- Search Adafruit
- SHOP
- BLOG
- LEARN
- FORUMS
- VIDEOS
- SIGN IN
- CLOSE MENU

0 Items
Sign In

- SHOP
- BLOG
- LEARN
- FORUMS
- VIDEOS

DotStar Pi Painter
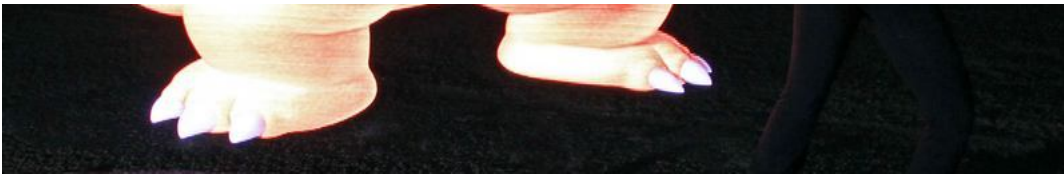
Ultimate light-painting rig!

Contributors

Phillip Burgess
Feedback? Corrections?
RASPBERRY PI LEDS / LED STRIPS

Raspberry Pi Setup

by Phillip Burgess

If using a single-USB-port system (Model A, A+ or Pi Zero), either use a USB hub so you can have keyboard and wireless attached during setup, or borrow a Model B/B+/Pi 2 for setup and then move the card over to the target system once finished.

This project works well with the **Raspbian "Lite"** operating system — a pared-down system that doesn't include any GUI apps. Don't bother with the full-featured Raspbian image for this project, it's *enormous!*

Fetch Raspbian Lite here:

Raspberry Pi Downloads Page

If you're new to Raspberry Pi and Linux, we strongly suggest working through the first few guides in the **Learn Raspberry Pi** tutorial series…know how to "burn" an SD image, perform a first-time setup and get the Raspberry Pi connected to a network. Some familiarity with one of the text editors (such as the simple *nano* or the more daunting *vi* or *emacs*) is also recommended.

Install the Raspbian Jessie Lite image on a **2GB or larger** microSD card. You'll need to connect a **monitor** and **USB keyboard** for basic system configuration, but this is only temporary…we'll set it up to run "headless" later. For networking, connect either an Ethernet cable or a USB WiFi adapter.

The first-time boot takes longer than usual; this is normal as the system does some self-configuration.

Log in as user "pi", password is "raspberry", then run the system configuration tool:

Copy Code

```
1.  sudo raspi-config
```

The following options are **required:**

- Expand Filesystem
- Under "Advanced Options," enable SPI and load kernel module by default

The following are optional but **recommended:**

- Change User Password
- Under "Internationalisation Options," Change Keyboard Layout. *If getting unexpected symbols when typing, this is why. I typically use Generic 104-key PC, English (US).*
- If your Pi will be on a network (wired or wireless), then under "Advanced Options" select:
  - Change Hostname to distinguish it from other Rasperry Pis on your network; mine is called "lightpaint."
  - Enable SSH for remote administration (this makes it easier to copy-and-paste the installation commands later).

You can further twiddle system settings to your liking. Tab over to the "Finish" button and **reboot** the system when prompted.

If you're setting up on a wired network, skip ahead to "Installing Packages" below. Otherwise, log in as "pi" again and let's set up WiFi…

Copy Code

```
1.  sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

Add the "network" section shown here, changing the network name and password to match your setup:

Copy Code

```
1.  ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
2.  update_config=1
3.  network={
4.          ssid="my-network-name"
5.          psk="my-network-password"
6.          scan_ssid=1
7.  }
```

Note: this file is *super-persnickety* about formatting. The lines in the network section should be indented with a single tab (not spaces). If the Pi won't join the WiFi network, it may be an extra space or two somewhere in this file.

Save changes and reboot the system to activate wireless networking.

Copy Code

```
1.  sudo reboot
```

WiFi is handy for setup and administration, but the LED on most wireless adapters will create a light streak when painting; remove it for photography or cover the LED with tape.

Install Packages

With ssh enabled, you can use a terminal program and log into the system remotely at *lightpaint.local* or whatever name you've given the system. This is preferable as you can copy-and-paste commands directly. Otherwise, you can log in with screen and keyboard as usual, just be very mindful of spelling.

Lets install a few prerequisite packages and fetch the light-painting software:

Copy Code

