2021 ASEE ANNUAL CONFERENCE
Virtual Meeting | July 26–29, 2021 | Pacific Daylight Time
Paper ID #33334

Online, Interactive Tool for Studying How Students Troubleshoot Circuits

Amy Fritz, Stanford University

Amy Fritz is an electrical engineering PhD student at Stanford University who works with Professor Mark Horowitz in the department of electrical engineering on education research.

Prof. Mark Horowitz, Stanford University

Mark Horowitz is the Yahoo! Founders Professor at Stanford University and was chair of the Electrical Engineering Department from 2008 to 2012. He co-founded Rambus, Inc. in 1990 and is a fellow of the IEEE and the ACM and a member of the National Academy of Engineering and the American Academy of Arts and Science. Dr. Horowitz's research interests are quite broad and span using EE and CS analysis methods to problems in molecular biology to creating new design methodologies for analog and digital VLSI circuits.

Mr. Atindra Jha, Stanford University

Atindra is an undergraduate student at Stanford University pursuing Computer Science and Mathematics. He works under the guidance of Dr. Mark Horowitz and Amy Fritz, aiding their electrical engineering education research.

Online, Interactive Tool for Studying How Students Troubleshoot Circuits

Abstract

We have developed a new tool to look at how students interact with circuits during the troubleshooting process. The online tool was originally designed to analyze individual troubleshooting strategy for large classes, but it also works well in the COVID-era to facilitate remote learning. While there are a number of tools that allow students to virtually interact with circuits, none supported both breadboard graphics and recording all student interactions, which were necessary to create an authentic troubleshooting situation that could be analyzed by the researchers afterwards. Therefore, we created our own circuit and data analysis tool using HTML5, CSS, and JavaScript, which utilizes breadboard imagery from Fritzing and runs on most modern browsers. Unlike a traditional paper-and-pencil test, the interactive, online tool allows us to see how students react to new information and measure domain knowledge beyond theory—including interpreting physical circuits and making measurements. Instead of relying on students to tell us everything on their mind, we can use their actions as a proxy for their thought processes. This paper describes how we developed the tool and some preliminary data on how students debug.

Introduction

In our efforts to study how all 150 students in our introductory circuits class troubleshoot circuits, we needed a method more efficient than videotaping students in lab followed by time-consuming post-analysis to produce quantitative results. This paper describes our solution: an interactive, online tool to study troubleshooting. We designed the online tool to mimic an authentic lab experience. It includes physical representations of circuits that students would normally see in lab, like breadboards. It also includes a mechanism to take voltage measurements in the circuit. By logging all mouse and key clicks in the browser window as well as user replies, we observe both students' reasoning and actions as they work to understand the error in the circuit. Thus, our new troubleshooting tool allows us to capture rich data for large number of students quickly and effectively (and now remotely).

Using our new tool, we were able to better understand how students approach troubleshooting circuits. We were able to analyze where and why students got off track during the troubleshooting process. These results show that there are several key decisions that students make during the troubleshooting process that divides them into different groups including what measurements they chose to take, their comparison to theory, and whether they searched for a fix after determining something was wrong with the circuit. We also added the realistic elements to the tool step-by-step, starting with just a theoretical diagram moving to a picture of a breadboard with all measurements given and finally allowing students to take their own measurements on a breadboard, so we could see what effect each additional complexity had on students' troubleshooting success. We worried that as we added physical pictures of circuits and ways for students to make their own measurements to the system, that the task would become significantly more difficult. It turns out that the added complexities did not decrease performance and, in some cases, enhanced student performance. Finally, we investigated whether we could effectively use measurements as a proxy for thought process. Our results point to significant overlap between measurement patterns and final reasoning given.

Background

Since troubleshooting is a type of problem-solving, we follow the universal list of expert problem-solving decisions across the STEM fields identified by Carl Wieman's physics education research group [1]. Several relevant to our research include determining what information is needed and comparing predictions to data to draw appropriate conclusions. Troubleshooting also involves using a variety of different types of domain knowledge. The main three types of knowledge we distinguish between in this work is theoretical knowledge, physical knowledge, and measurement knowledge. Theoretical knowledge includes identifying relevant principles and equations, making calculations (like gain), and reading circuit schematics. Physical knowledge includes recognizing physical circuit components and being able to extract a circuit's schematic from a picture of these components on a physical breadboard. Finally, measurement knowledge includes knowing how to use measurement tools properly and interpret their readings. This division of troubleshooting domain knowledge is loosely based off of the cognitive task analysis research by Jonassen, who divided the task of troubleshooting into six types of knowledge states [2]. For a more thorough overview of troubleshooting as a form of problem-solving, see Rivera-Reyes and Boyle's meta-analysis of engineering troubleshooting literature [3].

