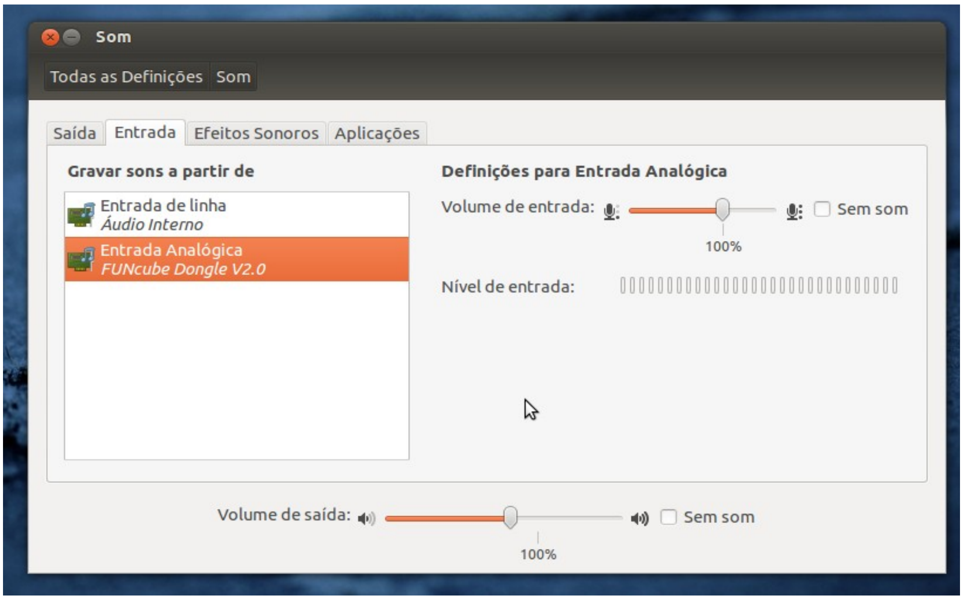
Getting Started with the FunCube Dongle Pro+ on Ubuntu 12.04 LTS (precise)

Step 1 – Making sure Ubuntu recognizes your FCD Pro+

Without installing any software just plug your FCD Pro+ into an available USB port. There should be no notifications or warnings of any kind.
Now go to Sound Settings (clicking on the volume control) and choose the Input tab, you should see FDC Pro+ detected like a device from which is possible to record.



Entrada de linha
Áudio Interno

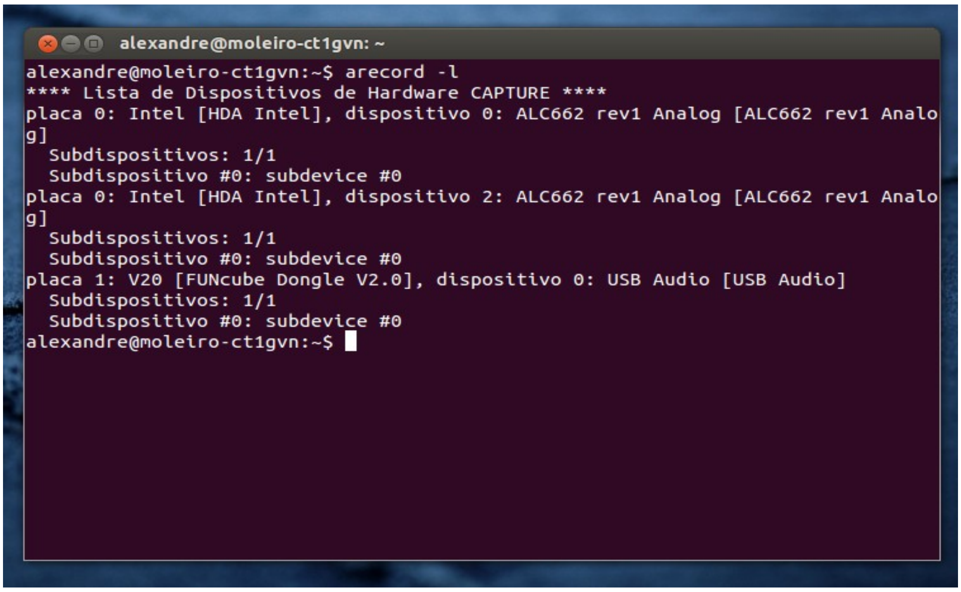
Entrada Analógica
FUNcube Dongle V2.0

Volume de entrada: 100% □ Sem som

Nível de entrada:

Volume de saída: 100% □ Sem som

Now open a Console window (Ctrl-Alt-T) and type **arecord -l**
You should see the FCD Pro+ listed.

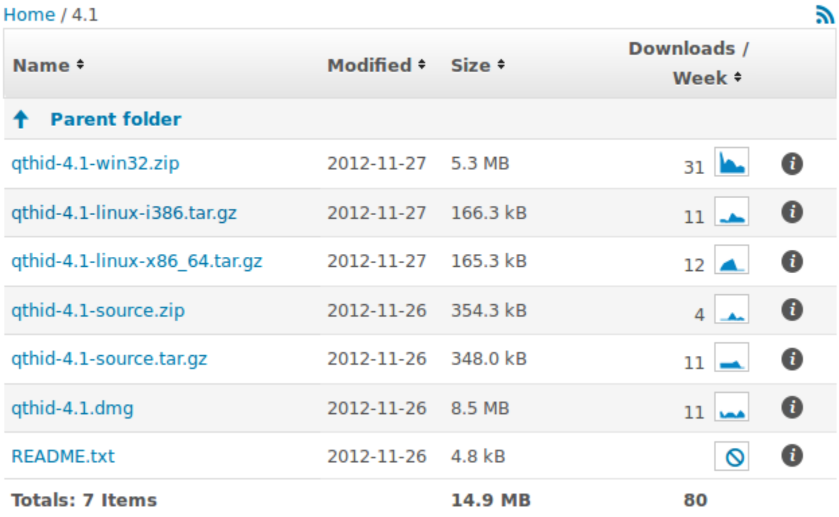


```
alexandre@moleiro-ct1gvn:~$ arecord -l
**** Lista de Dispositivos de Hardware CAPTURE ****
placa 0: Intel [HDA Intel], dispositivo 0: ALC662 rev1 Analog [ALC662 rev1 Analo
g]
  Subdispositivos: 1/1
  Subdispositivo #0: subdevice #0
placa 0: Intel [HDA Intel], dispositivo 2: ALC662 rev1 Analog [ALC662 rev1 Analo
g]
  Subdispositivos: 1/1
  Subdispositivo #0: subdevice #0
placa 1: V20 [FUNcube Dongle V2.0], dispositivo 0: USB Audio [USB Audio]
  Subdispositivos: 1/1
  Subdispositivo #0: subdevice #0
alexandre@moleiro-ct1gvn:~$ 
```

If all is OK proceed to Step 2.

Step 2 – Installing and testing qthid 4.1

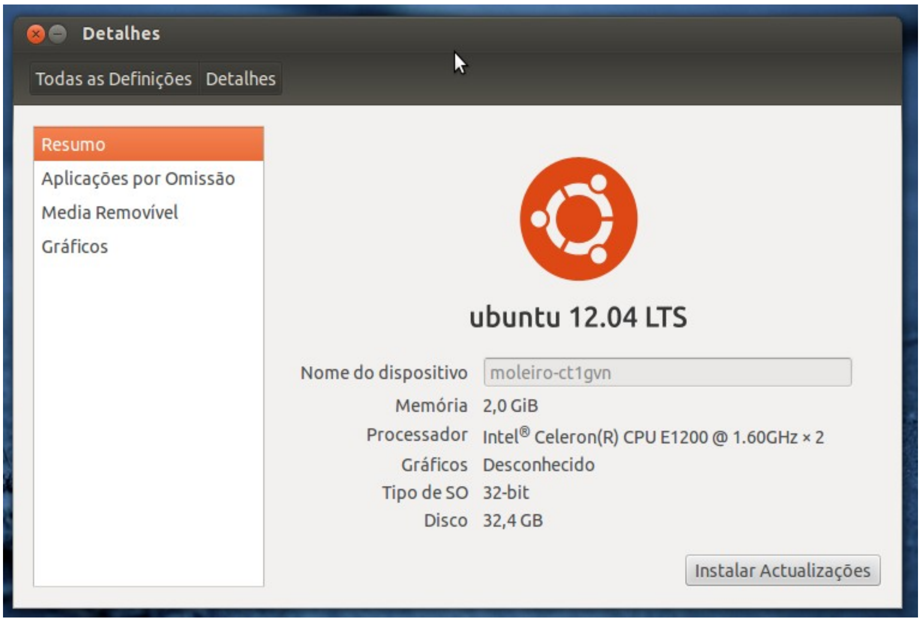
Point your browser to http://sourceforge.net/projects/qthid/files/4.1/ . If you are running the 32bit version of Ubuntu download qthid-4.1-linux-i386.tar.gz . If you are running the 64bit flavour choose qthid-4.1-linux-x86_64.tar.gz .



Home / 4.1

Name	Modified	Size	Downloads / Week
Parent folder			
qthid-4.1-win32.zip	2012-11-27	5.3 MB	31
qthid-4.1-linux-i386.tar.gz	2012-11-27	166.3 kB	11
qthid-4.1-linux-x86_64.tar.gz	2012-11-27	165.3 kB	12
qthid-4.1-source.zip	2012-11-26	354.3 kB	4
qthid-4.1-source.tar.gz	2012-11-26	348.0 kB	11
qthid-4.1.dmg	2012-11-26	8.5 MB	11
README.txt	2012-11-26	4.8 kB	
Totals: 7 Items		14.9 MB	80

If you are unsure which version you are running go to All Settings / Details.



