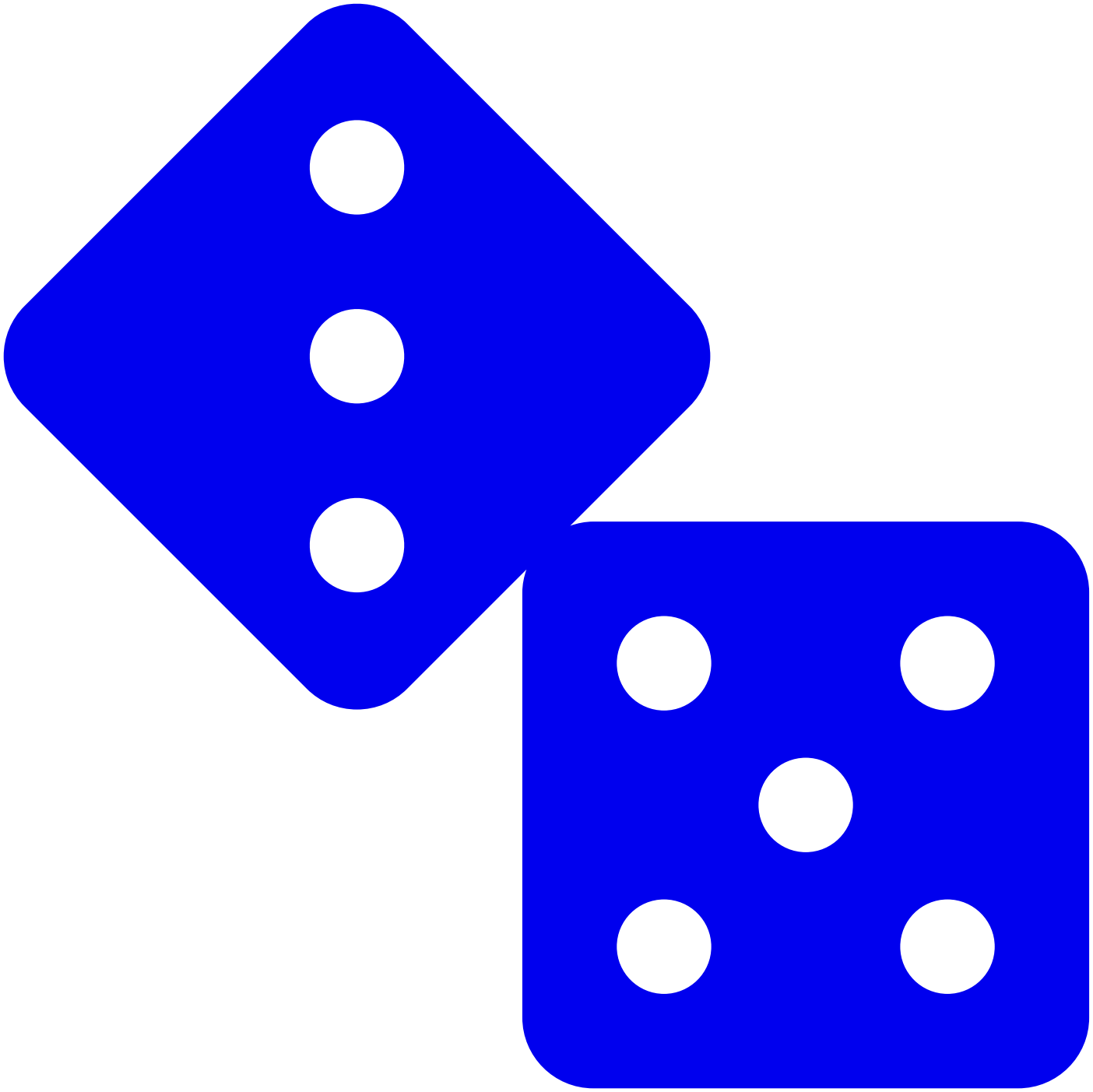
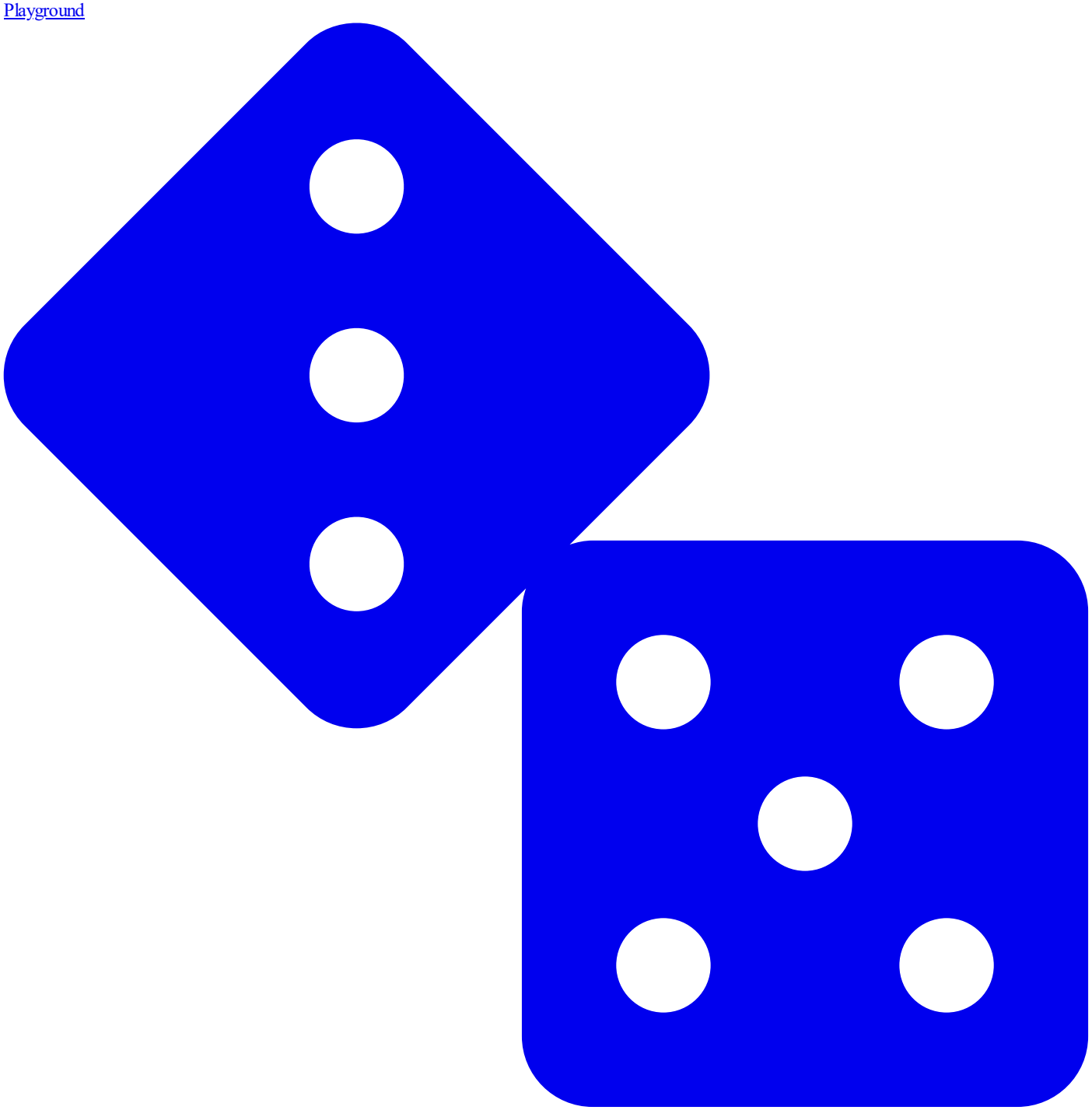
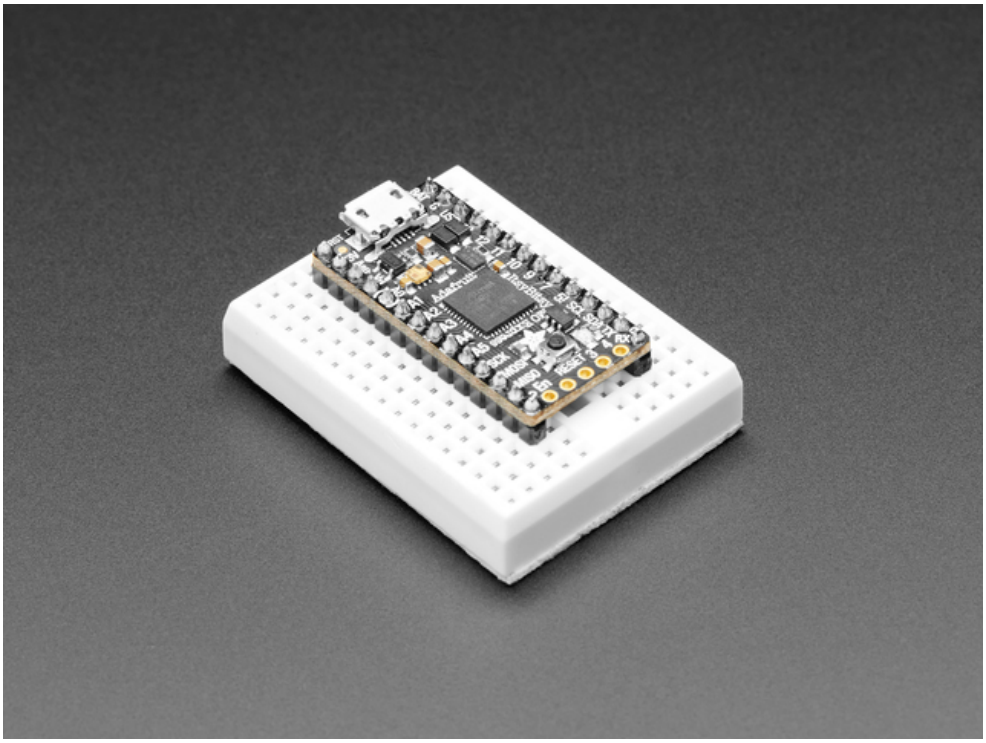


