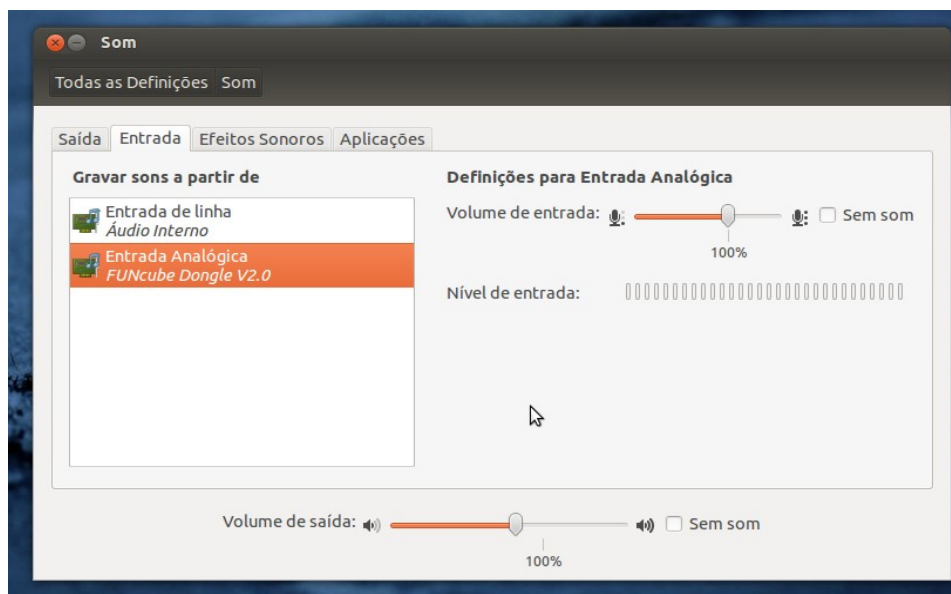


# Getting Started with the FunCube Dongle Pro+ on Ubuntu 12.04 LTS (precise)

## 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.



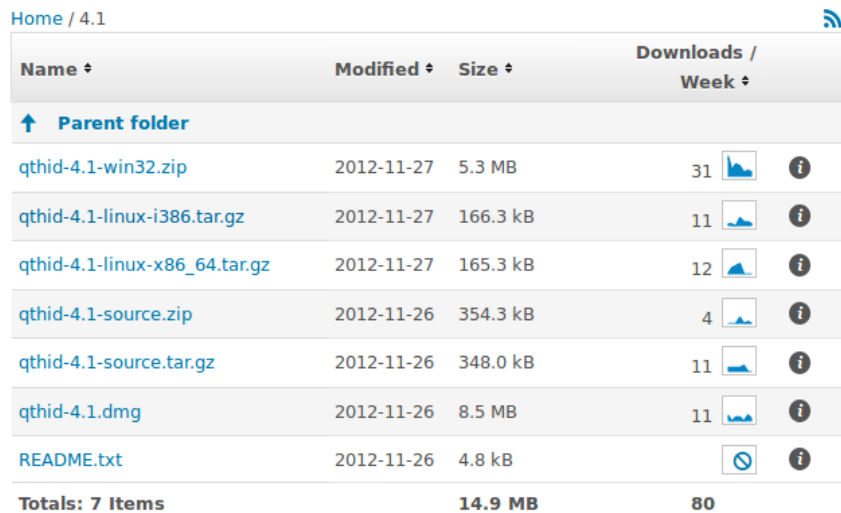
Now open a Console window (Ctrl-Alt-T) and type **arecord -l**  
You should see the FCD Pro+ listed.

```
alexandre@moleiro-ct1gvn: ~  
alexandre@moleiro-ct1gvn:~$ arecord -l  
**** Lista de Dispositivos de Hardware CAPTURE ****  
placa 0: Intel [HDA Intel], dispositivo 0: ALC662 rev1 Analog [ALC662 rev1 Analog]  
Subdispositivos: 1/1  
Subdispositivo #0: subdevice #0  
placa 0: Intel [HDA Intel], dispositivo 2: ALC662 rev1 Analog [ALC662 rev1 Analog]  
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

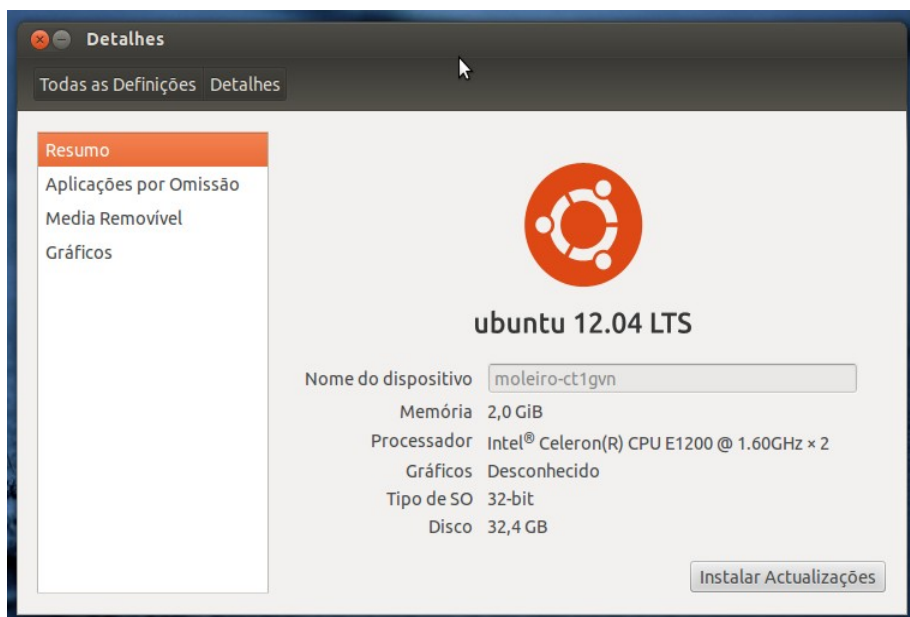
Point your browser to <http://sourceforge.net/projects/qthid/files/4.1/> . If you are running the 32bit version of Ubuntu download [qthid-4.1-linux-i386.tar.gz](#) . If you are running the 64bit flavour choose [qthid-4.1-linux-x86\\_64.tar.gz](#) .



Home / 4.1

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

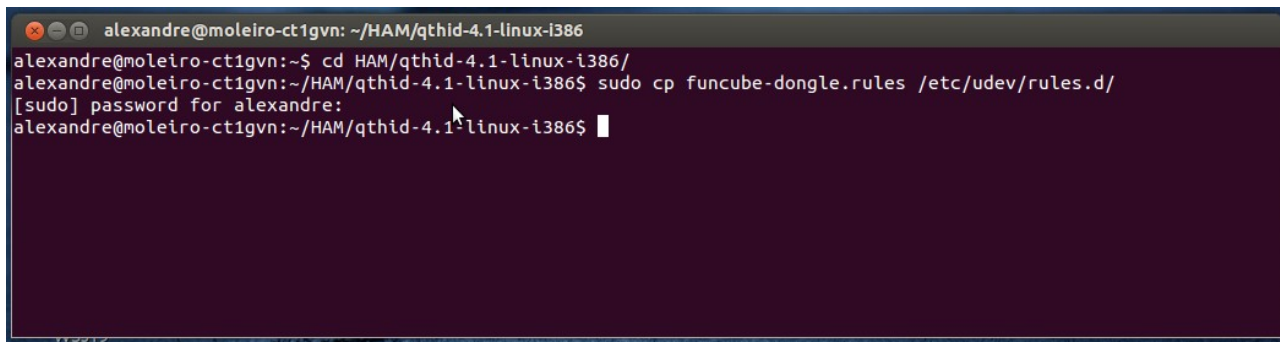
If you are unsure which version you are running go to All Settings / Details.



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).



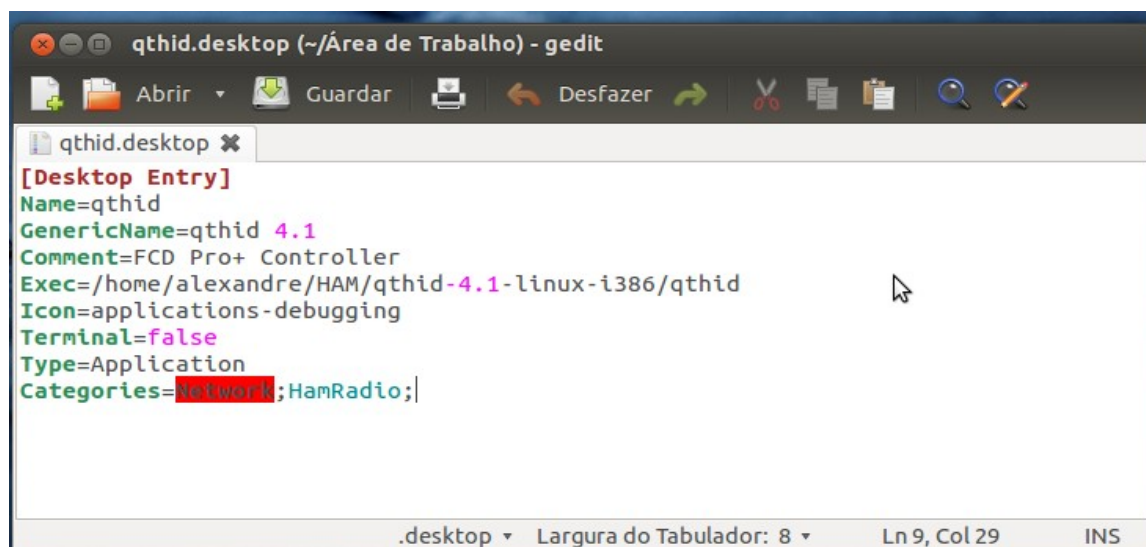
```
alexandre@moleiro-ct1gvn: ~/HAM/qthid-4.1-linux-i386
alexandre@moleiro-ct1gvn:~$ cd HAM/qthid-4.1-linux-i386/
