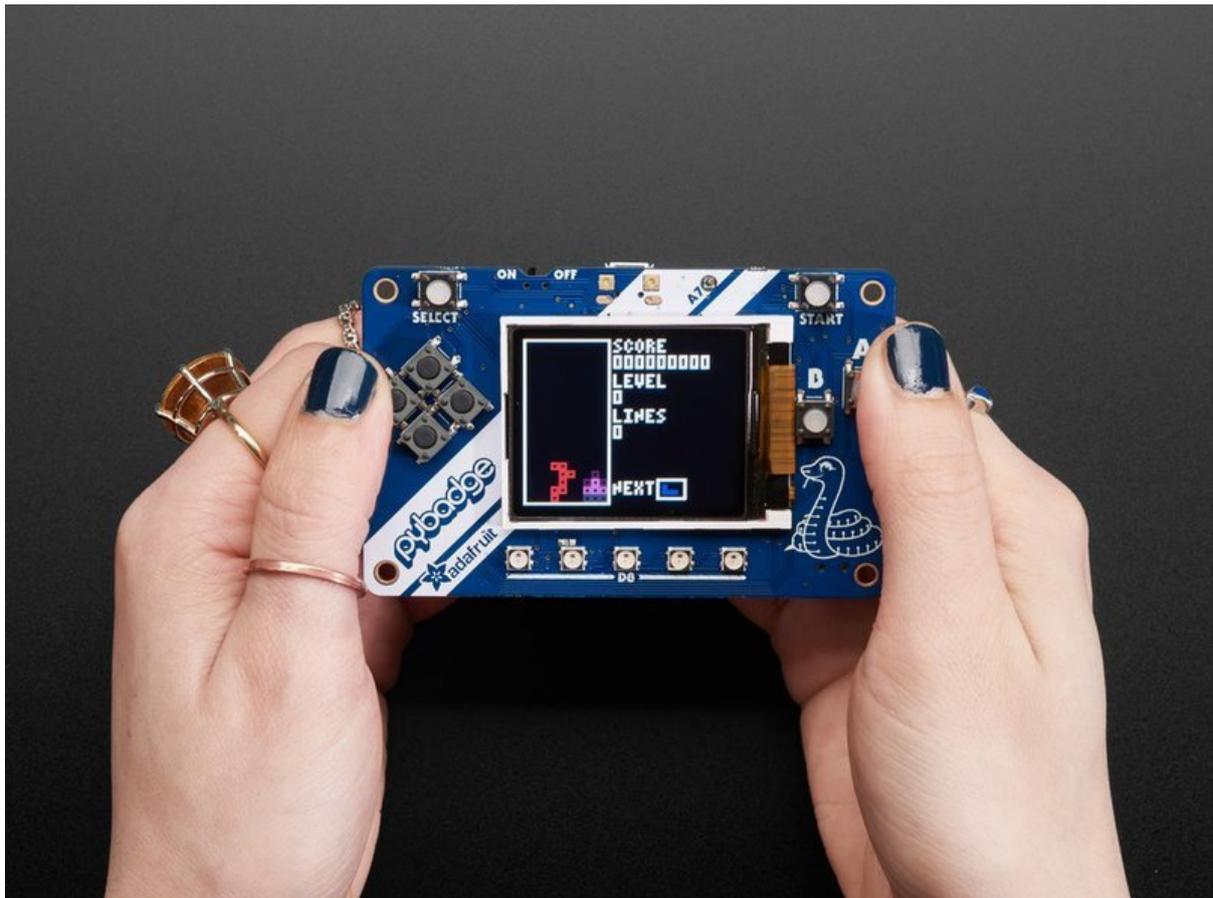




Playing Gamebuino META Games on Arcada

Created by lady ada



<https://learn.adafruit.com/playing-gamebuino-meta-games-on-arcada>

Last updated on 2025-12-02 03:17:05 PM EST

Table of Contents

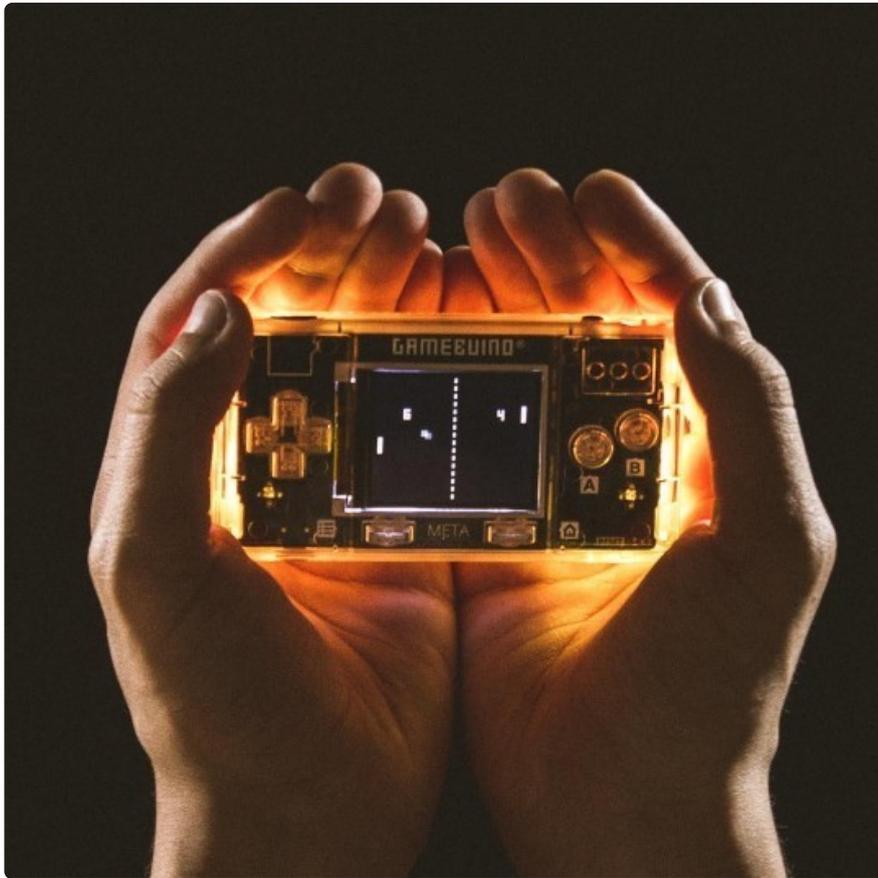
Overview	3
<ul style="list-style-type: none">• What can I use?	
Gamebuino Libraries	6
<ul style="list-style-type: none">• Step 1. Install Arduino IDE, drivers & board package• Step 2. Install Arcada Libraries• Step 3. Install Adafruit Version Gamebuino Library	
Gamebuino Test	7
<ul style="list-style-type: none">• Disk Drive Usage• Using the Filesystem for Games	
Some Examples	12
<ul style="list-style-type: none">• METAttris• META-Picomon	
Warnings & Notes	15
<ul style="list-style-type: none">• Yield• SD vs. QSPI Filesystem• Audio Format	

Overview



The Gamebuino META is a cool and colorful gaming handheld that features an Arduino-compatible ATSAM21, 160x128 color TFT screen, speaker, and 8 buttons. It's an amazing platform for writing small games and is a great looking handheld!

<https://adafru.it/EX3>



Since Arcada boards, like the PyGamer and PyBadge's are also Arduino compatible, it's easy to compile any Gamebuino games for these boards. The PyGamer/PyBadge have much faster processors with more memory and the same size screen (in fact, it's the same exact screen). We will not be 'emulating' the Gamebuino games - we really will just re-compile them.

The buttons, sound, display all work the same. A huge amount of the hardware interfacing code is all from Adafruit already, so adapting the Gamebuino library from the SAMD21 to SAMD51 was not too difficult!

You can use the SD card on the PyGamer if you like but we will prefer the internal QSPI memory since it exists on all Arcada boards.

The program-from-SD card bootloader wasn't ported, we'll be sticking to the UF2 bootloader we know and love.

chip on the Gamebuino META (ATSAMD21) and Arcada boards are compatible so we cannot use the 'bootload-from-SD' distribution system for Gamebuino. Each game must be compiled a-new. However, once it is compiled, you can share the UF2 for your particular board

What can I use?

We use the Arcade Arduino library to interface with the hardware, and can run on:



[Adafruit PyGamer Starter Kit](https://www.adafruit.com/product/4277)

Please note: you may get a royal blue or purple case with your starter kit (they're both lovely colors) What fits in your pocket, is fully Open...

