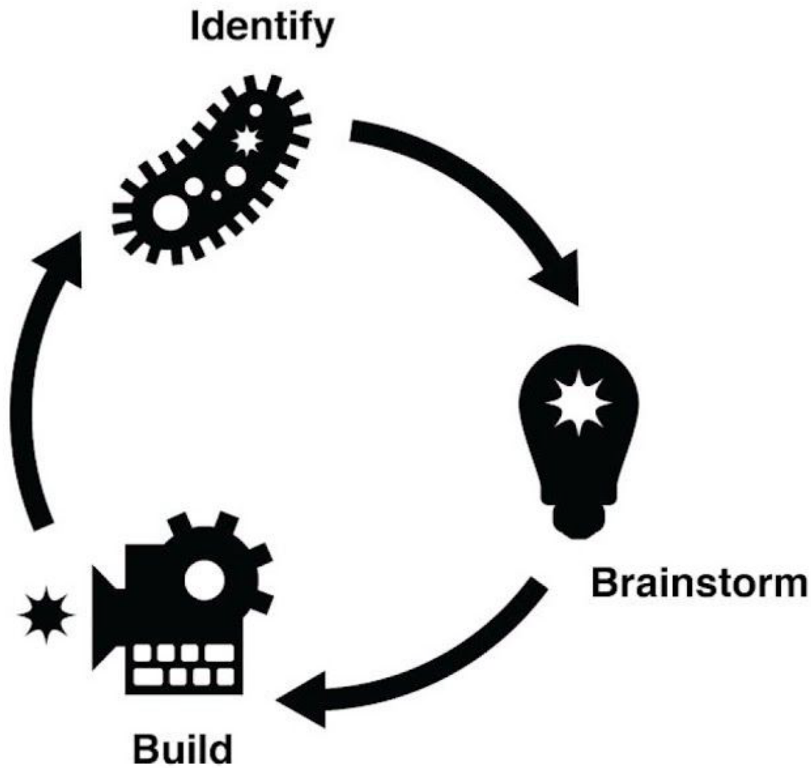


Invention × Nature

Future Humans

NYC Summer in the City
A New Approach to Summer Learning
Beam Center
STEM
mouse

Table of Contents

Program Overview

Dates and Times

July 2 – August 8, 2019
Monday – Thursday
8:30am – 2:30pm

In total, there will be 20 instructional days of Invention X Nature. There will be two additional field trip days.

Your Team

You will each lead this program in your own classroom, but you are part of a site-wide team that includes CUNY Tutors, Beam Center Program Managers, and Beam Center Program Assistants.

Beam Center Program Managers can provide support with pacing and scheduling, tool and material use, coordinating between classrooms, and can help clarify design challenges and curriculum design. Program Managers are also experienced educators and can help work directly with students. You can expect Program Managers to pop into your classroom and seek out ways to help. **We are not here to lead the class *or* evaluate your performance, we are only here to help. If you feel unprepared, please reach out.**

Beam Center Program Assistants can provide additional support with tool and material use as well as coaching students through the process of troubleshooting and building prototypes.

Beam Center Staff Contact

Bronx: M.S. 390
Beam Center Program Managers:
Kayla Miller kayla@beamcenter.org
Najm Qachi najm@beamcenter.org

Brooklyn: I.S. 096 Seth Low
Beam Center Program Managers:
Kyle Luntz kyle@beamcenter.org
JV Aine JV@beamcenter.org

Manhattan: Art and Design High School
Beam Center Program Managers:
David Golann david@beamcenter.org
Rachel Brown rachelbrown@beamcenter.org

Queens: Thomas A. Edison Career and Technical Education High School
Beam Center Program Managers:
Rachel Beane rachel@beamcenter.org
Andrew Cerito andrew@beamcenter.org

Staten Island: Ralph R. McKee Career and Technical Education High School
Beam Center Program Managers:
Yemi Afolabi yemi@beamcenter.org
Abib Coulibaly abib@beamcenter.org

Invention X Nature Program Directors
Lizzie Hurst lizzie@beamcenter.org
Allen Riley allen@beamcenter.org

Director, BeamWorks Programs
Matt Robinson matt@beamcenter.org

Who We Are

This summer, Invention X Nature is brought to you by Beam Center in collaboration with Mouse.

Beam Center's mission is to crystallize self-directed growth through ambitious, collaborative project-making.

We support and celebrate young people as producers of learning, culture and change who take bold steps towards personally meaningful futures and help guide compassionate, equitable, and vibrant communities.

We use old and new tools, technologies, and craft to honor the individual voice, celebrate the joy of producing something larger than ourselves, and inspire lasting wonder. We value the exchange of knowledge, perspective, and experience between youth and professional creators.

Mouse designs computer science and STEM curriculum on their online learning platform Mouse Create, trains K-12 educators, and engages students through the Design League and maker events. Mouse is committed to fostering greater diversity and humanity in STEM and empowering youth -- and all those that educate them -- to access and amplify technology as a force for good.

Invention X Nature Timeline

Your schedule will differ slightly day by day, but our recommended flow for the program is as follows.

DAY 1	AM	SESSION 1	Orientation
	PM	SESSION 2	Fundamentals: Journal
DAY 2	AM	SESSION 3	Fundamentals: Journal
	PM	SESSION 4	Documentation and Drawing
DAY 3	AM	SESSION 5	Fundamentals: Bionics
	PM	SESSION 6	Fundamentals: Bionics
DAY 4	AM	SESSION 7	Fundamentals: Bionics
	PM	SESSION 8	Fundamentals: Bionics
DAY 5	AM	SESSION 9	Fundamentals: Bionics
	PM	SESSION 10	Fundamentals: Bionics
DAY 6	AM	SESSION 11	Fundamentals: Bionics
	PM	SESSION 12	Fundamentals: Bionics
DAY 7	AM	SESSION 13	Domain 1: Chameleons
	PM	SESSION 14	Domain 1: Chameleons
DAY 8	AM	SESSION 15	Domain 1: Chameleons
	PM	SESSION 16	Domain 1: Chameleons
DAY 9	AM	SESSION 17	Domain 1: Chameleons
	PM	SESSION 18	Domain 1: Chameleons
DAY 10	AM	SESSION 19	Domain 1: Chameleons
	PM	SESSION 20	Domain 1: Chameleons
DAY 11	AM	SESSION 21	Domain 2: Bees

	PM	SESSION 22	Domain 2: Bees
DAY 12	AM	SESSION 23	Domain 2: Bees
	PM	SESSION 24	Domain 2: Bees
DAY 13	AM	SESSION 25	Domain 2: Bees
	PM	SESSION 26	Domain 2: Bees
DAY 14	AM	SESSION 27	Domain 2: Bees
	PM	SESSION 28	Domain 2: Bees
DAY 15	AM	SESSION 29	Synthesis: Future Humans
	PM	SESSION 30	Synthesis: Future Humans
DAY 16	AM	SESSION 31	Synthesis: Future Humans
	PM	SESSION 32	Synthesis: Future Humans
DAY 17	AM	SESSION 33	Synthesis: Future Humans
	PM	SESSION 34	Synthesis: Future Humans
DAY 18	AM	SESSION 35	Synthesis: Future Humans
	PM	SESSION 36	Synthesis: Future Humans
DAY 19	AM	SESSION 37	Showcase
	PM	SESSION 38	Showcase
DAY 20	AM	SESSION 39	Celebration
	PM	SESSION 40	Celebration

Invention x Nature Program Description

The program consists of three phases: **Fundamentals**, **Domains** and **Synthesis**. Each phase introduces new skills and challenges that students combine with what they learned in the previous phase.

- **Fundamentals:** (5 days) introduces the basic tools, materials, skills, and safety involved in the prototyping routine. It also includes the means by which students will document their work, which consists of a handmade journal and digital photography.

- Two **Domains** (4 days each): introduce specific design challenges inspired by the way living things move and react to their environment. Students will combine what they learned in Fundamentals with new circuits that include Arduino coding and physical computing.
- **Synthesis** (4 days): challenges students to bring together any tools, materials, skills, and code they choose from the previous phases to create an evolved creature prototype.

In Beam Center's Summer In the City program, **Invention x Nature**, Teachers will guide students through an iterative engineering design process themed around hands-on prototyping of interactive objects inspired by the natural world.

To prepare for the program, teachers will be trained by Beam Center Instructors to become **lead learners** and apply a set of skills, tools, and materials to a theme from nature.

Students begin the program by assembling their own **Prototype Journal** that they will use to develop projects over the summer.

During the first few days, students will be introduced to the **Fundamentals** of prototyping, which will include electronics, physical computing, and building cardboard mechanisms.

Students will then complete two **Domains**, design challenges inspired by the way that different animals survive and thrive. During this part of the program, they will be introduced to the particular fabrication skills, tools, and materials of that Domain and will create prototypes in collaboration with a partner.

In the final **Synthesis Project** project, students will design interactive prototypes that bring together a selection of everything they've learned to create a final prototype that brings together everything they've learned.

Online Resources

This public folder contains curriculum content in PDF format for Middle School and High School.

SITC.beamcenter.nyc

This public folder contains Arduino code and circuit diagrams for Middle School and High School.

Arduino.beamcenter.nyc

Prototyping in the Classroom

Safety Guidelines

1. No open toed shoes. Keep long hair, dangling clothing or jewelry tied back.
2. Put tools down and give teachers your full attention during demonstrations.
3. If you don’t know how: ask for help. If you do know how: offer help.
4. Respectful and clear communication is essential.
5. Everything in the space is a tool. Use them properly.
6. Keep work surfaces organized.
7. Use tools at designated stations.
8. Keep your fingers away from hot tips and sharp blades.
9. Be aware of your body in the workspace. Keep your eyes on the tool you’re using.
10. Set a good example. Do what you expect others to do.
11. Materials are shared by everyone. Use them purposefully.
12. Do not distract others when they are using tools.
13. Clean up is everyone’s responsibility.

Behavioural Guidelines

Beam Center trusts students to learn how to use real tools to make real things. To keep our trust, we ask students to listen to and follow instructions for how to use tools appropriately, and to treat every object in the shop as a serious tool.

Unsafe behavior is typically the result of student disengagement. Most behavioral issues can be resolved by redirecting energy back toward the project or by clarifying the project criteria.

If a student makes an *unintentional* mistake, we show them how to use the tool again and ask them to demonstrate correct usage. If a student makes an *intentional* mistake, we give them a chance to use the tool properly. If they continue to misuse the tool, they may lose the privilege of using that tool.

Guidelines for Teaching Safety

1. **Set a good example.**
    - We are always on stage when using tools. Do exactly what you expect your students to do. If you do it, they will too. If you don't do it, you can't expect them to.
2. **Provide clear instructions and discuss safety first.**
    - Demonstrate how to use each tool properly - tell them AND show them.
    - Be intentional about what's on the table when. Only hand out the tools and materials students will need within that work period.
3. **Don't hesitate to ask for help.**
    - We want to create an environment in which everyone is comfortable asking questions if they don't know how to do something. If you don't know how to use a particular tool or approach a task, just ask the Beam Center Program Manager.

What is a Prototype?

Invention X Nature is all about making **prototypes**.

A **prototype** is a first attempt at building something or realizing an idea. **Prototypes are like first-drafts**, but they should be functional, durable, and interesting to look at. They don't have to look perfect or polished. They are a way to share ideas with other people!

Invention X Nature is divided into three phases of collaborative prototyping. Each phase introduces new tools, skills, and materials and poses a design challenge for students working in groups.

During each phase, students follow the same **prototyping routine** for making prototypes:

1. Learn the circuitry, tools, skills, and materials
2. Hear the challenge, prompt, or theme
3. Brainstorm and play with materials
4. Make a plan and a diagram
5. Build
6. Revise
7. Reflect

The Prototyping Process

Often projects that involve making or prototyping begin with a problem to solve or a question to answer. Before beginning to create, think about whether you understand the prompt you've been given. Play with materials and explore ideas, and not to start with an idea of what you want to finished product to be. Some questions to consider.

- What is the problem/challenge I am trying to solve?
- Are there multiple ways to approach a solution?
- What is my end goal?
- What materials and ideas interest me?

You can help students by facilitating full-classroom brainstorms or grouping students based on their interests.

Drawing

Drawing is another way to communicate an idea and is especially helpful when you're working with other people. Artistic skills aside, information diagrams are a way to show someone how something might work, as a first step in building it. As long as the idea is communicated, the sketch is successful.

Guidance and prompts for making informational drawings

- Show how the circuit connects.
- Where is it interactive? What does it do?
- What's the scale? Write down measurements.
- How is it worn/held/attached or used?
- Add a title and label the parts.

Finishing Prototypes

When is a prototype finished?

Prototypes are “first draft” physical manifestations of an idea. A prototype is finished when it demonstrates the idea successfully. If you can’t find something to do on the prototype you were working on, improve or expand the idea, make it look better, make a version two, help someone else, or make something new.

Iterating on a prototype means that you’re improving on the function rather than improving the aesthetics.

Aesthetic decoration always comes last.

Prototype Presentation Guidelines for Students

Presentation Tips

- Face the crowd.
- Explain the steps you took to make it.
- Demonstrate how it works and what it does.

Prompts for Discussing Prototypes

- What’s the most interesting feature of your design? Why?
- What were the different steps you took to get your project to work?
- What was the hardest problem to solve?
- Did you have to do something a few times to get it to work? What?
- If you had more time, how would you improve your project?

Facilitation Guidelines

How to Demo Tools

Before demonstrating step-by-step instructions, lay an example of each of the components out on a table and explain what each is, what it does, and how to use it safely.

Classroom Management

Build Trust
Students need to feel safe before they allow themselves to take creative risks. Building trust is essential for fostering a sense of safety. Teachers can earn the trust of students by sharing knowledge of tools and materials and by communicating respectfully using constructive criticism. Students can earn the trust of teachers by listening closely and following instructions for using tools safely.

Redirect Energy
Disinterested students may just need a change of role. Help students who are disengaged become students who have skills that no one else has. Ask questions in order to help them figure out what interests them and give them entry points. Many behavioral issues can also be resolved by clarifying the design challenge criteria.

Constructive Criticism
Non-working prototypes do not indicate failure. To build prototypes, students must experiment, take risks, and try things that might not work. If a student's prototype does not work, it is important to provide constructive criticism rather than call it a failure.

Prompts for Constructive Criticism

- I like the way you ___
- I think ___ is successful
- This made me interested in exploring ___ more.
- I'm wondering about ____. Can you say more?
- What would you do differently?
- ____ but _____ because _______. (Compliment Sandwich).

Tool and Safety Mistakes
If a student makes a mistake using a tool, show them how to use the tool again and ask them to demonstrate correct usage.

Losing Tool Privileges
If a student repeatedly uses a tool in an unsafe way on purpose, they may lose the privilege of losing that tool.

Student Participation
If a student is visibly disengaged but is not creating a distraction, they may need a break. Interrupting long work periods with intentional breaks can be helpful. It is OK for students to quietly take a break to listen to music or check their phones as long as they remain in the classroom.

Students who aren't engaging with the main activity are welcome to work on alternative activities, tasks, or challenges that are constructive and non-disruptive.

Taking Breaks
Students may need to take breaks during prototyping periods.

Supporting Collaboration

Forming and Maintaining Student Groups
All of the projects in Invention X Nature are collaborative by nature. We recommend that every project is created by a group of 2-3 students. It is important that students work in consistent groups for the duration of a project. We recommend keeping a record of student groups to support consistency. It is OK for students to create new groups after each project.

Students Working Individually
If a student expresses the desire or need to work individually, it is up to your discretion as to whether this is appropriate.

Distributing Materials to Students

Don't give students all of the materials at once. Only give out materials for the current project. CUNY Tutors may be helpful in distributing materials.

It is recommended to only offer paint markers towards the end of a project. Students should spend the first few sessions focusing on form and function, and save color and decoration for last.

Music and Photography

Listening to Music
Listening to music can help create a positive laboratory environment. We recommend creating a system for students to take turns choosing music to listen to aloud in the classroom. If students listen to headphones during work periods, they must only have one earbud in. Students should not use headphones during instruction or while using any power tool.

Student Photography
Photo documentation is an important part of this program. It is OK for students to use their phones to photograph their work and each other, as long as they have permission.

Prototype Policies

Taking Projects Home
Student prototypes are collaborative, and this complicates the idea of a student taking their project home. Students may also need to disassemble earlier prototypes and reuse materials. For these reasons, it is important that students do not take their projects home until the last classroom day of the program. If the final day of STEM Summer In The City is a field trip, students should take their projects home the previous day.

Evaluating Students

Criteria for Success
There is no formal criteria of success for students in Invention X Nature. Success looks like students making something that they’re proud of, gaining skills, and becoming part of a community.

Positive behavior to look for:

- Skills using tools and materials
- Expressions of confidence
- Expressions of curiosity
- Understanding of how things work
- Ability to explain how they did something
- Ability to respond constructively to other people’s work

Classroom Setup Guidelines

Setting up Spaces

Students will often work standing and will move around the classroom during Invention X Nature. They will also work collaboratively using tools. To accommodate this, there are several spaces that will need to be prepared before the program begins. They are:

Meeting and Demonstration area
In-Progress Prototype Storage
Tool and Material Storage
Prototype Gallery
Personal Item Storage
Work Surfaces
Code Upload Zone
Hot Glue Zone
Soldering Station
Cardboard Cutter Zone
Recycling and Garbage

Meeting Area

Designate an area that stays (or can be made) clear which can be used to present ideas and demonstrate tool usage. This area could also serve as a spot where students can share their projects with the group. It should include a tabletop and a place to exhibit freestanding or wearable projects.

Prototype Gallery

Set up a gallery area where finished prototypes are displayed. Prototypes may need to be disassembled to re-use their parts after they are exhibited. If they are not disassembled, they should remain in the gallery until the end of the program.

Tool and Material Storage

Set up a "Materials Store" Using Trays
All materials should be stored in small trays. Create clearly labeled storage spaces for each type of material. Label each tray using masking tape and a marker.

It is recommended that you only pull out trays of materials that relate to the specific project that is happening. Materials for other parts of the program should be kept out of reach, unless specifically requested.

Group Similar Materials Together in Labeled Locations
Labeling a clear home location for every tool and material type will help kids clean up.

Examples of Storage Categories:

- Electronic Tools
- Electronics Parts
- Arduino
- Cardboard and paper scraps
- Glues and Tapes
- Fabric and Notions
- Cutting Tools
- Clamps

Personal Item Storage

Clothing and Bag Storage for Students
Students will not be able to keep their belongings at their specific desk as they would in a normal school class. Their bags and clothes will get in the way and create stress or danger if piled on tables and the floor. Designate a clear area for this stuff to go, preferably near the entry door, so it doesn't even really enter the room with kids and create distraction.

Work Surfaces

Select Appropriate Tables
Identify large tables that are good for working collaboratively on projects. Students will mostly work while standing, so tall tables are best.

If those are lacking, combine smaller desks together in groups of 2 and 4. Secure the desks together with tape or zip-ties, so they don't drift apart while kids are working on them.

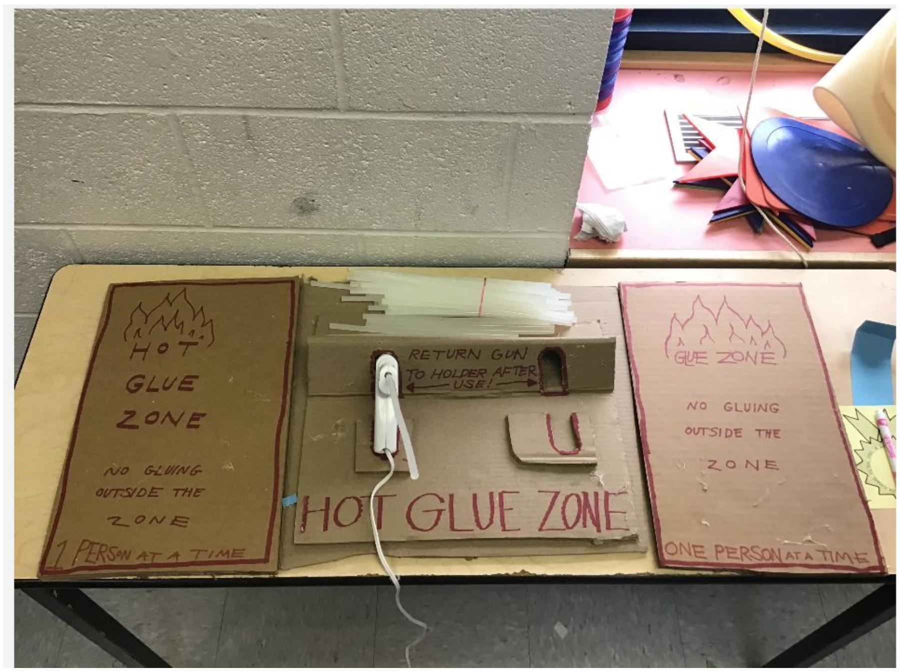
Protect Table Surfaces
Place brown butcher paper or cardboard sheets on top of tables to protect the tables from glue and tool scratches. There will be two rolls of brown paper at each site.

Leave Space Between Tables
Students will be moving around the room more than in a typical classroom. Move chairs to the side of the room to create extra space between tables.

In-Progress Prototype Storage

In-progress prototypes should be kept on the work surfaces where they are being built at the end of the day. Small parts or pieces may be kept in a labeled bag or box alongside the project.

Hot Glue Zone



Use cardboard sheets to create a clearly labeled hot glue zone. As pictured above, it is also possible to fabricate a simple hot glue gun holder by bending a sheet of cardboard into a triangular shape and cutting holes for the glue gun using a cardboard cutter.

Soldering Station

Set up a soldering station where students can sit down and solder at a safe distance away from where other students are building their prototypes. It should include basic soldering supplies such as hand tools, safety glasses, solder, electrical tape, steel wool and helping hands.

Cardboard Cutting Zone and Scrap Bin

Also create a clearly designated cardboard cutting station where students can clamp their work down. You can cut pieces of cardboard to the exact size of the table surface and tape them down securely in the corners.

If you have an empty bin or bag, use it to collect usable paper and cardboard scraps.

Code Upload Zone

Set up three Windows or Mac OS laptops with power supplies plugged in and with WiFi connected in a designated place in the classroom. Ensure that each laptop has a USB port and the appropriate USB cable for connecting an Arduino. Ensure that each laptop is equipped with the Arduino IDE software and can access the Arduino Web Editor. This is your Code Upload Zone.

Recycling and Garbage

Identify and label a container for paper recycling and garbage. Clean cardboard and paper scraps should be collected as recycling. Scraps with lots of glue must be discarded as garbage. It is wise to salvage any reusable electronic components or metal hardware from discarded prototypes.

Program Lesson Plans

Orientation - Lab Culture

Objectives
Build community, establish a safe space
Write community covenant
Establish idea of prototyping
Introduce toolkit and safety guidelines
Establish storage and cleanup routines

Project details
Students will introduce themselves, play a drawing game, make a community contract together, and share stories about their experience of making or fixing things.

Project Flow

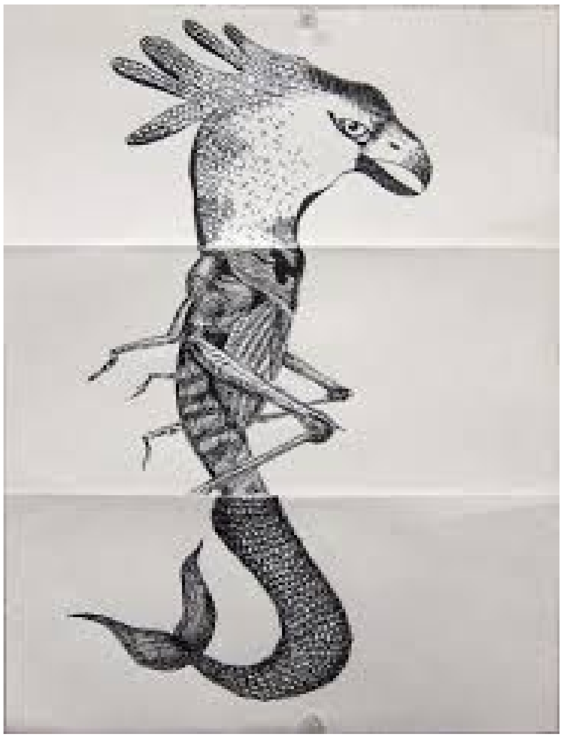
Session 1: Orientation to Invention x Nature

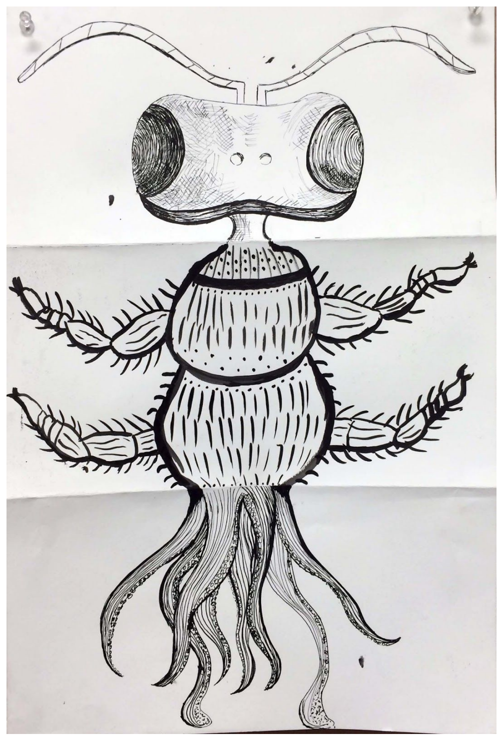
Classroom Setup
Chairs, open space to gather in a large circle
Markers and small paper for exquisite corpse game
Markers and one large paper for community contract

Welcome and thank you for choosing to be here!

Start with names or a name game.

Opening Ritual: Exquisite Corpse





Explain that you are going to play a collaborative drawing game.

Ask students to fold a piece of paper in three equal parts. **Demonstrate**.
Ask students to draw a head on the top part of the paper. Fold it so that you can no longer see the drawing, and pass to the left.
Ask students to draw a body/torso in the next part of the paper. Fold it so that you can no longer see the drawing and pass to the left.
Ask students to draw legs/feet in the next part of the paper. Fold it and pass to the left.

Ask the students to unfold the piece of paper, observe the character, name it or think about who it is. Share the results.

Save these drawings so that images of them can be uploaded to Mouse Create

Reflect : While doing the exercise, what values did you observe in the space? Reflect on the importance of collaboration and shared creativity.

Ask: What is a community?

- An *ecosystem* in which everyone supports each others growth.
- A structured environment in which every voice is heard.

Make a community contract together. What are our values? How do we transmit those values in an explicit way?

Write it down on a big piece of paper.

Example Community Contract

- One mic: show respect by listening while others are speaking.
- Everyone comes from somewhere. Appreciate diversity and respect difference.
- Honesty is the best policy.
- Support and share ideas.
- Make safe choices.
- Give and receive help.
- If you don't know, say so.
- Take chances, make mistakes.
- Leave it better than you found it.
- Be here now.

If you reinforce these expectations consistently, they will help to support and strengthen your community.

Share stories of making and fixing things.

- Have you ever followed a youtube video instructional to learn how to do something for yourself (dance moves, ikea, or hairstyles for example)?
- What was the first time you learned how to do something on your own?
- What tools “count?” Everyone has used a tool. What are some examples? What’s your favorite tool?

Explain: What will we do this summer?

- We will collaborate to design and build things.
- We will learn to use a variety of real tools and materials.
- We will make interactive prototypes that help us share our ideas with others.

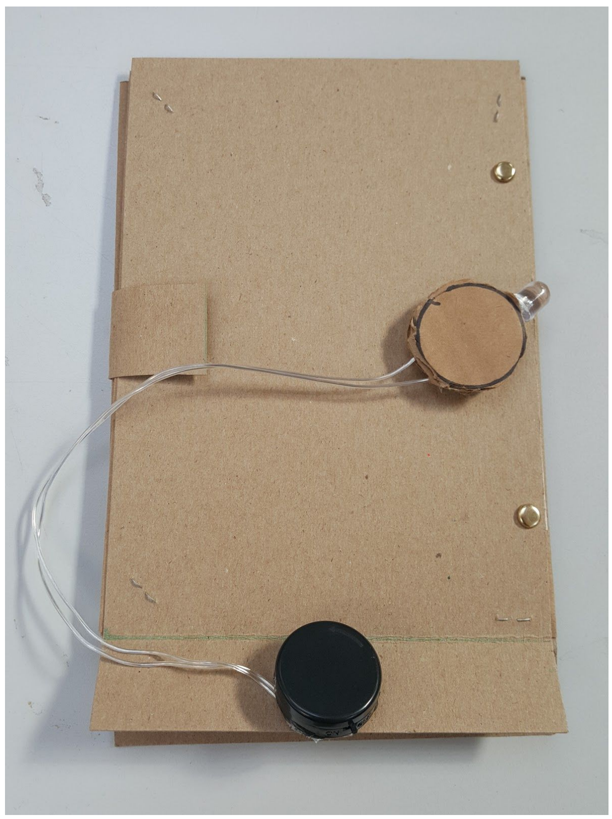
The focus of this program is to create a **laboratory environment** in which you will brainstorm, collaborate and build your own **prototypes** using real-world skills, tools, and materials. A **prototype** is a first attempt at building something or realizing an idea. **Prototypes are like first-drafts**, but they should be functional, durable, and interesting to look at. They are a way to share ideas with other people!

The structure of the program is as follows.

1. **The theme of our program is Future Humans.**
2. **Fundamentals** include the skills and tools that will help you throughout the entire summer. Specifically, we will
    a. Make your own **Prototype Journal** using cardboard, paper, UV LEDs and basic circuitry.
    b. Connect code to circuitry while making **Bionics** prototypes that imitate human and animal movement using flex sensors to control motors.
3. **Domains** are the application of skills from Fundamentals to a specific challenge and theme and include additional skills and materials. The domains are:
    i. Chameleons: wearable human adaptations that include automated patterns in light.
    ii. Bees: interactive sound systems that are triggered by body movement.
4. **Synthesis Project** is the final project that you will build. You will combine all of the knowledge you've gained during Fundamentals and Domains to build a final prototype.