This work builds upon previous troubleshooting studies like the work of the Lewandowski group that has studied the cognitive processes of students troubleshooting op amp circuits in a lab setting [4]. They approach troubleshooting with a model-based reasoning lens, which is similar to the expert problem-solving decisions surrounding comparing predictions and data mentioned above. Our contribution is creating a tool which enables data collection for Lewandowski group's type of analysis to large numbers of students. Other groups have shown the importance of multiple representations when students analyze and troubleshoot circuits [5] [6], which is why it was important to us to include multiple representations, like the breadboard and the circuit schematic, in our tool.

Researchers have also previously shown that students respond positively to troubleshooting activities in circuits classes and students state that their critical-thinking skills improved from the activities [7]. Our class experience is similar, and our tool makes creating this type of exercise easier. Another research group has shown that troubleshooting activities increase performance of underachieving students and help bridge the gap to their better-performing peers [8]. These studies highlight the potential of our tool as a curriculum intervention and not just a measurement tool for researchers.

Tool Design

We wanted our troubleshooting tool to mimic working with real hardware and allow us analyze the students' actions, which led to three requirements: 1) The visual should be able to be a physical circuit instead of just representing a circuit schematic 2) The students should be able to make changes to the circuit and see the result of their experiment 3) The program should record all student actions for later analysis.

We looked at other tools to see if any of them could fulfill our needs, but no tool met all the requirements. The VISIR platform [9] allowed students to remotely run experiments on a physical system, but it was missing some components we needed along with being too expensive and complex for our needs. TinkerCAD [10] and Edison Electricity & Electronics Education Software [11] had appropriate visuals and allowed students to make changes to the circuit, but it did not allow us to record students' actions. Fritzing [12] allowed us to make appropriate visuals, but it did not have a circuit simulator backend. Qualtrics [13], which we were previously using for our troubleshooting research, had good ways to record all students' actions, but no easy way for us to create a point and click interaction with a circuit.

The circuit analysis tool that we created, powered by basic HTML, CSS, JavaScript, and PHP programming, allows students to interact with a realistic circuit while enabling us to accurately map the series of steps that they followed during analyzing and troubleshooting the circuit. Students place the red and black probes in their desired breadboard holes to get the voltage difference displayed on the digital multimeter (DMM). We used Fritzing for the graphics of the breadboard and other circuit components and placed them on an HTML <canvas> element. This element allows for dynamic, scriptable rendering of two-dimensional shapes and bitmap images, which makes it possible for us to use JavaScript to draw probe wires on the fly and erase and reposition them swiftly when the connection is changed. We used two HTML radio buttons in order to get input on which of the two multimeter probes (the red or the black one) a student would like to connect to the breadboard at any given instant. Every time they need to change the probe, they can do so by selecting the radio button that indicates the name and color of the probe they'd like to switch to. The connection of a probe can be changed to a different hole than the one that it is connected to by simply clicking on the required hole. Students can continue inspecting the circuit and making measurements until they feel they have collected enough appropriate information to answer the follow-up questions regarding what is wrong with the circuit.

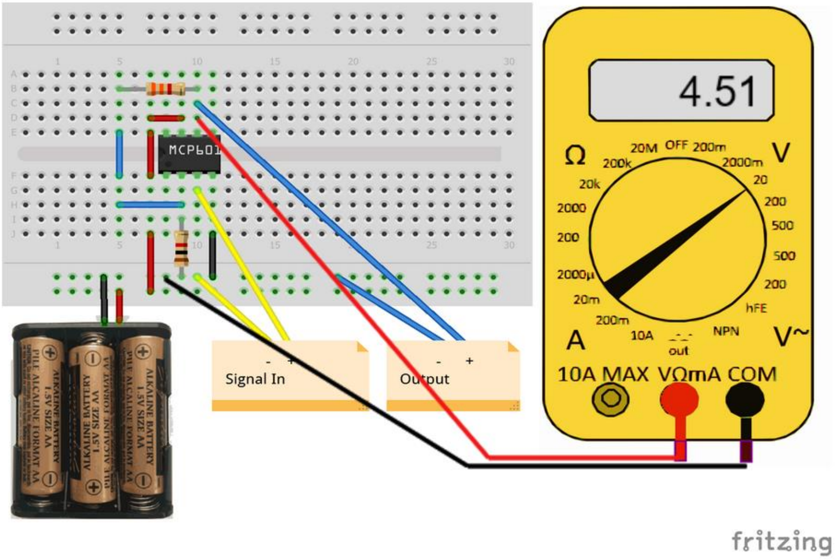


Figure 1. Graphical Interface. The experimental setup was portrayed using Fritzing. Students could click to attach probes from the DMM to the breadboard, which then displayed a voltage reading on the DMM.

