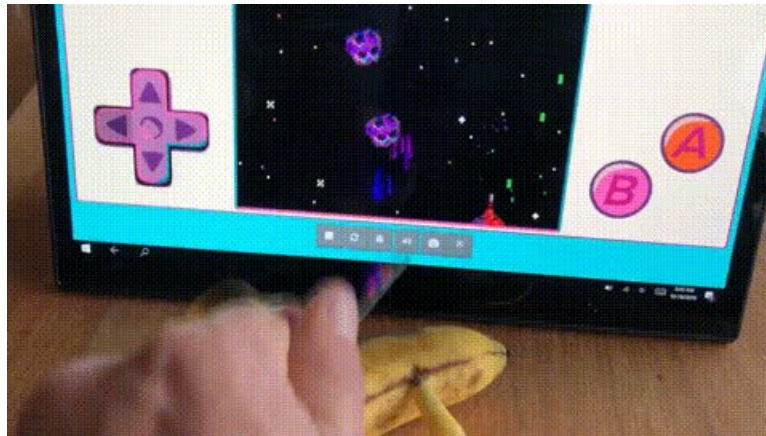




Custom Controllers for MakeCode Arcade

Created by Peli de Halleux

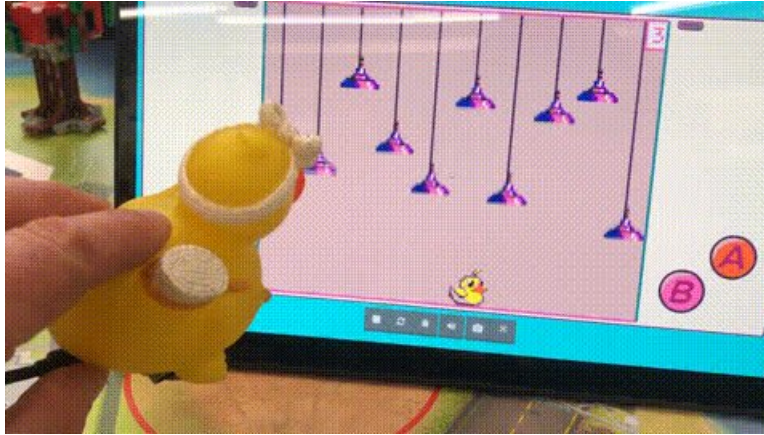


Last updated on 2019-10-22 05:58:40 PM UTC

Overview

[MakeCode Arcade](#) is a retro-gaming code editor to build cute 2D 1980's looking games. Arcade is well know here in Adafruit as it allows anyone to create games for the [pyBadge](#) or [pyGamer](#).

In this guide, we'll focus on a different approach: building custom controllers to play Arcade games in the browser.



Parts

You can choose from a number of microcontrollers supported in Microsoft MakeCode for Circuit Playground Express or MakeCode Maker. The Adafruit Circuit Playground Express has all kinds of sensors, buttons, and capacitive touch pads to make keypresses. But even the Adafruit Feather Express or Trinket MO can be used in MakeCode Maker.

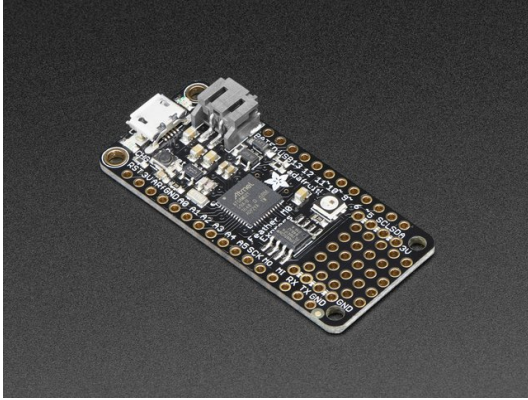
Tailor your needs to the boards available or have on hand.

