NeoPixie Dust Bag with Circuit Playground Express

Created by John Park

https://learn.adafruit.com/NeoPixieDustBag_CPX

Last updated on 2024-03-08 02:37:26 PM EST

Table of Contents

Overview 3

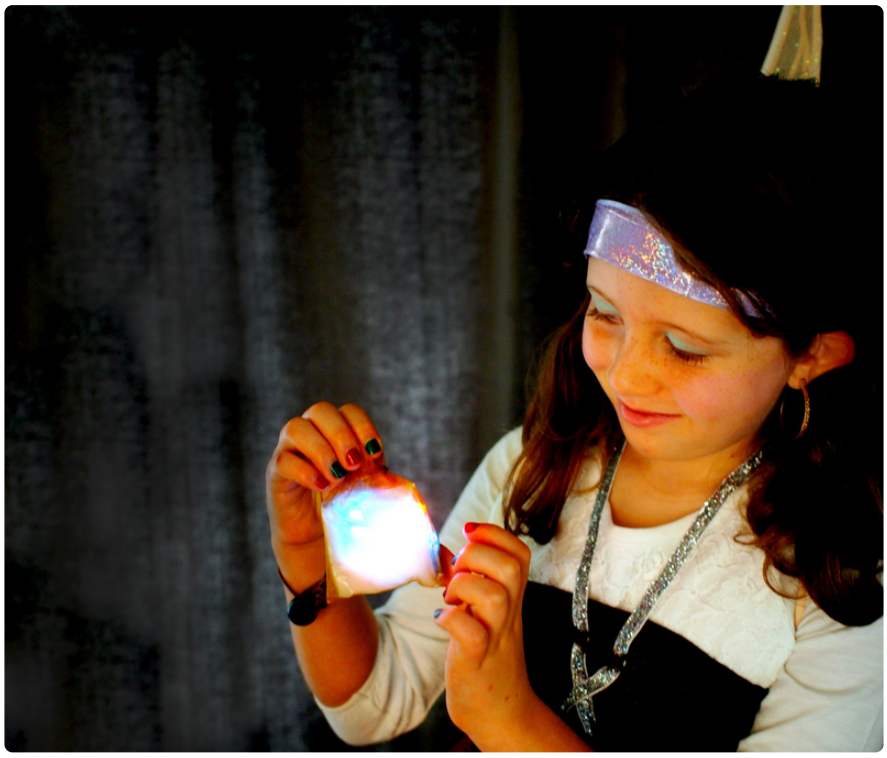
Code It 5

- Setup
- Animation
- Color Touch
- Variable Color
- Simulate It
- Pseudo ON/OFF
- Conditional If/Then/Else

Build and Use It 22

- Touch Sensor
- Bag in a Bag

Overview



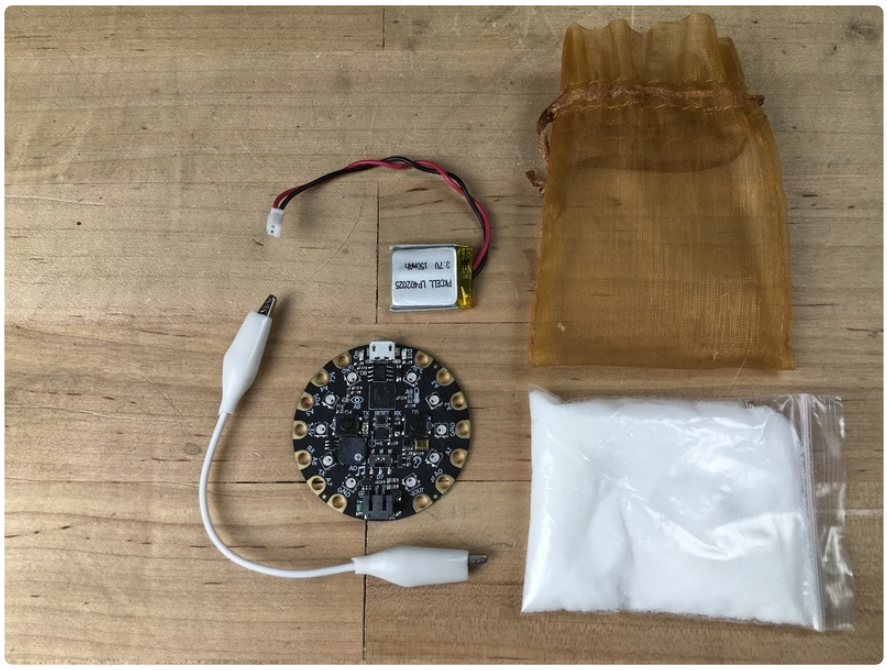
How can your fairy costume be complete without a glowing pixie dust bag?! It sparkles and blinks, and you can squeeze the touch sensor wire to change the colors on the beautifully twinkling NeoPixel ring! This fun, sophisticated-looking prop is easy to build with absolutely no soldering required.

This project updates and simplifies the classic GEMMA-based (https://adafru.it/xle)build by using the **Circuit Playground Express** (http://adafru.it/3333) -- what used to use multiple pieces of hardware is now built into a single board! Plus, all of the

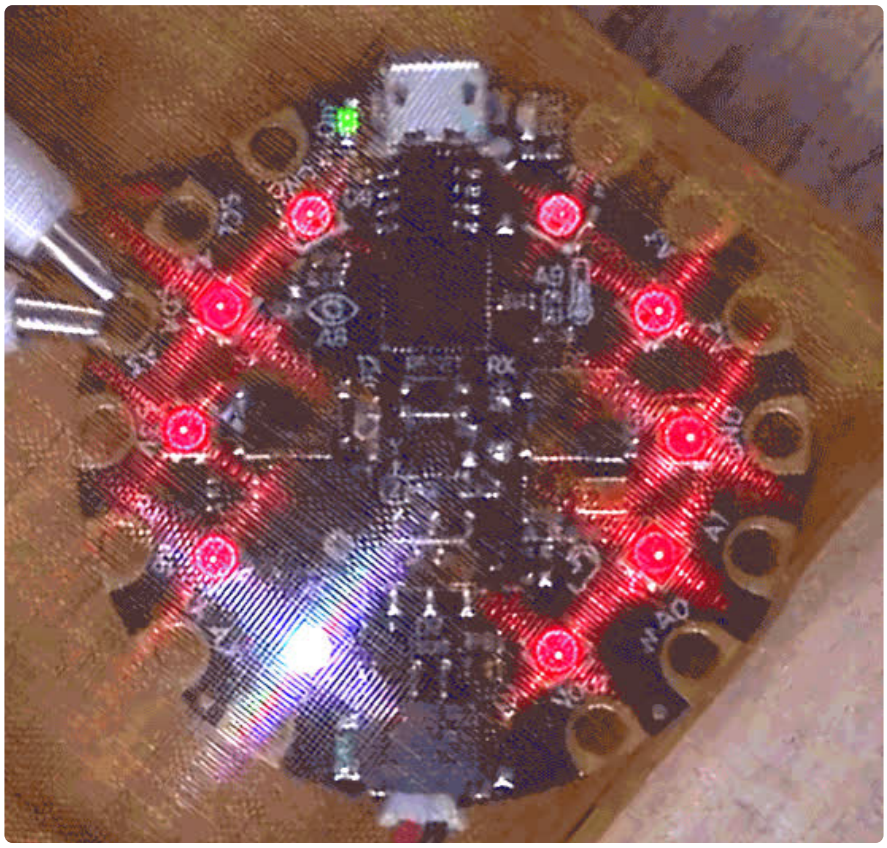
programming is done with the user-friendly graphical interface of MakeCode, so no need to type code or use the Arduino IDE if you don't want to.

Besides the Circuit Playground Express, you also need:

- 3 AAA battery pack with on/off switch, or
- 3.7V Lithium Ion battery with charger for a more compact build
- short wire alligator clip test leads
- translucent pixie dust bag
- small plastic bag
- flake salt, such as Morten's Kosher salt, used to as light diffusing "pixie dust"



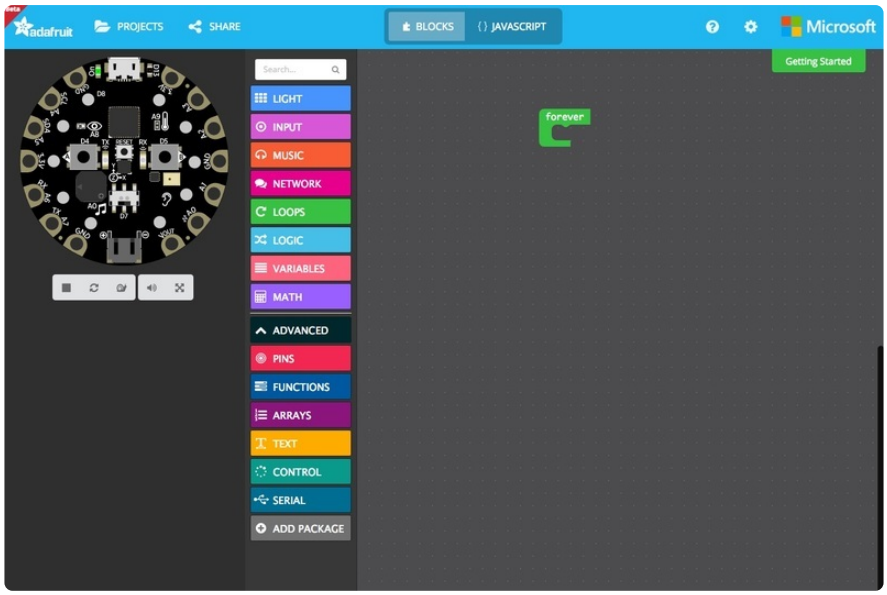
Code It



Your NeoPixie dust bag must twinkle beautifully -- what better way to do that than set the LEDs to a solid color and then randomly sparkle some of them white? You'll also want to be able to change the colors by tapping the capacitive touch sensor.