We stored the voltage values for the breadboard holes in a JSON file in a way that grouped together holes which must have the same voltage: the “power supply rows” on the top and bottom of the breadboard, and the upper and lower columns of pins in the center. The program gets the voltage at the point where a probe is connected by identifying which of these groups the hole belongs to. When both probes are connected, it follows the same process in order to obtain the voltages at both points where the probes are connected and then displays the voltage difference on the multimeter screen by carrying out a simple arithmetic operation. The program, still in its developmental stages, was written to ensure simplicity and flexibility.

Creating the JSON file of voltages in this manner will allow most educators to change the voltage values based on their specific circuit relatively quickly (since connected holes with the same voltage are grouped together) and easily without having to access and alter other frontend or backend files. Using a JSON file to store the system voltages, rather than using an underlying simulator, makes it easy to create bugs in a circuit. We can easily change the voltages across a component or represent circuits with broken components and loose wires, without changing the circuit representation. While the system does not currently allow students to make changes to the circuit, we hope to add a circuit simulator running in the backend in the future to allow for such changes and have it generate a revised JSON file.

We use the MySQL relational database management system to collect various types of data on each student. HTML DOM (Document Object Model) allows JavaScript to react to HTML events such as mouse clicks, drags, drops, and key presses. We used mouse click events to get the location of each breadboard hole the student clicked on while also keeping note of the color of the probe they used at any instant and stored this information sequentially in the table we created in our SQL database. We also collected and stored timestamps which not only allowed us to measure the time that it took an individual to complete the troubleshooting exercise in its entirety, but also to obtain the time difference between each successive breadboard-hole click. To ensure maximum flexibility and ease-of-use, the website was designed in a manner that allowed students unlimited time (not having sessions expire automatically after a given amount of time). The backend works in a fashion that allows us to retain the data collected on a student even if the session is ended (the tab or the browser is closed) at any point after beginning but before submitting the exercise. We used the read-only localStorage property in order to save and access the University ID entered by a student during authentication on the first page of the website. Subsequent pages of the exercise accessed this ID (across browser sessions) so that all the data collected for a single student can be linked back together and stored under the specific student’s ID in the SQL table. This allows us to reconstruct a timeline of the student’s entire interaction with the circuit with Python scripts later.

Troubleshooting Experiment

Op amp circuits were chosen for the first exercise using this tool, because op amps are easy to breadboard and the material occurs at the end of term when students will be most advanced in their troubleshooting skills. The setup uses an op-amp and power supply range that students are familiar with from their final lab, details of which were previously published at this conference [14]. Students were required to troubleshoot four different circuits, two of which were inverting amplifiers and two of which were non-inverting amplifiers (see Figure 2). Students were asked to determine whether the circuit was working or not. After looking at the measurement data, students

