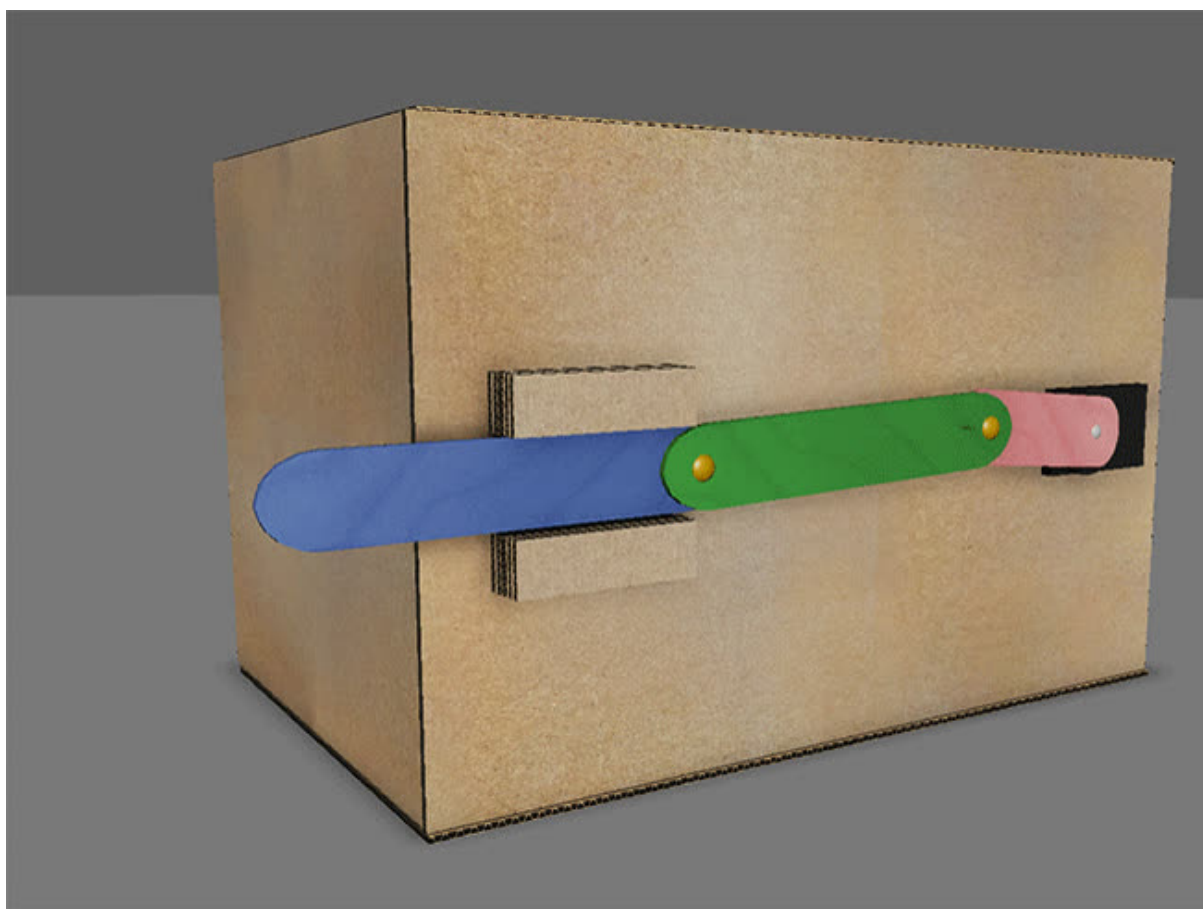




# Slider Crank Mechanism -- from Cardboard and Craft Sticks

Created by John Park



<https://learn.adafruit.com/cardboard-slider-crank>

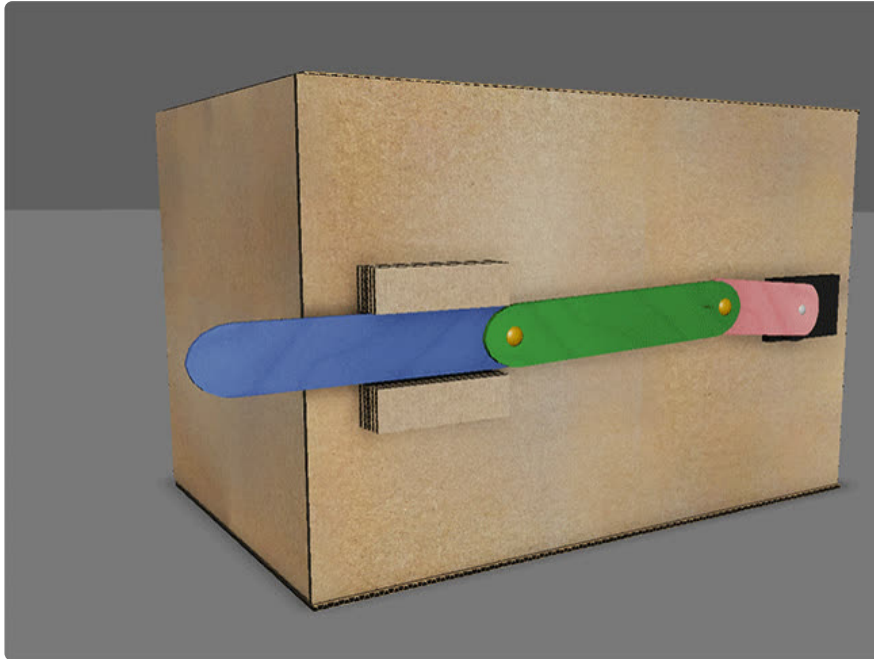
Last updated on 2024-03-08 02:59:45 PM EST

# Table of Contents

Overview	3
<hr/>	
• Materials	
• Tools	
Build the Slider Crank	5
<hr/>	
• Build the Linkage	
• Prep the Linkages	
• Fasten the Pivots	
• Servo Horn Connection	
• Wire the Servo	
Code it with MakeCode	13
<hr/>	
• Servo Control	
• Back and Forth	
• Angle Variables	
• Pulse Timing	
• Use the Angle Variables	
• Add Lights	
• Mode Switch	
• Conditions	
• Switch Reading	
• Button Clicking	
Mount the Slider Crank	25
<hr/>	
• Servo Attachment	
• Guide Block	
• Connect the Circuit Playground Express	
• Fire it Up!	

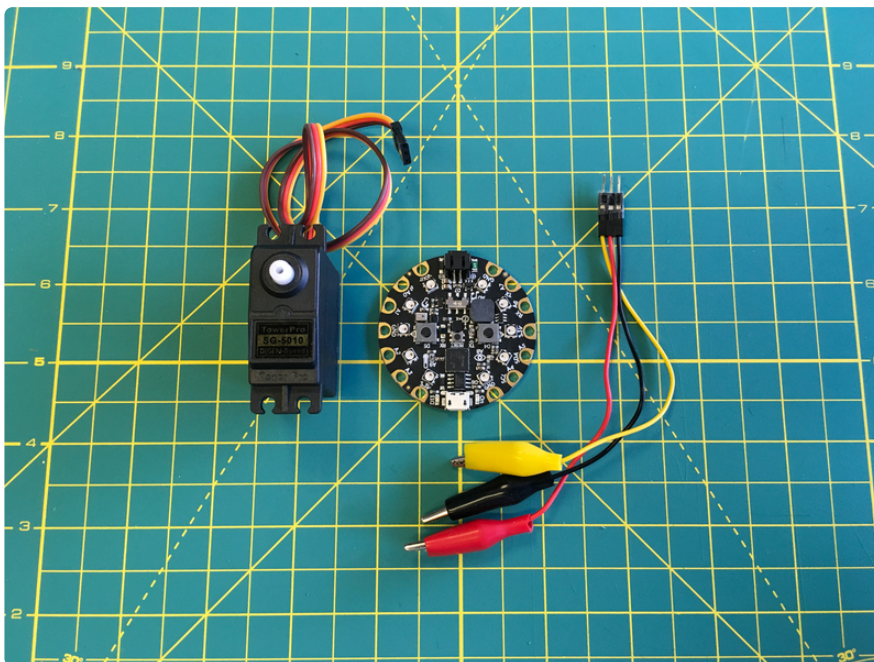
---

# Overview



A slider crank mechanism is a simple, effective way to convert rotation to linear motion. You can make one very easily out of craft sticks or cardboard, a hole punch, and some paper fasteners.

By attaching the crank shaft stick to a servo motor horn, we can drive our slider crank using a Circuit Playground Express programmed with MakeCode!



**1 x Circuit Playground Express**

<https://www.adafruit.com/product/3333>

Incredibly awesome microcontroller board

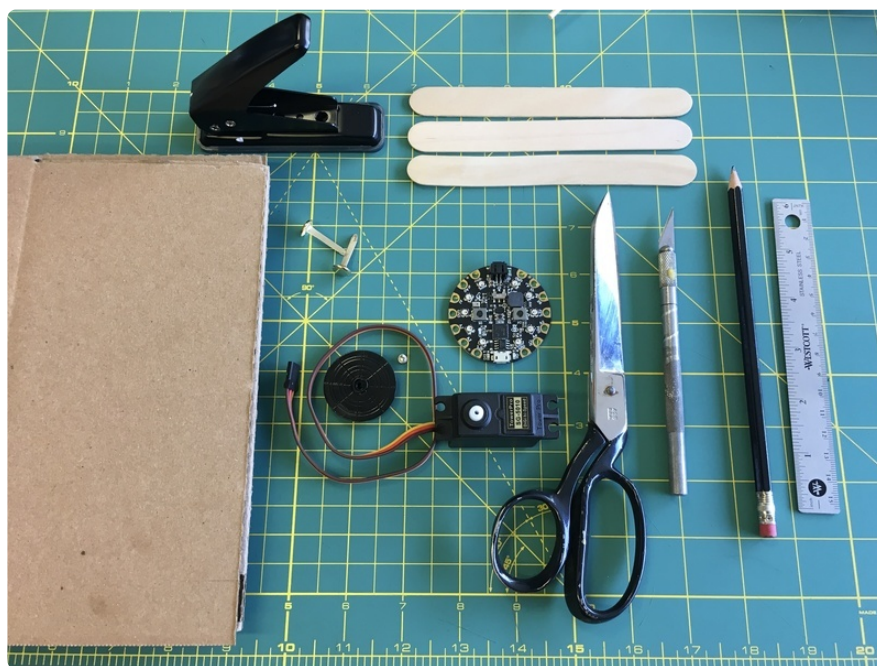
<https://www.adafruit.com/product/155>

6 pieces

<https://www.adafruit.com/product/3287>

<https://www.adafruit.com/product/3521>

3 pack



In addition to the parts above, you'll need:

- corrugated cardboard box
- three craft sticks
- brass paper fasteners



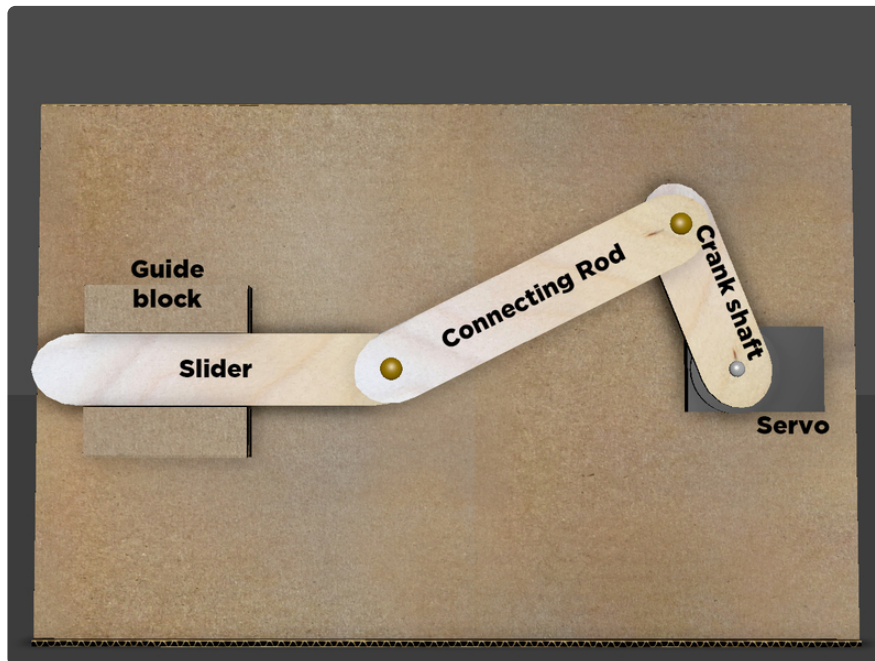
# Tools

You could build this entire project using only a pair of scissors, but here are some additional tools that can make things a bit easier:

- single hole punch
- hobby knife
- metal ruler
- hot melt glue gun and hot melt glue -- either low or high temperature is fine
- pencil or marker
- small screwdriver

---

## Build the Slider Crank



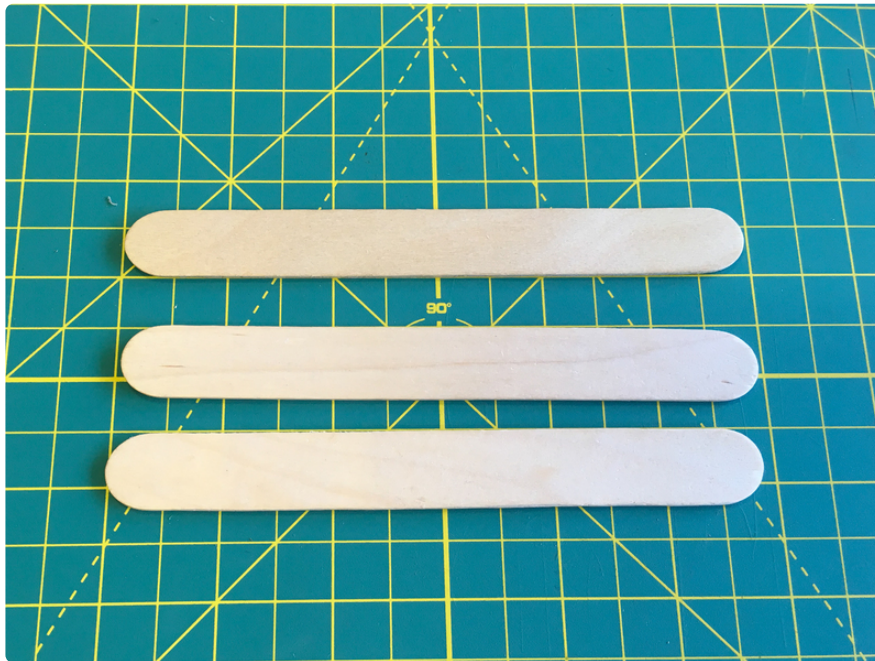
You'll build the slider crank assembly from three craft sticks. The mechanism consists of these key parts:

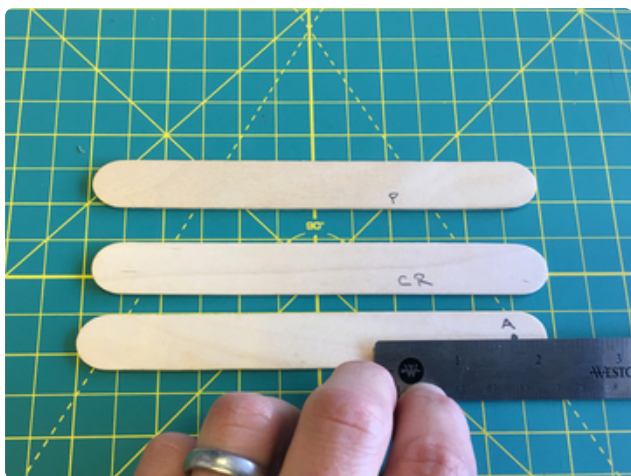
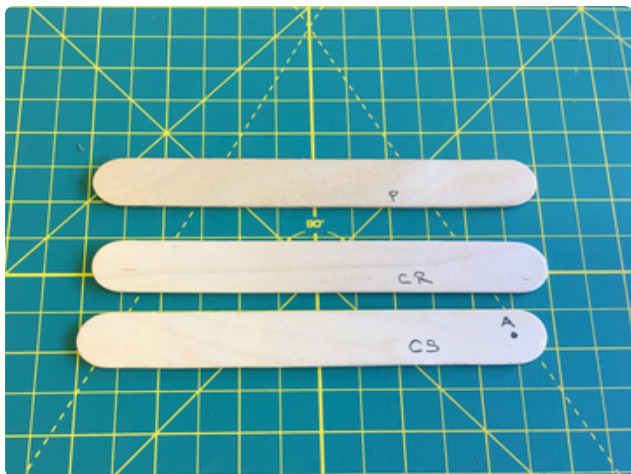
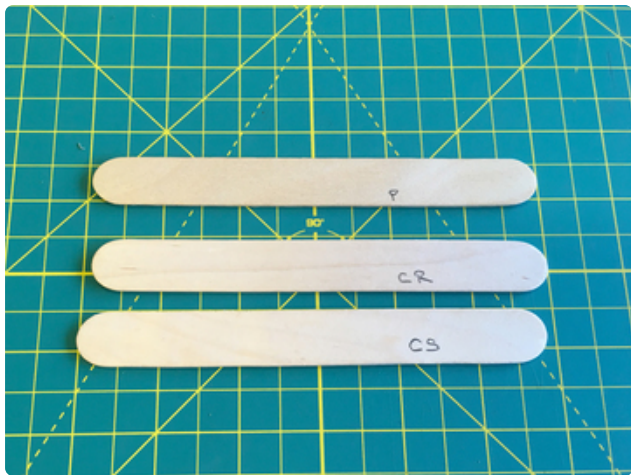
- **Crank shaft**
- **Connecting rod**
- **Slider** (also known as a **piston** in some mechanisms)

We'll drive the crank shaft with the **servo motor**, and the slider will translate back and forth along a **guide block** made of cardboard.

## Build the Linkage

Start by getting the three craft sticks.





## Prep the Linkages

Here, they've been marked 'CS' for crank shaft, 'CR' for connecting rod, and 'P' for piston (this is the same piece as the slider).

Mark a point 'A' near one end of the crank shaft

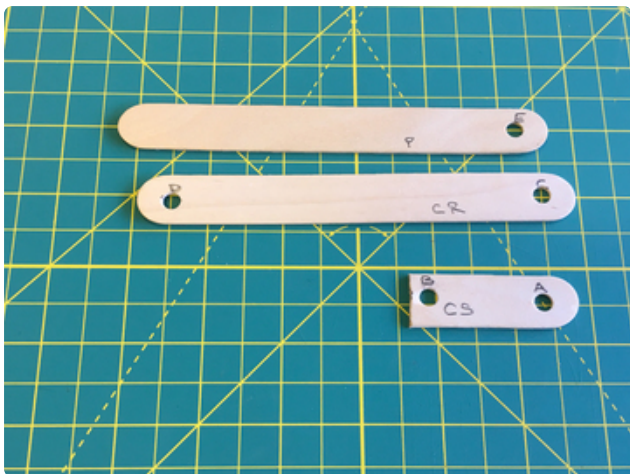
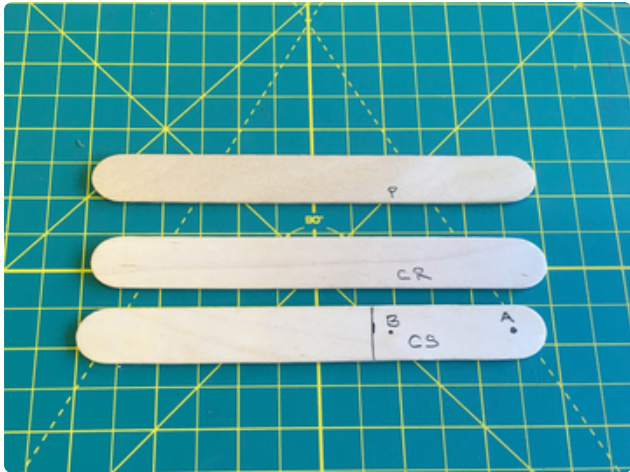
Measure and mark a line 2-1/4" from the A pivot end of the crank shaft

