

REMOTE RASPBERRY PI ROBOT (PS3 CONTROLLER) - FABLAB NERVECENTRE

30,267

60

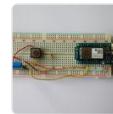
10

| Posted Sep. 5, 2014 in Technology (/technology/) > Raspberry-pi (/technology/raspberry-pi/)

| CC BY-NC-SA

<https://cdn.instructables.com/FK4/IN43/U7BDY6S/FK4/IN43U7BDY6S-LARGE.jpg>

(/member/TonyCanning/)

By **TonyCanning** (/member/TonyCanning/)[Follow](#)More by
the author: (/member/TonyCanning/)

These instructions try to assume very little. Hopefully, if you can boot a Raspberry Pi and have an internet connection you should be able to complete this great little project.

Basically, this is a project which will result in some specific areas which can be used in other projects.

The main outcomes will be:

Using a PS3 controller

Controlling DC motors

But along the way, you'll get a chance to [Download](#) or [Favorite](#) this Instructable and perform some often used linux commands in the terminal.

 Add Tip

 Ask Question

 Comment

[Download](#)

Step 1: What You Need



It's a bit of a list - but nothing is terribly expensive. I've also tried to keep the list to commonly used components that will be useful on any related project.

For example, this project could be adapted to control a garage door or to trigger some other project by incorporating a motion detector and/or camera.

So here's the list:

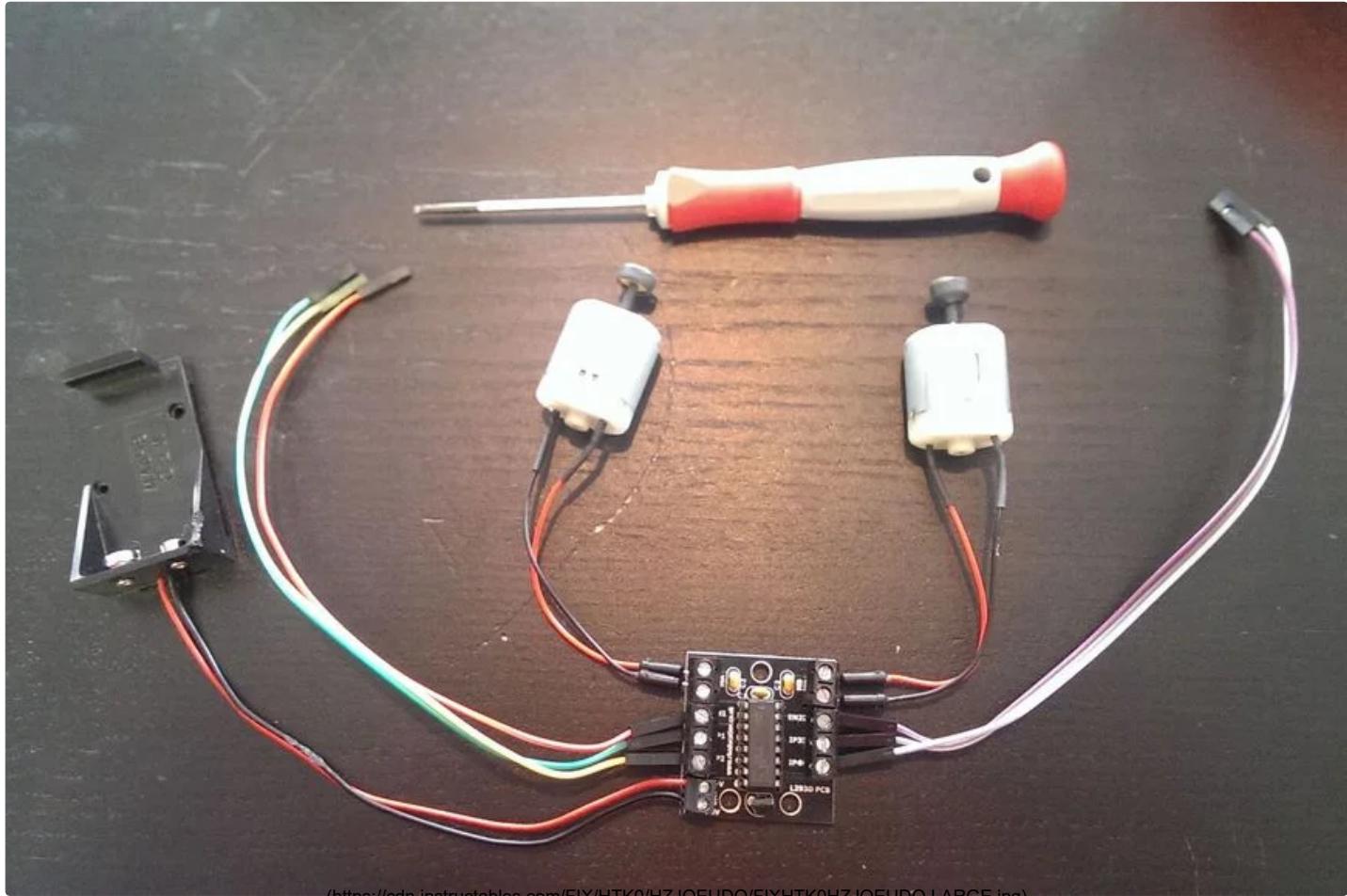
- Raspberry Pi & SD Card with Raspbian installed
- DC Motors (pictured are 2 different types - one set is geared)
- L293D motor driver pcb ([ChrisMason \(<https://www.instructables.com/id/Controlling-a-Raspberry-Pi-RC-Car-With-a-Keyboard/step4/Hardware/>\)](https://www.instructables.com/id/Controlling-a-Raspberry-Pi-RC-Car-With-a-Keyboard/step4/Hardware/), has described how to breadboard this chip if you don't want to buy the pcb)
- Powered USB hub
- 9V battery & holder
- 5V power source

