# Apache Labs LLC


# piHPSDR Controller

# Users Guide

**by**

**John Melton GØORX/N6LYT**

**This document contains references to the Apache Labs Transceiver products**
**http://www.apache-labs.com**

In cooperation with VK6PH, NRØV, W5WC, K5SO
and the OpenHPSDR Hardware and Software Projects
http://openhpsdr.org



November 26, 2016

# Contents - piHPSDR Controller Users Guide

## Introduction

## Introduction by John Melton GØORX/N6LYT

### Thu Aug 5, 2016 12:52 am (PDT) .

### Posted by: "John Melton" g0orx

I think it is time to answer some of the questions about the controller.

Some background ...

I presented a paper at DCC last year on how to build a CAT controller using an Arduino and some buttons and encoders, even one with an LCD screen showing the mode, frequency and an S meter. The idea of this was to try to get people building them.

When I returned I started looking at what a Raspberry Pi (model 2 at the time) could do as a standalone controller. I had already ported WDSP to Linux so had a good starting point. Very soon after I received an email from LA2NI, Kjell Karlsen asking if my Android code would run on the Raspberry Pi as he wanted to try to build a standalone system with a touch screen. I told him about my early development work on the Raspberry Pi and that started a project that is still on going today.

The idea was not to build an all singing/dancing copy of PowerSDR but to build something that could be used with a small portable system. Kjell has since gone on to build a very nice system using a Hermes and Apollo all built into a small case with the Raspberry Pi (now a model 3).

Abhi (Apache Labs) got interested in the project and wanted to build a controller initially without the built in Hermes/Apollo. Hopefully we will see a complete portable system sometime from Apache Labs.

So, that is what we have today. The software is still in development and new features are being added as well as bugs being fixed. It is designed to use with the radio, not as a remote console. The RPi does not have any audio input but does have audio output. It is currently possible to output the audio locally on the RPi but to connect a microphone you will need to plug a USB audio card into the device. I will be adding support for connecting a microphone to the this as well.

The RPi will work over WiFi using the built in controller. The signal quality will determine what sample rate it will run at. The RPi does not have GigE so cannot connect directly to a radio running the new Ethernet protocol, but will work through a switch or over WiFi. However, it has been tested using a USB to GigE dongle and that does work. Just remember this will not run at the higher sample rates.

I have looked at several other more powerful single board computers, but always I have run into problems with their implementation of the GPIO, which is used to interface the buttons and encoders.

We are using the Raspberry Pi official 7 inch touch screen. This has a resolution of 800x480 so we are struggling with screen real estate.

There is work in progress to also use the controller remotely across the Internet. This will require a server running at the radio that is connected to by a thin client running on the controller. This will stream the audio and spectrum information with all the DSP work being done on the server. It is not based on my ghpsdr3 project but of course that was a good starting point for the design.

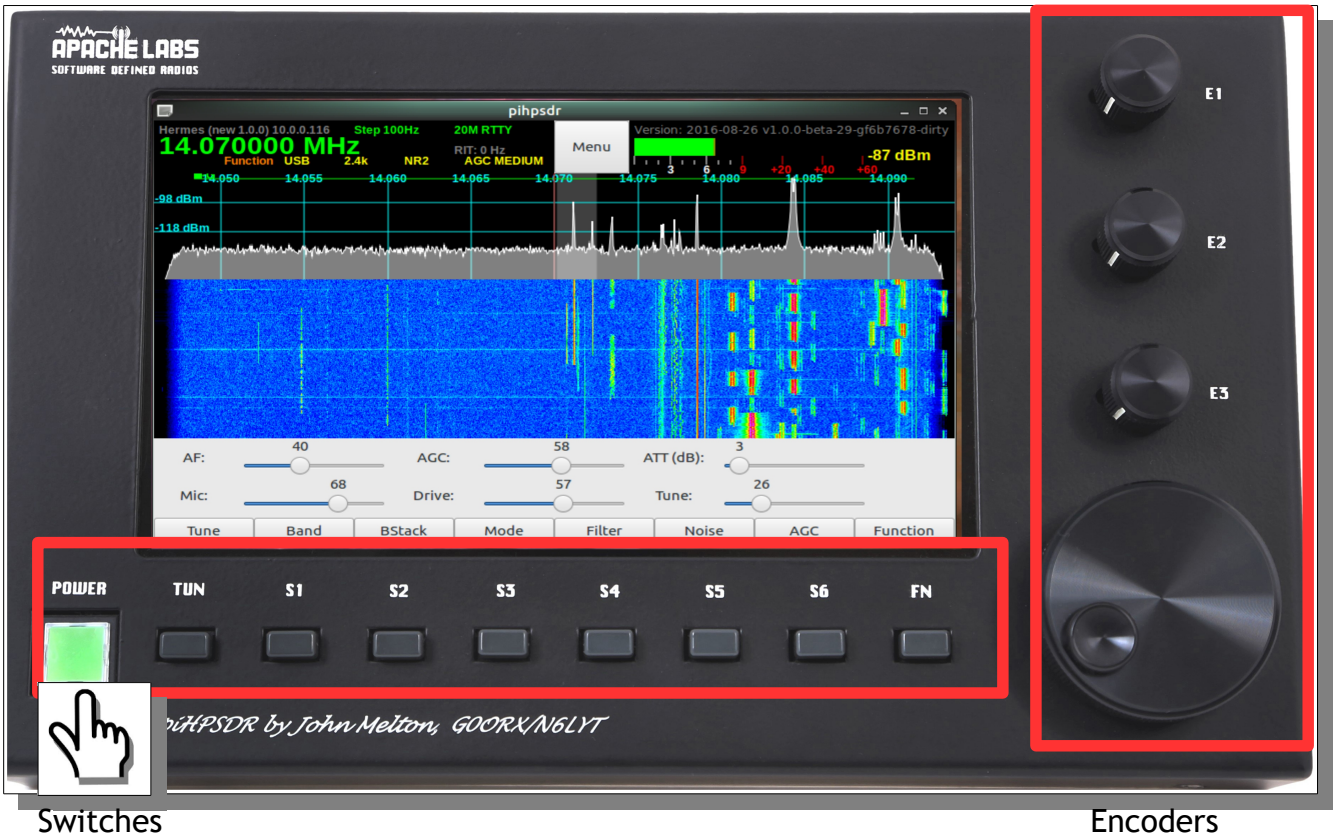
Finally, all the software is open source. It will run on a Raspberry Pi without the buttons/encoders. It will also run on other Linux systems and I have even built it and had it running on Mac OS X. I have not tried to build it on Windows as I do not have a Windows system.

-- John g0orx/n6lyt

---

**Note: This Note: this document assumes familiarity with the Apache Transceiver and standard software such as PowerSDR or Thetis. The Apache Transceiver Users Guides are located here: https://apache-labs.com/al-downloads/1001/ANAN-USER-GUIDES.html**

## 1. Front panel controls




Switches Encoders

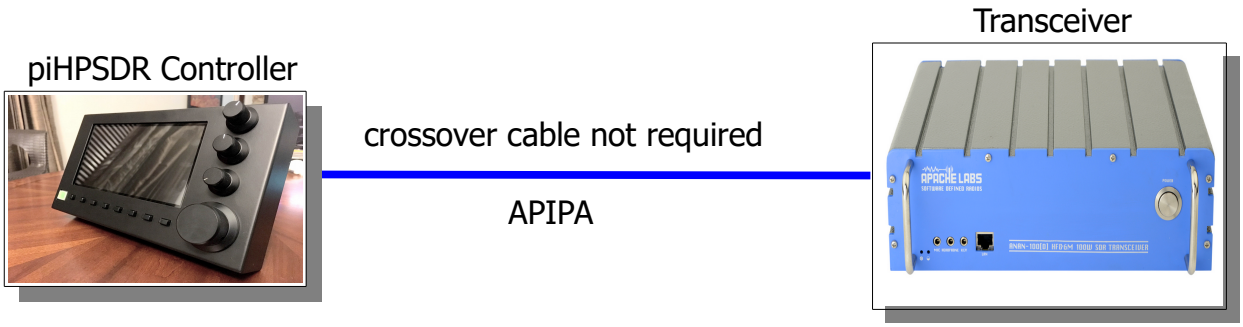
## Default assignments of switches and rotary encoders

- Power ON/OFF – switches external 12vdc 2A power to the Controller
- TUN – MOX/TUNE
- S1 - Band
- S2 – Band Stack
- S3 - Mode
- S4 – Filter
- S5 – Noise Blanker
- S6 - AGC
- FN – Function switch to toggle TUNE or MOX button and E1, E2, E3
- E1 -- two functions AF Gain / MIC Gain
- E2 – two function RF Drive / TUNE Drive
- E3 – two function AGC Gain / Rx Attenuation
- E4 VFO – main tuning knob

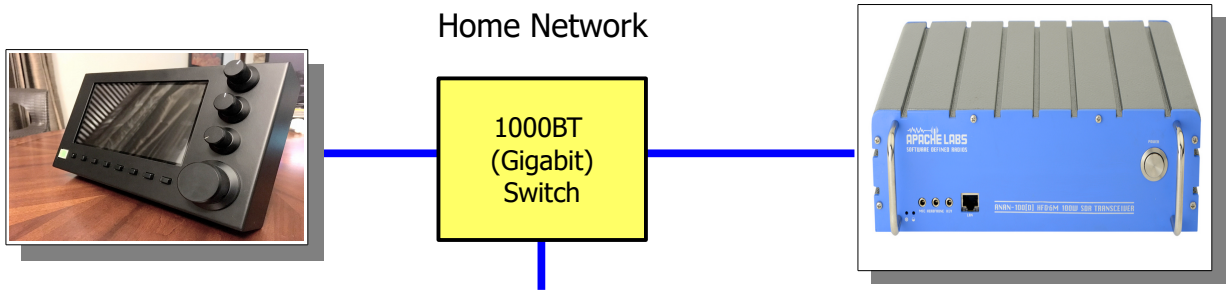
**Appendix** page 40 has a table showing the Switch, Encoder, and Touch Screen activation details.

## Network LAN connection

Auto-negotiation of a the RaspberryPi 100BT Ethernet connection allows operation of the Controller at the 48000, 96000, and 192Ksps sample rates.




The new piHPSDR-Controller can be connected directly to your ANAN Transceiver and each unit will make use of the APIPA assignment for an IP address.




CAT-5 or CAT-6 shielded cables are recommended for connection of the Controller

## Controller Hardware requirements

- 13.8vdc 2.5A (minimum 12vdc 2A) Power connection
- 100BT LAN connection to your Apache Transceiver

## Controller software requirements:

- Pihpsdr Controller Raspbian Operating System and application program are factory installed on an SDHC card.

  If you have purchased the piHPSDR-Controller KIT, you will need to create and install an SDHC card with the RaspberryPi operating system and pihpsdr application.

  Instructions for Installation are detailed in the Install pdf document that is available at:

  https://github.com/g0orx/pihpsdr/raw/master/release/documentation/pihpsdr-install.pdf

- If you wish to get the latest version of pihpsdr source code, executable binaries, and documentation the instructions are available at:

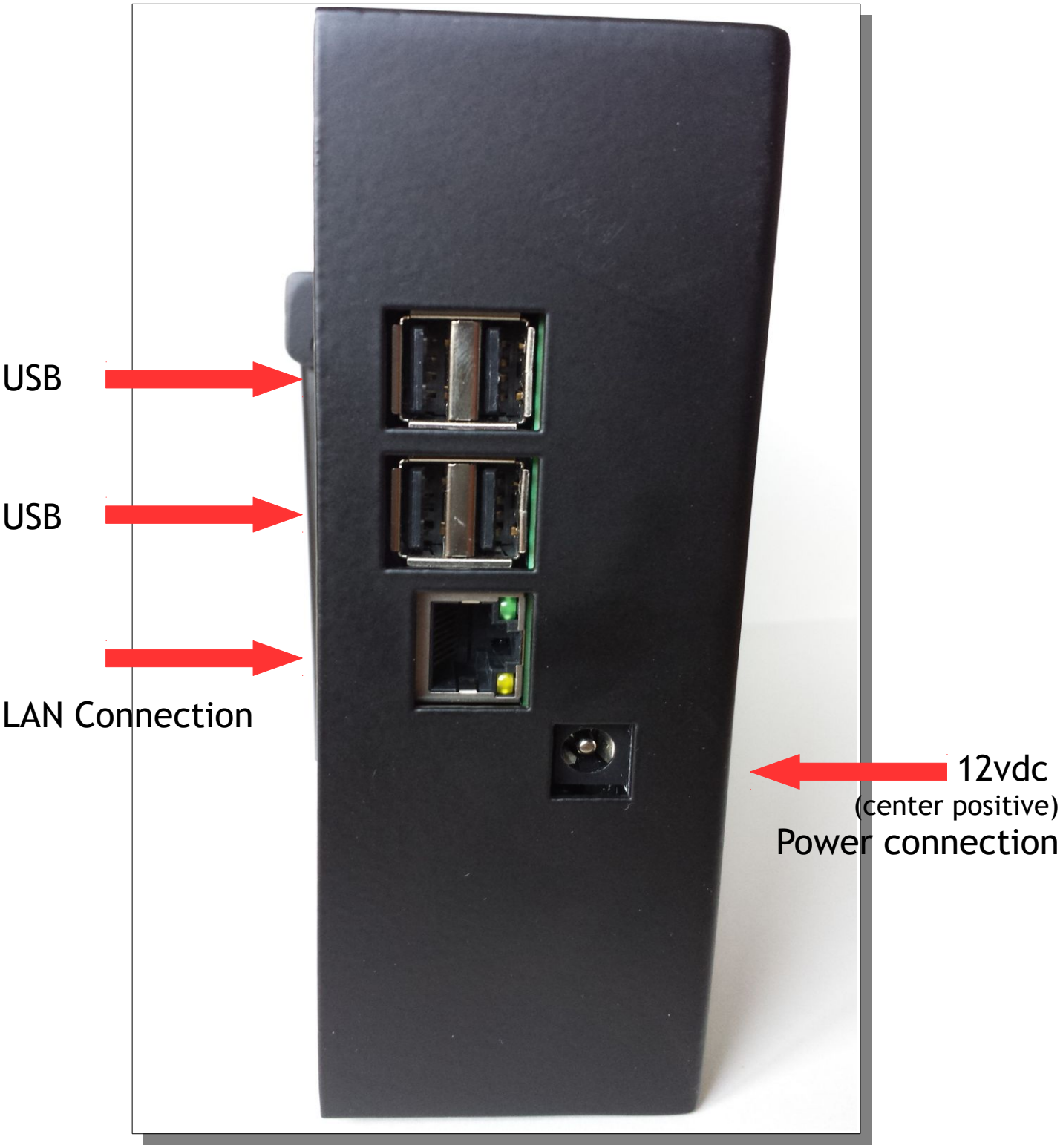
  https://github.com/g0orx/pihpsdr/raw/master/release/documentation/pihpsdr-build.pdf

**Note:** it is a good idea to make a backup of your Operating System and piHPSDR software. Use the RaspberryPi Menu → Accessories → SD Card Copier" utility. Use a new SDHC card of the same capacity as the original.

This is easily done using a USB – SDHC card reader/writer dongle plugged into one of the RPi USB ports on the left side of the piHPSDR Controller.

**A pdf document reader is included in the standard RaspberryPi operating system distribution.**

## PiHPSDR Controller - side panel



## 2. Quick Start Instructions

please refer to front and back panel illustrations

### Hardware Setup

- Carefully unpack the piHPSDR Controller.
- Connect a CAT 5/6 Ethernet cable between the rear panel LAN jack with proper access to your Apache Transceiver, Gigabit switch, or suitable router. Please see Network LAN connection page 9.
- Connect the supplied piHPSDR Controller power cable to a **fused** 12vdc typical Amateur Radio power supply.

### Software Setup

The piHPSDR Controller software is pre-formatted on the Operating System SDHC card inserted in the RaspberryPi.

Alternative software configurations are not provided by Apache Labs.

