

This repository | Search

Pull requestsIssuesGist

antibody-br / ServoControl

Unwatch1★ Star0Fork0

<> Code

🔔 Issues 0

🔗 Pull requests 0

📁 Projects 0

📖 Wiki

📶 Pulse

📊 Graphs

⚙ Settings

Remote servo control using Python, Arduino, OSC and TouchOSC App with Smartphone (iPhone iOS or Android)

Edit

remote

contol

python

osc

iphone

android

arduino

touchosc

ios

smartphone

Manage topics

📌 36 commits

🌿 1 branch

📦 0 releases

👤 1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

antibody-br committed on GitHub Update README.md

Latest commit d1c247b 4 minutes ago

image

Add files via upload

8 days ago

OSC_servo.py

Add files via upload

8 days ago

README.md

Update README.md

4 minutes ago

Readme.pdf

Add files via upload

8 days ago

Servo.touchosc

Add files via upload


8 days ago

Servo_Control.ino

Add files via upload

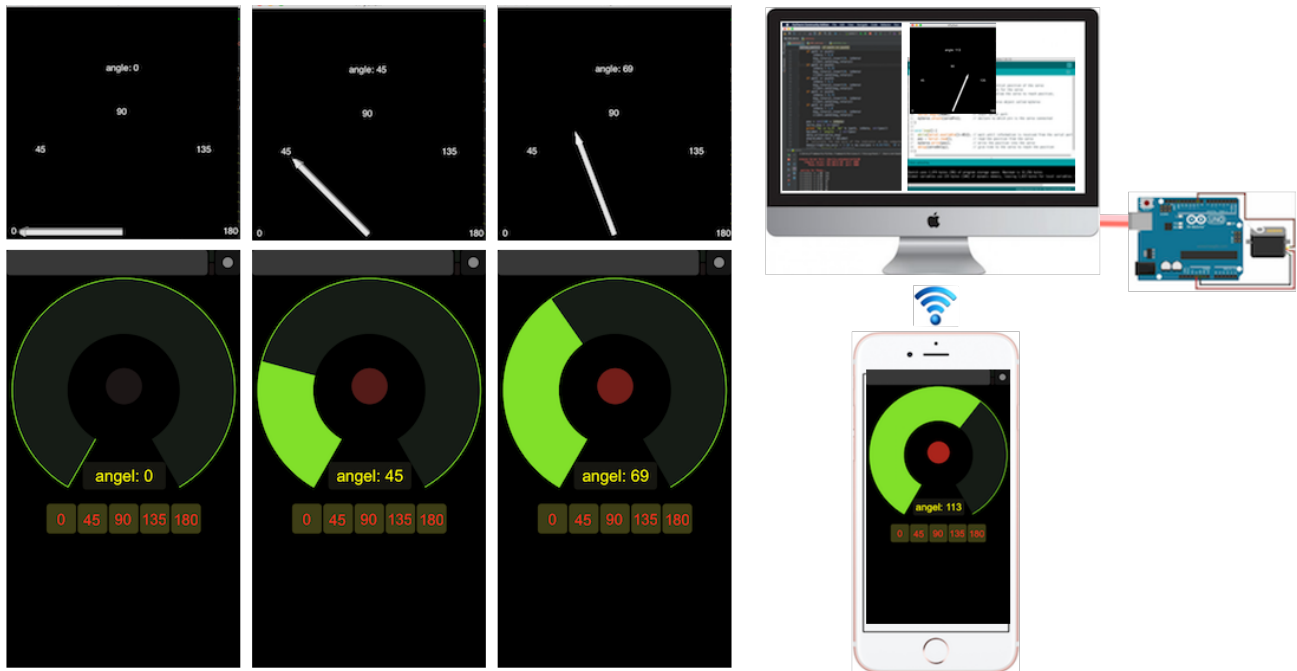
8 days ago

📖 README.md



ServoControl

Remote servo control using Python, Arduino, OSC and TouchOSC App with Smartphone (iPhone iOS or Android)



This program is based on the code and instructions from [SILVINO J. A. PRESA](#) <-THANK YOU! and adapted by me for remote control with the touchOSC app.

