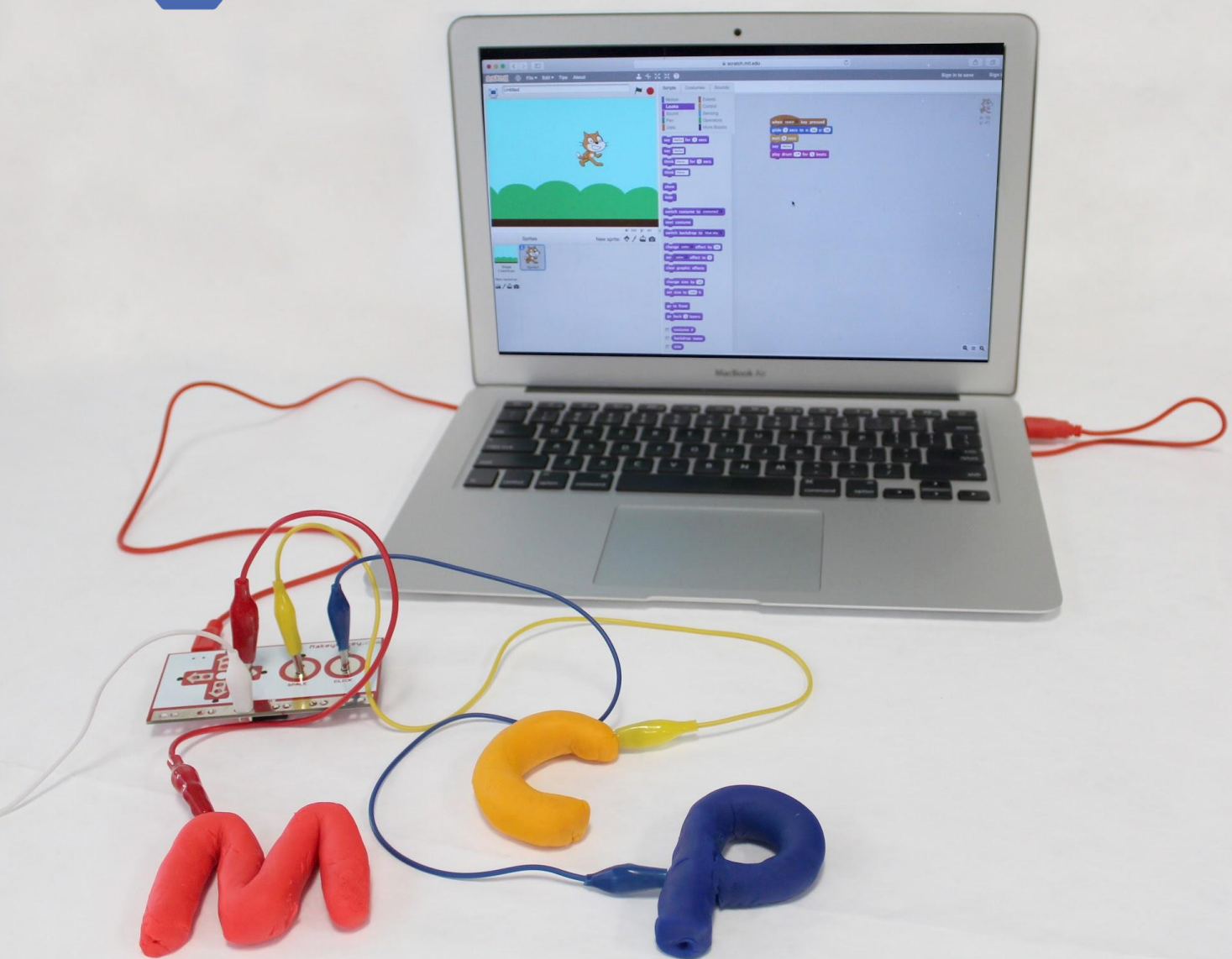


MAKE

CODE

PLAY



A PROJECT GUIDE TO GAME DESIGN BEYOND THE SCREEN

BY VEENA VASUDEVAN, YASMIN KAFAI, AND SARI WIDMAN

# TABLE OF CONTENTS

**I - INTRODUCTION** ..... 1

**II- GETTING STARTED**

WHAT IS THIS GUIDE AND HOW DO I USE IT? ..... 3

THINGS TO KEEP IN MIND WHILE USING THIS GUIDE ..... 4

**III - THE TOOLS YOU'LL NEED & HOW TO USE THEM: AN INTRODUCTION TO SCRATCH & MAKEY MAKEY**

MAKE: INTRO TO MAKEY MAKEY ..... 6-7

CODE: INTRO TO SCRATCH ..... 8-9

MAKE + CODE: INTRODUCING SCRATCH & MAKEY MAKEY TO YOUR STUDENTS ..... 10

PLAY: SETTING UP THE ARCADE ..... 11-12

**IV - PROJECTS**

Touchpads and Joysticks ..... 14 -19

    Overview ..... 14

    Suggested Materials ..... 15

    Curriculum ..... 16-17

    What Students Had To Say ..... 18

    Case Studies ..... 19

Board Games + ..... 20-24

    Overview ..... 20

    Suggested Materials ..... 21

    Curriculum ..... 22

    What Students Had To Say ..... 23

    Case Studies ..... 24

Wearable Controllers ..... 25-30

    Overview ..... 25

    Suggested Materials ..... 26

    Curriculum ..... 27-28

    What Students Had To Say ..... 29

    Case Studies ..... 30

**V- APPENDIX**

RESOURCES ..... 32-33

GLOSSARY ..... 34

CONTACTING US AND SHARING THE GUIDE ..... 35

## GAME DESIGN BEYOND THE SCREEN

More than 20 years ago, the first game design project took place in an elementary school in Boston. A class of fourth graders and their teacher transformed a traditional math class into a space to develop their own video games using the programming language `{Logo}`. Their goal was to design games that would teach fractions to younger students at their school. Besides learning programming and game design, students had to “advertise” their respective games to their younger counterparts. In doing so they likewise learned about product design, as well as writing advertisements and stories. The project was an unprecedented success with students and teachers alike, both of whom expressed enthusiasm for this integrated way to learn about mathematics content, computer coding, peer collaboration, and creativity all at once. Since this initial offering in the mid-1990s, hundreds of studies have demonstrated the successes of learning through making games, but most of these game designs have been limited to the computer screen.

Today we have expanded the game design project beyond the screen. Here students not only program or remix games on the screen but also engage in the making of physical controllers. They construct joysticks and touchpads that control the action on the screen. They augment the play of traditional board games with digital dice or cards. New construction kits like MaKey MaKey have made the design of such physical devices accessible to beginning programmers. Like with the original game design project, students learn about coding, graphic design, and writing stories. But through MaKey MaKey, they also learn about circuit design, conductivity, and crafting with various materials as they design and make their own controllers and digital extensions. More importantly, they learn firsthand how the technology they use every day is actually being made.



Photo Darryl Moran

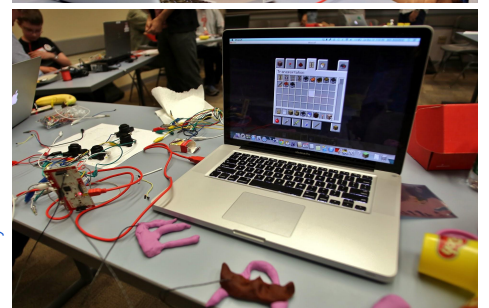


Photo Darryl Moran

## WHO WE ARE

Yasmin Kafai and Veena Vasudevan designed and implemented all of the projects described in this guide with Peter Endriss and his middle school students at the Penn Alexander School. Some of the projects were also tested during Philadelphia Scratch Day 2013, and with high school students from the Science Leadership Academy. Richard Davis and Eunyoung Lee participated in the design of touchpad classroom projects. Radium Lee Yang developed the prototype for the wearable controller design.

Veena Vasudevan drafted the first version of this guide, with additional content provided by Yasmin Kafai and Sari Widman. Sari Widman was responsible for the final layout design, based on an earlier version developed by Emily Dieckmeyer. Copy editing for this guide was done by Quinn Burke and Sari Widman. Photos were provided by Darryl Moran, Veena Vasudevan, and Yasmin Kafai, with additional images for the guide by Sari Widman.

The work described here was made possible through a National Science Foundation grant CDI# 1027736 to Yasmin Kafai. While we are grateful for this support, we need to add the customary disclaimer that any opinions, findings, conclusions, and recommendations expressed in this guide are entirely ours and do not reflect the views of the National Science Foundation or the University of Pennsylvania.

# GETTING STARTED

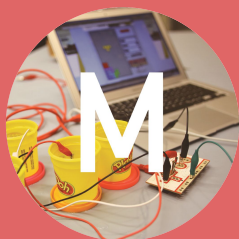
In this section, we'll show you how to move through this guide, and what to expect from it. We'll also review some concepts to keep in mind when considering how to bring these projects into your classroom.



## WHAT IS THIS GUIDE AND HOW DO I USE IT?

In MAKE CODE PLAY we describe three projects that expand game design beyond the screen. The projects range in difficulty level from starter to intermediate. Some projects are collaborative, while in others, students preferred to individually make their own game designs. Over the last three years we have piloted these projects with over a hundred elementary, middle, and high school students and their teachers. We have used materials that are commonly found in classrooms, schools, or homes, so they can be easily adapted in classroom or workshops. Our construction kit MaKey MaKey can be purchased and reused for the projects, and a programming tool, Scratch that is accessible and freely available around the world.

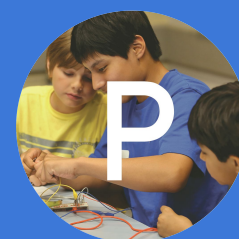
Three ideas guide each project: “MAKE,” “CODE,” and “PLAY.”



**MAKE:** In each project the crafting of a physical controller, interface, or board for the game is a key activity. We use MaKey MaKey to connect these controllers and boards to a Scratch game on the computer screen.



**CODE:** In each project students write or remix the code of games designed in Scratch. The code for the sample games can be found in the Scratch studio ([scratch.mit.edu](https://scratch.mit.edu)). Sometimes students re-write some lines of code that comprise these existing games; in other cases they might modify whole procedures.



**PLAY:** In each project students {playtest} each other’s games, among themselves or with other students from other classes. We conclude each project with an “arcade” in which students demo the final versions of their game and controller designs.

### AS YOU MOVE THROUGH THIS GUIDE WE WILL:

- Introduce you to two primary tools, Scratch and MaKey MaKey; you’ll need each for the projects
- Show you how to use them, where to get them, and how to introduce them to your students
- Outline three design projects, including learning goals and timelines, case studies, and activities
- Describe how to set up an “arcade” for playtesting game designs and controllers
- Provide resources for you to explore, troubleshoot, and expand upon your knowledge

### {RESOURCE CALLOUT}:

As you move through this guide you may find yourself wondering about a particular word, phrase, or topic. If the source of confusion has {brackets} around it, you’re in luck! These brackets indicate that you can find more information in the Resources section, from expanded definitions to suggested readings.



## THINGS TO KEEP IN MIND WHILE USING THIS GUIDE

### ORGANIZING AND SELECTING MATERIALS IS IMPORTANT:

Providing students with materials that are both engaging and well organized will go a long way towards creating a positive and successful project experience. Simple materials are great, and there are many inexpensive options including recycled materials and things you can find around your house and classroom. We recommend encouraging students to bring materials from their own homes that they would like to use.

### DESIGNING GAMES AND CONTROLLERS IS AN ITERATIVE PROCESS:

Students have an idea, build it, test and revise it, and try it out again. There is a lot of tinkering and fine-tuning involved in getting everything, especially the connection between the game on the screen and the physical controllers to work.