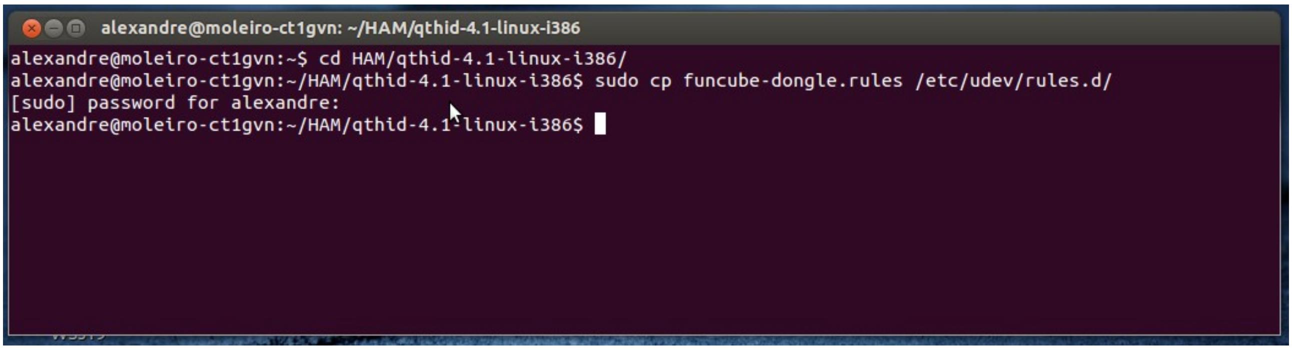
ubuntu 12.04 LTS

Nome do dispositivo moleiro-ct1gvn
Memória 2,0 GiB
Processador Intel® Celeron(R) CPU E1200 @ 1.60GHz × 2
Gráficos Desconhecido
Tipo de SO 32-bit
Disco 32,4 GB

After downloading double-click on the .gz file. Drag and drop the contained folder into anywhere in your personal folders.

Now open a Console window (Ctrl-Alt-T) and change into the folder you just created with the **cd** command. (In my case I have put it inside the HAM folder.)

After changing the current directory type **sudo cp funcube-dongle.rules /etc/udev/rules.d/** This command is necessary for using qthid and access the FCD Pro+ as a regular user (not admin).



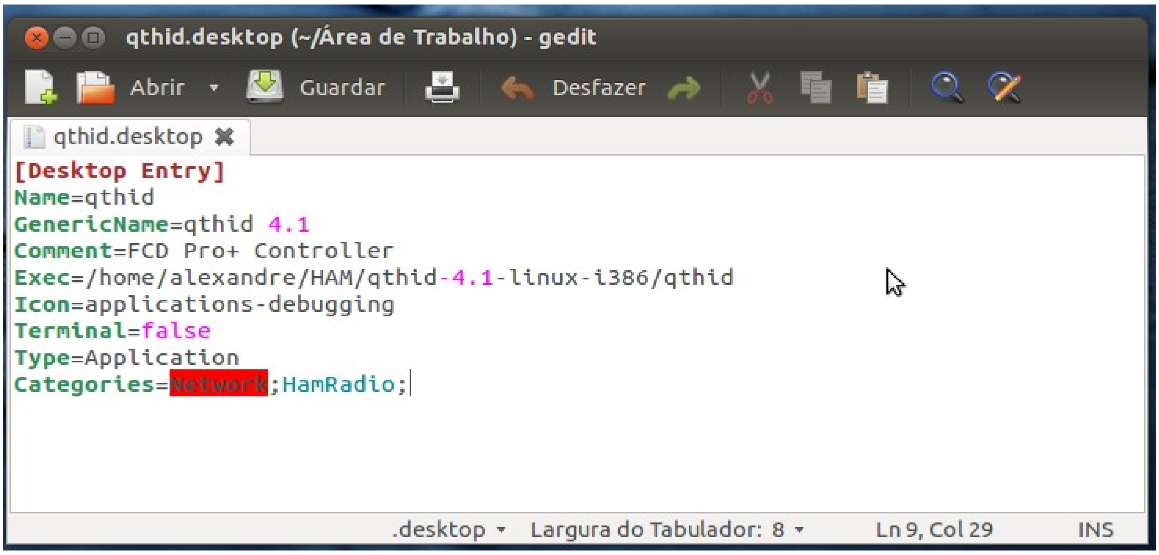
```
alexandre@moleiro-ct1gvn:~$ cd HAM/qthid-4.1-linux-i386/
alexandre@moleiro-ct1gvn:~/HAM/qthid-4.1-linux-i386$ sudo cp funcube-dongle.rules /etc/udev/rules.d/
[sudo] password for alexandre:
alexandre@moleiro-ct1gvn:~/HAM/qthid-4.1-linux-i386$ 
```

If you get a message saying that you are not on the sudoers list you should follow the instructions on this page: http://www.pendrivelinux.com/how-to-add-a-user-to-the-sudoers-list/

Now open a text editor (**gedit** for example) and insert the following text, making sure you change the folder in the Exec line to reflect your installation path:

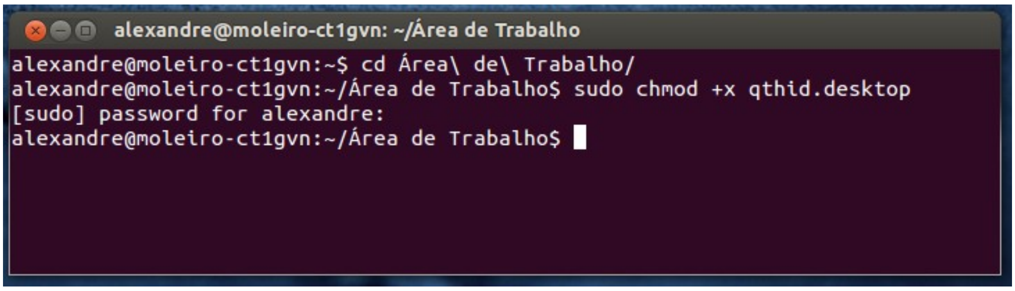
[Desktop Entry]
Name=qthid
GenericName=qthid 4.1
Comment=FCD Pro+ Controller
Exec=/home/alexandre/HAM/qthid-4.1-linux-i386/qthid
Icon=applications-debugging
Terminal=false
Type=Application
Categories=Network;HamRadio;

Save the file in your Desktop folder using the filename **qthid.desktop**



```
[Desktop Entry]
Name=qthid
GenericName=qthid 4.1
Comment=FCD Pro+ Controller
Exec=/home/alexandre/HAM/qthid-4.1-linux-i386/qthid
Icon=applications-debugging
Terminal=false
Type=Application
Categories=Network;HamRadio;
```

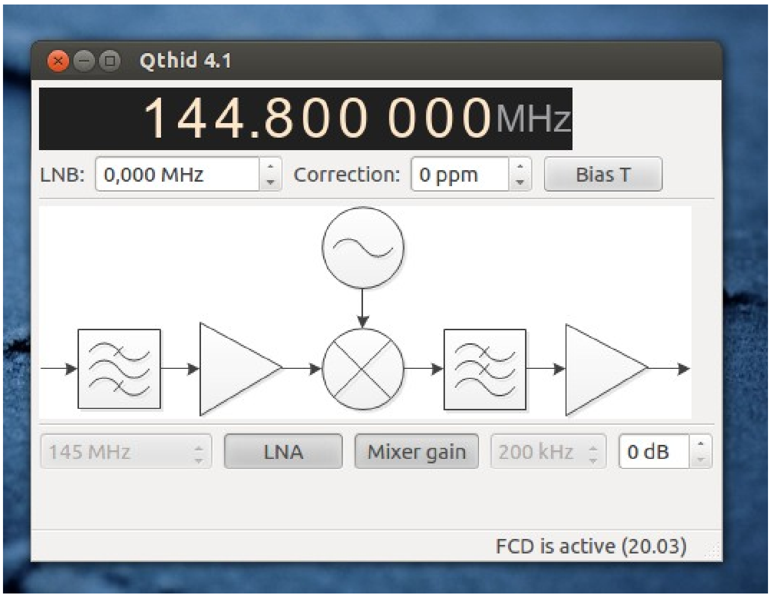
In the Console window change to your desktop folder with **cd ~/Desktop** and then type the command **sudo chmod +x qthid.desktop**



```
alexandre@moleiro-ct1gvn: ~/Área de Trabalho
alexandre@moleiro-ct1gvn:~$ cd Área\ de\ Trabalho/
alexandre@moleiro-ct1gvn:~/Área de Trabalho$ sudo chmod +x qthid.desktop
[sudo] password for alexandre:
alexandre@moleiro-ct1gvn:~/Área de Trabalho$
```

You have just created a desktop shortcut to qthid!

Now just double click on the shortcut you just created, it should open the qthid 4.1 user interface.



Qthid 4.1

144.800 000MHz

LNB: 0,000 MHz Correction: 0 ppm

145 MHz 200 kHz 0 dB

FCD is active (20.03)

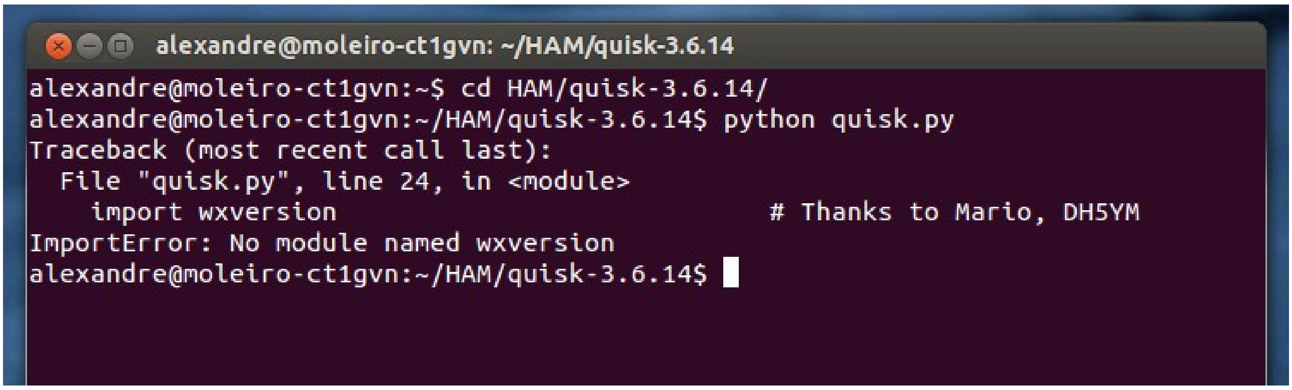
If the status bar reads “FCD is active” Congratulations, you have a working FCD Pro+
If the status bar reads “No FCD detected” try closing qthid 4.1, removing and inserting the FCD Pro+ and opening qthid 4.1 again. If that fails try repeating Step 2 from the beginning again.