- PS3 Controller Download Favorite
- All the necessary cables/power supplies for the above
- You'll also need to download the *.sh files (attached) to your /home/pi directory.

 1-dependencies.sh	Download (https://cdn.instructables.com/ORIG/F27/VBX7/HZMJ6TKR/F27VBX7HZMJ6TKR.sh) (https://cdn.instructables.com/ORIG/F27/VBX7/HZMJ6TKR/F27VBX7HZMJ6TKR.sh)
 2-downloads.sh	Download (https://cdn.instructables.com/ORIG/FE9/87QY/HZMJ6TL2/FE987QYHZMJ6TL2.sh) (https://cdn.instructables.com/ORIG/FE9/87QY/HZMJ6TL2/FE987QYHZMJ6TL2.sh)
 3-buildfromsource.sh	Download (https://cdn.instructables.com/ORIG/F6M/9TEL/HZMJ6TLH/F6M9TELHZMJ6TLH.sh) (https://cdn.instructables.com/ORIG/F6M/9TEL/HZMJ6TLH/F6M9TELHZMJ6TLH.sh)
 4-tidyup.sh	Download (https://cdn.instructables.com/ORIG/FO5/F7DF/HZNYGOMF/FO5F7DFHZNYGOMF.sh) (https://cdn.instructables.com/ORIG/FO5/F7DF/HZNYGOMF/FO5F7DFHZNYGOMF.sh)
 PS3_connect_controller.sh	Download (https://cdn.instructables.com/ORIG/FPR/4OHD/HZMJ6TMH/FPR4OHDHZMJ6TMH.sh) (https://cdn.instructables.com/ORIG/FPR/4OHD/HZMJ6TMH/FPR4OHDHZMJ6TMH.sh)

 Add Tip  Ask Question  Comment [Download](#)

Step 2: Wiring the Motors



When I began doing bits and pieces with Arduino and Raspberry Pi, I didn't get why motors couldn't be connected directly to them. Here's the quick explanation.....

If you look at the specifications of a motor, it will tell you about its RPM, ampage, Download Favorite ratio etc. Also included will be a voltage span within which it can operate. For the majority of motors used by hobbyists and makers, this voltage span is likely to be 6V or greater.

For the motors I'm using in this project, I'll be using a 9V battery. Now, a 9V battery is too much to run the Raspberry Pi from. Also, hooking up the motors directly to the RPi means that the greedy little things will try to draw more current than the RPi (or indeed an Arduino) can provide.

