



Crickit Controlled Animatronic Eyeball

Created by John Park



<https://learn.adafruit.com/crickit-controlled-animatronic-eyeball>

Last updated on 2024-06-03 02:23:29 PM EDT

Table of Contents

Overview	3
<hr/>	
• Materials	
Make the Eye and Mechanism	4
<hr/>	
• Eyeball	
• Micro Servo	
• Bracket	
• Full Size Servo	
Crickit Circuit	17
<hr/>	
Code with MakeCode	18
<hr/>	
• Crickit Extension	
• Servo Zeroing	
• Eyeball Range of Motion	
• Eye Darts	
• Interactive Tilt Sensing	
• Going Beyond	

Overview

Make your own animatronic eyeball that can look around -- left/right and up/down -- to add to your costumes and props, or simply as a (slightly unnerving) decoration!

With a pair of servo motors and a Crickit board paired with Circuit Playground Express, you can control the animatronic gaze with pre-programmed movements, random glances, or even interactive control.

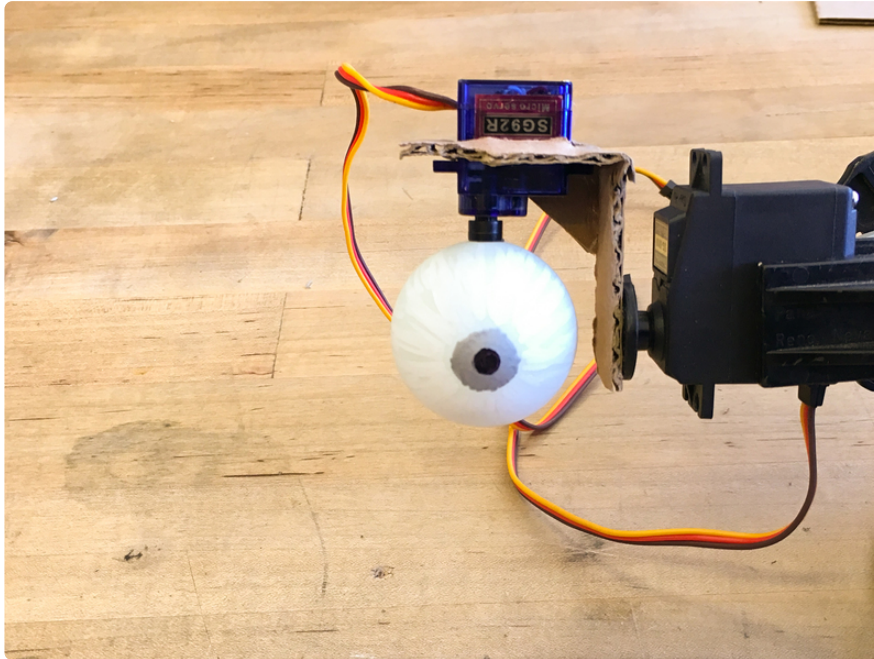
1 x Adafruit CRICKIT For Circuit Playground Express	https://www.adafruit.com/product/3093
1 x Circuit Playground Express Incredibly awesome microcontroller board	https://www.adafruit.com/product/3333
1 x Standard servo 180 degree rotation servo	https://www.adafruit.com/product/155
1 x Micro Servo Tiny 180 degree rotation servo	https://www.adafruit.com/product/169
1 x Waterproof 3xAA Battery Holder with On/Off Switch	https://www.adafruit.com/product/771
1 x Male DC Power adapter 2.1mm plug to screw terminal block	https://www.adafruit.com/product/369
1 x Alkaline AA batteries 3 pack	https://www.adafruit.com/product/3521

Materials

For this project, in addition to the above parts, you'll also need:

- Ping pong ball
- Corrugated cardboard
- Hot glue and glue gun or white glue
- Mask or other prop to contain the your eyeball and mechanism
- Markers or craft paint for eyeball decoration

Make the Eye and Mechanism

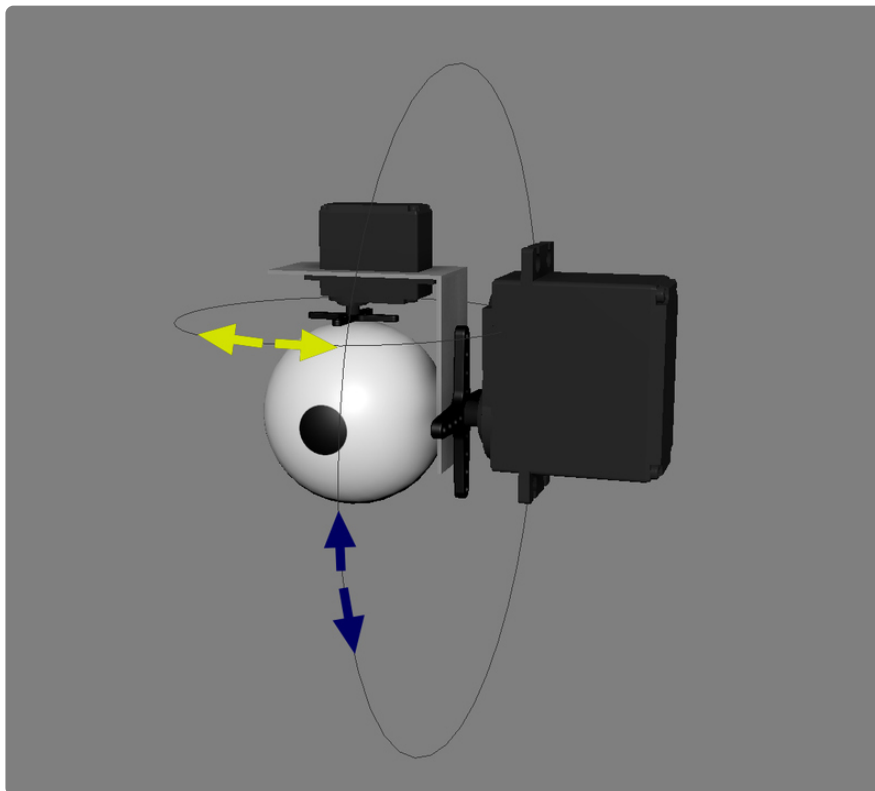
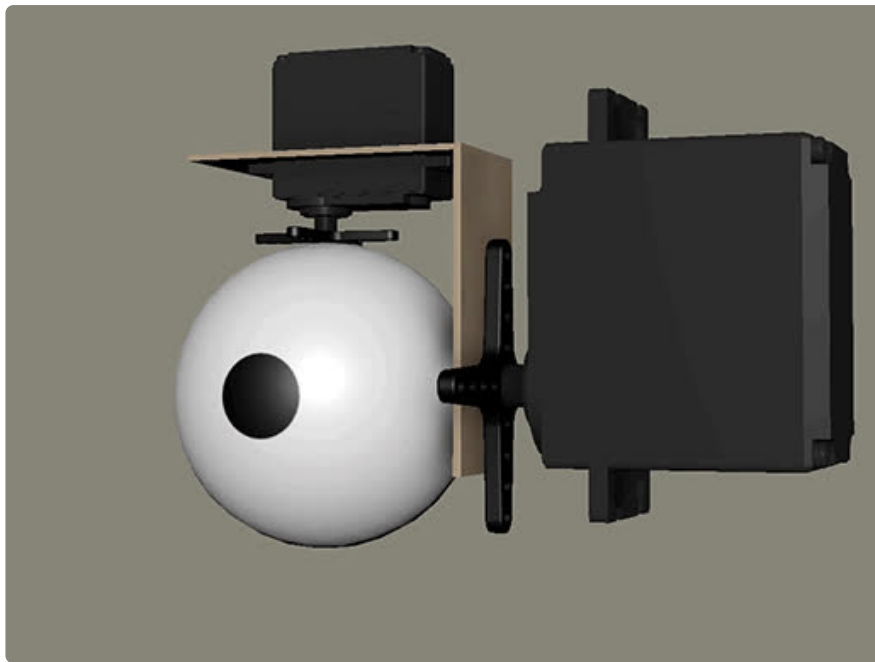


We'll make a two-axis gimbal to enable the eye to look left and right, as well as up and down, and a mix of these positions.

There are many interesting, clever, and sometimes complicated ways to build animatronic eyes. You can use fixed servos with linkages, and even multiply the linkage layout to drive two eyes simultaneously. But for this project, we'll keep it simple.

