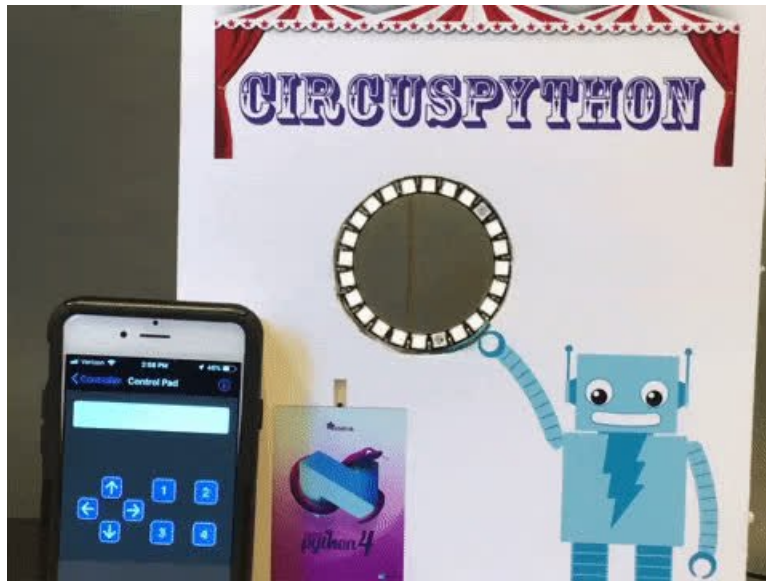


# CircusPython: Jump through Hoops with CircuitPython Bluetooth LE

Created by Dan Halbert



Last updated on 2019-01-30 11:14:11 PM UTC

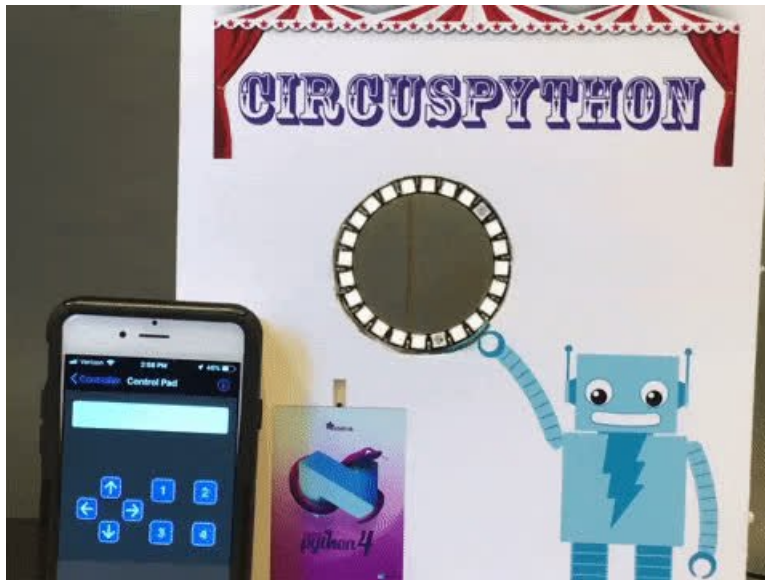
## Guide Contents

|  |    |
|--|----|
| Guide Contents                                       | 2  |
| Overview   | 3  |
| Parts  | 3  |
| Other Materials and Tools                            | 4  |
| Demonstration during Adafruit Show & Tell            | 4  |
| Build the Circus                                     | 5  |
| Make the Front Panel                                 | 5  |
| Wire the NeoPixel Ring                               | 6  |
| Mount the Parts                                      | 6  |
| Attach the Servo Arm                                 | 8  |
| CircuitPython Code                                   | 9  |
| Getting Familiar                                     | 9  |
| Download Library Files                               | 9  |
| CircuitPython Program                                | 10 |
| How the Program Works                                | 11 |
| Control the Circus with the Bluefruit LE Connect App | 12 |
| Download the App                                     | 12 |
| Connect to CircusPython                              | 12 |
| Troubleshooting                                      | 12 |
| Device Menu  | 12 |
| Controller Menu                                      | 13 |
| Move the Servo Arm                                   | 14 |
| Change Color   | 15 |
| Wow!   | 16 |

