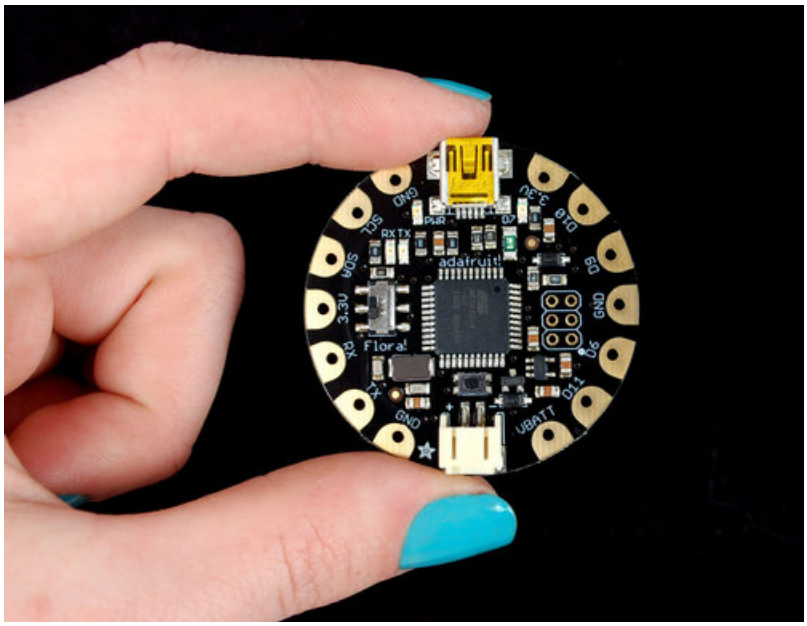
adafruit learning system

# Getting Started with FLORA

Created by Becky Stern



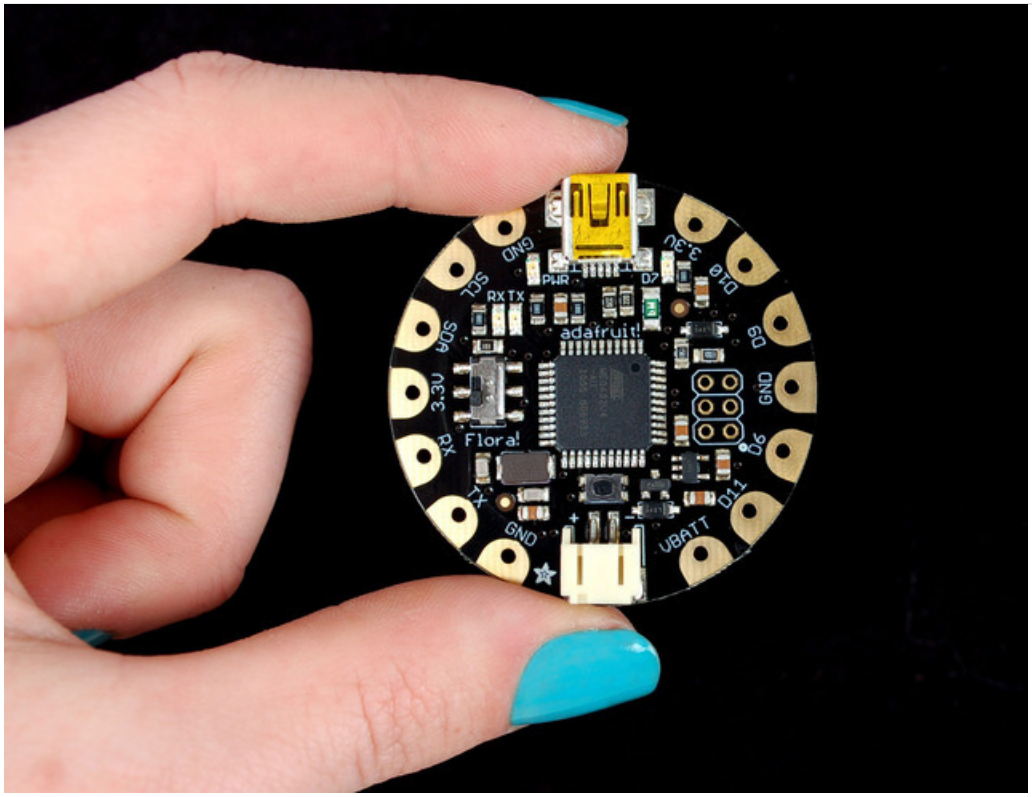

Last updated on 2018-08-22 03:32:32 PM UTC

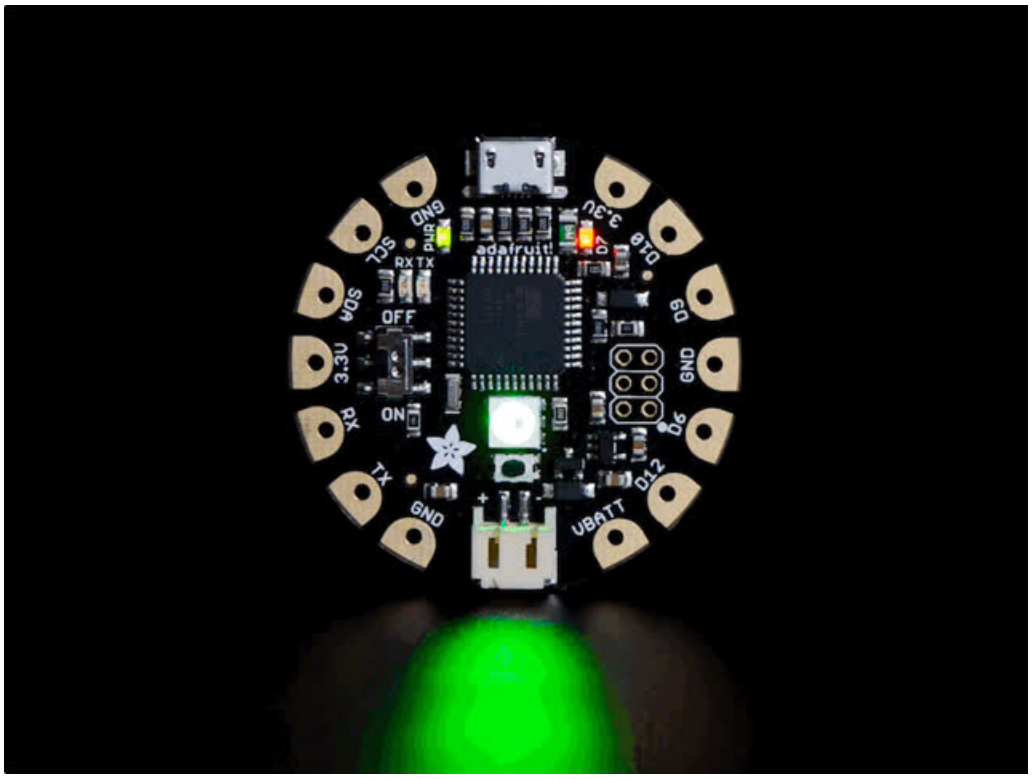
## Guide Contents


Guide Contents 2
Overview 3
Windows Driver Installation 6
Manual Driver Installation 7
Download software 9
Blink onboard LED 10
Blink onboard NeoPixel 13
Install the NeoPixel Library 13
Demo Code 13
Power your FLORA 16
The VBATT Pad 17
FLORA pinout diagram 18
- What about SPI? 18
- Some Technical Specs: 18

