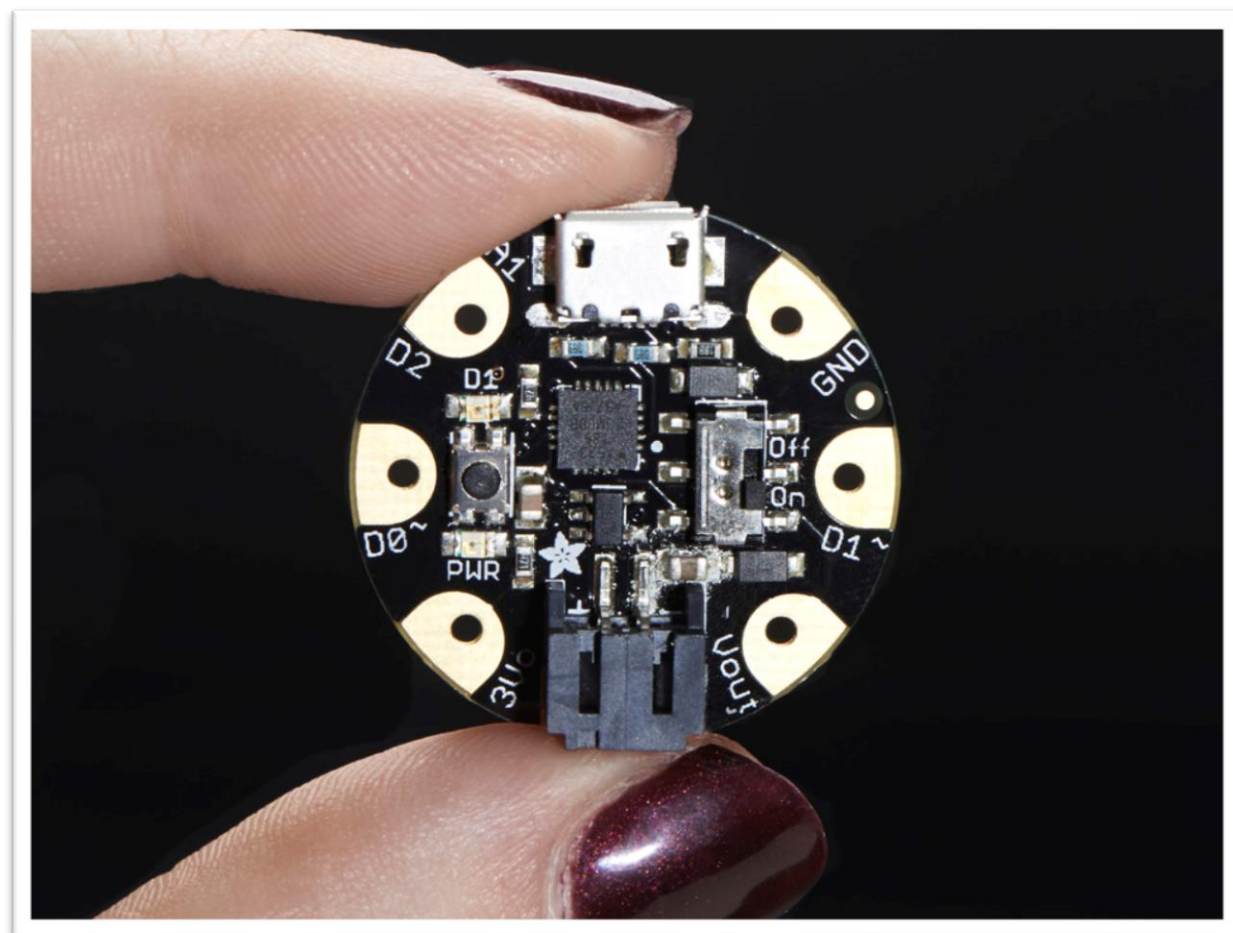


# GEMMA CONFIGURATION AND USE GUIDE



extia

2018

# Table of contents

I. Presentation .....	3
II. Warnings.....	4
III. Configuration.....	5
1. Drivers' installation (Windows only).....	5
2. Arduino IDE Setup.....	5
IV. Use .....	8