## Overview

Here's a simple demonstration project to get you started with Bluetooth Low Energy (BLE) support in CircuitPython. Blinka jumps through a NeoPixel ring of fire, using a servo. It's all controlled wirelessly via the Adafruit Bluefruit LE Connect app from your phone or tablet, using a Feather nRF52840 Express and a CRICKIT FeatherWing.

You can adapt and expand upon this code for your own projects. Add more servos, add sound, whatever you want. It's up to you!



## Parts

---

### 1 x [Feather nRF52840 Express](#)

A Feather board that supports BLE, running CircuitPython.

OUT OF STOCK

---

### 1 x [Adafruit CRICKIT FeatherWing](#)

Use with any Feather board

ADD TO CART

---

### 1 x [5V 2A \(2000mA\) switching power supply - UL Listed](#)

You'll need this or another 5V power supply with a 2.1mm plug for the CRICKIT.

ADD TO CART

---

### 1 x [NeoPixel Ring - 24 x 5050 RGB LED with Integrated Drivers](#)

A NeoPixel ring for light

ADD TO CART

---

### 1 x [Small Alligator Clip Test Leads](#)

Cut these in half and solder one end to the NeoPixel Ring terminals.

ADD TO CART

---

### 1 x [Male/Male Jumper Wires](#)

Connect from the CRICKIT board to the alligator leads on the NeoPixel ring.

ADD TO CART

---

1 x [Standard servo - TowerPro SG-5010 - 5010](#)

A smaller servo would probably work too.

ADD TO CART

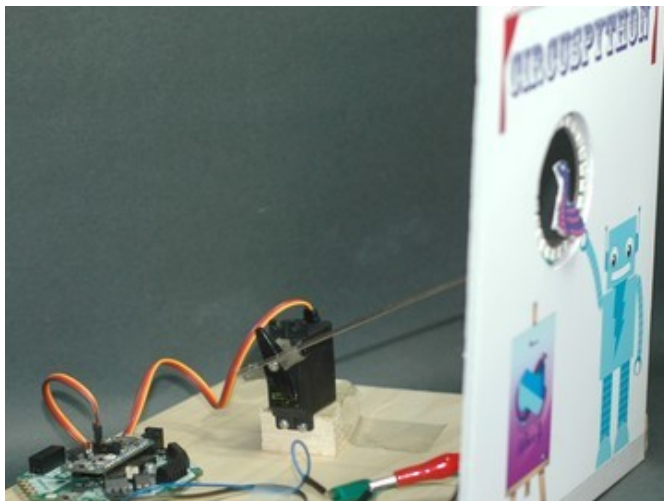
---

## Other Materials and Tools

- A wood base, about a foot square.
- A scrap of wood, for mounting the servo. I used wood screws to hold the servo on by one set of its ears.
- Foam core board, for mounting the front panel.
- A rod, such as a piece of coat hanger, a chopstick, or even a straw. I used a thin piece of metal salvaged from a discarded windshield wiper, so I could bend one end to mount Blinka more easily.
- Clear packing tape and office tape.
- Hobby knife.
- Glue stick.
- Wire strippers.

Demonstration during Adafruit Show & Tell

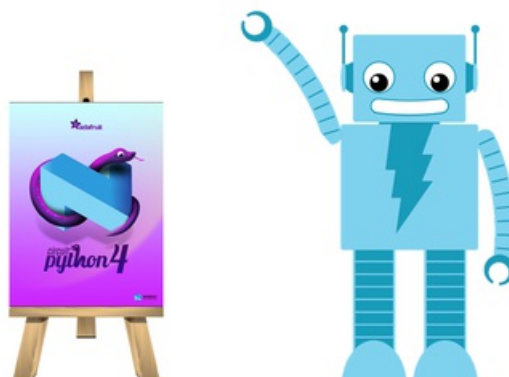
## Build the Circus



This project was built as quick demonstration of Bluetooth LE support in CircuitPython in about 90 minutes. So I cut a lot of corners on the construction, using tape liberally.