Cut the crank shaft to length

Punch a hole at each end of the crank shaft

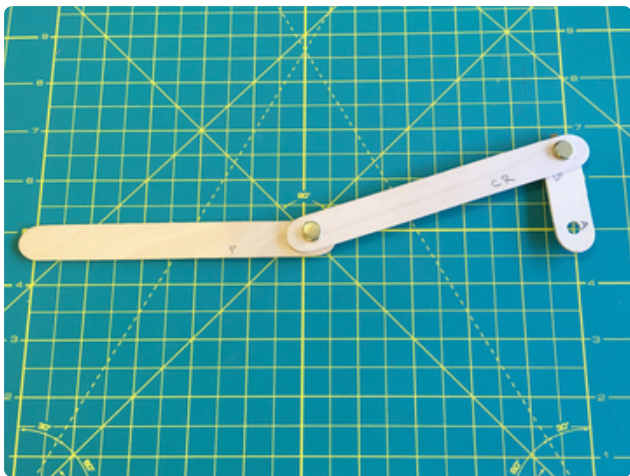
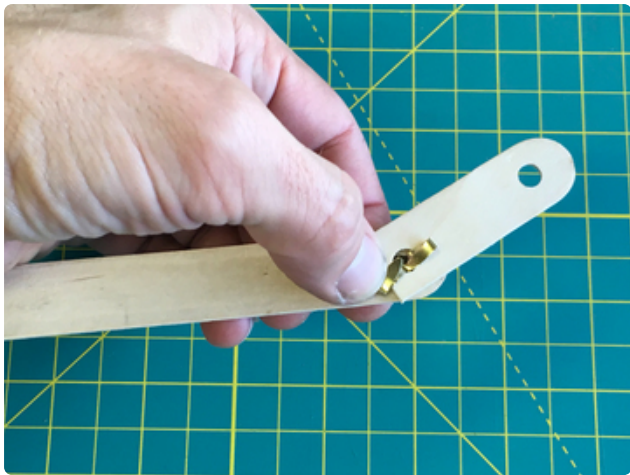
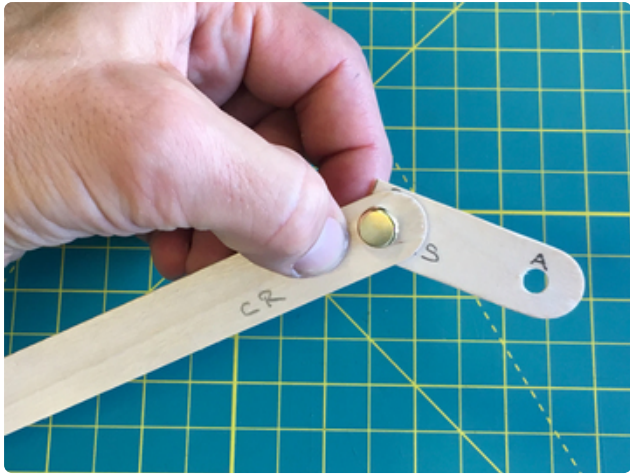
Punch a hole at each end of the connecting rod

Punch a hole at one end of the pivot or slider



Next, we'll join the linkages using the brass paper fasteners.





## Fasten the Pivots

Overlap the connecting rod on top of the B pivot end of the crank shaft, then join them with a paper fastener

Spread the fastener tabs wide on the other side, then fold them in

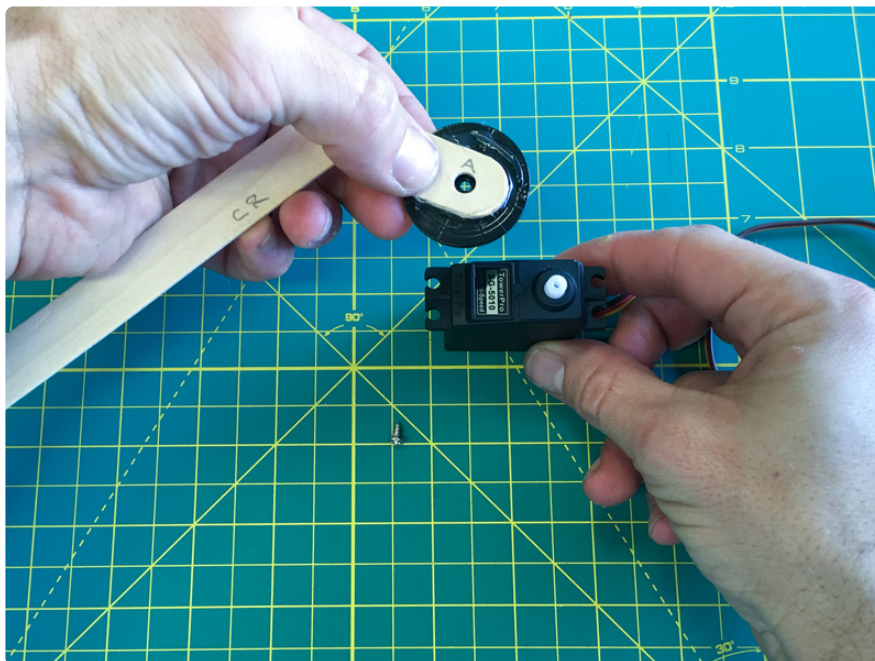
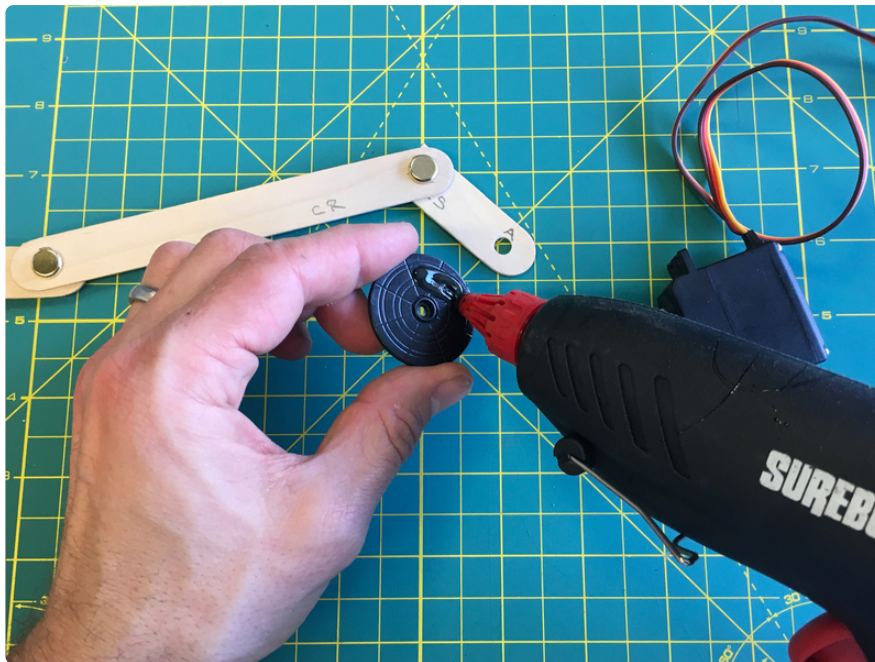
Repeat this process, overlapping the connecting rod D pivot with the E pivot of the piston/slider and then fastening it as shown

## Servo Horn Connection

Use hot melt glue to adhere the A pivot end of the crank shaft to the large, round servo horn. Once it cools, press it onto the servo's splined shaft.

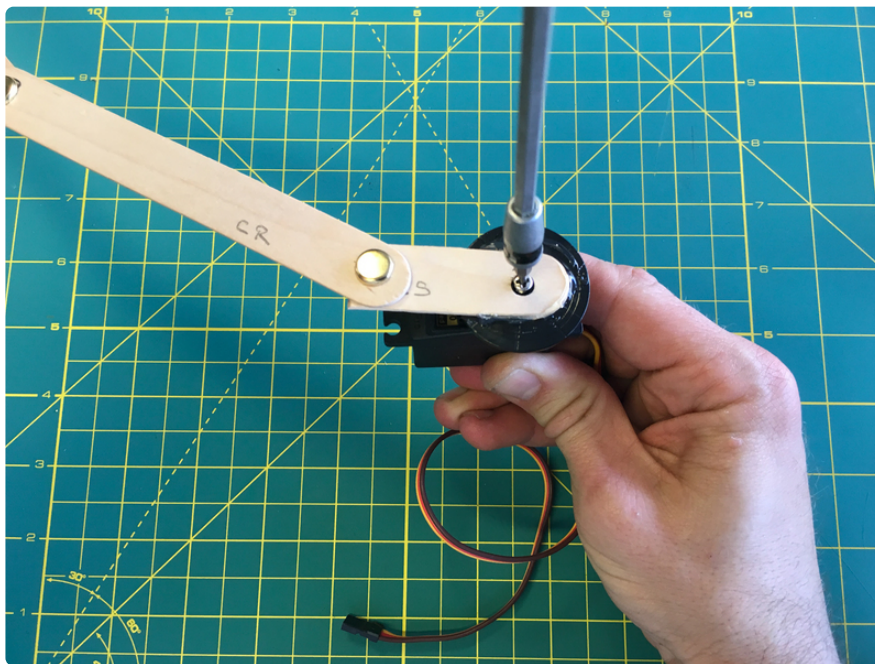
Use the horn to rotate the servo shaft to the far left and then remove, rotate, and reattach the horn to the shaft so the linkages are in a straight line to the left as shown.







You can secure the servo horn to the servo shaft with the small screw and a screwdriver.



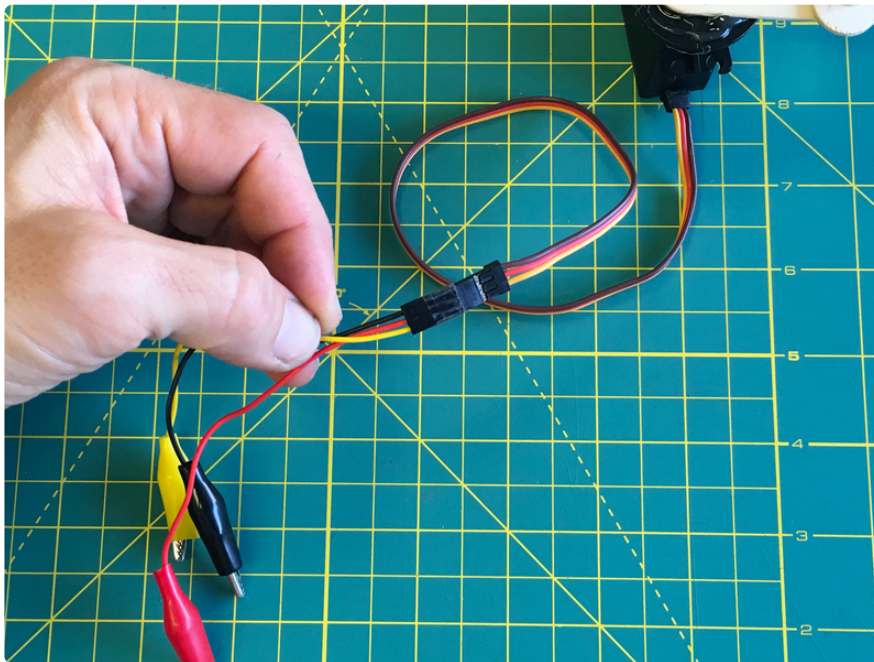
## Wire the Servo

You'll power and control the servo with the Circuit Playground Express. Plug the three alligator clip jumper wires into the servo cable housing:

- Black to Brown
- Red to Orange

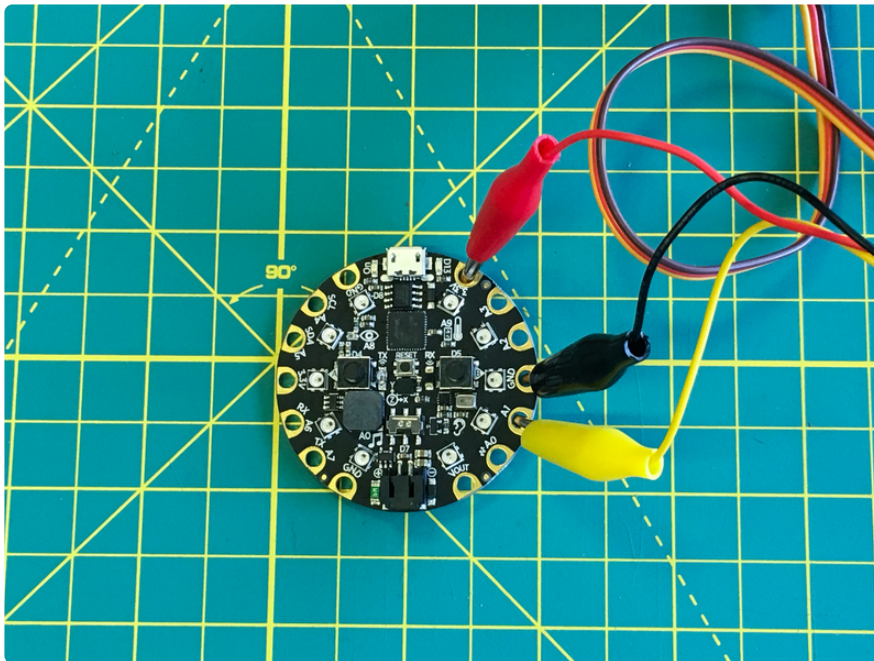


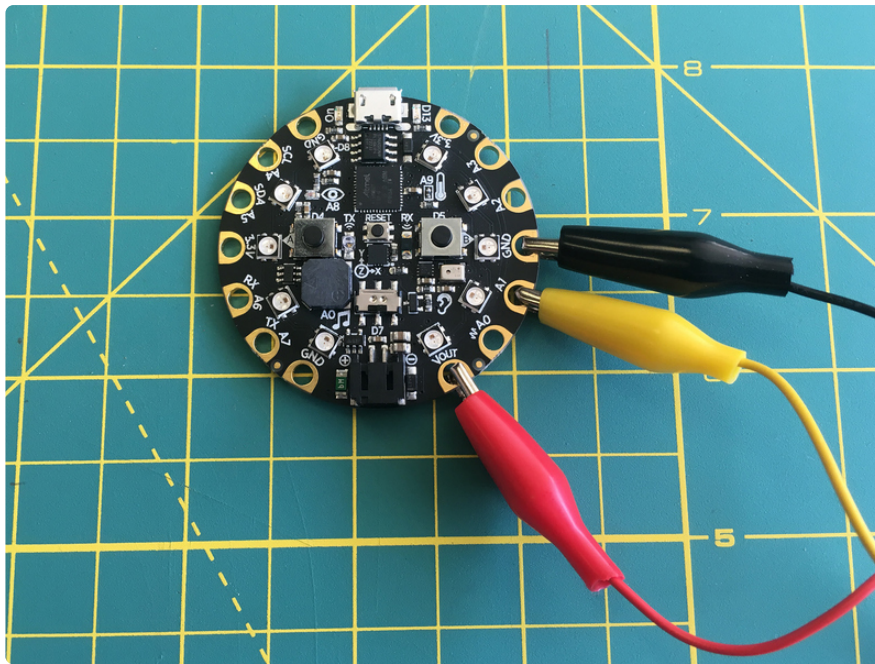
- Yellow to Yellow



Connect the alligator clips to the Circuit Playground Express pads:

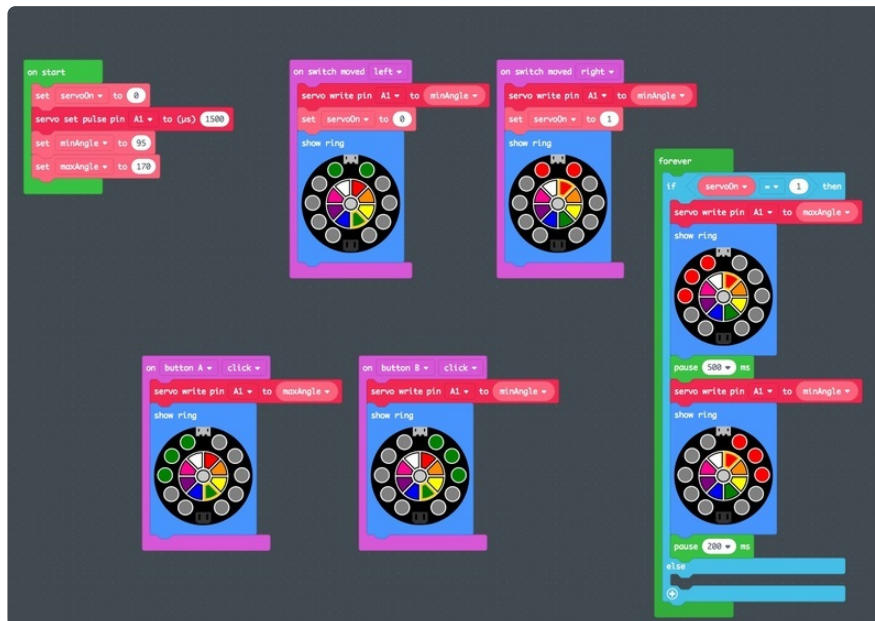
- Red to 3.3V (or, you can connect to VOUT for more power!)
- Black to GND
- Yellow to A1





Next, we'll program the Circuit Playground Express using MakeCode.

## Code it with MakeCode



Using MakeCode, we'll create a program to control the slider crank. We'll give it two modes -- automatic and manual. In automatic mode, the servo will oscillate back and forth between extended and retracted positions on the slider

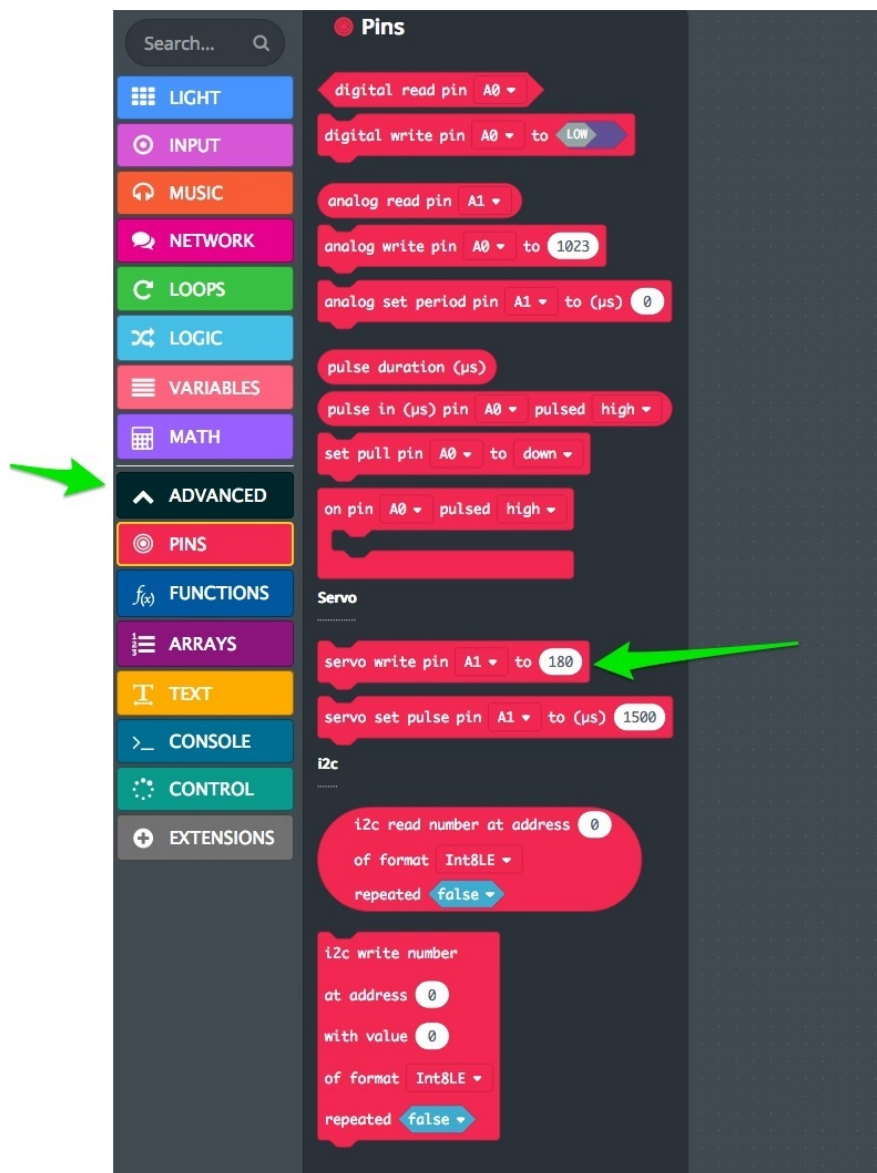
In manual mode, you will be able to press the A and B buttons on the Circuit Playground Express to extend or retract the slider.

If you're new to MakeCode, start by going through the [introductory guide here \(https://adafruit.it/wB5\)](https://adafruit.it/wB5). Then, return to this guide to continue.

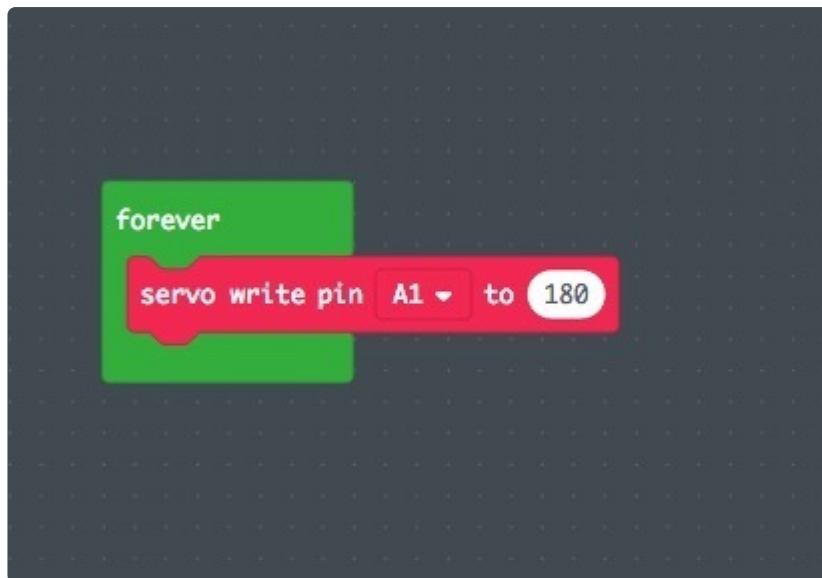
## Servo Control

First, let's have a look at controlling a servo with MakeCode. Start a new program by going to [makecode.adafruit.com \(https://adafruit.it/wpC\)](https://adafruit.it/wpC) and clicking **New Project**.