```
1. sudo apt-get update
2. sudo apt-get install git usbmount python-dev python-imaging python-pip
3. sudo pip install evdev
4. git clone https://github.com/adafruit/Adafruit_DotStar_Pi
5. git clone https://github.com/adafruit/DotStarPiPainter
```

usbmount is a package for sensing when USB drives are plugged in or removed. We need to install a couple of scripts so that it can inform our light-painter code of these events:

Copy Code

```
1. cd DotStarPiPainter
2. sudo cp 99_lightpaint_mount /etc/usbmount/mount.d
3. sudo cp 99_lightpaint_umount /etc/usbmount/umount.d
```

Dry Run

You don't need any LEDs connected yet, let's just make sure all of the software parts are working as expected. Start the light-painting program with:

Copy Code

```
1. sudo python DotStarPiPainter.py
```

If all goes well, *nothing should happen*...it's waiting for a USB drive to be inserted. Plug in a USB flash drive with one or more images on it, and after a moment you should see it print a message that it's loading and processing an image. Fantastic! Press Control+C to break out of the program and we'll do some final setup.

If the program throws an error, or if it doesn't detect the USB drive:

- Usually it's a missing package. DId you do the entire *apt-get* and *pip* commands above?
- Are the *usbmount* scripts installed in the correct locations?
- If it complains about the *dotstar.so* file, you may need to recompile that (type "make" in the *Adafruit_DotStar_Pi* directory) and copy it to this directory.

Some USB flash drives have a power-on LED. Cover the LED with tape, or switch to a "dark" drive with no light.

Success!

Let's set up the Pi to run fully headless now.

I recommend installing our *gpio-halt* utility. This tiny program lets you connect a button that performs an orderly system shutdown. (Just pulling the plug on a Linux system is a bad idea and can corrupt the SD card, so you'd have to start over.)

Copy Code

```
1. git clone https://github.com/adafruit/Adafruit-GPIO-Halt
2. cd Adafruit-GPIO-Halt
3. make
4. sudo make install
```

Then edit the rc.local file to start up our code automatically at boot time:

Copy Code

```
1. sudo nano /etc/rc.local
```

Just before final "exit 0" line, insert this line:

Copy Code