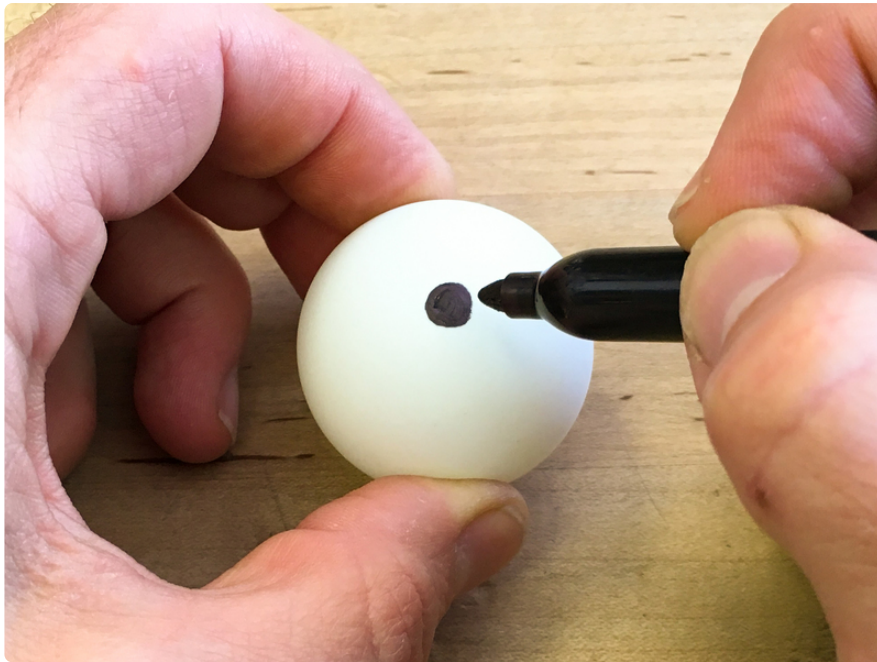
We'll connect the eyeball directly to the micro servo used for left/right movement, and then connect the micro-servo to the full-size servo. This means we don't need any complex or fiddly linkages. The only trick is to get the center of rotation of each servo to match the pivot of the eyeball on both axes.

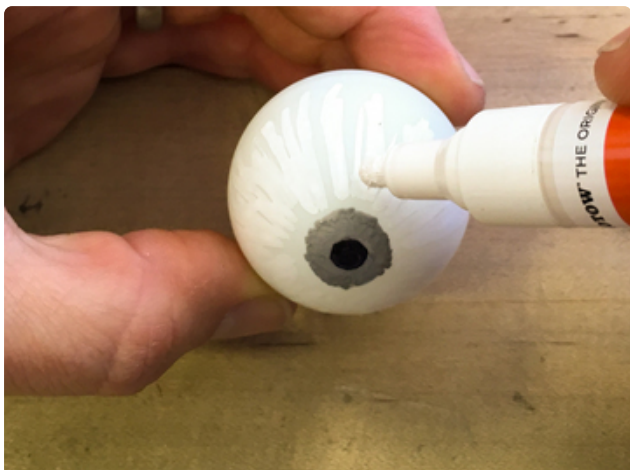
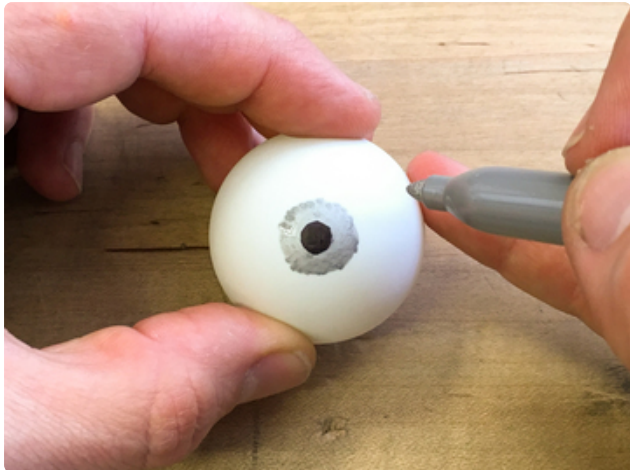
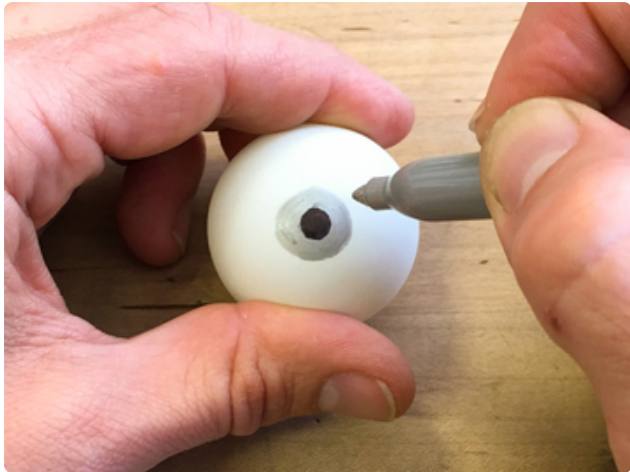
We can do this by connecting the micro servo to the full sized servo via a small bracket made of cardboard! You can see how this works in the diagram below.



Eyeball

First, we'll prepare the eyeball. You can start with a white ping pong ball and use markers to add a pupil and iris. A white paint marker makes the sclera look a bit more realistic. You can even add some veins with a thin red marker!



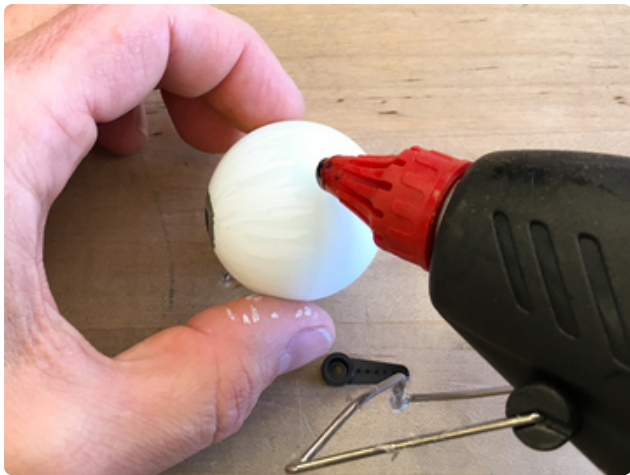
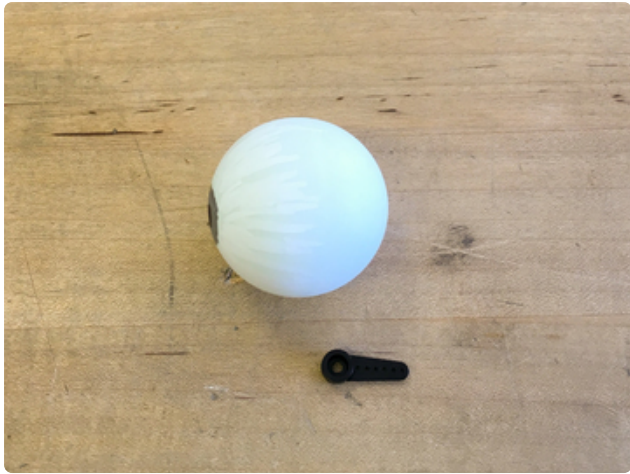


Start with a black marker for the pupil
 Add a lighter colored iris
 You can add subtle streaks in the iris
 White paint or paint marker can be used
 for extra detail in the sclera



Micro Servo

You can use hot glue to attach the top of the ping pong ball to the micro servo. Pick a servo horn and glue it in place, then place the horn on the splined shaft when it is dry.

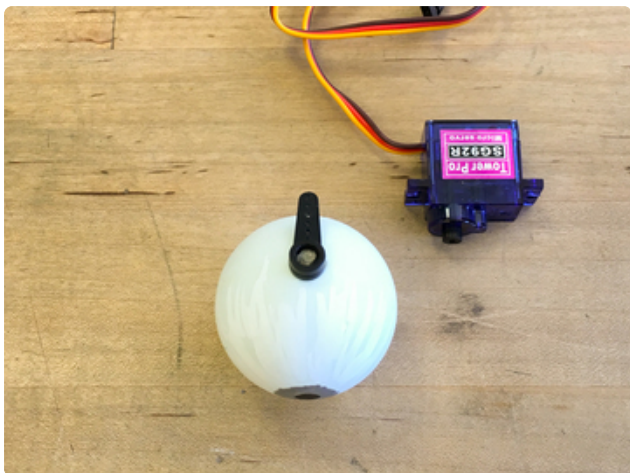
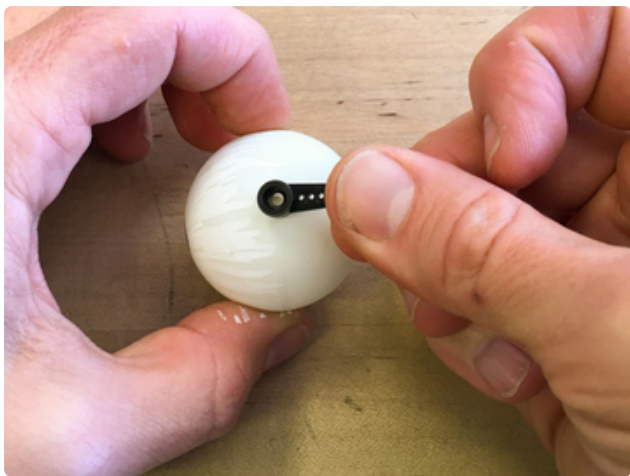


