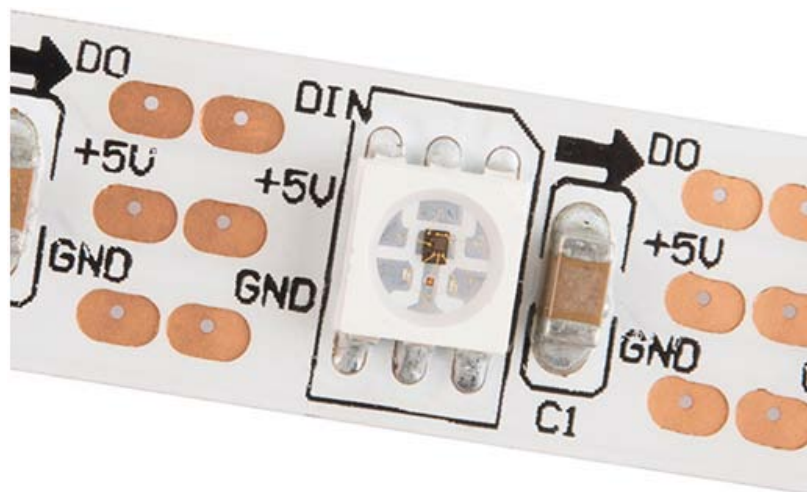


User Guide

WS2812B RGB LED Strip

These LED strips are just about the best way to get tons of colorful LED light with a minimum of wiring and fuss! Each strip is 50cm in length and contains 15 light elements. Each element contains a red, green, and blue LED, as well as a tiny WS2812 controller chip. This means that each element is individually addressable and can accept brightness values of 0-255 for each of its LEDs, giving you 24 bits of color precision overall. Each LED is activated by using a special serial data protocol. This protocol allows many LEDs to be daisy chained together so that one microcontroller can control the whole lot – with a single data signal. Any number of these LEDs can be chained together by connecting one module's data-out (DO) pin to another's data-in (DI) pin.



SKU: [DSP-1159](#)

Brief Data:

- Strip of 15 pieces/50cm of 5050 Red, Green, Blue (RGB) LEDs.
- Each LED element is individually addressable with 24-bit color (8 bits each for R, G & B), 16 million colors.
- 3-Wires easy connection, 2-power supply lines and 1 data line.
- 50cm in length.
- Interface: Serial (Shift Register).
- Data Speed: 800Kbps.
- Power Supply: 5V.

Application Introducing to WS2812B LED Strip with Arduino:

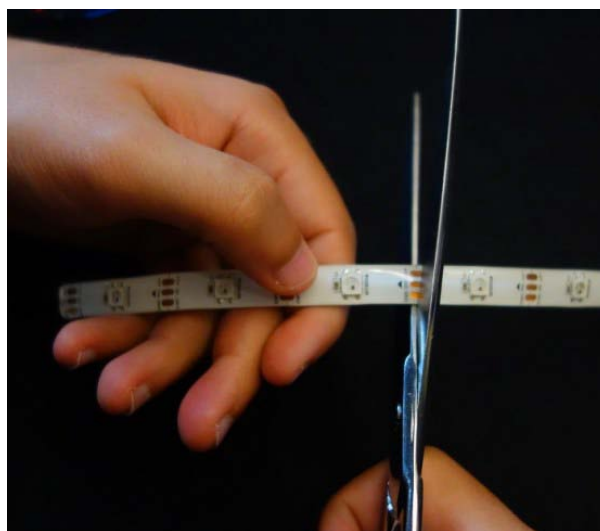
This is the coolest type of LED strips in the market today. You can control the brightness and the color of each LED individually, which allows you to produce amazing and complex effects in a simple way. This LED strip is made by WS2812B LEDs wired in series. These LEDs have a IC built right into the LED. This allows a communication via a one-wire interface. This means that you can control lots of LEDs using just one digital pin of your Arduino. In the following figure you can see the chip inside the LED. The LED is an RGB LED and works like so.



This kind of strips are very flexible and can be cut to any length you want. As you can see, the strip is divided into segments, and each segment contains one RGB LED.



You can adjust its size by cutting the strip with a scissors in the right place (the proper places to cut the strip are marked).



These strips come with connector pad at each end. I've decided to cut the connectors pad, and solder header pins. It's more handy if you want to connect the strip to an Arduino or to a breadboard.

