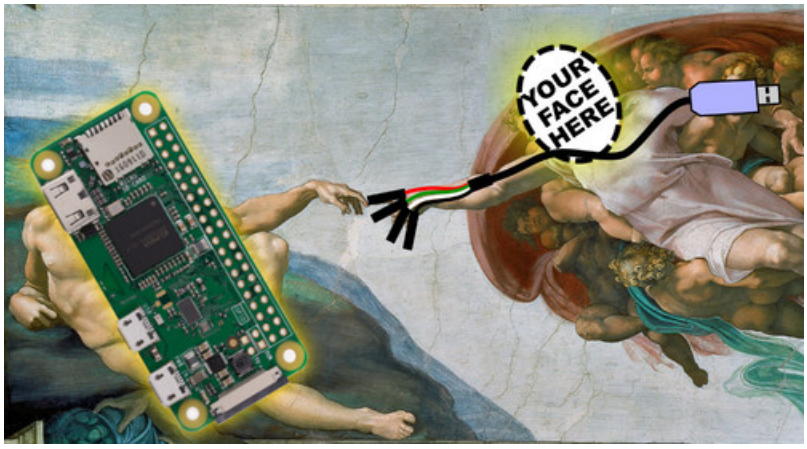
adafruit learning system

Raspberry Pi Zero Headless Quick Start

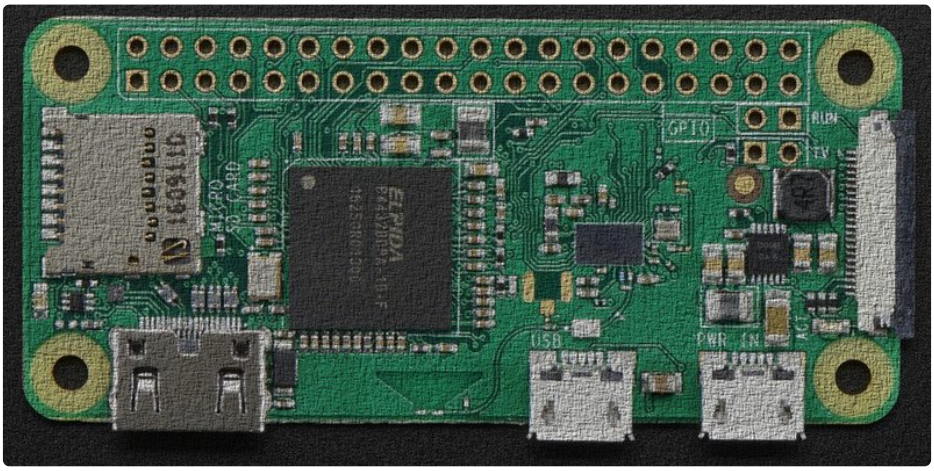
Created by Carter Nelson



Last updated on 2021-06-30 09:47:55 AM EDT

Guide Contents

Overview



This guide shows how to bring up a Raspberry Pi Zero or Zero W without needing to attach a keyboard/mouse/monitor. Basic settings will be configured by editing text files directly on the SD card using an editor on your main PC prior to first boot.

Latest OS tested: Raspbian Buster Lite 2019-06-20

In this guide the term "Pi Zero" will be used for both the Zero and the Zero W.

Suggested Items

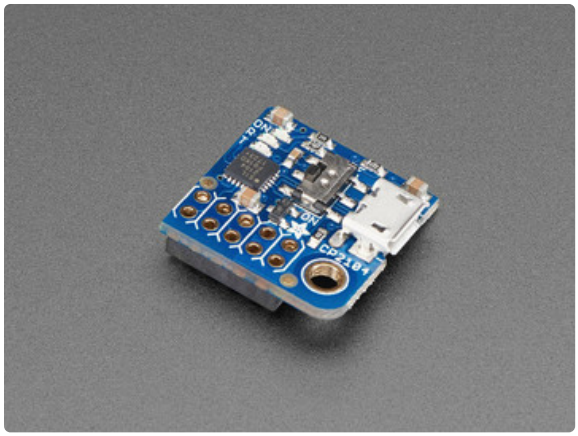
If you need to troubleshoot for any reason, this item is your best friend. For $10, it makes working with a headless Pi Zero so much easier. It's also included with a lot of the kits.

USB console cable. (https://adafru.it/dDd)

Don't leave home without it.

This is a slightly fancier option specifically made for the Pi. It has the convenience of allowing you to use a standard micro USB cable. It was mainly designed for attaching to male headers, so if you've soldered on

something different you'll have to find some way to jumper it.



Adafruit PiUART - USB Console and Power Add-on for Raspberry Pi (https://adafru.it/zib)

Of course you'll also need an SD card for the operating system, a Pi Zero (plus WiFi adapter (https://adafru.it/elp)) or Zero W (https://adafru.it/vla), and some form of header pins to access the GPIO.

- Male Headers (https://adafru.it/vlb)
- Female Headers (https://adafru.it/vlc)
- 90deg Female Headers (https://adafru.it/vld)

But I Want To Connect A Keyboard/Mouse/Monitor

That's fine but not covered in this guide. Troubleshooting HDMI issues, configuring display settings, and dealing with possible HDMI and/or USB cable and/or power problems are worth a separate guide.

Other Pi Models

This guide is targeted for the Pi Zero and Zero W as those models are more difficult to attach a keyboard / mouse / and monitor to. However, these setup instructions should also work on any Pi with WiFi. It has been tested to work with:

- Pi 3 Model B
- Pi 3 Model B+
- Pi 3 Model A+
- Pi 4 Model B

Is There Even Life?

You can skip this section unless you have reason to believe your Pi Zero isn't alive.

THE ZERO DOES NOT HAVE A POWER LED

The Pi Zero doesn't have much in the way of blinky LEDs to give you a warm fuzzy that it's doing anything or even alive. And if the GPU doesn't find a valid OS image, it doesn't even turn on the green ACT LED and looks totally dead. Typically this just means something is up with the SD card. Bad card. Bad image. Out of date image. Whatever. **It does not mean the Pi Zero is dead.**

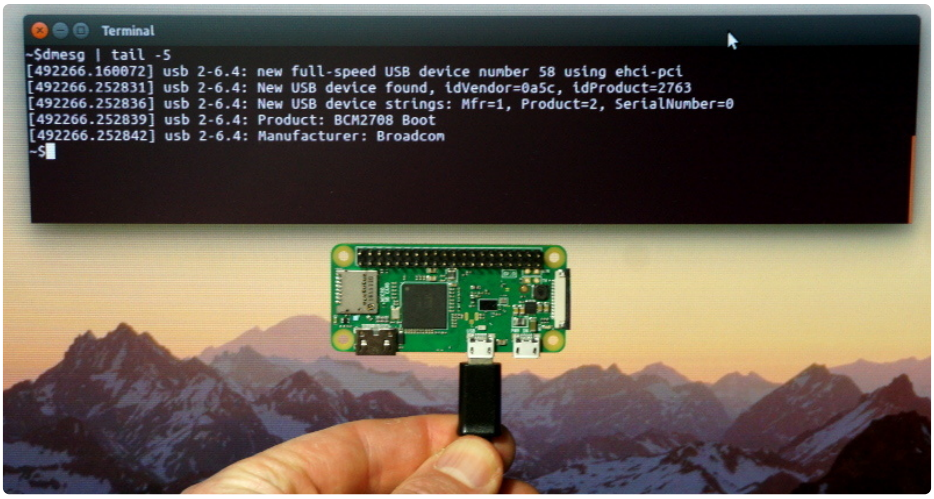
Here's how to run a sanity check to verify if the Pi Zero is OK.

(taken from here (https://adafru.it/upa) and also provided here (https://adafru.it/vle))

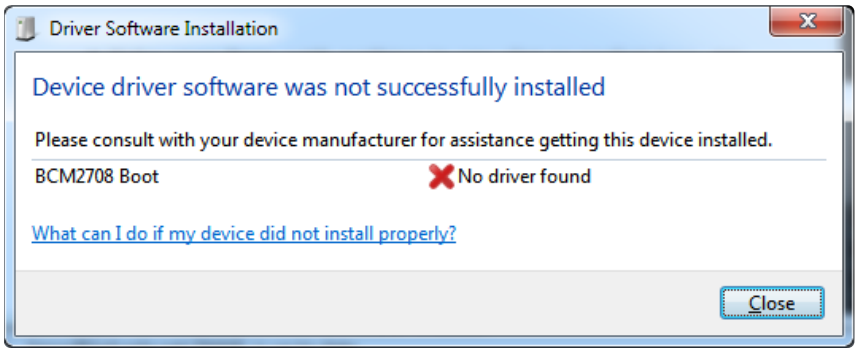
- Take your Zero, with nothing in any slot or socket (yes, **no SD-card is needed** or wanted to do this test!).
- Take a normal micro-USB to USB-A **DATA SYNC** cable (not a charge-only cable! make sure its a true data sync cable!)
- Connect the USB cable to your PC, plugging the micro-USB into the Pi's USB, (***not the PWR_IN***).
- If the Zero is alive, your Windows PC will go ding for the presence of new hardware & you should see "BCM2708 Boot" in Device Manager.
- Or on linux, run **sudo lsusb** or run **dmesg** and look for a `ID 0a5c:2763 Broadcom Corp` message. If you see that, so far so good, you know the Zero's not dead.

