



HAM Radio avec le Raspberry Pi

Partie 8



Sommaire :

Partie 8 : HamPi et RadioPi



Prérequis : Découverte de la Raspberry Pi partie 1

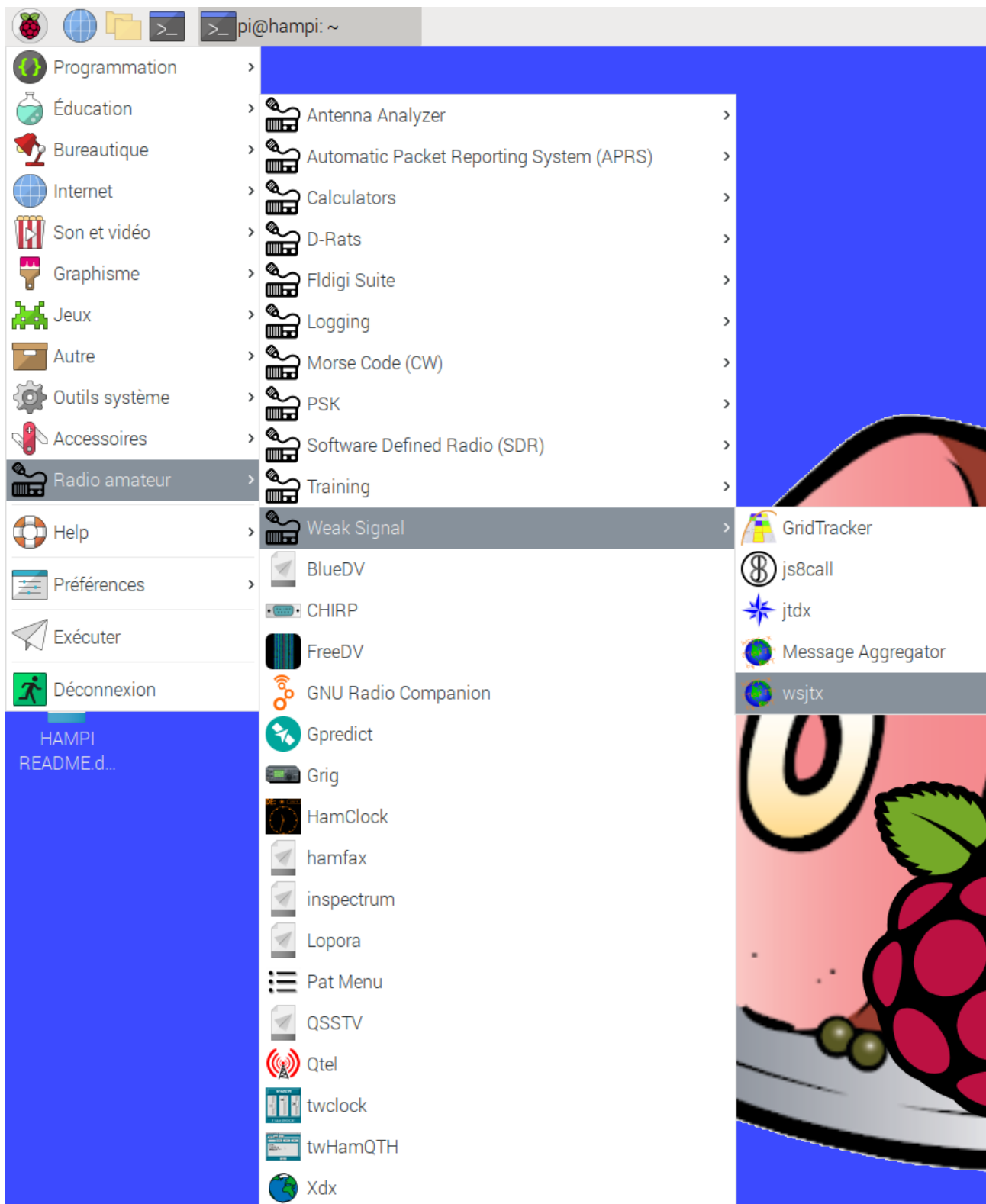
8.1 Présentation de HamPi	P. 2
8.2 Installation de l'image disque HamPi	P. 3
8.3 Prise en main de HamPi	P. 5
8.4 Présentation de radioPi	P. 9
8.5 Installation de l'image disque radioPi	P. 10
8.6 Prise en main de radioPi	P. 11
Annexe	P. 13

Version du 25/09/2021 V1.3

8.1 Présentation de HamPi

HamPi 1.x est une distribution Raspbian contenant plus de 100 logiciels préinstallés créée par Dave Slotter, W3DJS. La liste des logiciels préinstallés est en annexe page 9.

L'organisation du menu contenant les logiciels est très bien réalisée.



8.2 Installation de l'image disque

Commencer par se rendre sur le site officiel.

<https://github.com/dslotter/HamPi>

Pour connaître les nouveautés de la dernière version.