alexandre@moleiro-ct1gvn:~/HAM/qthid-4.1-linux-i386$ sudo cp funcube-dongle.rules /etc/udev/rules.d/
[sudo] password for alexandre:
alexandre@moleiro-ct1gvn:~/HAM/qthid-4.1-linux-i386$
```

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**



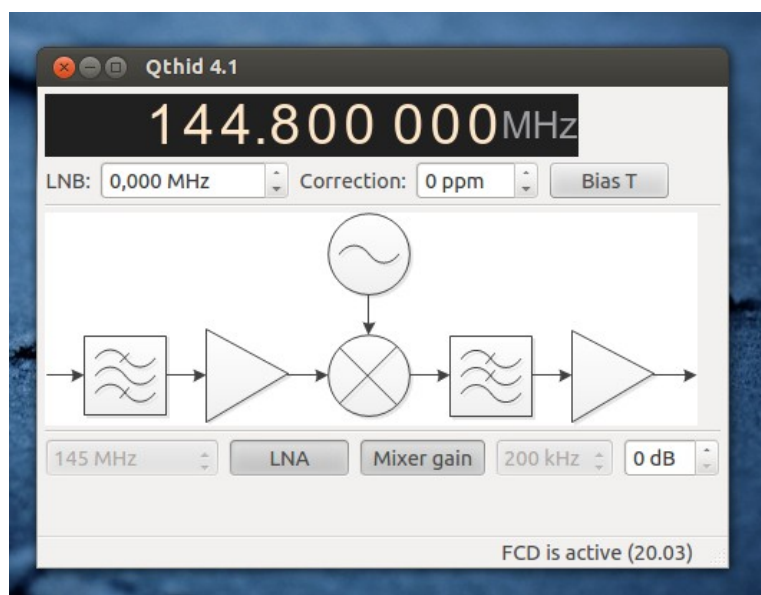
```
qthid.desktop (~/Área de Trabalho) - gedit
[Desktop Entry]
Name=qthid
GenericName=qthid 4.1
Comment=FCD Pro+ Controller
Exec=/home/alexandre/HAM/qthid-4.1-linux-i386/qthid
Icon=applications-debugging
Terminal=false
Type=Application
Categories=Network;HamRadio;
```

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.



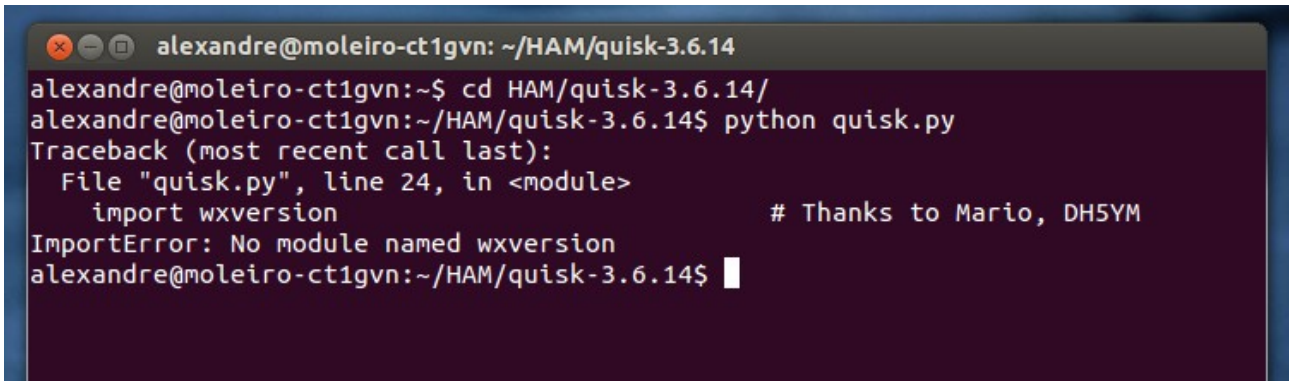
If the status bar reads “FCD is active” Congratulations, you have a working FCD Pro+  
If the status bar reads “No FCD detected” try closing qthid 4.1, removing and inserting the FCD Pro+ and opening qthid 4.1 again. If that fails try repeating Step 2 from the beginning again.

### 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](http://james.ahlstrom.name/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**



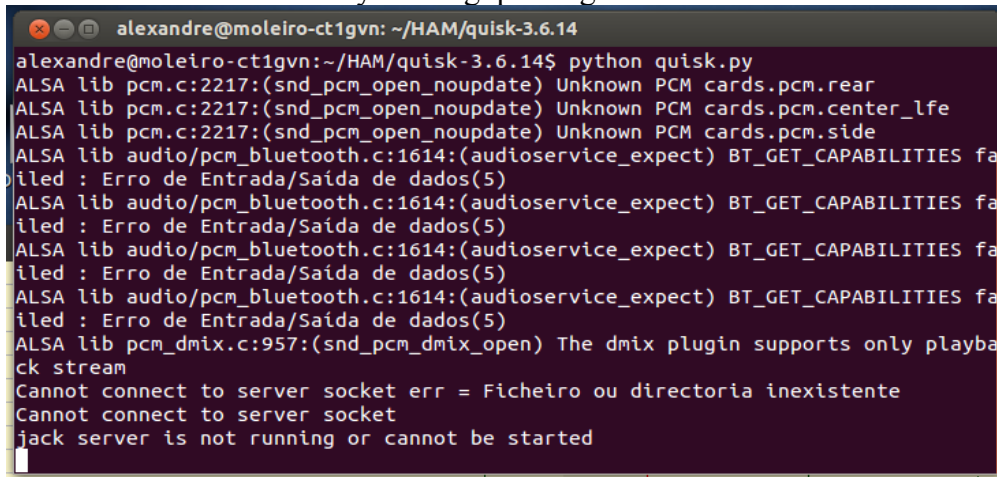
```
alexandre@moleiro-ct1gvn: ~/HAM/quisk-3.6.14
alexandre@moleiro-ct1gvn:~$ cd HAM/quisk-3.6.14/
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ python quisk.py
Traceback (most recent call last):
  File "quisk.py", line 24, in <module>
    import wxversion                                     # Thanks to 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):

