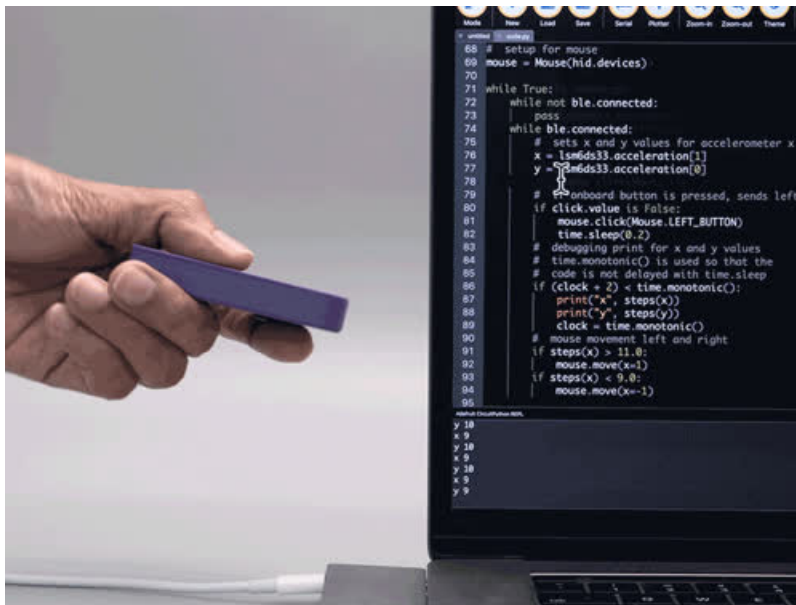


Programmable Wireless BLE Gesture Mouse

Created by Liz Clark



Last updated on 2021-10-01 08:52:51 AM EDT

Guide Contents

| | |
|---|----|
| Guide Contents | 2 |
| Overview | 3 |
| Parts | 4 |
| CircuitPython on Feather Sense | 6 |
| Set up CircuitPython Quick Start! | 6 |
| Code the BLE Gesture Mouse | 8 |
| Upload the Code and Libraries to the Feather nRF52840 Sense | 10 |
| How the CircuitPython Code Works | 10 |
| 3D Printing | 13 |
| CAD Parts List | 13 |
| CAD Assembly | 13 |
| Slicing Parts | 13 |
| Design Source Files | 14 |
| Wiring Switch | 15 |
| Wire for Switch | 15 |
| Tinning Wire | 15 |
| Solder Wire | 15 |
| Wired Switch | 16 |
| Solder Switch to Feather | 16 |
| Feather Switch | 16 |
| Assembly | 17 |
| Screws for Case | 17 |
| Install Hardware | 17 |
| Secure Feather | 17 |
| Install Switch | 18 |
| Switch Actuator | 18 |
| Connect Battery | 18 |
| Install Battery | 19 |
| Install Cover | 19 |
| Snap Fit Cover | 19 |
| Case USB port | 20 |
| BLE Pairing | 21 |
| BLE Pairing – MacOS | 21 |

Overview

Most computers require a mouse to navigate the user interface. In this project, you'll learn how you can build a mouse that connects via Bluetooth BLE and converts accelerometer readings to mouse cursor movements.



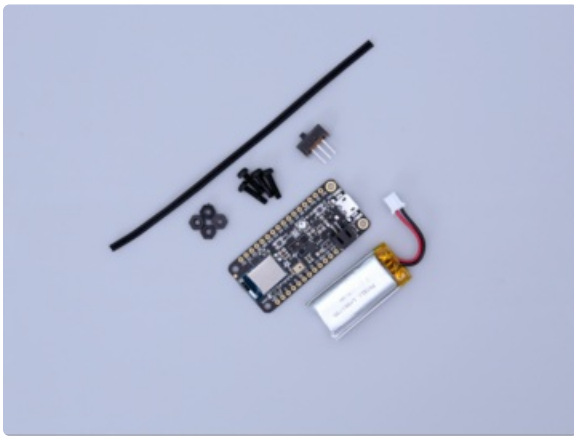
The project is built around the Adafruit Feather nRF52840 Sense. It has BLE connectivity and a lot of onboard sensors. Its accelerometer is used for sending movement data that is converted to move your mouse cursor around the screen.



