

Εγκατάσταση Raspberry PI OS

Το διαδικτυακό ραδιόφωνο λειτουργεί σε ένα Raspberry PI 3 με 1GB ram και κάρτα microSD των 16Gb. Η έκδοση είναι η Raspbian 11 βασισμένη στο Debian bullseye και συγκεκριμένα η Lite δηλαδή χωρίς GUI.

Κατεβάζω το image από το επίσημο site του raspberry.

Κατεβάζω την έκδοση Lite που είναι συμπιεσμένη σε xz.

Το αποσυμπιέζω και το γράφω στην sd με το balena etcher.

Ξεκινάω με οθόνη και επιλέγω πληκτρολόγιο, username και password και από raspi-config ξεκινάω τον ssh server. Ελέγχω ip με ifconfig και επανεκκίνηση και μετά συνδέομαι με ssh.

Από `sudo raspi-config` επιλέγω locale `en_GB.UTF-8` και `en_US.UTF-8` και ζώνη ώρας `Europe/Athens`.

Reboot

`sudo apt update` και `sudo apt upgrade` ώστε να πάρει τις πρόσφατες ενημερώσεις.

Από σύστημα το οποίο λειτουργεί ελέγχω την έκδοση του OS με:

```
lsb_release -a
No LSB modules are available.
Distributor ID: Raspbian
Description: Raspbian GNU/Linux 11 (bullseye)
Release: 11
Codename: bullseye
```

και αν είναι 32 ή 64bit με:

```
getconf LONG_BIT
32
```

Ρύθμιση WiFi

Από `raspi-config` επιλέγω χώρα και βάζω ασύρματο δίκτυο.

Εναλλακτικά διορθώνω το αρχείο `/etc/wpa_supplicant/wpa_supplicant.conf`:
`ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev`
`update_config=1`

```
network={
    ssid="Wifi1"
    psk="Secret1"
}

network={
    ssid="Wifi2"
    scan_ssid=1
    psk="Secret2"
}
```

Εγκατάσταση lirc

`sudo apt install lirc`

Στο `/etc/lirc/lirc_options.conf` διορθώνω τις τρεις πρώτες γραμμές:

```
[lircd]
nodaemon      = False
driver        = default
device        = /dev/lirc0
```

Στο αρχείο `/boot/config.txt` βγάζω σχόλιο από γραμμή και διορθώνω το 17 σε 18:

```
# Uncomment this to enable infrared communication.
dtoverlay=gpio-ir,gpio_pin=18
#dtoverlay=gpio-ir-tx,gpio_pin=18
```

Reboot

Γράφω `sudo mode2 -d /dev/lirc0` και πατάω κουμπιά στο RC για να δω αν λειτουργεί.

Με την εντολή:

```
irrecord --device /dev/lirc0 --driver default stav1.txt
```

Μαθαίνω τα κουμπιά του RC και τα αποθηκεύω στο αρχείο με όνομα `stav1.txt`. Με `irrecord --list-namespaces` | `more` βλέπω τα ονόματα των πλήκτρων που πρέπει να δίνω π.χ. `BTN_0` ή `BTN_1`.

Αφού είναι όλα καλά αντιγράφω το αρχείο στο `/etc/lirc/lircd.conf` και επανεκκινώ το lirc με

```
sudo systemctl restart lircd.service
```

Με `irw` ελέγχω αν καταλαβαίνει τα κουμπιά του RC με τα ονόματα που έδωσα.

```
stavros@WebRadio:/etc/lirc $ irw
```

```
00000000000001015 00 BTN_5 TV1
00000000000001015 01 BTN_5 TV1
00000000000001015 02 BTN_5 TV1
00000000000001015 03 BTN_5 TV1
00000000000001015 04 BTN_5 TV1
00000000000001018 00 BTN_8 TV1
00000000000001018 01 BTN_8 TV1
00000000000001018 02 BTN_8 TV1
00000000000001018 03 BTN_8 TV1
```

Κάνω σύνδεση από το vscode με remote SSH ή Remote Explorer και εγκαθιστά το vscode server και τα Python Extensions.

Από python δοκιμάζω το `ir_test.py` και ελέγχω αν λειτουργεί σωστά το RC.

Εγκατάσταση Mplayer

```
sudo apt install mplayer
```

Στο αρχείο `~/.mplayer/config` προσθέτω γραμμή `nolirc = yes` ώστε να μην εμφανίζει το μήνυμα:

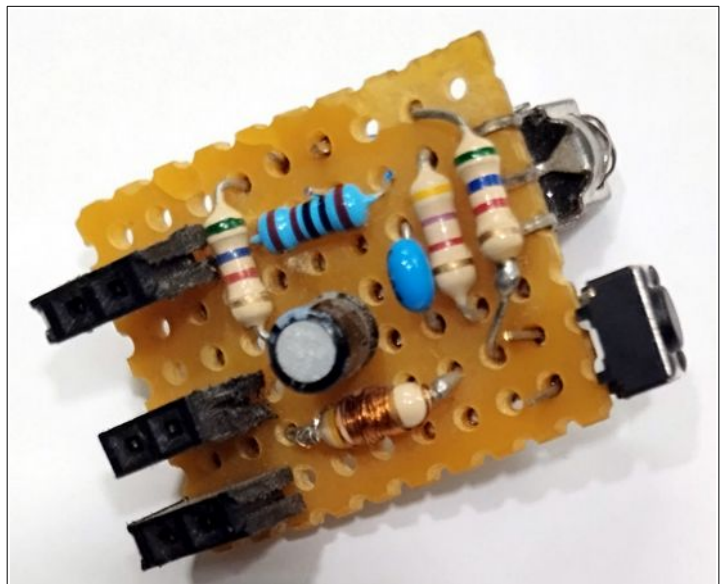
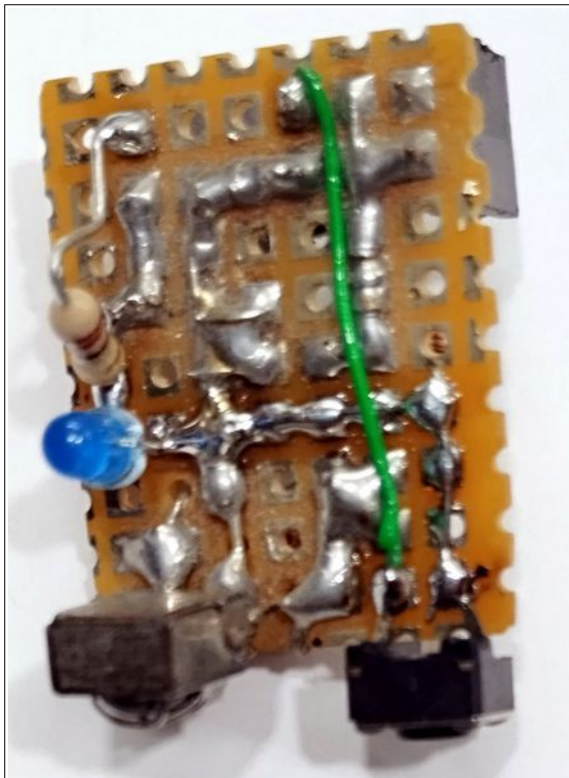
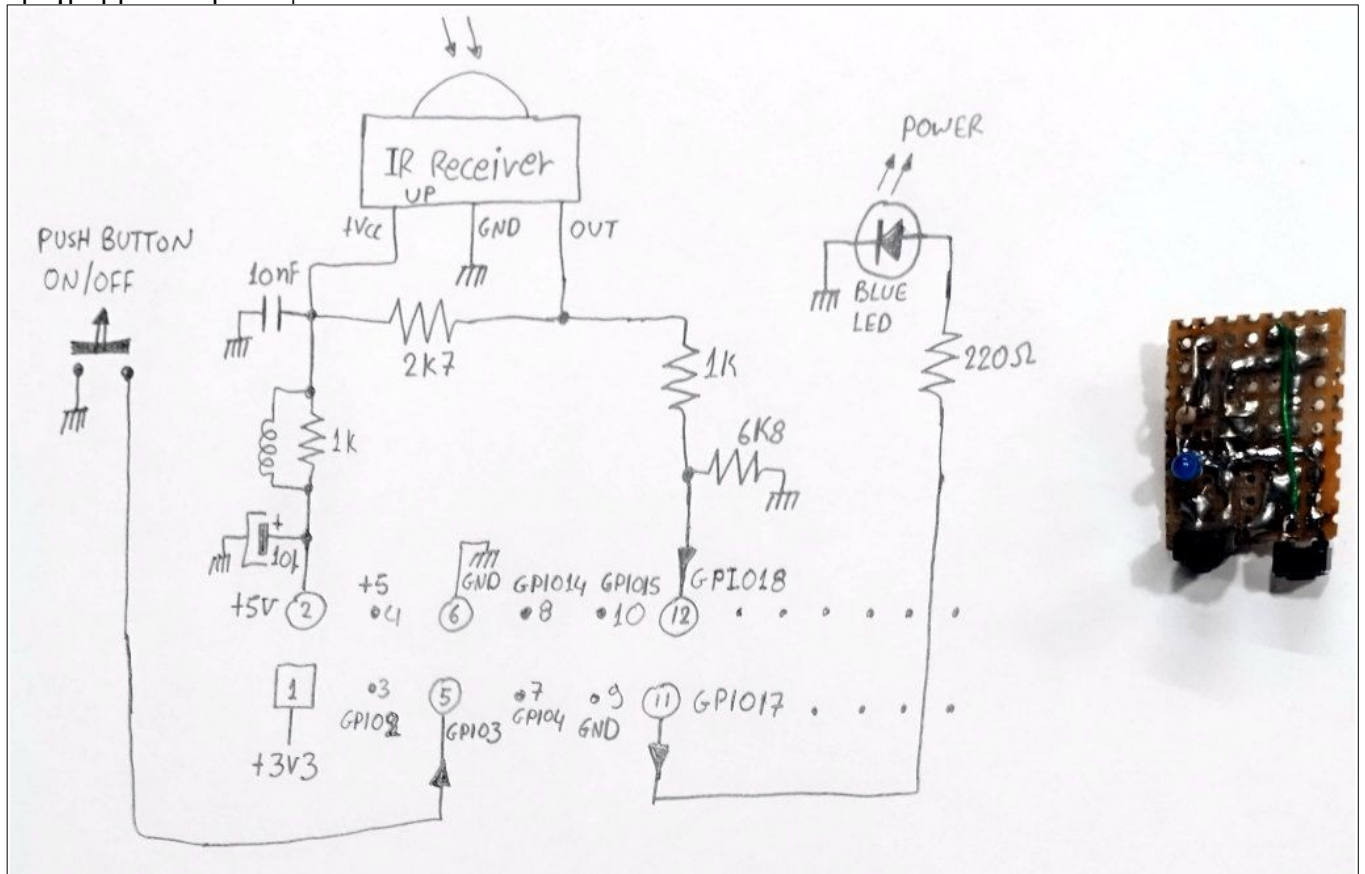
```
mplayer: could not open config files /home/stavros/.lircrc and /etc/lirc/lircrc
```