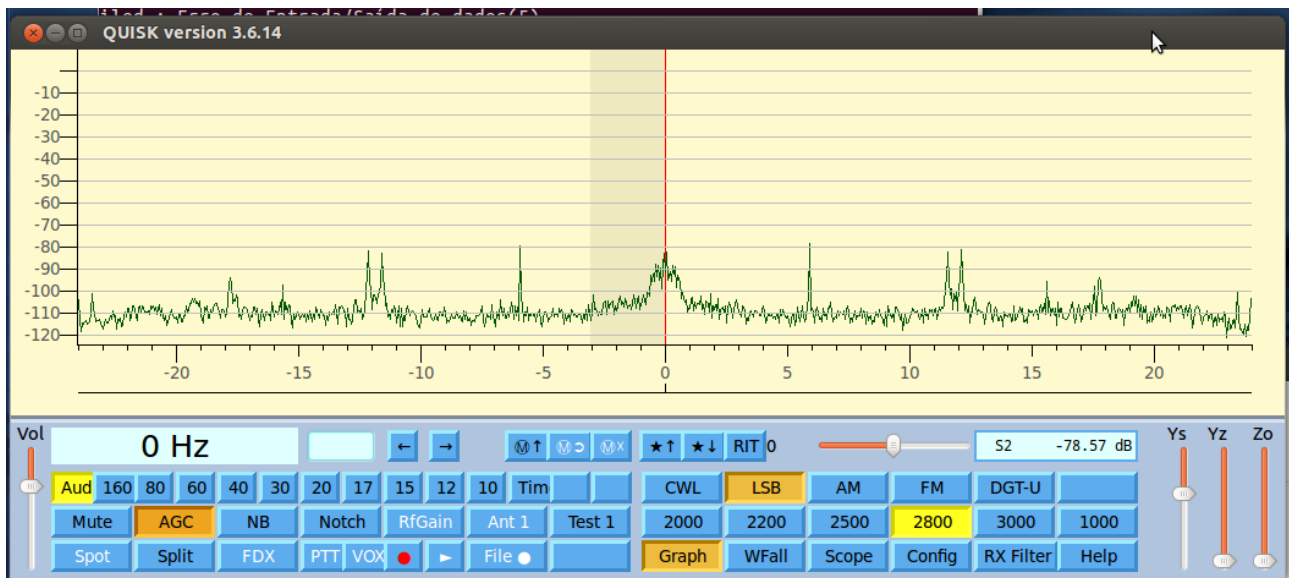
- python2.7
- python-wxgtk2.8
- ruby-fftw3

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



```
alexandre@moleiro-ct1gvn: ~/HAM/quisk-3.6.14
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 then choose the **Sound** tab.

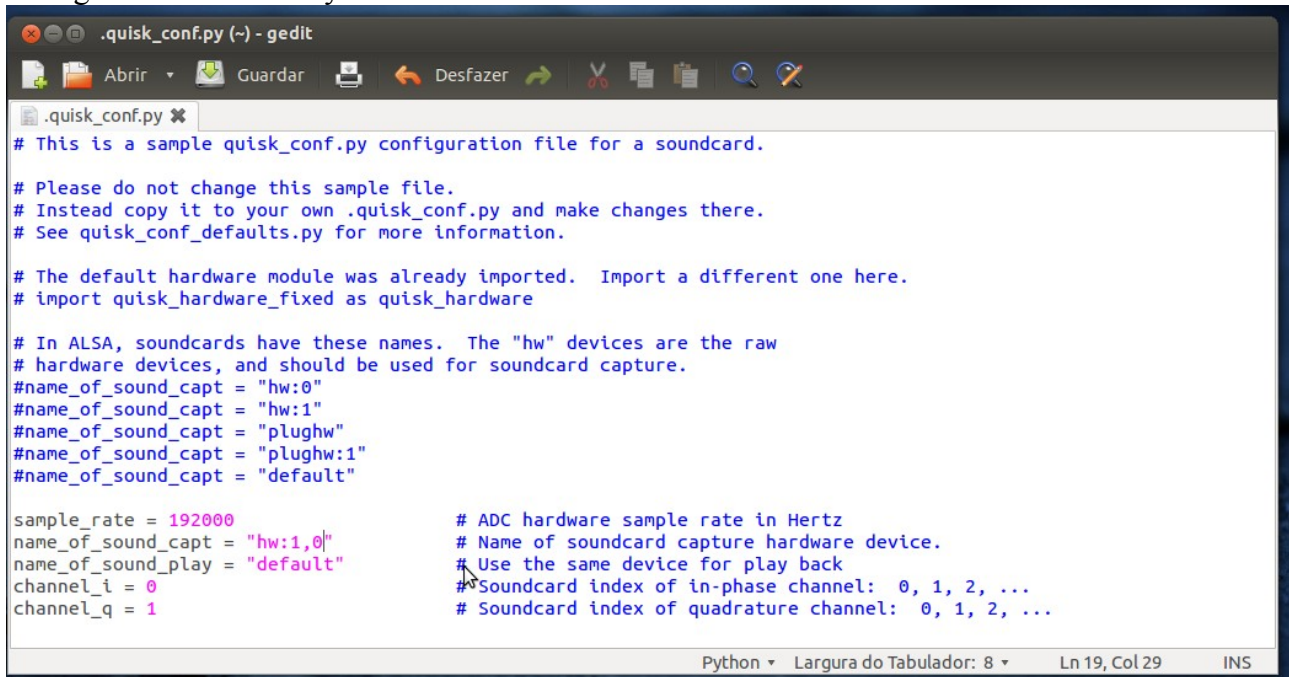


Write down the hw code you see for the FCD Pro+, in my case hw:1,0

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:



```
.quisk_conf.py (~) - gedit
# 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, ...

Python  Largura do Tabulador: 8  Ln 19, Col 29  INS
```

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.

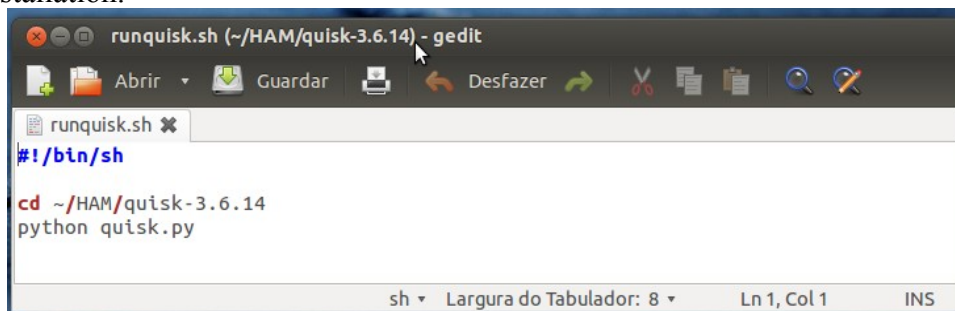
You can close the text editor.

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.



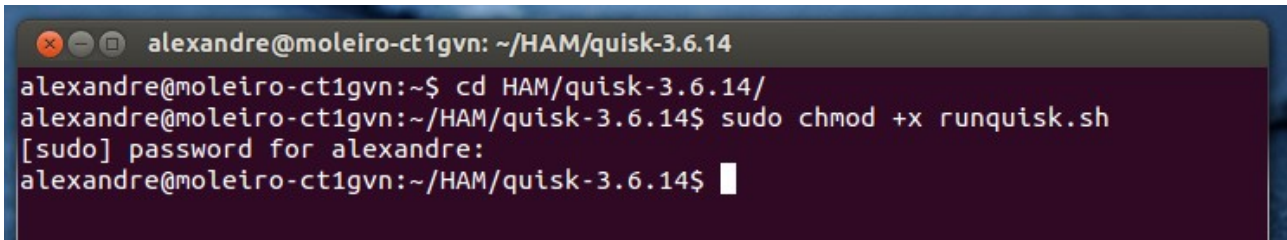
```
runquisk.sh (~/.HAM/quisk-3.6.14) - gedit
#!/bin/sh

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

sh  Largura do Tabulador: 8  Ln 1, Col 1  INS
```

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**

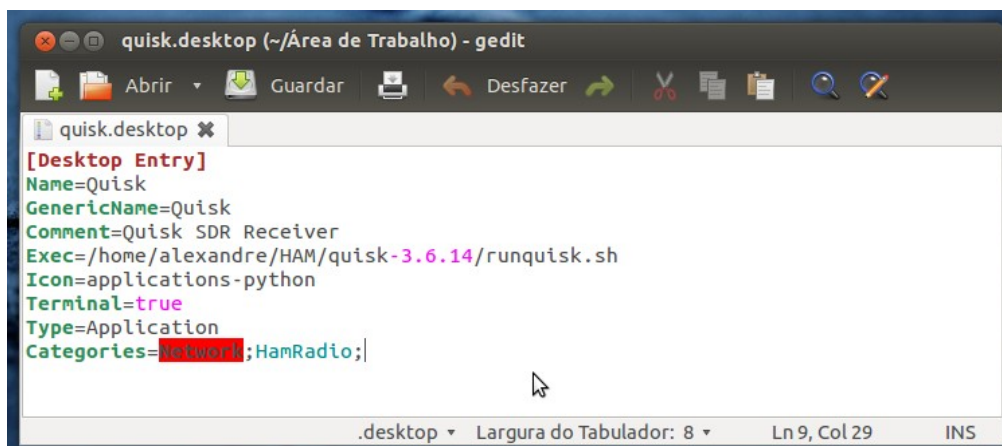


```
alexandre@moleiro-ct1gvn: ~/HAM/quisk-3.6.14
alexandre@moleiro-ct1gvn:~$ cd HAM/quisk-3.6.14/
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$ sudo chmod +x runquisk.sh
[sudo] password for alexandre:
alexandre@moleiro-ct1gvn:~/HAM/quisk-3.6.14$
```

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 installation 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;
```

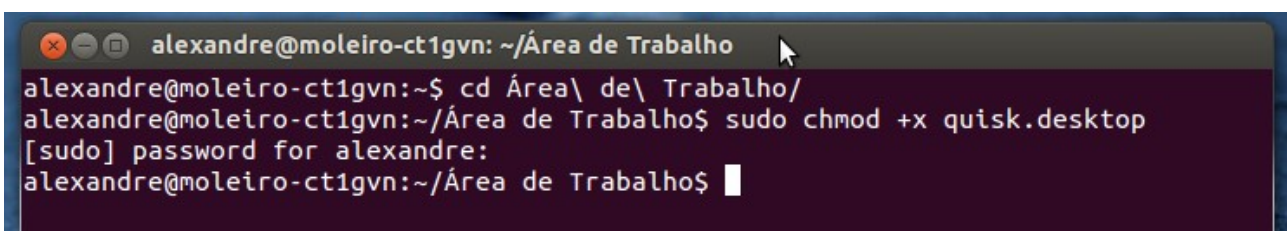


```
quisk.desktop (~/Área de Trabalho) - gedit
quisk.desktop
[Desktop Entry]
Name=Quisk
GenericName=Quisk
Comment=Quisk SDR Receiver
Exec=/home/alexandre/HAM/quisk-3.6.14/runquisk.sh
Icon=applications-python
Terminal=true
Type=Application
Categories=Network;HamRadio;
```