### Make the Front Panel

I used Microsoft PowerPoint for placing and scaling clip art and Adafruit graphics. After printing it on cardstock, I cut a hole just large enough for the NeoPixel ring by tracing around it and then using a hobby knife. Here is the PowerPoint file, and also PDF, and PNG versions, but you can be creative and make your own!



<https://adafru.it/DLh>

<https://adafru.it/DLh>

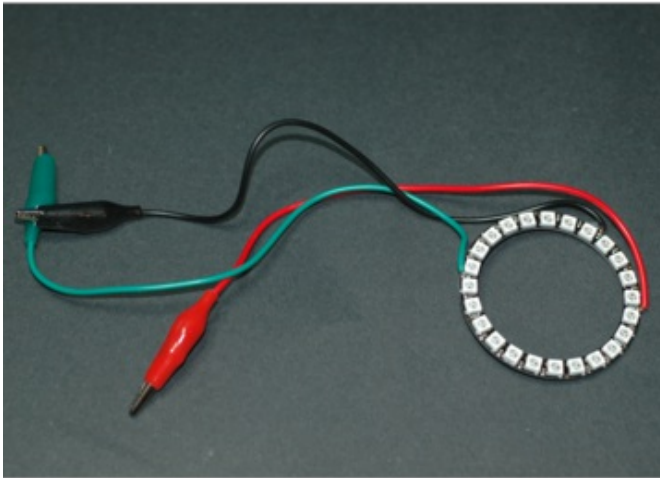
<https://adafru.it/DLi>

<https://adafru.it/DLi>

<https://adafru.it/DLj>

<https://adafru.it/DLj>

## Wire the NeoPixel Ring



Next, wire alligator clips to the NeoPixel ring. Cut a **red**, a **black**, and some other color alligator clip lead in half. Strip a small amount of insulation off the ends, enough to stick through the connection holes on the NeoPixel ring. Twist the stranded ends. Put the black wire in a **GND** hole, the red wire in a **PWR +5V** hole, and the other color in the **DATA INPUT** hole. Bend the wires over, solder, and trim.

The NeoPixel ring pictured has the wires coming out the front of the ring. This was what I had on hand, but for this project, coming out the back is better.



Alternatively, you can solder single pin headers to the holes on the ring. Pictured is a smaller ring using this technique.

## Mount the Parts

Tape the front panel to the bottom of the board, using one piece of tape on the front wrapped around to the bottom of the board, and one on the back attaching the panel to the top of the board.

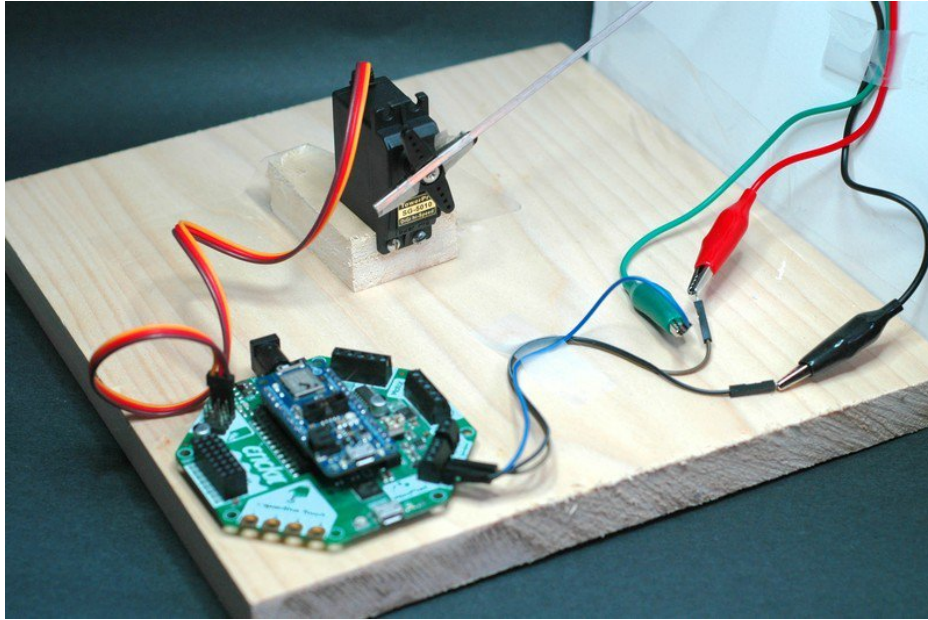
Put the NeoPixel ring into the front panel hole from the back, and tape it there.