Εγκατάσταση espeak για φωνητικά μηνύματα offline.

```
sudo apt install espeak
```

Το υλικό

Αυτό είναι το θεωρητικό κύκλωμα του δέκτη IR το οποίο έχει κατασκευαστεί σε διάτρητη πλακέτα. Έχει και ένα push button για εκκίνηση – τερματισμό και ένα LED το οποίο ανάβει όταν λειτουργεί το ρυθμόν πρόγραμμα του ραδιοφώνου.



Και εδώ φαίνεται το σύστημα σε λειτουργία.



Raspberry PI radio ως service

Αρχικά φτιάχνω το πρόγραμμα σε python 3 και το δοκιμάζω από το ~ με ./radio.py.
Φτιάχνω αρχείο στο *lib/systemd/system* και όνομα webradio.service.

```
[Unit]
Description=My Web Radio App
After=multi-user.target

[Service]
Type=simple
Restart=on-failure
User=stavros
WorkingDirectory=/home/stavros
ExecStart=/home/stavros/radio.py

[Install]
WantedBy=multi-user.target
```

και μετά γράφω:

```
sudo systemctl daemon-reload
sudo systemctl enable webradio
sudo systemctl start webradio
ή
sudo systemctl restart webradio
ή
sudo systemctl stop webradio
```

για να απενεργοποιήσω την υπηρεσία γράφω:
`sudo systemctl disable webradio`

για να ελέγξω την κατάσταση της υπηρεσίας γράφω:
`sudo systemctl status webradio`

```
• webradio.service - My Web Radio App
  Loaded: loaded (/lib/systemd/system/webradio.service; enabled; vendor preset: enabled)
  Active: active (running) since Sun 2024-09-15 22:54:10 EEST; 4min 16s ago
  Main PID: 2336 (python)
    Tasks: 4 (limit: 1595)
      CPU: 18.491s
  CGroup: /system.slice/webradio.service
          └─2336 python /home/stavros/radio.py
            └─2419 mplayer -really-quiet http://nitro.live24.gr/nitro4555
              └─2420 mplayer -really-quiet http://nitro.live24.gr/nitro4555
```

Sep 15 22:54:10 raspberrypi systemd[1]: Started My Web Radio App.

Με επανεκκίνηση δοκιμάζω την σωστή λειτουργία.

Το γραφικό περιβάλλον διαχείρισης φαίνεται παρακάτω:

Python Web Radio - Google Chrome

Python Web Radio x KOSMOS - EPTecho x +

Μη ασφαλής 192.168.42.25:8080/ctrl:vol=88 ☆

Εφαρμογές Xiaomi Redmi... Athens Voic... webmail.sch.g... Όλοι οι σελιδοδείκτες

Stavros Radio

Το raspberry PI λειτουργεί ως ραδιόφωνο.

Παίζει τώρα: Δεύτερο λαϊκά

Ένταση φωνής: 88%

1 Rebel	2 Zeppelin	3 Athens Voice	4 Galaxy 92 Αθήνα	5 Kiss-FM Αθήνα
6 Velvet Αθήνα	7 Rainbow Θεσσαλονίκη	8 Velvet Θεσσαλονίκη	9 Rock FM 96.9 Αθήνα	10 Rock Radio 104.7 Θεσσαλονίκη
11 1055 Rock Θεσσαλονίκη	12 DRS Ηράκλειο Κρήτης	13 Pepper 96.6 Αθήνα	14 Δίεση 101.3 Αθήνα	15 Μελωδία 99.2 Αθήνα
16 Πάλκο	17 Love Radio 97.5 Αθήνα	18 Ελληνικός	19 More Θεσσαλονίκη	20 Vanilla Smooth
21 Vanilla Fresh	22 Vanilla Deep	23 Δεύτερο πρόγραμμα	24 Δεύτερο λαϊκά	25 Κόσμος

<< Προηγούμενος Επόμενος >>

(c)2023-24 Stavros S. Fotoglou