### PROVIDING TIME AND SUPPORT IS KEY:

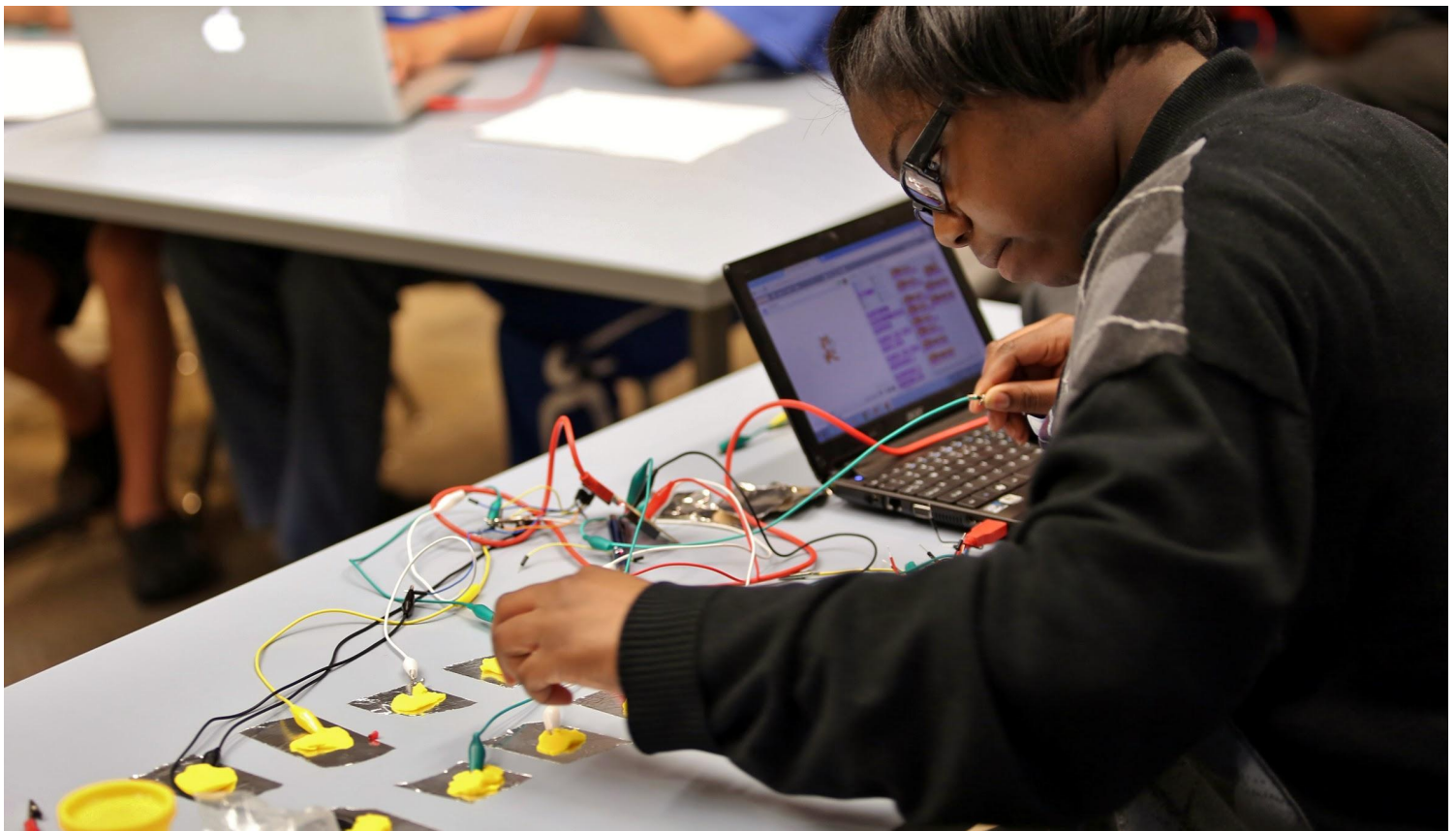
Lots of things don't work the first time when students make the connections between physical and screen designs. Encourage students to talk and discuss their design challenges so that they can learn from each others.

### FAILURE IS PART OF THE PROCESS:

Keep in mind that students will have points of frustration while working on their projects. Remind students before they begin that sometimes things won't work and that's ok. Encountering difficulties is part of the learning process.

### SHARING DESIGNS + GETTING FEEDBACK IS PART OF THE PROCESS:

We have built many feedback mechanisms into project activities such as play testing each other's designs and setting up the arcade at the project end. Students greatly value the feedback and use it to improve their designs.



# TOOLS YOU'LL NEED & HOW TO USE THEM: AN INTRODUCTION TO SCRATCH & MAKEY MAKEY

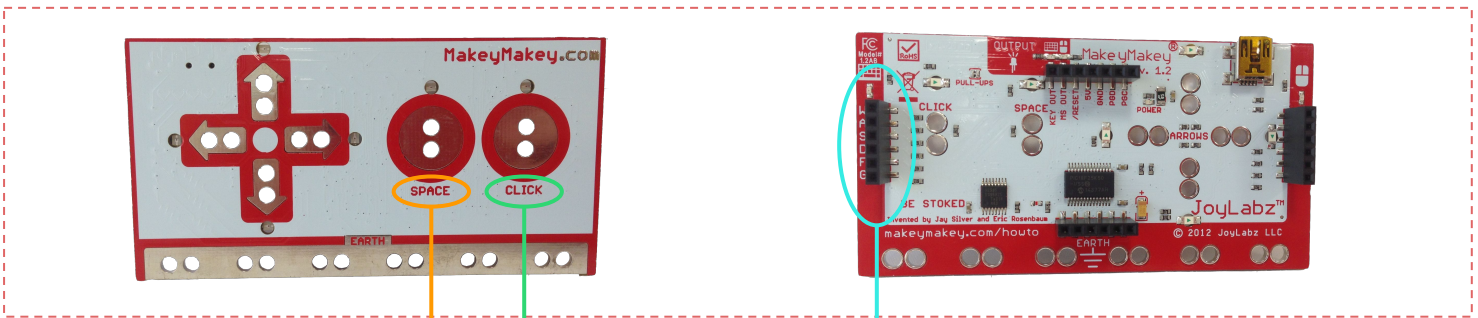
In this section, using the framework of “Make”, “Code”, and “Play”, we’ll guide you through the basics of Scratch, Makey Makey, and setting up the “Arcade”. You’ll be able to apply the information in this section to each of the projects outlined in the guide.

Let’s get started!

# MAKE: INTRO TO MAKEY MAKEY

## WHAT IS MAKEY MAKEY?

The very short answer is that the MaKey MaKey is a kit that allows you to connect any **{conductive}** material to your computer, and use that conductive material to control what's happening on the screen. In many ways, the design of the MaKey MaKey mirrors the look of an old-school Nintendo controller, and looking at it as a game controller may be a helpful way of thinking about it.



The slightly longer answer is that the MaKey MaKey is a modified **{Arduino}** board with some pre-programmed functions. If you look at the front of your MaKey MaKey you will see connection points that correspond to actions on your computer keyboard, like the **spacebar** and **mouse clicks**. If you flip the MaKey MaKey around, the back reveals more connections that allow for even more control. These **connections** are pre-programmed and link to particular letter keys on your keyboard. These connections in the back give you even more room to play and experiment with your controller.

(We say “pre-programmed” because with an Arduino board, inputs can be programmed to perform different actions. For example, you could use Arduino code to change a pin or “connection”, from a “K” key to a “5” key. You can reprogram some pins on the MaKey in this way if you choose to get more advanced in future projects; however you will not be doing this for any of the projects in this guide.)

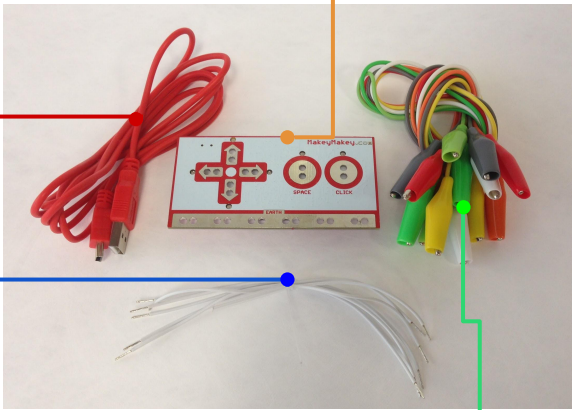
YOUR MAKEY MAKEY WILL COME WITH:

**Makey Makey Board**

**USB Cable**

**Jumper Wires**

**Alligator Clips**



## WHAT CAN YOU DO WITH YOUR MAKEY MAKEY?

All kinds of things! The MaKey MaKey is what will help you bring your game design beyond the screen. Students will program and design games in Scratch, and then create physical controllers or game boards using conductive materials. You will then use the MaKey MaKey to connect your game controller to the computer, allowing others to use the controllers that you have made in order to play the Scratch-designed games. The MaKey MaKey is your key to bridging the computational world and the physical world.

**Watch this video of the MaKey MaKey in action to get a sense of how it works and what it can do:**

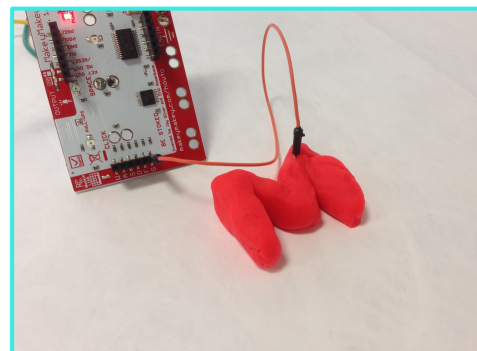
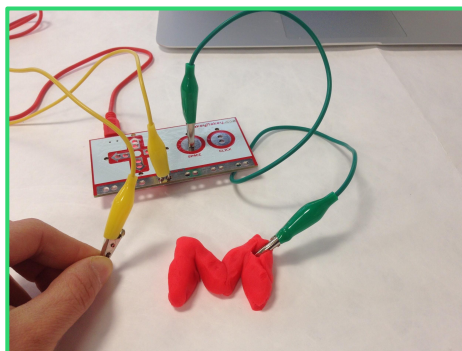
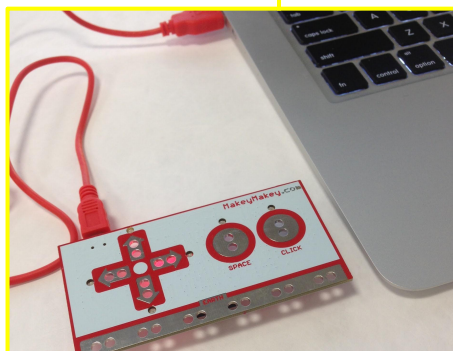
<https://youtu.be/rfQqh7iCc0U>



## MAKE: INTRO TO MAKEY MAKEY

### HOW DO I SET IT UP?

When setting up the MaKey MaKey, you'll need to connect it to 3 things: the computer, your conductive material, and yourself (or whoever is playing). First, the MaKey MaKey plugs into the computer using a simple **mini-USB to USB cord**.



Next, you'll connect your conductive material, in this case a PlayDoh "M". Using one of the **alligator clips** that comes with the kit, you'll attach an end to one of the connection points on the front of the board (either "Space", "Click", "Up", "Down", "Left", or "Right"), and the other end to the "M".

Lastly, you'll take a second alligator clip, clipping an end to the bottom of the board where it says **"Earth"**. You (or whoever is playing) will hold the other end of this alligator clip, **{grounding}** our circuit. Now you're ready to use your Makey Makey.

Let's say the "M" is connected to "Space". When the user holds the grounding wire in one hand and touches the "M" with the other, your computer will react exactly as if the computer's space bar has been pressed.

You can simultaneously connect as many materials to the MaKey MaKey as there are inputs. Once you get the hang of the MaKey you can experiment with this for more complex games. When you are ready to use the connections in the back, you can use the small **connection wires** that come with the kit.

### HOW DO I GET IT?

You can get your MaKey MaKey from the MaKey Shop (<http://shop.makeymakey.com/>). There are two options, the MaKey MaKey Classic, and the MaKey MaKey Go. The MaKey Go, while less expensive and more portable, only has one input, limiting usability significantly. MaKey MaKey can also be purchased from other outlets, like Amazon.com.

### OPPORTUNITIES FOR LEARNING

**Exploring {Conductivity} of Materials:** Students can try hooking up various objects to the MaKey MaKey and finding out through trial and error which are conductive. This can be followed by a discussion of the materiality of these objects, and the distinction between conductive and nonconductive materials.

**Making Simple Circuits:** Students will learn how to connect simple circuits between the MaKey MaKey and an object like the banana or an LED with the alligator clips. This can be followed by a discussion of where else simple circuits are used, for instance in light switches.

**Learning About Grounding:** One of the key challenges is that the MaKey MaKey needs to be grounded.

### RESOURCES FOR TROUBLESHOOTING AND BEYOND

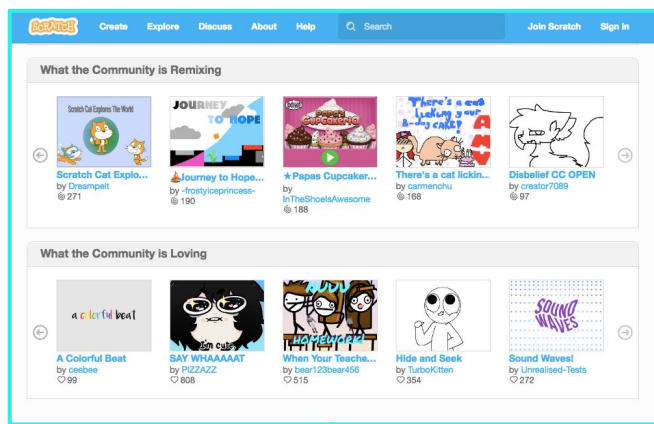
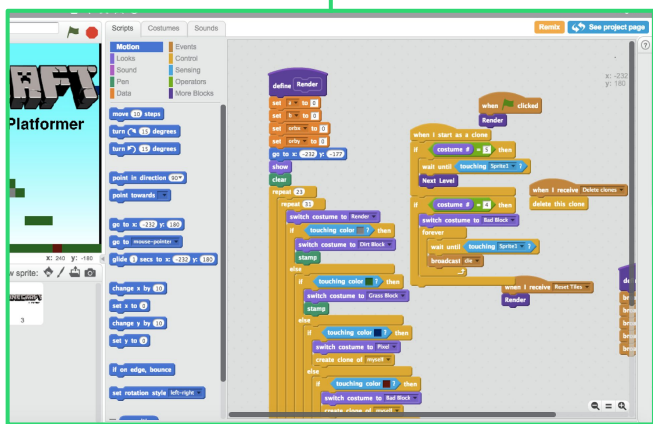
Want to explore more? See the APPENDIX for more resources to help you troubleshoot, and get inspired. You'll also find more in depth explanations of **Arduino**, **conductivity**, and **grounding**.