We will work collaboratively throughout the program and share our projects with each other along the way.

Sessions 2-4: Prototyping Journals



Project Overview

Outcomes
Basic safety and fabrication tools training (staplers, hole punch, scissors, hot glue)
Tracing and cutting precise shapes, taking measurements and using a straightedge
Introduction to a series electronic circuit and basic circuitry tools (wire strippers, crimps)
Drawing models, schematics, and illustrations

Project details

Each student will make one journal, which is theirs to keep. Any personal thoughts or doodles can be drawn using invisible ink and kept private. They should customize the cover with their names so that journals are differentiated. Students will display the journals at the end of the program during the showcase.

Prototype Journals include the following elements:

- **Cover** made from chipboard
- **Accordion pages** made from folded strips of paper
- **Tiny notebook** bound with brass tacks
- **Velcro binding** to hold it together
- **UV Lamp** for secret messages

Tools and Materials:

- Journal Fabrication
    - Chipboard
    - Larger white paper
    - Permanent markers and paint markers
    - Brass tacks
    - 4.25 x 7” paper
    - Binder clips
    - Scissors
    - Elmers glue sticks
    - Stapler
    - Cardboard cutters
    - Rulers
    - Hole punch
    - Masking tape
    - Hot glue station
    - Velcro dots
    - Laser-cut cover template
    - Laser-cut template for tiny notebook
    - Alphabet stencils
    - Spring clamps
- Secret Message LED

- 10mm UV LEDs
- 3v batteries (2 per project)
- 2x3v battery holder
- Soldering Iron
- Solder
- Helping Hands
- Needle nose pliers
- Safety glasses
- Wire snips
- Electrical tape
- Wire strippers

Project Flow

Session 2: Intro to Prototyping Journals

Classroom setup
Example Prototype Journal, UV marker, shop safety slideshow.

Chipboard, large paper, permanent markers, scissors, staplers, tape, Elmers glue sticks, rulers, cover templates. Mini paper, hole punch, brass tacks, acetate, template for tiny cover, binder clips.

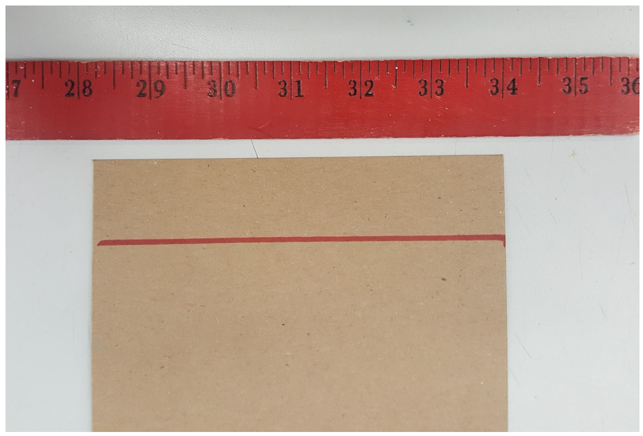
Go over complete instructions and demonstrations for making the journal first, then let students make journals at their own pace using your prototype journal as a reference.

1. **Introduce** the project.
    a. The focus of this program is to create a laboratory environment in which you will brainstorm, collaborate and build your own prototypes using real-world skills, tools, and materials. A prototype is a first attempt at building something or realizing an idea. Prototypes are like first-drafts, but they should be functional, durable, and interesting to look at. They are a way to share ideas with other people!
2. **Journals**

a. **Show** example journal.
b. **Explain**: Why are we making this? What will we use our journals for?
    i. We'll be making prototyping journals that you can use throughout the program and keep when you leave at the end of the summer.
    ii. We're building an ***accordion*** style notebook that will make it easy to display our work at the end of the program. There's space in the journals for notes in the mini notebook, and for schematics and drawings in the larger accordion notebook.
    iii. Even if you don't consider yourself a good artist, it can be important to make drawings in order to communicate your project ideas to other people or to instructors.
    iv. There's also a way for you to keep personal thoughts and doodles that only you can see! Our journals will incorporate UV lights so that you can include doodles and thoughts privately using invisible ink.
    v. The journal-making activity is your first opportunity to learn some simple fabrication skills, basic circuitry and safety guidelines while making something useful.

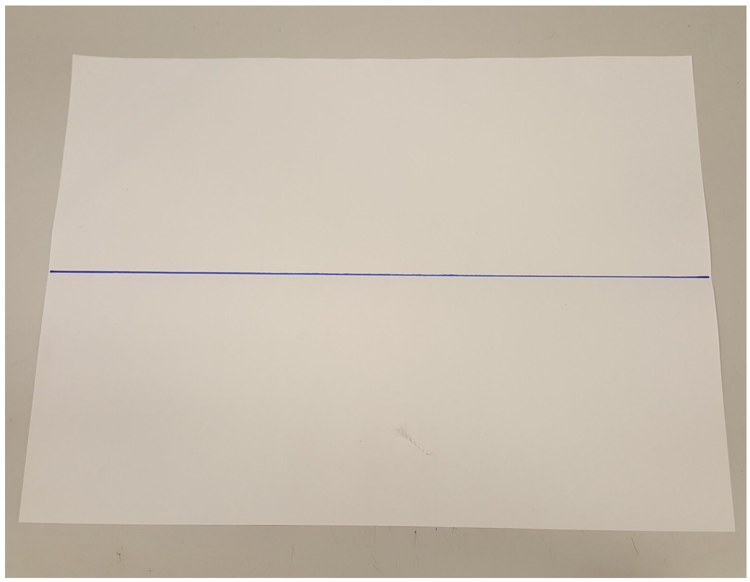
3. **Review** shop safety guidelines if necessary.
    a. This is a space where we make things together. It's important to always observe safety guidelines and use tools properly.
4. **Demonstrate** making the **accordion book**.
    a. Demonstrate how to hold and **use scissors to cut chipboard** after tracing the template of cover. Each student should cut *two* chipboard cover pieces using the template. They should both be 7" x 10.5." *Try to align the template with one of the corners of your cardboard to reduce waste and necessary cutting!*

b. Using a ruler, mark a line straight across the bottom of each cover piece, about 1” up from the bottom.

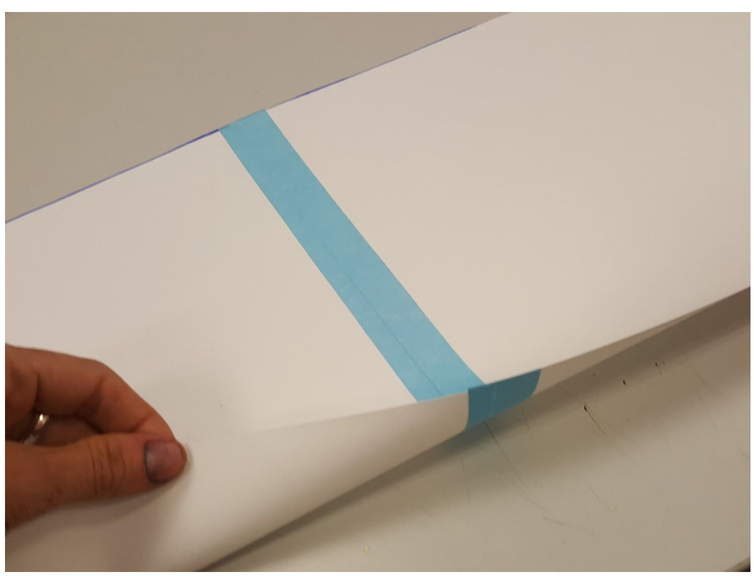


c. Fold, using the ruler to help you fold a straight line.

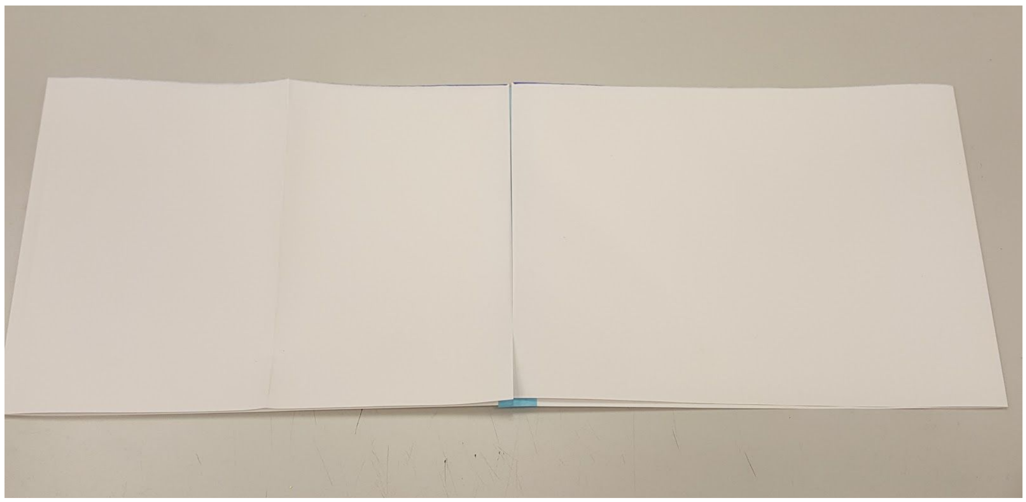
d. Take a piece of large paper and fold it in half ***lengthwise***. Unfold and mark the fold line with a marker and ruler.



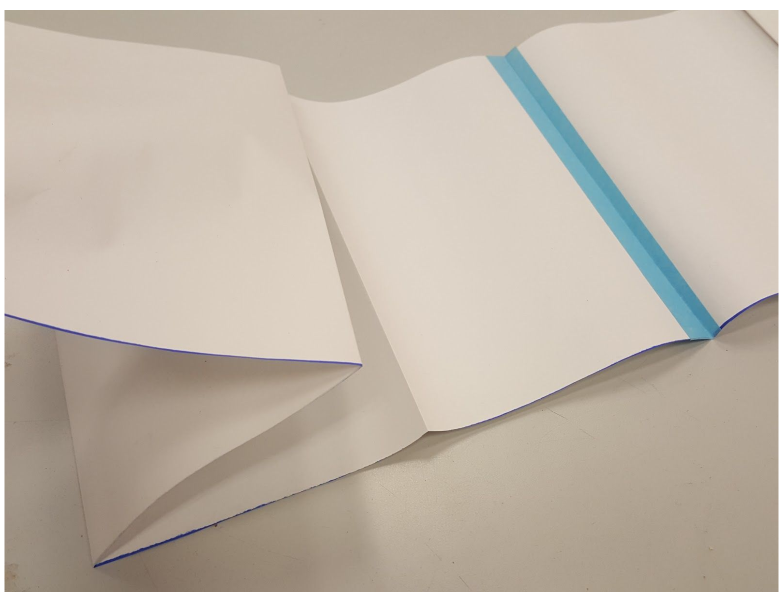
e. Cut along the line so you have two long strips of paper. Tape the two pieces of paper together, using masking tape on both sides. *It might be helpful to complete this step with a partner!*



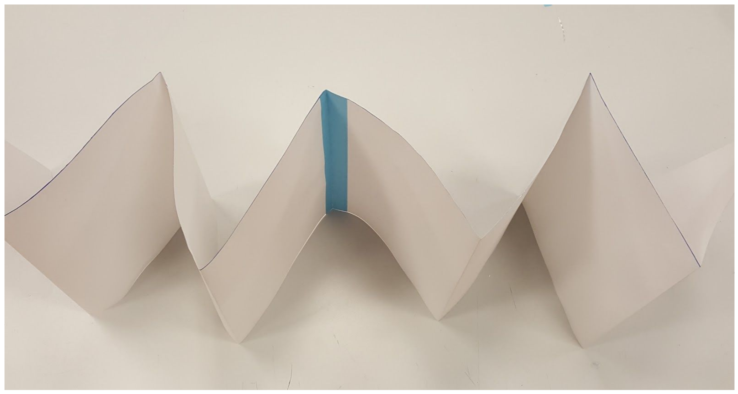
f. Fold the two paper "wings" in so they meet **in** the middle. Pretend the centerline is a mirror.



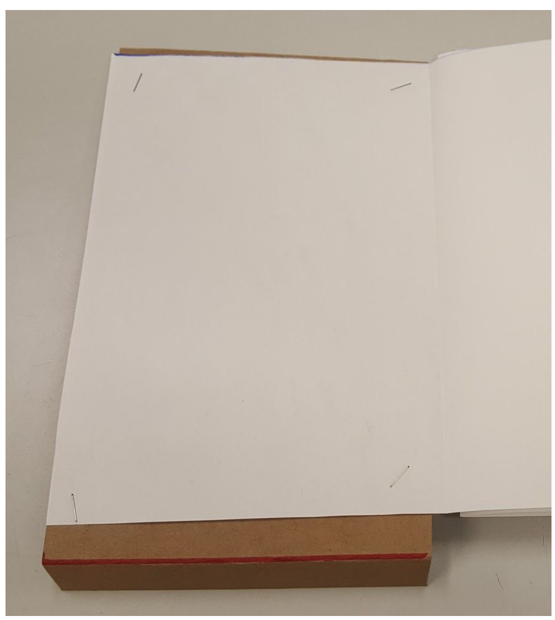
g. Fold the paper "wings" back **out**.



h. Flip it over, and fold **in**. Then, fold **out** the middle section to complete the accordion.

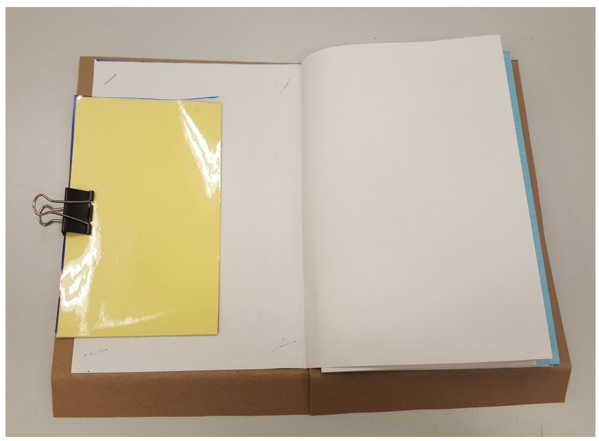


i. Align in this order: **cover**, **paper accordion**, **cover**. Glue the first page of each end of the accordion onto the covers using Elmers glue sticks, making sure the pages are glued *above* the folded line you made.
j. Reinforce the glue by stapling each corner. Make sure <u>not</u> to staple more than one page of the accordion to the cover! Try to position your staples so the sharp points are on the inside of your journal.

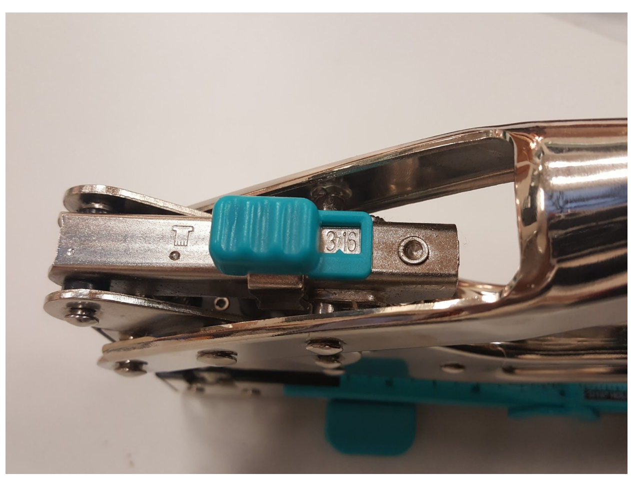


5. **Demonstrate** the following journal-making steps for making and binding the **tiny notebook.**

a. Collect about 15 sheets of tiny paper. Trace and cut a cover for the tiny notebook out of acetate. Again, try to reduce waste by cutting close to the edge!
b. ***Align*** all the pages and the cover by tapping them gently on the table top.
c. Hold the mini notebook in place on the inside of the front cover using a binder clip. *It might be helpful to complete this step with a partner to help hold the paper while you clip it!*



d. Set the hole punch to the 3/16” hole setting using the turquoise adjuster on the top. You can use the measuring device on the base of the hole punch (set it to about ¾”) to ensure that your holes are aligned along the edge of the notebook. It may also be helpful to mark where the holes should go beforehand.

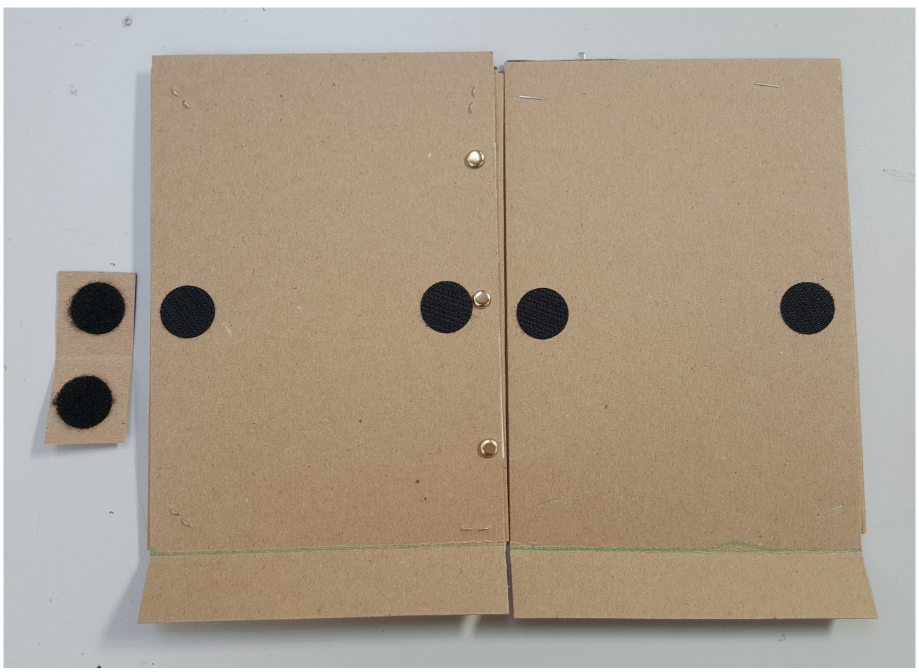


e. Use the hole punch to make two holes down the side oof the tiny notebook. *Do not remove the mini clips after cutting holes!* They'll help keep everything aligned until the brass tacks are in.
f. Insert brass tacks into the holes to bind the tiny notebook to the rest of the journal, with the legs of the brass tacks on the inside of the journal.

g. Remove and return the binder clip.

6. **Demonstrate** the following steps to add **velcro binding** to your journal.
    a. With the person next to you, **determine** who is going to be a velcro ***hook*** and who is going to be a ***loop***. There should be an even number of hook and loop people at each table. If you're a hook, all of the velcro pieces on your journal will be hooks, and you'll put loops on everything that *connects* to your journal (the binding, the LED).
    b. Add a piece of *your* kind of velcro in the middle of each side of the journal - if you're a loop person, you should use loops for this! Each student will need four pieces of velcro to do this.

c. Then, cut one roughly 1” x 3” tab of chipboard out of your scrap, and put a piece of the *opposite* type of velcro on each end.
d. **Demonstrate** hot glue gun use and safety precautions.
    i. Hot glue gun tips and fresh hot glue can burn you. Never touch the tip or any un-cooled hot glue. Hot glue turns white when it’s dry.
    ii. Only use hot glue guns in designated areas. Never play with hot glue.
    iii. Glue slowly. Rushing can lead to injuries.
e. Reinforce all velcro circles using a ring of hot glue around the outside.



f. This velcro tab can be switched back and forth from one side to the other in order to create two back-to-back books! It can be removed altogether to unfold the accordion for display, and stored by sticking it to the front of the journal.

7. While waiting to use the necessary tools, students can add names to covers and personalize journals using UV or permanent markers and stencils.
8. **Clean up** and save any useful cardboard scraps for later.

Session 3: Secret Message LED

Classroom setup

Alligator clips, UV LEDs, 3v battery clips, coin cell batteries, wire strippes, soldering irons, Helping hands, solder, cardboard, cardboard cutters, spring clamps, electrical tape, scissors, velcro, markers, electrical tape.

1. **Introduce** electrical circuits by showing students the physical objects.
    a. **Explain:** How does a battery work?
        i. *Electricity* is a form of energy we get from the movement of electrons.
        ii. A battery contains two materials; one has too many electrons and the other has too few.
        iii. This makes one side *positively charged* (+) and the other *negatively charged* (-).
        iv. It's a force of nature for the electrons to seek equilibrium, so if we connect a path from + to - the electrons will flow through it and make electricity.
        v. For our journals, we will be using two 3 volt batteries.
    b. **How does electrical wire work?**
        i. Inside is *conductive* metal, outside is *insulator* rubber.
        ii. Conductors let electricity flow, insulators resist it.
        iii. The color of the wire doesn't matter; it's just to help you stay organized.
        iv. Sometimes wire is *stranded* and sometimes it is *solid-core*. The wire inside of our journal battery clips is *stranded*, which means that it's made up of a bunch of tiny strands that can separate from each other.
    c. **What is a circuit?**
        i. The pathway from plus to minus is called a *circuit*. Electrons flow through the pathway in a *circle*.
        ii. The ***components*** in a circuit are there to perform a specific action (light up, spin, etc.), and absorb a specific amount of voltage in order to do this.
        iii. A ***short circuit*** happens when the pathway of the circuit bypasses the components it is meant to go through. Because it skips the components, the circuit not only won't work but could get very hot. This usually

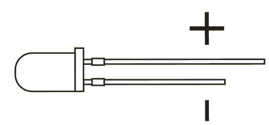
happens when two conductive surfaces touch each other that aren’t supposed to.

    iv. Short circuits can be avoided by ***insulating*** exposed wires and crimps with electrical tape or hot glue, and by trimming off stray wires that could create unwanted pathways for electricity to flow.

  d. **How does a LED work?**

    i. The LED uses a small amount of electricity and emits it as light.

    ii. These UV LEDs are extra large and can take six volts of electricity, the combined voltage from our two 3 volt coin cell batteries.



    iii. Most LEDs will take less voltage than this. If you give an LED too much voltage, it burns out. When an LED burns out, the entire circuit will stop working.

    iv. LEDs are *polarized*, and only work on one direction. The longer leg is *plus* and the shorter leg is *minus*, or ground. So in a circuit, the longer leg should always face the + side of the battery.

2. **Demonstrate** *prototyping* the UV LED circuit. Each student will need an LED, a 3v battery clip, and two 3 volt batteries. Tables will also need wire strippers and permanent markers.

    i. ***Wire strippers*** are for stripping plastic coating away from wires before soldering.

        1. Wrap the wire you’re stripping around your finger so that you can get a good grip.
        2. Strip the ends of the wires on the battery clip to about ½” - or, the width of a finger.
        3. The size of the hole needed to strip wires will vary with different types of wire - the best approach is to try different hole sizes until you find the right one.

    ii. The battery clip has ***stranded*** wire, which means there are a bunch of tiny wires inside. It’s a good idea to twist the ends tiny wires together using your fingers.

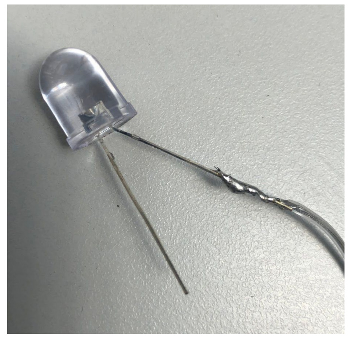
    iii. Put two 3v batteries into the clip with the plus side facing up. Screw the top back on and turn the clip *ON* using the tiny switch on the side.

    iv. Separate the legs of the LED. Touch the ends of the wire to the legs of the LED to see which is positive and which is negative. You may need to reverse which wire touches which leg. ***Mark*** *the end of the wire with permanent marker once you know which is positive.*

v. Make sure to keep the legs of the LED from touching, otherwise you'll get a ***short circuit***!

vi. *Students should all prototype the circuit so that they know which wire on the battery clip is positive and have marked it. Once they're done, they can turn OFF the battery switch.*

3. **Introduce** soldering tools.
    a. **Explain**: What is soldering? Solder is a metal that melts at about 750 degrees in order to create semi-permanent connections between conductive materials such as wire and the terminals on electrical components. It allows us to create a conductive pathway for electricity to flow. Solder fuses electronic parts together, but it is not glue. To avoid short circuits, only apply solder between the intended connections.
    b. Wire snips, wire strippers, and pliers look similar but have very different uses. Each tool should only be used for its intended purpose while soldering.
    c. ***Wire snips*** are for cutting wires and snipping excess after soldering.
    d. ***Pliers*** are for bending or securing wire as necessary. To wrap two wires together, hold them end-to-end with the pliers and twist one while holding the other with the pliers.
    e. **Helping hands** are clips that hold wires in place while soldering. Clipping as close as possible to where you're soldering will give you more stability.
4. **Explain**: Soldering safety.
    a. Always use helping hands to hold your work for you.
    b. Never touch the tip of the iron to any object or surface besides your circuit.
    c. Hold the soldering iron like a pencil. Never hold it past the grip-line.
    d. Never pass from person to person or lay iron on the table. Always return to holder when done using.
    e. Solder melts quickly. Keep your fingers at least 1” from the end as you work.
    f. Wires could become hot after soldering. Avoid touching.
5. **Demonstrate** soldering technique.
    a. Match the positive leg of the LED with the positive wire on the battery pack. **Make sure to turn off the battery pack before soldering.** Twist the positive wire from the battery pack around the long leg of the LED. Put the LED in the helping hand and heat the connection before pushing soldering into it.



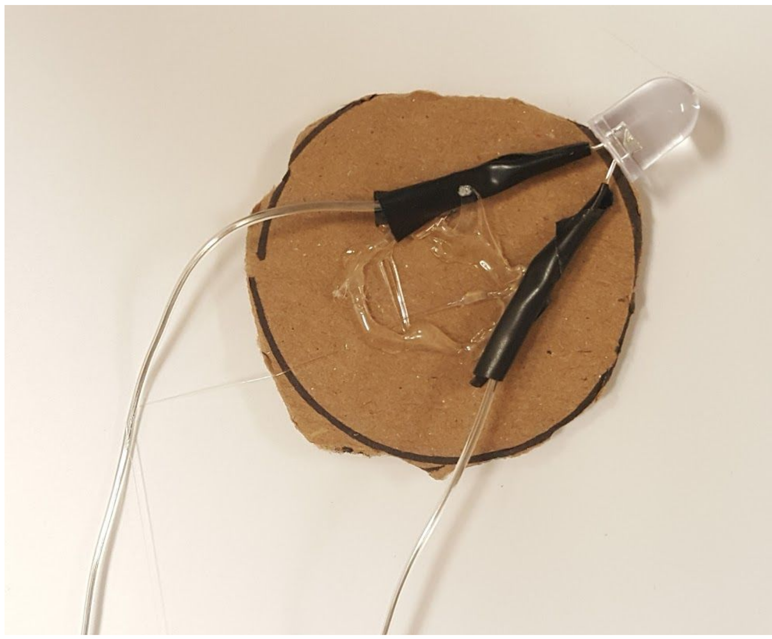
b. It’s always a good idea to ***insulate*** the legs of LEDs after soldering, so you don’t short circuit. ***Electrical tape*** is used to insulate wires and can be cut with normal scissors.
c. Solder and insulate both legs!

6. Demonstrate making a LED sandwich to mount on the journal.

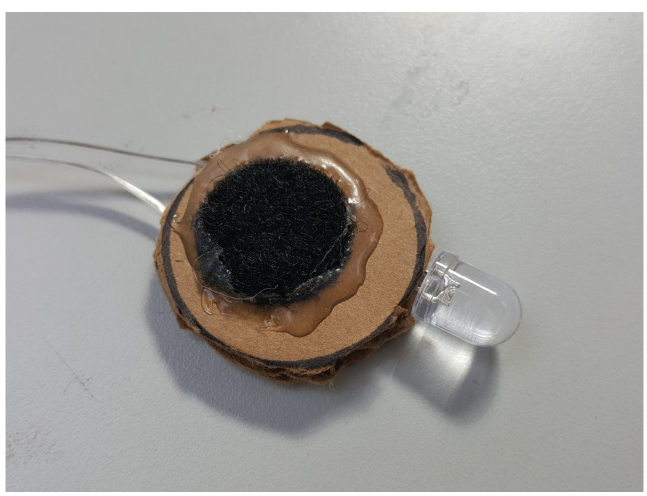
a. Demonstrate cutting two cardboard circles out of cardboard using an electrical tape roll as a template. Your two circles (or whatever other shape your choose) should be roughly big enough to cover the legs of the LED.

b. **Demonstrate** clamps and cardboard cutter safety precautions. Clamp your cardboard down to the table and use cardboard cutters and/or scissors to cut out your two circles.
   i. Hold the cardboard cutters like a butter knife.
   ii. Cut slowly. Rushing can lead to injuries.
   iii. Always cut away from you and never place your hand anywhere near the pathway of the blade.
   iv. You can always use scissors to finish your cut.

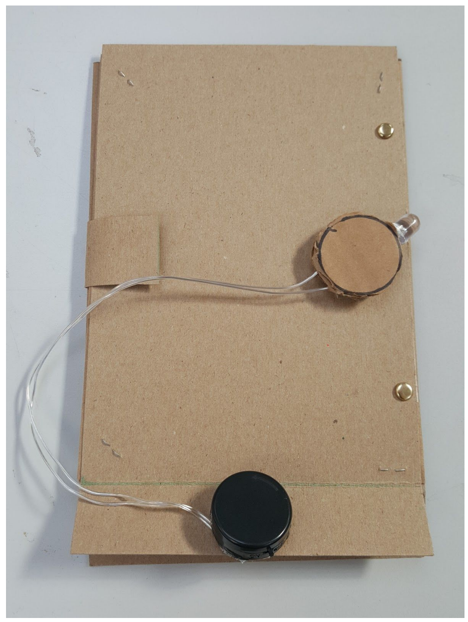
c. Sandwich the LED in the middle of the two cardboard circles using hot glue, making sure the legs of the LED are insulated or at least not touching. Hot glue the two pieces of cardboard together to make an LED sandwich.



d. Put a piece of the opposite type of velcro on one side of the LED sandwich. This way you can stick your LED sandwich to the front of your journal.
e. Reinforce the velcro by drawing a ring of hot glue around the outside of it.



f. Hot glue the ***base*** of the battery pack into place on the middle of one foot of the journal.



g. Test it to make sure the LED turns on! Then turn your battery clip off again.