FLORA projects 20
FLORA techniques 21
FLORA modules 22
Downloads 24
Files 24
Schematic & Fabrication Print 24

## Overview



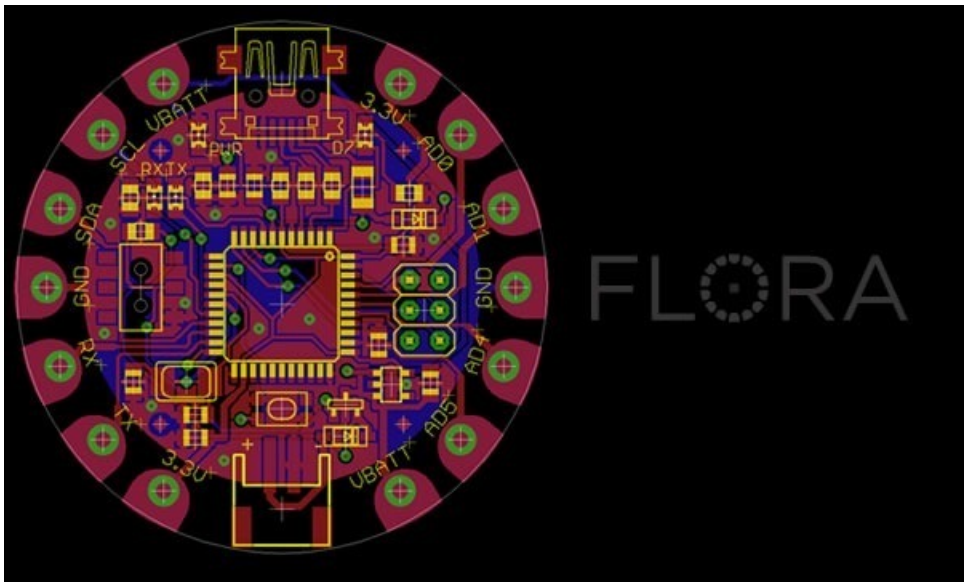

FLORA is Adafruit's wearable electronics platform. We designed it from the ground up to provide the best experience for Adafruit's community of makers, hackers, crafters, artists, designers and engineers. It's built around the Atmega32u4 chip, which has built-in USB support. No pesky special cables or extra parts for programming, just plug it

in and get started making the wearables project of your dreams! Works on Windows and Mac.

**New! As of May 12th, 2015, we're now selling the Flora v2! The Flora v2 now comes with a micro-USB port instead of a mini-USB port and a programmable NeoPixel installed** (it's connected to Digital 8, ready for your blinky commands)

The FLORA is extremely beginner-friendly-- it is difficult to destroy the FLORA by connecting a battery backwards due to polarized connector and protection diodes. The onboard regulator means that even connecting a 9V battery will not result in damage or tears.

This guide is the first step in using FLORA, plugging it in and programming it to blink its onboard LED! After that you can move on to our Flora RGB Smart Pixels tutorial (https://adafru.it/aRT) and/or Flora GPS tutorial (https://adafru.it/aRP).




Adafruit created the FLORA from scratch after many months of research and we really think we came up with something that will empower some amazing wearable projects.

The FLORA is small (1.75" diameter). We wanted the smallest possible board for our wearable platform. It's based on our experiences shipping our own, shipping, customer-tested Atmega32u4 (http://adafru.it/296) Breakout Board (http://adafru.it/296).

FLORA has a small but easy to use onboard reset button to reboot the system. The power supply is deigned to be flexible and easy to use. There is an onboard polarized 2 JST battery connector with protection schottky diode for use with external battery packs from 3.5v to 9v DC in (It can run on 3V but 3.5V-5V or higher is ideal). Can be used with LiIon/LiPoly, LiFe, alkaline or rechargeable NiMh/NiCad batteries of any size.
The FLORA does **not** have a LiPo charger included by design, this allows safe use with multiple battery types and reduces risk of fire as it is not recommended to charge these batteries on fabric.

FLORA has built-in USB support. Built in USB means you plug it in to program it, it just shows up. No additional purchases are needed! Works with Mac, Windows, Linux, any USB cable works great. Currently the PCB comes with a mini B connector but future versions may change to microUSB.
The FLORA has USB HID support, so it can act like a mouse or keyboard, etc.

adafruit
FLORA

## Windows Driver Installation

Mac and Linux do not require drivers, only Windows folks need to do this step

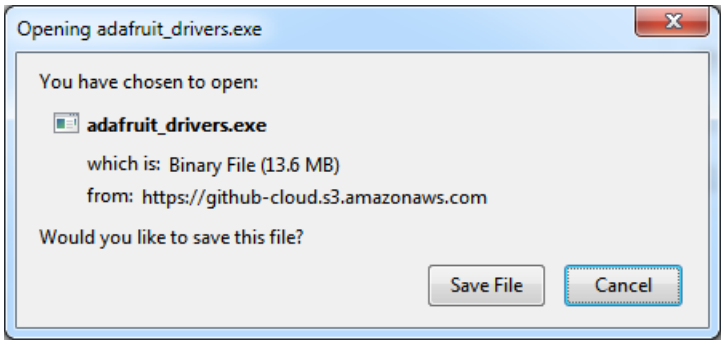
Before you plug in your board, you'll need to possibly install a driver!

Click below to download our Driver Installer.

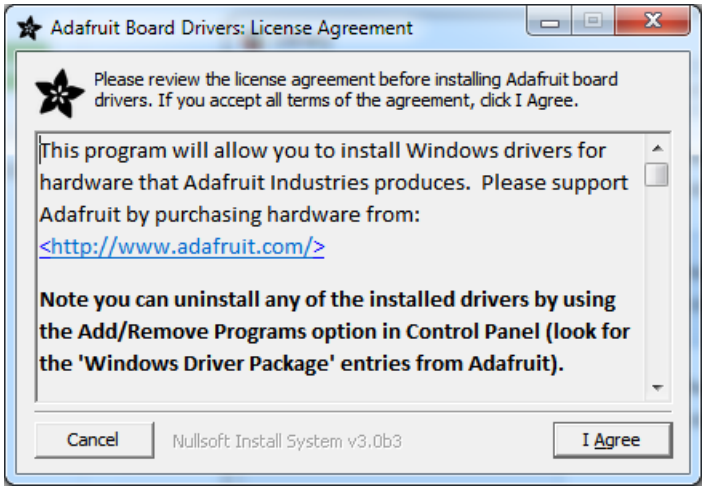
https://adafru.it/AB0

https://adafru.it/AB0

Download and run the installer.




Run the installer! Since we bundle the SiLabs and FTDI drivers as well, you'll need to click through the license




Select which drivers you want to install, we suggest selecting all of them so you don't have to do this again!

