

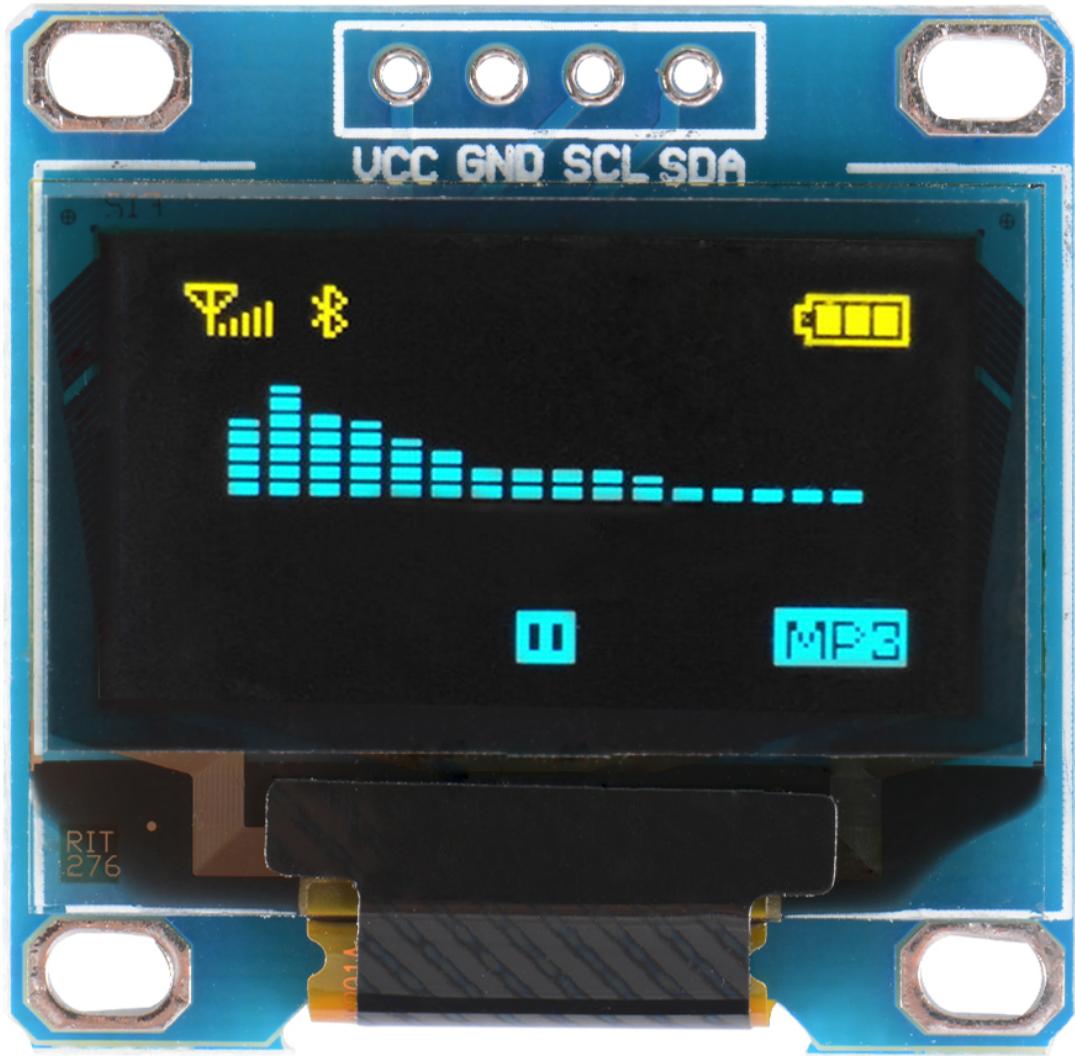
DIY Arduboy

TINKERCAD

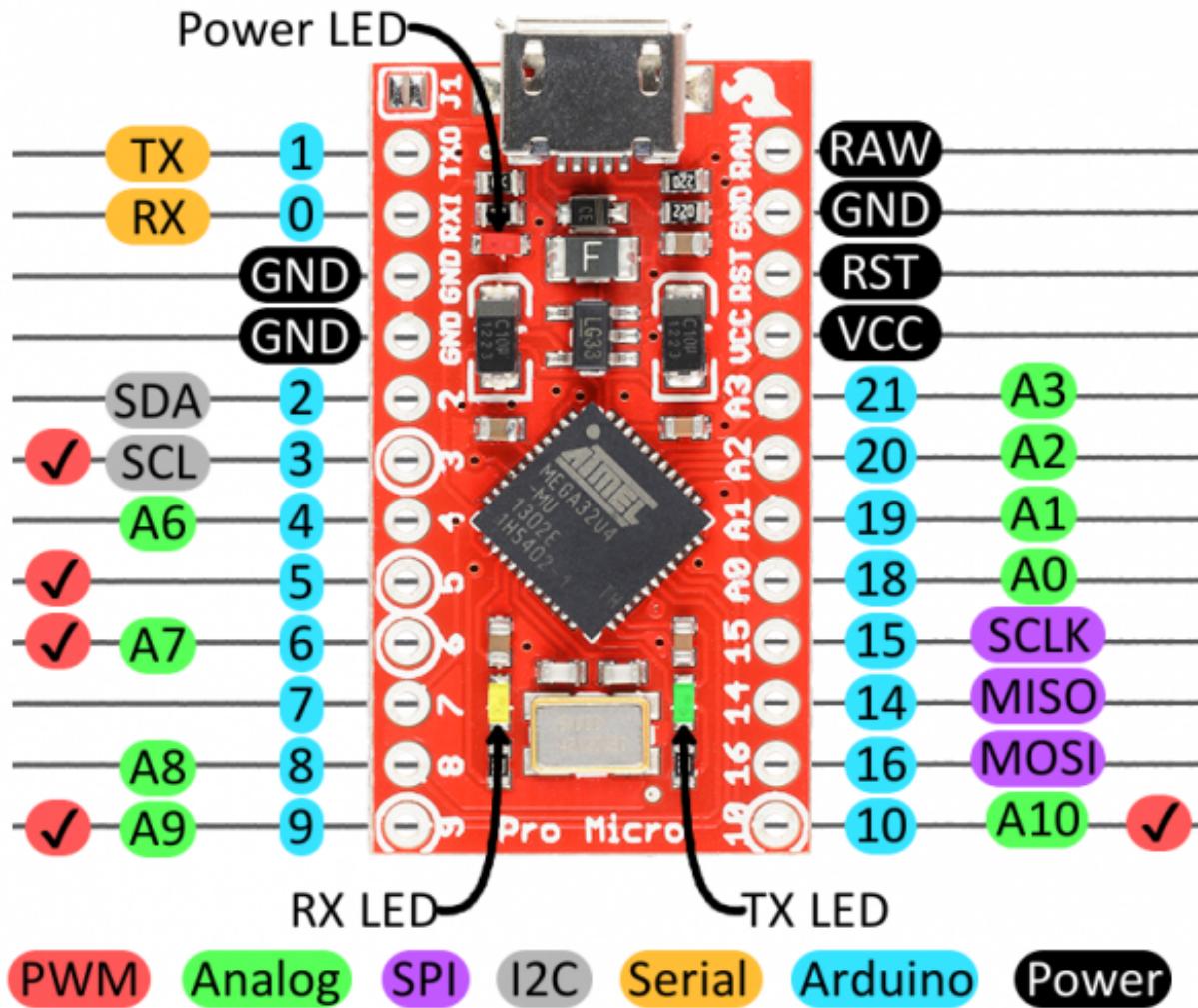
(sample: <https://www.tinkercad.com/things/6nSdspNbZXi-arduino-lcd-game#/>)
Tinker Circuits

[Arduboy Homemade - Tinker Circuits](#)

OLED 0.96 I2C - The Pinout



Sparkfun Pro Micro - The Pinout



Pro Micro & Fio V3 Hookup Guide

Learn how to setup IDE to work with Sparkfun Pro Micro board and run some examples

Arduino Micro just blinks, won't connect to PC

(bonus:

- The case of the broken Pro Micro
- The case of the broken Pro Micro (reprise)

)

OLED I2c Display With Arduino and, OLED I2C Display Arduino Tutorial

Notable: [i2c_scanner](#)