Step 3 – Installing and configuring quisk

NOTE: this step only works in 32bit Ubuntu, for the 64bit version you need to recompile quisk or skip to the next Step

Go to http://james.ahlstrom.name/quisk/ and download the quisk-3.6.14.tar.gz file. After downloading double-click on it and drag and drop the contained folder into anywhere in your home folder.

Open a Console window and change into the just created quisk folder using the **cd** command. Then type **python quisk.py**



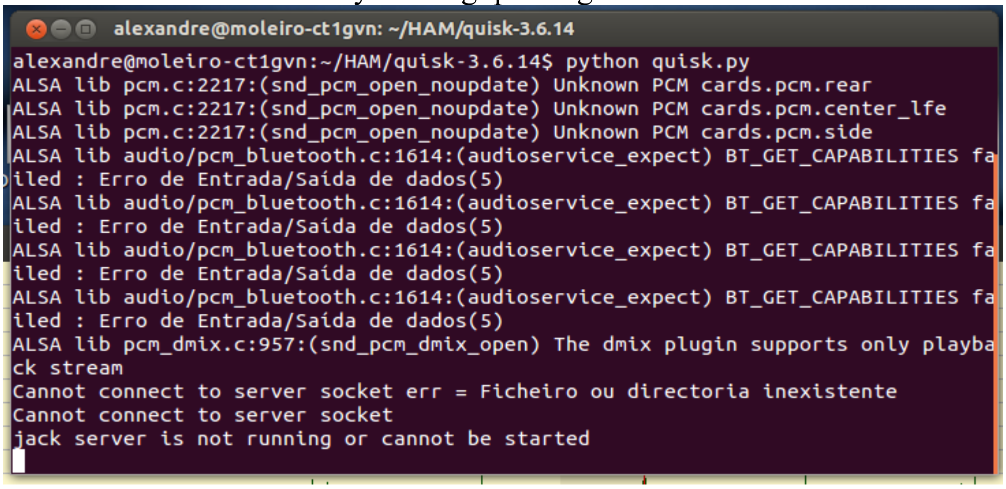
```
alexandre@moleiro-ct1gvn:~$ cd HAM/quisk-3.6.14/
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ python quisk.py
Traceback (most recent call last):
  File "quisk.py", line 24, in <module>
    import wxversion                        # Thanks to Mario, DH5YM
ImportError: No module named wxversion
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ 
```

You will probably get an error message, like I did.

Before proceeding you need to open Ubuntu Software Center and install the following packages if they are not already installed on your system (just type the names on the search box and they should appear):

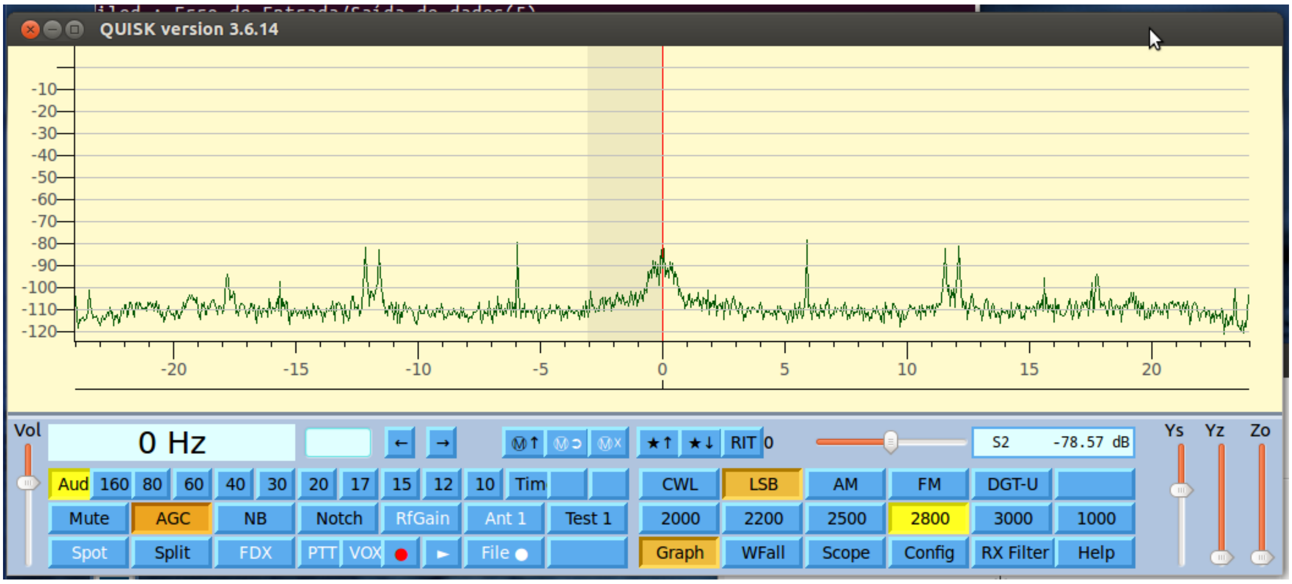
- python2.7
- python-wxgtk2.8
- ruby-fftw3

Go back to the Console window and try running quisk again.



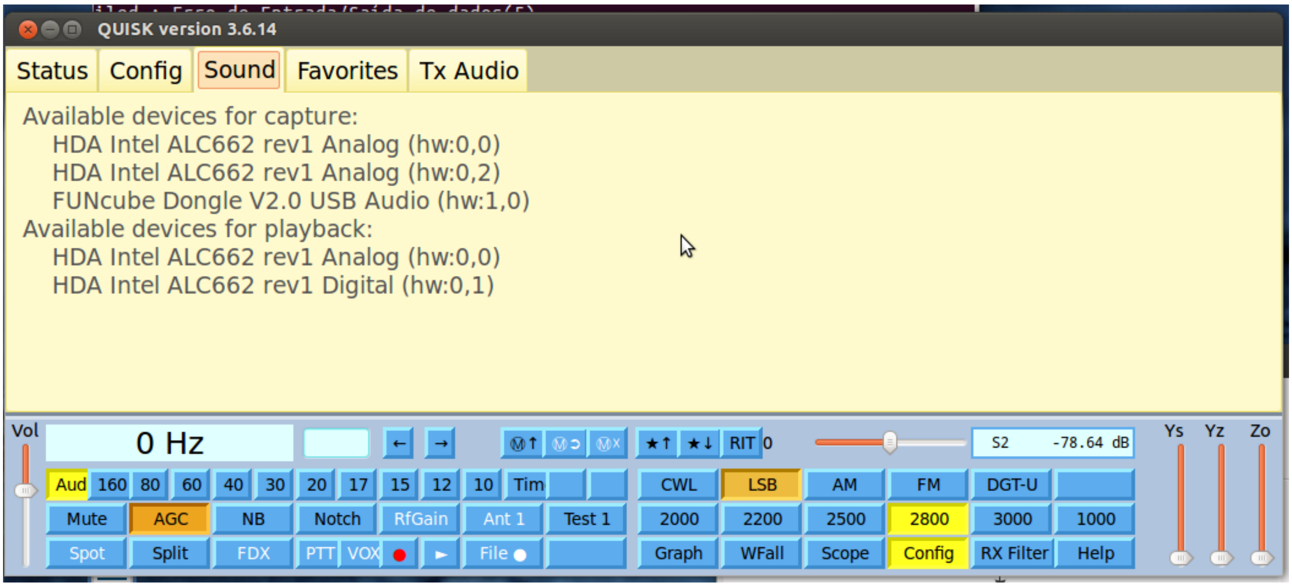
```
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ python quisk.py
ALSA lib pcm.c:2217:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.rear
ALSA lib pcm.c:2217:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.center_lfe
ALSA lib pcm.c:2217:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.side
ALSA lib audio/pcm_bluetooth.c:1614:(audioservice_expect) BT_GET_CAPABILITIES fa
iled : Erro de Entrada/Saída de dados(5)
ALSA lib audio/pcm_bluetooth.c:1614:(audioservice_expect) BT_GET_CAPABILITIES fa
iled : Erro de Entrada/Saída de dados(5)
ALSA lib audio/pcm_bluetooth.c:1614:(audioservice_expect) BT_GET_CAPABILITIES fa
iled : Erro de Entrada/Saída de dados(5)
ALSA lib audio/pcm_bluetooth.c:1614:(audioservice_expect) BT_GET_CAPABILITIES fa
iled : Erro de Entrada/Saída de dados(5)
ALSA lib pcm_dmix.c:957:(snd_pcm_dmix_open) The dmix plugin supports only playba
ck stream
Cannot connect to server socket err = Ficheiro ou directoria inexistente
Cannot connect to server socket
jack server is not running or cannot be started
```

After a few seconds you should get some errors on the console but also a nice graphical window with quisk running.



Also you should be hearing some audio from your regular sound card, make sure you can adjust that audio level with the **Vol** control on the left side of quisk.

Now click the **Config** button on quisk, and the choose the **Sound** tab.