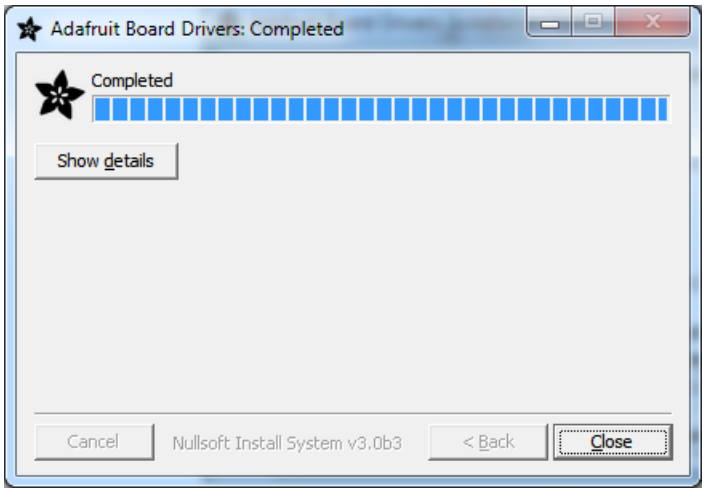
On Windows 7, by default, we install a single driver for most of Adafruit's boards, including the **Feather 32u4, the Feather M0, Feather M0, Express, Circuit Playground, Circuit Playground Express, Gemma M0, Trinket M0, Metro M0 Express**. On Windows 10 that driver is not necessary (it's built in to Windows) and it will not be listed.

The **Trinket / Pro Trinket / Gemma / USBtinyISP** drivers are also installed by default.

You can also, optionally, install the **Arduino Gemma** (different than the Adafruit Gemma!), **Huzzah and Metro 328** drivers.

Click **Install** to do the installin'.

Note that on Windows 10, support for many boards is built in. If you end up not checking any boxes, you don't need to run the installer at all!




## Manual Driver Installation

If windows needs the driver files (inf/cat) for some reason you can get all the drivers by downloading the source code zip file from this link:

https://adafru.it/AB0

https://adafru.it/AB0

And point windows to the **Drivers** folder when it asks for the driver location

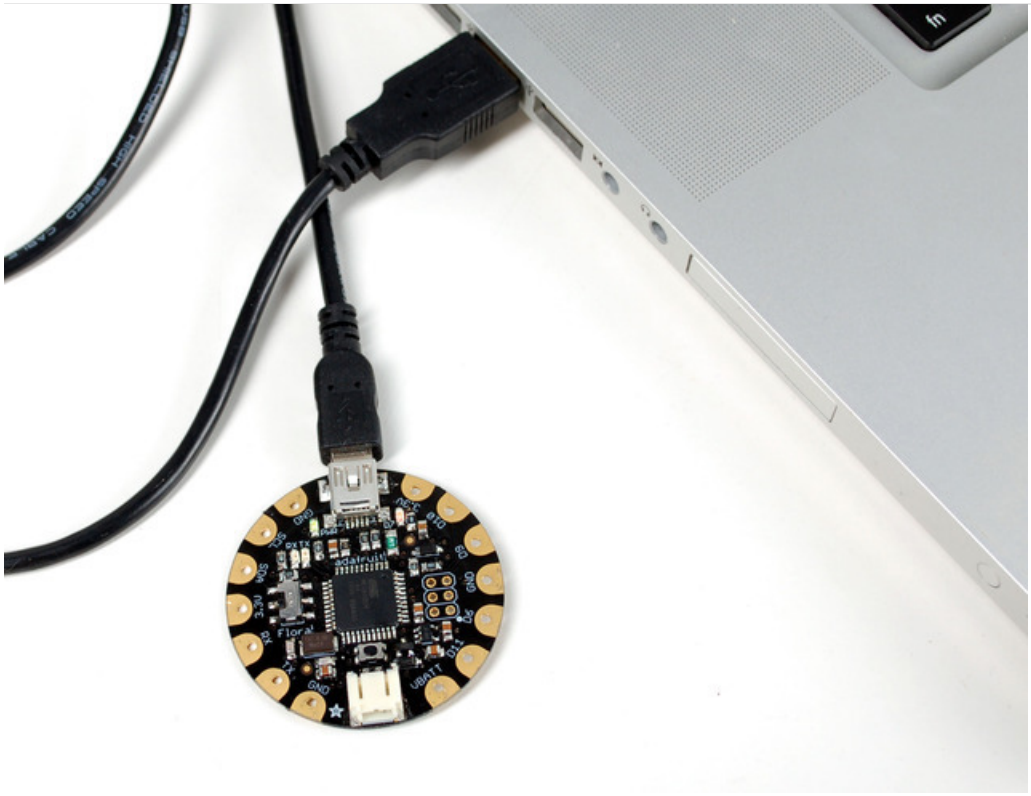
## Download software

To install the software for programming Flora just follow the steps in the steps in the **Adafruit Arduino IDE setup guide (https://adafru.it/jDQ)** to easily install support for Flora into your Arduino IDE 1.6 or higher

When you're finished installing the IDE come back to this page to continue the Flora guide.

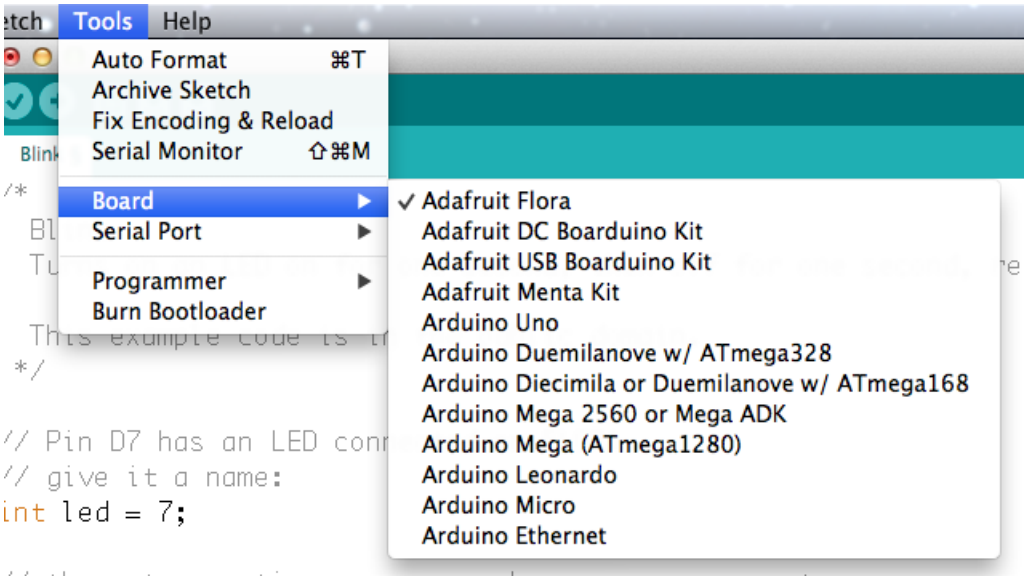
> On Windows be sure to install the Flora driver as noted on the Windows install page here: https://learn.adafruit.com/adafruit-arduino-ide-setup/windows-setup