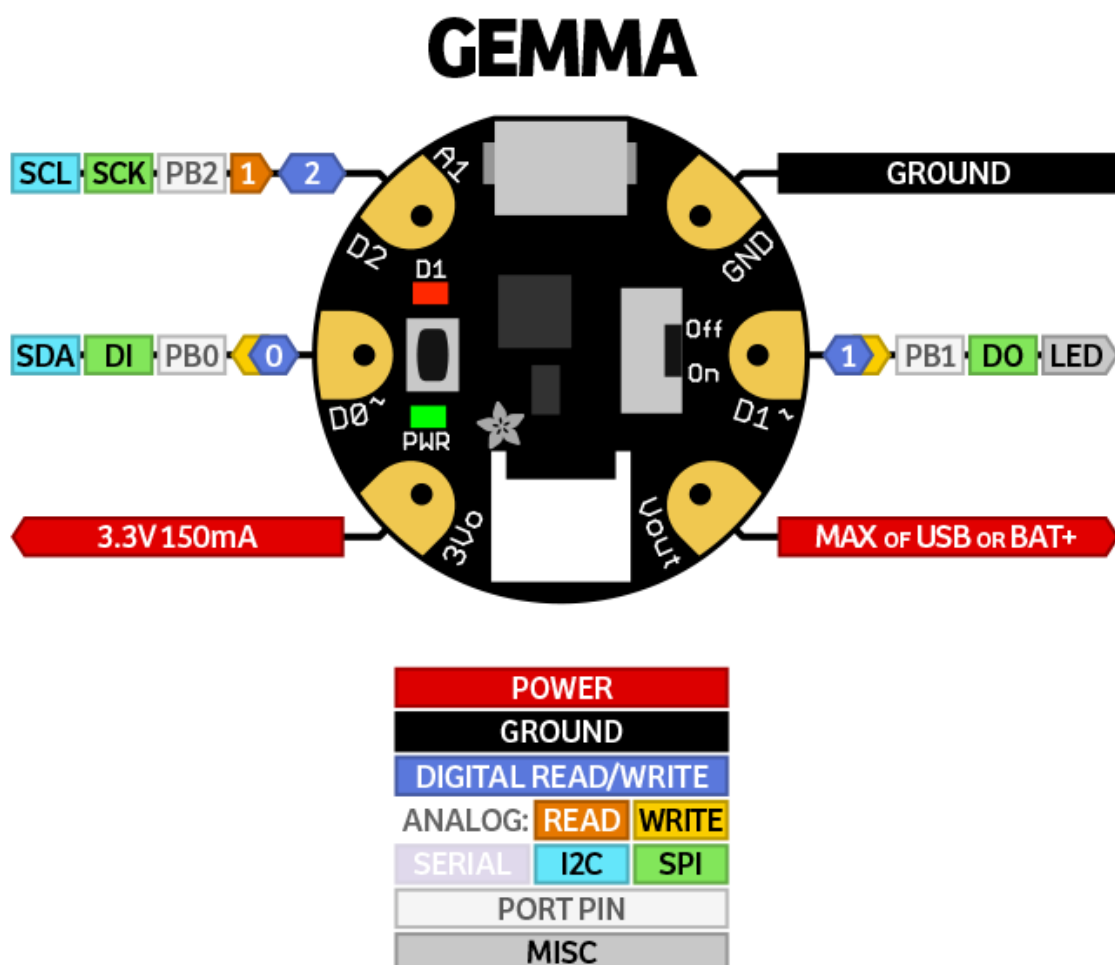
## I. Presentation

GEMMA is a microcontroller from **Adafruit** powered by an Attiny85 and programmable with an Arduino IDE over USB.

Among its main features, it includes 8K (5.25k available because 2.75 are occupied by the bootloader) of flash, 512 bytes of SRAM, 512 bytes of EEPROM, and an internal oscillator running at 8 MHz.