# CODE: INTRO TO SCRATCH

## WHAT IS SCRATCH?

Scratch is a visual programming language developed by a team at the MIT Media Lab and other researchers, among them Yasmin Kafai, to introduce children to coding. It uses a **“block” structure**, allowing students to code by snapping together blocks (not unlike Legos or puzzle pieces). These connected blocks serve as coding scripts which generate actions on the Scratch stage.

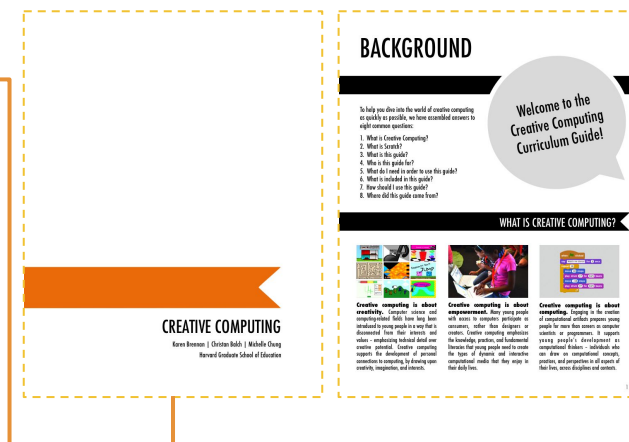


## WHAT CAN I DO WITH IT?

Scratch is such a useful tool because it has a low barrier of entry, but a high ceiling in terms of what students can create and how advanced they can get. Students can create games, animations, and more using image and sound. Another prominent part of Scratch is the **Scratch Community**. Users can share their creations, and explore the many many projects generated by other users. They can also “remix” these projects, editing the code to generate entirely new projects. The process of learning through remixing plays an important role in the projects described in this guide.

In this guide, we do not provide an in depth introduction to programming with Scratch. Download the **Creative Computing Guide** for a comprehensive look at using Scratch in your classroom. There is are two guides you can utilize, one for educators, and one for learners:

<http://scratched.gse.harvard.edu/guide/index.html>



## HOW DO I SET IT UP?

Ask students to navigate to [scratch.mit.edu](http://scratch.mit.edu) in their web browser. For new users, they can create a new account; have them click on the “sign-in” link on the right hand side of the screen. Keep in mind that minors will need a parent or guardian’s email address to create an account.

## HOW DO I GET IT?

[scratch.mit.edu](http://scratch.mit.edu)





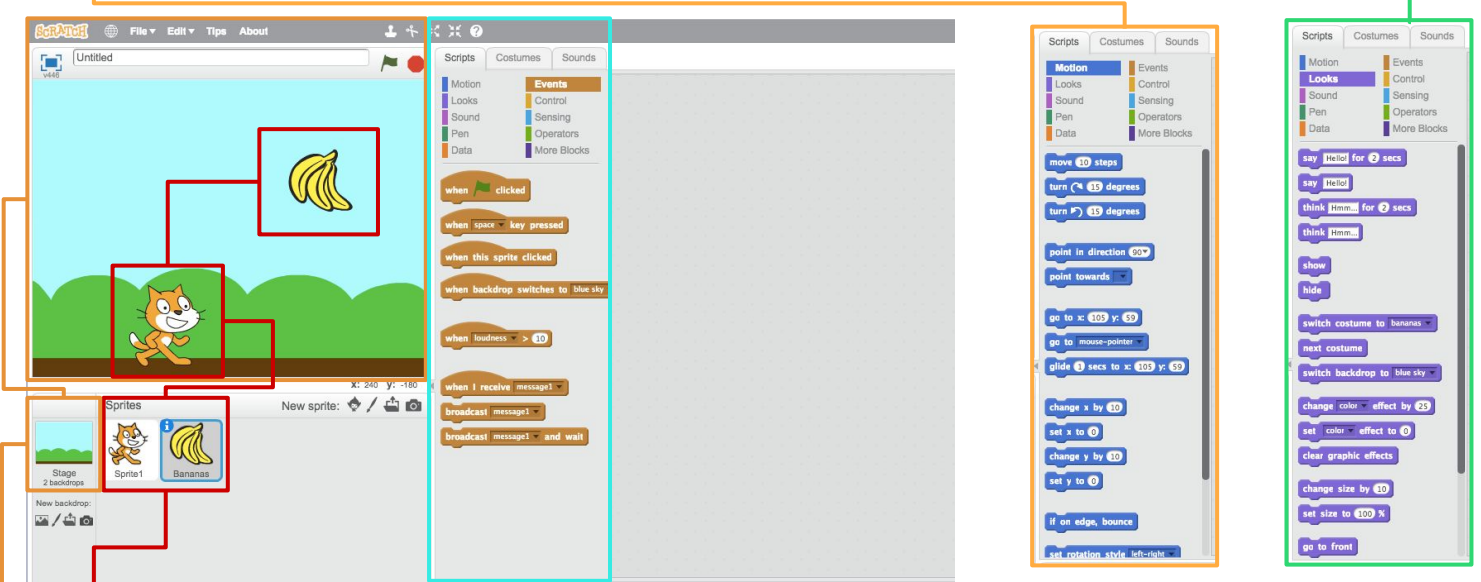
# CODE: INTRO TO SCRATCH

## OPPORTUNITIES FOR LEARNING

### Defining Basic Elements

To orient your students to some of the elements they'll be using in Scratch, have them click on *Create* and point out the stage, the sprite and some of the basic scripts. We recommend starting with: Sprites, Motion, Looks, and the Event "When Green Flag Clicked". Here is some language you can use when defining these elements for your students:

- Looks:** This is how you change the appearance of your sprite and the basis of animation and graphic effects.
- Motions:** These blocks will help you move your sprite around. Try moving the cat sprite by clicking *Motion* and the Script "Move 10 Steps". What happens?



- Sprites:** A sprite is like an actor. You can "direct" them to follow your commands (code). You can ask the sprite to speak, to move, and to play sounds.
- Stage:** This is where all the activity happens. Your sprites move around and do things on the stage.
- Events:** The most important is "When Green Flag Clicked" – it gets your program started.

**Remixing** Many student participants in our game design projects have little to no prior programming experience. Remixing existing games is a great way to introduce beginners to programming, so they can start out learning to read and understand code before writing their own. First steps can be as simple as changing the color of sprites or changing timing. More advanced steps can be adding new code to the program, changing both its look and its functionality.

Finding sample code for your students to remix is easy. Go to the Scratch site and type in the name of a popular game such as "Pong", "Mario Brothers" or Flappy Birds". You will see a large selection of games that other Scratchers have created. Games will vary in complexity in terms of code and graphics. Select one that is appropriate for your students' programming level and use the code and graphics as starter material for game designs. You can also get great ideas for other game elements from the games posted on Scratch and select code samples rather than using the whole game.

### RESOURCES FOR TROUBLESHOOTING AND BEYOND

The *Creative Computing Guide* will serve as your introduction to Scratch basics. For complete beginners, start with unit 0. As you gain familiarity with basic functionality, check out unit 4 which explains how to design games in Scratch.



## INTRODUCING SCRATCH & MAKEY MAKEY TO YOUR STUDENTS

Use the steps below to introduce Scratch and Makey Makey to your students at the beginning of each project.

### WONDERING HOW TO GET STARTED? TRY INTRODUCING THE PROJECT TO YOUR STUDENTS THIS WAY:

A good question for starting the project is to ask students what games they know and what games are their favorites to play. Collect all their answers on the blackboard. Do these games have physical components (controllers, dice, etc)? Do they have digital components (computer, game console, etc)? Both?

You can also ask them if they know what “programming” or “coding” is, and talk about the concept of coding in relationship to games that students have mentioned. (A great example that someone is bound to bring up is Minecraft.)

Once you have students thinking about this, you can introduce the design project and the tools.:

“We are going to be starting a design project today. Your goal will be to design our own video games and make your own controllers to play the games. We’ll be using two new tools to do this, Scratch and MaKey MaKey. “

### OVERVIEW OF SCRATCH AND MAKEY MAKEY

To introduce Scratch and MaKey MaKey use these short videos:

Scratch: <https://youtu.be/-SjuiawRMU4>

Makey Makey: <https://youtu.be/rfQqh7iCcOU>

After seeing the MaKey MaKey video, ask students what their favorite design were and why. Then give students a piece of paper and ask them to draw their own designs.

You can also ask students to go to the online Scratch site and direct them to a selection of games you have set up for them in Scratch studio. (See Unit 0 in the *Creative Computing Guide* for more about setting up a studio). Or, ask students to search the Scratch site by typing in names of games they know and seeing whether other Scratch members have programmed and posted a version on the site.

At the end of this first game play session, provide students with some time to share their first impressions. You can ask them to do this in small groups with other students, or you can do this as a whole class activity. Here are some questions that you can ask to get the discussion started: What did they like about the games they played? What did they not like? Did they check out the code? What kind of controllers would they make?



Photo Darryl Moran



Photo Darryl Moran



## PLAY: SETTING UP THE ARCADE

In this section we'll outline how to set up an "arcade" for students to play & test games. We are not just interested in having students learn how to make and code but also to collaborate and play in meaningful ways. You can use this model for the arcade for each of the projects outlined in this guide.

### PLAYTESTING IN CLASS

We ask students throughout the workshop to playtest their games and controllers in small groups to get feedback on the design and functionality from someone other than themselves or their teacher. Often students will do this on their own, but sometimes it helps to set up a special time to do this. Playtesting is an important part of the design process itself and shouldn't wait until the end. It's good practice to include it from the beginning when designing any software or hardware. These playtesting sessions can be short, just 5-10 minutes, and we often conclude them with a whole class discussion where we ask students to reflect on what they have learned as designers and users.

### MODELING PLAYTESTING

In preparation for the arcade, it also helps to have a student designer volunteer to demo a game with another student acting as the play tester. The students can play out how they would conduct a testing session for the whole class. What works about the game? Is gameplay intuitive? What may be missing in terms of the game and/ or the controls by which to play it? The teacher, as well as other students, can make suggestions for model questions or feedback.

### INVITING TO THE ARCADE

To celebrate the completion of the game designs and share them with other students in school, we set up arcades at the end of each project. Sometimes these arcades take place on the last day of the class; at other times they can come weeks later, for instance at the occasion of a larger school event. To invite other students to these arcades, we send out emails or ask designers to make flyers to announce the event. These invitations can be hand drawn and also include images of their Scratch screens and MaKey MaKey controllers.

### RUNNING THE ARCADE

When student playtesters come to the arcade, we often bring them in as small groups, directing them to a station with a Scratch game on the screen and the controller attached. Here are some suggestions on how to instruct the student play testers so that they can engage in **{“constructive criticism”}**. We suggest that they ask the designers to give them a demonstration of how to play the game and operate the controller. We ask playtesters to try out the game for a few minutes before taking notes on their evaluation cards (see following page for more on these). To keep play testing groups moving along, we call out every five or ten minutes when it's time to move to the next station.



Photo Darryl Moran



Photo Darryl Moran



# PLAY: SETTING UP THE ARCADE

## EVALUATING GAMES AND CONTROLLERS

To help student play testers with providing feedback about the game and controller designs, we hand out evaluation cards with simple questions such as: "What did you like about the game?" "What can be improved?" leaving some room for answers. On the top of evaluation cards, we ask students to write name of the game and designer(s). We ask the student playtesters to complete these cards and hand them back to the teacher, or the student or team who designed the game, at the end of their session.

## MAKING YOUR OWN EVALUATIONS

Another way to help student designers with setting up and organizing the playtesting, is to have them make their own evaluation cards. They can work in small groups or as a class and brainstorm together what kind of feedback or information about their game they want from their playtesters and what kinds of questions they want to ask. This way, each game designer could ask very specific questions about the graphics of their Scratch game and about the functionality and usability of their MaKey MaKey game controllers.

## DEBRIEFING AFTER THE ARCADE

After the arcade is over, it is important to hear from student designers as to what they saw playtesters doing with their games and controllers. Did everything work as they expected or did they have to fix code or controller designs? What did they ultimately learn about improving their games? This is also a good time to ask what surprised them most about the feedback and the responses they received.

