Chapter 6

Making Connections Across Disciplines in High School E-Textile Workshops

Yasmin Kafai, Deborah Fields, and Kristin Searle

Introduction

At the end of the last day, John reflected on the process of making his first electronic textile project: "I didn't know it took all this to light stuff up."

Most of today's technology designs make invisible or blackbox what makes them work. For everyday purposes, not knowing how your computer or software works might well be appropriate. Yet for educational purposes visibility might be more beneficial in promoting better understanding of functionality and access to computing (Buechley 2010; Eisenberg et al. 2006). In working with e-textiles, the process of making technology visible is both simple and complex at the same time. As the comment by John, a 14-year-old high school student, illustrates, it's difficult to know what it takes to make lights work—simple at first sight but more complex upon closer inspection.

Working with e-textiles involves the multiple disciplines of computer science, engineering, and the arts as designers engage in crafting, coding, and circuitry that, as Ngai, Chan, and Ng argue in Chapter 2, can be difficult, if not cumbersome to understand for those new to e-textiles. Further, projects with multiple types of designs of circuitry, coding, crafting can go wrong in many ways (Resnick, Berg, and Eisenberg 2000). Identifying, debugging, and solving these problems are at the crux of being able to design functional e-textile designs. Indeed, Sullivan (2008) argues that solving functional design problems helps learners develop intricate inquiry skills that include an iterative feedback loop of observation, hypothesis generation, hypothesis testing, and evaluation of solutions.

At the same time, the fabrication of stitches, circuits, and codes reveals the underlying structures and processes in tangible and observable ways that, we argue, are crucial to students' learning. When asked about the circuitry, crafting, and coding components of e-textiles, Tamieka, a 14-year-old African American girl, reflected, "I think it's good because...it takes three things and combines it. And it shows how like, you might think it's pretty hard at first, but when you do it, it's really easy...I learned, like, even more than what I knew before about it" (May 27, 2011, Interview). In other words, the multiple components that make up e-textiles render visible how technology is designed and built in ways that add to, rather than detract from, learning for the novice designer.

It is precisely these intersections between crafting, circuitry, and coding that make visible the challenges and opportunities for learning with e-textiles. While learning how to sew, how to make functional circuits, and how to write code that lights up LEDs and controls sensors are each important and valuable practices in their own right, learning how to see the connections between these parts is what makes learning to design e-textiles greater than its individual parts. In the following sections, we first describe the context of our e-textile design activities and then provide vignettes from a series of workshops with high school students as illustrations for making learning and, by extension, technology visible.

The Design of E-Textile Workshops for High School Students

We organized a series of three workshops ranging in length from 4 to 6 weeks in partnership with a local science museum that hosted after-school classes for a high school. Workshop sessions were held once a week and lasted about two hours. During this time period, students learned how to design and create their own e-textile projects, beginning with aesthetic drawings, followed by circuit schematics, sewing/crafting of designs with the LilyPad Arduino or LilyPad Simple Board, and programming. Ultimately the process was less linear than this description makes it seem. As students gained experience and knowledge, their projects tended to increase in complexity and they experienced greater success.

Participants included 35 freshmen, 14–15 years old, from a public magnet high school focused on science and technology in a large urban school district. The students' self-identified demographic makeup was 23% African American, 29% Caucasian, 14% Asian, and 17% mixed race/ethnicity. Five students chose not to identify their race/ethnicity in survey responses. Just under half of the participants were girls (n = 15). Overall, the demographics of the workshops reflected the diversity found in the school and district at large. In spite of students' interests in science and technology, only a few of our participants had prior programming experience, and none had ever worked with electronic textiles when they elected to participate in the workshops.

Although the workshop description above appears to breakdown learning e-textile design into its constituent parts (crafting, coding, circuitry), it is quite difficult to separate out these components in practice. In fact, where students were the most challenged and learned the most was when domains intersected in ways that were not always in keeping with their prior knowledge of the individual domains. In our analysis we have coded these learning moments as intersections between crafting/

circuitry, circuitry/coding, crafting/coding, and, finally, debugging, which involves all three domains (Kafai, Fields, and Searle 2012). We provide three examples of what takes place when high school youth engage with intersecting domains and how learning about crafting, coding, and circuitry becomes visible. Our first example occurred at the junction of crafting and circuitry.