## Blink onboard LED

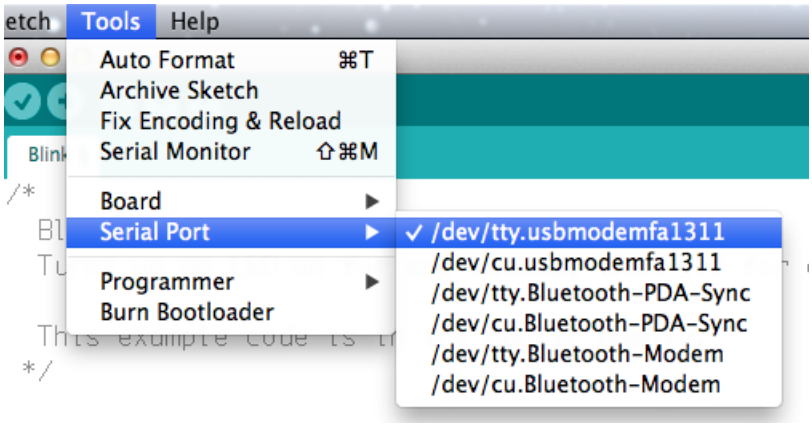


Next it's time to load up a program on your FLORA. There is an LED on board, so let's blink it! Plug in the USB cable and paste the following code into the Adafruit Flora IDE:

```
// Pin D7 has an LED connected on FLORA.
// give it a name:
int led = 7;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);               // wait for a second
}
```



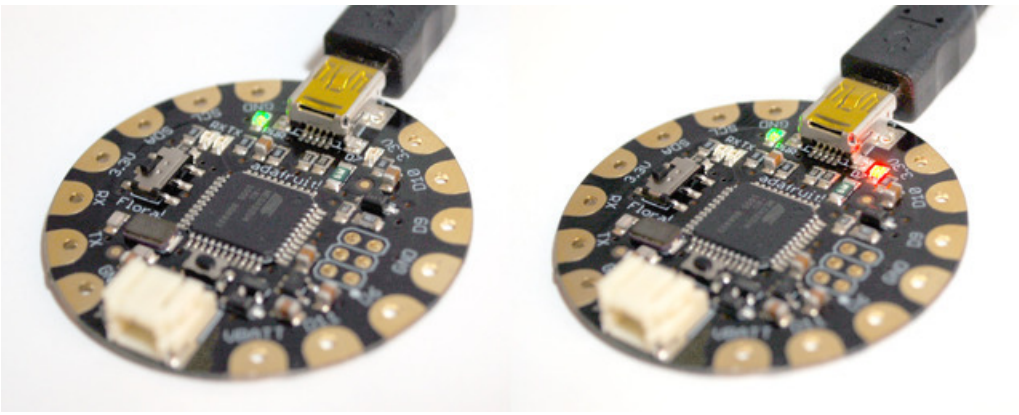

From the Tools menu, under "Board," choose "Adafruit Flora"




Also in the Tools menu, under "Serial Port," choose the one that contains the phrase "usbmodem" if you have a Mac.

If you're using a Windows computer, it will be named **COM**something, but not COM1 or COM2 (so it will be whatever comes after those two if they exist, such as COM3 or COM4).

Press the Upload button to transmit the program to the FLORA. It looks like an arrow pointing to the right.



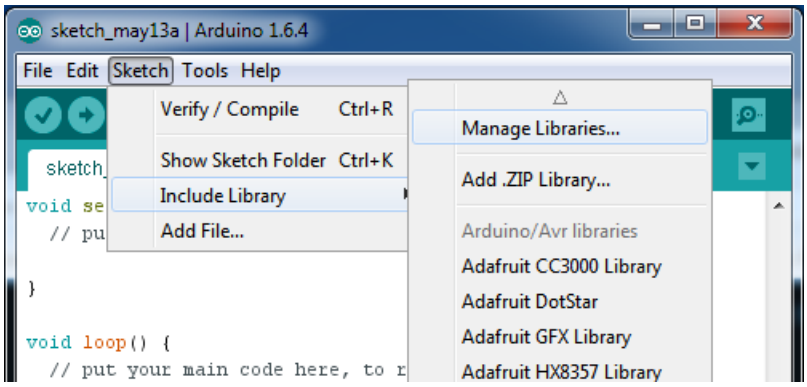
That's it! The on board LED marked "D7" should blink on and off repeatedly, and you've successfully programmed your FLORA!

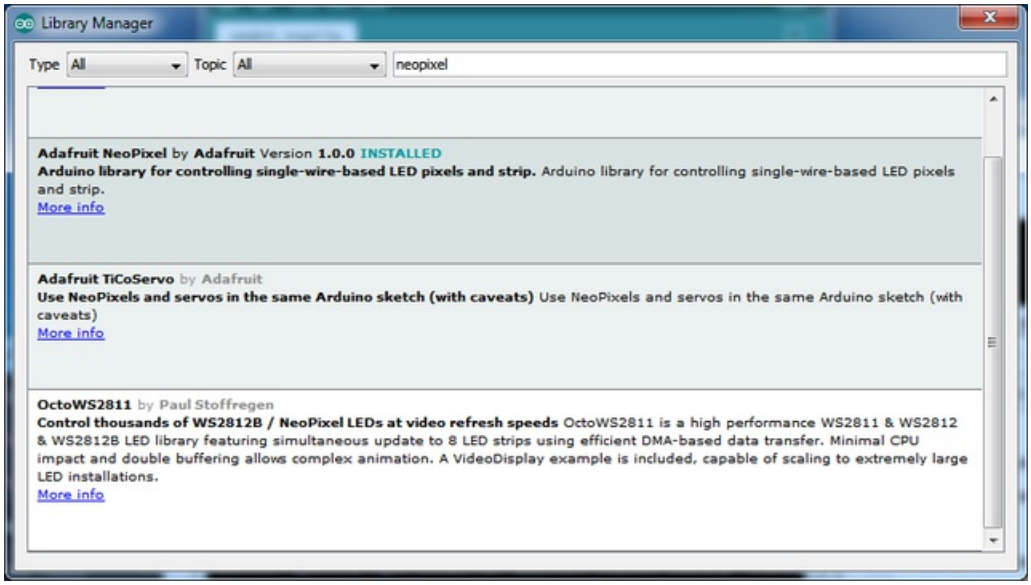
## Blink onboard NeoPixel

If you have a Flora v2, your Flora comes with an onboard NeoPixel! This lets you have a nice glowy LED using only one pin (Digital 8)

## Install the NeoPixel Library

If you are running Arduino IDE 1.6.1 or higher, you can install the library using the built in library manager, search for and install **Adafruit_NeoPixel**