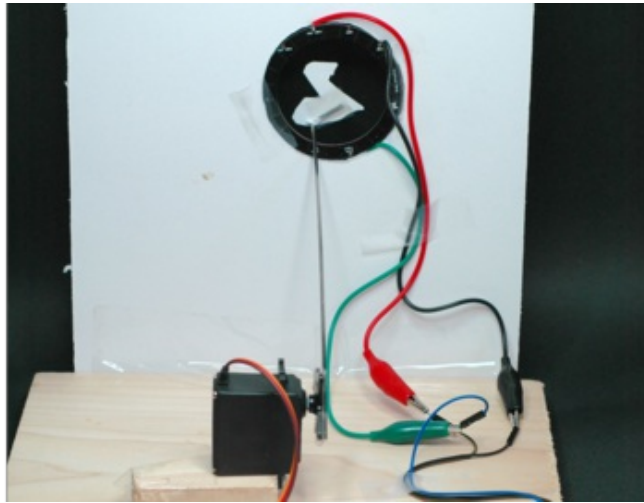
Fasten the servo to a scrap piece of wood by putting a couple of wood screws through its ears on one side. Then tape the scrap to the base with packing tape.

Fasten the CRICKIT to the base with hook-and-loop tape, screws, or double-sided sponge tape. I used hook-and-loop for easy removal.

Plug the servo into Servo **Port 1** on the CRICKIT, with the yellow wire pointing to the outside. Attach three jumper wires to the NeoPixel screw terminals on the CRICKIT. Connect a **red** one to **+5V**, a **black** one to **GND**, and some **other color** to the **signal pin** (the arrow). Connect the alligator clips to their appropriate colors

Dress the wiring with tape for neatness.

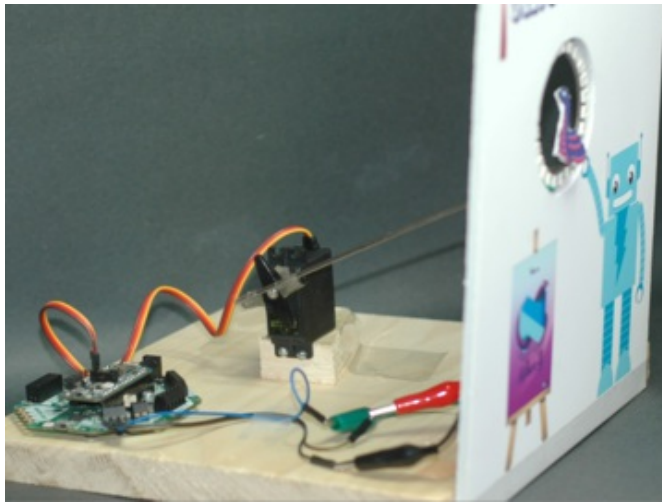




## Attach the Servo Arm

Before you attach the servo arm, you need to set the servo to 0 degrees. Your servo should have come with a variety of plastic "horns", with mounting screws and possibly small metal sleeves for the screws. Slip a horn onto the servo shaft without a screw and turn the shaft clockwise until it stops. Don't force it. There will be a little resistance, but if you're not sure, try going back and forth to feel how it moves.