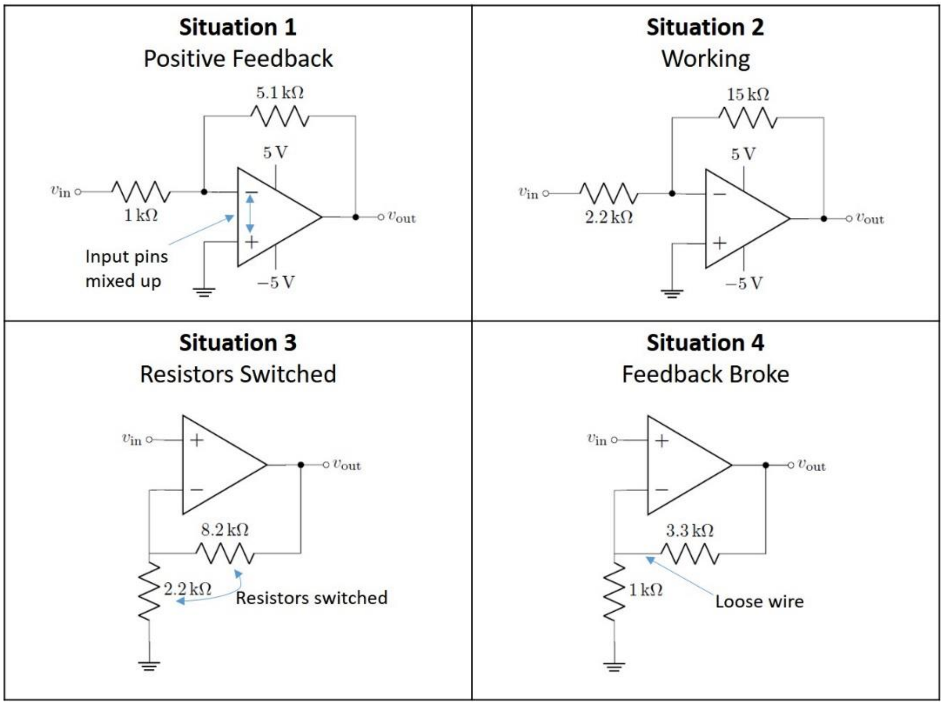
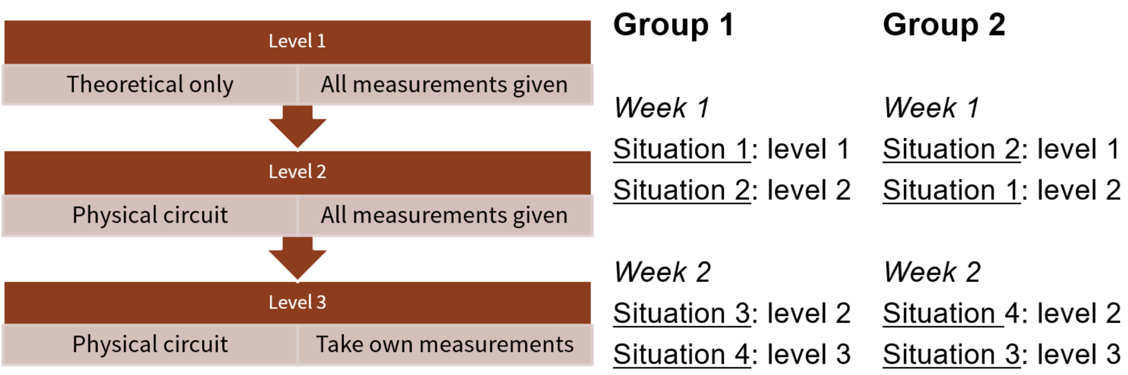


Figure 2. Over the course of two weeks, students participated in troubleshooting four op amp circuits, which are shown above. The problem with each circuit is indicated on the schematic.

had to choose whether the circuit was “working” or “not working” and then give reasoning in a blank below. If a student said the circuit was working, they were asked to explain how they knew it was working and what they expected the circuit to do in the first place. If they said it was not working, they were asked what was wrong with the circuit and what measurements indicated there was a problem with the circuit. They were also asked how to fix the circuit.

Over the course of two weeks, we gradually introduced students to the full troubleshooting tool. The goal was to add an additional domain of knowledge to each version of the assignment, so we could see how performance changed as the tool became more authentic. First, students were given a situation to troubleshoot where they were only given the theoretical circuit diagram and a list of measurements taken at various points of the circuit. We will call this level 1, since it required very little physical domain knowledge and measurement strategy (see Figure 3). Next, they were given a physical picture of the circuit with voltage measurements marked on the physical picture. This required students to translate the voltage readings from the physical circuit back to the theoretical circuit diagram to interpret the results. We will call this level 2. Finally, we had an interactive circuit where students had to take their own measurements (shown in Figure 1). This final situation tests their ability to use their measurement knowledge and take proper measurements. We will call this level 3, because it requires the use of knowledge from all three different domains—theoretical, physical, and measurement domains.



Group 1	Group 2
Week 1	*Week 1*
Situation 1: level 1	Situation 2: level 1
Situation 2: level 2	Situation 1: level 2
Week 2	*Week 2*
Situation 3: level 2	Situation 4: level 2
Situation 4: level 3	Situation 3: level 3

Figure 3. Experimental Design. Students were gradually introduced to the tool over two weeks, so we could measure how student performance changed as we added extra cognitive steps (levels) to the troubleshooting simulation. Students were broken into two groups and given the same troubleshooting situation at different levels so we could control for differences in performance across different troubleshooting situations.