This can result in damaging your hardware.

You can wire the motors, battery holder and wires to the L293D pcb as shown. The 9V battery will independently power the motors. The motors themselves are attached to terminals marked "MA" and "MB" on the board I have.

The other 3 pins (per side) are inputs 1 & 2 and an enabling terminal also. The enabling terminal could be regarded as a kind of handbrake, you have to enable the motor on the channel in order to drive it. The other 2 pins are switch controls which determine the direction the motor will spin.

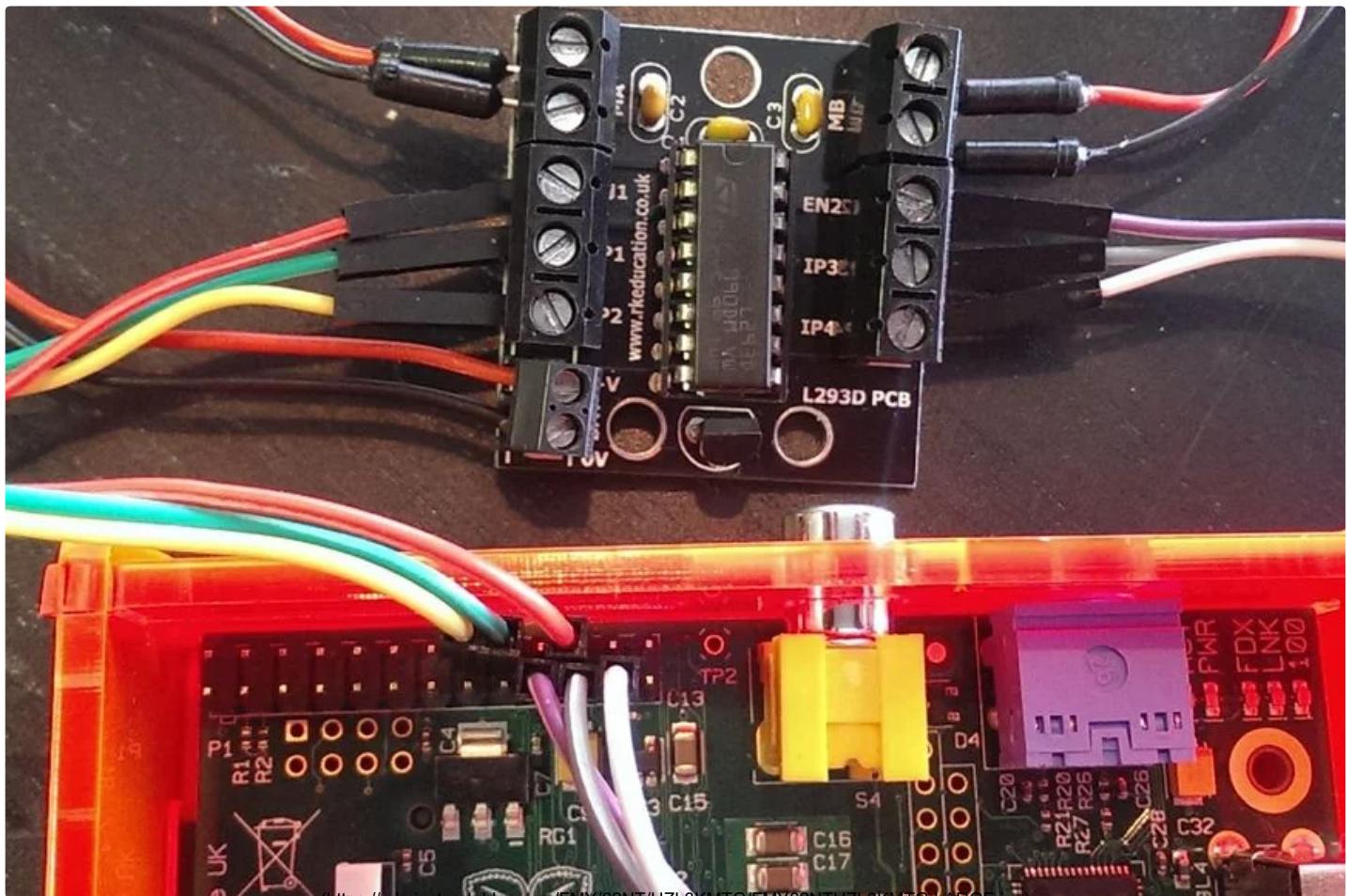
 Add Tip

 Ask Question

 Comment

Download

Step 3: Connecting the Motor Driver Pcb to the RPi



Apparently you shouldn't connect anything to the GPIO pins while the RPi is running so it's worth making sure to heed this advice or it could be expensive.

