Make It a Keyboard

Created by Anne Barela
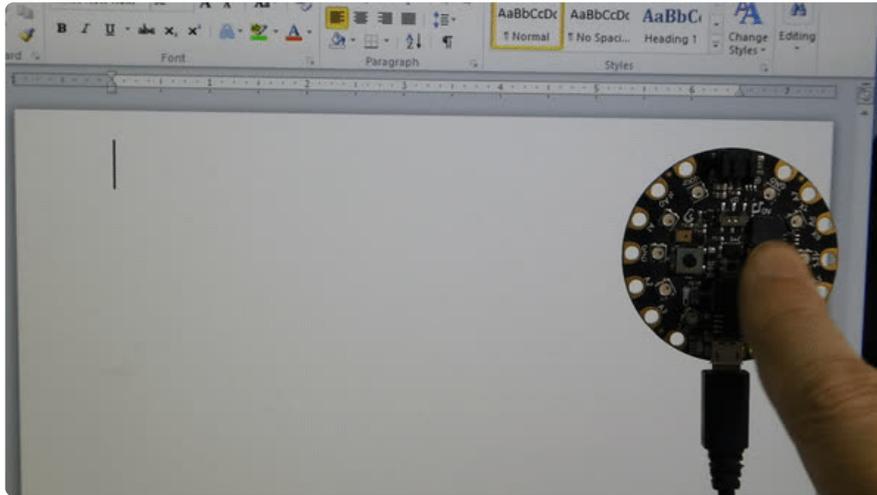

https://learn.adafruit.com/make-it-a-keyboard
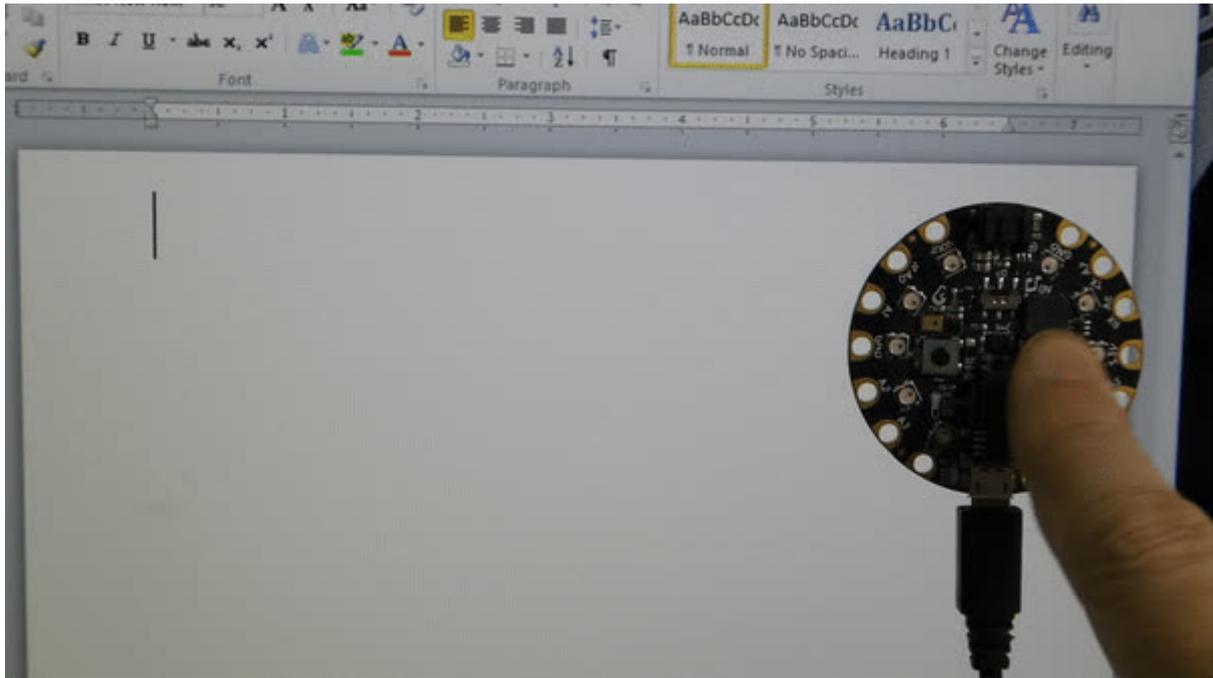
Last updated on 2024-06-03 02:26:34 PM EDT

Table of Contents

Overview

There are a great number of things one can do if they are able to interface their electronics to their desktop / laptop computer. This process used to be kinda hard but it isn't now, thanks to USB!