QUISK version 3.6.14
Status Config Sound Favorites Tx Audio

Available devices for capture:
    HDA Intel ALC662 rev1 Analog (hw:0,0)
    HDA Intel ALC662 rev1 Analog (hw:0,2)
    FUNcube Dongle V2.0 USB Audio (hw:1,0)
Available devices for playback:
    HDA Intel ALC662 rev1 Analog (hw:0,0)
    HDA Intel ALC662 rev1 Digital (hw:0,1)

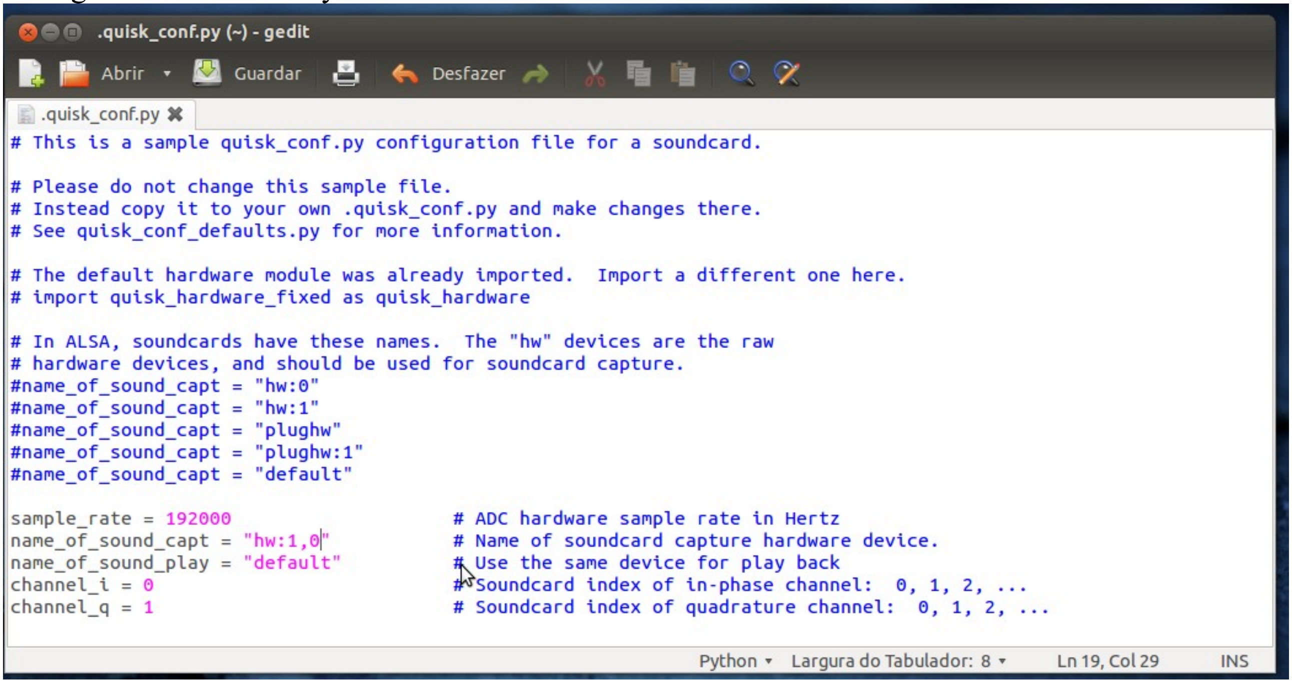
Vol
0 Hz
RIT 0
S2 -78.64 dB
Ys Yz Zo
Aud 160 80 60 40 30 20 17 15 12 10 Tim
Mute AGC NB Notch RfGain Ant 1 Test 1
Spot Split FDX PTT VOX File
CWL LSB AM FM DGT-U
2000 2200 2500 2800 3000 1000
Graph WFall Scope Config RX Filter Help

Write down the hw code you see for the FCD Pro+, in my case hw:1,0

Now close quisk window, you should get the prompt back in the Console window.

Using a text editor (**gedit** for example) open the file **quisk_conf_model.py** that is located on the folder where you installed quisk.

Change the values so they look like these:



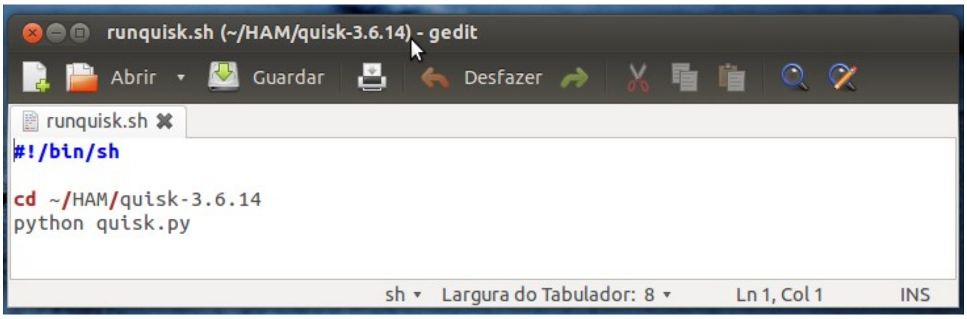
```
# This is a sample quisk_conf.py configuration file for a soundcard.

# Please do not change this sample file.
# Instead copy it to your own .quisk_conf.py and make changes there.
# See quisk_conf_defaults.py for more information.

# The default hardware module was already imported.  Import a different one here.
# import quisk_hardware_fixed as quisk_hardware

# In ALSA, soundcards have these names.  The "hw" devices are the raw
# hardware devices, and should be used for soundcard capture.
#name_of_sound_capt = "hw:0"
#name_of_sound_capt = "hw:1"
#name_of_sound_capt = "plughw"
#name_of_sound_capt = "plughw:1"
#name_of_sound_capt = "default"

sample_rate = 192000                    # ADC hardware sample rate in Hertz
name_of_sound_capt = "hw:1,0"           # Name of soundcard capture hardware device.
name_of_sound_play = "default"          # Use the same device for play back
channel_i = 0                           # Soundcard index of in-phase channel:  0, 1, 2, ...
channel_q = 1                           # Soundcard index of quadrature channel:  0, 1, 2, ...
```

Replace the “hw:1,0” with whatever code you wrote down earlier.

Don't save over the original file, use SAVE AS and save the file in your home folder with the name: **.quisk_conf.py**
Only if you save it in your home folder using exactly this filename will the configuration become active.
You can close the text editor.
Now go back to the Console window and start quisk again by typing **python quisk.py**
After a few seconds quisk should start and you should be hearing some noise, this time from the FCD Pro+ !

Next we need to create a script to launch quisk without having to type **python quisk.py** everytime we need to launch the program.

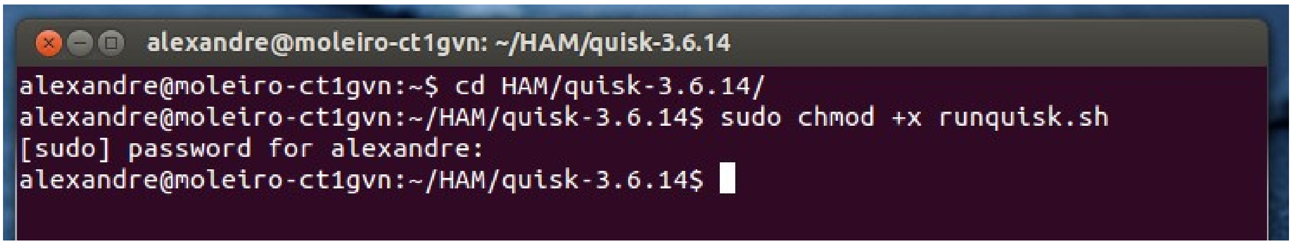
Using the text editor enter the following commands making sure you replace the folder to reflect your own installation.



```
#!/bin/sh

cd ~/HAM/quisk-3.6.14
python quisk.py
```

Then save the file as runquisk.sh on the folder where you installed it.

Open a Console window and change into your quisk folder using the command **cd**
Then issue the following command **sudo chmod +x runquisk.sh**

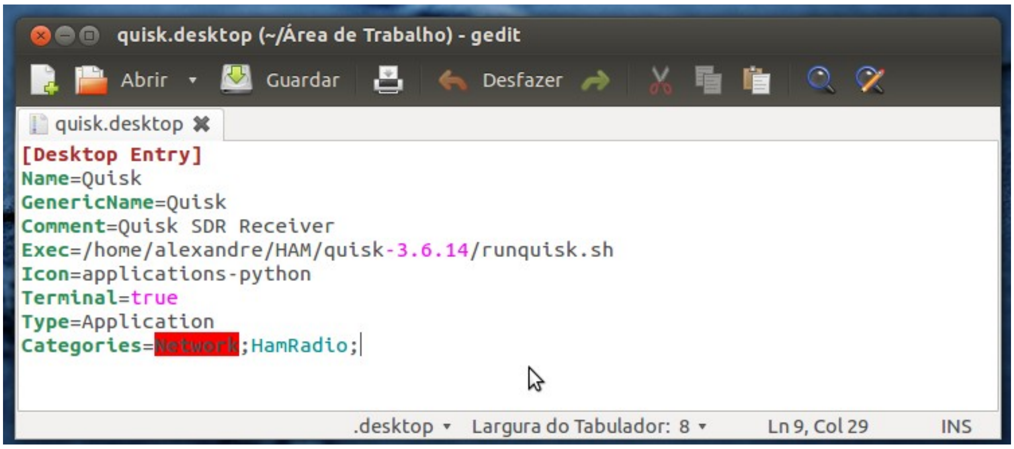


```
alexandre@moleiro-ct1gvn:~$ cd HAM/quisk-3.6.14/
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ sudo chmod +x runquisk.sh
[sudo] password for alexandre:
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ 
```

This will make the runquisk.sh file executable.