```
1. /usr/local/bin/gpio-halt 21 &
```

Change the "21" to whatever GPIO pin your shutdown button is connected to (the other leg of the button should connect to a ground pin). On Pi models with the **40-pin header, GPIO 21** is at the very end (nearest the USB ports) with an adjacent ground pin, so it's very convenient to connect a button across these two pins. On older Pi models with a **28-pin header, GPIO 7** is similarly situated at the end of the header:
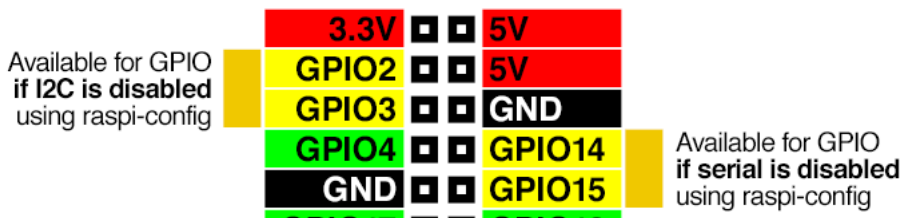
	GPIO17		GPIO18
	GPIO27		GND
	GPIO22		GPIO23
	3.3V		GPIO24
Used by DotStars GPIO unavailable	GPIO10		GND
	GPIO9		GPIO25
	GPIO11		GPIO8
	GND		GPIO7
	DNC		DNC
GPIO Availability:	GPIO5		GND
No	GPIO6		GPIO12
Maybe	GPIO13		GND
Yes	GPIO19		GPIO16
	GPIO26		GPIO20
	GND		GPIO21

is above this line are present on all Raspberry Pi boards

Pins below line on Models A+, B+ and Pi 2

This will take effect on the next reboot. Then, to shutdown, tap the button (or jumper across the pins with a screwdriver if no button installed yet) then wait at least 15 seconds before disconnecting power. Test it once with a monitor connected to confirm that it's working before running the system headless.

[OVERVIEW ASSEMBLY PART 1](#)
Last updated on 2015-12-16 at 03.24.36 PM Published on 2015-12-11 at 01.35.28 PM

Adafruit DotStar Digital LED Strip - White 144 LED/m - One Meter
$74.95 [Add To Cart](#)

Raspberry Pi 2 - Model B - ARMv7 with 1G RAM
$39.95 Add To Cart

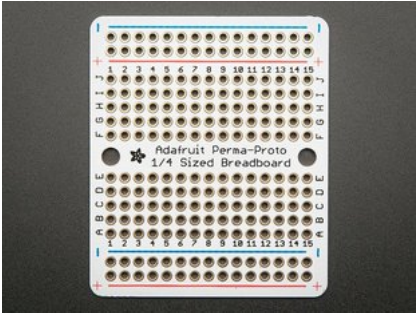Raspberry Pi Model A+ 512MB RAM
$24.95 Add To Cart

Adafruit Perma-Proto HAT for Pi Mini Kit - No EEPROM
$4.95 Add To Cart

Adafruit Perma-Proto Quarter-sized Breadboard PCB - Single
$2.95 Add To Cart

74AHCT125 - Quad Level-Shifter (3V to 5V)
$1.50 Add To Cart

Colorful Square Tactile Button Switch Assortment - 15 pack
$5.95 Add To Cart

USB Battery Pack for Raspberry Pi - 10000mAh - 2 x 5V outputs
$39.95 Add To Cart

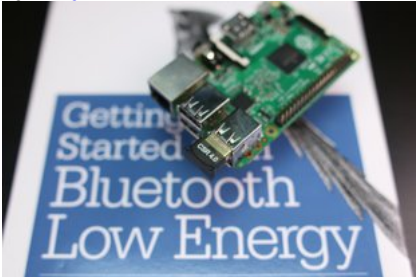USB DIY Connector Shell - Type A Male Plug
$0.95 Add To Cart
ADD ALL TO CART

RELATED GUIDES

Install bluez on the Raspberry Pi

How to compile and install bluez, the Linux Bluetooth classic & low energy system, on the Raspberry Pi.
by Tony DiCola

Learn how to download and install the latest version of the bluez Bluetooth library on the Raspberry Pi. Start using your Pi with Bluetooth classic & low energy in no time!
POPULAR

5" Display Kippah Portable Raspberry Pi
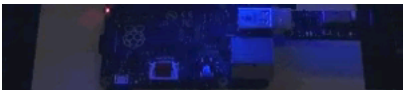
Make a portable Raspberry Pi
by Ruiz Brothers

Build a portable Raspberry Pi with a 5" display.

Reverse Engineering a Bluetooth Low Energy Light Bulb

Control a Bluetooth Low Energy device with your own code!
by Tony DiCola

Learn how to sniff Bluetooth low energy traffic using the Bluefruit LE sniffer and reverse engineer a simple Bluetooth low energy device like a light bulb. Then control the device yourself using a Raspberry Pi and Bluetooth low energy USB adapter!

NextBus transit clock for Raspberry Pi

Should I stay or should I go now?
by Phillip Burgess
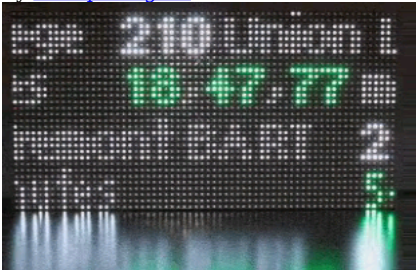
For transit-bound people, NextBus — a free internet service providing arrival data for over 135 transit agencies — is a tremendous convenience. Knowing when a bus is due means less standing out in the rain...one can use that time inside to get a little extra work done, or finish that cup of coffee. Using the Raspberry Pi computer, we'll create a dedicated NextBus "appliance" that informs at a glance.
×

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 ®
"Achieving a goal provides immediate satisfaction; the process of achieving a goal is a lasting pleasure" - Evelyn Berezin