Right-clicks are sent with the onboard button. Scrolling is done by covering the proximity sensor and tilting the Feather nRF52840 Sense up or down depending on the direction that you want to scroll.

The 3D printed case features cut-outs on the top so that you can access the button and proximity sensor while keeping your board protected.

Parts



Parts from Adafruit used to build this project.

- [Feather nRF52840 Sense \(https://adafru.it/JA1\)](https://adafru.it/JA1)
- [Slide Switch \(https://adafru.it/drN\)](https://adafru.it/drN)
- [400mAh Battery \(https://adafru.it/D7i\)](https://adafru.it/D7i)
- [M2.5 Hardware \(https://adafru.it/wsc\)](https://adafru.it/wsc)
- [Silicon Ribbon Wire \(https://adafru.it/CJj\)](https://adafru.it/CJj)

Adafruit Feather nRF52840 Sense

The Adafruit Feather Bluefruit Sense takes our popular Feather nRF52840 Express and adds a smorgasbord of sensors...

\$32.50

In Stock

Add to Cart

Breadboard-friendly SPDT Slide Switch

These nice switches are perfect for use with breadboard and perfboard projects. They have 0.1" spacing and snap in nicely into a solderless breadboard. They're easy to switch...

\$0.95

In Stock

Add to Cart

Lithium Ion Polymer Battery Ideal For Feathers - 3.7V 400mAh

Lithium ion polymer (also known as 'lipo' or 'lipoly') batteries are thin, light and powerful. The output ranges from 4.2V when completely charged to 3.7V. This battery...

\$6.95

In Stock

Add to Cart

Black Nylon Machine Screw and Stand-off Set – M2.5 Thread

Totalling 380 pieces, this M2.5 Screw Set is a must-have for your workstation. You'll have enough screws, nuts, and hex standoffs to fuel your maker...

\$16.95

In Stock

Add to Cart

Silicone Cover Stranded-Core Ribbon Cable - 10 Wire 1 Meter Long

For those who are fans of our silicone-covered wires, but are always looking to up their wiring game. We now have Silicone Cover Ribbon cables! These may look...

\$3.95

In Stock

Add to Cart

CircuitPython on Feather Sense

[CircuitPython](https://adafru.it/tB7) (<https://adafru.it/tB7>) is a derivative of [MicroPython](https://adafru.it/BeZ) (<https://adafru.it/BeZ>) designed to simplify experimentation and education on low-cost microcontrollers. It makes it easier than ever to get prototyping by requiring no upfront desktop software downloads. Simply copy and edit files on the **CIRCUITPY** drive to iterate.

The following instructions will show you how to install CircuitPython. If you've already installed CircuitPython but are looking to update it or reinstall it, the same steps work for that as well!

Set up CircuitPython Quick Start!

Follow this quick step-by-step for super-fast Python power :)

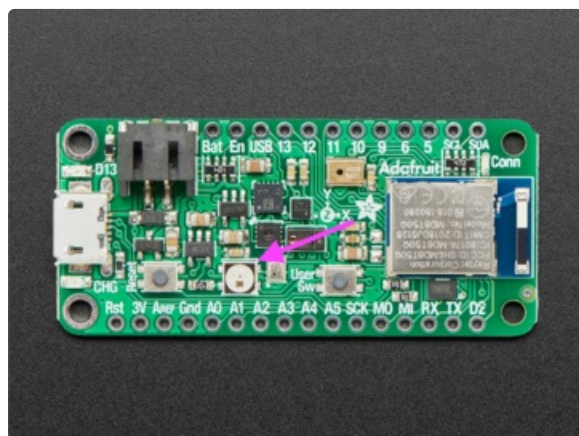
<https://adafru.it/JqE>

<https://adafru.it/JqE>



Click the link above to download the latest UF2 file.

Download and save it to your desktop (or wherever is handy).