Arduboy on Arduino Nano + I2C-Display Using a Nano with a I2C oled

Modified

Code: <https://github.com/harbaum/Arduboy2>

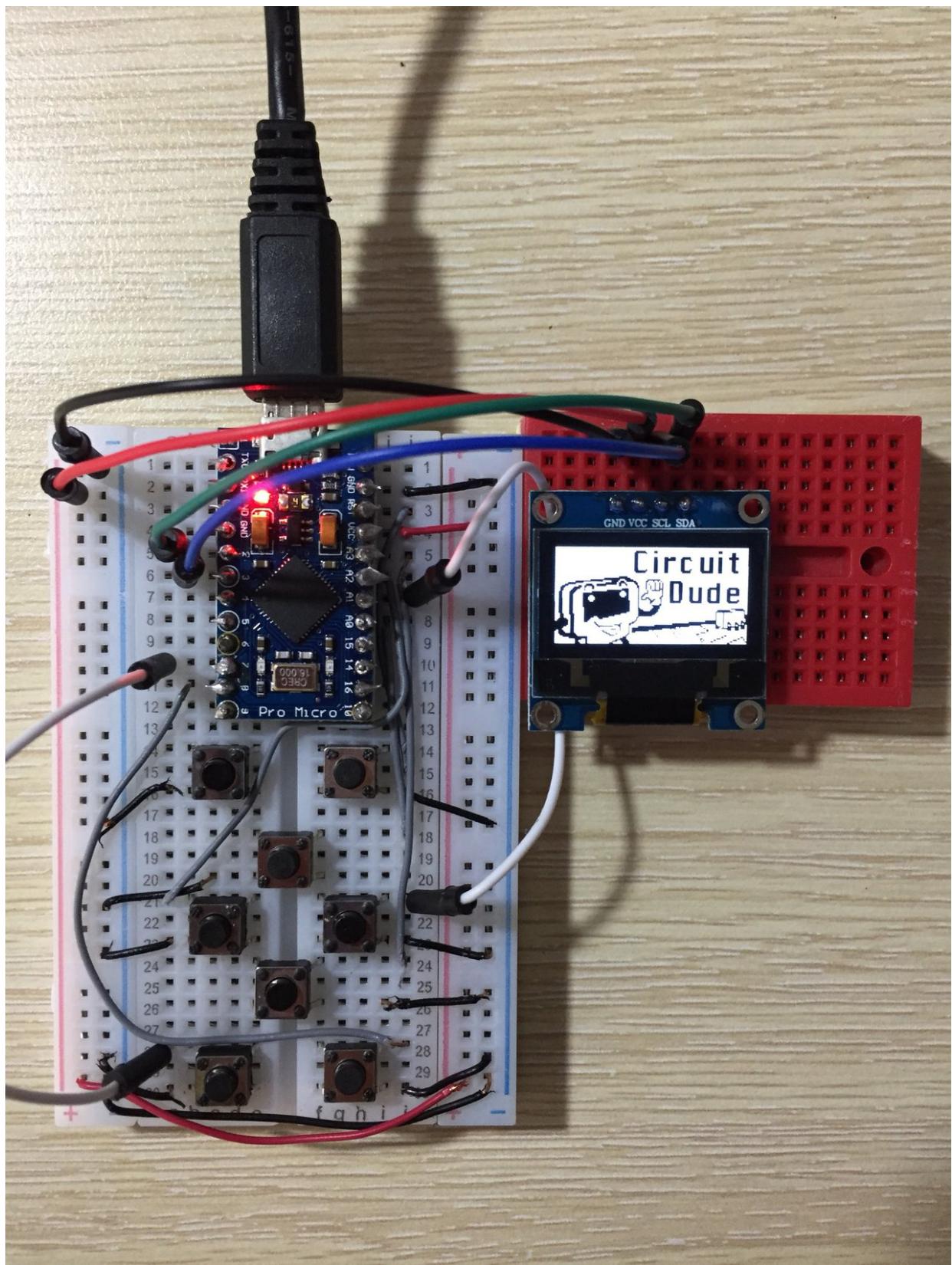
My own version

Arduboy Homemade - Pro Micro + I2C OLED

<https://github.com/huyduino/Arduboy2>

(actually I don't need to make these modification if I buy another display which supports SPI interface, instead of I2C.
however, it's fun doing these)

Final product



Video: <https://youtu.be/OHe54QShM9c>

Refs

Arduboy → Homemade

- \$12 Arduboy compatible system
- Breadboy a 100% non soldering Homemade Arduboy
- Scootmatt's Arduboy Clone
- Hardware clone (esp8266 + arduino nano)
- Where can I get a dev kit?
- Homemade Arduboy with acrylic glass case and flash cart
- PS2 Arduboy based on ESP8266

others

- <https://www.youtube.com/watch?v=x3wqad1CnH0>
 - DIY Video Game Using Arduino (Arduboy Clone)
-
-
-

