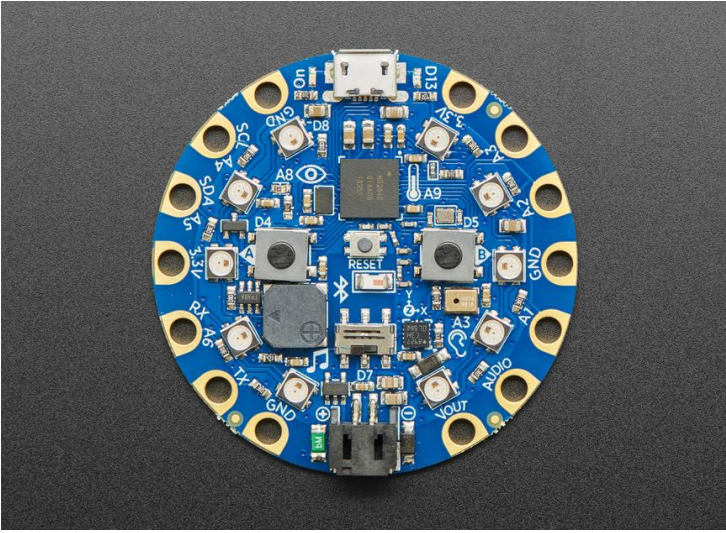


Circuit Playground Bluefruit CircuitPython Quickstart



The Adafruit Circuit Playground Bluefruit (CPB) has CircuitPython on board! It's an nRF52840 Cortex M4 processor with BLE support, plus a 2MB external flash chip for the **CIRCUITPY** USB drive. The board is loaded with all kinds of sensors, LEDs, touchpads, buttons and more!!

Check out these Adafruit Learn Guides and Links!

Welcome to CircuitPython: adafru.it/cpy-welcome

CPB Guide: adafru.it/adafruit-cpb

CP Made Easy on CPB: adafru.it/cp-made-easy

Download CircuitPython for CPB: adafru.it/cp-cpb

PyCon 2023 Content: adafru.it/pycon2023

Plug It In!

Use a micro-USB cable with data (beware charge/power-only cables). A USB drive called **CIRCUITPY** will appear. If there's a **code.py** on **CIRCUITPY**, it will run automatically.

Auto-Reload

Every time you write a file, **code.py** will be re-run, unless you are in the REPL. Simply edit **code.py** and see it run right away. This makes for a fast workflow!

Editing Code (Recommended Editors)

If you already have a favorite code editor, you can use it. Be sure you're using one that writes back immediately: VS Code, Atom (with fsync-on-save package), Sublime, gedit, vim with **-n** option, emacs, PyCharm with Safe Write, IDLE. **Don't use** Notepad, nano. **Welcome->Advanced Setup->Recommended Editors** has more details.

Another Editor Option

Mu is the easiest editor to use: it includes a Python editor and easy serial REPL access. See **Welcome->Installing Mu Editor**. The latest versions for all operating systems are available at <https://codewith.mu>, with install instructions. For Linux, or any OS, you can create a **venv** (if you wish), and use **pip3** to install Mu:
`pip3 install --user mu-editor`

Libraries

CircuitPython has built in native libraries, but also has libraries written in Python (which are compiled into **.mpy** files to save space). The board has a **lib** folder containing all the necessary libraries for the provided examples. If you want to try more complex examples or use external accessories, you may need to download more libraries. See

Welcome->CircuitPython Libraries.

Avoiding Filesystem Corruption

Windows and Linux **don't** write back data to **CIRCUITPY** immediately: they can delay for 10s of seconds. (Not an issue on MacOS.) **Eject or sync after you copy files, and always before you unplug or press the Reset button.** Otherwise **CIRCUITPY** may become corrupted. See **Editing Code** for recommended editors. If **CIRCUITPY** does get corrupted, see **Restoring or Installing CircuitPython** in this Quickstart.

Restoring or Installing CircuitPython

<https://adafru.it/cp-cpb> has the current version of the CircuitPython UF2 for the Circuit Playground Bluefruit.