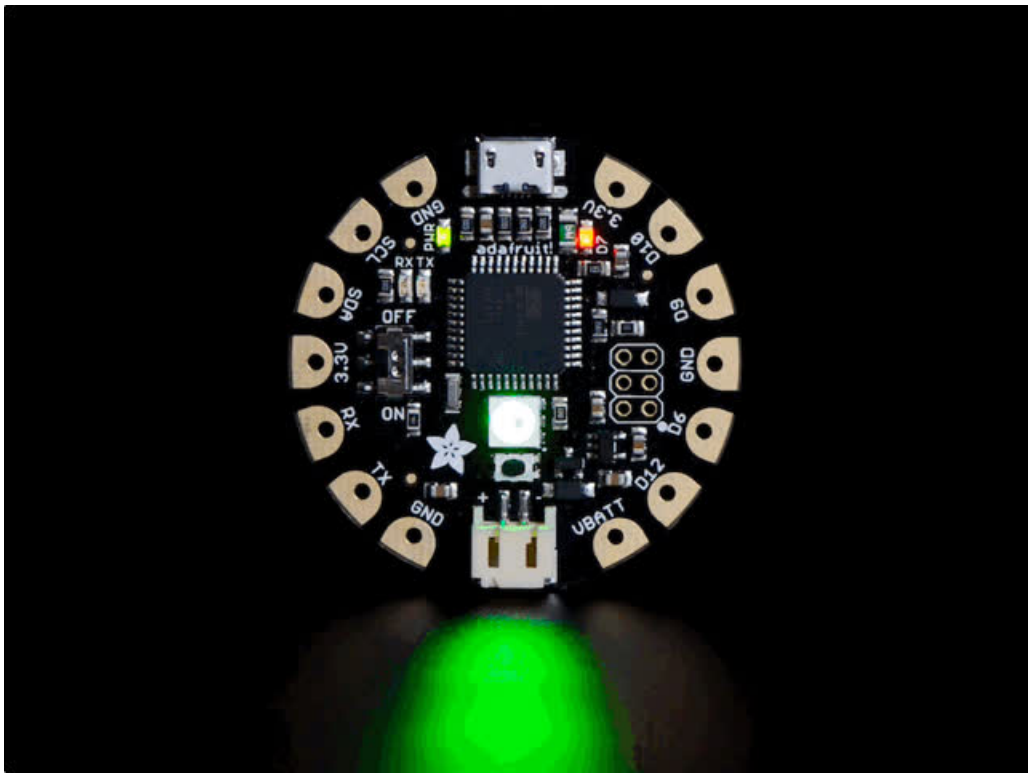

Install and close out the Library manager.

If you're not able to use the library manager or have an older IDE, you can always 'manually' install the library! (https://adafru.it/dhw)

## Demo Code

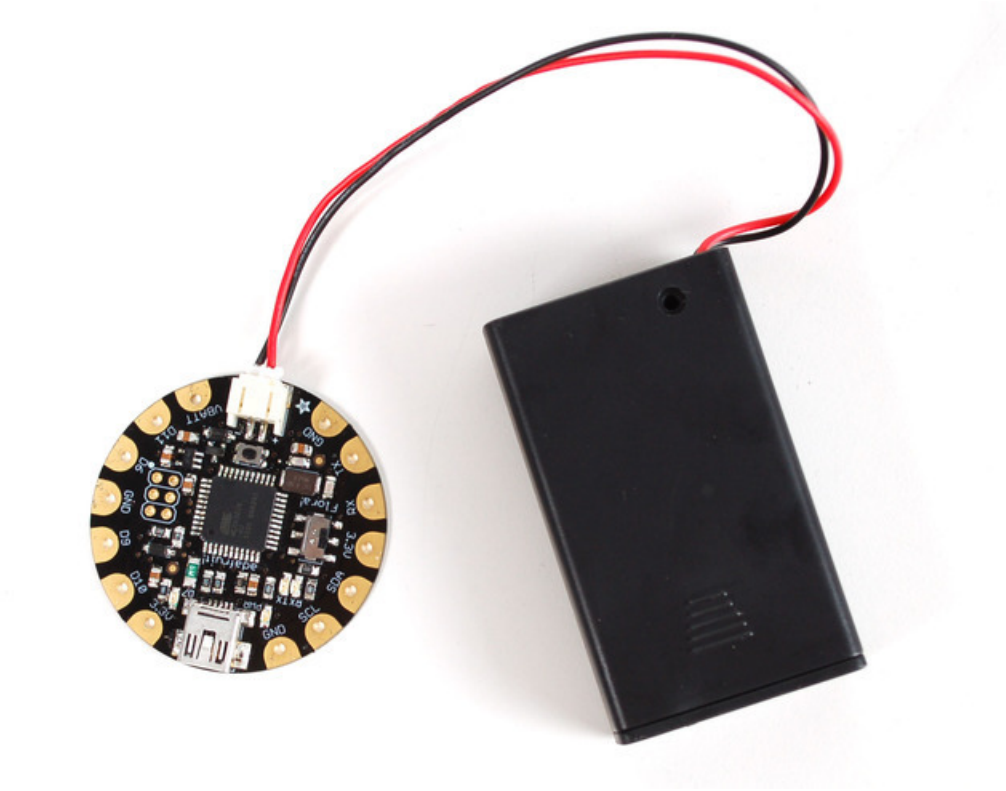
Upload the following sketch, note that we consider the 'strip' to be 1 pixel long, and connected to **Digital 8**

```
#include <Adafruit_NeoPixel.h>

#define PIN 8

Adafruit_NeoPixel strip = Adafruit_NeoPixel(1, PIN, NEO_GRB + NEO_KHZ800);

void setup() {
  strip.begin();
  strip.setBrightness(50);
  strip.show(); // Initialize all pixels to 'off'
}

void loop() {
  // Some example procedures showing how to display to the pixels:
  colorWipe(strip.Color(255, 0, 0), 500); // Red
  colorWipe(strip.Color(0, 255, 0), 500); // Green
  colorWipe(strip.Color(0, 0, 255), 500); // Blue
  rainbowCycle(20);
}

// Fill the dots one after the other with a color
void colorWipe(uint32_t c, uint8_t wait) {
  for(uint16_t i=0; i<strip.numPixels(); i++) {
      strip.setPixelColor(i, c);
      strip.show();
      delay(wait);
  }
}

// Slightly different, this makes the rainbow equally distributed throughout
void rainbowCycle(uint8_t wait) {
  uint16_t i, j;

  for(j=0; j<256*5; j++) { // 5 cycles of all colors on wheel
    for(i=0; i< strip.numPixels(); i++) {
      strip.setPixelColor(i, Wheel(((i * 256 / strip.numPixels()) + j) & 255));
    }
    strip.show();
    delay(wait);
  }
}

// Input a value 0 to 255 to get a color value.
// The colours are a transition r - g - b - back to r.
uint32_t Wheel(byte WheelPos) {
  WheelPos = 255 - WheelPos;
  if(WheelPos < 85) {
   return strip.Color(255 - WheelPos * 3, 0, WheelPos * 3);
  } else if(WheelPos < 170) {
    WheelPos -= 85;
   return strip.Color(0, WheelPos * 3, 255 - WheelPos * 3);
  } else {
   WheelPos -= 170;
   return strip.Color(WheelPos * 3, 255 - WheelPos * 3, 0);
  }
}
```