Connections Between Crafting & Circuitry

One of the first things students in our workshops struggled with were unique attributes of sewn circuits that lie at the intersection of sewing and circuitry. Sewing for conductivity is not the same as traditional sewing. At the most elementary level, students learned that they must sew an electronic component to ensure *conductivity* as well as stability. For instance, if an electronic component like an LED is sewn too loosely or the stitches are too large, the connection between the component and the conductive thread may be lost when the textile bends. Knot-tying reveals another layer of this connected relationship: in order to direct electricity to flow through an LED one must consciously tie a knot to end the positive connection on one side of the LED, cut the thread, and start a completely new line of thread on the negative

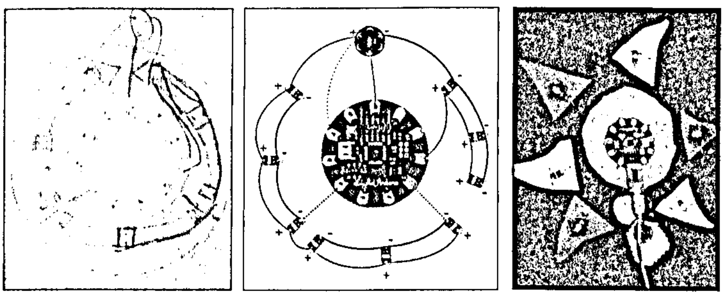

Fig 44 Patrick's blueprint (L to R): schematic diagram of his LilyPad design, and final sewn design (simplified with only 6 LEDs)

side. In the following example, Patrick tried to work out where he would have to cut the thread and make knots when he actualized his design (Figure 44) and solve his dilemma of how to connect two negative lines together.

Sewing for Conductivity

Patrick (student) and Deborah (researcher) work on the blueprint of Patrick's e-textile design. Deborah has just pointed out that since he is connecting the negative sides of several LEDs, he does not need to cut the thread at each point. When Patrick refers to "looping" he references the idea that a piece of conductive thread must be sewn through an electronic component three times to ensure conductivity and stability.

Patrick: So you don't have to loop it every time.
Deborah: That's right. Well you still need to loop it because of the electrical connection...
Patrick: Every time, okay (nodding, staring at the blueprint)

Deborah: Because if it's just the thread, the thread is pretty thin, right? ... So the more times you wrap it, there's more electrical connection there...
Patrick: Okay...

Tying Knots to End Electrical Connections

Deborah: Draw me the other places where you would need knots.
Patrick: Here, here, here, here, here. Oh yeah, definitely here. The one... What's up with that?

After pointing out where one would have to cut the thread and tie knots, Patrick asks what to do when he connects two negative lines. (February 16, 2011, Video)

One intersection between crafting and circuitry was evident when we asked students to "sew three times through," what Patrick referred to in the above vignette as "looping." Unlike "hard" electrical connections made by soldering two wires together, where the metal provides the needed electrical link between the wires, conductive thread serves as both wire and soldering material in e-textiles design. Thus, it is necessary to sew a component like an LED or a port on the LilyPad "three times through" in order to assure a significant electrical connection as well as stability. Youth in our workshops, however, often thought that sewing through a component only once was sufficient because this attached the component to the fabric backing. In the above excerpt, Patrick noticed that he could connect multiple negative lines together without having to make extra knots. Yet Deborah pointed out that he still needed to "loop it," reminding him that he still had to "sew three times through" for a strong, functional electric connection.

This same vignette also shows how Deborah asked Patrick to think through a sewable design by asking him to point to where he would need knots. This encouraged Patrick to think through the beginnings and endings of each positive and negative line. Many of the youth we worked with had no prior experience with sewing or circuitry, and simple things like threading a needle or tying a knot (regardless of its conductive purposes in e-textiles) were quite difficult for them. Indeed, we found that knowing where to tie knots and then actually tying the knots was one of the most dreaded moments of creating e-textiles for our youth. As Patrick began to sew, he needed knots "here here here here here...." He also realized that he could connect two negative lines and was not sure how to manage that ("What's up with that?"). This question provided an opportunity for Patrick to learn that he could simply connect two negative lines (see middle diagram of Figure 44 for connections between positive sides and between negative sides). This demonstrates his beginning understanding of how crafting and circuitry intersect in e-textile design and what he had to do in order to have a working project. These instances of Patrick struggling with the connections between crafting and circuitry illustrate some of the complexities of teaching e-textiles to youth.

Connections Between Circuitry and Coding

Another site of confusion occurred when students experienced an unexpected intersection of circuitry and coding. As youth began to understand the affordances of the LilyPad and how they wanted their projects to blink, they often realized they needed to change their circuitry designs to afford more flexible coding. Sometimes this meant using fewer parallel circuits because lights sewn in parallel could not be programmed individually, meaning they always turn on and off at the same time. In another example, a 14-year-old girl, Amari, faced a dilemma because she had connected one LED to both the positive and negative ground pins on the LilyPad.

Amari programmed her lights to turn on, making them blink. The LED that she had connected to the + and the – pins of the LilyPad stayed on continuously. "I know that you told me it would always be on and I did it anyway, but I want to re-do it," Amari said. Then she cut off the positive line of that LED and re-sewed it to a different, programmable pin (#3) on the LilyPad all on her own. (May 25, 2011, Field note excerpt).

