



Raspberry Pi Pedal for Youtube

Created by Jeff Epler



<https://learn.adafruit.com/raspberry-pi-pedal-for-youtube>

Last updated on 2024-06-03 03:16:41 PM EDT

Table of Contents

Overview	3
• Parts	
Install the cadence sensor	5
Set up Raspberry Pi OS	6
• Do Not Use sudo When Running pip3	
• Add User to bluetooth Group	
• Raspberry Pi 4B Firmware Fix	
Set up Chrome Browser Extension	8
Using and Customizing the script	10
• Customization	

Overview



Need a little more incentive to meet your exercise goals on a stationary bike? With Adafruit Blinka and a Raspberry Pi 4, you can use a Bluetooth BLE cadence sensor to play YouTube videos only when you are pedaling above a target RPM value.

This project does not require any soldering or wiring or even modification to your stationary bike - the sensor can be taped onto the bottom of the pedal, and the Raspberry Pi reads the cadence data over wireless Bluetooth!

The code is easily adjustable, you can add goals, HIIT pacing, even turn it around so you have to pedal harder to stop yourself from watching disgusting videos!



Just hook your Pi to a TV with a Micro HDMI to HDMI cable and get pedaling!

Parts



[Raspberry Pi 4 Model B - 4 GB RAM](https://www.adafruit.com/product/4296)

The Raspberry Pi 4 Model B is the newest Raspberry Pi computer made, and the Pi Foundation knows you can always make a good thing better! And what could make the Pi 4 better...

<https://www.adafruit.com/product/4296>

You will want a Pi 4 because playing youtube videos can take a lot of processor time, however you could adapt this project to play videos off the SD card in which case a less powerful Pi would work



[Micro HDMI to HDMI Cable - 2 meter](https://www.adafruit.com/product/1322)

Connect an HDMI device with a micro HDMI port to one with a regular size HDMI port together with this basic HDMI cable. It has nice molded grips for easy installation, and is 2 meter...

<https://www.adafruit.com/product/1322>



[Official Raspberry Pi Foundation Raspberry Pi 4 Case - Red White](https://www.adafruit.com/product/4301)

Keep your Raspberry Pi® 4 Model B computer safe and snug in this solid ABS acrylic enclosure. You can even pick up an official...

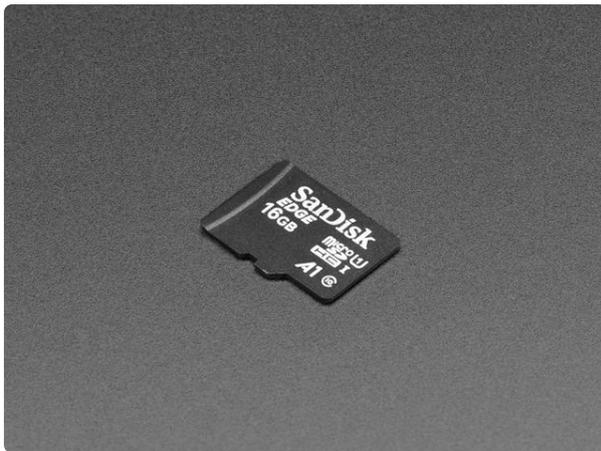
<https://www.adafruit.com/product/4301>



Official Raspberry Pi Power Supply 5.1V 3A with USB C

The official Raspberry Pi USB-C power supply is here! And of course, we have 'em in classic Adafruit black! Superfast with just the right amount of cable length to get your Pi 4...

<https://www.adafruit.com/product/4298>



16GB Card with NOOBS 3.1 for Raspberry Pi Computers including 4

NOOBS 3.1 is the fastest way to have a variety of operating systems on your Pi. Available on a 16G card, you can now boot multiple OS's such as Raspbian, Pidora, RaspBMC,...

<https://www.adafruit.com/product/4266>

1 x Wahoo RPM Cycling Cadence Sensor, Bluetooth/ANT+

Other Bluetooth Low Energy Cadence Sensors may also work

<https://www.amazon.com/Wahoo-Cycling-Cadence-Sensor-Bluetooth/dp/B00L9XNFPY/>

Install the cadence sensor

There are multiple ways to install a cadence sensor, but the key is to make sure it is going around in a big circle while you're pedaling. The manufacturer may include a variety of options.

This sensor uses an accelerometer so you don't need a magnet or other external sensor, its a self-contained blob!

I used the included double side adhesive to install it under one of my bike's foot pedals. It took just a minute and works great!



Set up Raspberry Pi OS

Raspberry Pi OS (the new name for Raspian), which is based on Debian Linux, also comes with Python. The latest version as of this writing, buster, comes with Python 3.7, invoked with the `python3` command. It also comes with Python 2.7, invoked as `python`. Do not use Python 2 for this guide.