In the MakeCode editor, click on the **ADVANCED** button to reveal additional categories. Then, click on **PINS** and from the **Servo** section, drag a **servo write pin A1 to 180** block onto the program canvas area and into the **forever** block.





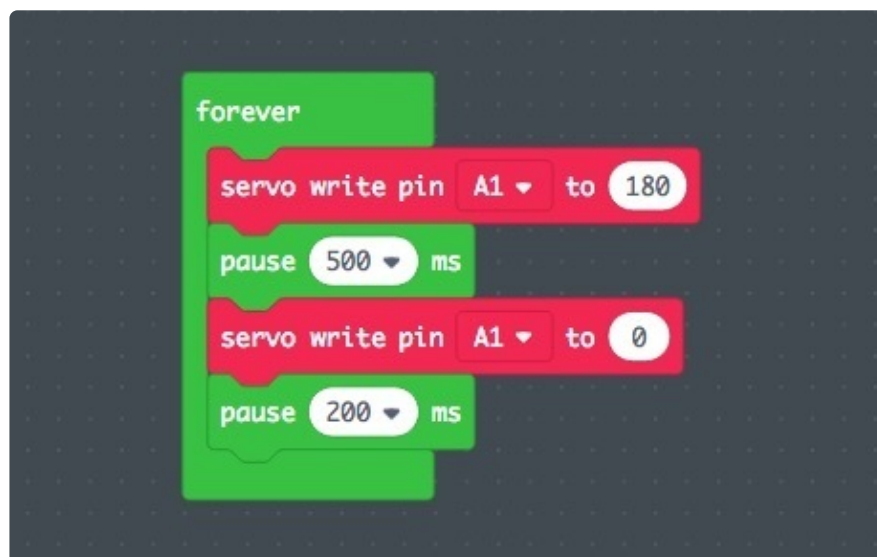
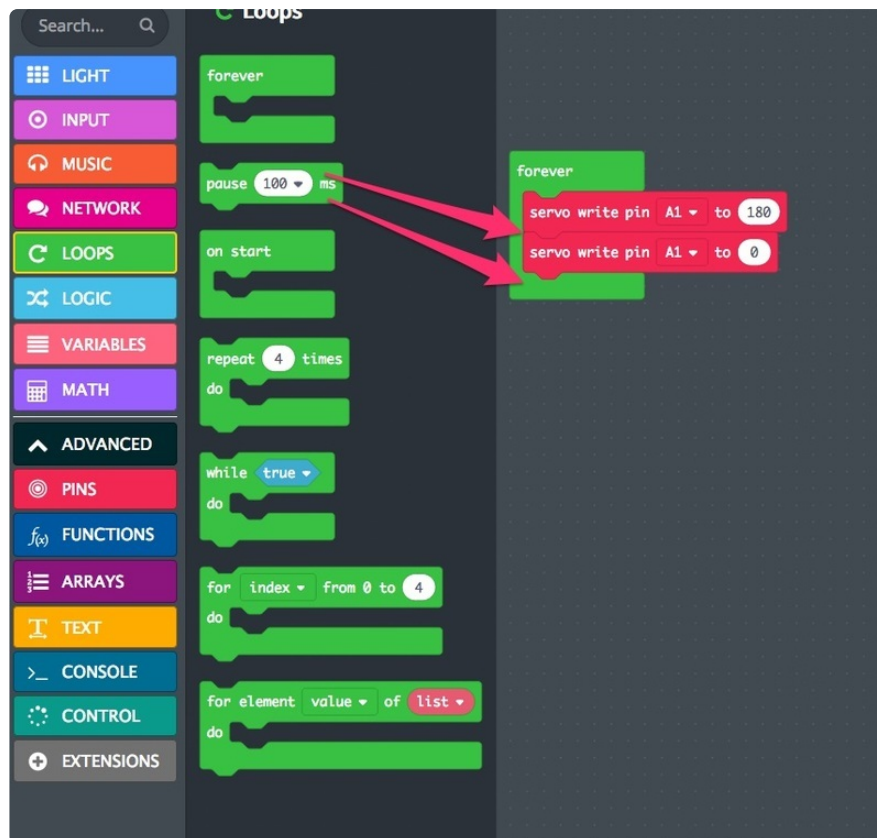


With the servo attached to power, ground, and pad **A1** on your Circuit Playground Express, download the program, connect the CPX to your computer over USB, press the reset button to enter bootloader mode. Then drag the downloaded .uf2 file to the CIRCUITPY drive.

When the program finishes uploading, the servo will quickly rotate to 180 degrees.

## Back and Forth

We can make the servo move back and forth with a pause in between by duplicating the **servo write A1 to 180** block, changing its value to 0, and then adding a pair of pause blocks.

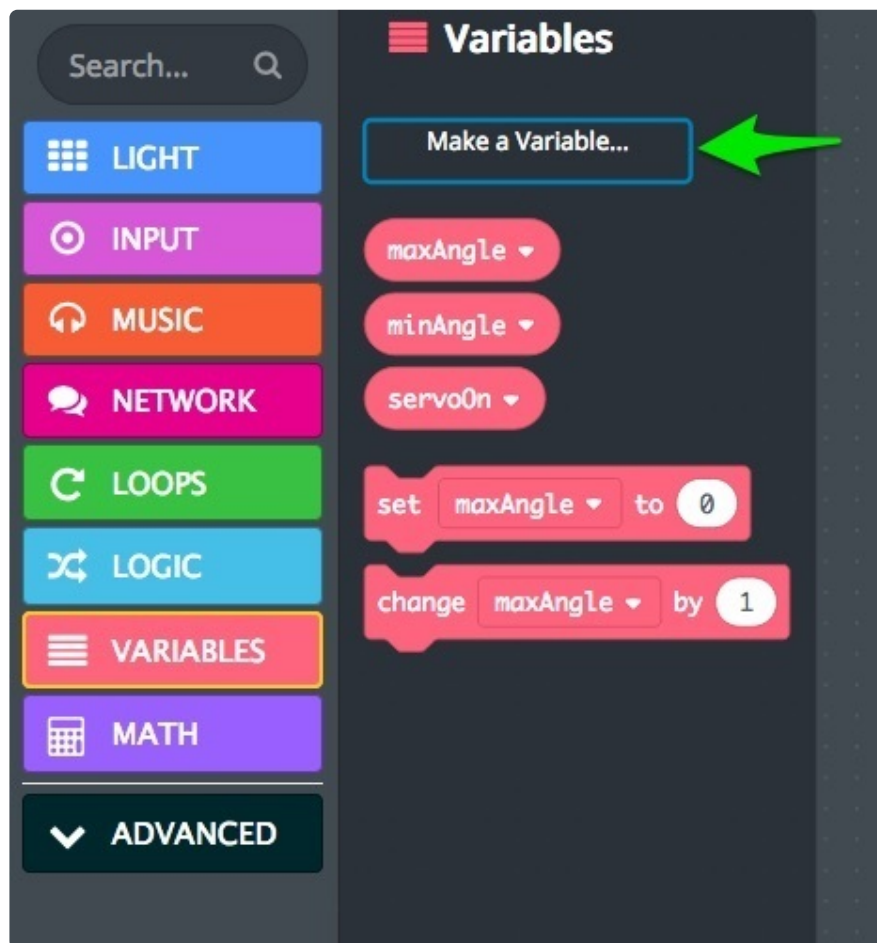


## Angle Variables

Now, these angles are a bit extreme for our slider crank mechanism. Let's make the maximum angle 170 and the minimum 95.

Since we'll use these values in a number of places throughout our code, and we may want to fine tune them, we can create variables to represent each. Then we only need to change the variables' values in one place.

In the **VARIABLES** category, click on **Make a Variable...** and create a variable called **maxAngle** and another called **minAngle**. Also, create one called **servoOn** which we'll use later.

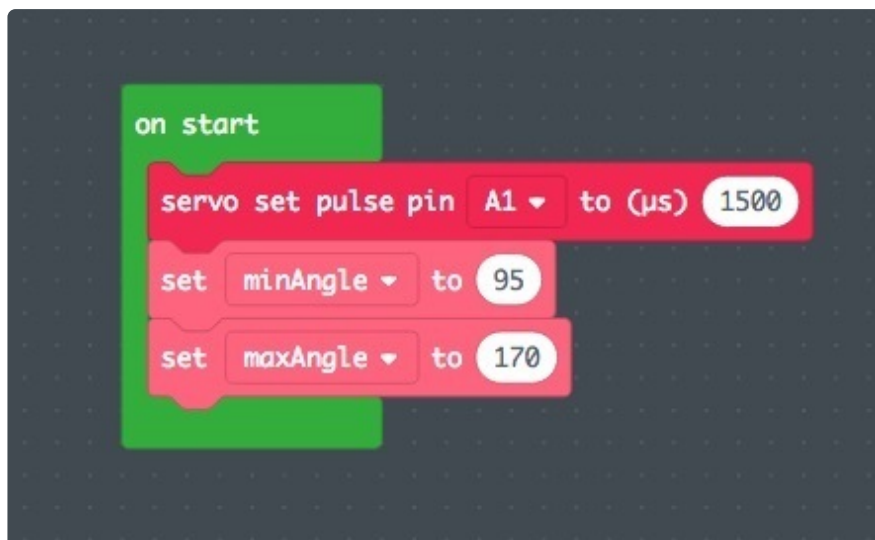


We'll want to set the values of the servo angle variables when the program begins. Drag an **on start** block from the **LOOPS** category. Then, put into it a pair of **set maxAngle to** blocks into the **on start**. You'll used the drop down menu to pick the variables in each block, change one of them to **minAngle**. You can then set the values as shown.

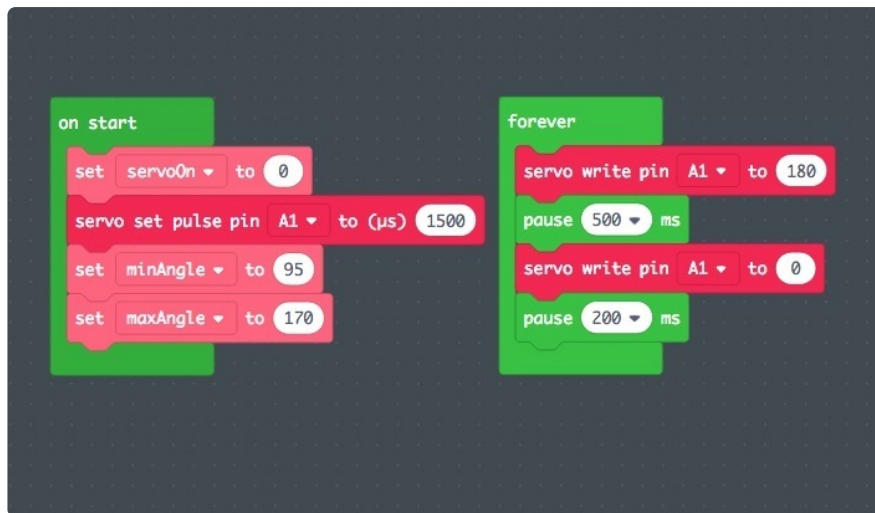


## Pulse Timing

Different servos expect different timing for their command pulses. If you need to change this value, you can add a **servo set pulse pin A1 to** block to the **on start** as shown here.