Next we need a desktop shortcut, so open the text editor and type (make sure you change the Exec line to reflect your instalation path):

[Desktop Entry]
Name=Quisk
GenericName=Quisk
Comment=Quisk SDR Receiver
Exec=/home/alexandre/HAM/quisk-3.6.14/runquisk.sh
Icon=applications-python
Terminal=true
Type=Application
Categories=Network;HamRadio;

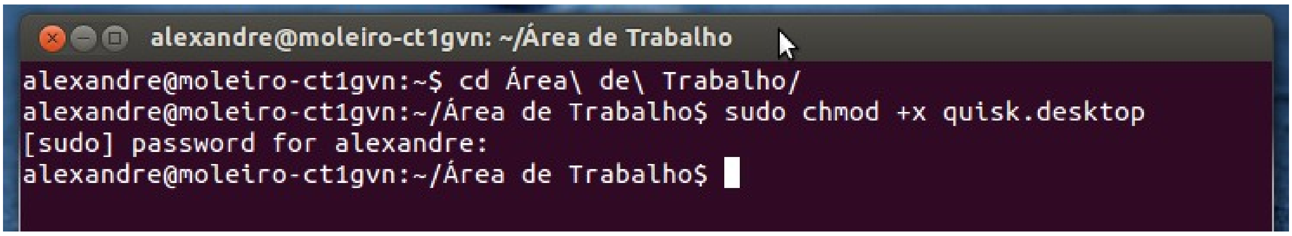


```
[Desktop Entry]
Name=Quisk
GenericName=Quisk
Comment=Quisk SDR Receiver
Exec=/home/alexandre/HAM/quisk-3.6.14/runquisk.sh
Icon=applications-python
Terminal=true
Type=Application
Categories=Network;HamRadio;
```

Note: you can change **Terminal=true** to **Terminal=false** if you don't want a console window to open with quisk

Save the file in your Desktop folder under the name **quisk.desktop**

Back in the console, change to your Desktop folder and issue the command:
sudo chmod +x quisk.desktop



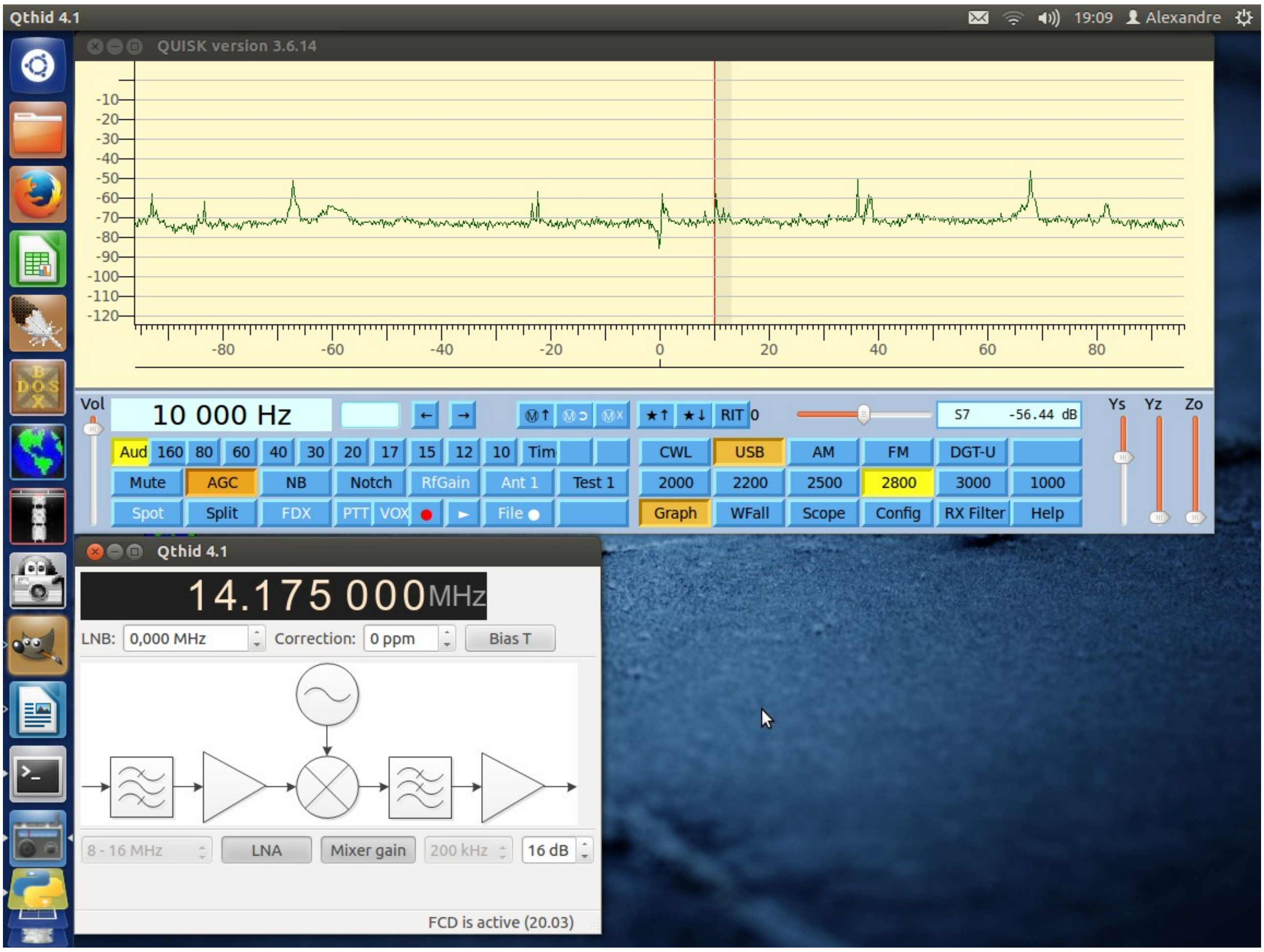
```
alexandre@moleiro-ct1gvn:~$ cd Área\ de\ Trabalho/
alexandre@moleiro-ct1gvn:~/Área de Trabalho$ sudo chmod +x quisk.desktop
[sudo] password for alexandre:
alexandre@moleiro-ct1gvn:~/Área de Trabalho$ 
```

You should now have a desktop icon for quisk.

Now launch qthid and then quisk. You can now control the center frequency with qthid and use quisk to demodulate signals from the passband.

Note that quisk doesn't directly control de center frequency, so even if you click the buttons to change band the center frequency will be allways the one set by qthid.

In the example below I was listening to 14.185 MHz USB, the center frequency of 14.175MHz is set in qthid plus 10kHz set in quisk.



QUISK version 3.6.14

10 000 Hz

S7 -56.44 dB

Qthid 4.1

14.175 000MHz

LNB: 0,000 MHz

Correction: 0 ppm

8 - 16 MHz

200 kHz

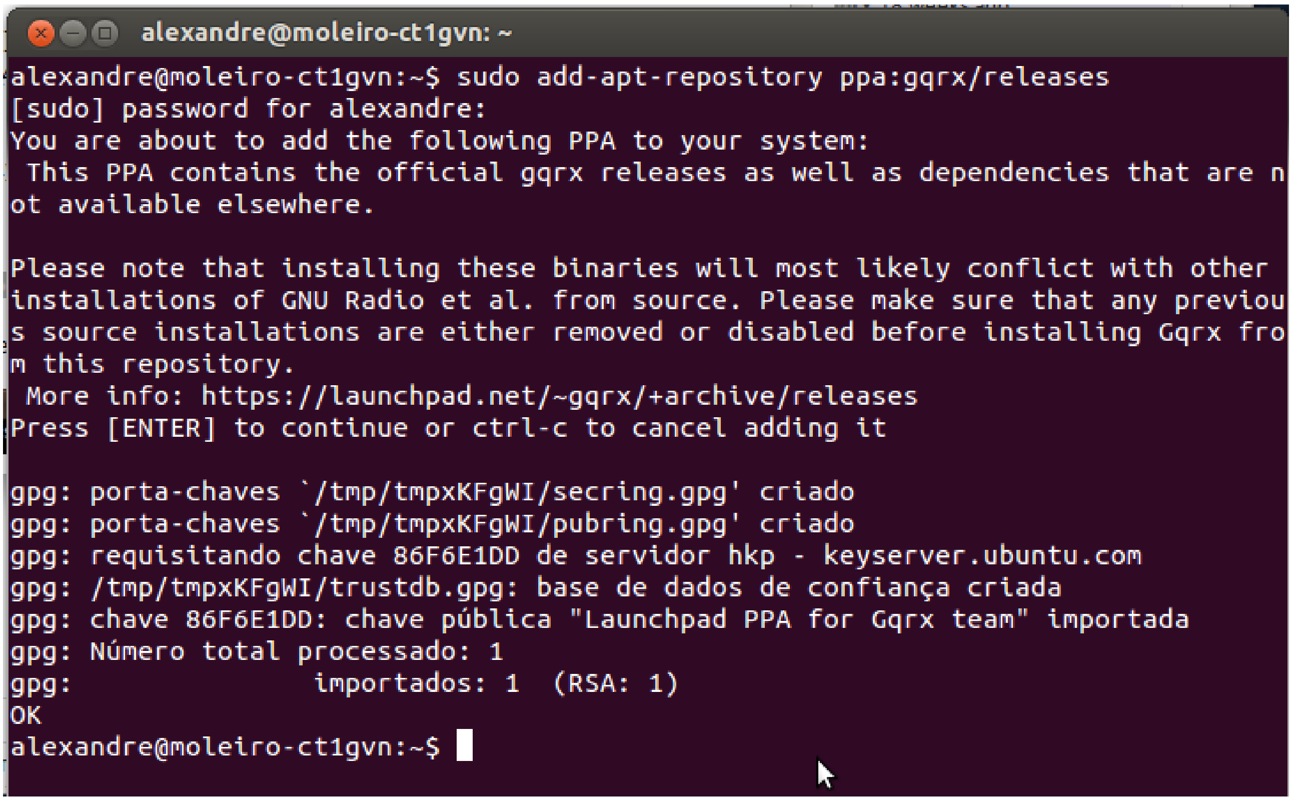
16 dB

FCD is active (20.03)

This completes the basic configuration of quisk.

Step 4 – Installing and configuring gqrx

First you need to add the Personal Package Archive of **gqrx** so that Ubuntu can find the required files. You need to open a Console window and type
sudo add-apt-repository ppa:gqrx/releases

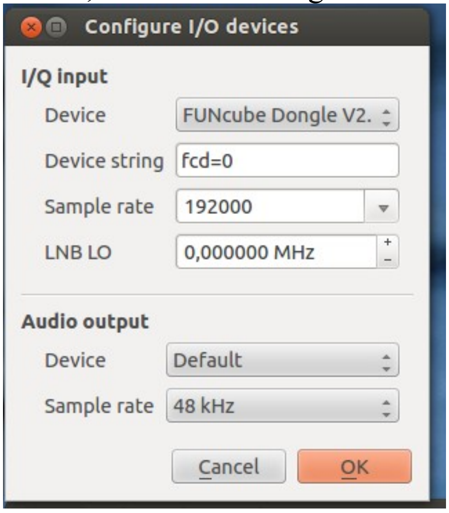


```
alexandre@moleiro-ct1gvn:~$ sudo add-apt-repository ppa:gqrx/releases
[sudo] password for alexandre:
You are about to add the following PPA to your system:
 This PPA contains the official gqrx releases as well as dependencies that are n
ot available elsewhere.

Please note that installing these binaries will most likely conflict with other
installations of GNU Radio et al. from source. Please make sure that any previou
s source installations are either removed or disabled before installing Gqrx fro
m this repository.
 More info: https://launchpad.net/~gqrx/+archive/releases
Press [ENTER] to continue or ctrl-c to cancel adding it

gpg: porta-chaves `/tmp/tmpxKFgWI/secring.gpg' criado
gpg: porta-chaves `/tmp/tmpxKFgWI/pubring.gpg' criado
gpg: requisitando chave 86F6E1DD de servidor hkp - keyserver.ubuntu.com
gpg: /tmp/tmpxKFgWI/trustdb.gpg: base de dados de confiança criada
gpg: chave 86F6E1DD: chave pública "Launchpad PPA for Gqrx team" importada
gpg: Número total processado: 1
gpg:               importados: 1  (RSA: 1)
OK
alexandre@moleiro-ct1gvn:~$ 
```

Again, on the Console, you need to issue the command **sudo apt-get update**
This will update the list of available packages, it may take some time, wait till it finishes.