You can connect the wiring to the RPi as shown in the image.

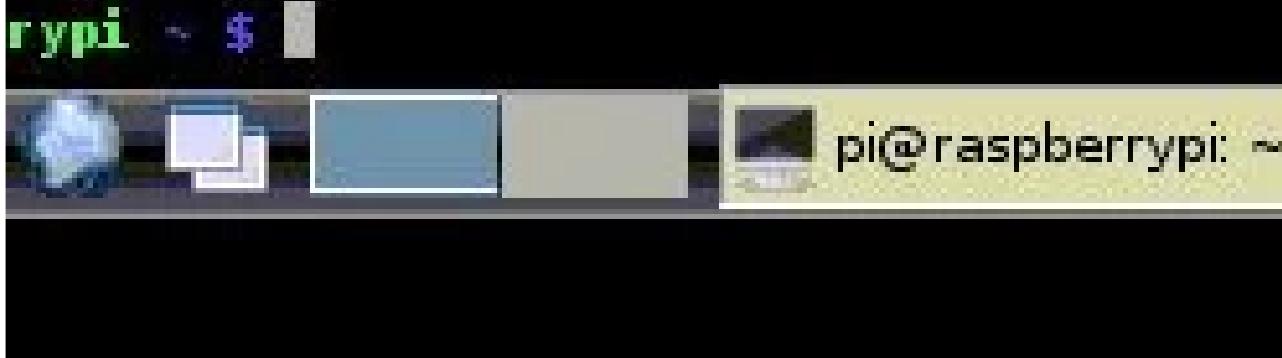
Don't put the 9V battery in just yet. I've found that if the battery is left in, it will slowly drain. Given that setting up the RPi for this could take some time, we'll just leave it out until we're ready for testing.

Another solution to this is that you can add a simple switch to the battery holder.

 Add Tip  Ask Question  Comment Download

Step 4: Getting the PS3 Controller Running on the Raspberry Pi

```
libbluetooth-dev (4.99-2)
mscompress (0.3-4)
triggers for menu ...
pi@raspberrypi ~ $ hciconfig
hci0: BR/EDR  Bus: USB
      Address: 00:15:83:0C:BF:EB  ACL MTU: 339:8  S
RUNNING PSCAN
      bytes:931 acl:0 sco:0 events:32 errors:0
      bytes:366 acl:0 sco:0 commands:25 errors:0
```



We now have the connections all made and so can move on to setting up the PS3 controller.

There are a good few tutorials on how to do this. The one I initially used is available here:

<https://github.com/petrockblog/RetroPie-Setup/wiki...>

(<https://github.com/petrockblog/RetroPie-Setup/wiki/Setting-up-a-PS3-controller>).

For anyone who is a beginner at using the terminal and is new to Linux going through all the commands and encountering some problems can be really frustrating. To help with that, I've started off with a completely clean install of Raspbian so that I can hopefully identify every issue you might encounter.

The first thing you need to do is install some background programs and libraries. These are known as "dependencies".

I'm going to give you something of an easy option here by providing script files for you to run. This should make the process easier. To run a scriptfile you just type "sudo sh ./scriptfilenamehere.sh" at the command prompt.

The first to be run is **1-dependencies.sh** (*Get yourself a cup of tea*)

After running this script we need to check if the RPi is seeing the bluetooth dongle. Run "hciconfig" at the command prompt. This will give an output the same as the image above if you're bluetooth dongle has been attached before booting up. (If not seen, you may need to reboot)

The next script to run is **2-downloads.sh**. This script will create a directory and download some necessary files to it. This shouldn't take long.

