RAD 2.0, particularly on the accessibility end, and **Section 6.5** highlights design decisions made by the BLV participants to improve the tool's accessibility features. However, due to sample size limitations (N=8 for RAD 1.0, N=3 for RAD 2.0), we relied on qualitative analysis, rather than quantitative results, to capture more thorough insight and personal experiences.

We categorized our qualitative findings to address the three areas of the research questions stated in **Section 1**: designing and simulating a circuit (**Section 6.1**), debugging (**Section 6.2**), and the impact of the hardware-digital interface (**Section 6.3**). In addition, **Section 6.5** summarizes the design decisions that contributed to the tool design evolution, as highlighted in **Section 5.1.**

6.1 Circuit Design and Simulation

Initial quantitative results from the RAD 1.0 final survey indicated that the prototype generally enabled participants to design circuits according to given requirements. All eight participants indicated that the tool allowed them to experience both, hands-on circuit design and digital simulation to a good or large extent. Moreover, seven out of eight participants agreed to a good or large extent that the tool allowed them to check, verify, and fix any errors in their work.

Grid-Based Spatial Layout: Participants noted that the grid-based spatial layout better supported their understanding, with participant P1 stating, "I felt **confident** with the act of transcribing the blocks in their position on the grid to the position on the [digital] simulator (block, column, row, orientation). It made me realize the importance of **spatial awareness** for the activity which added to my … growing **understanding** of the material."

Building Intuition: One of the new activities during RAD 2.0 asked participants to vary a resistance parameter value in order to build intuition about expected trends for the current value flowing through the resistor in a simple series circuit with a fixed voltage source. P1 reflected on this activity, sharing, "It was affirming to have the formulae, do the math and have the tool confirm or tell me otherwise. For instance, it felt very **quick and easy to test** current flow [through] resistors to quickly explore the **trend** of … increasing resistance in our demo circuit. I feel like I gained enough confidence to make me curious and want to experiment and learn from replicating or building bigger circuits." Reviewing the collected artifacts in the form of participants' activity workbooks, P6, whose tool interaction process is displayed in **Figure 6**, wrote down his answers to the questions that guided him in building intuition. He expected that "when the resistance increases, the current decreases." In a later question, he connected the simulated results to Ohm's law, writing, "Yes, [the simulation outputs agree with Ohm's law]. There's an **inverse relationship** between Resistance and Current. When the resistance increases, the current decreases." Therefore, not only did circuit design and simulation tasks support the participants in building intuition about expected trends, they were also able to verify electrical engineering theories via their simulation results.



Figure 6: P6 designs and simulates a parallel circuit according to given requirements. (a) P6 reviews the exercise in his activity workbook after setting up his hardware circuit design. (b) P6 goes through the hardware-to-digital transfer process, defining each tile on the hardware grid in the digital grid cells. (c) P6 simulates his digital circuit design.

6.2 Debugging

Detecting and Fixing Errors: According to the participants, the simulator allowed them to detect and fix errors. P5 shared that "after the simulation outcome, [they] pretty much [were] able to **identify where** the errors were and [were] able to **resolve** most of them." The recorded videos during RAD 2.0 showcased different types of errors being detected and debugged by the participants after reading their digital simulation results. For instance, P1 identified a missing ground component and an unconnected node of a power source. P3 also identified an unconnected node of a resistor, in addition to a wrongly oriented wire component which led to another unconnected node.

Multiple Approaches to Debugging: During RAD 2.0, participants debugged a broken electric circuit heater, utilizing IncluSim in a real life-like scenario. Unlike other activities where participants created the digital designs from scratch, for this activity, the participants were given the digital circuit heater design, and they loaded it into the digital IncluSim tool. Based on the video recordings, P3 attempted to explore and debug the digital design first before resorting to using the hardware tiles, while P1 and P6 preferred to directly replicate the digital design using the hardware tiles. P6 found debugging the circuit to be straightforward, identifying the

Table 2: Selected survey items from RAD 1.0 and RAD 2.0 post-program surveys. Both surveys had a 100% response rate. P1 to P8 completed the RAD 1.0 survey. P1, P3, and P6 completed the RAD 2.0 survey. Participants were asked to rate, on a 5-point Likert scale (from 1 to 5), to what extent they agreed or disagreed with the statements listed in the table, regarding their experience with the hardware-digital simulator tool.

Survey Question	RAD 1.0, N=8 (Mean; Variance)	RAD 2.0, N=3 (Mean; Variance)
This tool allowed me to experience both hands-on circuit design and digital simulation.	(4.75; 0.21)	(5.0; 0.0)
This tool was accessible for me to use as a user who is blind or has low vision.	(2.75; 1.64)	(4.33; 0.33)
This tool allowed me to check, verify, and fix any errors in my work.	(4.13; 1.84)	(5.0; 0.0)

broken wire, as he "also **traced** the circuit tactilely to make sure there were no other "broken" wires/open circuits."

Understanding the Error Causes: According to P1, not only did the tool allow for error detection, they found it "very affirming to be able to **troubleshoot** with the blocks and learn **what** I did wrong and **why**," understanding the root causes of the errors. Furthermore, the simulation process was instantaneous and efficient. P7 stated, "[The tool] enabled me to simulate the circuit, and I got **instant feedback** letting me know if all was hooked up properly."

6.3 Impact of the Hardware-Digital Interface on the Tool Use Experience

Multiple Representations for a Full Picture: Participants linked their use of both the hardware and digital tool components to a better grasp on the fundamental theories and functionality of the circuit design itself. P8 shared that "[they] had a good time using the tool. [They were] able to create a **digital** representation of [their] circuit and [were] able to **understand** the simulated circuit", and P1 reflected that "having the **physical** layout really helped with understanding the **formula** [they] had to solve." Most participants generally followed a hardware-to-digital design process and iterated as needed. P7 "began by using the tactile blocks to **physically** build [their] circuit and **then** proceeded to build the circuit [in] the **digital** simulator. [They] found it extremely useful to have both, the physical representation and the digital representation of the circuit, to get a **full picture** of the circuit that [they were] building." P2 thought that the "two media helped **reinforce** each other in making [them] understand what [they were] doing."