It is ultra low power, drawing only 9 mA while running and has an on-board 3.3V power regulator with 150mA output capability.

Finally, it includes 3 GPIO - The 3 independent IO pins have 1 analog input and 2 PWM output as well.



## II. Warnings

Even though you can program GEMMA using the Arduino IDE, it's not a fully 100% Arduino-compatible and suffers from some restrictions :

- GEMMA **does not have a Serial port connection** for debugging, so the serial port monitor will not be able to send/receive data (*nonetheless, you can overpass this issue using the **SoftwareSerial** library that emulates the Serial Ports over two available IO pins, making them unavailable for other applications*)
- Some computers' **USB v3 ports don't recognize** the GEMMA's bootloader. Simply use a USB v2 port or a USB hub in between
- GEMMA **is not guaranteed supported on Linux** operating system at this time - try Mac OS or Windows ! (*However, you can try the following - it does work for some computers : Linux is fairly picky about who can poke and prod at the USB port. You can always run avrdude or Arduino IDE as root, which will make sure you have the proper permissions. If you want to be super-cool you can add a udev rule which will let any user (who is not root) connect to the USBtiny driver. That way you don't have to be root all the time !*)

### III. Configuration

#### 1. Drivers' installation (Windows only)

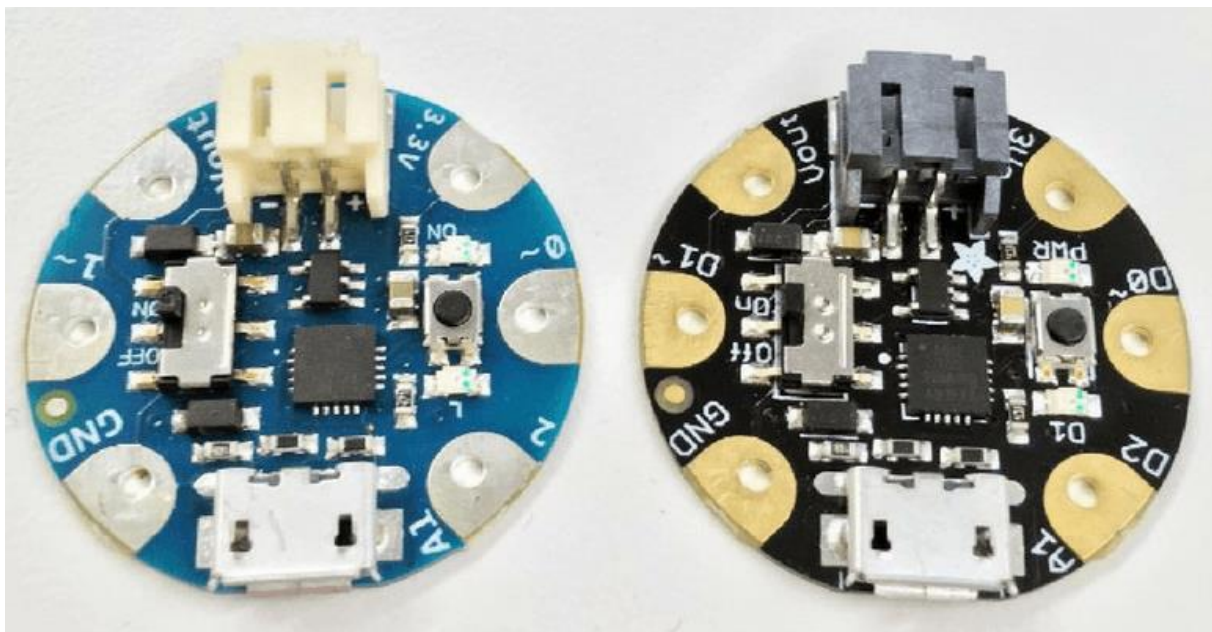
If you are using **Windows**, you may first download and install the necessary drivers needed for its use (Mac OS users can avoid this part)

They are available on [Adafruit's official website](https://www.adafruit.com/drivers).