WARNING: In rare cases, updating can result in the loss of any files on CIRCUITPY on the board - backup your files if possible first! To restore or update the board, double-tap the reset button found in the center of the board. The LEDs will flash red and then turn green, and you'll see a **CPLAYBTBOOT** drive show up on the computer. Copy the **.uf2** file to **CPLAYBTBOOT**. It will disconnect and the drive will disappear. A few seconds later, CIRCUITPY will reappear. If this does not resolve the issue, check out **Welcome->Troubleshooting->To erase CIRCUITPY** for instructions to fully erase the filesystem. **The steps found here WILL erase everything on the board.**

Connecting to the Serial Console

The serial console and REPL are built into **Mu** - simply click the icon labeled "Serial".

If not using Mu, try **Putty** or **Tera Term** (Windows), or **screen**, **tio** or **picocom** (Mac and Linux), or any other terminal emulator you may already be using. Use tab completion for the paths on Mac `/dev/tty.usbmodem*` or Linux `/dev/ttyACM*` while entering screen commands.

To connect using screen on Mac:

```
screen /dev/tty.usbmodem* 115200
```

To connect using screen on Linux:

```
screen /dev/ttyACM0 115200
```

Beyond the Circuit Playground Bluefruit

There are more than 70 CircuitPython-compatible Adafruit boards - adafru.it/cp-boards. There are also a multitude of sensor breakouts, accessories, and more. If you're interested in continuing your exploration, check out the Adafruit shop at adafruit.com. Use code PYCON2023 for a 10% discount!

The CircuitPython REPL!

Once connected to the serial console, type Enter if necessary to start the REPL. If **code.py** is running, type ctrl-C, then press enter. Type ctrl-D to soft-restart and reload the serial console.

REPL example:

```
>>> 1+2
3
```

To paste indented code blocks into the REPL, type ctrl-E to enter paste mode.

Blink - the “Hello World” of CircuitPython!

Make the CPB red LED blink. Run the REPL or **code.py**:

```
import time
from adafruit_circuitplayground import cp
while True:
    cp.red_led = not cp.red_led
    time.sleep(0.5)
```

Use the slide switch to light up the red LED!

```
from adafruit_circuitplayground import cp
while True:
    cp.red_led = cp.switch
```

Light Levels!

```
import time
from adafruit_circuitplayground import cp
while True:
    print(cp.light)
    time.sleep(0.5)
```

What’s the Temperature?

```
import time
from adafruit_circuitplayground import cp
while True:
    print(cp.temperature)
    time.sleep(0.5)
```

Capacitive Touch Detection!

```
import time
from adafruit_circuitplayground import cp
while True:
    if cp.touch_A1:
        print("Touched A1!")
        time.sleep(0.05)
```

Button Presses!

```
from adafruit_circuitplayground import cp
while True:
    if cp.button_a:
        print("Button A pressed!")
    if cp.button_b:
        print("Button B pressed!")
```

Light Up the First NeoPixel LED!

```
from adafruit_circuitplayground import cp
cp.pixels.brightness = 0.3
while True:
    cp.pixels[0] = (255, 0, 0)
```

Light up all the NeoPixel LEDs!

```
from adafruit_circuitplayground import cp
while True:
    cp.pixels.fill((0, 50, 0))
```

Make a tone!

```
from adafruit_circuitplayground import cp
while True:
    cp.play_tone(292, 1)
```

Now try combining the concepts to see what you can do!

Use the buttons to light up the NeoPixel LEDs!

```
from adafruit_circuitplayground import cp
while True:
    if cp.button_a:
        cp.pixels.fill((0, 50, 0))
    if cp.button_b:
        cp.pixels.fill((0, 0, 50))
```

Use the buttons to play tones!

```
from adafruit_circuitplayground import cp
while True:
    if cp.button_a:
        cp.start_tone(262)
    elif cp.button_b:
        cp.start_tone(294)
    else:
        cp.stop_tone()
```

Use the touch pads to play tones and light up NeoPixels!

```
from adafruit_circuitplayground import cp
cp.pixels.brightness = 0.3
while True:
    if cp.touch_A1:
        cp.pixels.fill((255, 0, 0))
        cp.start_tone(262)
    elif cp.touch_A2:
        cp.pixels.fill((255, 15, 0))
        cp.start_tone(294)
    elif cp.touch_A3:
        cp.pixels.fill((180, 180, 0))
        cp.start_tone(330)
    elif cp.touch_A4:
        cp.pixels.fill((0, 180, 180))
        cp.start_tone(349)
    else:
        cp.pixels.fill((0, 0, 0))
        cp.stop_tone()
```