Microcontrollers

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

\$24.95
IN STOCK

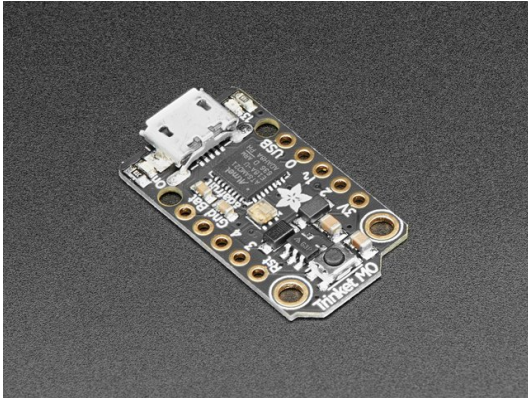
Add To Cart



Adafruit Feather M0 Express - Designed for CircuitPython

\$19.95
IN STOCK

Add To Cart



Adafruit Trinket M0 - for use with CircuitPython & Arduino IDE

\$8.95
IN STOCK

Add To Cart

Switches

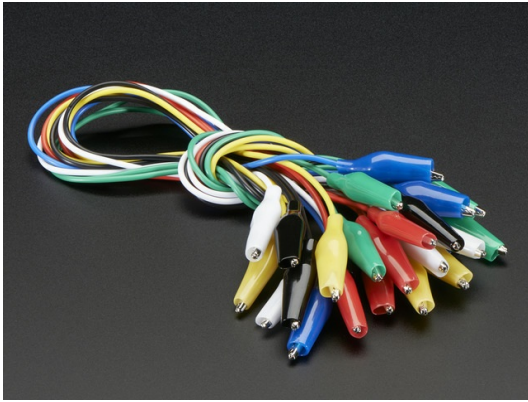
Adafruit carries a huge array of switches, including arcade buttons. You can use alligator clips or wires to connect them to your microcontroller. Here are just two of the dozens of products you can choose from on adafruit.com (<https://adafruit.com>).



Arcade Button - 30mm Translucent Clear

\$5.95
IN STOCK

Add To Cart



Small Alligator Clip Test Lead (set of 12)

\$3.95
IN STOCK

Add To Cart

USB Cable

Do get a known good USB cable. An extra long, bendy one like the purple one below work exceptionally well.



Pink and Purple Braided USB A to Micro B Cable - 2 meter long

\$3.95
IN STOCK

Add To Cart

Everything is a Keyboard

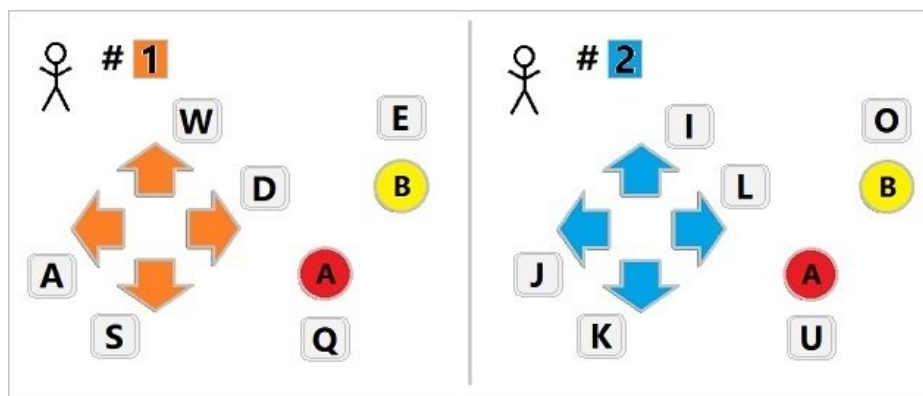
The browser-based Arcade game player supports a number of inputs like keyboard, mouse or gamepads. **If you can emulate a keyboard, you can build a controller!**

The idea to build custom controller by emulating a keyboard was pioneered by Makey Makey. In this guide, we will show how to use MakeCode to turn your beloved Adafruit boards, like Circuit Playground Express, into a keyboard and ultimately an awesome custom Arcade controller.

Inputs to your game can be:

- Buttons on the Circuit Playground Express
- Capacitive touch (metal, fruit, etc.)
- Any sort of switch, like arcade style buttons, big mashable buttons, etc.
- Sensor readings on the Circuit Playground Express like light, sound, movement, etc.

MakeCode Arcade supports 2 players on a keyboard using typical game key assignments.