<https://www.adafruit.com/product/4277>



[Adafruit PyGamer for MakeCode Arcade, CircuitPython or Arduino](https://www.adafruit.com/product/4242)

What fits in your pocket, is fully Open Source, and can run CircuitPython, MakeCode Arcade or Arduino games you write yourself? That's right, it's the Adafruit...

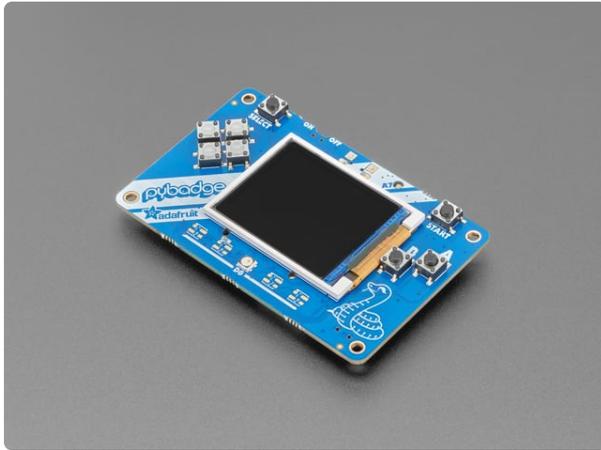
<https://www.adafruit.com/product/4242>



[Adafruit PyBadge for MakeCode Arcade, CircuitPython, or Arduino](https://www.adafruit.com/product/4200)

What's the size of a credit card and can run CircuitPython, MakeCode Arcade or Arduino? That's right, its the Adafruit PyBadge! We wanted to see how much we...

<https://www.adafruit.com/product/4200>



Adafruit PyBadge LC - MakeCode Arcade, CircuitPython, or Arduino

What's the size of a credit card and can run CircuitPython, MakeCode Arcade or Arduino even when you're on a budget? That's right, it's the Adafruit...

<https://www.adafruit.com/product/3939>



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>

Gamebuino Libraries

Step 1. Install Arduino IDE, drivers & board package

Before you start, you'll need to follow the guide for your particular board to install the Arduino IDE, drivers and board support package so that you can upload code to it via Arduino

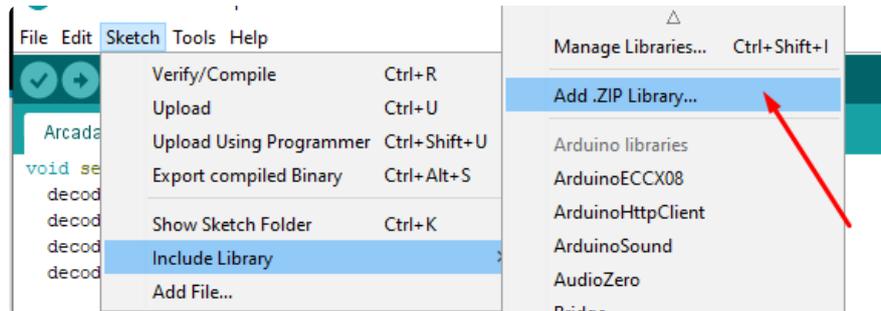
Step 2. Install Arcada Libraries

We use quite a number of libraries to access the display, sound, buttons, etc.

[Visit the Arcada library page to install all the libraries listed \(yes that's a dozen!\)](https://adafru.it/EUK) (<https://adafru.it/EUK>)

Step 3. Install Adafruit Version Gamebuino Library

In addition to the Arcada libraries, you'll need the Adafruit-tweaked versions of the Gamebuino library. We will install it manually, not through the library manager, so download the Zip and then select **Add .ZIP Library...** to install

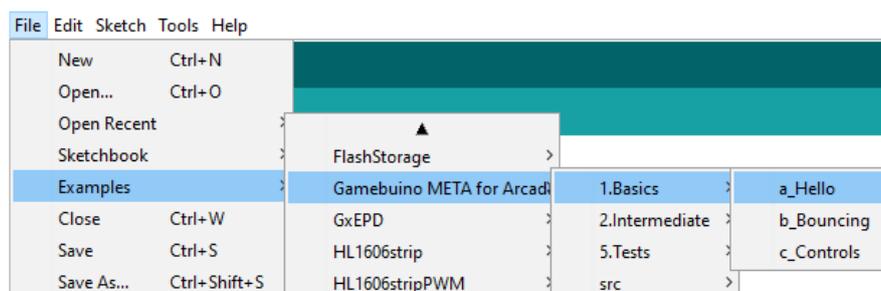


These replace the existing library but adds support for Arcada boards, for that reason, you should remove any existing library you have with this name!

