Deconstruction Kits for Learning: Students' Collaborative Debugging of Electronic Textile Designs

Deborah A. Fields
Utah State University
2800 Old Main Hill
Logan, UT 84322
001 215 7799004
stareyes@gmail.com

Kristin A. Searle
Utah State University
2800 Old Main Hill
Logan, UT 84322
001 215 7799004
kristin.searle@usu.edu

Yasmin B. Kafai
University of Pennsylvania
3700 Walnut Street
Philadelphia, PA 19104
001 215 7463468
kafai@upenn.edu

ABSTRACT

Learning to use a construction kit to design, make, and program electronic textiles (e-textiles) has been found to be a rich context for students' learning of crafting, engineering and programming. We propose the development of what we call a 'deconstruction' kit—the design of faulty e-textile artifacts that students need to de- and reconstruct—as an alternative to gain insights into students' learning. We designed e-textile projects with strategically poor crafting, non-functional circuitry, and insufficient coding to investigate high school students' understanding of coding, circuit design and creation (through sewing) with the LilyPad Arduino. We videotaped and analyzed ten students collaborating in pairs as they engaged in debugging, or fixing, various problems in provided e-textile artifacts. Our findings indicate that these deconstruction kit projects are not only promising tools for evaluating students' understanding of e-textiles but can also become valuable learning tools on their own, especially when peer collaboration is taken into account.

CCS Concepts

• **Social and professional topics → Computer science education** • **Social and professional topics → Computational thinking**

Keywords

Maker movement, debugging, electronic textiles

1. INTRODUCTION

Driven by the availability of low-cost hardware, digital fabrication tools, and open source software, the Maker Movement has provided a rich context to engage students in crafting, engineering, and computing [6]. As these maker activities increasingly move into formal classroom spaces, one major issue is how to authentically assess what students are learning through making. The challenge here is to understand and assess students' learning in making a complex design such as an electronic textile (hereafter e-textile) [4] that involves multiple domains—such as crafting, engineering, and computing—and tools both onscreen and offscreen while taking into account the collaborative, often distributed nature of learning in maker communities.

To address this challenge, different solutions have been proposed. Some have developed tests of specific aspects of knowledge used in maker designs, for instance assessing knowledge of simple, series, and parallel circuits in e-textiles through design tasks with paper and stickers that use familiar parts [10]. However, this type of test only covers one aspect (circuitry) of the distributed learning that happens when students are making designs that have physical, electrical, and computational components. Others have developed more generalized survey-based assessments that study students' changes in confidence and performance with various maker types of technologies (such as microcontrollers, laser cutters, 3D printers, see [2]). While this survey can be used in many different fabrication technology settings, it lacks the specificity needed to more deeply understand what students are learning when they engage in complex design tasks.

We propose the development of a deconstruction kit in which students fix, or debug, strategically built-in problems as a potential solution to the assessment issue. While such screen-based code debugging tasks are popular in computer science education (e.g., [1]), these have rarely been used for artifacts that have both physical and digital components. In this paper we discuss the design and testing of a deconstruction kit focused on assessing students' understanding of coding, circuitry, and crafting (through sewing) in the context of e-textiles activities made with the LilyPad Arduino construction kit [3]. In the design of our deconstruction kit, we targeted knowledge and dilemmas identified in prior research of students' struggles with multiple dimensions of learning with e-textiles [7]. Using lessons learned from the assessments of maker design tasks described above, the deconstruction kit was grounded in tools and materials familiar to students yet presented an original situation to assess their ability to fix mistakes. We videotaped 10 students collaborating in pairs as they worked to turn on LED lights in a project strategically designed with poor crafting, non-functional circuitry and insufficient coding. Our analyses focused on the solutions generated by each pair and how the collaborative setting of the debugging task influenced student learning. In the discussion we address how the design of such a deconstruction kit can become both a promising tool for evaluating students' learning of e-textiles and a valuable learning tool, especially when peer collaboration is taken into account.