The maker-controller extension

Microsoft built a MakeCode extension that exposes custom blocks and functions to turn any hardware into a custom controller. It takes care of emitting the keystrokes with the correct order and timing so that it really looks like a real keyboard.

The extension is hosted on GitHub at <https://github.com/microsoft/pxt-maker-controller> and we'll be showing how to use it the next pages.

Get your microcontroller and plug it into your computer via a known good data + power USB cable.

Circuit Playground Express

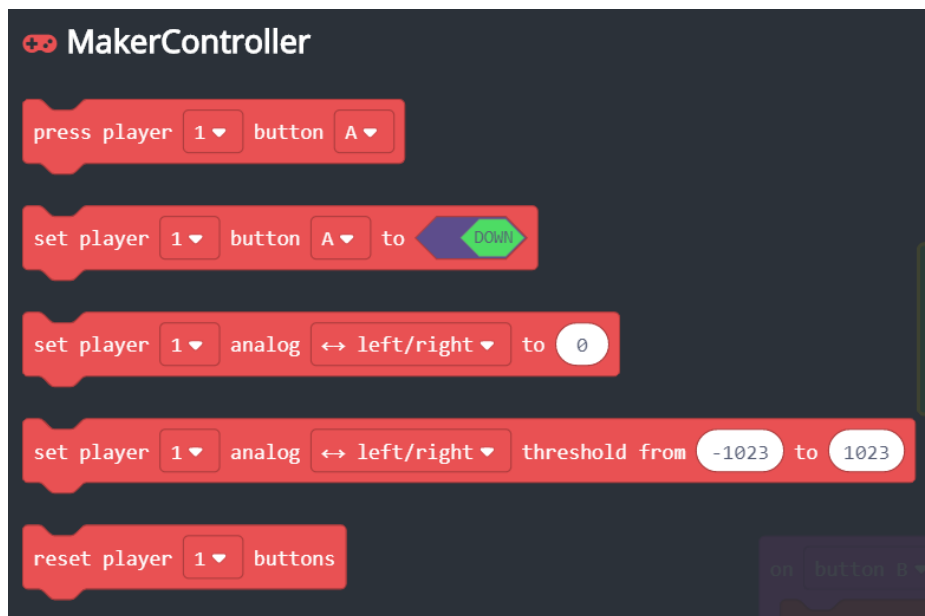
To use the maker-controller extension with the Circuit Playground Express,

- In your web browser, open <https://makecode.adafruit.com/beta>
- Click on **New Project**
- Click on **Extensions** under the gearwheel menu (upper right on the screen)
- Type or copy <https://github.com/microsoft/pxt-maker-controller> into the search box.

Maker (Feather Express, Metro Express, and other MakeCode Maker compatible boards)

To use the maker-controller extension with the MakeCode Maker,

- In your web browser, open <https://maker.makecode.com/>
- Click on **New Project**
- Click on **Extensions** under the gearwheel menu (upper right on the screen)
- Type or copy <https://github.com/microsoft/pxt-maker-controller> into the search box.



Remember... it's a keyboard!

Remember that when you'll plug your custom controller into your computer, it will start generating keystrokes regardless where the focus is. So make sure to leave any text editor before using it...

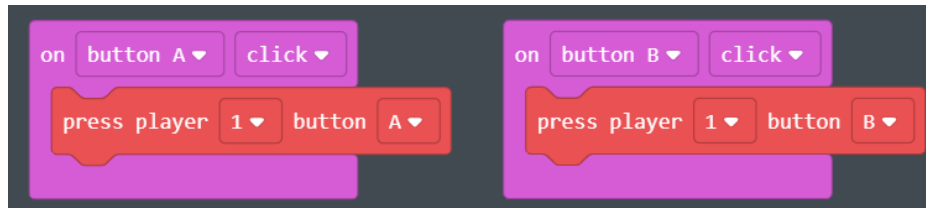
You can unplug your board from USB to stop it from sending characters if need be.

Button Presses

You can use the **press player .. button ..** block to simulate button presses in the game.