This guide assumes that you are using the "full" version of Raspberry Pi OS that boots to a desktop operating system, that way you can play YouTube videos in the browser.

We assume that you've already got your Pi set up, Pi OS (Raspbian) Full installed, keyboard and monitor attached, and you're logged in with WiFi/Ethernet networking set up you know your way around a Terminal command line

Do not use `sudo` when running `pip3`.

Do Not Use `sudo` When Running `pip3`

You may see Internet advice to install libraries using `sudo pip3` on Linux or MacOS. This is in general a bad idea, because you can damage the libraries that the underlying system depends on, and in general end up trashing your system in mysterious ways. Always install using `pip3` without `sudo`. If your `pip3` is old, you may need to specify `pip3 --user`, but these days `--user` is often the default.

Here's a detailed discussion (<https://adafru.it/Od6>) of why `sudo pip3` is a bad idea.

Run the following command to install the required Python packages for the project:

```
pip3 install --upgrade adafruit-blinka-bleio adafruit-circuitpython-ble adafruit-circuitpython-ble-cycling-speed-and-cadence python-xlib
```

Add User to `bluetooth` Group

On Linux, including on Raspberry Pi, you must also add your user to the `bluetooth` group. Reboot after doing this to ensure your user is added to the group and also to ensure that `~/local/bin` is added to your path (after doing `pip3` above). To add yourself to the group, do this (the second command will restart your computer, so save any documents or files first!):

```
sudo usermod -a -G bluetooth $USER
sudo reboot
```

Raspberry Pi 4B Firmware Fix

Raspberry Pi **4B** boards use different hardware for BLE and WiFi than Raspberry Pi **3B** and Pi **Zero W** boards. There is currently (as of October 2020) a bug in the firmware for the 4B boards, in package `bluez-firmware_1.2-4+rpt5_all.deb`. You must roll back the firmware package to an earlier version (`rpt2`) to get one that works. Do this **only for 3B+ and 4B boards**.

This bug is being tracked here: <https://github.com/RPi-Distro/bluez-firmware/issues/6> (<https://adafru.it/Oda>).

The third command will restart your computer, so save any documents or files first!

Anytime you upgrade packages, the incorrect version of this package may be reinstalled. Just run the commands below after an upgrade.

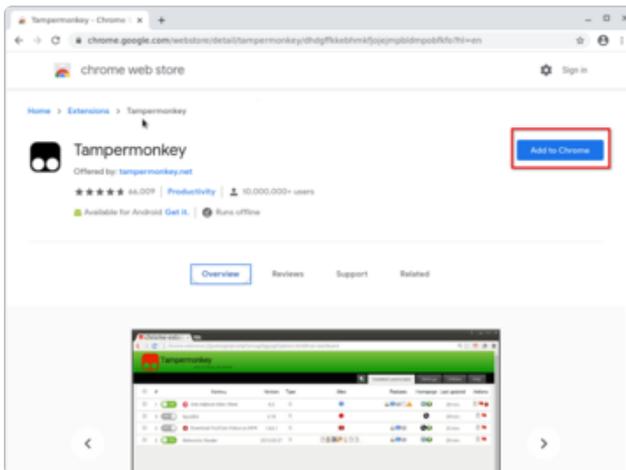
```
wget http://archive.raspberrypi.org/debian/pool/main/b/bluez-firmware/bluez-firmware_1.2-4+rpt2_all.deb
sudo dpkg --install bluez-firmware_1.2-4+rpt2_all.deb
sudo reboot
```

Set up Chrome Browser Extension

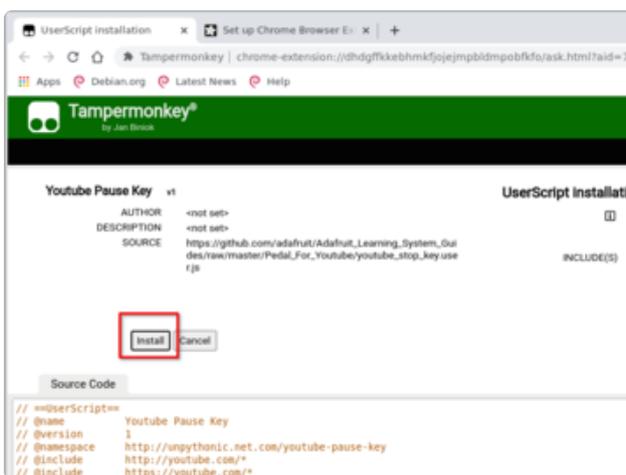
A common but frustrating user-interface thing that we see in video or audio playing devices is they use one key for both play and pause. That's easy for humans, but annoying for us because we want to know that when we sent a certain keypress the video stops, and a different key will start it, so we don't end up 'de synchronized' if the keypress is missed or double-counted somehow.