Now go to Ubuntu Software Center and search for and install the **gqrx** package.

After it finishes installing open a Console window and type **gqrx**
You should get a configuration window, use the following values:



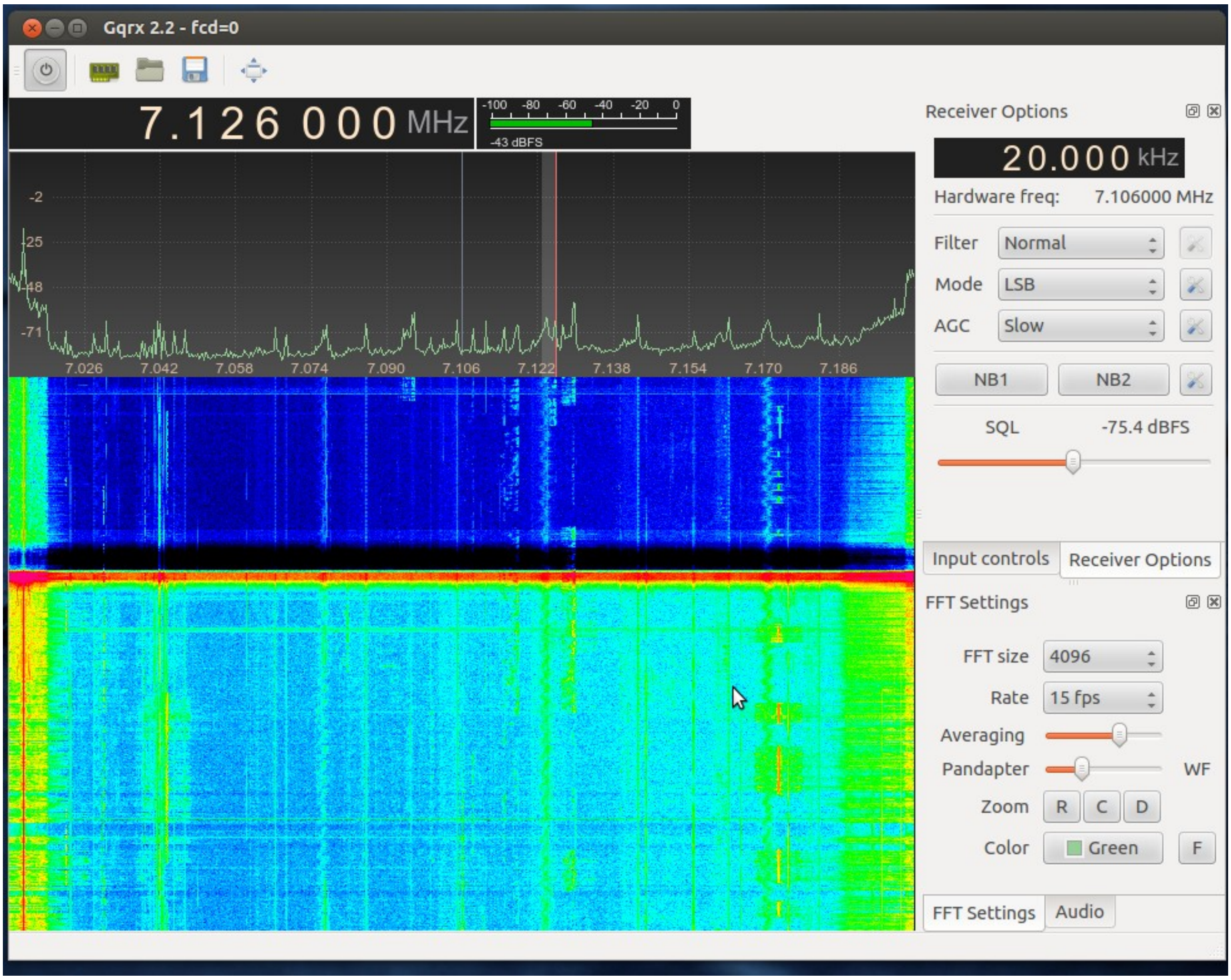
I/Q input

Device: FUNcube Dongle V2.
Device string: fcd=0
Sample rate: 192000
LNB LO: 0,000000 MHz

Audio output

Device: Default
Sample rate: 48 kHz

After you press ok you get the gqrx main window.



Gqrx 2.2 - fcd=0

7.126 000 MHz

-43 dBFS

Receiver Options

20.000 kHz

Hardware freq: 7.106000 MHz

Filter Normal

Mode LSB

AGC Slow

SQL -75.4 dBFS

FFT Settings

FFT size 4096

Rate 15 fps

Color Green

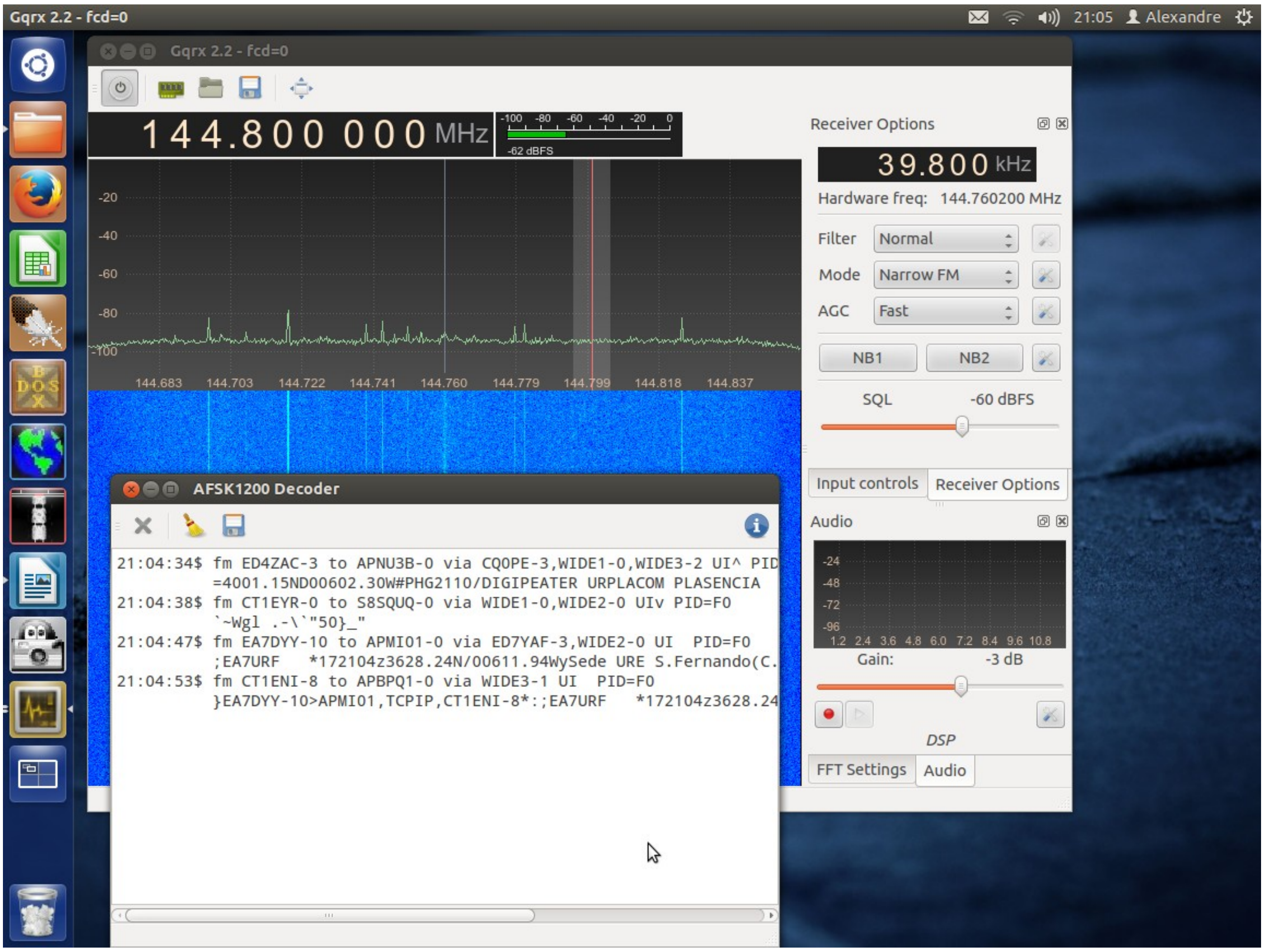
Last step is to create a desktop shortcut to gqrx.

Using the file manager navigate to the folder **/usr/bin** and find gqrx
Then, drag and drop while holding Ctrl and Shift the gqrx icon to your desktop.

With gqrx you can directly control the center frequency of the FCD Pro+, so you don't need to use qthid.

You should also try some of the settings in the Input controls section, selecting DC Cancel helps reduce the spike at the center frequency.

Also, with gqrx you can directly decode Packet radio signals at 1200 bps, just tune into an APRS frequency and open the AFSK1200 decoder from the Data menu.