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: ~/Área de Trabalho
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$
```

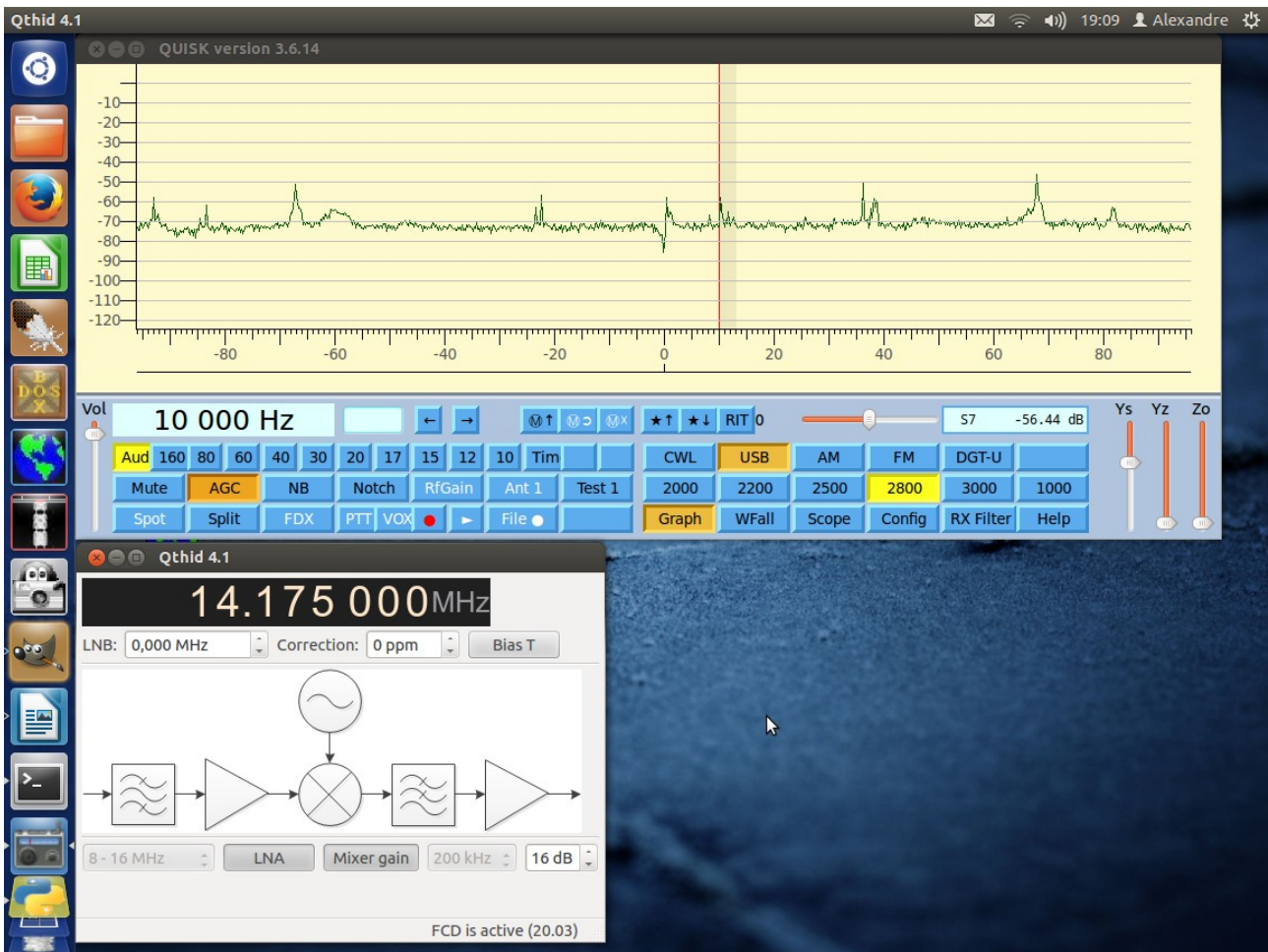


You should now have a desktop icon for quisk.

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 the center frequency, so even if you click the buttons to change band the center frequency will be always 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.



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: ~
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 not 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 previous source installations are either removed or disabled before installing GqrX from 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:~$
```

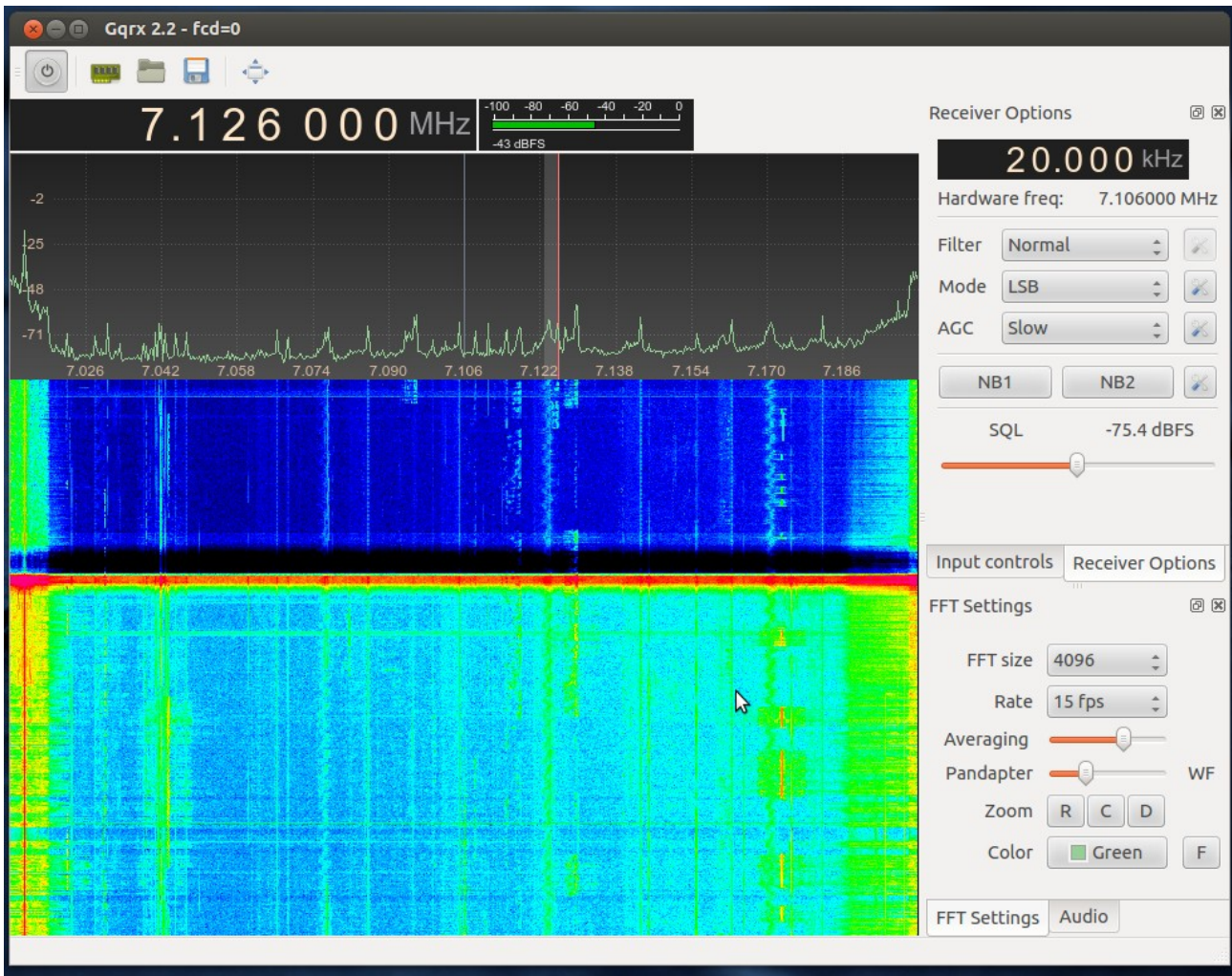
Again, on the Console, you need to issue the command **sudo apt-get update**. This will update the list of available packages, it may take some time, wait till it finishes.