Then take the rod mentioned in the parts list and tape it to the horn, so that it points through the front panel hole. Reposition the horn if necessary, and then screw it on. Adjust the length of the rod so it will clear the top of the hole when it moves up.



Now cut out a Blinka picture and attach it to the end of the rod with tape. The PowerPoint file below has several Blinka pictures in various sizes (you can adjust the sizes), or find one on the [Adafruit website](https://adafruit.com) (<https://adafruit.com>).

<https://adafruit.com>

<https://adafruit.com>

OK, you're ready to try out the program now!



# CircuitPython Code

## Getting Familiar

CircuitPython is a programming language based on Python, one of the fastest growing programming languages in the world. It is specifically designed to simplify experimenting and learning to code on low-cost microcontroller boards. Here are some guides which cover the basics:

- [Welcome to CircuitPython! \(https://adafru.it/cpy-welcome\)](https://adafru.it/cpy-welcome)
- [Introducing the Adafruit nRF52840 Feather \(https://adafru.it/DLI\)](https://adafru.it/DLI)

Be sure you have the latest CircuitPython loaded onto your board per the second guide.

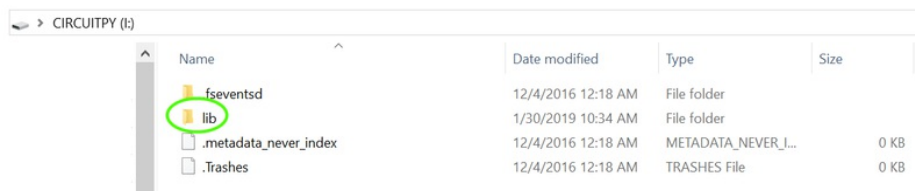
CircuitPython is easiest to use within the Mu Editor. If you haven't previously used Mu, [this guide will get you started \(https://adafru.it/ANO\)](https://adafru.it/ANO).

## Download Library Files

Plug your nRF Feather board into your computer via a USB cable. Please be sure the cable is a good power+data cable so the computer can talk to the board.

A new disk should appear in your computer's file explorer/finder called **CIRCUITPY**. This is the place we'll copy the code and code library. If you can only get a drive named **FTHR840BOOT**, load CircuitPython per the guide above.

Create a new directory on the **CIRCUITPY** drive named **lib**.



Download the latest CircuitPython libraries to your computer using the green button below. **Match the library you get to the version of CircuitPython you are using.** Save to your computer's hard drive where you can find it.

Note: This project uses the Bluetooth functionality available in CircuitPython 4.0.0-beta.1 and later, on nRF52840 boards such as the Feather nRF52840 Express. Make sure you are using that version or later.

<https://adafru.it/y8E>

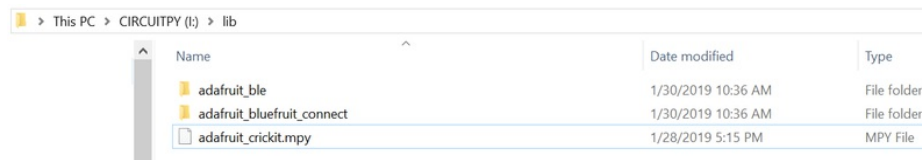
<https://adafru.it/y8E>

With your file explorer/finder, browse to the bundle and open it up. Copy the following folder from the library bundle to your **CIRCUITPY lib** directory you made earlier:

- **adafruit\_ble** (folder)
- **adafruit\_bluefruit\_connect** (folder)
- **adafruit\_crickit.mpy** (file)
- **adafruit\_motor** (folder)

- `adafruit_seesaw` (folder)

All of the other necessary code is baked into CircuitPython!