Here, we see coding and circuitry intersecting with one another in Amari's learning process as she shifted from having a light that was always on (connected to the positive and negative grounds) to a light that was connected to a programmable port (Figure 45). She then proceeded to program the lights to fade in and out—an effect that would not have been possible without her re-sewing the circuits.

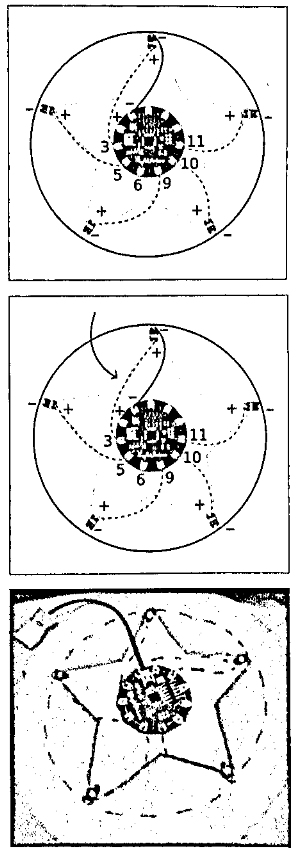

Fig 45 Changes in Amari's sewn designs (T to B): she connected the top LED to the hardwired positive and negative pins of the LilyPad; she re-sewed the top LED to the #3 pin to computationally control it; her final, sewn project

Though it was frustrating to us to watch students make decisions that would limit their designs despite our admonitions, we were surprised at how many students were willing to re-design their projects, some going to great trouble to do so, when they finally understood why certain circuit designs would not allow for the programmed lighting effects they desired. Though Amari did not initially realize the full implications of connecting an LED to the hardwired pins until she had sewn and programmed her project, when the implications of her circuitry design on her programming capabilities registered and she went to the trouble of undoing and redoing part of her project so that there were greater affordances for programming. It was truly a process of learning *through* design.

Problem Solving: Is it the Code, the Circuits, or the Crafting?

In the end, when projects had been designed, sewn, and coded, inevitably some component of the project caused it to fail. When this happened youth were faced with identifying the problem. Was it their code, their circuitry design, or their sewing? Because the underlying issue was rarely clear, the final step of problem solving, or "debugging," involved all three domains, and many youth like Nathaniel and Bailey had to test, identify, and solve the problems. Describing his finished project and summing up the most difficult parts, Nathaniel commented:

> Nathaniel: So it goes in a wave from like blue [LED] on, white on, blue off, red off, white off: like a wave. Stops and then starts over again... I thought it would look really cool and it does.
>
> Researcher: What went wrong and how did you fix it?
>
> Nathaniel: There was a couple of, like, wires that were crossing, and on the back I had to tape it down once and on the front I had to cut some of the yarn off, but that was it.
>
> Researcher: So what advice would you give to someone?
>
> Nathaniel: I would check the circuits before I jumped to conclusions that it's not working. 'Cause that's what was happening. (December 15, 2010, Video)

As Nathaniel discussed in the vignette above, when he finished his project and tried to program the LEDs to blink, they did not turn on. What was the problem? He thought he had programmed the LEDs correctly, but he did not know whether the difficulty was in the circuit design, the sewing, the coding, or some combination of the above. Then he discovered that some of his conductive thread was causing short circuits both on the front of his project and on the back. This project malfunction involved both Nathaniel's sewing and circuitry. His knots were messy with long loose ends, resulting in touching threads that caused a short circuit. Merely having messy knots might have been seen as an aesthetic eyesore on a sewing project, but with conductive sewing the consequences were much more severe—a project that did not work. Nathaniel solved this by taping down (insulating) some of the knots in the back and trimming loose threads on the front. Once this was done, his programming worked and the lights blinked in the order that he wanted, turning each light on, then each light off in turn. However, until he had solved the short circuit problem, he could not identify whether his coding worked. This is why at the end, Nathaniel advised others to first check their circuitry before assuming that the project did not work, essentially suggesting testing elements one by one to figure out the underlying problem.

Bailey also had difficulty getting her lights to turn on, but for slightly different reasons. She used a simple design to create a robot with a heart whose eyes blinked (Figure 46). She came to the final day of the workshop with the entire project sewn but despite efforts to code it, she could not make the lights turn on. With the help of a researcher, she first checked the back of her robot, saying "Uh oh," when she saw threads lying helter skelter and crossing each other, resulting in short circuits). She proceeded to trim the threads and taped one knot down. Then she plugged it into her computer and remixed the code. The project blinked, and she waved her hands up and down in excitement as it lit up for the first time. However, when she disconnected the robot from the computer and put in a battery, she discovered that it no longer worked. Together with the researcher she tested the positive, then the negative connections to the battery, and even tested it with another unused battery and battery holder, but it did not work. With a grimace, she ripped out the stitches to the battery and re-sewed them. Finally her robot design worked.