This script alters behavior on youtube.com so that pressing "p" will pause the video if it was playing. Youtube has a keypress "k" which will pause a playing video or restart a paused video, but for our program we need a way to get the video into a known state. Otherwise, the program could work backwards--pausing when you pedal, playing when you stop.

A simple extension solves this problem and gives us different keys for playing and pausing videos.



Open the Chromium browser on your Pi, go to the Chrome Web Store, and install the "tampermonkey" extension. This extension allows the installation of "user scripts", which can modify the behavior of websites.



Once tampermonkey is installed, [click here to download and install the script shown below. \(https://adafru.it/OZd\)](https://adafru.it/OZd)

Tampermonkey will pop up a screen with an "install" button. Click it.

[Head over to YouTube \(https://adafru.it/OZe\)](https://adafru.it/OZe) and check that you can pause the playing video by typing "p" and continue playback by pressing "k". If so, you're good

to go. Otherwise, you will need to double check the above steps to make sure that the Tampermonkey extension and the Youtube Pause Key userscript are correctly installed.

(There are other extensions that add capabilities like this to YouTube, but they are very complex. We wrote our own so that you can see exactly what it does, and be confident it isn't doing anything you don't want.)

[Click here to download and install the script shown below. \(https://adafru.it/OZd\)](https://adafru.it/OZd)

```
// ==UserScript==
// @name      Youtube Pause Key
// @version   1
// @namespace  http://unpythonic.net.com/youtube-pause-key
// @include   http://youtube.com/*
// @include   https://youtube.com/*
// @include   http://www.youtube.com/*
// @include   https://www.youtube.com/*
// @grant none
// ==/UserScript==

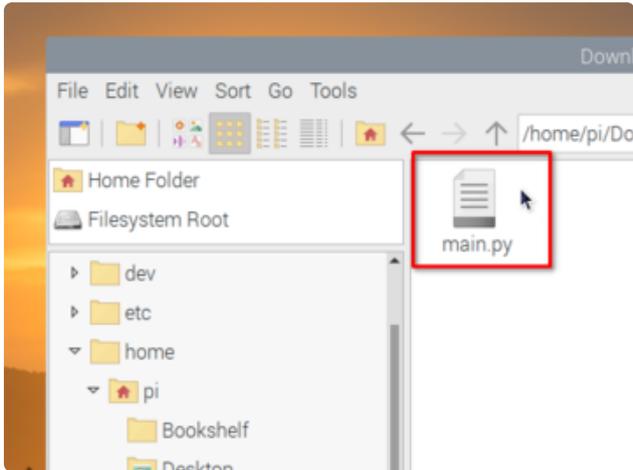
(function () {
  var inject = function() {
    var is_press_key = function (e, key) {
      document.last_target = e.target;
      document.last_event = e;
      var keyCode = e.keyCode ? e.keyCode : e.which;
      return (! (e.altKey || e.ctrlKey || e.metaKey)
        &&& keyCode == key);
    }

    var gr_in_bg_event = function(e) {
      var pause_key = 112; // 'p' key
      var videos = e.target.getElementsByTagName('video');
      if(videos.length == 0) { return true; }
      if (is_press_key(e, pause_key)) {
        videos[0].pause();
        e.stopPropagation();
        e.preventDefault();
        return;
      }
      return true;
    }

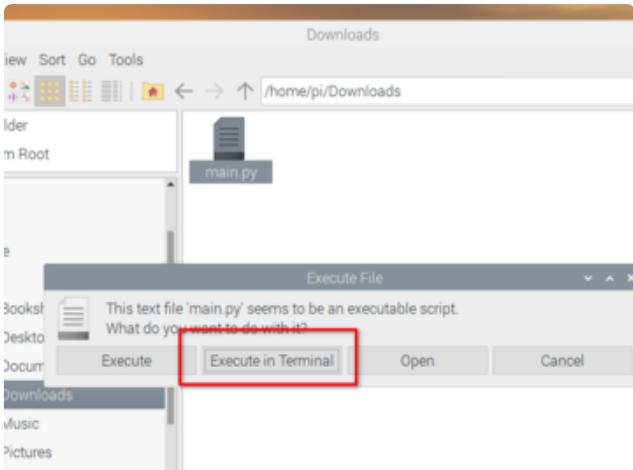
    document.addEventListener("keypress", gr_in_bg_event, true);
  }

  var newel = document.createElement("script");
  var newcontent = document.createTextNode("(" + inject + ")()");
  newel.appendChild(newcontent);
  document.body.appendChild(newel);
})();
```