Mais le lien de téléchargement est ici

<https://sourceforge.net/projects/hampi/>

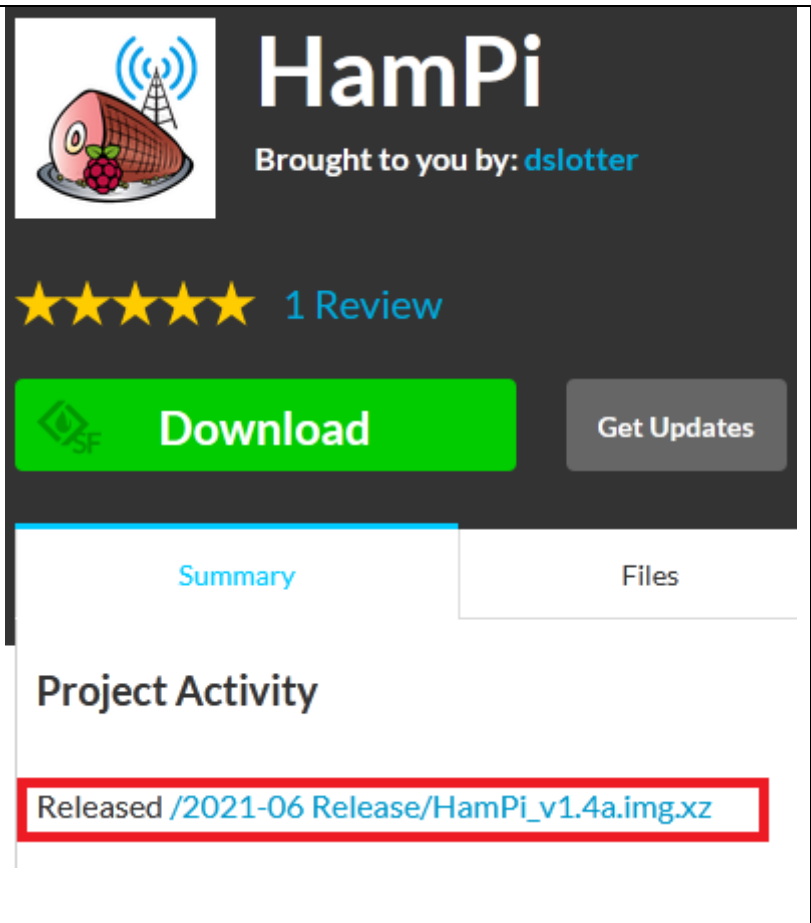
Puis télécharger le fichier


HamPi_vx.x.img.xz

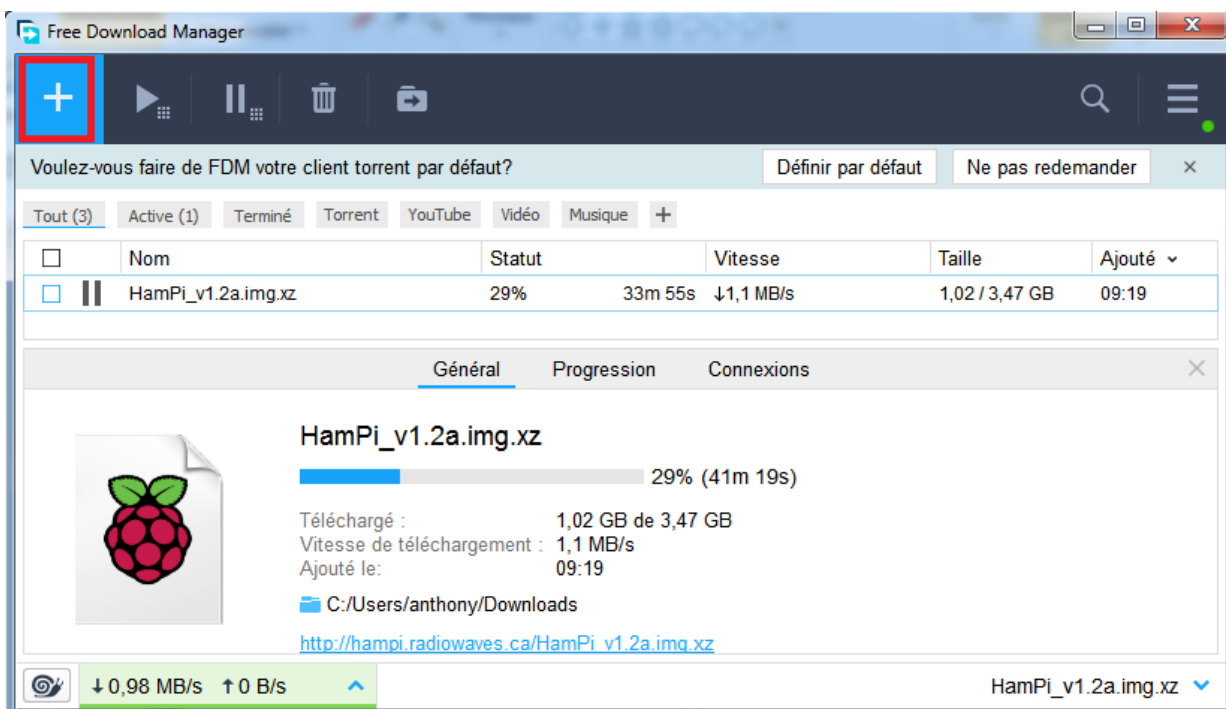
Celui-ci fait 2 à 3 Go

Il faudra prévoir une carte mSD de **32Go**

Les copies d'écran montrent l'installation de la version v1.2a, le procédé est identique pour les versions plus récentes

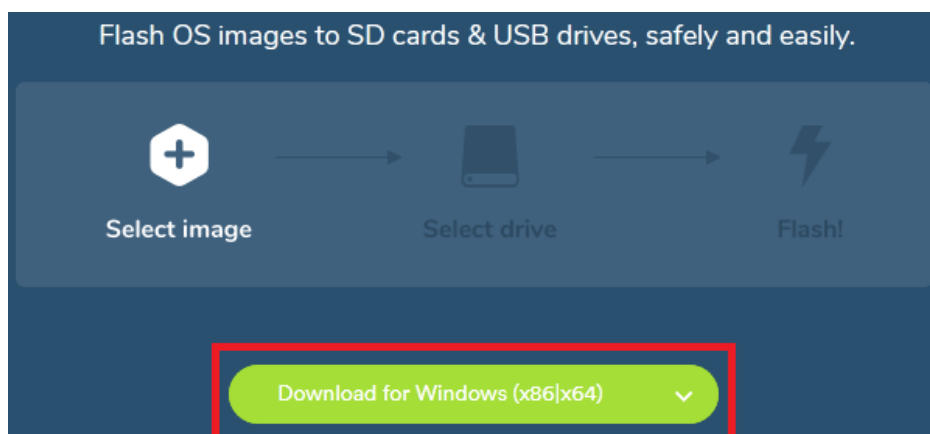


Pour faciliter le téléchargement, utiliser l'utilitaire [Free Download Manager](#). Cela permet de reprendre le téléchargement en cas d'interruption. Après avoir cliqué sur l'icône , coller l'URL du fichier à télécharger.



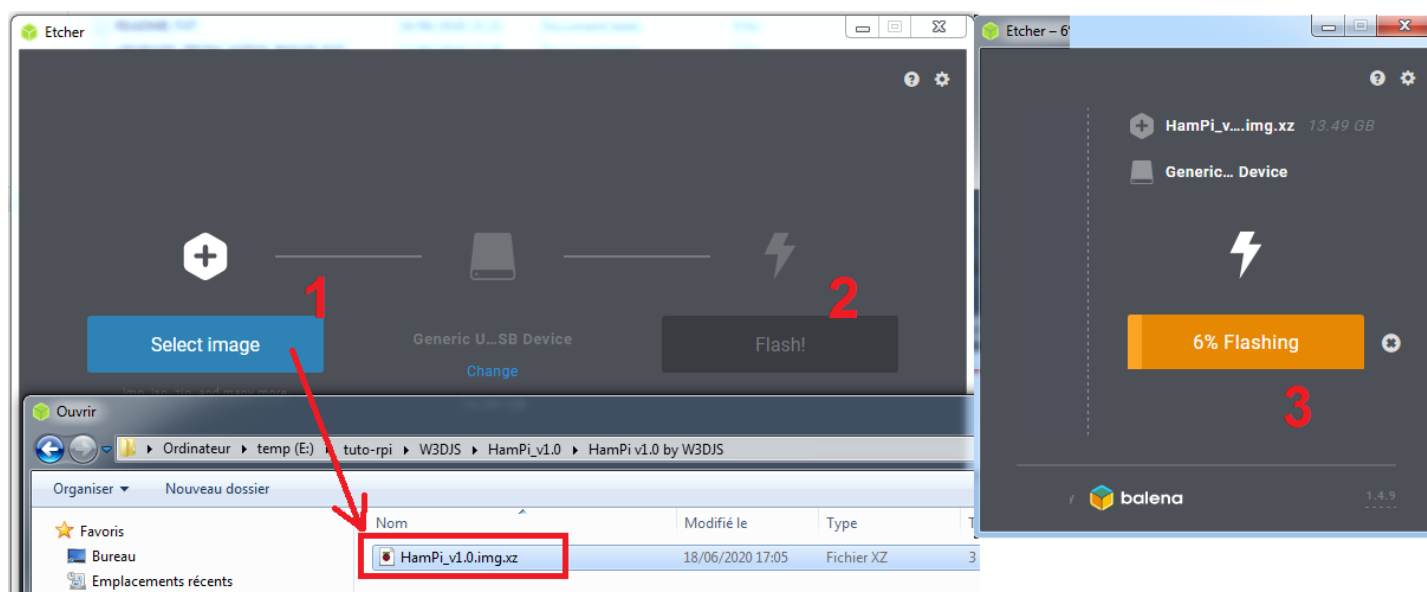
Télécharger et installer l'utilitaire Etcher pour copier l'image sur une carte mSD

<https://www.balena.io/etcher/>



Sélectionner le fichier **HamPi_vx.x.img.xz**, le lecteur de la carte mSD, puis cliquer sur Flash.