## CircuitPython Program

Download the `circuit.py` program below and save it on the **CIRCUITPY** flash drive as `code.py`. As soon as CircuitPython restarts, the servo arm will move into position, away from the NeoPixel ring.

```
# CircusPython!
# For use with the Adafruit BlueFruit LE Connect app.
# Works with CircuitPython 4.0.0-beta.1 and later running on an nRF52840 board.

import random
import time

from adafruit_crickit import crickit
from adafruit_ble.uart import UARTServer

from adafruit_bluefruit_connect.packet import Packet
# Only the packet classes that are imported will be known to Packet.
from adafruit_bluefruit_connect.color_packet import ColorPacket
from adafruit_bluefruit_connect.button_packet import ButtonPacket

# Initialize the NeoPixel ring to a fire color, not too bright.
crickit.init_neopixel(24, brightness=0.1)
color = (25, 12, 0)

# Creates a sparkly "fire"-like effect.
def sparkle():
    crickit.neopixel[random.randrange(24)] = (0, 0, 0)
    crickit.neopixel[random.randrange(24)] = color
    crickit.neopixel[random.randrange(24)] = color
    crickit.neopixel[random.randrange(24)] = color

uart_server = UARTServer()

# Increase this to slow down movement of the servo arm.
DELAY = 0.0

# Angle for Blinka before jumping through the ring.
UP_ANGLE = 50

# Go to this angle when jumping through the ring. Adjust
# slightly as necessary so you don't bump into the ring.
DOWN_ANGLE = 2

advertising_now = False
crickit.servo_1.angle = UP_ANGLE
angle = UP_ANGLE

while True:
```

```

while True:
    sparkle()
    if not uart_server.connected:
        if not advertising_now:
            uart_server.start_advertising()
            advertising_now = True
        continue

    # Connected, so no longer advertising.
    advertising_now = False

    if uart_server.in_waiting:
        packet = Packet.from_stream(uart_server)
        if isinstance(packet, ColorPacket):
            # Change the fire color.
            color = packet.color
        elif isinstance(packet, ButtonPacket):
            if packet.pressed:
                if packet.button == '5' and angle != UP_ANGLE:
                    # The Up button was pressed.
                    for a in range(angle, UP_ANGLE+1, 1):
                        crickit.servo_1.angle = a
                        # Sparkle while moving.
                        sparkle()
                        time.sleep(DELAY)
                    angle = UP_ANGLE
                elif packet.button == '6' and angle != DOWN_ANGLE:
                    # The Down button was pressed.
                    for a in range(angle, DOWN_ANGLE-1, -1):
                        crickit.servo_1.angle = a
                        # Sparkle while moving.
                        sparkle()
                        time.sleep(DELAY)
                    angle = DOWN_ANGLE

```

## How the Program Works

The first part of the program sets up a `UARTServer`, which is a BLE service that wirelessly sends streams of characters between a BLE device and a client computer (which could be a phone or tablet). The program then starts advertising its UART service to anyone listening. The client computer receives the advertisement, and tells the user it's available. The user can then choose to connect as a client of the service.

Once a connection is established, the program code waits for incoming command packets from the client, specifically either a `ColorPacket` or a `ButtonPacket`. When one arrives, the program either adjusts the NeoPixel color, or sees which button has been pressed: up-arrow or down-arrow. If it's a button press, the code moves the servo one degree at a time in the specified direction, until it hits a limit. During the servo movement, the code continues to call `sparkle()` so that the "fire" animation will not stop.

# Control the Circus with the Bluefruit LE Connect App

## Download the App

To control the servo and NeoPixels in CircusPython from a client computer (phone or tablet), you use the Adafruit [Bluefruit LE Connect App](https://adafru.it/iCi) (<https://adafru.it/iCi>). Install it from the [Apple App Store](https://adafru.it/BYj) (<https://adafru.it/BYj>) or [Google Play App Store](https://adafru.it/f4G) (<https://adafru.it/f4G>).

