

Adafruit will not be shipping orders Monday September 5, 2016 for the federal holiday, Labor Day. Orders placed after 11 AM EST Friday September 2 will go out Tuesday September 6.

0

-

- [SHOP](#)
- [BLOG](#)
- [LEARN](#)
- [FORUMS](#)
- [VIDEOS](#)
- [SIGN IN](#)
- [CLOSE MENU](#)

[0 Items](#)[Sign In](#)



- [SHOP](#)
- [BLOG](#)
- [LEARN](#)
- [FORUMS](#)
- [VIDEOS](#)



[Adafruit DotStar LEDs](#)

[Imagine NeoPixels with a double shot of espresso...](#)

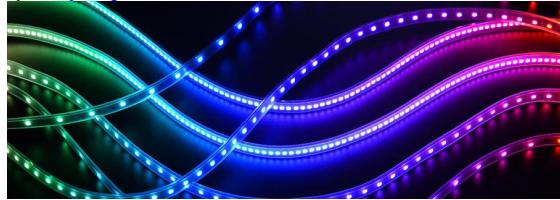
- [Overview](#)
- [DotStar Matrices](#)
- [•](#)
- [Single Page](#)
- [Download PDF](#)

Contributors

[Phillip Burgess](#)
[lady ada](#)
[Feedback? Corrections?](#)
[LEDS / LED STRIPS](#)

Overview

by Phillip Burgess



NeoPixel LEDs are the bee's knees, but in a few scenarios they come up short...connecting odd microcontrollers that can't match their strict timing, or fast-moving persistence-of-vision displays. Adafruit DotStar strips deliver high speed PWM and an easy-to-drive two-wire interface, bridging the gaps in the spectrum of awesome.

DotStars vs NeoPixels

The basic idea behind DotStars and NeoPixels is the same: a continuous string of individually-addressable RGB LEDs, driven by a microcontroller. The way each goes about it is a little different. DotStars aren't necessarily a better thing in every situation...there are tradeoffs, each has its pros and cons to consider...

DotStars

NeoPixels

- | | |
|---|---|
| + Extremely fast data ¹ and PWM ² rates, suitable for persistence-of-vision displays. | + More affordable. |
| + Easier to interface to a broader range of devices; no strict signal timing | + Wide range of form-factors (pixels, rings, matrices, etc.). |

requirements.

- + Works with Servo library, tone() function, interrupts, etc.
- Slightly more expensive.
- Strip form only; no other shapes.
- Needs two pins for control.

- + Works from a single microcontroller pin.
- Strict 800 KHz data rate; not all systems can generate this, and speed is a bottleneck on very long strands.
- 400 Hz refresh/PWM rate not suitable for persistence-of-vision effects.
- Not compatible with interrupts (e.g. Arduino Servo library or tone() function).

¹ Up to 8 MHz on Arduino, up to 32 MHz on Raspberry Pi.

² 19.2 KHz.

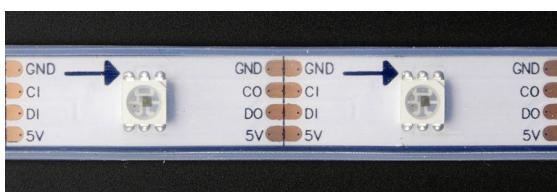
Powering DotStar LEDs

The power requirements for DotStars are pretty much identical to NeoPixels...in fact, we'll simply [refer you to the relevant page of the NeoPixel Überguide for pointers on estimating and routing power](#). In summary:

- Estimate up to 60 millamps peak for each pixel at full brightness white.
- For long strips, try to add a power tap every meter or so. This prevents a brown-out effect toward the end of the strip.
- It remains to be seen whether DotStars are as delicate as NeoPixels...but adding the power capacitor is a good idea anyway, do it!
- A ground connection is required between the microcontroller and strip, in addition to the signal lines.