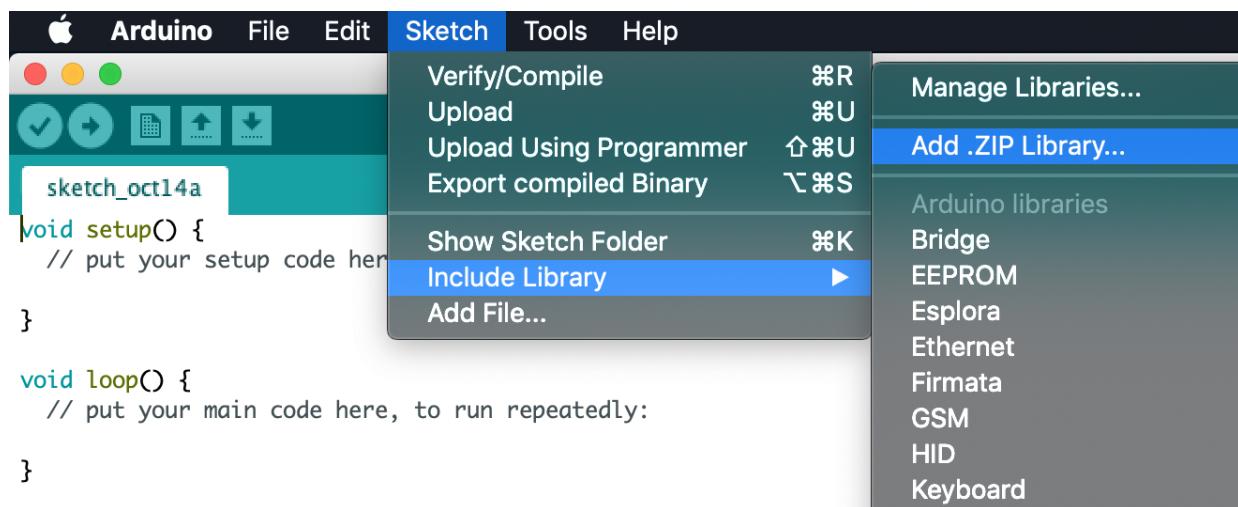
Deployment

Preparation

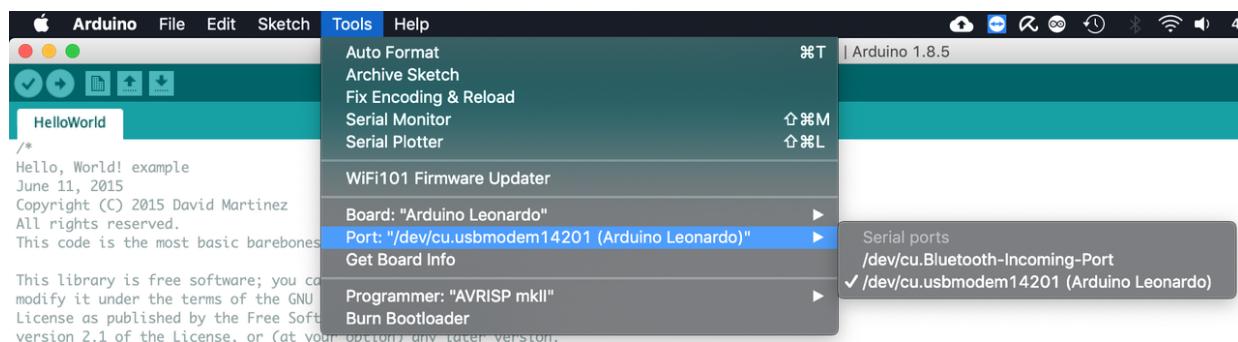
Before you can upload the game to your Arduboy (or homemade version in my case), you need to setup working environment. Follow this tutorial [Learning to Program the Arduboy](#) with a minor modification at **step 3**.