I am not an experience Python programmer, so I wrote a mickey-mouse code to control a servo with my iPhone. There are for sure much clever way's to do this, so feedback and improvements are very welcome.

For instance I use an UDP server to get the controller path from the touchOSC app and then I start an OSC server to control it. There must be a smarter way to do this.

so here we go:

index:

code	program	device
OSC_servo.py	python IDE	computer
Servo.touchOSC	touchOSC editor	computer => phone touchOSC
Servo_Control.ino	Arduino IDE	computer => Arduino

Steps:

- 1) Set up Arduino Circuit
- 2) Computer: load `Servo.touchosc` layout in TouchOSC editor
- 3) synchronize layout with Phone TouchOSC app.
- 4) load `Servo_Control.ino` sketch in Arduino IDE and compile to Arduino device
- 5) load `OSC_Control.py` in python IDE, and run it.
- 6) Now you can control the servo with your Phone

Requirements:

Software

on Mac or PC:

- **Python editor:** I prefer [PyCharm](#) (mac/win)
- [Arduino IDE](#)
- [touchOSC editor](#) => scroll to the end of the page: Downloads -> choose your OS.
- touchOSC app: [iOS](#) or [android](#) price: \$5

Python Modules

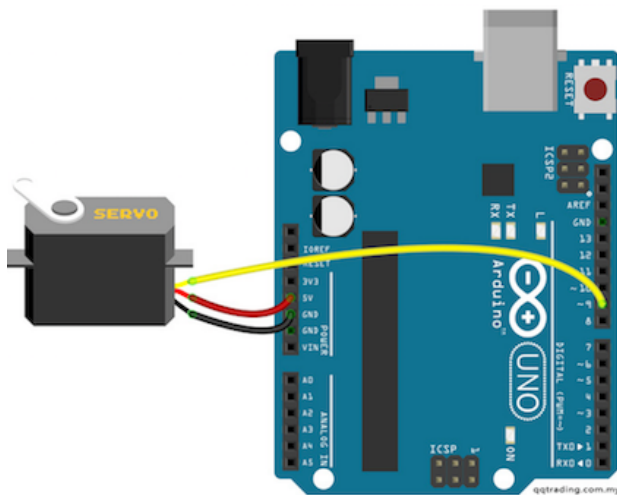
I use [Python 2.7.13](#). On the **Mac** be sure you use the **FrameWork version**, otherwise the Vpython graphic display will not work: `/opt/local/Library/Frameworks/Python.framework/Versions/2.7/bin/python2.7`

- **OSC:** Unix terminal: `$ pip install pyosc` or download: if you have trouble installing it, you can also put the `OSC.py` that is in the downloaded package, in the same folder you have `OSC_Servo.py`
- **Vpython:** `$ pip install vpython`
- **serial:** `$ pip install pyserial`
- **numpy:** `$ pip install numpy`
- **socket:** I guess it is a standard Python package, but had a lot of trouble getting it working. Be sure you have no other `socket.py` some where in your directories.

Hardware

- [Arduino Board \(Uno\)](#) and many other companies. \$15, My UNO comes from China (Oops!)
- **Servo Motor:** I use TowerPro SG90, do a google search and you get a lot of hits. Around \$5

Circuit diagram



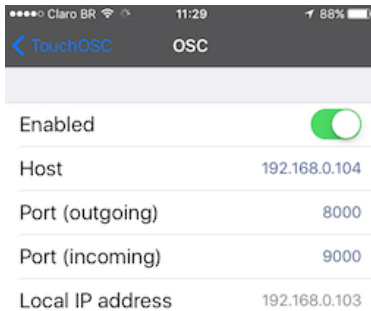
TouchOSC editor on Computer

Double click the `Servo.touchosc` file, and the `TouchOSC editor` should open or open it from inside the editor. Click the `Sync` button, in the upper right of the window => [touchOSC sync](#)