Media for Flexible Hardware and Digital Debugging Steps: **Figure 7** showcases the design approach of P8 as an example of the hardware-to-digital process. He started by placing the blocks on the hardware grid board, followed by exploring the features of the digital simulator, then transferred his hardware circuit design into the digital grid. After that, he simulated his design digitally, received an error message, decided to move back to the hardware grid to debug, and then inputted his updated hardware circuit design into the digital platform to receive a successful simulation result.

Cognitive Offloading: After RAD 1.0, when asked if they could rely solely on the digital simulator component, the participants

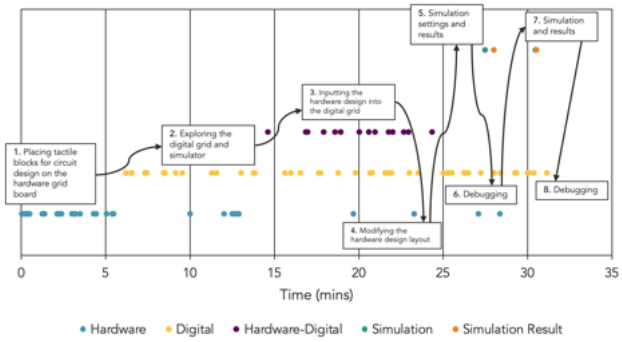


Figure 7: A plot of the hardware, digital, and hardware-digital interaction processes of P8 with the preliminary simulator prototype tool during a RAD 1.0 circuit design activity. Colored dots on the plot represent an interaction instant with the tool.

highlighted the opportunity for cognitive offloading that the tool affords, especially with larger circuit designs. P4 stated, "I think I would be able to use the simulator without the physical components, but they do provide me a way to check my work so I **wouldn't have to totally rely on my memory**." After RAD 2.0, P6 recalled the cognitive offloading feature of the hardware, stating that "without the visual representation, [simulation] would be much harder because it would entail reading and **memorizing** the board, circuit, and values."

Clearer Spatial Navigation Decisions via Hardware: P1 said she would find it hard to exclusively rely on the digital medium, sharing, "One challenge with a fully text-based [digital] circuit is knowing where to go next, especially when they get complicated such as with the heater exercise [shown in **Figure 8**], when squares next to one another got filled. **Which way** do you Arrow to correctly follow the circuit without jumping wires and getting **lost**? I think I would need a physical iteration to complement the digital arrangement in order to understand the circuit pathway more clearly." Therefore, according to P1, navigating the digital design spatially may get confusing, especially at points where multiple components are connected in different directions.



Figure 8: P1 sets up the circuit design of the malfunctioning electric heater exercise on the hardware grid during RAD 2.0. A tactile reference guide is to her left on the table.

6.4 A Case of an Iterative and Successful Debugging Process Using the Hardware-Digital Simulator

The following case demonstrates the iterative hardware-digital approach that P1 adopted using the simulator prototype to debug and fix her circuit design. This video analysis was completed using the VVVA method. One of the activities asked P1 to design a circuit using a 5 Amps current source and two resistors, all connected in parallel. The steps below break down P1's debugging process using the hardware-digital platform and highlight circuit design skills and tool features that she utilized to successfully identify and resolve the error in her circuit design.

(1) *Theoretical calculations:* P1 completed theoretical calculations first, discovering that the current through the first resistor should be 2 Amps, and the current through the second resistor should be 3 Amps. The sum of the current through these resistor branches should add up, as a result, to 5 Amps. This is a property of the parallel configuration.
(2) *Interpreting the simulation results to identify a problem:* At the 38 minute mark, the simulator results stated that the current through each resistor is 5 Amps. P1, thinking aloud, asked if the resistors "are in series [and not in parallel]? Because it's the whole 5, right?"
(3) *Hardware debugging:* Returning to the hardware grid and tracing the tactile symbols on the blocks with her finger, P1 probed around and found the resistors connected at one common node, but not another.
(4) *Detecting the error and resolving it via the hardware first:* Realizing that the resistors needed to be connected at both their nodes for a parallel configuration, P1 decided to "find another T"-shaped wire. She moved the current source block back and left an empty cell between the resistors to place the T-shaped wire block.
(5) *Digital simulation and referring to theory to verify results:* P1 attempted to simulate again, hoping that "maybe [she] can get this." Then, she reviewed the results out loud, "Okay. Two amps? Yeah, that was the first [current value]. Yay. Three [Amps for the second current value], which will make the five [total]. All right!" P1 linked the simulation results to the theoretical calculations she had done prior, and the values matched. The total debugging process took P1 around 5 minutes.

6.5 Co-Design Decisions

When the BLV co-designers were asked in discussion and via surveys whether they felt that the research team actually implemented their design decisions in the new tool design, P1 responded, "Yes! Wow!" and P3 stated, "Yes, [it] felt really great seeing the improvements based on [my] feedback, and being part of the [co-design] journey." **Table 3** showcases the evolution of the tool design in line with three core design decisions made by the participants as co-designers and the feedback that they provided throughout the process.

Design Decision 1: Tutorial Design and Pre-Program Demo: During RAD 1.0, participants used the tool for the first time during their in-person individual simulator activities. As a result, many participants struggled with figuring out how to best interact with the simulator app using their screen reader, what shortcuts to keep in mind, and how to make the most of the available app features. P2 shared that "there was a **learning curve** to know how to interact with the **grid**, how to interact with the **dialog**, where **focus** goes, what's expected, and what requires workarounds." Reflecting similarly, P8 stated that a "big help would be a Readme" and P3 preferred to have had "a **walkthrough demo** of how the simulator is expected to behave with the screen reader." As a result, in preparation for RAD 2.0, the research team introduced a tool tutorial, which participants completed during the pre-program onboarding virtual session. P3 reflected that after completing the tutorial, his "learning curve decreased,... as [he] got **familiar** with the flow." P6 shared that the tutorial "training... over Zoom helped tremendously with being **prepared** for... the in-person session," as it helped him learn the "shortcuts, tool's controls and functions, [and] interaction with the tutorial." As for P1, she highlighted a specific feature that she particularly appreciated, sharing, "Having the warnings and errors with **advice** [in the tutorial] also made a huge difference, making me feel like I was being **set up for success**. Like a good tutorial should, it prepared me ahead of time and taught me what to do, so I was ready for the in-person segment." P3 and P6 listed new keyboard commands via keyboard-only operation of the digital interface as some of their favorite tool features, especially with the additional shortcuts, all introduced by the tutorial.