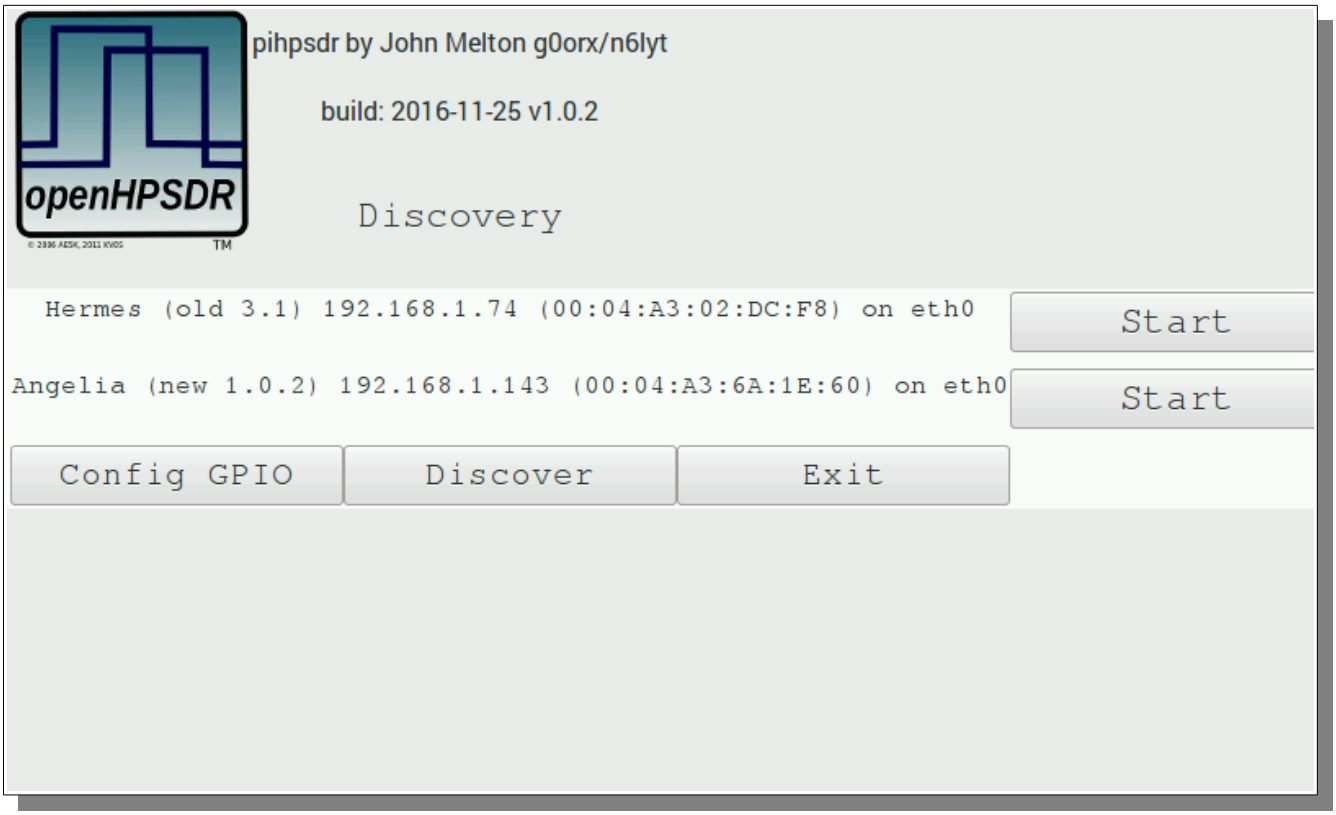
A person familiar with the Raspbian operating system may wish to modify or update the contents of the SDHC card using appropriate Linux tools. Please refer to page 10.

Note: pihpsdr is OpenSource. If you would like to compile the program on a Linux system, the source, binaries, and documentation are located at https://github.com/g0orx

## 3. Discovery

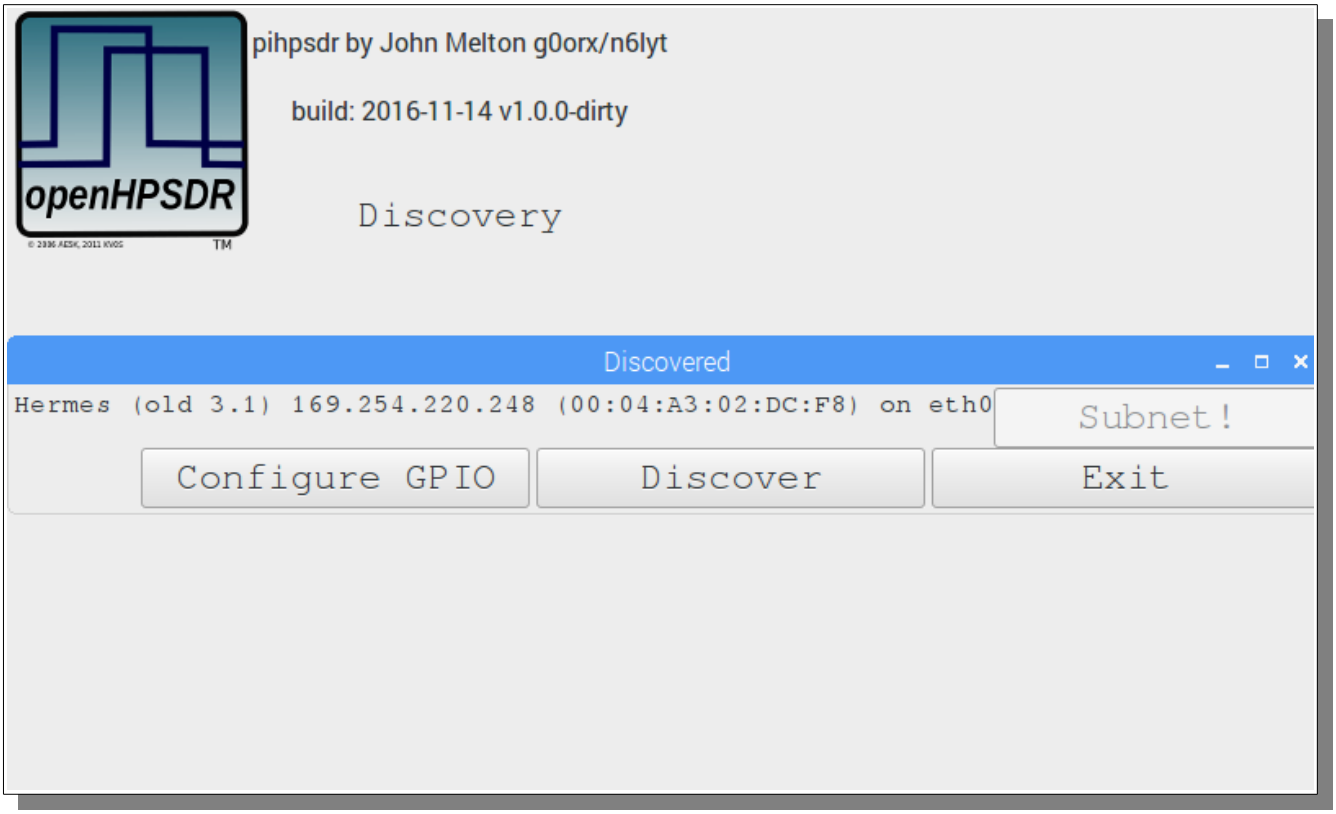
When first started, piHPSDR will try to discover all the HPSDR compatible radios on the network. It will look for devices running both the original (old) and the new Ethernet (new) protocol.

**If one or more Transceiver interfaces are found they will be identified by the device type,** the software version, the IP address and the MAC address of the device.



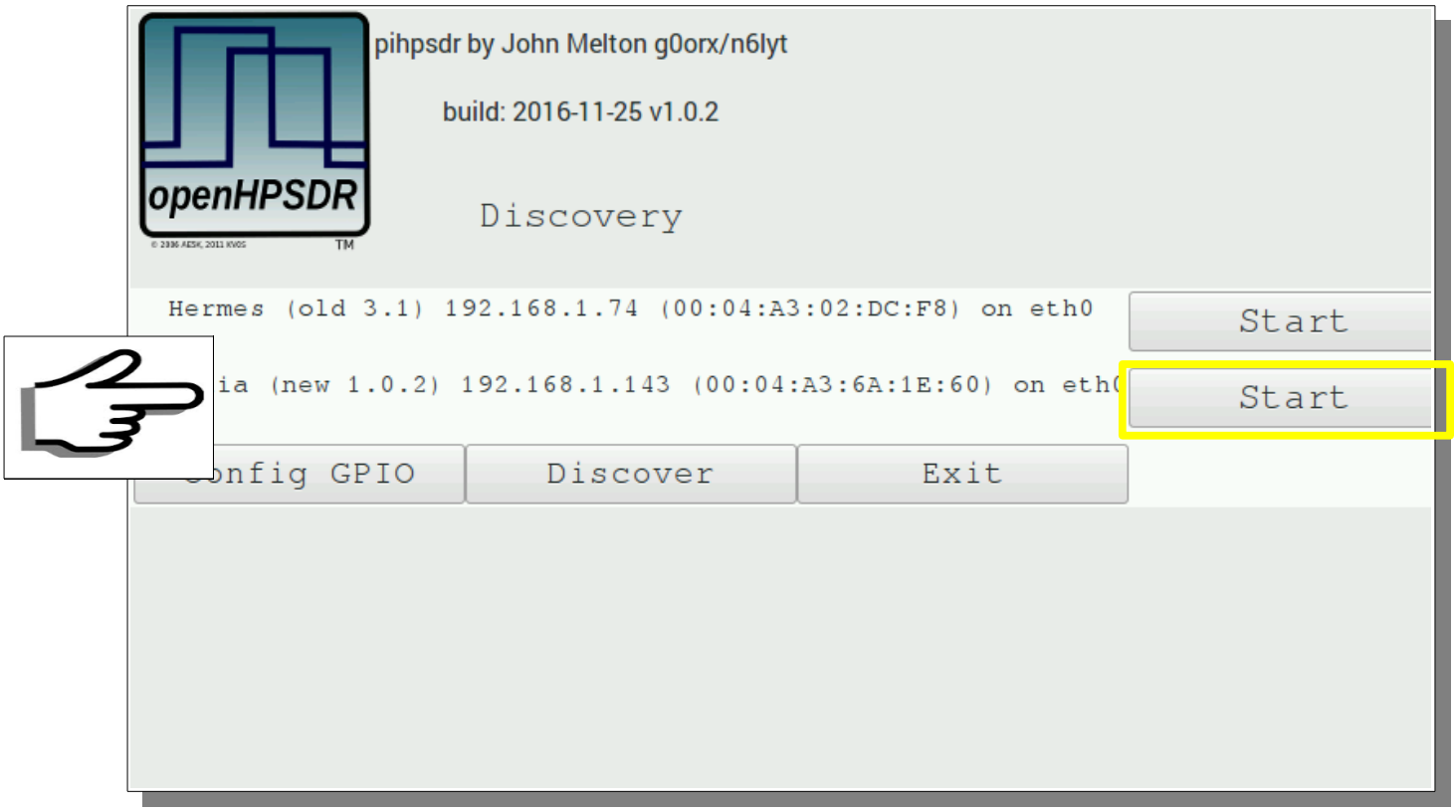

If no devices are found you should check connectivity between the radio and the network that piHPSDR is connected to. Tapping on OK will try discovery again. Tapping on Cancel will exit piHPSDR back to the Raspberry Pi desktop.

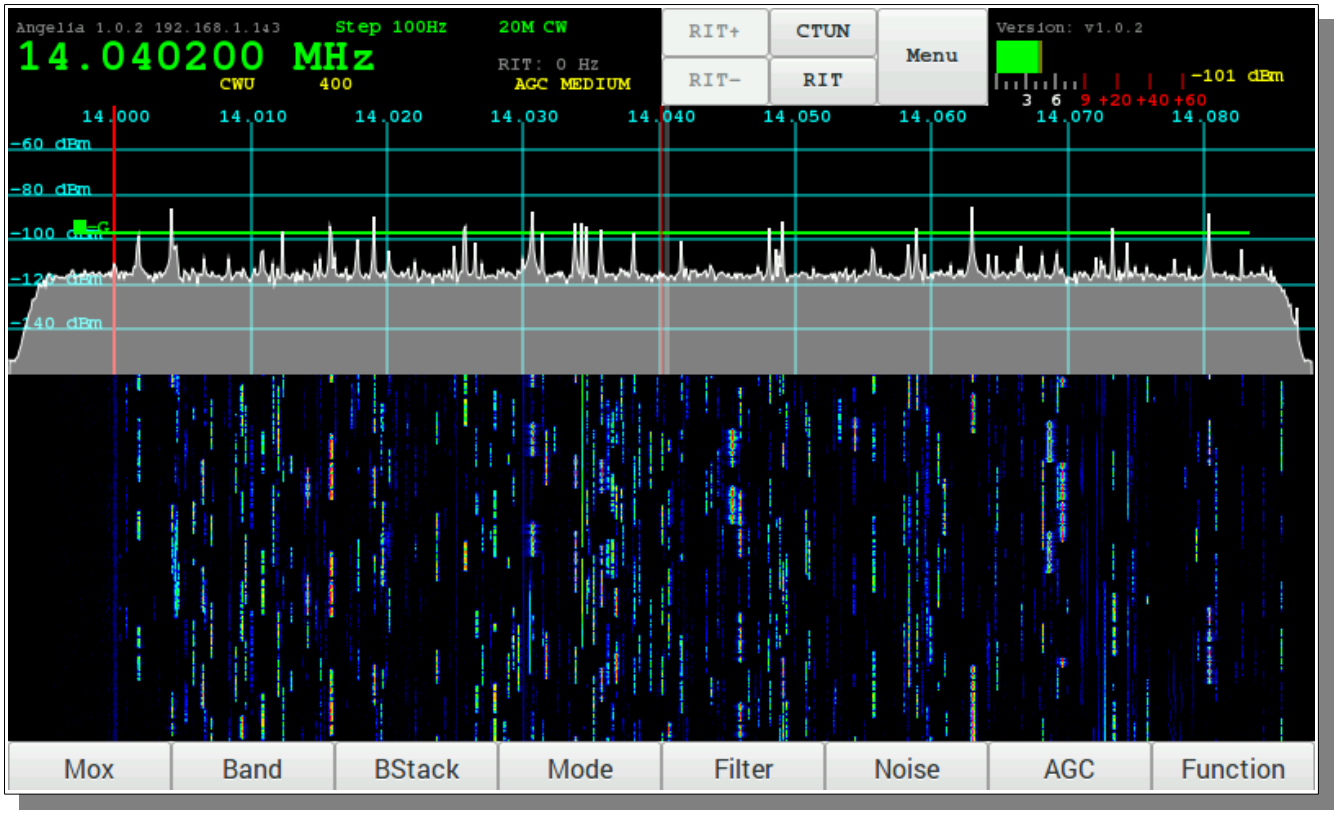
The discovery protocol will allow a device to see and respond even if they are not on the same subnet. If this is the case, the Start button will be disabled and the text replaced with **Subnet** The most common cause of this problem is usually that the radio has not been able to get a DHCP address and has defaulted to the Self Assigned IP Address, or the device has a static IP address that is not on the same subnet as the piHPSDR.