- Explore & Learn
- New Guides
- Playground





Introducing ItsyBitsy M0 Express

By lady ada
What's smaller than a Feather but larger than a Trinket? It's an ItsyBitsy!

CircuitPython Servo

In order to use servos, we take advantage of `pwmio`. Now, in theory, you could just use the raw `pwmio` calls to set the frequency to 50 Hz and then set the pulse widths. But we would rather make it a little more elegant and easy!

So, instead we will use `adafruit_motor` which manages servos for you quite nicely! `adafruit_motor` is a library so be sure to grab it from the library bundle if you have not yet! If you need help installing the library, check out the CircuitPython Libraries page.

Servos come in two types:

- A **standard hobby servo** - the horn moves 180 degrees (90 degrees in each direction from zero degrees).
- A **continuous servo** - the horn moves in full rotation like a DC motor. Instead of an angle specified, you set a throttle value with 1.0 being full forward, 0.5 being half forward, 0 being stopped, and -1 being full reverse, with other values between.

Servo Wiring

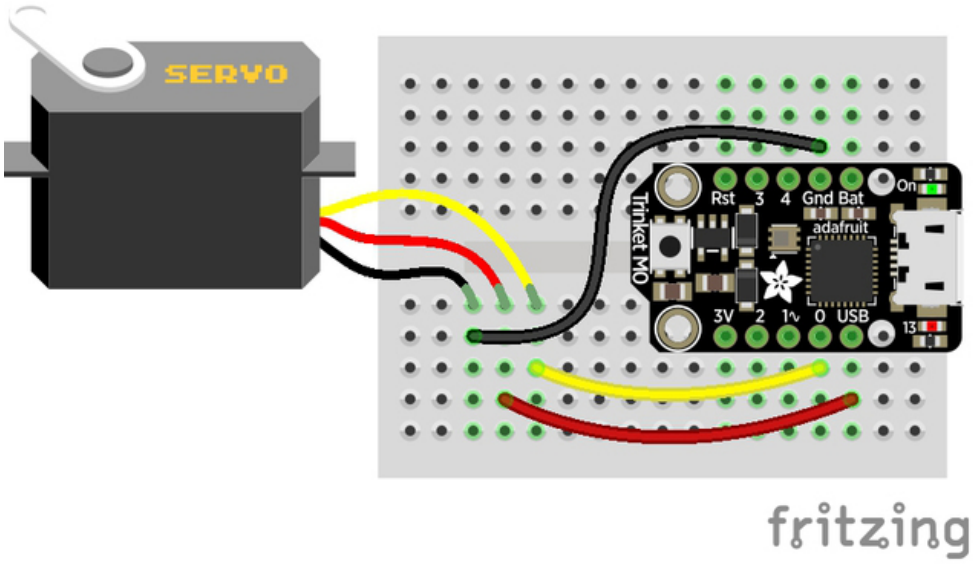
Servos will only work on PWM-capable pins! Check your board details to verify which pins have PWM outputs.

The connections for a servo are the same for standard servos and continuous rotation servos.

Connect the servo's **brown** or **black** ground wire to ground on the CircuitPython board.

Connect the servo's **red** power wire to 5V power, USB power is good for a servo or two. For more than that, you'll need an external battery pack. Do not use 3.3V for powering a servo!

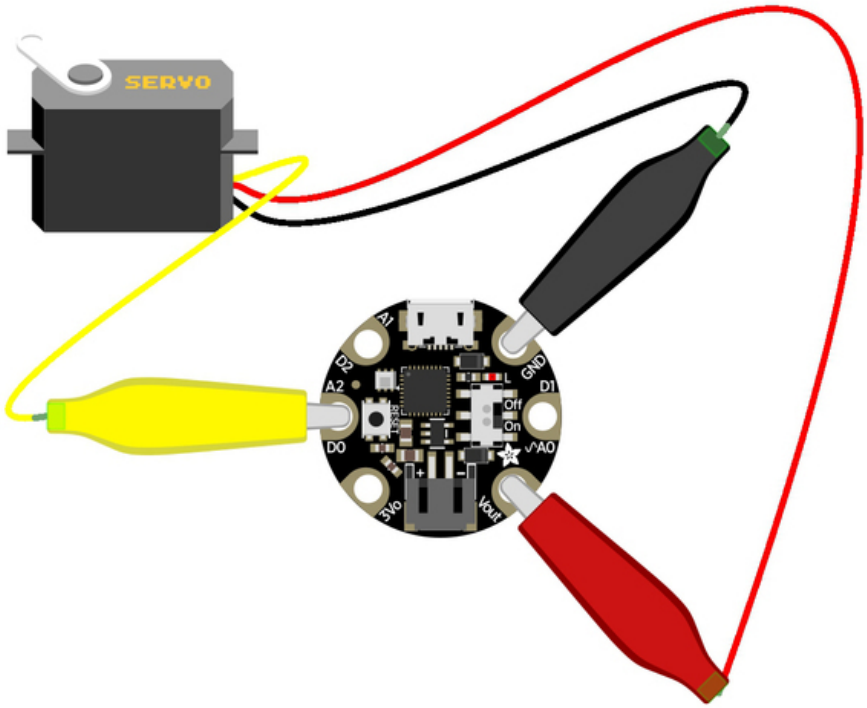
Connect the servo's **yellow** or **white** signal wire to the control/data pin, in this case **A1 or A2** but you can use any PWM-capable pin.



-

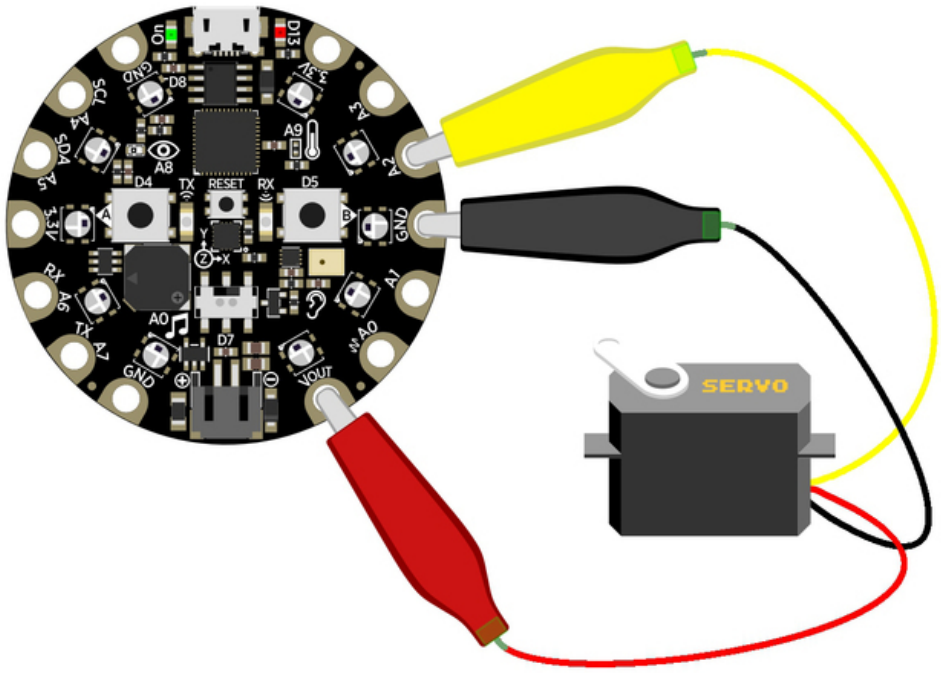
For example, to wire a servo to **Trinket**, connect the ground wire to **GND**, the power wire to **USB**, and the signal wire to **0**.