9. **Clean up!** Your tables should look like they did when you began.

Session 4: Drawing and Documentation

Classroom setup

Alligator clips, UV LEDs, 3v battery clips, coin cell batteries, wire strippes, soldering irons, Helping hands, solder, cardboard, cardboard cutters, spring clamps, electrical tape, scissors, velcro, markers, electrical tape.

Classroom setup
Prototyping journals, phone cameras, pencils or markers.

1. **Introduce** drawings and schematics! Throughout the summer, we will be using our journals to make diagrams and collaboratively plan our projects.
2. **Explain**: Using your journals
    - **a.** You can use invisible ink anytime to doodle or store personal writing that you don't want other people to see. The UV light will allow you to make this text visible, when you want to.
    - **b.** Journals can be used as a place to paste printed images from the internet using glue sticks. The accordion part of the journal is what visitors at the showcase will see - include printed inspiration images or sketches here.
    - **c.** You can use your journals after a project is finished to record your own or other people's ideas that you think worked well. Sharing ideas and building off of other people's discoveries is an important part of being in a prototyping community.
    - d. You do not need to be a "good artist" to make an interesting and useful drawing or diagram!
2. **Explain**: Informational drawings
    - a. A ***drawing*** is an illustration of something - something from inside your head or something from life. A drawing can tell people someone something funny or interesting about your project. Your exquisite corpse creature is an example of a drawing.
    - b. A ***diagram*** is an informational drawing, and is really important for communicating your ideas when you're working with other people, making a detailed plan, or keeping track of important information relating to building your prototype.
    - c. Guidance and prompts for making useful diagrams.
        - i. Draw the object.

ii. How big is it? What are the ***dimensions*** or exact measurements?
iii. Show how the circuit connects. Where is the motor, LED, or sensor going to go? Where are the “brains”, or the Arduino?
iv. What are you making? Title your schematic or name your project.
v. What are the steps you need to take to make it? List them in order so you know how to begin.

3. **Introduce:** photo documentation. A portfolio of good photographs of your work is really valuable! There are a different ways to take photographs for your portfolio.
    - **a.** A ***cluttered background*** distracts people from the thing you are trying to photograph. There are too many things for their eye to sort through to see the main subject of your photo!
    - **b.** A ***neutral background*** with nothing in it makes your project stand out. It’s what most artists and photographers use to create images for websites. With a clear background anyone looking at your photo will find their eyes naturally focus on your project.
    - c. A ***portrait*** is a photo that focuses on a person and tries to communicate something about them. A portrait of you with your project shows how the project relates to your personality and sense of style. It shows your role as a creator and mastermind.
    - d. Have your friend take a portrait so they can back up and zoom in on you a little. Find a nice background with good light.
4. **Document projects!** Use phone cameras to take photographs of your exquisite corpse drawing, your journal, or yourself holding your project.
5. Mouse Create will help us build an ***online portfolio*** over the course of the summer! We will start using Mouse Create next week. For now, save the images you took to upload later.
6. If there’s extra time, free write in journals or play a game together.

Sessions 5-12: Bionics


Project Overview

Objectives
Connecting to an Arduino using jumper wires
Following circuit diagrams
Using journals to sketch and record ideas
Designing and building compound moving parts
Extending the motion of servo motors to create interesting movement

Project details
Students will work in pairs to use servo motors and Arduinos in order to make bionic limbs that move in response to their own body movements. They'll first learn how to connect an Arduino to a servo motor, and will then build simple mechanisms that extend the movement of the motor using string, rubber bands, popsicle sticks, and cardboard. They'll bring their mechanisms to life using flex sensors.

It's recommended that students work in pairs.

Tools and Materials

- Electronics and Circuitry
    - Arduinos
    - Mini screwdriver
    - Servo motors
    - Jumpers
    - Solid core wire
    - Wire strippers
    - AA batteries
    - 4AA battery clips
    - Screw terminals
    - Breadboard
    - USB cables
    - Laptops
    - Soldering supplies
    - Safety glasses
    - 10k resistors
- Fabrication
    - Cardboard
    - Zip ties
    - Rubber bands
    - Brass tacks
    - Tape
    - String
    - Skewers
    - Popsicle sticks
    - Craft circles
    - Cardboard cutters
    - Cardboard perforator
    - Cardboard tubes
    - Nuts and bolts
    - Washers
    - Rulers
    - Hole punch
    - Squeeze clamps
    - Permanent markers

- Scissors
- Staplers
- Hot glue and gun
- Brain Box template
- Journals
- Pencils
- Conductive foam
- Copper tape
- Template popsicle stick

Project Flow

Session 5: Introduction to Bionics and Arduino Circuits

Classroom setup
Mouse Create, screw terminals, battery holder, AA batteries, breadboards, Arduinos, jumper wires, servo motors, mini screwdrivers, circuit diagrams. USB cables, laptops.

Cardboard, brain box template, cardboard cutters, cardboard perforators, rulers, tape, markers, rubber bands.

1. **Explain**: What are we doing? Use Mouse Create to introduce the project.
    a. The theme of our overall program is Future Humans. Throughout the summer, we will be learning the skills needed to prototype technological or biological ***adaptations*** (gadgets, wearables, or physical extensions) needed to exist as **Future Humans**, techno-futurist extensions of our best selves navigating a complex ecosystem and protecting what's most important. An ***adaptation*** is any alteration in the structure or function of human beings that allows them to better survive in their environment.
    b. During the Fundamentals part of the program, you will learn the basic, *fundamental* skills needed to build motorized prototypes that can imitate ***organic*** movement. In robotics, ***bionics*** is the application of the way animals and plants move to the design of robots for human life. Bionics is about ***extending*** our abilities to increase our potential.
    c. During this Fundamental, you will be working in pairs and using servo motors to make mechanical structures that move organically in response to your own body

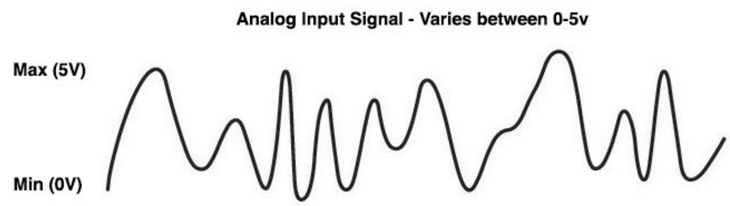
movements. We'll start by exploring the Arduino and motor alone, then we'll learn how to use simple building techniques to extend the movement of our motors to do more interesting things.

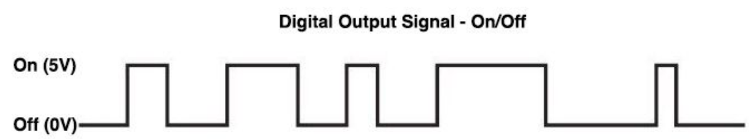
2. **Explain:** What's an ***Arduino***?
    a. Before we create our bionic mechanisms, we will create the ***central nervous system*** of our prototype using an Arduino. The ***central nervous system*** is where information is stored and activity is coordinated.
    b. An Arduino is a small computer that controls the behavior of a circuit. The Arduino controls the behavior of electrical circuits using code. Code is a set of instructions written in language a computer can understand. For example, "turn this motor on and off every second". In most cases, editable code will be pre-written to make the servo motors move back and forth. Code is edited on a computer and sent to an Arduino using a USB cable.
    c. Arduino also has pins for jumper wires to go into it. ***Jumper wires*** connect the servo motor to the circuit. The color and length of the jumper wires doesn't matter - the color is just to help you know what's what.
3. **Explore** the Arduino. Arduino also has input pins for jumper wires to go into just like the breadboard does.

    a. Arduino's take inputs and give outputs. ***Inputs*** are incoming information that the computer uses, and ***outputs*** are what the computer uses the information to do.
    b. ***Digital pins*** are typically used as ***outputs***. They are either on or off. When they're on, they send a 5V signal like the + side of the battery. When they're off,

they don't send any voltage. *Our servo motor will be controlled using digital pins.*

c. ***Analog pins*** are ***inputs***. They accept *varying* electrical signals between 0V and 5V and translate the voltage into a number that we can use in our code. 0V = 0 and 5V = 1023. *Our sensors will plug into analog pins.*

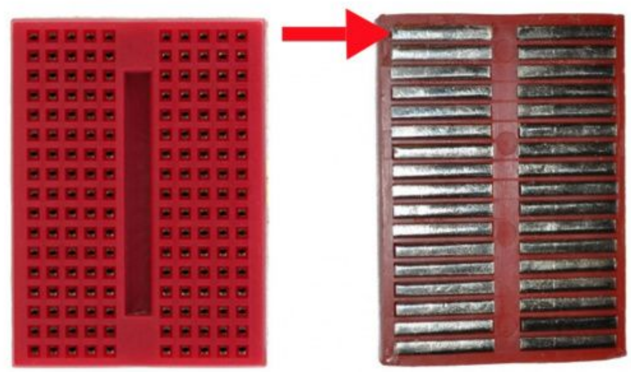




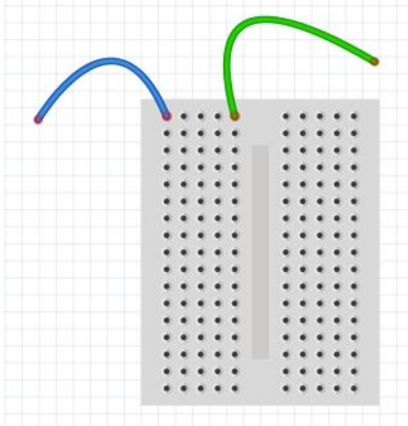
d. The ***5V pin*** outputs a steady 5V signal like the + side of the battery, the difference between this pin and a digital pin is that it is always on.

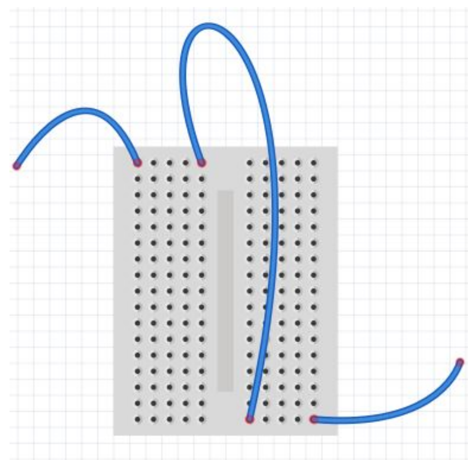
e. ***GND*** pins are ground pins and are similar to the negative (**-**) side of a battery. We will always need to end our Arduino circuits at a GND pin.

2. **Explain:** What's a breadboard and jumpers?
3. ***Breadboards*** are a tool to connect electrical components for testing out circuits. Arduino projects usually connect to circuits on breadboards.
4. Breadboards have a grid of ***pins*** that give us conductive pathways along which to connect our different components. We build circuits by connecting the pins on the breadboard using ***jumper wires*** and components like servo motors or LEDs.
5. Each row of the breadboard is a set of 5 horizontal connections, kind of like a series of power strips - the lengthwise indent along the board divides the two sides. For example, the first five pins in row one are connected and the second five pins in row one are connected to create electrical pathways.

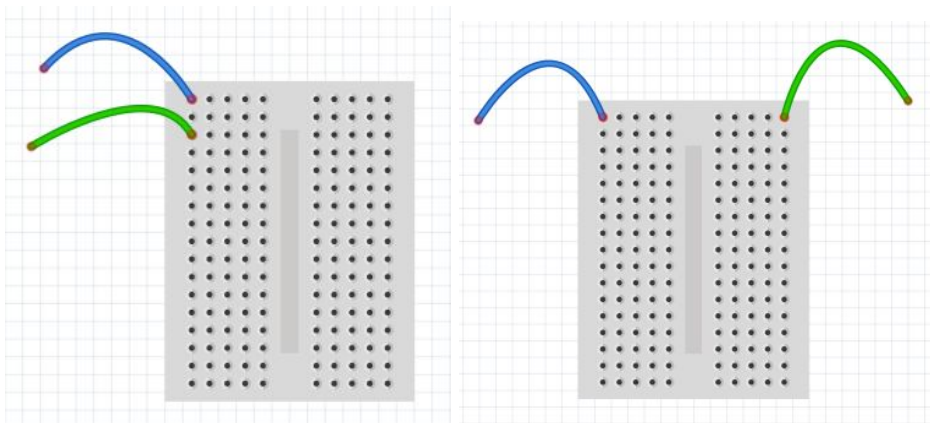


These are connected:



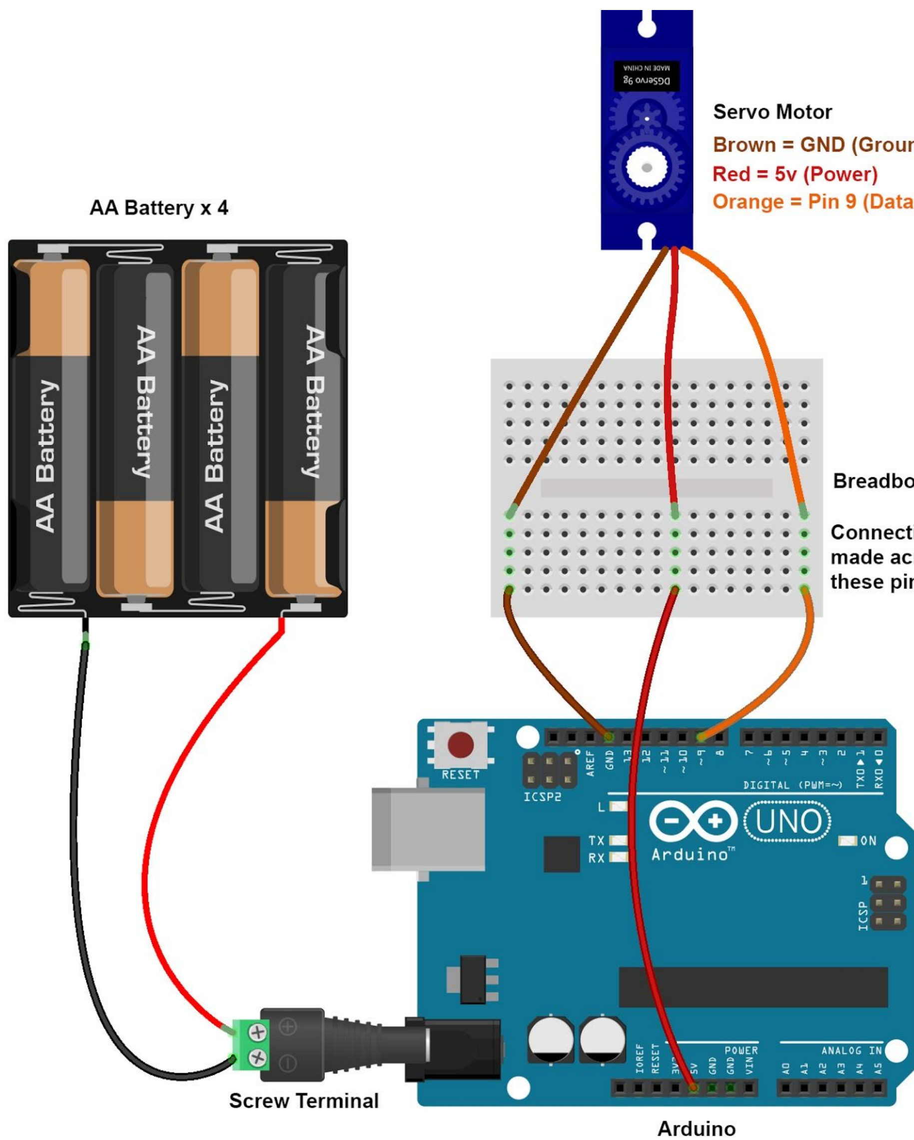


These are **not** connected:



4. **Explain: Servo motors** are small geared motors that can be programed to spin in a specific speed or direction.
    a. 180° servo motors move back and forth in a semi-circle. The *degree* that a 180° servo travels can be controlled by an Arduino.
    b. There are two parts to a servo motor:
        The ***body*** of the motor, which drives the movement of the shaft and should be glued down to a base.
        The ***shaft*** of the motor, which is the moving part.
    c. Servo motors can only move very light materials. Keep in mind that your projects should be as small as possible while still be interesting and functional.
    d. Be extremely careful: these motors can break if you over extend them or turn them too much! There are gears inside that can break. Don't try to move it further than it wants to. Never force it. Don't turn it too fast. If you hear a funny sound, stop turning it.
    e. You can play with the motor to get a sense of how it moves, but treat it like a tool and be careful with it.
    f. Servo motors always have three wires - power and ground (just like an LED), and signal. ***Signal,*** or *data,* is how the computer tells the motor what to do.
5. **Introduce:** What is a sensor?
    a. With Arduino, you can control a circuit using real-world sources like sound, light, and temperature using ***sensors***. We will be using ***flex sensors*** for this domain, which cause your servo motor to move when the sensor is flexed or bent. *We will add sensors later on in the project.*

6. **Upload code to Arduino.** Each pair or group of students should build one circuit. *For now, we'll upload a simple Sweep sketch that doesn't include a sensor. Later on in fundamentals, we'll incorporate a sensor and upload slightly different code.*
    a. Plug your arduino into a laptop using a USB cable.
    b. Open the basic **Bionics** code in the Arduino IDE.
    c. Hit the check mark in the upper left-hand corner to verify the code was copied correctly.
    d. Under the *Tools* bar, go to *Port* and select the option that has *USB* in it.
    e. Once you're connected through the port, hit the arrow button in the upper left-hand corner to send the code to the Arduino.
    f. A message should appear at the bottom of the sketch once the code has successfully been sent.
    g. Unplug the Arduino from the laptop.
    h. You will have an opportunity to look at the code later on in the fundamental.
7. **Demonstrate building the basic servo circuit step-by-step**: in pairs, follow the diagram to make connections between the Arduino and servo motor using jumper wires.



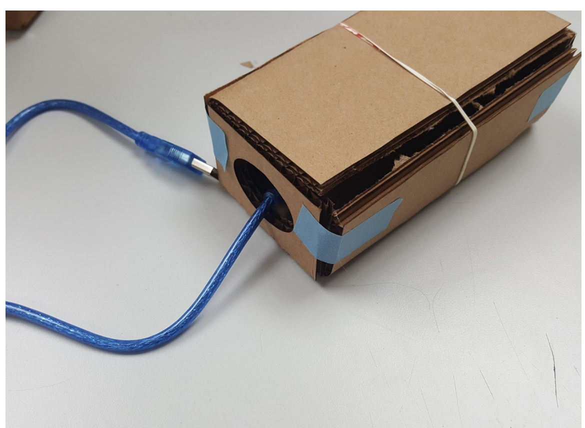
a. Connect a ***screw terminal*** to a battery pack using a mini screwdriver. Plug it into the Arduino. The lights on the Arduino should turn on when you put batteries in the battery pack.
b. Connect pin 9, GND, and 5v on the Arduino to different rows of the breadboard.
c. Connect the servo ***motor shaft*** to the motor using the mini screws and screwdriver.

    d. Connect pin 9 on the Arduino to the yellow wire of the servo, via the breadboard. The yellow wire is for signal and pin 9 is for digital motor control data.
    e. Connect 5V on the Arduino to the red wire of the servo via the breadboard. Connect GND to brown wire in the same way. The red wire is for power, the brown wire is for ground.

8. **Demonstrate**: You can extend the wires of the servo motor when necessary by stripping and using solid core wire instead of jumpers. You can use zip ties or binder clips to hold loose wires together.
9. **Give students time to build their Arduino circuits in pairs.**
10. **Demonstrate:** Building cardboard Brain Box
    a. Many of our projects this summer will involve making some kind of box to protect parts of our projects.
    b. A complete box needs six sides and be made up of 90° angles. Trace the template box provided. Cut it using cardboard cutters and/or scissors.
    c. ***Score*** along the interior lines of the box.
        i. ***Scoring*** cardboard refers to slightly cutting the internal support system of the cardboard without creating a complete cut, allowing it to bend more easily.
        ii. When you trace the template, mark the perforated bend line on the face of the cardboard with a marker. Align a ruler with the marks, and cut with perforator OR press the end of a cardboard cutter toward the surface of the cardboard. Gently pull it along the ruler edge.

d. Tape or glue the box together... It can be helpful to temporarily tape it together from the outside while you hot glue it on the inside.

e. Hold the arduino and breadboard in place inside the box using tape, velcro or a tiny bit of hot glue. Make sure the power input on the Arduino is near where you cut the hole in your cardboard box.
f. Make a top for the box by tracing it's outline onto cardboard. Use a rubber band to hold it shut.

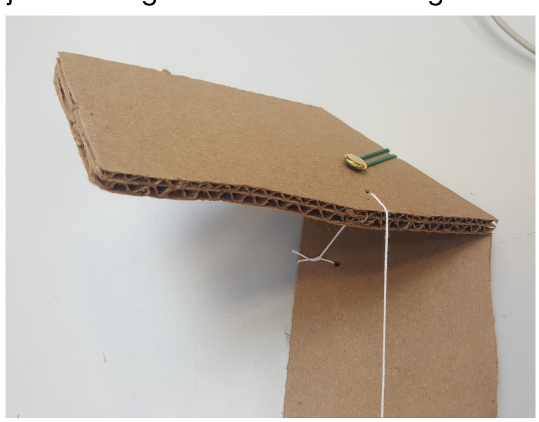


2. **Give students time to build casing for their Arduino circuits in pairs.** Make sure students put their names on the box that holds their circuit.
3. **Clean up.** Unplug the screw terminal from the Arduino to turn it off, and set aside. Students will not need their Arduino circuits for the following part of the fundamental.

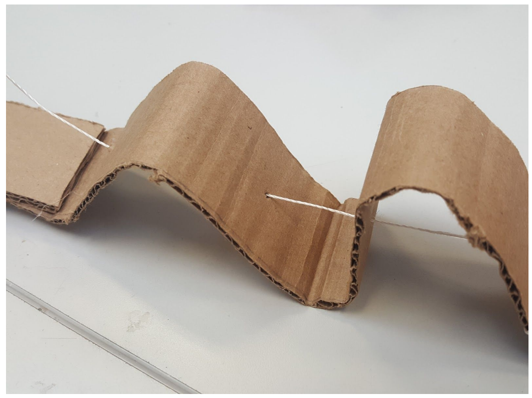
Session 6: Body-Building with Cardboard

Classroom setup
Cardboard, cardboard cutters, squeeze clamps, cardboard tubes, skewers, hole punch, string, brass tacks, scissors, staplers, rubber bands, markers, tape, popsicle sticks, washers, popsicle stick templates.

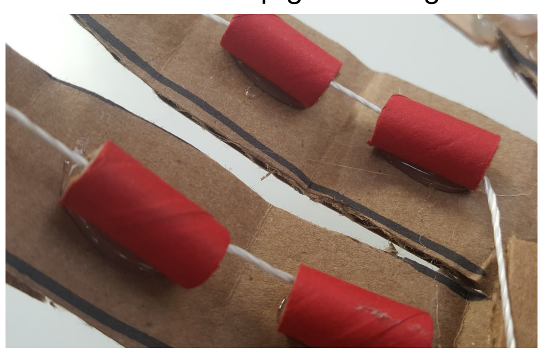
1. **Explain the main objective**: During this fundamental, we'll be using simple building techniques to ***extend*** the motion of the tiny servo motor to bring a mechanism to life. Servo motors can move back and forth in a semi-circle only (180º), but using string, levers, rubber bands, and other materials you can translate this back-and-forth movement into more complex and interesting movements.
2. **Introduce *anatomy*** of bionic mechanisms.
    a. ***Limbs*** are parts of the prototype that will move or bend - like tentacles or arms - and are made up of joints.
    b. ***Joints*** are the points on the limbs that open and close - like knees and elbows.
2. **Point out** the following contracting and expanding **joint types** and anatomical parts using GIFs or physical prototypes.
    a. ***Elbow joint****s* are made by ***scoring*** cardboard to get a sharp hinge. You can score elbow joints straight across or at an angle.



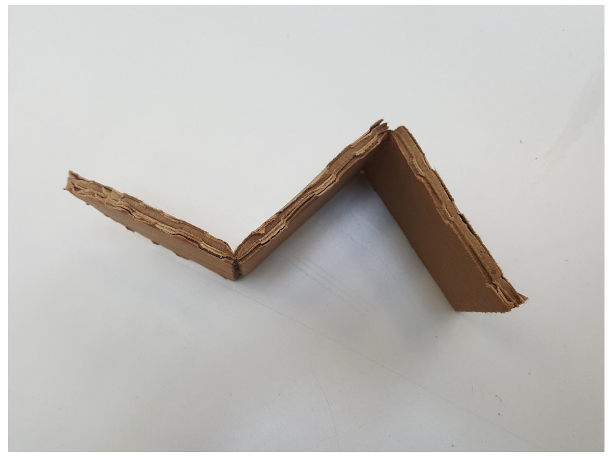
    b. ***Curved joints*** are made by bending cardboard.



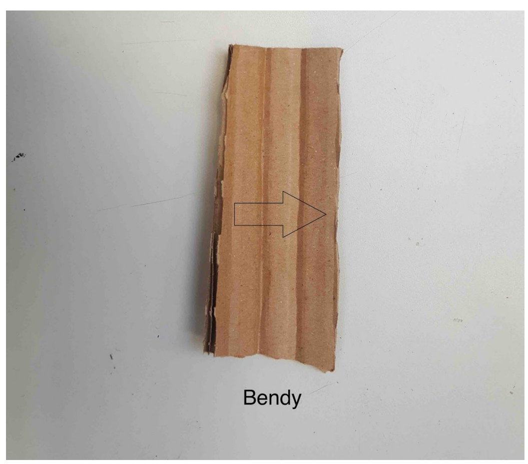
c. **Explain:** controlling elbow and curved joints.
   i. ***Muscles*** and ***tendons*** make cardboard joints move using string and rubber bands. The string (with help from a motor) pulls a joint closed, and the rubber band snaps it back open again.
   ii. Cardboard tubes can help guide string.



d. **Demonstrate** how to make a compound joint by bending or scoring your cardboard in one or both directions. Does it fold like an accordion or open and close like a finger?
   i. ***Double ply*** cardboard is great for creating rigid motion like this. You can reinforce hinges with tape if you cut too far or if it's starting to rip - just make sure to tape the opposite side from the one you scored.



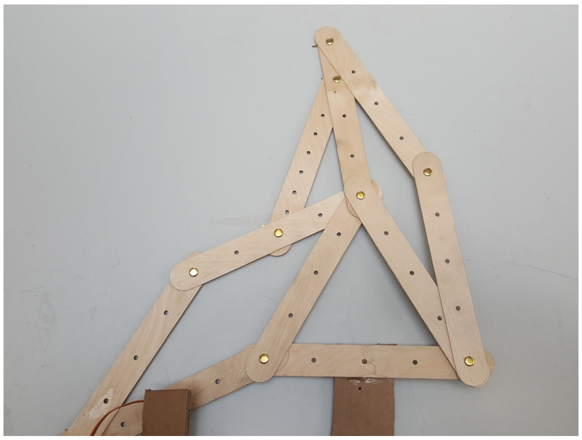
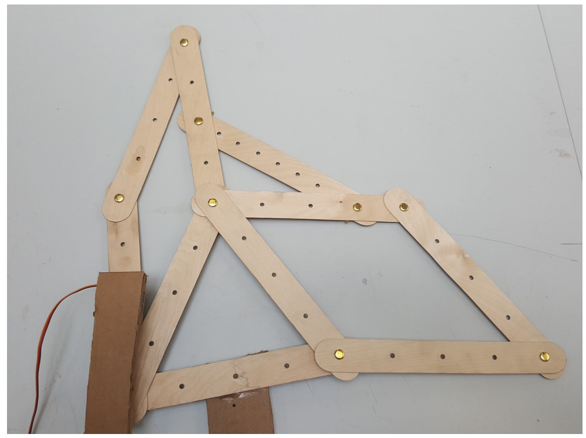
ii. Demonstrate ***bending*** for ***curved joints*** using the edge of a table. ***Single ply*** cardboard is great for non-rigid, wavy motion. Consider ***grain direction*****:** cardboard bends along with the corrugation - the lines and hollow grooves should align with your bend.



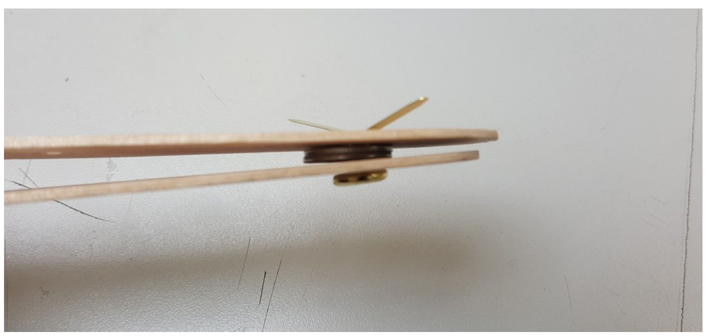
iii. Make a ***tendon*** by attaching rubber bands to the ***outside*** of the joint using brass tacks. The ***outside*** of the joint is the open scored side. You can adjust the strength of the tendon by moving the brass tacks closer or further from each other.

b. Poke holes through the center of each joint for string to go through. Close the joint, decide where the string needs to go through the two sides, and mark the points. Use a hole punch or pencil to punch through both sides of the cardboard.
c. Make the limb string-operated by threading strings through the holes you just made. You can tie string to the ends of limbs or use staples to hold it in place. Imagine a motor at the *other* end of the limb, where you're holding it, pulling and releasing the string to make the limb bend and contract.
d. ***Levers*** can extend the motion of the tiny servo motor as it pulls the string to create more interesting movement. Consider:
   i. What ***plane*** is the lever moving along? For example, does it move up and down, or from side to side? Consider how your motor would need to attach in order for the lever to move in this direction.
   ii. How long does your lever need to be? You can punch holes in popsicle sticks or tie string to skewers for longer levers, or punch holes in craft circles for small sturdy levers.