Student Feedback Cards

Name of the Game:	Spike Jump Thing
What did you like about the game?	I like flappy bird
What did you like about the controller?	A little hard to figure out
What would you add or change?	Flashy colors could cause seizures, solid colors would be better

Name of the Game:	Spike Jump thing
What did you like about the game?	Graphics
What did you like about the controller?	it fit on the controller
What would you add or change?	the bugs

Name of the Game:	spike jumping
What did you like about the game?	I like how hard it is.
What did you like about the controller?	It is Airily easy to use.
What would you add or change?	Less glitching out.

1. Game You Tried:	Animal vs. Carrot
2. Your Grade:	K 1 2 3 4 (5) 6 7 8
3. One Thing You Liked:	I loved the imagination
4. One Thing You Would Change/Add:	
5. Would you come back for another arcade?	YES NO

1. Game You Tried:	Kill the Alien
2. Your Grade:	K 1 2 3 (4) 5 6 7 8
3. One Thing You Liked:	challenging
4. One Thing You Would Change/Add:	more than one alien on the screen
5. Would you come back for another arcade?	YES NO

1. Game You Tried:	Animal vs. Carrot
2. Your Grade:	K 1 2 3 4 (5) 6 7 8
3. One Thing You Liked:	creative keys
4. One Thing You Would Change/Add:	Better keys
5. Would you come back for another arcade?	YES NO

1. Game You Tried:	kill the Alien
2. Your Grade:	K 1 2 3 (4) 5 6 7 8
3. One Thing You Liked:	the crab
4. One Thing You Would Change/Add:	harder to get the alien
5. Would you come back for another arcade?	YES NO

# PROJECTS

In this section we will outline three projects you can try out in your classroom to extend game design beyond the screen. The projects are ordered according to complexity/difficulty level, from beginner to intermediate:

- + JOYSTICKS & TOUCHPADS
- + BOARDGAMES +
- + WEARABLE CONTROLLERS

Remember you can remix, iterate, and redesign these projects to meet your classroom's needs. The processes outlined in the *Setting Up Scratch and Makey Makey With Your Students* and *Setting Up the Arcade* sections found earlier in the guide can be applied to each of these projects.

## JOYSTICKS & TOUCHPADS

### OVERVIEW

In this introductory project, youth get to design controllers for remixed Scratch games. Youth begin the project by getting an introduction to Scratch and the MaKey MaKey and then move onto remixing an existing game from the Scratch starter projects studio. Then, using conductive materials like Playdoh and foil, they design a controller and in the process have to think about usability and conductivity.

### Project Timeline

8 CLASS SESSIONS (APPROXIMATELY 8 HOURS)

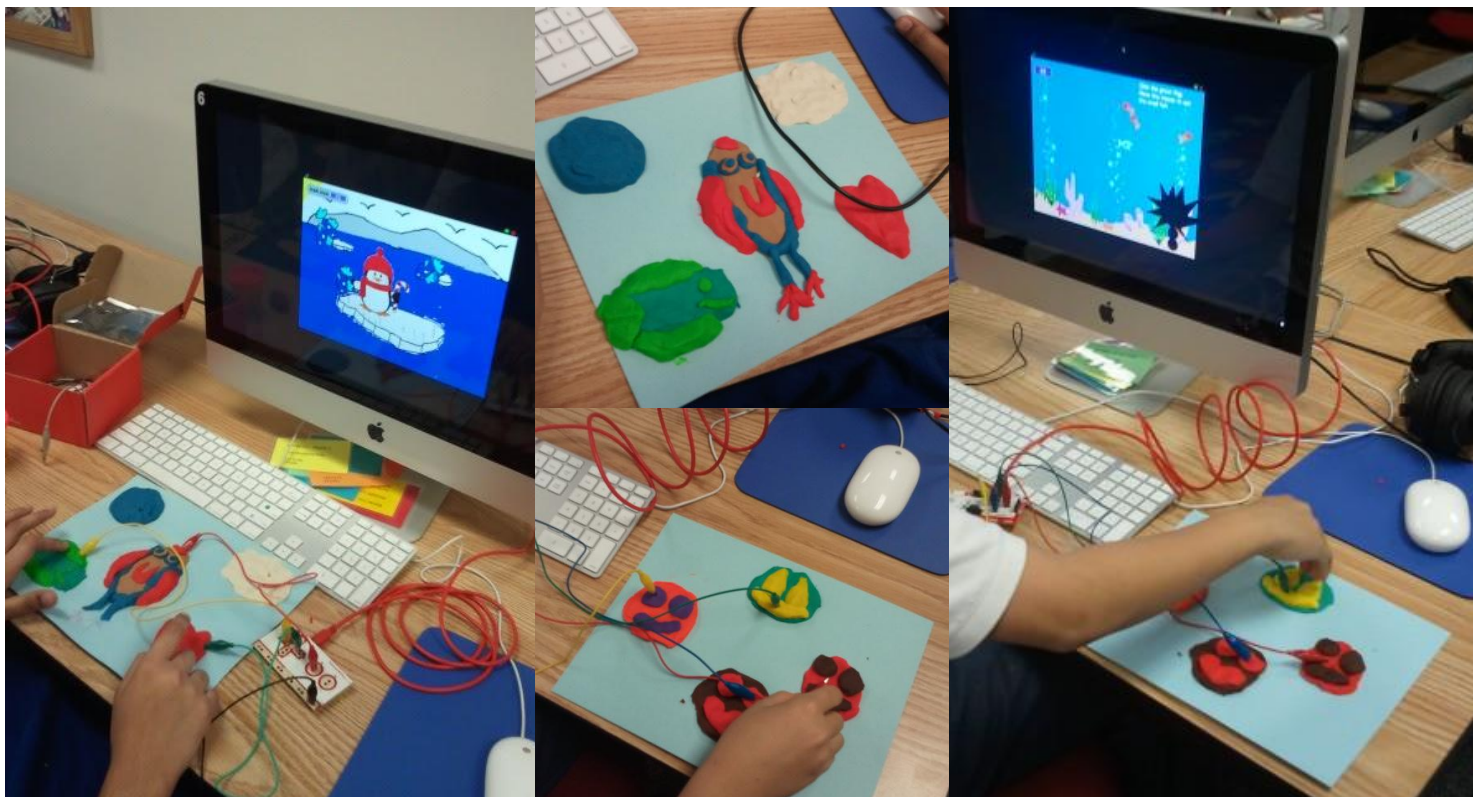
The activity arc includes time to develop Scratch skills. However, if your students are already comfortable with Scratch, you can go straight to remixing games and then moving on to controller design from there.

[ Session 1 ]	[ Session 2 ]	[ Session 3 - 5 ]	[ Session 6 ]	[ Session 7 ]	[ Session 8 ]
<b>Intro to Scratch + Makey Makey</b>	<b>Learning Scratch Basics</b>	<b>Remixing A Game In Scratch Starter Projects Studio</b>	<b>Designing and Testing Controllers</b>		<b>Playing in the Arcade</b>

### Learning Computational Thinking

These are the {computational thinking} concepts, practices and perspectives that are touched on by this unit.

CONCEPTS	PRACTICES	PERSPECTIVES
<ul style="list-style-type: none"> <li>• SEQUENCES</li> <li>• PARALLELISM</li> <li>• EVENTS</li> </ul>	<ul style="list-style-type: none"> <li>• TESTING &amp; DEBUGGING</li> </ul>	<ul style="list-style-type: none"> <li>• CONNECTING</li> </ul>



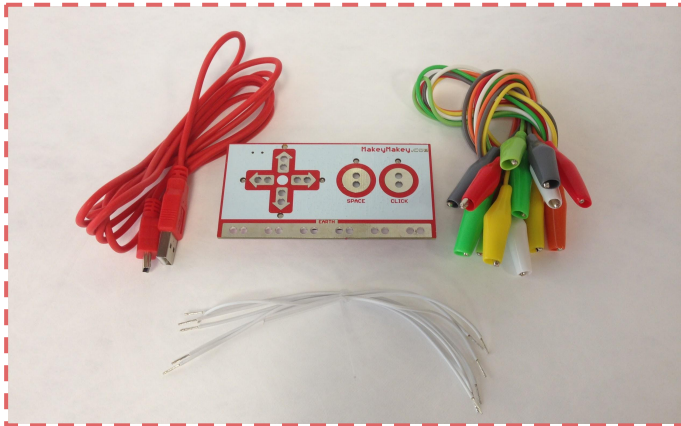


# JOYSTICKS & TOUCHPADS

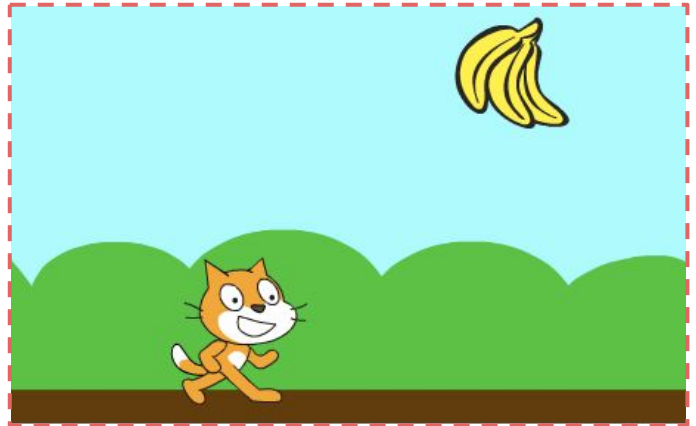
## SUGGESTED MATERIALS

Below is a list of materials that we've used in workshops in the past. You can add to or tweak the materials list based on what you have available. As mentioned earlier in the guide, another way to get students involved is for them to bring conductive items in from home. Our list is intentionally simple because we don't want students to get lost in the materials.

**MAKEY MAKEY**



**SCRATCH**



**PLAY DOH**



**ALUMINUM FOIL**



**PIPE CLEANERS**



**CONDUCT**

**MARKERS**



**CARD STOCK/ CARDBOARD**



**GLUESTICKS**



**CONSTRUCT**

## JOYSTICKS & TOUCHPADS

### (1) INTRODUCTION TO SCRATCH AND MAKEY MAKEY

See the Tools chapter for recommendations on how to get started with your students.

### (2) REMIXING A GAME FROM SCRATCH STARTER PROJECT STUDIO

#### Working with Existing Scratch Games

The goal with this project is to familiarize students with how games are built in Scratch and for them to think about the different components of different types of games.

1. Ask students to navigate to the Scratch starter projects studio and click on games. Click here for the Starter Projects.
2. Give students time to play each of the games. The games are simple so students should be able to play all of them within 10-15 minutes.
3. Once they have had a chance to play all the games, start a discussion about the games. Then, ask students to join you in a discussion about the games.

#### Remixing Games

Ask students to pick a game they want to remix from the options available. Have them brainstorm ideas for how they might remix their game. Here are some options for them to consider:

- Adding a score
- Adding a way to win/lose the game
- Creating new levels
- Devising new obstacles (e.g. something that results in losing points and increases difficulty)

Next, have students sketch their ideas. Provide students with a sheet of paper that asks them to draw a game screen and then to describe in a few sentences what they expect will happen in their game.

After this, they will be ready to start working with code. To give them some direction, have students identify 1-2 things they want to work on. Use code from a Scratch game that you have programmed yourself or taken from an already existing Scratch game. If you are remixing an existing Scratch project from the online community, this is also a good moment to demonstrate for students how to give credit to the original project creator.

Take a few class periods (we recommend up to 3 depending on students' prior programming experience and the complexity of their projects) to support students with remixing their games. They don't have to perfect everything; there will be time after they create their controllers to build, test, and tweak.

For beginners it's much easier to start with MaKey MaKey inputs like up and down, right and left, and spacebar. As your students are remixing, keep this in mind so you can help them think about controller integration later. Also keep in mind that starter projects like *Hide and Seek* ([https://scratch.mit.edu/projects/85963232/?tip\\_bar=hide](https://scratch.mit.edu/projects/85963232/?tip_bar=hide)) in Scratch are great games for remixing but are not very easy to remix for a controller design project because they require a mouse input.

#### TIPS!

#### MANAGING QUICK MOVERS

If your students are moving quickly through their remixes, this is a good time to let them test their games with a rough controller button. This will encourage other students to keep moving along to get to the controller design phase. You can also use this as an opportunity for students to test out each other's games and give informal feedback. Testing and thinking about the games of others will help get them thinking about improvements they can incorporate into their own games.

#### DISCUSSING GAMES

- a. How would they classify the games?
- b. What features do the different games have?
- c. Can the games be won or lost?