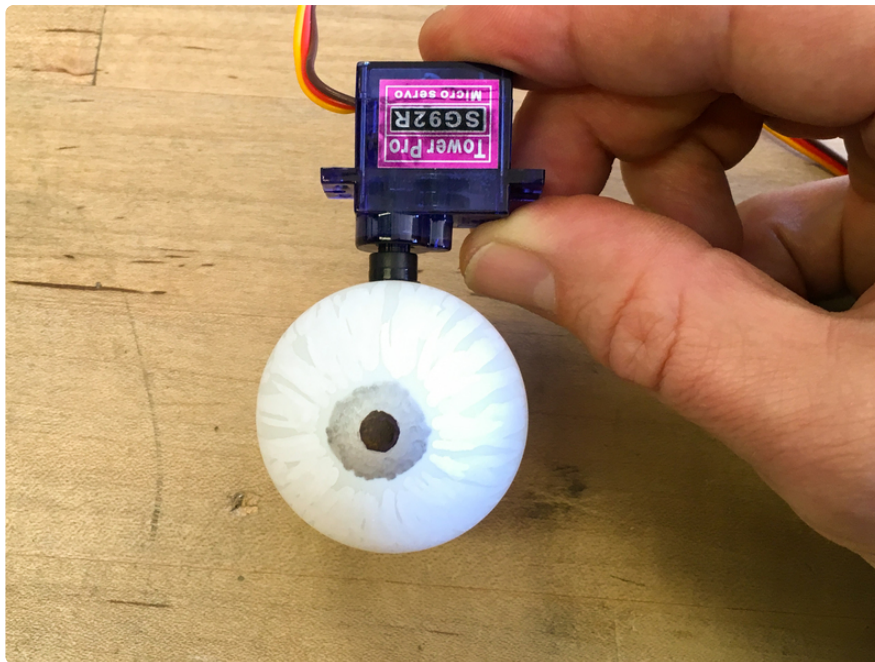
Choose a point at the top pole of eyeball and place a dab of hot glue or white glue -- don't use too much glue or it will fill up the horn's center and it won't fit over the servo shaft

Set the servo horn on the glue so that the larger, open splined part will still be able to fit over the servo shaft

Hold it in place as the glue cures

Place the horn onto the micro servo's shaft

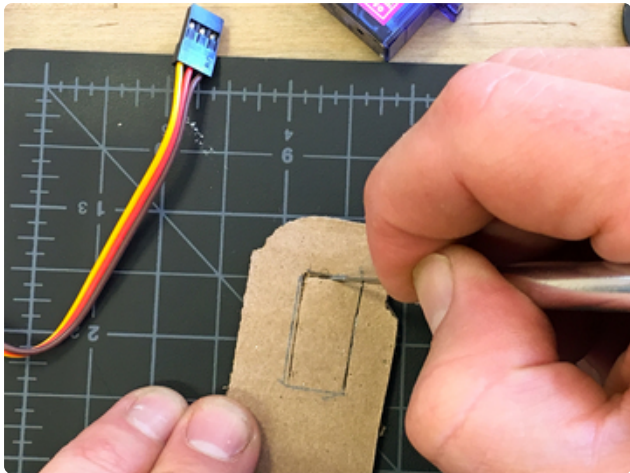
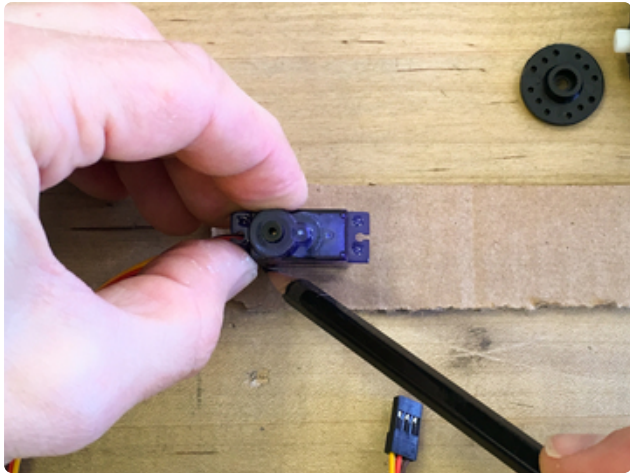
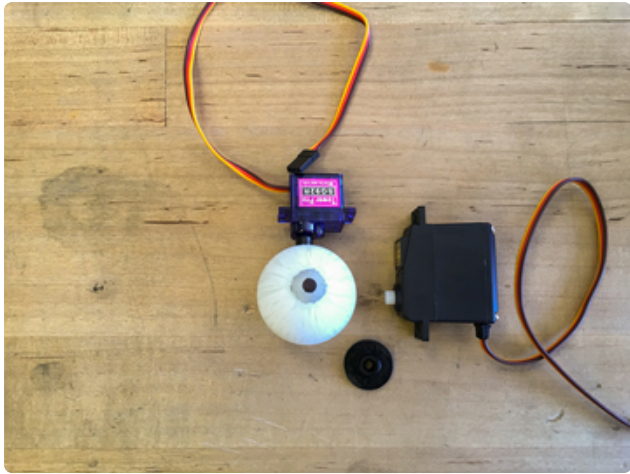




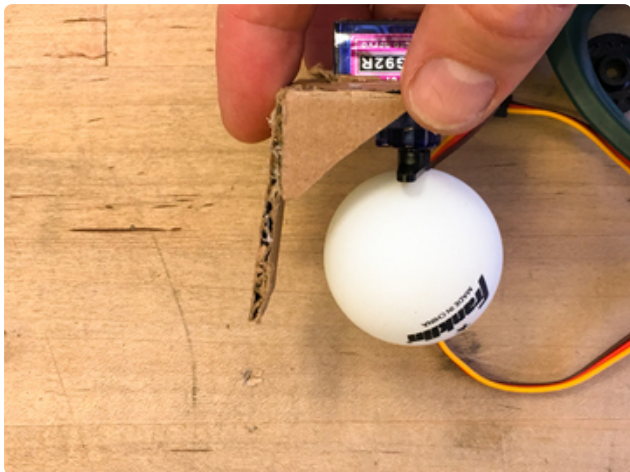
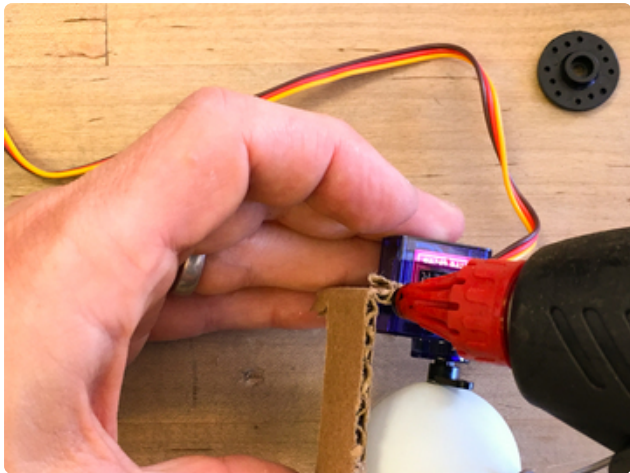
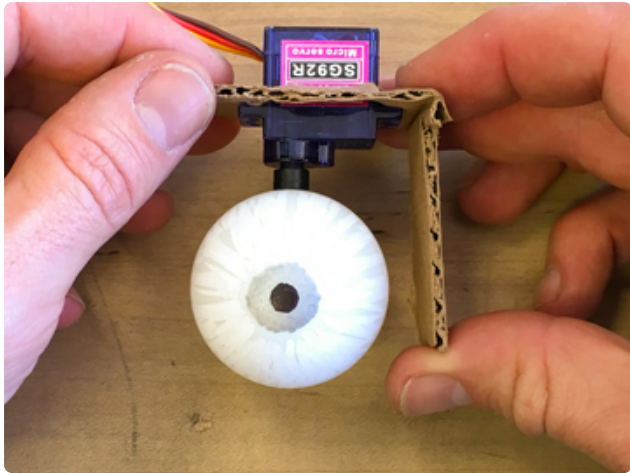
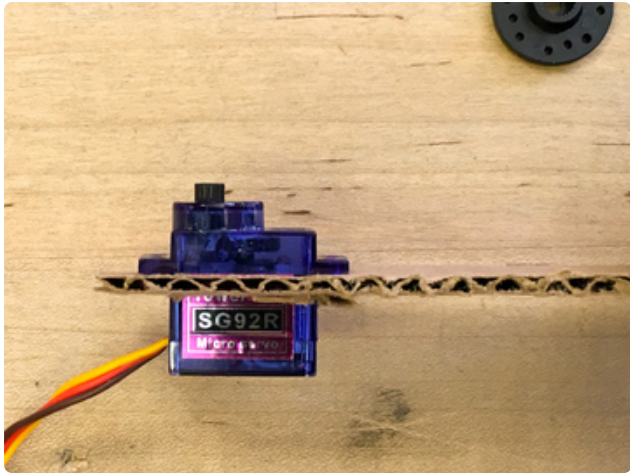
Bracket

Add a small cardboard bracket to hold the micro servo in place and connected to an arm on the full sized servo. The important part here to place the pivot of the eye at the same center as the full size servo's shaft.

You will glue on a corner brace to keep the 90° bracket stable.