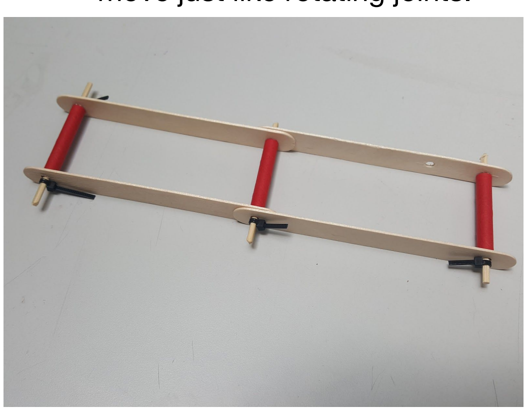
4. **Explain:** controlling rotating joints.
   a. ***Rotating joints*** are made up of a series of folding popsicle sticks that expand and contract like a skeleton. They open and close on a single ***plane***.



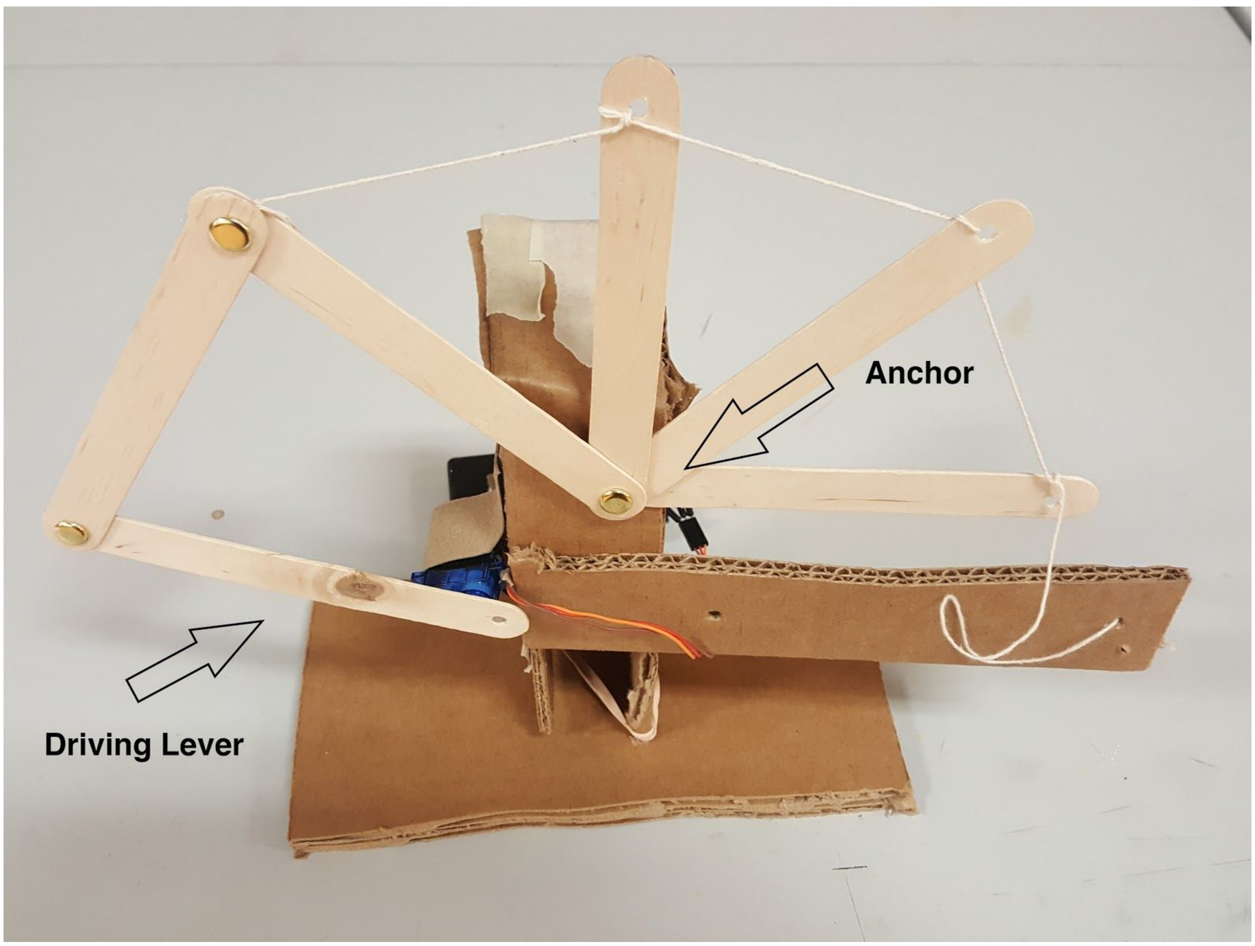
b. ***Washers*** do a couple of things - they can be used to reduce ***friction*** between two surfaces or to create space between two surfaces, as needed.



c. ***Ladder joints*** can be built using skewers, cardboard tubes, and popsicle sticks, with zip ties or rubber bands to hold the skewers in place. These joints bend and move just like rotating joints.



d. The ***driving lever*** extends from the motor shaft, similar to how levers function with elbow and curved joints. The driving lever controls
e. The ***following motion*** is made up of all the joints connected to and controlled by a driving lever.
f. The ***anchor***(s) are fixed points where rotating joints are mounted on the supporting structure. Anchors can either be stationary or move on a brass tack.



5. ***Play*!** Explore materials to find out what works and what's interesting.
   a. Start by making a simple limb that expands or contracts *without* a servo motor. Small practice prototypes will help us understand how to build moving parts.

b. It's important to play first to get a sense of what's possible, rather than starting with an idea and trying to figure out how to build it.
c. Then, we'll use what we learned to combine and expand our prototypes and *automate* them using servo motors.

6. Clean up.

Session 7: Test, Combine Joints, and Build

Classroom setup

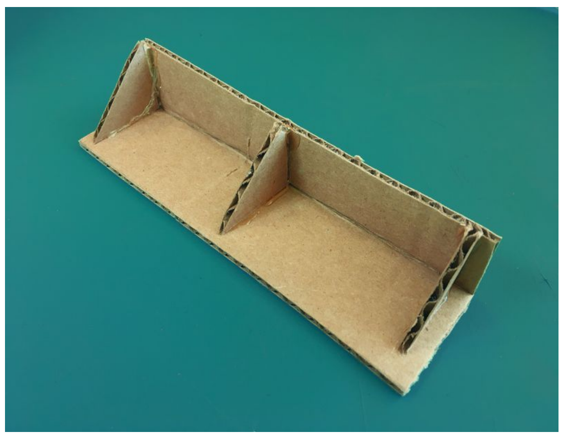
Cardboard, cardboard cutters, skewers, hole punch, zip ties, string, brass tacks, scissors, staplers, rubber bands, hole punch, markers, tape, squeeze clamps, hot glue and glue gun, popsicle sticks, washers, journals and pencils.

1. **Evolve** your prototypes
    - **a.** **Share** results with each other within groups. In your groups, combine what you learned and design a new mechanism together. Consider:
        - **i.** What kind of mechanism could you make using these techniques and materials that imitates human movement in response to the movement of a flex sensor?
        - ii. What kind of gesture is it? What does it do?
    - **b.** **Draw** a diagram of what you want to make. As you work and build on your ideas over the course of the next few days, add the following information to your schematic.
        - **i.** Where is the motor mounted?
        - ii. Where is the driving lever?
        - iii. What kind of joints are there and how many? What is the following motion?
        - iv. Where are the anchor points?
        - v. How will the flex sensor connect to your body?
2. **Build!** Test and improve.
    - a. Build a simple mechanism as a group and move it using only your hands.
3. Clean up.

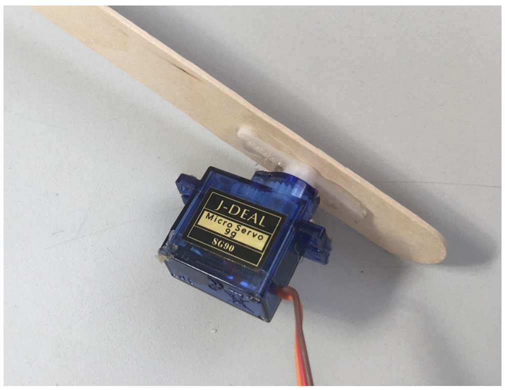
Session 8: Mount Servo Motors

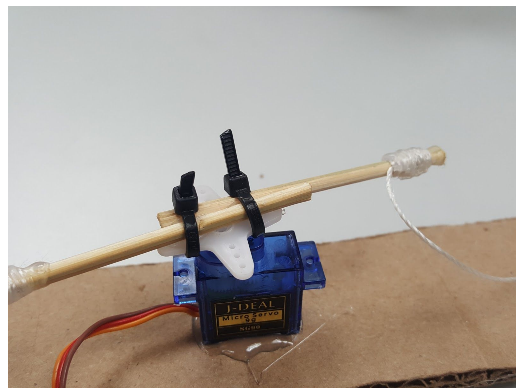
Classroom setup
The same as session 7.

1. **Explain**: The motor needs to be mounted to a supporting structure or fixed onto the same surface as the joint. **Demonstrate** some techniques for building strong structures.
    a. Triangles can strengthen structures

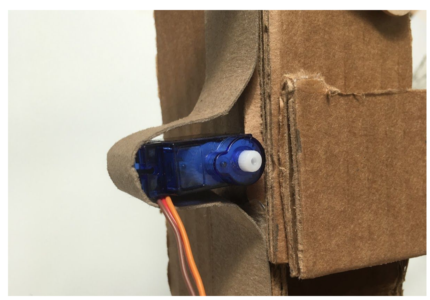


    b. **Cubes** can be built to mount the servo motors using the templates provided.
2. **Demonstrate:** how to incorporate servo motor.
    a. Move the prototype using just your hands and notice where the motion extends from. Where does the ***driving lever*** need to be? Where does the motor need to be mounted?
    b. Hot glue or zip tie levers to motor heads. Glue, staple or tie string to levers. You can control multiple joints with one motor by attaching string to multiple points of the same lever.





c. Where does the motor need to be based? Tape, or a cardboard strap can be used to secure a motor.



d. Avoid getting hot glue in or near the motor shaft otherwise it could clog the motor and prevent it from moving!

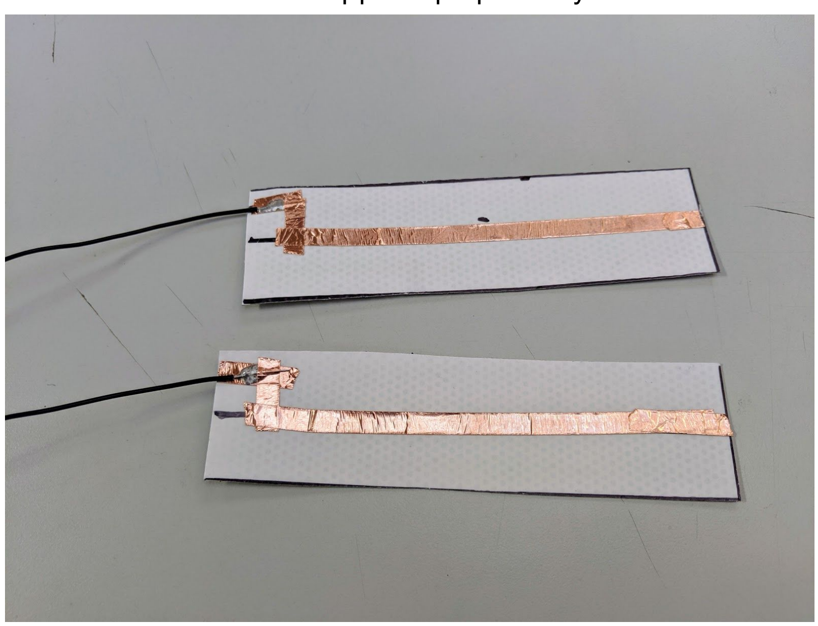
3. **Continue** building bionic mechanism prototypes.
    a. Once you've built a prototype and connected it to a servo motor, test it by plugging in the Arduino to power.
    a. Does it move the way you want it to? How can the design be improved?
    b. Ask other people about what they did and incorporate their ideas. Cooperation and sharing are important parts of natural evolution.
11. Clean up.

Session 9: Controlling Movement with Flex Sensors

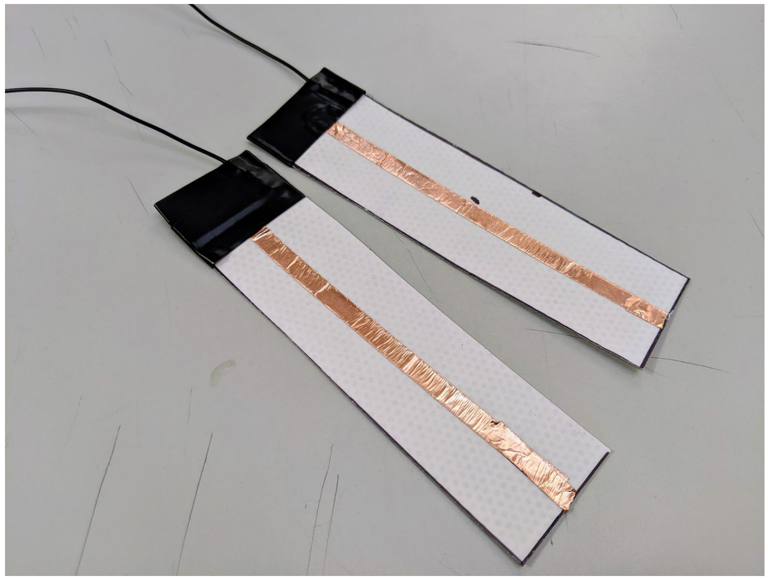
Classroom setup

Cardboard, cardboard cutters, squeeze clamps, copper tape, conductive foam, vinyl, scissors, electrical tape, 10k ohm resistors, hot glue station, soldering station, permanent markers, rulers, circuit diagrams.

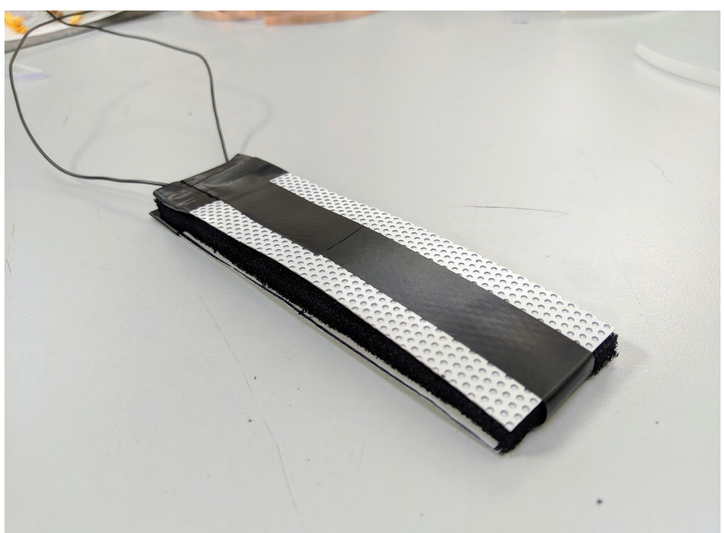
1. **Demonstrate:** Building a flex sensor.
    a. ***Flex sensors*** are sensors that change in ***resistance*** as the sensor bends. Resistance is a quantity that measures how much *electric current* can flow. The arduino will be able to read how the resistance changes as the flex sensor opens and closes, and use that to control the servo motor. You can connect flex sensors to the joints of your fingers, arms, or legs so that your body movement affects the movement of the servo motor. Gaming gloves often use flex sensors.
    b. To build a flex sensor, cut two small strips of vinyl about 1" x 6". On the smooth side of the vinyl, mark a line down the center of each strip (longways) and align copper tape down the center. Make a copper tape pathway going off to the side of each center pathway, as in the image below.
    c. Solder a wire to the end of each copper tape pathway.



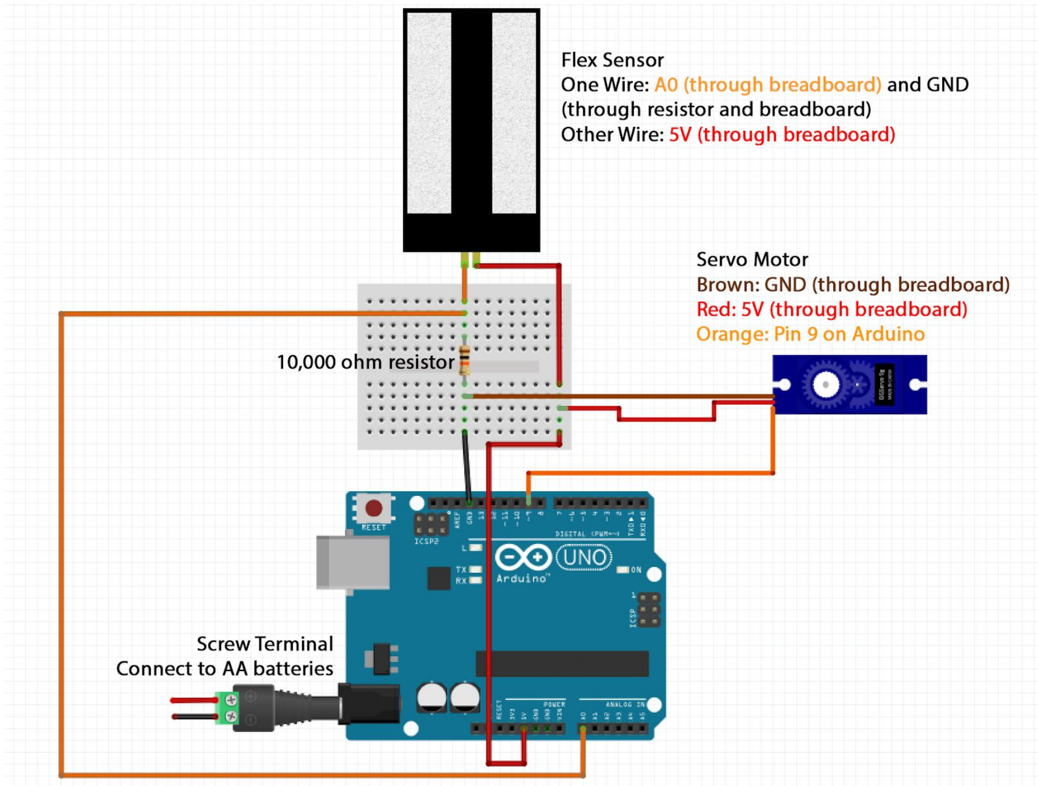
    d. Wrap the two areas where the wires are soldered to the copper with electrical tape.



e. Cut one piece of foam to be the same size as the vinyl. Sandwich the piece of foam in between the vinyl.
f. Tape the sandwich together in this order: vinyl, foam, vinyl. Make sure that the wires are coming out of the same end of the sensor, but are not both coming out from the same *corner* of the sensor.



2. Each group should build a flex sensor and connect it to their Arduino circuit using the diagram below.



3. **Demonstrate:** editing ***code***
    a. Upload ***Bionics Sensor*** code to Arduinos.
    b. Explain anatomy of code using **Bionics Code Cheat Sheet**.
4. **Allow students time to change code and see how sensor can control their prototypes.**
5. Clean up.

Session 10: Build!

Classroom setup
The same as session 9.

1. **Continue** building and finalizing prototypes.
2. Cleanup.

Session 11: Narrative Building

Classroom setup
The same as session 10.

1. **Continue** building and finalizing prototypes.
2. **What is it?** Use available materials to add sculptural elements to prototypes such as claws or fingers to give meaning to the movement you've created.
3. Clean up.

Session 12: Prototype Fair

Classroom setup
Prototypes, cameras or phone cameras, Mouse Create.

1. **Prototype fair!**
    - **a.** Create a setting in which all students can come together and share what they've made with each other.
    - a. Students should explain what their prototype is and demonstrate how it works.
    - b. Encourage them to ask questions about each other's projects.
2. **Document** projects using a camera.
3. **Demonstrate** for students how to log into Mouse Create and upload images to start building a **digital portfolio**.
    - a. Describe your prototype. How is it used by a future human? How does it work?

Bionics Code Cheat Sheet

Bionics Sketch

This code is meant to be loaded onto an Arduino, so that students can observe a servo plugged into pin 9 moving back and forth. Since it's more of a demonstration, it doesn't have to be modified.

Bionics Flex Sketch

This code will move a servo motor depending on the reading that a flex sensor gives. Students can modify this sketch if they want to. The entire code will be explained, and parts that students will especially want to experiment with are highlighted like this.

Code Comments

If you write something in the coding window, the computer will think that it is code. We can get around this by putting // before anything we want to write in regular human language. This is helpful if you want to leave notes or explanations in your code.

```
// This is a comment!
// It needs to have the two forward-facing slashes before it,
// and it will last until the next line of the program.
```

If the text turns a gray color, you will know that it worked.

Naming your Servo

This is telling the Arduino that we want to use special code to control our servo motor, and then telling it that we want to use one servo in our program. You have to name every motor that you use - we are naming ours "myservo". A second servo might be called "myservo2."

Notice the colors of the code - Arduino gives different commands different colors. Orange is common, but you will also see blue-green, green, and black!

```
#include <Servo.h>  // this tells th
Servo myservo;      // we are tellin
```

VOID SETUP()

This is an important area in the code that every program needs in order to run. We put code here that sets up things we want to use later, like motors or lights. We have to put the setup code between brackets {   }.

```
void setup() {         // Beginning of setup().

  myservo.attach(9);   // this command tells the computer that we
                       // plugged into pin 9.

  Serial.begin(9600);  // this starts up our Serial Monitor, whicl

}                      // End of setup().
```

VOID LOOP()

This is the most important area in any program. It contains all the “action” parts of the program, and they will repeat over and over until the Arduino is unplugged. Like with setup(), your code in this area needs to be between brackets {   }.

The rest of our code in this program will be put in our loop().

Serial Reading and Smoothing

A flex sensor will report a number back to the Arduino based on how much it is bent or pressed.

This number can range from 0 to 1023.

Sensors can be jittery, so we are going to take 30 readings from the sensor at once and average them out.

```
int smoothedReading = averageReadings(analogRead(A0), 30);
```

Printing in the Serial Monitor

The Serial Monitor is a tool we can use to see information being sent to or from the Arduino.

After we upload our code, we can activate it by clicking the icon in the upper-right corner.
We will use it to show our smoothed-out readings from our sensor. Use it to play with your sensor and determine the range of numbers it gives you!

```
Serial.println(smoothedReading);        // Print out our smoothed sens
                                        // to the Serial Monitor so we
```

Mapping Sensor Information to Servo Motion

Next, we want to take the range of numbers that our sensor is reading, and map them to our servo's range of motion so that our motor moves according to the sensor. We can do this using a command called map() and two ranges of numbers - in this case, our sensor range and our motor range. With the map() command, the lowest possible sensor reading is anchored to the lowest motor position, and the highest possible reading is anchored to the highest motor position.

```
int servoPosition = map(smoothedReading, 400, 1023, 0, 180);   // We are taking ou
                                                                // and naming it "s
```

sensor range: `400, 1023`

motor range: `0, 180`

You **will** have to change the sensor range to match the range of values that your sensor actually gives once you plug it in and use it. You'll do this by looking at the serial port, and noticing what the lowest and highest numbers are when you use your sensor. Then, replace the first two numbers in map() with your numbers.

Moving Your Servo with Sweep()

“sweep()” is a command that lets us control the speed and position of our servo motor. Inside the (), there are three things in order:

1) The name you gave your servo motor (in this case, it's "myservo").
2) The degree the servo will move to, using servoPosition - servos can move from 0 to 180 degrees.
3) The speed that you want it to move with. The bigger the number, the slower it will go. As an example: 0 is very fast, 10 is medium-slow, 100,000 will take years, etc.

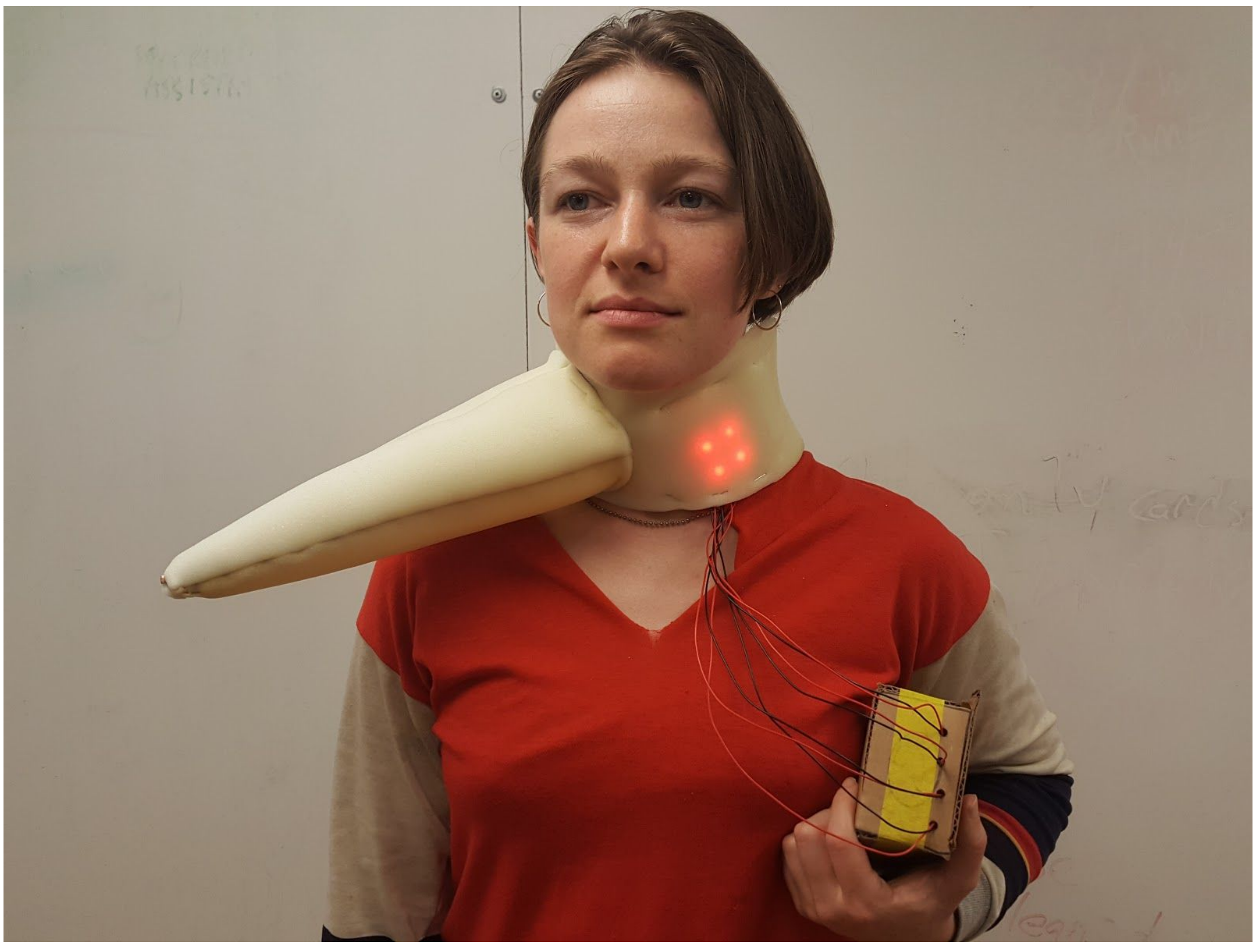
```
sweep(myservo, servoPosition, 0);
```

Students may want to adjust the speed of the servo motor.

End of Loop()

This is the end of our main program! There is some code underneath our program, but we do not need to change it. If you are curious, it is custom code that lets us sweep our motors back and forth more easily.

Sessions 13-20: Chameleons



Project Overview

Objectives
Creating a colorful, textural, responsive wearable for a future human
Creating interesting patterns in light using Arduino, LEDs, and photoresistors
Constructing soft forms with foam using yarn and staples
Using armature wire to create bendable forms

Pattern-making out of paper

Project Details

Students will design and build wearable prototypes that allow them to adapt to their future environments and communicate visually with other future humans. They will design a body adaptation or device using organic forms made with foam, and incorporate visual light patterns using Arduino and various kinds of LED lights.

It's recommended that students work in pairs.

Tools and materials

- Electronics and Circuitry
    - Arduino
    - 220 and 10k ohm resistors
    - UV and colored LEDs
    - Photocells
    - Laser cut LED holders
    - Electrical tape
    - 4AA battery clips
    - Solid core wire
    - Breadboard
    - Needle nosed pliers
    - Soldering supplies
    - Perfboard
    - Safety glasses
- Fabrication
    - Staple pliers
    - Foam
    - Armature wire
    - Colorful yarn
    - Elastic
    - Acetate
    - Hot glue and hot glue guns
    - Cardboard
    - Cardboard cutters

- Velcro
- Permanent markers
- Paint markers
- Yarn sewing needles
- Hole punch
- Scissors
- Measuring tapes

Project Flow

Session 13: Introduction

Classroom setup
Mouse Create, Arduinos, breadboards, LEDs, jumpers, photoresistors, soldering supplies, solid core wire, 10k and 220 ohm resistors, battery packs, laptops, USB cables.

Cardboard, box template, cardboard cutters, perforators, markers, tape, hot glue station.