## Discovery → Start

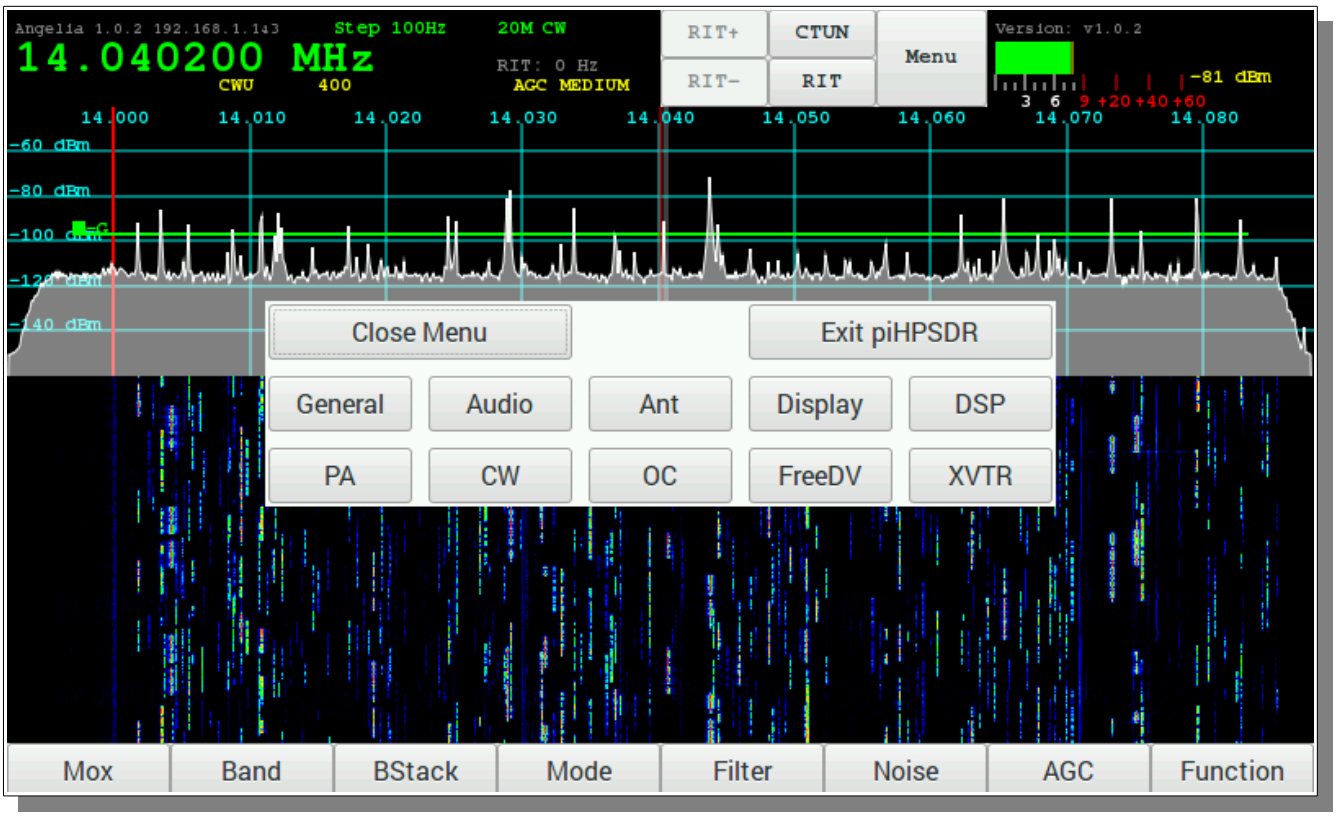
Tapping on the Start button will start the radio running.



## 4. Menu

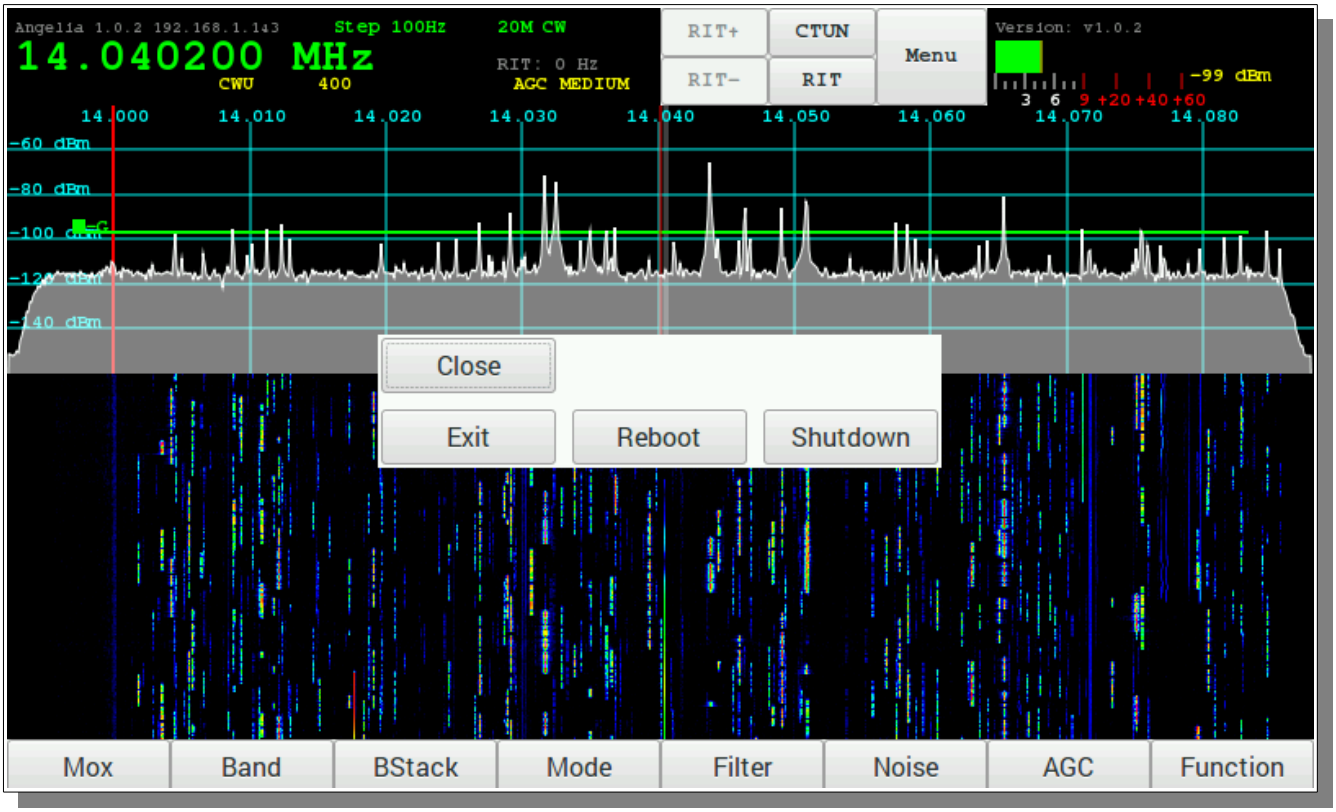
Tapping the Menu button brings up the menu dialog.




The menu dialog lets you select the different menu options for configuring the radio.

You can close the menu by tapping on the Close Menu button or by tapping on the Menu button again.

## Menu Exit piHPSDR

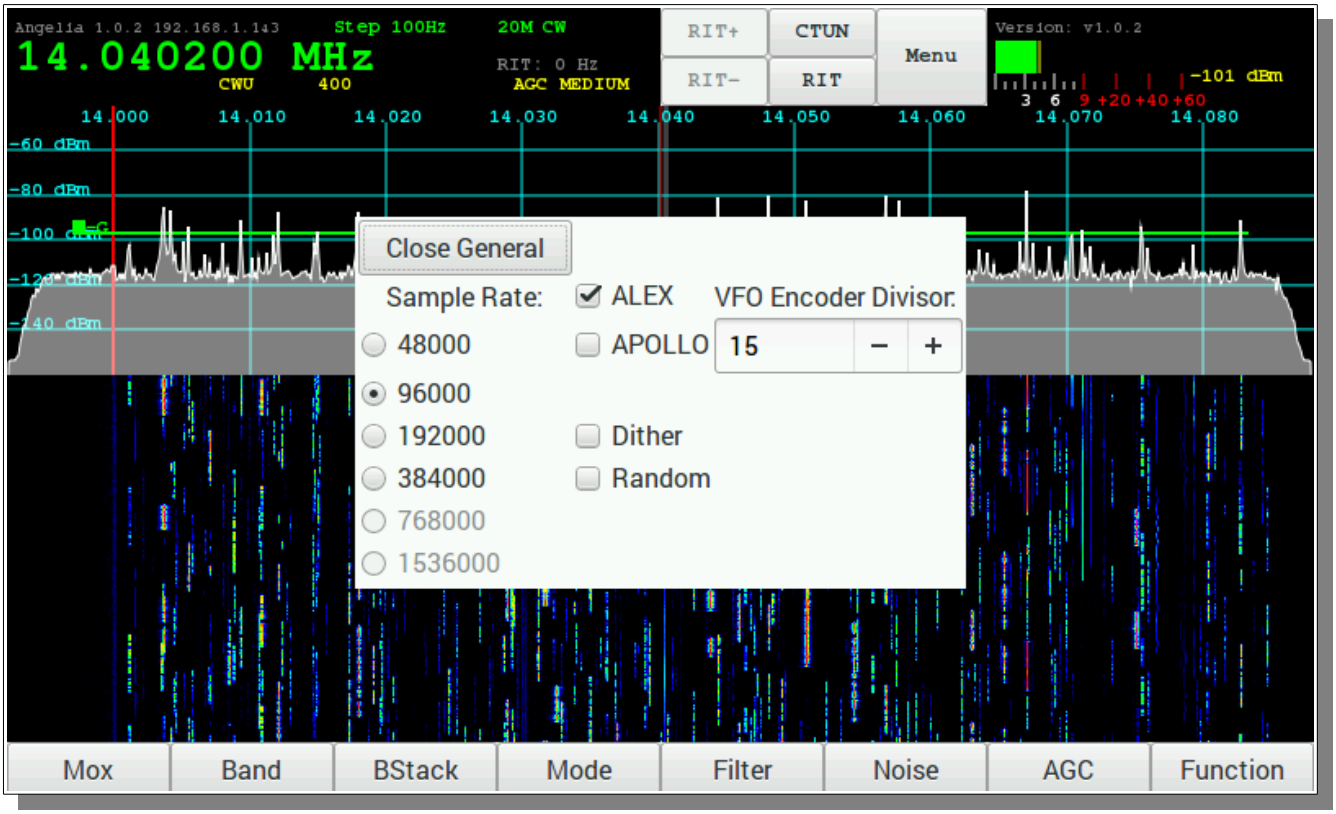



The Exit piHPSDR button will bring up another dialog allowing you to end the piHPSDR application in a number of ways:

- **Close –** close this dialog box
- **Exit** – exit piHPSDR back to the Raspberry Pi Desktop
- **Reboot** – reboot the Raspberry Pi
- **Shutdown** – Shutdown the Raspberry Pi

You can also tap the Menu button again to close the Exit Menu and bring up the Menu selection.

## Menu → General




- **Sample Rate** - The Sample Rate selection selects the width of the Panadapter displayed on the piHPSDR Controller screen. This is the rate which **pihpsdr** uses to decode a portion of the 30Mhz spectrum data from the Apache Transceiver.

  **Note: 768k and 1.536K are not available on Rpi.**

- **ALEX – selects** the automatic LPF/BPF filters selected inside each Apache Transceiver

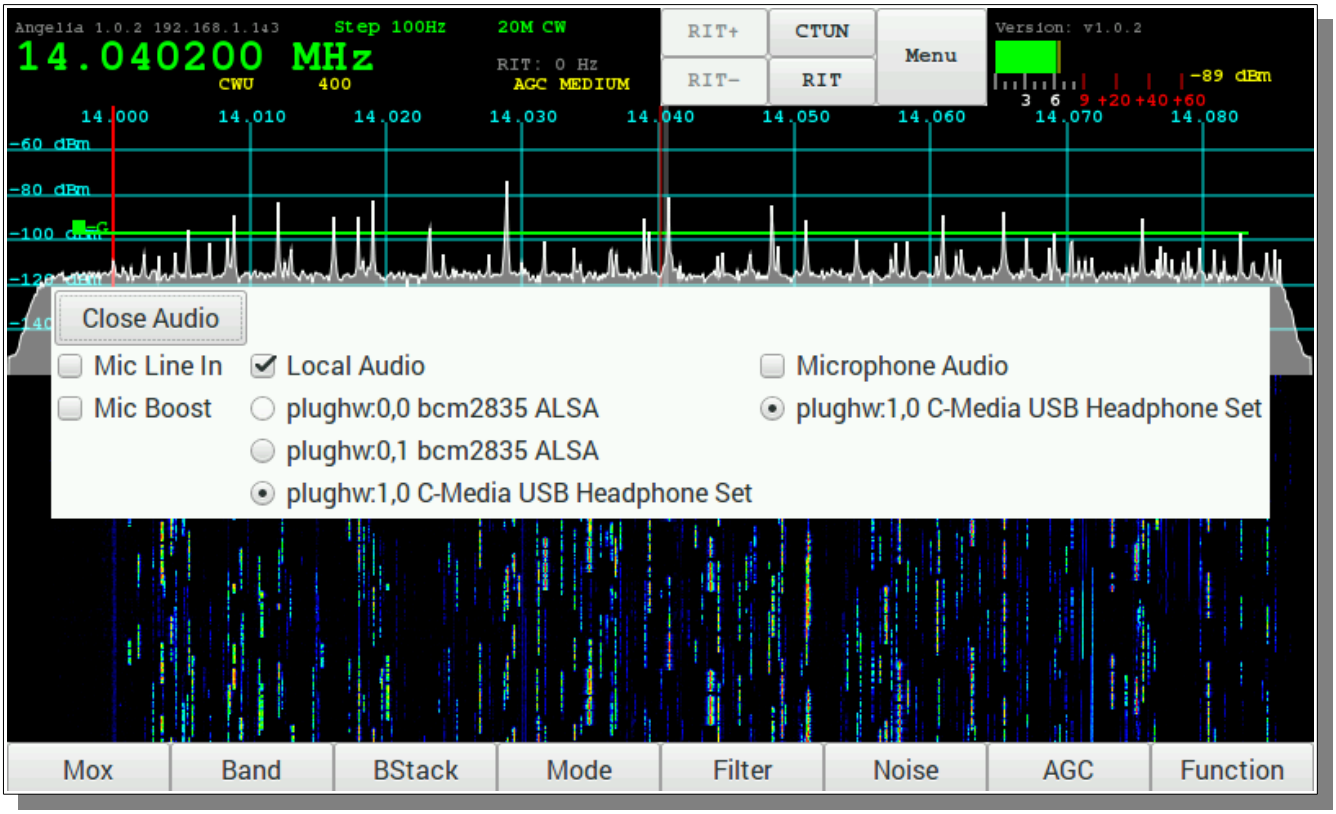
- **APOLLO –** refers to the Linear Amplifier and filters designed by Kjell LA2NI.

- **Dither** - Dither is a built-in electronic feature of the Linear Technologies LTC-2208 Analog to Digital Converter chip inside each Apache Software Defined Radio Transceiver.

- **Random** - Ramdom is a built-in electronic feature of the Linear Technologies LTC-2208 Analog to Digital Converter chip inside each Apache Software Defined Radio Transceiver.

- **VFO Encoder Divisor** - The VFO Encoder Divisor lets you adjust the sensitivity of the VFO encoder. The value specifies the number of encoder steps required to step the VFO up or down.

## Menu → Audio




- **Mic Line In** – When enabled the audio from the radio will come from Line In rather than the microphone socket.

- **Mic Boost** - is a 20db boost of the MIC connected to your Apache Transceiver.