Connecting DotStar LEDs

Our LED suppliers sometimes make unannounced production changes to the wiring. Therefore, **the best way to identify connections is a close visual examination of the strip.**



First, look for arrows printed along the strip next to each LED. These show the direction of data moving down the strip...your microcontroller connects at the originating ("in") end, the arrows point toward the "out" end. (In the photo above, our microcontroller would be located off the left side.)

Second, look for labels on the strip to identify the function and order of the four wires: ground, 5 Volts, data and clock...usually labeled GND, 5V, D or DI (data input) and C or CI (clock input). These will go to corresponding pins on your microcontroller and power supply.

Again, due to production variations, you can't always count on wire colors or plug genders as a reliable indication of function, even if ordered at the same time. Take a close look to confirm before connecting anything.

DotStars are 5V devices. While you might get them to respond to 3.3V signals, this is not a guaranteed thing and should not be counted on. For low-voltage microcontrollers and systems such as Raspberry Pi, a logic level shifter (e.g. [74AHCT125](#)) is recommended for both the data and clock pins.

Libraries and Example Code

We have an [Arduino library](#) to get you started:

[Click to download Adafruit_DotStar library for Arduino](#)

Unzip and rename the resulting folder "Adafruit_DotStar", move it to your Arduino/Libraries folder and restart the Arduino IDE. Library installation is a frequent stumbling block...if you need assistance, our [All About Arduino Libraries](#) guide spells it out in detail!

The "strandtest" example shows basic library use; declaring a strip object, setting LED colors and pushing this data to the strip via the show() method. For more advanced examples, nearly any NeoPixel code should compile and run with DotStars, just changing the library #include and the strip declaration.

Another production batch variation is the order in which color bytes are sent. Your own code can always use red, green, blue order when specifying colors, but the library needs to know the LED color sequence so it can adapt...you can see this as an optional last parameter to the constructor (change 'BRG' to 'GRB' or whatever data order your LEDs require):

[Copy Code](#)

```
1. Adafruit_DotStar strip = Adafruit_DotStar(60, 4, 5, DOTSTAR_BRG);
```

[Another option for Arduino is the FastLED library](#), featuring cutting-edge code with more color-handling and mathematical support functions. However, it's a little more tricky to use...so, if connecting DotStars for the first time, we ask that you **start with the Adafruit_DotStar library**. Once the hardware is confirmed working, you can then graduate to whatever code or library you'd like!

There's also a [Python module](#) that works with [Raspberry Pi](#):

[Click to download Adafruit_DotStar Python module for Raspberry Pi](#)

The Python code includes three examples. One is a simple strand test just to confirm that the LEDs are working (start here). The other two show how light painting and persistence-of-vision can be implemented using the Python Image module, which must first be installed with:

[Copy Code](#)

```
1. sudo apt-get install python-imaging
```

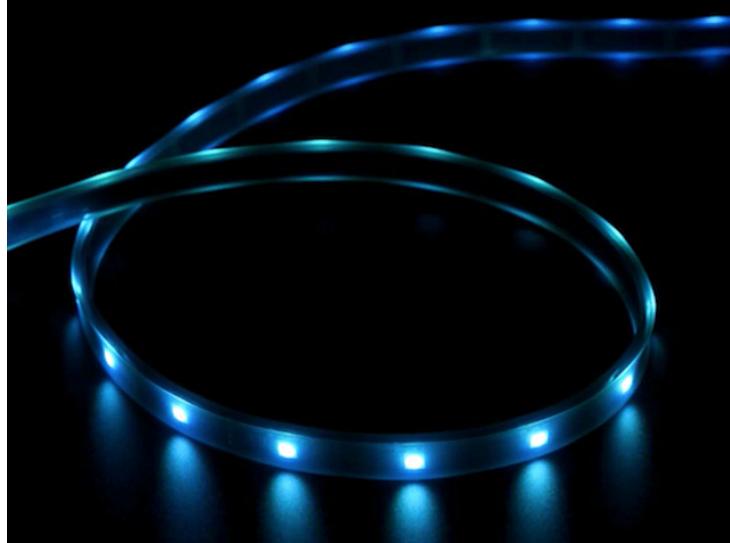
To use the SPI output mode in the examples (as opposed to the default "bit-bang" mode), you must enable the SPI kernel module in raspi-config (under "Advanced Options").