## JOYSTICKS & TOUCHPADS

### (3) CREATING TOUCH PAD CONTROLLERS

#### Prototyping Controllers with MaKey MaKey

The first step in supporting students to design a touchpad controller is to have them spend time prototyping so they can get a sense of the components they'll be working with, and what they'll need to consider in their design. Let's say a student chose to remix the Maze Starter (<https://scratch.mit.edu/projects/10128431/>) game. Maze Starter is a good game to choose because it utilizes the up, down, right and left arrow keys. Initially, you would want to have the student test their game with something simple like four pieces of foil wired to the MaKey MaKey. To get them started with prototyping:

1. Have them take stock of all the components to make a rough prototype (e.g. MaKey MaKey, four conductive pieces, wires, Play Doh).
2. Spend some time discussing conductivity and why things that are conductive can act as buttons" when touched.
3. Have students set up their prototype, testing each button as they go.
4. Ask students to spend time playing their game and make observations about what works regarding
  - a) Placement of buttons
  - b) Cohesion with their game.
5. Give students time to make tweaks to their game and retest using the prototype controllers.

When we have done this in our own game design workshops, students often forget to provide code for each of the up, down, right and left arrow keys. Other students thought about laying things out in a certain way or tweaking their game based on the new input.

### (4) DESIGNING CONTROLLERS

A good way to start is to lay out all of the materials you will be working with. Then, have students think about their prototyping and consider:

- a. How many buttons do they need for their controller?
- b. What material do they want to use?
- c. Will their controller somehow mirror their screen design?

Have students visualize and articulate their plan by sketching it out. Make sure their sketched diagram includes the MaKey MaKey and where a user might hold the earth button. As they work, check in with students, asking them about design choices and helping them think through challenges. Then, have students test their games several times with their own controllers.

### (5) PLAYTESTING IN THE ARCADE

When they are ready to share, have students play each other's games and give feedback. See the "Setting Up the Arcade" section for recommendations on how to structure this activity for your students.

#### TIPS!

#### DISCUSSING DESIGN

Use this prototyping process as a chance to talk about design. Have students consider questions like: What did we learn about our games that we didn't know before we tested them with the prototype controllers? What changes did you have to make to your games? Can you think of other games that have special controllers - what is similar or different about what you want to design?

#### INTRODUCING SKETCHING

Sometimes students can get bogged down by anxiety over the "look" of their sketch. When asking students to create their sketches, be sure to emphasize the exploratory and experimental nature of sketching. It is not about drawing a work of art but rather mapping out their ideas. Explain to them that it is just another way for them to plan and think through their ideas.

## JOYSTICKS & TOUCHPADS

### WHAT STUDENTS HAD TO SAY

#### On designing your own game...

Donald: "Um, I liked the, um, part where you get to play your own games, see what it's real like, and get your own visual thing, like...Um, sometimes when I'm like playing Xbox 360, I'm like why doesn't it have this? Like I can make it like that."

#### On designing controllers...

Tasfia: "Because some people they just use ordinary keyboards. It's kinda boring, but if you make your own keyboard and get to use it that's better."

#### On the arcade...

Tasfia: "But actually it was really fun cause hmm.. it was kind of hmm.. I was kind of getting angry because you have to say the same thing to the kids. But it was really fun cause you have to spend time hmm.. without doing them at class? Hmm, so hmm, spending the time with those kids, it was kinda fun cause they really like my game.... But it was kinda fun to explain it to them. Like, it was like. They were really like so interested."

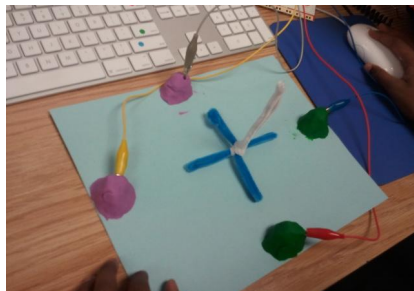
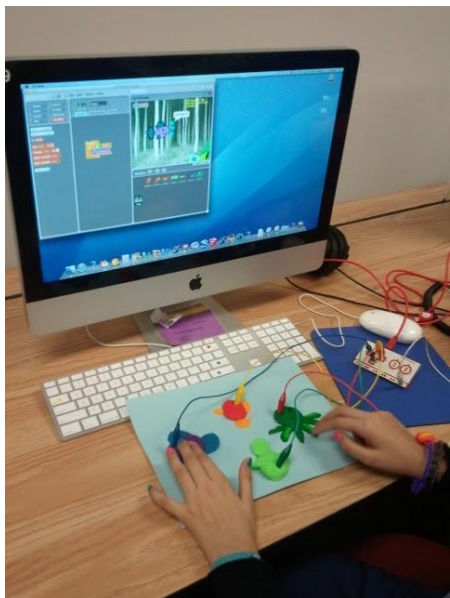
Donald: "Um, well at first I thought my game was ... um a bit boring. But, um, when the fifth graders just played it a lot, the two classes, um, a lot came to it. They played it said it was fun and...that made me think that okay, it's...it's good, it's fun."

Vincent: "I think it was like a really fun, we had a really good time..."

#### On my favorite part...

Donald: "Um, my favorite part was the, the favorite part of the workshop was playing other people's games. And like I said before, play my own. Cuz then I also get to see what other people did."

Student  
Projects





## JOYSTICKS & TOUCHPADS

### CASE STUDY: CONTROLLING THE ZOMBIES

We consider Amani, a sixth grade girl, who remixed a simple starter project that we provided her. She created a game where the player controls a zombie that has to consume brains to survive. How did she remix? First, she found a zombie graphic by searching online. She also downloaded images of brains, some of which she recolored in the Scratch paint editor. She customized a background by drawing a city scene, again using the Scratch paint editor. She also selected Michael Jackson's *Thriller* as her background music because she felt it well fit her theme (the *Thriller* video is famously one where Zombies dance around).

In terms of programming, Amani's initial design was simply a zombie eating brains. However, in a second workshop on the topic, Amani added complexity to her game by adding both good (pink) and poisonous (green) brains. If the zombie ate the pink brains, they would earn points, but if the zombie ate bad brains, the player would lose points. She also correlated the size of each of the brains to a point value so the brains with higher point values were larger while those with lower point values were smaller in size (see top, Figure 3). During an arcade for younger students, Amani also opted to add a high score to "win" the game because she found that her game was so enjoyable that her players would go on playing for several minutes. By adding a way to win, she was able to move others through the line and give her game a definitive end.

Like her Scratch game, Amani's touchpad controller went through a few iterations (see bottom, Figure 3). She designed the first version to have two arrows and a space bar to match the zombie's left, right and jump movement. One challenge with keeping the entire touchpad on one surface was that the moisture from the Play-Doh began to seep into the paper, thus causing short circuits and stopping her controller from working. Accordingly, in the next iteration, she added a thin orange outline and placed the pieces on foil tape. However, when she tested this second version of the interface, it also did not work because it kept short-circuiting. In this case, the entire surface became conductive. In her final version, Amani made sure that each key (right, left and space) was a separate button. She also opted to use pink as the color for her buttons to match the color of the brains in her Scratch game. She shaped them into arrows for more intuitive usability.

Amani, through this design, iterated on her original idea and eventually composed a more sophisticated game that reflects important instincts. The implementation of score, the use of challenges like green brains, and the differing point values all resulted in a more sophisticated (and more playable) video game. Her Halloween-inspired theme created a definitive tone to her game, as evident with its carefully-chosen soundtrack and even with the pink hue of her game controllers.



Student  
Projects

## BOARDGAMES +

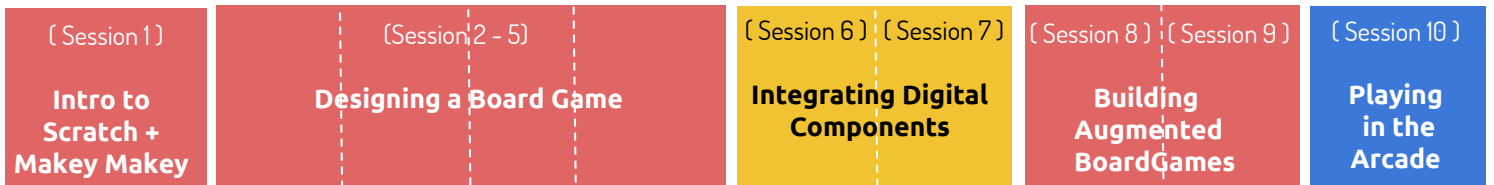
### OVERVIEW

In this workshop students will design their own augmented board games. First, they will design a board game and when they have worked out the game logic and mechanics, they will decide to augment (digitally) some of the components. For example, if the game has a dice or a spinner, they will use Scratch and MaKey MaKey to integrate a digital version of these components. So when students play their augmented board games, instead of rolling dice, they will touch a hot spot on their boards. This project is meant to be collaborative and works best with 3-4 students.

### Project Timeline

10 CLASS SESSIONS (APPROXIMATELY 10 HOURS)

The activity arc includes time to develop Scratch skills. However, if your students are already comfortable with Scratch, you can go straight to remixing games and then moving on to controller design from there.



### Learning Computational Thinking

These are the {computational thinking} concepts, practices and perspectives that are touched on by this unit.

CONCEPTS	PRACTICES	PERSPECTIVES
<ul style="list-style-type: none"> <li>• SEQUENCES</li> <li>• PARALLELISM</li> <li>• EVENTS</li> </ul>	<ul style="list-style-type: none"> <li>• TESTING &amp; DEBUGGING</li> <li>• BEING ITERATIVE &amp; INCREMENTAL</li> </ul>	<ul style="list-style-type: none"> <li>• CONNECTING</li> </ul>



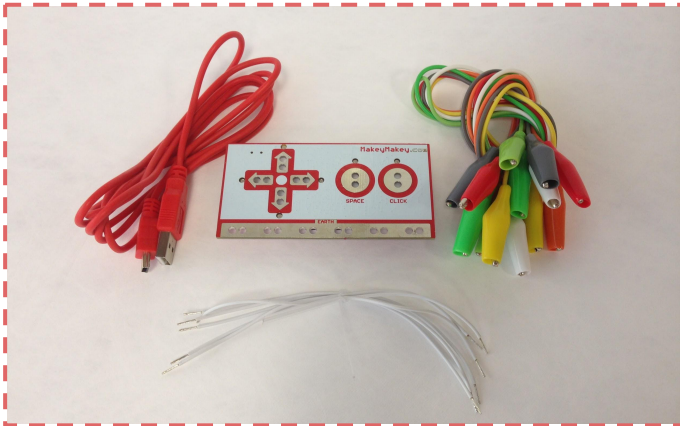


### BOARDGAMES +

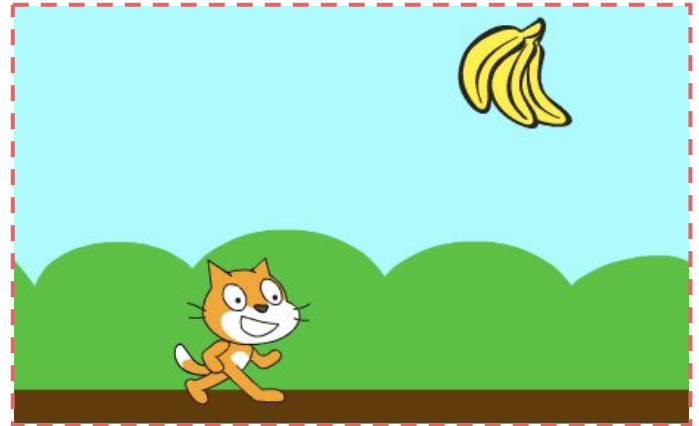
#### SUGGESTED MATERIALS

Below is a list of materials that we've used in workshops in the past. You can add to or tweak the materials list based on what you have available. As mentioned earlier in the guide, another way to get students involved is for them to bring conductive items in from home. Our list is intentionally simple because we don't want students to get lost in the materials.

#### MAKEY MAKEY



#### SCRATCH



#### CARD STOCK/ CARDBOARD



#### FABRIC PAINT/ MARKERS



#### CONDUCTIVE THREAD



#### ALUMINIUM FOIL

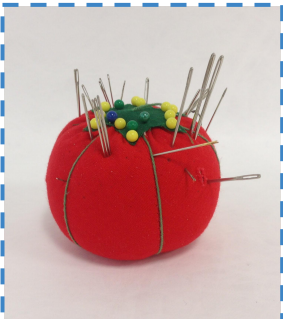


CONDUCT

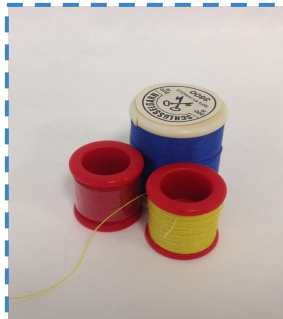
#### HOT GLUE



#### NEEDLES



#### THREAD



#### SCISSORS



#### EMBELLISHMENTS



CONSTRUCT

## BOARDGAMES +

### (1) INTRODUCTION TO SCRATCH AND MAKEY MAKEY

See the Tools chapter for recommendations on how to get started with your students.