<https://adafru.it/EX4>

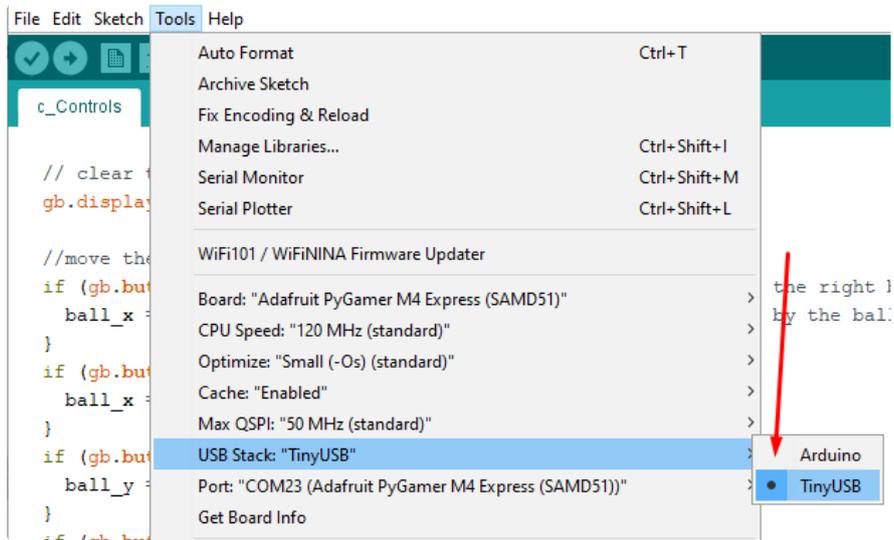
Gamebuino Test

OK, let's test out that this worked. Restart the Arduino IDE and select the board you'll be uploading to, then open the **Gamebuino META for Arcada** examples. Start with **a_hello**

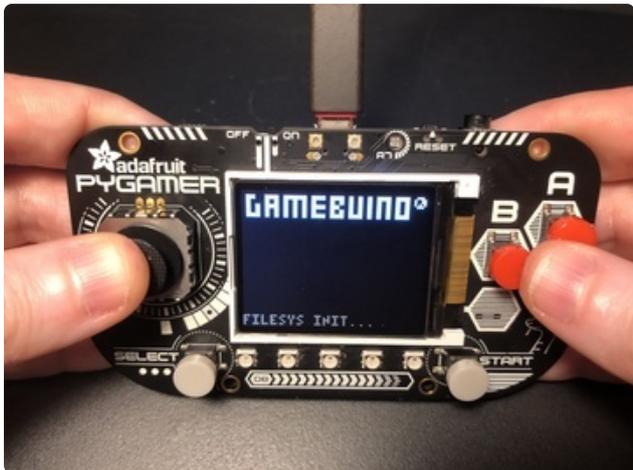


Be sure to select your Arcada board. You don't need to overclock or optimize so leave those settings alone.

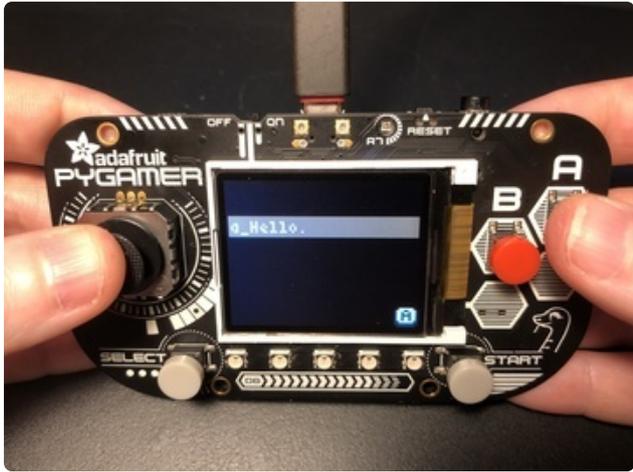
You must select TinyUSB as the USB Core - that's what we use to have the internal storage appear as a disk drive. It's not the default so be sure to re-select it whenever you open the Arduino IDE



Compile and upload a_Hello to your board. You will see the starting animation and GAMEBUINO text...