Remember, **A2 on Trinket is labeled "0"**.



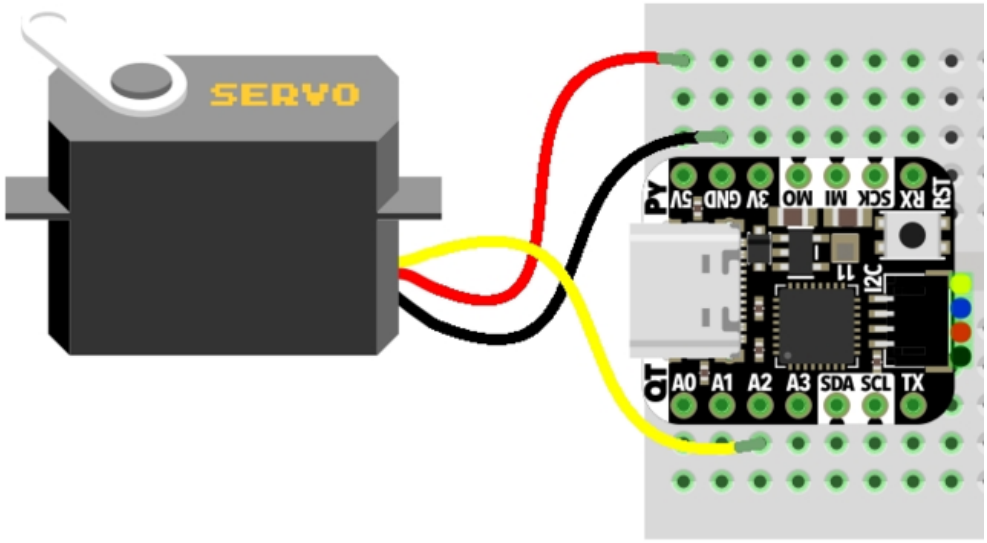
-

For **Gemma**, use jumper wire alligator clips to connect the ground wire to **GND**, the power wire to **VOUT**, and the signal wire to **A2**.



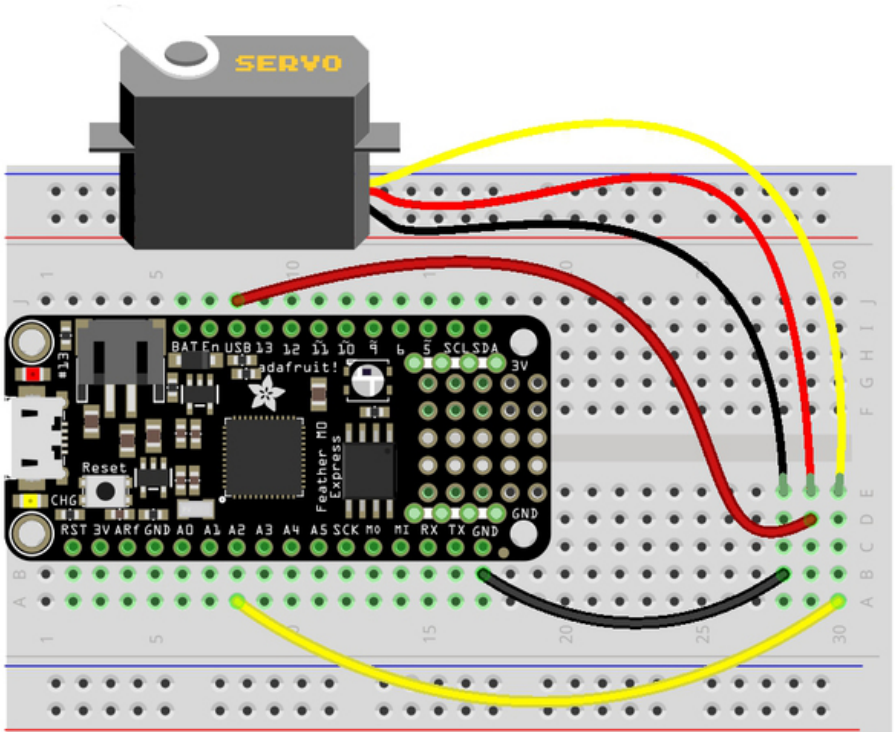
-

For **Circuit Playground Express and Circuit Playground Bluefruit**, use jumper wire alligator clips to connect the ground wire to **GND**, the power wire to **VOUT**, and the signal wire to **A2**.

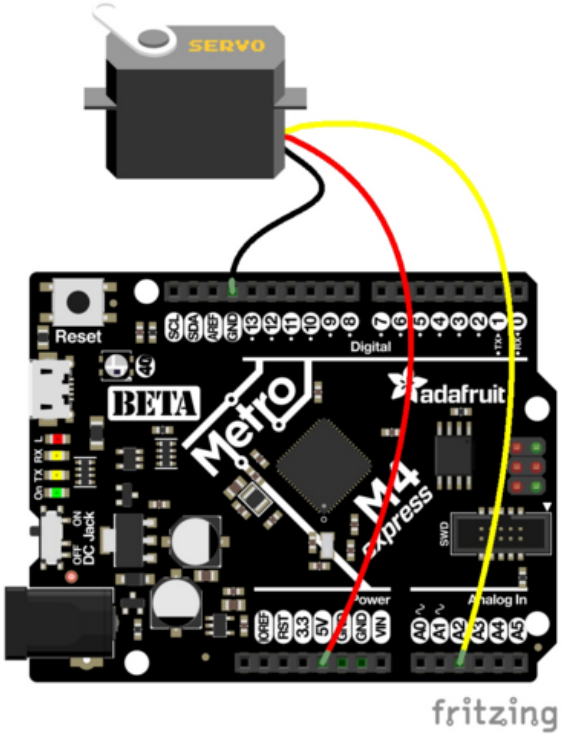


-

For **QT Py M0**, connect the ground wire to **GND**, the power wire to **5V**, and the signal wire to **A2**.



For boards like **Feather M0 Express**, **ItsyBitsy M0 Express** and **Metro M0 Express**, connect the ground wire to any **GND**, the power wire to **USB or 5V**, and the signal wire to **A2**.



For the **Metro M4 Express**, **ItsyBitsy M4 Express** and the **Feather M4 Express**, connect the ground wire to any **G or GND**, the power wire to **USB or 5V**, and the signal wire to **A2**.

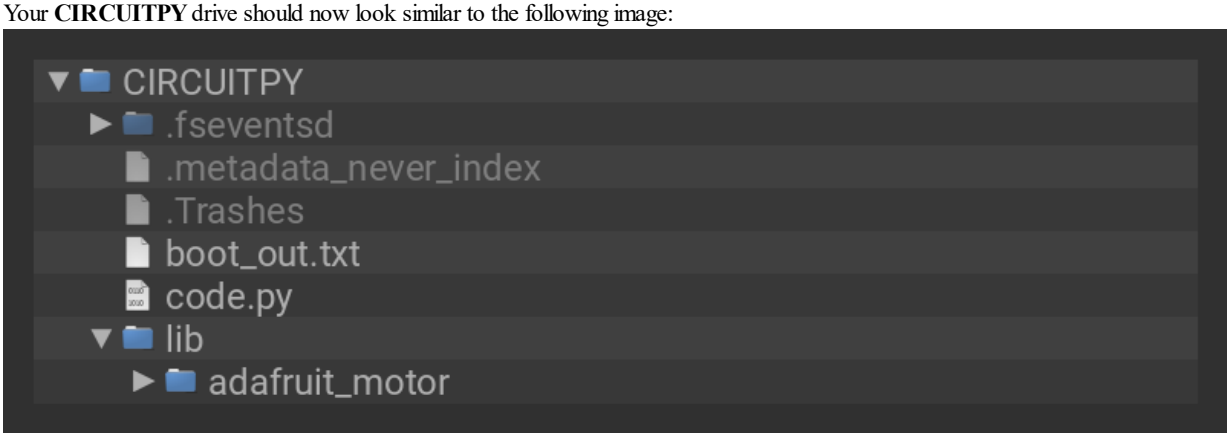
Standard Servo Code

Here's an example that will sweep a servo connected to pin **A2** from 0 degrees to 180 degrees (-90 to 90 degrees) and back.