1. **Explain**: What's the main idea? Use Mouse Create to introduce the project.
    a. Visual communication is the conveying of information about our identity or our intentions in ways that can be seen. There are different ways that humans and animals communicate visually.
    b. For example, adaptations to our physical bodies affect how we function in our environment: form relates to function. Consider the tough protective armor of a crab or the gills of a fish that allow it to breath underwater. Color change and light is also used by animals and humans to indicate important information.
    c. Chameleons are distinguished by their zygodactylous feet - like some birds, they have two toes facing forward and two facing backwards, which allows them to climb tree trunks more easily. Chameleons' eyes are independently mobile, but when aiming at prey they focus forward in coordination.
    d. Some chameleon species are able to change their skin colouration. Different chameleon species are able to vary their colouration and pattern through combinations of pink, blue, red, orange, green, black, brown, light blue, yellow, turquoise, and purple. Colour change in chameleons functions as camouflage and social signaling. Chameleons tend to show brighter colours when displaying

aggression to other chameleons, and some species adjust their colours for camouflage in accordance with the vision of the specific predator species by which they are being threatened.

e. ***Camouflage*** is an adaptation that allows animals to blend in with certain aspects of their environment. Camouflage increases an organism's chance of survival by hiding it from predators.
f. ***Aposematism*** is a biological means by which a dangerous or noxious organism advertises its dangerous nature to a potential predator using visual signalling such as bright patterns or colors that make them stand out.
g. Can you think of ways that humans communicate visually?
h. **Explain**: in this domain, we will use Arduino to create LED patterns in different colors that can be controlled using a ***photoresistor***. We will start by building a practice circuit with one LED but you can include up to six in your final prototype.

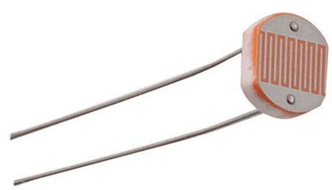
2. **Explain:** what are LEDs?
   a. The LEDs we'll be using are exactly the same as the UV LED we used for the journal, except they are smaller and they're colored.
   b. Remember to always keep the two legs separated, otherwise they short circuit.
   c. LED's are ***polarized***, which means that they are one directional. Pay attention to which side of the circuit the long leg goes on.

3. **Explain:** what's a resistor?
   a. We'll be using a ***resistor*** as part of our circuit. Resistors are often used to reduce voltage in circuits to prevent LEDs from burning out. They are also used here with ***photoresistors*** to create an anchor point for the sensor so it can compare its resistance readings to one baseline resistance reading.

220 Resistor

10k resistor

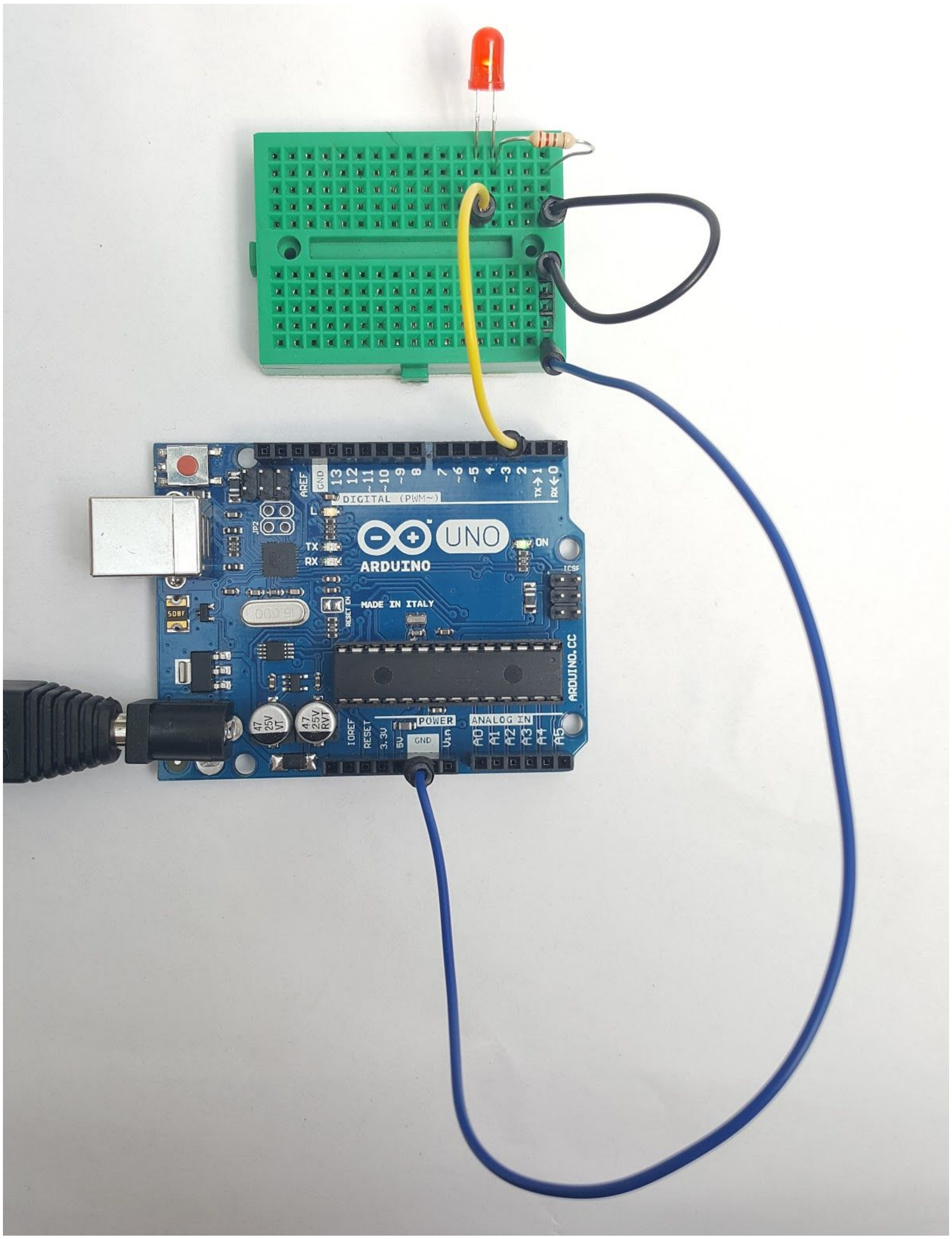
4. **Explain:** what's a photoresistor?
    a. A light sensor, or ***photoresistor,*** varies its resistance based on exposure to light in a way that is measurable by the Arduino. The Arduino can use values from the photoresistor to control other parts of the circuit. The code that we're working with will turn your LEDs on when they are in the dark.



5. **Upload code to Arduino.** Each pair or group of students should build one circuit.
    a. Plug your Arduino into a laptop using a USB cable.
    b. Open the **Chameleons** code in the Arduino IDE and upload it to your Arduino.
    c. *You will have an opportunity to edit the code later on in the domain.*
6. **Demonstrate** building an LED circuit step-by-step using a breadboard. In pairs, follow the diagram to make connections between the Arduino, breadboard and LEDs using solid core wire or jumpers. Power the Arduino first using the battery pack, then connect to the breadboard and the components.

    a. Connect a ***screw terminal*** to a battery pack using a mini screwdriver. Plug it into the Arduino. The lights should turn on.
    b. Students can include up to 6 LEDs. Before incorporating LEDs into projects, make sure to test them by holding them to a coin cell battery to make sure they work!

c. Connecting LEDs to breadboard:
   i. Use a wire to connect from GND to one corner of the breadboard. Then use another small wire to bridge the gap to the other side. These two breadboard rows will be your "ground bar."
   ii. Connect a jumper wire from pin ~3 on the Arduino to the breadboard. In general, you can use any pin with a ~ symbol next to it to fade LEDs.
   iii. Put the positive leg of the LED into the same row as the ~3 jumper wire, with the negative leg facing to the right.
   iv. Add a 220 ohm resistor with one end in the same row as the negative leg of LED and the other connecting to the ground bar (All your LEDs will share the same ground bar). It should look like this:

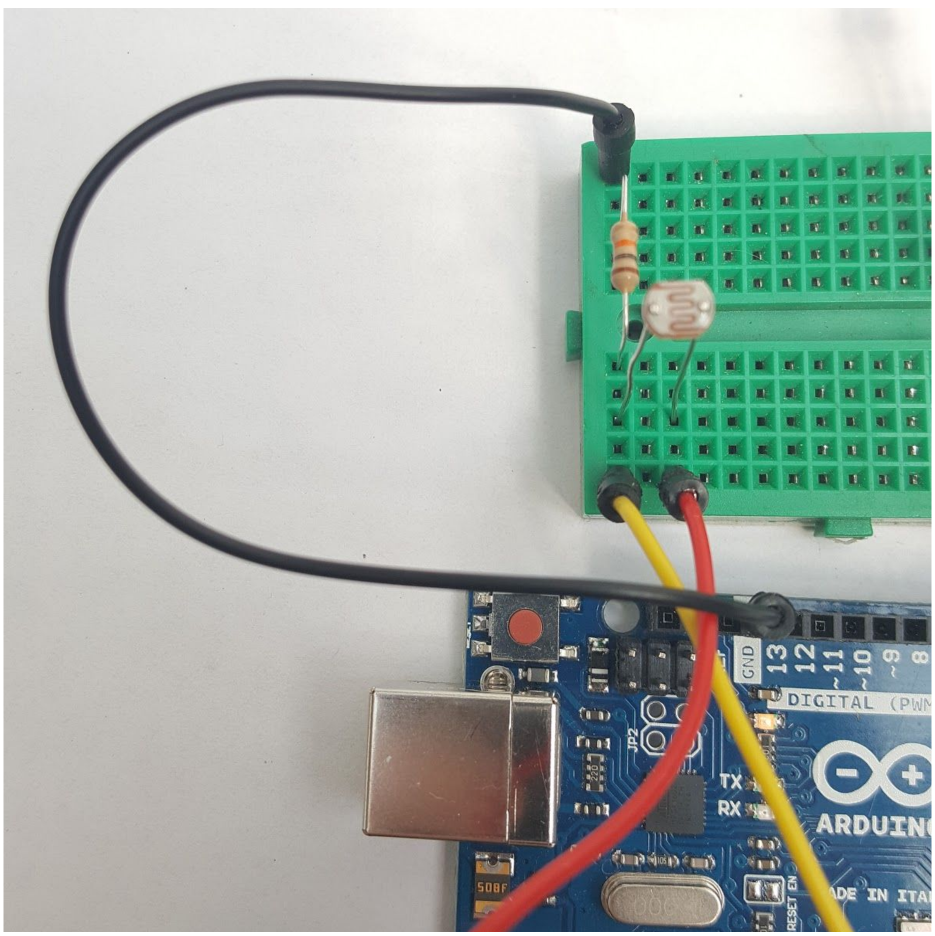

UNO
ARDUINO
MADE IN ITALY
DIGITAL (PWM~)
ANALOG IN
POWER

v. Use another wire to connect from the other Arduino GND to make another ground bar on the other side of the breadboard. This is where the photoresistor will go. The photoresistor part of the circuit is a little more complicated.

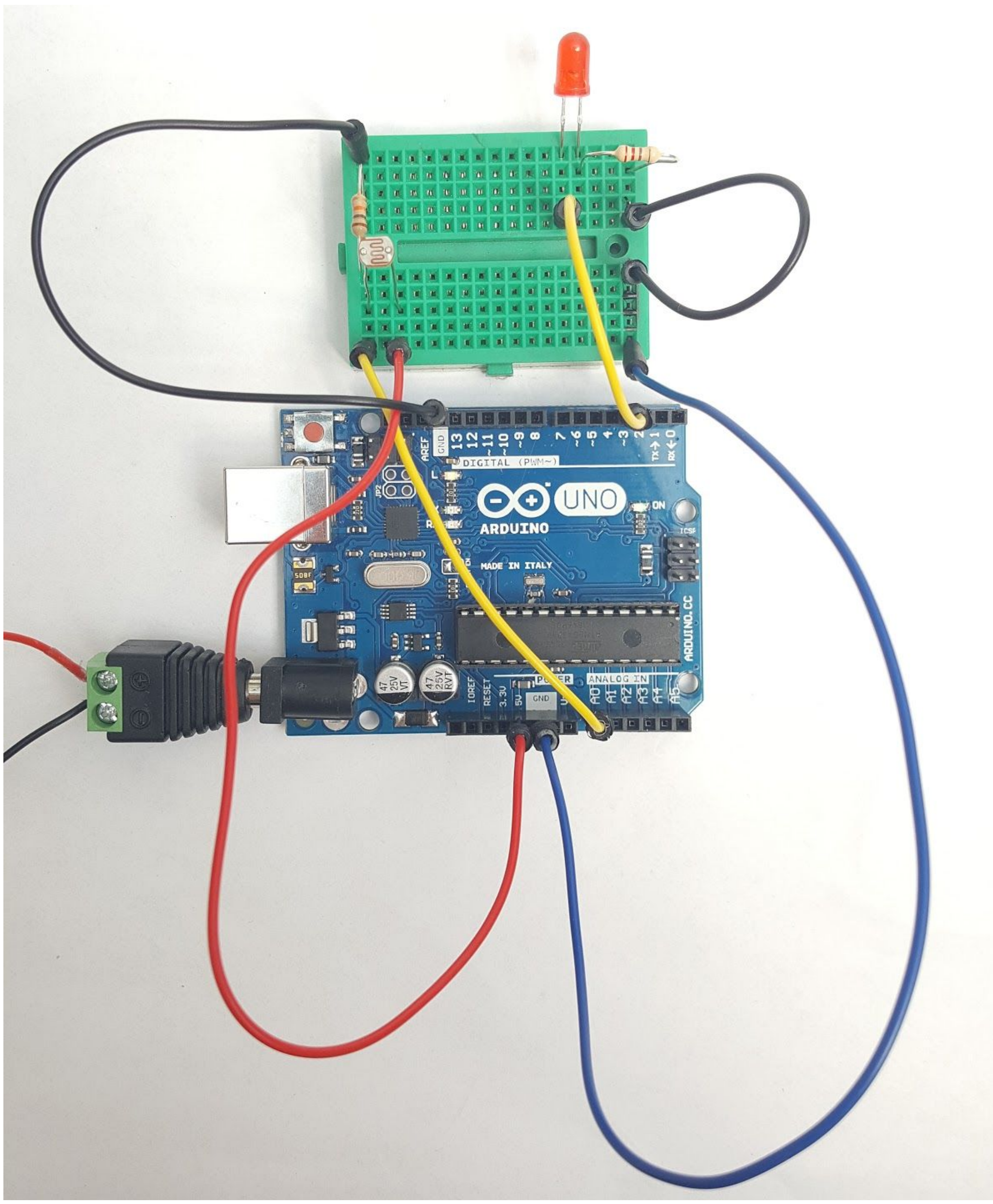
vi. Use a 10K ohm resistor to jump from the photoresistor ground bar across the gap of the breadboard. Then, in the same row plug in one side of the photoresistor - doesn't matter which one - and a jumper that goes to A0.

vii. Make a connection from the other photoresistor leg to the 5V pin on the Arduino.

It should look like this:



The completed circuit will look like this:



3. Give students time to build their Arduino circuits in pairs. Test to make sure it works.
4. Build or recycle a box to hold the arduino circuit.
5. **Clean up**. Put arduino circuits away for now. You will not need them for the following session.

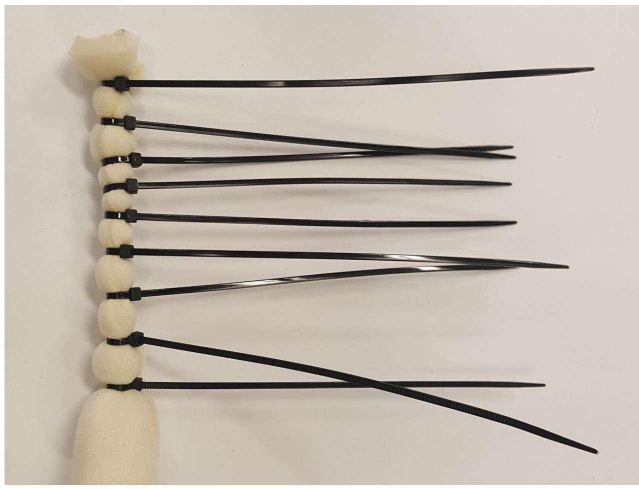
Session 14: Brainstorming and Fabrication

Classroom setup
Prototypes and Mouse Create, cardboard, cardboard cutters, squeeze clamps, scissors, staplers, paper, journals. Foam, armature wire, colorful yarn, large sewing needles, acetate, hot glue guns, popsicle sticks, brass tacks, permanent markers.

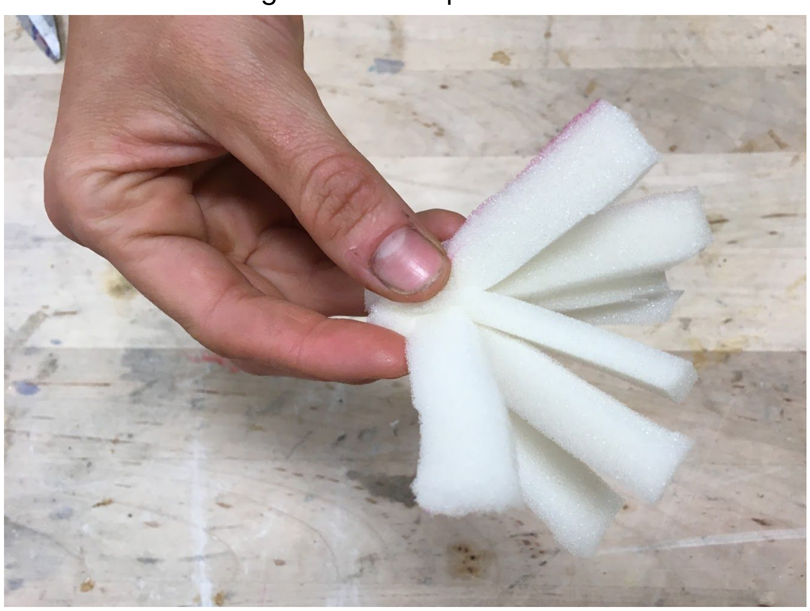
1. **Continue** building circuits, if students aren't finished.
2. As a class, brainstorm and write down a list of aspects of an imaginary future world.
    a. How is the ***environment*** different? What is the climate? The terrain? Are we on earth? Are you in a city? In a town? On a vessel? In a new wilderness? In the water? Underground?
    b. What is future human ***culture*** like in this world? Make a list of aspects of future cultures in that environment. Are humans the dominant species in this world? How will people collect the resources that they need to survive? Is it a dark period for humans, or utopian?
    c. Considering the environment and culture, what ***adaptations*** does your future human need to be successful? How can their appearance assist them in their lives?
3. **Design and Sketch:** Based on this class brainstorm, groups should write down and discuss ideas that interest them and make sketches of their ideas in journals.
4. **Look** at GIFs or finished prototypes of wearable future human adaptations to show how the following tools and materials can be used.
5. **Demonstrate** the following techniques for working with foam to create wearables.
    a. You can use staplers to hold foam together, instead of sewing.
        i. Remember to leave about a fingers-width of ***seam allowance*** as you staple. Seam allowance is the distance between the foam edge and the staple line on two pieces of foam that are being stapled together.
        ii. Place staples right next to each other as you go.
        iii. If a stapler gets jammed, you can pull staples out using needle nosed pliers.
    b. Appendages such as tentacles can be made by using a paper pattern to cut out two of the same shape and stapling them together with ***armature wire*** inside. The armature wire will allow the shape to bend and keep its form.

c. You can place an LED or photocell on the end of an armature wire appendage.

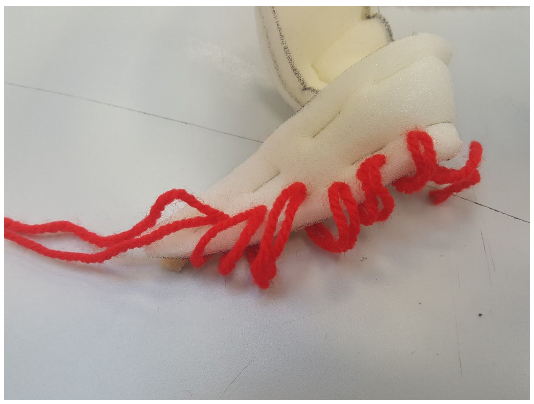
d. You can use cardboard to reinforce a structure, make a base or an underlying skeleton.
e. You can staple on velcro or elastic to hold things together.
f. You can use zip ties decoratively or to hold things together.



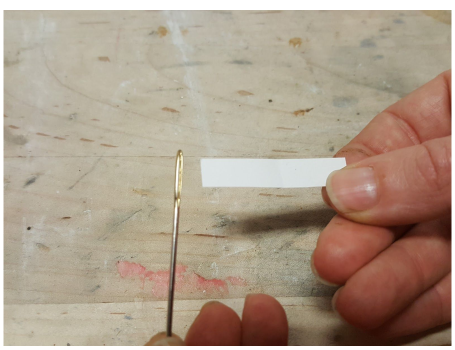
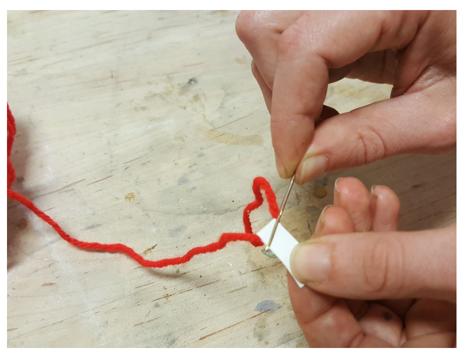
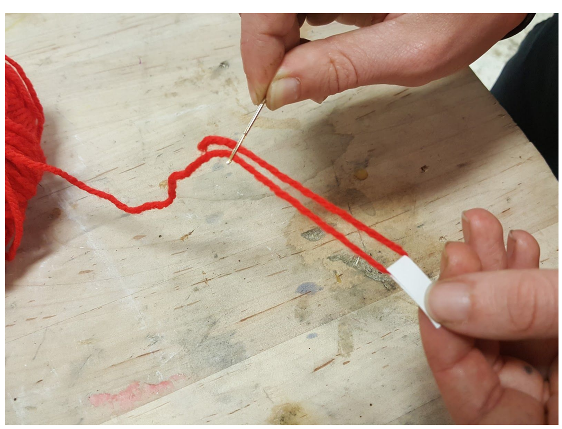
g. You can use also use brass tacks and popsicle sticks to hold things together.
h. You can cut smaller shapes such as warts, spikes, or pom poms and poke them through holes in the foam or cardboard to create surface texture. You can poke holes in the foam using a skewer or pencil.



2. **Demonstrate** sewing with yarn to add colorful detail, fur, or to help hold things together.

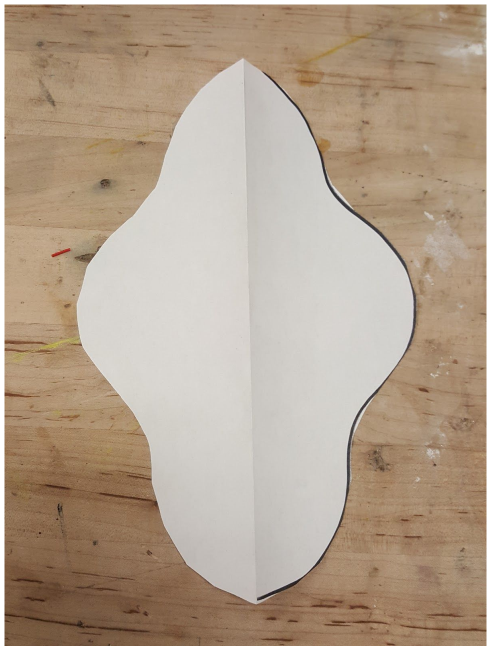


    a. Thread the needle with yarn and pull it all the way through using a small piece of paper that can fit through the needle.

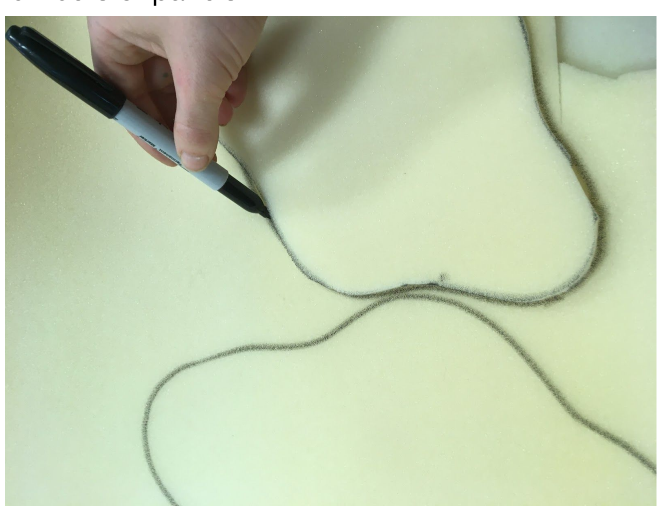




   b. Pull the yarn through the needle and tie a knot at the end of the ***tail***. As you sew, hold the needle over the hole so that the yarn doesn't pull through.
   c. As you poke the needle through the foam, support the foam from behind (being careful not to poke your hand).
3. **Demonstrate** pattern making by showing images or examples.
   a. ***Pattern making*** is a tool that allows you to ***trace*** and repeat a paper shape multiple times to create wearable prototypes.
   b. You can make ***3D forms*** using the following technique. Think of the way that all the panels on a football are the same shape and combine to create a rounded form.

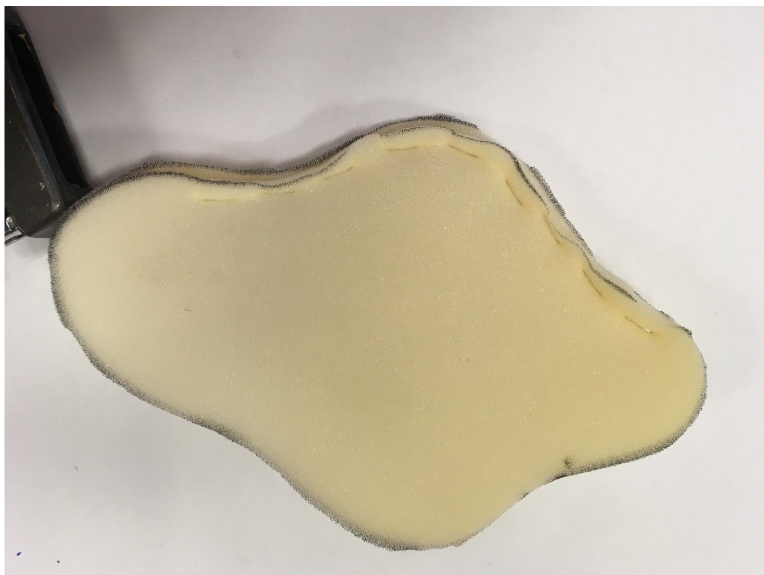
      i. Fold a piece of paper in half. Draw and cut out a shape. Unfold so you have a ***symmetrical*** shape.



ii. Trace the outline of your paper shape onto foam, staying close to the edge of the foam and to each other to avoid waste! Decide how many ***panels***, or sides, you want your 3D form to have: Move the paper and trace at least three times. You'll get different results with different numbers of panels.



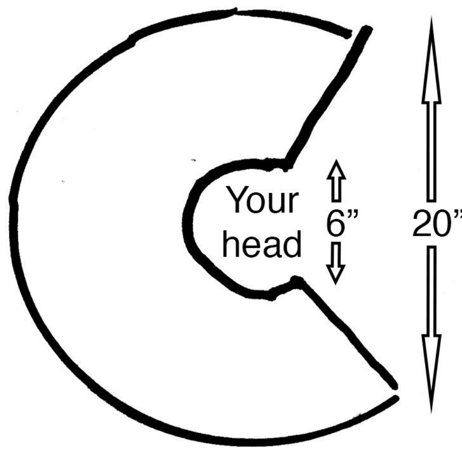
iii. Line up the edges of two of the foam shapes with marker marks on the outside, and staple along the edge, starting at the top where the fold mark was on the paper and moving all the way down one side.



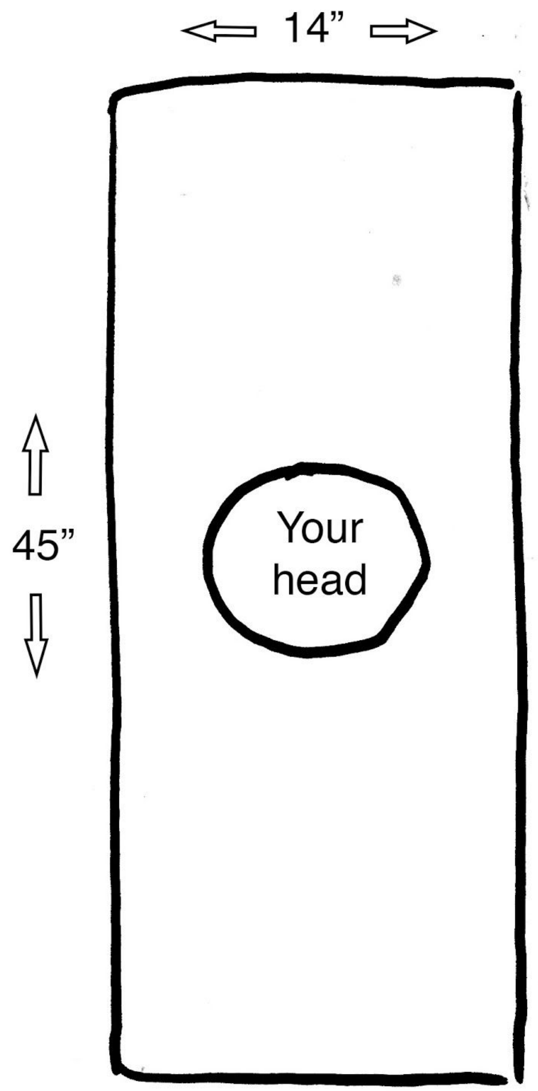
iv. Once you staple your way down, match up the edges of the *next* cut-out foam shape and staple that shape to the previous one, again starting at the top of the fold mark and moving all the way down to the bottom. Do this for each shape until you've added all of your panels.

v. Before you completely staple your form together, turn it inside out to hide your ***seams***!