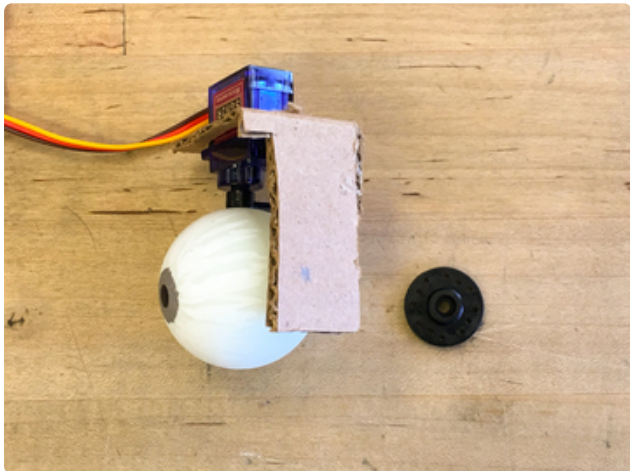
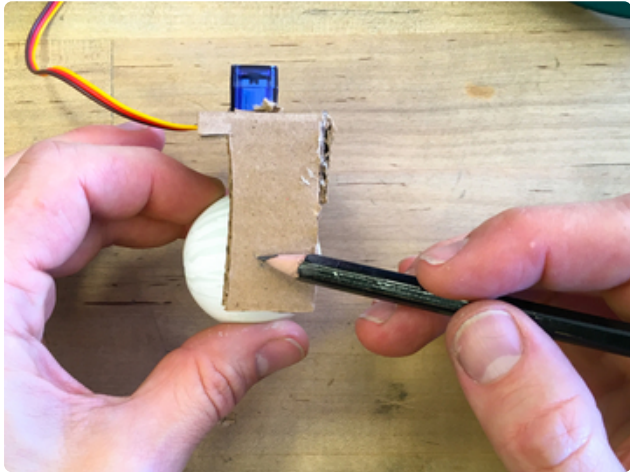


Cut a small strip of cardboard and trace the micro servo as shown
 Cut out the slot for the micro servo
 Press fit the servo into place as shown
 With the eyeball in place, bend down the cardboard into an 'L' bracket so it reaches a bit below the center of the eyeball -- this is where the bracket will attach to the full size servo for up/down rotation
 Cut and glue a small triangle onto the bracket for support as shown



Full Size Servo

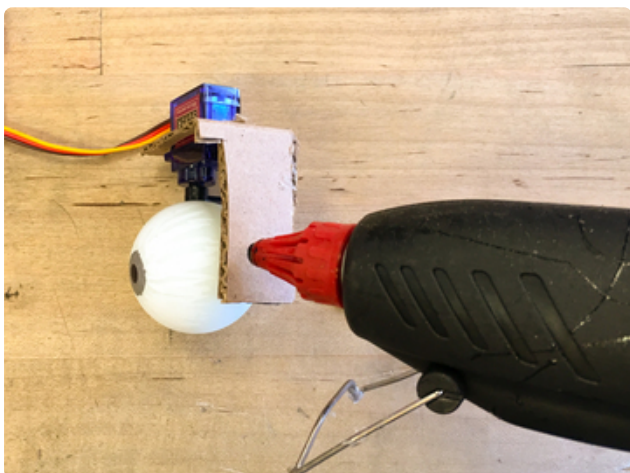
The full size servo will rotate the bracket, micro servo, and eyeball around the up/down pivot point.

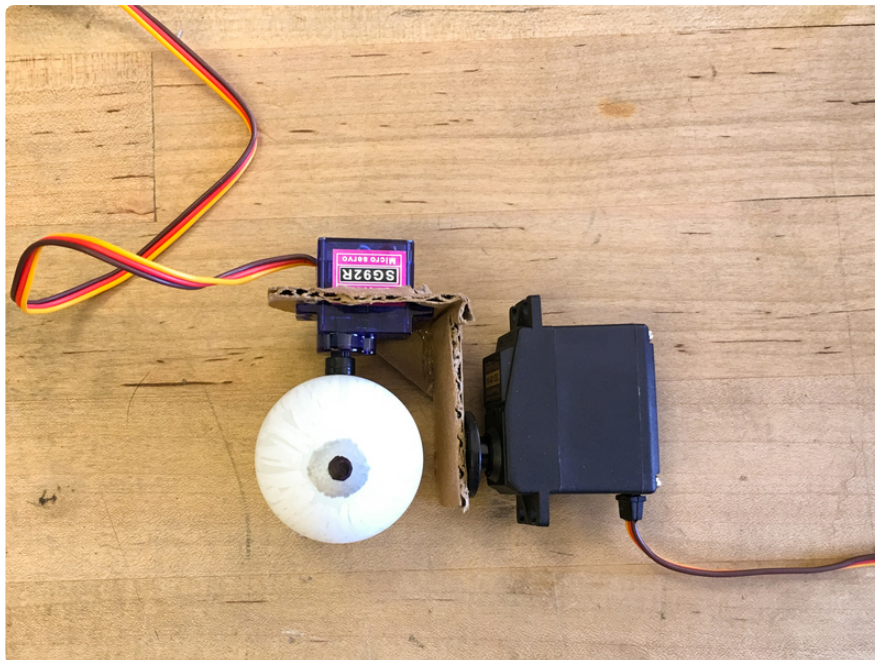
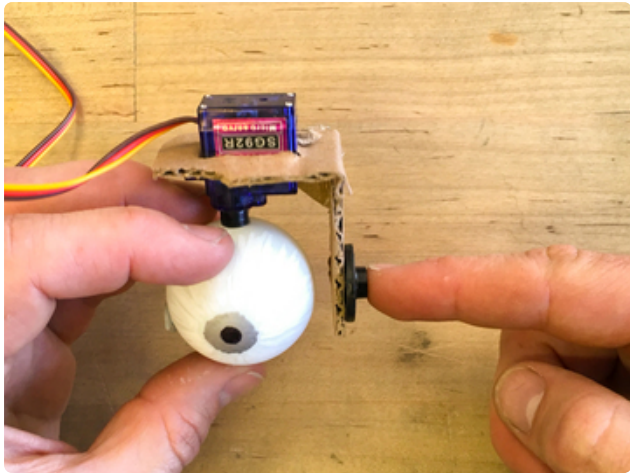
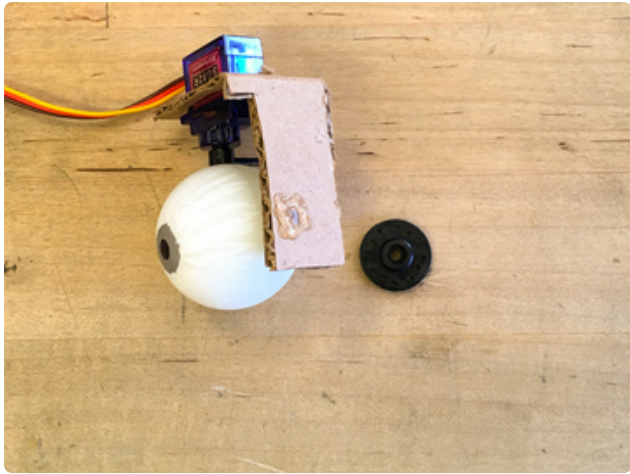


Mark the up/down center of the eyeball on the bracket

Glue a small servo horn to the pivot point of the bracket

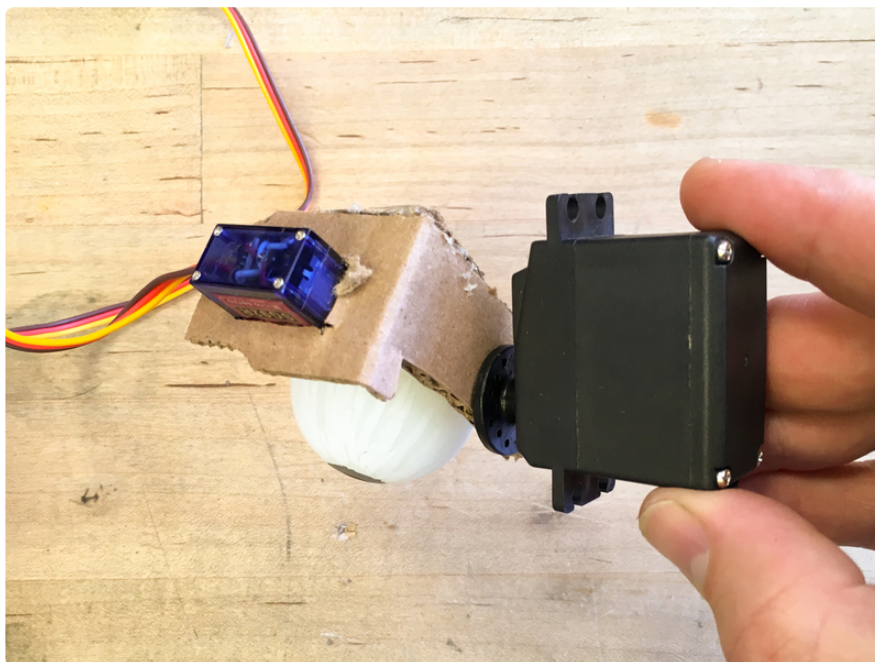
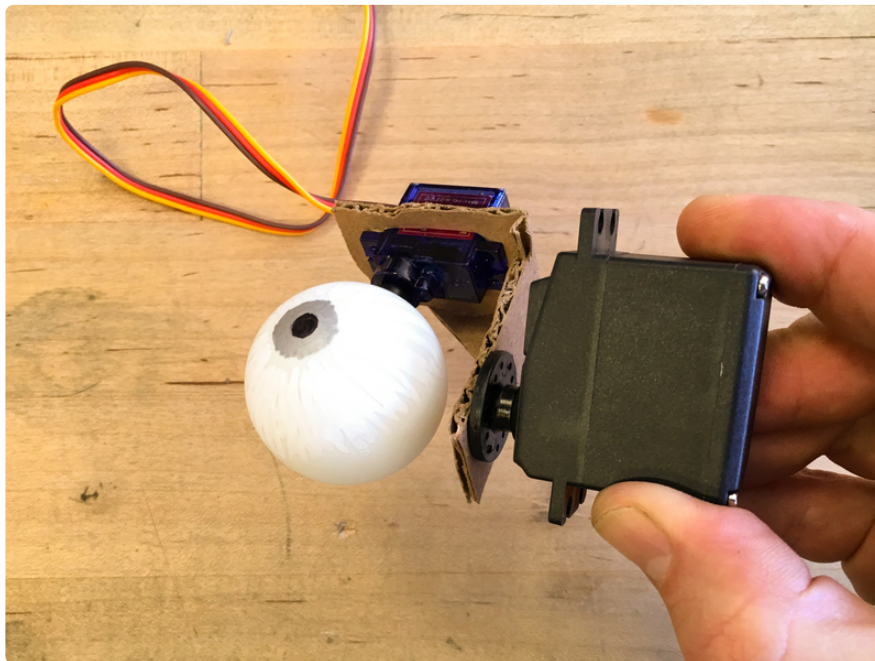
Once the glue cures, press the horn onto the servo shaft as shown