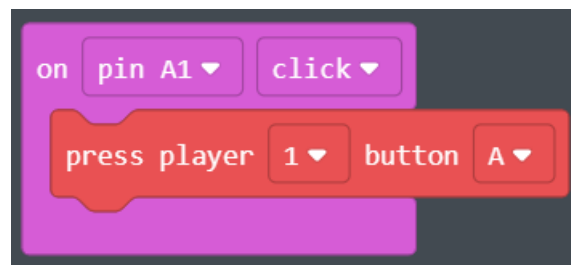
Button clicks

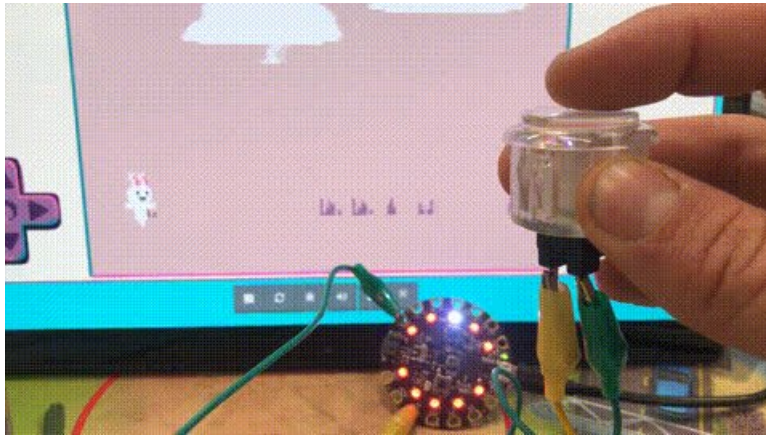
For example, the snippet below responds to button A/B on the CPX and converts them into A/B button keyboard keystrokes.



Buttons on Pins

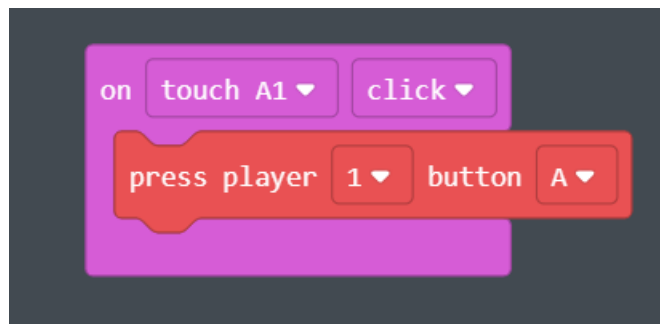
You can also detect clicks on pins by closing the circuit to the ground pins. For example, you can use alligator clips between the pins (a data pin and **GND**) and an Arcade button.





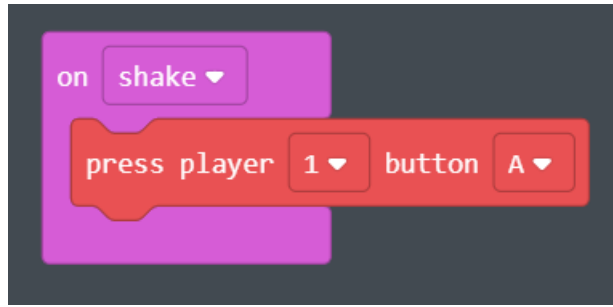
Capacitive touch

Get your apples and bananas! It's totally possible to use capacitive touch to build controllers.

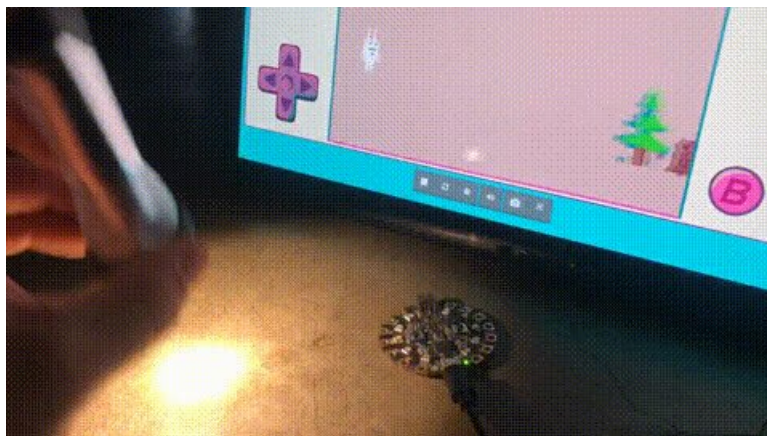


Shake and other events

Any event triggered by a sensor could be used to generate a keystroke. For example, shake can be used to make the bunny jump.