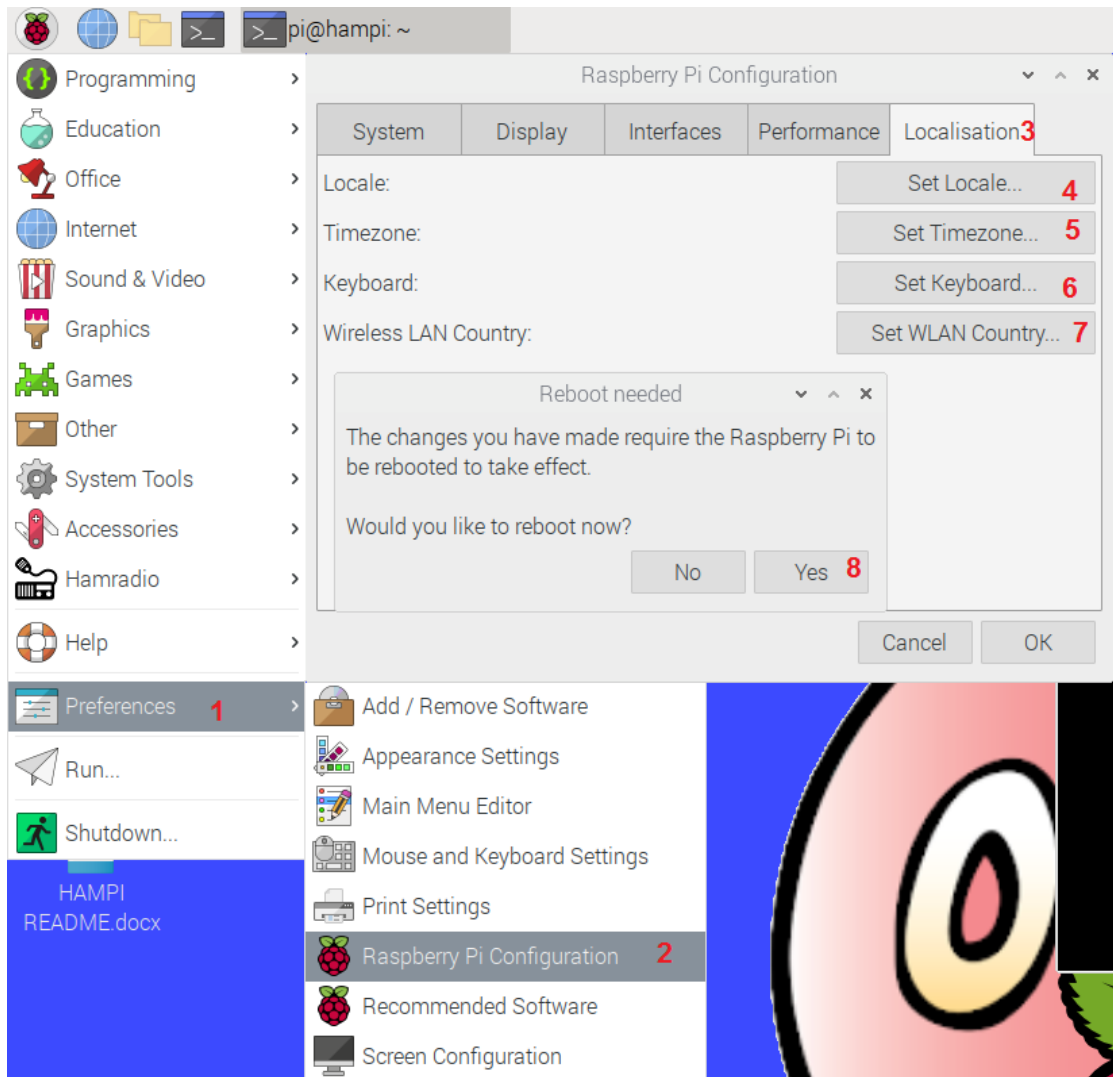
Attention, une carte mSD de **32Go** est nécessaire



Il faudra attendre un bon quart d'heure le temps que l'écriture et la vérification soient terminées.

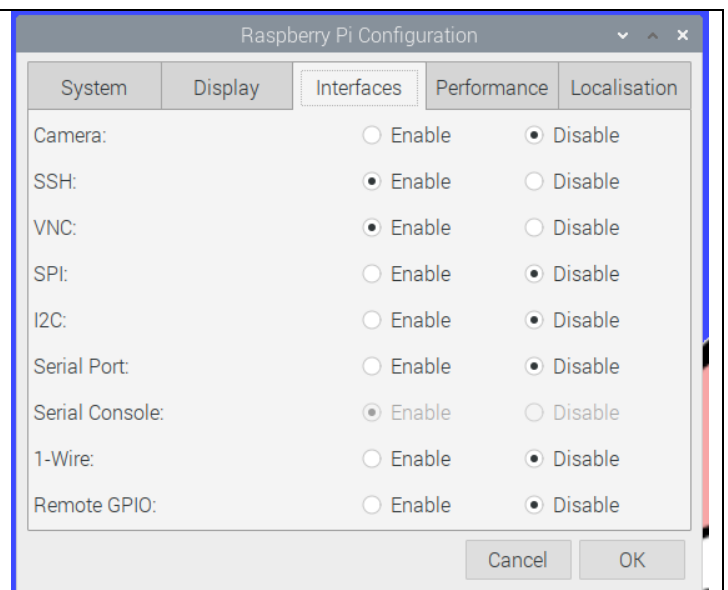
8.3 Prise en main de HamPi

Dès la mise sous tension, il faudra configurer le pays et la langue comme expliqué dans la partie 1



Par défaut la connexion en SSh et VNC est activée.

Le contrôle de la Raspberry via PuTTY en SSh et l'utilisation de VNC viewer sont expliqués dans la 1^{ère} partie de mes tutoriels.



J'ai commencé par brancher la clé `rtl-sdr`, mais la commande `rtl_test` ne fonctionnait pas (même avec la version 1.4a). Il a fallu que je fasse la manipulation expliquée dans la partie 3 de mes tutoriels, page 6 avec le terminal en ligne de commande.

