

OscIntRadio

Project: OscIntRadio

Author: Staffan Melin, staffan.melin@oscillator.se

License: GNU General Public License v3.0

Version: 20200418

Project site: <http://www.oscillator.se/opensource>

Introduction

The goal of this project is to use a Raspberry Pi (RPi) as an internet radio player that you can connect to your stereo amplifier (or active speakers or headphones). The radio is controlled using a web interface, so you can use any device connected to your Wi-Fi as a controller (computer or smartphone).

This guide assumes you know how to operate a command prompt and use some basic text commands.

Table of Contents

Introduction.....	1
How to use it.....	2
How to build it.....	4
1 Equipment.....	4
2 Install Raspbian.....	4
A. Install Raspbian without a desktop environment.....	4
B. Install Raspbian with a desktop environment.....	4
C. Modify an existing installation.....	5
3 Connection.....	5
4 SSH communication.....	5
5 Settings.....	6
6 Install mpd.....	6
7 Install a web server.....	6
8 Copy PHP program code and example playlist.....	7
9 Test.....	7
10 Playlist.....	7
Internet radio station directories.....	8
Todo.....	8
Interesting links.....	8

How to use it

Connect your RPi to your network and connect it to power.

Let it boot (it should take less than a minute).

With another device (smartphone, laptop) connected to the same wifi, start a browser and visit the homepage on the RPi.

To connect you will have to lookup the ip (eg 192.168.1.85) of the RPi.

As I run GNU/Linux, I used the command

```
nmap -sL 192.168.1.0/24
```

in a terminal to list all connected devices.

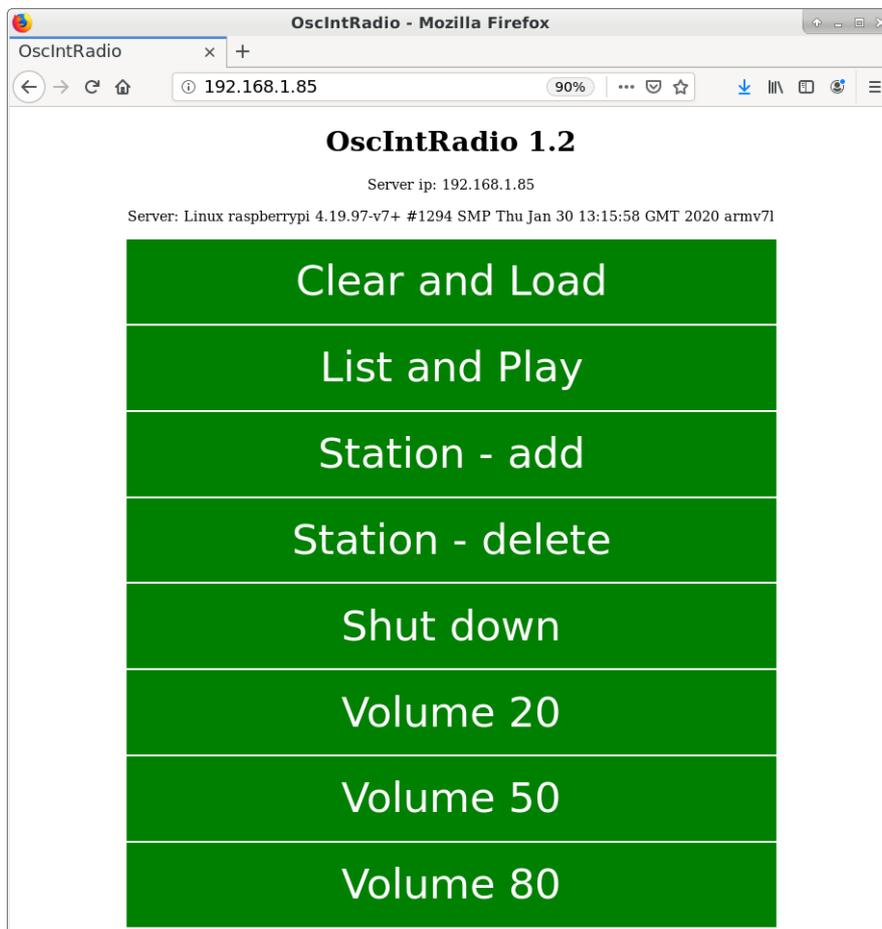
It will show a line something like this

```
Nmap scan report for raspberrypi.lan (192.168.1.85)
```

which means that your RPi is connected to the ip 192.168.1.85.