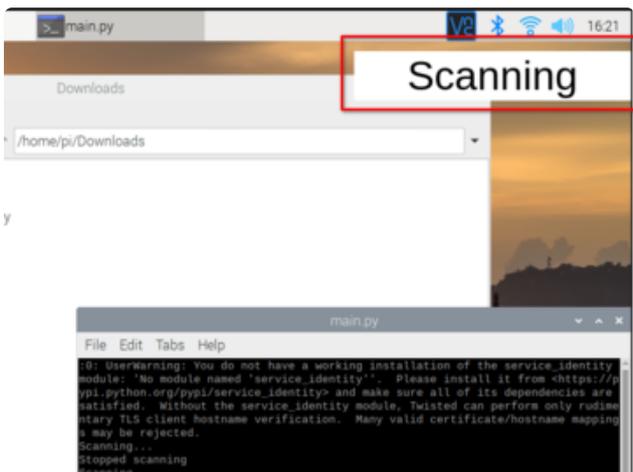
Using and Customizing the script



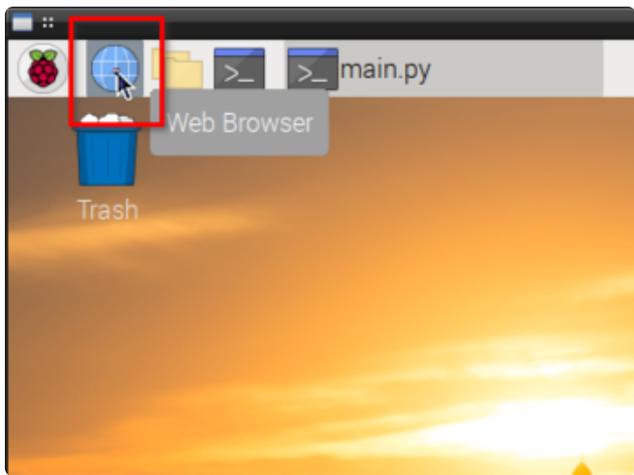
Download the main script from [github \(https://adafru.it/VFE\)](https://adafru.it/VFE) and save it as code.py.



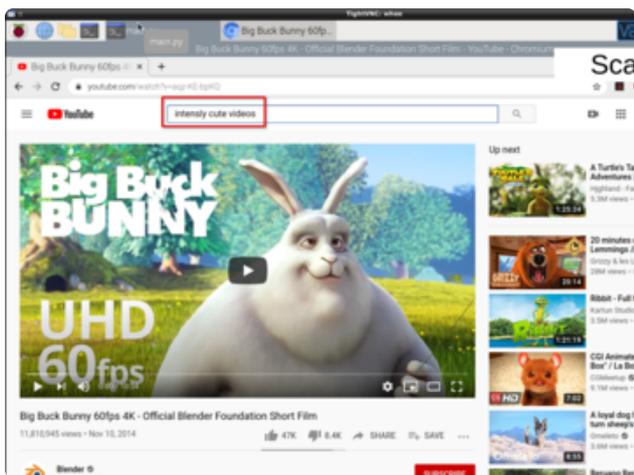
Double-click it and choose "Execute in terminal".



The terminal will show a scary (but unimportant) message. More importantly, the OSD will appear and show "Scanning".



Start the Chrome browser, navigate to youtube.com, and pick whatever kind of video is your catnip.



Start pedaling after a few seconds (give it a good 15 seconds), the sensor will be detected and your cadence in RPMs will be displayed.



In the default script, if you are in the red zone (below 60RPM), the video will pause after 2 seconds. If you're in the yellow zone (60 to 72RPM) or green zone (above 72RPM), the video will continue playing. You can customize these values by editing the script and restarting it.

The script is able to detect when the active window is a Chromium browser window but not whether you're on a YouTube video. If your computer is randomly typing "p"

and "k" into websites, make sure you've stopped the script! Just return to the terminal window where you started the script and use the close button in the corner (x) or press **Ctrl-C** to interrupt it.

Customization

The script is set up so that you can easily customize the following values:

```
MINIMUM_RPM = 60
```

```
TARGET_RPM = 72
```

```
GRACE_TIME = 2
```

Ready to do something more in depth? Here are just a few of the other ideas the author and his friends had while writing this guide:

- The On Screen Display of cadence is fairly crude. How would you improve it?
- Can the program manage a whole workout, with changing target cadences and timed breaks (where the video doesn't stop?)
- Make it to work with additional video services or local video players such as **vlc**
- Interface with other BLE sensors or devices—Adafruit Blinka BLE can work with a whole range of Bluetooth Low Energy modules and sensors such as heart rate monitors, temperature sensors, BLE light bulbs, or developer boards like the CircuitPlayground Bluefruit or CLUE