The USB specification has several ways to characterize devices. This includes telling the computer that the device you are plugging into the USB port is a keyboard or mouse (https://adafru.it/C1T). These peripherals are categorized as human interface devices (HID).

Many Adafruit products can communicate with other computers using HID and emulating a keyboard.

What types of projects work well emulating a keyboard? Some examples:

- **Macro keys** - activating to send a predefined sequence of keystrokes
- **Assistive Technology** - the ability to trigger keyboard input using a multitude of interface devices connected to the microcontroller
- **Device control** - the ability to send multimedia commands for volume, etc.
- **Sensor input** - Turn a banana or some slime into a keyboard with the use of capacitive touch
- **Fun** - a simple prank would send some characters to someone's screen at random times

This guide will show you how to quickly and easily emulate a keyboard on an Adafruit Circuit Playground Express board in both Microsoft MakeCode and CircuitPython.

Parts List
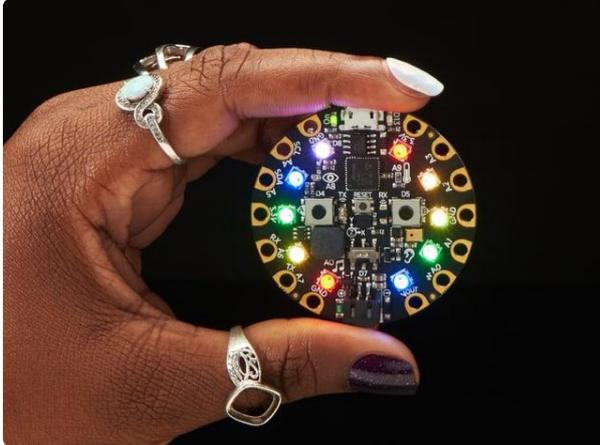
Circuit Playground Express

Circuit Playground Express is the next step towards a perfect introduction to electronics and programming. We've taken the original Circuit Playground Classic and...

https://www.adafruit.com/product/3333

USB cable - USB A to Micro-B

This here is your standard A to micro-B USB cable, for USB 1.1 or 2.0. Perfect for connecting a PC to your Metro, Feather, Raspberry Pi or other dev-board or...

https://www.adafruit.com/product/592
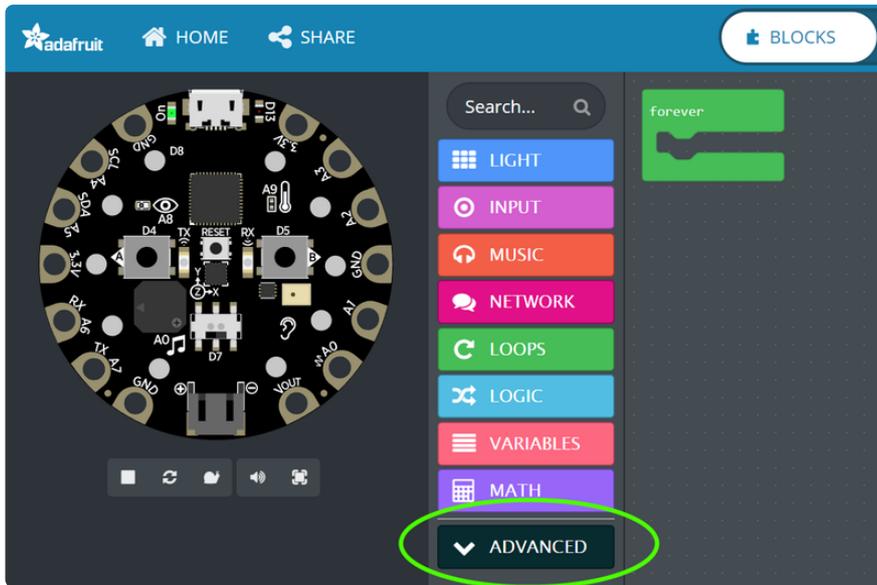
Microsoft MakeCode