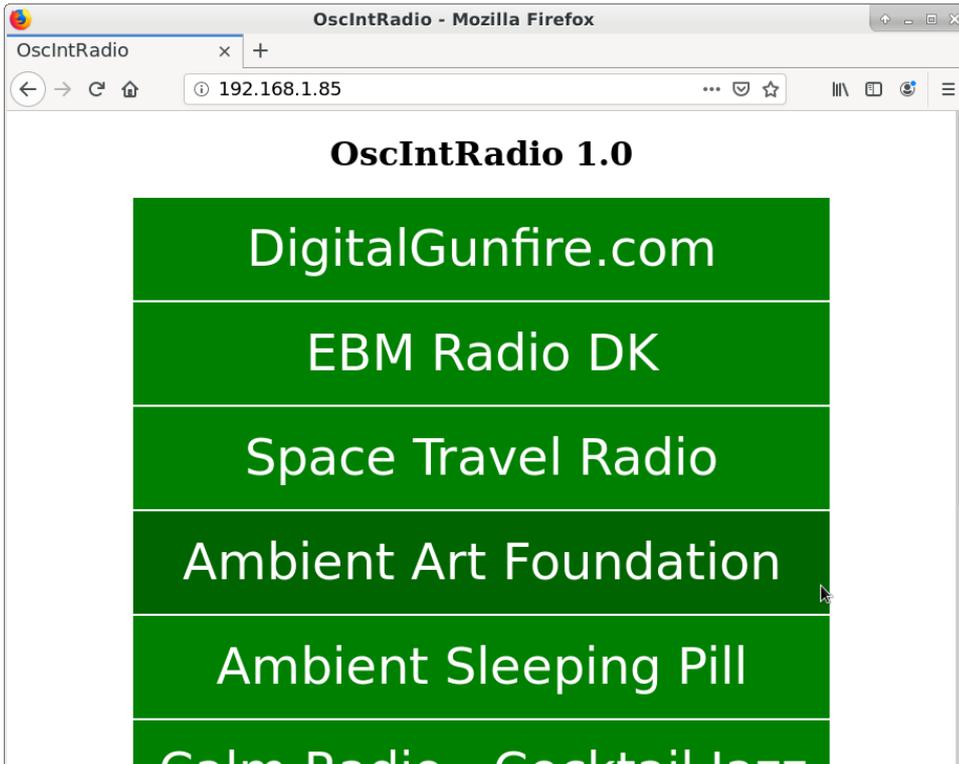
If you have an Android phone you can use Network Discovery (on F-Droid) or for example Network Scanner (on Play Store).

Type this ip number into your browser and you will see the OscIntRadio homepage:



Click on "Clear and Load" to reset the radio and load the playlist.

Click on "List and Play" to select a radio station to play:



Click on the channel of your choice.

Click on "Shut down" to shut down your RPi before you disconnect it from power.

Add and delete stations and change the volume with the other options. Note that you will have to click "Clear and Load" to play newly added stations.

My wifi router always seem to assign the same ip to my RPi, so I don't have to look it up every time.

A good place to find new stations is <http://www.radio-browser.info>.

How to build it

1 Equipment

- a computer running GNU/Linux (you can also use a computer with another OS but then my instructions can't be followed exactly)
- a Raspberry Pi (RPi)
- if you have a RPi without wifi you need a wifi dongle
- SD-card
- card reader (if your computer hasn't got one built in)
- local network (wifi router)
- wifi connection

2 Install Raspian

Download and install Raspian using the instructions on:

```
https://www.raspberrypi.org/downloads/raspbian/
```

and

```
https://www.raspberrypi.org/documentation/installation/installing-images/README.md
```

As we are not going to connect the RPi to a monitor, keyboard and mouse, the desktop software is not necessary. We are going to run our RPI headless.

There are three different roads to take.

A. Install Raspian without a desktop environment

Download Raspbian Buster Lite and install it.

While you have the SD-card still inserted in the computer, create a file named "ssh" on the boot partition so enable SSH (see below).

B. Install Raspbian with a desktop environment

Download "Raspian Buster with desktop" and install it.

Connect it to a screen, a keyboard and a mouse and use

```
sudo raspi-config
```

to make it boot into CLI, ie without a desktop environment (raspi-config > 3 Boot options).

At the same time you can enable SSH (raspi-config > 5 Interfacing options).

C. Modify an existing installation

If you already have a RPi running Raspian you can make it headless by removing everything associated with the window system:

```
sudo apt-get remove --auto-remove --purge 'libx11-.*'  
sudo apt-get autoremove --purge
```

(Repeat apt-get autoremove --purge until no orphans remain)

Source: <https://raspberrypi.stackexchange.com/questions/4745/how-to-uninstall-x-server-and-desktop-manager-when-running-as-headless-server>

Remember to enable SSH before you remove the window system. If not, you can always enable SSH by putting a file named "ssh" on the boot partition using another computer.

Useful links:

- <https://howtoraspberrypi.com/how-to-raspberry-pi-headless-setup/>

3 Connection

For this project to work the RPi has to be connected to your wifi.

If you don't have a RPi with wifi, and you don't have a wifi dongle (USB wifi network card), you can connect your RPi using the ethernet port.