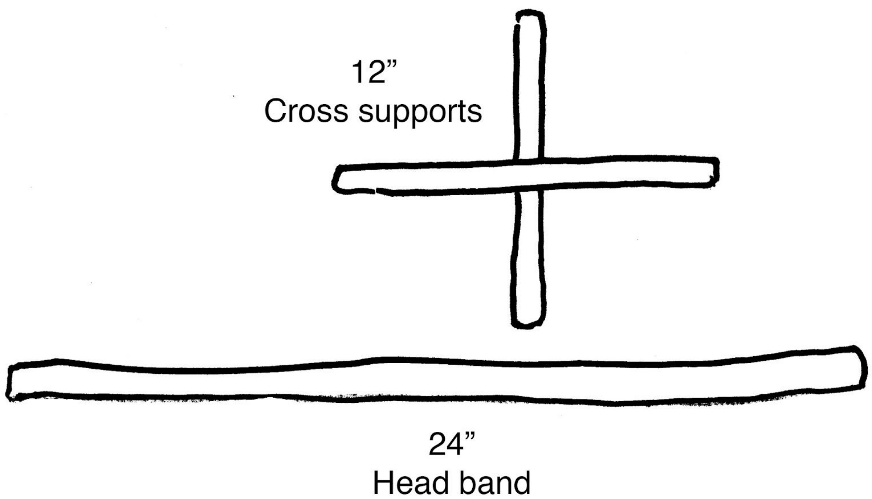
4. The following patterns are for you to build off of or connect new parts to make your project unique. The pattern is intended as a base for you to build from.
    a. **Cape** patterns can be used to make prototypes that can be worn around your shoulders.



    b. **Bib** patterns can be used to make prototypes that can be worn over a T-shirt.



c. **Headgear** is anything that goes on your head. You can use the headgear pattern to build a skeleton out of cardboard.



5. **Build guidelines:**
    a. Work together as a group to make one prototype wearable that combines your ideas.

b. After building, we will use Arduino to create repeating patterns in light and control the speed of the blinking patterns using a photocell. Consider how lights and a sensor can be used to bring your idea to life.
c. You can incorporate any other available materials you know how to use.
d. **Draw** a diagram of what you want to make. As you work and build on your ideas, add the following information to your schematic.
    i. What does the main form look like? What necessity inspired its shape?
    ii. What other physical characteristics does it have?
    iii. Are there moving parts?
    iv. Where are the LEDs and what color are they? Where is the photocell? How is it covered and uncovered?
    v. Draw the circuit into your diagram.

6. **Play! Explore your ideas** by cutting shapes out of paper and using staples to hold things together. Make a paper first draft to make sure it fits the way you want it to.
7. Clean up.

Session 15 and 16: Build Prototypes

Classroom setup
The same as session 14.

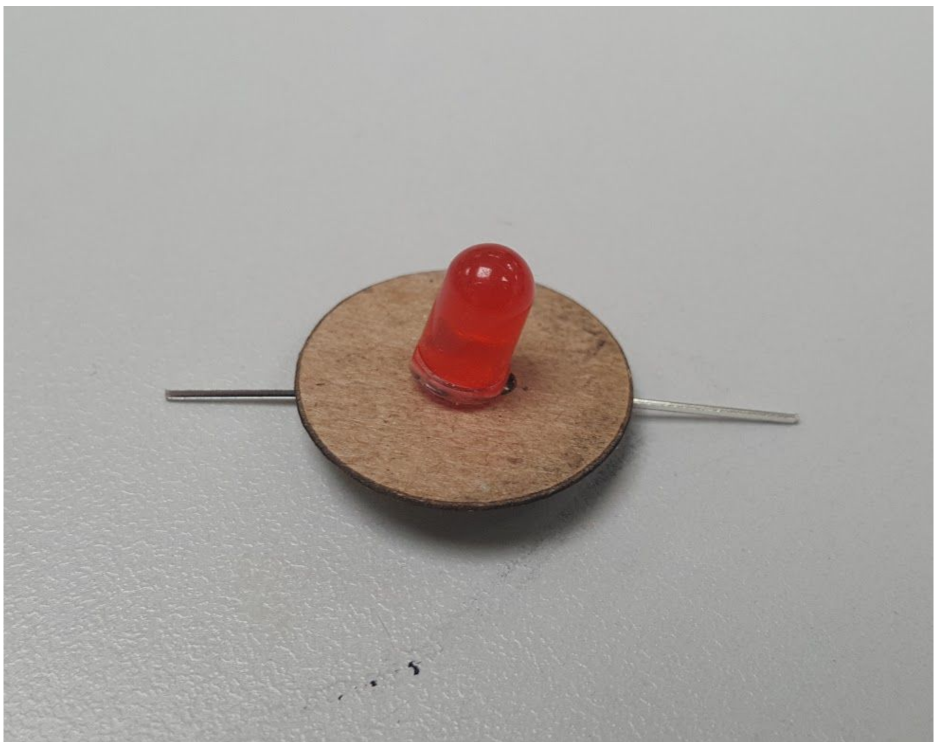
1. **Build!** Try on and improve.
    a. Determine where you want LEDs and photocell to be incorporated.
2. Clean up.

Session 17: Incorporate Electronics

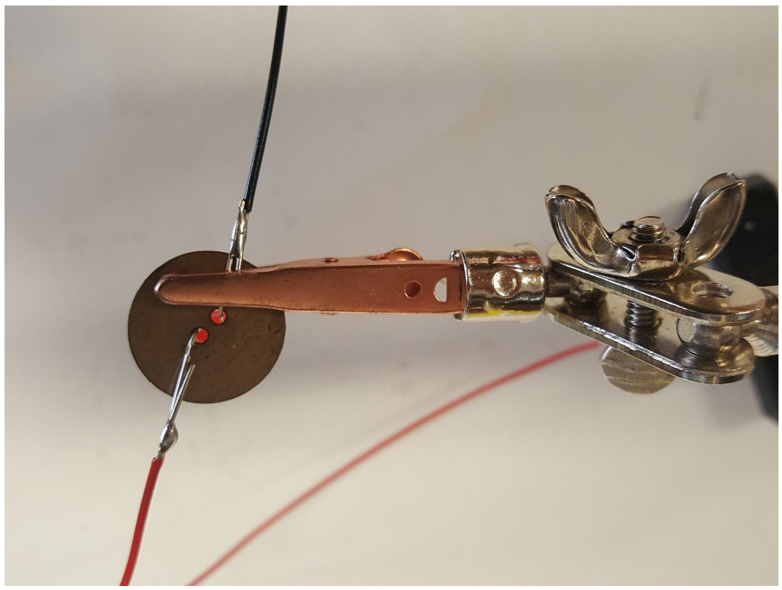
Classroom Setup
Wire, soldering supplies, LEDs, LED holders, wire strippers, electrical tape.

1. **Demonstrate** methods for how to incorporate LEDs into projects.
    a. First, consider how long your wires need to be in order to reach the Arduino. Cut the wires *at least 2 feet long* and separate the ends from each other. You can cut between the two wires to split them. Strip the wires using size 22 holes on the wire strippers.

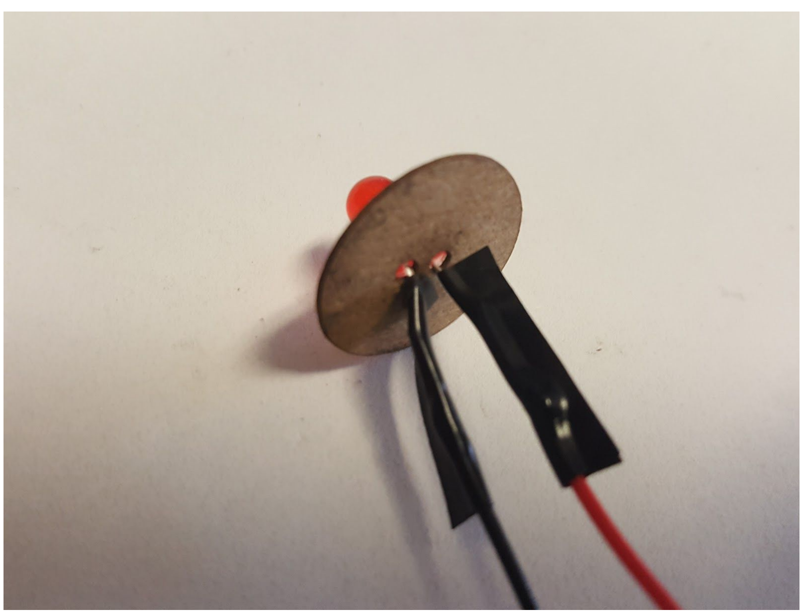
b. Feed the legs of the LED through the LED holder and separate them to temporarily help hold the LED in place. Make sure to notice which leg is longer - the longer leg is positive. (Students don't need to use the LED holders, but they'll help for surface mounting LEDs onto foam).



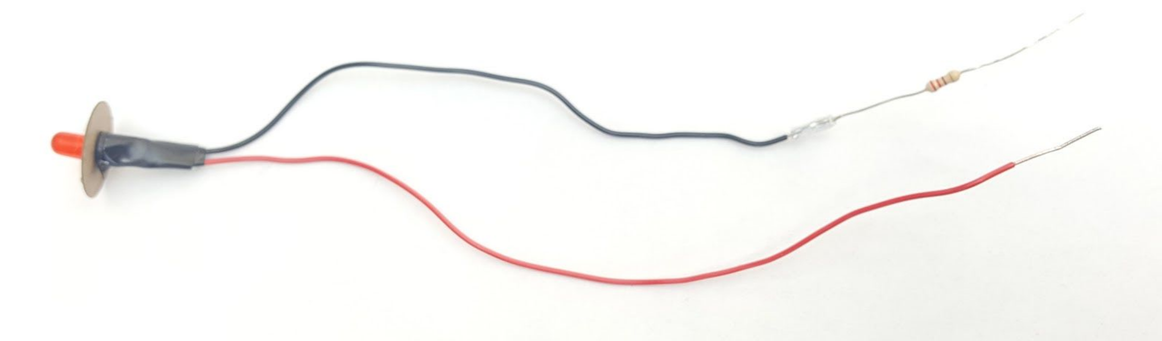
c. Connect the longer leg of the LED to the colored wire (positive) and the shorter leg to the black or darker color wire (ground). Solder the legs of the LEDs to the wires.



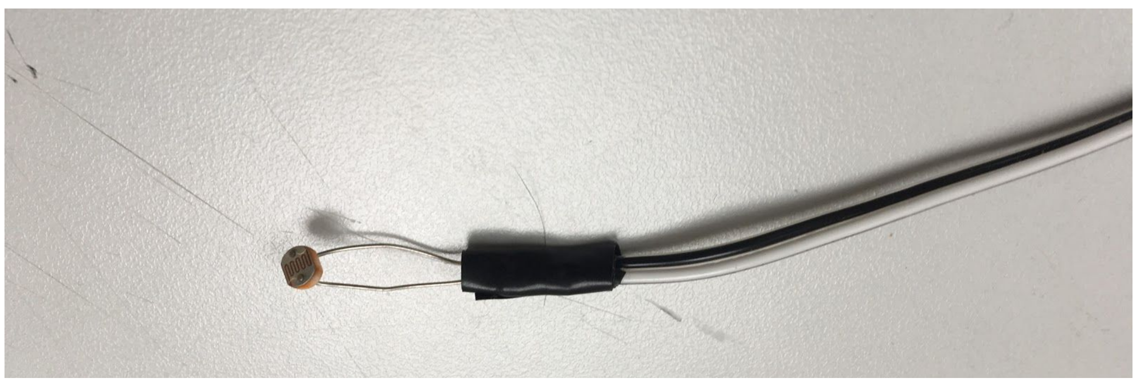
d. **Insulate** the individual legs using electrical tape so that you don't get a short circuit on the LEDs!



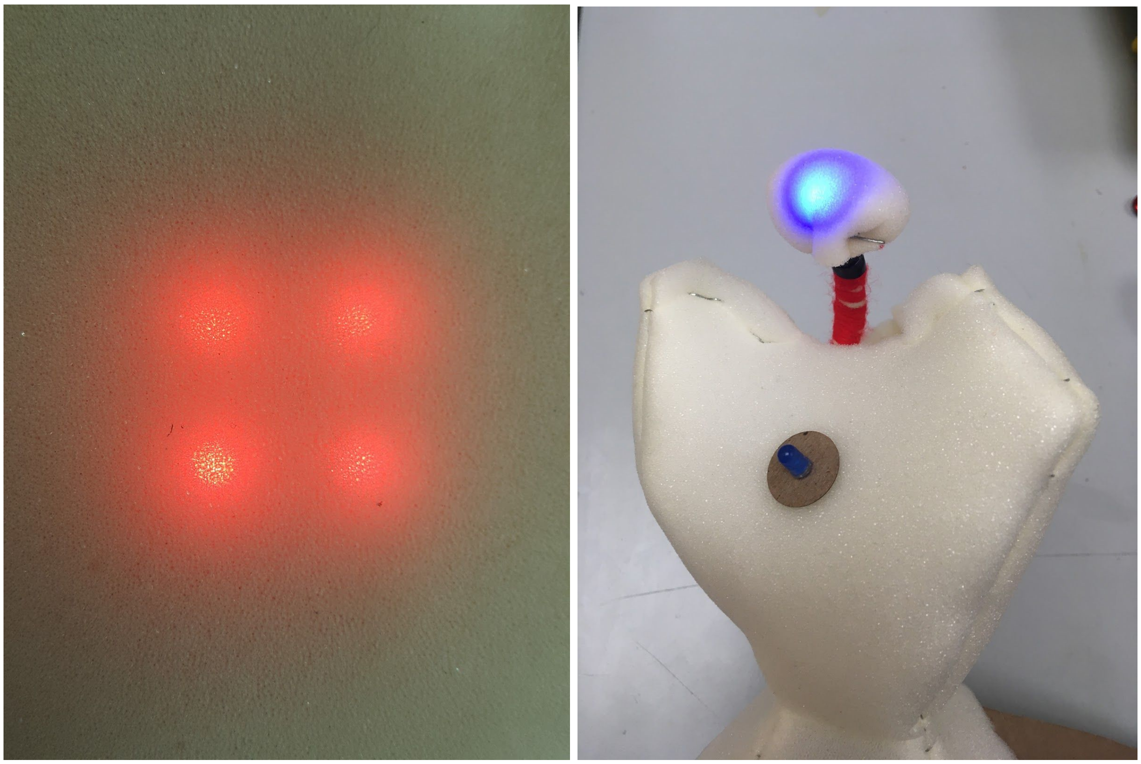
e. Solder a 220 ohm resistor to the *negative* side of the LED.



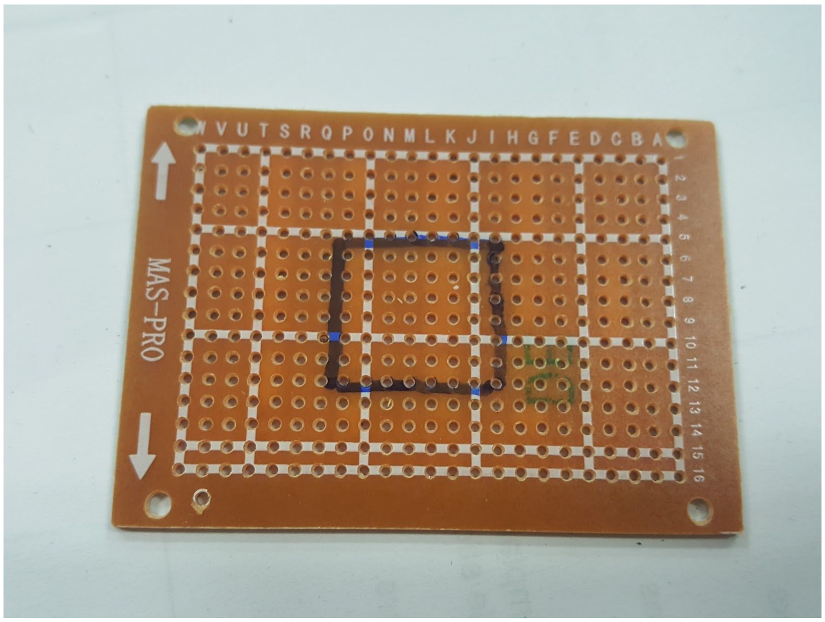
f. Extend the legs of the photoresistor by soldering wires to both legs. The photoresistor does not need a resistor. Insulate the photoresistor legs as well.

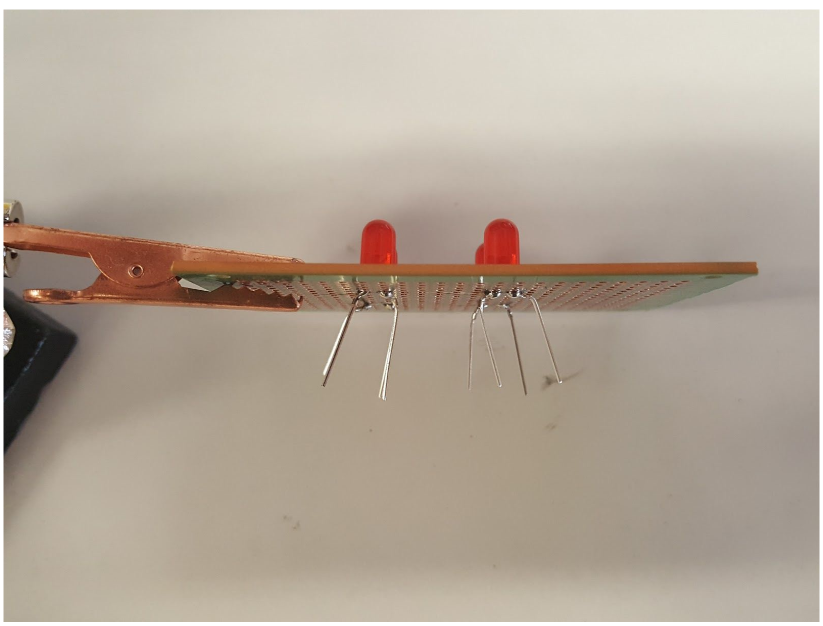


g. Using a skewer or a pencil, punch a hole in the foam where you want the LEDs or photoresistor to sit, and thread the wires through the holes so that they're hidden.
h. You can ***diffuse*** the LED light and create a softer glowing effect by adding foam over it or putting it inside of a form - like a lantern.



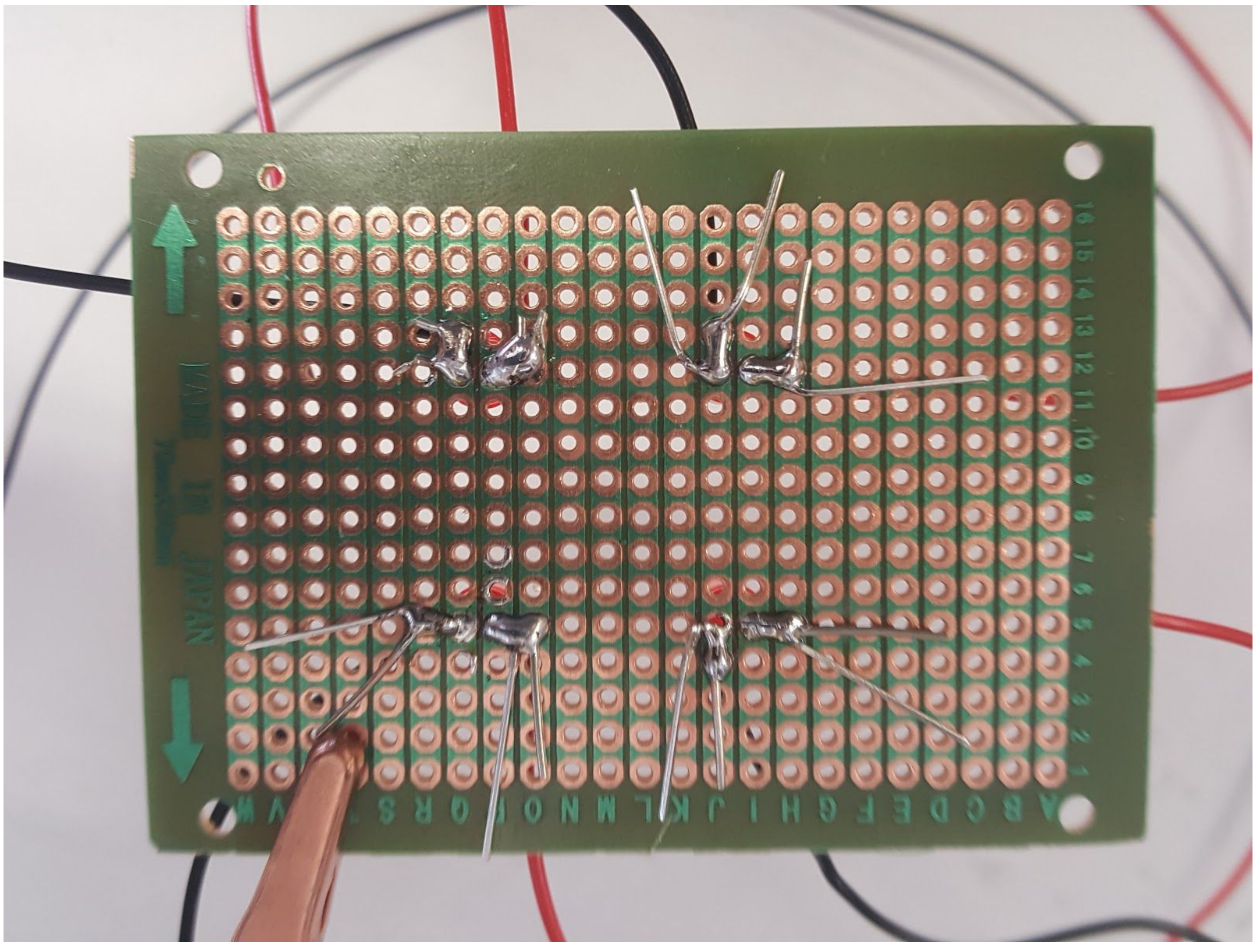
2. **Demonstrate** how to solder LEDs to perfboard to create grid like patterns and shapes.
    a. In order to use the perfboards to organize your LEDs, start my drawing you desired pattern in permanent marker on the perfboard. Place your LED legs through the holes in the perfboard so that the legs come out of the conductive side.



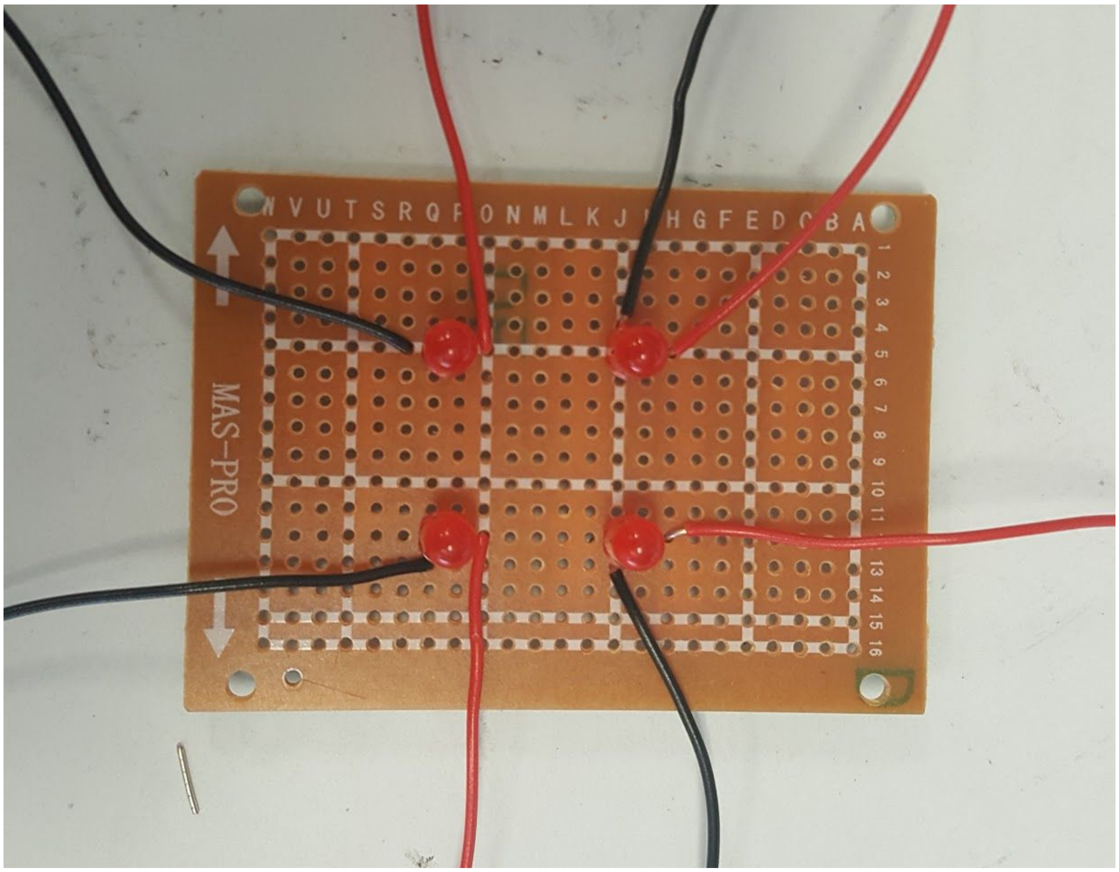


b. Bend the legs to the sides, away from each other. Remember to never let any two legs touch! Stick the wires through the perfboard from the front, so that they meet the LED legs on the conductive side of the perfboard. The wires should stick through holes that are close to where the LED legs are so that they can be soldered together.

c. Trim the excess wires from the back of the perfboard.



d. The top surface of the perfboard will look like this at the end.

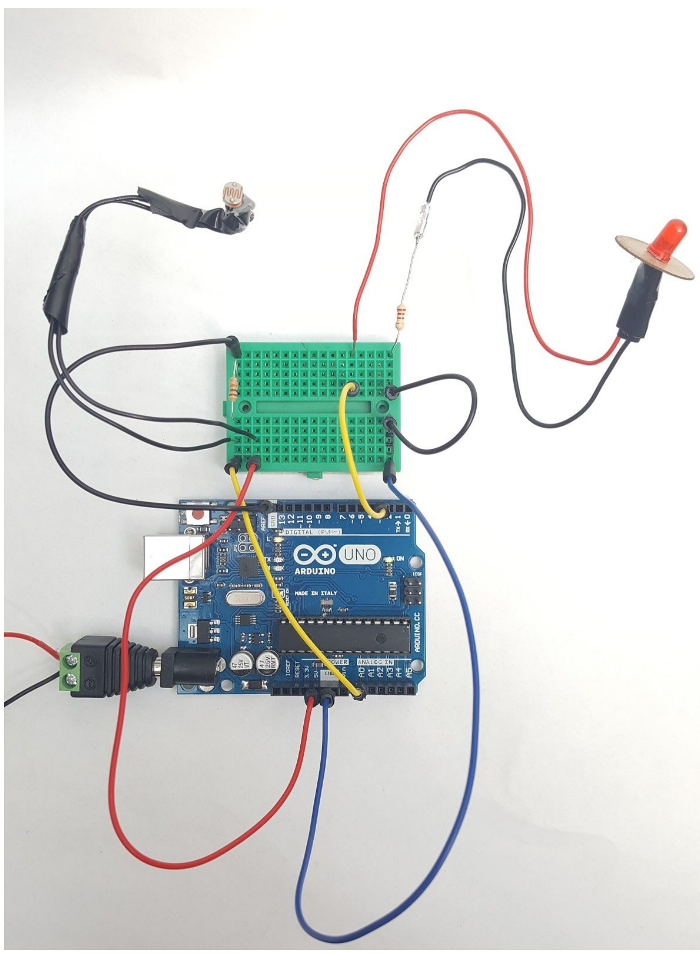


3. Students should incorporate LEDs and photocells into their projects before moving on to connect them to the Arduino and breadboard.
4. Clean up.

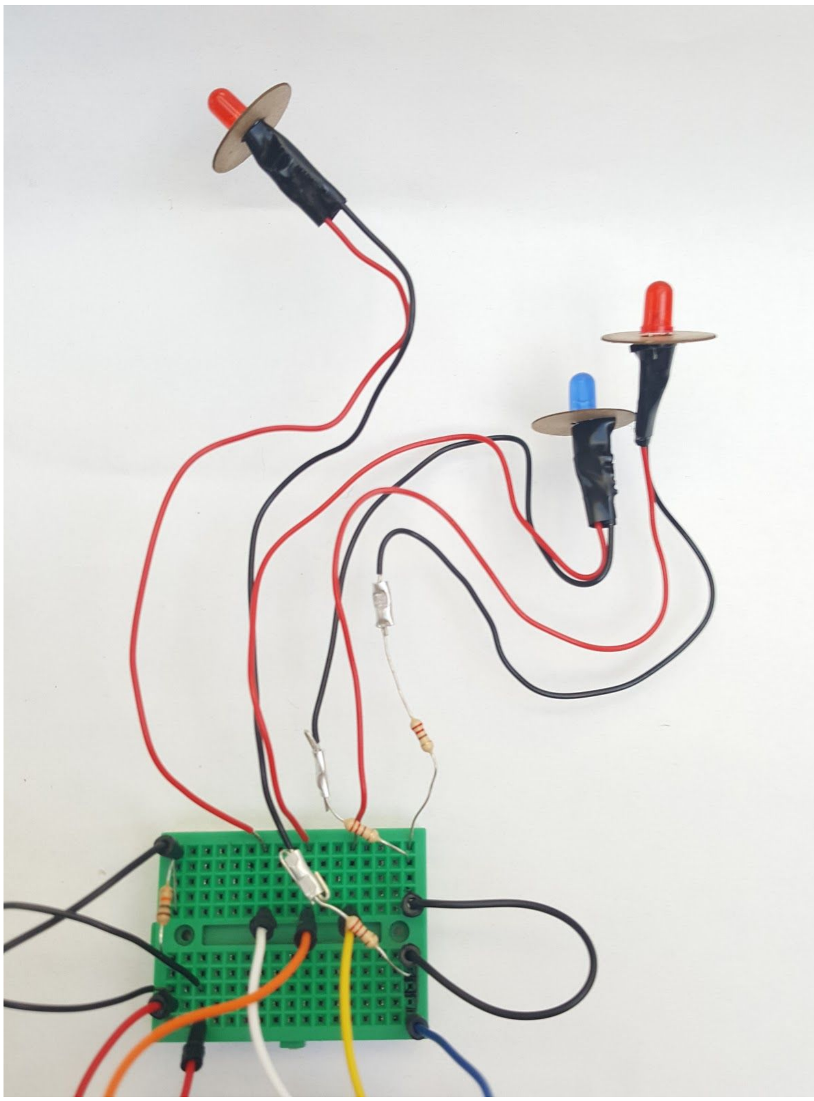
Session 18: Creating Light Patterns

Classroom setup
Perfboard, Arduino circuits, soldering supplies, laptops, USB cables.