The Python code is modeled after the Arduino library (in turn based on the NeoPixel library), so the available functions are essentially the same, and existing NeoPixel examples can probably be ported with minimal fuss.

The Raspberry Pi is a 3.3V device. See notes above ("Connecting DotStar LEDs") regarding level-shifting.

DOTSTAR MATRICES

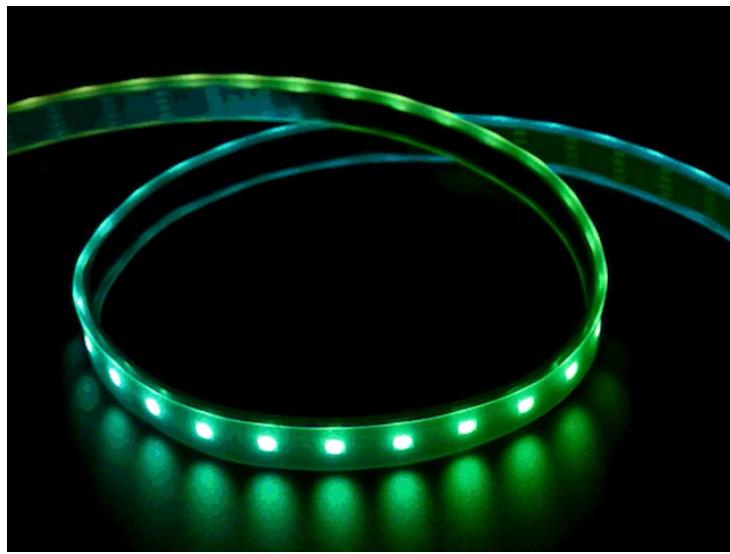
Last updated on 2015-10-24 at 07.18.45 PM Published on 2014-12-24 at 05.03.53 PM



Adafruit DotStar Digital LED Strip - Black 30 LED - Per Meter

\$19.95 [ADD TO CART](#)

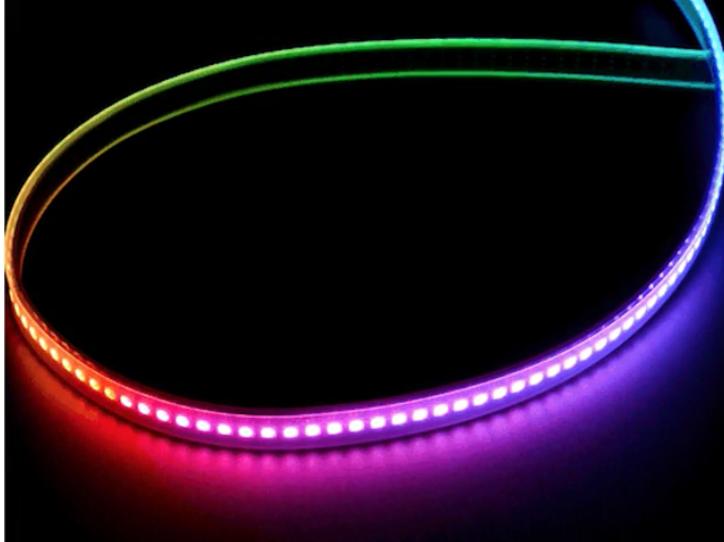
Adafruit DotStar Digital LED Strip - White 30 LED - Per Meter

\$19.95 [ADD TO CART](#)