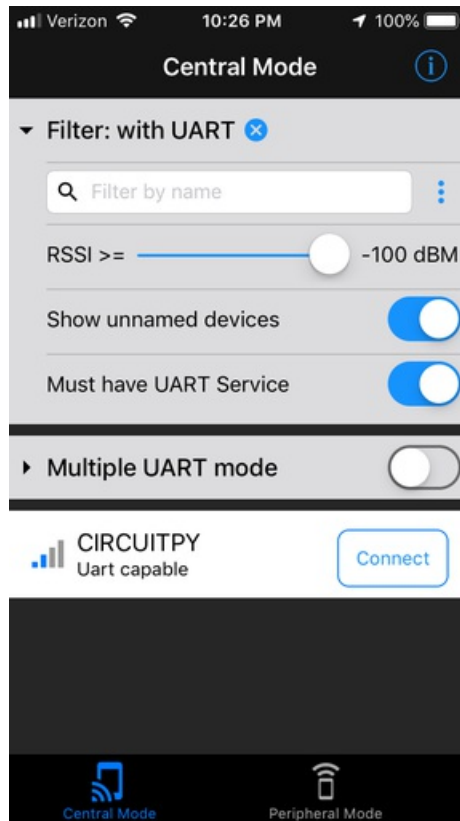
## Connect to CircusPython

Start the main program running on your CircusPython project. Then start up the Bluefruit LE Connect app, and make sure it's in Central Mode (left button on the bottom). When you start the app, you should see a device named **CIRCU** or **CIRCUITPY**. If there's a long list of devices, you can shorten it by turning on the "Must have UART Service" switch.

To connect to CircusPython, touch the **Connect** button. You should see "Connecting" and then "Discovering Services".

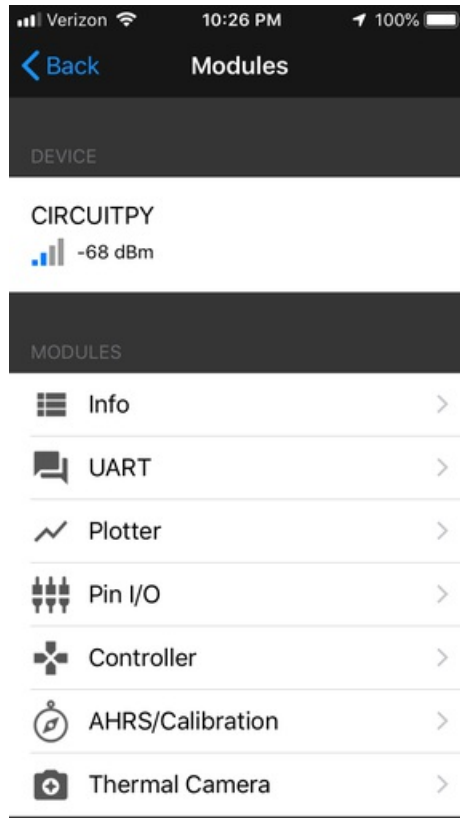
## Troubleshooting

If you don't see **CIRCU** or **CIRCUITPY** right away, try pulling down to refresh. If that doesn't work, try turning Bluetooth off and back on on your phone or tablet and restarting the app.