To account for variance in difficulty between different troubleshooting problems, we split the class into two groups. Each week students did two different troubleshooting problems sequentially in one exercise, where the level increased from the first problem to the second problem. One group did circuit Situation 1 first and the other group did circuit Situation 1 second. In other words, half the class was given Situation 1 at Level 1 and half the class was given Situation 2 at Level 1, and then they swapped situations when they moved up a level (see Figure 3). By splitting the students into two groups, we were able to compare performance on Level 1 and Level 2 for the **same** circuit problem.

Results + Discussion

Data was collected in Fall 2020. Roughly 115 students took the introductory circuits course in the fall, and labs were conducted remotely at home. About 100 students completed the homework assignment each week, leaving about 50 students in each half group. We first compared performance across the various conceptual levels. We looked at whether students correctly identified whether the circuit was working or not and if the circuit was not working, whether they correctly isolated the cause of the problem. The results are shown in Figure 4.

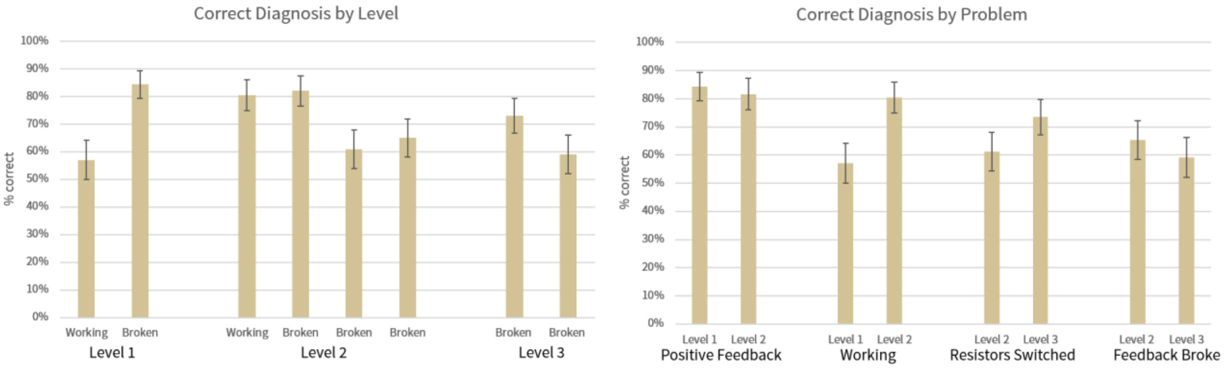


Figure 4. We looked at what percentage of students correctly diagnosed the circuit was working or not working in each situation. (Left) Variance in performance is dominated by the particular troubleshooting situation, rather than how authentic the tool is. (Right) Looking at the four different troubleshooting situations, half the time students performed better on the tool with more realistic representations, and the other half of the time the opposite was true.

Perhaps not surprisingly, our results indicate that differences in performance appear to be dominated by the specific problem with the circuit rather than the conceptual level versions of the tool. This makes sense, since some circuit errors are harder to find than others and some problems like miswirings can be easier to find when presented with a physical layout. More interesting was that our results indicate that performance did not decline as more cognitive levels were introduced. In fact, in some cases, students' performance increased. We are currently collecting more data to better understand this relationship.

The measurement data also provided interesting insight into students' troubleshooting journey, since we were able to track the measurements they took. For the most difficult troubleshooting challenge, the op-amp with the "loose" wire in the feedback loop (Situation 4), 30 students correctly diagnosed the circuit as not working, but only 10 students correctly isolated the problem with the circuit to the feedback loop when asked for their reasoning in follow-up questions. To better understand why some students were able to find the right reasoning, we looked at their measurement data and noticed that all these students measured the feedback loop many times compared to their peers. For example, student 11's final six measurements were clicking back and forth between the two breadboard rows on either side of the loose feedback wire. This makes sense, since from the picture presented, the circuit is wired correctly, and the voltage difference between these points is surprising. Many feedback measurements strongly indicate that the student noticed something was wrong in this region, but it is not sufficient to indicate that they understood the root cause of the issue: 5 students measured the feedback loop at least 4 times without writing the correct cause in the follow-up questions. Since students are so variable in the amount they write in open-ended questions, their measurement data can provide additional insight into their troubleshooting.

