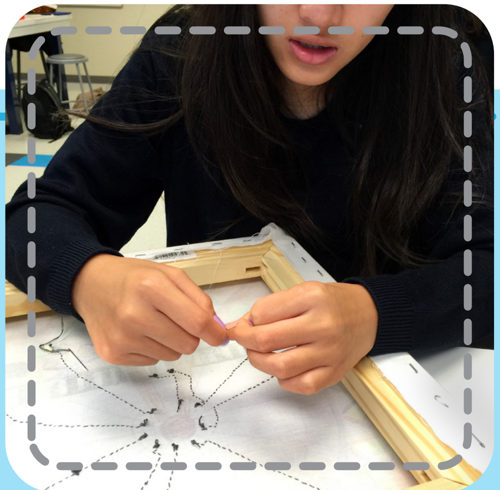
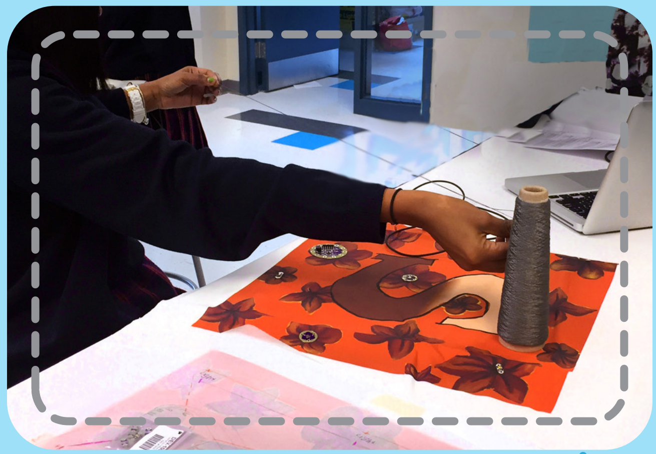
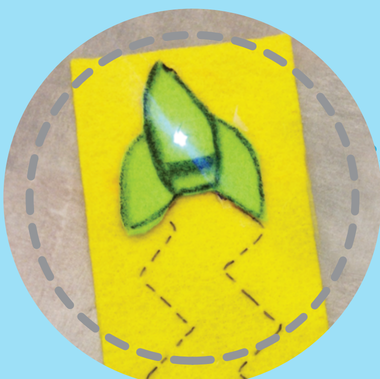
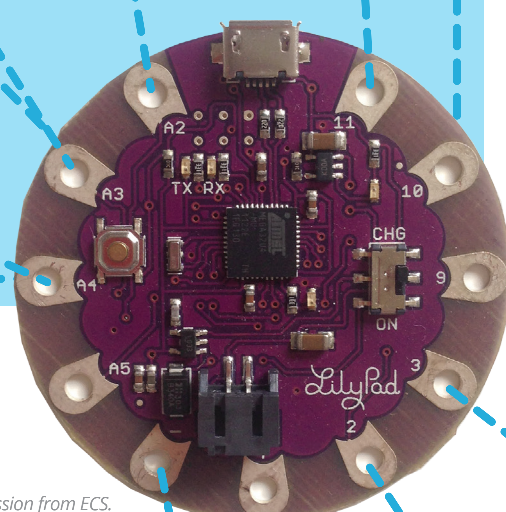
E-textiles

Unit 6 Alternate Curriculum

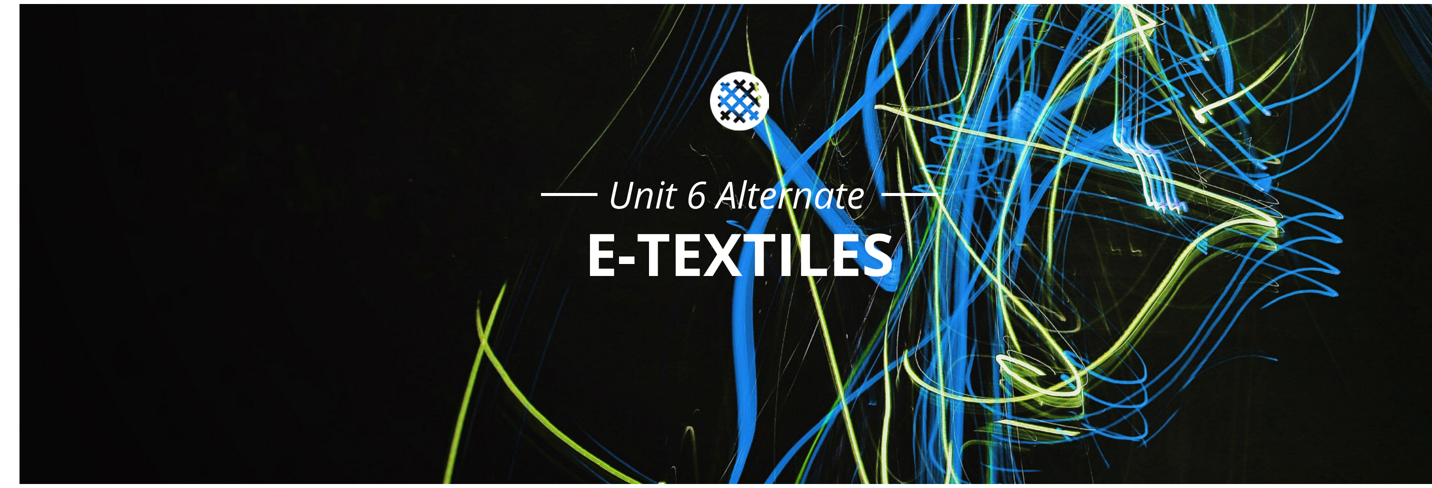








Fields, D. A., Landa, J., Nakajima, T., Kafai, Y. B., Goode, J., Margolis, J. & Ottina, J. (2018). *Stitching the Loop: An Electronic Textiles Unit in Exploring Computer Science.* Exploring Computer Science. Available at http://exploringcs.org.



Introduction

Welcome to our unit on electronic textiles (e-textiles) for Exploring Computer Science. This unit is intended to serve as an alternate to either Unit 5 (Data) or Unit 6 (Robotics). From 2015-2018 we designed, researched, and revised the unit until it we felt that it was robust. Results from our third year of study with 15 teachers show positive pre/post results in students' sense of competency with computer science, interest in computer science, and creativity with computer science (see below).

On our websites we have a number of additional resources for the curricular unit, including the accompanying technical guide and help in identifying material and computational resources needed for the unit. There is also an introductory video on the e-textiles unit. Be sure to check them out!

Some general guidelines for teaching the unit

First, ***make all the projects yourself.*** Get help from friends, parents, and forums where you need it: few people have all the underlying skills of sewing, circuitry, and coding in Arduino that are needed for the e-textiles unit. We say this from experience. When we have seen people not complete projects before teaching them to students, they tend to teach things that are incorrect and have a very frustrating experience.

Record the mistakes you make as you create the projects. This is to develop your own knowledge & to develop troubleshooting skills as well as to understand what students will experience.

Understand some of the design principles behind the unit. We have written them out succinctly in this paper[1]. Our early research into this unit has showed that it can help improve students' engagement with CS — their sense of being creative and personally expressive, their feeling of being able to identify and solve problems, and their overall interest in CS. Solving the many problems that come up in making e-textiles is an opportunity for greater learning! The portfolios in the unit can help students meta-reflect on this process of learning through challenges and revisions and on who they themselves are in the burgeoning world of CS.[2] References for this research is below, and we will continue to update our website with additional research findings as they are published.

1. Kafai & Fields, 2018.
2. Fields, Shaw, & Kafai, 2018.

When you teach the unit, reflect after each lesson and write down things to remember for next year. This is part of applying the principle of iteration to your own teaching!

Recommended timeline

The unit should take around 10 weeks, but in the first year of teaching the unit it often takes 2-3 weeks longer. You will want to leave extra weeks (12-13 weeks total) the first time you teach it.

6 months before the unit begins:

- Skim the curriculum and the technical guide.
- Plan time to MAKE ALL FOUR PROJECTS YOURSELF *before* you start the unit. (We cannot emphasize the importance of this enough).
- Meet with your administration to set up a purchase order for materials. See Materials Guide for costs and recommended materials. Plan to order some supplies as early as 3 months in advance to allow time for manufacturing in case materials are out of stock.
- Meet with the tech leader at your school and decide on the coding platform [link to coding platform guide] that will work best for you. Plan a timeline for installing software and apps, and getting all permission taken care of.

3 months before the unit begins:

- Place orders for materials
- Make the projects. Some projects take 8-10 hours, especially your first time.

1 week before the unit begins:

- Check coding platform on all student computers. Meet with tech administrator for any necessary permissions required.

A note on Journals and Design Notebooks

Throughout this curriculum we refer to both Journal entries and Design Notebook entries. Journal entries are similar in style to journal questions through the ECS curriculum. Design Notebook entries are more focused on elements of designing or reflecting on the design of e-textiles. We recommend the Design Notebook entries go in an actual notebook because this will occasionally include things that need to be drawn out, like sketches and circuit diagrams. Journal entries can also go in the notebook, or if you have done journal entries in a different way throughout the year, you can stay with your earlier practice.

Please cite this work as:

Fields, D. A., Landa, J., Nakajima, T., Kafai, Y. B., Goode, J., Margolis, J. & Ottina, J. (2018). *Stitching the Loop: An Electronic Textiles Unit in Exploring Computer Science.* Exploring Computer Science. Available at http://exploringcs.org.

References

Kafai, Y. B., Fields, D. A., Lui, D., Walker, J. T., Shaw, M., Jayathirtha, G., Nakajima, T., Goode, J., & Giang, M. T. (2019). Stitching the loop with electronic textiles: Promoting equity in high school students' competencies and perceptions of computer science. SIGCSE 2019, Minneapolis, MN: ACM.

Kafai, Y. B., & Fields, D. A. (2018). Some reflections on designing constructionist activities for classrooms. In V. Dagiene & E. Jastuė, *Constructionism 2018: Constructionism, Computational Thinking and Educational Innovation: conference proceedings*, Vilnius, Lithuania, pp. 606-612. Available at http://www.constructionism2018.fsf.vu.lt/proceedings.

Fields, D. A., Shaw, M. S., & Kafai, Y. B. (2018). Personal learning journeys: Reflective portfolios as "objects-to-learn-with" in an e-textiles high school class. In V. Dagiene & E. Jastuė, *Constructionism 2018: Constructionism, Computational Thinking and Educational Innovation: conference proceedings*, Vilnius, Lithuania, pp. 213-223. Available at http://www.constructionism2018.fsf.vu.lt/proceedings.

— *E-Textiles* —

Daily Overview Chart

INSTRUCTIONAL DAY	TOPIC
1-2	Introduce basic electronic components and circuitry. Students will design and create a simple circuit on paper (Electronic Greeting Card) using a cell battery, copper tape, and one LED light.
3	Explore the DIY Community.
4-7	Introduce electronic sewing, students stitch a wearable parallel circuit that includes a switch.
8-9	Introduce how LEDs can exhibit various behaviors through computational circuits.
10-11	Introduce Arduino programming.
12	Use one switch to control behavior of a Circuit Playground.
13-18	Students create an interactive mural.
19	Use a sensor as analog input.
20-31	Design and craft a human sensor project.

DAILY LESSON PLANS

Instructional Days 1-2: Ecard!

Topic Description

This lesson introduces students to basic electronic components and circuitry. Students will design and create a simple circuit on paper (Electronic Greeting Card) that uses a cell battery, copper tape, and one LED light.

Objectives

The student will be able to:

- Draw a circuit diagram with a simple circuit
- Identify necessary components of a simple circuit system (power, components, polarity, etc.)
- Peer review one another's circuit diagrams according to a rubric
- Create a paper greeting card with electronic components (including designing it, crafting it, debugging it, and sharing it with classmates)

Outline of the Lesson

- Journal Entry (5 minutes)
- Introduction to Electronic Textiles Unit (10 minutes)
- Introduction to Electronic Card Making (10 minutes)
- Electronic Card Circuit Design (20 minutes)
- Electronic Card Construction (50 minutes)
- Share Electronic Cards (10 minutes)
- Design Notebook Entry (5 minutes)

Student Activities

- Complete journal entry.
- Design an electronic card circuit.
- Construct an electronic card.
- Share electronic cards with classmates.
- Reflect in Design Notebook.

Resources

- Materials for each student:
    - Printout of Circuit Drawing Handout
    - 1 LED with lead wires
    - Paper for circuit sketch
    - Paper for the project (plain copy paper is fine)
    - Copper tape (approximately 18 inches)
    - 1 cell battery
    - Binder clip or paper clip
- Materials for student groups:
    - Scotch tape
    - Scissors
    - Colored pencils/markers
    - Paper
- E-Textiles Technical Guide:
    - Circuitry (pg. T9)
    - Materials (pg. T51)
    - Paper Circuit Design Example (pg. T57)

Resources continued on next page. →

Resources continued

- Website with video, circuit template, and samples: http://chibitronics.com/simple-circuit-tutorial
- Design Notebooks Electronic Card Sample Rubric (pg. E11)

Teaching/Learning Strategies

Journal Entry: *What are some ways you can apply problem solving to designing and building things by hand?*

- Share responses with elbow partner
- During the large group discussion, bring out connections to following the steps of an algorithm, using logic, problem solving, and debugging.

