

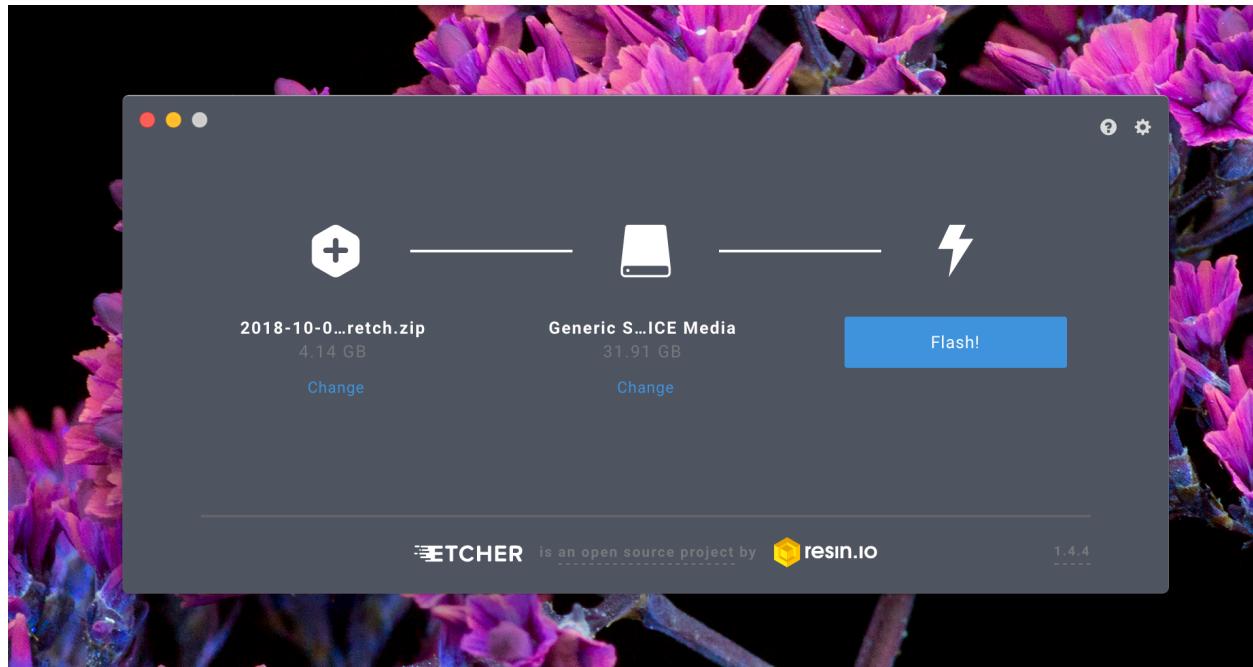
Raspberry Pi Setup For Scientific Python and Visual Stimuli

10/10/2018 - cdeister@brown.edu

Download the latest version of “Raspian” with desktop.
https://downloads.raspberrypi.org/raspbian_latest

Raspian is a Debian-based Linux distribution compiled for the ARM chip on the pi. You will need to burn this image to an SD card. I recommend San Disk 32GB. There is a method that allows booting from a USB thumb drive, but there is no practical benefit as the USB on the pi is pretty slow, even for USB2.

Use etcher (etcher.io) to burn the image. It’s a simple drag-and-drop operation:
This will take a while to burn (10-20 min). Once you have burned it, eject the SD card from your computer, but if you plan to use wifi, place it back in. You can auto-configure the wifi and then set everything up over ssh.



Make a text file called “wpa_supplicant.conf” and save it to the root of the SD card. Be sure to modify the ssid and psk for your network.