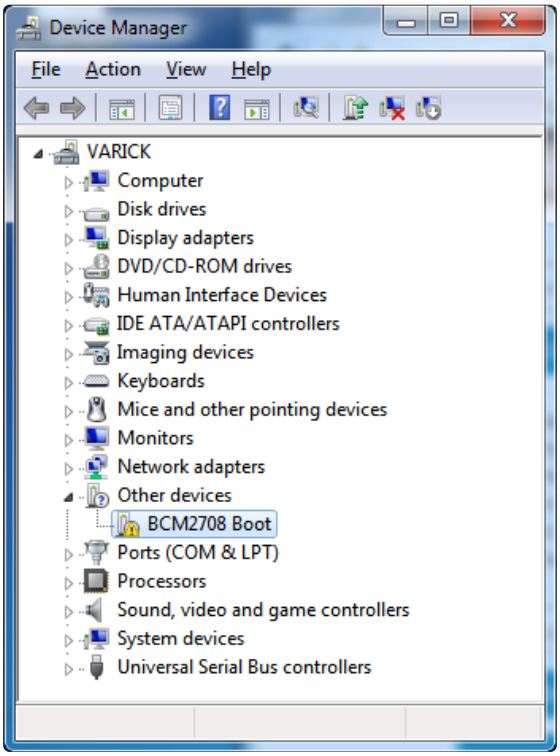
It may take a few seconds for the messages to show up.

Below is a Pi Zero connected to a Linux computer via a USB cable and the resulting dmesg output. **Note: there is no SD card installed, USB cable is in USB port, and there are no lights.**



Here's what our Windows machine showed:





Looks dead, but it's not.

Install OS on to SD Card

This guide uses Raspbian Lite as the starting point. Download the latest version from here:

https://adafru.it/fQi

https://adafru.it/fQi

You will get a .zip file. Unzip that and you should get a .img file. Then follow these excellent instructions to burn the OS image to an SD card:

https://adafru.it/jd0

https://adafru.it/jd0

You live in a world where an entire operating system can fit on a wafer thin piece of plastic smaller than your finger tip. And you can run this on a $5 (or $10) computer that is small enough to give away on the cover of a magazine (https://adafru.it/vlf). You should be amazed, excited, and happy about this.

Text File Editing

If you burned the OS image successfully, you should see a folder named **boot** appear on your main computer. If it doesn't, try removing and reinserting the SD card. If it still doesn't, try burning the image again.

There are three text files we will create/edit in **boot**.

1. **wpa_supplicant.conf** - wifi settings
2. **config.txt** - global system settings
3. **ssh** - an empty text file to enable ssh

We are going to edit these directly on the SD card before putting it in the Pi Zero. This way you can edit these files using your favorite text editor on your main computer. Try to avoid using a word processor though.

Configure WiFi

The Pi Zero W has built in WiFi so nothing additional is required. **WiFi is only supported on 2.4 GHz band**. If you are using the original Pi Zero, you will need some form of WiFi Adapter (https://adafru.it/elp) and a way to connect it: cable (https://adafru.it/we0) or adapter (https://adafru.it/sTD).

The WiFi configuration file does not exist and needs to be created. The name of the file should be **wpa_supplicant.conf** and its contents will get copied to the system folder at boot time. It will then be deleted. So this a one time only process. If you want to try again, you have to recreate the file and reboot.

The contents of the file should be as shown below. Replace **YOURSSID** and **YOURPASSWORD** with whatever is used for your network setup.

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=US

network={
    ssid="YOURSSID"
    psk="YOURPASSWORD"
    scan_ssid=1
}
```

Don't put any spaces around the = in this file. It doesn't like that.

Like this:

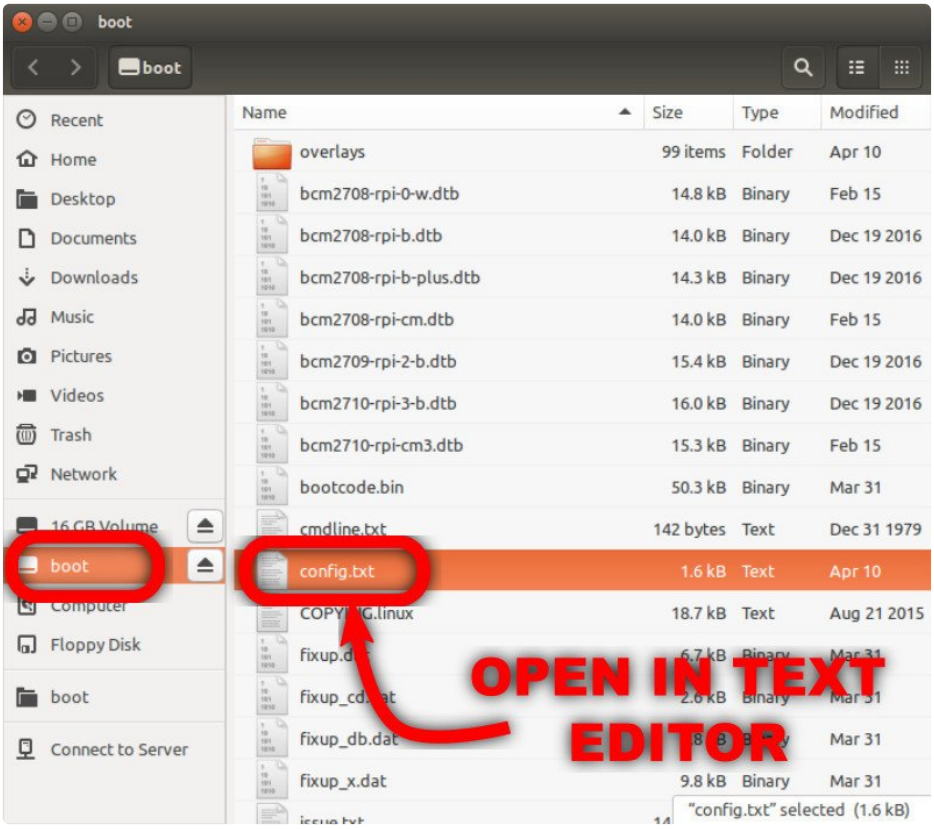
https://adafru.it/B1R

https://adafru.it/B1R

Save file and move on to next step. (more details here (https://adafru.it/yPd))

Enable UART

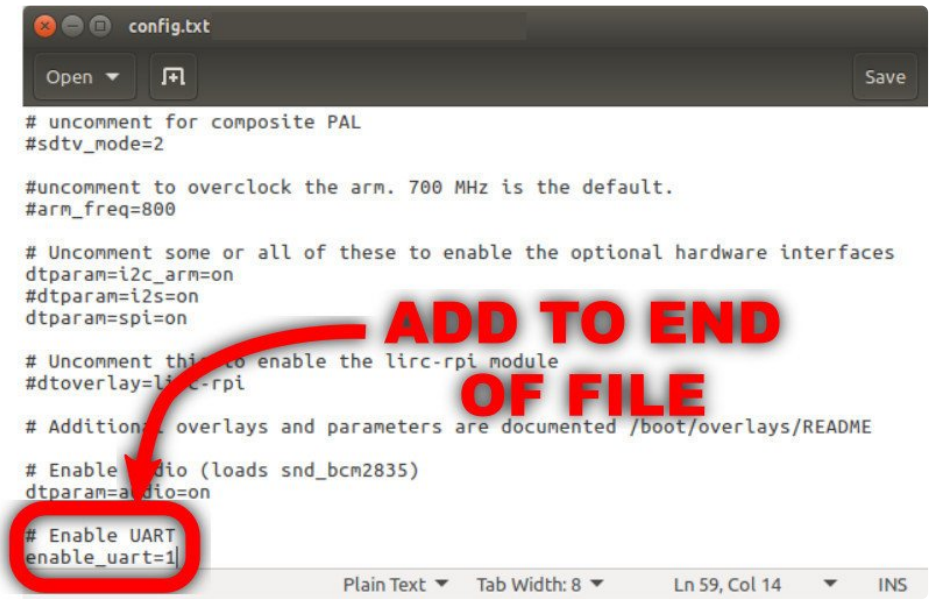
The file called **config.txt** already exists, we are just going to edit its contents. We will add some text to the bottom to enable the UART on the GPIO header pins. This allows a USB console cable to be attached later for troubleshooting.



Open the file in a text editor and add the following text to the bottom.

