

Ubuntu 16.04 LTS in DigitalOcean.

Auto Installation.

```
adduser pysoc_server
usermod -aG sudo pysoc_server
su pysoc_server
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install git
cd /home/pysoc_server
git clone https://github.com/notalentgeek/pysoc
cd /home/pysoc_server/pysoc/script/script-setup
sudo ./setup_ubuntu_1604_server.sh
```

Manual Installation.

Raspbian Jessie in Raspberry PI 3B.

Auto Installation.

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install git
cd /home/pi
git clone https://github.com/notalentgeek/pysoc
cd /home/pi/pysoc/script/script-setup/
sudo ./setup_raspbian_jessie_rpi.sh
```

Manual Installation.

1. Update and upgrade Raspbian Jessie.
Call these commands from terminal.

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

2. Prevent general energy saving features and screen saver.
 1. Description.
Raspbian Jessie default screen saver is ten minutes and then goes to soft shut-down within the next 30 minutes. Pedge, need to have the operating system to be always available unless the user turned it off. Hence, general energy saving features should not active.
 2. Locate or create `/etc/xdg/lxsession/LXDE-pi/autostart`.
Add these lines.

```
@xset -dpms
@xset s noblank
@xset s off
```
 3. Locate or create `/etc/xdg/lxsession/LXDE/autostart`.
Add these lines.

```
@xset -dpms
@xset s noblank
```

@xset s off

4. Locate or create `/home/pi/.bashrc`.
Add this line.
`setterm -blank 0 -powerdown 0`
5. Locate `/etc/lightdm/lightdm.conf`.
Search and replace this line `#xserver-command=X` with `xserver-command=X -s 0 -dpms`.
6. Locate `/etc/kbd/config`.
Search and replace this line `BLANK_TIME=30` with `BLANK_TIME=0`.
Search and replace this line `POWERDOWN_TIME=30` with `POWERDOWN_TIME=0`.

2. Setup LIRC (Linux Infrared Remote Control).

1. Description.

LIRC is a library for Linux to interface general infrared devices. Although mostly related to TV infrared remote, LIRC can be used to assign infrared signals to running program. However, the infrared signals need to be assigned into constants those refer to TV infrared remote. In this setup I set to assign three infrared signals into constants (`KEY_1`, `KEY_2`, `KEY_3`).

2. Locate `/etc/modules`.

Add these lines.

```
lirc_dev
lirc_rpi gpio_in_pin=23 gpio_out_pin=22
```

3. Delete `/etc/lirc/hardware.conf`.

4. Create `/etc/lirc/hardware.conf`.

Add these lines.

```
#####
```

```
# /etc/lirc/hardware.conf
```

```
#
```

```
# Arguments which will be used when launching lircd
```

```
LIRCD_ARGS="--uinput"
```

```
# Do not start lircmd even if there seems to be a good config file
```

```
# START_LIRCMD=false
```

```
# Do not start irexec, even if a good config file seems to exist.
```

```
# START_IREXEC=false
```

```
# Try to load appropriate kernel modules
```

```
LOAD_MODULES=true
```

```
# Run "lircd --driver=help" for a list of supported drivers.
```

```
DRIVER="default"
```

```
# usually /dev/lirc0 is the correct setting for systems using udev
```

```
DEVICE="/dev/lirc0"
```

```
MODULES="lirc_rpi"
```

```
# Default configuration files for your hardware if any
```

```
LIRCD_CONF=""
```

```
LIRCMD_CONF=""
```

```
#####
```

5. Locate `/boot/config.txt`.

Add this line.

```
dtoverlay=lirc-rpi,gpio_in_pin=23,gpio_out_pin=22
```

6. Copy `/etc/lirc/lircd.conf` as `/etc/lirc/lircd_backup.conf`.

7. Delete `/etc/lirc/lircd.conf`.

8. Create `/etc/lirc/lircd.conf`.

Add these lines.

```
# Please make this file available to others
# by sending it to <lirc@bartelmus.de>
#
# this config file was automatically generated
# using lirc-0.9.0-pre1(default) on Sat Jan 7 22:45:56 2017
#
# contributed by
#
# brand: /home/pi/lircd.conf
# model no. of remote control:
# devices being controlled by this remote:
#
```

```
begin remote
  name pysoc
  bits 13
  flags RC5|CONST_LENGTH
  eps 30
  aeps 100
  one 924 840
  zero 924 840
  plead 970
  gap 113287
  toggle_bit_mask 0x0
  begin codes
    KEY_1 0x1001
    KEY_2 0x1002
    KEY_3 0x1003
  end codes
end remote
```

9. Delete `/home/pi/.lircrc`.

10. Create `/home/pi/.lircrc`.

Add these lines.