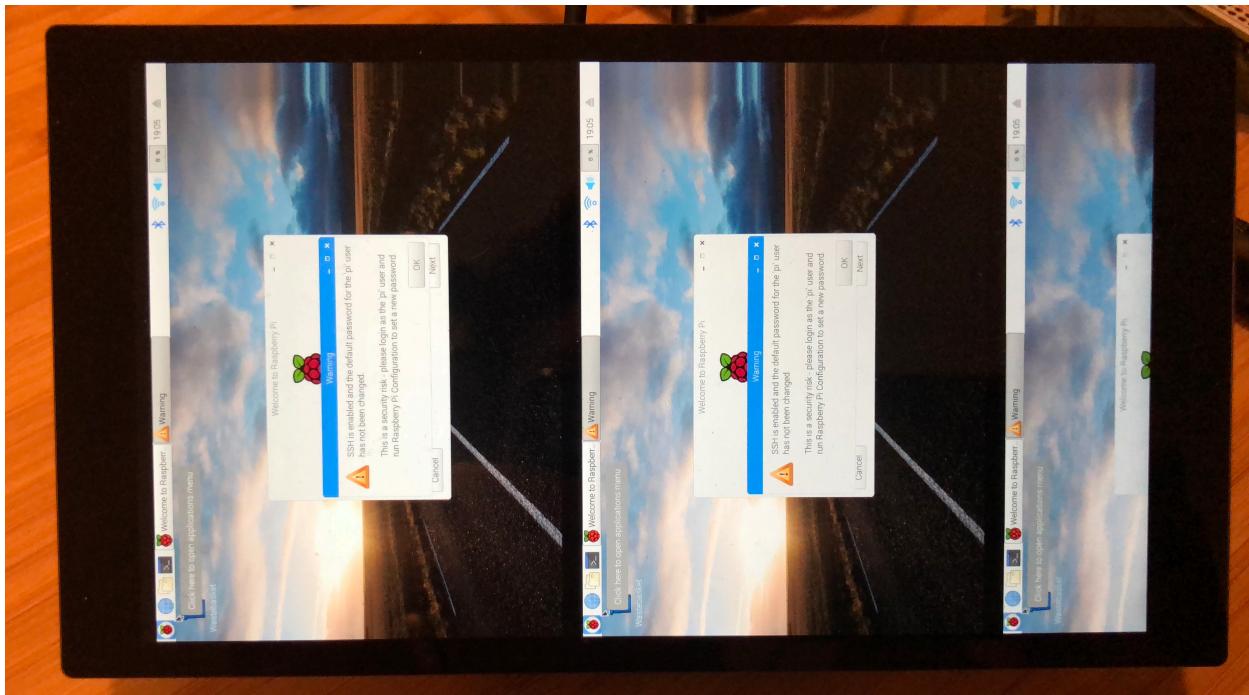
```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
network={
    ssid="YOUR_NETWORK_NAME"
    psk="YOUR_PASSWORD"
    key_mgmt=WPA-PSK
}
```

Also, ssh will not be enabled by default. You can add a blank text file to root called “.ssh” to enable it. You can do this in a terminal by:

```
cd /Volumes/boot  
touch ssh
```

Now put the SD card into your pi, and boot. The first boot will resize the SD card from 4.14 GB (the size of the image you burned) to the size of the SD card. If you have a screen attached this will be clear, if not, it should reboot on its own, but check the blinking lights on the pi, if it is just red for a while unplug and plug the power back in.

If you are using a Manga Screen 2, for visual stimuli. You will need to configure it to look right.



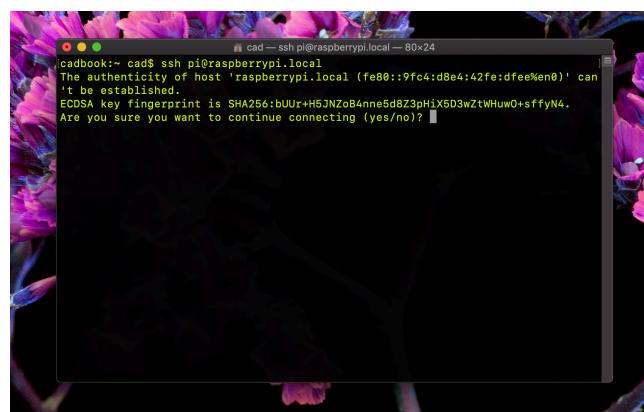
To do this, and to configure the pi in general, ssh into it via a terminal. You will need to know its IP address, which you can get from the pi itself via a terminal. Just type “sudo ifconfig” and look for a sensible IP on the adapter you are using (e.g. ethernet0 vs wlan0; wifi is WLAN). If you can not get to the pi it’s default hostname is “raspberrypi” so you should have decent luck trying that.