To use with CircuitPython, you need to first install a few libraries, into the lib folder on your **CIRCUITPY** drive. Then you need to update **code.py** with the example script.

Thankfully, we can do this in one go. In the example below, click the **Download Project Bundle** button below to download the necessary libraries and the **code.py** file in a zip file. Extract the contents of the zip file, open the directory **CircuitPython_Essentials/CircuitPython_Servo/** and then click on the directory that matches the version of CircuitPython you're using and copy the contents of that directory to your **CIRCUITPY** drive.

Your **CIRCUITPY** drive should now look similar to the following image:



Download Project Bundle
Copied!

```
# SPDX-FileCopyrightText: 2018 Kattni Rembor for Adafruit Industries
#
# SPDX-License-Identifier: MIT

"""CircuitPython Essentials Servo standard servo example"""
import time
import board
import pwmio
from adafruit_motor import servo

# create a PWMOut object on Pin A2.
pwm = pwmio.PWMOut(board.A2, duty_cycle=2 ** 15, frequency=50)

# Create a servo object, my_servo.
my_servo = servo.Servo(pwm)

while True:
    for angle in range(0, 180, 5):  # 0 - 180 degrees, 5 degrees at a time.
        my_servo.angle = angle
        time.sleep(0.05)
    for angle in range(180, 0, -5): # 180 - 0 degrees, 5 degrees at a time.
        my_servo.angle = angle
        time.sleep(0.05)
```

View on GitHub

Continuous Servo Code

There are two differences with Continuous Servos vs. Standard Servos:

1. The `servo` object is created like `my_servo = servo.ContinuousServo(pwm)` instead of `my_servo = servo.Servo(pwm)`
2. Instead of using `myservo.angle`, you use `my_servo.throttle` using a throttle value from 1.0 (full on) to 0.0 (stopped) to -1.0 (full reverse). Any number between would be a partial speed forward (positive) or reverse (negative). This is very similar to standard DC motor control with the **adafruit_motor** library.

This example runs full forward for 2 seconds, stops for 2 seconds, runs full reverse for 2 seconds, then stops for 4 seconds.

To use with CircuitPython, you need to first install a few libraries, into the lib folder on your **CIRCUITPY** drive. Then you need to update **code.py** with the example script.

Thankfully, we can do this in one go. In the example below, click the **Download Project Bundle** button below to download the necessary libraries and the **code.py** file in a zip file. Extract the contents of the zip file, open the directory **CircuitPython_Essentials/CircuitPython_Continuous_Servo/** and then click on the directory that matches the version of CircuitPython you're using and copy the contents of that directory to your **CIRCUITPY** drive.

Your **CIRCUITPY** drive should now look similar to the following image:

Download Project Bundle
Copy Code

```
# SPDX-FileCopyrightText: 2019 Anne Barela for Adafruit Industries
#
# SPDX-License-Identifier: MIT

"""CircuitPython Essentials Servo continuous rotation servo example"""
import time
import board
import pwmio
from adafruit_motor import servo

# create a PWMOut object on Pin A2.
pwm = pwmio.PWMOut(board.A2, frequency=50)

# Create a servo object, my_servo.
my_servo = servo.ContinuousServo(pwm)

while True:
    print("forward")
    my_servo.throttle = 1.0
    time.sleep(2.0)
    print("stop")
    my_servo.throttle = 0.0
    time.sleep(2.0)
    print("reverse")
    my_servo.throttle = -1.0
    time.sleep(2.0)
    print("stop")
    my_servo.throttle = 0.0
    time.sleep(4.0)
```

View on GitHub

Pretty simple!

Note that we assume that 0 degrees is 0.5ms and 180 degrees is a pulse width of 2.5ms. That's a bit wider than the *official* 1-2ms pulse widths. If you have a servo that has a different range you can initialize the `servo` object with a different `min_pulse` and `max_pulse`. For example:

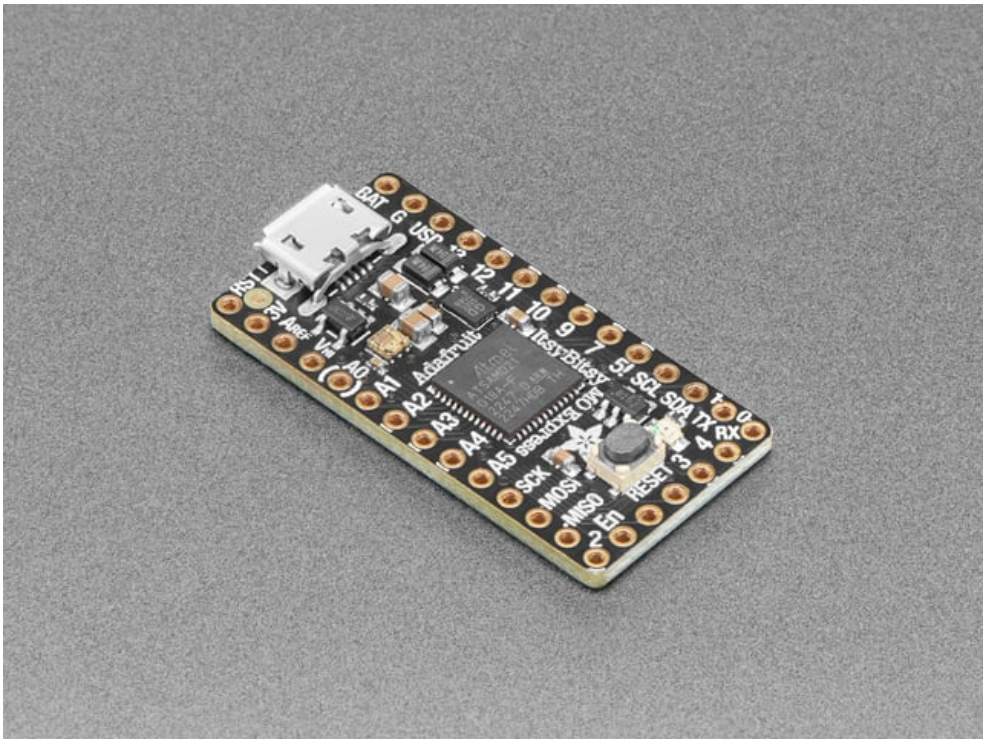
```
my_servo = servo.Servo(pwm, min_pulse = 500, max_pulse = 2500)
```

For more detailed information on using servos with CircuitPython, check out the CircuitPython section of the servo guide!
CircuitPython PWM CircuitPython Cap Touch

This guide was first published on May 27, 2018. It was last updated on May 27, 2018.

This page (CircuitPython Servo) was last updated on Aug 22, 2023.