DOI: http://dx.doi.org/10.1145/3003397.3003410

2. BACKGROUND

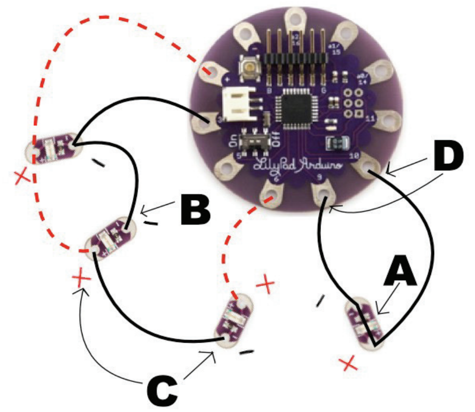
In designing the deconstruction kit, we built on prior research that focused on debugging as a key computational practice that all programmers, novices and experts alike, have to engage in when designing software (e.g., [11]). Because debugging involves implementing many new skills at once, it is challenging for novices. Being able to read and debug code written by someone else is especially difficult and does not necessarily coincide with one's ability to write code [9]. Research shows that students' success or failure in screen-based debugging often relates to their conceptual knowledge of what the program is intended to do and their ability to attend to program state (e.g., [1]). The recent increased interest in bringing computing back into K-12 education has caused computer science educators to return to helping students become more proficient in debugging their programs.

However, extending debugging beyond the screen makes things more complicated for learners who must not only attend to a variety of code-based errors, but also physical errors such as incorrect placement of a sensor on a robot or incorrect circuitry in an e-textiles project. Identifying, debugging, and solving these problems is at the crux of being able to design functional computational and engineering projects. For instance, Sullivan [12] argues that solving functional design problems helps students develop intricate inquiry skills that include an iterative feedback loop of observation, hypothesis generation, hypothesis testing, and evaluation of solutions. She analyzed individual students debugging a carefully designed set of robotics dilemmas that, combined with pre- and post-tests, demonstrated how robotics activities elicited the thinking and science process skills associated with scientific literacy (observation, estimation, hypothesis generation, etc.) and improved students' system understanding.

In designing an e-textile deconstruction kit, we built on Sullivan's research examining debugging of robotic design by moving into the domain of electronic textiles materials and by making the debugging a collaborative activity where students worked in pairs. In our analysis, we address the following two research questions: (1) In what kinds of thinking and process skills did students engage while debugging e-textile problems? and (2) How did the collaborative task contribute to students' learning through debugging an e-textile?

3. METHODS

Based on observations from two prior workshops (see [7]), we had a fairly good understanding of different types of mistakes and challenges common to novices in e-textiles. Building on our own prior work creating problem-embedded non-computational e-textile projects as learning tools to encourage students to deconstruct circuit designs [5], we developed more advanced versions to examine students' problem solving in e-textiles: including crafting and computational components. We thus physically sewed and programmed a set of five projects in the deconstruction kit that, while technically identical, looked aesthetically different so as to avoid students comparing solutions to the problems in circuit design and coding (see Figure 1). Each project presented three debugging challenges related to circuitry design (i.e., engineering challenges), and three challenges related to coding the e-textile artifact.



Circuit challenges:
1. Short circuits: Short circuits with overlapping conductive threads; also at point **A**, where LED4's positive and negative ends are sewn together.
2. Electronic topology (series versus parallel circuits): At point **B**, LED2 is in parallel with LED1, but should be independent with LED1 (sharing either a positive or negative connection but not both).
3. Polarization: LED3's polarity is reversed in point **C.** Both circuitry and code need adjusting.

Coding challenges:
4. Constant versus variable pins: Students need to distinguish coding LEDs connected to a constant polarization pin (e.g., LED1, 2 & 3) and coding LEDs connected to variable polarization pins (e.g. LED4).
5. Control flow: Students must code one end of an LED constant while programming the other end to turn the LED on and off. At point **D** in Figure 1, LED4 needs unique blink code compared to the other LEDs since both of its respective pins are variable.
6. End-state definition: The task required a clear understanding of the end-state for the code. The students were challenged to figure out that the code had to command the LEDs to turn off

Figure 1. Electronic textile deconstruction kit using flying pig image (top); Circuit diagram schematic using LilyPad Arduino and 4 LEDs (middle) with dashed lines positive and solid lines negative; and Bug listing and explanations. ©Deborah Fields