The third script to run is **3-buildfromsource.sh**. This will compile and install the sixaxis manager software. You will be prompted to answer a few questions. To do so, type nothing and just hit [ENTER] each time you are asked anything.

The next script is **4-tidyup.sh**. I'm sure you can guess what this is for - just a bit of housekeeping to keep things clean.

 Add Tip  Ask Question  Comment Download

Step 5: Connecting to Your PS3 Controller



```
control/
pi $ ls
dfromsource.sh  PS3_connect_controller.sh  si
up.sh          QtSixA-1.5.1
pi $ sh ./PS3_connect_controller.sh
00:15:83:0c:bf:eb
00:15:83:0c:bf:eb

Started, press the PS button now
to connect to sdp session
Sony Computer Entertainment Wireless Control
```

At this point you've done the majority of the work to having your PS3 remote controlled robot up and running. Now, we're going to make sure you can connect up to the PS3 controller and that your Raspberry Pi is receiving inputs from it.

Make sure your PS3 controller is connected to a USB port on your RPi (or to your powered USB hub).

You can now run the **PS3_connect_controller.sh** script which will pair your controller and then ask you to press the PS button. *Unplug* the controller and press the PS button - you should then see the message as per the image.

Achievement unlocked!

Your PS3 controller is now paired and ready to use with any game or robot which can take its signals.

Possible issues:

- When running PS3_connect_controller.sh you may not manage to connect due to the bluetooth dongle possibly "sleeping". The immensely technical solution to this is to remove it and then plug it back in!
- Everything *should* run correctly - if not, please comment and I can try to troubleshoot. (& update this instructable)
- One likely issue you will meet at some point is that the PS3 controller may not be seen while plugged in to USB. Type "**ls /dev/input**". This will list attached input devices and

should include "js0". If not, plug out and [Download](#) [Favorite](#) the PS3 controller USB connection and wait before listing input devices again to look for js0.

 Add Tip

 Ask Question

 Comment

[Download](#)

Step 6: Controlling the DC Motors With the PS3 Controller

Raspberry pi Bluetooth remote



At this point you now have the hardware connections to the motors complete and can read from the PS3 controller.

Now you can download the attached code which is written in Python.

I'd recommend placing this code in a folder - call it "PS3_robot" or something. It's just easier and safer to do this as things can easily get lost or deleted when left sitting in /home/pi.

Assuming you have only just turned on your RPi (or perhaps have moved your SD card to another Pi) you will need to open 2 terminals. In the first terminal, run:

- `sh ~/PS3control/PS3_connect_controller.sh`

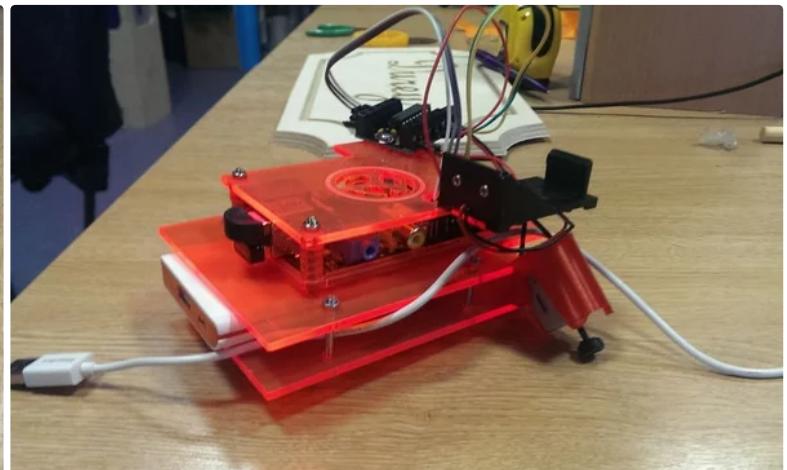
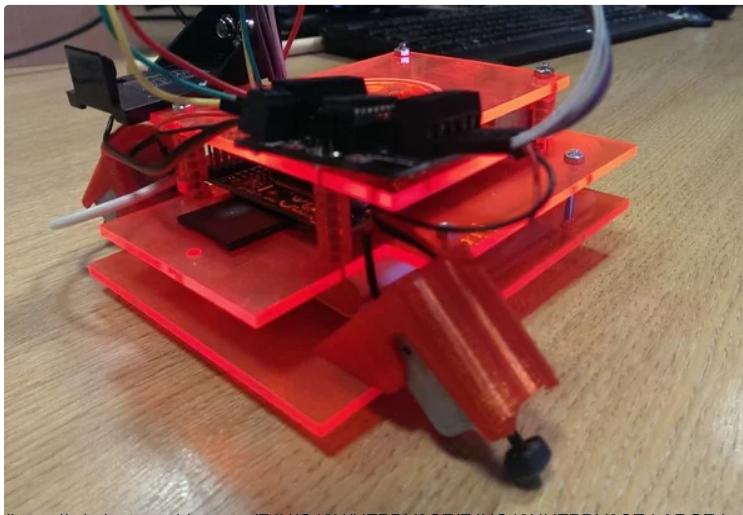
In the second terminal, run:

- `sudo python ~/PS3_robot/PS3_motors.py`

Now you can have a go at moving the motors using the PS3 controller. Each joystick will control forward and reverse movement (see video).

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

Step 7: Creating a Chassis



For motor mounts I 3d modeled a few parts - here's the one's I used for this project on [thingiverse](http://www.thingiverse.com/thing:450351) (<http://www.thingiverse.com/thing:450351>).

See the attached pdf file for the chassis and the images for some of the assembly.

You'll need a decent mess of 3mm nuts & bolts to put this together.

To assemble the pi covering spacers, a circular one goes at each corner followed by the L-Shaped spacer.

With respect to the different spacers included, some have quadrants cut from them - these 5 go above the L-shaped layer on the corner of the Ethernet connection.

The few which have flats on them are for going directly above the L-shaped layer at the power socket. The remaining circular spacers make up the rest of the height to get to the top cover.

Once assembled....

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

Step 8:Enjoy!

And if you want to know a bit more about the scripts you have run and/or the python code you can do so by typing "nano ~/<directoryname>/<filename>".

The *.sh script files can be a reference project if you wish to try to "automate" any work which would otherwise be far more tedious.



Add Tip



Ask Question



Comment

Download

Share

Did you make this project? Share it with us!

I Made It!

Recommendations



Download

Favorites



Post Comment

We have a **be nice** policy.
Please be positive and constructive.

Add Images

Post

10 Comments



hunterb40 (/member/hunterb40/) 1 year ago
(/member/hunterb40/)

[Reply](#)

Can you use a different kind of controller like an xbox 360 or xbox 1



samimac03 (/member/samimac03/) 1 year ago
(/member/samimac03/)

[Reply](#)

Do you need the Wifi dongle for anything other than installing the software? or is it necessary. Thanks



PeterB57 (/member/PeterB57/) 2 years ago
(/member/PeterB57/)

[Reply](#)

Awesome! Just what I needed. Thanx!

Got it into a directory in home/pi.

Couldn't get it to work at first, but after:

sudo update-rc.d sixad defaults (from <https://github.com/RetroPie/RetroPie-Setup/wiki/Se...> and reboot it worked.

Was able to control a servo motor with it.

Just need to work out how to re-connect after controller goes to sleep without rebooting...



Jordan76149 (/member/Jordan76149/) 2 years ago
(/member/Jordan76149/)

[Reply](#)

Hi

I have the same problem as Michael and i cant seem to get it right

Im new to this stuff so please bear with me, Im sure the second script doesnt create a new directory and the scripts, once downloaded lose their names

Any help would be much appreciated



MichaelV2 (/member/MichaelV2/) 2 years ago
(/member/MichaelV2/)

[Reply](#)

I was succesful in running scripts 1-3, but when i tell it to run the tidyup script the pi says "mv: cannot stat 'PS3_connect_controller.sh' : No such file or directory"

which is weird, and so i try to skip that one and go on to the next file, telling it to run the PS3_connect_controller file, and then it says "sudo: ./sixpair: command not found"
help

this is exactly what my screen says:

pi@raspberrypi ~ \$ sudo sh ./4-tidyup.sh

mv: cannot stat 'PS3_connect_controller.sh' : No such file or directory

pi@raspberrypi ~ \$ sudo sh ./PS3_connect_controller.sh



MichaelV2 (/member/MichaelV2/) 2 years ago in reply to MichaelV2 (/member/MichaelV2/)
(/member/MichaelV2/)
Reply