Text editor powered by tinymce.
Difficulty: Beginner
Guide Type: Project
Contributors: lady ada, Kattni Rembor
Categories: Adafruit Products
Arduino Compatibles
CircuitPython
29 Saves
Featured Products

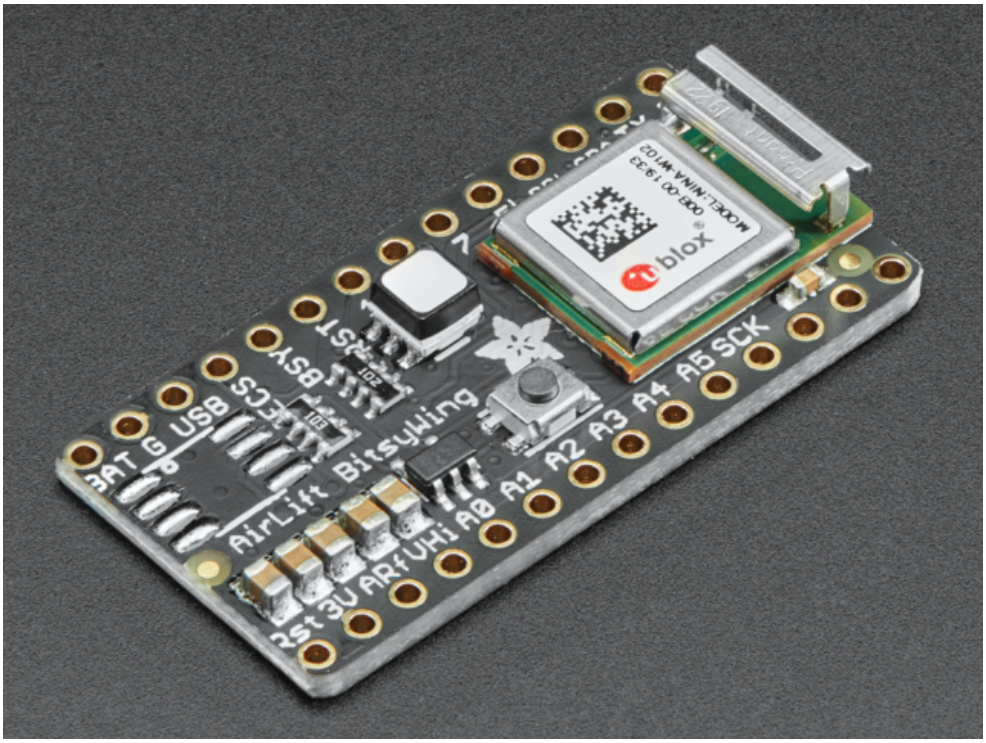


Adafruit ItsyBitsy M0 Express - for CircuitPython & Arduino IDE
$11.95
Add to Cart

Blinka the CircuitPython Temporary Tattoo
$0.95
Add to Cart

Blinka the CircuitPython Limited Edition Enamel Pin
Out of Stock

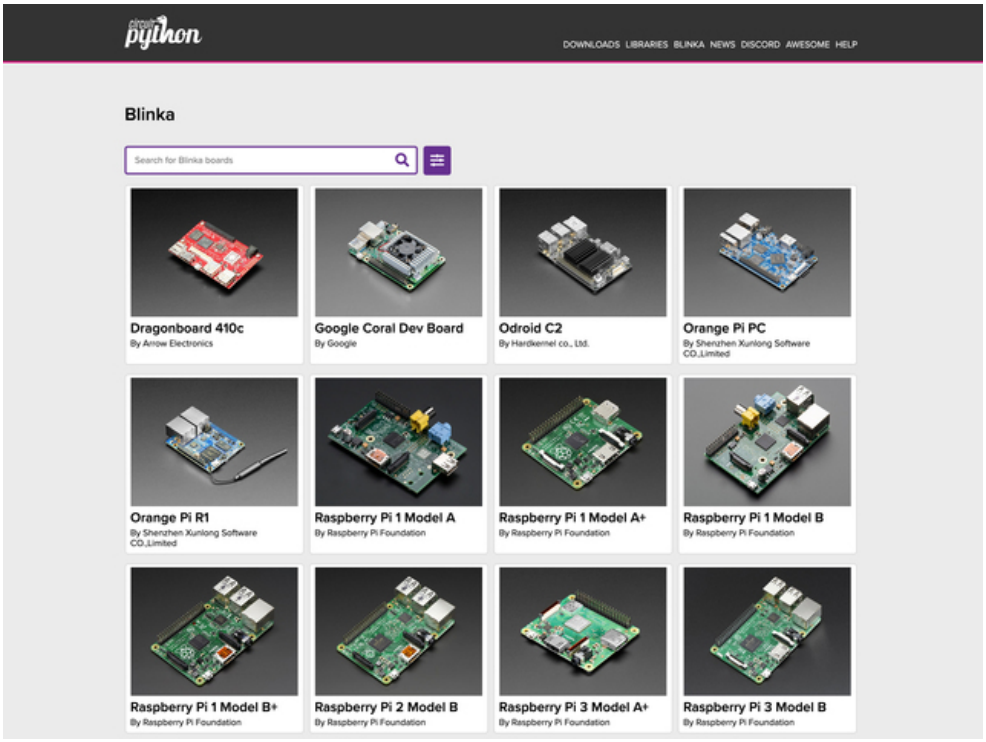
Blinka the CircuitPython Sticker
$1.25
Add to Cart
Related Guides



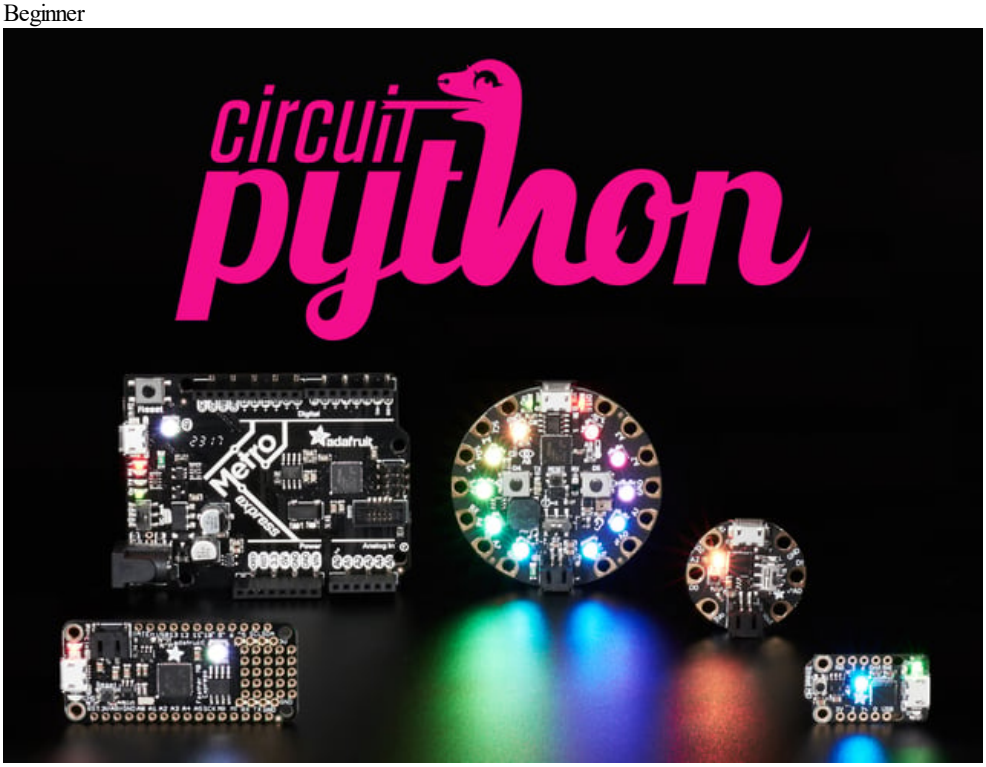
[Adafruit Airlift Bitsy Add-On - ESP32 WiFi Co-Processor]
By [Bryan Siepert]
13
Beginner
Updated

```
const mp_obj_property_t mymodule_myclass_question_obj = {
    .base.type = &mp_type_property,
    .proxy = {(mp_obj_t)&mymodule_myclass_get_question_obj,
              (mp_obj_t)&mp_const_none_obj},
};

const mp_obj_property_t mymodule_myclass_answer_obj = {
    .base.type = &mp_type_property,
    .proxy = {(mp_obj_t)&mymodule_myclass_get_answer_obj,
              (mp_obj_t)&mp_const_none_obj},
};

STATIC const mp_rom_map_elem_t mymodule_myclass_locals_dict_table[] = {
    // Methods
    { MP_ROM_QSTR(MP_QSTR_deinit), MP_ROM_PTR(&mymodule_myclass_deinit_ob
    { MP_ROM_QSTR(MP_QSTR___enter__), MP_ROM_PTR(&default___enter___obj)
    { MP_ROM_QSTR(MP_QSTR___exit__), MP_ROM_PTR(&mymodule_myclass___exit_
    { MP_ROM_QSTR(MP_QSTR_question), MP_ROM_PTR(&mymodule_myclass_questio
    { MP_ROM_QSTR(MP_QSTR_answer), MP_ROM_PTR(&mymodule_myclass_answer_ob
};
STATIC MP_DEFINE_CONST_DICT(mymodule_myclass_locals_dict, mymodule_myclas

const mp_obj_type_t mymodule_myclass_type = {
    { &mp_type_type },
    .name = MP_QSTR_Meaning,
    .make_new = mymodule_myclass_make_new,
    .locals_dict = (mp_obj_dict_t*)&mymodule_myclass_locals_dict,
};
```

