Stroboscopic Zoetrope

Created by John Park

https://learn.adafruit.com/strobe-zoetrope

Last updated on 2023-08-29 03:46:18 PM EDT
# Table of Contents

## Overview
- Parts
- Materials & Tools

## Make the Zoetrope
- Create an Animation Cycle
- Draw Faces on Ping Pong Balls
- Make a Spinning Disk Platform
- Divide into Six Segments
- Head Attachments
- Attach to Motor

## Crickit Setup
- Mount the Zoetrope
- Mount the Motor
- Light Angle
- Secure the Crickit

## Program Crickit and CPX with MakeCode
- Crickit Install for MakeCode
- Flashing Lights
- Buttons, Switches, Variables, and Modes
- Switch Mode
- Buttons

## Animate!
- Change It Up!
Overview

A zoetrope is a type of animation machine. It relies on the principle of persistence of vision -- just like turning the pages rapidly of a simple flip book or watching an animated film go by 24 frames per second on a projector — persistence of vision is an illusion that fools our brains into interpreting rapid changing images as animated motion.

You can build one that is purely mechanical, where small slits in a rotating disc cause our eyes to see rapidly blinking images, or you can use quickly blinking (stroboscopic) light instead.

In this project we will use the Circuit Playground Express and its NeoPixels to very rapidly blink light. We'll use the Cricket to spin a motor to drive a disk onto which our animated objects are attached.

You will use the buttons on the Circuit Playground Express to increase and decrease the speed of the blinking and of the motor in order to get the stroboscopic effect to synchronize.

Parts

1 x Adafruit CRICKIT for Circuit Playground Express

https://www.adafruit.com/product/3093
### Round, Awesome Microcontroller Board

<table>
<thead>
<tr>
<th>Item</th>
<th>Quantity/Description</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 x Circuit Playground Express</td>
<td></td>
<td><a href="https://www.adafruit.com/product/3333">https://www.adafruit.com/product/3333</a></td>
</tr>
<tr>
<td>1 x TT Motor DC Gearbox</td>
<td>200 RPM 3 to 6VDC</td>
<td><a href="https://www.adafruit.com/product/3777">https://www.adafruit.com/product/3777</a></td>
</tr>
<tr>
<td>1 x TT Motor Pulley</td>
<td>36mm Diameter</td>
<td><a href="https://www.adafruit.com/product/3789">https://www.adafruit.com/product/3789</a></td>
</tr>
<tr>
<td>1 x Male DC Power adapter</td>
<td>2.1mm plug to screw terminal block</td>
<td><a href="https://www.adafruit.com/product/369">https://www.adafruit.com/product/369</a></td>
</tr>
<tr>
<td>1 x Alkaline AA batteries</td>
<td>3 pack</td>
<td><a href="https://www.adafruit.com/product/3521">https://www.adafruit.com/product/3521</a></td>
</tr>
<tr>
<td>1 x USB Cable</td>
<td>A/MicroB - 3ft</td>
<td><a href="https://www.adafruit.com/product/592">https://www.adafruit.com/product/592</a></td>
</tr>
<tr>
<td>1 x 3 x AA Battery Holder</td>
<td>with 2.1mm Plug</td>
<td><a href="https://www.adafruit.com/product/3842">https://www.adafruit.com/product/3842</a></td>
</tr>
</tbody>
</table>

### Materials & Tools

You'll just need a few common materials and tools to build the zoetrope disc, stand, and animated faces:

- 6x ping pong balls
- Marker or paint pen
- Thin, stiff cardboard (chipboard) for the spinning disk
- Small cardboard box to mount the motor, Crickit, and battery pack
- Pencil
- Scissors
- Compass
- Hobby knife
- Hot glue and hot glue gun
- Small philips screwdriver
Make the Zoetrope
Create an Animation Cycle

A simple six frame animation cycle is what we'll be using for our zoetrope. Here are three really great, expressive examples from everybody's favorite animator, Joe Bowers! Alongside many feature animation credits at Disney and DreamWorks, Joe is the creator of the famous Dovahbear animation series, the beloved Blarfish, and others. He was kind enough to contribute these three cycles for use in this project. Thanks Joe!
When creating your own cycles, you will want them to hook up from the last frame to the first frame, since they will be cycling over and over. Take a look at the examples above to see how the last frame of each cycle leads back into the first frame. In the second animation, note how the extreme hits on frame four or five and then the smile gets smaller on frame six in preparation for the smallest mouth shape of frame one.