1. **Explain**: Now prototypes have been built and LEDs and photoresistors incorporated, it’s time to connect the wires to the Arduino and breadboard.
    a. Use this image to help you connect your LEDs and photocell to the breadboard. The circuit will be the same, except that negative leg of the LED (with the resistor on it) can go straight to the ground bar.



b. Any additional LEDs will follow the same pattern: the positive leg connects to a ~ pin, and the resistored leg goes to the ground bar. **Make sure to insulate any exposed wire!** *Plug your LEDs into the Arduino in the order that you want them to turn on.*



c. If you've tweaked your code at all, upload it now.

2. **Evolve**: Once you've built a prototype and connected it to an arduino, test it. Does it function? Does it look good? How can the design be improved?
2. Clean up.

Session 19: Complete Prototypes

Classroom setup
The same as session 26. Paint markers.

1. **Continue** building and finalizing prototypes.
    a. Use paint markers to add detail or color to prototypes.
    b. Use available materials to add meaning or sculptural elements to prototypes.
    c. Incorporate arduino circuit and connect to your project somehow.

Session 20: Prototype Fair

Classroom setup
Mouse Create, phones or cameras, journals.

1. **Prototype fair!**
    a. Create a setting in which all students can come together and share what they've made with each other.
    b. Students should demonstrate and explain what their prototype is and demonstrate how it works.
2. **Document** projects using a camera.
1. **Upload** to Mouse Create
    a. Upload images of journal sketches and final product.
    b. Describe your prototype. How is it used by a future human? How does it work?

Chameleons Code Cheat Sheet

Chameleons Sketch

This code lets students control a sequence of LEDs using a photoresistor as a sensor.

VOID SETUP()

In this sketch, we are setting up the Serial Monitor (so we can see our photocell information) and the digital pins the LEDs plug into. We are telling the Arduino that certain pins will OUTPUT electricity so that they can power the LEDs.

Students will start their circuits by incorporating only one LED and photocell, but they can eventually add up to six LEDs using the pins listed below. These are all the pins that have ~ next to them.

Students shouldn't really need to change setup().

```
void setup() {

  Serial.begin(9600);  // start up the Serial Monitor.

//////  SETTING UP THE PINS FOR YOUR LEDS
  pinMode(3, OUTPUT);        // These are the pins that have the ~ symbol on the Ardu
  pinMode(5, OUTPUT);        // They have special capabilities that let us fade LEDs
  pinMode(6, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);

}
```

VOID LOOP()

Just like in the previous programs, loop() is where we will put the input and output parts of our program. This time, our input will be a photoresistor, and our output will be lighting up LEDs.

Sensor Reading

Just like with Bionic Mechanisms, we are going to read information off of a sensor. This time you'll be using a photoresistor (light sensor), which will most likely report a different range of numbers than your flex sensor did. Upload the code, play with covering up the sensor or shining light on it, and see how it reacts in the Serial Monitor .

```
////// SENSOR READING
  int sensorReading = analogRead(A0);    // read what the sensor is telling u
  Serial.println(sensorReading);         // show what the sensor is telling u
```

If Statement: Using a Sensor to Control LEDs

The rest of our program will be contained within an **if statement.** An if statement is a special piece of code that has a **condition** and a **result**. If the condition is met, the result code will run. If the condition is NOT met, the result code will NOT run. We can use this to make our LEDs shine only if the photoresistor sees darkness.

The **condition** is located in (   ), and the **result** code is between {   }.

Students may want to change the **condition** based on the numbers their photoresistor gives them from the Serial Monitor.

Switching LEDs on and off

Next are the commands to turn LEDs on and off. We can do that using the digitalWrite() command with an Arduino pin number and HIGH or LOW. You can control how long they stay on or off with the delay() command. The delay() command takes a number in *milliseconds*, and 1000 milliseconds equals 1 second.

Students may want to change the order and timing of their LEDs. They can also delete any commands that refer to pins they're not using.

```
////// SWITCHING YOUR LEDS ON AND OFF
    digitalWrite(3, HIGH);  // The LEDs on our six pins switch ON,
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(10, HIGH);
    digitalWrite(11, HIGH);
    delay(2000);            // and then they shine for two seconds.
    digitalWrite(3, LOW);   // Then, they all get switched OFF,
    digitalWrite(5, LOW);
    digitalWrite(6, LOW);
    digitalWrite(9, LOW);
    digitalWrite(10, LOW);
    digitalWrite(11, LOW);
    delay(500);            // and they stay off for half a second.
```

Fading LEDs Up and Down

In addition to switching LEDs on and off, we can also use a *fade* effect to gradually fade them up or down. We can do this with the glowLED and dimLED commands, which both need two numbers to run. The first number is the pin of the LED, and the second number is how much time it takes to completely fade (a higher number will take longer). A speed number from 1 (fast) to 10 (slow) is recommended.

Students may want to change the order and speed of their fades. They can also delete any commands that refer to pins they're not using.

```
////// FADING YOUR LEDS UP AND DOWN
    glowLED(3, 4);     // The LEDs on our six pins will fade UP at a relat
    glowLED(5, 4);
    glowLED(6, 4);
    glowLED(9, 4);
    glowLED(10, 4);
    glowLED(11, 4);
    delay(2000);       // They will shine at full fade for 2 seconds,
    dimLED(3, 4);      // And then they will fade DOWN at the same speed.
    dimLED(5, 4);
    dimLED(6, 4);
    dimLED(9, 4);
    dimLED(10, 4);
    dimLED(11, 4);
    delay(1000);      // The program will wait 1 second before the loop re
```

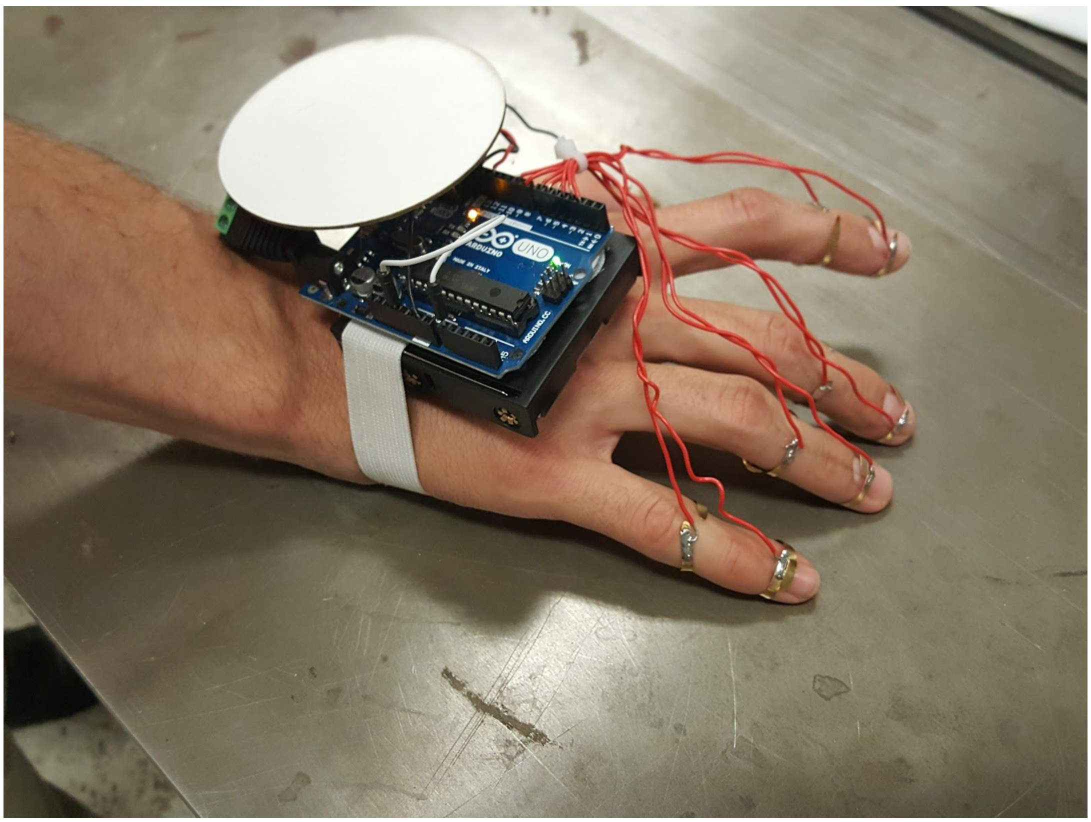
Ending Our If Statement and Loop

The final part of our program is just closing out the { }s for our if statement and our loop(). Then, as always, the loop will start over from the beginning.

```
  }                              // End of our if statement.

}                                // End of our loop() area.
```

Sessions 21-28: Bees



Project Overview

Objectives
Using sound to sense your environment with hand made switches
Manipulating musical patterns with Arduino code
Building speakers and interactive sound-systems

Project Details
Students will build mechanical switches that trigger musical patterns to play from an Arduino. Then, they'll design the way that the sound is experienced by building speakers and vibrational

sound systems.

It’s recommended that students work in pairs.

Tools and materials

- Electronics and Circuitry
    - Solid core wire
    - Breadboards
    - Arduinos
    - Exciters
    - Soldering supplies
    - Potentiometers
    - Conductive foam
    - AA Batteries
    - 4AA battery packs
    - Multimeters
    - Safety glasses
- Fabrication
    - Colored paper
    - Copper tape
    - Armature wire
    - Cardboard
    - Felt
    - Foam
    - Tape
    - Tubing
    - Funnels
    - Staplers
    - Velcro
    - Cardboard cutters
    - Scissors
    - String
    - Markers
    - Elastic
    - Brass tacks

Project Flow

Session 21: Introduction

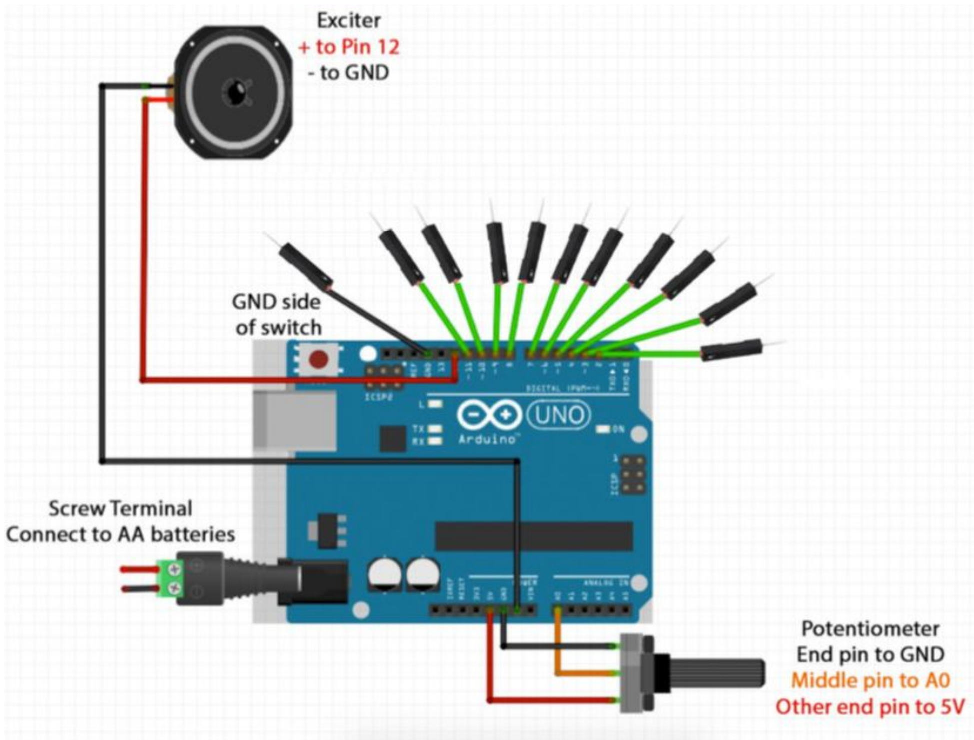
Classroom setup
Prototype, Mouse Create, Bees circuit diagram, Arduino, breadboard, jumper wires, potentiometer, exciter, soldering supplies, solid core wire, AA batteries, battery holder, laptops, USB cables.

Cardboard, cardboard cutters, perforators, brain box template, tape, hot glue, markers, rubber bands, rulers.

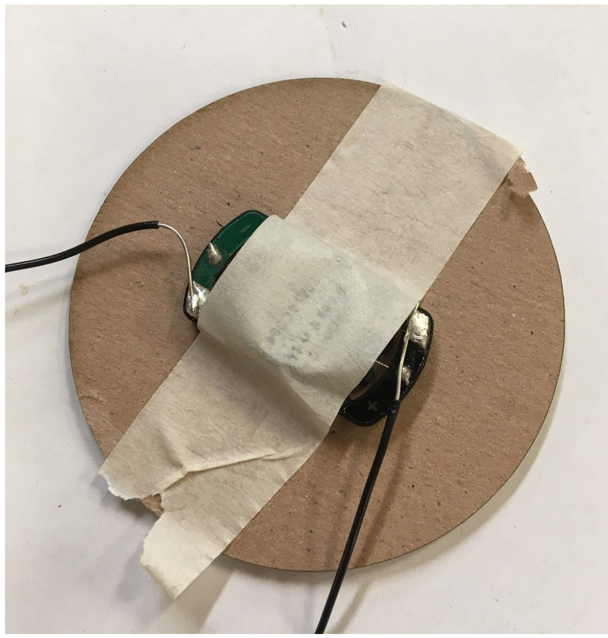
1. **Introduce**: What's the main idea? Use Mouse Create to introduce the project.
    a. Vibration is used by lots of different animals to sense and communicate information about the environment. Animals have adapted in different ways to incorporate high level vibration sensors and receptors into their bodies. For example, specific receptors inside the antennae of a bee can detect movement allow the bee to judge its speed. Honeybees "dance" generates vibrations that communicate things to one another, such as the location of nearby food sources.
    b. **What other animals communicate using vibration or sound? Ask** students to think about times when they've sensed something was happening through sound or vibration only.
    c. **Explain:** ***Sound*** is nothing but vibrations of the air around us which, in turn, vibrates our eardrums. A ***Tone*** is a ***periodic vibration*** which can be heard or felt by human - the ***frequencies*** of these vibrations are what we perceive as different musical ***Notes***. Speakers generate sound by an electromagnet being vibrated by an electrical signal.
2. **Explain the main objective**: During this domain, we'll be building sound systems using ***exciters*** to play patterns of ***tones*** of different frequencies from an Arduino, which we'll trigger using home-made mechanical switches. First, we'll build a prototype circuit.
3. **Explain:** What's an ***exciter***?
    a. ***Exciters*** are speakers that do not have a cone attached. When an exciter vibrates, you will *feel* the sound waves but not hear them. Exciters make audible

sound *only* when physically connected to a piece of paper, cardboard, a tabletop or surface that can vibrate air.

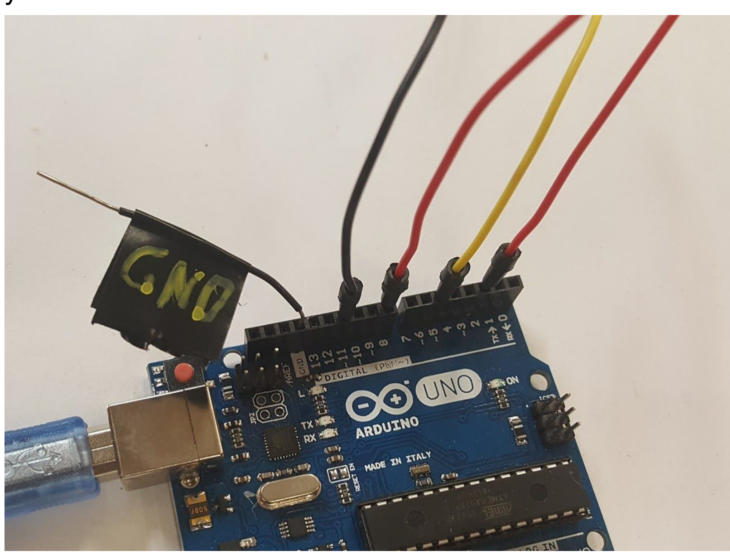
4. **Explain:** What's a ***potentiometer***?
    a. Potentiometers are ***variable resistors*** that allow us to change the length of the ***delay*** we hear between tones (how spaced out the notes are). When the potentiometer is turned all the way in one direction and we close a switch, our code will generate a solid tone. When it's turned all the way in the other direction, we'll hear ***pulsing*** notes (also known as an ***arpeggio*** in musical terms). In between, we'll hear a range of speeds.
    b. A potentiometer is similar to the flex sensor you built during fundamentals, or the photoresistor you used in the Chameleons domain. You will have the choice, later in the domain, to substitute the potentiometer for a flex sensor or photoresistor in your prototype!
5. **Explain:** What's a switch***?***
    a. A ***switch*** or button is just a mechanism that physically connects and disconnects two conductive surfaces within a circuit. Any ***conductive*** material can be used as part of a circuit to create a switch.
    b. A ***mechanism*** is made up of controllable moving parts. For example, buttons use foam to separate two conductive surfaces which can be connected by pressing them together. Light switches are levers that bridge a gap between two conductive surfaces, completing the circuit.
    c. We will prototype the circuit first by simply touching jumper wires together to make temporary switches. When you build your prototypes, you will make actual switches.
6. **Upload code to Arduino.** Each pair or group of students should build one circuit.
    a. Plug your Arduino into a laptop using a USB cable.
    b. Open the **Bees** code in the Arduino IDE and upload it to your Arduino.
    c. *You will have an opportunity to edit the code later on in the domain.*
7. **Demonstrate building circuit step-by-step**: in pairs, you will follow the diagram to make connections between the Arduino, jumper wires, potentiometer and exciter.



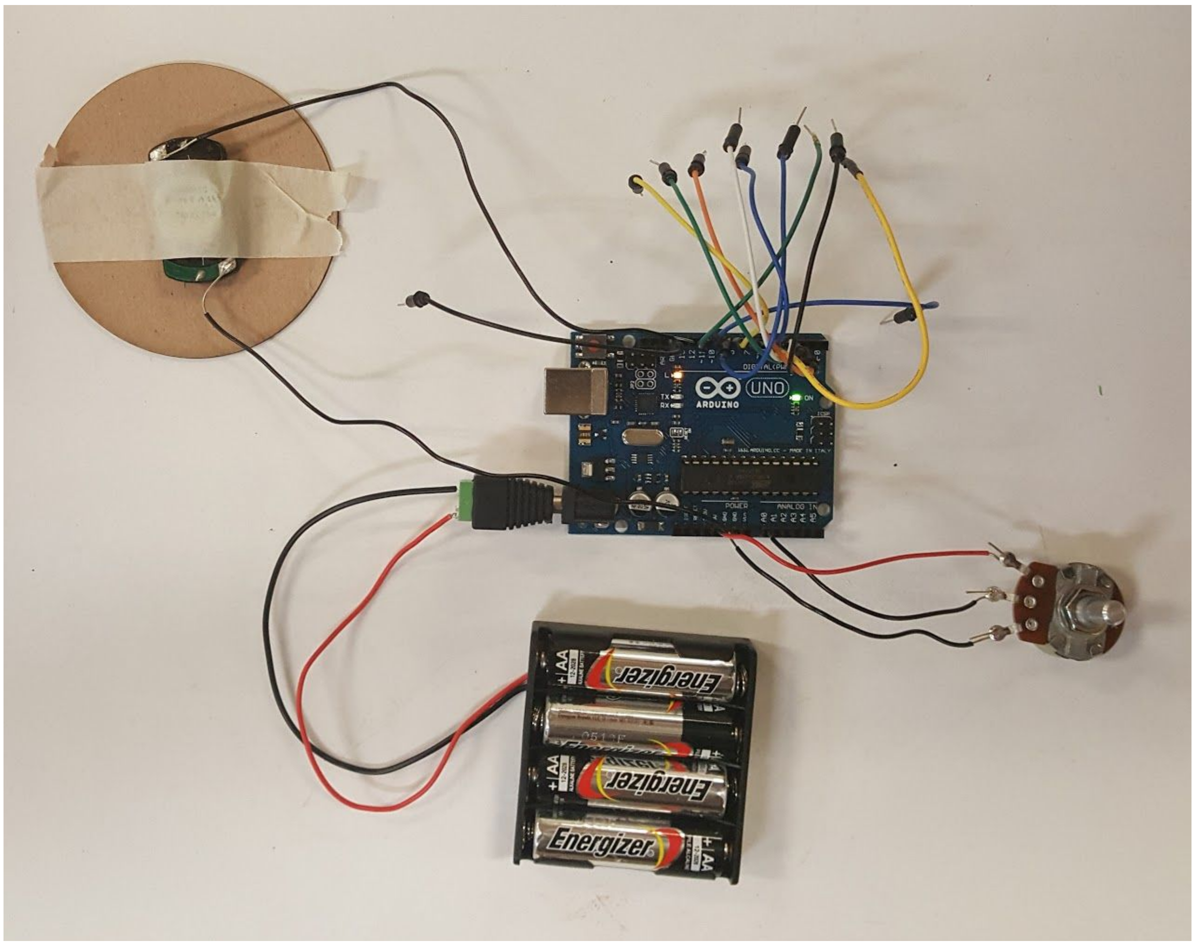
a. Strip and solder a wire to each of the two tabs on the back of the exciter. Tape the exciter to a cardboard circle so that you can hear the sound when it plays.



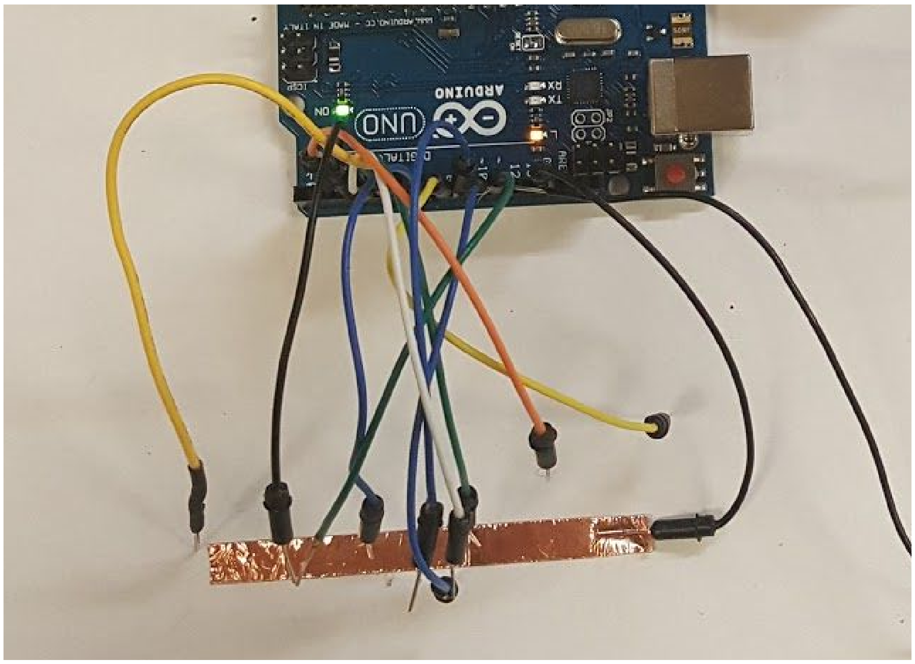
b. Plug a battery pack into the Arduino. The lights should turn on.
c. Connect the exciter to pins 12 and GND on the Arduino. Connect wires to all of the pins from 2-11 on the Arduino. Each pin corresponds to a different tone.
d. It may be useful to label the GND wire.



e. Strip and solder one wire to each of the three terminals of the potentiometer. The left wire goes to GND, the middle wire goes to A0 and the right wire goes to 5V on the Arduino.
f. Your circuit should look like this:



g. **Demonstrate how to play!** When contact is made between the free GND jumper and any one of the digital pins (or multiple pins at a time), an electrical signal will output from pin 12 driving the exciter to make a tone. When you turn the knob on the potentiometer, it will affect the spacing of the tones.
h. You can also attach GND to a piece of copper tape, and then touch the jumpers to the copper tape strip.
i. You can remove the cardboard circle and experience the sound by touching the exciter.



8. **Give students time to build their Arduino circuits in pairs and test them.**
9. **Build** or recycle a box to hold the arduino circuit.
10. Clean up.

Session 22: Building Sound Systems

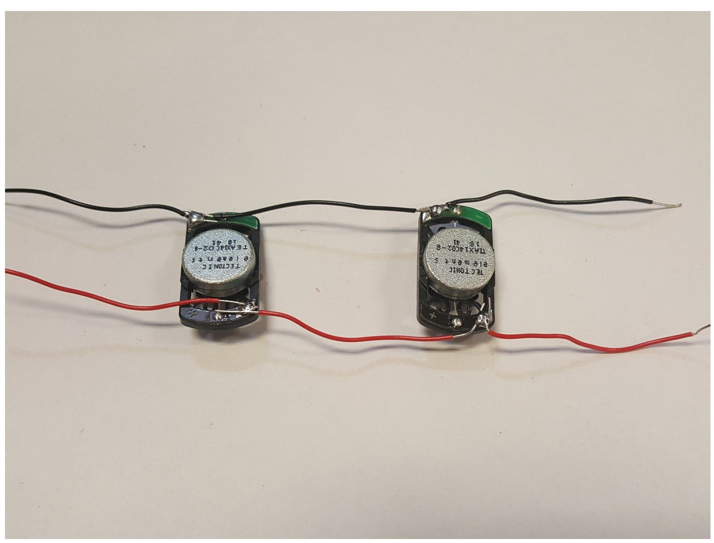
Classroom setup
Cardboard, cardboard cutters, colored paper, scissors, hot glue station, tape, funnels, plastic tubing, copper tape, soldering supplies, staplers, felt, foam, armature wire, velcro, elastic, markers, brass tacks.

1. **Finish building arduino circuits, if necessary.**
2. **Explain** anatomy of a sound system using GIFs or physical prototypes. Students have two main considerations: speakers and switches.
    a. **Speakers**: How is the sound felt or heard?
        i. Use the cardboard building techniques you've already learned to make your speakers that can be held, worn, or experienced.
        ii. You can experience audio signal as either vibration when you touch the exciter, or as audible sound when you connect the exciter to a surface.

Different surfaces will yield different results! The ideal material for making audible sound with an exciter is a thin, lightweight sheet of material that can bend. Paper, cardboard, plastic cups, windows, walls, cereal boxes, or table tops will all work.

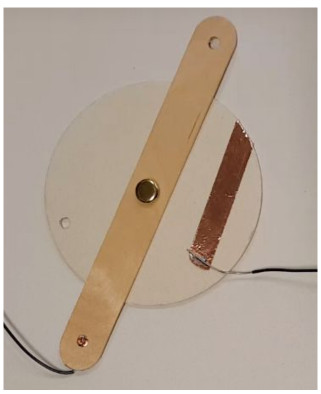
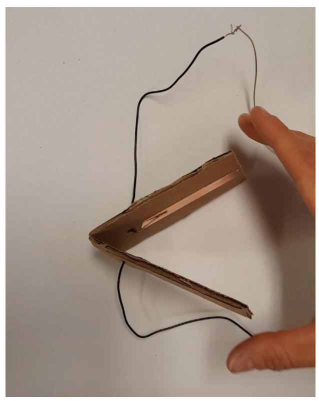
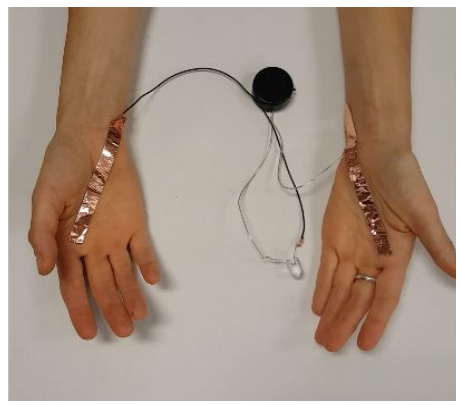
iii. The stickered side of exciters should be mounted to surfaces that can vibrate using the sticker provided or a tiny bit of careful hot gluing around the outside -

iv. You can solder two exciters in ***parallel*** in order to get sound out of both at once, as in the picture below.

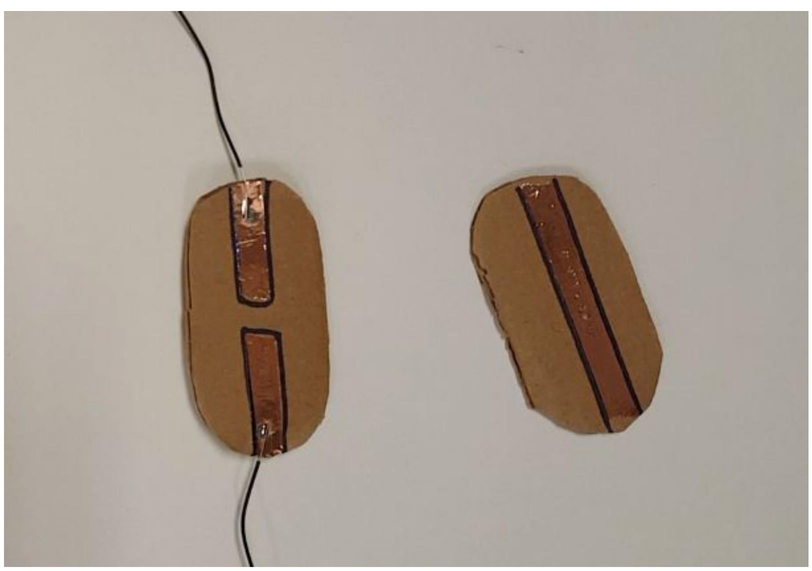


v. **Consider**:
   1. Is the vibration or audible sound intended for you or for other people, near or far?