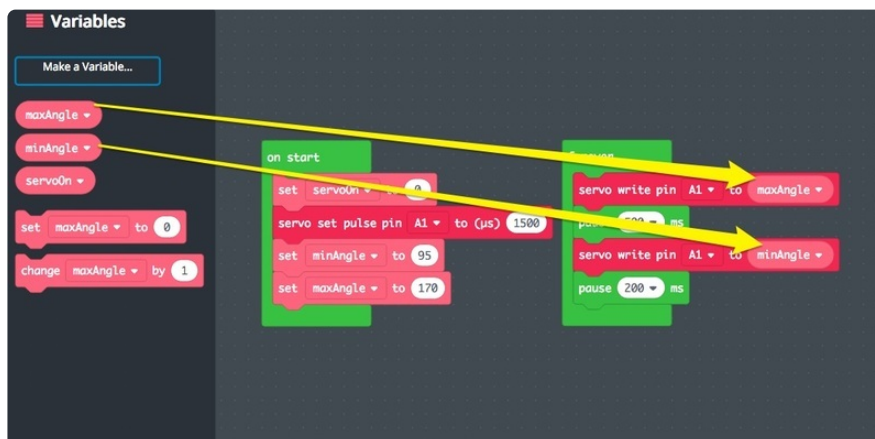


Complete the **on start** block by adding one more **set maxAngle to** block, and change the dropdown to the **servoOn** variable with a value of 0. We'll use this a little bit later.



## Use the Angle Variables

Let's put those variables to use. From the **VARIABLES** category, drag a **minAngle** and **maxAngle** into the **servo write pin A1** blocks in your **forever** loop.

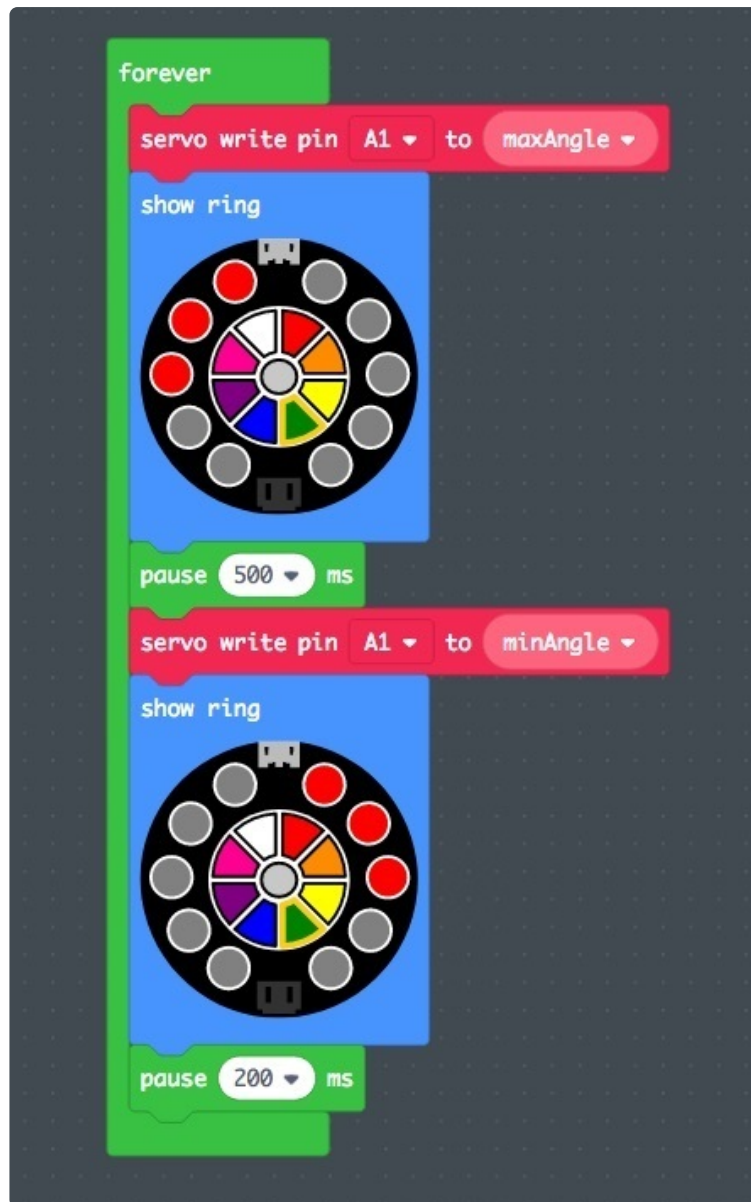


## Add Lights

We'll add some NeoPixel lighting now. We can make the CPX LEDs light up to indicate the direction of travel.

From the **LIGHT** category, add a couple of **show ring** blocks as shown here. You can click the circles twice to "blank" them into these patterns.

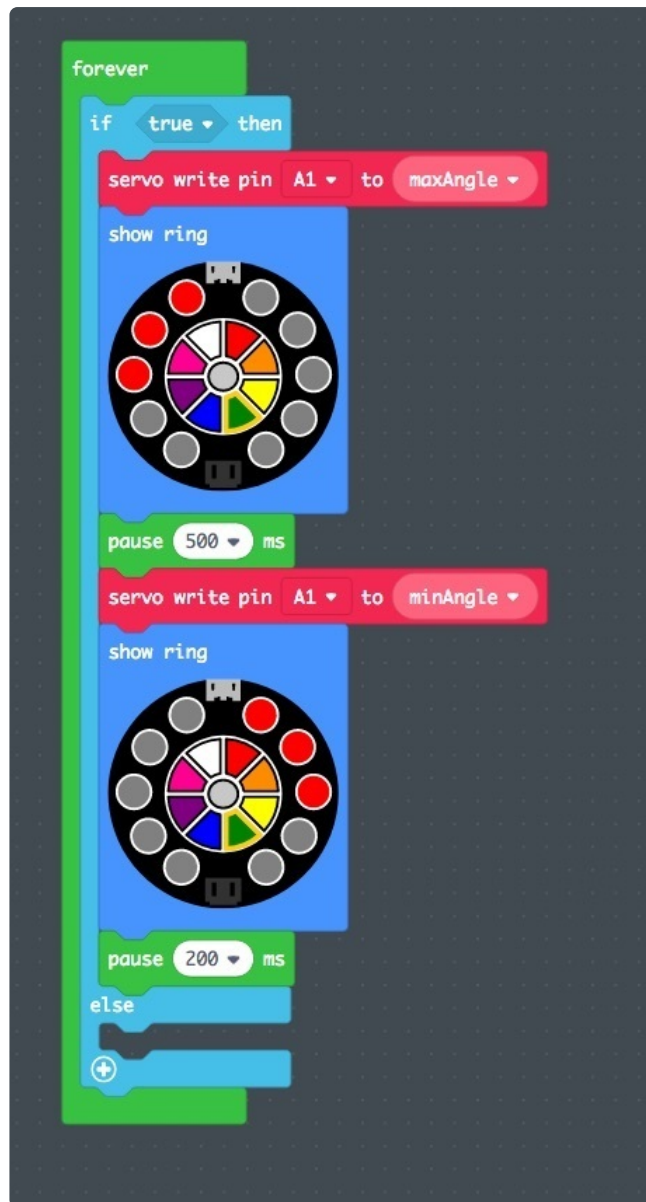




## Mode Switch

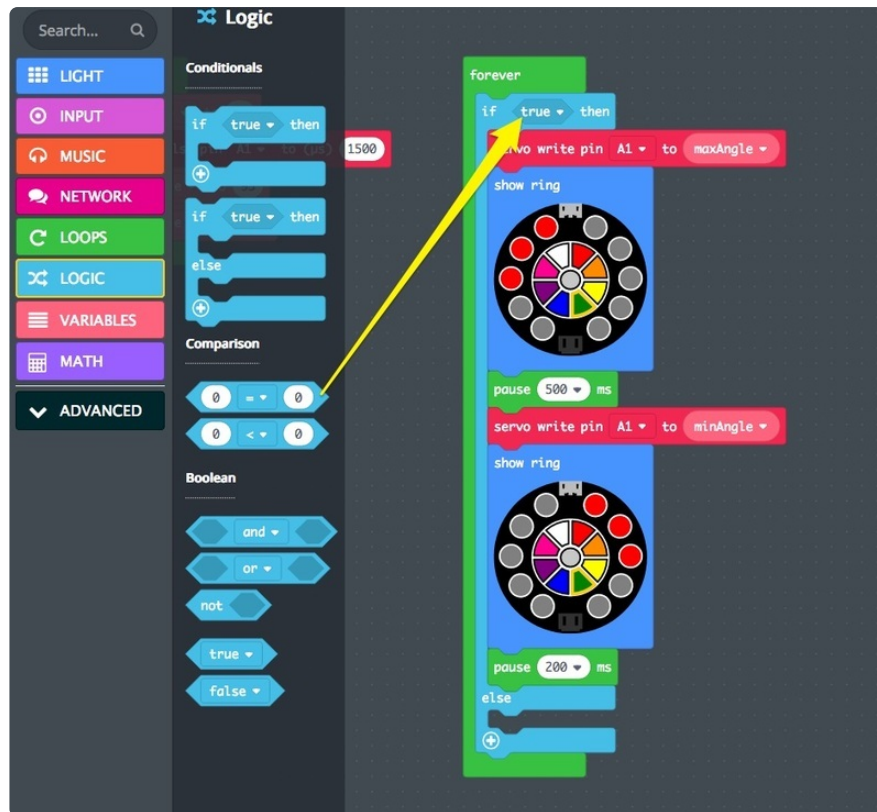
You may get tired of the servo swinging back and forth, so next you can set up a mode switch. This will read the small selector switch on the CPX. If it is moved to the right, the servo swings back and forth, if not, it won't!

Get an **if true then else** block from the **LOGIC** category. Transfer the contents of the **forever** block into the if section as shown.