Introduction to Electronic Textiles Unit

- Students will learn about electronic circuits, make several projects, work in groups and alone, and learn to program using a text-based coding language.
    - Give a high level description of the projects, especially the final one.
    - Describe the Design Notebook.
        - The purpose of the notebook is to allow students to reflect on the processes they engaged and not just the final products.
        - It can include reflections, procedures, reminders, tips, sketches, ideas, etc.
        - There will be prompts after certain lessons.
        - The idea is inspired by Leonardo Da Vinci's notebook: https://www.bl.uk/collection-items/leonardo-da-vinci-notebook
        - It will be turned in as part of the Portfolio at the end (see pg. E60).
    - Describe the Portfolio.

Introduction to circuitry

- Distribute one wired LED to each student. Invite students to discuss and write observations in their Design Notebook. Pass out one coin cell battery to each student. Invite them to turn on their LEDs. Students should be encouraged to help one another to determine that all of the LEDs work properly and can turn on.
- Invite observations. Which side of the battery is positive? Does it matter which leg of the LED is touching the positive side of the battery? Which side of the LED needs to be connected to the positive side of the battery? (*positive-to-positive, negative-to-negative*)
    - How can you tell which side of the LED is positive or negative?
        - Invite elbow partners to come up with ways to remember which side is which. Share a few ideas as a class. Write on the board.
        - (Two ways: The longer lead wire is positive. There is also a small notch in the plastic on the positive side of the LED.)
        - *Pro tip: It might help to color one leg (e.g., positive) of the LED with a marker to see it more easily.*
- Show the Chibitronics tutorial video (Circuit Sticker #1): http://chibitronics.com/simple-circuit-tutorial/.
- If desired: Demonstrate some eCards as examples from Technical Guide pg. T57. Or share a Google Images search on: "paper circuit LED" and "copper tape."

Circuit design

- Show the circuit diagram from the Chibitronics website (Circuit Drawing Handout).
- *Note:* The copper tape provided in the Adafruit kits has conductive adhesive. Both the metallic side and the adhesive side conduct electricity.
- *Note:* Pay special attention to the details, like how to turn corners without breaking the tape and maintaining conductivity, leaving a gap for the LED, etc. Explain to students that they will be using a regular LED with wire legs that needs to be taped onto the tape or under the tape (not LED stickers as in the video).
- Students should gather all of their necessary materials (paper, estimate how much copper tape they will need, battery, paperclip/binder clip, etc).
- Make one paper circuit, following the diagram on the sketch (The LED will need to be reused on the final card, so treat it with care).
- Distribute scratch paper and pencils to each student so they can design their eCard in a circuit drawing ("blueprint"). Recommended: Do this in the Design Notebook!
- Elbow partners approve one another's blueprints before students move on to crafting their actual projects.

Electronic card construction

- Remind students that they may need to use Problem Solving and Debugging strategies to get them to work.
- Make available colored pencils, markers, etc. so that students can embellish their cards. They are welcome to use additional materials as well.
- Optional: If students finish quickly there are more advanced projects they can do here (making a switch, adding a light, etc.): https://chibitronics.com/learn/.

eCard Science Fair

- Split class in half and have half the group circulate room while the other half stay at their desk and present their projects to the circulators. Encourage circulators to ask questions like, “How did you do that?” “Why did you make those decisions?”
- Have two groups switch roles.

Design Notebook

- Record some expert tips on circuits and crafting electronic greeting cards that you would share with a beginner. You may include sketches or diagrams.

Electronic Card Sample Rubric

Name: ______________________________

DO YOU HAVE?	POINTS POSSIBLE	YES	NO	POINTS EARNED
Blueprint of a working circuit design: polarity labelled, (+, —), LED position labelled	5			
eCard design that incorporates an LED	5			
LED lead wires are connected to the copper tape	5			
LED turns on when the circuit is completed (i.e., battery in place	5			
Thought and intention is shown in the design of the card (it could be simple or elaborate)	5			
TOTAL:	**25**			

Instructional Day 3: DIY Community

Topic Description

This lesson is an exploration of the DIY Community.

Objectives

The student will be able to:

- Define DIY in their own words
- Explore the online DIY community
- Identify skills that their friends and family have that could be useful for DIY projects

Outline of the Lesson

- Journal Entry (5 minutes)
- Innovation Videos (10 minutes)
- DIY Community Web Hunt (30 minutes)
- Define DIY (10 minutes)

Student Activities

- Complete journal entry.
- Complete DIY Community Web Hunt.
- Discuss the definition of DIY.

Resources

- MacGuyverism #44 Video: http://bit.ly/MacGuy44
- Fixing zippers: http://bit.ly/fixingzip
- India's Menstruation Man: http://bit.ly/IndiasMM
- DIY Community Web Hunt
- www.Instructables.com
- http://macgyverisms.wonderhowto.com/
- www.doityourself.com/forum/
- Design Notebooks

Teaching/Learning Strategies

Journal Entry: *Write about a time that you made something yourself instead of buying it, or bought something that you changed to make better.*

- Share responses with elbow partner

Play videos

- Play one or more video(s) that model the innovation and creativity that is often used in DIY projects (Tip: do just one or two videos if you are short on time).
  - MacGuyverism #44 Video: http://bit.ly/MacGuy44
  - Fixing zippers: http://bit.ly/fixingzip
  - India's Menstruation Man: http://bit.ly/IndiasMM
  - Have students reflect on the videos with their elbow partners.

DIY Community Web Hunt

- Have students complete the web hunt in pairs.
  - Have students spend 10 minutes to do as much of #1 as possible. Each student in the pair should look for different terms. They can pick the suggested search terms or their own.
  - Groups spend 2 minutes sharing their responses to #1 with each other.
  - Student groups spend 10 minutes doing as much of #2 as possible.
  - Groups spend 2 minutes sharing their responses to #1 with each other.
  - Have student groups spend 6 minutes completing #3 and #4.

DIY Discussion

- Have students discuss their responses to what DIY (Do It Yourself) means to them.
- Have students share the skills that they, their friends, or their family have that can be used to complete DIY projects.

DIY Community Web Hunt

1. Complete at least the first 2 rows of the following table by going to www.instructables.com and finding a project you like. Record its name and the skills involved in completing it. You may use the suggested search terms below or choose your own. If you are working with a partner, you may want to look up different search terms.

   Suggested search terms: cooking, gardening, etextiles, cardboard, technology

SEARCH TERM	PROJECT NAME	SKILLS INVOLVED TO DO PROJECT

2. Complete at least 2 rows of the following table by exploring any of the sites listed. Find projects that match the given criteria. Record its name and the skills involved in completing it.

   Use any of the following sites or one of your own:
   - www.instructables.com
   - macgyverisms.wonderhowto.com
   - www.doityourself.com/forum

	PROJECT NAME	SKILLS INVOLVED TO DO PROJECT
Find a project that you've never thought of on your own.		
Find a project that you would want to try at home.		
Find a project that you think is the "coolest."		

3. In your own words, define DIY (Do-It-Yourself).

4. What might be some skills that you, your friends, or your family have that could be useful for completing DIY projects?

Instructional Days 4-7: Light-up Wristband

Topic Description

This lesson introduces electronic sewing and stitching a wearable parallel circuit that includes a switch.

Objectives

The student will be able to:

- Use conductive thread to sew electronic components
- Design and create a working parallel circuit with three lights
- Demonstrate how a switch works to turn electricity flow on and off
- Use an iterative design process

Outline of the Lesson

- Journal entry (5 minutes)
- Introduce Electronic Wristband (5 minutes)
- Aesthetic Drawing (15 minutes)
- Introduction to electronic sewing (15 minutes)
- Sewing a simple circuit (40 minutes)
- Design Notebook (10 minutes)
- Parallel circuits (15 minutes)
- Design circuit (20 minutes)
- Electronic Wristband (30 minutes)
- Design Notebook (10 minutes)
- Adding switches (40 minutes)
- Refine and decorate (15 minutes)
- Design Notebook (15 minutes)

Student Activities

- Complete journal entry.
- Practice basic electronic sewing.
- Complete Design Notebook entry — at end of each day.
- Design a parallel circuit.
- Craft an Electronic Wristband.
- Revise design to include a switch.
- Refine and decorate wristband.

Resources

- Materials for each student:
  - sewing needle
  - cell battery
  - cell battery holder
  - 3 LilyPad LEDs
  - conductive thread
  - 2 pairs of sewable snaps
  - felt sheets (~12" long, ~2" wide)
- Materials for student groups/class:
  - fabric scissors
  - seam rippers
  - alligator clips
  - rulers
  - scotch tape
  - craft glue
  - black pencil
  - red pencil
  - felt scraps (for practice sewing or decorations)
  - embroidery thread
  - sewing thread
  - tracing paper

Resources continued on next page. →

Resources continued

- Wristband Diagram images (fig. 1, fig. 2, fig. 3, fig. 4, fig. 5)
- Electronic Wristband Sample Rubric (pg. E25)
- Design Notebooks
- E-Textiles Technical Guide
  - Sewing (pg. T18)
  - Troubleshooting Electrical Problems (pg. T34)
  - Example of Wristbands (pg. T59)

Teaching/Learning Strategies

Journal Entry: *Thinking back to the Electronic Card, what were the important components of a circuit?*

- Share with elbow partners.
- You may want to show "Paper Circuit" from Figure 1 below.
- With the help of the students, construct the circuit diagram for the Electronic Card on the board (see left side of "Circuit Drawing" in Figure 1 below). This will be used as a starting place to introduce electronic sewing.

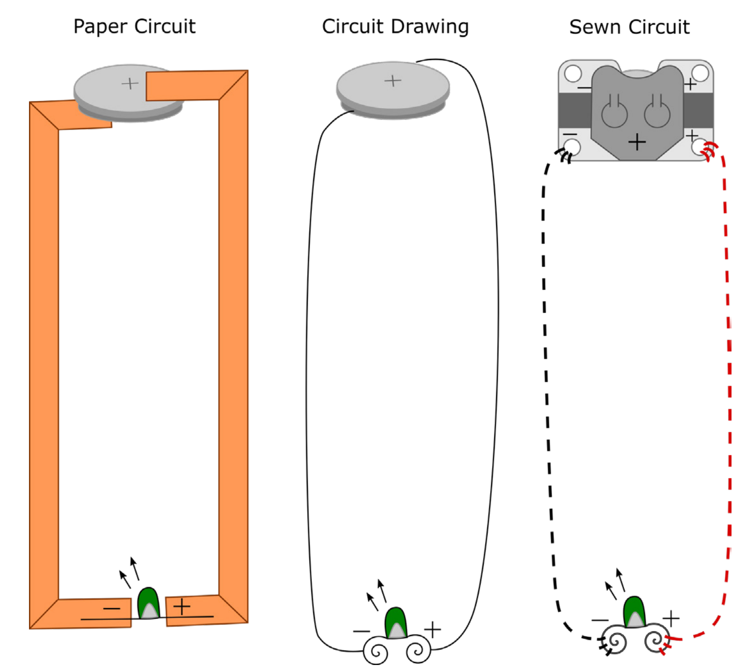


Figure 1: Paper Circuit to Sewn Circuit. Image credit Janell Amely.

Introduce Electronic Wristband or Decorative Cuff

- The finished project can be worn as a wristband or cuff and can also be used to decorate backpacks, chairs, etc. Show some samples. See E-Textiles Technical Guide: Activity 3: Electronic Wristband (pg. T59).
- Explain that students will create their wristband through an iterative design process.
    - First they will learn to sew a simple circuit with a single LED.
    - Then they will craft a wristband with 3 LEDs.
    - Finally, they will add a switch to their design.
- See Electronic Wristband Sample Rubric (pg. E25).

Aesthetic Drawing

- Invite students to draw what they would like their wristband to look like. It should have a battery holder (display) and three lights. *They do not need to work on correct circuitry yet. Simply get an idea on paper for what style their wristband can have.*
- *Optional:* Show examples of wristbands from the E-Textiles Technical Guide: Ideas & Inspiration (pg. T59). Show your own wristband as an example.

Introduction to electronic sewing

- Consult E-Textiles Technical Guide: Sewing Basics for Electronic Textiles (pg. T18).
- For each topic (thread a needle, create a starting knot, sew a running stitch, make an end knot) ask if any students know how to do it and have them demonstrate for the class. Use students' prior knowledge, allowing for different demonstrations for how to do each task.
- *Optional:* Students can practice sewing on pieces of felt with non-conductive thread. They can remove the stitches with a seam ripper before they start their actual project.

Sewing a simple circuit

- Explain to students that they will do this project in stages, breaking a problem down into parts. First they will learn how to connect and sew one light. Then they will add on the other two lights. Finally they can add a switch (using snaps) so that the wristband only turns on when snapped together.
- Using the circuit diagram from their journal, introduce the corresponding components that will be used for this project (new LEDs, battery holder, thread, etc).
- Choose which light to draw in and sew first. (Ask students which light might be best to do first. Share their reasoning.)
- Create a sewn circuit drawing next to the aesthetic drawing with one LED connected. They will add each component as you introduce them (see "Sewn Circuit" in Figure 1 above). Consult E-Textiles Technical Guide: Sewing Basics (pg. T18) for more information on each component on the next page:
    - *Note:* The same battery will be used in this project, but instead of using a paperclip to hold it in, the project uses a sewable battery holder.
    - Explain the parts of the battery holder. ***WARNING:*** The top metal part of the battery holder is positive. Not all of the holes need to be used for the circuit. Discourage sewing between two holes directly as that can create a short circuit on the negative side of the battery holder.
        - Add the battery holder to the sewn circuit drawing.
    - The LED will be a LilyPad LED with special holes for sewing.
        - Note the "+" and "–" to denote positive and negative.
        - Add the LED to the sewn circuit drawing.
        - *Note:* Figure 1 depicts a traditional LED. (We use LilyLEDs.)
    - Instead of copper tape, students will use conductive thread.
        - Add the thread as stitches to the sewn circuit drawing.
        - Red for positive, black for negative.
- Have students sew the first LED based on the sewn circuit drawing. They will know it works if it turns on when the battery is connected. Debug as necessary.

