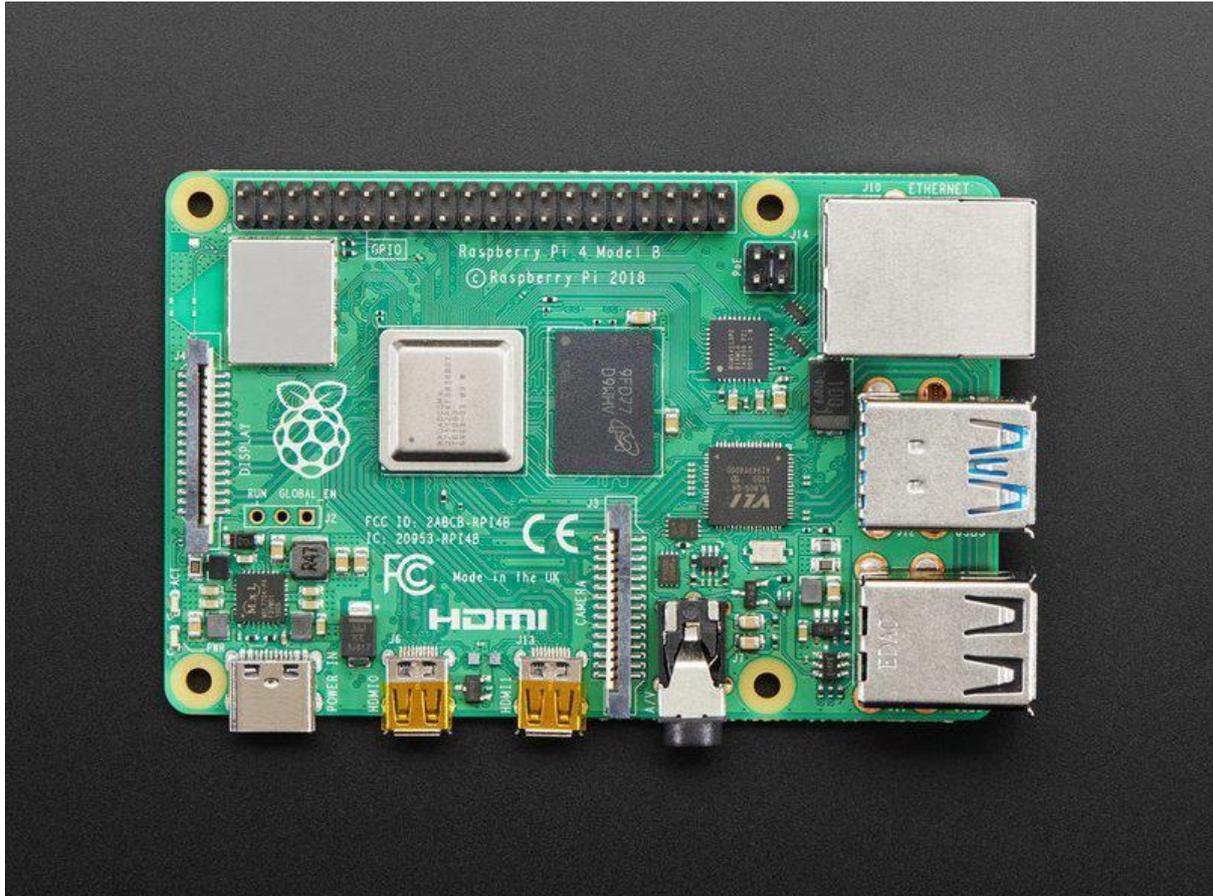




Raspberry Pi Computer Quick-Start

Created by lady ada



<https://learn.adafruit.com/raspberry-pi-computer-quick-start>

Last updated on 2023-08-29 04:26:12 PM EDT

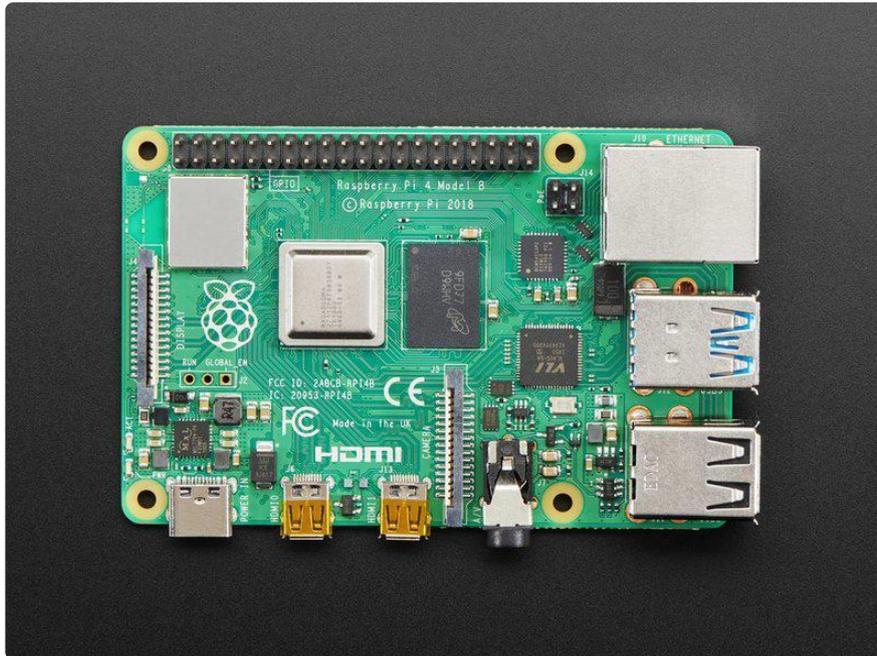
Table of Contents

Raspberry Pi Quick Setup

3

- [SD Card Prep](#)
- [Power to the Pi](#)
- [Ping](#)
- [SSH](#)
- [Change the Password!](#)
- [Rename the Pi](#)
- [Update Application List](#)
- [Software Setup](#)
- [Make sure you're using Python 3!](#)
- [Install Blinka](#)

Raspberry Pi Quick Setup



This quick-start guide will get you ready to connect and run all sorts of Adafruit projects on your Raspberry Pi computer

You will need:

- [A modern Raspberry Pi computer \(\)](#), we recommend the 'full' size Pi 3 B+ and later, that come with Ethernet and WiFi built in
- Matching power supply ([USB C type for Pi 4 and later \(\)](#), or [Micro USB type for Pi 3 B+, Pi Zeros \(\)](#) or earlier)
- [8 GB or larger micro SD card \(\)](#) and USB reader/writer

SD Card Prep

First we need to perform the basic set up the Raspberry Pi. These are the first steps:

- download the latest Raspbian desktop operating system
- burn the OS to an SD card
- enable SSH
- add WiFi settings and credentials

For details on doing these things, [follow the first four pages of this guide \(\)](#) to get your Pi setup with Raspbian Lite (for 'headless' or remote operation) or NooBs (for graphical / desktop use), WiFi, and ssh enabled.

Once the SD card is prepared, come back here so we can continue.

Power to the Pi