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:



After you press ok you get the gqr main window.



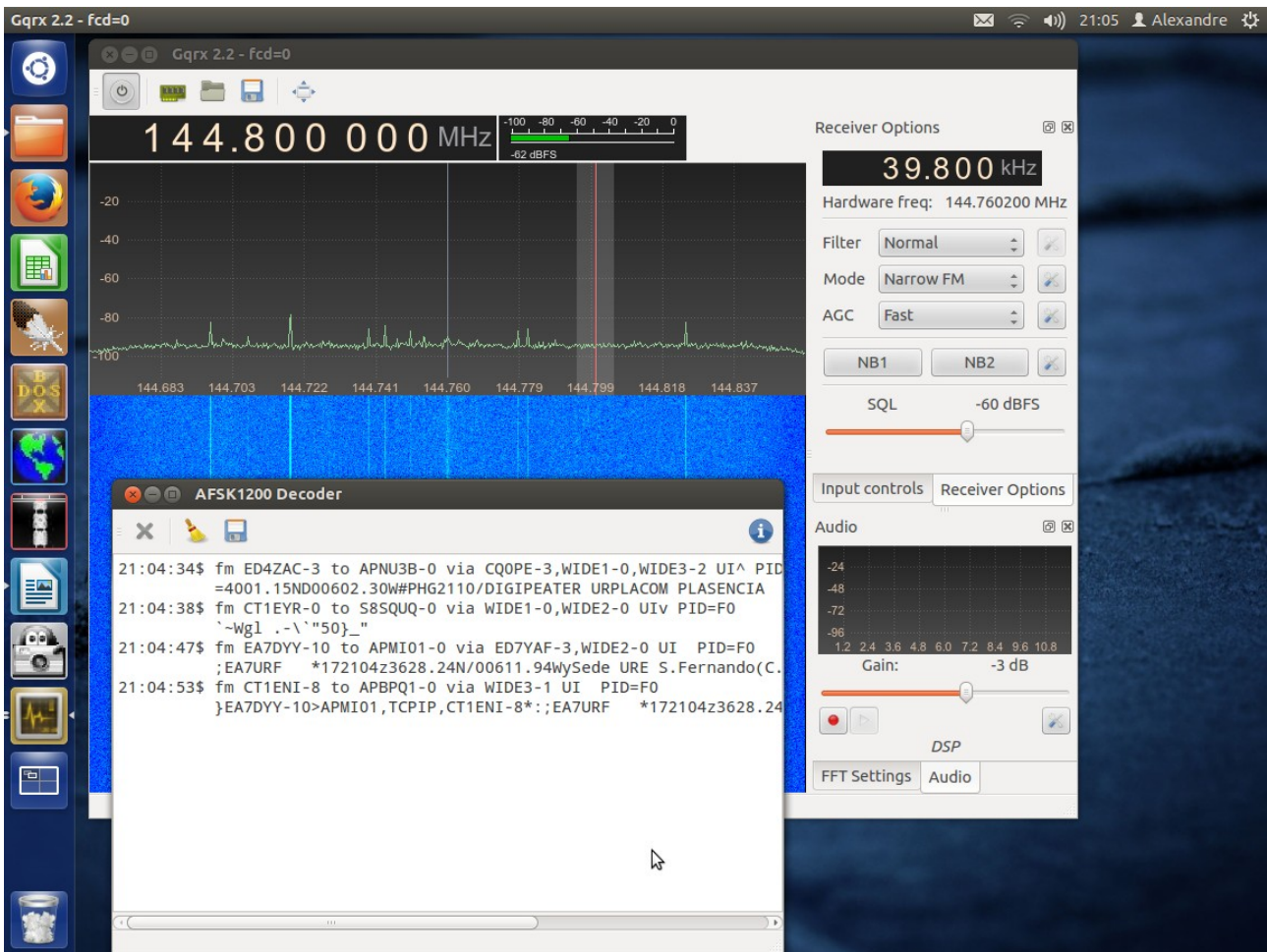
Last step is to create a desktop shortcut to gqr.

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

With gqr 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.



## 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.

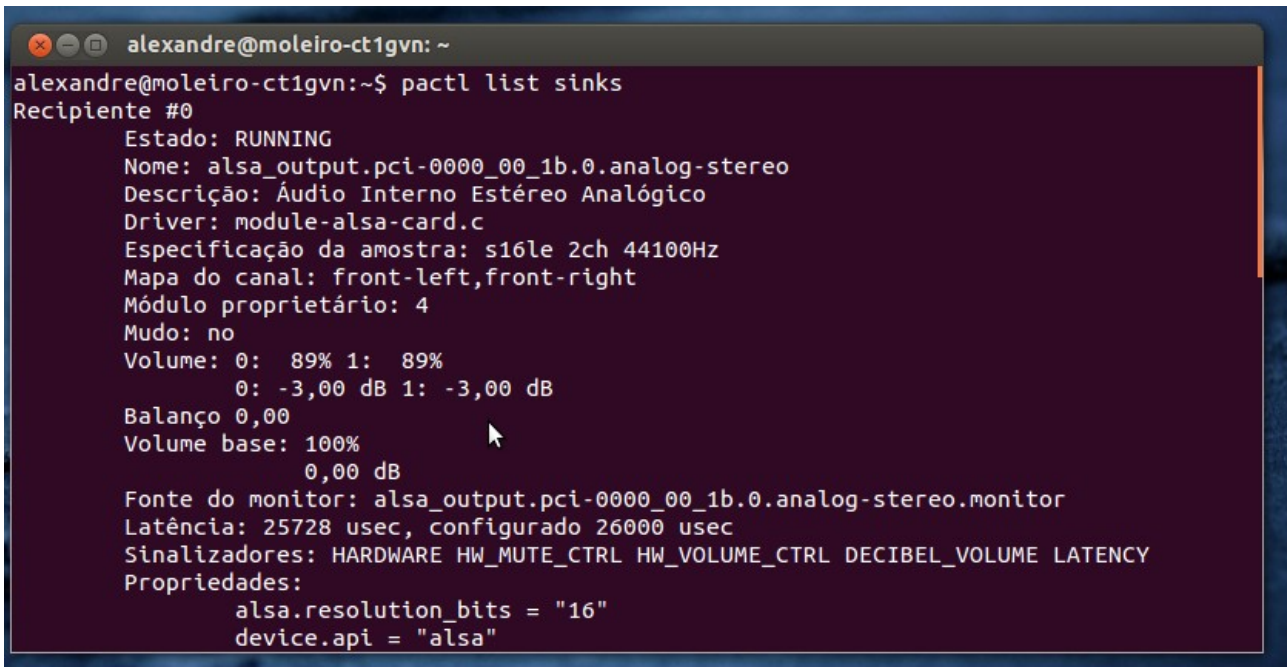
For receiving digital modes you need dedicated programs.