Figure 2. Three variations on electronic textile deconstruction kits using a removable LilyPad Arduino. Though aesthetically distinct, each deconstruction kit has identical circuitry and code problems. ©Deborah Fields

Five pairs of students, who had participated in at least one of a series of previous e-textiles workshops, completed the activity during their one-hour lunch period. The pairs were composed of students of varying ability levels and genders (2 male-male pairs, 1 female-female pair, and 2 male-female pairs). During the hour-long task, students were provided with the deconstruction kit (see Figures 1 and 2) and code that would turn on each of the four lights. Each pair was instructed to make each of the four LED lights blink independently of the others and asked to think aloud. During the last 15 minutes of the hour, the researcher provided select hints if the students were stuck on a particular problem. Each debugging session was video recorded and then logged, resulting in a close but not word-for-word transcript of what students said. Drawing on Sullivan's (2009) robotics-based coding scheme, we focused our analysis on which problems pairs solved with and without hints and what thought processes they used.

4. FINDINGS

All student pairs solved most of the problems in the deconstruction kits, though no group solved every problem in the time allotted without help. Since we purposefully designed the deconstruction kit projects to be on the edge of students' capabilities, this was a promising finding. Students struggled the most with the programming, especially with control flow and end-state definition (see Table 1).

Table 1. Number of groups solving each problem, with or without hints.

Challenge	Solved, no hints	Solved with hints	Did not solve
1. Short circuits	3	2	0
2. Electronic topology	1	3	1
3. Polarization	5	0	0
4. Constant vs. variable pins	4	0	1
5. Control flow	0	1	4
6. End state definition	1	4	0

One reason for this might be that the students had very limited experience with coding because they spent most of the workshop time on making their projects by crafting designs out of felt and designing and sewing their circuits. Less time was spent on writing code. However, analysis of students' debugging of Scratch programs suggests that attention to state is a difficult concept for students to grasp [9]. Furthermore, students strategically isolated and prioritized the order of problem solving, an important strategy since more than one issue could cause a light not to work (e.g., bad circuitry design combined with poor coding). In many instances the pairs laid out multiple issues and carefully chose the order in which to work through them in order to isolate one problem at a time. Finally, students also had to select and appropriately use the right tools to fix problems including: multi-meters, needles, specific scissors or seam rippers, and available pins on the LilyPad Arduino. Debugging an e-textile with problems not just in the program code but also in the circuit design, or in crafting turned out to be a complex challenge for all students, still engaging them in multiple cycles of observation, hypothesis generation, hypothesis testing, and evaluation of their solutions in the process observed by Sullivan [12].

Unlike most debugging tasks used for individual assessments, we engaged students as teams with the e-textiles deconstruction kit, which produced two unanticipated learning benefits. First, students were able to assist each other: exchanging ideas, answering each others' questions, and catching each others' mistakes. Second, in generating hypotheses about the bugs and evaluating their solutions, students had to justify their reasoning, explaining why they thought something was a problem or why a solution worked. This forced them to make their own circuit and programming knowledge more explicit. These justifications also provided insight into some of students' otherwise hidden learning processes. For instance, Aaliyah identified short circuits caused by loose threads on the back of the project but her partner, Saul, did not understand why that was an issue. Aaliyah explained, "[I]t like shorts out the thing…I remember [instructor] telling [another student] that they needed to be careful about how long the length of the string is at the end, 'cause when they cross it won't work." Here Aaliyah explained how crossed positive and negative threads could short a circuit. Interestingly, she learned this through overhearing an instructor help another student in the workshop. Revoicing what she heard, she was able to think through and explain the issue to her partner.