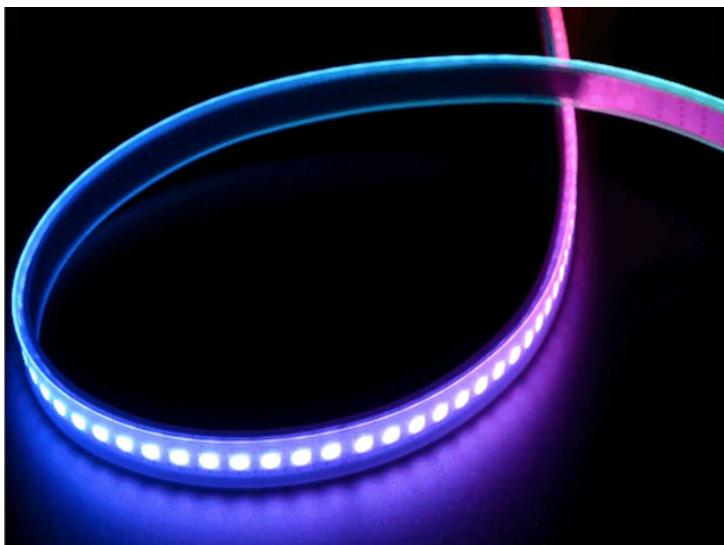
Adafruit DotStar Digital LED Strip - Black 60 LED - Per Meter

\$29.95 [ADD TO CART](#)

Adafruit DotStar Digital LED Strip - White 60 LED - Per Meter

\$29.95 [ADD TO CART](#)

Adafruit DotStar Digital LED Strip - Black 144 LED/m - One Meter

\$74.95 [ADD TO CART](#)

Adafruit DotStar Digital LED Strip - White 144 LED/m - One Meter

\$74.95 [ADD TO CART](#)

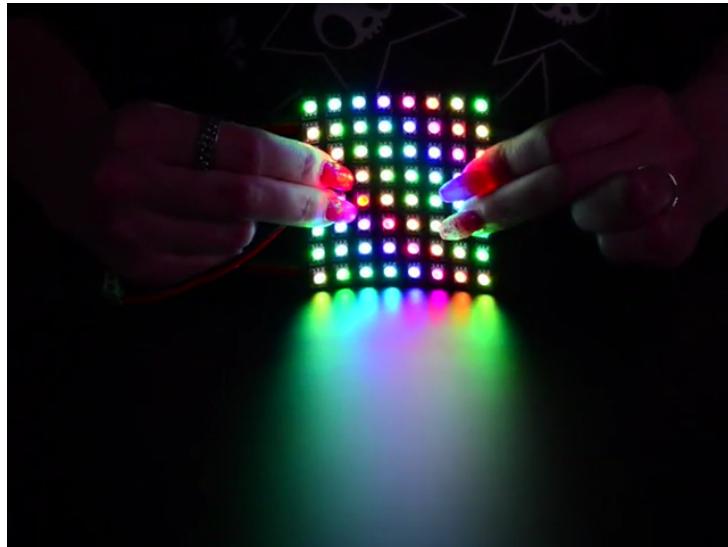
Adafruit DotStar Digital LED Strip - Black 144 LED/m - 0.5 Meter

\$39.95 [ADD TO CART](#)

9/5/2016

Overview | Adafruit DotStar LEDs | Adafruit Learning System

Adafruit DotStar Digital LED Strip - white 144 LED/m - 0.5 Meter
\$39.95 [ADD TO CART](#)



Flexible Adafruit DotStar Matrix 8x8 - 64 RGB LED Pixels
\$49.95 [ADD TO CART](#)



Flexible Adafruit DotStar Matrix 16x16 - 256 RGB LED Pixels
\$99.95 [OUT OF STOCK \(NOTIFY ME\)](#)



Flexible Adafruit DotStar Matrix 8x32 - 256 RGB LED Pixels

\$99.95 [ADD TO CART](#)

74AHCT125 - Quad Level-Shifter (3V to 5V)

\$1.50 [ADD TO CART](#)

5V 4A (4000mA) switching power supply - UL Listed

\$14.95 [ADD TO CART](#)