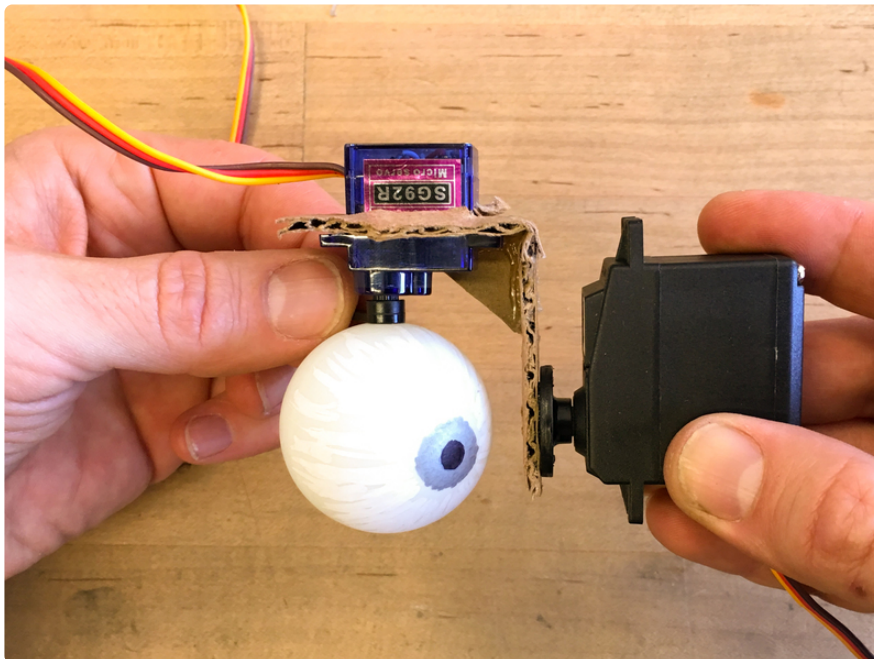
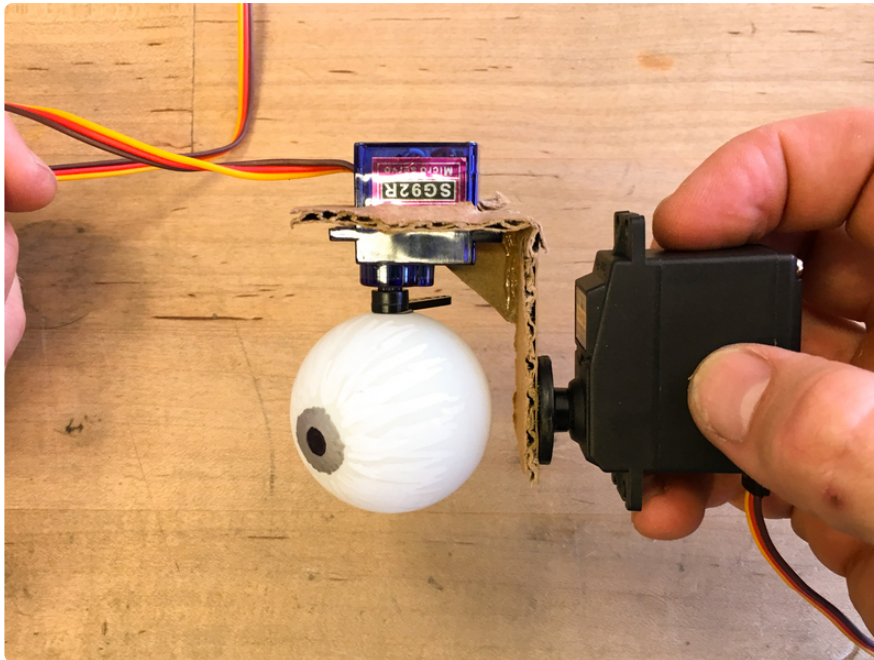


Test out the motion by rotating the servo shafts (this only works when they aren't plugged in and powered up). You can turn them each about 180°.

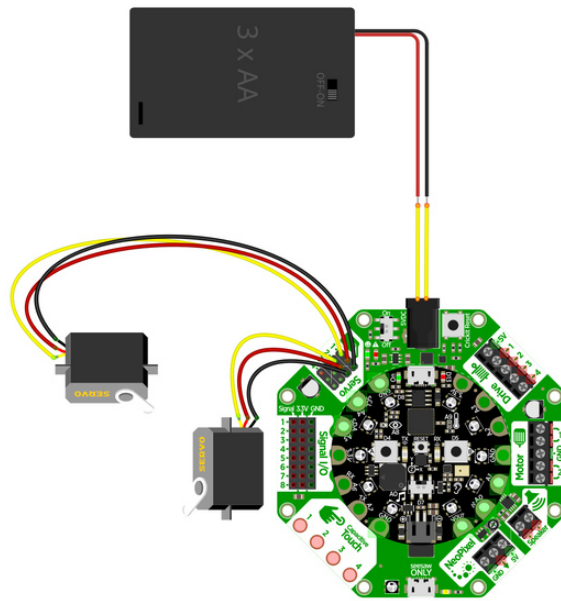
You may find that the shafts weren't centered when we attached the servo horns. That's no problem -- we'll find the center positions and re-connect the bracket and eyeball later to match.



Your eyeball gimbal is complete! Time to program it.

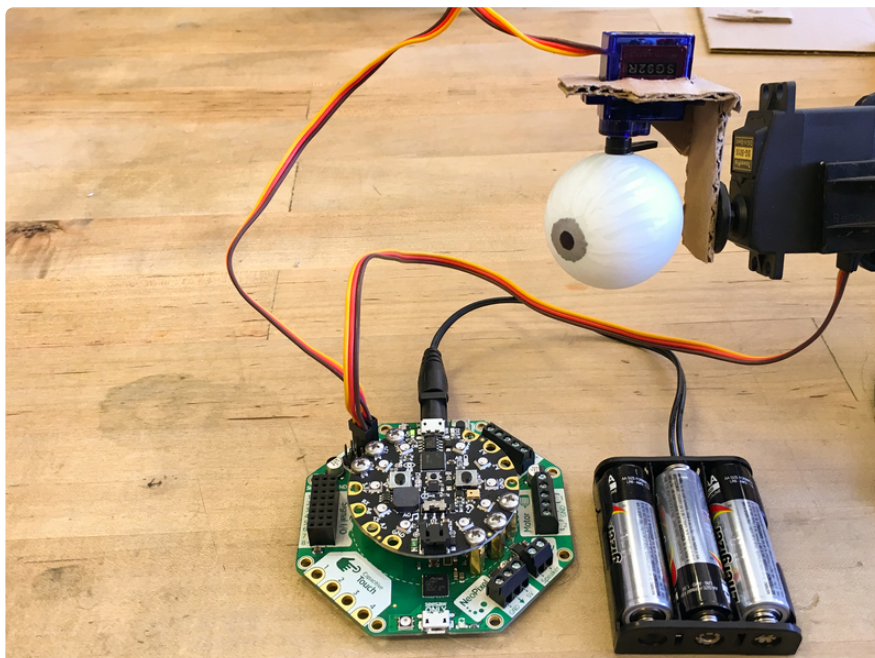


Crickit Circuit



We'll drive the two servos from the Crickit's servo outputs 1 and 2. Plug the micro servo into port 1 and the full size servo into port 2.

Then, plug the battery pack into the DC power jack on the Crickit and you'll be ready to start programming it!