Safely eject the SD card from your main computer. Then, with the Pi unplugged from power, insert the SD card into the Pi's SD card slot.

Then, plug in the power to the Pi. You should see some LEDs light up.

From here forward, we'll be connecting to the Pi remotely from our main computer. Make sure your computer is on the same WiFi network as the Pi.

Ping

After a minute or two, you can try and ping the Pi to see if it has connected to your network. Open a terminal on your computer, such as Terminal.app on a mac (Windows users will require some additional setup. [Read here \(\)](#)), and type this and press return:

```
ping -c 3 raspberrypi.local
```

You should see an output similar to this:

```
jep@machine ~ % ping -c 3 raspberrypi.local
PING raspberrypi.local (192.168.1.68): 56 data bytes
64 bytes from 192.168.1.68: icmp_seq=0 ttl=64 time=14.993 ms
64 bytes from 192.168.1.68: icmp_seq=1 ttl=64 time=20.565 ms
64 bytes from 192.168.1.68: icmp_seq=2 ttl=64 time=50.571 ms

--- raspberrypi.local ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 14.993/28.710/50.571/15.625 ms
```

This tells us that the Pi is on the network, and has an address of 192.168.1.68.

SSH

We can now connect to it through the secure shell (SSH) so we can control it remotely.

[Here's some great info on SSH \(\)](#) from the Raspberry Pi Foundation.

Type the following in your terminal:

```
ssh pi@raspberrypi.local
```

or

```
ssh pi@192.168.1.68
```

 (or whatever address you saw returned earlier when you pinged it.)

You may see a message warning you that the authenticity of the host can't be established, Are you sure you want to continue connecting (yes/no)?