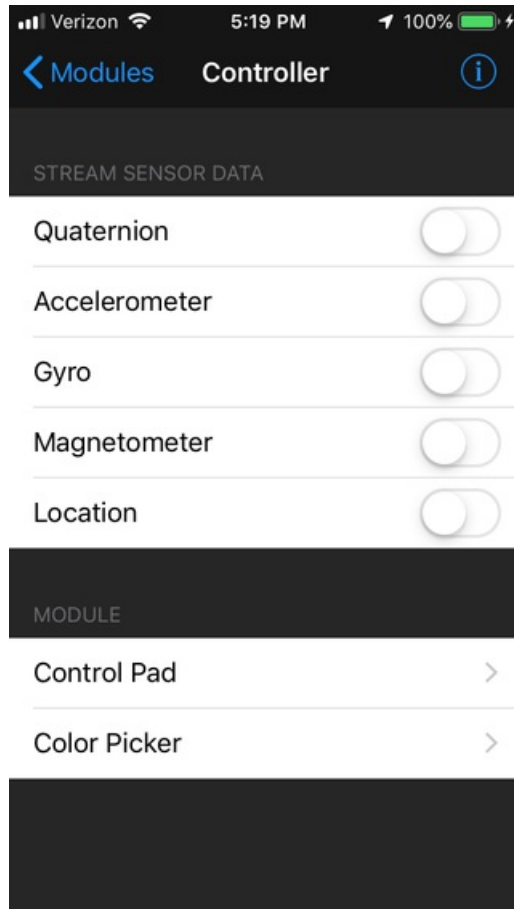
## Device Menu

After you connect, you'll see a menu of how you can interact with the device. Choose **Controller**.



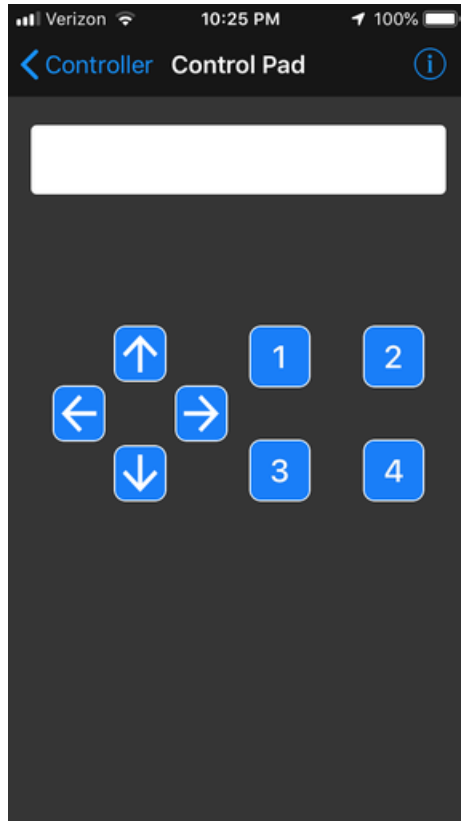
## Controller Menu

After you choose Controller, you'll see another screen with more choices. **Control Pad** and **Color Picker** are used for this project.



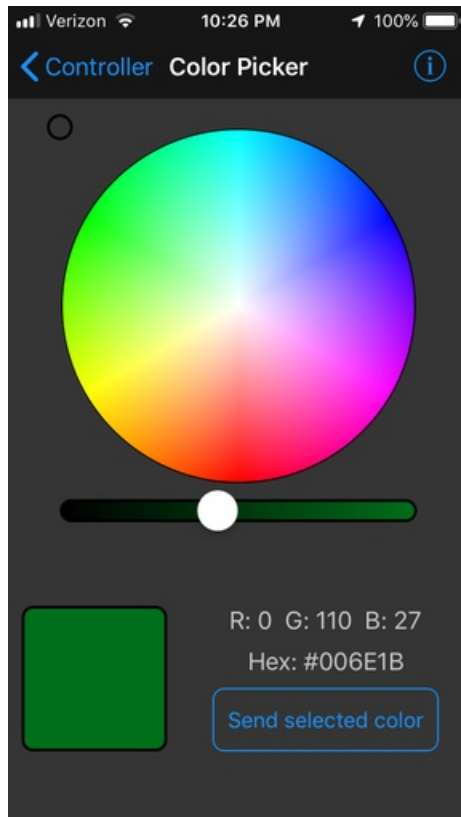
## Move the Servo Arm

Here's the **Control Pad** screen. Use the **Up** and **Down** buttons to move the servo arm up and down and make Blinka jump through the NeoPixel ring.



## Change Color

To change the color of the NeoPixels, use the **Color Picker** screen. Choose a color and press **Send Selected Color**.



Wow!

That's it! This simple project is only a jumping-off point. Add more servos or other things, and use the other buttons ([see the codes for the other buttons \(https://adafru.it/iCI\)](https://adafru.it/iCI)) to control them.