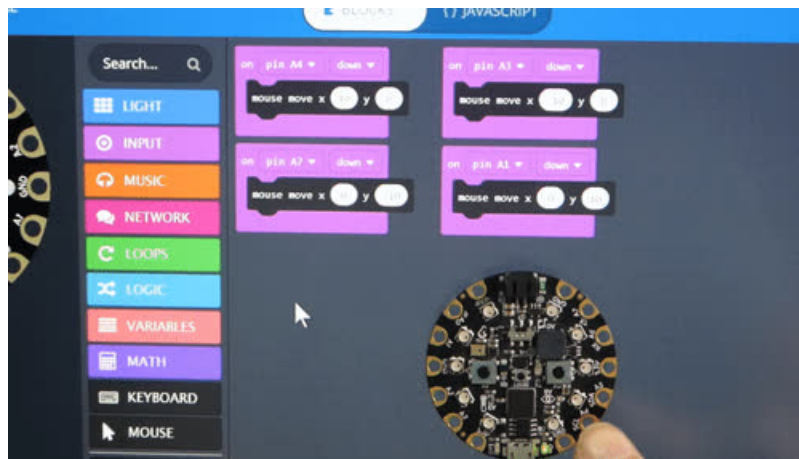




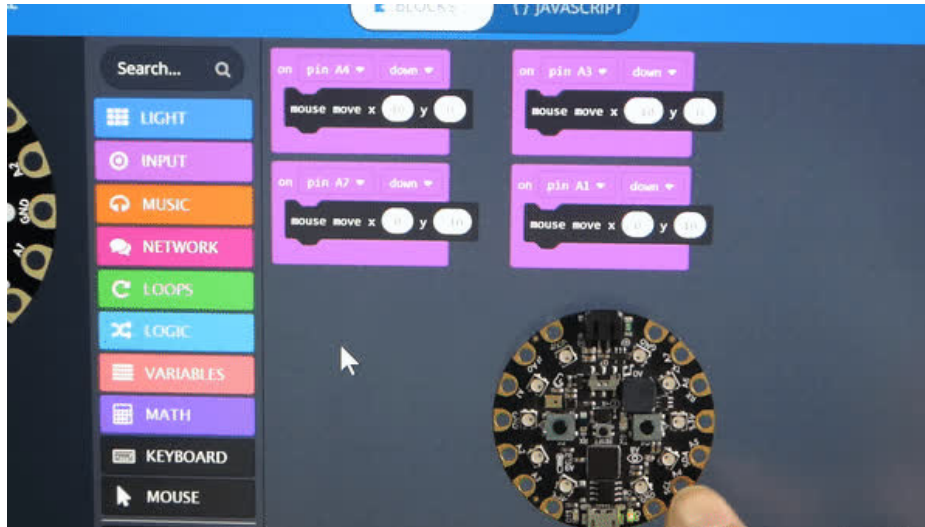
# Make It a Mouse

Created by Anne Barela



Last updated on 2020-01-08 12:14:42 AM UTC

## Overview



There are a lot of things one can do if they are able to interface their electronics to their desktop / laptop computer. It's easier than ever to make these connections, especially with Circuit Playground Express.

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](https://adafru.it/C1U) or mouse. These peripherals are categorized as human interface devices (HID).

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

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

- Assistive Technology - the ability to trigger keyboard input using a multitude of interface devices connected to the microcontroller
- Simulating mouse clicks with another device.
- Making a mouse jiggler to keep a computer from falling asleep

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

## Parts List

Your browser does not support the video tag. [Circuit Playground Express](#)