To this question you can reply by typing `yes` and pressing return. This will add the Pi to your list of known hosts and should not pop up again with this question.

You will now see the password prompt for the Pi:

```
pi@raspberrypi.local's password:
```

Go ahead and type in the default password, which is: `raspberry`

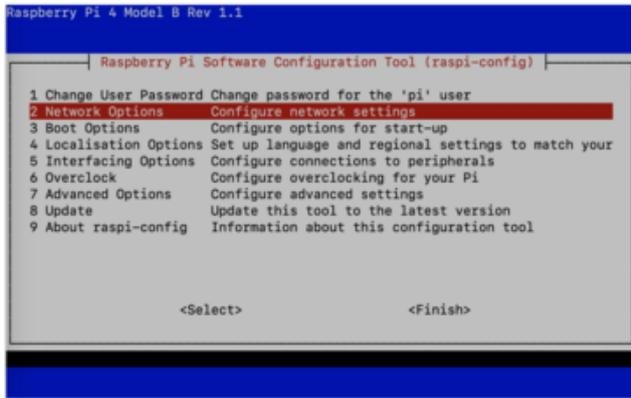
Change the Password!

You will now be logged into the Pi over ssh! There will be a warning about changing your password from the default and instructions on how to do so. Do it now! Type `passwd`, hit return and follow the instructions to create a new, secure, non-default password!

Rename the Pi

You may end up with multiple Pi computers on your network, which can get confusing if they are all named raspberrypi. To rename this one for this project, type in `sudo raspi-config` and press return.

`sudo`? What's `sudo`? It stands for "superuser do", and it's a program that lets you run powerful commands which are normally restricted for reasons of safety and security. Or just to prevent accidents!



Use the arrows and enter key on your keyboard to navigate. Pick 2 Network Options and press return on your keyboard.

Then, pick N1 Hostname.

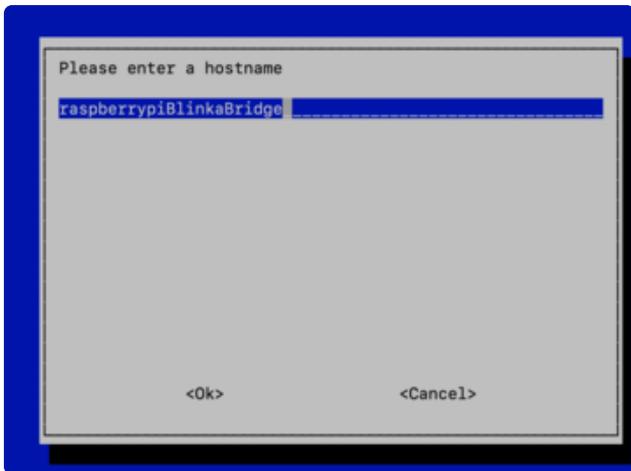
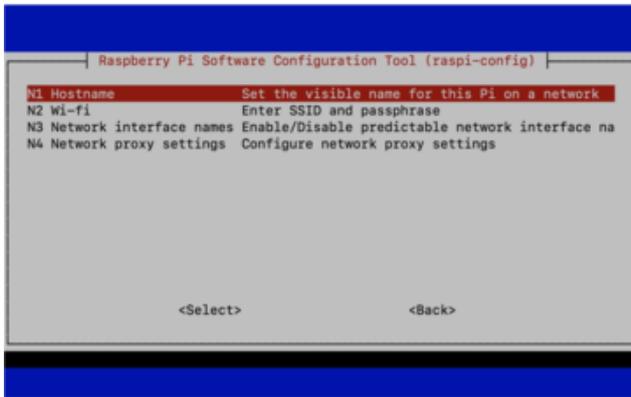
Enter a new hostname -- I typed

`raspberrypiBlinkaBridge`

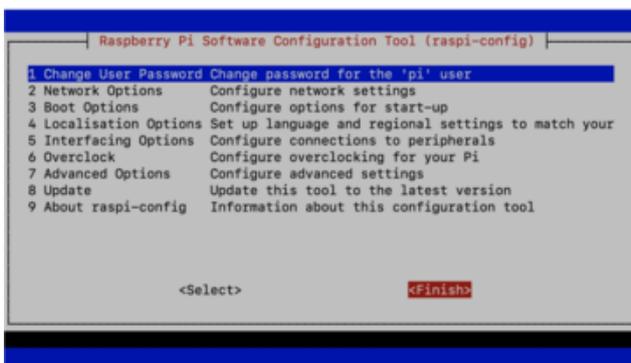
Use the Tab key to get to the <Ok> button and press return.