Design Notebook: *Write down some sewing tips you would give to a beginner. (10 min)*

- Have students share with elbow partners.
- Have a few groups share their tips with class.

Parallel circuits

- Return to the sewn circuit design of students' wristbands and sewn circuit drawing (see "Original" in Figure 2 on the next page) and ask students how to add the additional 2 LEDS from their aesthetic drawing. In pairs have students discuss how they might do this. Invite students to suggest how to make the modification to the sewn circuit diagram.
    - Add 2 LEDs below the "Original" circuit diagram from Figure 2. Ask students to share out how they might connect them. Connect them to the circuit as in "Add LEDs Outside" in Figure 2 on the next page.
        - Emphasize the positive and negative sides of the circuit.
    - Going back to the "Original" circuit diagram, ask students if they could add LEDs in between the battery and the current LED. Draw your LEDs connected as in "Add LEDs Inside" from Figure 2 on the next page.
    - Going back to the "Original" circuit diagram, ask students if it would be possible to have the positive leads of the LEDs right next to each other. Draw them as in "Add LEDs by Legs" in Figure 2 on the next page. Have students suggest how to connect the negative stitches.

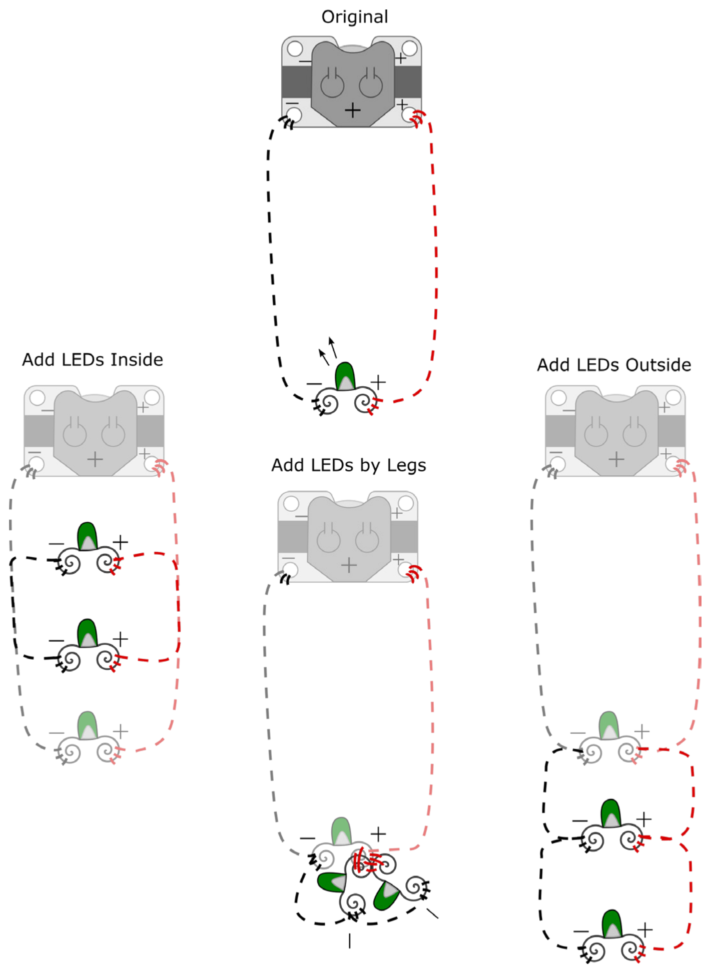


Figure 2: Adding Lights in Parallel. Image credit Janell Amely.

Design Circuit

- Have students return to their aesthetic drawing of their wristband on paper which shows the relevant decorative elements they wish to add. Have them either revise their sewn circuit diagram ("blueprint") to include all three LEDs or draw a new sewn circuit diagram that shows how the different components connect. They may want to use tracing paper for one layer. Designs should include the battery holder, 3 LEDs, and thread path, making sure to indicate the positive and negative polarities (which can be indicated using black or red pencils).
    - See Figure 3 on the next page for example circuit diagrams and Figure 4 for their corresponding aesthetic drawings with circuit diagrams visible.
    - Show your example wristband as another sample.
- Have elbow partners review each other's circuit designs.

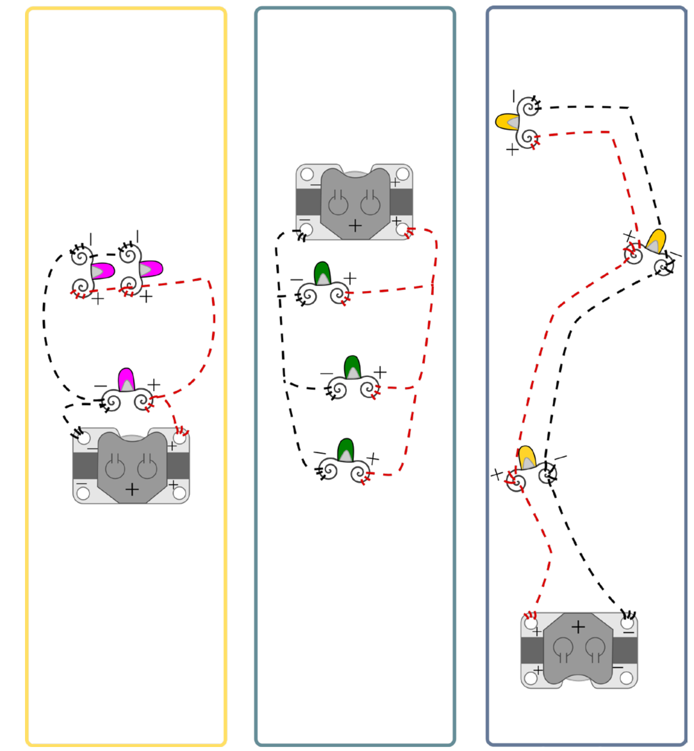


Figure 3: Example circuit diagrams. Image credit Janell Amely.

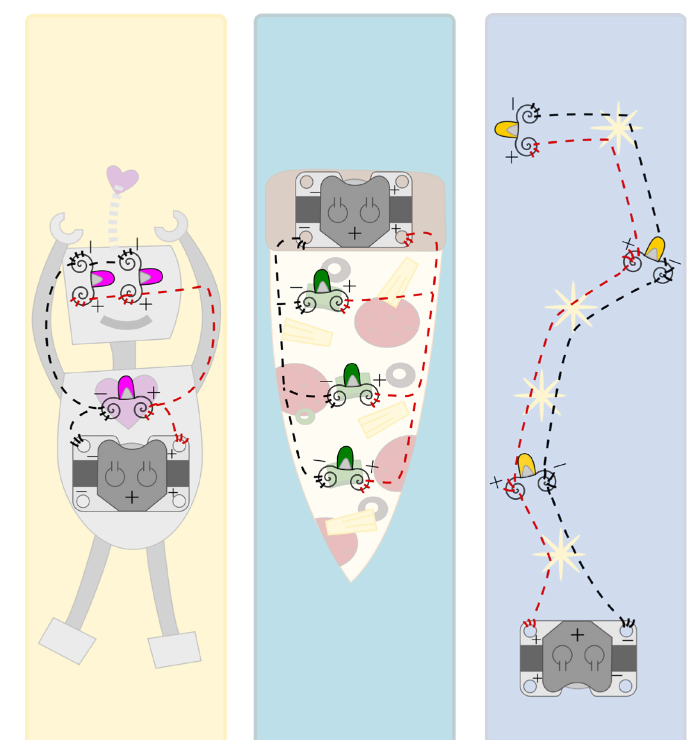


Figure 4: Example aesthetic drawings and circuits. Image credit Janell Amely.

Electronic Wristband

- Have students sew two more lights onto wristbands (in parallel as discussed above).
- *WARNING:* Crossing opposite polarities will cause a SHORT CIRCUIT!
- See E-Textiles Technical Guide: Activity 2: Electronic Wristband (pg. T59).
- When students are finished, have them test their wristbands with a battery.
- Use alligator clips to debug circuits as necessary. See E-Textiles Technical Guide: Troubleshooting - Electrical Problems (pg. T34).

Design Notebook: *Respond to one of the following prompts:*

- Draw/write tips for sewing LEDs in parallel.
- What are some of the important aspects of a parallel circuit?
- What are some ways a parallel circuit is different from a simple circuit?
- *Encouraged:* In addition, take a picture of your Wristband progress so far.

Adding Switches

- Currently students' wristbands will always be on. Show students a completed Electronic Wristband and ask them how the lights are switched on and off. Ask them what might be the switch for the Wristband. The snap is the switch!
- Referring to one of the drawings from Figure 4, have a conversation about creating a switch:
    - Draw one pair of snaps without connecting them to the circuit.
    - Explain that the snaps are conductive like the thread because they are made of metal.
    - Explain how each of the two ends of the snaps can be made to connect to positive or negative only and how snapping them together completes the circuit.
    - Invite students to edit the design on the board to incorporate the snap as a switch. See Figure 5 on the next page.
    - Emphasize that part of their circuit *must be cut* to add the switch. Ask why this is so.
    - Demonstrate how the wristband snaps. Ask students what they need to keep in mind when sewing the snaps. Emphasize that one snap must be on the back of the wristband. This is why each of the wristbands in Figure 5 have a corner folded up to reveal the snap on the flipside (see two pictures on left).
    - An additional set of snaps can be added for more security - these have no electrical properties, they are not connected to the circuit and can be sewn with regular (non-conductive) thread.
    - See examples in the E-Textiles Technical Guide for more information (pg. T59).
- Have students revise their own circuit diagrams.
- Students will then sew on snaps as switches to their Wristbands.
    - Regular sewing (non-conductive) thread or fabric glue can be used to attach any other extra decorations,to secure the rest of the battery holder, or the extraneous snaps to the fabric.
    - Debug as necessary; alligator clips may be used to troubleshoot.

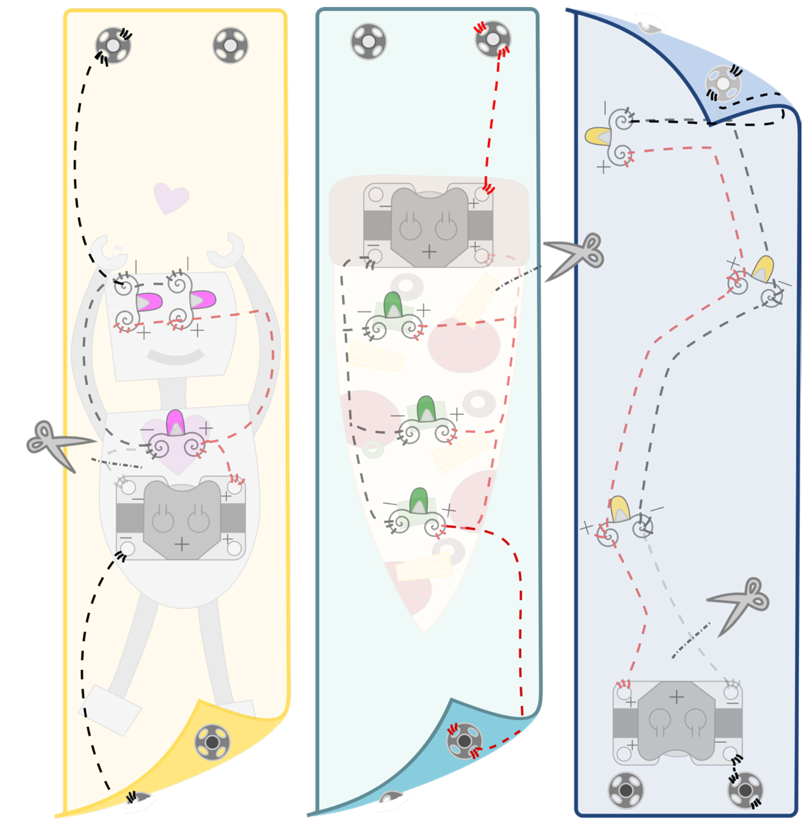


Figure 5: Adding snaps as switches. Image credit Janell Amely.

Refine & Decorate Wristband

- Once snaps are complete, students can take time to add to the aesthetics of their wristbands. Felt shapes can be cut and placed on top of LEDs, with or without holes to show the light (small dots of glue or a few stitches with regular thread can keep these pieces on).
- Craft glue can be used to secure the conductive thread knots by preventing fraying.
- Students can also take home Wristbands to finesse them. But they must *bring them back for later activities!*

Design Notebook

- Thinking about your Wristband project, what expert tips would you give someone who is just starting this project (that you wished someone had told you when you started)?
- Share with elbow partners.
    - Have a few groups share their tips with class and/or write them on a board.
- *Encouraged:* Take a picture of your wristband.

Volt Ohm exercise; or, How & why to use a multimeter (30 min)

Materials: Students will need a finished or partially done project, such as the wristband project. The project should have conductive thread stitching to collect data on. Students may use the same stitching and measure different lengths and parts of it. Students will also need multimeters. The student may work individually, pairs, or groups of four; group sizes mostly depend on how many multimeters there are.