**\$24.95**  
IN STOCK

Add To Cart



USB cable - USB A to Micro-B

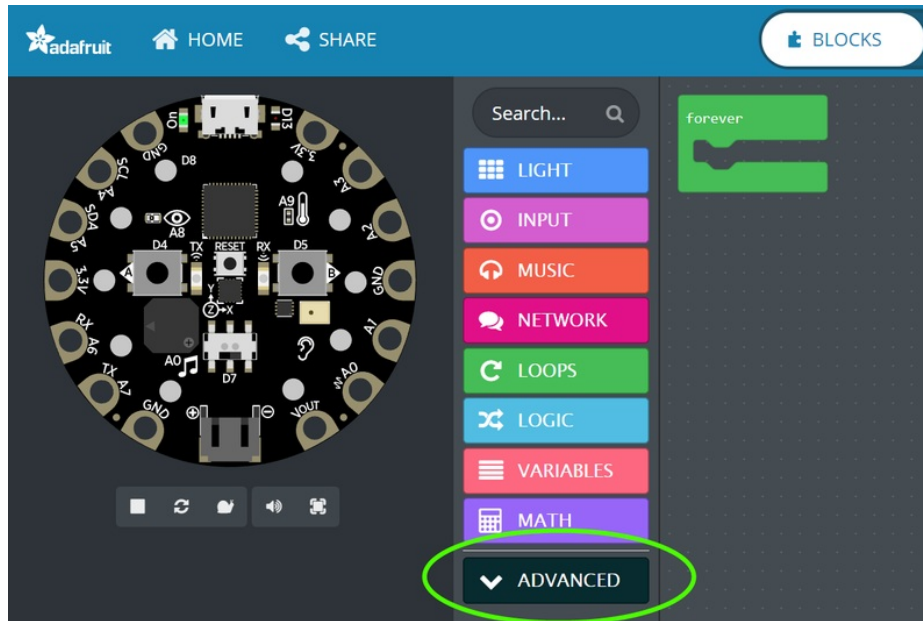
\$2.95  
IN STOCK

Add To Cart

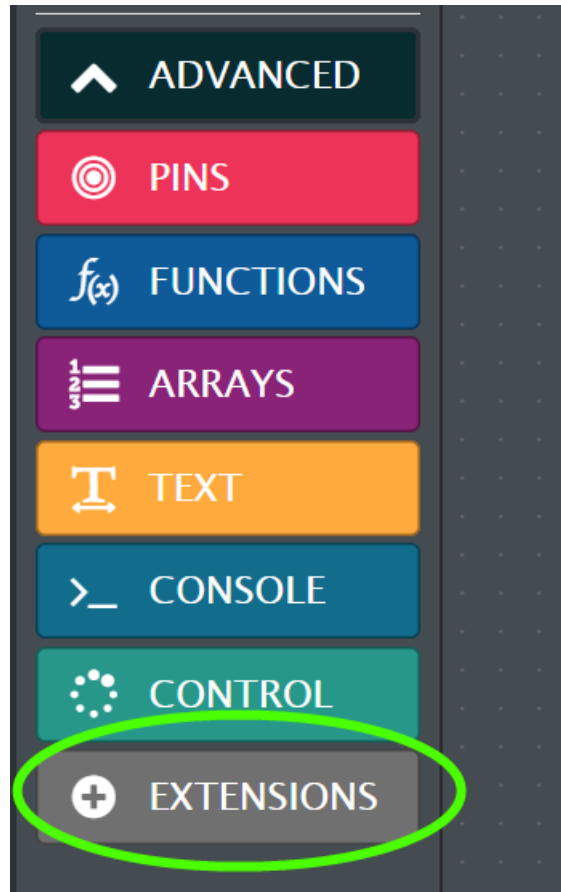
## Microsoft MakeCode

If you are new to Microsoft MakeCode, you can [learn the basics of MakeCode here \(https://adafru.it/wWd\)](https://adafru.it/wWd).

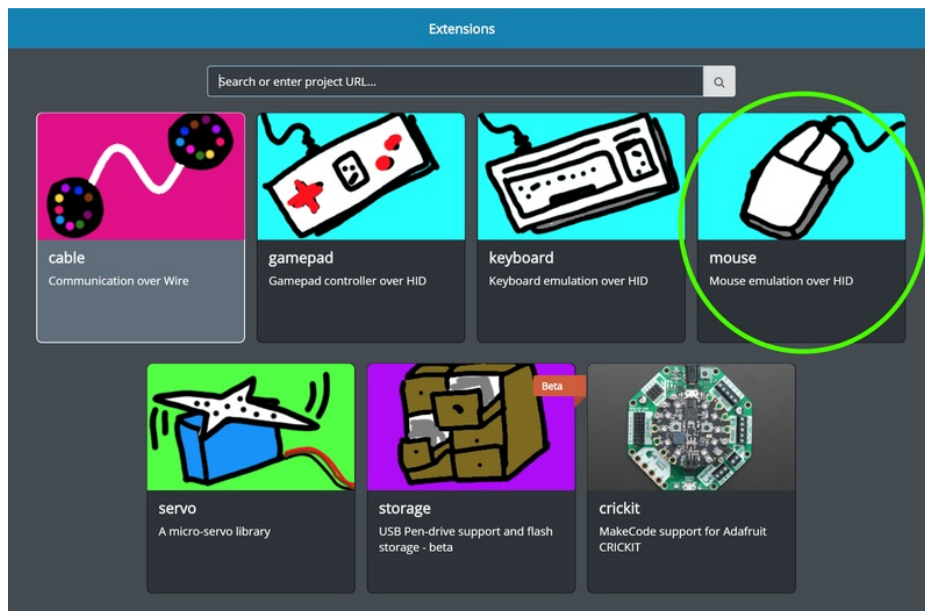
To add keyboard support, MakeCode needs to load an extension - to bring in some blocks that are not usually loaded in the main program. To do this, click on the **ADVANCED** blog group:



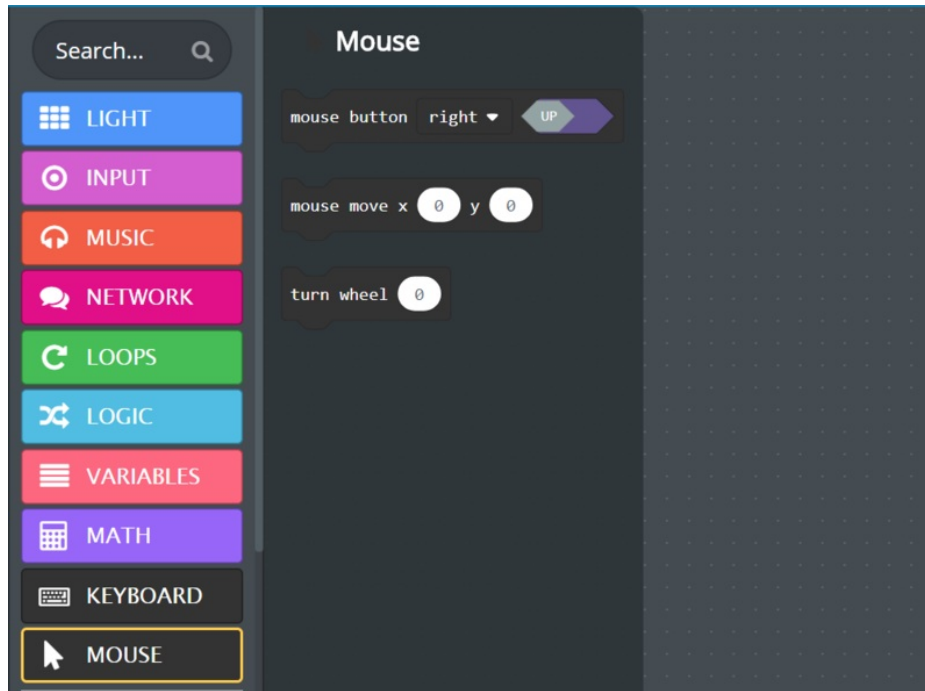
Then click on the EXTENSION block:



And select the **mouse** - Mouse emulation over HID extension:



This adds a new block group on the main MakeCode page called **MOUSE** with several new functional blocks when you click the group button:



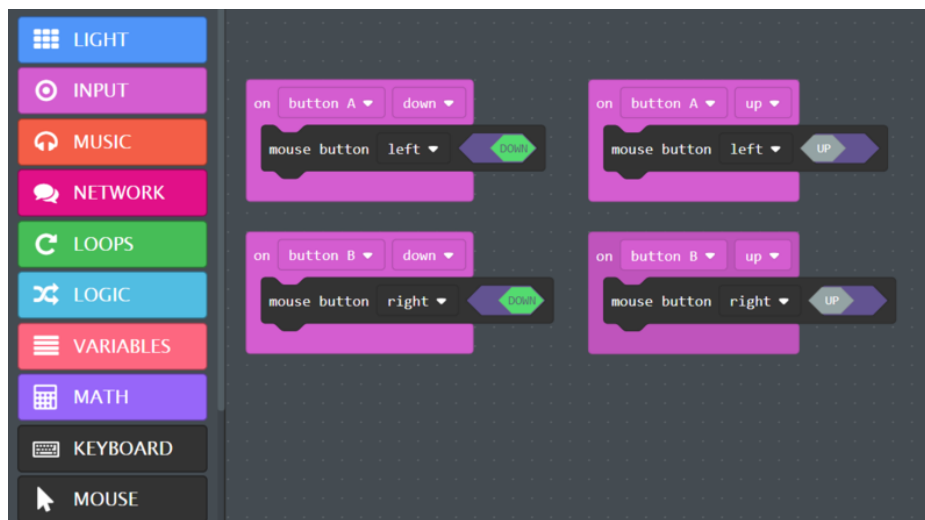
At this point you will want to define some action that lets your program know you wish to use one of the mouse functions. This can be anything but for example you may want to trigger mouse input on:

- 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 move the mouse after several seconds, minutes, hours, etc.

For a simple demonstration, here are four button event blocks from the **INPUT** block group **on button click** - the A button on Circuit Playground Express clicks the left mouse button (down when A is down, up when A is up). If Button B is pressed, it simulates the right mouse button clicked down, up when B is released.



Buttons Switched? There may be a bug in MakeCode, we alerted them to an issue, just switch button A and button B in the code and it should work if you are finding A is acting like Right Click and B as Left Click.