As we analyzed our results, we realized that we could make the exercise better for our data collection and the students' learning outcome. Our initial exercise allowed students to exit when they thought they figured out the problem, independent of whether their answer or reasoning was correct. For students who reached a wrong conclusion, we could not track what would happen if they discovered data that contradicted their conclusion, and they lost the learning experience that occurs when data challenges one of your assumptions. Our tool can easily provide the needed feedback, so our current activities include real time analysis of students' answers and provide additional information when they are wrong. These "guardrails" should allow us to better track the thinking process of students who find troubleshooting confusing, allowing us to create better interventions to help their learning.

Conclusion

Finding and fixing errors in an electrical circuit you built is a critical learning task for new circuit designers. We created a tool which helps students practice this skill in a constrained environment. This tool eases the creation of interactive circuit debugging activities, making it possible for students to practice these skills outside of a lab setting and allowing us the possibility to provide feedback to keep students on track. Furthermore, collecting information about students' measurement journeys allows us to better understand students' thought processes while debugging and helps us uncover students' stumbling blocks, which will hopefully lead to better teaching interventions. We have continued to modify the experiments and used this tool in subsequent terms, in efforts to improve the tool and gather more data about how students debug. We are happy to share source code with others who would like to help test out this system. We look forward to sharing additional insights into students' debugging processes in the near future.

Bibliography

[1] A. Price, et al. "A Detailed Characterization of the Expert Problem-Solving Process in Science and Engineering; Guidance for Teaching and Assessment," submitted to *CBE Life Sciences Education*, 2020. Available: ArXiv, https://arxiv.org/abs/2005.11463 [accessed Feb. 6th, 2021].

[2] D. Jonassen and W. Hung. "Learning to Troubleshoot: A New Theory-Based Design Architecture," *Educational Psychology Review*, vol. 18, no. 1, pp. 77-114, 2006.

[3] P. Rivera-Reyes and R. Boyles, "Training in Troubleshooting Problem-Solving: Preparing Undergraduate Engineering Students for Industry," in *2013 ASEE Annual Conference & Exposition, Atlanta, GA, June 23-26, 2013.*

[4] D. Dounas-Frazer, et al., "Investigating the role of model-based reasoning while troubleshooting an electric circuit," *Phys. Rev. Phys. Educ. Res.*, vol. 12, 2016.

[5] J. Ortega-Alvarez, et al., "Exploring Undergraduate Students' Computational Modeling Abilities and Conceptual Understanding of Electric Circuits," *IEEE Transactions on Education*, vol. 61, no. 3, August 2018.

[6] W. Sanchez, et al., "Undergraduate Engineering Students' Representational Competence of Circuits Analysis and Optimization: An Exploratory Study," in *2016 ASEE Annual Conference & Exposition, New Orleans, LA, June 26-29, 2016.*

[7] J. Delvicario, et al., "Promoting Critical Thinking Through Troubleshooting Exercises in Fundamental Electric Circuits Labs," in *2018 ASEE Annual Conference & Exposition, Salt Lake City, UT, June 24-27, 2018.*

[8] R. Ramos, "Guided Modules Emphasizing Process-Based Troubleshooting Techniques Help Below-Average Performing Students Improve Instrumentation Skills," in *2018 ASEE Annual Conference & Exposition, Salt Lake City, UT, June 24-27, 2018.*

[9] I. Gustavsson, et al., "The VISIR project – an Open Source Software Initiative for Distributed Online Laboratories," *in REV 2007.*

[10] Autodesk®, "Tinkercad® Circuits". [Online.] Available: https://www.tinkercad.com/. [Accessed March 2nd, 2021].

[11] DesignWare Inc., "EDISON v5 for Windows—Multimedia Lab for Exploring Electricity and Electronics". [Online.] Available: http://www.designwareinc.com/edison.htm. [Accessed March 2nd, 2021].

[12] Fritzing, "Fritzing—Electronics Made Easy". [Online.] Available: https://fritzing.org. [Accessed March 2nd, 2021].

[13] Qualtrics®, "Online Survey Software". [Online.] Available: https://www.qualtrics.com/core-xm/survey-software. [Accessed March 2nd, 2021].

[14] S. Bell and M. Horowitz, "Rethinking Non-major Circuits Pedagogy for Improved Motivation," in *2018 ASEE Annual Conference & Exposition, Salt Lake City, UT, June 24-27, 2018.*