Design Decision 2: Reduced Hardware Size: During RAD 1.0, there was consensus among the participants that the hardware resources had larger dimensions than need be, and this impacted their tool use experiences and the potential use of the tool in classroom settings. According to P3, "the physical board and blocks were a bit **overwhelming** to explore and in terms of **arm span and length**." Similarly, P8 noted that "the physical board is **cumbersome**... It doesn't seem like something that could be provided to a **large group**." The research team, consequently redesigned significantly smaller and more compact hardware tiles and grid board, as shown in **Figure 5**. During the RAD 2.0 "Dive Into the Past" activity of the co-design reflection session, all participants unanimously preferred the new, reduced dimensions. When discussing whether the new

Table 3: Introduced design improvements to the tool and relevant participant decisions and feedback during RAD 1.0 and RAD 2.0.

RAD 1.0 Design Feedback	Design Decision and/or Improvement	RAD 2.0 Design Feedback
1.A steep learning curve	Tutorial design and pre-program demo	A smooth and affirming tutorial experience
2.Very bulky hardware	Hardware dimensions reduced significantly	Better size, could use firmer structure for tile placement
3.Missing color contrast	Black labels on white background	Mixed feedback on contrast order

tool design could be used in its current state in classrooms by students, P3 simply responded, "**Definitely**. Totally." That being said, some participants found the new functional shapes of the tiles to be less sturdy than the typical square blocks in the square cells of the grid. This especially applied to round tiles, which were now small and light enough to be rotated by mistake in place. P6 elaborated, "The blocks could be easily **bumped out** of position or alignment. I guess it depends on one's personality as to whether they like things to rigidly remain in place or tolerate a bit of disorder. For the fully blind users, I guess it would help to easily trace the circuit lines without fear of moving the blocks out of place."

Design Decision 3: Introducing Color Contrast: The tactile embossing (braille, schematics, markers) adopted for the prototype was done on white paper. A low-vision participant remarked that "since [they] haven't had experience with braille, [they] didn't have the tactile sensitivity to distinguish the small dots. It would've helped to have higher **contrast** between the dots and the blocks' background." As a result, the research team introduced black color lining to all tactile content. However, during RAD 2.0, participants with low vision had different reactions to the implementation. Comparing P1's video-recorded experiences using the blocks during RAD 1.0 to using the colored tiles during RAD 2.0, she fully changed her approach from asking the researcher for help in finding the needed tactile blocks, to successfully and quickly identifying the colored tiles by looking at them up close. As for P6, he recommended more specific contrast, digitally and physically. Regarding the hardware, he shared, "It would help to have more contrast between the pieces and the board, so that the circuit is more accentuated for low-vision users. It would help to have **more contrast** for the wire circuits." As for the digital interface, he "disliked having parts of the screen having different black and white backgrounds," as the digital grid had alternating light and dark column colors, and the tutorial bubble had a dark background. On the other hand, P1 shared during her individual simulator activity that she enjoyed the alternating column colors as they served as an easy identifier of distinct neighboring cells.

7 Discussion

7.1 Addressing the Initial Research Questions

Participants overall enjoyed the program. They "wished [they] could stay for the rest of the day" and described the program as "challenging and rewarding, [a] perfect balance." In addition, they found the improved IncluSim tool design to be "simple and intuitive" according to P3, "accessible and fun" according to P6, and offering a "low risk and high reward" simulation experience, according to P1.

The collective findings from RAD 1.0 and 2.0, presented in **Section 6** directly address our initial research questions, listed in **Section 1**. IncluSim enabled users to complete *electronic circuit design and simulation* tasks according to given requirements in an accessible manner, integrating smooth interaction with digital screen readers as well as tactile hardware with color contrast. Across different activities, participants *debugged* their circuits successfully, appreciating the hints and advice that IncluSim offered to better interpret simulation results. Finally, whether it be designing a circuit from scratch or figuring out an issue with a given circuit, participants often brought up the importance of the *hardware-digital* interface in facilitating their experience. The hardware allowed participants to spatially navigate their designs and provided a cognitive offloading medium. The digital interface provided instant feedback to verify theoretical calculations and identify any design warnings or errors. This timeliness in feedback is an important feature in contrast to the very slow debugging process that Kulkarni had to go through, as described in **Section 3.2**.

7.2 Is a Hardware Medium Necessary?

A strong case can be made for a software-only tool design which can significantly scale and expand digitally at no cost to more learners and education spaces, in comparison to the current version which requires the replication of the open-source hardware design resources, though the latter remain relatively affordable for the time being. Based on participant feedback and findings from the literature, we argue that the circuit design and simulation tasks in this work necessitate the presence of a hardware medium.

Participants' feedback in **Section 6.3** emphasized the importance of a hardware medium for accessible, smooth, and successful simulation. The participants described the hardware grid board as a cognitive offloading medium to support memory and knowledge retention. This is corroborated by research establishing *cognitive artifacts* as memory aids and their role in engineering education [26][23][4]. In line with Norman's definition of cognitive artifacts to "maintain, display, or operate upon information in order to serve a representational function," according to participants, the hardware medium enabled the tactile display of the circuit design and its spatial navigation [47]. To further illustrate the importance of hardware as a cognitive artifact, we highlight the specific aspects of the IncluSim design and simulation process. First, every tactile tile includes an orientation marker, which could be rotated in four directions, and a schematic of the electronic component, which

may have one to four different terminals that connect to other components, with an example shown in **Figure 5**. While the digital screen prints the orientation of the tile as top, bottom, left, or right, this *text-based* description does not suffice to describe where the terminals of the component can end up connecting. It would be hard to remember mentally what the orientation implies in terminal placement and connection for tens of components simultaneously. The hardware provides a tactile reference to do so. In addition, while smaller circuits could be constructed digitally, a circuit may include series and parallel branchings and multiple connections to the same component. The screen reader can navigate the digital grid using the Arrow keys, but only one cell at a time. This navigation would not be representative of larger parallel or series connections and would again require the learner to memorize simultaneous connections. However, using the hardware grid, participants can trace the paths of circuit connections and identify local and broader types of circuit branches. Some participants used two hands at a time during RAD studies to trace the different connections to one component simultaneously using the hardware. Thus, the hardware artifact offers an accessible and convenient medium to aid memory and navigation, where the software medium falls short.