You will see the starting animation, LEDs flashing, and GAMEBUINO text...

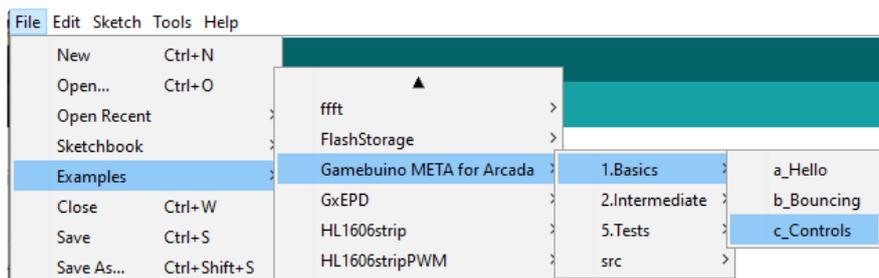


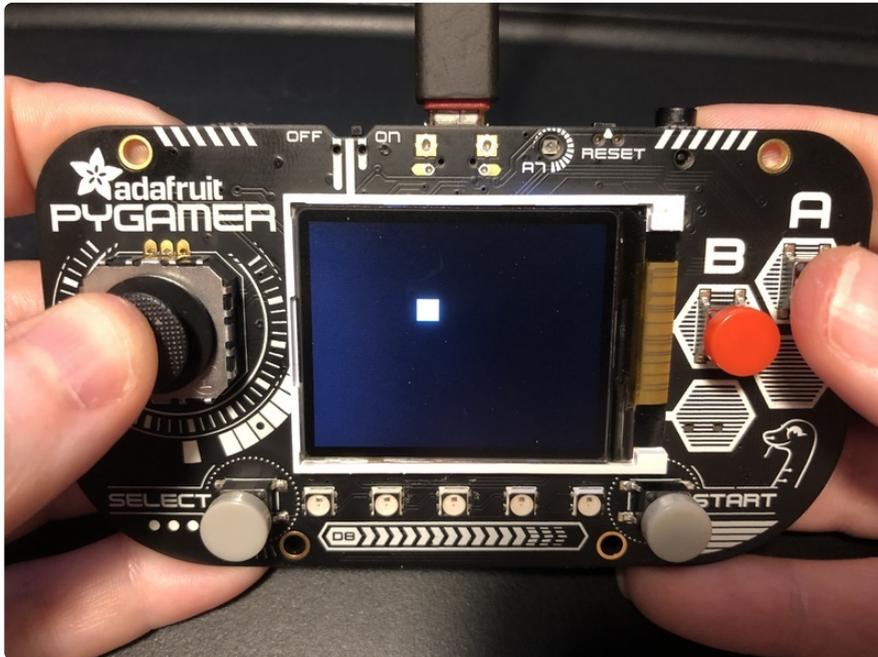
Then the 'splash screen' for the game, in this case it just shows the name and indicates for you to press the **A** button to continue



Finally, the Hello World text and a counter will appear. That's it!

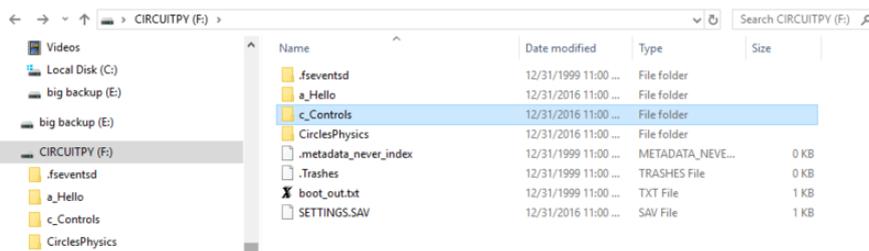
The next one we suggest you try is `c_Controls` because you can test the D-Pad (move the ball) and buttons (make sounds)