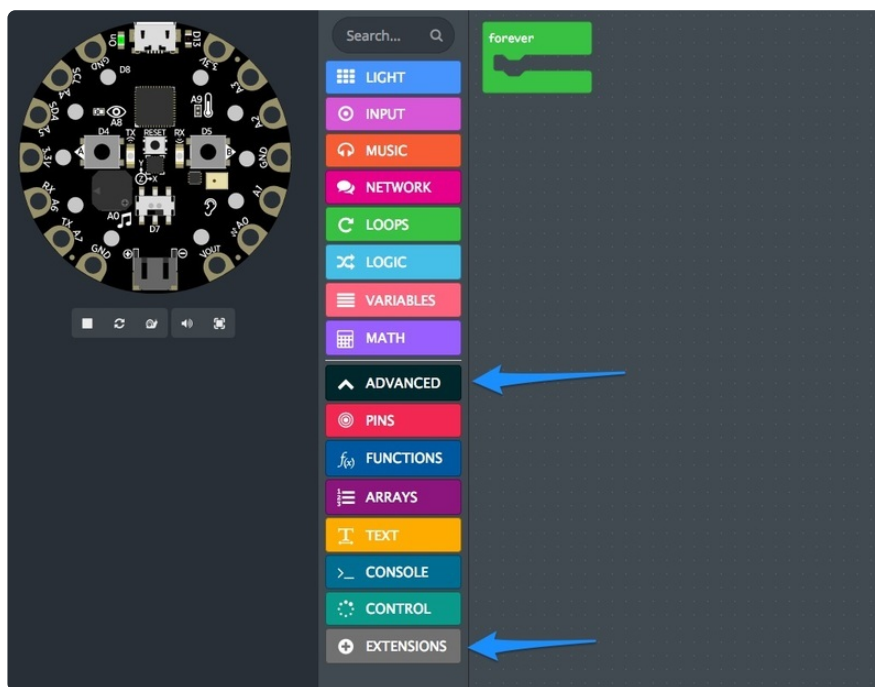
Code with MakeCode

You can code your Crickit and Circuit Playground Express using MakeCode! If you aren't already familiar with MakeCode, [check out this guide \(https://adafru.it/wWd\)](https://adafru.it/wWd) to get started, then return here when you're ready.

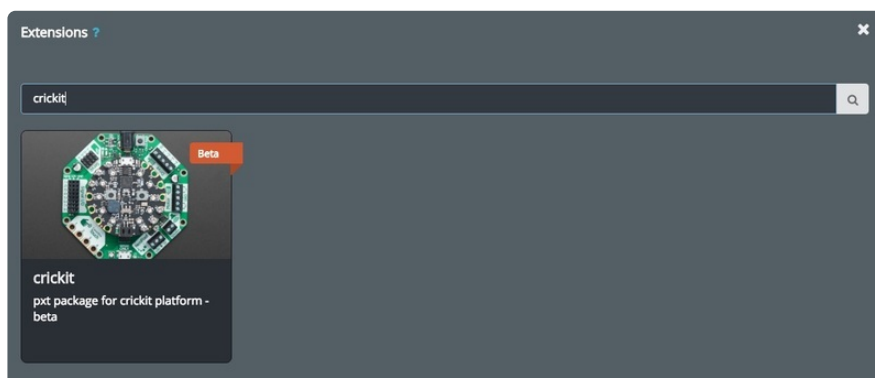
Crickit Extension

Inside of MakeCode, we'll add the Crickit extension so that we can access these powerful additional blocks!

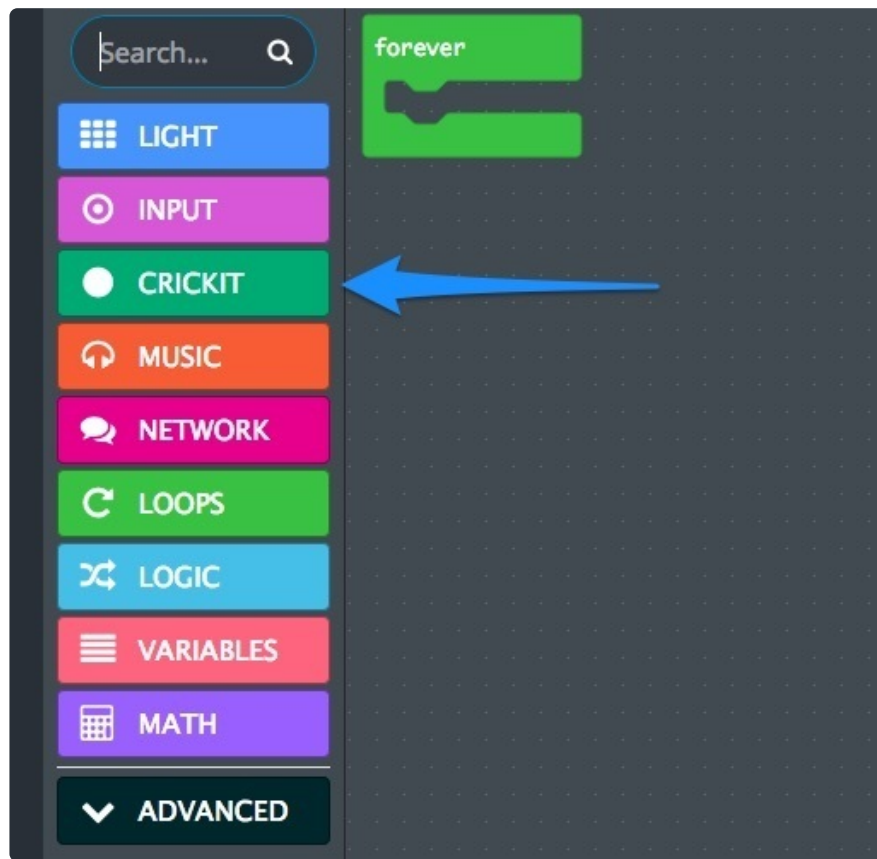
To enable Crickit functionality in MakeCode, click on the **Advanced** category, then click on **Extensions**.



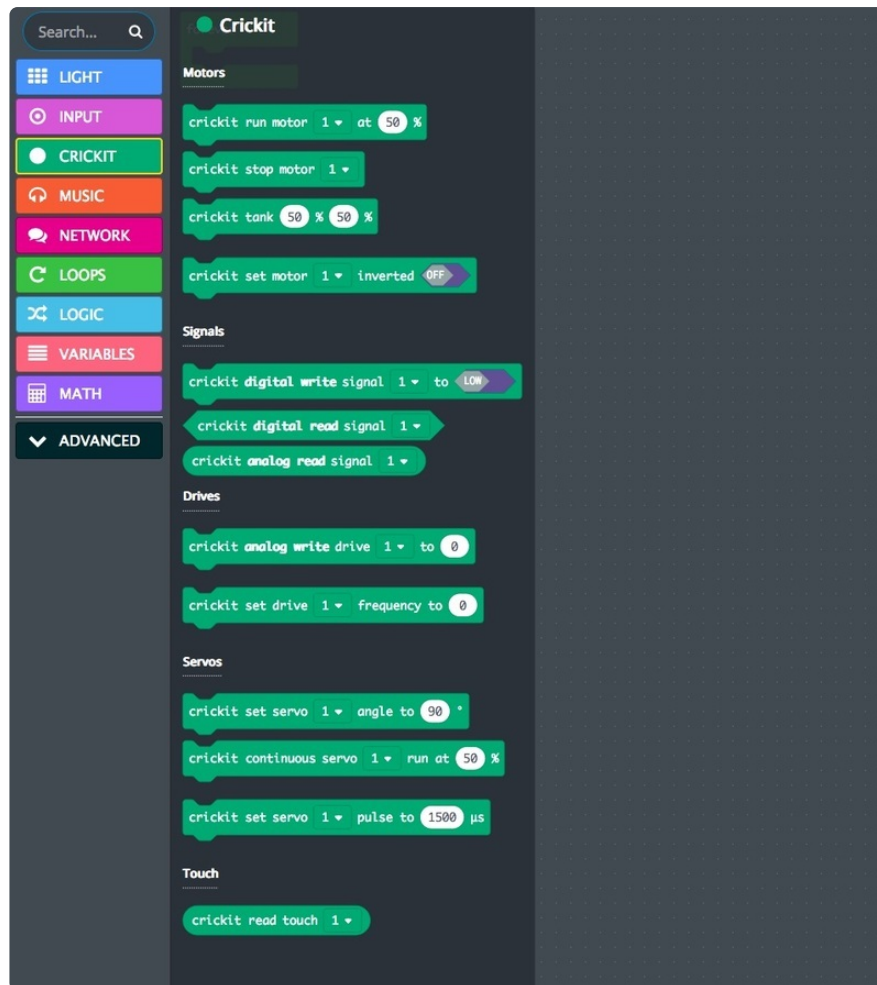
Then, type 'crickit' in the text field and press enter. The Crickit package will appear. Click on it and it'll be added to MakeCode!