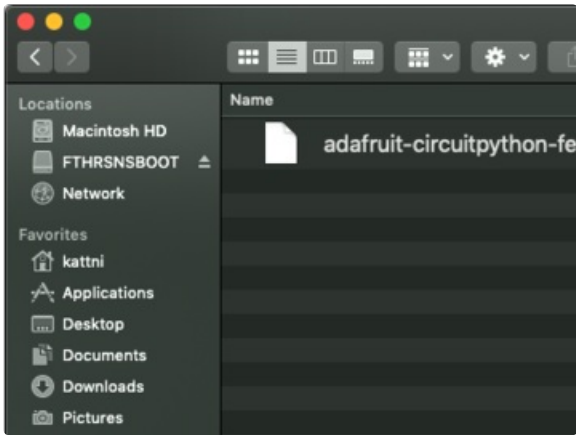


Plug your Feather Sense into your computer using a known-good USB cable.

A lot of people end up using charge-only USB cables and it is very frustrating! So make sure you have a USB cable you know is good for data sync.

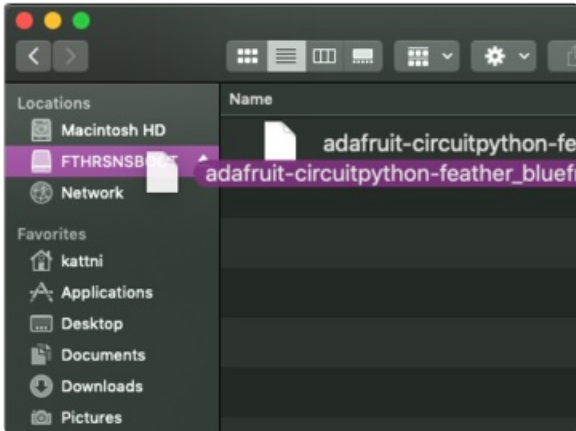
Double-click the **Reset** button next to the USB connector on your board, and you will see the NeoPixel RGB LED turn green (identified by the arrow in the image). If it turns red, check the USB cable, try another USB port, etc. **Note:** The little red LED next to the USB connector will pulse red. That's ok!

If double-clicking doesn't work the first time, try again. Sometimes it can take a few tries to get the rhythm right!



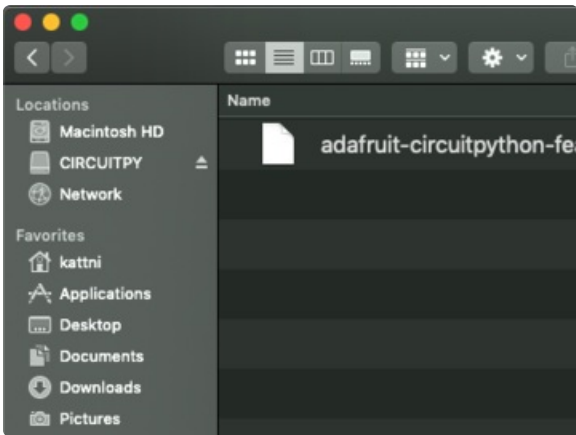
You will see a new disk drive appear called **FTHRNSBOOT**.

Drag the `adafruit_circuitpython_etc.uf2` file to **FTHRNSBOOT**.



The LED will flash. Then, the **FTHRNSBOOT** drive will disappear and a new disk drive called **CIRCUITPY** will appear.

That's it, you're done! :)



Note: Some early release Sense boards had the drive named FTHR840BOOT. You can still copy .UF2s to the board, just copy to the board name appearing when the board is plugged in.

Code the BLE Gesture Mouse

Once you've finished setting up your Feather nRF52840 Sense with CircuitPython, you can access the code and necessary libraries by downloading the Project Bundle.