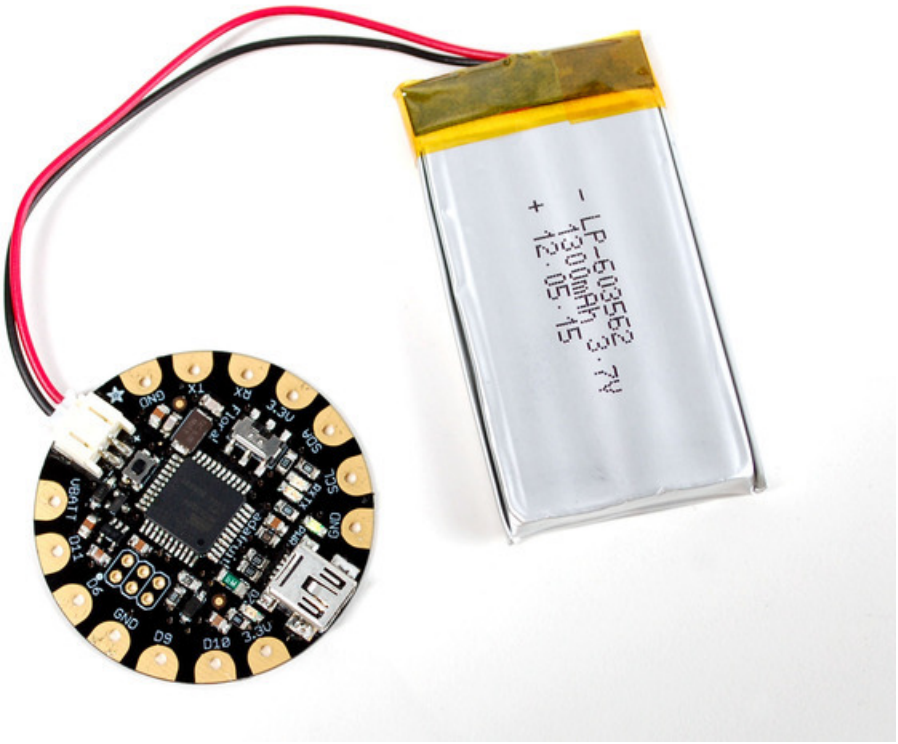

GND
SCL
RX TX
SDA
OFF
3.3V
ON
RX
TX
GND
adafruit
3.3V
D10
D9
GND
D6
D12
VBATT

## Power your FLORA

The Flora runs at 3.3V with an onboard regulator to keep the voltage steady. You'll need to power the board with a 3.6V or greater battery. For that reason, we find 3 x AA or AAA or a lithium-polymer battery to be ideal. You can only power FLORA through its onboard JST port.



After loading your program onto the FLORA, you'll want to unplug the USB cable and go portable with battery power! USB will always power the board, but battery power (through the JST connector) can be turned on and off with Flora's onboard switch (near the "ON" text on the board). We recommend our 3xAAA holder (http://adafru.it/727) for beginners, which conveniently plugs into FLORAs JST port.

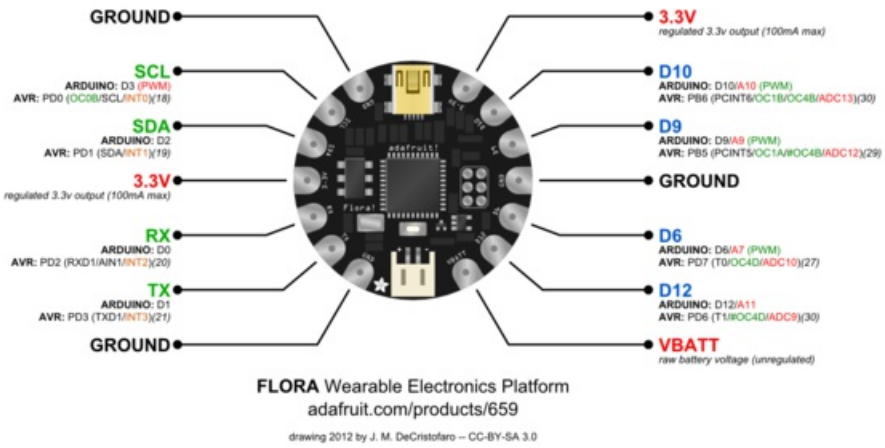



More advanced users may wish to use a rechargeable lithium polymer battery (http://adafru.it/258) for their smaller size and longer duration. These batteries pose added risk if abused, shorted, bent, crushed, or punctured. FLORA does not have onboard charging, so you would also need a separate LiPoly charger (http://adafru.it/259).

## The VBATT Pad

The most commonly confused part of the Flora is the **VBATT** pad and how power management is handled in the Flora. We designed the flora to be mainly powered via the JST connector. You can power it with a battery pack or lipo battery. That input is polarity protected and also controlled via the on/off switch. The battery output **after** the switch is **VBATT**. You shouldn't use **VBATT** as an input as you might damage your battery if you also plug in the USB connector to a computer! The ideal use of **VBATT** is when you want to power something like NeoPixels or a servo, something that requires more than the 150mA available from the onboard regulator.

## FLORA pinout diagram

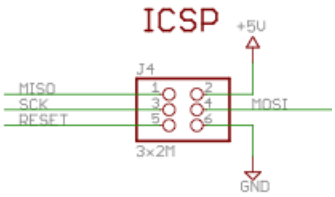



For handy reference, we've created this pinout diagram illustrating all the alternate functions for each of the pins on the Flora.

The most commonly confused part of the Flora is the **VBATT** pad and how power management is handled in the Flora. We designed the flora to be mainly powered via the JST connector. You can power it with a battery pack or lipo battery. That input is polarity protected and also controlled via the on/off switch. The battery output **after** the switch is **VBATT**. You shouldn't use **VBATT** as an input as you might damage your battery if you also plug in the USB connector to a computer! The ideal use of **VBATT** is when you want to power something like NeoPixels or a servo, something that requires more than the 150mA available from the onboard regulator.

### What about SPI?

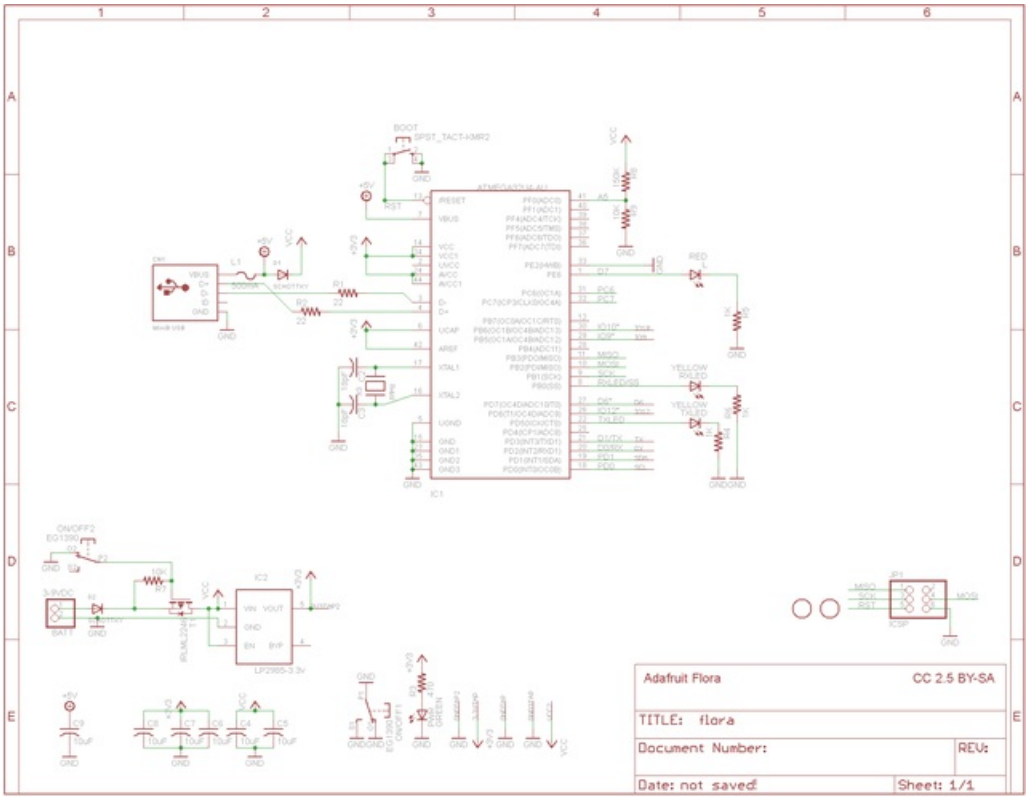
The Flora also supports hardware SPI.  The SPI pins are on the 2x3 header near the center of the Flora.  A small white dot indicates pin 1.  The SPI header pinout is shown here:




### Some Technical Specs:

- **Battery input (JST)**: 3.5-16V (the regulator is MIC5225-3.3 (https://adafru.it/dQO)) with reverse polarity protection, 2A max rated connector. We recommend no more than 6VDC so that the 3.3V regulator does not overheat
- **USB input**: 4.5V-5.5V with 500mA fuse
- **3.3V output pad**: we recommend no more than 100mA, as the regulator can supply 150mA and the Flora Mainboard uses 20mA. but you can probably draw up to 250mA in spikes
- **Clock speed:** 8MHz
- **Chipset:** ATmega32u4
- **VBAT output pad:** the higher of the VBAT JST input voltage and USB voltage (two schottkey diode connection)

- **Current Draw:** 8mA quiescent, another 2mA when the pin #D7 LED is on.



## FLORA projects

Flora Pixel Brooch (https://adafru.it/aTj)

Flora TV-B-Gone (https://adafru.it/kAd)

GPS Jacket (https://adafru.it/aWF)

Capacitive Touch with Conductive Fabric (https://adafru.it/aWG)

Wearable Piezo Tones (https://adafru.it/aWH)

LED Ampli-Tie (https://adafru.it/c1u)

Plush Game Controller (https://adafru.it/c9F)

## FLORA techniques

Flora snaps (https://adafru.it/aUl)

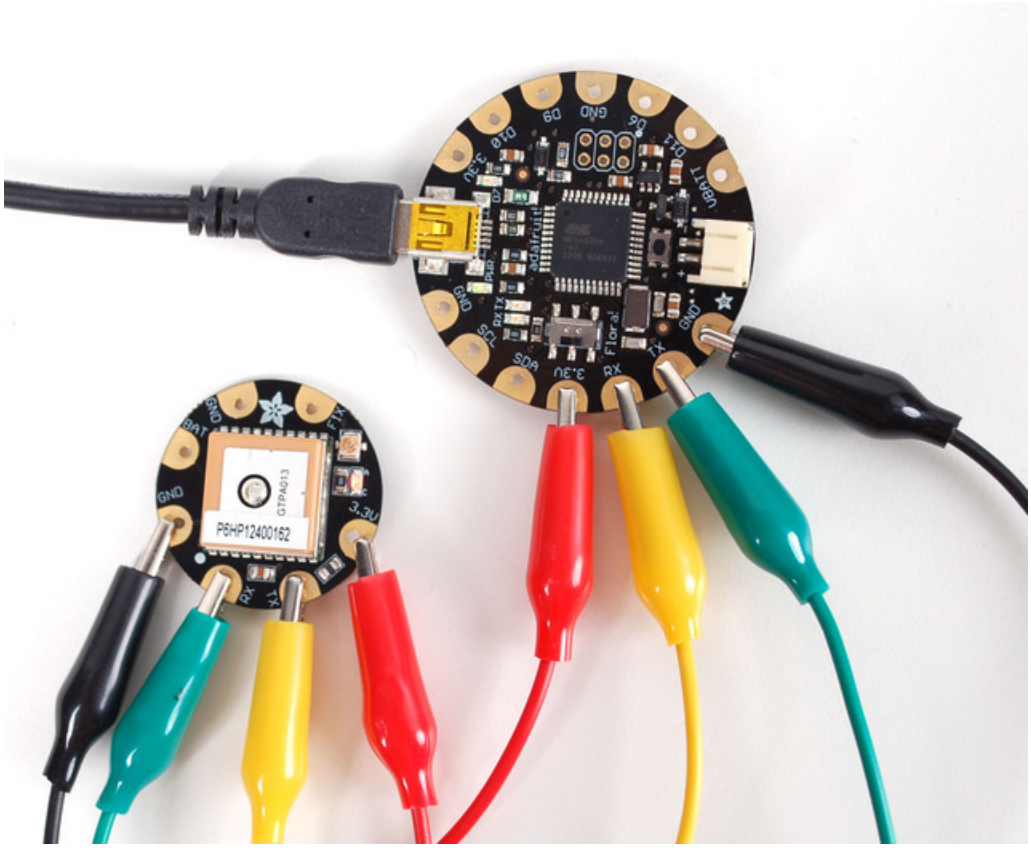
Conductive thread (https://adafru.it/aVx)

Capacitive Touch with Conductive Fabric & Flora (https://adafru.it/aWG)

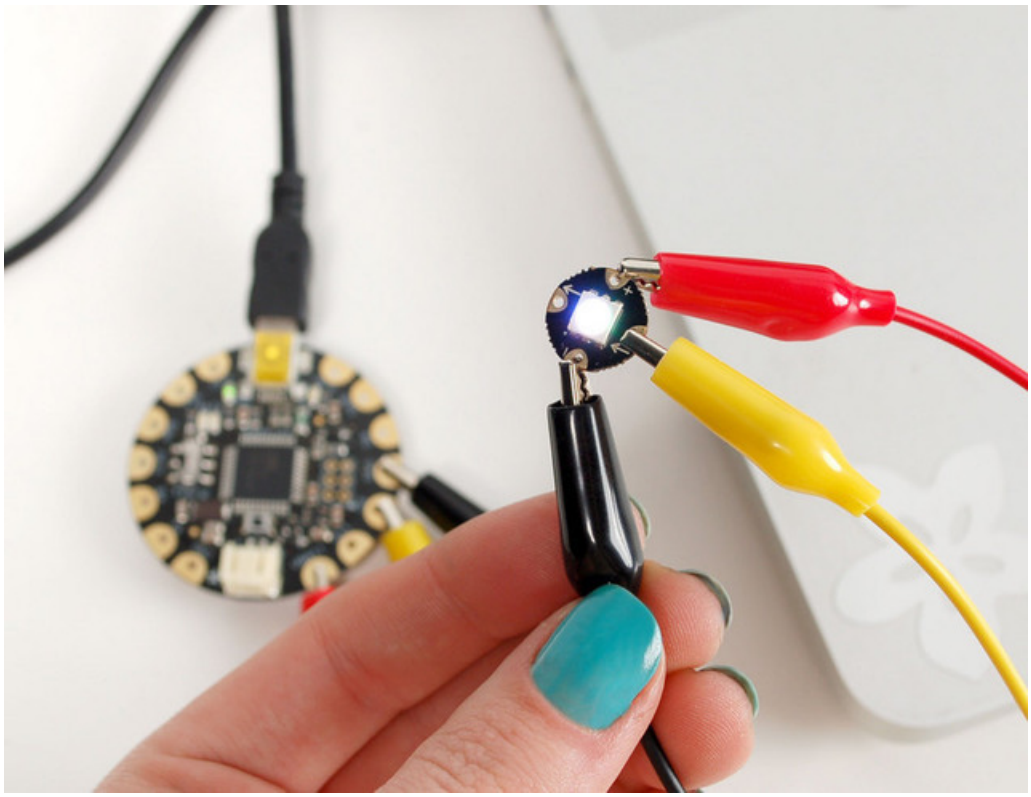
## FLORA modules

FLORA Accelerometer + Compass (https://adafru.it/aYS)