```
# Enable UART
enable_uart=1
```

Like this:



Save it and move on to next step. (more details here (https://adafru.it/upa))

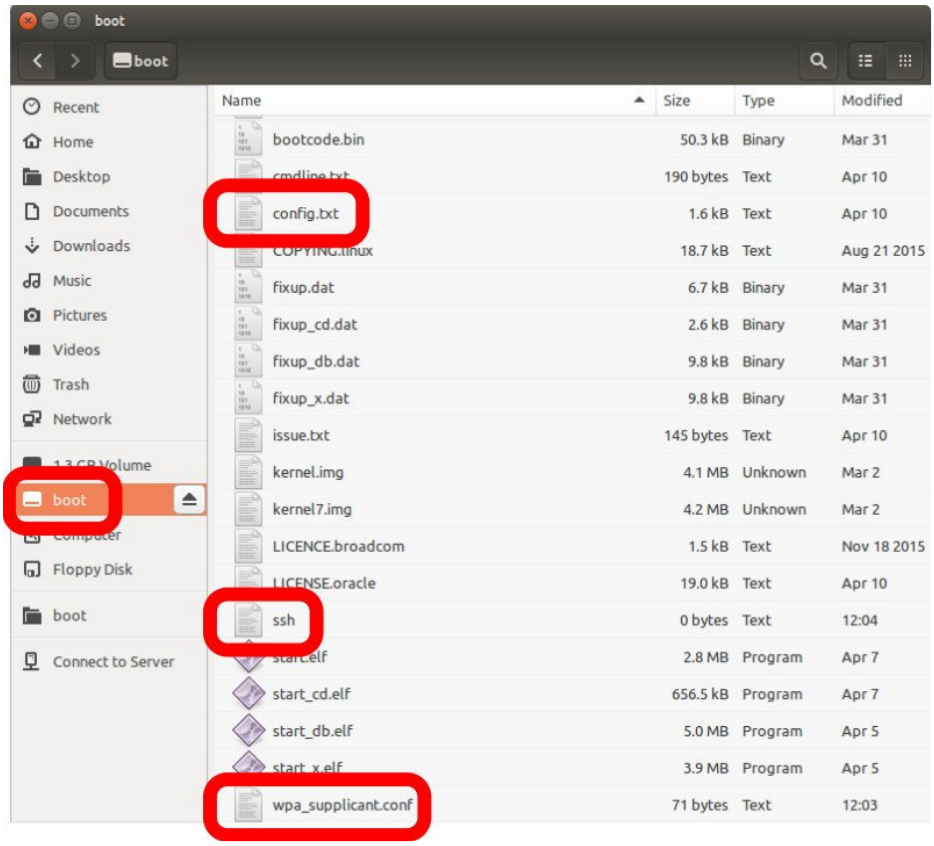
Enable SSH

SSH used to be enabled by default, but was then (Nov 2016) turned off by default. This was due to security concerns since the pi user id and password are well known. However, you'll likely want this enabled so you can remotely connect to the Pi Zero.

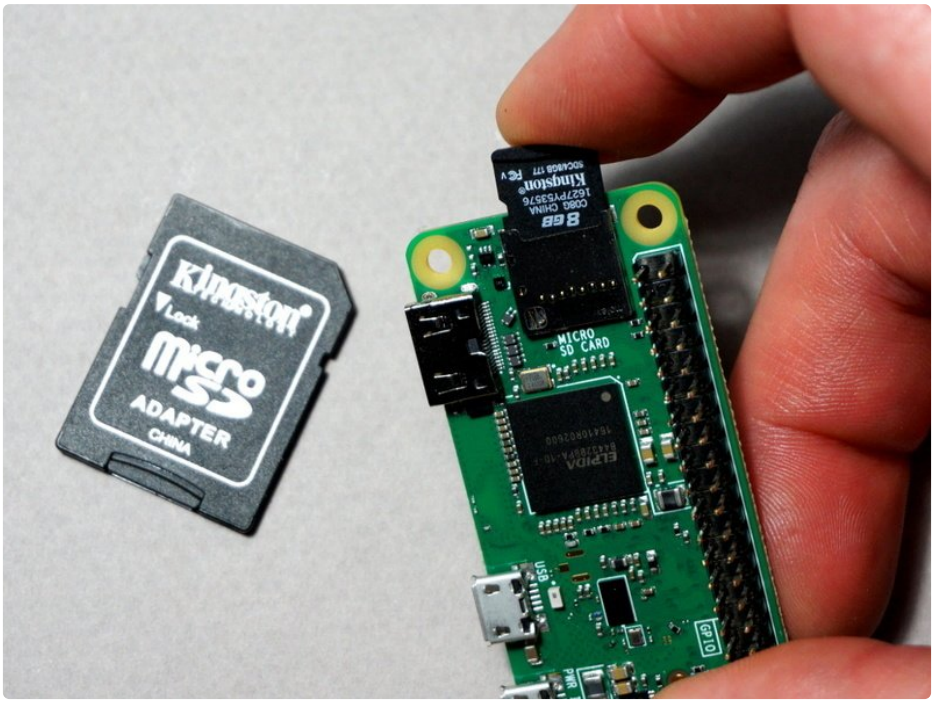
To do so, we simply create a file called **ssh**. This file does not exist and needs to be created. It can be empty. The system looks for it at boot time and will enable ssh if it is there. It is then deleted. So just create a new file and save it as **ssh** to the **boot** folder.

Final Check

After the above steps, you should have the following files on the SD card in the **boot** folder.



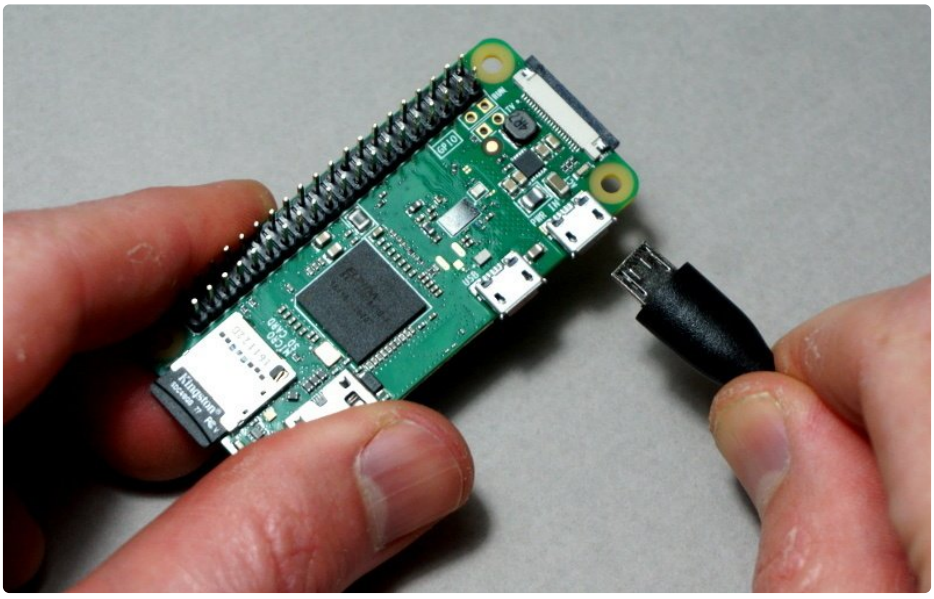
Safely remove the SD card from the main computer and install it in the Pi Zero.



Give It Life

OK. Let's power it up.

With the SD card inserted, apply power via a USB cable to the **PWR IN** connector as shown.



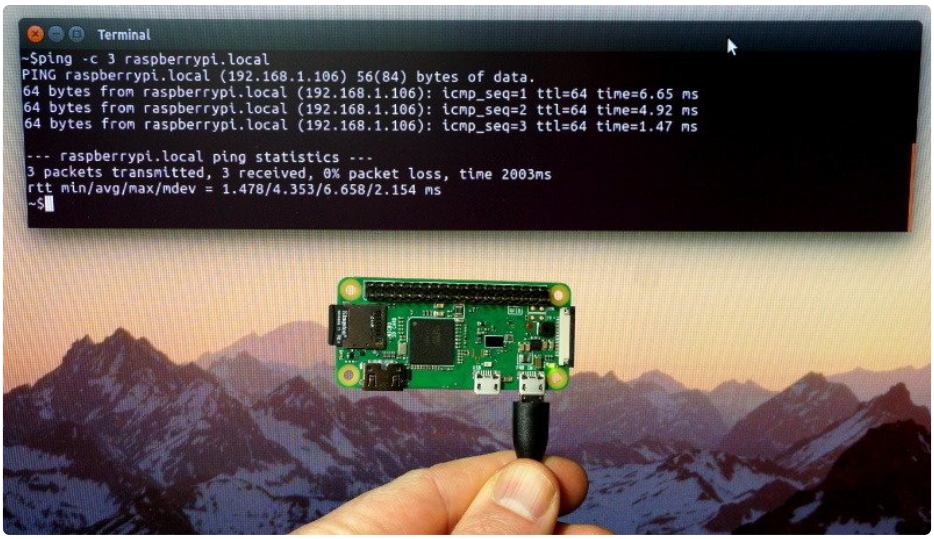
You should see some activity on the green LED light. This means the Pi Zero found a good OS image and is booting.

After a minute or two, you can try and ping the Pi Zero to see if it has connected to your network. You can access the Pi Zero using mDNS style addressing.

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