During the installation, you need to agree the terms of uses and select which drivers you want to install, we suggest selecting all of them so you don't have to do this again !

#### 2. Arduino IDE Setup

Depending on the GEMMA board you own, there are different selections to apply.



Arduino GEMMA

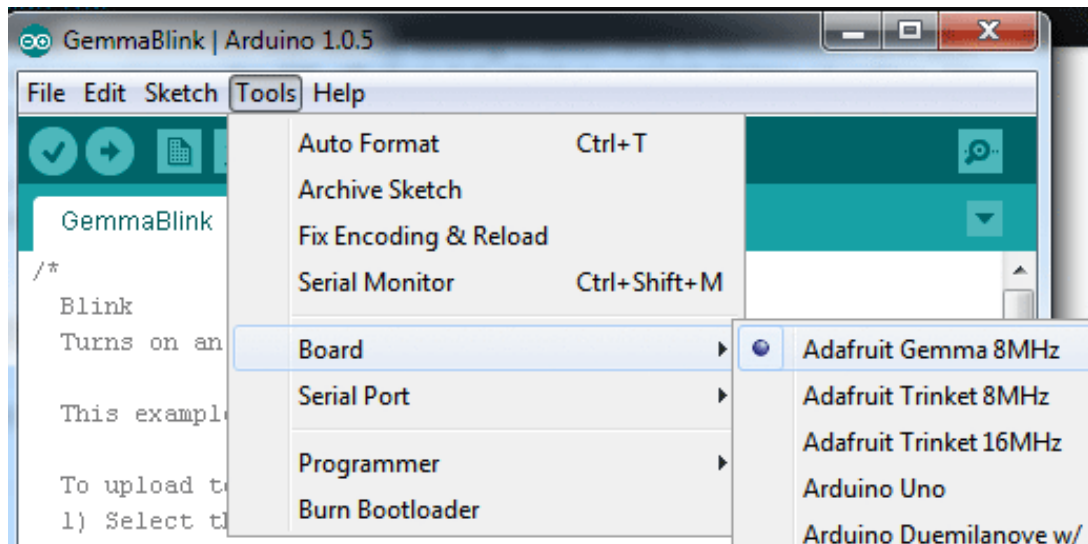
Adafruit GEMMA

Make sure to pick the right version of the Gemma board in the IDE ! If you're using the Adafruit Gemma it won't program correctly when selecting the Arduino Gemma

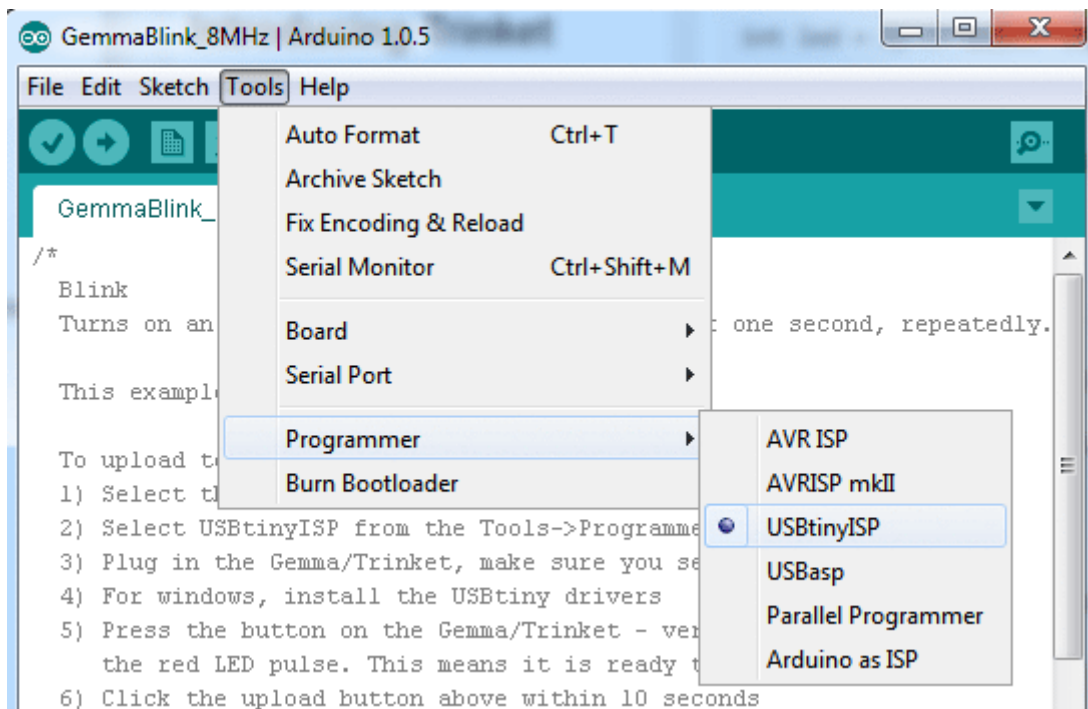
option and vice versa ! Carefully read the instructions below to pick the right board.