If you are new to Microsoft MakeCode, you can learn the basics of MakeCode here (https://adafru.it/wWd).
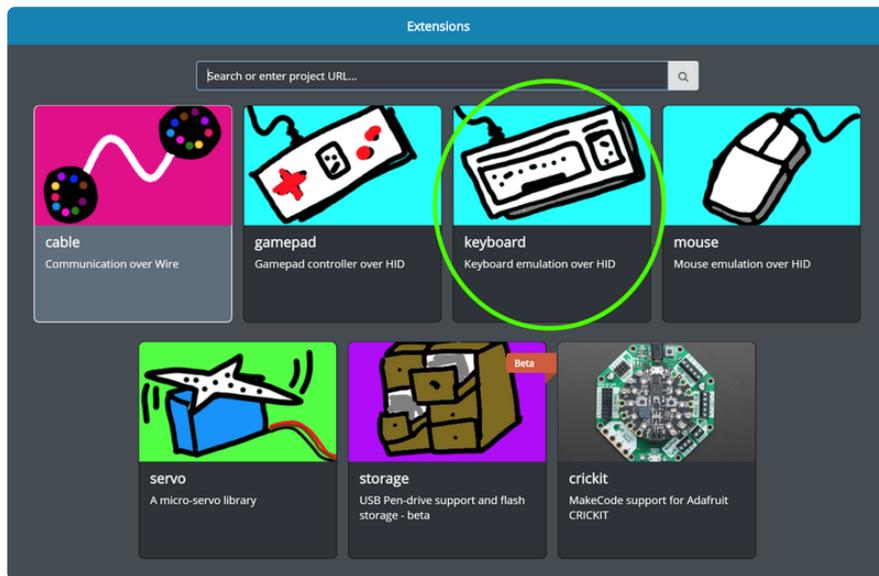
To add keyboard support, MakeCode needs to load an extension - bring in some blocks not in the main program. Click on the **ADVANCED** blog group:

Then click on the **EXTENSION** block:

And select the **keyboard** - Keyboard emulation over HID extension:

This adds a new block group on the main MakeCode page called **KEYBOARD** with several new functional blocks when you click the group button:
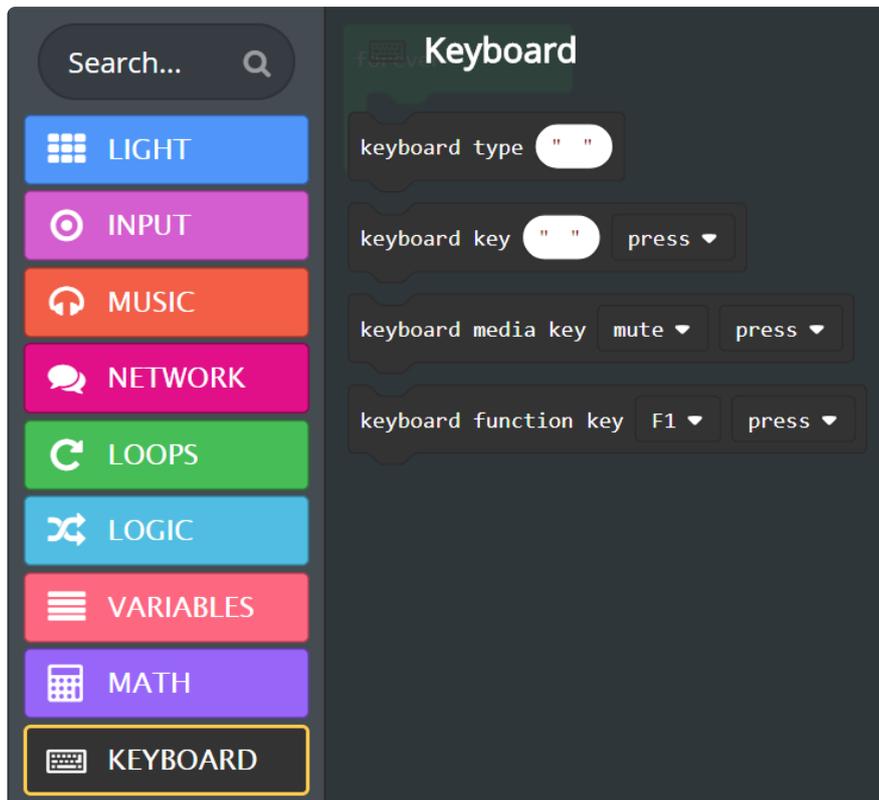
At this point you will want to define some action that lets your program know you wish to use one of the Keyboard functions. This can be anything but for example you may want to trigger keyboard input on:

- Press of the A or B buttons or the slide switch
- Press of a capacitive touch pad (marked A1 to A7)
- On a loud sound, shake of the board, or bright light

• Periodically send a key after several seconds, minutes, hours, etc.

For a simple demonstration, here are two button event blocks from the **INPUT** block group `on button click` - the A button on Circuit Playground Express prints ABC as text, it can be any text you want to type with only a button press. If Button B is pressed, if you have a music app playing it will pause playing - it works for sure if you have iTunes or Spotify, but not all apps listen for the media key.
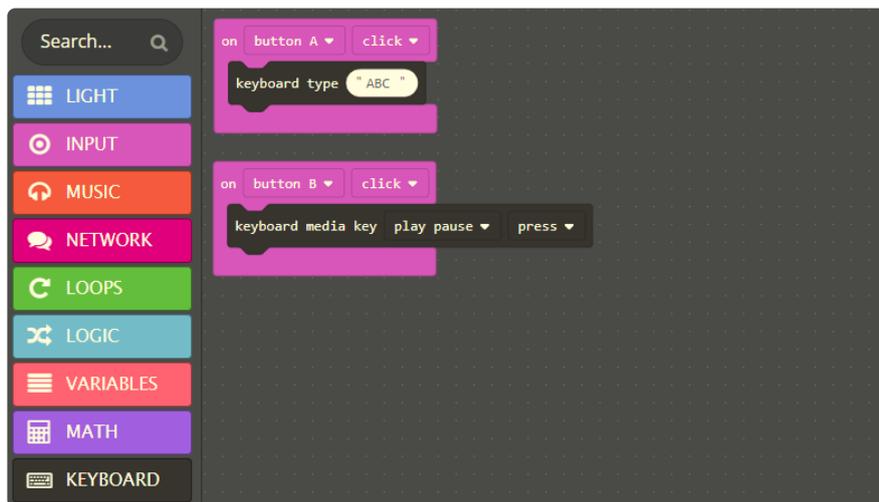
Download this Example in MakeCode

https://adafru.it/B-5

Feel free to define other actions and have fun with your new keyboard!

Frequently Asked Questions

Do I have to unplug my other keyboard?

No, all modern operating systems allow more than one keyboard to be connected at a time.

How can I see the Circuit Playground Express as a device?

For Windows, open up the Control Panel application, go to Devices and Printers then look under the Devices group for a keyboard icon labeled "PXT Device":

PXT Device

CircuitPython

If you are new to CircuitPython, we recommend you start with the Welcome to CircuitPython Guide (https://adafru.it/Bid) then come back here.

Adafruit recommends installing and using the Mu editor, available for Windows, macOS, and Linux. You can learn about Mu in this guide (https://adafru.it/ANO).

The adafruit_hid Library

Adafruit has developed libraries to easily use CircuitPython devices which have HID capabilities as keyboard devices. There are several functions you may wish to use in your programs, this guide will show them with examples.

On the Circuit Playground Express, this library is built-in! On other CircuitPython boards you may have to install the library by hand, see this guide on how to do that (https://adafru.it/ABU), it's easy and just requires dragging a file onto your board.

The adafruit_circuitplayground Library

Adafruit has developed a library to access much of the functionality on the Circuit Playground Express called `adafruit_circuitplayground`.

You can use the library by defining `from adafruit_circuitplayground.express import cpx` at the start of your code. The library has functions for all the sensors, buttons, etc. The examples will use `cpx.button_a` and `cpx.button_b` to read the pushbuttons.