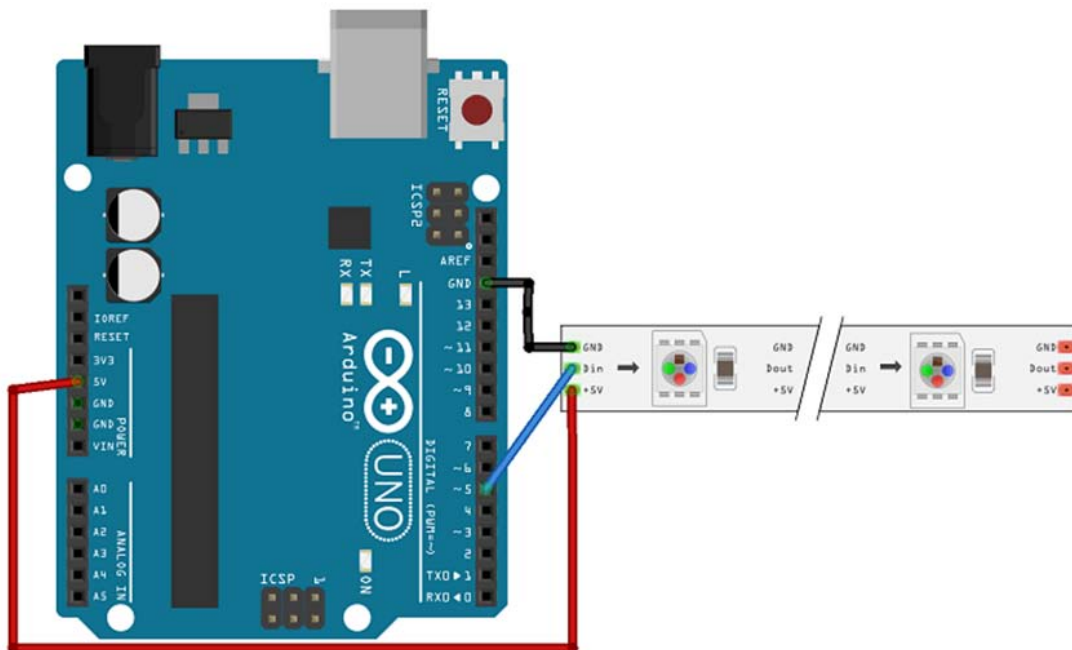


Powering the WS2812B LED Strip

The LED strip should be powered using a 5V power source. At 5V, each LED draws about 50mA, when set to its full brightness. This means that for every 30 LEDs, the strip may draw as much as 1.5 A. Make sure you select a power source that matches the strip's needs. If you end up using an external supply, don't forget to connect the power supply ground to the Arduino ground.

Schematics

In this example, the WS2812B LED strip will be powered using the 5V Arduino pin. In this case, we are controlling 15-LEDs/50cm strip. Bear in mind that if you want to control many LEDs, you'll need to use an external power supply.



Code

To control the WS2812B LED strip, you'll need to download the **FastLED** library.

Installing the FastLED library

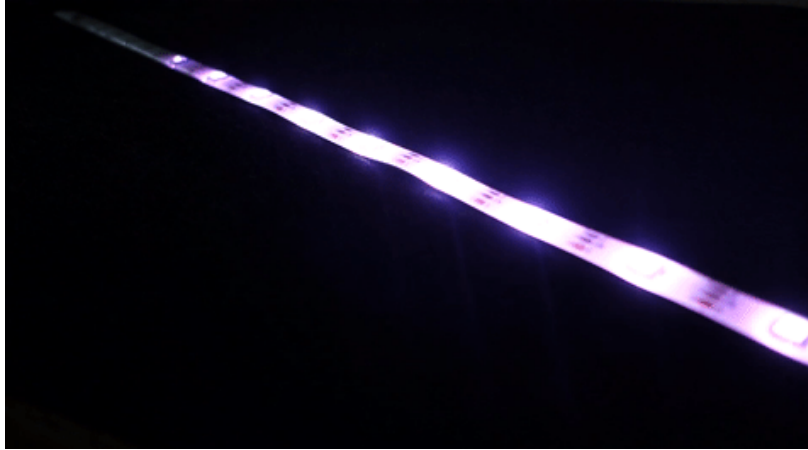
1. [Click here to download the FastLED library](#). You should have a .zip folder in your Downloads folder
2. Unzip the .zip folder and you should get **FastLED-master** folder
3. Rename your folder from **FastLED-master** to **FastLED**
4. Move the **FastLED** folder to your Arduino IDE installation **libraries** folder

5. Finally, re-open your Arduino IDE

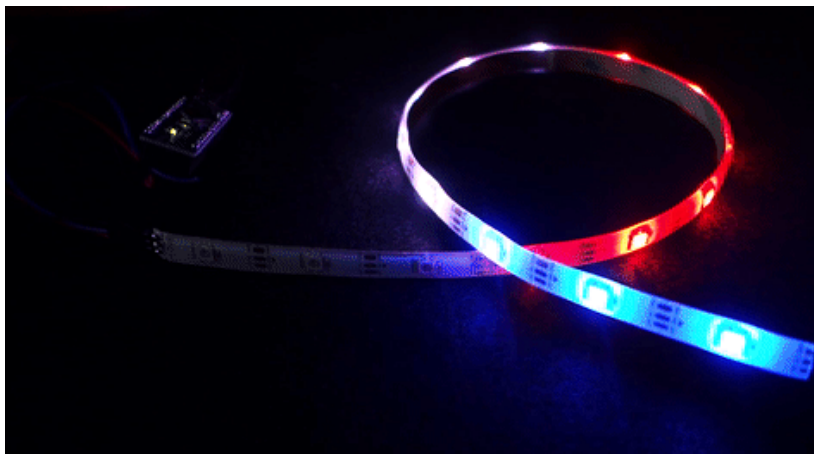
After installing the needed library, upload the following code to your Arduino board (this is an example sketch provided in the library examples folder). Go to File > Examples > FastLED > ColorPalette .