FLORA Luminosity Sensor (https://adafru.it/c9H)




FLORA GPS (https://adafru.it/aRP)



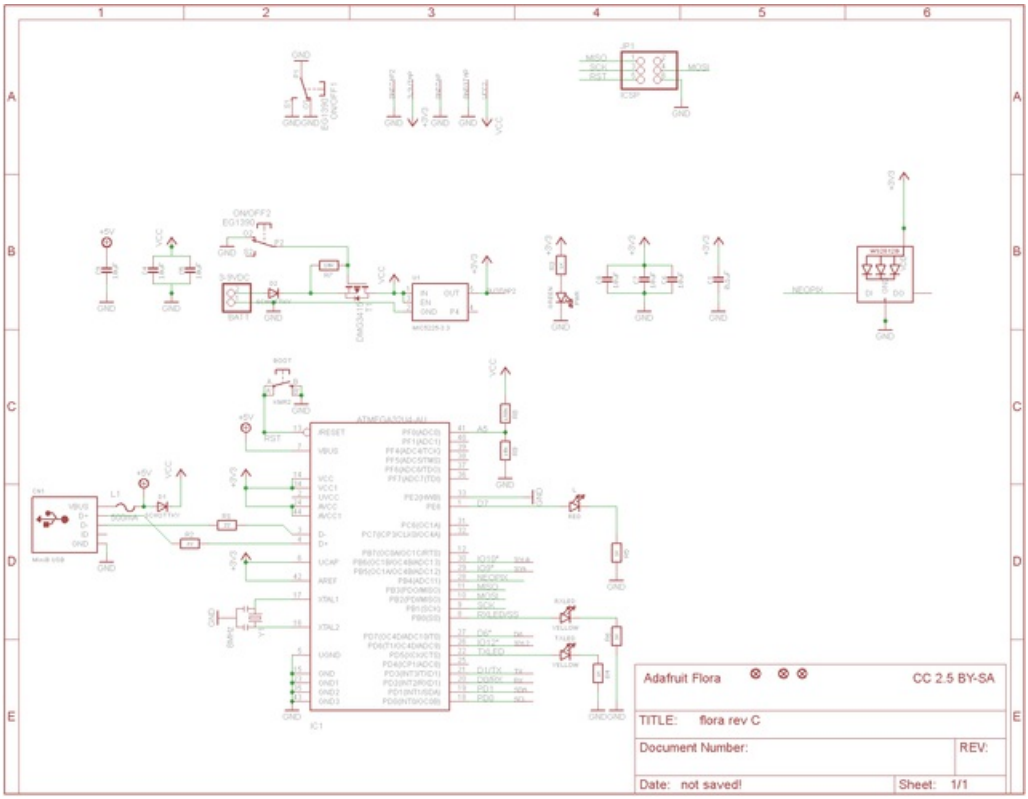
Flora RGB Smart NeoPixels (https://adafru.it/c9J)

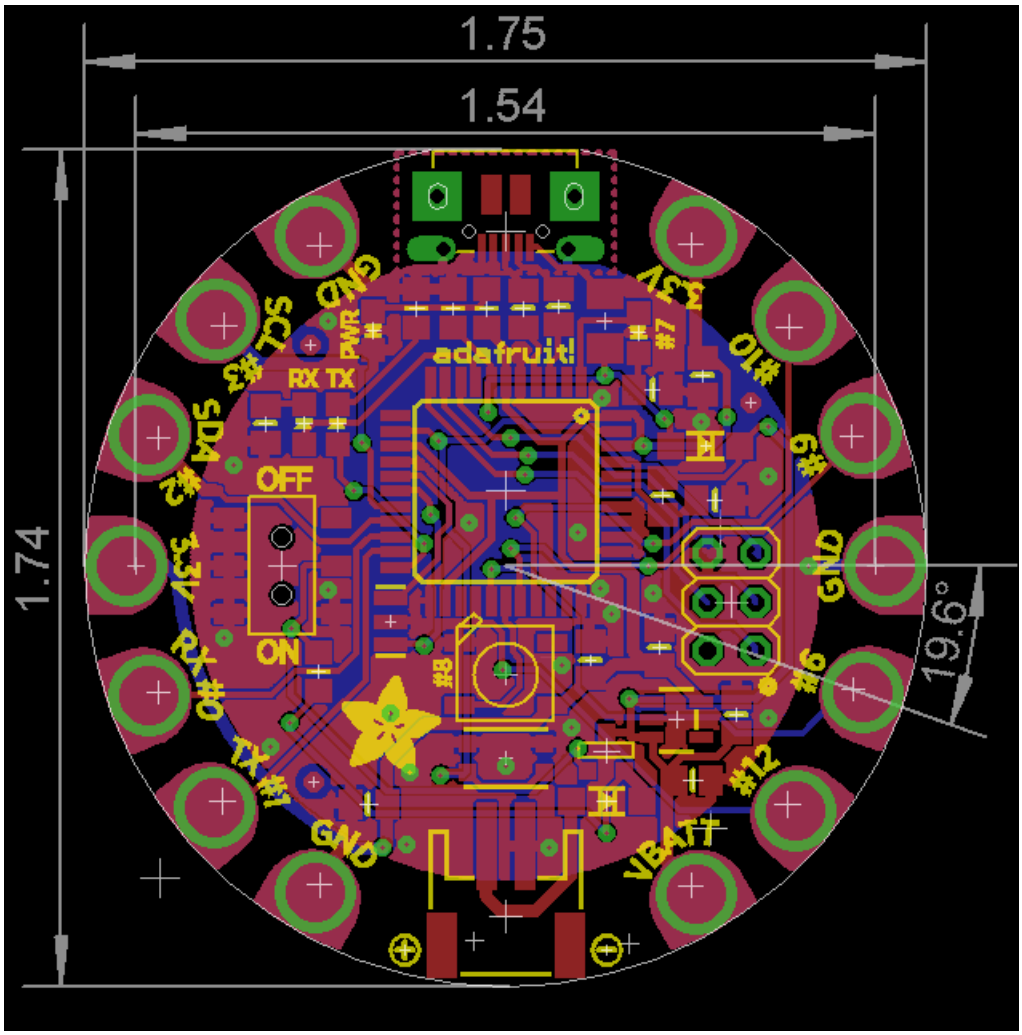
## Downloads

## Files

- EagleCAD PCB files on GitHub (https://adafru.it/rTe)
- Fritzing object in Adafruit Fritzing library (https://adafru.it/aP3)

## Schematic & Fabrication Print




1.75
1.54
1.74
19.6°
adafruit
GND
3.3V
SCL
#3
SDA
#2
3.3V
RX
#0
TX
#1
GND
VBAT
#12
#6
GND
#9
#10
#7
#8
PWR
RX TX
OFF
ON

# OCEAN CHIPS
## Океан Электроники
## Поставка электронных компонентов

Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;
- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком);
- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «**JONHON**», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «**FORSTAR**».

AVIC JONHON

**«JONHON»** (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

**«FORSTAR»** (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).

FORSTAR®

Телефон: 8 (812) 309-75-97 (многоканальный)
Факс: 8 (812) 320-03-32
Электронная почта: ocean@oceanchips.ru
Web: http://oceanchips.ru/
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А