7.3 Design Decisions in Context

Different design decisions made during the iterative co-design process of IncluSim can be linked to relevant literature and inclusive design frameworks. Low-vision participants prioritized the addition of *color contrast*, as supported by literature [70][13]. BLV participants decided to *reduce the tool hardware size* for more convenient use individually and in classroom settings. Blind members of our research team additionally brought up the importance of non-conspicuous physical tool design that would be used in classroom and group settings to avoid feelings of self-awareness, and research has often commented on the need for less bulky, more non-conspicuous BLV tools [41][17][20][40]. Moreover, as participants strongly encouraged the integration of a BLV-accessible tool tutorial, the research team developed a tutorial in line with recent research on non-visual authoring and screen-reader friendly interactions for BLV tutorial design, unlike traditional app tutorials which significantly rely on pop-ups and visual pointers to screen content [60][61].

Participant feedback on the *grid-based* hardware and digital interface design emphasizes the importance of spatial and directional navigation, rather than text-based information, in exploring spatial circuit design and simulation, in line with the importance of "space" in blind spatial cognition research [19]. Furthermore, recent work by Lee et al. confirmed that a *tile-based* design paradigm can "support spatial understanding and manipulation of a variety of visual content" in canvas and images of objects, allowing for more intuitive mental mapping of spatial content [38]. Their proposed tool, AltCanvas, offers BLV users a medium for tile-based image editing and interaction using keyboard navigation, aligning "more closely with natural spatial cognition building," away from conventional text-based descriptions. The IncluSim grid-based interface design elicited similar feedback from participants pertaining to multi-directional navigation abilities, starting from one cell or tile, and allowing for tactile spatial cognition in designing, verifying, and modifying circuit designs.

Finally, in implementing many design changes and decisions, we relied on the UDL and WCAG frameworks mentioned in **Section 4.2** to ensure that the integrated features were inclusive and BLV-accessible. We believe that by adopting the co-design method with the BLV community, our design process also inherently adhered to the ability-based design principles [72]. In **Table 4**, we discuss some of the IncluSim design features or processes in relation to principles of multiple accessibility-centric design frameworks [62][7][37].

7.4 Expanding the Design Criteria of BLV-Inclusive Engineering Education Tools

We propose additional considerations to expand the list of inclusive design criteria for accessible engineering education tools in the work of Kulkarni et al. [37].

(1) *Comprehensive tutorial objectives:* Principle 8 in the work of Kulkarni et al. advocated for a tool that must not be time-consuming to learn to use [37]. We expand this principle to incorporate tutorial guidance on the different modes of interaction that the learner needs for success, particularly from an accessibility perspective. The IncluSim tutorial introduced the user to the **main functionalities** of the simulator, the **relevance** of the hardware resources, the **shortcuts** to facilitate keyboard-only interaction with the tool, and it offered guidance on **efficient screen reader interaction** with the digital interface. We believe that these are core areas to be covered by tutorials of accessible tools, in order to significantly smoothen the learning curve of the user with the tool *on all fronts*, in contrast to a more conventional tutorial that would have focused solely on the simulator functions.
(2) *Beyond one-size-fits-all design decisions:* Our findings demonstrate how a common and expected design criterion, such as color contrast, may not elicit unified feedback from the BLV community. For instance, P1 and P6, both of whom have low vision, had opposing thoughts on alternating column colors in the digital grid, as discussed in **Section 6.5**. Thus, integrating customizable features in the tool design can actively include more learners with different accessibility preferences. Co-design is a powerful design method to elicit such diverse reactions.

7.5 Limitations and Opportunities for Future Work

The current IncluSim digital version is optimized for use with MacOS and the VoiceOver screen reader. We plan on developing versions that are compatible with other operating systems and screen readers to ensure larger access. Moreover, we hope to increase our sample size and evaluate the use of the tool in classroom settings. We are also investigating the learning outcomes of using IncluSim and its different features from embodied and sociocultural learning perspectives.

Furthermore, there are multiple areas which future research can capitalize on to expand the capabilities of IncluSim. First, the current tool design is most suitable for use in introductory engineering courses and makerspaces. More advanced courses may deal with larger circuits. A possible design change to accommodate for more complex circuits would be allowing the user to save and import

Table 4: IncluSim design processes and features in relation to the principles of different accessibility-centric design frameworks.

IncluSim Design Feature or Process	**Design Framework or Guideline *(Specific Principle)***
An open-source digital tool with low-cost, replicable hardware	Ability-based design *(Availability)*
Iterative co-design with the BLV community, adapting the tool design to the community's needs and design decisions	Ability-based design *(Ability; Accountability)*
Hardware with tactile schematics and with color contrast	Ability-based design *(Adaptability)* UDL *(Principle 1. Provide options for perception)*
Hints and advice on interpreting warning and error messages in the digital interface	UDL *(Principle 7.2. Minimize threats and distractions)*
Accessible tool interaction and navigation using a digital screen reader	UDL *(Principle 4.2. Optimize access to tools and assistive technologies)* WCAG *(Guideline 2.4. Navigable)*
Modular hardware tiles	Kulkarni et al. *(Principle 9. Modularity)*
Import and *Export* functionalities to save new circuit designs and load previous ones	Kulkarni et al. *(Principle 5. Retention of memory and knowledge)*

sub-circuit modules into the grid, with each module representing a whole circuit. This can save the learner space in the grid. Another major area for future work is exploring AC analysis simulation results. IncluSim currently runs DC simulations, which often produce exact numerical results, while AC simulations generate alternating waves to represent the variation of a signal over time or frequency. Sonification is a promising technique that can represent such graphical simulation outputs for BLV users [53][25][71][1]. Finally, we invite the research community to consider what a BLV-accessible, software-only, scalable simulation environment could entail.