The default password is raspberry and the default user is pi.

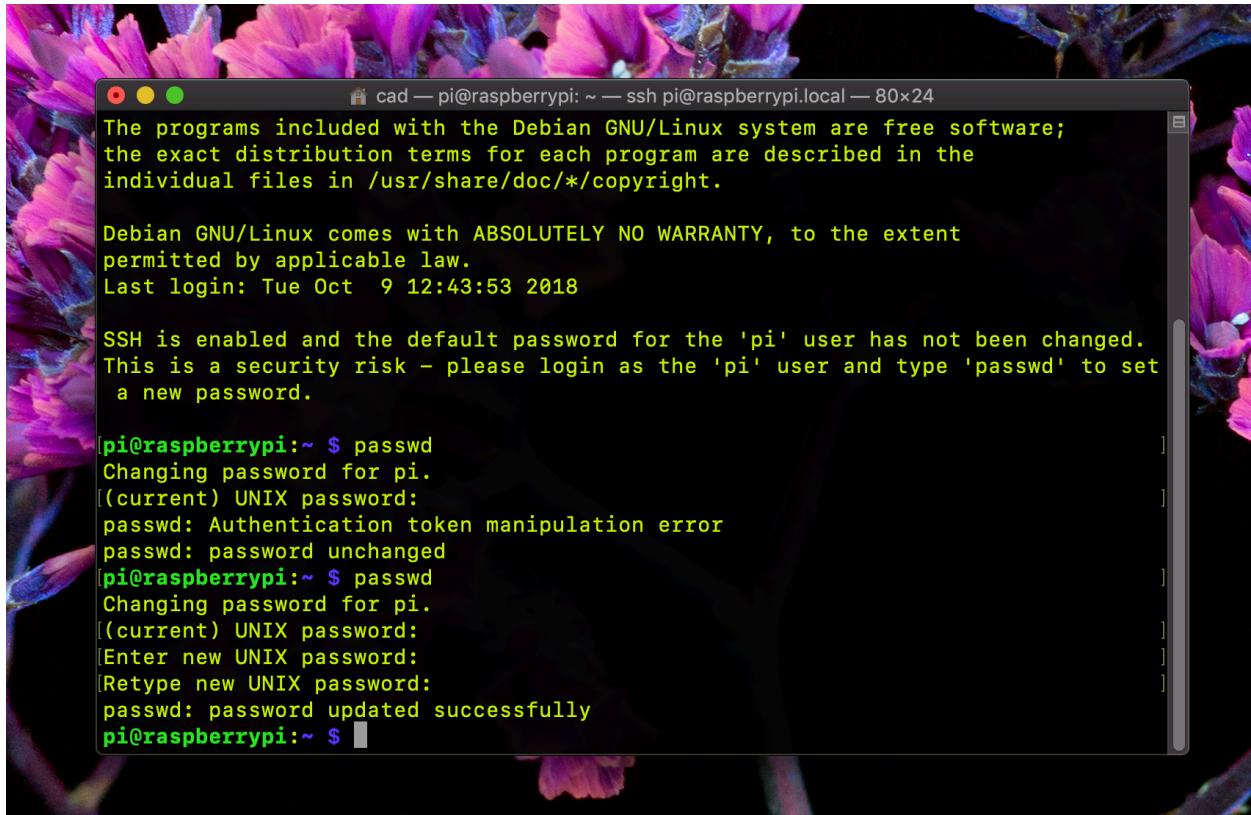
```
ssh pi@raspberrypi.local
```

It will ask you if you trust it, and the password. Currently the Pi is in an unsecure state, but the image is fine and not going to mess you up.