**Draw Faces on Ping Pong Balls**

Using a marker, draw your animation frames on each of six ping pong balls. Try to keep the same relative position and size of the drawings so that things don't jitter around too much once the zoetrope is spinning and your eye will see each drawing essentially in the same position.
Make a Spinning Disk Platform

Next, we'll create a platform for the heads. Using a compass, draw a circle with a diameter of 6-1/2" onto a piece of chipboard (thin, stiff cardboard). To do so, make the opening of your compass half of that desired diameter, or 3-1/4". Don't cut out the circle just yet, however.
Divide into Six Segments

It's very important that the ping pong ball heads are evenly spaced. Otherwise the stroboscopic effect will be difficult-to-impossible to synchronize. In order to do this, we'll divide the circumference of the circle into six even arcs.

The simplest way to do this is to set the pivot point of your compass at any point on the circle's edge and create another identically sized circle. So make sure your compass is still set at the original measurement of 3-1/4" before drawing the second circle.

Now, pick up the compass, place its pivot point on one of the two intersections you just created with the original circle. Draw another circle.

Repeat this process going all the way around the original circle. Look at all of the intersection points between the new circles and the original one -- these are perfectly spaced and now give us our six even sections!
Use a straight edge to connect three pairs of opposite intersection points on your original circle, you will create the perfect indicators of your six sections.

Now, we'll create one more measurement indicator so we know where to exactly place each ping pong ball. Use the compass to again draw a circle starting at the center of our disk, but this time shorten the radius distance to 2". Each intersection with your six lines marks the spot where a ball will go.
Using scissors or a hobby knife, cut out the circle disk.
Head Attachments

Now, use hot glue to attach each ball at one of the six intersections, with the faces facing outward from the center. Make sure to place the balls in the same order as your animation cycle!
Attach to Motor

Push the motor pulley screw through the top of the disk. This will help you center the pulley.

Put a few spots of hot glue on the pulley as shown, and then affix it to the disk.
Once the glue has cured, unscrew the screw, push the hub onto the motor shaft, then drive the screw back into place to secure it.
Crickit Setup

Plug the motor leads into the Crickit's Motor 1 port. Connect the red wire to the outer terminal and the black one to the inner terminal.

You can create other disks with different animation cycles on them, simply unscrew and drop on the new ones when its time to change them out!

Also, plug in the battery pack to the Crickit's DC power jack.
Mount the Zoetrope

For a mounting base, you can use a small cardboard box, such as the one from which you unpacked AdaBox 008!
Mount the Motor

Flip the box over (we'll use the opening flap to mount the Crickit and CPX in a moment), and poke a hole in the center with a pencil to accommodate the motor shaft as shown.

Us a bit of double stick foam tape to secure the motor onto the box as seen here, making sure the shaft is free to move easily.
Tuck the battery pack inside the box -- it's weight will serve to stabilize the zoetrope base.
You can use another piece of double stick foam tape on the flap -- we'll secure the Crickit there in a moment.

**Light Angle**

We can angle the flap so that the Crickit and CPX are pointing at the ping pong balls, yet still easy to access the buttons and switches on board. Open the flap about 20° and then use a Makedo Scru to lock in the flaps as shown here. (You can use a toothpick or other material to hold the flaps in place if you like. Making it easily removable will allow for battery changes later.)
Repeat this for the other side.

**Secure the Crickit**

Now, you can use that second piece of foam double stick tape to secure the Crickit to the flap as shown here. You could substitute zip ties, pipe cleaners, screws, or other hardware if you choose.

