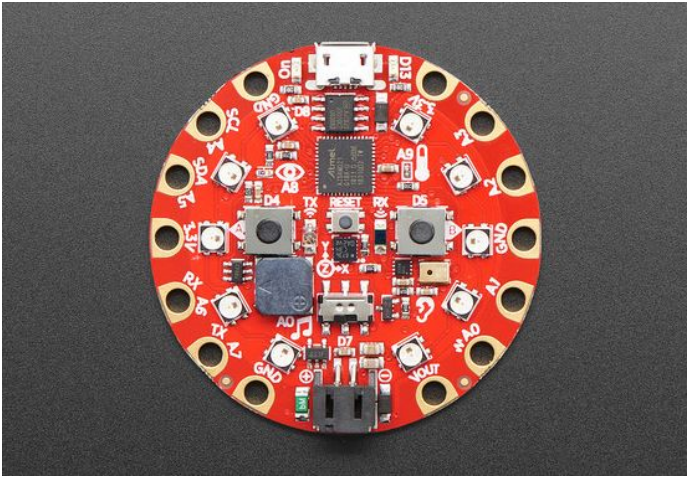# Circuit Playground Express CircuitPython Quickstart

The Adafruit Circuit Playground Express (CPX) has CircuitPython on board! It's a Microchip SAMD21 microcontroller running at 48 MHz, with 256kb flash, plus a 2MB external flash chip for the CIRCUITPY USB drive. It's loaded with sensors, LEDs, touch pads, buttons and more!

Check out these Adafruit Learn Guides and Links!
Welcome to CircuitPython: adafru.it/cpy-welcome
CPX Guide: adafru.it/adafruit-cpx
CP Made Easy on CPX: adafru.it/cp-made-easy-on-cpx
Download CircuitPython for CPX: https://adafru.it/cp-cpx
PyCon 2019 Content: https://adafru.it/pycon2019

Are you on Windows 7?
You need to install drivers before plugging in! See **Welcome->Installing CircuitPython**.
Windows 10, Mac, and Linux don't need drivers.

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.

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. Continue reading to see editors that write immediately so you don't need to Eject or sync every time you edit. If **CIRCUITPY** does get corrupted, see **Restoring or Installing CircuitPython** in this Quickstart.

Editing Code
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. *Don't use* Notepad, nano, IDLE. **Welcome-> Creating and Editing Code** 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 version for Windows and Mac are available at https://codewith.mu. For Linux, or any OS, you can create a **venv** and use **pip3** to install Mu:
```
pip3 install --user mu-editor
```

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!

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 does not currently have a **lib** folder because all of the necessary libraries for these examples are included in CircuitPython for CPX. But, if you want to try more complex examples or use external accessories in the future, you'll need to download the right libraries. See **Welcome->CircuitPython Libraries**.

Restoring or Installing CircuitPython
https://adafru.it/cp-cpx has the current version of the CircuitPython UF2 for the Circuit Playground Express.
WARNING: This process will rarely result in the loss of any files on the board - backup your files if possible first!
To restore or update your 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 **CPLAYBOOT** drive show up on your computer. Copy the **.uf2** file to **CPLAYBOOT**. It will disconnect and the drive will disappear. A few seconds later, CIRCUITPY will reappear. If this does not resolve your issue, check out **Welcome->Troubleshooting-> CIRCUITPY Drive Issues** 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** 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
```

Interact with the 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 your CPX red LED blink. Type this into the REPL or **code.py**:
```
import time
from adafruit_circuitplayground.express import cpx
while True:
    cpx.red_led = not cpx.red_led
    time.sleep(0.5)
```

Use the slide switch to light up the red LED!
```
from adafruit_circuitplayground.express import cpx
while True:
    cpx.red_led = cpx.switch
```

Light Levels!
```
import time
from adafruit_circuitplayground.express import cpx
while True:
    print(cpx.light)
    time.sleep(0.5)
```

What's the Temperature?
```
import time
from adafruit_circuitplayground.express import cpx
while True:
    print(cpx.temperature)
    time.sleep(0.5)
```

Capacitive Touch Detection!
```
import time
from adafruit_circuitplayground.express import cpx
while True:
    if cpx.touch_A1:
        print("Touched A1!")
        time.sleep(0.05)
```

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

Light Up the First NeoPixel LED!
```
from adafruit_circuitplayground.express import cpx
cpx.pixels.brightness = 0.3
while True:
    cpx.pixels[0] = (255, 0, 0)
```

Light up all the NeoPixel LEDs!
```
from adafruit_circuitplayground.express import cpx
while True:
    cpx.pixels.fill = (0, 50, 0)
```

Make a tone!
```
from adafruit_circuitplayground.express import cpx
while True:
    cpx.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.express import cpx
while True:
    if cpx.button_a:
        cpx.pixels.fill((0, 50, 0))
    if cpx.button_b:
        cpx.pixels.fill((0, 0, 50))
```

Use the buttons to play tones!
```
from adafruit_circuitplayground.express import cpx
while True:
    if cpx.button_a:
        cpx.start_tone(262)
    elif cpx.button_b:
        cpx.start_tone(294)
    else:
        cpx.stop_tone()
```

Use the touch pads to play tones and light up NeoPixels!
```
from adafruit_circuitplayground.express import cpx
cpx.pixels.brightness = 0.3
while True:
    if cpx.touch_A1:
        cpx.pixels.fill((255, 0, 0))
        cpx.start_tone(262)
    elif cpx.touch_A2:
        cpx.pixels.fill((255, 15, 0))
        cpx.start_tone(294)
    elif cpx.touch_A3:
        cpx.pixels.fill((180, 180, 0))
        cpx.start_tone(330)
    elif cpx.touch_A4:
        cpx.pixels.fill((0, 180, 180))
        cpx.start_tone(349)
    else:
        cpx.pixels.fill((0, 0, 0))
        cpx.stop_tone()
```