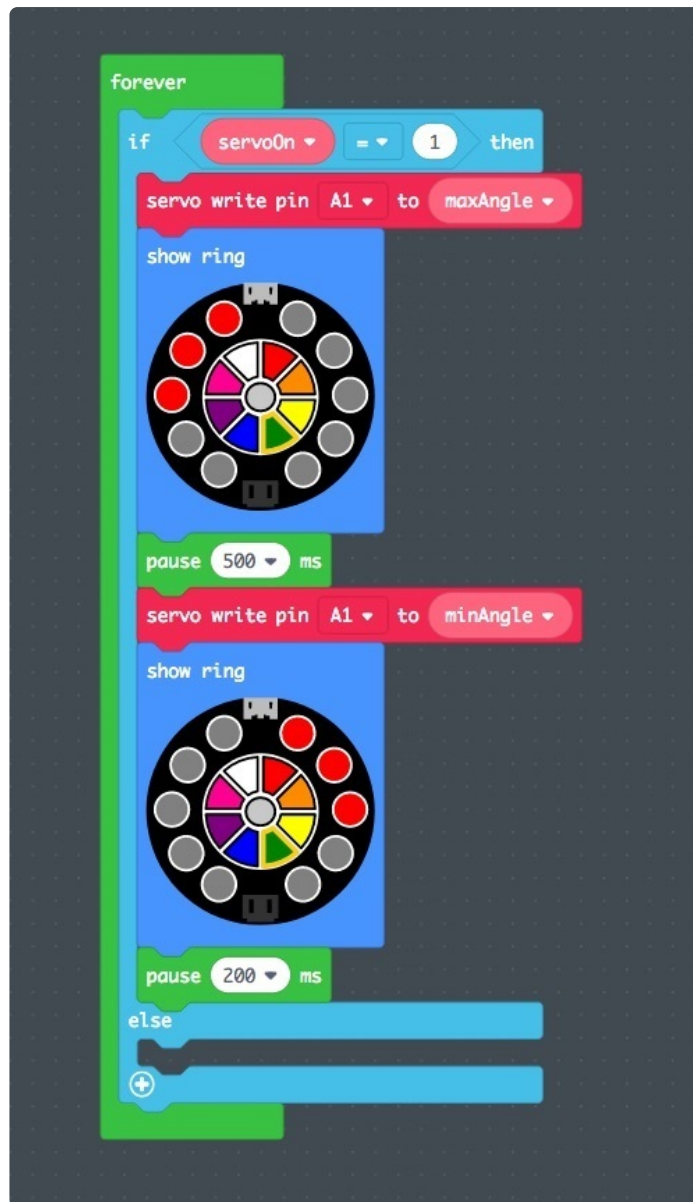


## Conditions

We now need some condition to be met that will be tested by **if** statement. From the **LOGIC** category get a **0 = 0** comparison block and move it onto the **true** block.



Drag in a **servoOn** variable from **VARIABLES** and set the value to 1.

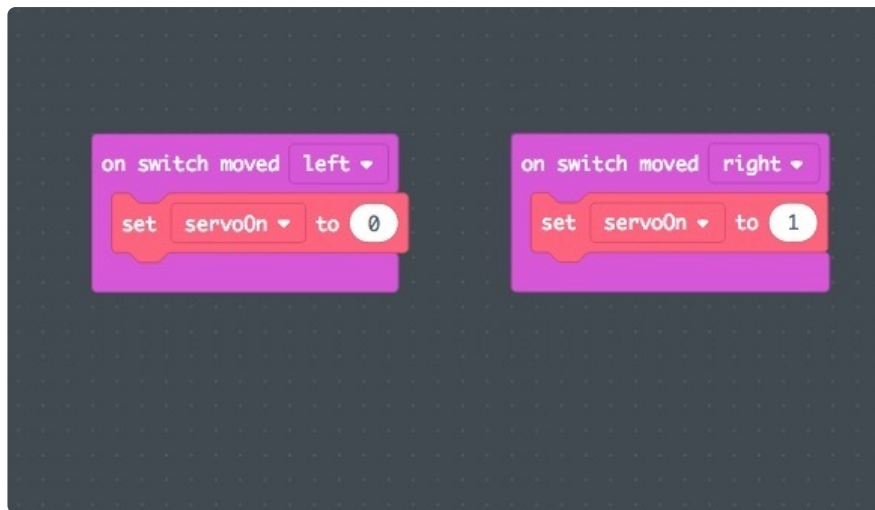


## Switch Reading

We will now set up blocks to read the switch position, and change the **servoOn** value accordingly.

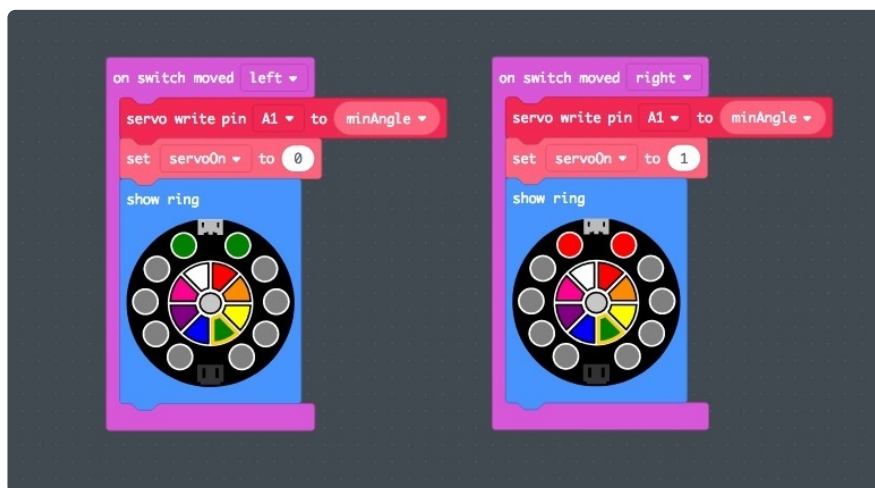
From **INPUT**, drag in two **on switch moved left** blocks. Change one of them to **right**.

Add from the **VARIABLES** category a **set servoOn to** block to each switch block, setting the right one's value to 1.



Now, when you flip the switch left and right you'll toggle the value of the **servoOn** variable!

To give visual feedback when the mode switch is flipped, add **show ring** blocks as seen here, as well as resetting the servos to their minimum angles.

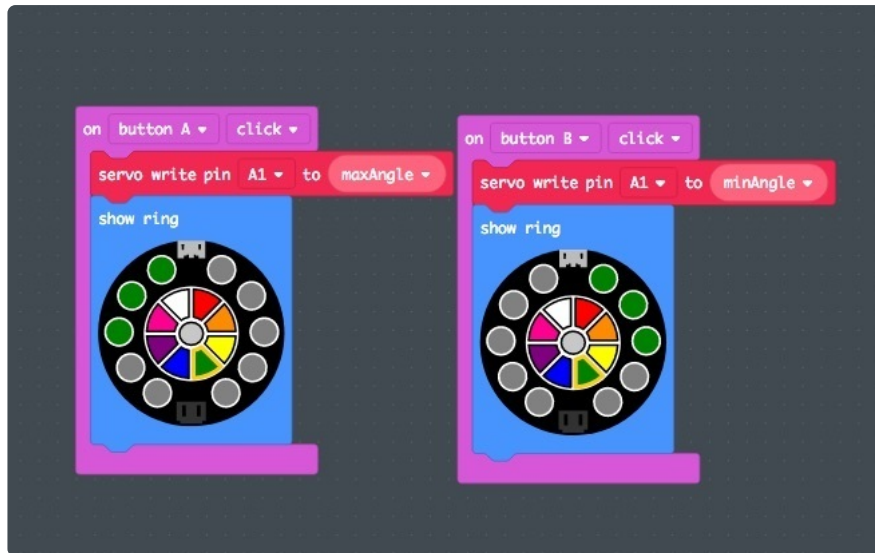


## Button Clicking

The final element is to add button control. from **INPUT** get a couple of **on button A click** blocks. Change one to **button B**.

Add to these a **servo write pin A1** each, and a **show ring** as seen here.





The program is complete!

Here's the final code, you can now download it to your computer and then reset the CPX and drag the .uf2 program file onto the CIRCUITPY drive:

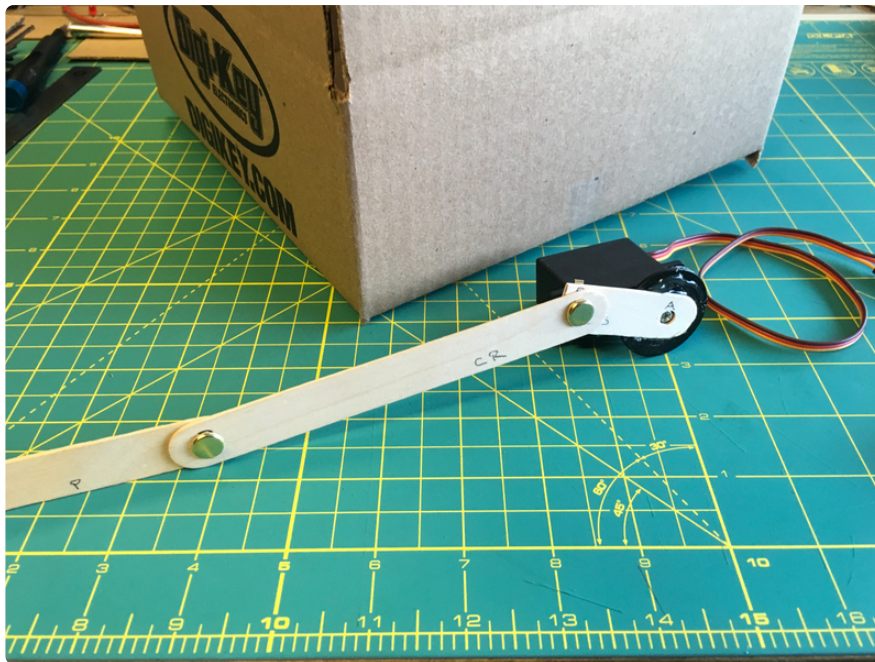
Now, we'll build the base and guide to secure the mechanism onto our cardboard box.