To do this, click on the **Download Project Bundle** button in the window below. It will download as a zipped folder.

```
import time
import board
import digitalio
import simpleio
import adafruit_lsm6ds.lsm6ds33
import adafruit_apds9960.apds9960
from adafruit_hid.mouse import Mouse

import adafruit_ble
from adafruit_ble.advertising import Advertisement
from adafruit_ble.advertising.standard import ProvideServicesAdvertisement
from adafruit_ble.services.standard.hid import HIDService
from adafruit_ble.services.standard.device_info import DeviceInfoService

# setup I2C
i2c = board.I2C()

# setup accelerometer
lsm6ds33 = adafruit_lsm6ds.lsm6ds33.LSM6DS33(i2c)
# setup proximity sensor
apds9960 = adafruit_apds9960.apds9960.APDS9960(i2c)

# enable proximity sensor
apds9960.enable_proximity = True

# setup for onboard button
click = digitalio.DigitalInOut(board.SWITCH)
click.direction = digitalio.Direction.INPUT
click.pull = digitalio.Pull.UP

# rounding algorithm used for mouse movement
# as used in the HID mouse CircuitPython example
mouse_min = -9
mouse_max = 9
step = (mouse_max - mouse_min) / 20.0

def steps(axis):
    return round((axis - mouse_min) / step)

# time.monotonic() variable
clock = 0

# variable for distance for proximity scrolling
distance = 245
```



```

# setup for HID and BLE
hid = HIDService()

device_info = DeviceInfoService(software_revision=adafruit_ble.__version__,
                                manufacturer="Adafruit Industries")
advertisement = ProvideServicesAdvertisement(hid)
advertisement.appearance = 961
scan_response = Advertisement()
scan_response.complete_name = "CircuitPython HID"

ble = adafruit_ble.BLERadio()

if not ble.connected:
    print("advertising")
    ble.start_advertising(advertisement, scan_response)
else:
    print("already connected")
    print(ble.connections)

# setup for mouse
mouse = Mouse(hid.devices)

while True:
    while not ble.connected:
        pass
    while ble.connected:
        # sets x and y values for accelerometer x and y values
        # x and y are swapped for orientation of feather
        y, x, z = lsm6ds33.acceleration

        # map range of horizontal movement to mouse x movement
        horizontal_mov = simpleio.map_range(steps(x), 1.0, 20.0, -15.0, 15.0)
        # map range of vertical movement to mouse y movement
        vertical_mov = simpleio.map_range(steps(y), 20.0, 1.0, -15.0, 15.0)
        # map range of mouse y movement to scrolling
        scroll_dir = simpleio.map_range(vertical_mov, -15.0, 15.0, 3.0, -3.0)

        # if onboard button is pressed, sends left mouse click
        if not click.value:
            mouse.click(Mouse.LEFT_BUTTON)
            time.sleep(0.2)
        # if the proximity sensor is covered
        # scroll the mouse
        if apds9960.proximity > distance:
            mouse.move(wheel=int(scroll_dir))
        # otherwise move mouse cursor in x and y directions
        else:
            mouse.move(x=int(horizontal_mov))
            mouse.move(y=int(vertical_mov))

        # debugging print for x and y values
        # time.monotonic() is used so that the
        # code is not delayed with time.sleep
        if (clock + 2) < time.monotonic():
            print("x", steps(x))
            print("y", steps(y))

```

```
clock = time.monotonic()

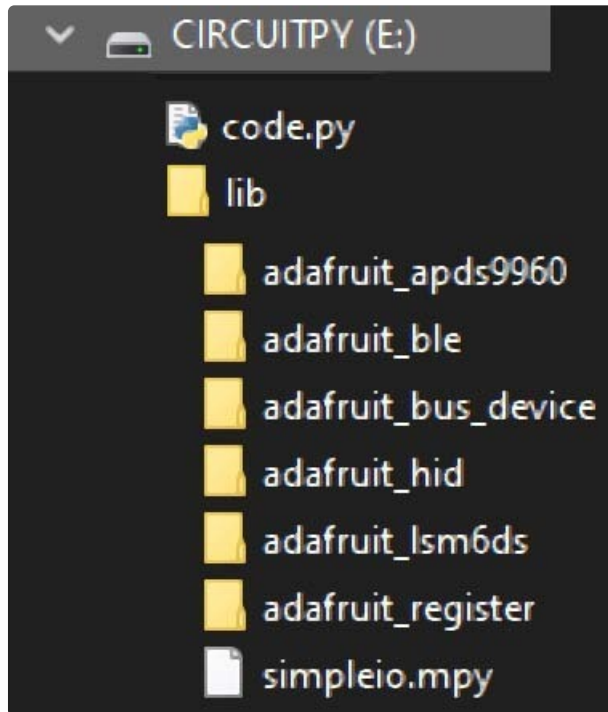
ble.start_advertising(advertisement)
```

Upload the Code and Libraries to the Feather nRF52840 Sense

After downloading the Project Bundle, plug your Feather nRF52840 Sense into the computer's USB port. You should see a new flash drive appear in the computer's File Explorer or Finder (depending on your operating system) called **CIRCUITPY**. Unzip the folder and copy the following items to the Feather nRF52840 Sense's **CIRCUITPY** drive.