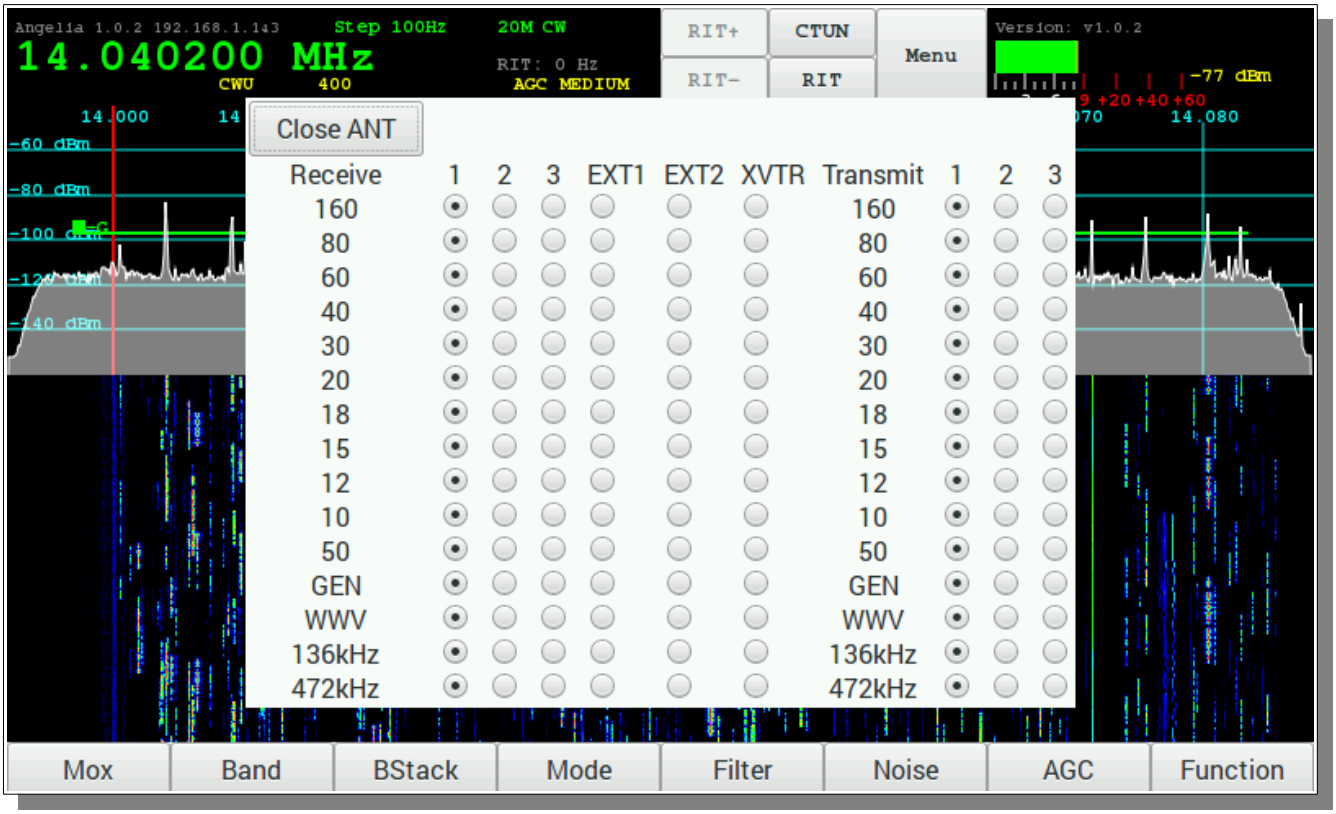
  The Raspberry Pi does not have any audio input capability. However, you can connect a USB sound dongle to the Raspberry Pi which will enable additional audio input and output.

- **Local Audio** – When enabled selects the selected audio interface on the Raspberry Pi to output sound.

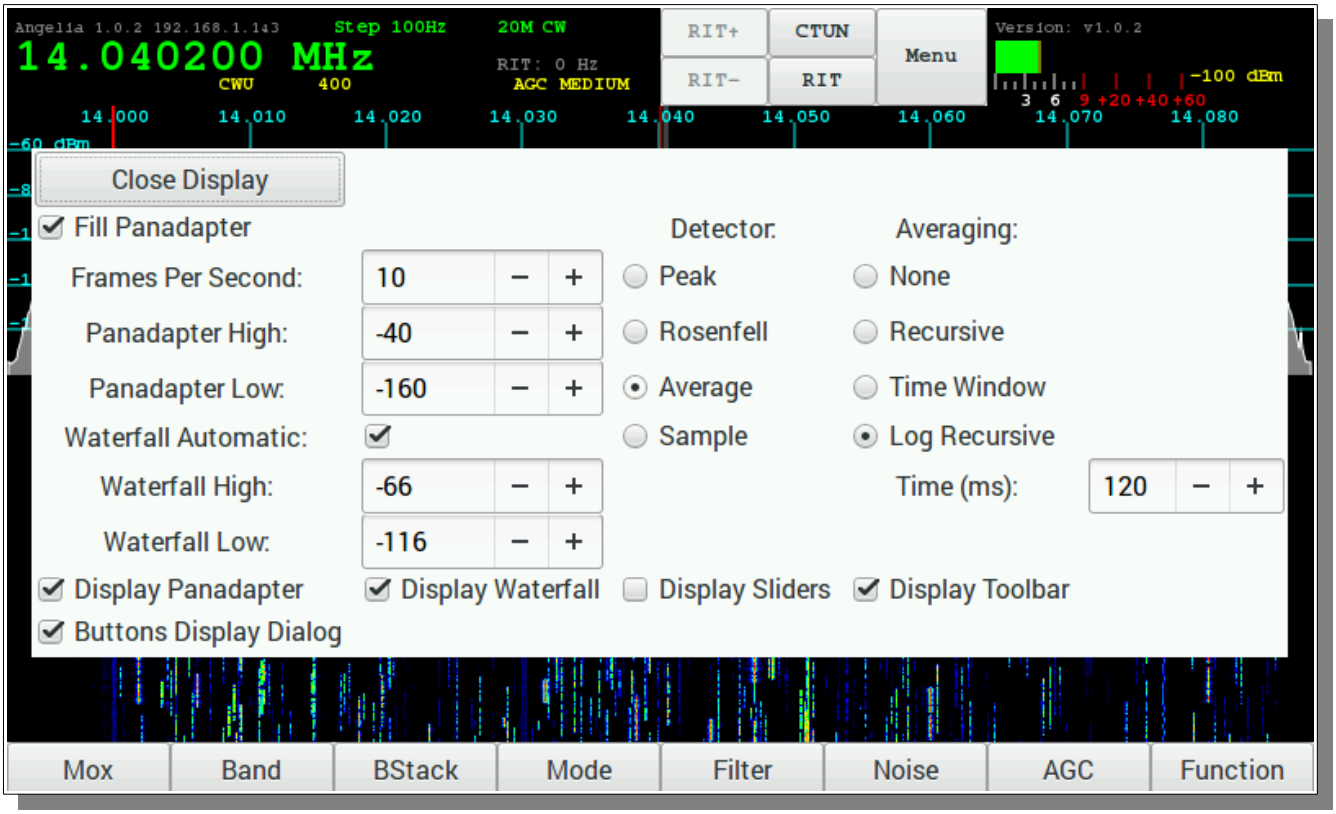
- **Microphone Audio** – When enabled selects the selected audio interface on the Raspberry Pi for the microphone input.

## Menu → ANT

The Ant menu selects which antenna is used for receive and transmit on each band.



## Menu → Display




- **Fill Panadapter** – when enabled the panadapter graph will be filled. When not enabled it will be drawn as a line.

- **Frames Per Second** – Update rate of Panadapter and Waterfall

- **Panadapter High** - Maximum signal level displayed in Panadapter

- **Panadapter Low** – Minimum signal level displayed in Panadapter

- **Waterfall Automatic** – When enabled the Waterfall High and Waterfall Low are adjusted automatically.

- **Waterfall High** – When Automatic is not set this will set the maximum signal used in the waterfall.

- **Waterfall Low** - When Automatic is not set this will set the minimum signal used in the waterfall.

- **Detector** – Selects Peak, Rosenfell, Average or Sample for the Panadapter display.

- **Averaging** – Selects the method for averaging the Panadapter display.

- **Display Panadapter** – when selected the Panadapter is displayed on the main screen.

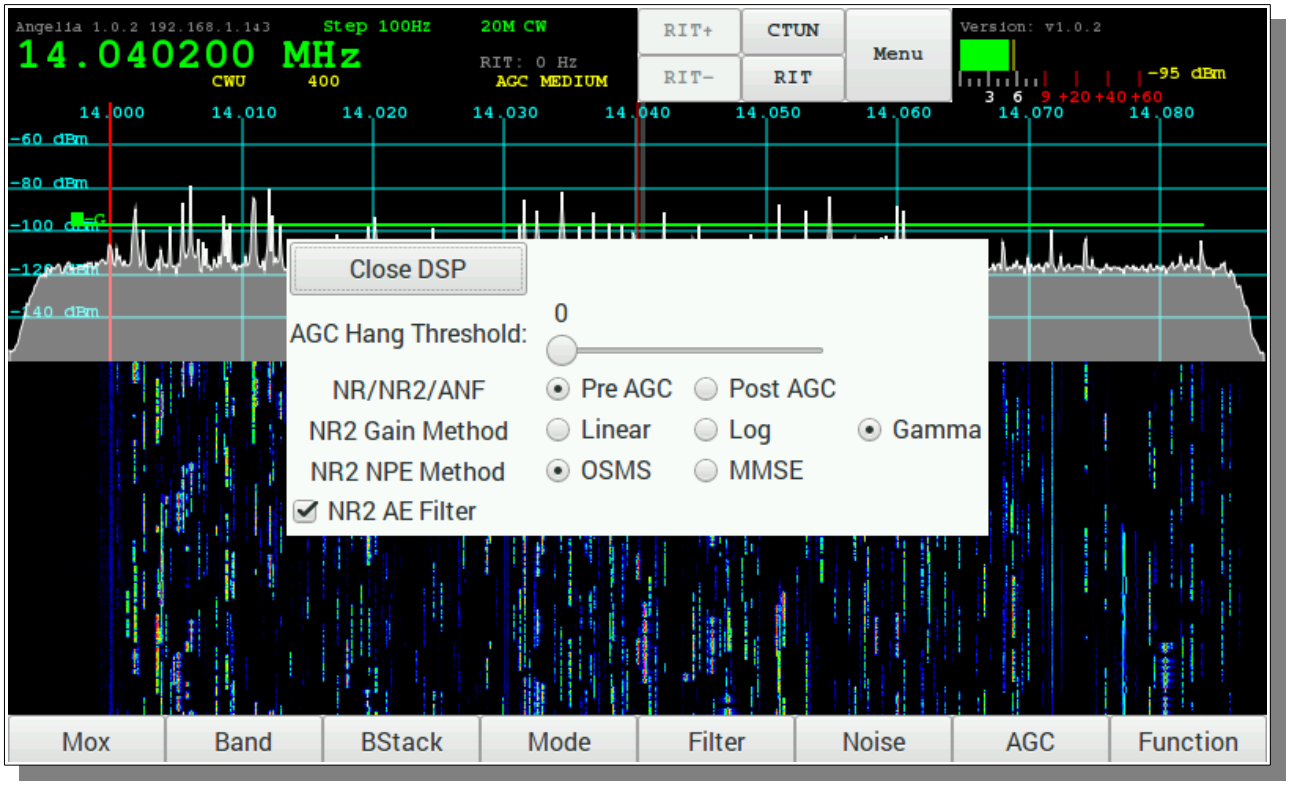
- **Display Waterfall** – when selected the Waterfall display is displayed on the main screen.

- **Display Sliders** – when selected the slider controls are displayed on the main screen.

- **Display Toolbar** – when selected the toolbar is displayed on the main screen.

- **Buttons Display Dialog** – when selected the presses on the toolbar or the physical buttons displays a dialog to select the options. When not set the button steps forward or reverse (if Function is enabled) through the options.

## Menu → DSP




The DSP has options for the DSP functions.

- **AGC Hang Threshold** – sets the HangThreshold for the AGC when enabled.

- **NR/NR2/ANF** – selects where in the DSP processing the noise reduction functions are performed/ The default id Pre AGC processing.

  Pre AGC – perform noise reduction pre AGC

  Post AGC – perform noise reduction post AGC

- **NR2 Gain Method** – selects the method used for the gain processing. The default is Gamma.

  Linear - Gaussian speech distribution, linear amplitude scale

  Log - Gaussian speech distribution, log amplitude scale

  Gamma - Gamma speech distribution

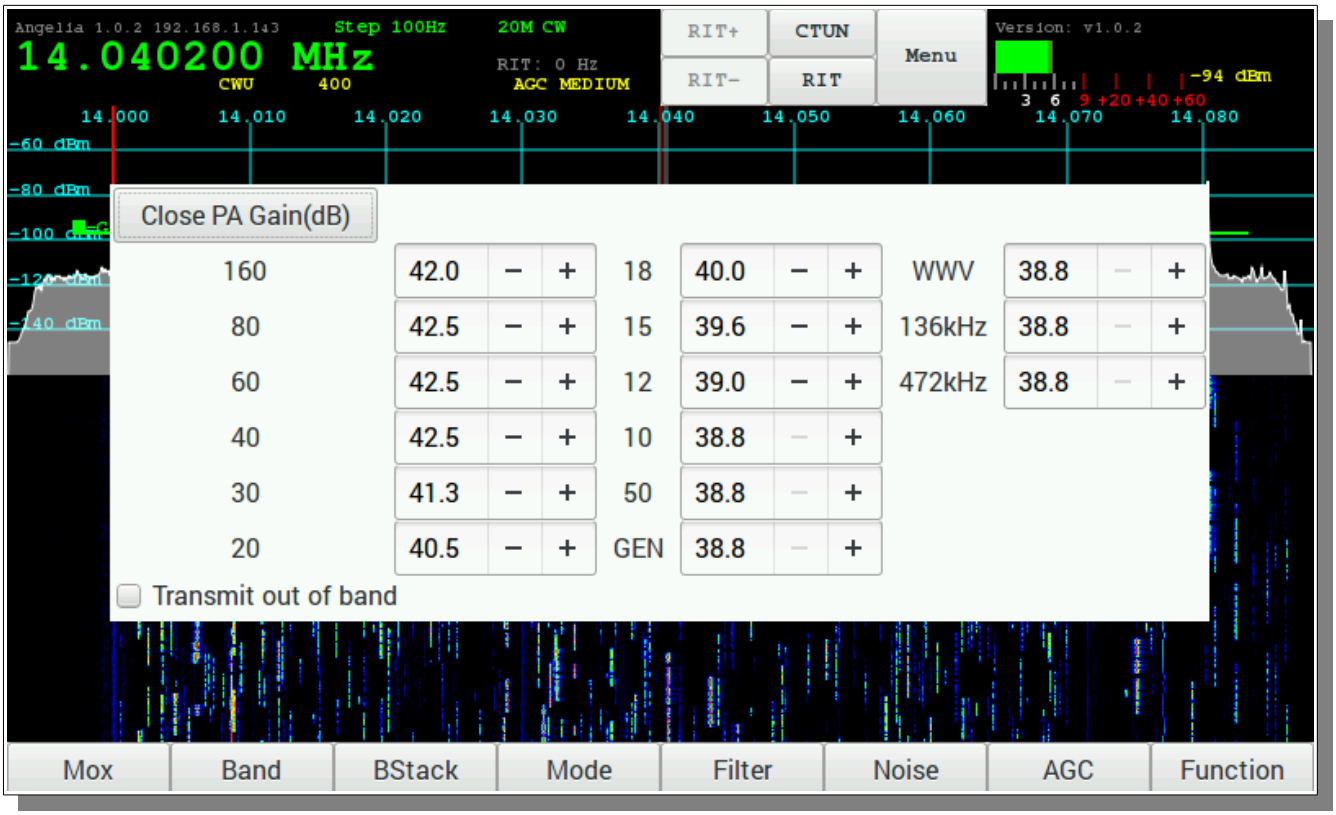
- **NR2 NPE Method** – selects the Noise-Power-Estimation method. The default is OSMS.

  OSMS - Optimal Smoothing Minimum Statistics

  MMSE - Minimum Mean -Square Error

- **NR2 AE Filter** - Enable Artifact elimination. Default enabled.