Sending Individual Keys

A keyboard typically has well over a hundred keys with various combinations of shift, ctrl, alt, and caps. CircuitPython can send all of these but you need to know what you want to send. The following sends a capital A on Button A and a ctrl X on Button B:

```
# SPDX-FileCopyrightText: 2018 Anne Barela for Adafruit Industries
#
# SPDX-License-Identifier: MIT

import usb_hid
from adafruit_circuitplayground.express import cpx
from adafruit_hid.keyboard import Keyboard
from adafruit_hid.keycode import Keycode

kbd = Keyboard(usb_hid.devices)

while True:
    if cpx.button_a:
        kbd.send(Keycode.SHIFT, Keycode.A)  # Type capital 'A'
        while cpx.button_a: # Wait for button to be released
            pass

    if cpx.button_b:
        kbd.send(Keycode.CONTROL, Keycode.X)  # control-X key
        while cpx.button_b: # Wait for button to be released
            pass
```

If a button is pushed, `kbd.send` sends a defined key. Then the `while` loop waits while the button is still down before checking for a new key to prevent rollover, getting a lot of sent characters for one button press.

For a list of all the keys and codes you can send, see this code (https://adafru.it/Ene).

Send Multimedia Keys

You can also send multimedia keycodes. The `adafruit_hid.consumer_control` library provides the functionality and the

`adafruit_hid.consumer_control_code` library provides predefined values for multimedia keys.

The keys you can select are defined in [this Read The Docs page](https://adafru.it/AOi) (https://adafru.it/AOi).

The following program will increment and decrement the volume on a USB controllable multimedia device (such as my Windows 10 PC) using the Circuit Playground Express Buttons A and B. The program will only send one code per press, if the button is held down it only sends one code due to the `while` statement. The `while` statement blocks can be removed to simulate press and hold to change volume.

```
# SPDX-FileCopyrightText: 2018 Anne Barela for Adafruit Industries
#
# SPDX-License-Identifier: MIT

import usb_hid
from adafruit_circuitplayground.express import cpx
from adafruit_hid.consumer_control import ConsumerControl
from adafruit_hid.consumer_control_code import ConsumerControlCode

cc = ConsumerControl(usb_hid.devices)

while True:
    if cpx.button_a:
        cc.send(ConsumerControlCode.VOLUME_INCREMENT)
        while cpx.button_a:
            pass
    if cpx.button_b:
        cc.send(ConsumerControlCode.VOLUME_DECREMENT)
        while cpx.button_b:
            pass
```

Sending Multiple Characters at Once

Here is an example of sending out a string of characters at once:

```
# SPDX-FileCopyrightText: 2018 Anne Barela for Adafruit Industries
#
# SPDX-License-Identifier: MIT

import usb_hid
from adafruit_circuitplayground.express import cpx
from adafruit_hid.keyboard import Keyboard
from adafruit_hid.keyboard_layout_us import KeyboardLayoutUS

kbd = Keyboard(usb_hid.devices)
layout = KeyboardLayoutUS(kbd)

while True:
    if cpx.button_a:
        # Type 'Jane Doe' followed by Enter (a newline).
        layout.write('Jane Doe\n')
        while cpx.button_a:
            pass
```

Keyboard Layouts

The `adafruit_hid` library currently only supports the US keyboard layout with `layout.write()`. However, layouts for keyboards for other languages and regions are available at https://github.com/Neradoc/Circuitpython_Keyboard_Layouts (https://adafru.it/UYD).

Documentation

The online documentation for the Adafruit HID library is available at on readthedocs (https://adafru.it/B-7).

Adafruit has another guide (https://adafru.it/BQj) that goes over the functionality of the adafruit_circuitplayground (CPX) library functions.

Going Further

Besides defining your own keyboard, there are some other applications of this technology. Adafruit has another guide which demonstrates using Circuit Playground Express as a keyboard for Android devices. Then, using the Android Google keyboard GBoard, it can convert Morse Code to text. If this interests you, see the guide Android GBoard Morse Code Control with Circuit Playground Express (https://adafru.it/B-8).