- **lib** folder
- **code.py**

Your Feather nRF52840 Sense **CIRCUITPY** drive should look like this after copying the lib folder and the **code.py** file.



How the CircuitPython Code Works

The Feather nRF52840 Sense is able to use the onboard accelerometer's x and y position data to move your computer's mouse around the screen. The [lsm6ds33](#) is setup at the beginning of the code.

```
# setup accelerometer
lsm6ds33 = adafruit_lsm6ds.lsm6ds33.LSM6DS33(i2c)
```

In the loop, `x` and `y` are setup to hold the value of the `lsm6ds33`'s coordinates.

The `map_range()` function from the `simpleio` library is used to convert the accelerometer data to usable mouse movement data for both vertical and horizontal movement. The degree of motion from the accelerometer also affects the speed of the mouse's cursor movements.

Additionally, `map_range()` is used for scrolling, converting the mapped vertical movement of the mouse to scrolling direction and speed.

```
while ble.connected:
    # sets x and y values for accelerometer x and y values
    # x and y are swapped for orientation of feather
    y, x, z = lsm6ds33.acceleration

    # map range of horizontal movement to mouse x movement
    horizontal_mov = simpleio.map_range(steps(x), 1.0, 20.0, -15.0, 15.0)
    # map range of vertical movement to mouse y movement
    vertical_mov = simpleio.map_range(steps(y), 20.0, 1.0, -15.0, 15.0)
    # map range of mouse y movement to scrolling
    scroll_dir = simpleio.map_range(vertical_mov, -15.0, 15.0, 3.0, -3.0)
```

Left-clicks are sent with the Feather nRF52840 Sense's onboard button.

```
# if onboard button is pressed, sends left mouse click
if not click.value:
    mouse.click(Mouse.LEFT_BUTTON)
    time.sleep(0.2)
```

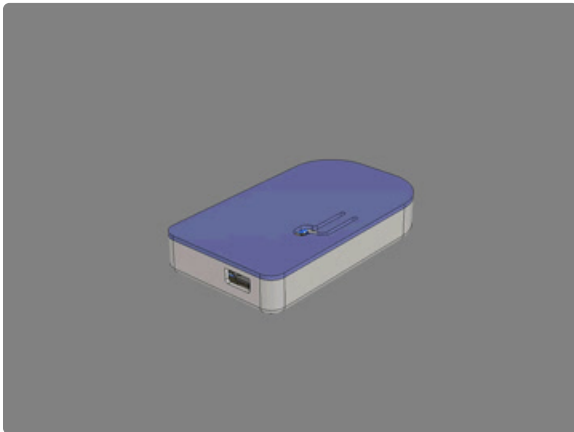
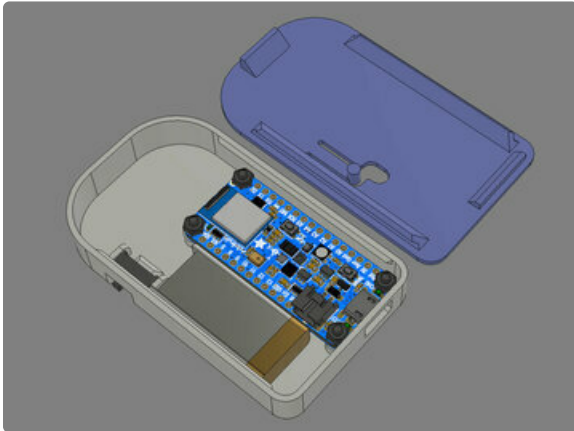
Scrolling is done by covering the APDS9960 proximity sensor and tilting the Feather Sense up or down. If the proximity sensor is covered, then `mouse.move(wheel=int(scroll_dir))` is sent. Otherwise, the mouse's cursor will move freely around the screen.

```
# if the proximity sensor is covered
# scroll the mouse
if apds9960.proximity > distance:
    mouse.move(wheel=int(scroll_dir))
# otherwise move mouse cursor in x and y directions
else:
    mouse.move(x=int(horizontal_mov))
    mouse.move(y=int(vertical_mov))
```