You can program all of these effects without typing a line of code, simply by dragging and dropping elemental code blocks, you can create Circuit Playground Express software in no time, and even test it out with the virtual Circuit Playground Express simulator built into the page.

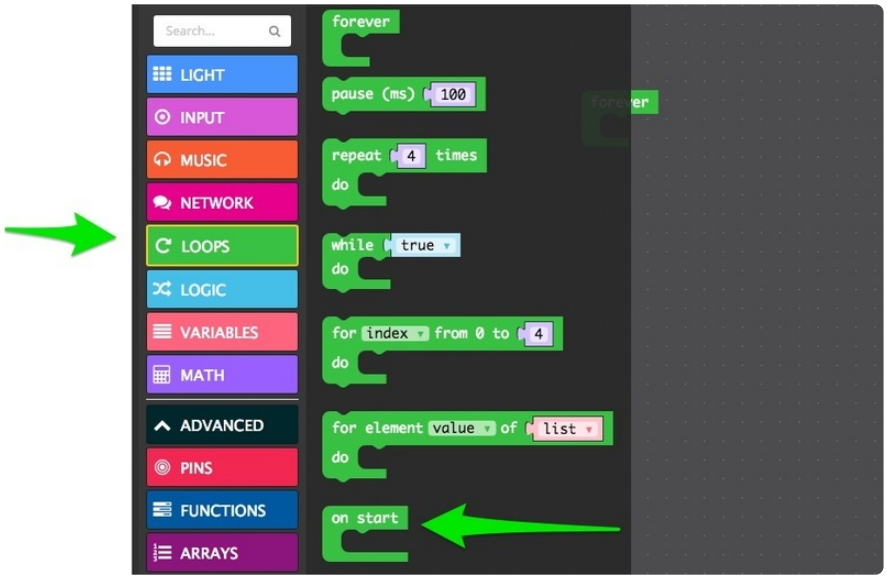
Be sure you've gone through the tutorial here on the basics of MakeCode (https://adafru.it/wB5) for Circuit Playground Express to make sure you know the basics of creating and uploading your code.



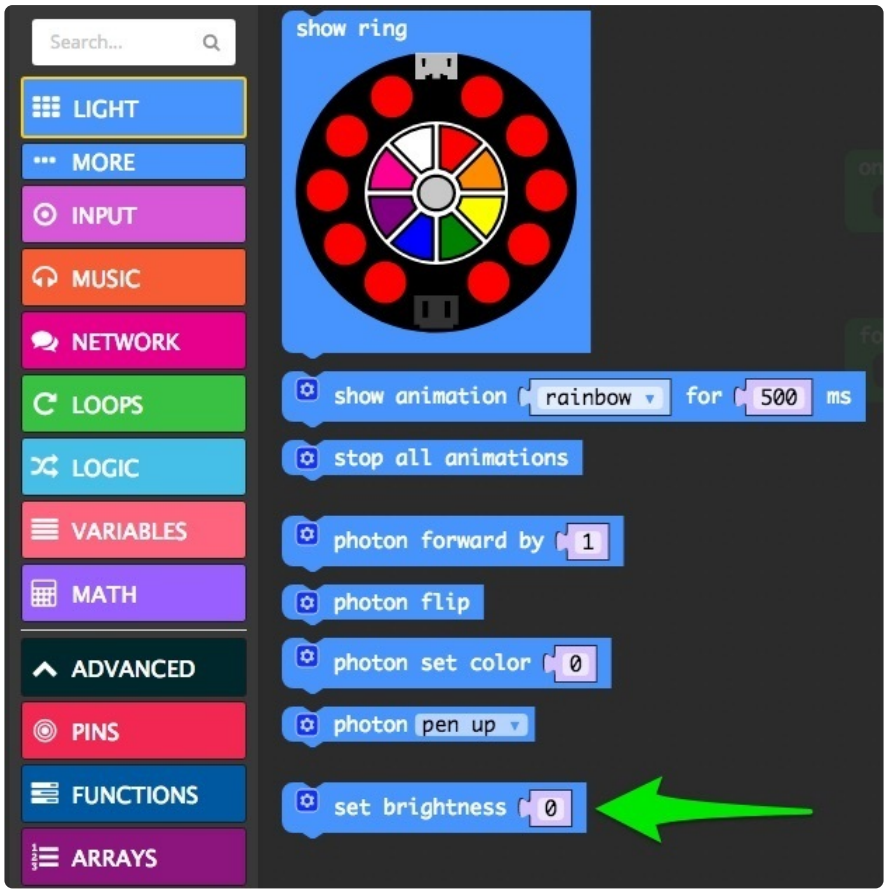
Setup

You'll begin with a nice clean canvas -- nothing but a single, empty **forever** loop block. This is where all of the repeating command blocks will live, running over and over and over again.

Let's also add an **on start** block to set one parameter from the beginning -- this is a good place for any commands that need to run once each time the Circuit Playground Express is turned on or reset. Think of it as a sort of initialization block.



From the **Light** category, add a **set brightness** block to the **on start** block and set the value to something moderately bright, say 80.

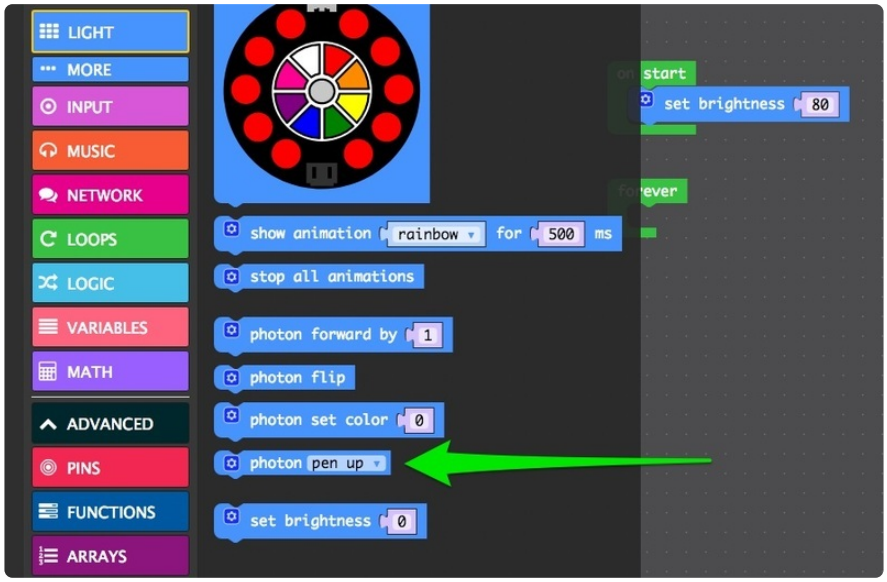


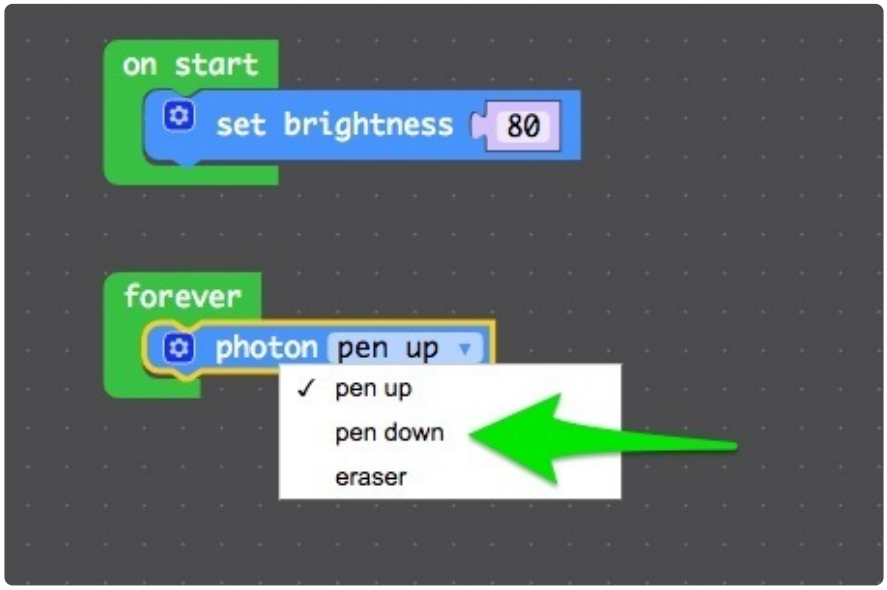
Animation

Next, you'll light up the NeoPixel ring using **photon** blocks.