Now that the package has been added, you'll see a new block category appear named **Crickit**



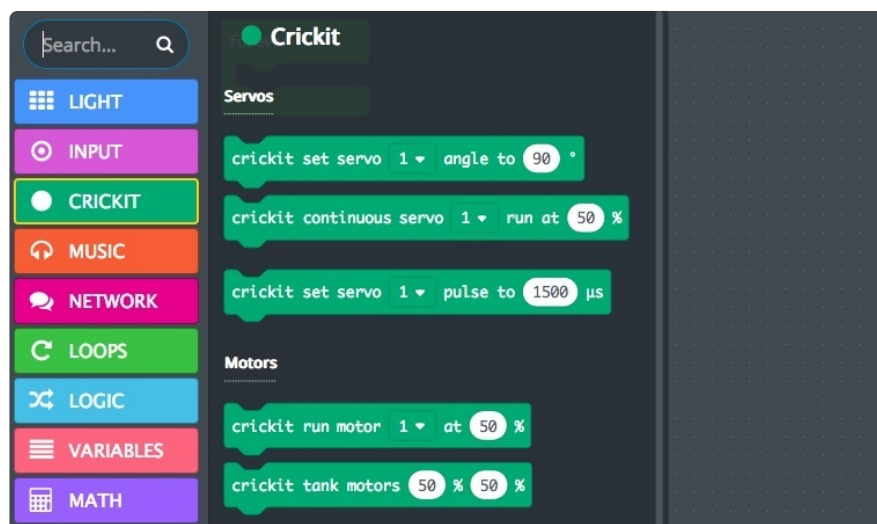
Click on the **Crickit** category to see all of the things you'll be able to do.

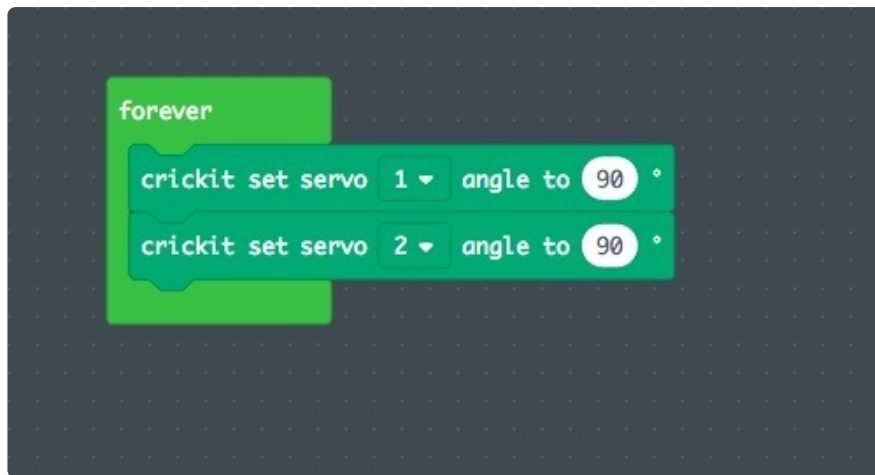


Servo Zeroing

Click the **CRICKIT** category and then add the **crickit set servo 1 angle to 90°** block to your program.

Place this block into the **Forever** loop, and then right-click on it and create a duplicate. Change the duplicate to set **servo 2** as shown.





Be sure you use the green Servo blocks under the CRICKIT group and NOT the red Servo blocks under the PINS block group!

Change the program's name from 'untitled' to 'Animatronic Eyeball' and then download it to your local hard drive. With the Circuit Playground Express (CPX) plugged in via USB, press the CPX's reset button once so the NeoPixels turn green. This places it into bootloader mode.

Drag the 'circuitplayground-Animatronic Eyeball.uf2' file you downloaded onto the **CPLAYBOOT** drive.

The CPX will reboot and the two servos will go to their center positions, since 90° is the halfway point of the servo's 0° to 180° range.

If the eye is no longer looking straight ahead, remove the servo horns and reposition them. They are now 'zeroed'!

Eyeball Range of Motion

Now, let's have the eyeball look around a little bit. First, we'll run through the range of motion of the left/right and up/down motions.

We can create this effect by adding a **pause 100 ms** block from the **LOOPS** category, and then duplicating the **crickit set servo** blocks as shown.

First, set the horizontal and vertical rotations to **90°** so the eyeball looks forward. Then after a half second pause, look left (**0°** on **servo 1**) pause a second, and look right (**180°**).

After another short pause, we'll center the eye horizontally (**90°** on **servo 1**) as we have it look down (**0°** on **servo 2**). Pause one second, then look up (**180°** on **servo 2**).

Pause again, and then repeat forever!



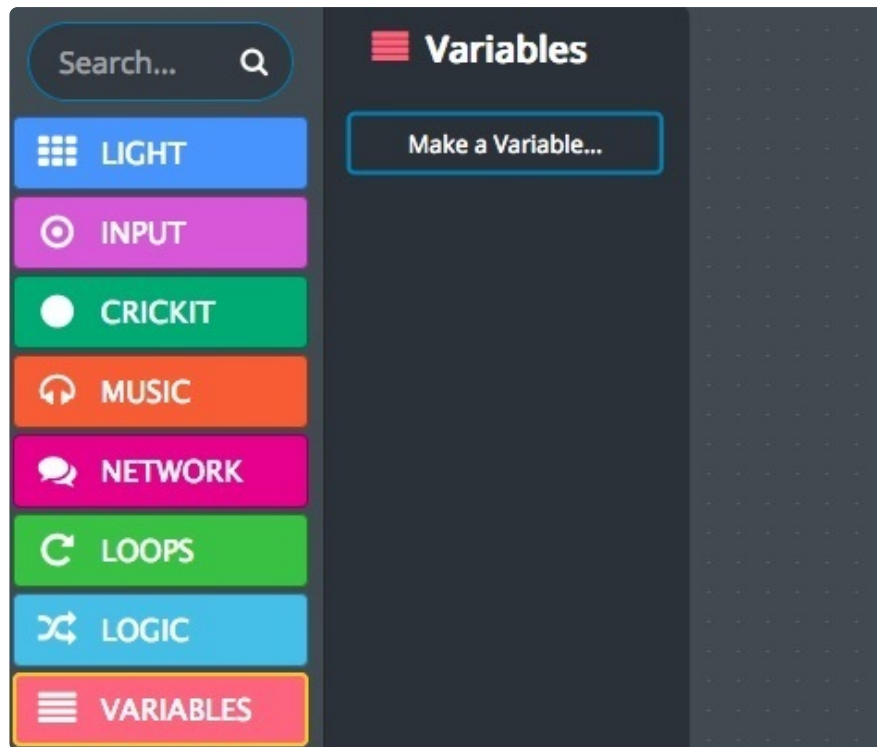
[Here's a link \(https://adafru.it/BBT\)](https://adafru.it/BBT) to the shared project.

Eye Darts

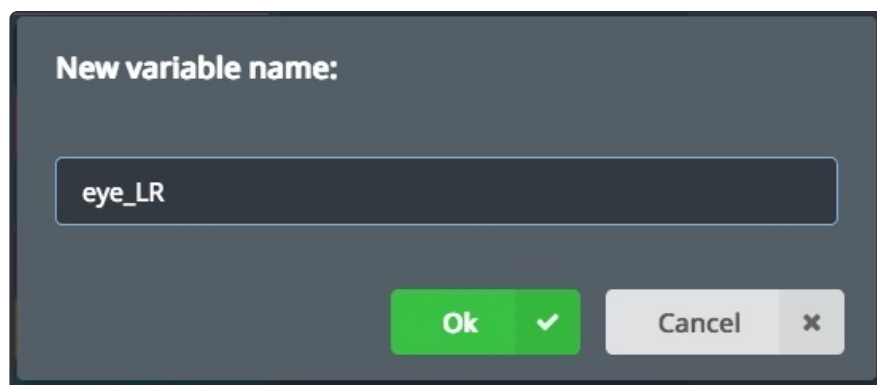
Random eye darts a great way to add some realistic life to your animatronic eyeball!

For this, we'll create three variables to represent the two servo angles and a delay time. We can then randomize those values.

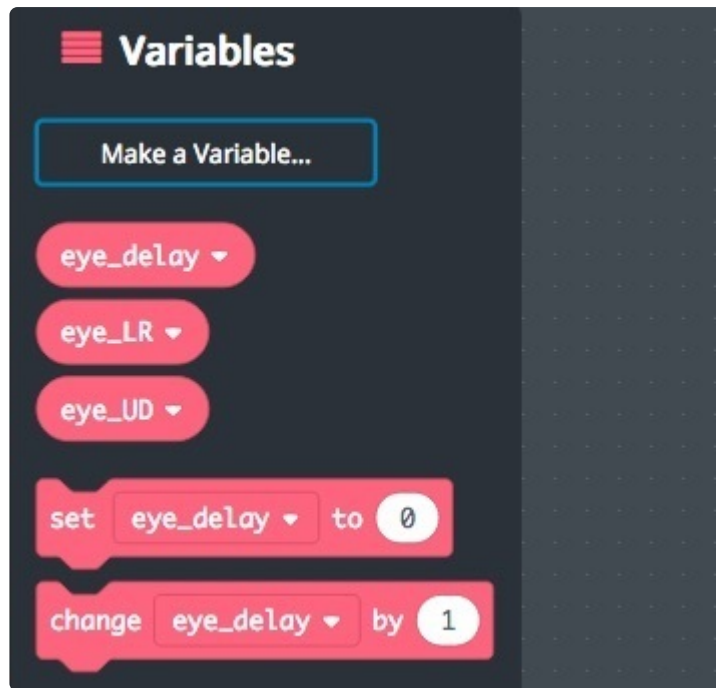
First, click on the **VARIABLES** category and then click on **Make a Variable...**