The value that triggers scrolling can be adjusted with the `distance` variable if you need to increase or decrease sensitivity.

```
# variable for distance for proximity scrolling  
distance = 245
```

3D Printing



CAD Parts List

STL files for 3D printing are oriented to print "as-is" on FDM style machines. Parts are designed to 3D print without any support material. Original design source may be downloaded using the links below.

- [blemouse-top-cover.stl](#)
- [blemouse-case.stl](#)

CAD Assembly

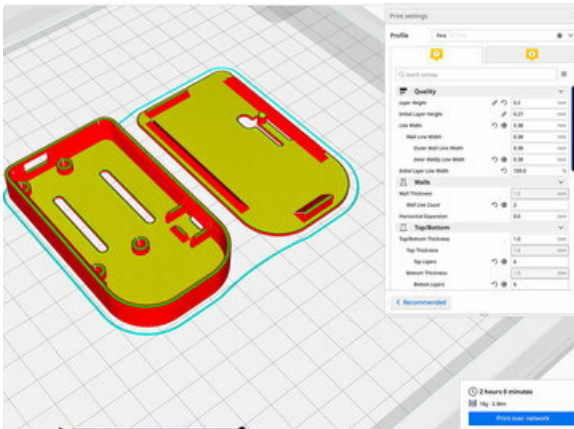
The Feather nRF52840 Sense is secured to the case with 4x M2.5 x 10mm screws and hex nuts. The slide switch press fits into the built-in holder. The battery is placed next to the Feather. The top cover snap fits over the case with access to the proximity sense and a button actuator for the on-board user switch.

<https://adafru.it/UEg>

<https://adafru.it/UEg>

<https://adafru.it/UEh>

<https://adafru.it/UEh>

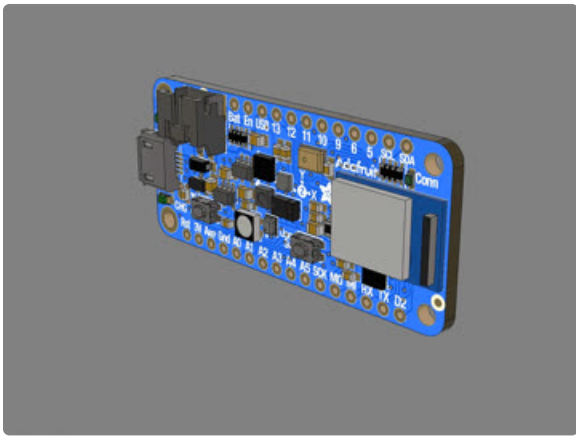


Slicing Parts

No supports are required. Slice with setting for PLA material.

The parts were sliced using CURA using the slice settings below.

- PLA filament 220c extruder
- 0.2 layer height
- 10% gyroid infill
- 60mm/s print speed
- 60c heated bed



Design Source Files

The project assembly was designed in Fusion 360. This can be downloaded in different formats like STEP, STL and more. Electronic components like Adafruit's board, displays, connectors and more can be downloaded from the [Adafruit CAD parts GitHub Repo](#) (<https://adafru.it/AW8>).

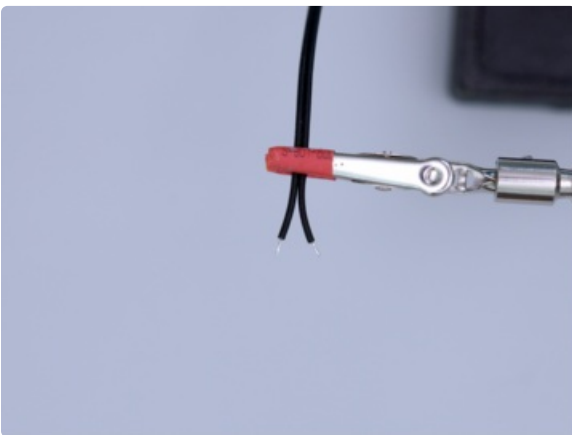
Wiring Switch



Wire for Switch

Use a piece of ribbon cable to connect the switch to the Feather. A 2-wire piece will connect to the Enable and Ground pins.