See these guides on how to connect to a wifi network on a headless machine:

- <https://desertbot.io/blog/headless-raspberry-pi-3-bplus-ssh-wifi-setup>
- <https://styxit.com/2017/03/14/headless-raspberry-setup.html>

```
My wifi dongle was a RTL 8188EUS, so I had to follow the  
instructions on  
  
https://www.raspberrypi.org/forums/viewtopic.php?  
uid=81098&f=28&t=62371&start=0#p462982  
to make it work.
```

4 SSH communication

You communicate with the headless RPi using SSH.

As I run GNU/Linux, I used the command

```
nmap -sL 192.168.1.0/24
```

in a terminal to list all connected devices.

It will show a line something like this

```
Nmap scan report for raspberrypi.lan (192.168.1.85)
```

which means that your RPi is connected to the ip 192.168.1.85.

Start a terminal window on your computer (not your RPi) and enter:

```
ssh pi@192.168.1.85
```

Login with your password (the user is usually "pi"). Per default the password is "raspberry".

The password can be changed by;

```
sudo raspi-config
```

5 Settings

It is also a good idea to force the audio output to the audio output jack using:

```
sudo raspi-config
```

Go to "7 Advanced Options" and select "A4 Audio".

Now reboot your RPi.

Now we have a RPi running Raspbian, connected to your Wi-Fi, and accessible using SSH.

6 Install mpd

Now we can install mpc and mpd - the music player client and 'daemon' background controller:

```
sudo apt-get install mpd mpc
```

The commands we are going to use in our PHP code are:

- mpc load oscplaylist - load playlist
- mpc playlist oscplaylist - lists all items in playlist
- mpc play <pos> - play <pos> in playlist

7 Install a web server

Our user interface will be provided by a web server on the RPi. The web server we will use is called Apache. Install it using the following commands:

```
sudo apt install apache2
```

Enter these two commands to easier manage your sites files:

```
sudo chown -R pi:www-data /var/www/html/
```

```
sudo chmod -R 770 /var/www/html/
```

You also need to install PHP, the language our user interface is written in:

```
sudo apt install php php-mbstring
```

You don't need to install a database engine.

Source: <https://howtoraspberrypi.com/how-to-install-web-server-raspberry-pi-lamp/>

Useful links:

- <https://www.raspberrypi.org/documentation/configuration/security.md>

- <https://www.instructables.com/id/Simple-and-intuitive-web-interface-for-your-Raspbe/>

As the web interface should be able to shut down the RPi, you have to add the following line

```
www-data ALL=(ALL) NOPASSWD: /sbin/shutdown
```

to the file `/etc/sudoers` using the command

```
sudo nano /etc/sudoers
```

which launches a simple text editor. Make the changes, press CTRL+O to save and CTRL+X to exit.

Source: <https://raspberrypi.stackexchange.com/questions/26357/shutting-down-raspberry-pi-with-php>

8 Copy PHP program code and example playlist

To complete the project you have to copy the included PHP file (`index.php`) into the directory `/var/www/html`.

You also have to copy the example playlist (`oscpplaylist.m3u`) into the directory `/var/lib/mpd/playlists/`.

Enter the three commands below to allow the PHP program in `index.php` to modify the playlist:

```
sudo usermod -a -G audio www-data
sudo chmod g=rw /var/lib/mpd/playlists/oscpplaylist.m3u
sudo reboot
```

Note that the Raspberry Pi reboots after the last statement.

Useful links:

- <https://www.poftut.com/execute-shell-commands-php-exec-examples/>
- <http://www.hackingwithphp.com/4/12/0/executing-external-programs>

9 Test

To test your RPi, follow the "How to use it" section above.

10 Playlist

All stations are kept in a text file called a playlist. The playlist is located on the RPi in the following place: `/var/lib/mpd/playlists/oscpplaylist.m3u`

The format of this file is as follows:

```
#EXTM3U
#EXTINF:0,DigitalGunfire.com
http://dg.darkonline.org:8080/listen
```

The file starts with the line `"#EXTM3U"`

Every internet radio channel takes two lines:

1. "EXTINF:0," followed by the name of the channel (this is shown on the web interface)
2. The URL (web address) of the channel

URLs for radio stations can be found on the web, for example at www.radio-browser.info.

Internet radio station directories

- <http://www.radio-browser.info>
- <https://directory.shoutcast.com>

Todo

Ideas for improving OscIntRadio.

Modify the web interface:

- use tags on channels
- sleep mode that shuts down the server after a specified amount of time

Other:

- assign a domain name or static ip
- make a user interface with hardware (inspiration: <http://lofi.fi/using-a-arduino-lcd-shield-with-raspberry-pi-and-node/>)

Interesting links

<https://www.instructables.com/id/Raspberry-Pi-Internet-Radio/>

<https://www.pimusicbox.com/>

https://bobrathbone.com/raspberrypi/pi_internet_radio.html