You should first change the password! Because ssh is on, you could be vulnerable to an attack in the long run. Just type in “passwd” and type in the “current” default password, then it will ask you for a new one.



If you did everything right you should see a prompt like this:



The screenshot shows a terminal window titled "cad — pi@raspberrypi: ~ — ssh pi@raspberrypi.local — 80x24". The window displays the following text:

```
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Oct  9 12:43:53 2018

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $ passwd
Changing password for pi.
(current) UNIX password:
passwd: Authentication token manipulation error
passwd: password unchanged
pi@raspberrypi:~ $ passwd
Changing password for pi.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
pi@raspberrypi:~ $
```

Get csBehavior

Go ahead and make sure you are at your home folder and clone the repository:

```
cd ~
git clone https://github.com/cdeister/csBehavior.git
```

This should clone all of csBehavior, setting your branch to master. If you want to change the branch (for example cadDevelop) type the following

```
cd csBehavior/
git checkout cadDevelop
```

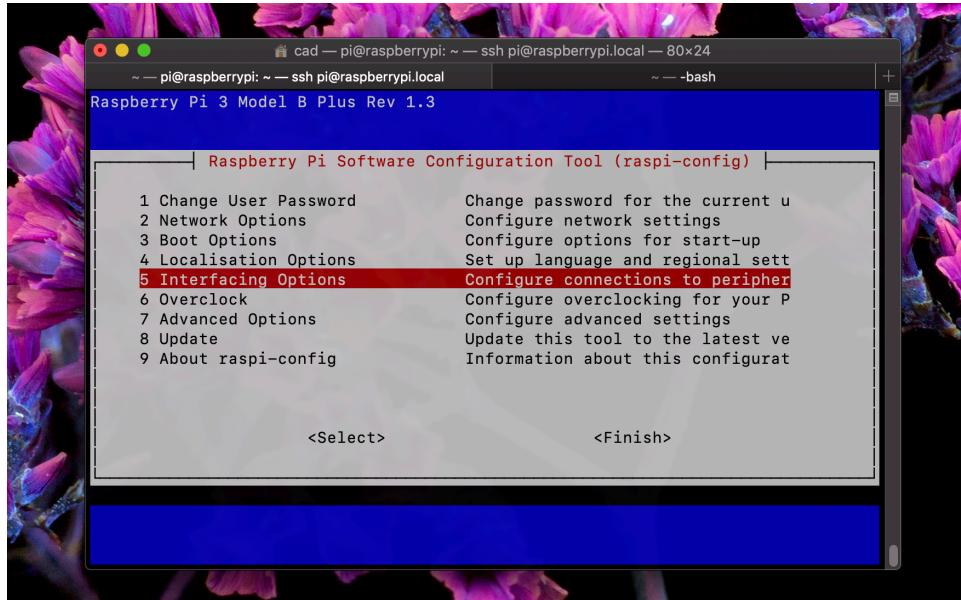
Now you should be set. In documentation/scripts/ I have a host of shell scripts that can speed all this up.