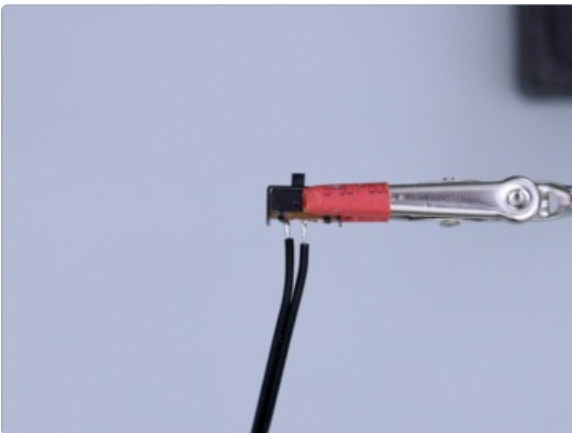
Wire Length: 9cm (3.5in)



Tinning Wire

Using wire stripper, remove a bit of insulation for the tips of each wire. Add a bit of solder to the exposed wire. This will help to prevent the strands of wire from fraying.

Use a pair of helping third hands to hold the wire in place while soldering.

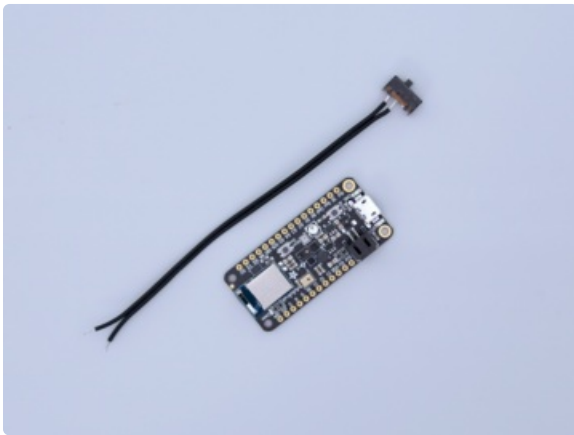


Solder Wire

Cut one of the three leads from the slide. Either the far left or right but not the middle! Trim the remaining leads short, about half the original length.

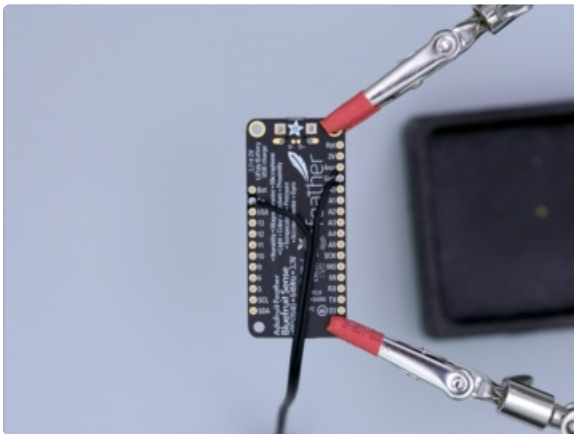
Add a bit of solder to the leads of the slide switch. This will make it easier to attach the wires.

Solder the 2-wire ribbon cable to the leads on the slide switch.



Wired Switch

Double check the solder joints in the wire and slide are solid and properly soldered.