Windows users will require some additional setup. Read here (https://adafru.it/q1e). Also, the ping options on Windows are different, so the above would be:

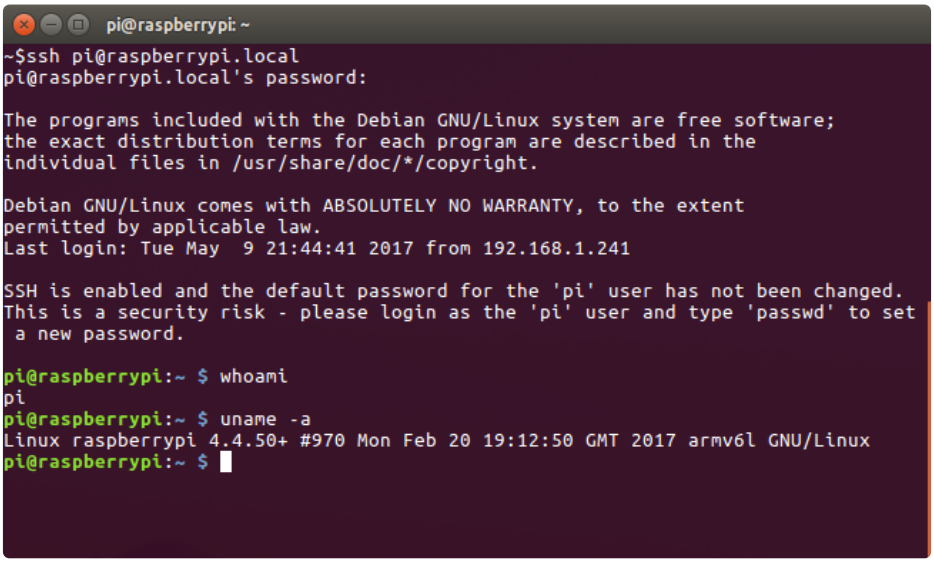
```
ping -n 3 raspberrypi.local
```



You should also be able to **ssh** into the Pi Zero.

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

The default password is **raspberry**.



Done?

If that worked, then you're pretty much done. The Pi Zero has connected to your network, and assuming your network is connected to the Internet, so is the Pi Zero. If you want, you can read the Suggested Initial Setup (https://adafru.it/wvE) section for some suggested first steps.

If that did not work, it's time to troubleshoot. Go to the Using A Console Cable to Troubleshoot (https://adafru.it/wvF) section.

Suggested Initial Setup

Where you go next is entirely up to you and what your plans are. However, it's a good idea to start by running a system update first. Connect to the pi via ssh and run the following commands:

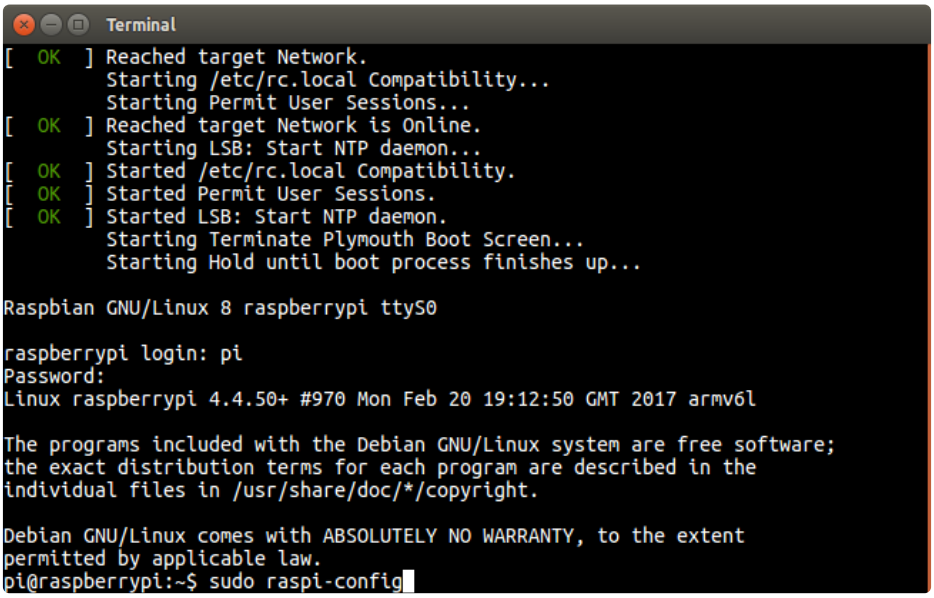
```
sudo apt-get update
sudo apt-get upgrade
```

Both commands may take a while to complete. Raspbian Lite is a pretty minimal install, so your next steps are probably to install a bunch of packages. By running the above first, you will make sure to bring in the most up to date versions.

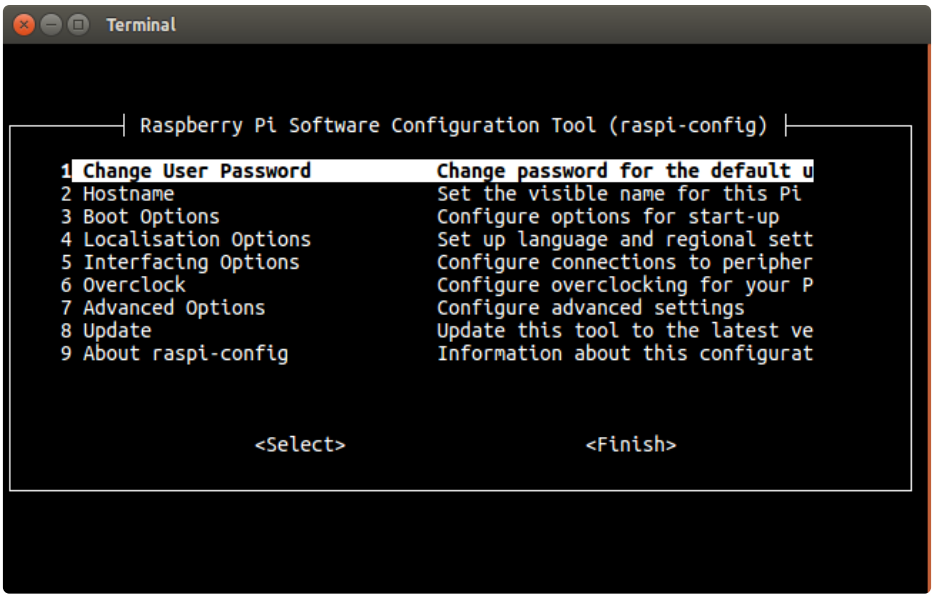
System Configuration

General system configuration is done by running the **raspi-config** utility.