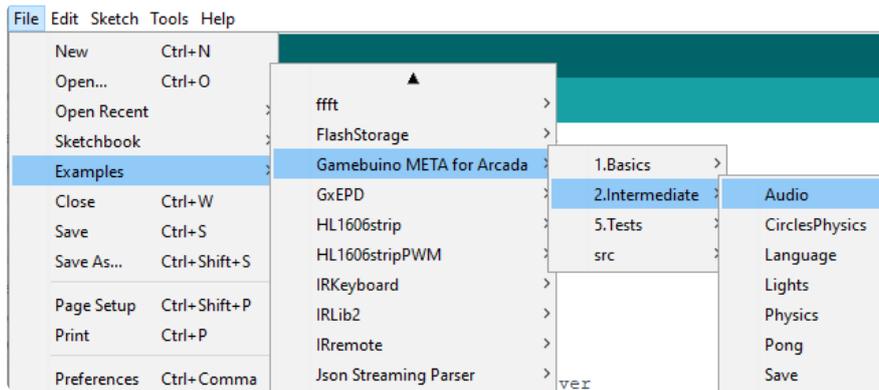
Disk Drive Usage

You'll note as you run these tests that a **CIRCUITPY** drive will appear on your computer. This disk drive has the contents of the filesystem (QSPI or SD Card) that you can browse and manage with your computer. Folders appear on the disk drive for each project/game you compile & upload, and will contain files or settings for that game.



Using the Filesystem for Games

Some games will want you to store files on the filesystem. Normally this is an SD card only, but we can use the QSPI storage as the USB disk drive. To demonstrate, load up the **Audio** demo

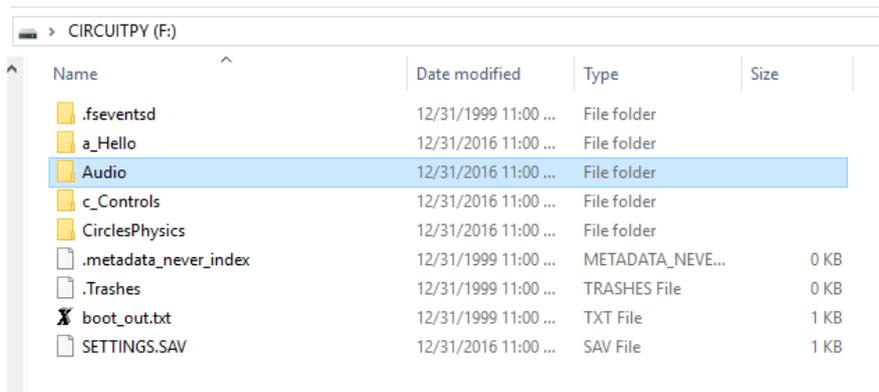


You'll notice that you can press **B** to play the bleep effect but **A** doesn't work yet.

In the code **A** should trigger this line:

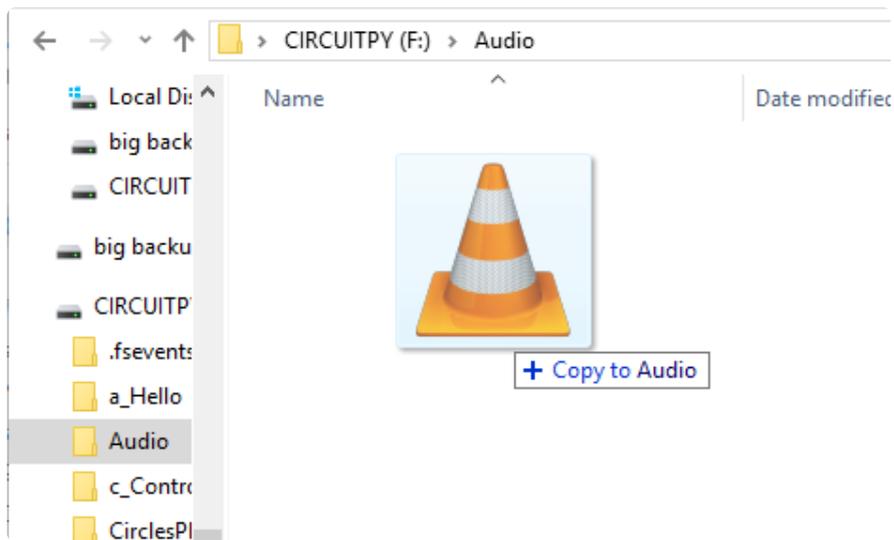
```
music = gb.sound.play("test.wav", true); // true for infinite looping
```

In this case its looking on the filesystem. Press reset to restart the sketch, the next time the USB drive appears you'll see a new folder called **Audio**



Download and save this **test.wav** file into the folder

<https://adafru.it/EX6>



Once it finishes writing, wait a moment, then try pressing **A** on the sketch to play the audio!