Hardware freq: 144.760200 MHz

Filter: Normal

Mode: Narrow FM

AGC: Fast

SQL: -60 dBFS

Gain: -3 dB

```
21:04:34$ fm ED4ZAC-3 to APNU3B-0 via CQ0PE-3,WIDE1-0,WIDE3-2 UI^ PID
          =4001.15ND00602.30W#PHG2110/DIGIPEATER URPLACOM PLASENCIA
21:04:38$ fm CT1EYR-0 to S8SQUQ-0 via WIDE1-0,WIDE2-0 UIv PID=F0
          `~Wgl .-\`"50}_"
21:04:47$ fm EA7DYY-10 to APMI01-0 via ED7YAF-3,WIDE2-0 UI  PID=F0
          ;EA7URF   *172104z3628.24N/00611.94WySede URE S.Fernando(C.
21:04:53$ fm CT1ENI-8 to APBPQ1-0 via WIDE3-1 UI  PID=F0
          }EA7DYY-10>APMI01,TCPIP,CT1ENI-8*:;EA7URF   *172104z3628.24
```

Step 5 – Receiving digital modes by feeding decoded audio to applications

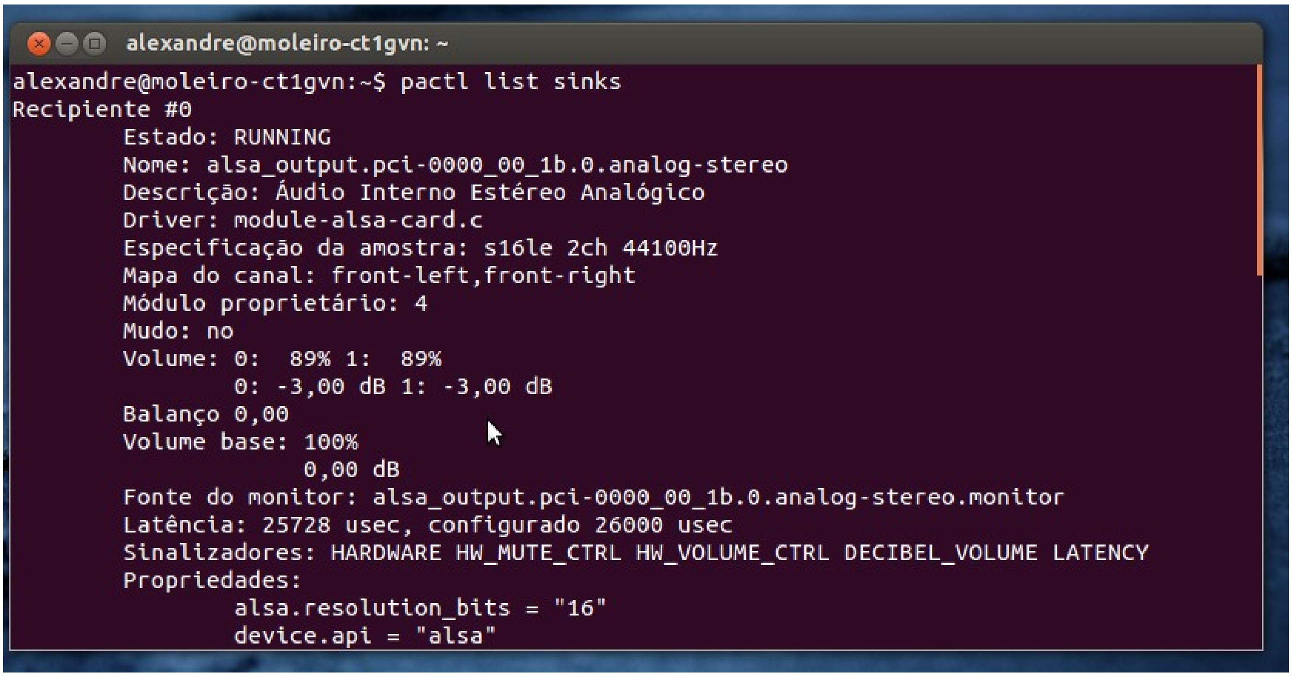
If you completed Steps 2, 3 and 4 you now have two fully functional receivers with AM, FM and SSB capabilities.

For receiving digital modes you need dedicated programs.

You also need to feed the audio output of quisk or gqrx into a digital mode dedicated program.

You will need the **pavucontrol** program, so first go to Ubuntu Software Center and install it.

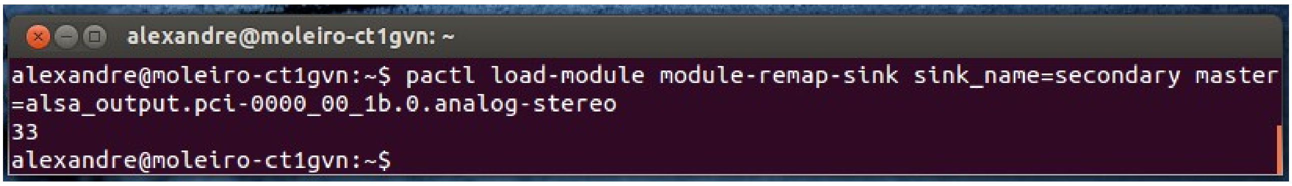
Then, open a Console window and type: **pactl list sinks**
Look for something like **alsa_output.pci-0000_00_1b.0.analog-stereo** in the output of the command and write it down.



```
alexandre@moleiro-ct1gvn:~$ pactl list sinks
Recipiente #0
        Estado: RUNNING
        Nome: alsa_output.pci-0000_00_1b.0.analog-stereo
        Descrição: Áudio Interno Estéreo Analógico
        Driver: module-alsa-card.c
        Especificação da amostra: s16le 2ch 44100Hz
        Mapa do canal: front-left,front-right
        Módulo proprietário: 4
        Mudo: no
        Volume: 0:  89% 1:  89%
                0: -3,00 dB 1: -3,00 dB
        Balanço 0,00
        Volume base: 100%
                     0,00 dB
        Fonte do monitor: alsa_output.pci-0000_00_1b.0.analog-stereo.monitor
        Latência: 25728 usec, configurado 26000 usec
        Sinalizadores: HARDWARE HW_MUTE_CTRL HW_VOLUME_CTRL DECIBEL_VOLUME LATENCY
        Propriedades:
                alsa.resolution_bits = "16"
                device.api = "alsa"
```

Now issue the command (all in one line):
pactl load-module module-remap-sink sink_name=secondary master=alsa_output.pci-0000_00_1b.0.analog-stereo
Making sure you replace the code after master= with the one you've wrote down.

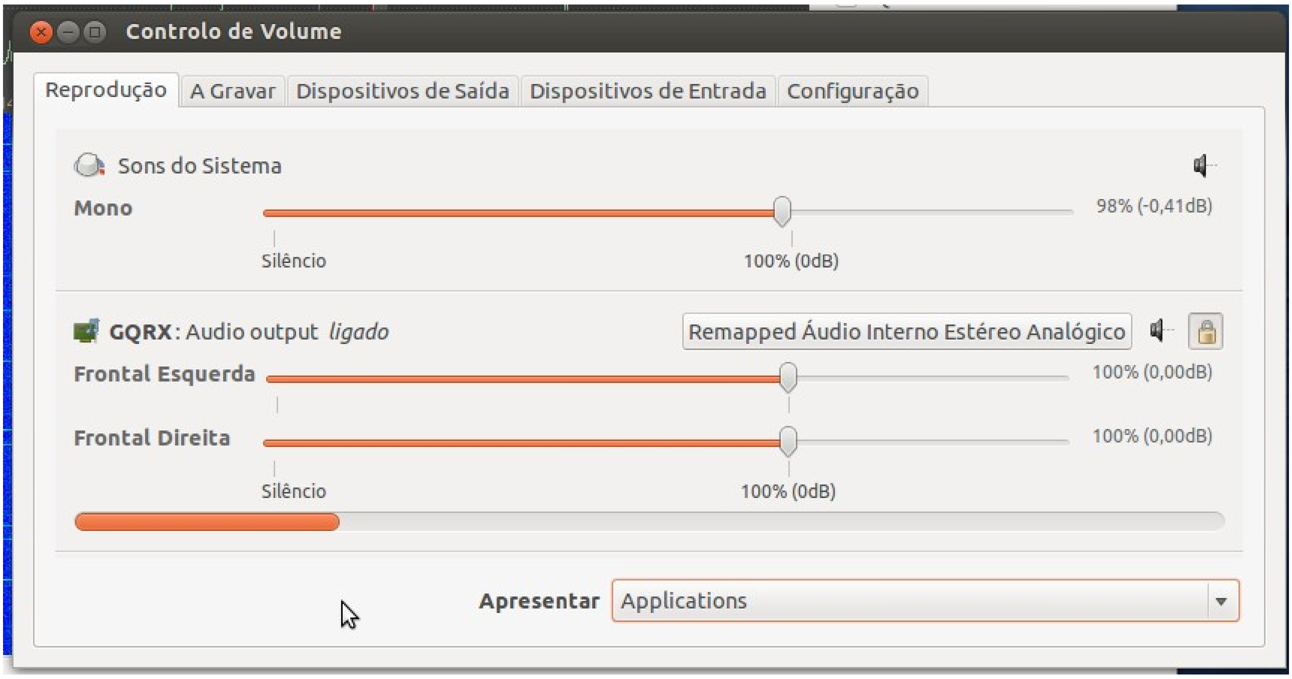
The command should output some number, thats OK.



```
alexandre@moleiro-ct1gvn:~$ pactl load-module module-remap-sink sink_name=secondary master
=alsa_output.pci-0000_00_1b.0.analog-stereo
33
alexandre@moleiro-ct1gvn:~$
```

Start quisk or gqrx and tune into a digital mode frequency using the appropriate mode (usually USB for HF digital modes e.g.).

Now open **pavucontrol** (just type pavucontrol in the launcher) and select the Playback tab. Find the application that is playing audio (quisk or gqrx) and change from normal audio to Remapped audio.