### (2) DESIGNING A BOARD GAME

#### Playing Games to Make Games

To begin this part, we recommend having students spend a little while playing sample board games. Break students up into groups of 4–5 and have them play some traditional board games. Then, spend a few minutes discussing the features of the board games. Ask students to consider the following: What is needed to produce a board game? What makes it compelling? Interesting? Challenging?

#### Sketching Out an Idea

Once students have played existing board games and they have some ideas in their minds, it's time for them to brainstorm ideas for their game. Give them a chance to brainstorm in their small groups and have them consider the following:

1. THEME: What is the game about?
2. TYPE OF GAME: What type of game? E.g. start to finish, monopoly style, solve the puzzle
3. NUMBER OF PLAYERS: How many players can play the game?
4. COLOR SCHEME: What should their game look like
5. GAME MECHANICS: How will players win or lose the game? Move around the board? Playing cards? Dice? Spinners?

When they've spent time thinking through these things, ask students to build a basic prototype. All they will need is a piece of paper and something to write with. The idea is for students to create a sketch that helps them concretize all of the brainstorming.

#### Playtest Boards

After students have a chance to bring their ideas together, have them design a playtest board. This will require larger paper and/or poster paper so they can make a more realistic outline of the board. Have students draw a more life-like version of their boards and also work on creating any of the components they need to create (e.g. playing cards). Depending on the facilities and the student's own tech savvy, they can opt to do these things on the computer or by hand. Once the students have had a chance to create their playtest boards and the related components, have them play the game several times. Through playing they will start to see what works and what needs to be fixed.

#### Augmenting Playtest Boards

Once students have had a chance to work out the issues with their playtest boards, it is time for them to integrate some of the digital components and work on remixing these pieces.

### (3) PLAYTESTING IN THE ARCADE

When they are ready to share, have students play each other's games and give feedback. See the "Setting up the Arcade" section for recommendations on how to structure this activity for your students.

#### TIPS!

##### SHARING IDEAS

After students sketch their basic prototypes, it is a great time to bring them together and have the groups share ideas with the rest of the class. This will also give them a chance to practice giving and getting feedback and present their ideas.

##### ADDING DIGITAL COMPONENTS

In augmenting their board games, students will be taking analogue pieces of their game, and making them digital. For example, if students want to digitize their dice, they can set up a hotspot on their playtest board for the dice and then play their game with this new component. As they play, they will observe what works | what doesn't work. Encourage them to integrate one component at a time.

**BOARDGAMES +**

WHAT STUDENTS HAD TO SAY

**On the experience...**

Astan: "The way it came out. I think me and Laila did a really good job on this. We could probably do a little more neater but I think it turned out pretty good. We put our hearts and work in it. Yeah. My favorite part was the board because I got ideas me and Laila got ideas from our head and put them on the paper. And we're like yes that's go there. Let's do bad cards let's do good cards. Let's do this let's do that. And then it was just so good. And we came up with the name because we had so many spots and I was like yea, That Spot."

Syeda: "Um I think it was valuable coz I got to um learn more new things about that stuff [technology] coz I never knew how this works."

Ingrid: "Well I just think it's cool that we got to make a digital game. Like, we got, we made a board game and then transferred it onto the computer. And I think ... probably the way we, like we really worked on the cards the most and when you draw a card it comes up on the screen and it reads it to you and we have it down. And so I thought that was pretty cool."

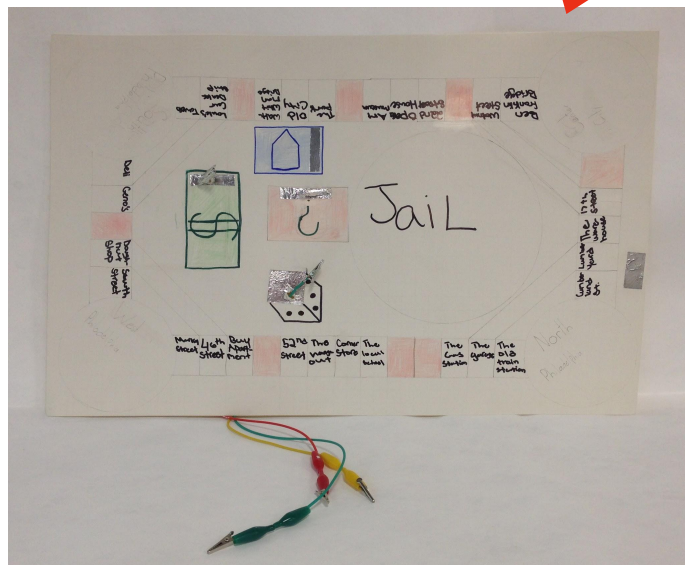
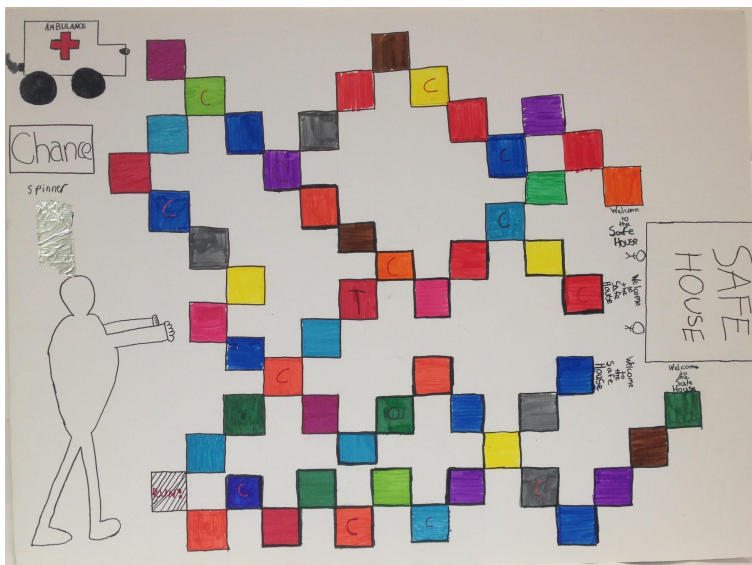
**On digitally augmenting board games...**

Zoe: "Um well the dice well you have to roll something and we uh in our family we lose the dice a lot and also um the these would be digital because the cards would be so much easier..it would look cooler if the cards popped up like they were in shuffled decks then they pop up. That would like add another interest to our game. That would like people would like more. And also um we don't wanna make like 100 cards out of paper if we digitalize it then that would be so much easier."

**On collaboration...**

Grey: "Working with another partner was hard just coz..we both have so many ideas. And sort of hard to collaborate...sacrifice some of your ideas for his and vice versa. So um you know if you're working by yourself, all of your ideas are the best ideas. And they automatically just go down to the paper."

**Student Projects**





## BOARDGAMES +

### CASE STUDIES: ROAD TRIP

Here we share a collaborative effort among three sixth grade girls, Evie, Kimberley, and Alanna. The trio designed a board game entitled *Road Trip*. This start-to-finish game was dreamt up by the girls in an early brainstorming session during our workshop. In the game, four cars begin on a start space and traverse challenges while trying to get across the country to a finish line. Each time a player rolls, they land on a blank space or one that is labeled with a shamrock (Lucky card), an X mark (Danger) or an Exclamation point (Trouble). The players are at the whim of the dice roll and cards as they try to get to the end.

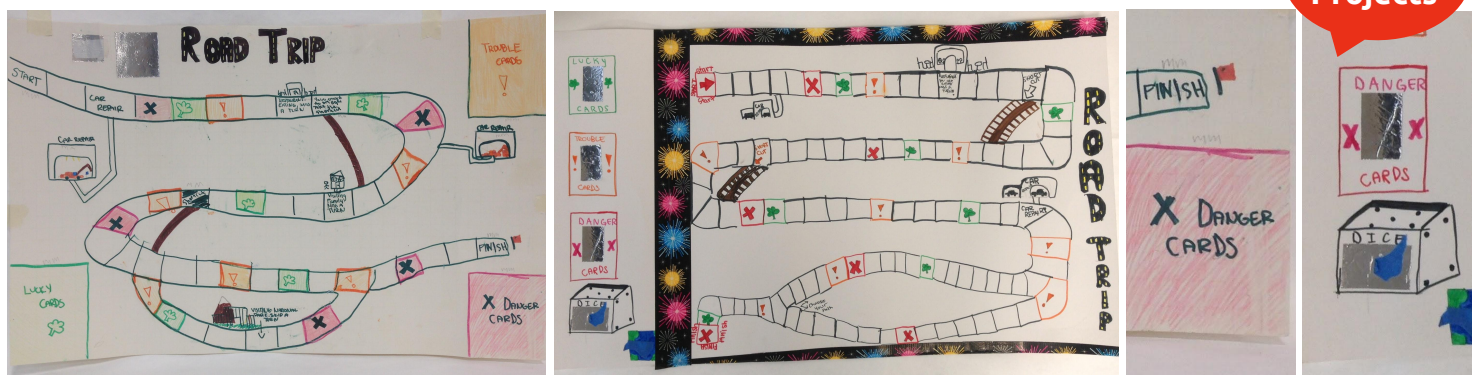
The girls began designing a play test board by sketching out their initial idea using paper and pencil. Even though it wasn't required, the girls opted to test out the colors and design elements they hoped to include in the final design. During our workshop we had students play each other's games so they could get feedback from their peers. As their peers played their game, team *Road Trip* observed that their board was too simple and needed more spaces for greater variety and challenge. They took note of these issues so they could make adjustments in future iterations.

After play testing and talking through the possibilities for digitizing certain components, the girls opted to digitize the playing cards that were originally hand-written on paper. In order to make this switch from paper to screen, they had to determine where to add new hotspots on their board game and how to remix the code for playing cards. They also had to generate additional content for the digital playing cards that was theme-specific. The girls felt this was extremely time consuming and challenging. They mentioned that the research for the trivia questions was particularly challenging. But they also concluded that it was entirely worthwhile in the end.

When it was time to move to the final board game, the girls meticulously drew the new board. They maintained their original symbols for danger ("X"), trouble ("!") and lucky cards (a shamrock) and also made sure that the hot spots were aligned. To accommodate for their digitizations, the girls added a foam panel for their now digital dice and a connection to earth (or a connection to ground the circuit required by MaKey MaKey). They drew a picture of dice as well as the earth as two visual cues to remind players. This design made it easier to roll the dice and get a playing card because a player would simply have to touch earth and one other option on the board rather than having to hold it in their hand.

The girls were proud that they designed a game that their peers were subsequently impressed by. Alanna explained, "It's cool that we got to make a digital game. Like, we got — we made a board game and then transferred it onto the computer. We really worked on the cards the most and when you draw a card it comes up on the screen, and it reads it to you and we have it down. And so I thought that was pretty cool." The experience of designing a game and then digitizing components including playing cards resulted in a hybrid that was quite different (and enjoyably so) than pulling cards from a paper stack. In this case, the girls were able to collaborate on a vision and bring it to fruition from just a simple idea on paper, blending high and low tech to create a new artifact.

Student  
Projects



## WEARABLE CONTROLLERS

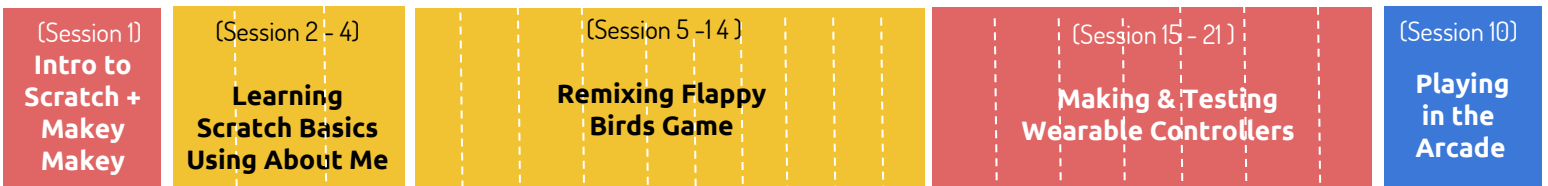
### OVERVIEW

In this workshop students will design (or remix) their own video games in Scratch and make a wearable controller to play their games. This unit will involve learning the basics of Scratch functionality, remixing or designing unique Scratch games, learning about circuits via MaKey MaKey and designing wearable controllers. As they move through the various phases, students will be positioned as designers or makers, having to think through the aesthetics, game logic and functionality of their on-screen (Scratch game) and off-screen (wearable controller) components.