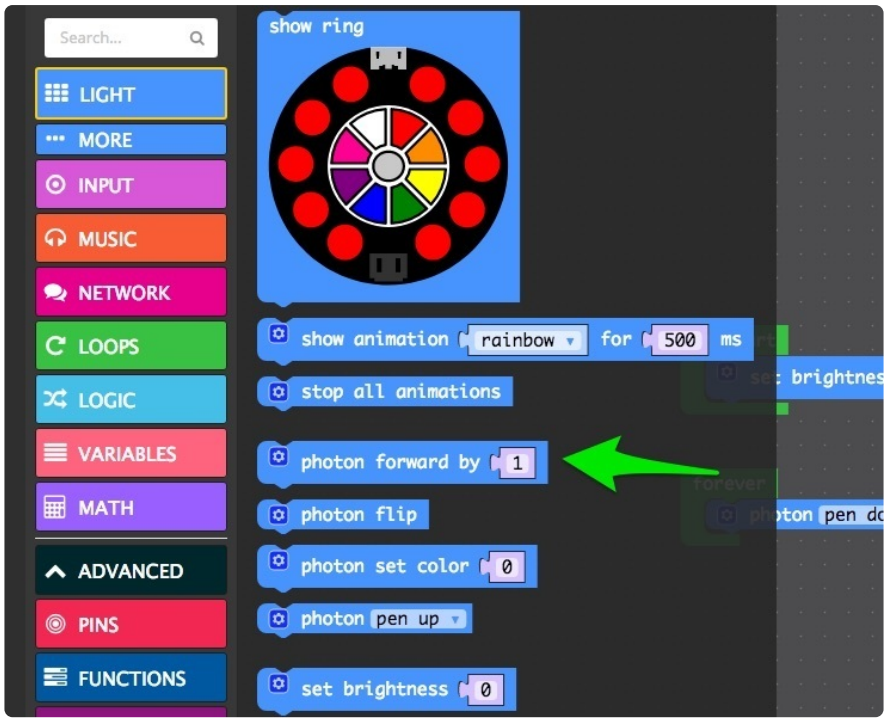
Photon blocks act like a sort of pen that can move to any position on the NeoPixel ring (0-9) and light or color a NeoPixel when it is "down". It doesn't change the color of any NeoPixels when it is "up", just like a pen that is lifted off the paper.

From the **Light** category, grab a **photon pen up** block and place it into the **forever** loop. Then, change the dropdown to **pen down**.

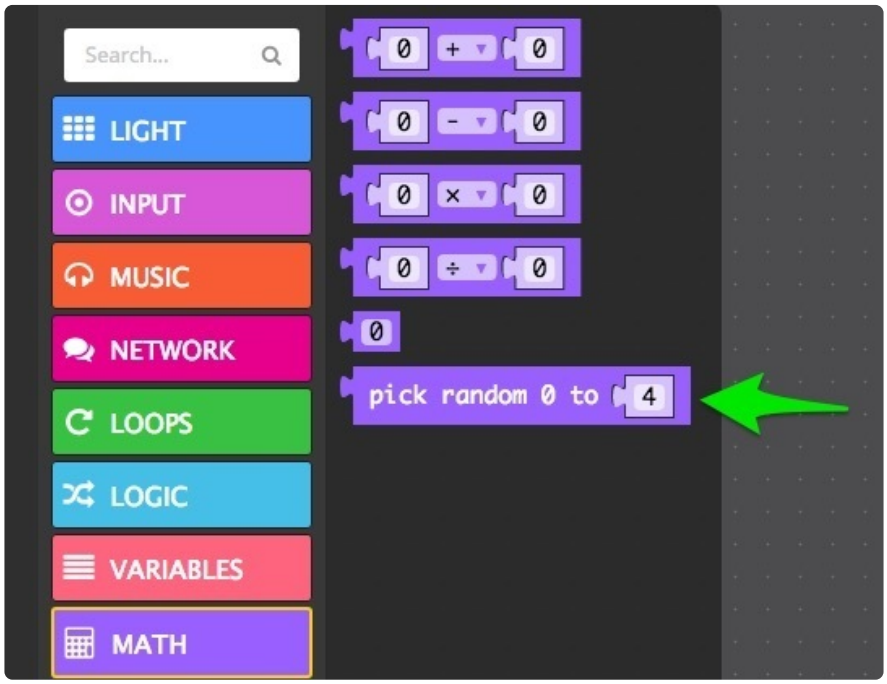


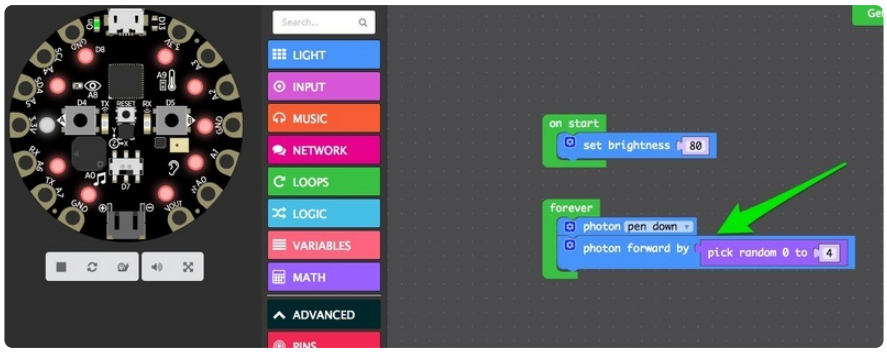


You'll want to light up and color your NeoPixels by moving the photon pen around while the pen is down, so add a **photon forward by 1** block to the **forever** loop.

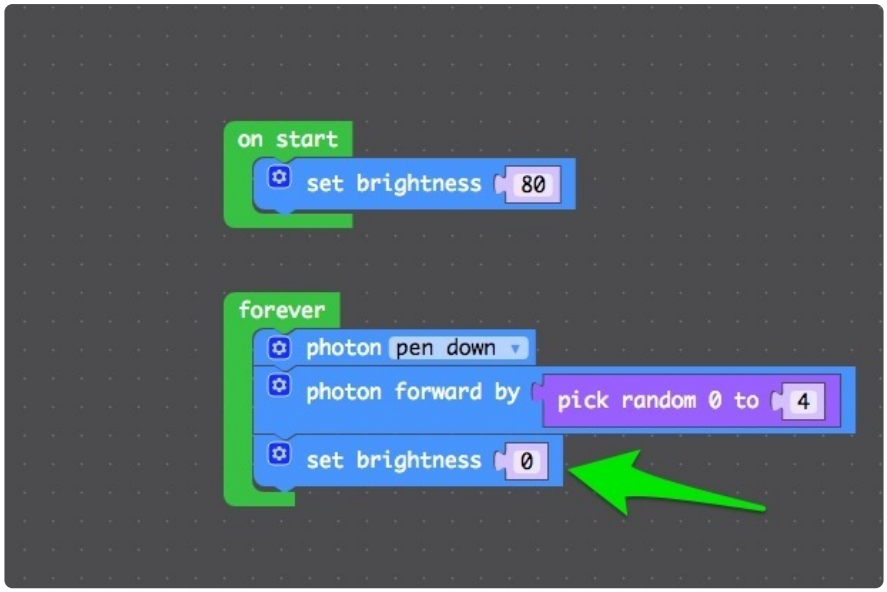


To create the organic, irregular sparkle pattern, you'll have the photon pen move a random number of positions (between 0-4), by adding a **pick random 0 to 4** block to the **photon forward by** block.

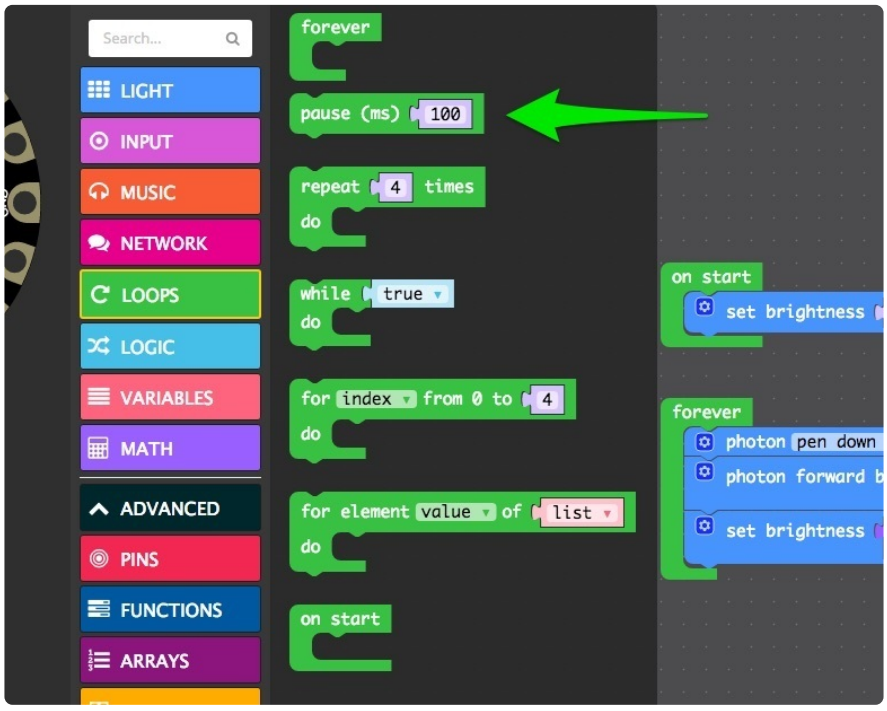




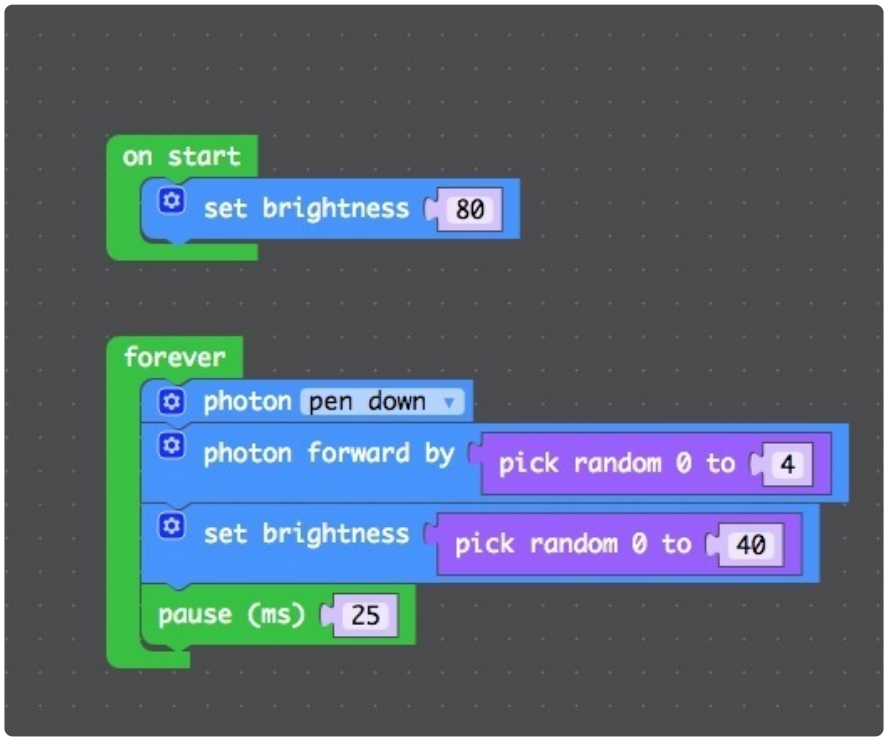
You'll do the same sort of trick to vary the brightness, using the **set brightness block** and another **pick random** block, with the value range set from **0 to 40**