5. DISCUSSION

The analysis of the e-textile deconstruction kit provided insights into what students understood and the thinking processes they utilized in solving problems that involved the overlapping areas of knowledge (crafting, circuitry, and coding) in designing e-textiles. In our case, students struggled the most with certain areas of coding (control flow and end-state definition), allowing us to adapt future iterations of workshops to teach these concepts better and provide more time for learning them. In solving problems students exhibited several key scientific thinking processes, including multiple iterations of observation, hypothesis generation, hypothesis testing, and evaluation of solutions. Beyond other examples of debugging assessments that take place solely on-screen or with robotics kits, we noted that students had to work to isolate overlapping problems. In particular, the need to work with the physicality of the projects made prioritizing and thinking through the order of solutions an important step in the problem solving process. This supports conclusions drawn from the analysis of clinical interviews with college-age students relatively advanced in e-textiles design [8]. There the authors identified "crafting" as a key area of cognitive work on par with coding and circuitry design in being able to think through how to spatially and physically adapt a project to meet certain constraints. The academic work connected to crafting also goes along with our findings regarding the importance of connecting knowledge and skills with a range of domain specific tools to solving problems in this area.

Completing the debugging task collaboratively provided deeper insight into students' thinking processes as they justified and argued for certain problems and solutions. It also allowed students to teach each other and contributed to the task not only as a summative assessment of understanding but also as a formative assessment. Furthermore, engaging in collaborative debugging allowed students to solve problems they might not have been able to solve individually given that students with lesser understandings of coding, circuitry and creation tended to design simpler projects themselves and thus not encounter all of the problems provided in the deconstruction kit. Indeed, many students reported that they were more confident in their understanding of e-textiles after solving the problems embedded in the deconstruction kit. These reflections open up the potential of deconstruction kits as key learning tools in makerspaces, FabLabs, and e-textiles workshops. Future research is needed to explore the possibilities of deconstruction kits with different types of fabrication beyond e-textiles and with more advanced students.

6. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation (CIS-0855868) to the third author. The views expressed are those of the authors and do not necessarily represent the views of the National Science Foundation, Utah State University, or the University of Pennsylvania.

7. REFERENCES

[1] Ahmadzadeh, M., Elliman, D., & Higgins, C. (2005). An analysis of patterns of debugging among novice computer science students. In, *Proceedings of ITiCSE'05* (pp. 84-88). Monte de Caparica, Portugal: ACM.

[2] Blikstein, P., Kabayadondo, Z. & Martin, A. (under review). The exploration and fabrication technology (EFT) index: Assessing unexplored technology and engineering literacies in makerspaces and educational FabLabs.

[3] Buechley, L. & Eisenberg, M. (2008). The LilyPad Arduino: Toward wearable engineering for everyone. Pervasive Computing, 7(2), 12-15.

[4] Buechley, L., Peppler, K., Eisenberg, M., & Kafai, Y. B. (Eds.) (2013). *Textile Messages: Dispatches from the World of Electronic Textiles and Education.* New York, NY: Peter Lang Publishers.

[5] Griffin, J., Kaplan, E., Burke, W. Q., & Kafai, Y. B. (2011, March). A Deconstruction Kit for the LilyPad Arduino: Designing Debugging Sets for Learning about Circuitry & Programming for High School Students. Presented at *SIGCSE 2011,* Dallas, Texas.

[6] Halverson, E. R., & Sheridan, K. (2014). The maker movement in education. *Harvard Educational Review*, *84*(4), 495-504.

[7] Kafai, Y. B. Fields, D. A., & Searle, K. A. (2014). Electronic textiles as disruptive designs in schools: Supporting and challenging maker activities for learning. *Harvard Educational Review*, 84*(4)*, 532-556.

[8] Lee, V. R. & Fields, D. A. (2013). A clinical interview for assessing student learning in a university-level craft technology course. In the Proceedings of FabLearn 2013, Palo Alto, CA.

[9] Lewis, C.M. (2012). The importance of students' attention to program state: A case study of debugging behavior. In *Proceedings of the International Computing Education Research Association Conference (ICER'12)* (pp. 127-134). Auckland, New Zealand: ACM.

[10] Peppler, K. & Glosson, D. (2013). Stitching circuits: Learning about circuitry through e-textile materials. *Journal of Science Education and Technology, 22*(5), 751-763.

[11] Soloway, E. & Spohrer, J.C. (Eds.). (1989). *Studying the novice programmer*. Hillsdale, NJ: Lawrence Erlbaum.

[12] Sullivan, F. R. (2009). Robotics and science literacy: Thinking skills, science process skills and systems understanding. *Journal of Research in Science Teaching*, 45(*3*), 373-394.