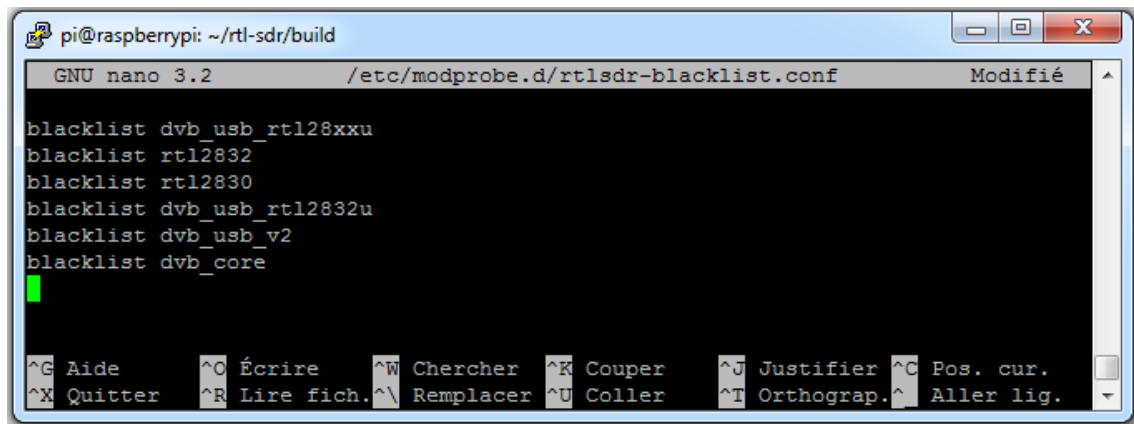
```
sudo nano /etc/modprobe.d/rtlsdr-blacklist.conf
```

```
pi@raspberrypi:~/rtl-sdr/build $ sudo nano /etc/modprobe.d/rtlsdr-blacklist.conf
```

```
blacklist dvb_usb_rtl28xxu
blacklist rtl2832
blacklist rtl2830
blacklist dvb_usb_rtl2832u
blacklist dvb_usb_v2
blacklist dvb_core
```

Copier-coller la liste ci-dessus dans l'éditeur nano

Rappel clic droit pour coller le texte dans l'éditeur nano



Enregistrer le fichier à l'aide des touches **Ctrl + o**, puis la touche entrée pour valider l'enregistrement.



Sortir de l'éditeur à l'aide des touches **Ctrl + x**.



Redémarrer le Raspberry Pi

```
pi@raspberrypi:~/rtl-sdr/build $ sudo reboot
```

sudo reboot

Vérification : Brancher la clé rtl-sdr sur un port USB de la Raspberry Pi.
Avec LXTerminal ou Putty, taper la ligne de commande suivante :

lsusb

Localiser la clé rtl-sdr : **RTL238 DVB-T**

Cas de la Raspberry Pi 4 :

```
pi@raspberrypi:~ $ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 003: ID 0bda:2838 Realtek Semiconductor Corp. RTL2838 DVB-T
Bus 001 Device 002: ID 2109:3431 VIA Labs, Inc. Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Avec LXTerminal ou Putty, taper la ligne de commande suivante : (attention **tiret du bas**)

rtl_test

La clé rtl-sdr devrait être reconnue, si ce n'est pas le cas, débrancher et rebrancher la clé et recommencer le test.

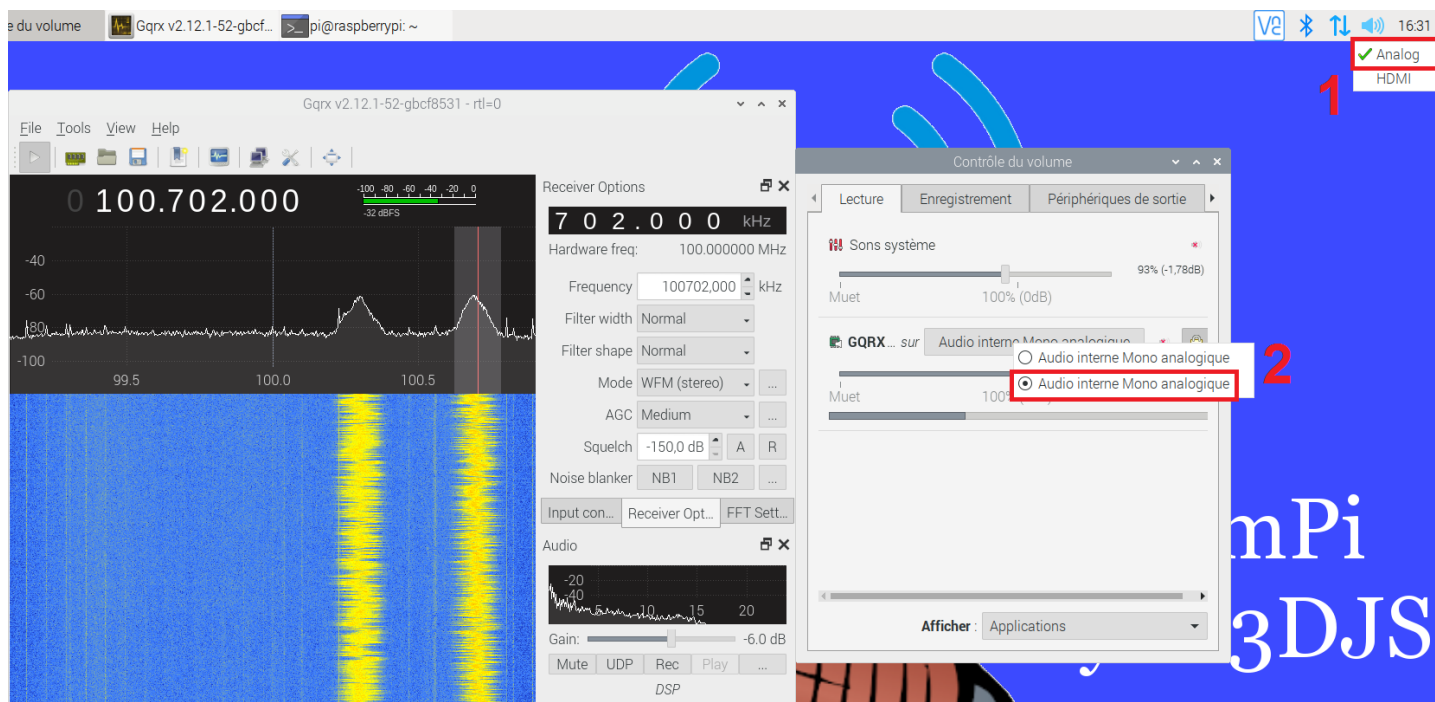
```
pi@raspberrypi:~ $ rtl_test
Found 1 device(s):
 0: Realtek, RTL2838UHIDIR, SN: 00000001

Using device 0: Generic RTL2832U OEM
Found Rafael Micro R820T tuner
Supported gain values (29): 0.0 0.9 1.4 2.7 3.7 7.7 8.7 12.5 14.4 15.7 16.6 19.7
 20.7 22.9 25.4 28.0 29.7 32.8 33.8 36.4 37.2 38.6 40.2 42.1 43.4 43.9 44.5 48.0
 49.6
[R82XX] PLL not locked!
Sampling at 2048000 S/s.

Info: This tool will continuously read from the device, and report if
samples get lost. If you observe no further output, everything is fine.