To slow things down just a touch, add in a slight pause with the **pause (ms)** block.



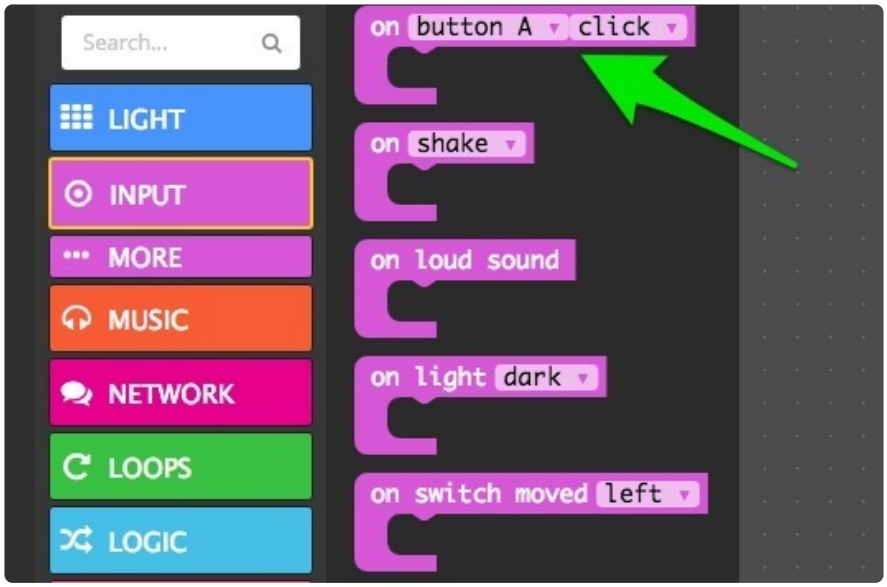
Here's what the animation now looks like -- go ahead and test it out!



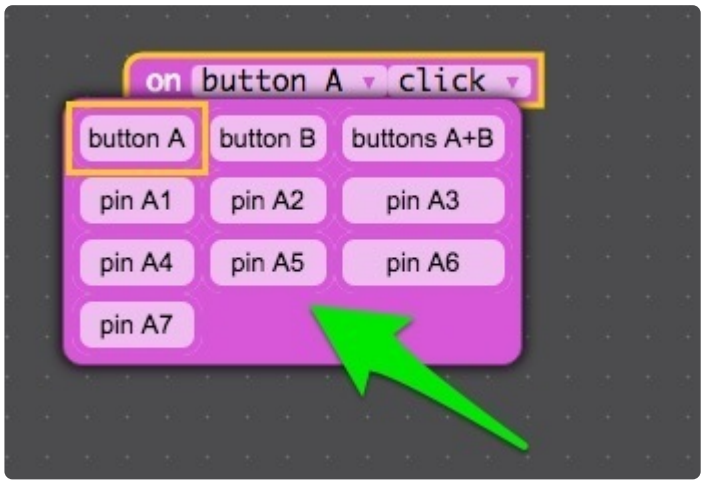
Color Touch

By default the LEDs will light up red. Red is nice, but how about some other colors, too, when the mood strikes? You'll now add touch input that allows one of the capacitive touch pads of the Circuit Playground Express to switch the color values every time you tap it.

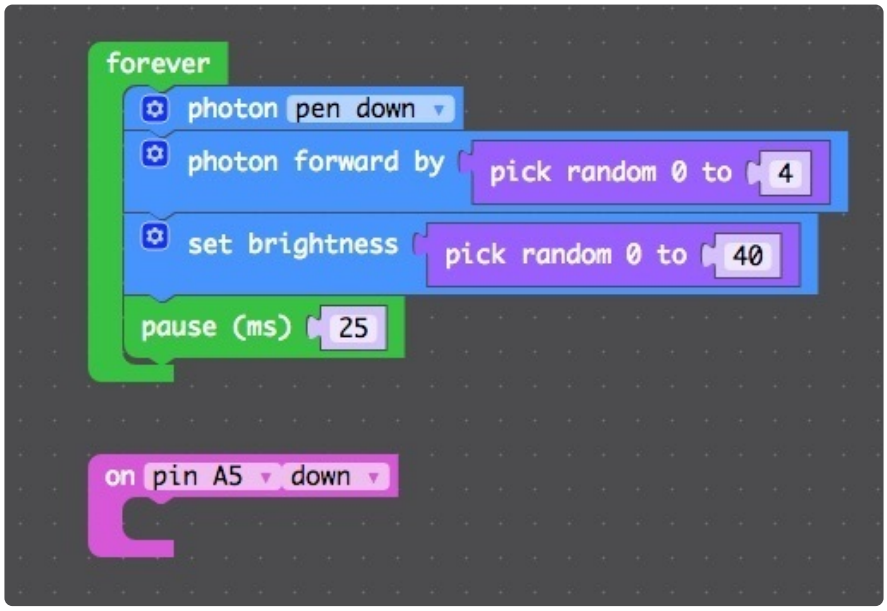
From the **Input** category, add an **on buttonA click** block.



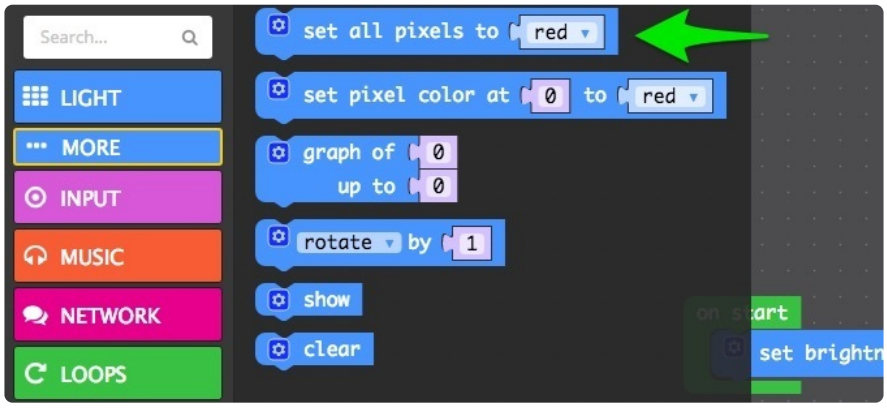
Change the input to one of the pin pads, in this case **pin A5**. (Any of the seven pin A pads will work as capacitive touch pads.)



Change the input mode to **down** instead of **click**. This means the input condition is met the moment you touch the pad, not when you touch and release.



Add a **set all pixels to red** block from the **light** category to the **on pin A5 down** block.

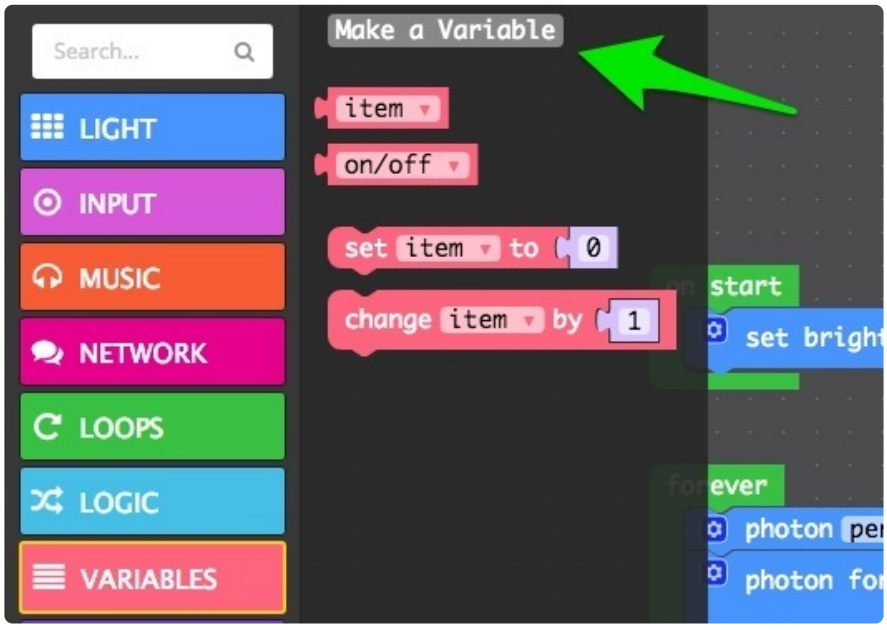


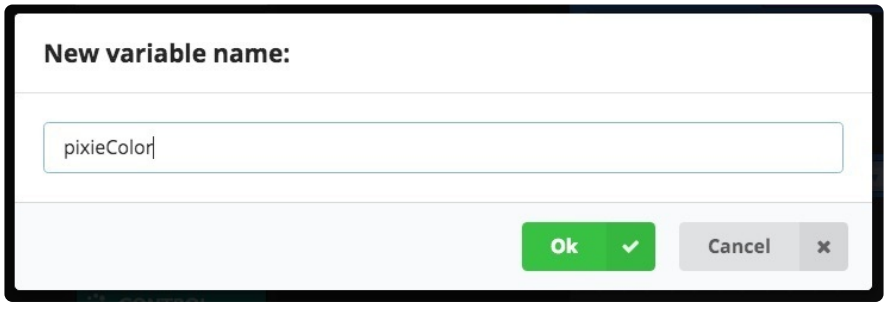
To make the ring flash white when the pad is touched, change the dropdown from **red** to **white.**

Variable Color

To change the color of the photon in the **forever** block, we'll use a variable. This variable, called "pixieColor", will store a value that changes each time you touch the capacitive pad, and the value will be used by the **photon set color** block.