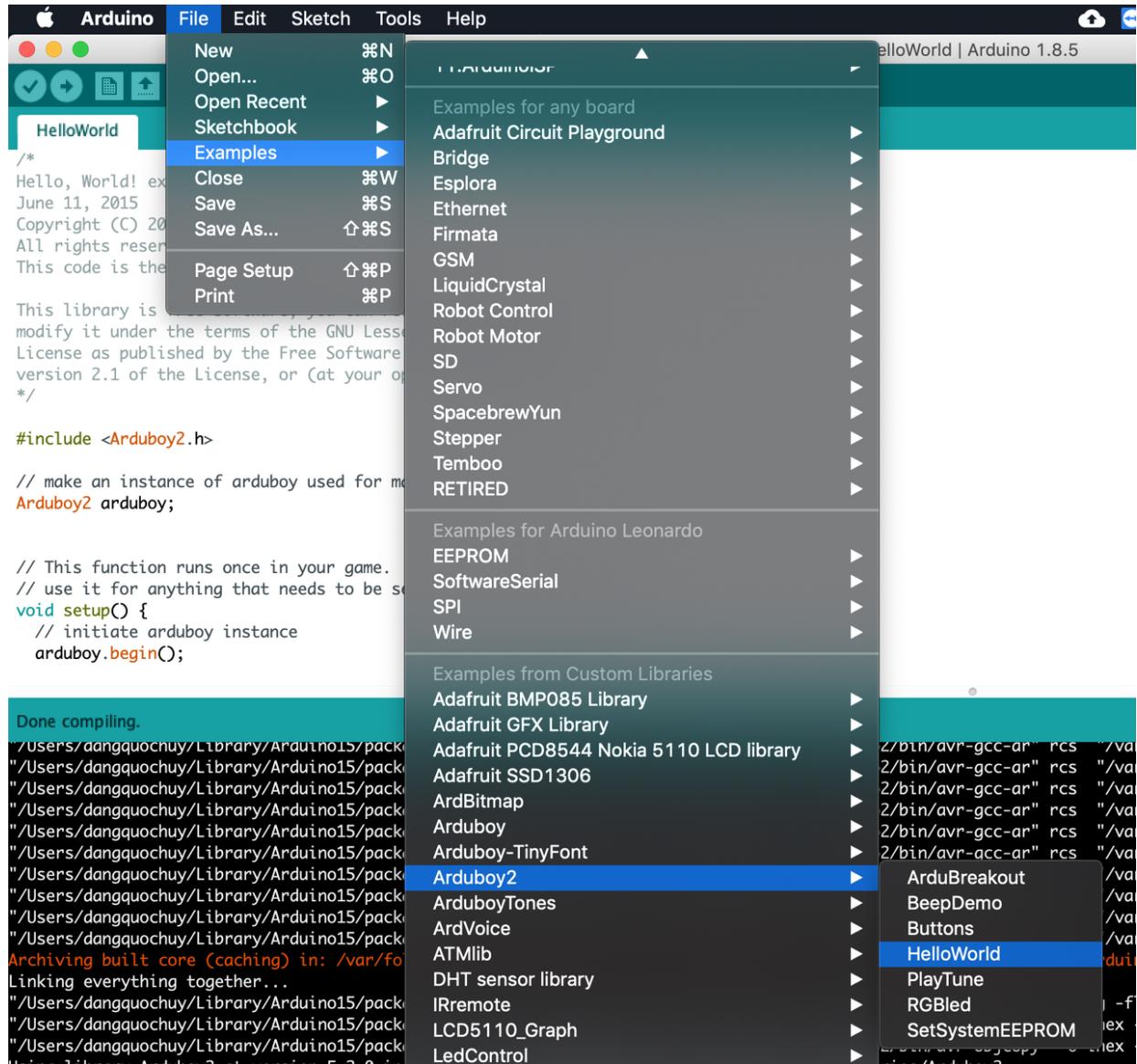
Don't install **Arduboy2** library from Library Manager, but