DC electricity (5-10 min)

- Teacher discusses properties of DC electricity.
    - Positive and negative parts of circuit.
    - Open and closed circuits.
    - Electricity goes through the most direct path with least resistance.
    - Ohms are the measurement we use for resistance.
- Introduce volt ohm meter:
    - http://bit.ly/voltohm1 (from electronicsnotes.com) or http://bit.ly/voltohm2 (from fluke.com)
    - Which tutorial you use depends on which multimeters you have in class. Be sure to demonstrate how (and why) to use them.

Students measure ohms and collect data to share (20 min)

WARNING: Make sure there is no current (no power source) connected to the Circuit Playground, such as a battery or connected to computer via USB.

- Students should collect, organize, and compare the resistance as measured by a multimeter over the following data:
    - Different lengths of conductive thread
    - Conductive thread tension
    - Stitches per inch
    - Students should also look at what might be the causes of poor connections or intermittent connections.
- Groups share.
- Class discussion.
- Class organizes data and discusses what they have found regarding resistance of the conductive thread. Address the following questions: How does the length of the conductive thread affect the resistance? How does the quality of the sewing such as stiches per inch and tension on thread affect the resistance?

Journal Entry: *From what you learned about the class data, how will that affect your design choices and execution of sewing skills?*

Electronic Wristband Sample Rubric

Note: For rubric below, giving partial credit is encouraged if students partially meet a requirement.

DO YOU HAVE?	POINTS POSSIBLE	POINTS EARNED
CRAFTING		
1. Wristband has at least 3 lights sewn in parallel.	10	
2. The battery holder, LEDs, and snaps are securely sewn in (stitched through at least three times).	5	
3. Stitches are evenly spaced and no wider than ¼ inch in length.	5	
4. Starting and ending knots are secure. No loose or loopy threads are present in the back or front of the fabric.	10	
Comments on Crafting:		
CIRCUITRY		
5. Circuit design blueprint with accurate markings: positive and negative lines, positive and negative sides of the LED, switch, all components are labelled correctly.	15	
6. Project matches the circuit design.	10	
7. All LEDs light up.	5	
8. Wristband turns on and off when it is snapped and unsnapped.	10	
Comments on Circuitry:		
OVERALL		
9. Design Notebook entries completed.	15	
10. Aesthetics: The design appears to be purposeful and/or personal. Care has been taken in the look and feel of the project. Students can justify their design if helpful (written, verbal).	15	
Comments on Overall Aesthetics:		
CRAFTING: ___ CIRCUITRY: ___ OVERALL: ___	TOTAL:	/100

Instructional Days 8-9: Introduction to Circuit Playground and Computational Circuits

Note: This lesson requires you to pre-program (upload code to) some Circuit Playgrounds for your students to use. Please plan for the extra prep time.

Topic Description

This lesson introduces how LEDs exhibit various behaviors through computational circuits.

Objectives

The student will be able to:

- Design a computational circuit using a pre-programmed mini-computer
- Understand the role of a common ground

Preparation

- Pre-upload "preprogrammedCP.ino" code onto enough Circuit Playgrounds for each pair of students to use.
- If you have questions about the Circuit Playground, check out this overview by its developer, Adafruit: https://learn.adafruit.com/introducing-circuit-playground/overview

Outline of the Lesson

- Journal entry (10 minutes)
- Circuit Playground introduction (10 minutes)
- Exploring computational circuits with worksheets (40 minutes)
- Design Notebook (5 minutes)
- Design project (15 minutes)
- Share designs (15 minutes)

Student Activities

- Complete journal entry.
- Explore computational circuits.
- Complete Design Notebook entry.
- Design project.
- Present designs.

Resources

- Materials for student pairs:
    - Circuit Playground
    - USB micro cables with computer to charge (or LiPo battery with battery charger)
    - Alligator clips
    - Wristband projects from days 4-7 (one per student)
- Exploring the Circuit Playground.pdf
- Exploring the Circuit Playground - Teacher.pdf
- E-Textiles Technical Guide
- preprogrammedCP.ino

Teaching/Learning Strategies

Journal Entry: *Think about what would make your wristband more awesome. What kind of functionality would be cool? (I.e., If you could customize the lights on your wristband to do more than just be on or off, what else might you make the lights do?) (10 min)*

- Have students share with elbow partners.

Exploring computational circuits

- Have students work together in pairs.
- Students will need their wristband projects from days 4-7. Each pair will need both members' wristbands.
- Distribute a Circuit Playground (CP) and USB micro cable (to plug the CP into a computer), and alligator clips for each pair of students.
    - Have students examine the different pins and markings on the Circuit Playground (i.e. GND, 3.3V, VBATT, 0, 1, 2, etc.).
    - Especially note the GND (negative), 3.3V (positive), and numbered pins on the edges.
    - ***WARNING:*** The pin marked VBATT is connected directly to the powersource. Warn students *not* to connect things up to it—the voltage is too high (5V) for most LEDs and could burn them out.
    - Ask pairs to brainstorm how they might connect all the components (wristbands, alligator clips, Circuit Playground, power source).
    - *Details on what each pin does and where each sensor and switch are can be found here:* https://learn.adafruit.com/introducing-circuit-playground/pinouts.
- In pairs, have students complete Activity One ("Test it Out") from the Exploring Circuit Playground Student Guide (Exploring the Circuit Playground.pdf). The answers to Activity One are below (students may use different words to describe the effects):
    - Pin 1 (TX 1): Blink (on/off regularly)
    - Pin 3 (SDA 2): Twinkle (blinking at different brightnesses in different intervals)
    - Pin 9: Heartbeat (regular "beating" rhythm for fading)
    - Pin 6: Fade (steady fade in and out of brightness)
- Facilitate a discussion on computational circuits.
    - Draw a circuit diagram on the board similar to the circuit used in Activity One.
    - Draw an additional LED and have students brainstorm how they would connect it to the current circuit.
    - Have students share their ideas and update their drawings.
    - Highlight one student diagram that has the LEDs connected in parallel (to same pin on Circuit Playground) and ask students what the behavior of each light might be.
    - Highlight one student diagram that has LEDs connected to different pins on the Circuit Playground and ask students what the behavior of each light might be?
- In pairs, have students complete Activity Two ("Designing Computer Circuits") from the Exploring Circuit Playground Student Guide.
    - Have some students share out their designs and the expected behavior of their LEDs.
        - Make connections to the student diagrams from computational circuits.

- Highlight a design that uses parallel circuits. Explore the idea of a common ground (one negative line connecting the LEDs' negative sides to the negative of the Circuit Playground or battery holder) instead of each LED separately connecting all the way back to the battery holder. Do you think this might change the behavior of the lights? In Activity Three they will practice making parallel circuits.
- Students complete Activity Three ("Add More Lights with Parallel Circuits") from the Exploring the Circuit Playground guide.
  - For possible solutions, see Exploring the Circuit Playground - Teacher.pdf.

Design Notebook: *Respond to one of the following:*

- Reflecting on the day, what are some things you learned about computational circuits?
- Explain (draw or write) how to design a computational circuit such that all the lights perform different behaviors.

Design project

- In their Design Notebooks, have students *brainstorm* at least one project that they could make with the pre-programmed Circuit Playground. They should describe which pre-programmed light patterns they would use for their projects and how the chosen light patterns would enhance the design of their projects. Students should feel free to include sketches or diagrams.
- If students need some inspiration, here are some examples:
  - A plush bear that has a light up heart that uses the heartbeat pattern.
  - A cityscape with lights on the skyscrapers that blink differently.
- Have students share with elbow partners.
- Have a few students share with the class or invite students to put their sketches up on a bulletin board.

Instructional Days 10-11: Introduction to Programming in Arduino

Note: Make sure that your programming environment (see the Coding Platform Guide on the ECS e-textiles website) works on student computers before starting this lesson. Always test ahead of time to avoid losing class time.

Topic Description

This lesson introduces students to Arduino programming.

Objectives

The student will be able to:

- Program a blinking light pattern in Arduino with their wristbands and a Circuit Playground

Outline of the Lesson

- Journal Entry (5 minutes)
- Connect the Circuit Playground to wristband (5 minutes)
- Program layout (10 minutes)
- Programming a blinking light (15 minutes)
- Blinking patterns (20 minutes)
- Debugging activities (20 minutes)
- Finish blinking patterns (20 minutes)
- Gallery walk (10 minutes)
- Design Notebook (5 minutes)

Student Activities

- Complete journal entry.
- Participate in Circuit Playground and program layout discussion.
- Connect wristband to Circuit Playground.
- Program a blinking light.
- Create a light pattern.
- Debug an existing program.
- Exhibit their light pattern and see others during the gallery walk.
- Complete a Design Notebook entry.

Resources

- Materials:
  - Circuit Playground
  - Micro usb cable
  - Computer
  - Alligator clips
  - Student Wristbands (one per student/two per pair)
- TurnOnTheLights_Starter.ino
- BasicBlink_Example.ino
- Light Pattern Storyboard.pdf
- Debugging Fun Blink1 Handout
- Debugging Fun Blink2 Handout
- debuggingFunBlink1.ino
- debuggingFunBlink2.ino
- Debugging Fun Blink1 Solution
- Debugging Fun Blink2 Solution
- E-Textiles Technical Guide: Coding (pg. T25)

Teaching/Learning Strategies

Journal Entry: *What were some ways we used the "wait_secs" block in Scratch?*

- It may help to display a visual of the "wait_secs" block from Scratch.
- Share responses with elbow partner.

Connect the Circuit Playground to wristband

- Connect the Circuit Playground to the computer, and open the development environment (i.e. Codebender or Arduino). Students work in pairs.
  - Show students how to connect the Circuit Playground to a computer using a micro-USB cable.
    - *Tip: Sometimes there is an error message about a "keyboard" not working. Ignore this. The computer sometimes assumes that the Circuit Playground is an external keyboard.*
  - Open the chosen development environment and show the students how to select the board (Circuit Playground) and port (usually COM1 or COM[X] for Chromebooks or Windows machines, usbmodem for Apples) .
  - See E-Textiles Technical Guide: Basic Coding for Electronic Textiles (pg. T26).

Review layout of program

- Use TurnOnTheLights_Starter.ino.
- Naming section: "name" the components connected to the pins of the Circuit Playground.

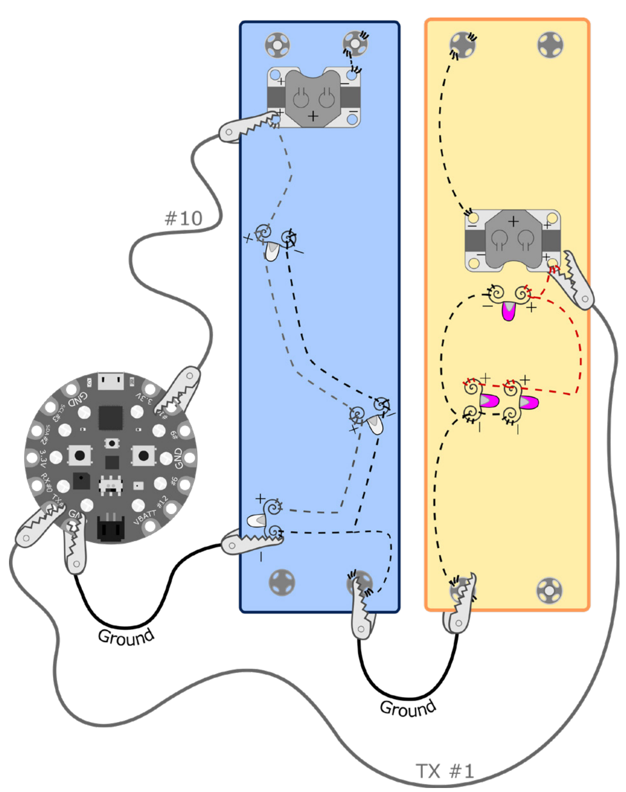


Figure 6: Connecting wristbands to the micro controller.
Image credit Janell Amely.

  - They will be using their wristbands as LEDs. Looking at the naming section, have students guess which pin to connect the wristband to.
    - Since "wristband1 = 10", it should be connected to pin #10.
  - Have students connect their wristbands to Circuit Playground (see image on previous page).
  - Relate this to creating variables in Scratch when they made the nutrition game on day 14.
- setup(): tells Arduino which pins are input and which are output.
- loop(): the code here executes over and over.
  - Relate this to the "forever" block in Scratch.
- In pairs, have students compile, download, and test the TurnOnTheLights_Starter.ino code to see the light turn on.
- Make sure students have one wristband connected to pin 10 (or alter the code for the pin number connected).

Demonstrate how to blink one wristband

- Show students the line that turns the light(s) on.
- Ask students how they think they could program that same light to turn off
  - Have students compile and try their code.
  - When the light appears to remain on, ask students why.
    - Point out that since it is looping, the light is turning on and off so fast it appears to stay on. They might have run into something similar in Scratch.
  - Ask if they wanted to make the light blink on and off every second, how might they do this in Scratch.
    - Relate the "wait _ secs" block to the delay() function.
    - Explain that the delay() function takes the number of milliseconds instead of seconds like the "wait_secs" block in Scratch so delay needs to be called with the value 1000 for a 1 second delay.
    - Try the code with 1 delay() in between setting the light to be high and low.
    - When the blinking still is not quite right, lead students to add a delay after the line that turns the light off.
    - See BasicBlink_Example.ino.