The light sensor is also a great sensor to build analog controllers.



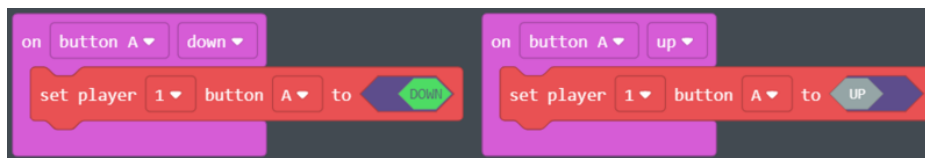
Ups and Downs

Some games use the up and down state of the buttons, so it can be more efficient to send those events instead of simple presses. For example, the duration of the button press can determine the strength at which the golfer hits the ball!

You can use **set player .. button .. to up/down** to set the button to up or down states.

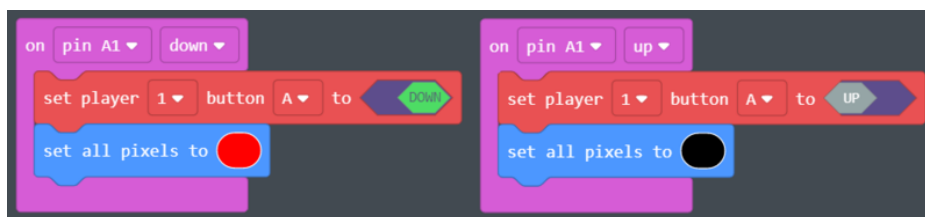
Buttons

Buttons, on board or using pins, have a **up** and **down** event that can be used to control the arcade buttons more precisely.



Pin Buttons

Close the circuit between **GND** and a pin to trigger down and up events. Real arcade buttons work great!



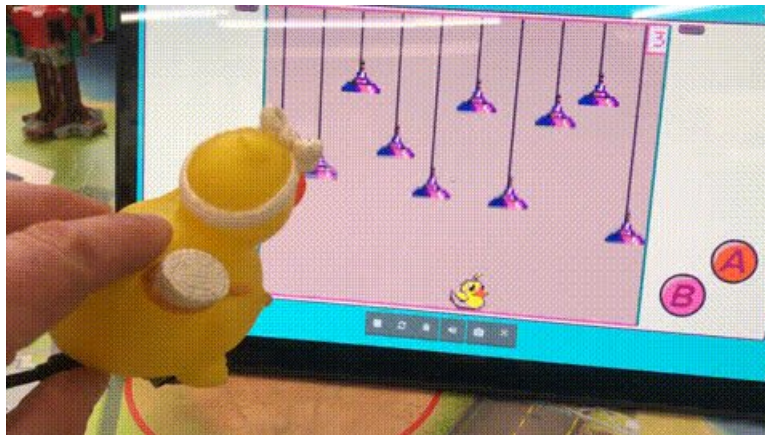


Analog DPAD

Use the **set player .. analog .. to value** block to convert an analog sensor value, like an accelerometer reading, into button ups and downs.

Left and right

We use the **set player .. analog ... to ..** block to convert the horizontal acceleration data into DPAD left/right buttons.



Reading Analog Values

For analog readings mapped to values in-game, you can use the **set player .. analog ..** blocks to change a value.

For example, if in-game, an analog reading controls an object, you can set what values you want in **aset player .. analog .. threshold** block. Then in game, the maximum and minimum values for an analog reading, like light level or potentiometer, map to the threshold values using a **set player .. analog .. to** block.

In the example below, the game responds to levels from 0 to 150 for a balloon on screen. The analog **light level** sensor block gives readings from 0 to 255. The analog blocks do all the mapping for getting the input the game is looking for.