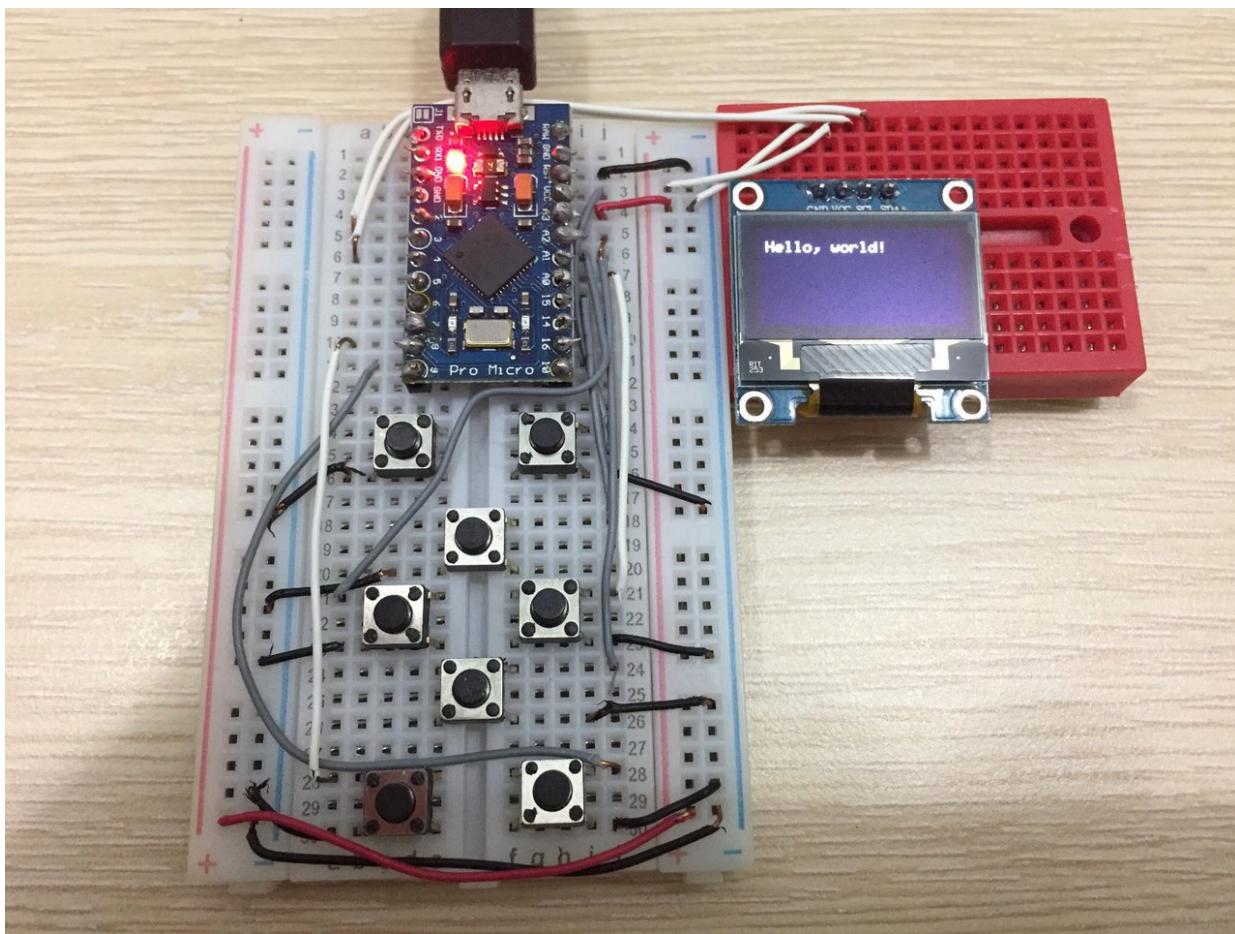
- Download the customized lib for this DIY Arduboy
at <https://github.com/huyduino/Arduboy2/archive/master.zip> (not sure is it needed, but recommend to unzip `Arduboy2-master.zip`, then rename the unzipped folder from `Arduboy2-master` to `Arduboy2`, then zip this folder to `Arduboy2.zip`. We will use this `Arduboy2.zip` file in the right below step).
- Select Add .ZIP Library... menu, point to zip file to import.



If your development environment is all set, you should be able to load the Hello World example as mentioned at step 5 in the tutorial. Below is mine.