```
begin
  button = KEY_1
  config = KEY_1
  prog = pysoc
end
begin
  button = KEY_2
  config = KEY_2
  prog = pysoc
end
begin
  button = KEY_3
  config = KEY_3
  prog = pysoc
end
```

11. Delete `/etc/lirc/lircrc`.

12. Create `/etc/lirc/lircrc`.

Add these lines.

```
begin
  button = KEY_1
  config = KEY_1
  prog = pysoc
end
begin
  button = KEY_2
  config = KEY_2
  prog = pysoc
end
begin
  button = KEY_3
  config = KEY_3
  prog = pysoc
end
```

3. Setup PiCamera.

1. Description.

Raspberry PI has two ways to connect camera: USB web cam and PiCamera. USB web cam is the usual USB web cam you can find in general consumer electronics store. PiCamera is a ribbon cabled camera that is low powered and smaller than common USB web cam. However, despite PiCamera has its driver pre-installed in Raspbian Jessie, it different setup route than USB web cam.

2. Locate `/boot/config.txt`.

Search and replace or create this line `start_x=1`.

Search and replace or create this line `gpu_mem=128`.

4. Setup PyAudio.

1. Description.

PyAudio is a high level library to interface sound input and output. PyAudio works across common desktop operating system, Linux, MacOS, and Windows. However, there is a problem in one of its dependency, PortAudio. Hence, to solve this issue, you need to recompile the stable version of PortAudio.

2. Download PortAudio stable version 19.

Call this command from terminal.

```
wget http://www.portaudio.com/archives/pa_stable_v19_20140130.tgz -P /home/pi
```

3. Extract the source code.

Call this command from terminal.

```
tar xf /home/pi/pa_stable_v19_20140130.tgz -C /home/pi
```

4. Compile the source code.

Call these commands from terminal.

```
cd /home/pi/portaudio
```

```
./configure
```

```
make
```

```
sudo make install
```

```
cd /home/pi
```

5. Locate `/home/pi/.bashrc`.

Add these lines.

```
LD_LIBRARY_PATH="/usr/local/lib"
```

```
LD_RUN_PATH="/usr/local/lib"
```

```
PATH=$PATH:/usr/local/lib/
```

```
export LD_LIBRARY_PATH
```

```
export LD_RUN_PATH
```

export PATH

5. Setup audio input.

1. Description.

Raspberry PI does not offer audio input. Microphone should be plugged into external audio driver. However, to make this sadder, there are only several kind of audio drivers has its driver pre-installed in Raspbian. This USB audio driver <https://www.adafruit.com/product/1475> is one of them.

To interface the USB audio driver settings need to be done. For example to turn off the default audio channel and setting up Raspbian Jessie to be able to listen to upcoming audio input.

2. Delete */etc/asound.conf*.

3. Create */etc/asound.conf*.

Add these lines.

```
pcm.!default{
    type hw card 1
}
```

```
ctl.!default{
    type hw card 1
}
```

4. Delete */home/pi/.asoundrc*.

5. Create */home/pi/.asoundrc*.

Add these lines.

```
pcm.!default{
    type hw card 1
}
```

```
ctl.!default{
    type hw card 1
}
```

6. Locate */usr/share/alsa/alsa.conf*.

Search and replace this line *defaults.ctl.card 0* with *defaults.ctl.card 1*.

Search and replace this line *defaults.pcm.card 0* with *defaults.pcm.card 1*.

Search and replace this line *defaults.pcm.device 0* with *defaults.pcm.device 1*.

Search and replace this line *defaults.pcm.subdevice 0* with *defaults.pcm.subdevice -1*.

6. Setup Pysoc.

1. Download Pysoc.

Call these commands from terminal.

```
cd /home/pi
```

```
git clone https://notalentgeek.com/notalentgeek/pysoc
```

2. Download *pip3* packages.

Call this command from terminal.

```
sudo pip3 install -r /home/pi/pysoc/req/req_raspbian_jessie.txt
```

7. Reboot.

8. Run Pysoc.

Call this command from terminal.

```
cd /home/pi/pysoc && sudo python3 -B pysoc.py --rpi --picam
```