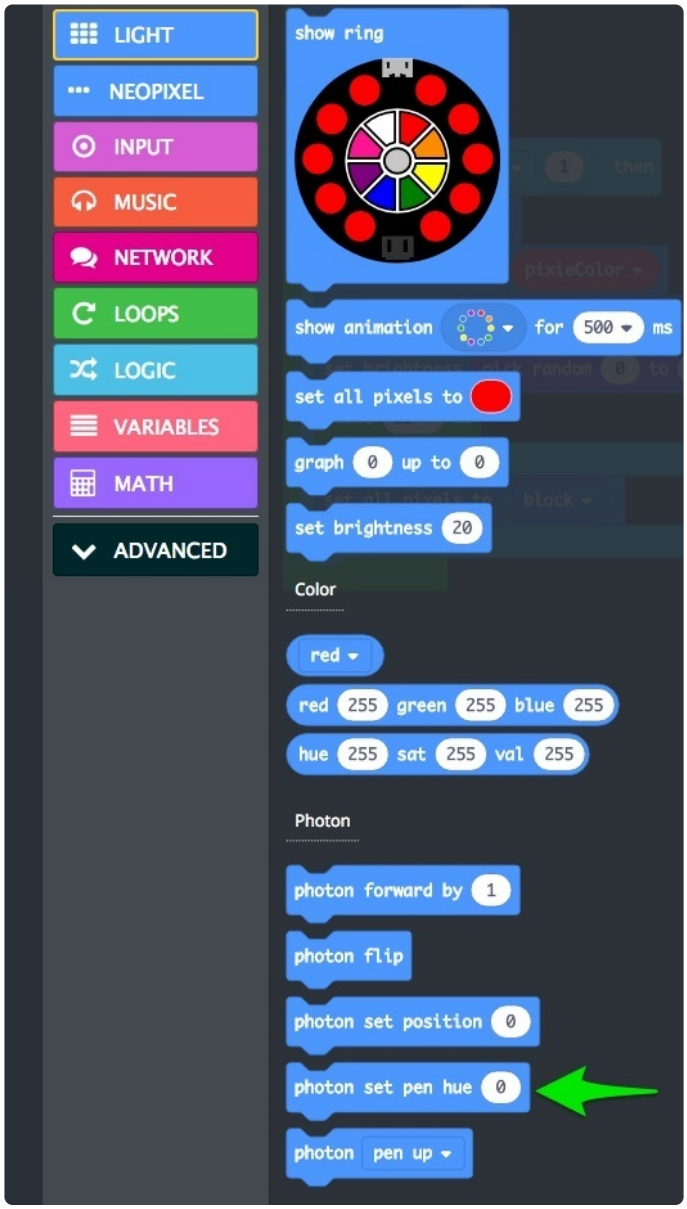
In the **Variables** category, click the **Make a Variable** button, then type the name **pixieColor** and click **Ok.**



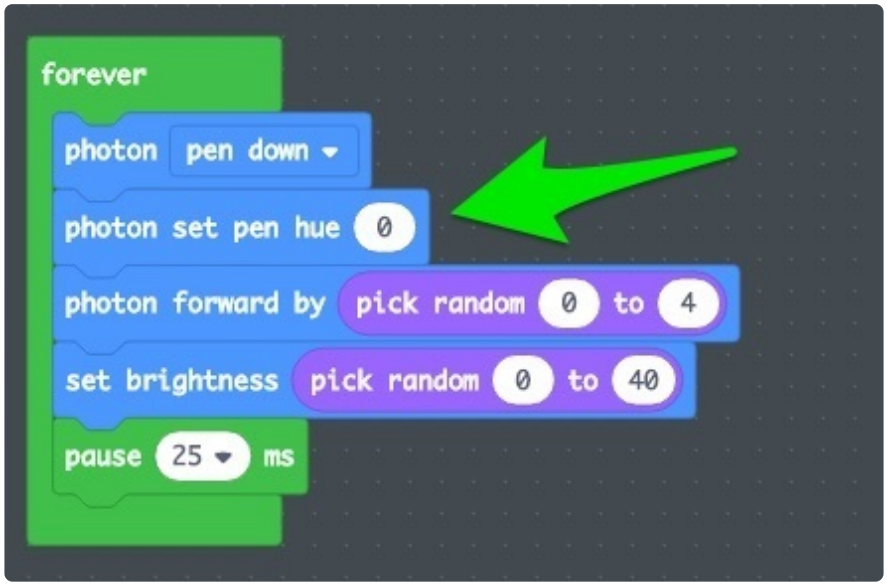


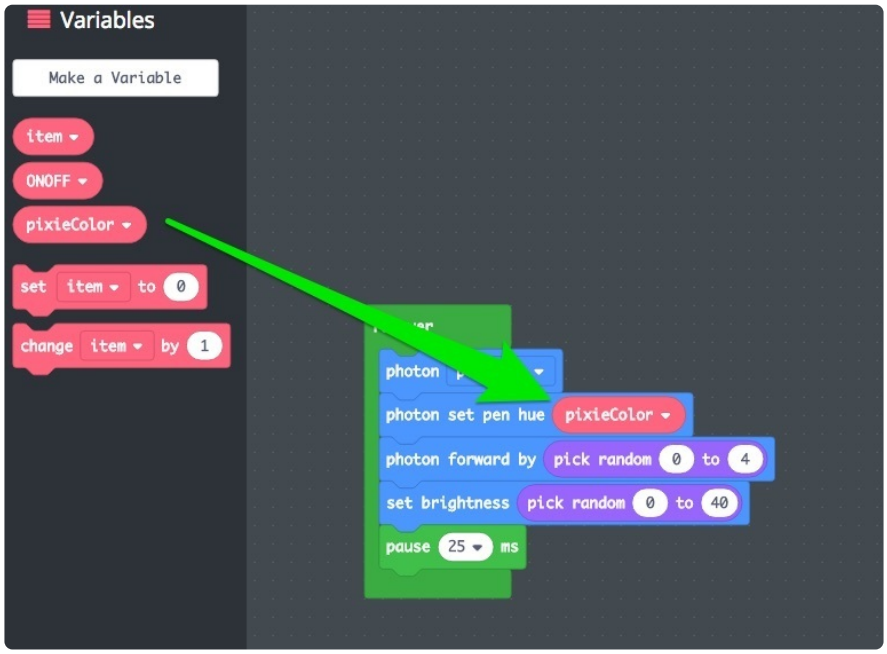
Drage the **change item by 1** block into the **on pin A5 down** block, then change the dropdown to your **pixieColor** variable. So that the color change is more dramatic, set the change value to 50.

Add a **photon set pen hue** block to the **forever** loop, just under the **photon pen down** block.



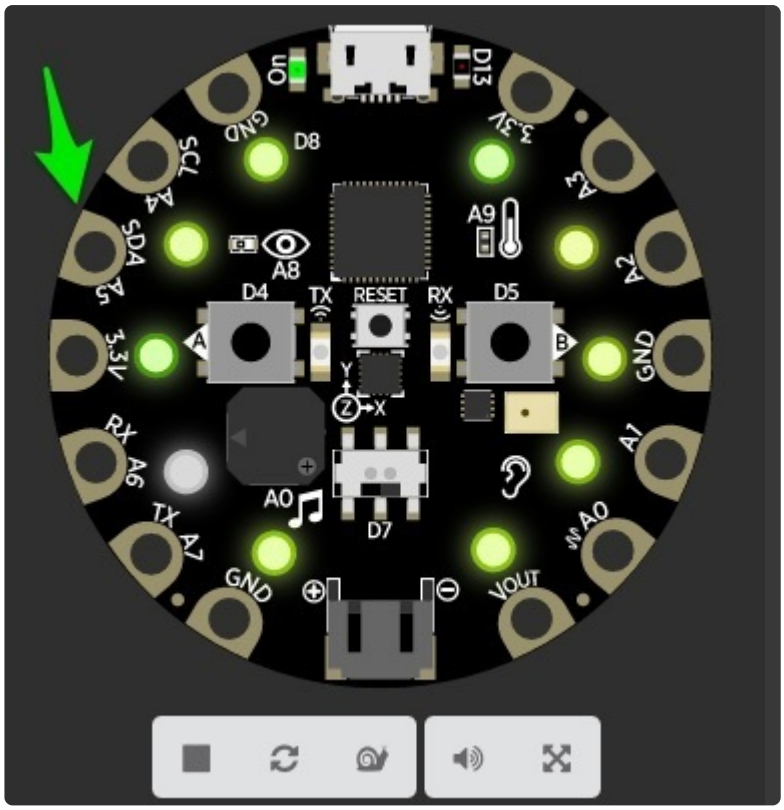
Instead of the default 0 value, drag an instance of the **pixieColor** variable block on to the **photon set pen hue.**