```
sudo raspi-config
```



This will bring up the main menu.



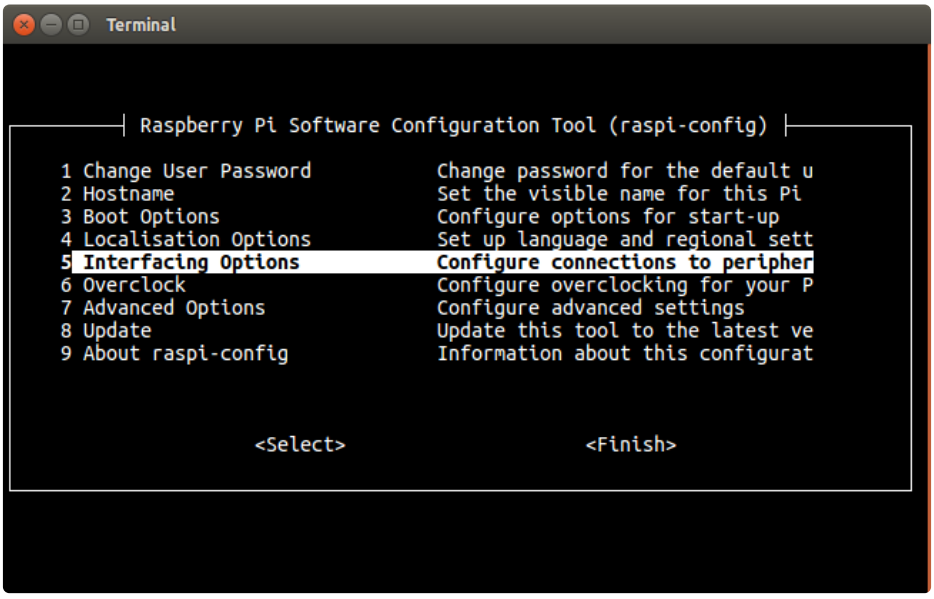
This is a good time to change the password to something other than the default.

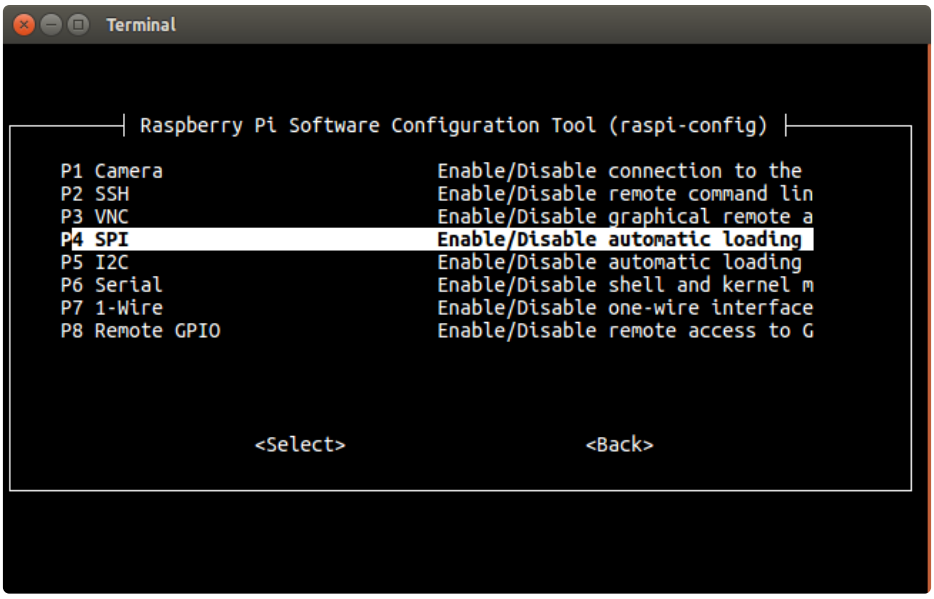
You can also do other things like change the time zone, keyboard layout, hostname, etc.

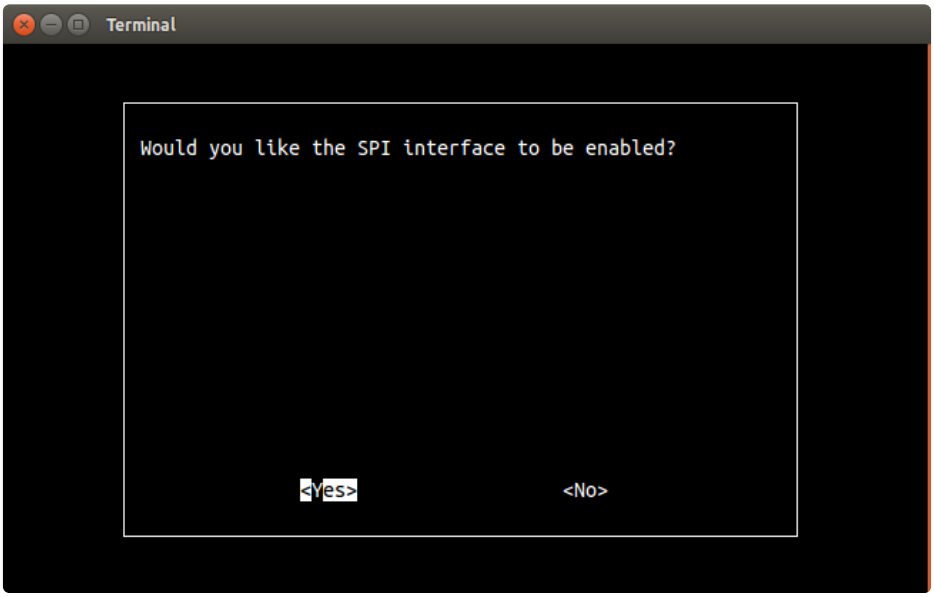
Enable SPI and I2C

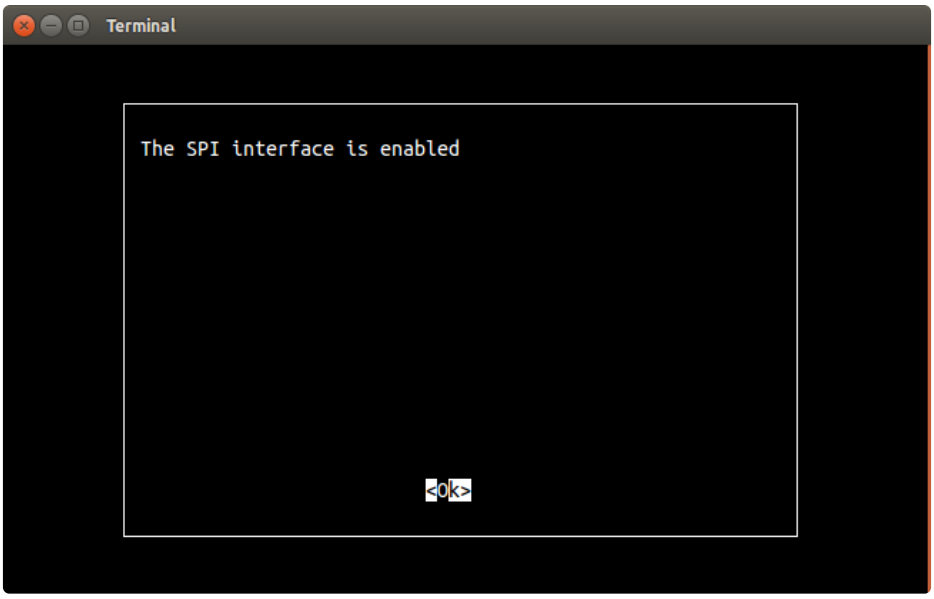
These are both used in a lot of projects but are disabled by default. It's a good idea to enable them now so you don't waste time later wondering why it's not working.

SPI

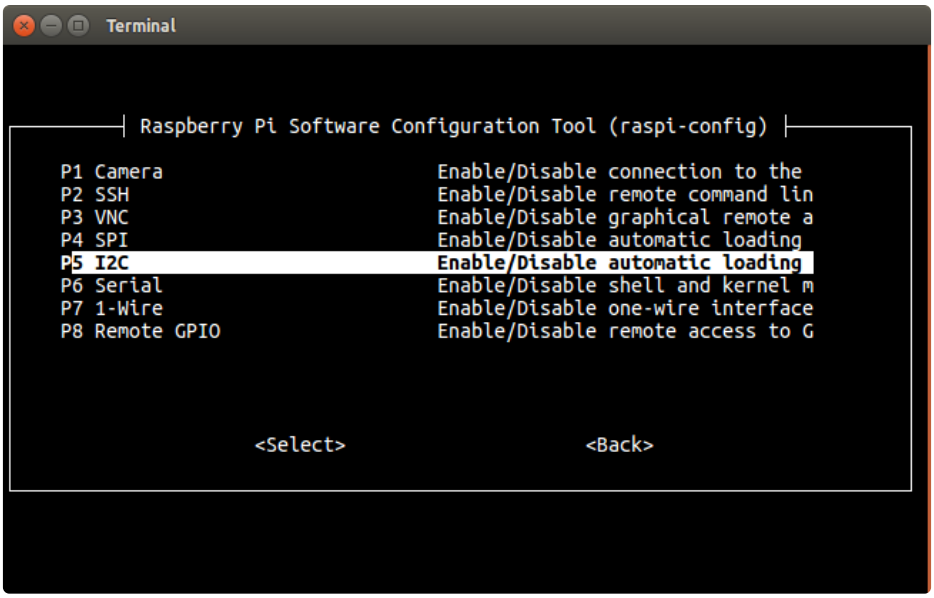


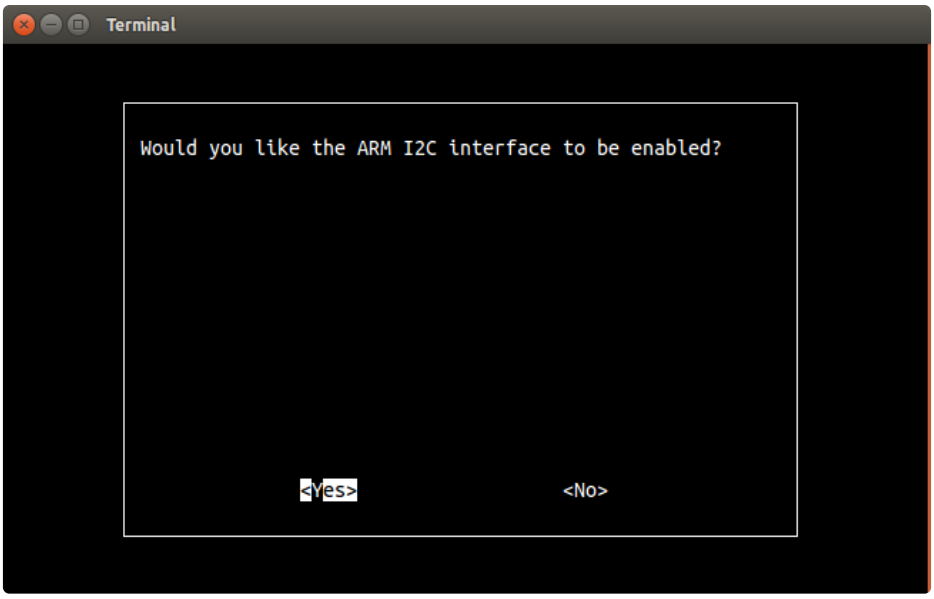

Terminal
Raspberry Pi Software Configuration Tool (raspi-config)
P1 Camera Enable/Disable connection to the
P2 SSH Enable/Disable remote command lin
P3 VNC Enable/Disable graphical remote a
P4 SPI Enable/Disable automatic loading
P5 I2C Enable/Disable automatic loading
P6 Serial Enable/Disable shell and kernel m
P7 1-Wire Enable/Disable one-wire interface
P8 Remote GPIO Enable/Disable remote access to G
<Select>
<Back>

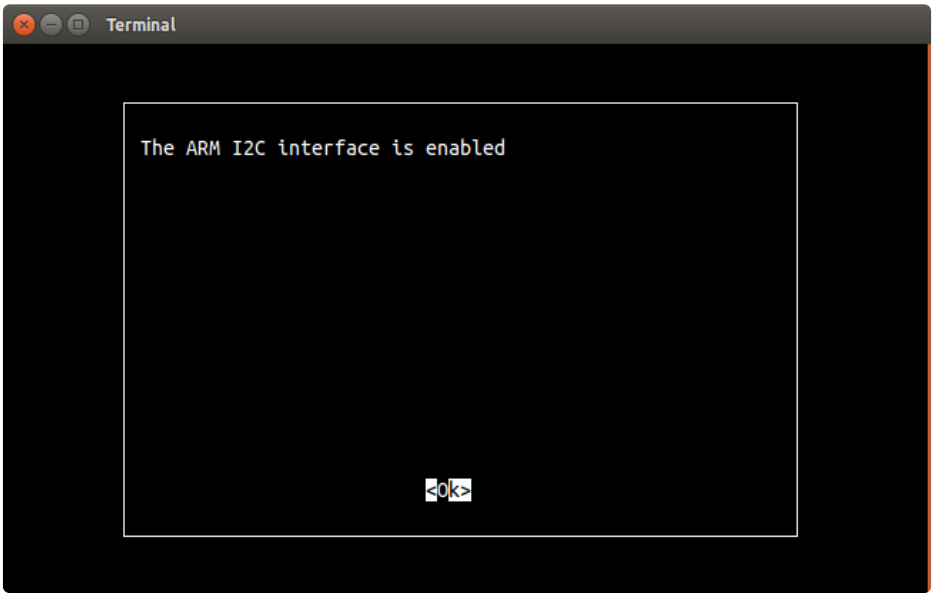

Terminal
Would you like the SPI interface to be enabled?
<Yes>
<No>


Terminal
The SPI interface is enabled
<Ok>

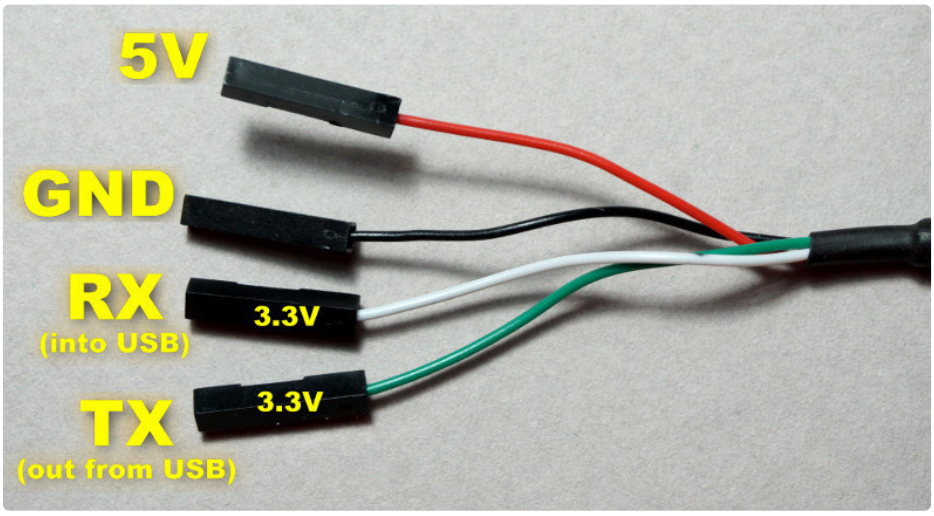
I2C