Demonstration

In the end, this is what you'll have. Amazing effects like this one:



Or this one...



And so on...

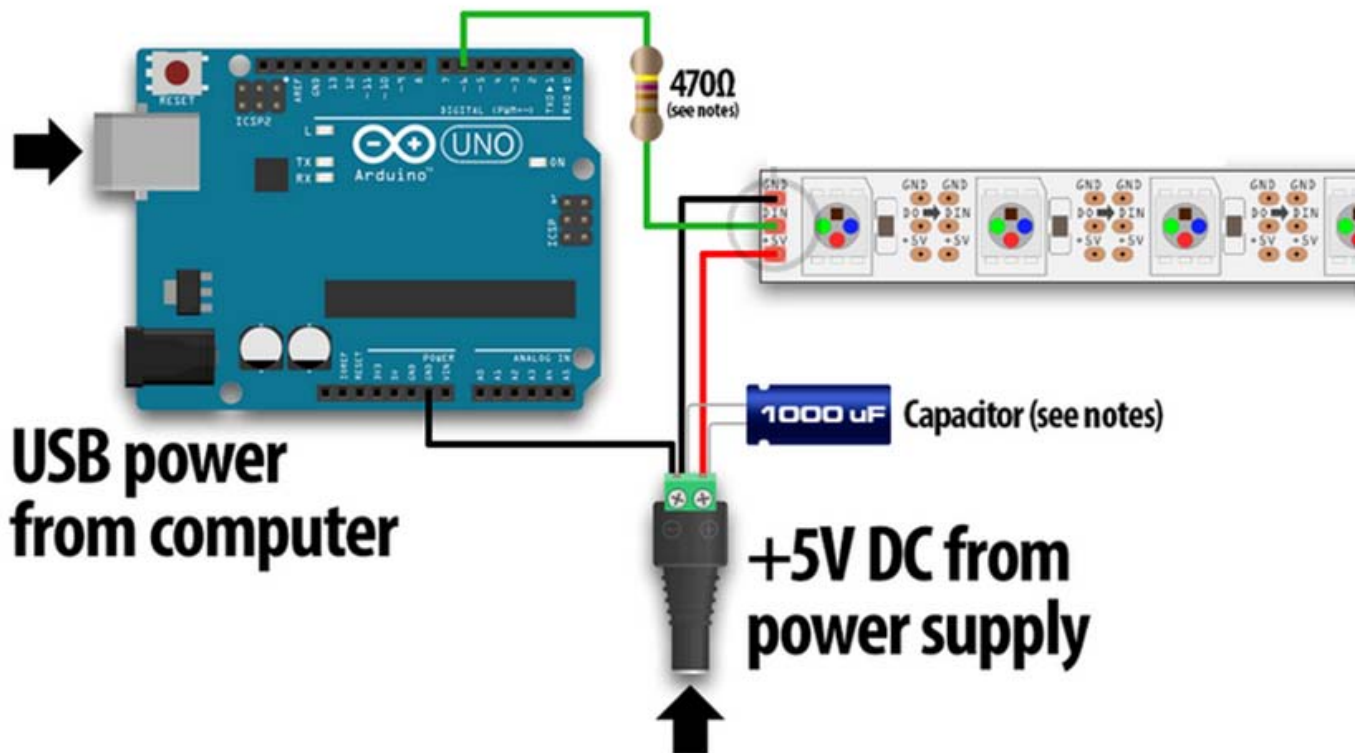


[Youtube Video Demo](#)

Suggest Connection Using External Power Supply:

For Arduino boards with a separate +5V DC power supply for the WS2812B RGB Strip: connect the +5V input on the strip to the + (positive) terminal on the power supply (don't connect to the Arduino), DIN to digital pin 6 (or any digital of your choice) on the Arduino, and – (minus or GND) on the strip must connect to both the minus (–) terminal on the DC supply and a GND pin on the Arduino (there are usually several — any will do).

When using a DC power supply, or an especially large battery, we recommend adding a large capacitor (1000 μ F, 6.3V or higher) across the + and – terminals. This prevents the initial onrush of current from damaging the LEDs.



Adding a $\sim 470\Omega$ resistor between the microcontroller's data pin and the data input on the RGB LED can help prevent spikes on the data line that can damage the first LED. Please add one between the micro and RGB LED!

Note: Suggest to use this external power supply connection if powering more than 60-LEDs as +5V on-board Arduino may not enough to power this.