Note the orientation -- this sideways rotation allows easy access to the USB port on the CPX for programming.
You've built your stroboscopic zoetrope, it's time to program it!
Program Crickit and CPX with MakeCode

Crickit Install for MakeCode

You can code your Crickit and Circuit Playground Express using MakeCode! [Head on over to the Crickit with MakeCode guide to get set up! ()](

The Stroboscopic Zoetrope code will need to do the following tasks:

- Spin the DC motor connected to the Crickit
- Allow the motor speed to be increased and decreased with the buttons on the Circuit Playground Express (CPX)
- Rapidly flash the NeoPixel ring on the CPX
- Allow the flashing speed to be increased and decreased with the buttons on the CPX
- Switch button modes using the slide switch on the CPX

Let's start off with a simple version that can strobe at a fixed rate and run the motor. Later we'll add some fancy controls.

Flashing Lights

The first thing we'll do is turn all of the NeoPixels white by adding a `show ring` block to an `on start loop` block.
Then, we'll flash the LEDs by turning their brightness up high, pausing, setting the brightness to zero, pausing, and repeat!

We also need to run the the DC motor — we can start by setting it to run at, say 80% speed. Use the Crickit category's `crickit run motor 1` at 80% block.
Try downloading and running this code now. Turn on the Crickit power and you should see the disk spin and the lights blink rapidly!

You probably won’t get the strobe effect, since it requires a bit of tuning. Instead of hand tuning the numbers in MakeCode, re-uploading and iterating this way, we’ll add controls so that the buttons on the CPX can be used to change the motor speed and strobe rate.

**Buttons, Switches, Variables, and Modes**

Here’s how we’ll do it -- three variables are created:

- `switch_mode`
- `strobe_rate`
- `motor_speed`

These are initialized in the on start block as shown. Note how instead of simply typing a number, we'll use a true block for the `switch_mode` variable's initial value. This makes the variable a "boolean" which means it can either be `true` or `false`. 
Switch Mode

Then, we'll get an `on switch moved left` block and add the `set switch_mode to` block from the variable category. Make the value a `true` block from the logic category.

Duplicate this block and change it to `on switch moved right`, and `set switch_mode to false` as seen here.

Now, when you toggle the Circuit Playground Express switch you'll be flipping the state of `switch_mode` between `true` and `false`. 
We'll used `true` to represent strobe mode, and `false` to represent motor mode for the A and B buttons.

**Buttons**

Finally, we'll set up both buttons. From INPUTS, bring in a `on button A click` block and change it to `down` instead of `click`. Add to it an `if true then else` block from the LOGIC category so the button can be used to change either `strobe_rate` or `motor_speed` depending on the switch mode.

Bring in a `0 = 0` comparison block from the LOGIC category to replace the simple `true` check, then change the first `0` to a `switch_mode` variable and the second one to `true` as shown here.

![Button Configuration](image)

In the `if` section, bring in a `change by` block from the VARIABLES category and choose `strobe_rate` and a value of `0.5`. This means the strobe pause will increase by 0.5 milliseconds (ms) each time the button is pressed when the switch is in strobe mode, thus slowing down the strobe.

Repeat this for the `else` section, but change to `motor_speed` and a negative change of `-0.5`. This means the motor speed will slow down by half a percent each time the button is pressed.
Now, duplicate this whole button input and change it to be **button B** the change values as shown. This way, the B button will reduce the strobe delay, thus speeding up the strobing, and can increase the motor speed when in motor speed mode.

You can click the button below to edit a live copy of the code.

[Download the MakeCode](#)

Here's what the final program looks like.
Upload this new code to your CPX and you're ready to run the zoetrope!

Animate!

Time to animate! First, make sure your room is dark -- the darker it is, the better the effect will be. Then, turn on the Crickit's on/off switch. The disk will spin and the lights will flash, but it may not yet look like the animation you expect.

To tune the sync of the spinning and flashing, you will want to increase and decrease the motor speed and the LED flashing rate. Flip the switch on the Circuit Playground Express to the left, and then try hitting the B button (on the right) a few times to increase the flashing speed until the strobe syncs with the motor. It may appear dimmer as it gets very fast, but this is OK, don't worry.

You can then flip the switch to the right and use the buttons to increase or decrease the motor speed to fine tune it.

When it starts to sync you may notice the faces are animating, but they appear to be rolling off to the left or right. Try tuning motor speed up or down to lock it in!

Change It Up!

Feel free to use one of the other animations provided or make your own. Have fun with your Zoetrope!