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.



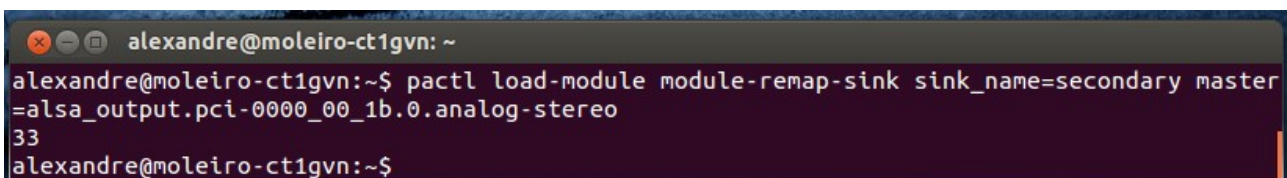
```
alexandre@moleiro-ct1gvn: ~
alexandre@moleiro-ct1gvn:~$ pactl list sinks
Recipiente #0
  Estado: RUNNING
  Nome: alsa_output.pci-0000_00_1b.0.analog-stereo
  Descrição: Áudio Interno Estéreo Analógico
  Driver: module-alsa-card.c
  Especificação da amostra: s16le 2ch 44100Hz
  Mapa do canal: front-left,front-right
  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.

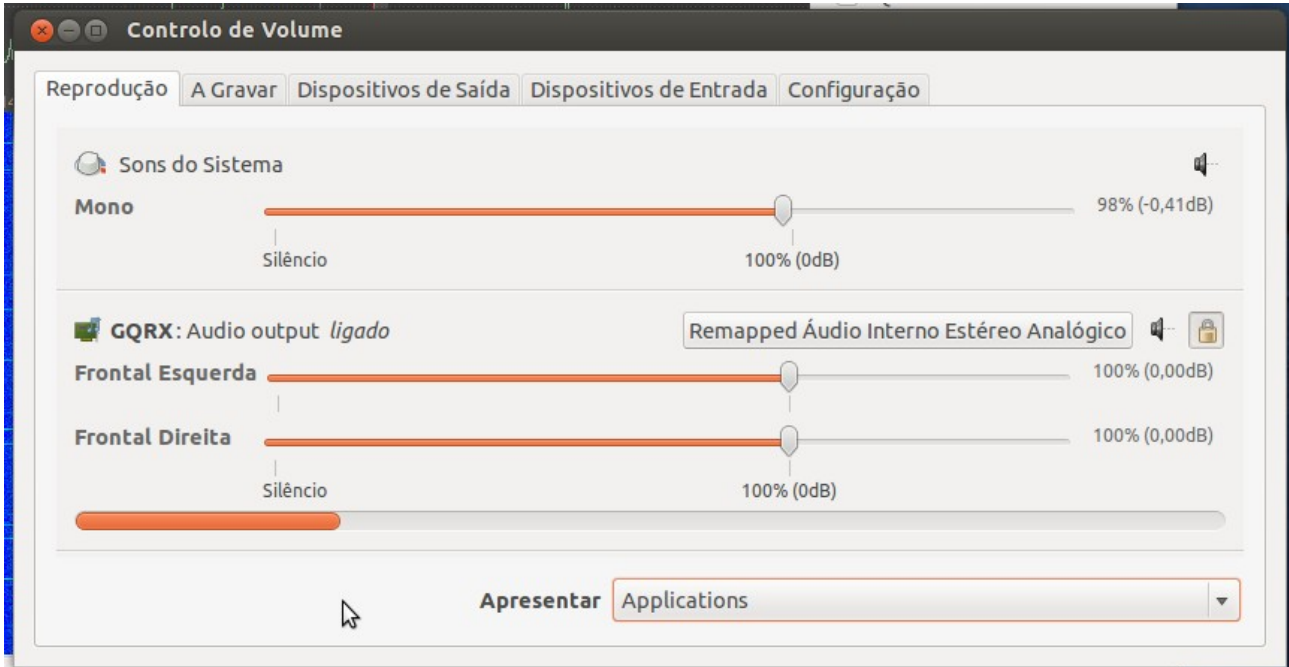
The command should output some number, thats OK.



```
alexandre@moleiro-ct1gvn: ~
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.).

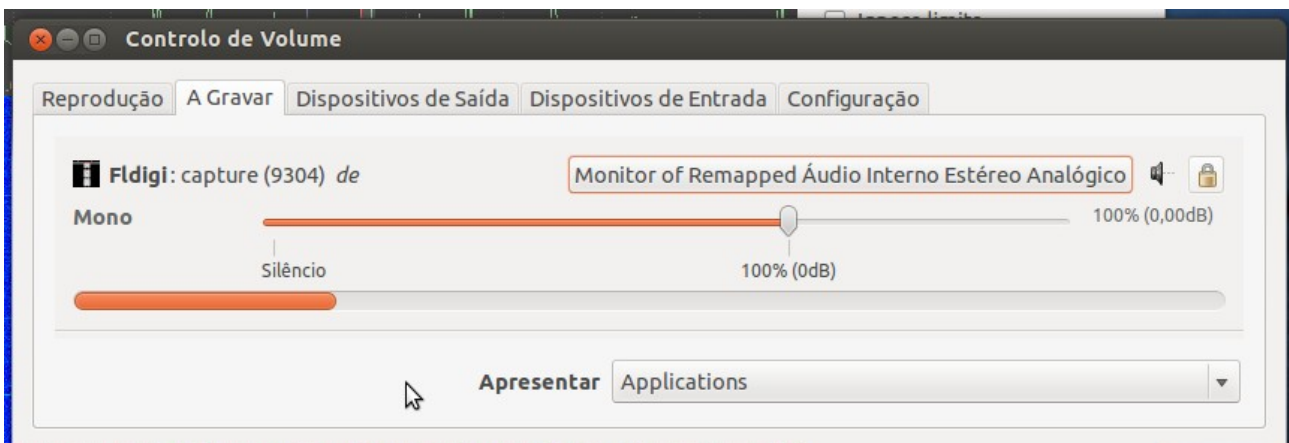
Now open **pavucontrol** (just type pavucontrol in the launcher) and select the Playback tab. Find the application that is playing audio (quisk or gqrX) and change from normal audio to Remapped audio.