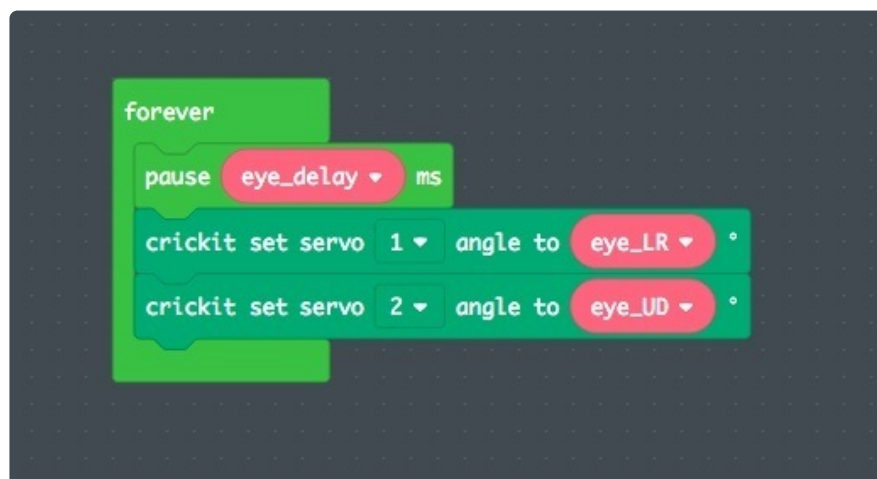
Then, type in the name **eye_LR** for the horizontal ('left/right') variable.



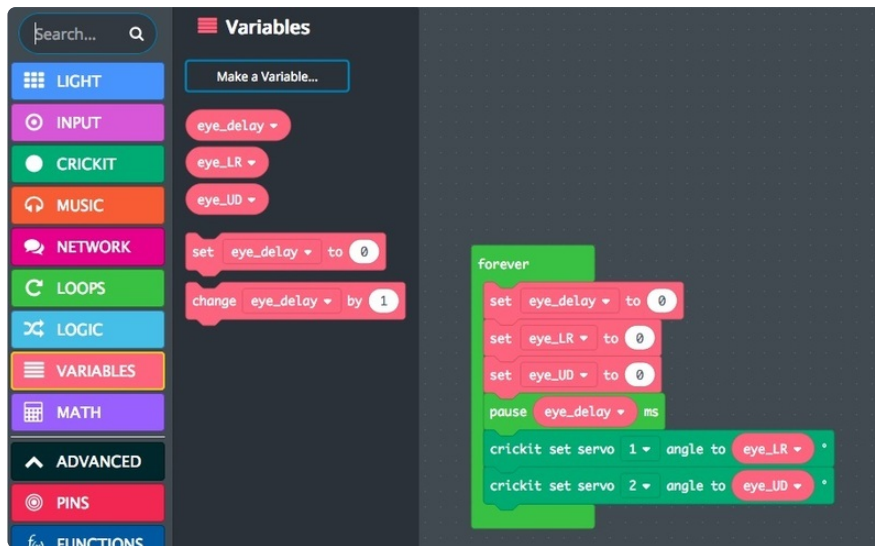
Do this two more times to create the **eye_UD** ('up/down') and **eye_delay** variables.



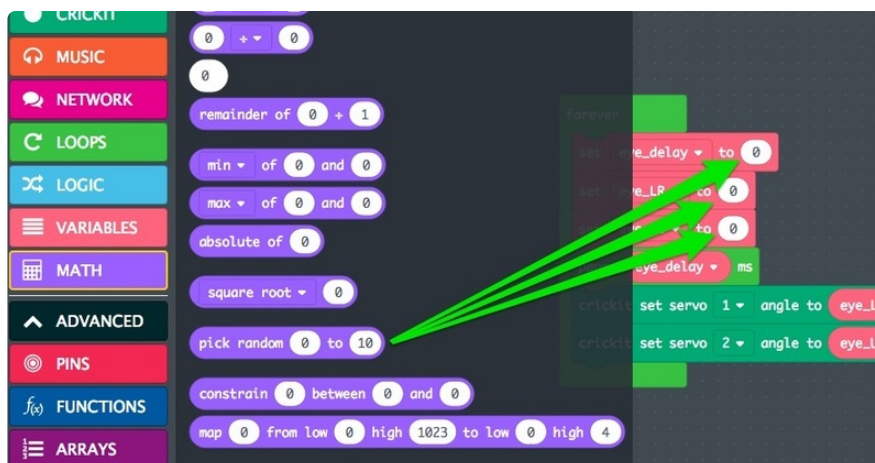
Now, create this simple set of blocks in the **Forever** loop and use the new variables in place of explicit values, as shown.



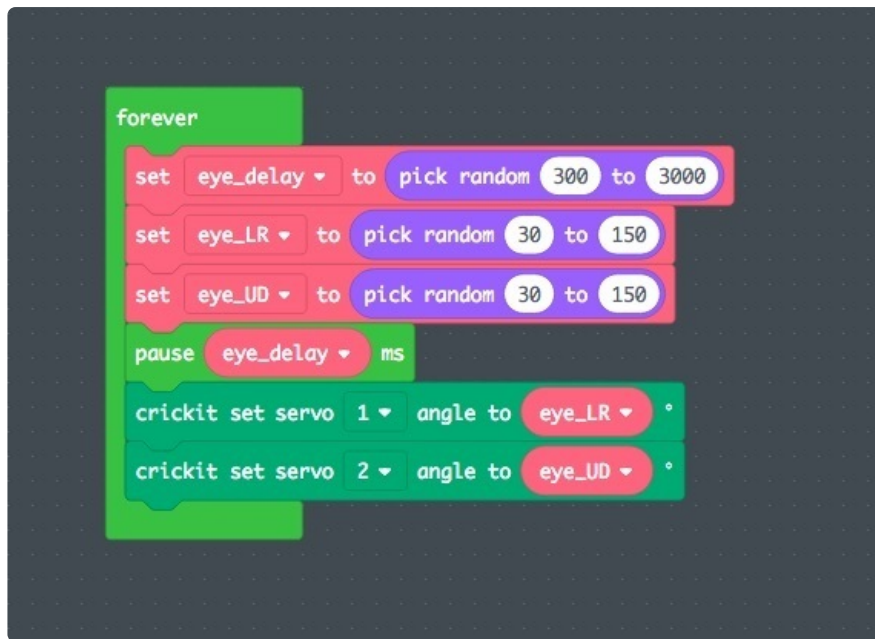
Currently, this won't do anything useful, because we haven't assigned values to any of the variables. Let's add some **set _variable_ to _value_** blocks as shown, then we'll add some randomness.



From the **MATH** category, drag in a **pick random** block onto each of the variable value assignments.



Each time the program loops it will find a new random value from within the specified range. Fill in some appropriate values for the randomization range -- for the delay I've picked anywhere from half a second to three seconds. For the angles, I've tightened up the range a bit so it looks like the eye is watching things, not just rolling around in it's imaginary eye socket!

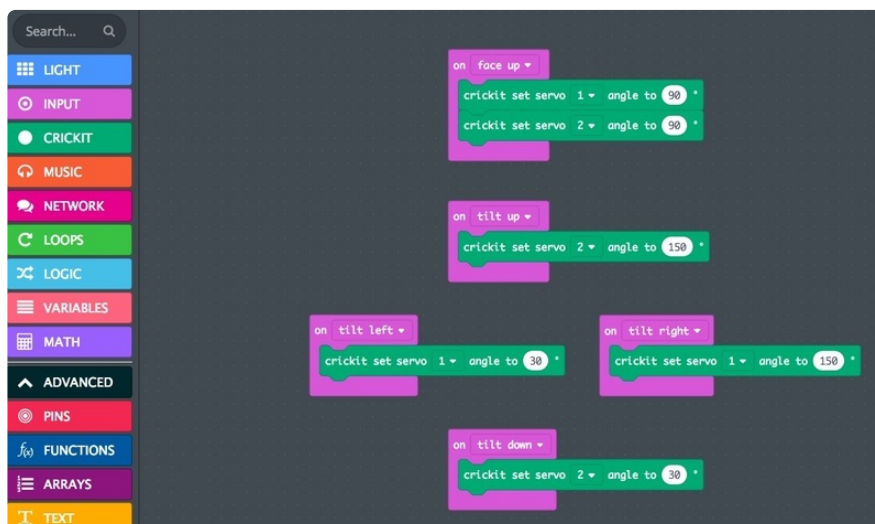


You can open the [shared project from this link \(https://adafru.it/BBU\)](https://adafru.it/BBU) if you like.

Interactive Tilt Sensing

The previous mode is self-running, but what if you want to interactively puppeteer your animatronic eyeball? Here's a simple way to do it using just the accelerometer that's built onto the Circuit Playground Express!

You can see that when the board is face up, the eye will go to center position. Then, as you tilt in each direction, the servos will respond in kind!



You can [click this link \(https://adafru.it/BBV\)](https://adafru.it/BBV) to check out the program.

Going Beyond

You're now all set up with your own animatronic eyeball. From here, you can decorate it, embed it inside of a mask, prop, sculpture, or who knows what?!

Try using other inputs for control such as capacitive touch sensing, the microphone, light sensor, buttons, and more!