### Project Timeline

20 CLASS SESSIONS (APPROXIMATELY 20 HOURS)

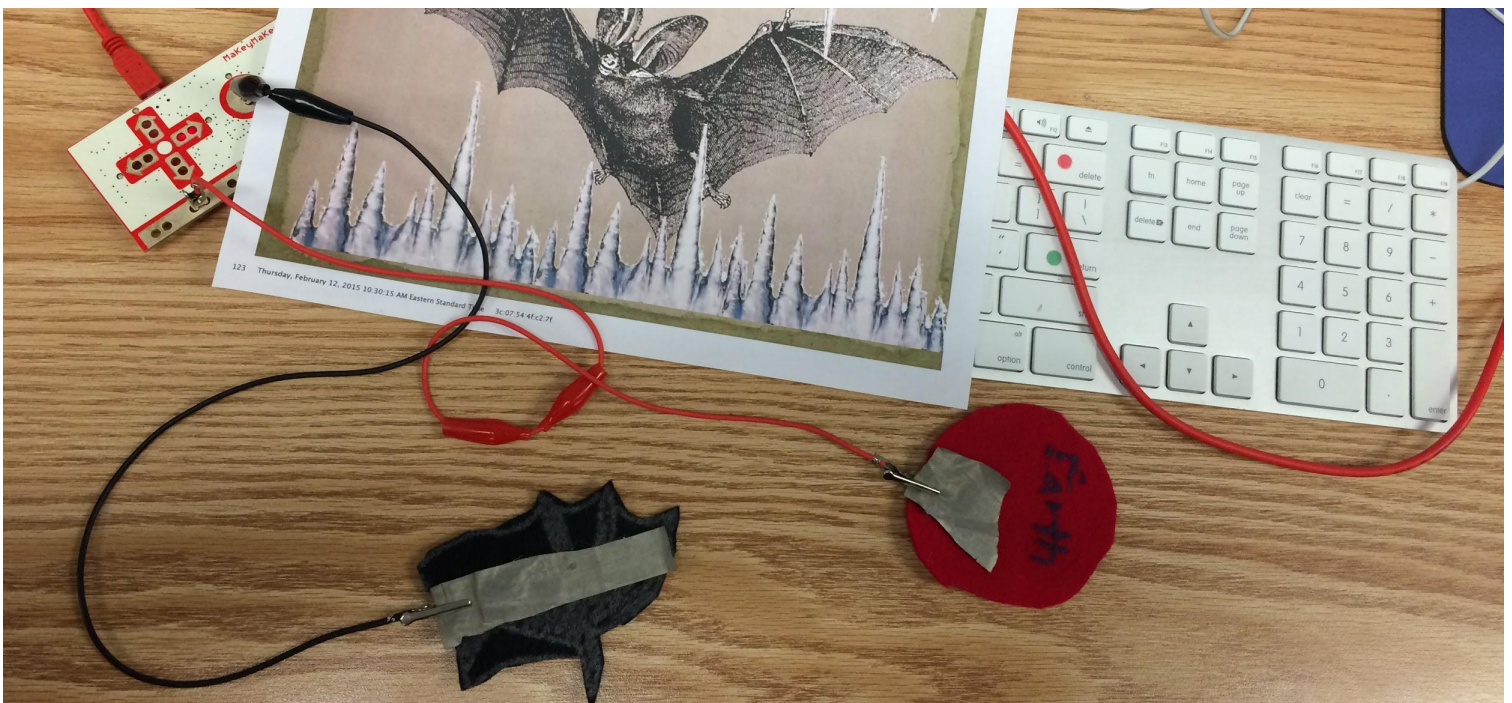
The activity arc includes time to develop Scratch skills. However, if your students are already comfortable with Scratch, you can go straight to remixing games and then move on to controller design.



### Learning Computational Thinking

These are the {computational thinking} concepts, practices, and perspectives that are touched on by this unit.

CONCEPTS	PRACTICES	PERSPECTIVES
<ul style="list-style-type: none"> <li>• SEQUENCES</li> <li>• PARALLELISM</li> <li>• EVENTS</li> </ul>	<ul style="list-style-type: none"> <li>• TESTING &amp; DEBUGGING</li> </ul>	<ul style="list-style-type: none"> <li>• CONNECTING</li> </ul>



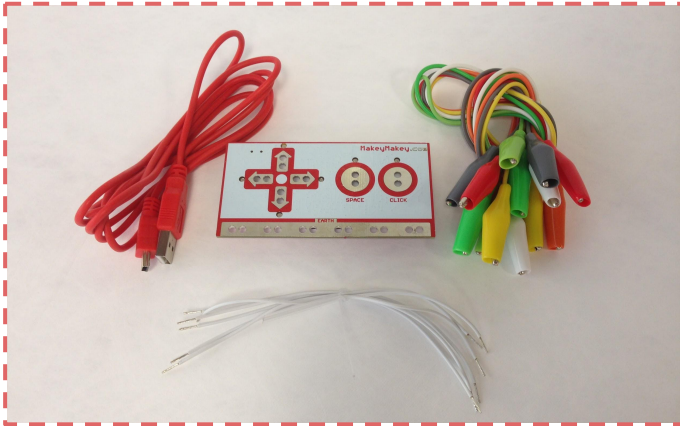


# WEARABLE CONTROLLERS

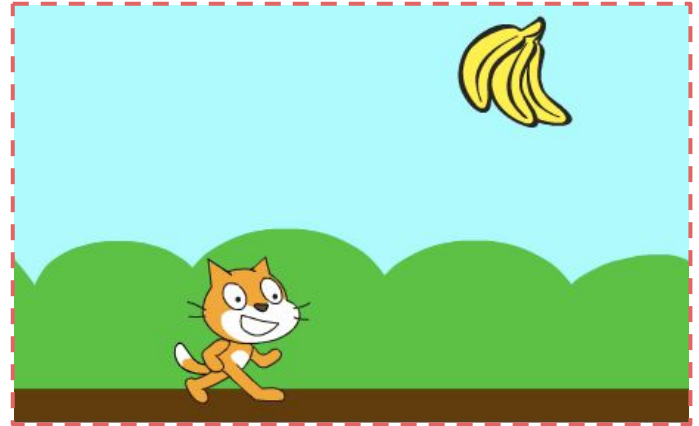
## SUGGESTED MATERIALS

Below is a list of materials that we've used in workshops in the past. You can add to or tweak the materials list based on what you have available. As mentioned earlier in the guide, another way to get students involved is for them to bring conductive items in from home. Our list is intentionally simple because we don't want students to get lost in the materials.

MAKEY MAKEY



SCRATCH



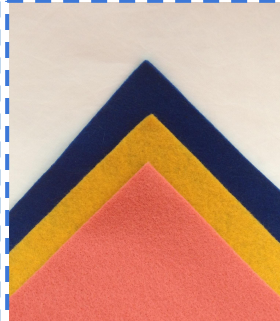
CARD STOCK/  
CARDBOARD



FABRIC PAINT/  
MARKERS



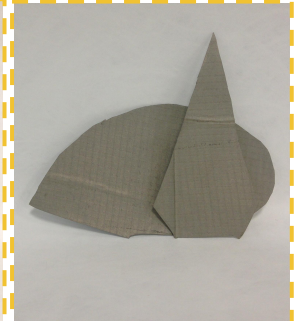
FELT



CONDUCTIVE THREAD



CONDUCTIVE FABRIC

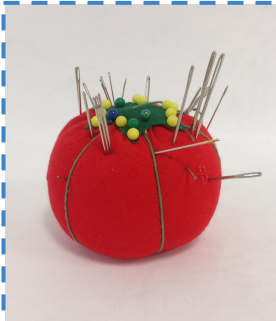


CONDUCT

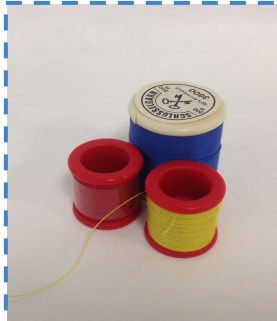
HOT GLUE



NEEDLES



THREAD



SCISSORS



EMBELLISHMENTS



CONSTRUCT



## WEARABLE CONTROLLERS

### (1) INTRODUCTION TO SCRATCH AND MAKEY MAKEY

See the Tools chapter for recommendations on how to get started with your students.

### (2) ABOUT ME

Once students are oriented to the basics, it is time to start the About Me project, or any other simple starter project you like for orienting students to the basic concepts in Scratch. The goal of this project is to get students to animate the letters of their name. If a student's name is Joe, they would create new sprites for the letters J-O-E and then animate them. This project forces students to think about movement, events, parallelism and sequences, all concepts that are important in game design. Other ways to personalize this project are to pick an upcoming holiday to be the inspiration for the program, or ask students to pick a word or phrase that means something to them.

You can carry this project on as long as it takes students to complete to their satisfaction. It most likely will be 1.5 - 2 classes (or class hours) for them to finish the project. Once you're done, have students share the projects in a studio and watch all of them.

Ask students:

What are some features of the games they've observed?

What is one piece of feedback they might give to other students?

What's one feature they admire in another's game?

This will be the first opportunity for students to make and share, so follow the conversation where it goes!

### (3) REMIXING FLAPPY BIRDS

#### Introducing Flappy Birds

There are several versions of flappy birds that have been developed in Scratch. We recommend looking at a few different versions by projecting them and showing students the game. Here are two links to get you started:

Simple version of flappy birds (<https://scratch.mit.edu/projects/26439299/>)

Sophisticated version of flappy birds (<https://scratch.mit.edu/projects/25828661/>)

Demo each of these versions with your students. Ask for volunteers to come try them as well. This play time is essential for students to observe the various features in each of the two games.

Try starting a discussion around what they're noticing about how the game functions:

1) What do they notice about the background? [Remember- the background scrolls, but the bird never moves!]

2) What do they notice about the bird? How does the bird stay afloat? [Get students to think about gravity].

#### TIPS!

#### DISCUSSING GRAVITY

If this project is embedded within a science or technology unit, you could use it as an opportunity to discuss gravity and what is required to keep the bird afloat. We recommend that you play both more and less sophisticated versions of Flappy Birds and think about how smooth (or not) the bird's movements are and how gravity plays a role in such movement.

#### PERSONALIZING PROJECTS

Projects like "About Me", are great to get students started because of their personalized nature. We've found that students love doing projects that incorporate visual representations of their names. It lets them explore expressions of their identity in a way that's simple and open ended while not being overly abstract. It adds an extra layer of engagement.

## WEARABLE CONTROLLERS

### Remixing: Making Choices

Once students are ready to design their own game give them some time to brainstorm ideas for what they want to make changes to. This will give them a clear sense of how to move forward. Some of the key areas to consider are:

1. THEME	What is the game about?
4. COLOR SCHEME	What should their game look like?
5. GAME MECHANICS	How will players win or lose the game? How will they move around the board? Will they use playing cards? Dice? Spinners?

### Designing Wearable Controllers

When students have a working remix of their game, it is time for them to create a prototype of their controller. Have students draw a sample design to get a sense of how they want to integrate the MaKey MaKey. They'll need to consider how the player will close the circuit, and what types of movements will make sense for the player when using the controller with the game.

Once students have produced a sketch they are happy with, have them construct a paper prototype so they can feel out the physicality of the controller, and how it will work with the user's body. This will give them a better idea of appropriate sizing and placement of components. They may use a simple conductive material like foil to test the controller out at this stage.

This is also a good time to have students test each other's controllers and give feedback. You can ask students to consider:

Is the controller comfortable?

Is it easy to use?

Is it fun to use?

Does it make sense with the game it was designed for?

After students have addressed any issues they encountered when testing their paper prototypes, they can begin to construct their final versions with their chosen materials.

### (3) PLAYTESTING IN THE ARCADE

When they are ready to share, have students play each other's games and give feedback. See the "Setting up the Arcade" section for recommendations on how to structure this activity for your students.

### TIPS!

#### STRUCTURE IN REMIXING

It's important to give students some structure to work within when starting to remix their games, so they don't become overwhelmed when they begin editing code. Try suggesting that they move through the code one edit at a time, and have them write out a step-by-step plan outlining how they will do this before they get started.

#### INTRODUCING THE IDEA OF THE USER

Before students begin to design their wearable controller, introduce the idea of the "user". Get them thinking about the different types of people that could be playing their game. What do they need to consider about the user and how it would affect their wearing the controller? Age, size, etc?

## WEARABLE CONTROLLERS

### WHAT STUDENTS HAD TO SAY

#### On the experience...

Sophia: "Um, well I really, I really like the idea. Well I just really like that I finished the game in time and it works right. Um and I also really like how the controller is kind of complex and we never really had to do anything like that in technology. So it was interesting to get to make and sew cause it was almost like sewing and technology and all these interesting things mashed together into a giant awesome sandwich."

Moritz: It was a good experience. I got to of course learn something new as I said before. I knew how to work with Scratch and the programming but I didn't, I think I reached a more advanced level, more sophisticated designing with the programming with that um, software.

Ingrid: Well it was cool that um, like we made our own game. Like when we play games at home, I don't think most of us know how they're made and then we got to do that. And yeah we got to go through the process of making a game and for people to play it, which was also really cool.