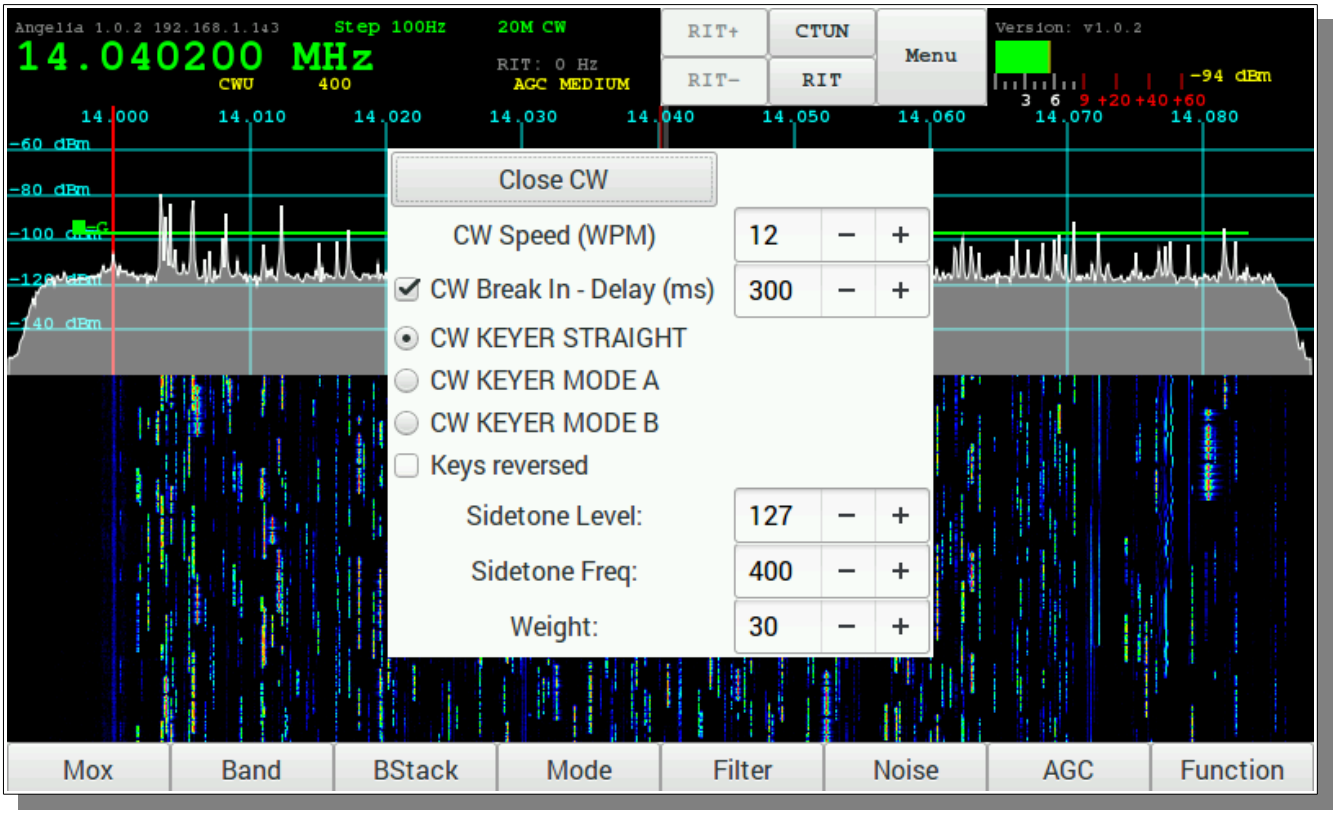
## Menu → PA Gain by Band




Sets the gain for each band to set the maximum drive level. Note that as the setting are increased the drive level is decreased.

- **Transmit out of band** – enables transmit outside of ham bands. Default disabled.

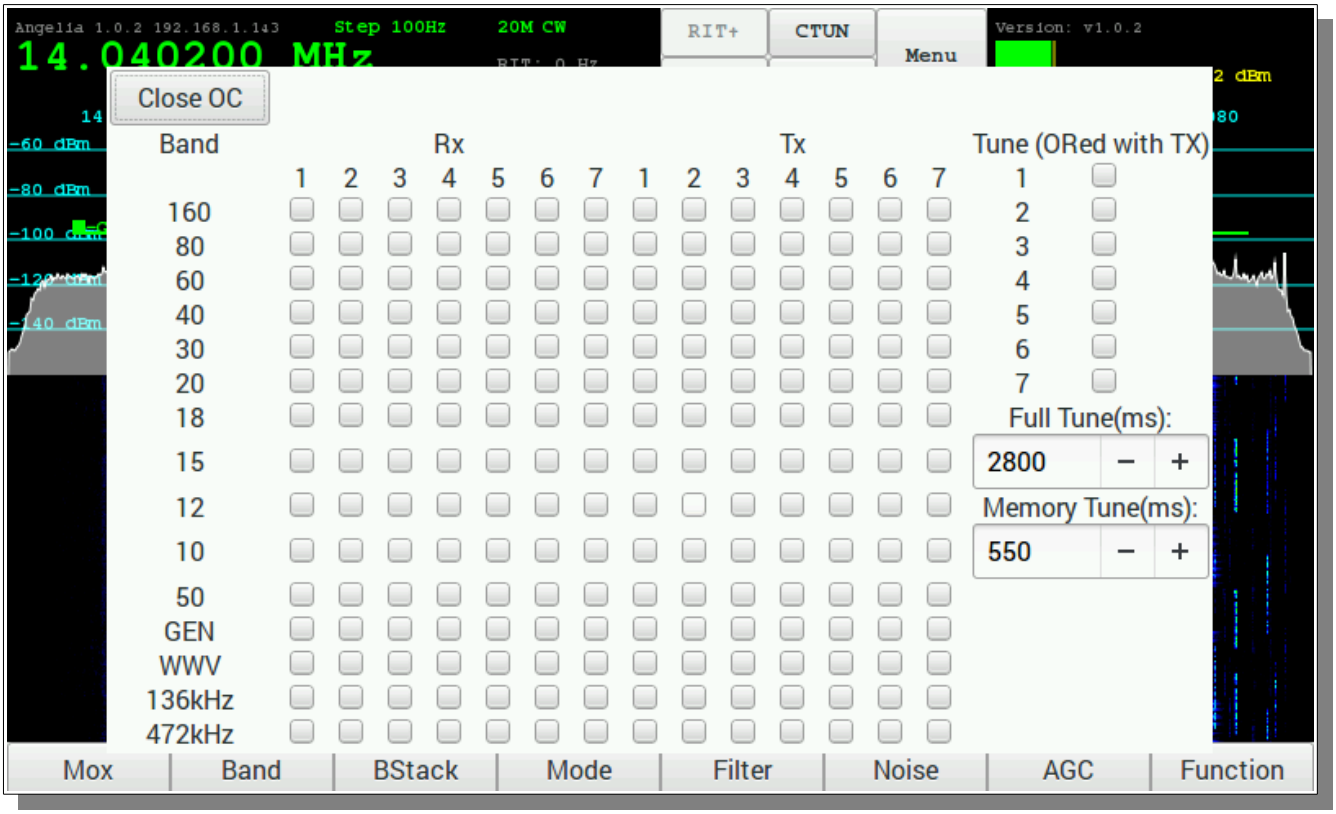
## Menu → CW




Controls the firmware CW Keyer.

- CW Speed (WPM) – sets speed of dot/dash generator when in Mode A or Mode B.
- CW Break In – when enabled sets the delay time in milliseconds to switching to receive.
- CW KEYER STRAIGHT – selects the key connected will be a straight key.
- CW KEYER MODE A – selects a paddle key running in Mode A
- CW KEYER MODE B – selects a paddle key running in Mode B
- Keys reversed – when enabled the dot/dash paddles are reversed.
- Sidetone Level – sets the audio level of the sidetone.
- Sidetone Freq – set the frequency of the sidetone.
- Weight – sets the dot/dash weighting.

## Menu → OC  Open Collector Aux I/O connectors

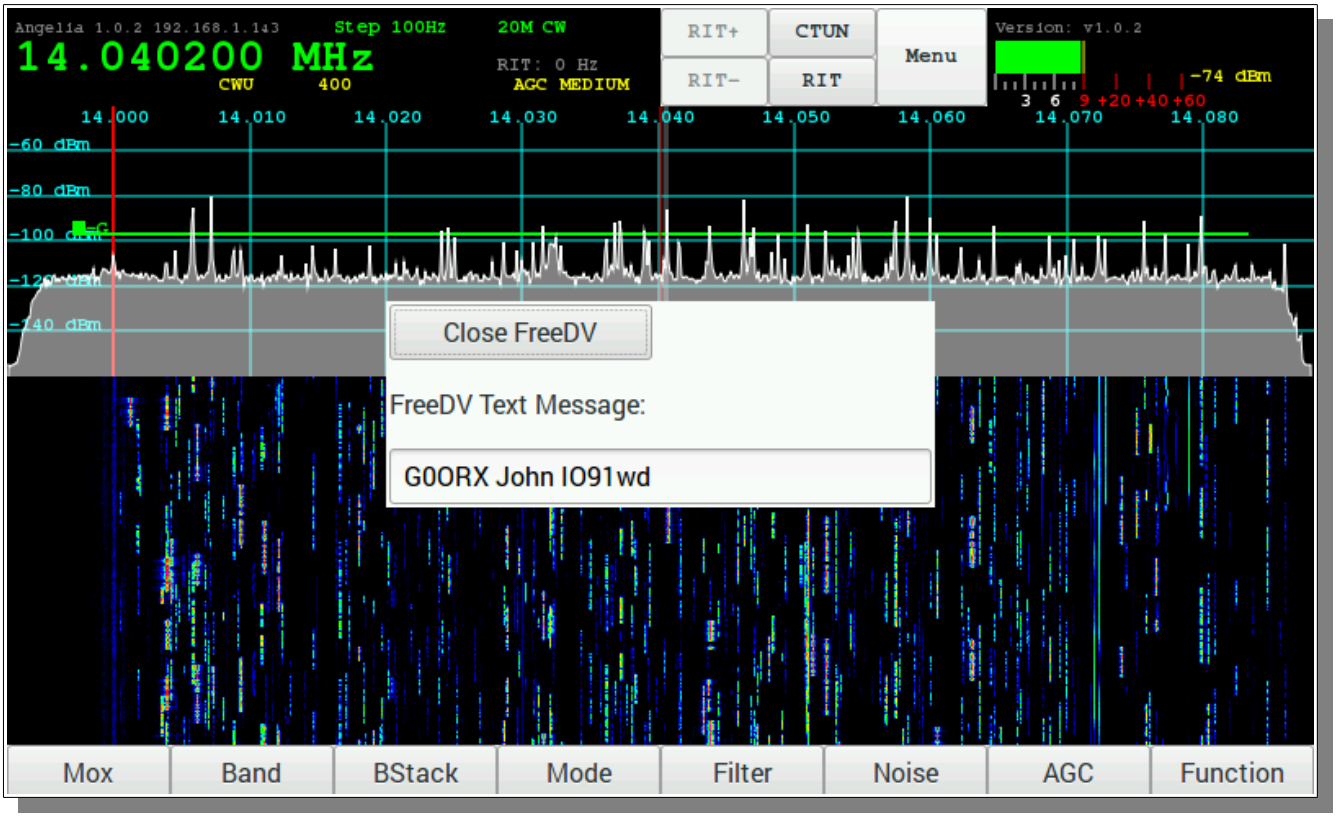



Configure the Open Collector outputs for each band.  These can be used to control an external filter when not using the Apache Labs filters or the Alex filters.  The default is none are enabled.

When an external ATU is used the Tune option can be used to signal to the ATU to start its tune function. The default is none are enabled.

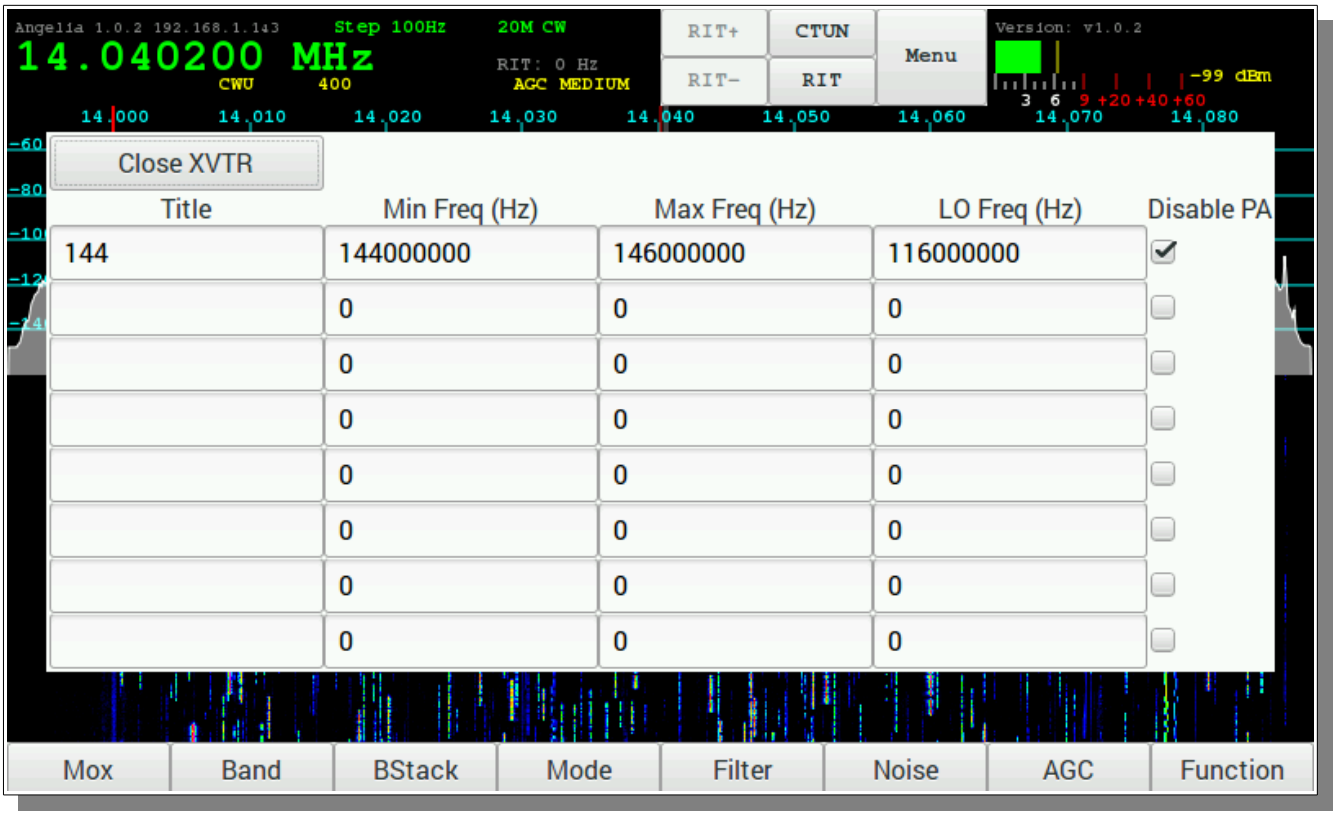
- Tune – configure an OC to be turned on when Tune is enabled.  Currently only the Full Tune is implemented.
- Full Tune – specifies the time the OC is enabled when the Tuning.

## MENU → FreeDV




Sets the FreeDV Text Message that is send along with the digitized voice.

## Menu → XVTR




Configure up to 8 transverters.