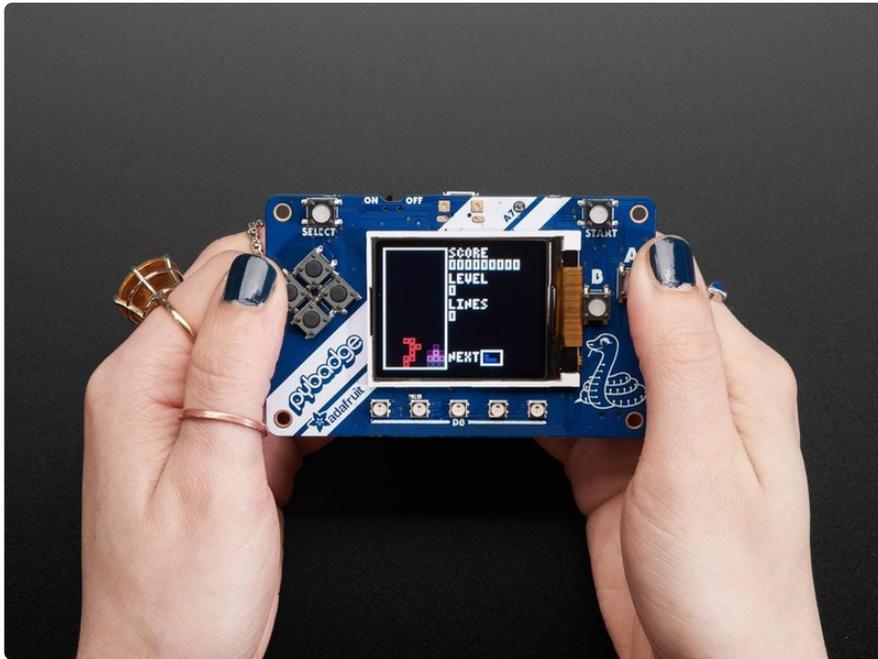
Some Examples

There's over 100 games available, and we don't want to play any favorites! We did try a couple games that are good demonstrations of the Gamebuino platform. Here are ready-to-go UF2s you can drag-n-drop onto the **BOOT** bootloaders of a PyGamer or PyBadge

METAtris

[This Tetris-like has awesome graphics and sound, incredibly addicting! \(https://adafru.it/EX7\)](https://adafru.it/EX7)

[The source code is on github, you can download/fork it from there \(https://adafru.it/EX8\)](https://adafru.it/EX8)