[Extending CircuitPython: An Introduction]
By [Dave Astels]
17
Advanced



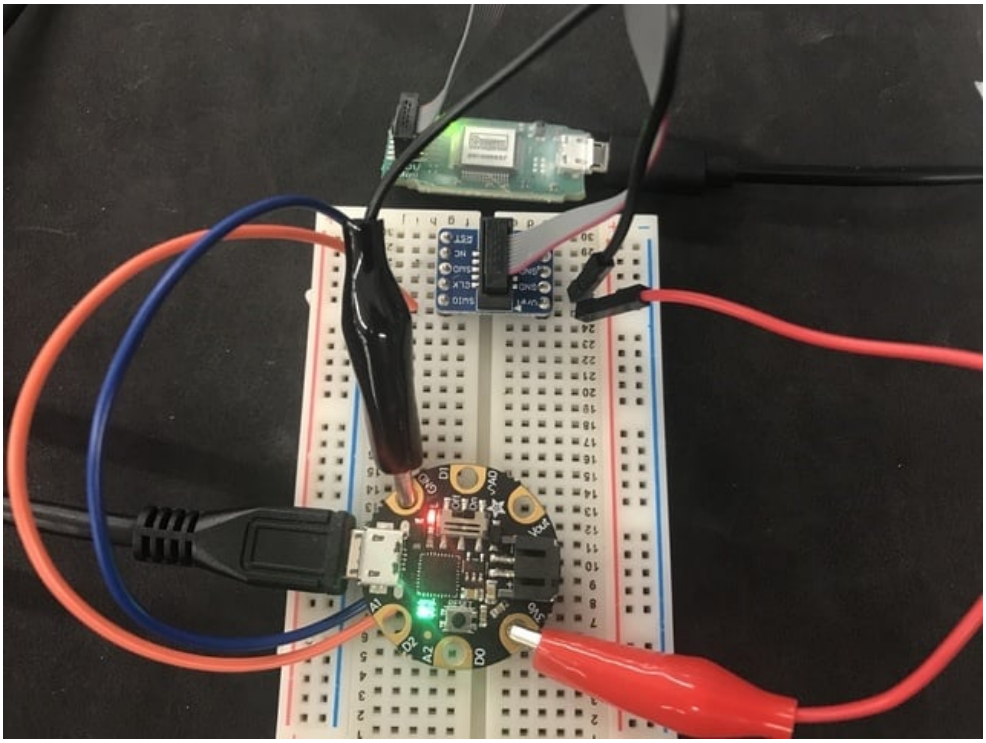
How to add a New Board to the circuitpython.org website
By M. LeBlanc-Williams
3
Beginner



How to Add a New Board to CircuitPython
By Kattni Rembor
14
Intermediate
Updated

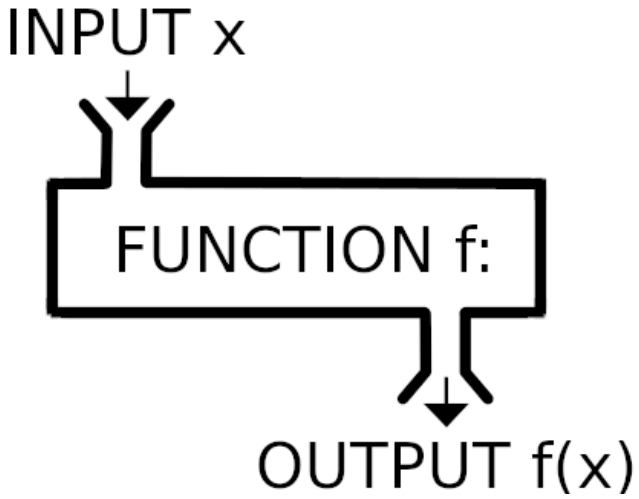
NeoPixel LED Necklace Insert with USB Charging
By Erin St Blaine
48
Intermediate