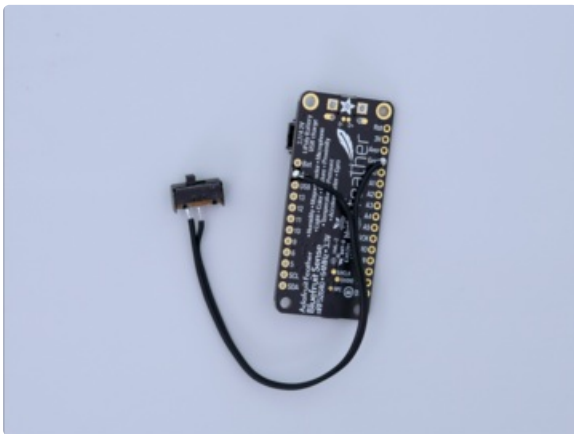


Solder Switch to Feather

Add a bit of solder to the Enable and Ground pins on the bottom of the Feather.

Solder the wires to the Enable and Ground pins on the Feather.

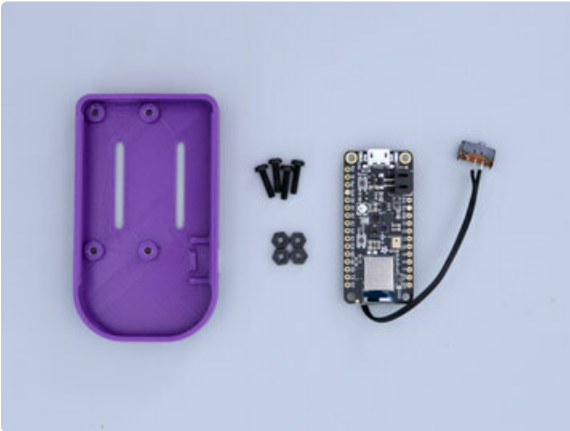
Use helping third hands to keep the board in place while soldering.



Feather Switch

Double check the Enable and Ground pins have been properly soldered to the wires.

Assembly



Screws for Case

Use the following hardware to secure the Feather to the case.

- 4x M2.5 x 10mm screws
- 4x M2.5 hex nuts



Install Hardware

Start by inserting an M2.5 screw through one of the mounting holes on the bottom of the case.



Secure Feather

While holding the screw in place, install the Feather into the case with the USB port facing the cut out. Line up the mounting holes and fit the screw through the mounting hole.

Insert and fasten hex nuts to the screw to secure the Feather to the case. Repeat for the remaining mounting holes.



Install Switch

Insert the slide switch into the built-in holder with the actuator fitting through the hole. Fit the body of the slide switch at an angle and firmly press fit down into place.



Switch Actuator

The actuator from the switch should be accessible through the side of the case. The built-in holder keeps the slide switch in place and prevents it from being pressed all the way into the case.



Connect Battery

Plug in the power cable from the 400mAh battery to the JST connector on the Feather.



Install Battery

Fit the battery into the case and place it right above the slide switch holder. Optionally use mounting tack or double-sided tape to keep the battery secured in place.



Install Cover

Orient the top cover with the case. The top cover features a built-in actuator for the User Switch on the Feather. The hole allows accessible to the proximity sensor.



Snap Fit Cover

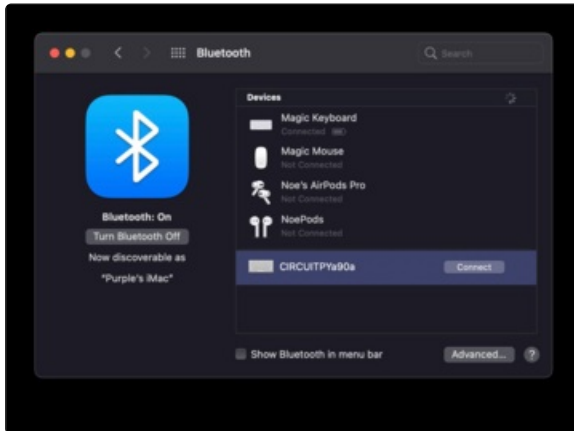
Place the top cover over the case and firmly press them together to snap fit closed.



Case USB port

The microUSB port is accessible on the back side of the case. Use the microUSB port for programming and recharging the 400mAh battery.

BLE Pairing



BLE Pairing – MacOS

To pair the CircuitPython device to macOS, open the System Preferences app. Click on the Bluetooth icon.

In the bottom of the list of BLE devices, click on the **CIRCUITPY** device. Click the connect button to pair.

To disconnect or remove the BLE device, click on the X icon next to the label.