---

## Mount the Slider Crank

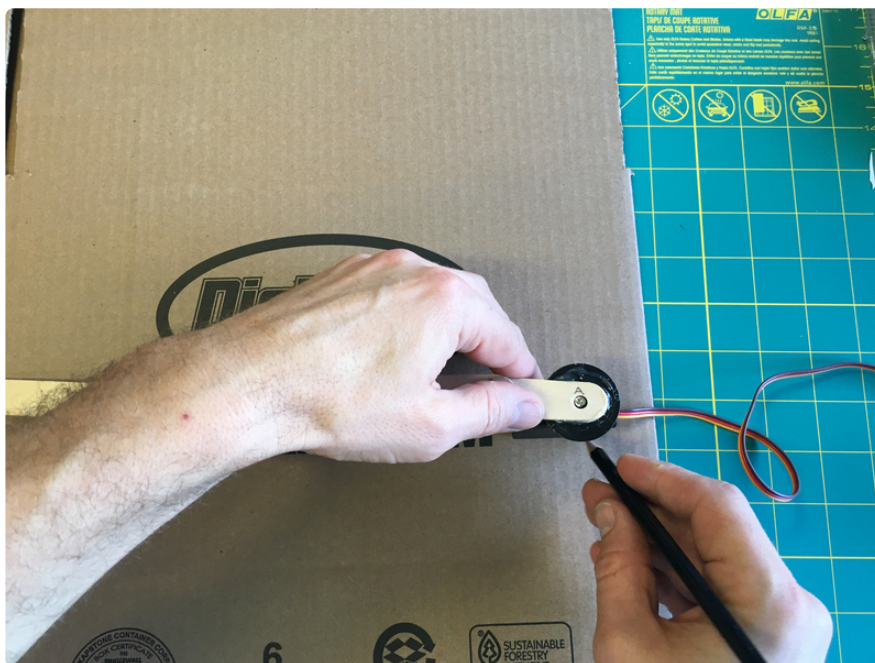
There are two fixed points in this mechanism -- the servo crank, and the guide block that will constrain the slider (otherwise it could flop all around!) -- we'll use cardboard to build the structure.

Start with a corrugated cardboard box, such as a medium shipping box.

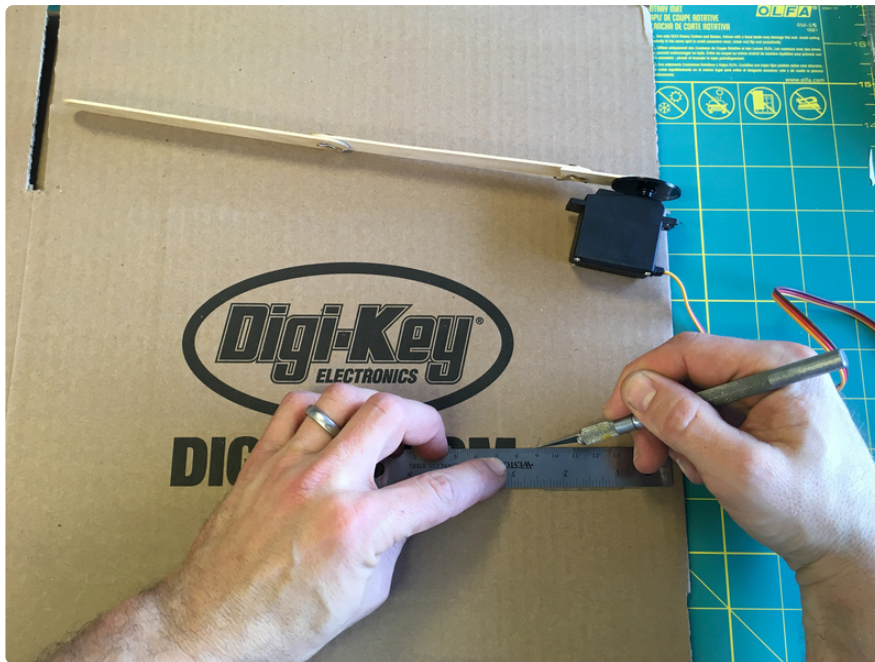


## Servo Attachment

Mark the servo base onto the box using a pencil or pen. Keep the markings very close to the true dimensions of the servo, so that you can cut a snug opening for it to press fit.



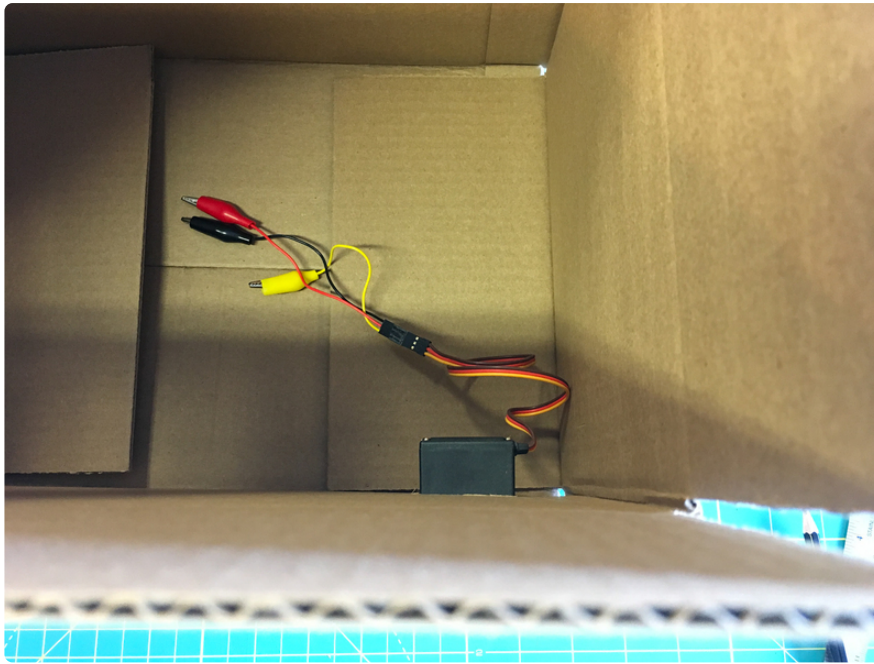
Use a hobby knife to cut out the servo opening. It's best to use a metal ruler or straight edge to guide the knife edge.



Insert the alligator clip wires into the opening, then press the servo body in.







## Guide Block

Now that the servo is fixed in place, we can determine the best position for the guide block.

With the crank rotated fully counterclockwise to its stopping point (180 degrees on the servo) extend and straighten the linkage as shown. Mark the top and bottom of the slider onto the box using a pencil or pen.



These two marks are where you will build up a small cardboard platform the rest the slider, and then two walls of cardboard for the upper and lower guides.







Cut four or five small pieces of cardboard to fit the markings  
 Rest the slider on them to make sure it is level with the servo horn -- this is necessary because of the offset of the servo from the box, as well as the slack in the system at the joints  
 Glue down the cardboard pieces in a stack



Place two more piece of cardboard like guide rails on the upper and lower "walls" of the stack. These should guide the slider, but not impede its motion.



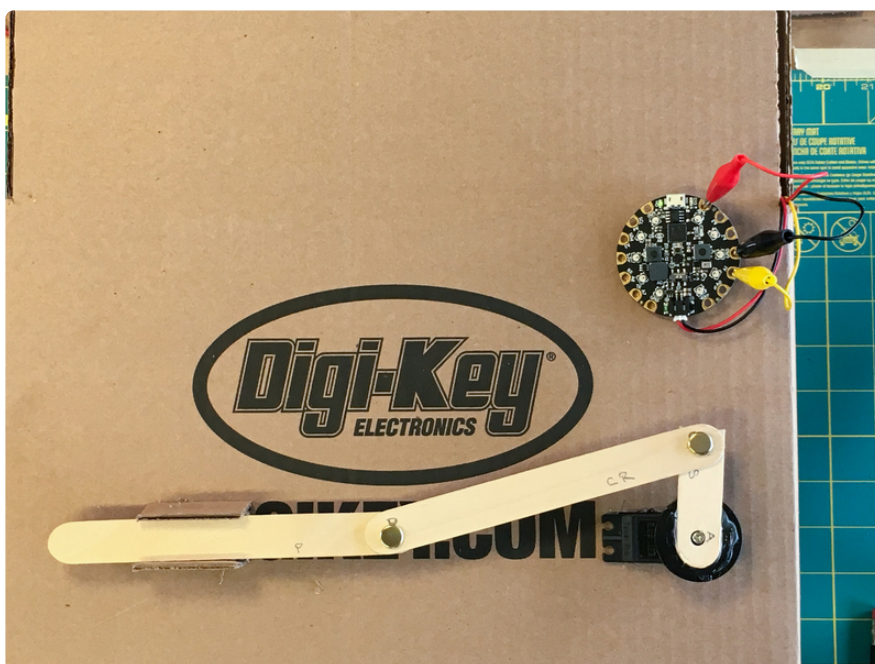




## Connect the Circuit Playground Express

Now, you can reconnect the alligator clips to the CPX as shown, by running the wires up from inside the box to the edge. Or you can poke a hole in the cardboard wherever you like -- that's one of the great features of cardboard!

Connect the battery pack to the CPX as well, using the JST connector. You can mount the battery pack inside the box with some tape.





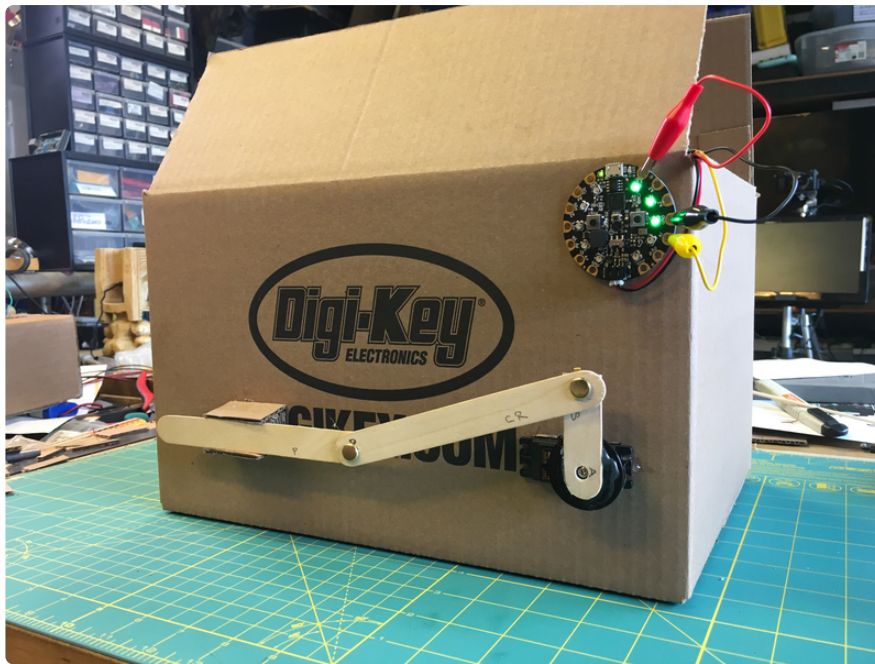
## Fire it Up!

Turn on the battery pack's power switch and away we go! Move the slider to chose between automatic or manual mode, where you can press the A and B buttons to actuate the slider.



If your motor struggles to turn check that all of the pivots can move freely and consider hooking the red alligator clip to VOUT on the Circuit Playground Express instead of 3.3V

You can mount the Circuit Playground Express to the box with a loop of masking tape, as I've done here, or get more creative using bamboo skewers as rivets, pipe cleaners, or double stick foam tape!



With the slider crank mechanism under your control you can now find some creative uses for it! You could attach a puppet, make a primitive train, or perhaps a lumbering box robot! Now that you have the power to convert rotation to linear motion, what will you make?