Reading samples in async mode...
^Csignal caught, exiting!
User cancel, exiting...
Samples per million lost (minimum): 0
pi@raspberrypi:~ $ ^C
```

Avec le logiciel GQRX, j'ai testé la réception correcte de la bande FM 88-108Mhz



Mais le son ne fonctionnait pas sur la sortie jack 3.5.

- 1- Cliquer alors avec le bouton droit sur l'icône du haut-parleur, puis vérifier **Analog**.
- 2- Avec l'utilitaire contrôle du « volume pulse audio », cliquer (bouton droit) sur « audio interne mono analogique » et sélectionner la 2eme ligne.

Je n'ai pas le temps de tester tous les logiciels. Chaque OM devra réaliser ses propres essais.

Il existe sur GitHub (inscription requise) une page d'aide où on peut poser des questions ainsi qu'un groupe d'utilisateurs de Raspberry Pi autour de la radio.

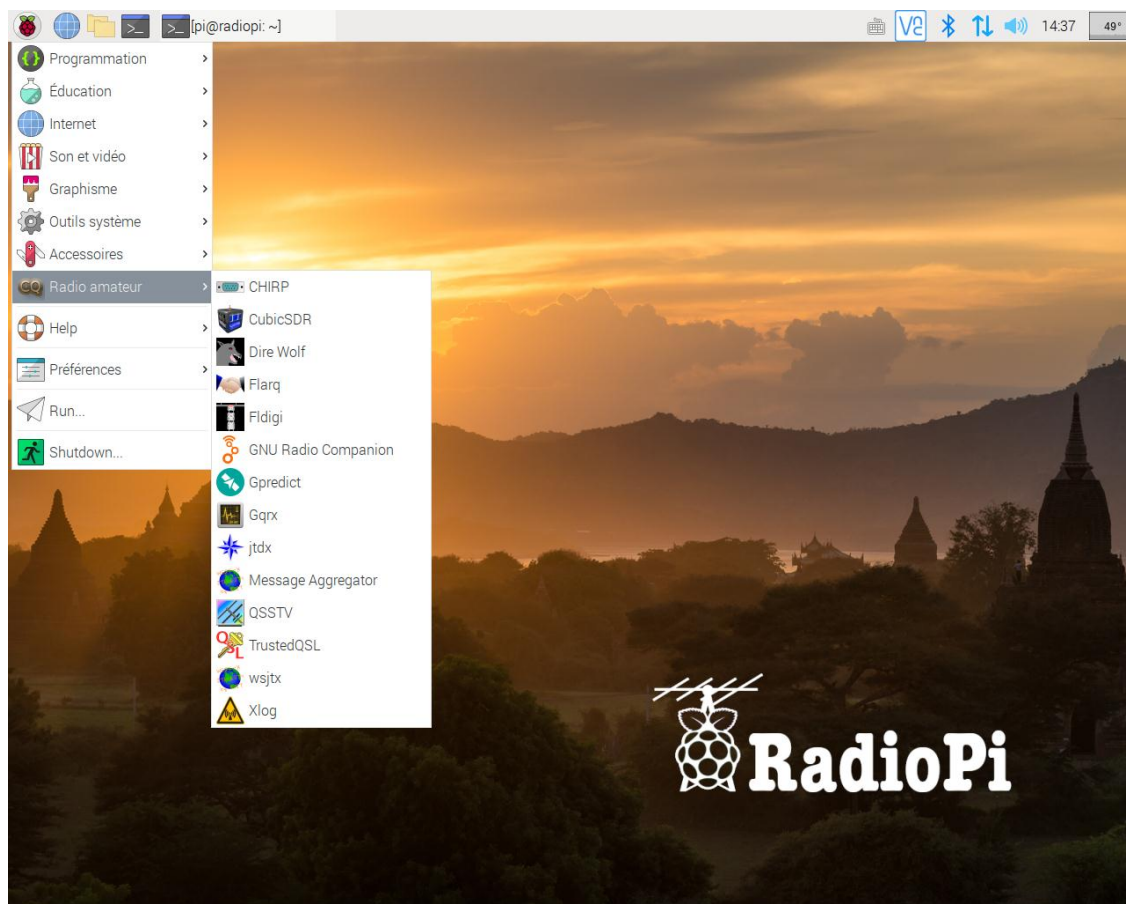
<https://github.com/dslotter/HamPi/issues>

<https://groups.io/g/RaspberryPi-4-HamRadio/>

8.4 Présentation de RadioPi

[RadioPi](#) est une distribution Raspbian créée par BG6LH, BG1TPT, BI1EIH contenant 15 logiciels préinstallés. L'avantage de cette image est de proposer un [wiki](#) de démarrage (mode d'emploi) ce qui n'est pas le cas pour HamPi. Par contre, cette image dispose de beaucoup moins de logiciels préinstallés que HamPi. En espérant que RadioPi bénéficie du même nombre de logiciels qu'HamPi à l'avenir.

- **WSJT-X**, 2.2.2, The extreme weak-signal communication software
- **JTDX**, 2.1.0-RC150, More features than WSJT-X
- **TQSL**, 2.4.3, The ARRL LoTW QSL client
- **Xlog**, 2.0.14, A light weight logging program
- **Fldigi**, 4.1.01, Amateur Radio in Digital Modes
- **CHIRP**, daily20190104, Programming amateur radios, supporting a large number of manufacturers and models, such as Yaesu, Icom, Kenwood, Baofeng, TYT and so on.
- **QSSTV**, 9.2.6, Receiving and transmitting SSTV and HAMDRM(DSSTV)
- **GNU Radio Companion**, 3.7.13.4, GNU Radio is a free & open-source software development toolkit that provides signal processing blocks to implement software radios.
- **GQRX**, 2.11.5, Gqrx is an open source software defined radio receiver (SDR) powered by the GNU Radio and the Qt graphical toolkit.
- **CubicSDR**, 0.2.5, Cross-platform SDR application
- **Direwolf**, 1.4, A software "soundcard" modem/TNC and APRS en/decoder