Terminal
Raspberry Pi Software Configuration Tool (raspi-config)
P1 Camera Enable/Disable connection to the
P2 SSH Enable/Disable remote command lin
P3 VNC Enable/Disable graphical remote a
P4 SPI Enable/Disable automatic loading
P5 I2C Enable/Disable automatic loading
P6 Serial Enable/Disable shell and kernel m
P7 1-Wire Enable/Disable one-wire interface
P8 Remote GPIO Enable/Disable remote access to G
<Select>
<Back>


Terminal
Would you like the ARM I2C interface to be enabled?
<Yes>
<No>


Terminal
The ARM I2C interface is enabled
<Ok>

Using A Console Cable to Troubleshoot



If you were unable to ping or ssh into the Pi Zero then you need to trouble shoot what's going on. This is where the USB Console Cable (https://adafru.it/dDd) comes in handy.

There is already an excellent guide that goes over the basics of using a USB console cable with instructions on requirements for the various major operating systems. **Read this first and then come back.**

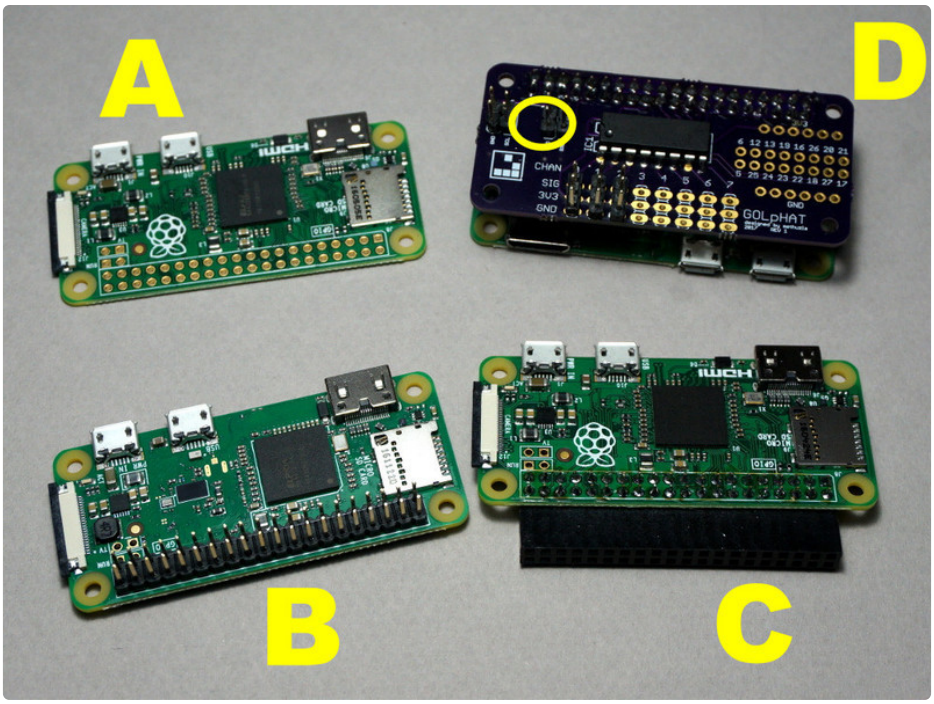
https://adafru.it/kgF

https://adafru.it/kgF

This guide will use screen (https://adafru.it/wD0) on Linux for the examples, but since pretty much everything is command line based and just a bunch of text, it should look the same on all OS's.

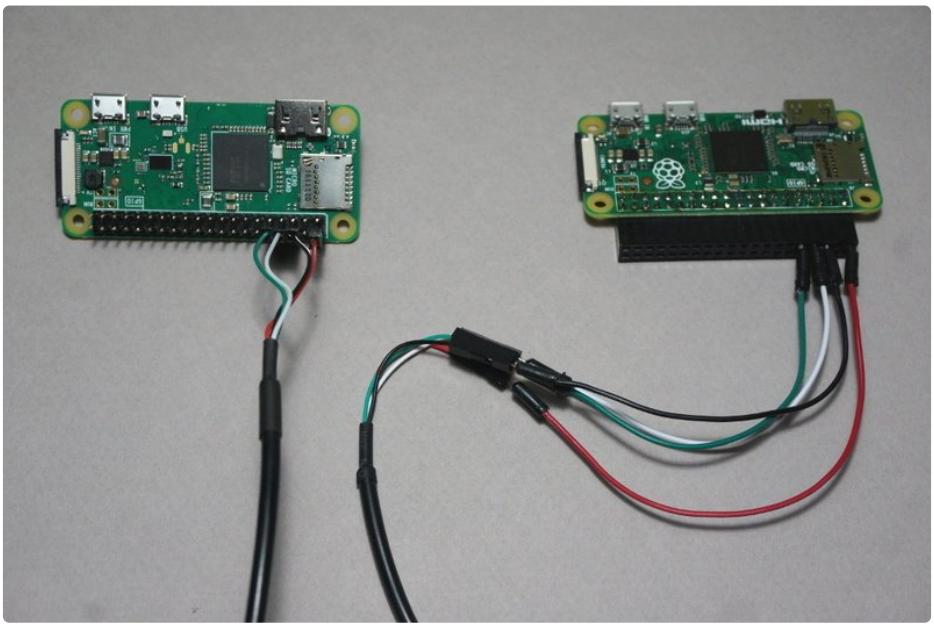
Interfacing To Pi Zero