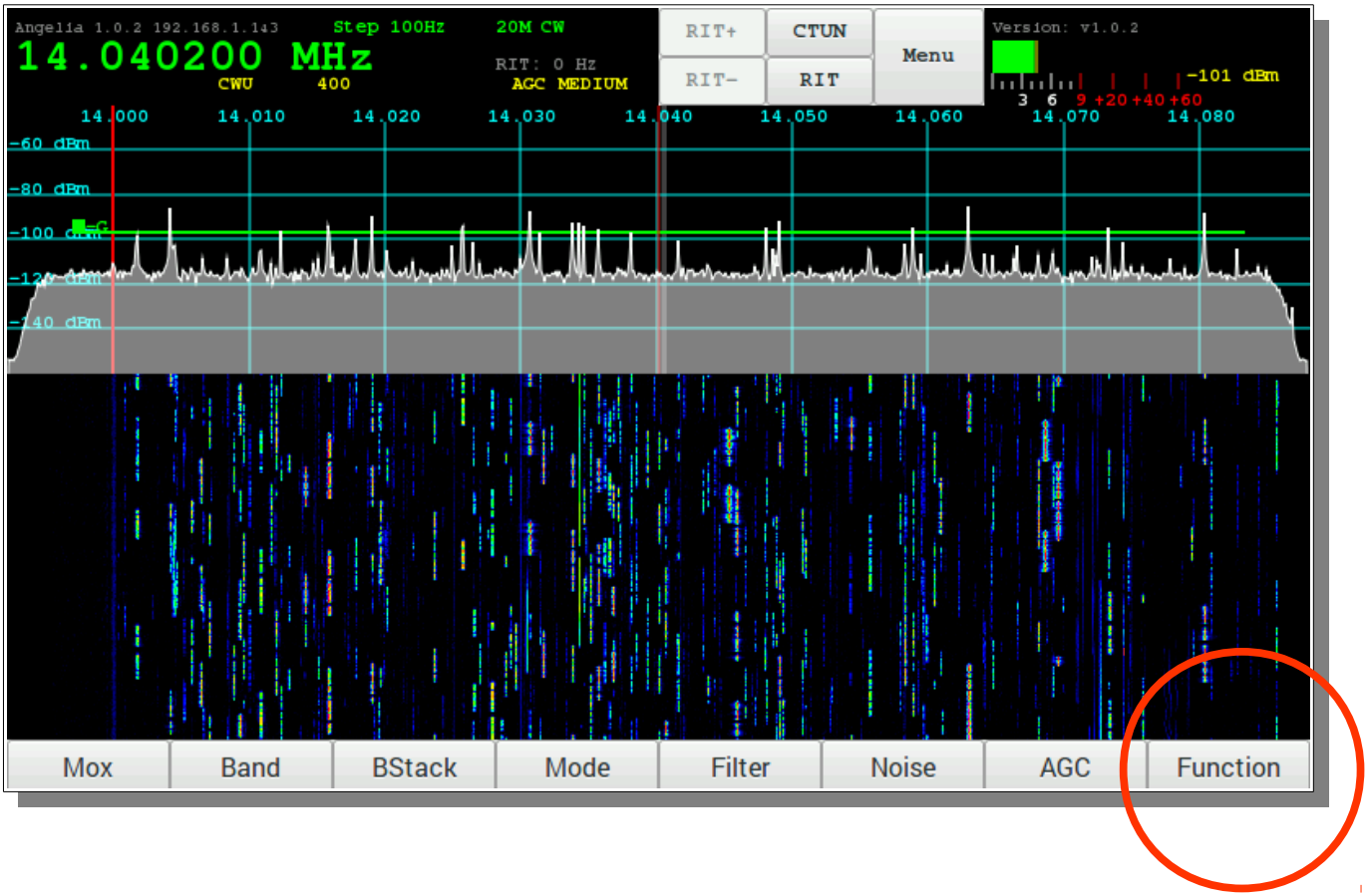
- **Title** – the name as it appears in the Band, Ant and OC menus.
- **Min Feq** – The minimum frequency in Hz.
- **Max Feq** – The maximum frequency in Hz.
- **LO Feq** – The Local Oscillator frequency in Hz.
- **Disable PA** – When checked the PA will be disabled on transmit.

Note that the frequency the radio is tuned to is the selected frequency minus the LO frequency. In the example above the 144MHz to 146MHz transverter frequency will be tuned to 28MHz to 30MHz on the radio.

When one or more transverters are configured they will appear in the band selection dialog and also in the ANT selection menu and the PA Gain menu.

## 5. Screen Controls and Push Buttons

### Function




The function touch screen button or physical button enable the optional functions of some of the buttons and encoders.

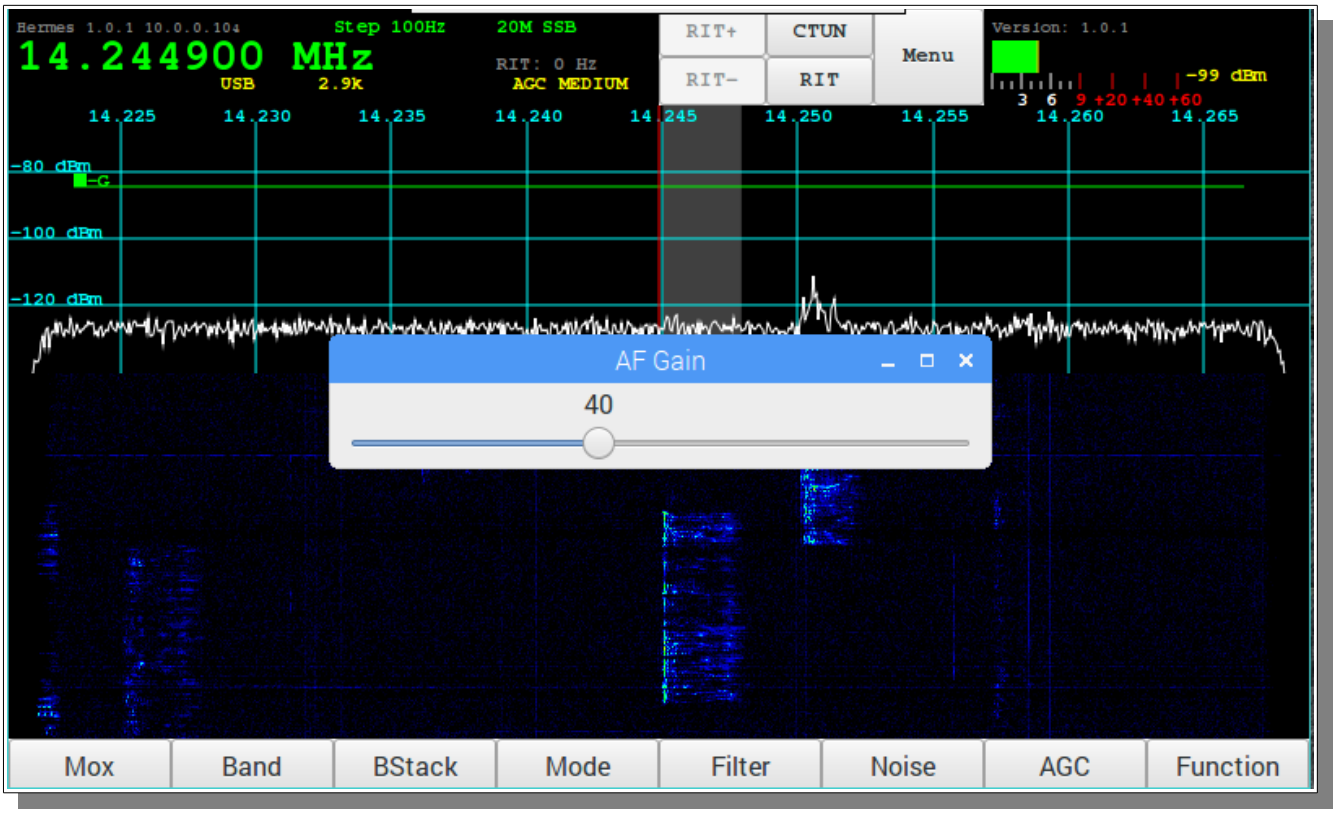
When enabled the **Function** text appears in the VFO panel and the Mox button changes to Tune.

In addition the 3 encoders change function:

- The **E1** encoders selects **AF Gain** or **Mic Gain**
- The **E2** encoder selects **PA Drive Level** or **Tune Drive Level**
- The **E3** encoder selects **AGC Gain** or **Rx Attenuation**