8 Conclusion

In this work, we proposed IncluSim, a novel, accessible electronic circuit simulator tool for blind and low-vision learners pursuing introductory electronics education. Over 2.5 years of design efforts and two studies, we co-designed this tool with the BLV community. Co-design gave the BLV participants the power to influence design decisions and shape the prioritized features of the tool. Our findings from both studies show that IncluSim, via its hardware-digital interface, enabled the users to design, simulate, and debug electronic circuit designs in an accessible and efficient manner. The tool code is open-source, and the IncluSim tool is available for free use by learners, educators, and interested BLV community members.

Acknowledgments

We thank our BLV participants for their full investment in our co-design process. We thank the Stanford RAISE program for the training, funding, and support that made our community partnership efforts possible. We also thank the Stanford Office of Accessible Education, the Stanford Designing Education Lab members, and the LightHouse for the Blind and Visually Impaired staff for their extensive mentorship, feedback, and guidance. Finally, this work gratefully builds upon the prior efforts of BLV researchers and allies.

References

[1] Safinah Ali, Laya Muralidharan, Felicia Alfieri, Monali Agrawal, and Jacob Jorgensen. 2020. Sonify: making visual graphs accessible. In *Human Interaction and Emerging Technologies: Proceedings of the 1st International Conference on Human Interaction and Emerging Technologies (IHIET 2019), August 22-24, 2019, Nice, France*. Springer, 454–459.

[2] Ayah Bdeir. 2009. Electronics as material: littleBits. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction* (Cambridge, United Kingdom) *(TEI '09)*. Association for Computing Machinery, New York, NY, USA, 397–400. https://doi.org/10.1145/1517664.1517743

[3] Cynthia L. Bennett, Abigale Stangl, Alexa F. Siu, and Joshua A. Miele. 2019. Making Nonvisually: Lessons from the Field. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility* (Pittsburgh, PA, USA) *(ASSETS '19)*. Association for Computing Machinery, New York, NY, USA, 279–285. https://doi.org/10.1145/3308561.3355619

[4] Jonte Bernhard. 2009. Learning through artifacts in engineering education: Some perspectives from the philosophy of technology and engineering science. In *Proceedings of the 6th European Forum on Continuing Engineering Education, Quality Developmentin Lifelong Learning-in Theory and Use.*

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

[6] Ingrid Burkett. 2012. An introduction to co-design. *Sydney: Knode* 12 (2012).

[7] Ben Caldwell, Michael Cooper, Loretta Guarino Reid, Gregg Vanderheiden, Wendy Chisholm, John Slatin, and Jason White. 2008. Web content accessibility guidelines (WCAG) 2.0. *WWW Consortium (W3C)* 290, 1-34 (2008), 5–12.

[8] Ruei-Che Chang, Wen-Ping Wang, Chi-Huan Chiang, Te-Yen Wu, Zheer Xu, Justin Luo, Bing-Yu Chen, and Xing-Dong Yang. 2021. AccessibleCircuits: Adaptive Add-On Circuit Components for People with Blindness or Low Vision. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 670, 14 pages. https://doi.org/10.1145/3411764.3445690

[9] Kathy Charmaz. 1983. The grounded theory method: An explication and interpretation. *Contemporary field research* (1983), 109–126.

[10] Josh Urban Davis, Te-Yen Wu, Bo Shi, Hanyi Lu, Athina Panotopoulou, Emily Whiting, and Xing-Dong Yang. 2020. TangibleCircuits: An Interactive 3D Printed Circuit Education Tool for People with Visual Impairments. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376513

[11] Analog Devices. 2024. *LTSPice Information Center.* Retrieved September 9, 2024 from https://www.analog.com/en/resources/design-tools-and-calculators/ltspice-simulator.html

[12] Samuel J Dickerson and Renee M Clark. 2018. A classroom-based simulation-centric approach to microelectronics education. *Computer Applications in Engineering Education* 26, 4 (2018), 768–781.

[13] Silvia Dini, Luca Andrea Ludovico, and María Joaquina Valero Gisbert. 2024. Bridging the color barrier: a review of techniques for improving color perception in the blind and visually impaired. *Color Research & Application* 49, 5 (2024), 515–534.

[14] Daniel Drew, Julie L. Newcomb, William McGrath, Filip Maksimovic, David Mellis, and Björn Hartmann. 2016. The Toastboard: Ubiquitous Instrumentation and Automated Checking of Breadboarded Circuits. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 677–686. https://doi.org/10.1145/2984511.2984566

[15] Sahar Fazeli, Judith Sabetti, and Manuela Ferrari. 2023. Performing qualitative content analysis of video data in social sciences and medicine: the visual-verbal video analysis method. *International Journal of Qualitative Methods* 22 (2023), 16094069231185452.

[16] Deborah Ann Fields, Yasmin Kafai, Tomoko Nakajima, Joanna Goode, and Jane Margolis. 2018. Putting making into high school computer science classrooms: Promoting equity in teaching and learning with electronic textiles in exploring computer science. *Equity & Excellence in Education* 51, 1 (2018), 21–35.

[17] Paul B Freeman. 2024. Low vision: overview and review of low vision evaluation and treatment. *Understanding and Managing Vision Deficits* (2024), 277–299.

[18] Amy Fritz, Mark Horowitz, and Atindra Jha. 2021. Online, Interactive Tool for Studying How Students Troubleshoot Circuits. In *2021 American Society for Engineering Education (ASEE) Virtual Annual Conference Content Access*.

[19] Nicholas A Giudice. 2018. Navigating without vision: Principles of blind spatial cognition. In *Handbook of behavioral and cognitive geography*. Edward Elgar Publishing, 260–288.