ItsyBitsy Keybow Mechanical Keypad
By John Park
16
Intermediate

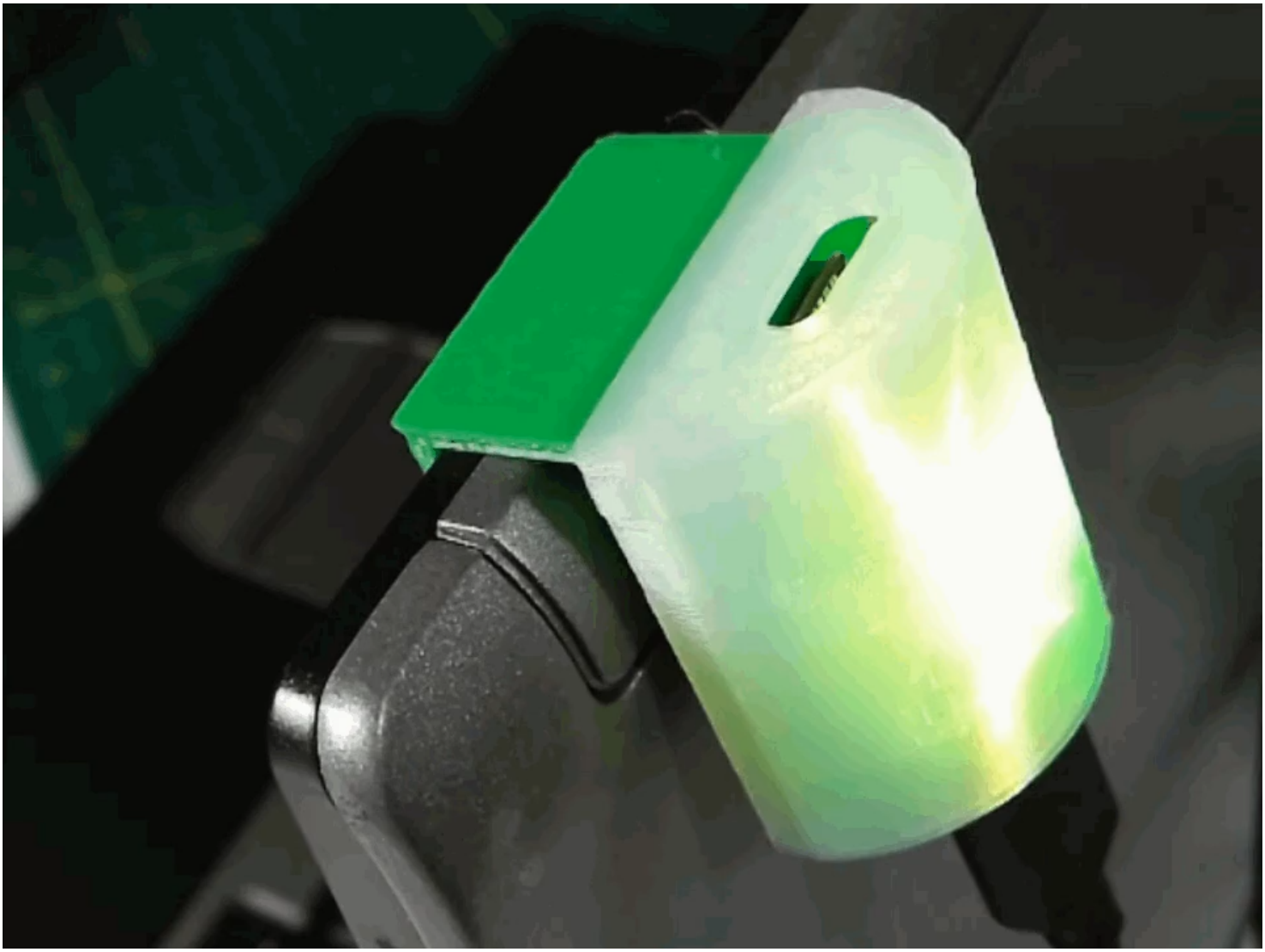


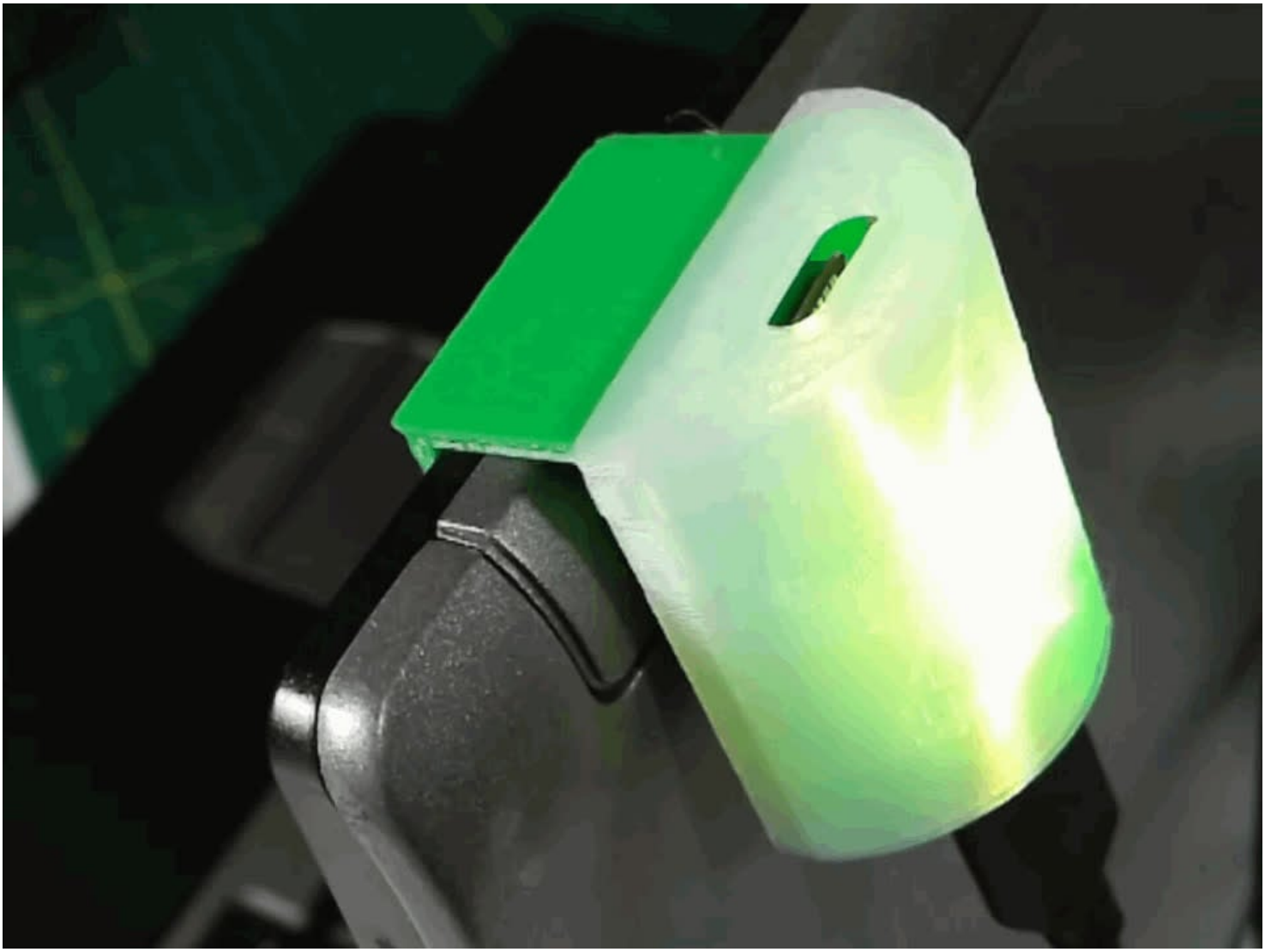
How to Program SAMD Bootloaders
By Brent Rubell
34
Intermediate

USB HID Crank Controller
By Ruiz Brothers
12
Beginner



CircuitPython 101: Functions
By Dave Astels
48
Intermediate



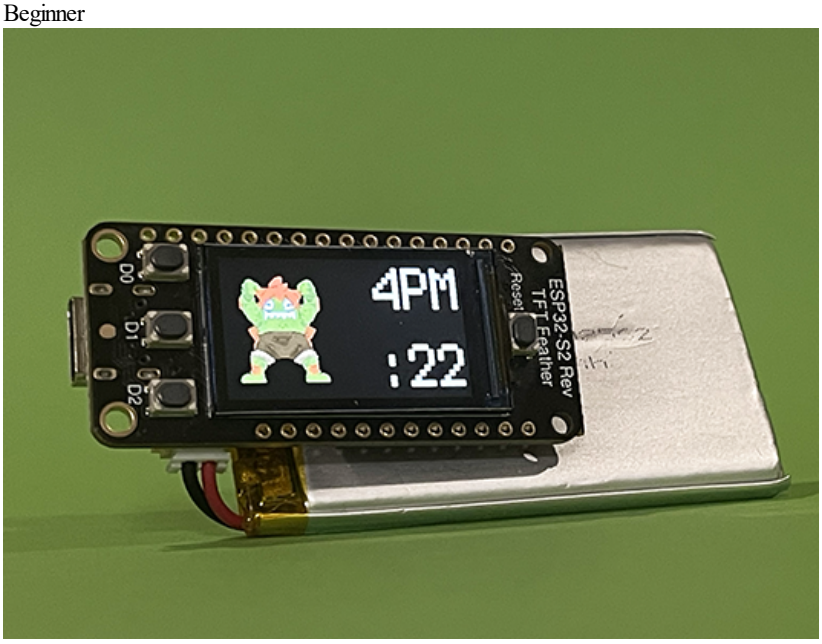


CircuitPython Powered AT Hand-Raiser
By Bill Binko
17
Beginner

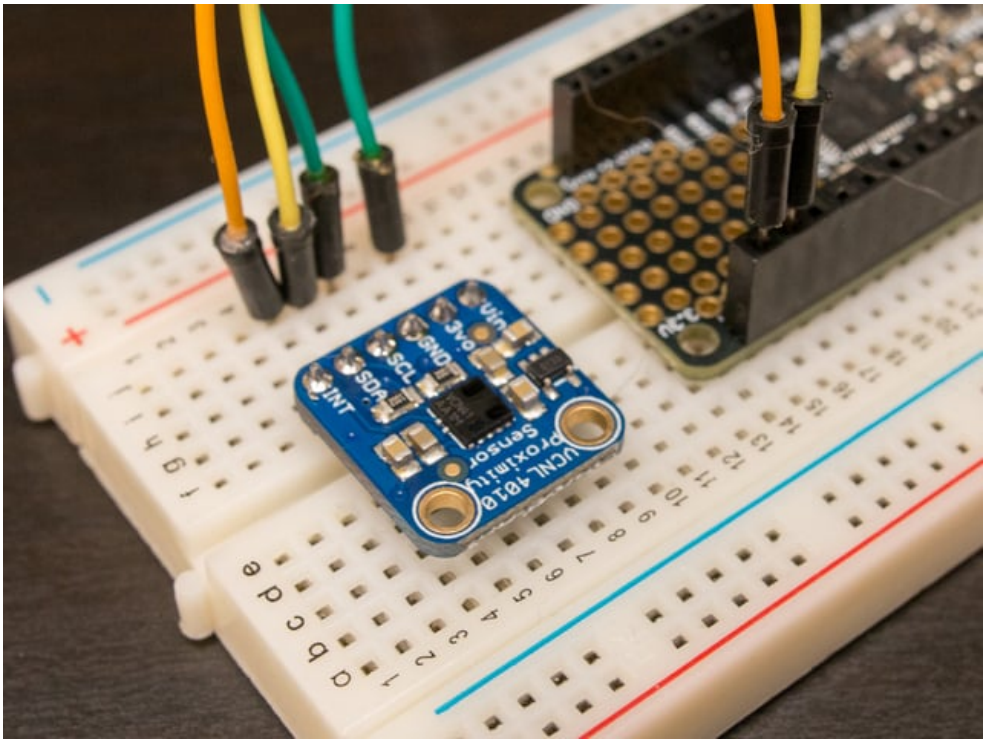
Sensor Plotting with Mu and CircuitPython
By Kattni Rembor
73
Beginner

Contribute to CircuitPython with Git and GitHub
By Kattni Rembor
37
Intermediate
Updated

Rotary Encoder in CircuitPython
By Kattni Rembor
30
Beginner



ESP32-S3 Reverse TFT Digital Clock Display featuring...
By Trevor Beaton
6
Beginner
New