<https://adafru.it/B-a>

<https://adafru.it/B-a>

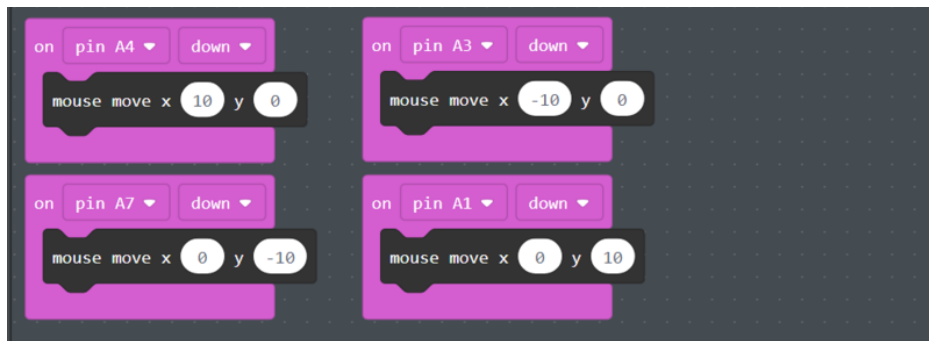
## Moving the Cursor

You can also define moving the cursor. Ideas include:

- Using the Circuit Playground Express accelerometer tilt
- Using buttons or capacitive touch on the outside pads

The following example uses the capacitive touch pads to move in the x and y direction.

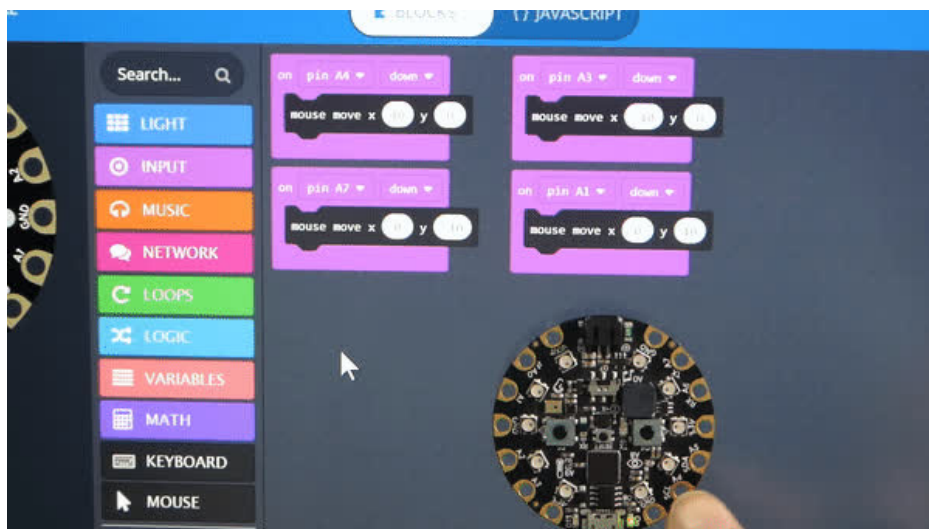
Pads **A3** and **A4** move the mouse horizontally, pads **A7** and **A1** move the cursor vertically.



<https://adafru.it/B-b>

<https://adafru.it/B-b>

Here is the code in action:



# CircuitPython



If you are new to CircuitPython, we recommend you start with the [Welcome to CircuitPython Guide \(https://adafru.it/Bid\)](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 \(https://adafru.it/ANO\)](https://adafru.it/ANO).

## The adafruit\_hid Library

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



These examples have been updated for version 4+ of the CircuitPython HID library. On the CircuitPlayground Express this library is built into CircuitPython. So, please use the latest version of CircuitPython as well. (At least 5.0.0-beta.3)

## 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.

## Performing Mouse Clicks

The following performs a mouse left click when Circuit Playground Express Button A is pressed, a right click when Button B is pressed:

Temporarily unable to load content:

When a button is pressed, the code sends the mouse button press notification. It then waits, doing nothing while the button is still pressed. When the button is not pressed anymore, the code sends a mouse release button call.

## Moving the Mouse Cursor



The following uses 4 of the capacitive touch pads on the Circuit Playground Express to signal having the mouse move in the +x, +y, -x, and -y axis. This can be changed to switch activations or other methods such as using the accelerometer chip to move based on tilt of the Circuit Playground circuit board.

Temporarily unable to load content:

## Documentation

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

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

## Going Further

The mouse keys and cursor code can be merged to one overall mouse program with buttons and cursor control.

This type of work is terrific for assistive technologies projects.

You can also make a mouse jiggle program which uses the `time` library along with the `mouse` library to move the mouse a little, sleep, then do it again. This may keep a computer awake if it has auto shutdown which you cannot change.