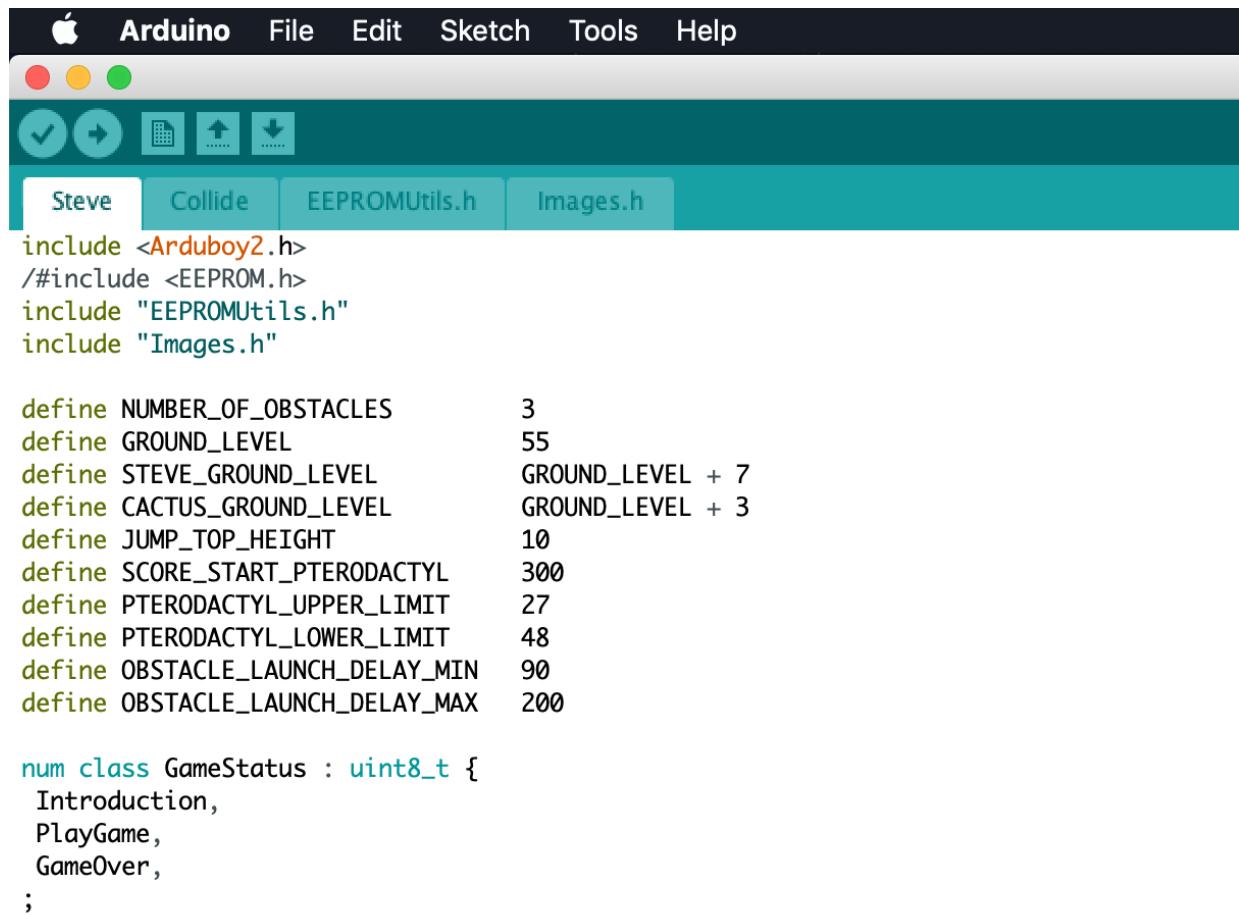
Upload game

(As easy as upload example sketch to the board above)

Choose the game you want to upload, in this example i will upload Steve - the Running Dinosaur for Arduboy
Download source code at <https://github.com/filmote/Steve/archive/master.zip>

Unzip and rename folder from Steve-master to Steve

Go into folder Steve and double click on Steve.ino the project will be opened in Arduino IDE as below



```

Arduino  File  Edit  Sketch  Tools  Help
Steve  Collide  EEPROMUtils.h  Images.h

include <Arduboy2.h>
#include <EEPROM.h>
#include "EEPROMUtils.h"
#include "Images.h"

define NUMBER_OF_OBSTACLES      3
define GROUND_LEVEL            55
define STEVE_GROUND_LEVEL      GROUND_LEVEL + 7
define CACTUS_GROUND_LEVEL     GROUND_LEVEL + 3
define JUMP_TOP_HEIGHT         10
define SCORE_START_PTERODACTYL 300
define PTERODACTYL_UPPER_LIMIT 27
define PTERODACTYL_LOWER_LIMIT 48
define OBSTACLE_LAUNCH_DELAY_MIN 90
define OBSTACLE_LAUNCH_DELAY_MAX 200

num class GameStatus : uint8_t {
    Introduction,
    PlayGame,
    GameOver,
};

;

```

Select board, port then compile and upload the game to the board. That's it, now it's time to play game!!!