5V 2A (2000mA) switching power supply - UL Listed

\$7.95 [ADD TO CART](#)



5V 10A switching power supply
\$25.00 [ADD TO CART](#)



Female DC Power adapter - 2.1mm jack to screw terminal block
\$2.00 [ADD TO CART](#)



4-pin JST SM Plug + Receptacle Cable Set
\$1.50 [ADD TO CART](#)



Adafruit DotStar LED Strip - APA102 Warm White - 30 LED/m
\$19.95 [ADD TO CART](#)

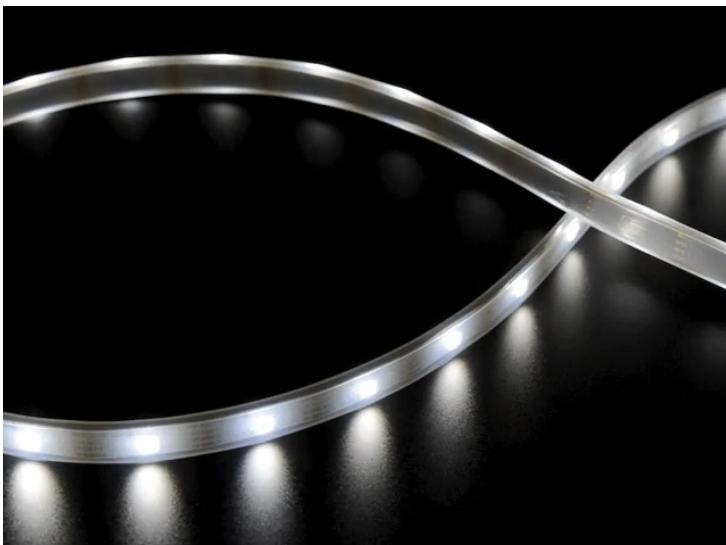




Adafruit DotStar LED Strip - APA102 Warm White - 60 LED/m
\$29.95 [ADD TO CART](#)



Adafruit DotStar LED Strip - APA102 Warm White - 144 LED/m
\$74.95 [ADD TO CART](#)



Adafruit DotStar LED Strip - APA102 Cool White - 30 LED/m
\$19.95 [ADD TO CART](#)



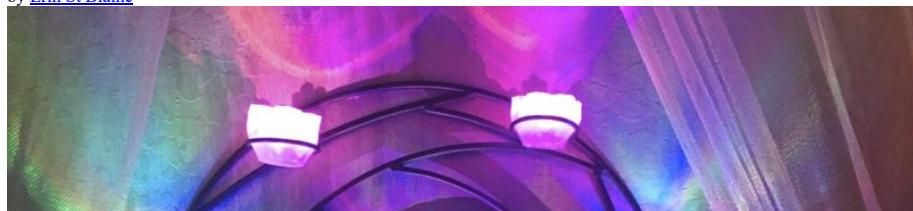
Adafruit DotStar LED Strip - APA102 Cool White - 60 LED/m

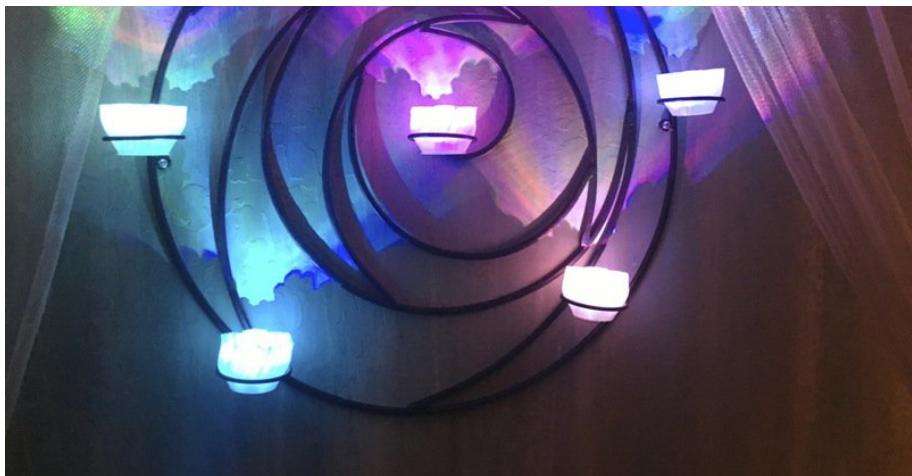
\$29.95 [OUT OF STOCK \(NOTIFY ME\)](#)