Sons do Sistema
Mono 98% (-0,41dB)

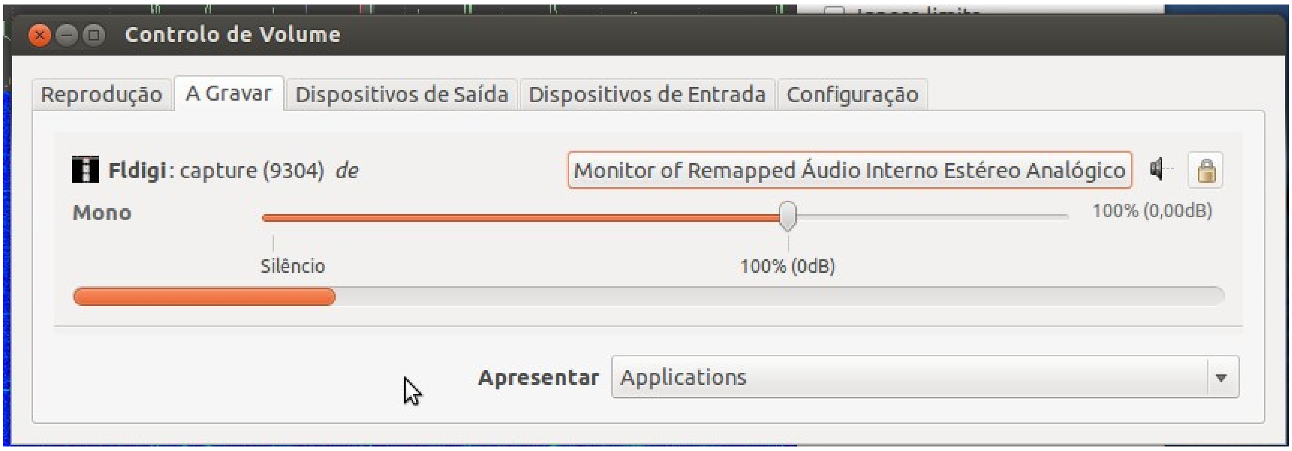
GQRX: Audio output *ligado* — Remapped Áudio Interno Estéreo Analógico
Frontal Esquerda 100% (0,00dB)
Frontal Direita 100% (0,00dB)

Apresentar: Applications

Then open the application for digital modes and start decoding. In this example I will use **fldigi.**

NOTE: Installation and configuration of a digital mode decoding program is not covered in this document

In **pavucontrol** now click on the Record tab. You should see the digital modes program recording audio. Change to Monitor of Remapped Audio.

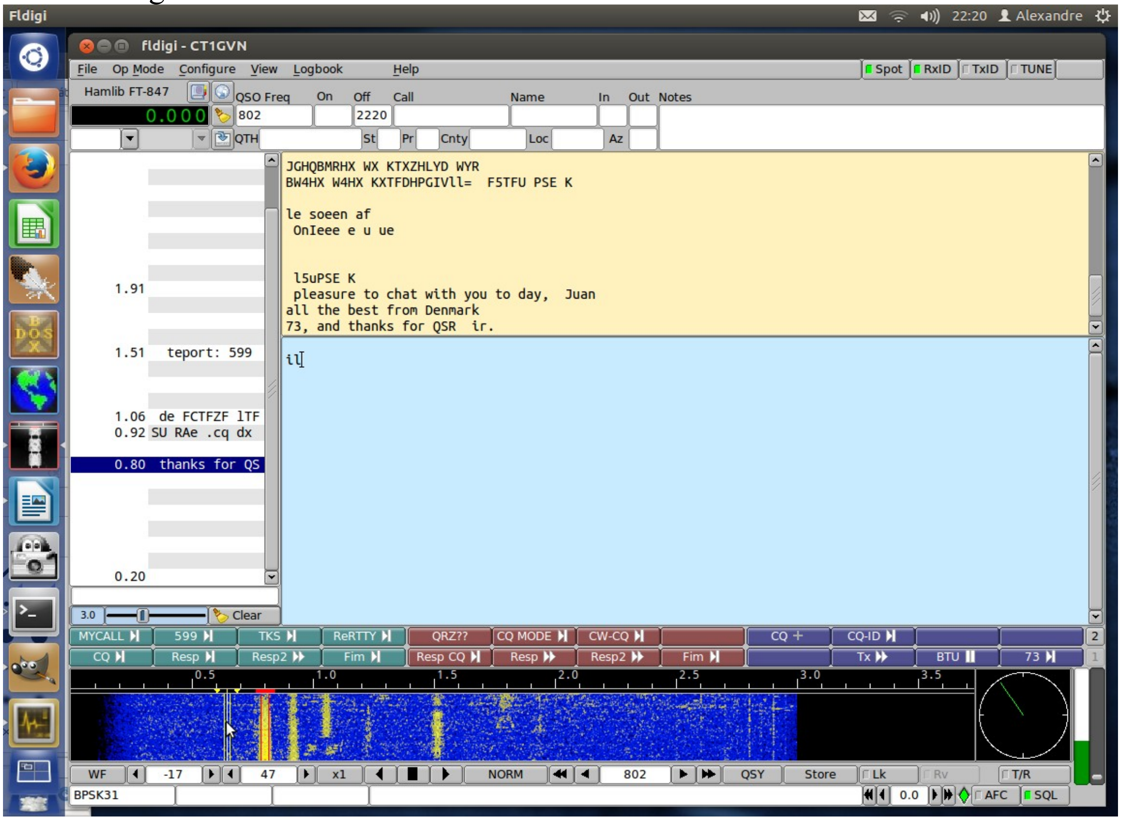


Fldigi: capture (9304) *de* — Monitor of Remapped Áudio Interno Estéreo Analógico
Mono 100% (0,00dB)

Apresentar: Applications

If all went well you digital modes decoding program should be now be “hearing” the output of quisk or gqrs.

Example with fldigi


Fldigi
22:20 Alexandre
fldigi - CT1GVN
File Op Mode Configure View Logbook Help
Spot RxID TxID TUNE
MYCALL 599 TKS ReRTTY QRZ?? CQ MODE CW-CQ CQ + CQ-ID
CQ Resp Resp2 Fim Resp CQ Resp Resp2 Fim Tx BTU 73
Clear
WF QSY Store Lk Rv T/R AFC SQL NORM

Hamlib FT-847

0.000

QSO Freq 802 On Off 2220 Call Name In Out Notes

QTH St Pr Cnty Loc Az

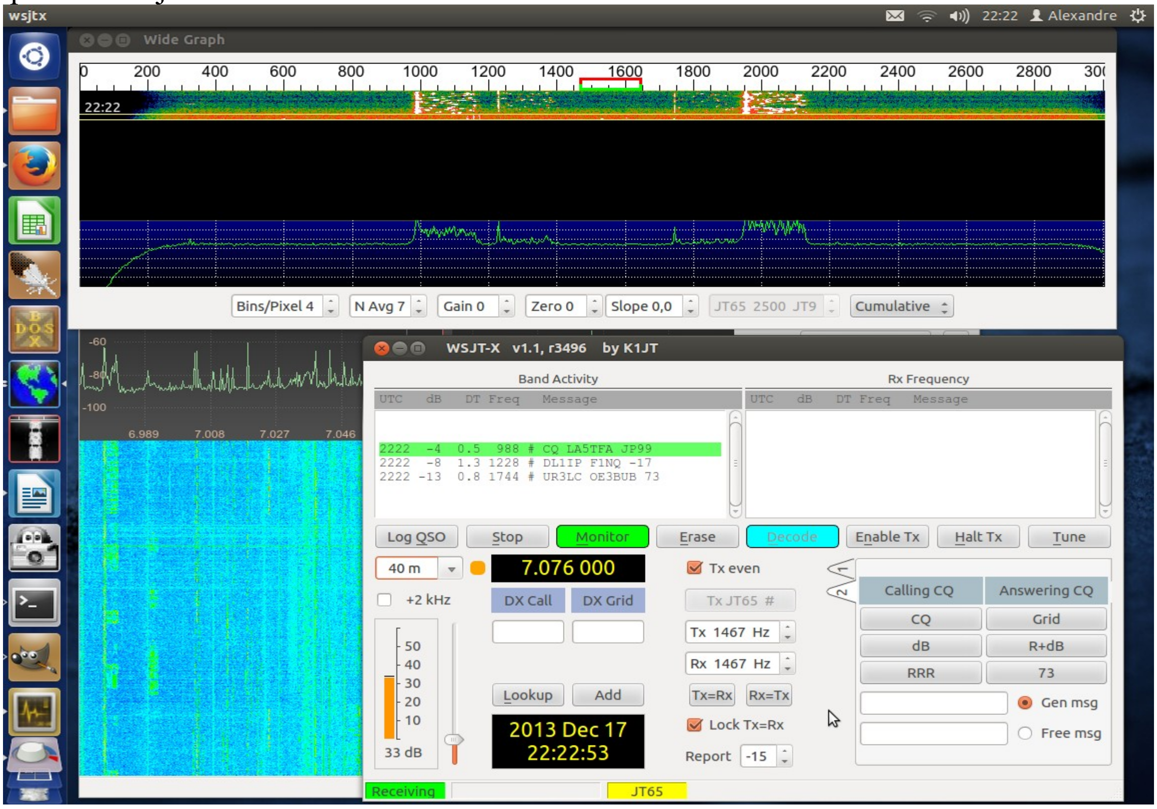
```
JGHQBMRHX WX KTXZHLYD WYR
BW4HX W4HX KXTFDHPGIVll=  F5TFU PSE K

le soeen af
 OnIeee e u ue


 l5uPSE K
 pleasure to chat with you to day,  Juan
all the best from Denmark
73, and thanks for QSR  ir.
```

```
1.91
1.51   teport: 599
1.06   de FCTFZF 1TF
0.92 SU RAe .cq dx
0.80   thanks for QS
0.20
```

WF -17 47 x1 802 0.0

BPSK31

Example with wsjt-x



Bins/Pixel 4 N Avg 7 Gain 0 Zero 0 Slope 0,0 JT65 2500 JT9 Cumulative

```
2222  -4  0.5  988 # CQ LA5TFA JP99
2222  -8  1.3 1228 # DL1IP F1NQ -17
2222 -13  0.8 1744 # UR3LC OE3BUB 73
```

40 m 7.076 000

☐ +2 kHz

☑ Tx even

Tx JT65 #

Tx 1467 Hz

Rx 1467 Hz

☑ Lock Tx=Rx

Report -15

● Gen msg

○ Free msg

33 dB

2013 Dec 17
22:22:53

Receiving JT65

Happy SDRing!

73 de Alex - CT1GVN