ToucOSC app on Phone

- click the white spot on upper right of the window, choose the upper item
- OSC: and fill in the IP address, variable in the OSC_Control.py => serverAdr = "192.168.0.104" <= **YOU HAVE TO CHANGE this to your computer IP address !!!**
- Port(outgoing) is set to 8000, variable in the OSC_Control.py => serverPort = 8000
- Port(incoming) is set to 9000, variable in the OSC_Control.py => clientPort = 9000
- Local IP address is variable in OSC_Control.py => clientAdr = "192.168.0.102" <= **YOU HAVE TO CHANGE this to your Phone IP address !!!**

Bellow the settings on my iPhone:



Return to <TouchOSC upper left corner and choose the item under LAYOUT, choose Add. Choose the host from the list. Now you can choose the Servo layout.

see also [touchOSC wifi transfer](#)

Arduino IDE

In the Arduino IDE on your computer: open the Servo_Control.ino file and send it to the Arduino. Be sure that the right type Arduino and serial Port is selected => Menubar -> Tools -> Board: / Port:

Now you see also the name of the **serialPort that you need to set** in OSC_Control.py => variable: serialPort = "/dev/tty.wchusbserialfa130". On the Mac it is similar like this, on WIN is some thing like COMn

see also: [serial port name](#)

First compile your sketch, **do not** open the serial monitor, **and then run** OSC_Control.py. Otherwise you get the message: **avrdude: ser_open(): can't open device "/dev/cu.wchusbserialfa130": Resource busy**

OSC_Control.py

to find your IP address: [Mac](#) or [Win](#)

****!!! change the variables according to your enviroment !!! ****

```
serialPort = '/dev/tty.wchusbserialfa130'
serverAdr = "192.168.0.104"
serverPort = 8000
clientAdr = "192.168.0.102"
clientPort = 9000
```

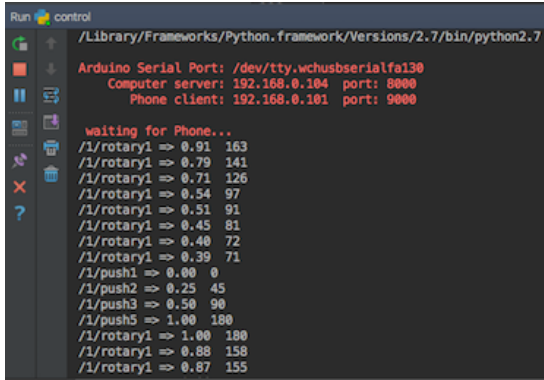
WARNING: if you not use a fix IP address, but get it from a DHCP server, the IP address can change, special on the Phone, that is re-connecting to your network if you return from an other place. error message is e.g:

OSCServer: NoCallbackError on request from 192.168.0.103:64550: No callback registered to handle OSC-address '/1/push2'

you see: if have my Phone IP defined as 192.168.0.102 but it is now: 192.168.0.103

Check again your touchOSC app to see if it is still the same.

it take a few moments to start, on my compter 20 sec., also its a bit slow to receive the fist messages. Keep turning the rotary till it gets it. Output looks like this:



```
Run control
/Library/Frameworks/Python.framework/Versions/2.7/bin/python2.7
Arduino Serial Port: /dev/tty.wchusbserialfa130
Computer server: 192.168.0.104 port: 8000
Phone client: 192.168.0.101 port: 9000

waiting for Phone...
/1/rotary1 => 0.91 163
/1/rotary1 => 0.79 141
/1/rotary1 => 0.71 126
/1/rotary1 => 0.54 97
/1/rotary1 => 0.51 91
/1/rotary1 => 0.45 81
/1/rotary1 => 0.40 72
/1/rotary1 => 0.39 71
/1/push1 => 0.00 0
/1/push2 => 0.25 45
/1/push3 => 0.50 90
/1/push5 => 1.00 180
/1/rotary1 => 1.00 180
/1/rotary1 => 0.88 158
/1/rotary1 => 0.87 155
```

conroller => output angle

Famous last words: "IT SHOULD WORK!"

Happy trouble shooting and debugging!!!

-b-

bert@temminck.net, April 2017, Anápolis-GO, BRASIL