- **Gpredict**, 2.3-33-gca42d22-1, A real-time satellite tracking and orbit prediction application
- **VNC Server**, 6.7.2, To remote control your rig remotely by VNC
- **PulseAudio Preferences**, 1.1, For simultaneous line-output, and transmit monitoring.



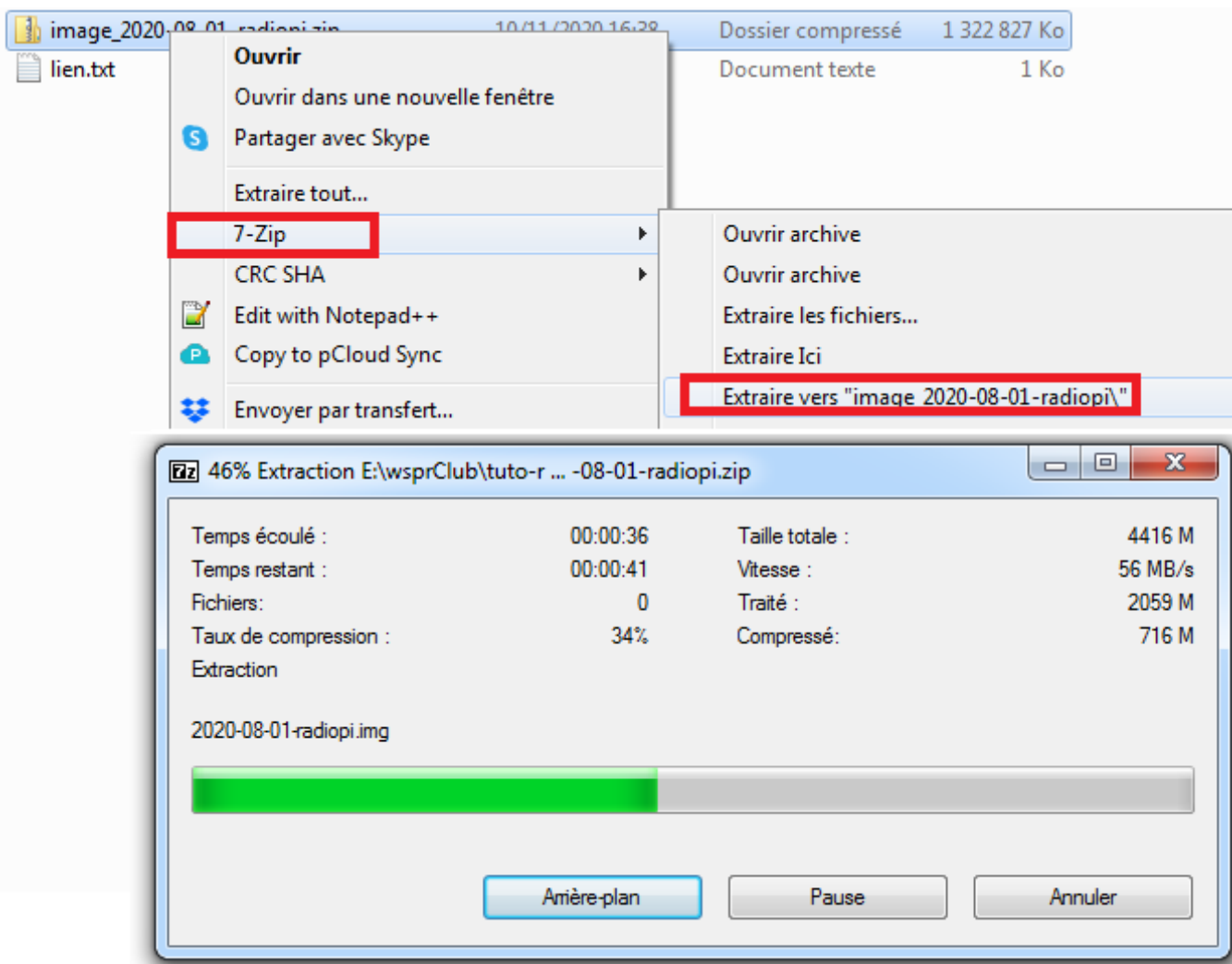
8.5 Installation de l'image disque radioPi

A partir du site <https://radiopi.club/> , télécharger la dernière version de l'image

Downloads

Image file: [image_2020-08-01-radiopi.zip](#)

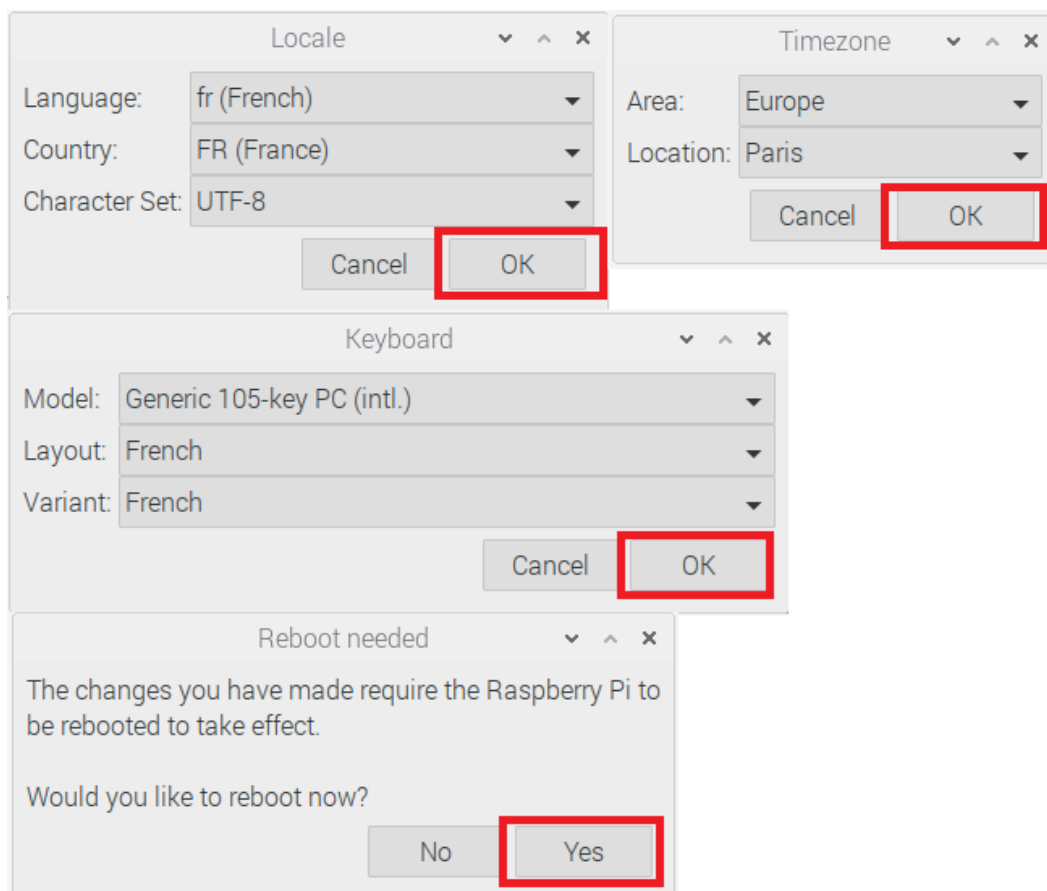
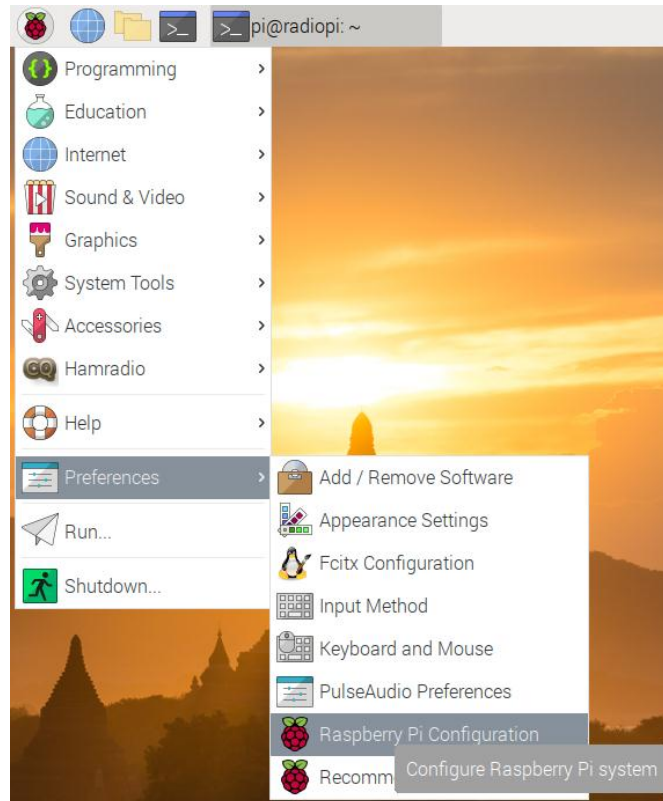
Puis décompresser l'archive avec l'utilitaire [7zip](#),



Pour copier l'image sur une carte mSD, utiliser la même procédure que pour HamPi (page 4 de ce même document) avec l'utilitaire Etcher.

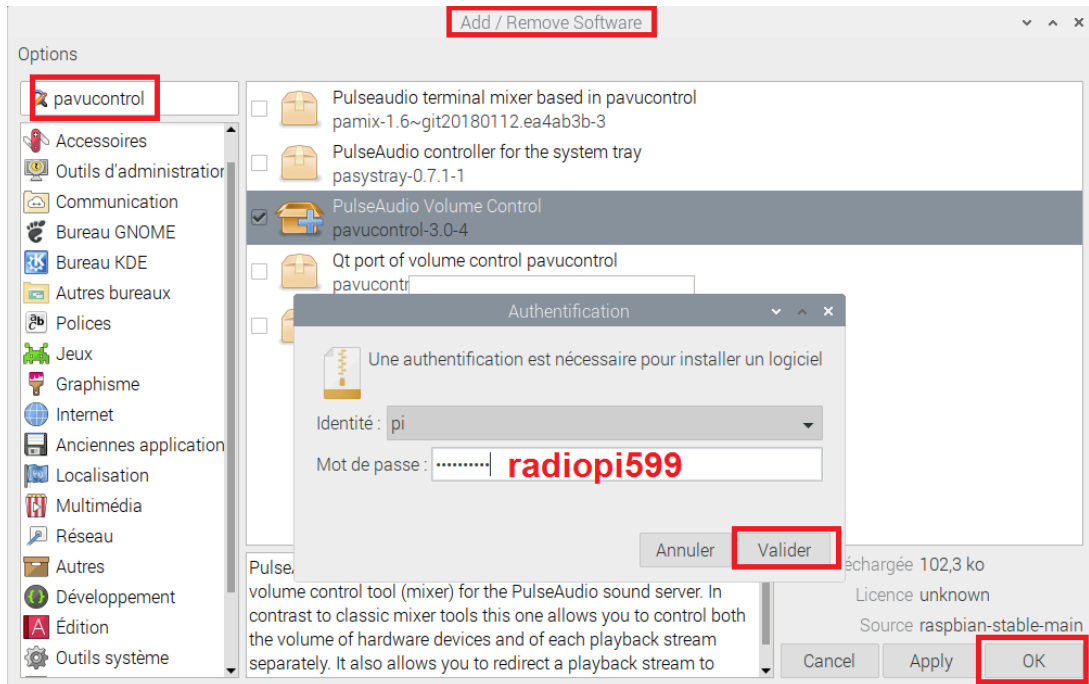
8.6 Prise en main de radioPi

Commencer par configurer le pays et le clavier en allant dans le menu « RaspberryPi configuration »

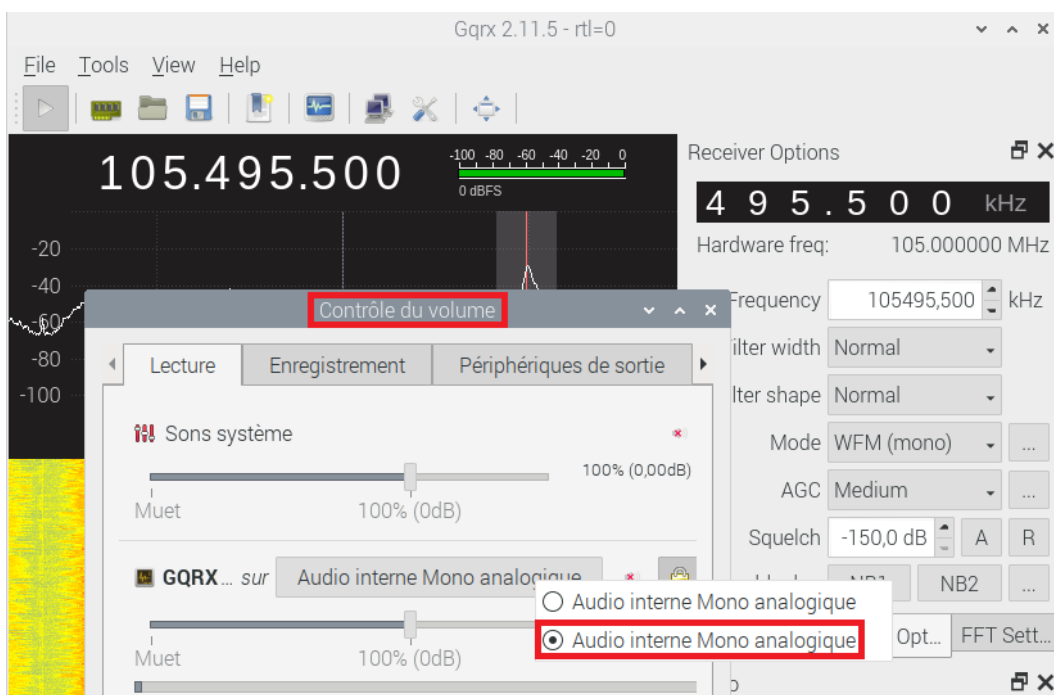


Effectuer les opérations suivantes afin de supprimer l'utilitaire fcitx qui perturbe l'utilisation du clavier en français.