### Adafruit Gemma (Black GEMMA)

If you're using the Adafruit Gemma (the one with a black PCB board) select the Adafruit Gemma 8MHz board from the Tools->Board menu.

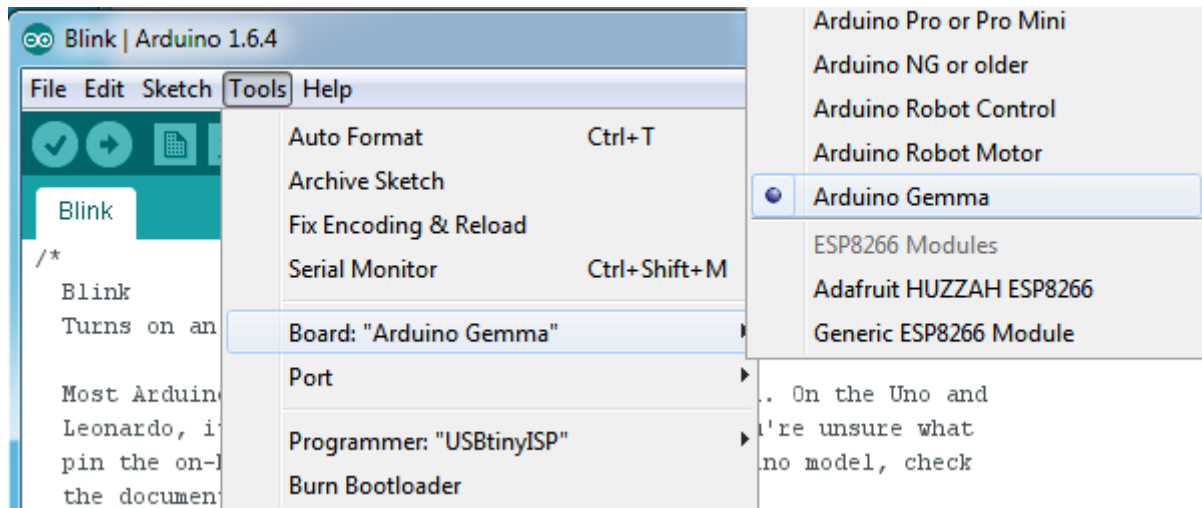


Then, select USBtinyISP from the Tools->Programmer sub-menu.