Adafruit DotStar LED Strip - APA102 Cool White - 144 LED/m

\$74.95 [OUT OF STOCK \(NOTIFY ME\)](#)[ADD ALL TO CART](#)

RELATED GUIDES

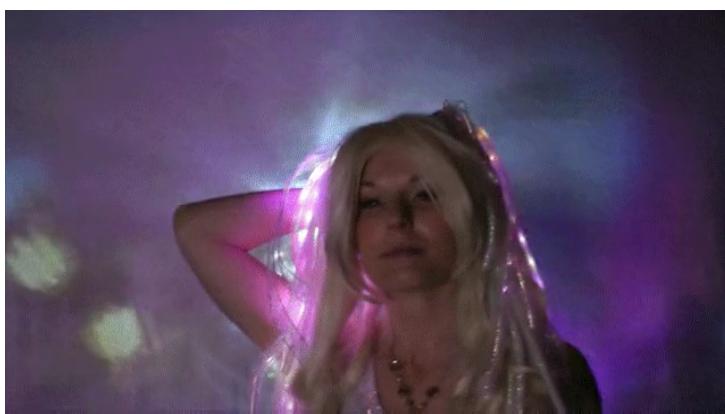
[Remote Controlled LED Tea Light Sconce](#)[Dotstars and IR control bring your wall to life](#)by [Erin St Blaine](#)



[Turn an old candelabra into a colorful 3d Printed LED display that will enhance the decor of your home or office. Change modes and brightness with a remote control, and harness the power of FastLED to add your own color animations.](#)

[**Pixie CyberWig - A Wearable WiFi LED Visualization**](#)

[Wireless video on your head](#)
by [Erin St Blaine](#)



[Dotstar LED cyber dreds wirelessly reflect the colors and movement on your computer screen. Use music visualizers to effortlessly project sound reactivity and color to your costume.](#)

[**LIGHTSHIP: LED Animation over WiFi**](#)

[Any sufficiently advanced technology is indistinguishable from magic.](#)
by [Phillip Burgess](#)



[Use the Adafruit Feather M0 WiFi board and DotStar LEDs to create mobile luminous art pieces — one or many — synchronized from your computer via wireless networking.](#)
[NEW](#)

[**Ladyada's Learn Arduino - Lesson #2**](#)

[Venture deep into the Blink sketch...](#)
by [lady ada](#)





OK you've gotten your Arduino set up and also figured out how to use the software to send sketches to the board. Powerful stuff! But...just running example sketches is a little boring. What we really want to do is use our own creativity and skill to write new sketches! That's what we'll be doing in this lesson. To start we will venture deep into the Blink sketch, looking at each line and trying to understand what its doing. Then we will start hacking the sketch, and maybe even meet an internationally-famous DJ and design custom hardware for him!

^

OUT OF STOCK NOTIFICATION

YOUR NAME
YOUR EMAIL

[NOTIFY ME](#)

- [CONTACT](#)
- [SUPPORT](#)
- [DISTRIBUTORS](#)
- [EDUCATORS](#)
- [JOBS](#)
- [FAQ](#)
- [SHIPPING & RETURNS](#)
- [TERMS OF SERVICE](#)
- [PRIVACY & LEGAL](#)
- [ABOUT US](#)

ENGINEERED IN NYC Adafruit ®

"Seek freedom and become captive of your desires. Seek discipline and find your liberty" - [Frank Herbert](#)