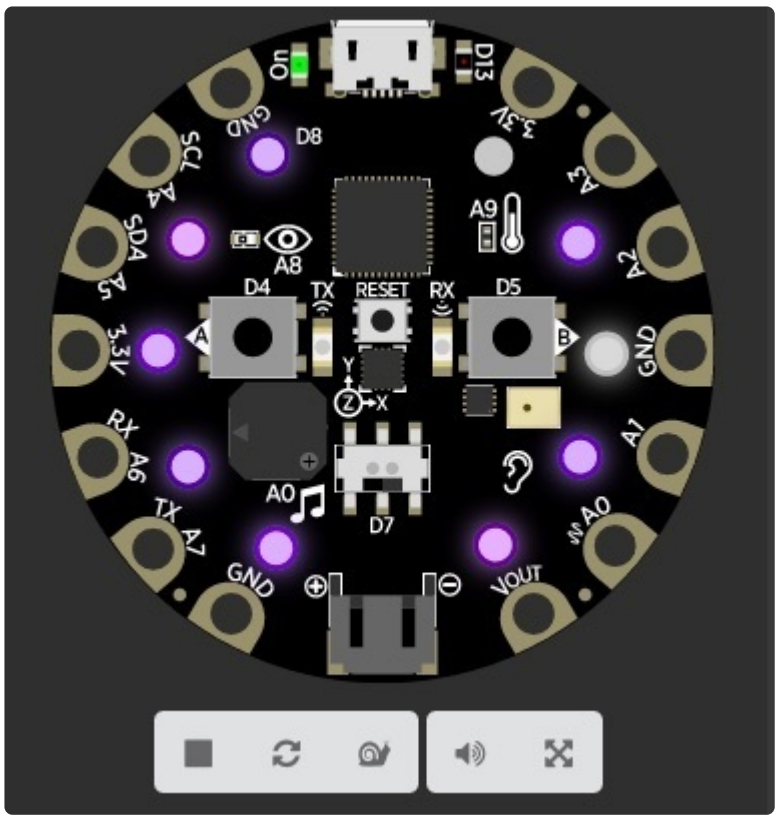


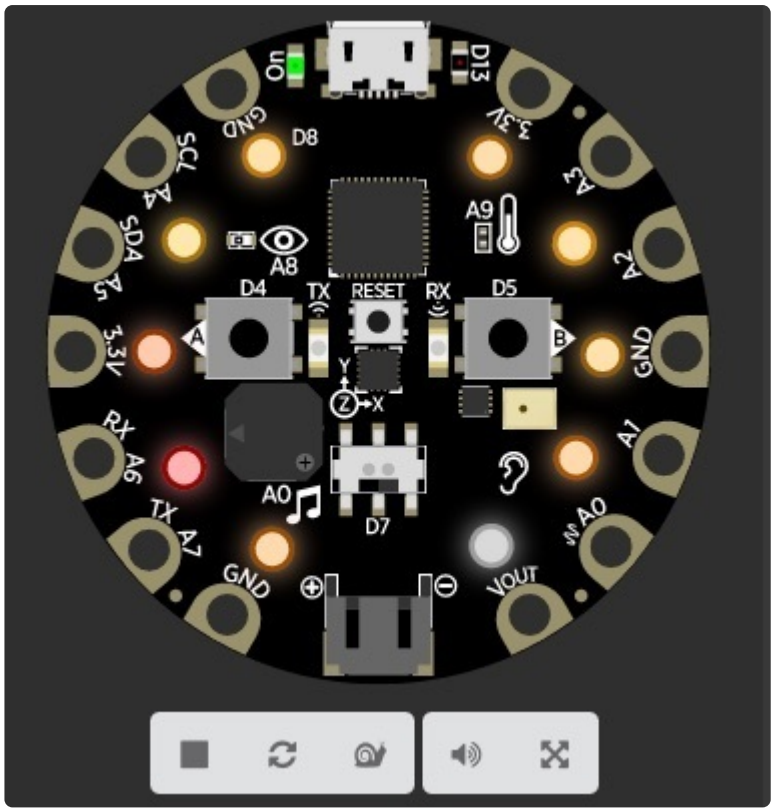


Simulate It

Test out the color change now by clicking the gold pad on the Circuit Playground Express simulator at pad A5. You'll see the color change each time you click it.


ON
D13
GND
D8
3.3V
SCL
A4
A3
SDA
A5
A8
A9
A2
D4
TX
RESET
RX
D5
3.3V
A
B
GND
Y
Z
X
RX
A6
A1
A0
D7
TX
A7
A0
GND
VOUT

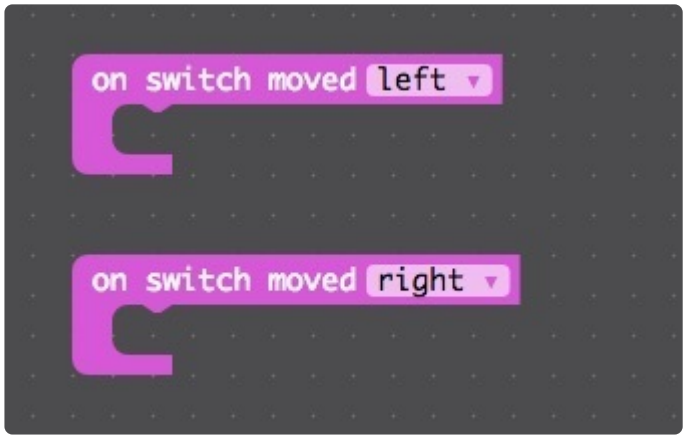

ON
D13
GND
D8
3.3V
SCL
A4
A3
SDA
A5
A8
A9
A2
D4
TX
RESET
RX
D5
3.3V
A
B
GND
Y
Z
X
RX
A6
A1
A0
D7
TX
A7
A0
GND
VOUT



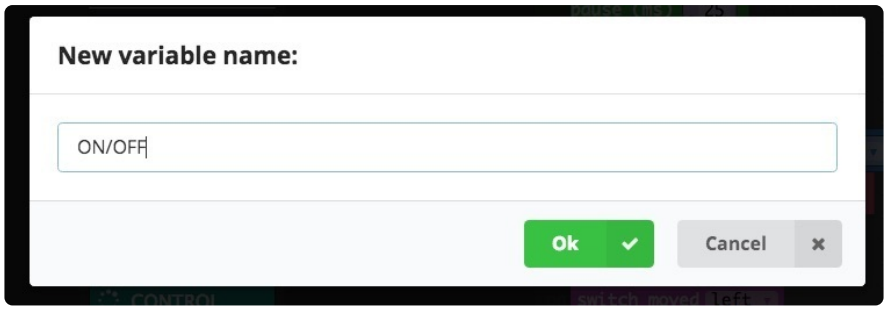
Pseudo ON/OFF

You may not always want your NeoPixie Dust Bag to sparkle, but it isn't convenient to unplug the battery all the time, so let's use the on-board slide switch to turn off the LEDs.

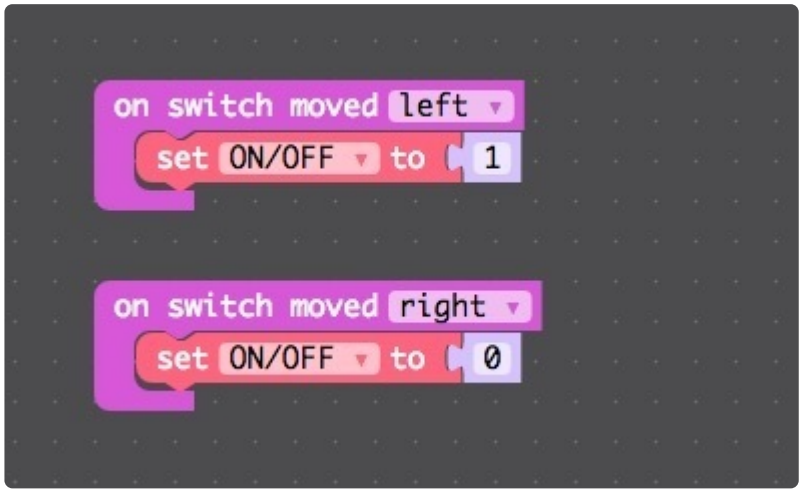
Grab an **on switch moved left** block from the **Input** category, then duplicate it and set the second one's dropdown to **right.**



Like you did before, create a new variable, this time naming it "ON/OFF".



Drag a **set item to 0** block into each switch block, and then change their variable dropdown choices to **ON/OFF.** The left position will be ON, so set the value to 1, leaving the right position to 0.