I opened up the 4th file and typed the stuff in manually since it wasn't very much and this time it says that it can't move the PS3_connect_controller.sh file to ~/PS3control/ because that file doesn't exist



TonyCanning (author) 2 years ago in reply to MichaelV2 (/member/MichaelV2/)
(/member/TonyCanning/)
Reply

Hi Michael,

It's important that you download all files that have been included in Step #1.

It looks like you have downloaded files 1, 2, 3 & 4 but not the PS3_connect_controller.sh file.

If it's not downloaded, it can't move it.

I hope you get this going - it's not the easiest RPi project to start with!



MichaelV2 (/member/MichaelV2/) 2 years ago in reply to TonyCanning (/member/TonyCanning/)
(/member/MichaelV2/)
Reply

I made sure that I had the PS3_connect_controller.sh file in the but it still thinks that I don't have it whenever I try to run the tidyup script.

I have one goal with this project and that is to get my controller connected to my raspberry pi, but no other tutorials have succeeded so far, possibly because I have a fake brand controller... any ideas?



TonyCanning (author) 2 years ago in reply to MichaelV2 (/member/MichaelV2/)
(/member/TonyCanning/)
Reply

Hi Michael,

If you've downloaded the PS3_connect_controller.sh file then you definitely have it.

But here's what I'm thinking; I may not have made the instructions perfectly clear and I suspect we can sort this out! So let's step through the bits in the script files.....

1-dependencies just updates & installs some bits your RPi will need.

2-downloads creates a directory called "PS3control" then downloads other bits that are needed for the project to that directory.

3-buildfromsource.sh puts all that downloaded code together & installs it.

PS3_connect_controller.sh needs to be downloaded before doing the tidyup - I think you've downloaded it already but perhaps not to where the tidyup script expects it to be.

I'll assume that you can't currently find it. Here's a few commands that might help - open a terminal to run them....

try:

find ~/ -type f -name "PS3_connect_controller.sh"

- you should get an output that tells you where the file is. If you don't, then you haven't got the file on your system.

So, let's say you get a response which tells you that the file is somewhere you didn't expect. somewhere like:

/home/pi/project/PS3/madeupdirectory/PS3_connect_controller.sh

Well, the tidy up script thinks that this file is in /home/pi. The tidyup tries to move it from home/pi into a directory to keep it for use with the project. It is trying to place it into the /home/pi/PS3control/ which was created by one of the previous scripts.

If you've run tidyup.sh already and this is the only thing outstanding then you can use the move command to place it in the correct directory:

mv /home/pi/project/PS3/madeupdirectory/PS3_connect_controller.sh /home/pi/PS3control

4-tidyup.sh - as mentioned, this moves the PS3_connect controller.sh to the home/pi/PS3control/ directory. ... Once done, all the stuff downloaded so that you don't end up cluttering up your pi.

If you end up having frustrations with controllers, make sure you try it with a genuine one before writing off the code - some aftermarket ones just won't work....

I hope this works for you Michael, let me know!



seamster (/member/seamster/) 3 years ago
(/member/seamster/)
Reply

Nice! Thanks for sharing this!

Post Comment

Categories

Technology (/technology/)
Workshop (/workshop/)
Craft (/craft/)
Home (/home/)
Food (/food/)
Play (/play/)
Outside (/outside/)
Costumes (/costumes/)

About Us

Who We Are (/about/)
Why Publish? (/about/create.jsp)
Jobs (/community/Positions-available-at-Instructables/)
Contact (/contact/)

Resources

Residency Program (<http://www.autodesk.com/pier9/residency/home>)
Community (/topics/)
Sitemap (/sitemap/)
Help (/id/how-to-write-a-great-instructable/)

Find Us

(<http://www.instagram.com>)
 (<http://www.pinterest.com>)
 (<http://www.facebook.com>)
 (<http://www.twitter.com>)