[20] Nicholas A Giudice and Gordon E Legge. 2008. Blind navigation and the role of technology. *The engineering handbook of smart technology for aging, disability, and independence* (2008), 479–500.

[21] Richard Goldberg. 2016. The 'Invisible Handshake'Project as a Practical, Hands-on Experience in a Biomedical Electronics Class. In *2016 American Society for Engineering Education (ASEE) Annual Conference & Exposition*.

[22] Liwen He, Yifan Li, Mingming Fan, Liang He, and Yuhang Zhao. 2023. A Multimodal Toolkit to Support DIY Assistive Technology Creation for Blind and Low Vision People. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23 Adjunct)*. Association for Computing Machinery, New York, NY, USA, Article 3, 3 pages. https://doi.org/10.1145/3586182.3616646

[23] Richard Heersmink. 2013. A taxonomy of cognitive artifacts: Function, information, and categories. *Review of philosophy and psychology* 4 (2013), 465–481.

[24] Steve Hodges, Sue Sentance, Joe Finney, and Thomas Ball. 2020. Physical Computing: A Key Element of Modern Computer Science Education. *Computer* 53, 4 (2020), 20–30. https://doi.org/10.1109/MC.2019.2935058

[25] Leona M Holloway, Cagatay Goncu, Alon Ilsar, Matthew Butler, and Kim Marriott. 2022. Infosonics: Accessible Infographics for People who are Blind using Sonification and Voice. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 480, 13 pages. https://doi.org/10.1145/3491102.3517465

[26] Edwin Hutchins. 1999. Cognitive artifacts. *The MIT encyclopedia of the cognitive sciences* 126 (1999), 127.

[27] The Smith-Kettlewell Eye Research Institute. 2023. *Blind Arduino Project*. Retrieved September 1, 2024 from https://www.ski.org/projects/blind-arduino-project

[28] Frederico Jacob, António Alberto, and Pedro Guimarães. 2021. Use of Tinkercad platform for Teaching Electronics Subject in Post-Secondary Technical Courses. In *Ninth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'21)* (Barcelona, Spain) *(TEEM'21)*. Association for Computing Machinery, New York, NY, USA, 543–547. https://doi.org/10.1145/3486011.3486517

[29] Karen H. Jin, Kathleen Haynie, and Gavin Kearns. 2016. Teaching Elementary Students Programming in a Physical Computing Classroom. In *Proceedings of the 17th Annual Conference on Information Technology Education* (Boston, Massachusetts, USA) *(SIGITE '16)*. Association for Computing Machinery, New York, NY, USA, 85–90. https://doi.org/10.1145/2978192.2978238

[30] R Burke Johnson. 1997. Examining the validity structure of qualitative research. *Education* 118, 2 (1997).

[31] Yasmin B. Kafai, Eunkyoung Lee, Kristin Searle, Deborah Fields, Eliot Kaplan, and Debora Lui. 2014. A Crafts-Oriented Approach to Computing in High School: Introducing Computational Concepts, Practices, and Perspectives with Electronic Textiles. *ACM Trans. Comput. Educ.* 14, 1, Article 1 (mar 2014), 20 pages. https://doi.org/10.1145/2576874

[32] Yoonji Kim, Youngkyung Choi, Hyein Lee, Geehyuk Lee, and Andrea Bianchi. 2019. VirtualComponent: A Mixed-Reality Tool for Designing and Tuning Breadboarded Circuits. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300407

[33] Yoonji Kim, Hyein Lee, Ramkrishna Prasad, Seungwoo Je, Youngkyung Choi, Daniel Ashbrook, Ian Oakley, and Andrea Bianchi. 2020. SchemaBoard: Supporting Correct Assembly of Schematic Circuits using Dynamic In-Situ Visualization. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 987–998. https://doi.org/10.1145/3379337.3415887

[34] André Knörig, Reto Wettach, and Jonathan Cohen. 2009. Fritzing: a tool for advancing electronic prototyping for designers. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction* (Cambridge, United Kingdom) *(TEI '09)*. Association for Computing Machinery, New York, NY, USA, 351–358. https://doi.org/10.1145/1517664.1517735

[35] Krzysztof Tomasz Konecki. 2011. Visual grounded theory: A methodological outline and examples from empirical work. *Revija za sociologiju* 41, 2 (2011), 131–160.

[36] Gunther Kress. 2009. *Multimodality: A social semiotic approach to contemporary communication*. routledge.

[37] Trisha Kulkarni, Gene S-H Kim, and Aya Mouallem. 2023. A Case for Improving the Accessibility of Electrical and Computer Engineering Education – Starting with a Blind Student's Autoethnography. In *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility* (New York, NY, USA) *(ASSETS '23)*. Association for Computing Machinery, New York, NY, USA, Article 96, 8 pages. https://doi.org/10.1145/3597638.3614551

[38] Seonghee Lee, Maho Kohga, Steve Landau, Sile O'Modhrain, and Hari Subramonyam. 2024. AltCanvas: A Tile-Based Editor for Visual Content Creation with Generative AI for Blind or Visually Impaired People. In *Proceedings of the 26th International ACM SIGACCESS Conference on Computers and Accessibility*. 1–22.

[39] Woojin Lee, Ramkrishna Prasad, Seungwoo Je, Yoonji Kim, Ian Oakley, Daniel Ashbrook, and Andrea Bianchi. 2021. VirtualWire: Supporting Rapid Prototyping with Instant Reconfigurations of Wires in Breadboarded Circuits. In *Proceedings of the Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Salzburg, Austria) *(TEI '21)*. Association for Computing Machinery, New York, NY, USA, Article 4, 12 pages. https://doi.org/10.1145/3430524.3440623

[40] Chen Liang, Yasha Iravantchi, Thomas Krolikowski, Ruijie Geng, Alanson P Sample, and Anhong Guo. 2023. BrushLens: hardware interaction proxies for accessible touchscreen interface actuation. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–17.

[41] Heather L Mason. 1999. Blurred vision: A study of the use of low vision aids by visually impaired secondary school pupils. *British Journal of Visual Impairment* 17, 3 (1999), 94–97.