Like many others, Bailey had to examine her crafting, circuitry, and coding. In this case her design was functional, but her sewing needed trimming and, as concerned the battery, re-sewing. She tested the project repeatedly to isolate multiple problems: testing first with the computer as power source, second with the battery as power source, then further testing individual connections to the battery and alternate materials.

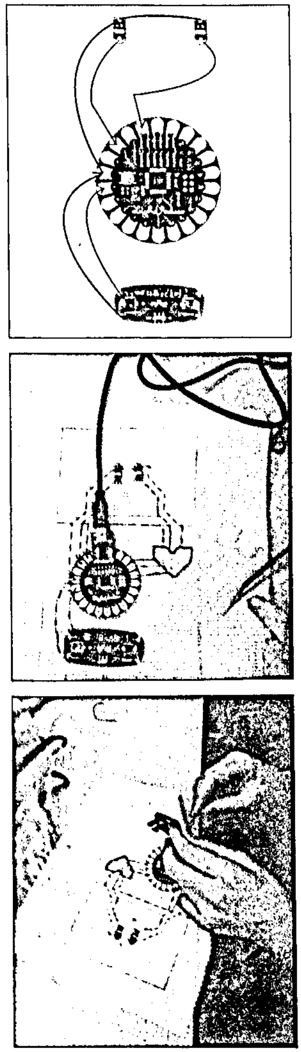

Fig 46 Three pictures of Bailey's Robot with a Heart (T to B): A schematic diagram of her blueprint, her final coded project, and Bailey re-sewing the battery to fix a circuit connection

Making Connections Visible: Lessons Learned

These examples highlight some of the many ways that the intersecting domains of crafting, coding, and circuitry challenged students' pre-existing knowledge and their expectations. One goal of our investigation was to understand how making technology visible could be beneficial in students' learning with e-textiles designs. In order to examine this aspect, we needed to understand first the particulars of how students approached the connections between crafting, circuitry, and code. Most of their problems fell into two areas: creating circuits with fabric and thread and programming physically laid out circuits. Remarkably, both of these broad categories sit at the juncture of the conceptual and the physical, such that the material qualities of e-textiles made visible the conceptual issues of electronics and programming and added an aesthetic dimension (Fields et al. 2012). In other words, the infusion of crafting into the domains of electronics and code made visible the inner workings of circuits and programming. Thus it was the initial move from circuit diagrams to actual sewn circuits and the subsequent move from abstract concepts of code to coding physical circuits embedded in cloth that held both the greatest challenge and, as we argue, some of the greatest opportunities for learning.

Knot-tying became a way to think about connections between positive ends or ways to direct electricity from thread into an LED. Choosing which LilyPad pins to connect one's LEDs involved not just selecting a pin that was close to where one wanted the LED but also considering whether that pin had the desired computational affordances. Finally, when projects had been fully sewn, programmed, and connected to a power source, there could still be several things wrong. In fact, it appeared that youth learned the most when they had to debug their own projects. Some young people, troubleshooting at the intersection of crafting and circuitry, had to rip out sewing where they had accidentally connected a positive to a negative line. Others worked laboriously on debugging their code by looking for missing semi-colons, brackets, or LilyPad pins they had not properly turned on as outputs. Testing, isolating, and fixing problems were excellent opportunities for youth to apply their new knowledge of e-textiles.

Thankfully, by the time youth got to debugging, they had, for the most part, invested eight or more hours in their e-textile projects and felt a sense of ownership. This sense of ownership helped them to persevere through solving difficult debugging challenges that introduced young e-textile designers into thinking about the interdependencies of individual parts in making visible the larger functional system. Indeed, we have even made debugging challenges into assessments to formally test and promote youths' understanding of particular concepts at the intersections of crafting, circuitry, and coding (Fields et al. 2012; Kaplan et al. 2011).

We see these findings as evidence that making technology visible can promote learning (Buechley 2010; Eisenberg et al. 2006; Resnick, Berg, and Eisenberg 2000). Rather than further simplifying e-textiles design for beginning designers, we argue for maintaining the complexity while scaffolding learning through supports such as pre-designed project blueprints and/or pre-sewn projects that can be coded by students, both of which we used during subsequent workshops with great success. Far too often technology designs are purposefully hidden away, which foregrounds the consumption of technology rather than the production of it (Jenkins et al. 2006). Of course, intentional invisibility or blackboxing can also be a strategic part of learning by highlighting some aspects for students while hiding others, focusing their attention on things they most need to learn, as the programming platforms Scratch and ModKit do by taking out minutiae in coding that often distract students (like punctuation and spelling). In the end, decisions on what to render visible or what to leave invisible are thought-provoking tasks for educators as we seek to challenge and scaffold students' learning.