Using VCNL4010 Proximity Sensor
By Tony DiCola
10
Beginner
×

OUT OF STOCK NOTIFICATION

YOUR NAME
YOUR EMAIL
NOTIFY ME
Search

Search

Categories

No results for query

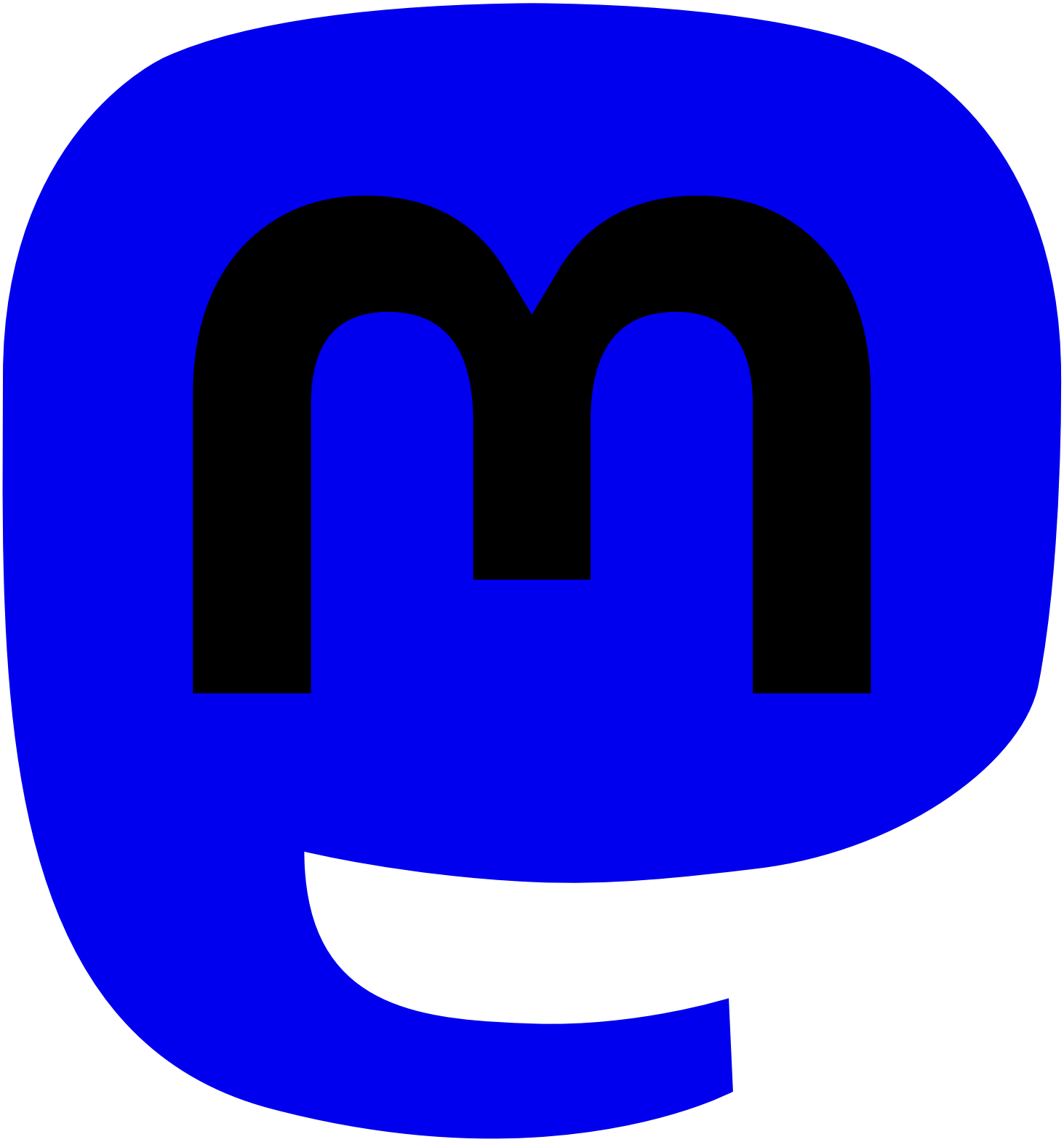
- «
- ‹
- 1
- ›
- »

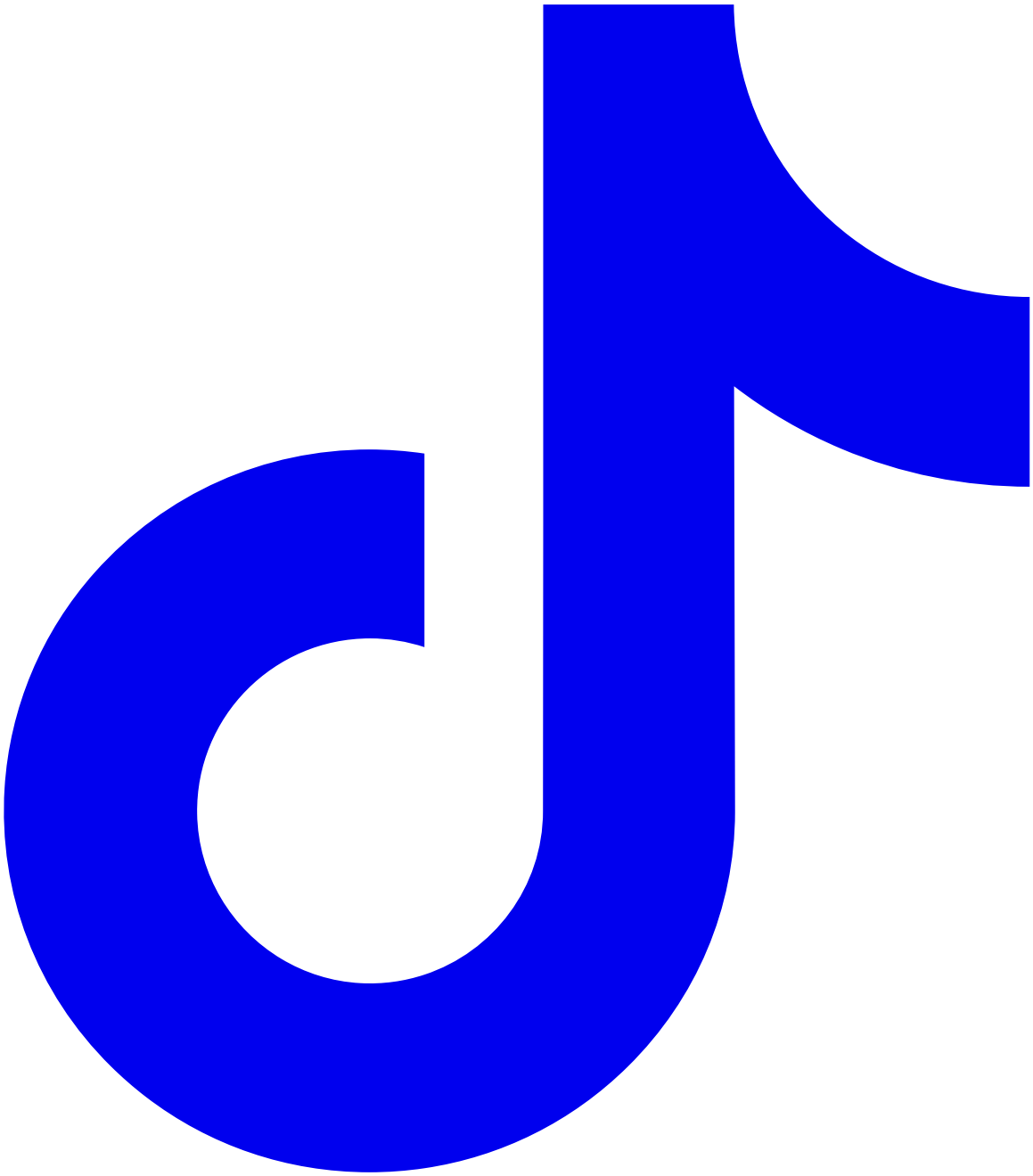
- Contact Us
- Tech Support Forums
- FAQs
- Shipping & Returns
- Freebies
- Terms of Service
- Privacy & Legal
- Website Accessibility

- About Us
- Press
- Educators
- Distributors
- Jobs

- Gift Cards

"Get as much education as you can. Nobody can take that away from you"
Eben Upton





A Minority and Woman-owned Business Enterprise (M/WBE)