Press tab twice to get to the <Finish> button and press return.

Now, we'll restart the Pi to make the changed host name show up on the network by typing `sudo reboot` and pressing return.



Note, this will close the SSH session. After the Pi restarts, wait about a minute, and then reconnect over SSH. This time, you'll use the new name: `ssh pi@raspberrypiBlinkaBridge`



Update Application List

Once logged into the Pi, we'll do some software updates to make sure we have the latest of everything.

We really really recommend the latest Raspbian only. If you have an older Raspbian install, run "sudo apt-get update" and "sudo apt-get upgrade" to get the latest OS!

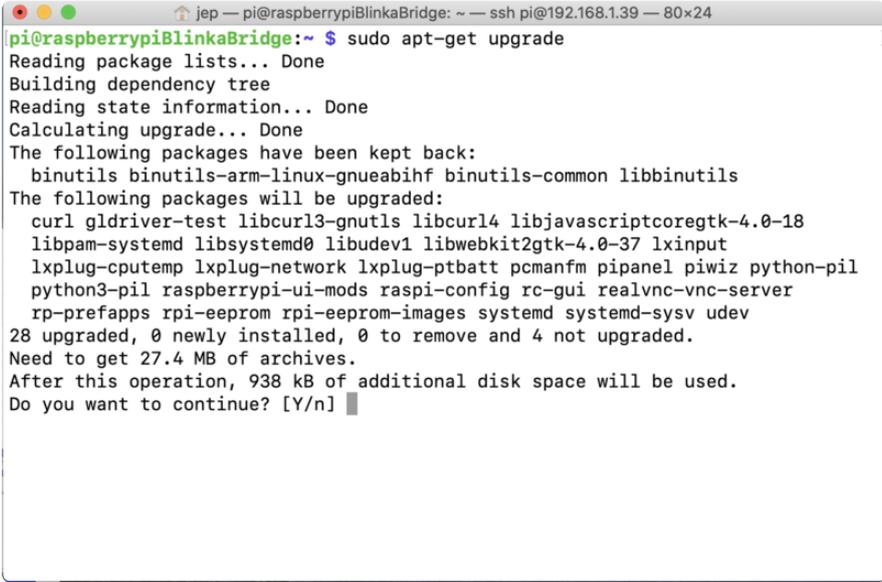
Type:

```
sudo apt-get update
```

And then press return. This will run for a little bit as it reads the available package lists, and then finish, reporting that it is done.

You can now apply any of these changes by typing:

```
sudo apt-get upgrade
```



```
pi@raspberrypiBlinkaBridge:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
 binutils binutils-arm-linux-gnueabi binutils-common libbinutils
The following packages will be upgraded:
 curl gldriver-test libcurl3-gnutls libcurl4 libjavascriptcoregtk-4.0-18
 libpam-systemd libsystemd0 libudev1 libwebkit2gtk-4.0-37 lxinput
 lxplug-cputemp lxplug-network lxplug-ptbatt pcmanfm pipanel piwiz python-pil
 python3-pil raspberrypi-ui-mods raspi-config rc-gui realvnc-vnc-server
 rp-prefapps rpi-eeprom rpi-eeprom-images systemd systemd-sysv udev
28 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
Need to get 27.4 MB of archives.
After this operation, 938 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Type **Y** and press enter to continue.

Software Setup

Now that the Pi is updated and renamed on the network, we'll make sure we have Python 3 setup, as Python 2 is no longer used or supported.

pip3, is the software package installer we'll use. Let's upgrade it to the latest version with this command:

```
sudo pip3 install --upgrade setuptools
```

If above doesn't work try

```
sudo apt-get install python3-pip
```

Once that has finished you'll be returned to the prompt.

Make sure you're using Python 3!

The default Python on your computer may not be Python 3. Python 2 is officially discontinued and all our libraries are Python 3 only.

We'll be using `python3` and `pip3` in our commands, use those versions of Python and pip to make sure you're using 3 and not 2.

Install Blinka

Blinka is the software that allows us to run CircuitPython and its libraries on Linux. You can [read more about it here](#) ().

To install blinka, type:

```
sudo pip3 install adafruit-blinka
```

That's it! You can reboot again if you like.