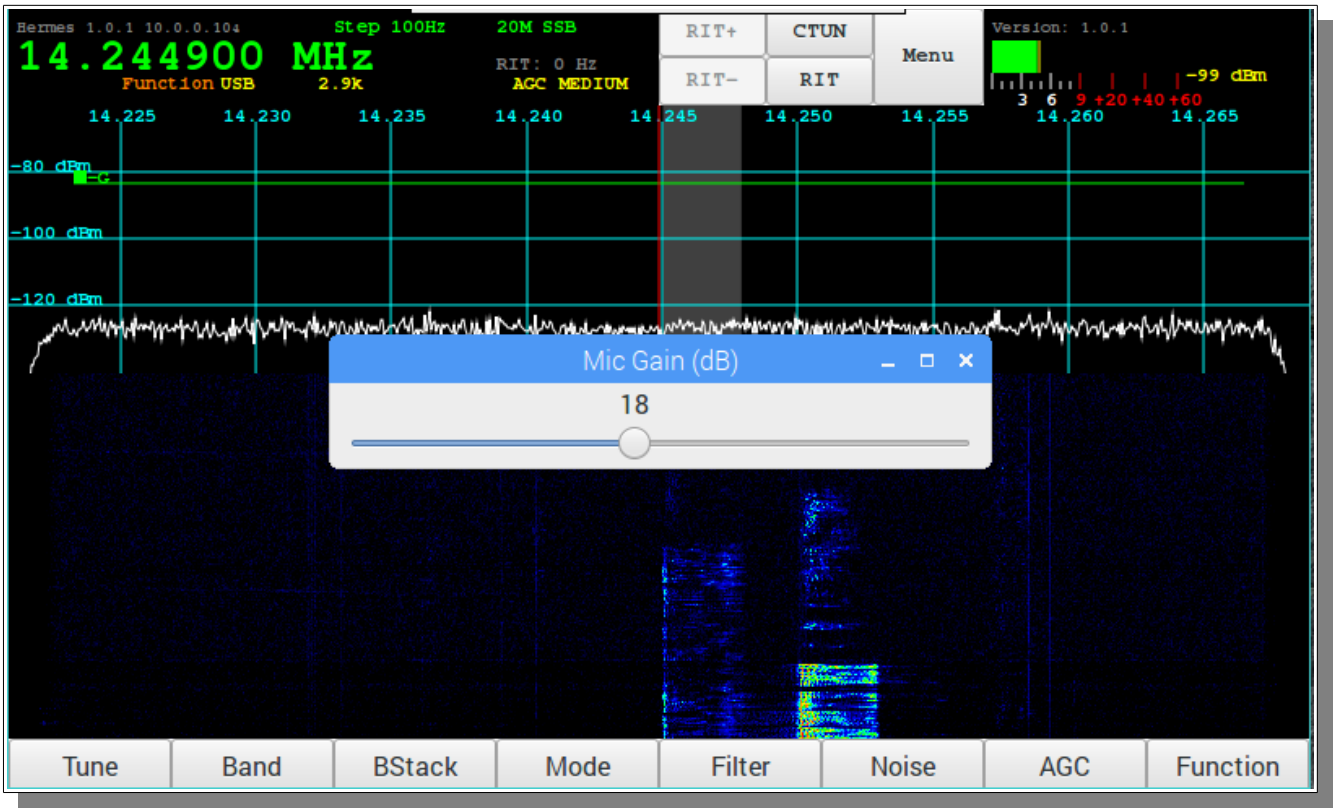
## AF Gain Slider

AF Gain Slider is selected using Function key and then rotating E1



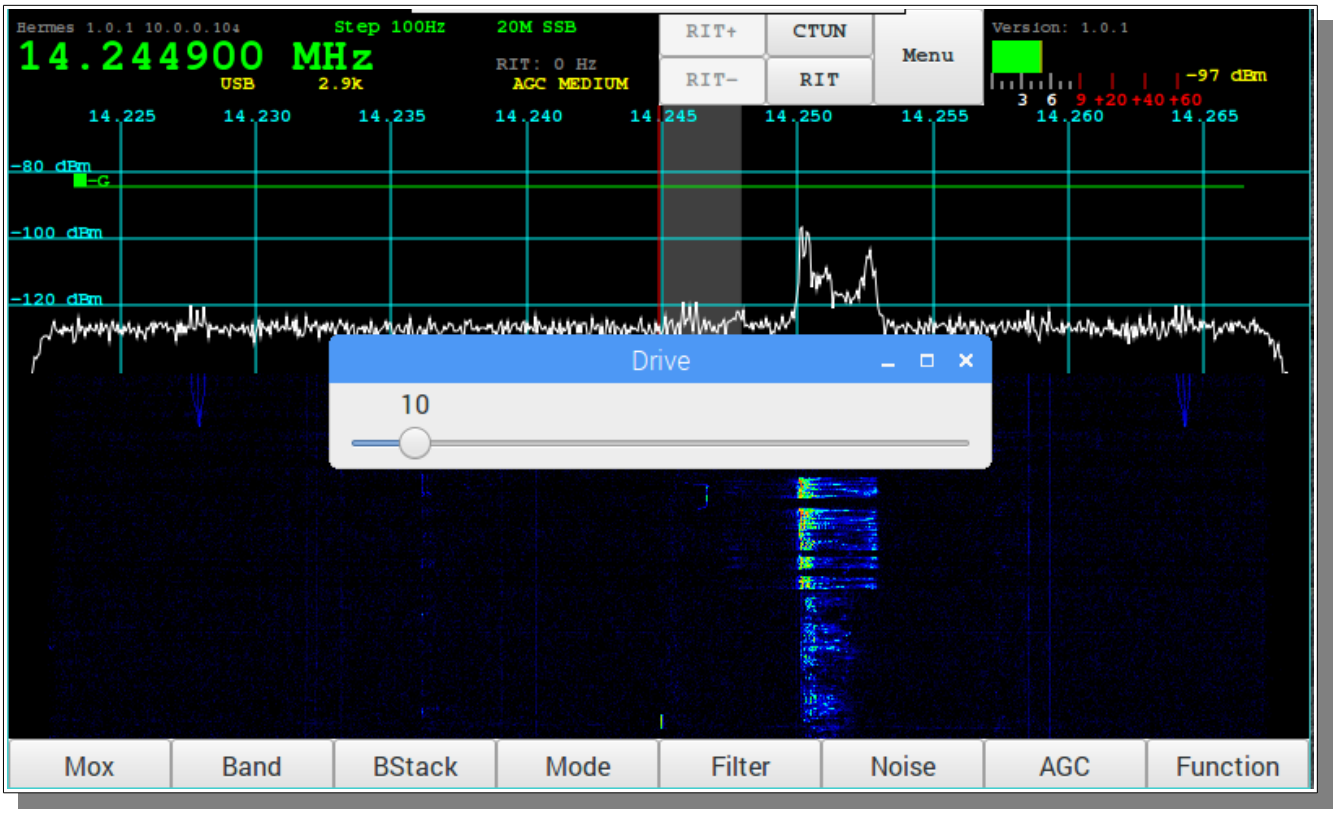

## MIC Gain Slider

MIC Gain Slider is selected using Function key and then rotating E1



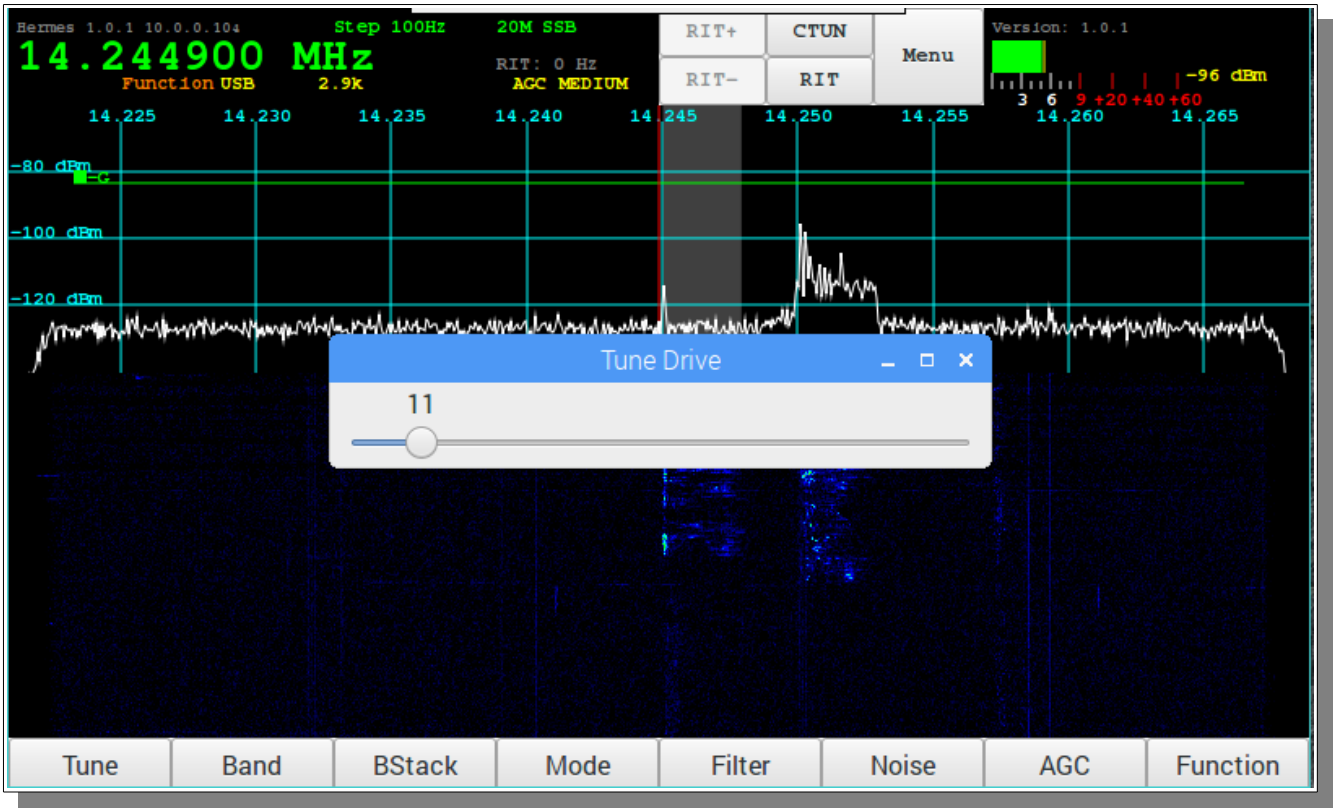
## PA Drive Slider

PA Drive Slider is selected using the Function Key and then rotating E2



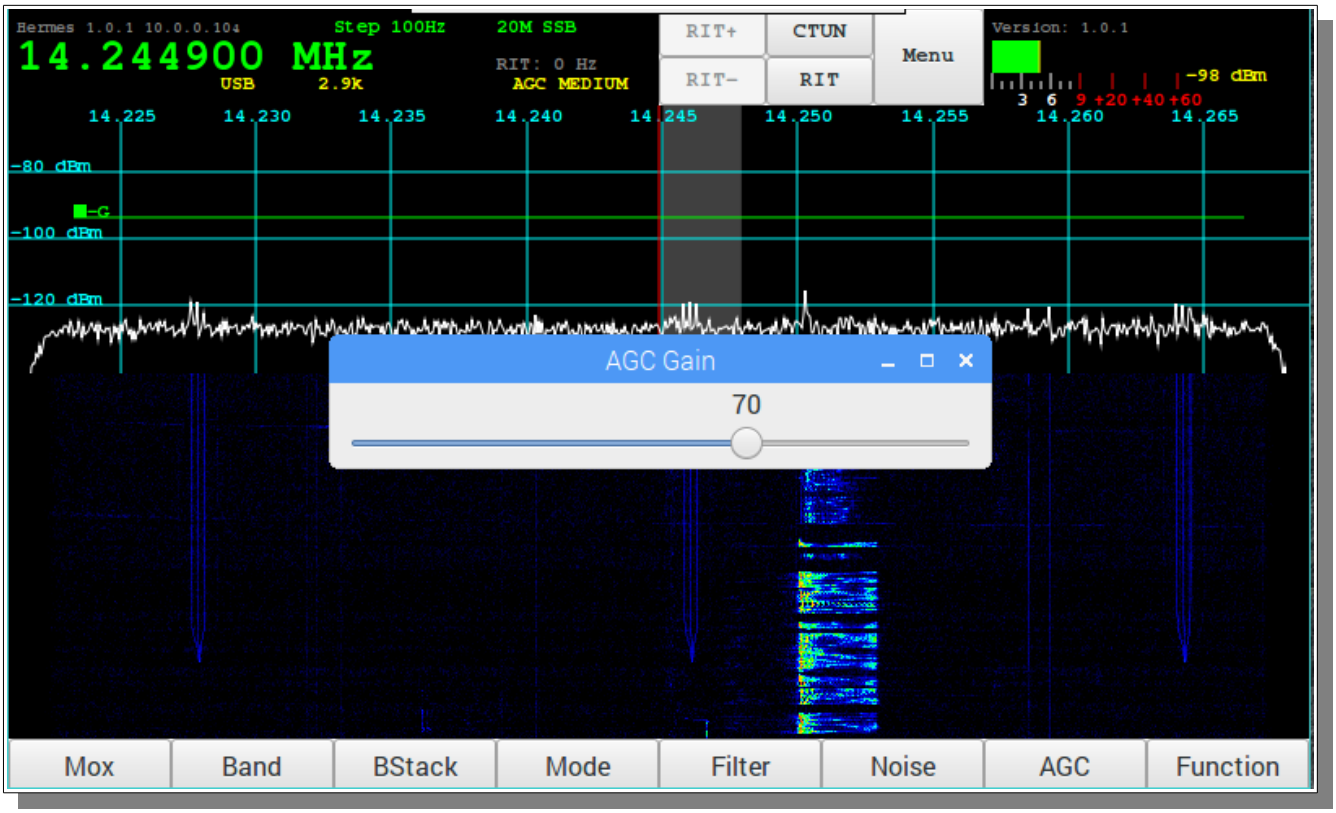

## TUNE Slider

Tune Drive Slider is selected using the Function Key and then rotating E2



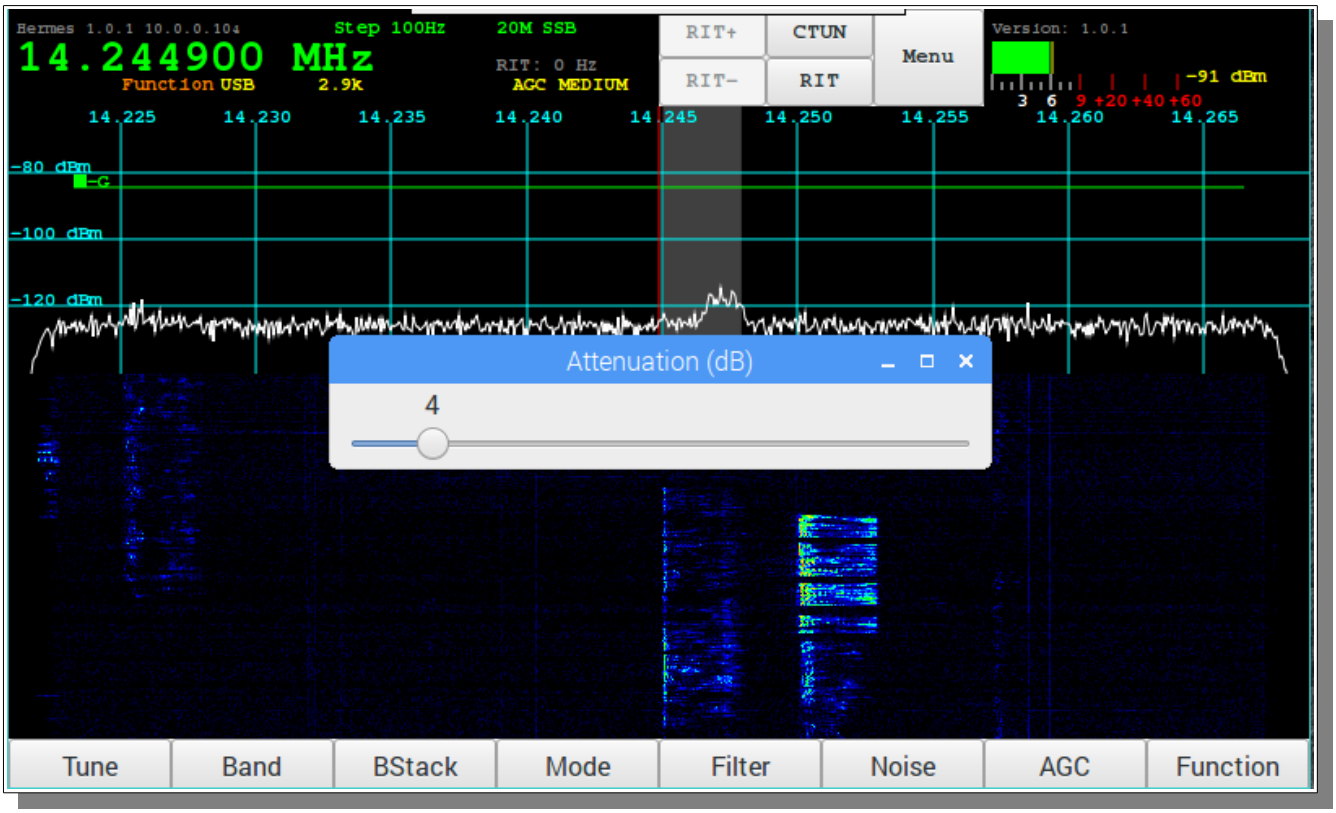
## AGC Gain Slider

AGC Gain Slider is selected using the Function Key and then rotating E3

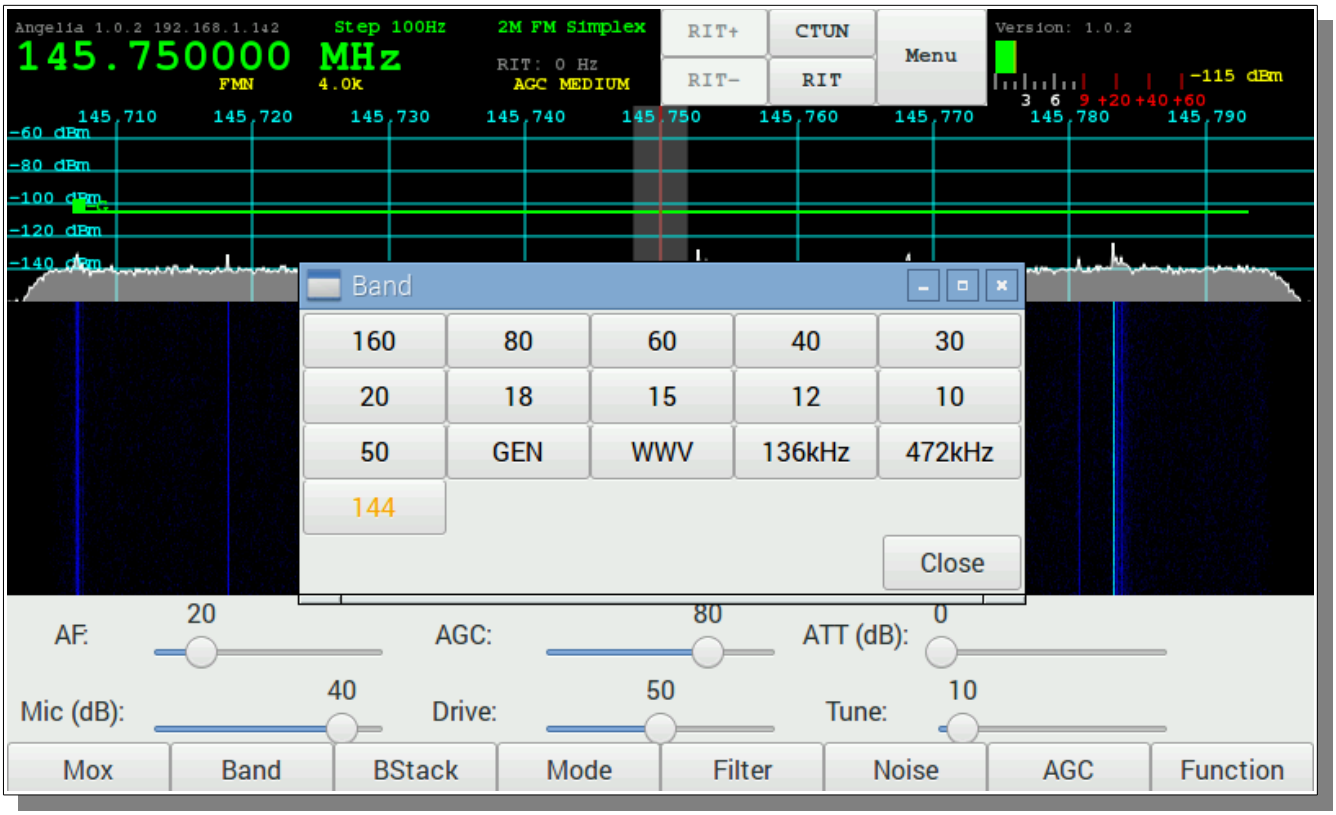



## Receive Attenuation Slider

Receive Attenuation Slider is selected using the Function Key and then rotating E3



## Band Selection

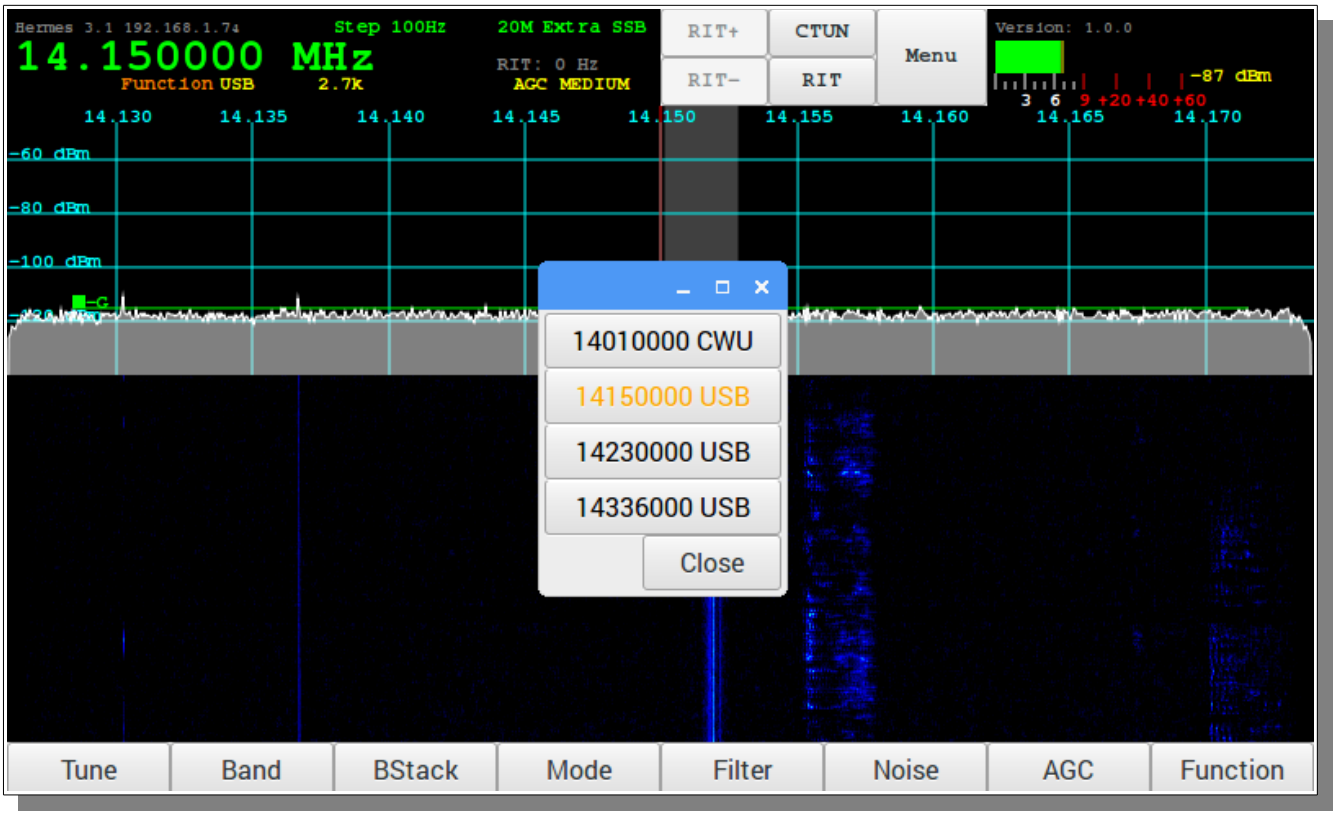



Tapping the Band touch screen button or pressing the physical button will display the list of bands with the current band highlighted in orange.