Pairs code their own blinking patterns with two wristbands

- Have students complete Light Pattern Storyboard.pdf on paper.
  - Students can alternate scribing so both students get to write.
- When students are ready to code, have driver and navigator switch and program a unique light pattern.
- Have driver and navigator switch and program a second light pattern.

Debugging Activity

- Assign half the pairs to do Debugging Fun Blink1 Handout and the other half to do Debugging Fun Blink2 Handout.
    - Have them all complete #1 on their assignments.
- Have students load the corresponding Arduino files: debuggingFunBlink1.ino or debuggingFunBlink2.ino.
    - Have them run the code and answer #2.
        - Alternatively, you can run the program for the students as a whole class and have them complete #2
- Have students complete their debugging assignments.
- Pairs will then join another pair that had the other debugging assignment and share their scenarios and answers.
- See Debugging Fun Blink1 Solution and Debugging Fun Blink2 Solution.

Finish blinking patterns

- Students finish programming their own blinking patterns, making sure each partner has had a chance to program one.
- Extension: If pairs finish programming two wristbands quickly, provide an option to add on more LEDs to program (i.e., individual LEDs, paper circuits, or more at your discretion).

Gallery walk of patterns

- Have students share their lighting patterns.
- Some ideas:
    - Students leave their blinking wristbands on the table and walk around the class, seeing other patterns.
    - Have students hold up their patterns.
- Encourage some compliments: What did you like? What new ideas did you get?

Design Notebook Entry: *Write down some expert tips for programming light patterns in Arduino.*

Instructional Day 12: Programming with One Switch (simple conditionals)

Topic Description

This lesson provides an introduction to using one switch and simple conditionals to control the behavior of a Circuit Playground.

Objectives

The student will be able to:

- Use a switch as input
- Program their own computational circuit using conditionals
- Write functions to organize and reuse code

Outline of the Lesson

- Journal Entry (5 minutes)
- Review conditionals (10 minutes)
- Switch as input (10 minutes)
- Creating light pattern functions (30 minutes)
- Recommended: One Switch Roleplay activity (40 minutes)

Student Activities

- Complete journal entry.
- Participate in review of conditionals.
- Discover how to use a switch to turn an LED on and off.
- Program the Circuit Playground to perform a chosen light pattern based on the position of the switch.

Resources

- Materials:
    - Circuit Playground
    - Micro usb cable
    - Computer
    - Wristbands
    - Alligator clips
- age solution.sb - from Unit 4
- Arduino Reference - If/Else: https://www.arduino.cc/en/Reference/Else
- One Switch Program Storyboard.pdf
- One_Switch_Starter
- One_Switch_ON_OFF
    - All code samples, starters, and examples available in Arduino code as well.
- Arduino Reference - Functions: https://www.arduino.cc/en/Reference/FunctionDeclaration
- Helper_Functions
- Helper Function Storyboard.pdf

Teaching/Learning Strategies

Journal Entry: *What were some ways we have used the "if" and "if/else" block in Scratch?*

- It may help to display a visual of the "if" and "if/else" blocks from Scratch.
- Share responses with elbow partners.

Review conditionals

- Have students share from their journal responses.

- Review how conditionals were used in Scratch for the Age project (Scratch Unit Day 15).
    - See age solution.sb from Unit 4.
- Show how conditionals are coded in Arduino vs Scratch
    - Place an "if/else" block from Scratch on the board and write the Arduino if/else code next to it point out the corresponding elements:
        - The conditions go inside the "()"s.
        - The curly braces determine encompass the code that will execute in the if and else.
        - See Arduino Reference - If/Else: https://www.arduino.cc/en/Reference/Else

Using the slide switch to turn an LED on and off

- Invite students to identify the slide switch on the Circuit Playground, ask which pin number is it assigned to. (#21)
    - *Note:* There are three switches on the Circuit Playground (two buttons and one slide switch). We are using only the slide switch for now.
- In pairs, have students complete #1-#3 of One Switch Program Storyboard.pdf.
- Lead class through completing #4.
    - Have students open One_Switch_Starter.
    - See One_Switch_ON_OFF for the solution.
    - Point out that the sensorVal will be "HIGH" when the slide switch is at "+" and "LOW" when it is switched to "–".
- In pairs, students complete #5 and #6.
    - Remember to connect one Wristband to the correct pin!

Create light pattern functions

- Introduce the BUILDING BLOCKS Section of their program.
    - They will create their own function for their light patterns.
        - They can reuse the light patterns by cutting and pasting from the previous lesson, or they can create new light patterns.
    - The syntax will be similar to the setup() and loop() functions they have been using:
        - "void" - meaning that the function does not return any values.
        - The name of the function (e.g., slowblink, funkyblink).
        - "()" - since we are not passing any arguments.
        - The body of the function goes between "{" and "}".
    - See Arduino Reference - Functions: https://www.arduino.cc/en/Reference/FunctionDeclaration
- Students complete complete #7 to #13 on One Switch Program Storyboard.pdf.
- OPTIONAL EXTENSION: Helper Functions
    - Optionally, share how functions can be used as building blocks.
        - See Helper_Functions.
    - Use Helper Function Storyboard.pdf

One Switch Class Roleplay (40 min total)

One of the most difficult concepts for students to understand is the need for a variable to hold the information from the switch (i.e., why "switch1storage" in the sample code is needed in addition to a name for the switch as in "switch1"). The activity below helps them to grasp this and to review DC circuit conditions versus micro controller programming logic. Use this activity as an introduction to the switch coding lesson. Students will understand the CS conditionals and variables.

Review of switches and variables (5-10 min)

- Discuss with class that the switches on the Circuit Playground are not part of the processor. The Circuit Playground designer added the switches to the board so that one would not need to connect switches to the Circuit Playground. Remind students that a switch, like the ones students made previously, are basically two wires that are not connected, thus are an open circuit. The switch can be manipulated so that the two wires touch and close the circuit. A switch does not have a "brain."
- Discuss what a variable is and why we use them in mathematics. Make the connection between variables in mathematics and coding.

One Switch Roleplay (30 min)

Note: We have used "switch1storage" as a variable name because it helps us remember what that variable does—stores the information from reading the switch. However, it can help if students make up their own name: swoosh, Bob, switchvalue, etc. Just like naming LEDs for convenience, we should name our data-storing variables with something that makes sense to us.

Roleplay Setup One Switch

- One student in front of class as the variable switch1storage with a two-sided card or felt such that one side represents HIGH and the other side represents LOW. This student only changes the card when told to "digitalRead" the human "switch" person in the back of the room (see next bullet).
- One student in the back of the classroom as the switch, (facing the front and can see the student that represents the variable, but cannot be seen by the class) with a two-sided card or felt such that one side represents HIGH and the other side represents LOW.
- The rest of class represents the processor in the Circuit Playground that executes the code written. They face forward and can only see the variable student and not the switch student.

Roleplay Activity One Switch

- Teacher or student conducts a discussion on the 4 parts of code on board or projected. Have someone (teacher or student) write the code on the board. Leave space in each section to write additional code later.

```
/* The first section is where to declare global objects and call up
additional files the program needs to use.  */

void setup () {
/* The second section is for things that only need to be done once
at the beginning of the program.  */
}

void loop() {
/* The third section is for things that happen repeatedly in the
program loop while the program is running. The code is executed in
the order coded. */
}
/* The fourth section is for functions that are called up by the
third section.  */
```

- Class writes the first section of code to declare pin and variables. Class discusses what they wrote and why. Add comment lines.
- Class then writes second section of code lead by teacher for things that have to happen once, such as pinMode. Class discusses what they wrote and why. Add comment lines.
- Class then writes third section of code lead by teacher to use digitalRead and assign variable values to switch1storage. Class discusses what they wrote and why. Add comment lines.
- Class decides what behavior happens when the variable value is HIGH and another behavior when the variable value is LOW. Such behaviors could be clapping or standing up. Class then writes the two functions in the fourth section. Class discusses what they wrote and why.
- Class then adds conditional statements to the third section that calls up the behavior functions in forth section for the two possible variable values. Class discusses what they wrote and why. Add comment lines.
- Once code is done teacher leads the class through the code line at a time and class roleplay their given roles. Have the switch student randomly pick a state to be in. Start from first section of code. Students will need to be reminded not to do the behavior until digitalRead, the code assigns a value to the human switch1storage variable, and have read the conditional code in the third section.
- Repeat the loop code, third section. The switch student changes their state randomly and the teacher continues to loop through the third section until they feel the students understand the CS concepts addressed in the lesson.

Journal Entry: *Why do we need to have a variable and digital read in our code for one switch?*

Instructional Days 13-18: Interactive Mural Project

Topic Description

This lesson has students create an interactive mural.

Objectives

The student will be able to:

- Craft part of a collaborative mural that uses a computational circuit and switches
- Program a computational circuit with complex conditionals that respond to user input

Outline of the Lesson

- Journal Entry (5 minutes)
- Introduction of project (20 minutes)
- Aesthetic and circuit design (30 minutes)
- Crafting and programming (35 minutes)
- Design Notebook (10 minutes)
- Two Switch Roleplay: Review of complex conditionals (45 minutes)
- Crafting and programming (35 minutes)
- Design Notebook (10 minutes)
- Debugging activities (20 minutes)
- Crafting and programming (100 minutes)
- Class mural exploration (20 minutes)
- Design Notebook entry (10 minutes)

Student Activities

- Respond to journal entry.
- Participate in introduction of project.
- Collaborate with partner to design their project.
- Begin crafting and programming.
- Participate in review of complex conditionals.
- Craft and program.
- Debug existing program.
- Finish crafting and programming.
- Explore class mural.
- Complete Design Notebook entry.

Resources

- Materials for each pair:
  - Circuit Playground board
  - Micro usb cable
  - Computer
  - 4-6 LilyPad LEDs
  - felt pieces for base and decoration
- Materials for each table group:
  - Craft glue
  - Thread wax
  - Sewing thread
  - Seam rippers
  - Rulers
  - Black and red pens
  - Scotch tape
  - Alligator clip set
- Design Notebooks
- Translating from Scratch Activities (Scratch Translation Lesson Notes.pdf, optional)
- Interactive Mural Sample Rubric (pg. E43)
- Mural_Starter code
- Mural Storyboard.pdf
- Debugging Fun Mural1 Handout
- Debugging Fun Mural2 Handout
- debuggingFunMural1 code
- debuggingFunMural2 code
- Debugging Fun Mural1 Solution
- Debugging Fun Mural2 Solution

Resources continued on next page. →

Resources continued

- Use_Two_Switches_Starter code
- Fleshed out Two Switches Example Code: Use_Two_Switches_EXAMPLE_PizzaCart_
- E-Textiles Technical Guide

Teaching/Learning Strategies

Journal Entry: *If you have 2 switches, how many different ways could you arrange them (i.e., On and Off)? How many ways are there total?*

- Share with elbow partner.
- Have a student share their answers with the class.
- Guide students to the fact that there are 4 configurations - OFF/OFF, OFF/ON, ON/OFF, and ON/ON. Write this on the board.
  - Relate the switch configurations to binary numbers (Unit 2, Days 10-12) and the amount of numbers that can be represented with 2 bits.

Introduce Interactive Mural Project

- Have class decide on a theme for their quilt-style mural considering how many panels the class can make.
  - It could feature different parts of the school (mascot, stadium, cafeteria, hang-out areas, ECS classroom) or their neighborhood/ school community.
  - Another idea could be a banner with letters such as: "ECS Rocks!" or "Go Bears!" (insert chosen slogan)
  - Or you can pick a favorite theme (Pacman, local plants, etc.).
  - The class may want to agree on a color scheme or other aesthetic details.
  - Each pair will craft one piece of the mural, each panel will feature one Circuit Playground and 4-6 LEDs.
  - Review where the switches are on the Circuit Playground (slide switch: #21, button switches: #4 and #19). They will program two of them (i.e., switch and button or two buttons). Explain how a button acts as a switch (by closing a circuit, but instead of left /right, it's up / down).
- See E-Textiles Technical Guide: Activity 3 Collaborative Mural Project (pg. T61).
- See Interactive Mural Sample Rubric (pg. E43).
  - Students may be assessed in pairs, one rubric for the group.

Aesthetic and circuit design

- Have student pairs draw both an aesthetic design and a circuit design for their panel.

- They should consider where to place the Circuit Playground and LEDs in both designs.
  - People will need to operate the switches on the Circuit Playground to make the lights change pattern.
- Students especially should consider how to connect (sew) the components to the Circuit Playground (circuit design).
  - They may use a common ground, but they will want to avoid stitches crossing over each other as it will cause crossed signals or shorts.
  - The multiple ground (GND) pins on the Circuit Playground may provide additional flexibility in circuit diagrams.
  - Note how thethe Li-Po battery needs to be attached after the project is finished. It will either hang in front of the Circuit Playground when connected, or students will need to find a place to store it (e.g., felt pocket, a slit in project to put it in the back, etc.).

Student pairs begin crafting and programming

- Have pair groups complete #1 - #5 of the Mural Storyboard.pdf.
- They may use Mural_Starter code to begin their programs. They can enter the code from steps #1 - #5 of the storyboard. They will complete the storyboard after reviewing conditionals.
- One member may want to work on the program while the other is crafting. Ensure that each member gets a chance to craft and program.

Design Notebook

- *(Can be completed at the end of class, either after aesthetic & circuit design or after some crafting time—whichever is more convenient.)*
- Take a picture of your work so far. What is something you have had to fix or something you have made a change to since you started on this project?