The Pi Zero leaves the GPIO header pins unpopulated. This is actually a great feature as it allows you to solder on whatever style makes most sense for your project.



In the image above there's (A) a Pi Zero with no header attached, (B) a Pi Zero W with male headers, (C) a Pi Zero with right angle female headers, and (D) a custom topper with serial pins indicated.

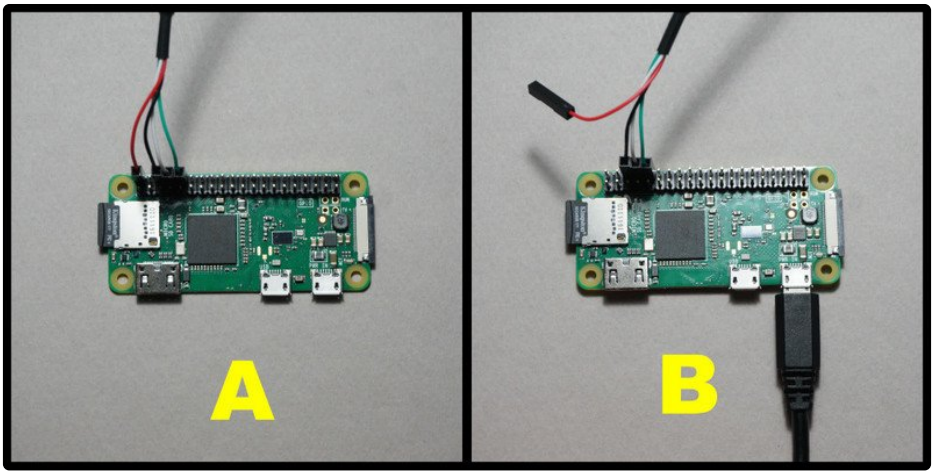
If you have male headers, you can connect the console cable directly. If you have female headers, just use some short lengths of prototyping wire.



Power Options

There are a couple of different ways to power the Pi Zero when using a console cable. Both options are

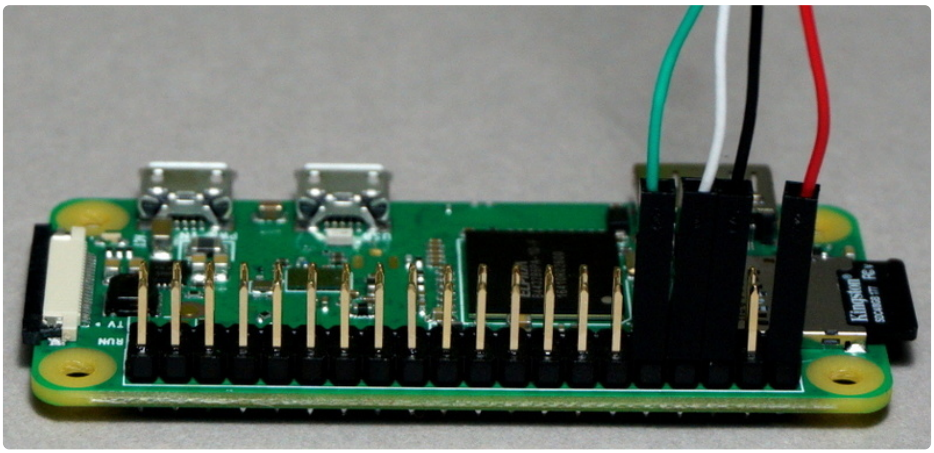
shown below.

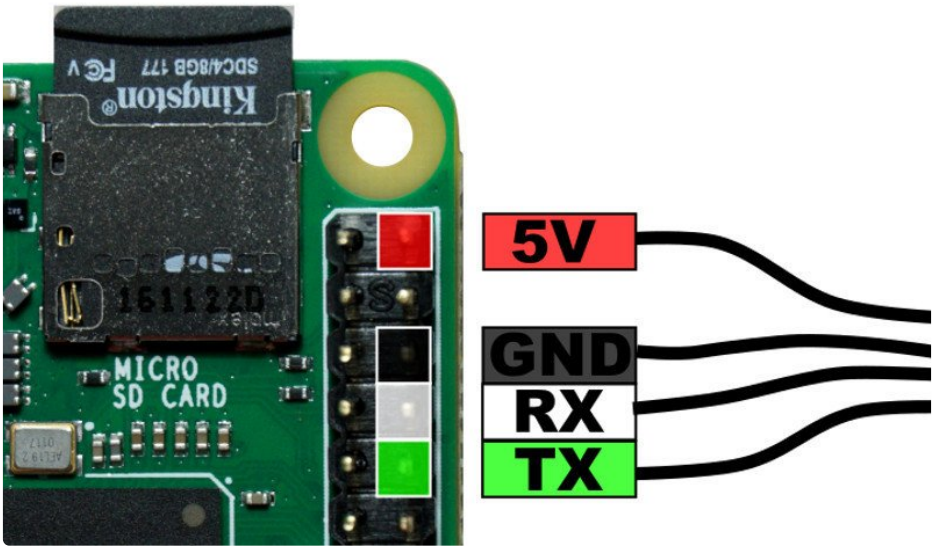


You should be able to power the Pi Zero directly from the USB cable (A). However, if you have other items attached to the Pi Zero that require more power, you may need to provide power through the micro-USB connector (B). In that case, do not connect the red power lead.

Booting With A Console Cable

Proper wiring is critical, esp. for the red power cable. Carefully examine the images below to insure you have the correct cable connections.

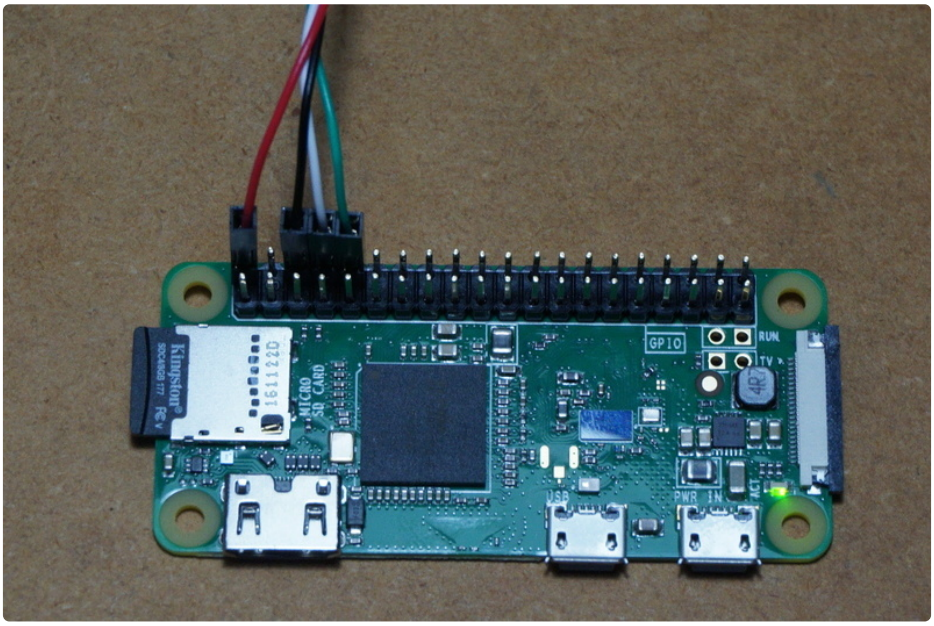




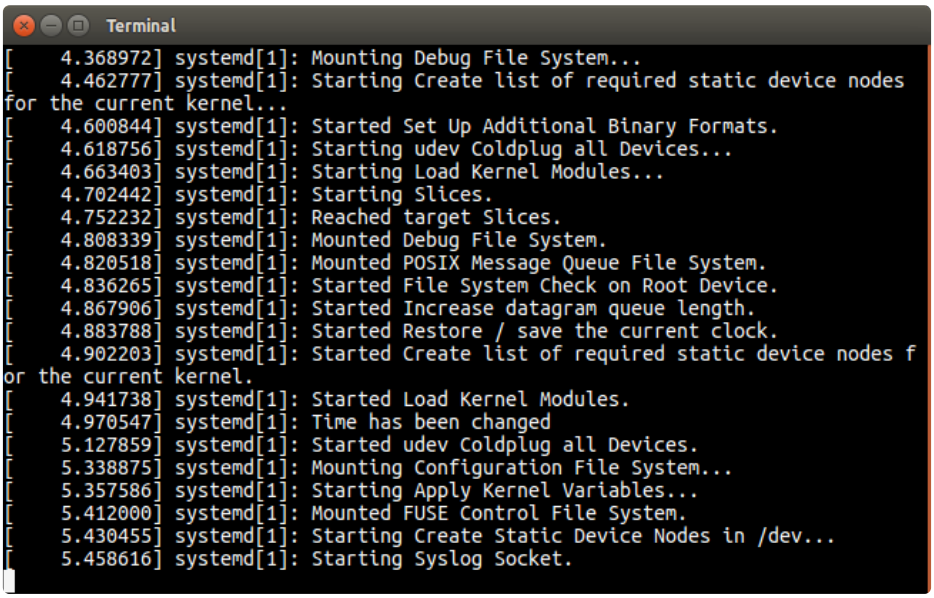