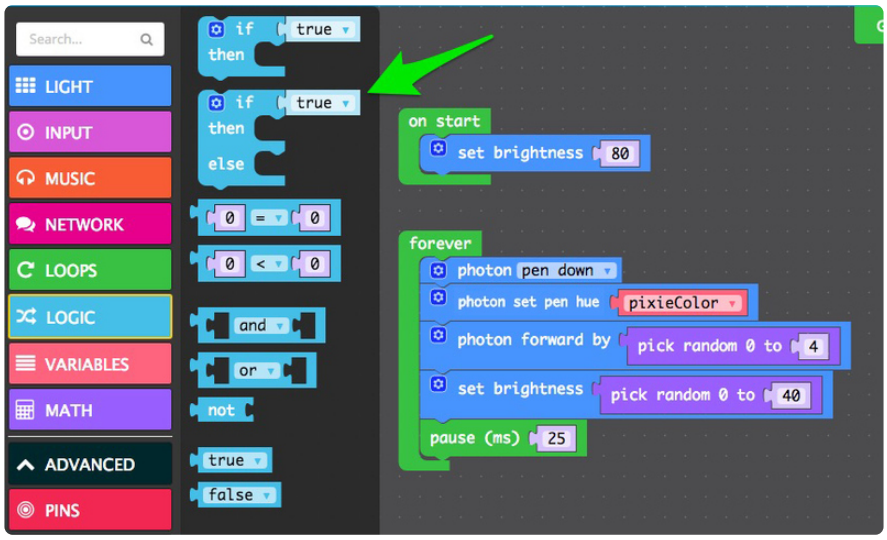
Conditional If/Then/Else

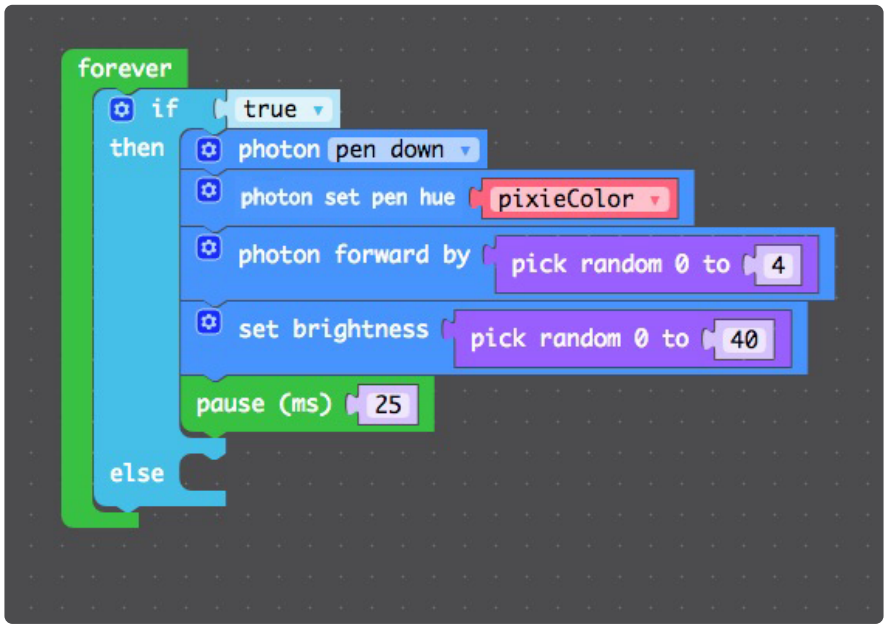
So you now have a variable named ON/OFF that changes depending on the slide switch position. But how do you actually use that to control the lights? The answer is with a conditional **if then else** block.

This block does two things: it tests if a certain condition is met, and then it executes the proper set of blocks depending on that answer.

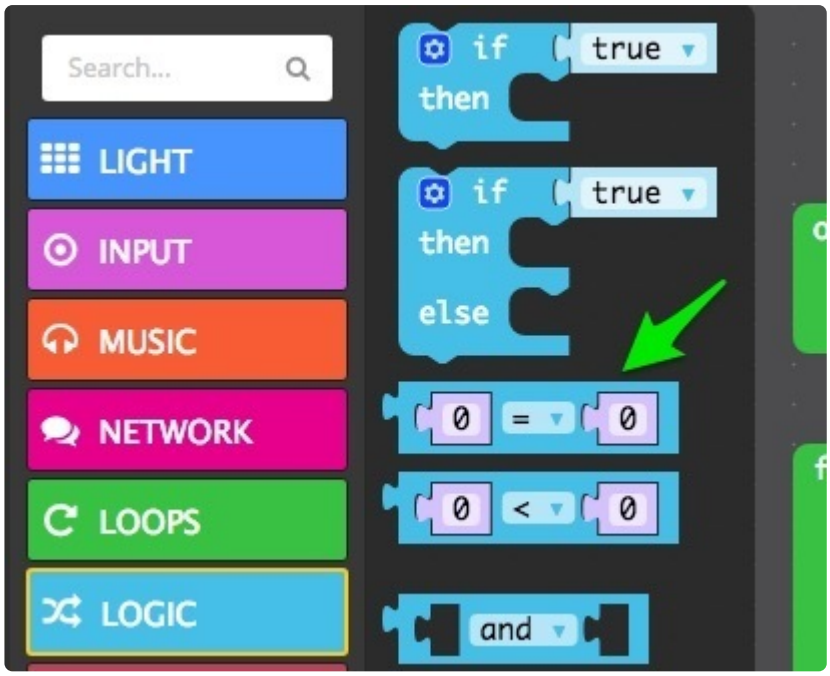
In our case: **if** the variable **ON/OFF** is equal to 1, **then** play our animation blocks, **else,** set all the pixels to black.

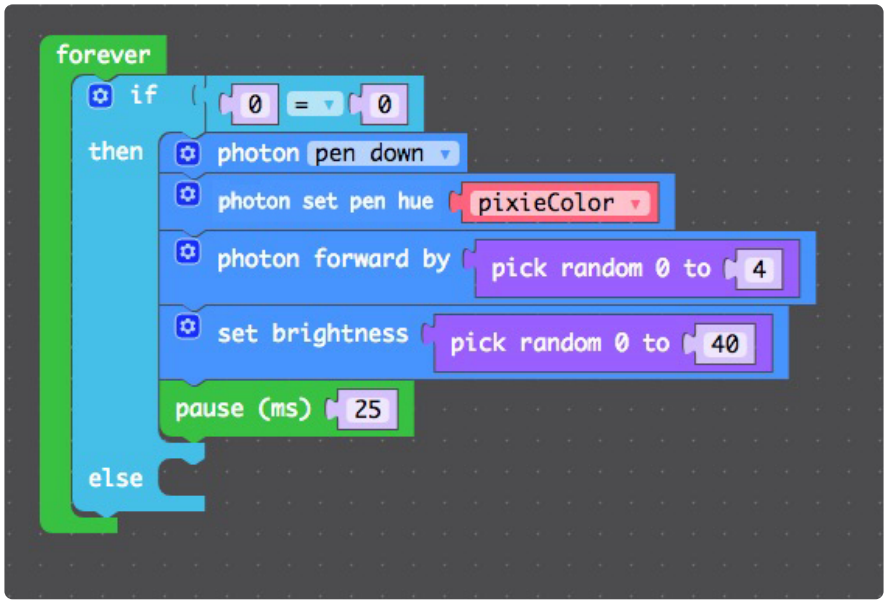
From the **Logic** category, get an **if then else** block and place it in the **forever** loop. Then place all of the animation blocks inside the **then** section.



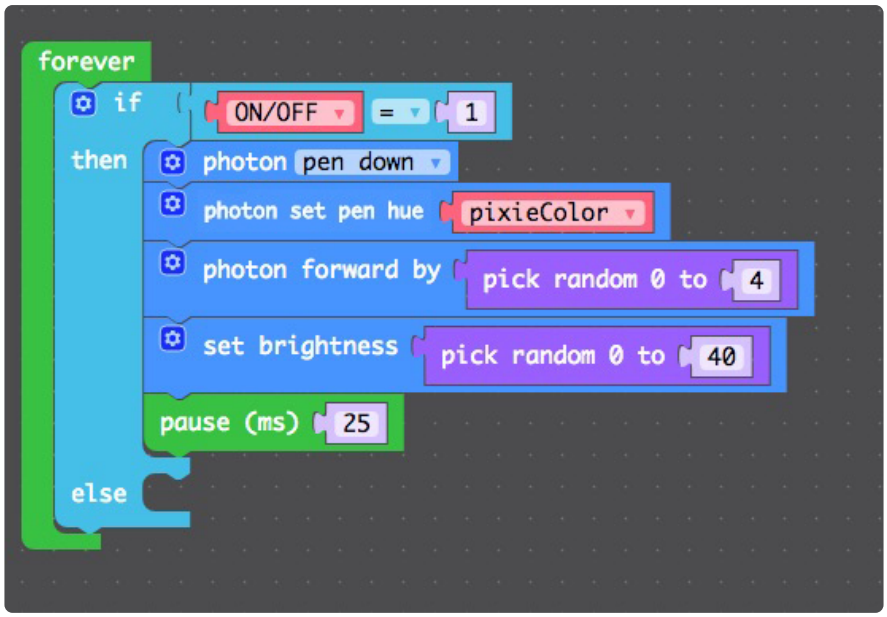


To test our condition, we'll need to get a **0** = **0** block from the **Logic** category and place that at the end of the **if** statement.