#### On the Arcade...

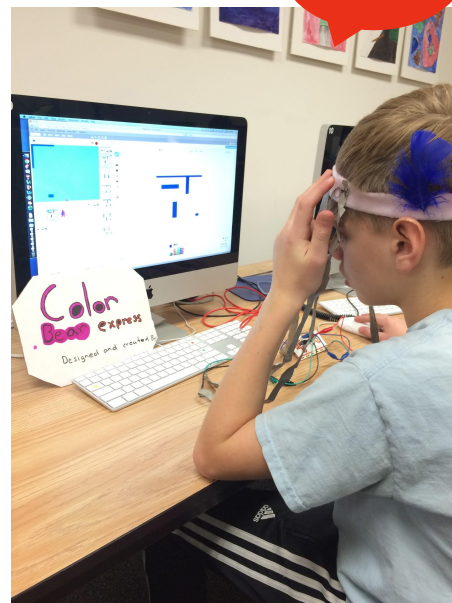
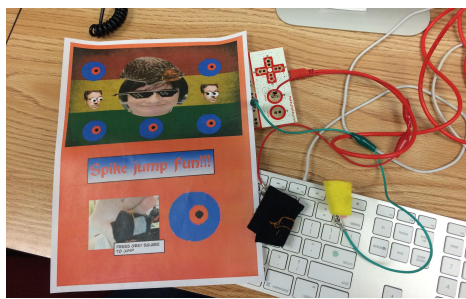
Moritz: "The arcade was fun. I got feedback from kids that weren't of my age. I got younger children's feedback, that helped me. So that's what gave me, you know, ideas for I can later improve my game... I got constructive feedback."

Ingrid: Um I think it was good that we had other people try our game because when you try your game it's probably different and I had to explain a lot to other people how, what to do and I think those are things you don't think about when making a game.

#### On what I'm most proud...

Ingrid: Well, just making a game probably because I never thought - or like when we play flappy birds I don't think we think about what goes on but then when you do it, you're like wow that took a lot of work and you have the product.

### Student Projects



## WEARABLE CONTROLLERS

### CASE STUDIES MICHAEL FLIES

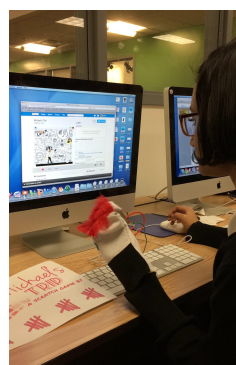
Sara is a seventh grade girl whose interest in music was clearly reflected by her remix of the popular game *Flappy Birds*. Sara's project was inspired by Michael, her favorite member from the boy band, *Five Seconds of Summer*. Sara felt confident remixing a slightly more sophisticated version of *Flappy Birds*, given her previous exposure to Scratch.

She began by transforming the aesthetics. She found a whimsical background, downloaded an image of Michael that she slightly tweaked, and created a secondary costume for her Michael sprite to make it appear he was flying. To do this, she had to edit the original image to make it look like Michael's legs are bent in the air. She then added wings to her protagonist to go with the "flappy" theme. For the soundtrack, Sara of course selected a song by *Five Seconds of Summer*, which plays as the character navigates the challenges in the game. Usually these challenges are by way of pipes that the flappy bird has to avoid but in lieu of traditional pipes, Sara chose scrolling obstacles. These included a little bat with a camera to represent the paparazzi that might follow the singer, and objects like donuts, bananas, and other things that would also act as deterrents to Michael staying afloat. Another twist Sara incorporated into her game was giving Michael the ability to stand up during slow parts of the game and to scroll during selective sections of play. She created all of the code for her scrolling challenges using trial and error.

Moving from the screen to the hand-held, for the wearable controller, Sara used the prototype as a guide, explaining it helped her to know where to locate the electronic fabric and connect components. First, she made a white felt hand by cutting a pattern out and sewing it together. Next, she used a hot glue gun to add red feathers that symbolized Michael's hair. She also added "googly" eyes and drew a small mouth to create a face on the glove. Attached black "pants" completed the look. Once the main construction was completed Sara began testing out the game with her controller. She temporarily pinned conductive fabric to the top and thumb section of her glove design to see if her code worked and if the flapping motion made sense with her game. When she had played it several times with success, she chose to hot glue the fabric to the top and thumb section of the inside of her controller. To play, a user would have to "flap" the controller to make the two hotspots touch each other.

The project gave Sara an opportunity to express her interest in music through a unique design. It also introduced Sara to new technologies that facilitated the production of a wearable controller which she had not done before. This all well-coupled with the Scratch skills that Sara had already attained. 7th grader Sara said of the experience, "What I liked about the project was ... I could like get in touch with my creativity."

### Student Projects



# APPENDIX

Excited? Curious? Confused about something? In this section you can find additional resources for educators and learners, as well as a glossary of terms.

+ RESOURCES

+ GLOSSARY

+ CONTACTING US AND SHARING THE GUIDE



## RESOURCES

MAKEY MAKEY

**PURCHASING:** Some retailers for buying your MaKey MaKey

**Joylabz:** <http://shop.makeymakey.com/>

**Sparkfun:** <https://www.sparkfun.com/products/11511>

**Amazon:** [http://www.amazon.com/Makey-Invention-Kit-Everyone/dp/B008SFLEPE/ref=sr\\_1\\_1?s=toys-and-games&ie=UTF8&qid=1464043595&sr=1-1&keywords=makey+makey&refinements=p\\_85%3A2470955011](http://www.amazon.com/Makey-Invention-Kit-Everyone/dp/B008SFLEPE/ref=sr_1_1?s=toys-and-games&ie=UTF8&qid=1464043595&sr=1-1&keywords=makey+makey&refinements=p_85%3A2470955011)

## GETTING STARTED

**Sparkfun's MaKey MaKey Quick Start Guide:** A resource for learning and understanding Makey basics

<https://learn.sparkfun.com/tutorials/makey-makey-quickstart-guide>

**Joylabz's Set Up:** A set up guide from the makers of MaKey MaKey

<http://makeymakey.com/how-to/classic/>

## TROUBLESHOOTING

**Joylabz Troubleshooting:** A troubleshooting guide from the makers of MaKey MaKey

<http://makeymakey.com/how-to/classic/#troubleshooting>

## ADVANCED

**Sparkfun's MaKey MaKey Advanced Guide:** A resource for reprogramming the Makey board with Arduino <https://learn.sparkfun.com/tutorials/makey-makey-advanced-guide>

## INSPIRATION

**Instructables:** An online community for people to post instructions for DIY projects. A great resource for inspiration and project ideas <http://www.instructables.com/howto/makey+makey/>

**Project Videos:** Playing Music with MaKey: <https://www.youtube.com/watch?v=xvmTav3SYs>

**Project Videos:** Palying Minecraft with MaKey:

[https://www.youtube.com/watch?feature=player\\_embedded&v=csjM4oDet0s](https://www.youtube.com/watch?feature=player_embedded&v=csjM4oDet0s)

## SCRATCH

**Creative Computing Guide:** Offers a rich selection of projects, curriculum, and resources to guide you through learning Scratch and bringing it into your classroom. There is a version of this guide for both educators and students. <http://scratched.gse.harvard.edu/guide/>

**Harvard GSE's ScratchED:** A community for Scratch educators <http://scratched.gse.harvard.edu/>

## RESOURCES

### FURTHER READING:

#### ACADEMIC PAPERS AND RESEARCH ASSOCIATED WITH THE PROJECTS IN THIS GUIDE:

All the projects described in this guide and more, have been documented in the research papers and book chapters listed below. In many instances they have more detailed project descriptions and case studies, and provide some additional background on the collaboration and learning that we observed in students' game designs beyond the screen. You can find copies of these papers at [yasminkafai.com](http://yasminkafai.com) or in many digital libraries.

#### Overview of Gaming Beyond the Screen:

Kafai, Y. B. & Vausdevan, V. (2015, November). Constructionist Gaming Beyond the Screen: Middle School Students' Crafting and Computing of Touchpads, Board Games, and Controllers. In *Proceedings of the 10th Workshop in Primary and Secondary Computing Education (WiPSCE '15)* (pp. 49-54). New York, NY: Association for Computing Machinery.

#### Detailed Reports:

Kafai, Y. B., Burke, W. Q., Mote, C. (2012, June). What makes things fun to participate? The role of audience for middle school game designers. *Proceedings of the Interaction Design for Children Conference (IDC10)*, Bremen, Germany.

Kafai, Y. B. & Vasudevan, V. (2015, June). Hi-Lo Tech Games: Crafting, Coding and Collaboration of Augmented Board Games by High School Youth. In *Proceedings of the 14th International Conference on Interaction Design and Children (IDC '15)* (pp. 130-139). New York, NY: Association for Computing Machinery.

Lee, E., Kafai, Y.B. Davis, R.L. & Vasudevan, V. (2014). Playing in the arcade: Designing tangible interfaces with MaKey MaKey for Scratch games. In A. Nijholt (Ed.), *Playful User Interfaces: Interfaces that invite social and physical interaction* (pp. 277 - 292). Singapore: Springer.

Mote, C., Kafai, Y. B., & Burke, Q. (2013). Virtual Design Competitions to Promote Communication, Collaboration, and Learning in Schools. *Learning and Leading with Technology*, 4(4), 16-21.

Vasudevan, V., Kafai, Y. B. & Yang, L. (2015, June). Make, Wear, Play: Remix Designs of Wearable Controllers for Scratch Games by Middle School Youth. In *Proceedings of the 14th International Conference on Interaction Design and Children (IDC '15)* pp. 339-342). New York, NY: Association for Computing Machinery.

Vasudevan, V., Davis, R. Kafai, Y. B., & Lee, E. (2013). Joystick Designs: Middle School Youth Crafting of Game Controllers. In Ochsner, A. & Dietmeier, J. (Eds.), *Proceedings of the Games + Learning + Society Conference*, Madison, WI.

#### General Background On Learning Programming, Crafting, and Game Making:

Kafai, Y. B. & Burke, Q. (2017). *Connected Gaming: What Making Video Games Can Teach Us About Learning and Literacy*. Cambridge, MA: MIT Press.

Kafai, Y. B. & Burke, Q. (2015). Constructionist Gaming: Understanding the Benefits of Making Games for Learning. *Educational Psychologist*, 50(4), 313-334.

Kafai, Y. B. & Burke, Q. (2014). *Connected Code: Why Children Need to Learn Programming*. Cambridge, MA: MIT Press.

## GLOSSARY

**ARDUINO:** Arduino is a programmable circuit board (or microcontroller), and an accompanying open-source software that can be used to program the board. There are several varieties of Arduino board including the LilyPad (e-textiles). They are popular for beginners exploring physical computing. <https://learn.sparkfun.com/tutorials/what-is-an-arduino>

**COMPUTATIONAL THINKING:** The process of thinking through a complex problem and its solution using concepts and practices from computer science.

**CONDUCTIVITY:** The degree to which electricity (or heat) to flow through a material.

**CONSTRUCTIVE CRITICISM:** Well-considered and thoughtful feedback given for the purposes of improving outcomes.

**GROUNDING:** In the case of the MaKey MaKey “grounding” simply means closing or completing the circuit, so electrical current can continuously flow through it.

**INTERFACE:** Hardware or software designed to communicate information or data between two things, including devices, programs, and users.

**LOGO:** A programming language designed as a learning tool. Developed in 1967, it could be considered a predecessor of sorts to Scratch. While it is no longer widely in use, it played an important role in the development of programming education for children. [http://el.media.mit.edu/logo-foundation/what\\_is\\_logo/logo\\_programming.html](http://el.media.mit.edu/logo-foundation/what_is_logo/logo_programming.html)

**PLAYTEST:** Playtesting is a standard practice in the game design industry where users play games at various stages of development, and give feedback on their playing experience. This serves to work out kinks and improve user experience in games before they go to market.

**SCROLLING:** In the context of a video game, a player can control an object while the background scrolls or slides across the screen, giving the illusion that the object is moving through space.

**USABILITY:** The degree to which a designed object can meet the goals and needs of its users, in terms of effectiveness, efficiency, and ease of learning. Considering usability in design means accounting for the prospective users needs, how those needs might shift and change over time, and the environment and circumstances in which it will be used.

## CONTACTING US AND SHARING THE GUIDE

### ATTRIBUTING AND SHARING THIS PROJECT GUIDE

This project guide is free.

We have licensed it under the Creative Commons License-Attribution-NonCommercial -ShareAlike. This license lets others remix, revise and expand on this work for non-commercial purposes, as long as you credit and license your work also under the identical terms.



### TO REFERENCE THIS GUIDE USE THE FOLLOWING CITATION

Vasudevan, V., Kafai, Y.B., & Widman, S. (2016). *Make Code Play: A Project Guide to Game Design Beyond the Screen*. Online publication. Philadelphia, PA: University of Pennsylvania. Available at:

<http://www.yasminkafai.com/mcp/>

### FEEDBACK AND QUESTIONS ABOUT THE PROJECT GUIDE

Please share your experiences, comments, or suggestions about the project guide and activities with [kafai@upenn.edu](mailto:kafai@upenn.edu)