<https://adafru.it/EX9>

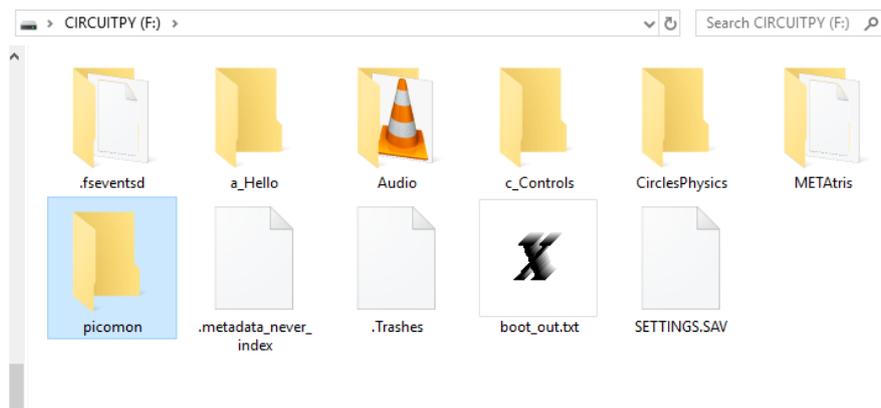
<https://adafru.it/EXa>

META-Picomon

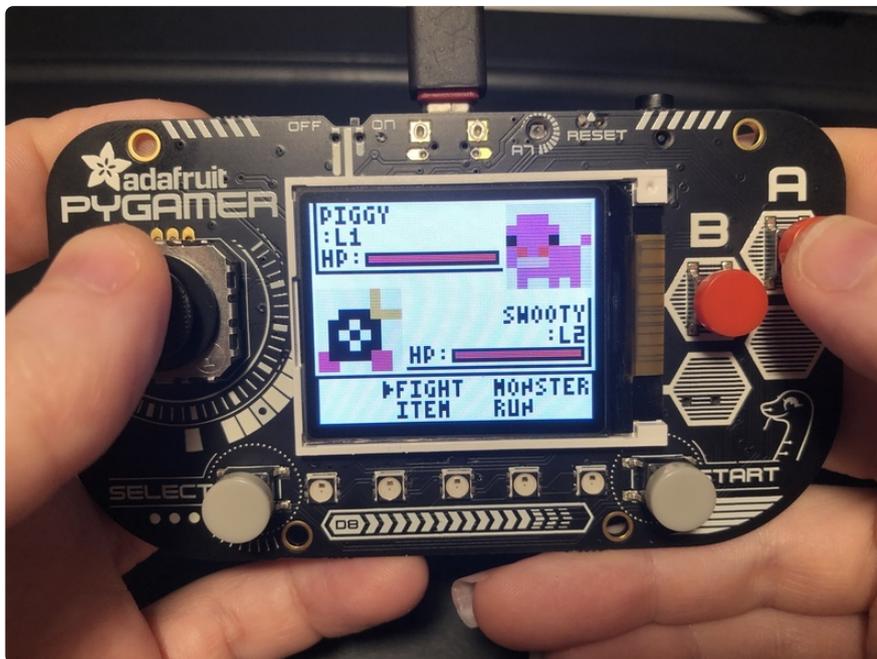
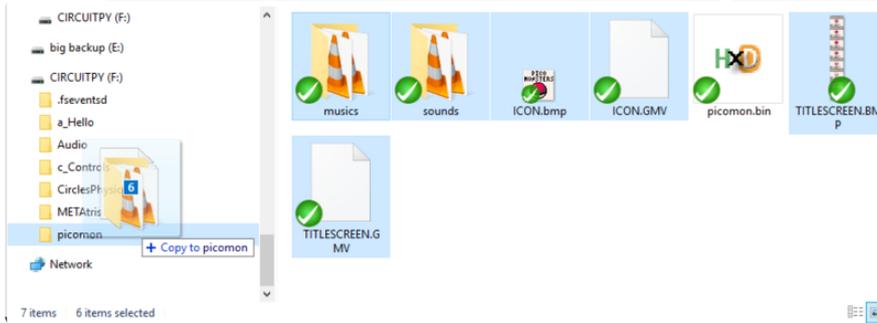
This Pokémon alike is a port from a pico8 game, thus the name

[Source code is on GitHub you can download/fork it here \(https://adafru.it/EXb\)](https://adafru.it/EXb)

This game comes with audio files, so after you run the game once, reset the Arcade board, and find the **picomon** folder on the USB drive that appears:



Then, from the downloaded source code, go into the **binaries/picomon** folder. You'll find a **musics** folder, **ICON.GMV** file and more. Drag all but the picomon.bin file into the **CIRCUITPY/picomon** folder



<https://adafru.it/EXc>

<https://adafru.it/EXd>

Warnings & Notes

Yield

The Gamebuino library defines a function called `yield()` but this is internally used by Arduino for USB and other RTOS tasks, for Gamebuino's `yield` we've renamed it to `gb_yield()`

SD vs. QSPI Filesystem

By default Arcada sets up to use the internal QSPI filesystem. If you want to use SD cards instead, and your board has an SD card slot, open up the `Adafruit_Arcada.h` and find the definition block for your board, for example

```
#elif defined(ADAFRUIT_PYGAMER_M4_EXPRESS)
```

Then change

```
#define ARCADA_USE_QSPI_FS  
//#define ARCADA_USE_SD_FS
```

to

```
//#define ARCADA_USE_QSPI_FS  
#define ARCADA_USE_SD_FS
```

And recompile your game

Audio Format

Audio WAV files must be in **Mono, 8-bit, 44.KHz** which is a little different than what we normally use in Adafruit projects. [Still, our audio-conversion guide may help! \(https://adafru.it/BvU\)](https://adafru.it/BvU)