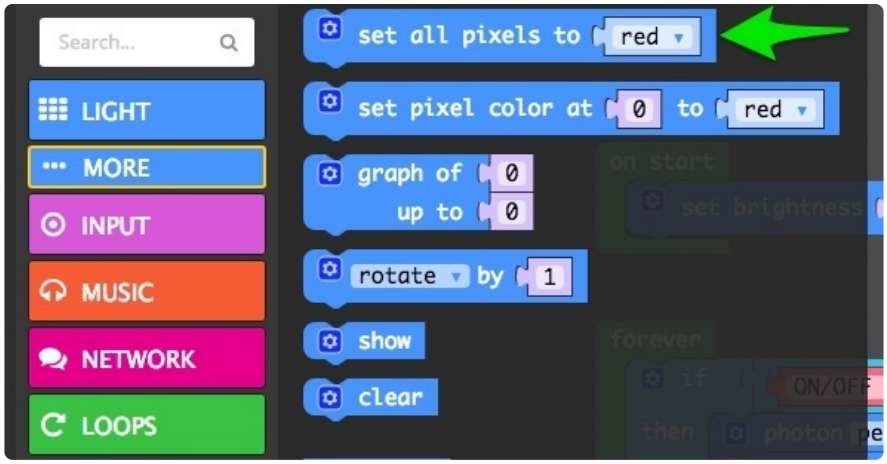




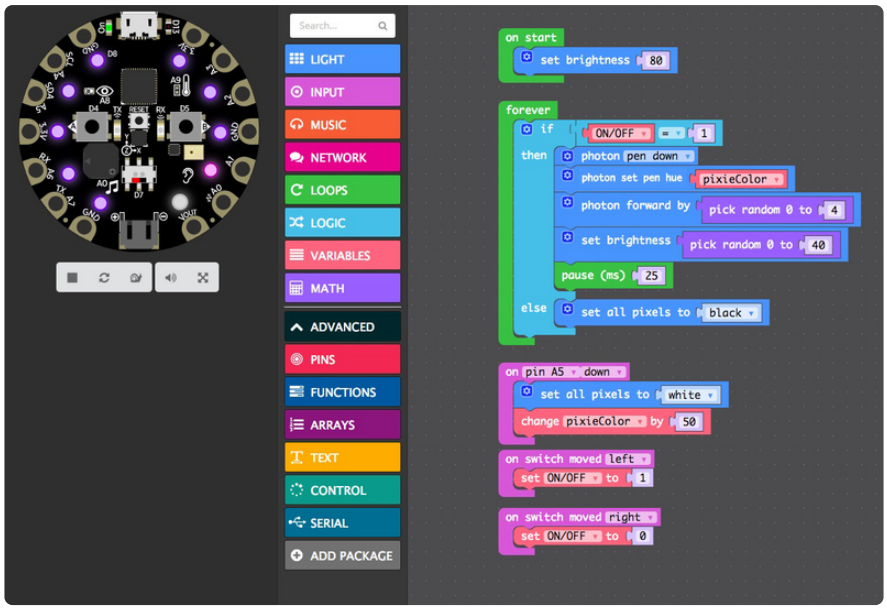
Place an **ON/OFF** variable into the first slot, and change the second slot to **1.** This means now that when the slide switch is to the left, the variable will be equal to 1 and the animation will play.



Now, put a **set all pixels to** block inside the **else** section, and change the dropdown to **black.**

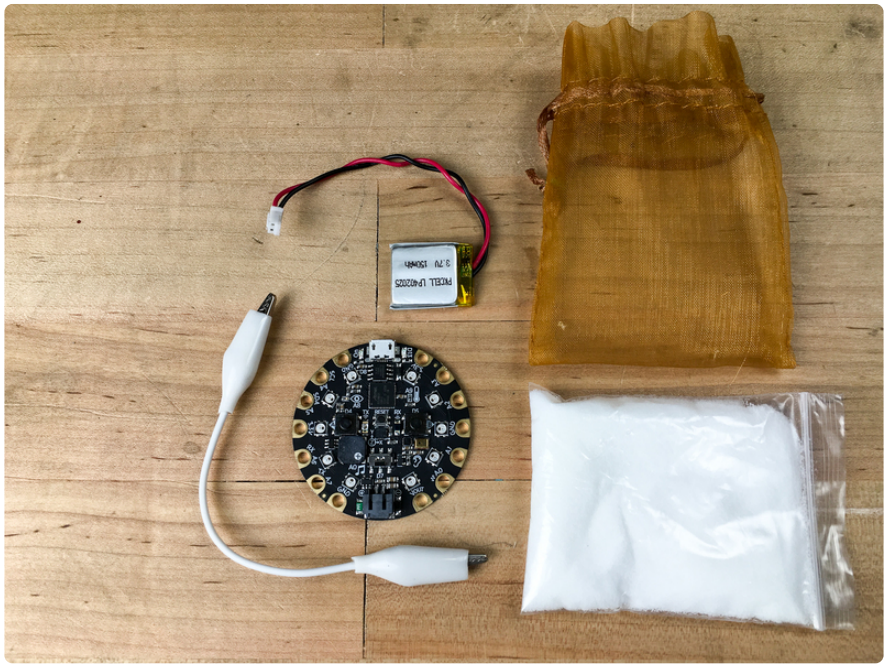


You finished programming your NeoPixie Dust Bag! Upload it to the Circuit Playground Express and then we'll move on to dressing it up with a battery, bag, and capacitive switch.



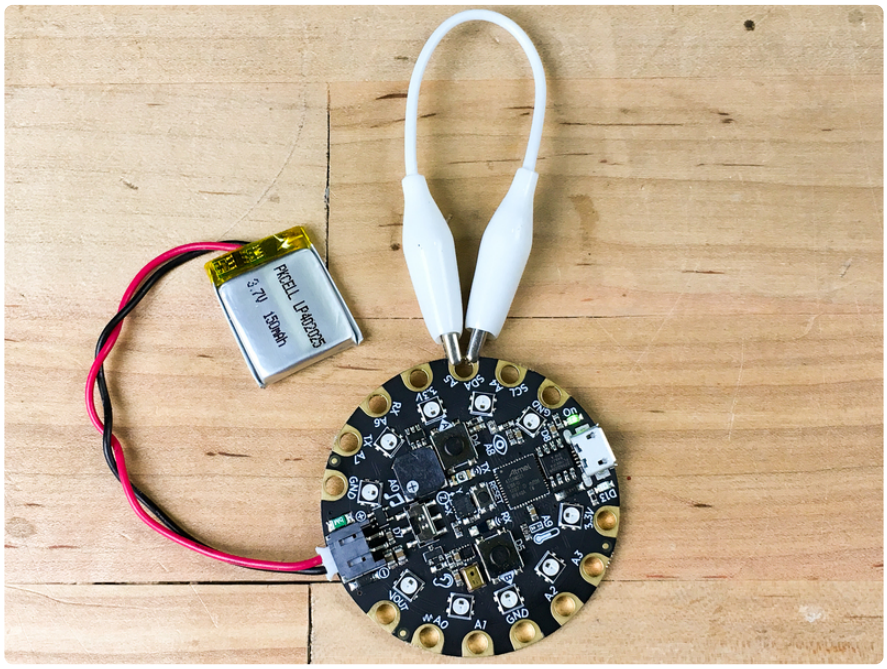
Here's what the finished code will look like -- feel free to try it out in the simulator (flip the slide switch, then click the A5 pad) and then download it to your Circuit Playground Express and give it a whirl!

Build and Use It

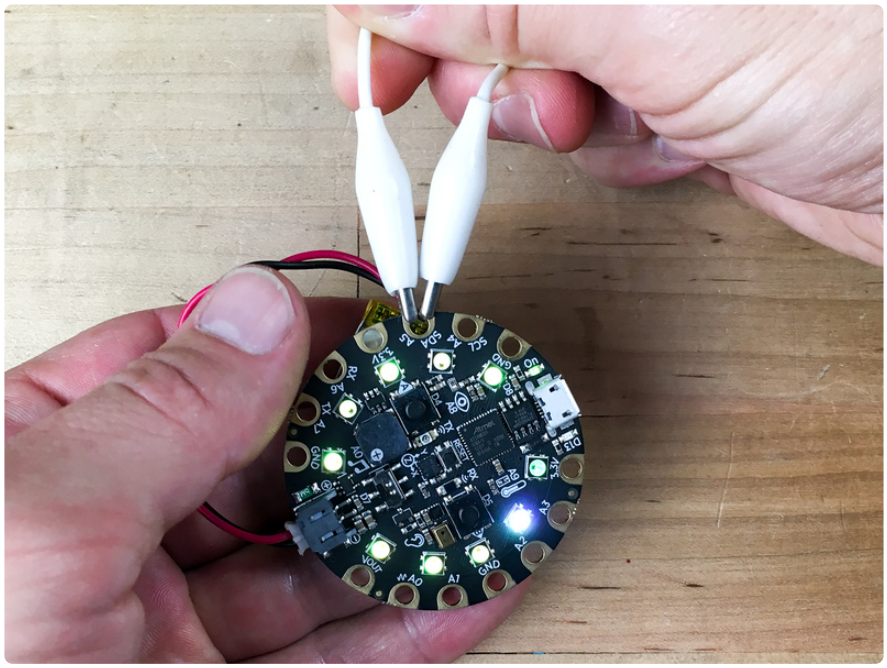


Touch Sensor

To make it easier to change colors when the Circuit Playground Express is in the bag, extend the Pin A5 sensor with a loop of wire. You can simply clip the alligator lead to the pad on both ends as seen here.



Plug in your battery, make sure the slide switch is set to the left, and then squeeze the wire while holding the board in order to trip the sensor and change colors.



Bag in a Bag

You'll use the salt to diffuse the light and make the bag look like it's full of pixie dust. To keep the salt off of the electronics, place them inside the plastic bag, then place all of that into the pixie dust bag.

Now, fill the pixie dust bag with salt. Instant pixie dust!

You can hold the bag's wire loop and then squeeze the pixie dust to change the color! Your capacitance is measured by the on-board sensor, and it works best when you have a hand near the board and the other squeezing the wire.

Now go off and spread magic -- maybe you can even teach someone to fly!