[42] David A. Mellis, Leah Buechley, Mitchel Resnick, and Björn Hartmann. 2016. Engaging Amateurs in the Design, Fabrication, and Assembly of Electronic Devices. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems* (Brisbane, QLD, Australia) *(DIS '16)*. Association for Computing Machinery, New York, NY, USA, 1270–1281. https://doi.org/10.1145/2901790.2901833

[43] Joshua Miele. 2018. *Blind Arduino Blog*. Retrieved September 1, 2024 from http://blarbl.blogspot.com

[44] Aya Mouallem, Mark Horowitz, and Sheri D Sheppard. 2023. The CARE methodology: a new lens for introductory ECE course assessment based on student Challenging And Rewarding Experiences. In *2023 American Society for Engineering Education (ASEE) Annual Conference & Exposition*.

[45] Aya Mouallem, Trisha Kulkarni, and Sheri D Sheppard. 2024. Leveraging the CARE Methodology to Enhance Pedagogical and Institutional Support for Blind or Low-Vision (BLV) Learners in Electrical and Computer Engineering (ECE). In *2024 American Society for Engineering Education (ASEE) Annual Conference & Exposition*.

[46] RM Nieswiadomy and C Bailey. 2002. Qualitative research designs. *Foundations of Nursing Research (4 th ed.). Upper Saddle River, New Jersey. Prentice Hall* (2002).

[47] Donald A Norman. 1991. Cognitive artifacts. *Designing interaction: Psychology at the human-computer interface* 1, 1 (1991), 17–38.

[48] Dan O'Sullivan and Tom Igoe. 2004. *Physical computing: sensing and controlling the physical world with computers*. Course Technology Press, Boston, MA, USA.

[49] Nils Pancratz and Ira Diethelm. 2018. Including part-whole-thinking in a girls' engineering course through the use of littleBits: A practical report on including part-whole-thinking into the content of computer science education. In *2018 IEEE Global Engineering Education Conference (EDUCON)*. 1134–1138. https://doi.org/10.1109/EDUCON.2018.8363357

[50] D. Pederson. 1984. A historical review of circuit simulation. *IEEE Transactions on Circuits and Systems* 31, 1 (1984), 103–111. https://doi.org/10.1109/TCS.1984.1085422

[51] Emma C. Pender and John J. Healy. 2022. Accessible Circuit Diagrams. In *2022 33rd Irish Signals and Systems Conference (ISSC)*. 1–6. https://doi.org/10.1109/ISSC55427.2022.9826152

[52] Ken Perry. 2016. *Blind Electronics*. Retrieved September 1, 2024 from http://www.blindelectronics.com

[53] Venkatesh Potluri, John Thompson, James Devine, Bongshin Lee, Nora Morsi, Peli De Halleux, Steve Hodges, and Jennifer Mankoff. 2022. PSST: Enabling Blind or

Visually Impaired Developers to Author Sonifications of Streaming Sensor Data. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 46, 13 pages. https://doi.org/10.1145/3526113.3545700

[54] Jie Qi, Leah Buechley, Andrew "bunnie" Huang, Patricia Ng, Sean Cross, and Joseph A. Paradiso. 2018. Chibitronics in the Wild: Engaging New Communities in Creating Technology with Paper Electronics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3173574.3173826

[55] Lauren Race, Chancey Fleet, Joshua A. Miele, Tom Igoe, and Amy Hurst. 2019. Designing Tactile Schematics: Improving Electronic Circuit Accessibility. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility* (Pittsburgh, PA, USA) *(ASSETS '19)*. Association for Computing Machinery, New York, NY, USA, 581–583. https://doi.org/10.1145/3308561.3354610

[56] Lauren Race, Claire Kearney-Volpe, Chancey Fleet, Joshua A. Miele, Tom Igoe, and Amy Hurst. 2020. Designing Educational Materials for a Blind Arduino Workshop. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3334480.3383055

[57] Lauren Race, Joshua A. Miele, Chancey Fleet, Tom Igoe, and Amy Hurst. 2020. Putting Tools in Hands: Designing Curriculum for a Nonvisual Soldering Workshop. In *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility* (Virtual Event, Greece) *(ASSETS '20)*. Association for Computing Machinery, New York, NY, USA, Article 78, 4 pages. https://doi.org/10.1145/3373625.3418011

[58] Michael Ray. 2021. *Raspberry VI*. Retrieved September 1, 2024 from http://www.raspberryvi.org/pages/index.html

[59] M. Resnick, F. Martin, R. Sargent, and B. Silverman. 1996. Programmable Bricks: Toys to think with. *IBM Systems Journal* 35, 3.4 (1996), 443–452. https://doi.org/10.1147/sj.353.0443

[60] André Rodrigues, Leonardo Camacho, Hugo Nicolau, Kyle Montague, and Tiago Guerreiro. 2018. Aidme: Interactive non-visual smartphone tutorials. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. 205–212.

[61] André Rodrigues, André Santos, Kyle Montague, Hugo Nicolau, and Tiago Guerreiro. 2019. Understanding the authoring and playthrough of nonvisual smartphone tutorials. In *Human-Computer Interaction–INTERACT 2019: 17th IFIP TC 13 International Conference, Paphos, Cyprus, September 2–6, 2019, Proceedings, Part I 17*. Springer, 42–62.

[62] David Rose. 2000. Universal Design for Learning. *Journal of Special Education Technology* 15, 4 (2000), 47–51. https://doi.org/10.1177/016264340001500407

[63] Uğur Sarı, Hüseyin Miraç Pektaş, Ömer Faruk Şen, and Harun Çelik. 2022. Algorithmic thinking development through physical computing activities with Arduino in STEM education. *Education and Information Technologies* 27, 5 (2022), 6669–6689.

[64] JooYoung Seo. 2019. Is the Maker Movement inclusive of anyone?: Three accessibility considerations to invite blind makers to the making world. *TechTrends* 63, 5 (2019), 514–520.

[65] Katherine Steele, Brianna Blaser, and Maya Cakmak. 2018. Accessible making: Designing makerspaces for accessibility. *International Journal of Designs for Learning* 9, 1 (2018), 114–121.