Two Switch Roleplay

- This activity is a revisit of the previous activity One Switch Roleplay (recommended on Instructional Day 12) with the addition of another switch. Students revisit CS concepts of coding structure, conditional statements, variables, switches, and digitalRead. In addition students will explore the mathematical concept of possible combinations and the CS concept of nested conditional statements.
- **Review One Switch Roleplay Activity (5 min)**
- **Explore possible combinations (10 min)**
  - Journal Entry: *Given two switches how many possible combinations of switch states for two switches?*
  - Students pair share.
  - Students share out what they discussed as pairs. Explore the different strategies used to determine the total possible combinations.
  - After class discussion students add to their journal entry.

- **Two Switches Roleplay (30 min)**
    - One of the most difficult concepts for students to understand is the need for a variable to hold the information from the switch (i.e., why "switch1storage" in the sample code is needed in addition to a name for the switch as in "switch1"). The activity below helps them to grasp this and to review DC circuit conditions versus micro controller programming logic and nested conditional statements. Use this activity as an introduction to the mural project two switches coding challenge. Students will understand the CS concepts of nested conditionals and variables.
- **Roleplay Setup Two Switches**
    - Two students in front of class as the variables "switch1storage" and "switch2storage" with two-sided cards or felt such that one side represents HIGH and the other side represents LOW. The two cards should be differentiated so students can tell which card represents which variable, such as each card is a different color.
    - Two students in the back of the classroom facing the front (and can see the students that represent the variables, but cannot be seen by the class) with two sided cards or felt such that one side represents HIGH and the other side represents LOW. The two cards should be differentiated so students can tell which card represents which variable, such as each card is a different color.
    - The rest of class represents the processor in the Circuit Playground that executes the code written. They face forward and can only see the variable students and not the switch students.

```
/* The first section is where to declare global objects and call up
additional files the program needs to use.  */

void setup () {
/* The second section is for things that only need to be done once
at the beginning of the program.  */
}

void loop() {
/* The third section is for things that happen repeatedly in the
program loop while the program is running. The code is executed in
the order coded. */
}

/* The fourth section is for functions that are called up by the
third section.  */
```


- **Roleplay Activity Two Switches**
    - Teacher or student conducts a discussion on the 4 parts of code on board or projected. Have someone (teacher or student) write the code on the board. Leave space in each section to write additional code later.
- **Class discussion. As a class:**
    - Decide the first section of code to declare pins and variables & write it on the board. Discuss what the class wrote and why.
    - Decide the second section of code for two switches for things that have to happen once, such as pinMode. Discuss what the class wrote and why. Add comment lines.
    - Decide the third section of code to digitalRead two switches and assign variable values to switch1storage and sensorVal2. Discuss what the class wrote and why. Add comment lines.
    - Decide what four behaviors happen for each possible combination of the variable values. Such behaviors could be clapping or standing up. Write the four functions in the fourth section. Discuss what the class wrote and why. Add comment lines.
    - Add nested conditional statements to the third section that calls up the four behavior functions in fourth section for the four possible variable value combinations. Discuss what the class wrote and why. Add comment lines.
- Once code is done lead the class through the code one line at a time and have the class roleplay their given roles. Have the switch students randomly pick their states to be in. Variable storage students (switch1storage, switch2storage) only change their values when told to "digitalRead". Start from first section of code. Students will need to be reminded not to do the behaviors until digitalRead, the code assigns value to the variables, and student have read the conditional code that applies in the third section.
- Repeat the loop code, third section. The switch students change their states randomly and the teacher continues to loop through the third section until the instructor feels the students understand the CS concepts addressed in the lesson.

Journal Entry: *Why do we need to have a variable and digital read in our code for one switch? If we used three switches instead of two how many possible behaviors could we have? Justify your response.*

Additional resources connecting Scratch and Arduino (optional)

- Translating from Scratch Activities (Scratch Translation Lesson Notes.pdf).

Crafting and programming

- Pairs will complete the rest of Mural Storyboard.pdf.
- Pairs will continue programming. One member may want to work on the program while the other is crafting. Codes can be tested by alligator clipping wristbands, paper circuits, or LEDs to the Circuit Playground. At this stage, there could be extra Circuit Playgrounds that could be made available for testing at your discretion.

- Ensure that each member gets a chance to craft and program.
- Extension: Groups who finish early can program all *three* switches for their panel of the mural.

Debugging Activity

- Assign half of the pairs to Debugging Fun Mural1 Handout and the other half to Debugging Fun Mural2 Handout.
- Have them complete #1 in pairs.
- Have students load the corresponding doc files: debuggingFunMural1 code or debuggingFunMural2 code.
    - Have them run the code in pairs and answer #2.
        - Alternatively, you can run the program for the students as a whole class and have them complete #2.
        - *Note:* The code will work with the onboard Circuit Playground light, so no LEDs need to be connected. Similarly, *any* LED connected to one of the pins on the Circuit Playground will work.
- Have students complete their debugging assignment.
- Have each pair form a group with another pair that had the other debugging assignment, and have each pair share their scenarios and answers with each other.
- See Debugging Fun Mural1 Solution and Debugging Fun Mural2 Solution.

Crafting and Programming

- Groups finish crafting and programming their mural piece.

Assemble the class mural and explore

- Have students interact with different parts of the mural to discover the different programmed behaviors of the different pieces.

Design Notebook entry

- Students can reflect on one or more of the following topics. They should include details and examples:
    - What are some things you learned from working in a group on this project?
    - What are you most proud of from this project?
    - What parts of this project were the most challenging?
    - Was there something you would do differently, if you were to do the project again?
    - What are some tips from this project that you would want to share with someone just starting out?
    - Thinking back on this project, what are some tips that you would like to remember for future projects?

Interactive Mural Sample Rubric

DO YOU HAVE?	POINTS POSSIBLE	POINTS EARNED
CRAFTING		
1. Each electrical component is securely sewn in on each pin (neatly stitched through at least three times).	5	
2. The running stitches are neat, evenly spaced and secure (the thread is pulled flat on both sides of the fabric, stitches are about a ¼ inch in length, there are no stray or tangled threads or big accidental knots, etc.).	5	
3. Starting and ending knots are neat and secure.	5	
Comments on Crafting:		
CIRCUITRY		
4. CircuitPlayground pins are clearly labeled (i.e., A4, A3, 5, 6, etc.) on the circuit diagram.	5	
5. Positive and negative pins/lines are distinct on the circuit diagram (i.e., clearly marked and not crossing).	5	
6. The system functions when a Circuit Playground is powered.	5	
7. There are at least four independently controllable LEDs (i.e. connected to different pins).	5	
Comments on Circuitry:		
CODING		
8. Each input and output has been declared in the Naming Section.	5	
9. Each pin is set to output or input as needed in the Setup Section.	5	
10. There are a total of four distinct conditions in the code.	10	

11. The panel has at least four different programmed lighting effects.	10	
12. Code is well commented – there are comments on each named line at the top, on at least half of the input/outputs in the setup section, on the conditions, and some description of each lighting pattern.	5	
Comments on Coding:		

COLLABORATION

13. Each person fulfilled individual responsibilities (half the sewing and half the programming).	5	
14. Pairs worked together to make overall design decisions & manage time together.	5	
15. The project reflects that the makers attended to the class theme and is visually cohesive with the other panels.	5	
Comments on Collaboration:		

OVERALL

16. Design Notebook entries were completed.	5	
17. Aesthetics: The design appears to be purposeful and/or personal. Care has been taken in the look and feel of the project. Students can justify their design if helpful (written, verbal).	5	
Comments on Overall:		

EXTRA CREDIT

18. Use of fading lights.	5	
19. Use of three switches to create additional effects.	5	

CRAFTING:		CIRCUITRY:		CODING:		TOTAL:	/100
COLLABORATION:		OVERALL:		EXTRA:			

Instructional Day 19: Programming Sensors as Inputs

Topic Description

This lesson introduces students to programming using a sensor as analog (vs digital) input.

Objectives

The student will be able to:

- Demonstrate understanding that analog input is for a range of input values
- Use the serial monitor to output sensor readings
- Write conditional expressions using the < and > operators

Outline of the Lesson

- Journal entry (5 minutes)
- Digital vs. Analog activity (20 minutes)
- Analog inputs (15 minutes)
- Light sensor storyboard (15 minutes)
- Program conditionals (15 minutes)
- Design Notebook (5 minutes)
- Recommended (25 minutes): Explanation of Digital versus Analog

Student Activities

- Complete journal entry.
- Participate in analog inputs discussion.
- Complete Light Sensor Storyboard worksheet.
- Program the conditions for various light patterns.
- Complete Design Notebook entry.

Resources

- Materials:
  - Circuit Playground
  - Micro usb cable
  - Computer
  - Flashlight/cell phone light
- https://learn.sparkfun.com/tutorials/analog-vs-digital
- https://www.diffen.com/difference/Analog_vs_Digital
- Difference between Analog & Digital (YouTube video): http://bit.ly/differenceAD
- "grades solution.sb" - from Unit 4
- TranslatingGradeSolution.pdf
- Light Sensor Storyboard.pdf
- Light_Sensor_Starter code
- Light_Sensor_Solution code

Teaching/Learning Strategies

Journal Entry: *If you had a sensor that could detect how bright the surrounding light is, what kinds of projects might you design?*

- Share with elbow partner.

Digital vs. Analog

- Show video then discuss as class the difference between analog vs. digital (4 min):
  - FYI w/video: https://www.diffen.com/difference/Analog_vs_Digital

- Video: http://bit.ly/differenceAD, can start video at clocks.
- Have the following discussion (10 min):
    - Think back to unit 1 when we learned how computers store and compute using binary, zeros and ones making up bits and bytes. The binary the computer uses is digital information or data. Analog is what happens in our physical world. The computer peripherals such as the keyboard convert your analog key strokes into digital information. The computer converts the digital information into an analog image on your monitor for you to read.
- Optional Journal Entry (6 min): *Why is it important to know what is digital and what is analog when you write code for a microcontroller?*

Analog inputs

- Hand out Light Sensor Storyboard.pdf worksheet.
    - In pairs, have students do #1.
    - Have them locate the placement of the light sensor on the Circuit Playground (A5).
- Demo the Circuit Playground running the program Light_Sensor_Solution.ino (up front by the teacher).
    - Connect the Circuit Playground.
    - Move a flashlight (or the light on your phone) closer to the light sensor. Do the same thing with your hand to block light.
    - Ask the students to describe how they think the behavior of the Circuit Playground is influenced by the light and your hand.
        - Guide them to discover that the brighter the light reaching the light sensor, the more lights are turning on.
    - The variable "sound" in Light_Sensor_Solution code can be set to "true" if you'd like to turn on beeping sounds that change with light intensity.
- Show how to display the serial monitor.
    - Click on the magnifying glass icon in the upper right hand corner of the Arduino window or Tools->Serial Monitor in the menu. (In Codebender, the Serial Monitor is on the upper right hand corner.)
    - Point out how the code "prints" to the serial monitor.
    - Redo light sensor flashlight/hand demo while displaying the serial monitor so students can see how the values change.
    - Have students guess what the highest and lowest possible brightness levels are.
    - Have students guess how the number of lit lights on the Circuit Playground is determined based on the brightness.
        - Guide them to identify ranges of values for each number of LEDs that light up on the Circuit Playground.
- Show students "grades solution.sb" from Unit 4 (if applicable).
    - Ask them how they wrote conditionals that worked on ranges of values.

    - Show them the side by side comparison from the last lesson of the Scratch grade code with the equivalent Arduino code from TranslatingGradeSolution.pdf.
    - Point out that in Scratch, they only used "<" and ">" (less than, greater than) because it is not easy to program the equivalent to "<=" and ">=" (less than or equal to, greater than or equal to) in Scratch. But in Arduino we could write:

```
if (grade >= 80 && grade < 90)
```

Storyboard

- In pairs, have students load Light_Sensor_Starter code.
- Pairs will complete Light Sensor Storyboard.pdf.

Program conditionals

- Have students complete the Activity Section of their program and test it.
    - Students' solutions may vary from the sample solution (Light_Sensor_Solution) due to the brightness of classroom, etc.
- Extensions:
    - Students could add more conditions based on the brightness level. Currently, the lights turn on in pairs. They could create conditions so that there are ranges that turn on only 1, 3, 5, 7, or 9 lights.
    - Students can program the Circuit Playground to play different sounds based on the light sensor input.

Design Notebook

- What are some pro tips you would like to remember about using the Circuit Playground for analog input? Record some words, diagrams, etc. in your design notebook.

Instructional Days 20-31: Human Sensor Project

Note: Please be aware that aluminum patches will need to be made in advance of the "Testing Aluminum Foil Sensors" lesson. Please plan for the extra prep time.

Topic Description

Students will design and craft a Human Sensor Project.

Objectives

The student will be able to:

- Apply their knowledge of E-Textiles to create a human sensor project
- Craft a sensor from aluminum foil
- Demonstrate understanding that humans can vary in their conductivity when interacting with the aluminum foil sensor

Preparation

- Before the project begins, it is helpful to iron aluminum foil to the iron-on adhesive in large sheets (see E-Textiles Technical Guide: Do-It-Yourself Sensors, pg. T54). These will be necessary for students' patches.