Configure Raspbian

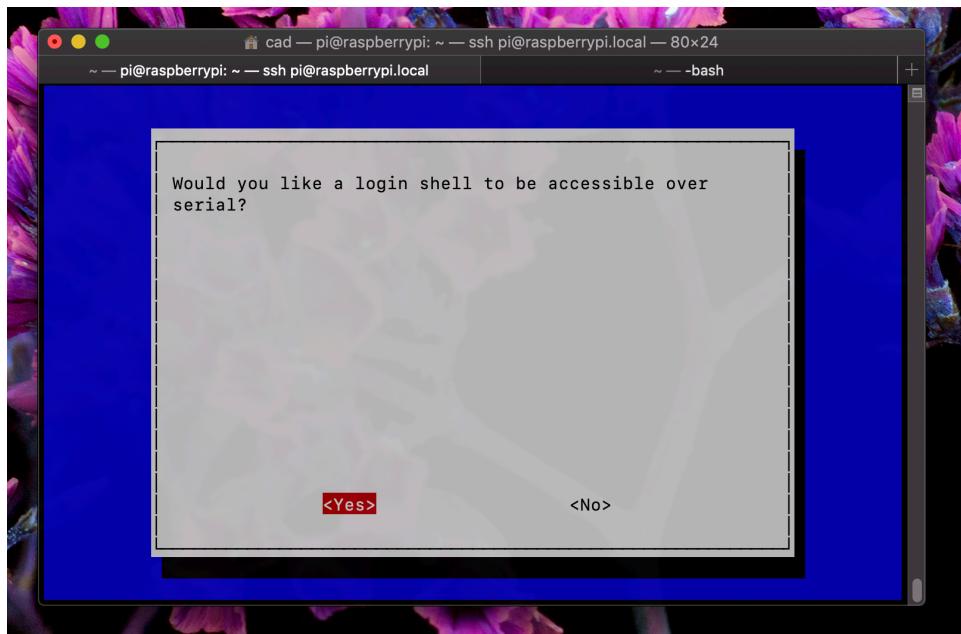
Raspbian has a terminal GUI that lets you configure system stuff. We mainly want to turn on all GPIO and make sure our display driver is “fakeKMS”

```
sudo raspi-config
```

This will bring up the following prompt:



Navigate to 5, interfacing options, and enable all but SSH, assuming you enabled it already. You will be kicked to the main screen after each enable so be patient. When you enable serial it will ask if you want the login shell to be accessible over serial (say no), then it will ask if you want to enable hardware serial (you do). Lastly, navigate to “Advanced Options” and navigate to “GL Driver.” Select “fakeKMS” Now you can exit, it will ask you to reboot and you should.



Update apt, and the OS

Raspbian is debian. So, it has apt-get. Lets do the following:

```
sudo apt-get update  
sudo apt-get upgrade -y  
sudo apt-get dist-upgrade -y
```

Install Scientific Python and PsychoPy

This is still usually more complicated than it needs to be. I have made a shell script in /csBehavior/documentation/scripts called “setupPiNoVNC.sh” and it will (as of this writing) successfully compile everything you need. You can always open it up in a text editor and cut and paste.

Manga Screen Configuration (applies to other HDMI displays, sorta):

I use a 1920x1080p Manga Screen 2 from thing-printer.com to display stimuli. It is a cheap hi-quality HDMI display, that was a tablet screen turned into an HDMI display. As of this writing many were available, and are \$99 each.

<https://www.thing-printer.com/product/manga-screen/>

The community has a wiki page with most info you would need.

http://wiki.thing-printer.com/index.php?title=Manga_Screen_2

Add the following via nano to /boot/config.txt

```
sudo nano /boot/config.txt  
hdmi_cvt=1080 1920 60 5 0 0 0  
hdmi_group=2  
hdmi_mode=87  
hdmi_timings=1080 1 100 10 60 1920 1 4 2 4 0 0 0 60 0 144000000 3  
max_framebuffer_width=1920  
max_framebuffer_height=1920  
sudo reboot
```

This will reboot your pi.

I do not use the rotate option, and the screen is natively 1080x1920, so it will display longwise. The reason is performance. The screen doesn't have a scaler and depends on the pi, the pi does this fast if you use the openGL driver, but that driver is buggy and I use the fakeKMS driver instead. It scales slow.

Set the desktop to be 50% gray and auto-hide the menubar.

We want to disable screen saver nonsense, which is not possible by default. You have to install “x screen saver”

```
sudo apt-get update  
sudo apt-get install xscreensaver -y
```

Then you can navigate to preferences and configure the screen saver (disable it).