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

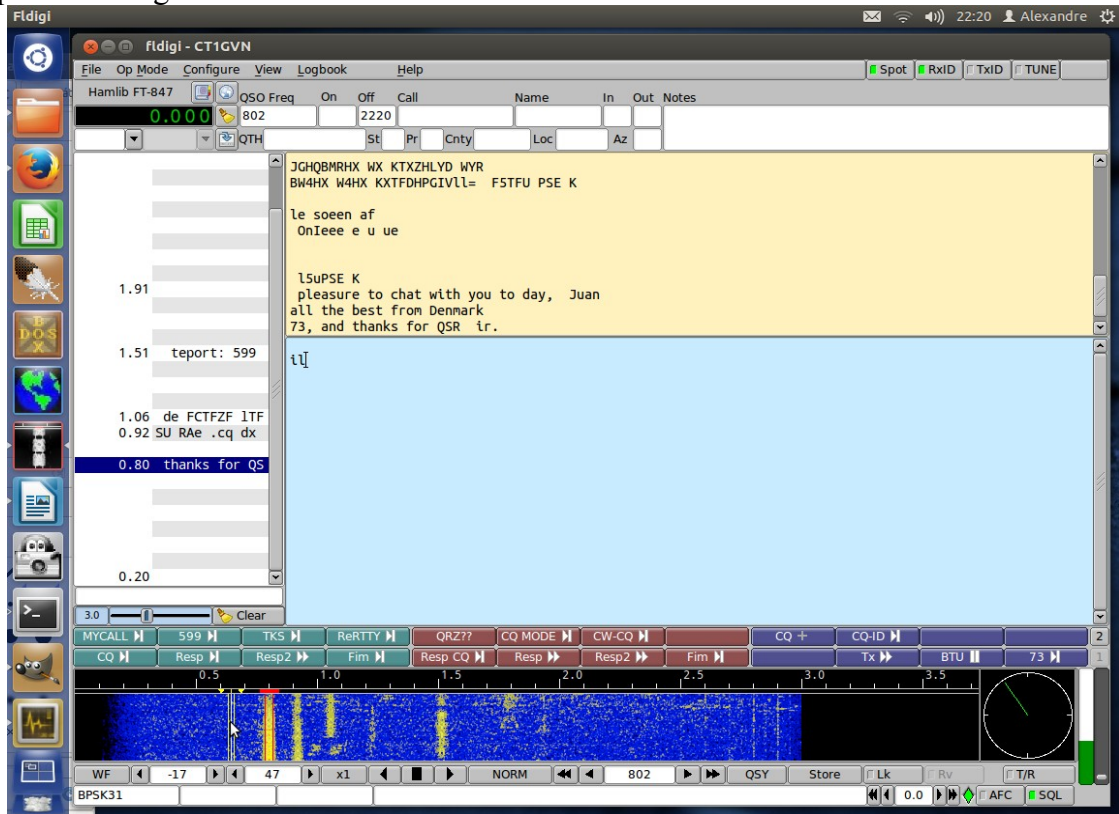
NOTE: Installation and configuration of a digital mode decoding program is not covered in this document

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

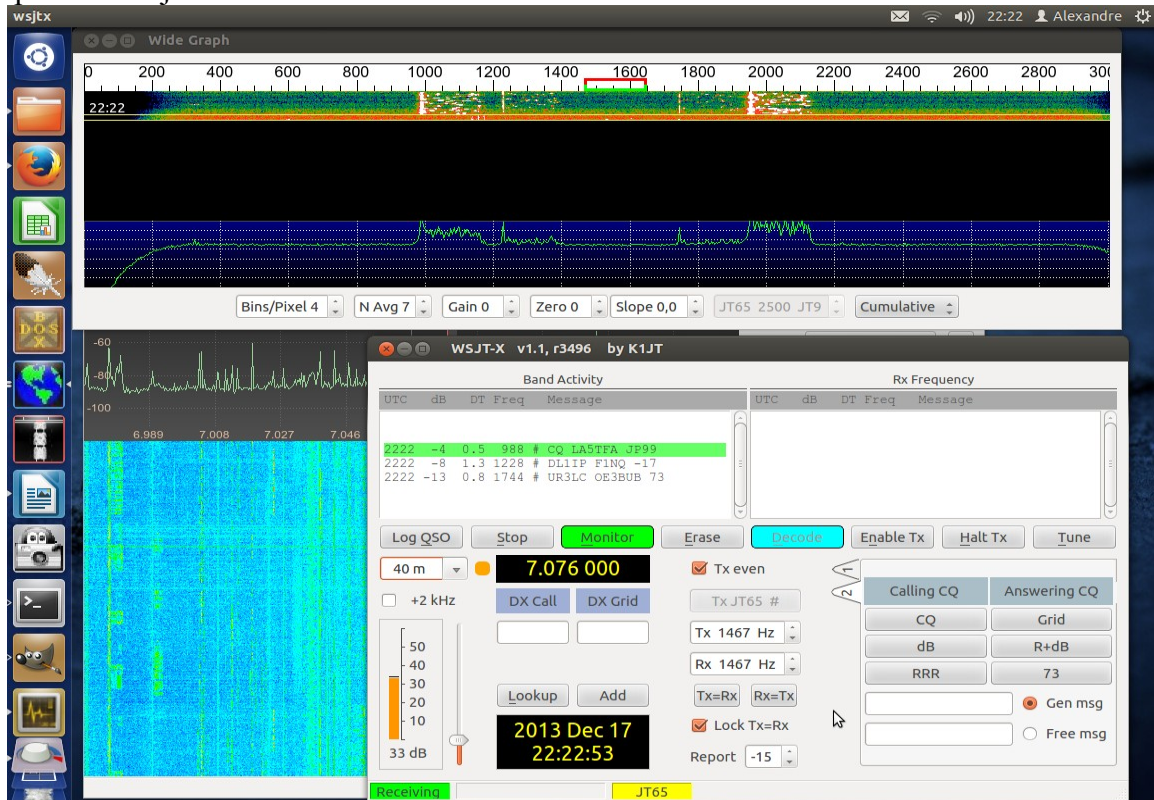


If all went well you digital modes decoding program should be now be “hearing” the output of quisk or gqrs.

## Example with fldigi



## Example with wsjt-x



Happy SDRing!

73 de Alex - CT1GVN