Detailed Outline of the Lesson

DESIGN project
- Introduce Project (20 minutes)
  - Brainstorming (25 minutes)
  - Introduce Portfolio (10 minutes)
- Design Human Sensor Project (50 minutes)
  - Design Notebook (5 minutes)

CONSTRUCT & TEST LIGHTS
- Crafting and debugging of lights (105 minutes)
- Design Notebook (5 minutes)

LESSON: SENSOR RANGES
- Testing aluminum foil sensors (reading a range in a program) (30 minutes)
  - Debugging Activities (35 minutes)

CONSTRUCT & TEST PATCHES
- Creating aluminum foil sensors (45 minutes)
  - Design Notebook (5 minutes)
- Crafting, programming, and debugging (155 minutes)
  - Design Notebook (5 minutes)

PORTFOLIO
- Portfolio Formation (110 minutes)
- Human Sensor Demos (55 minutes)

Student Activities

- Participate in project introduction.
- Brainstorm human sensor project ideas.
- Participate in portfolio introduction.
- Design their Human Sensor Project (aesthetic design, circuit design, storyboard for code).
- Respond to Design Notebook entry.
- Craft and debug light portion of project.
- Respond to Design Notebook entry.
- Test aluminum foil sensors.
- Debug existing code.
- Design and craft aluminum foil sensors.
- Respond to Design Notebook entry.
- Craft, program, and debug project.
- Respond to Design Notebook entry.
- Assemble portfolio.
- Demo human sensor project to class.

Resources listed on next page. →

Resources

- Materials for each student:
  - Circuit Playground
  - Computer
  - Micro usb cable
  - Li-Po battery
  - ~ 6 LilyPad LEDs
  - Pre-ironed aluminum patches
- Materials for each table group:
  - Craft glue
  - Thread wax
  - Conductive thread
  - Sewing thread
  - Seam rippers
  - Rulers
  - Black and red pens
  - Scotch tape
  - Alligator clips
  - Felt sheets and scraps of various colors
- Human Sensor Rubric options:
  - Portrait formatting
  - Landscape formatting
  - One page (summary)
  - Page E56
- Human Sensor Storyboard.pdf
- ReadYourHumanSensor_SAMPLE.ino
- Testing Aluminum Foil Sensors
- Human_Sensor_Starter_BASIC code
- ECS_HumanSensor_Starter_MOREHELP code
- Example of fully fleshed out human sensor project code: Human_Sensor_PizzaCart_1
- Debugging Activity:
  - Debugging Fun Sensor1 Handout
  - debuggingFunSensor1 code
  - Debugging Fun Sensor1 Solution
  - Debugging Fun Sensor2 Handout
  - debuggingFunSensor2 code
  - Debugging Fun Sensor2 Solution
  - Debugging Fun Sensor3 Handout
  - debuggingFunSensor3 code
  - Debugging Fun Sensor3 Solution
  - Debugging Fun Sensor4 Handout
  - debuggingFunSensor4 code
  - Debugging Fun Sensor4 Solution
- E-Textiles Technical Guide
  - Sample Human Sensor Projects (pg. T65)
  - Creating a handcrafted human sensor (pg. T54)
  - Troubleshooting Electrical Problems (pg. T34)
  - Troubleshooting Code (pg. T40)
- Human Sensor Student Portfolio Assignment (pg. E60)
- Exemplar portfolios and portfolio sample pages (folders in file package)
- Tips for Using Evidence Effectively handout
- Sample Portfolio Rubric (pg. E61)

Teaching/Learning Strategies

Introduce Project

- Project can modify an existing textile object like a hoodie, a stuffed animal, or something flat like a placemat, or can be made from scratch using felt.
  - Demo teacher project.
    - Point out the aluminum foil sensors and how the circuit between them has to be complete for the project to work.

    - The sensors give different readings, just like the light analog sensor.
    - Larger sensors usually give a wider range of values.
  - See E-Textiles Technical Guide: Electronic Textiles Design Ideas: Activity 4: Human Sensor Project (pg. T65) for examples.
- Talk about humans as conductive entities (we're mostly made of water!). Computers can measure the electricity we conduct with our human body.
  - Make connections to Human-Computer Interaction (can connect to other types of wearables such as Fitbits or heartrate shirts).
  - If making a wearable, where would be appropriate places to put patches that people might touch?
- Share Human Sensor Sample Rubric.

Brainstorming

- Have each student take a few minutes to come up with many different project ideas that can be used by them or someone else in the class.
- Have students share their ideas with elbow partners.
- Students will pair-share a few ideas with the whole class. They can be written on the board so undecided students can get inspiration.
- Have students decide what they want to do for their project. If they decide to use something from home, they should bring it in for the design phase.

Introduction of Portfolio

- Introduce the idea of the portfolio. Instead of a test or just a working project, the final evaluation of this project will include the project itself and a reflective "portfolio" where students can share things that went well, issues they had to address, and tips and tricks for future students working on this project.
  - See Portfolio Assignment description and Sample Portfolio Rubric.
- Share the two exemplar portfolios as well as sample pages (folders are in your file package). These are from other students, from ECS and elsewhere. Feel free to use the Tips for Using Evidence Effectively handout to help guide the discussion. Discuss as a class:
  - What makes a good portfolio? What helps you to understand the challenges people worked through?
  - What helps to make "evidence" understandable? (i.e., annotating pictures, putting an arrow to problem areas)
  - What could help you illustrate your progress? (i.e., taking pictures every day, taking screenshots of the code with error messages, noting things that you fixed along the way)
- Students may want to be especially aware of things that don't go as they planned, changes they make, and solutions they come up with. These will be written out for several journal questions, which they can use to make creating the portfolio easier. They can also take pictures during the creation of the project to use later.

- They should keep all versions of their circuit diagrams, journal responses, and any other things they want to note.
- *Every day before leaving class:* Take a picture of your project or save a version of your code. These can be used as artifacts in the portfolio.

Design Human Sensor Project

- Have students create a rough aesthetic drawing that focuses on design (what the project will look like, where electrical components things will be placed).
  - This should include the shape of their aluminum foil patches. These do not have to look the same, though they should be of appropriate size for good conductivity (*at least* one-inch in diameter, bigger is generally better).
  - Have students take a picture of their drawing!
- Have students create a circuit diagram (blueprint):
  - Labeling the pin number next to each component and indicating negative and positive lines in different colors will make crafting and programming easier.
  - Have students take a picture of iterations of their design and circuit diagrams.
  - *Note:* One aluminum foil sensor must be connected to pins 6, 9, 10, or 12 only. Only those pins can be used to take analog readings.
    - *Note:* On the Circuit Playground, four ports (6, 9, 10, and 12) serve as two separate pins. In other words, there are two pins connected to the same physical spot. However, these pins have different features. To cue the analog INPUT feature (to read a sensor) you must program the A pin on the port.
      - 6 shares A7
      - 9 shares A9
      - 10 shares A10
      - 12 shares A11
    - Example: To use A11, you would connect your sensor patch to 12/A11 and in your code use pinMode (A11, INPUT); and analogRead (A11);
    - ***WARNING:*** Do NOT use the Vbatt for anything. Ever. The voltage is too high and will burn out your components!
  - The other aluminum foil patch should be connected directly to a negative pin.
    - TIP: Other components can be connected to the negative patch or the specified GND pins as a ground.
  - Other pins on the Circuit Playground can be made into additional grounds to make connections easier.
    - Just program the pin as OUTPUT and digitalWrite it to be LOW.

- If students do this, make sure they indicate these changes clearly on their blueprint.
- Only pins 6/A7, 9/A9, 10/A10, and 12/A11 support analog reading (i.e., reading a range of values from a sensor).
  - Only pins 6, 9, and 10 analogwrite (i.e., writing a range of values to a light or sound output). For lights that fade, one of these pins must be used. *Note:* Fading is an optional extension or extra credit on the project.

Design Notebook: *Based on your design, what are some things you'd like to keep in mind while crafting your Human Sensor Project? (5 min)*

Crafting and debugging of lights

- Have students complete #1, #2, and #4 of Human Sensor Storyboard.pdf.
- Have students sew all of their lights and the Circuit Playground, following their blueprints.
- Testing human sensor project lights:
  - Students should test each light with a simple program that just turns the light on. They can use Human_Sensor_Starter_BASIC code as a starting point for their program.
  - Have students debug as necessary (lights that don't turn on due to faulty sewing, etc.).
  - You can use this as an opportunity for a class discussion: Why is it useful to test your lights just by turning them on (as opposed to creating a blinking pattern)?
  - *Note:* Students often have a hard time *breaking down a problem* into testable parts. *Simply turning the lights on*, without coding for sensors or complex lighting patterns, can isolate whether the circuitry is functional. *Remind students of the problem solving strategies in Unit 2!*

Design Notebook: *Choose one or more of the following questions to answer (5 min):*

- What were some modifications you had to make in your design?
- What were some bugs you had to fix and how did you fix them?
- What was one or more of your biggest challenges so far?
- *Cue students to remember that they can use these answers in their portfolio.*

Exploring aluminum foil sensors

- Have sets of foil patches (enough for 1 set per pair of students).
  - A set would be two similar sized patches.
  - Have various sized sets: small, medium, large, or extra large (optional).
- Distribute a set of foil patches to pairs of students.
- Connect one of the student's partially-completed Circuit Playground project using alligator clips to the provided sensors.

  - Have each pair load the code for reading the human sensor patches: ReadYourHumanSensor_SAMPLE.
  - Invite students to connect patches based on the code. One patch must be connected to ground (negative). The other patch must be connected to pin 6, 9, 10, or 12 for analog read.
    - Remind students that when the analog pins are used as input, they have different numbers in the code: Pin 6 = A7; Pin 9 = A9; Pin 10 = A10; Pin 12 = A11 (the code uses A11/Pin 12).
- Class Discussion: Experimenting with patches while looking at sensor readings. *(Consider/adapt the following questions & experimentation.)*
  - What happens when one of you squeezes both patches?
  - What happens when you touch the patches together? Why do you think it goes down that low?
  - What happens if each of you put one hand on one and then hold hands? (optional but quite fun to experiment with)
- Testing sensor ranges
  - In pairs, have students complete the first row in Testing Aluminum Foil Sensors handout.
  - Leave this form at each computer (either digitally open, or physically in front of the computer).
  - Each pair will then rotate to others' projects, writing down the ranges on the form at each computer as they squeeze. Students should cover as many stations as they can within a limited amount of time.
  - Have students return to their original stations and look at the results of their classmates on the computer.
- Class discussion:
  - As a class: Invite students to share the overall ranges that each group came up with (students may note some variance amongst individuals).
  - Ask students the following questions:
    - Why are these ranges different?
      - Answers might refer to the size of the patches or individual's personal conductivity.
    - Is there a relationship between the patch size and the number? Talk to your partner about that.
- Number line discussion:
  - Draw a number line on the board to talk through how students can translate the possible ranges on their sensor to useable ranges that they can use to program different effects.
  - Ask students the following questions:
    - Where might you put different ranges, such as "not touching" or "squeezing really hard"?

- How can you write those mathematically? (i.e., with >, <, = expressions)
- Students complete the handout in pairs:
  - Tell students to take a look at the ranges that were written on their spreadsheet and to think about what ranges would be good for all users. Ask, "If you wanted everyone in this room to use this project, what would be the best ranges for the different patterns? Talk about this with your partner, then type some numbers in."
  - Groups can come up with appropriate ranges and accompanying descriptions they would use for these sensors. Descriptions can be in their own words such as "not touching" or "lightly touching."
- Class discussion:
  - Have students share answers to questions at the bottom of the handout.
  - Ask what existing code could help us manipulate the conditionals for our human sensor projects.
    - Guide them to the light sensor code from Day 19.
- Optional extension: Students could begin coding some lights to work with the practice sensors, if they figure out the ranges quickly. (They could use the light sensor code as a model for how to do this.)

Debugging activities

- Assign pairs to work on one of the following (distribute them across different pairs): Debugging Fun Sensor1 Handout, Debugging Fun Sensor2 Handout, Debugging Fun Sensor3 Handout, or Debugging Fun Sensor4 Handout.
  - Have them complete #1.
- Have students load the corresponding code files, which they can test using their own Human Sensor project: debuggingFunSensor1 code, debuggingFunSensor2 code, debuggingFunSensor3 code, or debuggingFunSensor4 code.
  - If they use their own Human Sensor Project, make sure to remind them to modify the pin numbers to match their project in the Naming Section. Then have them run the code and answer #2.
  - Alternatively, you can run the program for the students as a whole class and have them complete #2.
  - Have students complete their debugging assignments.
- After finishing, have each pair join another pair with the other debugging assignments and share their scenarios and answers.
- See Debugging Fun Sensor1 Solution, Debugging Fun Sensor2 Solution, Debugging Fun Sensor3 Solution, and Debugging Fun Sensor4 Solution.

Creating & testing aluminum foil sensors

- Set up an ironing station. Have students craft the aluminum foil patches for their individual projects, iron them on, and sew their sensor circuits. Have students debug as necessary (e.g., short circuits, loose stitches, patches that are too small).

  - See E-Textiles Technical Guide: Do It Yourself Sensors (pg. T54).
  - Using the earlier activity/handout from Testing Aluminum Foil Sensors (see below) as a guide, ask students to test their own patches and write down the minimum and maximum ranges they can achieve. Students define appropriate ranges for their light behaviors and insert them in #3 of their Human Sensor Storyboard.pdf.

Design Notebook: *Write down/diagram some tips for crafting projects that use aluminum foil sensors.*

Finish crafting, programming, and debugging

- Have students complete their Human Sensor Storyboard.pdf.
- Have students finish crafting, programming, and debugging their human sensor projects.
  - If students need more guidance programming the ranges for their senses they may use: ECS_HumanSensor_Starter_MOREHELP code.
- See E-Textiles Technical Guide for circuit debugging tips (pg. T34).
- Remember to have students take pictures regularly and jot notes in their Design Notebooks about changes they make.