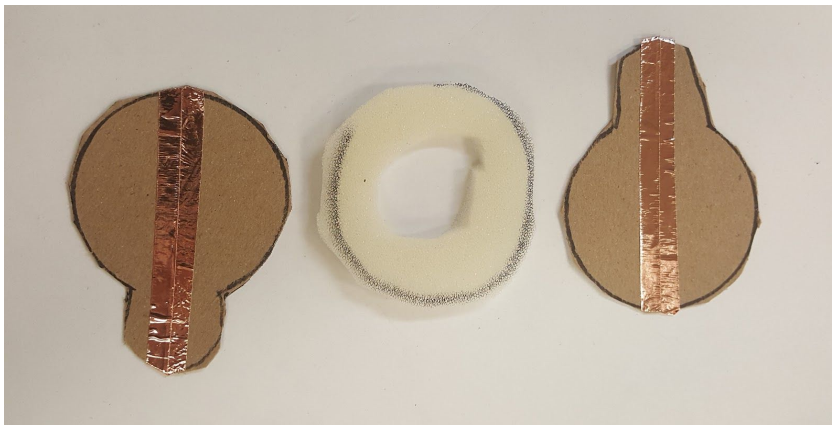
b. **Switches**: What's the interaction that triggers the sound?
   i. **Demonstrate** how a switch can be made out of any two conductive surfaces by using copper tape and wire to make a connection between GND and any pin.
   ii. You can use similar cardboard building techniques to make your switches.
   iii. Most switches have two conductive pathways that connect and disconnect.

   iv. A ***bridge*** type has two pathways that are 'bridged' by a *third* conductive surface.

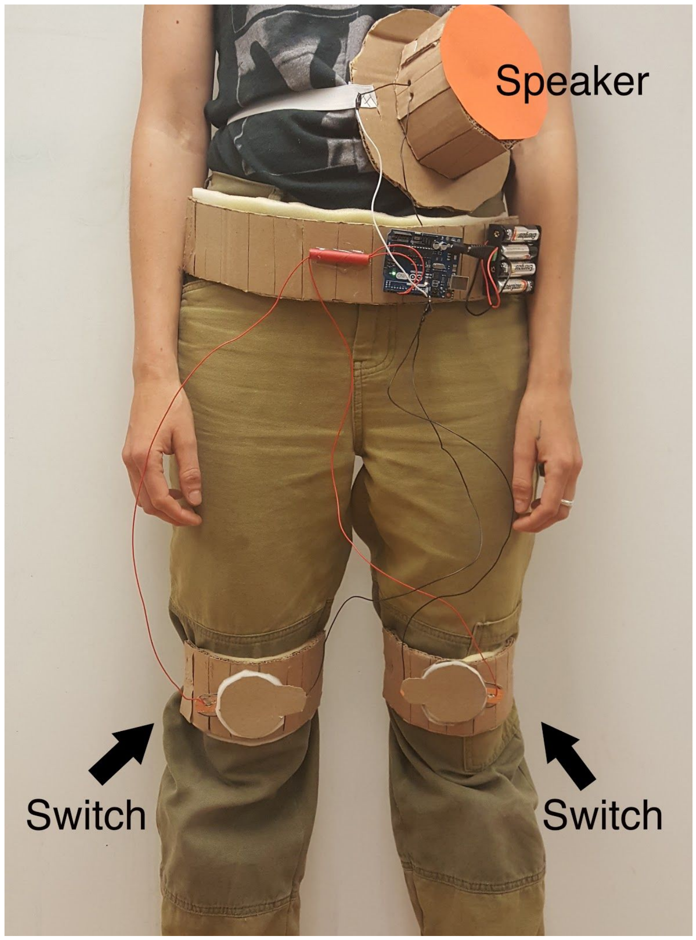


v. You can use foam to make buttons that bounce back after you press them closed.



vi. **Consider:**

1. How do the switches connect to and interact with your body, an object, or your environment?
2. How do the conductive surfaces connect and disconnect?



3. **Build guidelines:**
    a. Together, you will design a way for your future human self to send information about movement and action in the environment, using sound or vibration.
    b. Work together as a group to make one prototype that combines your ideas.
    c. We will eventually use Arduino to make choices about what we hear.
    d. You can incorporate any other available materials you know how to use.
    e. Draw a diagram of what you want to make. As you work and build on your ideas, add the following information to your schematic.
        i. You can use up to ten switches. Where are they located?
        ii. Are switches activated by interactions between more than one person or object?
        iii. Where is the exciter/speaker located?

iv. Will there be a potentiometer, photoresistor or a flex sensor used to control the speed of the notes? Or do you want solid tones?
v. How will the different parts of the sound system be connected using cardboard or other building materials? What does it look like? How is it held, attached, or worn?
vi. Draw the entire circuit into your sketch.
4. **Play!** Explore materials and construction techniques, and try making switches that can trigger sounds on the Arduino.
5. Clean up.

Session 23: Build

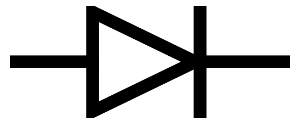
Classroom setup
The same as session 22.

1. **Build** mechanical switches and cardboard speaker structures! Test and improve.
2. Clean up.

Session 24: Soldering to Switches

Classroom setup
The same as session 23.

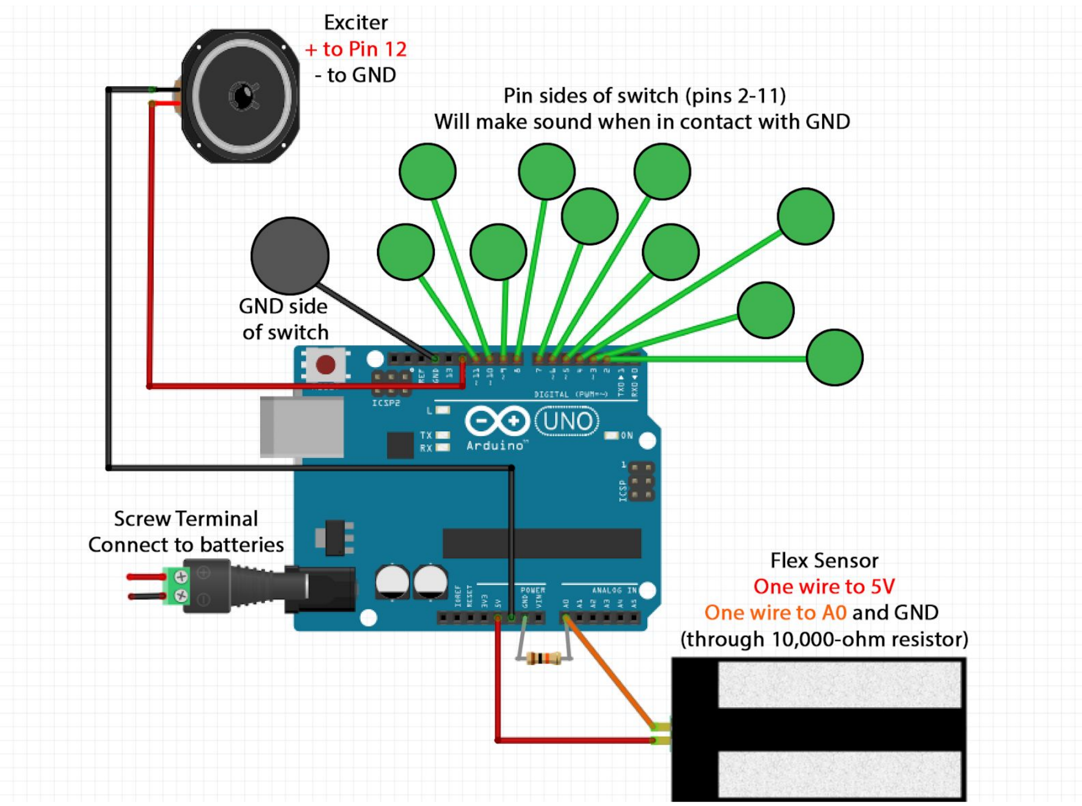
1. **Demonstrate** how to solder connections to the copper tape on home-made switches. On *each* switch, there should be two wires that lead back to the Arduino:
    a. A ground wire that connects to GND on the Arduino.
    b. A wire that connects to a digital pin between 2 and 11 on the Arduino
    c. *All of your switches can have their GND wires soldered together, but each of their second wires need to be plugged into different pins to get separate tones.*
2. **Demonstrate** how to test the switch using the *continuity* setting on a multimeter. If you touch the multimeter to each side of the switch when it's closed, it should beep or indicate a connection. The continuity symbol looks like this:



Session 25: Introduce Variable Resistors and Code

Classroom setup
The same as session 24. Laptops, USB cables. Conductive foam, vinyl, photocells, 10k resistors, copper tape.

1. **Explain**: if you want to hear a delay, you can use a ***variable resistor*** to determine how much delay there is. You can use any one of the following variable resistors:
    a. ***Potentiometers*** will enable you to specifically set the arpeggio speed, as you did in the prototype circuit.
    b. ***Flex sensors and photoresistors*** will enable you to change the speed constantly through movement or light. If you use a flex sensor or a photoresistor, you will have to incorporate a resistor at the base of the sensor like you did in the Chameleons Domain. This is illustrated in the diagram above.
    c. **You can also simply use a jumper wire to directly connect between pins A0 and 5V (fastest arpeggio), 3.3V (medium speed) or GND (slowest).**
2. **Explain:** students will connect their projects to their Arduino in the same way that they did at the beginning of the domain, but this time with their switches incorporated. With hand-made switches represented as green dots, the new circuit will look like this:



3. **Introduce** code using the Bees Code Cheat Sheet. Students can make changes to the code that affect the following:
    a. How many pins/switches - they can use up to ten different notes.
    b. If notes sound in a sequence (arpeggio) or interfere with one another
    c. If they are using flex sensors, photoresistors, potentiometers or jumper wires to control arpeggio speed.
    d. The direction of the arpeggio (tones going up or down) using boolean (true or false) logic.
    e. Which notes are triggered by which switches.
4. Continue prototyping.
5. Clean up.

Session 26: Build Prototypes

Classroom setup

The same as session 25.

1. **Check in** with each group. Some questions that might be helpful to ask:
    a. What is it? Is the soundsystem used for alerting, communicating, making music, or something else?
    b. What kind of movement activates your sound system?
    c. How is this device used in your future world and how is it helpful?
2. **Continue** building and finalizing prototypes.
3. Clean up.

Session 27: Complete Prototypes

Classroom setup
The same as session 26. Paint markers.

4. **Continue** building and finalizing prototypes.
    a. Use paint markers to add detail or color to prototypes.
    b. Use available materials to add meaning or sculptural elements to prototypes.
    c. Incorporate arduino circuit and connect to your project somehow.

Session 28: Prototype fair!

Classroom setup
Prototypes, cameras or phone cameras, Mouse Create.

1. **Prototype fair!**
    a. Create a setting in which all students can come together and share what they've made with each other.
    a. Students should explain what their prototype is and demonstrate how it works.
    b. Encourage them to ask questions about each other's projects.
2. **Review**: how to upload images to Mouse Create.
    a. Upload images of journal sketches and final product.
    b. On Mouse Create, describe your prototype. How is it used by a future human? How does it work?

Bees Code Cheat Sheet

Bees Sketch

This code lets students control a series of sounds using custom-made switches.

List of Musical Notes

This code involves using musical notes. We have to pair all the musical notes with their matching frequencies in order to use them with the Arduino. Since they are all in the code already, they don’t need to be changed and *should not* be deleted.

```
#define NOTE_B0  31
#define NOTE_C1  33
#define NOTE_CS1 35
#define NOTE_D1  37
#define NOTE_DS1 39
#define NOTE_E1  41
#define NOTE_F1  44
#define NOTE_FS1 46
#define NOTE_G1  49
#define NOTE_GS1 52
#define NOTE_A1  55
#define NOTE_AS1 58
```

Each note has a NAME (B, C#) and an OCTAVE (0 to 8, being lowest to highest). There are many more notes than this in the code, but you get the idea.

Selecting Our Specific Notes

Now that we have defined all the notes, we are selecting ten specific notes to use with our project. These ten notes can be played by closing switches connected to the ten digital pins (pins 2-11).

```
////// SELECTING OUR SPECIFIC NOTES

int Notes[] = {
  NOTE_C4, NOTE_D4, NOTE_E4, NOTE_G4, NOTE_A4,
  NOTE_C5, NOTE_D5, NOTE_E5, NOTE_G5, NOTE_A5
};
```

The notes in the base code are specially selected to make a musical progression that sounds nice. However, if you want to change them, you will have to make sure that ten notes are named in this section with commas between them, or else the program will give an error.

Booleans: Controlling the Program Options

Booleans are a special type of control that programming uses. They can only be set to **true** or **false**, and they are used to turn different parts of the code on or off.
In this case:

RISING will make the sequence of notes play from low to high if set to "true".
FALLING will make the sequence of notes play from high to low if set to "true".
(If both are set to true, the notes will play low to high AND then high to low.)
DELAYED will put space between every note if set to "true". If you disable delayed, you will get constant noise.

Play around with the combination of options to find what suits you best.

```
////// BOOLEANS: CONTROLLING THE PROGRAM "OPTIONS"

boolean rising = true;
boolean falling = true;
boolean delayed = true;
```

VOID SETUP() and LOOP()

Because we are controlling the options we want for our program with booleans, there is no need to edit the code in either setup() or loop() this time.

Sessions 29-36: Future Humans

Project Overview

Objectives
Bringing together everything you've learned to create something that synthesizes your knowledge.
Building Future Human adaptations that are unique and say something about who you are.
Designing ways for our Future Human selves to respond and react to each other or our environment.

Project Details

Students will work together to design and build Future Human adaptations, using the tools and materials of their choosing.

Session 29: Introduction

Classroom setup
Synthesis Worksheets, Mouse Create, selected building materials.

1. **Reflect** on the skills you've learned from past prototypes. What was exciting?
2. **Explain** synthesis.
    a. In Synthesis, you will build a Future Human adaptation that's informed by all of the projects that you've made, but is totally new. You'll work together to design and build something that ***synthesizes*** your knowledge and expertise!
3. **Explain: Synthesis criteria**
    a. You can't make something that already exists. You can combine things that already exist in a new way *or* invent new things.
    b. You should build something new from scratch.
    c. You can combine two types of electronic components, form any area of the program.

d. Your Future Human self must ***interact*** with other Future Humans, objects or the environment somehow.
e. You must be able to explain who your Future Human self is and where they live.

4. **Share** Synthesis Worksheet
    a. You can use the Synthesis worksheet to review all the materials you've used and skills you've learned. As a group, you can use this worksheet to help develop ideas about what you want to build.
5. Make a drawing!
6. **Build**!

Session 30-35: Build!

Classroom setup
Selected tools and materials.

1. Students will continue to build their synthesis projects. Test and improve!

Session 36: Prototype Fair

Classroom setup
Prototypes, cameras or phone cameras, Mouse Create.

1. **Prototype fair!**
    a. Create a setting in which all students can come together and share what they've made with each other.
    a. Students should explain what their prototype is and demonstrate how it works.
    b. Encourage them to ask questions about each other's projects.
2. **Upload** images to Mouse Create.
    a. Describe your prototype. What did you make?
    b. Reflect on why you made the choices you did.

APPENDIX

Electronic Circuits

Introduction

Electricity is energy created by the movement of electrons through a circuit. *Electrons* are the part of atoms that make them stick together to form molecules. *Electricity* is a form of energy we get from the movement of electrons.

A battery contains two materials; one has too many electrons and the other has too few. This makes one side *positively charged* (+) and the other *negatively charged* (-). It's a force of nature for the electrons to seek equilibrium, so if we connect a path from + to - the electrons will flow through it and make electricity. The path from + to - is called a *circuit*.

A circuit is a pathway for electrons to move from a negatively charged material to a positively charged material. Circuits are designed to make electrons flow through electrical components to do some type of job, like emitting light or making sound, and absorb a specific amount of voltage in order to do this. The amount of voltage they need to function will be designated.

A circuit is made of *conductive* materials such as wire or copper that have a crystalline structure that allows electrons to easily move through them. Insulators, such as plastic, surround wires in order to keep electricity flowing only where we want it to. Conductors let electricity flow, and insulators resist it.

A *short circuit* happens when the pathway of the circuit bypasses the components it is meant to go through and circles back to the battery too soon. Because it didn’t pass through the components in the circuit, it will have excess heat and could start a fire. Short circuits can be avoided by *insulating* exposed wires and conductive material with electrical tape or hot glue, and by avoiding stray wires that could create unwanted pathways for electricity to flow.

Soldering tools can be used to create more permanent connections between conductive materials such as wire and the *terminals* on electrical *components*. Solder is kind of like glue for electronics, except that solder is *conductive* and should be used precisely and only where needed. It allows us to create a conductive pathway for electricity to flow.

Typically, circuits are prototyped using breadboards and jumpers*,* which are tools for making temporary circuits and trying things out. Solder tools can be used to create more permanent connections between conductive materials such as wire and the *terminals* on electrical *components*. Solder is kind of like glue for electronics, except that solder is *conductive* and should be used precisely and only where needed. It allows us to create a conductive pathway for electricity to flow.

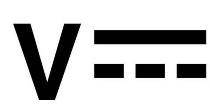
Electrical Signals and the Body
Beam projects typically only involve low voltages and low currents that are not dangerous. It is perfectly safe, for example, to touch the + and - poles of a battery with one hand. However, it is wise to avoid running electrical signals through the body and especially from one side of the body to the other.

Common Tools and Materials

Multimeter
Multimeters are prototyping tools that can be used to test circuits in various ways.

To test for *continuity* in a circuit or to see if an LED has burned out, hold the metal probes to the circuit and set the multimeter to the *continuity* setting. It will beep to indicate if there is a connection between two points.

To test to see how much voltage is left on a battery, set to *20 Volts DC* and touch the probes



to either side of the battery. The remaining voltage will be displayed on the screen of the multimeter. The DC voltage symbol is to the right.

Soldering Iron

Soldering irons are used to create more permanent circuits.

Soldering Setup

- Wear safety glasses
- Use in a well-ventilated area
- Maximum of 1 soldering iron per 2 sq. ft.
- Do not tangle the cables

Soldering Procedures

- The metal tip of the iron is typically heated to approximately 700ºF
- Don’t touch the tip or past the grip.
- Always put it back in the holder.
- Never pass or put the soldering iron on the table.
- Do not melt solder for no reason
- Use "helping hand" tools to secure wires in place when soldering
- Wires will become very hot during and after soldering
- Soldering irons must be turned off after use

Wire Snips are for cutting wires and snipping excess solder.

Wire Strippers are for stripping plastic coating away from wires before soldering. Match the gauge of the wire you're stripping with the correct numbered hole on the wire strippers.

Pliers are for holding wires. Theannot be used to cut or strip wire. Less commonly used in electronics.

Helping hands are alligator clips that hold wires in place while soldering.

Wire

Wire is a conductive material, like copper, encased in an insulating material, like rubber or plastic. There are two types of wire: stranded and solid. Stranded wire is flexible and is good for moving parts, but solid wire is much easier to insert into small holes. The color of wire is for organizational purposes only.

Copper tape is conductive, so it can be used to create circuits.

Battery

A battery is a container that holds a negatively charged material and a positively charged material. Negatively charged means the material has too many electrons. Positively charged means the material has too few electrons. When the negative and positive poles of the battery are connected through a circuit, electrons flow until the system reaches equilibrium and the battery "dies."

Voltage is the force or pressure of electricity in a circuit, coming from a battery. Different parts of a circuit use up different amounts of voltage. Amperage or current is the volume or amount of electricity in a circuit. The current is the same everywhere in a circuit.

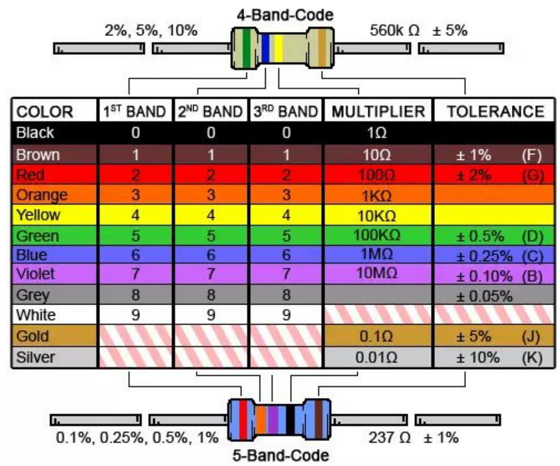
Projects will typically use either AA (1.5v), coin cell (3v), or 9 volt batteries. Batteries are incorporated into soldered circuits using their associated battery clips - never solder directly to a battery, and always remove the battery from a clip before soldering.

It is safest to store batteries in their packaging or with tape over one or both terminals. If the batteries short circuit while in storage, they will overheat and could start a fire.

Components

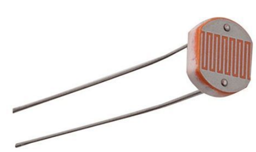
Resistor
A resistor is a component that resists the flow of electricity and reduces the overall voltage of the circuit. Resistors are used to protect components in cases where the amount of voltage needed by a component is less than the amount stored in a battery. A resistor contains a very long piece of wound conductive material that dissipates electricity as heat.

Resistors are labeled with color bands that indicate their resistance as measured in Ohms. The first two bands indicate the first and second numbers of the value, and the third band indicates how many zeros follow the first two numbers. Typical electronics projects will use resistors between 220 Ohms and 10k Ohms. Ohm's Law can be used to determine what value of resistor should be used to protect a circuit when it is not given.

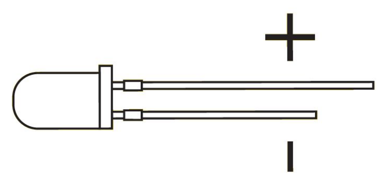


COLOR	1ST BAND	2ND BAND	3RD BAND	MULTIPLIER	TOLERANCE
Black	0	0	0	1Ω	
Brown	1	1	1	10Ω	± 1% (F)
Red	2	2	2	100Ω	± 2% (G)
Orange	3	3	3	1KΩ	
Yellow	4	4	4	10KΩ	
Green	5	5	5	100KΩ	± 0.5% (D)
Blue	6	6	6	1MΩ	± 0.25% (C)
Violet	7	7	7	10MΩ	± 0.10% (B)
Grey	8	8	8		± 0.05%
White	9	9	9		
Gold				0.1Ω	± 5% (J)
Silver				0.01Ω	± 10% (K)

Photoresistor
A photoresistor is a form of variable resistor that restricts the flow of electricity in response to light.



LED
LED stands for Light Emitting Diode. A diode is an electrical component that controls the direction of the flow of electrons. Because of this, LEDs are *polarized* and have a positive and negative terminal. The negative terminal of the LED must be installed in a circuit facing toward the negative terminal of the battery, otherwise the LED won't emit any light and the circuit will not work.



The LED uses a small amount of electricity and emits it as light, typically between 2 and 4 volts. The LED can only handle a small amount of electricity, so if we're using a battery that has more than 3 volts, we need to add a resistor to absorb some of the excess or else it will burn out. When it burns out, a small piece of metal melts inside it. The entire circuit will stop working because electricity can no longer flow.

A servo motor is a motor that can be precisely controlled using an Arduino. Servos typically only spin in 180° semicircles.

The red wire is always power, the brown wire is always ground, and the yellow wire is always data.

It's recommended that you attach the servo heads to the motors using a mini screwdriver before building projects.

Exciters are speakers that don't have a cone, or vibrating surface, attached. They emit audible sound when attached to paper, cardboard, or something that is flexible and can vibrate.

Physical Computing and Arduino Basics

What is physical computing?
Physical computing is the practice of controlling the behavior of electrical circuits using sensor-controlled computers and microcontrollers.

What is a computer?
A computer is a machine that physically organizes patterns of electricity to represent information. This information is measured in a unit named "bits." One bit is represented by a single electrical charge in the computer's processor microchip. For reference, 8 bits = 1 byte, and 8,000,000,000 bits = 1 gigabyte.

When we type on a computer, we are causing physical changes inside the computer's hardware. For example, typing a single letter in a plain text document creates 1 byte of information, which is represented by 8 electrical charges. If we type a second letter and save the file, the file size becomes 2 bytes. For every letter we type, more electrical switches are turned on inside the computer.

What is a microcontroller?

A microcontroller is like a tiny computer that has input and output pins that can be connected with an electronic circuit on a breadboard. We can control the circuit using real-world sources like sound, light, and temperature using sensors, which would be considered *inputs*.

Writing code

Arduinos control the behavior of electrical circuits using code. Code is a set of instructions written in language a computer can understand. For example, "turn this light on and off every second" or "turn this light on only when it's dark outside."

You use the Arduino IDE to create a *sketch* (a little block of computer code) that you upload to the arduino board through USB in order to tells the board what to do.

The programming language uses words and numbers to organize logical instructions for how our circuit will behave. The Arduino has a built-in program called an assembler that knows the Arduino language and translates our code into electronic behavior.

When something isn't working in a program it's called a bug. Finding, identifying and fixing the problem is called *debugging*. Grace Murray Hopper coined this phrase when she traced the cause of a malfunction in her computer to an actual moth that had gotten into the hardware. A bug may be a *syntax error* (a problem with how you wrote your code) or a *logic error* (a problem with the flow of your application).

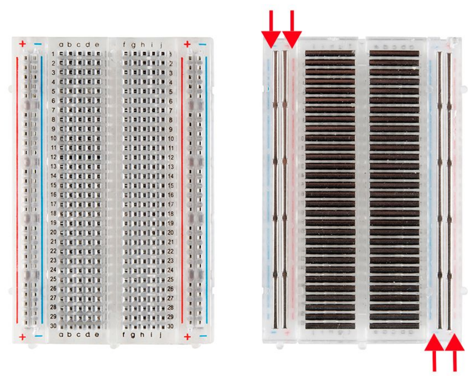
Arduino Code Editing Software

There are two ways to write and send code to an Arduino:

- Arduino Web Editor. Recommended because it saves your code online so you can access it from any computer. You must create an account and install the Arduino drivers. We recommend that you create an account with a simple password for each class to be shared by the students.
- Arduino IDE (Integrated Desktop Environment). This is a downloadable app that does the same thing as the Web Editor, but it does not save your code online.

Arduino Tools and Materials

Breadboard
Breadboards are another tool for prototyping circuits. They contain columns and rows of input pins that are used for connecting jumper wires and components. The photo on the right shows the conductive paths going along the two columns on either side and along the rows in the center of the breadboard.



Jumper cables
Jumper cables plug into arduinos and breadboards and can be used to prototype and make temporary circuits.

Sensors
A sensor is an electronic component that either varies its electrical resistance or generates a voltage based on physical conditions of the real world. For example a photoresistor varies its resistance based on exposure to light, and a piezo mic generates a voltage when pressure is applied to it.

In most Arduino circuits, you will be connecting a sensor into one of the Analog Input pins labeled A0-A5. The Analog Input pins measure voltage between 0V and 5V and translate that electricity into a number between 0 and 1023. We can use the changes in those sensor readings to control the behavior of our circuit.

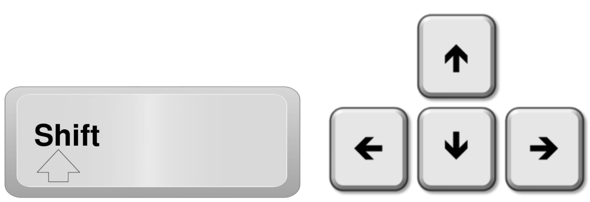
Typing Skills for Coding

Most students know how to type e-mails or write papers, but typing for coding requires certain additional skills.

Text Selection with the Mouse

- Clicking and dragging selects individual characters and lines
- Double-clicking a word selects one word
- Triple-clicking a word selects the entire line
- Quadruple-clicking a word selects the entire page

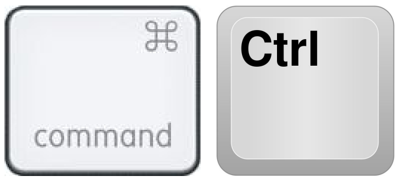
Text Selection with the Keyboard



Shift + Arrow Keys

-Hold down *shift* while moving the cursor with the *arrow keys* to select characters and lines
-Also hold down the command key to select entire lines or the whole page.

Copy/Cut/Paste



(On a Mac, use the command key. On a PC, Chromebook, or any other computer, use control.)

Command + C to copy
Command + V to paste
Command + X to cut
Command + Z to undo
Command + A to select all of the text

Fabrication Tools

Cardboard cutters are for cutting cardboard only. They can cut skin, so use them with care.

Cardboard perforators can be used to score cardboard without cutting all the way through.

Spring clamps can be used to hold cardboard in place while cutting with cardboard cutters. Make sure to clamp so the area you're cutting is *off* the table. Two clamps are better than one.

Hot glue guns are useful for quickly joining pieces of wood together, or for joining wood, plastic, and other materials.

- Glue guns are made of plastic and look like a toy, but they are actually serious tools.
- Glue guns tips can cause burns.
- Hot glue remains hot for several seconds after application.
- Glue guns must be unplugged after use.

Long Reach Hole Punch can punch holes in paper, cardboard, popsicle sticks, and many other materials. There are multiple settings on the top, but you will only really need to use the 3/16" setting. You can change the settings by moving the turquoise button. There is a moveable measuring tool on the bottom of the hole punch if you want to be more precise.

Staple pliers can be used to staple paper, foam, cardboard, and felt. They require a specific kind of staple, which can be loaded from the top by pulling the “tail” on the back of the handle. They get jammed sometimes - use pliers to pull the staples out of them.