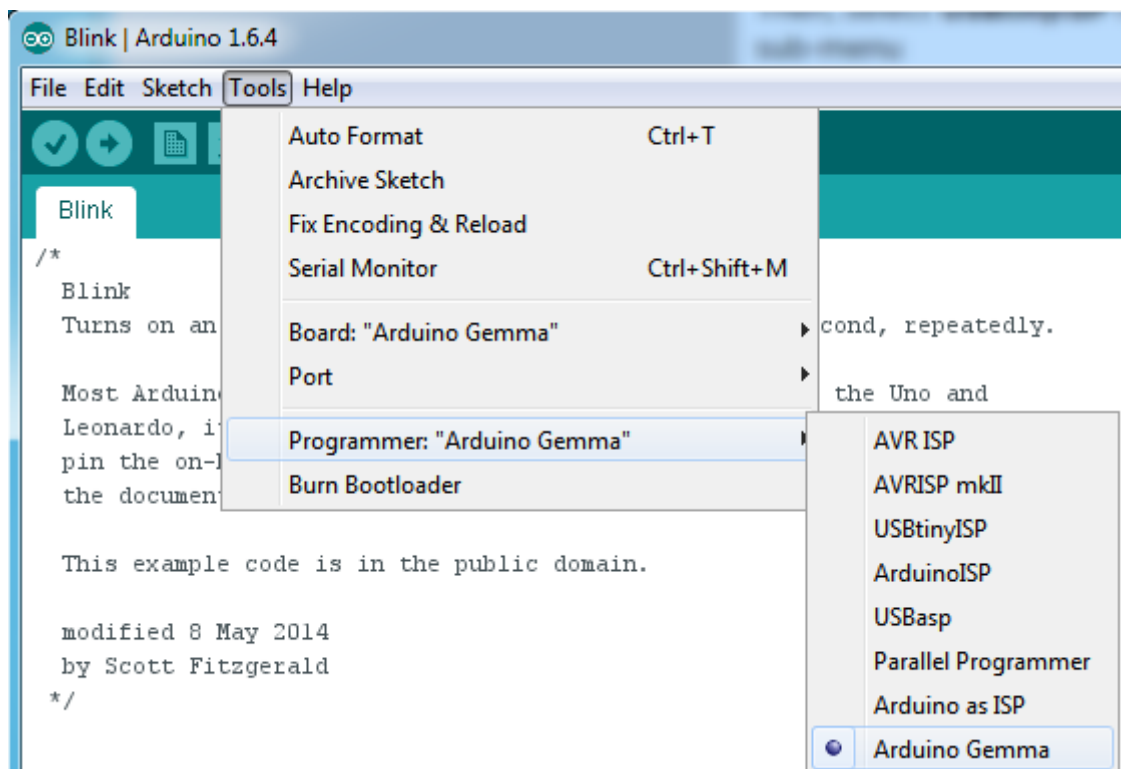


## Arduino Gemma (Teal GEMMA)

However if you're using the newer Arduino Gemma (with a teal PCB board) select the Arduino Gemma board from the Tools->Boards menu.



Then select Arduino Gemma as the Programmer type.



If you don't have the boards available inside your Arduino IDE, then you must import them before the previous steps. To do so, from the Tools->Boards menu click Boards Manager.



Search and install **Arduino AVR Boards**, which will give you access to the [Arduino GEMMA](#).

When it comes to the **Adafruit GEMMA**, navigate to the **Preferences** menu. You can access it from the **File** menu in Windows or Linux, or the **Arduino** menu on OS X. Add the following URL to the Additional Boards Manager URLs option: [https://adafruit.github.io/arduino-board-index/package\\_adafruit\\_index.json](https://adafruit.github.io/arduino-board-index/package_adafruit_index.json)

Finally, redo the same process as for the Arduino Gemma searching and installing Adafruit AVR Boards.

## IV. Use

After the previous setup, you will be able to correctly upload a code on the board.

To do so, **plug the board over USB to your computer** (preferably over USB v2). The state LED should start blinking and stop after a few seconds.

You may have noticed that no port selection has been done from **Tools->Port**. That is because, being devoid of Serial ports as it has been highlighted before, **the board will not show up**.

Check if the board is well connected and recognized correctly by your computer in the Device Manager.

At the end of the day, **enter Bootloader mode** pushing on the board's **pushbutton** (make sure that the board is still recognized inside the Device Manager), the **LED blinks again**, and it is now time to perform the upload. Once it has ended, it is confirmed by the IDE and the LED stops again.