Design Notebook: *Reflecting on your project, how did it turn out compared to your original plan? What changed?*

Portfolio Formation

- Re-introduce the Student Portfolio Assignment (can be passed out as a one-page handout, see below). There are three sections: description of final project, description of process (revision or challenges), and reflection.
- Teacher may want to re-model sample portfolios or sections of them. Ask students to remember what makes a portfolio helpful for the viewer. If helpful, use the exemplar portfolios and sample pages for reference again (folders in your file package). Share the Tips for Using Evidence Effectively handout as a reference for students.
- Have students compile their artifacts, and choose which challenges or revisions they want to write about.
- See Sample Portfolio Rubric.

Share Human Sensor Projects!

- Have students share their human sensor projects as a capstone to the unit.
- Options:
  - Gallery walk or e-textiles show. Have students walk around the class, interacting with projects. For feedback: Put a piece of paper by each project and ask students to write down something they like.
  - Have students do a brief show and tell (if time is a problem, use a timer for 1-2 min per student).

Human Sensor Rubric (100 points total)

This is a long and detailed rubric. Feel free to simplify it according to your needs.

Note: This rubric is also available in its own separate file:

- Portrait formatting
- Landscape formatting
- One page formatting (simplified for printing & reporting, but *please use the full versions above & below* to know what each value means, i.e., what 10, 6, 3, and 0 points mean in each category).

Design & Craft

	10 PTS	6 PTS	3 PTS	0 PTS	POINTS EARNED
BASIC REQUIRE-MENTS	There are at least four independently controllable LEDs attached (i.e. connected to different pins) and two conductive patches in the design	Most of the project components are there but not all. Some LEDs might be missing or the patches might not be connected. Still, the project mostly meets the basic requirements.	This project needs more work. Some components are there (a couple of LEDs, maybe a sensor patch) but a lot is missing.	No lights, no sensor patches, basically no project at all.	
	10 PTS	**6 PTS**	**3 PTS**	**0 PTS**	
DESIGN NOTE-BOOK	All Design Notebook entries completed.	Most Design Notebook entries completed.	Only a few Design Notebook entries completed.	No Design Notebook entries are complete.	
	15 PTS	**10 PTS**	**5 PTS**	**0 PTS**	
SEWING *(continued on next pg.)*	• Each electrical component is securely sewn in on each pin (neatly stitched through at least three times). • Stitches are neat, evenly spaced and secure (The thread is pulled flat on both sides of the fabric, stitches are about a ¼ inch in length, there are no stray threads or big accidental knots, etc.). • The back of the project is as neat as the front. Knots are secure	• Some electrical components are sewn in well (neatly stitched through at least three times), but some are not. • Stitches are overall neat but some are uneven in ways that affect the longevity of the project (i.e., could be snagged or pulled easily). • Some knots are well secured but others are loose or fraying in ways that could compromise the circuitry.	• Most electrical components are insecure and loose, affecting the steadiness of the electrical connections. • Stitches are big and untidy; this could affect the project in the long term (i.e., they could be snagged or pulled out). • The back of the project is a mess and there are many loose threads or fraying knots that are compromising how	• This project is so poorly sewn it's almost non-existent. Everything is loose or unconnected. • The back is a mess with loads of touching threads. Pieces are falling off. • Or maybe you used non-conductive thread to sew electronic pieces on.	

SEWING *(continued)*	and tight. No loose threads are present.		the project works.		
	Comments on sewing:				
	15 PTS	**10 PTS**	**5 PTS**	**0 PTS**	
DESIGN	• The design appears to be purposeful and/or personal. Care has been taken in the look and feel of the project. How it looks is intentional. • Decorative parts are sewn or glued on with care • The project is finished. All final touches are done. • Students can justify their design if helpful (written, verbal).	• The design is okay, but not very personal or purposeful. It seems a bit thrown together. • Decorative parts are somewhat haphazardly attached. • The project isn't quite finished. Some final touches are clearly needed.	• Did you really put much thought into the design? It seems as though you just threw it together without thinking about who would enjoy it or how it would look. • Decorative parts are haphazardly attached and the project looks messy. • The project is very unfinished: pieces are missing, parts were slapped together last minute, the whole thing is far from coming together.	There is no design; there is basically no project. No effort at all has been taken on actually making something that is intentional for whatever reason.	
	Comments on design:				

Circuitry

	15 PTS	10 PTS	5 PTS	0 PTS	POINTS EARNED
DIAGRAM	• The diagram is clear, readable, and functional (it would work if constructed). Someone else could use this to make the design themselves! • Circuit Playground pins are clearly labeled (i.e., 0, 1, 2, 12, etc.) on the circuit diagram. • Positive and negative pins/lines are distinct on the circuit diagram.	There is a circuit diagram and it would functional, but it needs to be improved in order for it to be thoroughly useable. Perhaps the pins or polarity (plus & minus lines) are not clearly labeled.	The circuit diagram needs a lot of improving. It would not be functional if constructed (there are crossed lines) and it is almost entirely unlabeled. It's hard to tell what part is which and how things are supposed to be connected.	There is no circuit diagram for this project.	

	5 PTS	3 PTS	1 PT	0 PTS	
LIGHTS-ON	• The lights function when a Circuit Playground is powered. • *Note:* Patterns are considered below. This is just whether the lights turn on - i.e., a test of basic circuitry and not code.	Most of the lights turn on but not all.	One or two lights turn on when the Circuit Playground is connected to power.	No lights function when the Circuit Playground is connected to power.	

Comments on circuitry:

Coding

	5 PTS	3 PTS	1 PT	0 PTS	POINTS EARNED
FOUR COMPLETE LIGHTING PATTERNS	There are four functional lighting patterns (i.e., would work if circuits were perfect). Each is different in some way.	Two or three lighting patterns have been programmed. Each is different in some way.	Only one lighting pattern has been programmed.	No lighting patterns have been programmed.	
	15 PTS	**10 PTS**	**5 PTS**	**0 PTS**	
CODING	• The code is great! • Each input and output has been declared in the Naming Section and set to output/input as needed in the Setup Section. • Needed variables are entered correctly and used consistently. • Conditionals are programmed effectively and are functional.	• The code is there but some minor improvements are needed (mostly syntactical). • Examples: There are extra (or missing) brackets; a variable/pin has been mislabeled; an input/output is missing or mislabeled.	• The code is there but a lot of improvements are needed. • Some sections need to be finished (maybe not all variables are listed, inputs/outputs are missing, conditionals are poorly programmed). • Or perhaps the conditionals are semantically mis-programmed.	• There is no code at all. • Or it is so poorly done that there might as well be no code.	

	5 PTS	3 PTS	1 PT	0 PTS	
COMMENTED CODE	Code is well commented—there are comments on each named lined at the top, on half of the input/outputs in the setup section, on the conditions, and some description of each lighting pattern.	Code is somewhat well commented. There are a few comments there saying what different parts do, but is it not consistent.	There are a couple of comments on the code but it has not been done consistently.	There are no comments on the code or they are incorrect (i.e., a direct copy/paste from sample code that no longer apply to the current code—like a mislabeled pin).	
	5 PTS	**3 PTS**	**1 PT**	**0 PTS**	
SENSORS	Aluminum foil sensors work and detect at least four variable levels of touch. The sensors are programmed to be continuous (or very intentionally non-continuous).	• Foil sensors are programmed but there are some minor problems. • Perhaps the ranges are not continuous or are miswritten (i.e., a minor problem with the operators, not using <= or >=). • Perhaps there are only 3 coded ranges.	• Some attempt has been made to code the ranges, but these have not been tested and do not work well. • The ranges are not continuous; there are large unintentional gap. • Perhaps there are only 1-2 coded ranges.	Aluminum foil sensors are not coded beyond the starter code.	

Comments on coding:

Extra Credit

	POINTS EARNED
Use of additional coding elements such as fading, random light patterns, or sound.	
Additional conditions (i.e., more than 4 lighting patterns or similar outputs) triggered by the sensor.	
Some other form of going above and beyond on this project. *Specify:*	

Totals

DESIGN & CRAFT:		CIRCUITRY:		CODING:		EXTRA CREDIT:		TOTAL:	/100

Portfolio Assignment for Electronic Textiles: Human Sensor Project

Note: This assignment is also available in its own separate file.

PART I: Description of final project (at least one section/page/slide)

WRITTEN	EVIDENCE
Answer all prompts: • What does your project look like? • Where are the sensors and LEDs placed? • What are your four different light patterns and how are they activated?	Include pictures of your project to help the reader understand what you made. Make sure to show the placement of your LEDs, sensors, and the Circuit Playground.

PART II: Process of making your project (at least two sections/pages/slides)

WRITTEN	EVIDENCE
Write about two 'process moments', which can be either: • 2 challenges you faced (include problems and solutions); OR • 2 revisions you made (include the changes and why); OR • 1 challenge you faced (include problem and solution) and 1 revision you made (include the change and why). *Please note, at least one moment must focus on your Arduino programming. The other may focus on your sewing, circuitry, design, or coding.*	For each 'process moment', help the reader understand what happened with pictures: • For a coding moment, include relevant screenshots of your code (include before/after). • For a circuitry or crafting moment, include relevant pictures of your project, circuit diagram, engineering notebook or journal entries (including before/after).

PART III: Reflection across the entire e-textiles unit (at least one section/page/slide)

WRITTEN	EVIDENCE
Describe one specific skill or thing that you improved upon over the course of the e-textiles unit. This may either be something concrete like your sewing skills, or something more abstract like your planning abilities.	Include images to help the reader understand your improvement. This may either be screenshots of code, pictures of your project or circuit diagram, pictures from your engineering design notebook or journal entries. You may include pictures from multiple projects, if you wish.

For all evidence, please remember to label your evidence and point to the areas your readers should look at! Refer to the **Tips for Using Evidence worksheet for advice.*

Portfolio Rubric (100 points total)

Note: This rubric is also available in its own separate file.

PART I - Project Description (25 points maximum)

	15 PTS	10 PTS	5 PTS	0 PTS	POINTS EARNED
WRITTEN PART	The section contains complete explanations for the physical and functional descriptions of the project. This means that students specify the placement of the electronic components in their project and describe their four different light patterns.	The section is mostly complete but may miss some parts of the description, either the physical layout of the components or the complete functional description of the project.	The section either contains only one of the project descriptions (physical or functional) or both the sections are incomplete.	There is neither a mention of the physical or the functional description of the project.	
	10 PTS	7 PTS	3 PTS	0 PTS	
USE OF EVIDENCE	Provides relevant evidence that demonstrates both the physical layout and the technical function of the project that is made legible and understandable through appropriate textual annotation and/or visual markers (e.g., captions, labels, arrows, highlights).	Provides relevant evidence that demonstrates both the physical layout and the technical function of the project but with no textual annotation or visual markers (e.g., captions, labels, arrows, highlights) to assist with legibility or understanding.	Provides some evidence that does not match or support the claims made in the written description.	No evidence provided in this section.	
Comments on project description:					

PART II - Process of making your project (50 points maximum)

	30 PTS	20 PTS	10 PTS	0 PTS	
WRITTEN PART	The section contains complete explanations of both the episodes (two challenges or two revisions or one of each) with at least one of them involving coding. The challenges should include the challenge and the resolution, and the revision should include the change and the rationale for it.	The written section may contain either incomplete descriptions of both the episodes or may have not mentioned coding in either of the episodes. Examples for incomplete descriptions - challenges without resolutions or revisions without reason for the revision of plan.	Explanation contains only one of the episode, either fully or partly. This maybe in any domain (coding, crafting, circuitry, designing).	Portfolio doesn't contain any written description of either the challenges or revisions.	

	20 PTS	14 PTS	6 PTS	0 PTS	
USE OF EVIDENCE	Provides relevant evidence to support both the episodes that is made legible and understandable through appropriate textual annotation and/or visual markers (e.g., captions, labels, arrows, highlights).	Provides relevant evidence that supports both the episodes but with no textual annotation or visual markers (e.g., captions, labels, arrows, highlights) to assist with legibility or understanding.	Provides some evidence that does not match or support the claims made in the written description.	No evidence provided in this section.	

Comments on project description:

PART III - Reflection across the unit (25 points maximum)

	15 PTS	10 PTS	5 PTS	0 PTS	
WRITTEN PART	Mentions one specific skill or the thing they improved upon in the reflection section with complete explanations. Complete explanation will include both the identification of skills and lessons about self, along with appropriate justification.	Mentions a skill or the thing they improved up, attempts to explain but lacks complete explanation. This can mean that they identify the improvement but does not explain why they think so.	Only mentions the skill or the thing they improved upon over time. Does not make an attempt to provide any explanation.	Does not attempt to reflect at all.	
	10 PTS	**7 PTS**	**3 PTS**	**0 PTS**	
USE OF EVIDENCE	Provides relevant evidence that supports the reflection and are made legible and understandable with appropriate textual annotation and/or visual markers (e.g., captions, labels, arrows, highlights).	Provides relevant evidence that supports the reflection but with no textual annotation or visual markers (e.g., captions, labels, arrows, highlights) to assist with legibility or understanding.	Provides some evidence that does not match or support the claims made in the written description.	No evidence provided in this section.	

Comments on reflections:

TOTALS

DESCRIPTION:		PROCESS:		REFLECTION:		TOTAL:	