OK. Let's bring it up. Do things in this order:

1. Insert SD card into Pi Zero.
2. Connect the green, white, and black cable leads to the GPIO header. **Leave red disconnected for now.**
3. Insert USB console cable into host computer.
4. Launch whatever terminal program you use.
5. Make 110% sure you know where the 5V pins are on the GPIO header.
6. Swiftly and carefully attach the red cable lead to one of the 5V pins on the header. **THIS IS THE SPARK OF LIFE!** (or death if you chose the wrong pin)

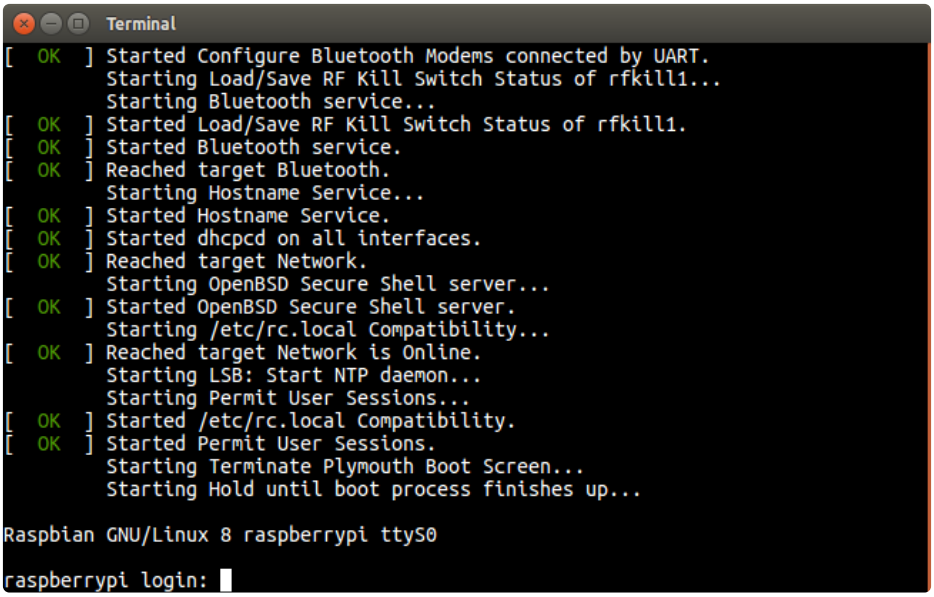
You should now see some signs of life on the green ACT LED.



And in your terminal window, you should see boot messages scrolling by.

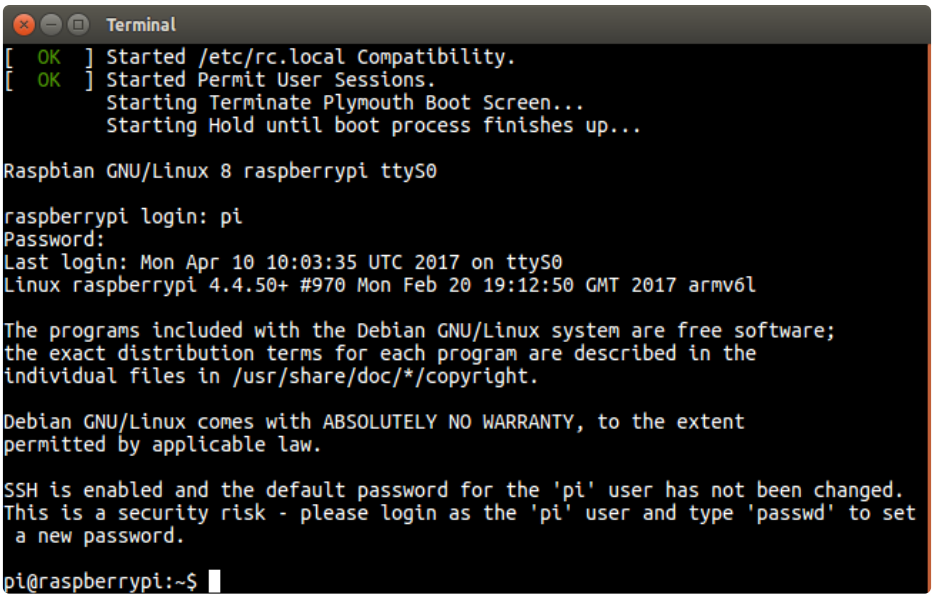


Eventually you should get to a login prompt.



Use the following user name and password to login:

- pi
- raspberry



And you're in.

What's next depends on what the problem is. We can't cover them all here. But now you have access to the Pi Zero and can navigate the system and edit things as needed based on whatever guidance you receive.

Some common basic commands are:

- **dmesg** to check the kernel log for anything odd
- **ifconfig** to see network status and settings
- **lsusb** to see what's connected to USB port
- **lsmod** to check for kernel modules