[66] Marc Steen. 2013. Co-Design as a Process of Joint Inquiry and Imagination. *Design Issues* 29, 2 (04 2013), 16–28. https://doi.org/10.1162/DESI_a_00207 arXiv:https://direct.mit.edu/desi/article-pdf/29/2/16/1715163/desi_a_00207.pdf

[67] Evan Strasnick, Maneesh Agrawala, and Sean Follmer. 2021. Coupling Simulation and Hardware for Interactive Circuit Debugging. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 667, 15 pages. https://doi.org/10.1145/3411764.3445422

[68] Anselm Strauss and Juliet Corbin. 1994. Grounded theory methodology: An overview. *Handbook of qualitative research* (1994).

[69] Abbas Tashakkori and Charles Teddlie. 2003. The past and future of mixed methods research: From data triangulation to mixed model designs. *Handbook of mixed methods in social and behavioral research* 671 (2003), 701.

[70] Garreth W Tigwell. 2021. Nuanced perspectives toward disability simulations from digital designers, blind, low vision, and color blind people. In *Proceedings of the 2021 CHI conference on human factors in computing systems*. 1–15.

[71] B. N. Walker and L. M. Mauney. 2010. Universal Design of Auditory Graphs: A Comparison of Sonification Mappings for Visually Impaired and Sighted Listeners. *ACM Trans. Access. Comput.* 2, 3, Article 12 (mar 2010), 16 pages. https://doi.org/10.1145/1714458.1714459

[72] Jacob O. Wobbrock, Shaun K. Kane, Krzysztof Z. Gajos, Susumu Harada, and Jon Froehlich. 2011. Ability-Based Design: Concept, Principles and Examples. *ACM Trans. Access. Comput.* 3, 3, Article 9 (apr 2011), 27 pages. https://doi.org/10.1145/1952383.1952384

[73] Theodore Zamenopoulos and Katerina Alexiou. 2018. *Co-design as collaborative research*. Bristol University/AHRC Connected Communities Programme.

A Circuit Design and Verification Tasks

(1) Set up a circuit design with a specific configuration using the hardware blocks. Participants had to design an electronic circuit composed of a current source and two resistors, all connected in parallel
(2) Run some theoretical calculations using formulas and a calculator to find a current or voltage value across specific circuit components. Participants used the conventional current equations across resistors in parallel
(3) Simulate the circuit design to find the current or voltage value using the digital simulator and referring to the hardware blocks design. Participants had to calculate the currents flowing through each resistor
(4) Compare simulation results to theoretical calculations to verify work or debug further. Participants also had to verify that the sum of currents branching into each resistor was equal to the current flowing from the current source
(5) Vary parameter values in a given circuit design and build intuition about expected trends in line with Ohm's law (only during RAD 2.0)
(6) Debug and fix the circuitry of a malfunctioning electric heater (only during RAD 2.0)

B Design Thinking and Co-Design Tasks

- Reflect individually in a written journal entry on personal experiences with the hardware and digital components of the tool. Some guiding prompts for this task were: (1) What went well with using the tool today, and/or what didn't go well? (2) Did you find the use of any tool features challenging? (3) Were any of the experiences you had rewarding?
- Complete a think-pair-share activity, by discussing specific design prompts in pairs then reflecting with the larger group
- Identify common and contrasting experiences with specific tool features and empathize with others' experiences by asking 5 Why's consecutively
- Ideate and propose modifications to existing features (i.e., What features could be removed or modified?), and brainstorm new features that participants would like to have implemented in the upcoming design iteration
- Make design decisions based on turning points in the design process as described by researchers
- "Dive Into the Past" (activity during which participants are handed the old and new tool hardware designs): Re-explore both prototypes haptically to reflect on the integrated changes (only during RAD 2.0)

C Participation Confirmation Survey Sample Questions

In the participation confirmation survey, we asked basic questions about the refresher document content to verify that participants had read it. Some of these questions were:

- You are given a resistor with resistance R=10 Ohms and current I=2 Amps. Find the voltage value V across the resistor.
- What keyboard key do you press to launch the simulation profile using the digital simulator? *Multiple choice options: S key, L key, B key.*

D Final Survey Sample Questions

The following are sample questions selected from the final surveys of RAD 1.0 and RAD 2.0.

Part A: Simulator Activities

(1) To what extent do you agree or disagree with the following statements?
Answer format: *Likert-scale options (Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree).*
   (a) This tool allowed me to experience both: hands-on circuit design and digital simulation.
   (b) This tool allowed me to check, verify, and fix any errors in my work.
   (c) This tool was accessible for me to use as a user who is blind or has low vision.

(2) Walk us through your thought process of using this tool, including the hardware grid board, tactile tiles, and digital simulator, as you worked on Part A of your activity workbook. Did you iterate through the hardware and digital components? How did you find this process?
Answer format: *Open-ended response.*

(3) If you had any, tell us about your experience debugging (finding and fixing errors) with this tool.
Answer format: *Open-ended response.*

Part B: Co-Design Activities

(1) To what extent do you agree or disagree with the following statements?
Answer format: *Likert-scale options (Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree).*
   (a) I enjoyed learning and exploring new perspectives and ideas by fellow participants and team members.
   (b) I feel confident that my contributions to the design discussion will be taken into consideration by the design team to influence the future design of the tool.

(2) Did you feel that the research team had taken your prior recommendations from the previous RAD study in December 2023 into consideration, in the improved simulator design? Please elaborate.
Answer format: *Open-ended response.*

(3) Did your experience co-designing the simulator tool influence your learning experience using the tool? If yes, how?
Answer format: *Open-ended response.*

E Open-Source IncluSim Resources

E.1 Hardware Resources

The do-it-yourself (DIY) design guide for the hardware components of IncluSim (hardware grid board, tactile tiles, and reference grid) can be found at bit.ly/IncluSimHardwareGuide.

E.2 Digital Resources

The installation and set-up guide for the digital IncluSim simulator interface can be found at bit.ly/IncluSimDigitalGuide. The open-source, anonymized code repository for the project can be found at https://anonymous.4open.science/r/IncluSim/.