```
sudo apt purge fcitx
sudo apt autoremove
sudo apt update
sudo reboot
```



GQRX avec la clé rtl-sdr fonctionne du 1^{er} coup.



HamPi 1.2 – The Raspberry Pi Ham Radio Image by W3DJS

General Ham Radio Applications

[HamLib](#) - Ham Radio Control Libraries

[grig](#) - graphical user interface to the Ham Radio Control Libraries

[CHIRP](#) - Radio Programming Software

[APRS Message App for JS8Call](#) - GUI to send APRS messages via JS8Call

[QTel](#) - EchoLink client

[QSSTV](#) - Slow Scan TV (e.g. "Fax")

[Gpredict](#) - Satellite prediction

[FreeDV](#) - Free digital voice vocoder

[BlueDV](#) - Client for D-Star and DMR

[WsprryPi - WSPR software](#)

[ADS-B Flight Tracking Software](#)

[Pi3/4 Stats Monitor](#) - by [W1HKJ](#)

[VOACAP](#) - HF propagation prediction

[GPS Support](#)

[wxtoimg](#) - NOAA weather imaging software

[twHamQTH](#) - an online callsign look up program

[twclock](#) - a world clock and automatic ID for amateur radio operators

[acfax](#) - Receive faxes using your radio and sound card

[colrconv](#) - converts client with sound and ncurses color support

[D-Rats 0.3.9 \(by new maintainer Maurizio Andreotti\)](#) - A communication tool for D-STAR

[fbb](#) - Packet radio mailbox and utilities

[gcb](#) - Utility to calculate long and short path to a location

[glfer](#) - Spectrogram display and QRSS keyer

[Xdx](#) is a DX-cluster client

[DXSpider](#) - DX Cluster Server

[fccexam](#) - Study tool for USA FCC commercial radio license exams.

[gnuais / gnuaisgui](#) - GNU Automatic Identification System receiver

[hamexam](#) - Study guide for USA FCC amateur radio (ham radio) license examinations.

[hamfax](#) - Qt based shortwave fax

[inspectrum](#) - tool for visualising captured radio signals

[predict-gsat](#) - Graphical Predict client

[splat](#) - analyze point-to-point terrestrial RF communication links

[wwl](#) - Calculates distance and azimuth between two Maidenhead locators

[AX.25](#) – Packet Radio drivers for ax.25 protocol

[linpac](#) - terminal for packet radio with mail client

[PyBOMBS](#) - GNU Radio install management system

[AMBEserver](#) – AMBE vocoder chip support

[HamClock](#) – GUI HamClock by WBOEW

[Adifmerg](#) – command-line ADIF conversion utility

[Lopora](#) – QRSS Beacon Reception

[Universal Ham Radio Remote](#) (UHRR) – UHRR provides remote radio operation

[RpiTx](#) -- Turns Raspberry Pi into low power transmitter

[ACARS Decoder](#) – for tracking aircraft transponders

[CygnusRFI](#) – RFI analysis tool for ground stations and radio telescopes)

[Update Scripts](#) -- to update Fldigi suite and WSJT-X (so one doesn't have to wait for new HamPi release.)

Antenna Ham Radio Applications

[antennavis](#) - Antenna Visualization Software

[gsmc](#) - A GTK Smith Chart Calculator for RF impedance matching

[nec2c](#) - Translation of the NEC2 FORTRAN source code to the C language

[xnecview](#) - NEC structure and gain pattern viewer

[yagiuda](#) - software to analyse performance of Yagi-Uda antennas

Digital Mode Ham Radio Applications

[WSJT-X](#) - Weak Signal (FT8, FT4, etc.) by [W1JT](#)

[GridTracker](#) - Graphical mapping companion program for WSJT-X or JTDX

[JTDX](#) - Alternate client for Weak Signal (FT8, FT4, etc.)

[JS8Call](#) - Messaging built on top of FT8 protocol by [KN4CRD](#)

[JS8CallTools](#) - Get Grid coordinates using GPS

(FLDigi is in its own section below.)

[gnss-sdr](#) - GLONASS satellite system Software Defined Receiver

[linpsk](#) - amateur radio PSK31/RTTY program via soundcard

[multimon](#) - multimon - program to decode radio transmissions

[multimon-ng](#) - digital radio transmission decoder

[psk31lx](#) - a terminal based ncurses program for psk31

[twpsk](#) - a psk program

Software Defined Radio

[CubicSDR](#) - Software Defined Radio receiver

[cutesdr](#) - Simple demodulation and spectrum display program

[GQRX](#) - Software defined radio receiver

[LeanSDR](#) – Lightweight, portable software defined radio

[SDRAngel](#) - SDR player

[lysdr](#) - Simple software-defined radio

[SoapyAudio](#) - Soapy SDR plugin for Audio devices

[SoapyHackRF](#) - SoapySDR HackRF module

[SoapyMultiSDR](#) - Multi-device support module for SoapySDR

[SoapyNetSDR](#) - Soapy SDR module for NetSDR protocol

[SoapyRemote](#) - Use any Soapy SDR remotely

[SoapyRTLSDR](#) - Soapy SDR module for RTL SDR USB dongle

[SoapySDR](#) - Vendor and platform neutral SDR support library

[SoapySDRPlay](#) - Soapy SDR module for SDRPlay

Support for [RTL-SDR](#)

Support for [SDRPlay SDR](#)

Support for [HackRF SDR](#)

Support for [AirSpy](#) and [AirSpy HF](#)

[SoapySDRAirSpy](#)- Soapy SDR module for AirSpy SDR

[SoapySDRFUNCube Dongle Pro+](#)- Soapy SDR module for FUNCube Dongle Pro+

[SoapySDRPlutoSDR](#)- Soapy SDR module for Pluto SDR

[SoapySDROsmoSDR](#)- Soapy SDR module for Osmo SDR

[SoapySDRRedPitaya](#)- Soapy SDR module for Red Pitaya SDR

[SoapyUHD](#)- Soapy SDR module for Ettus ResearchUHD SDR

[SoapySDRVOLKConverters](#) - Support for VOLK-based type converters

APRS Applications

[Xastir](#) - APRS GUI client / Digipeater / Igate

[YAAC](#) - Yet Another APRS Client

[DireWolf](#) - Software "soundcard" AX.25 packet modem/TNC and APRS encoder/decoder

[aprsdigi](#) - digipeater for APRS

[aprx](#) - APRS Digipeater and iGate

[soundmodem](#) - Sound Card Amateur Packet Radio Modems

FLDigi Application Suite from [W1HKJ](#)

[flrig](#) - Rig Control program which interfaces with fldigi

[fldigi](#) - [Digital Modes](#) Communications

[flaa](#) - RigExpert Antenna Analyzer Control Program

[flamp](#) - File transmissions via Amateur Multicast Protocol

[flarg](#) - ARQ data transfer utility for fldigi

[flcluster](#) - Telnet client to remote DX Cluster Servers

[fllog](#) - Logbook application which can use same data file as fldigi

[flmsg](#) - Editor for ICS 213 Forms

[flnet](#) - Net Control Assistant for Net Activities (Check-In Application)

[flpost](#) - NBEMs post office

[flwrap](#) - File encapsulation and compression for transmission over amateur radio

[flwkey](#) - Winkeyer (or clone) control program for K1EL Winkeyer series

Logging Applications

[TrustedQSL](#) - LotW client

[CQRlog](#) - Ham Radio Logging Application

[PyQSO](#) - Logging software (written in Python)

[klog](#) - The Ham Radio Logging program

[tlf](#) - console based ham radio contest logger

[tucnak2](#) - VHF/UHF/SHF Hamradio contest log version 2

[twlog](#) - basic logging program for ham radio

[upload_adif_log](#) – Upload only new log entries to LotW, eQSL.cc and ClubLog

[wsjtx to n3fjp](#) - Logging adapter to allow WSJT-X to log to N3FJP

[xlog](#) - GTK+ Logging program for Hamradio Operators

WinLink Applications

[Pat WinLink](#) - WinLink for Raspberry Pi (and other platforms)

[ARDOP](#) support for Pat WinLink

[ARDOP-GUI](#) - Provides graphical representation of ARDOP connections

[Find ARDOP](#) - Retrieves local ARDOP sources by [KM4ACK](#)

[Pat Menu 2](#) – Menu for Pat by KM4ACK

[PMON](#) - a PACTOR® Monitoring Utility for Linux

Morse Code Applications

[aldo](#) - Morse code training program

[cw](#) - sound characters as Morse code on the soundcard or console speaker

[cwcp](#) - Text based Morse tutor program

[xcwcp](#) - Graphical Morse tutor program

[cwdaemon](#) - morse daemon for the serial or parallel port

[ebook2cw](#) - convert ebooks to Morse MP3s/OGGs

[ebook2cwgui](#) - GUI for ebook2cw

[morse](#) - training program about morse-code for aspiring radio hams

[morse2ascii](#) - tool for decoding the morse codes from a PCM WAV file

[morsegen](#) - convert file to ASCII morse code

[grq](#) - High speed Morse telegraphy trainer

[xdemorse](#) - decode Morse signals to text