Tapping the current band will step through the band stack for that band. Tapping on any other band will switch to that band and the last used band stack entry.

Tapping on Close will close the menu.

## Band Stack

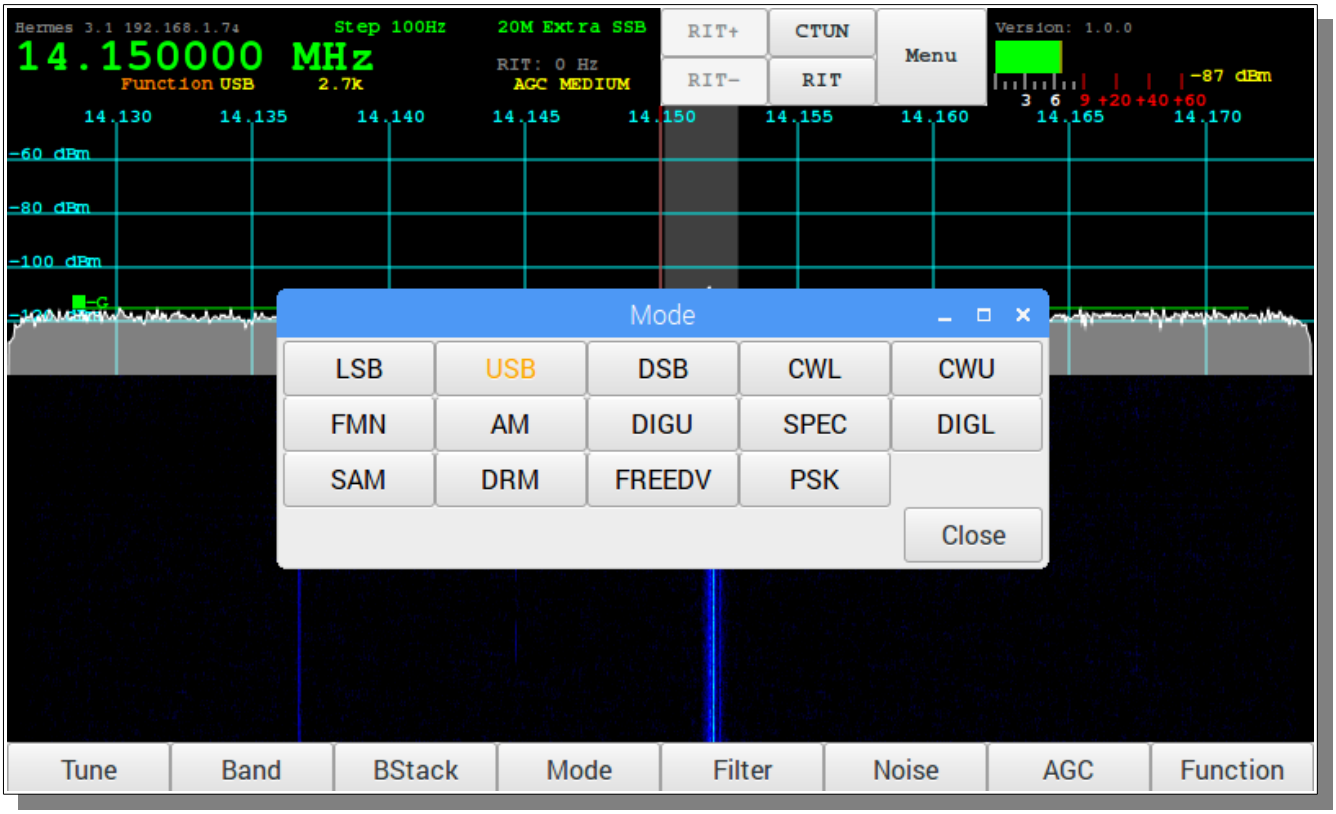



Tapping the BStack touch screen button or the physical button will display the list of Band Stack entries for the current band with the current selection highlighted in orange.

Tapping on an entry will switch to that entry.

Tapping on close will Close the menu.

## Mode

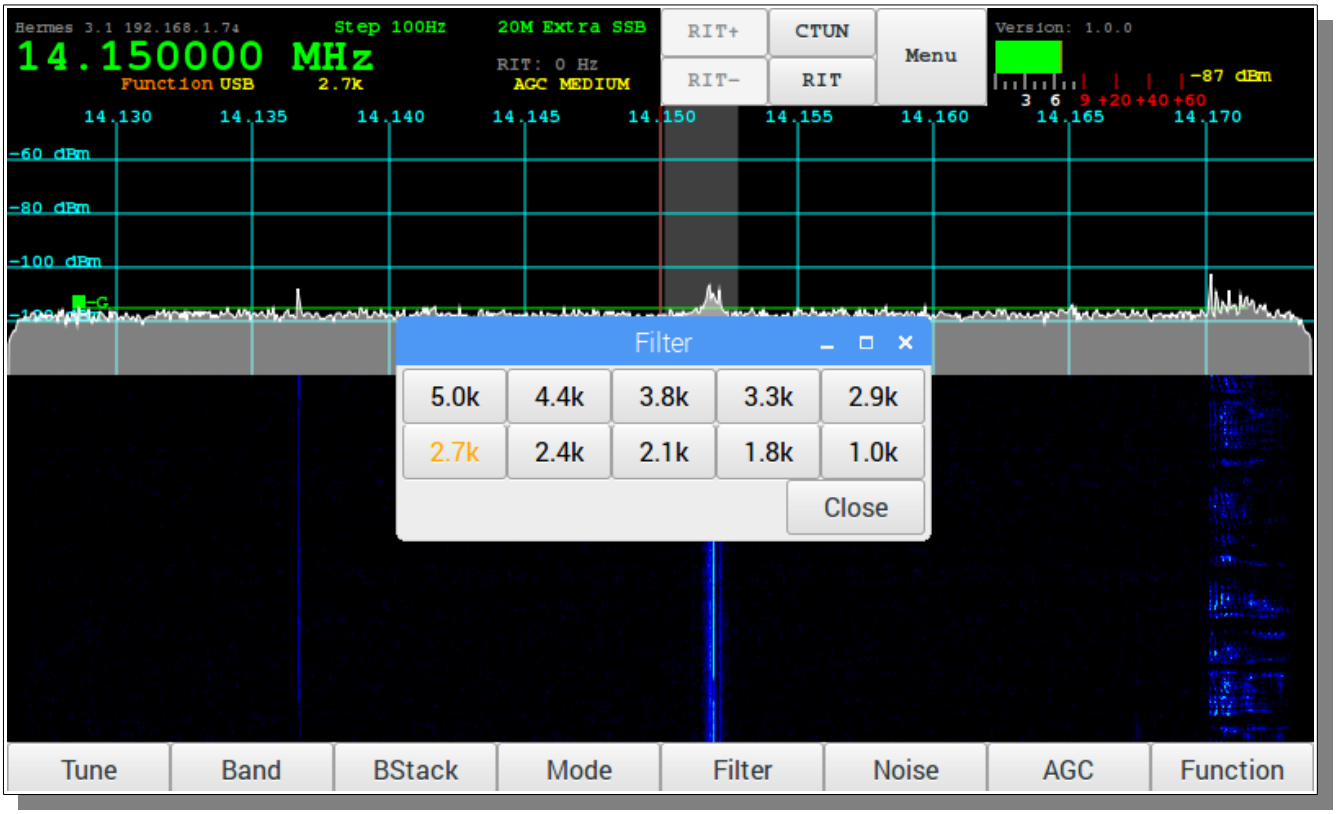



Tapping the Mode touch screen button or pressing the physical button will display the list of modes with the current mode highlighted in orange.

Tapping a mode will switch to the selected mode.

Tapping on Close will close the menu.

## Filter

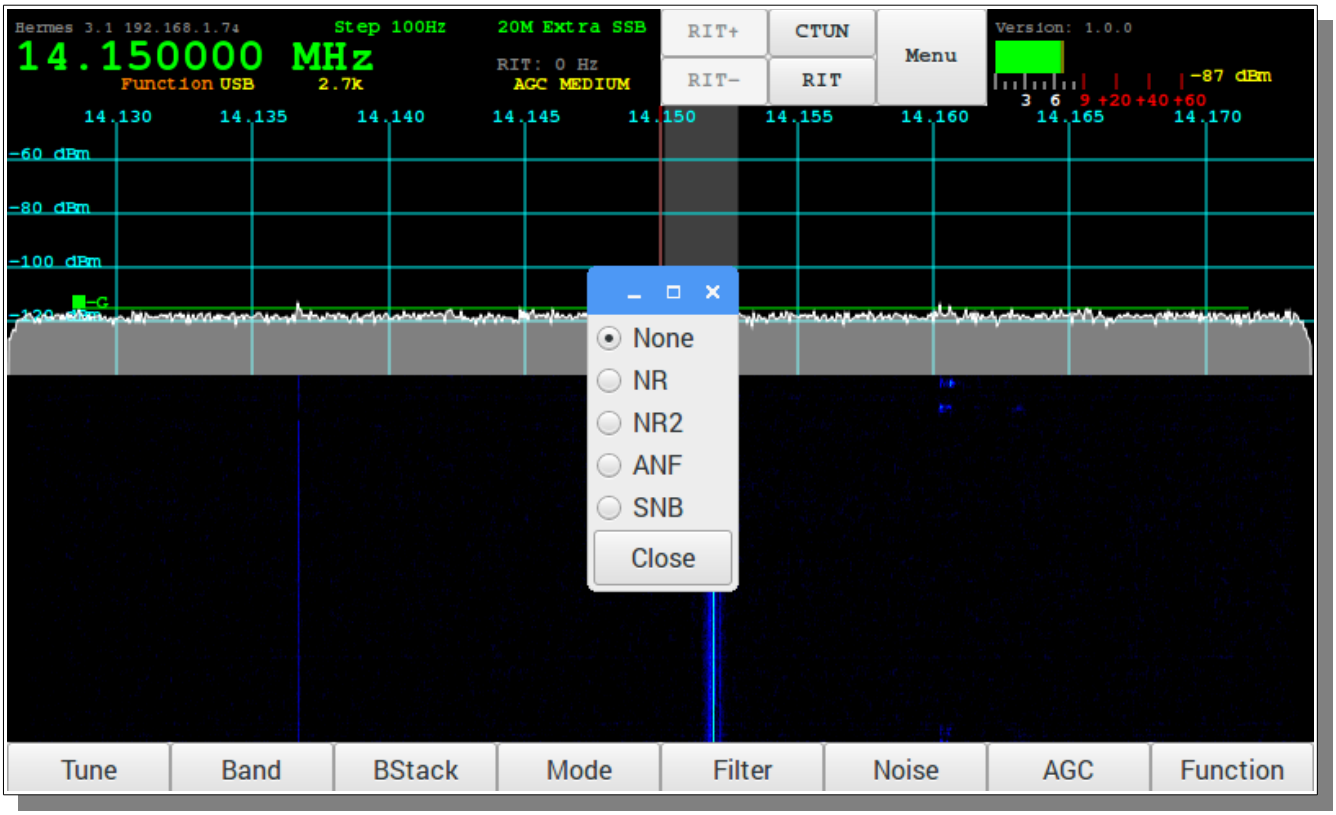



Tapping the Filter touch screen button or pressing the physical button will display the list of filters with the current filter highlighted in orange.

Tapping a filter will switch to the selected filter width.

Tapping on Close will close the menu.

## Noise

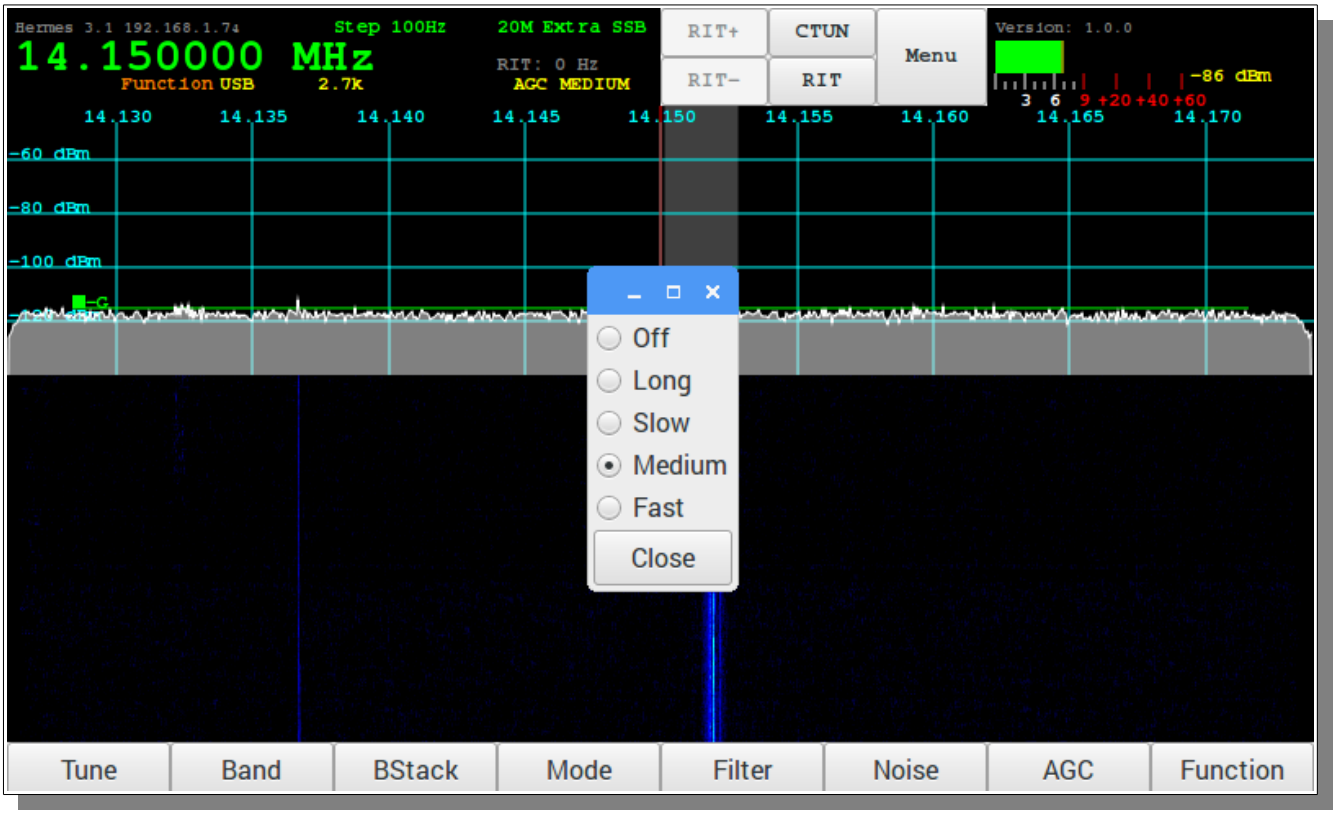



Tapping the Noise touch screen button or physical button will display the list of noise reduction modes are available.

You can select one of the following functions:

- **NR** - Noise Reduction
- **NR2** - New NR0V algorithm
- **ANF** - Automatic Notch Filter
- **SNB** - Spectral Noise Blanker

## AGC

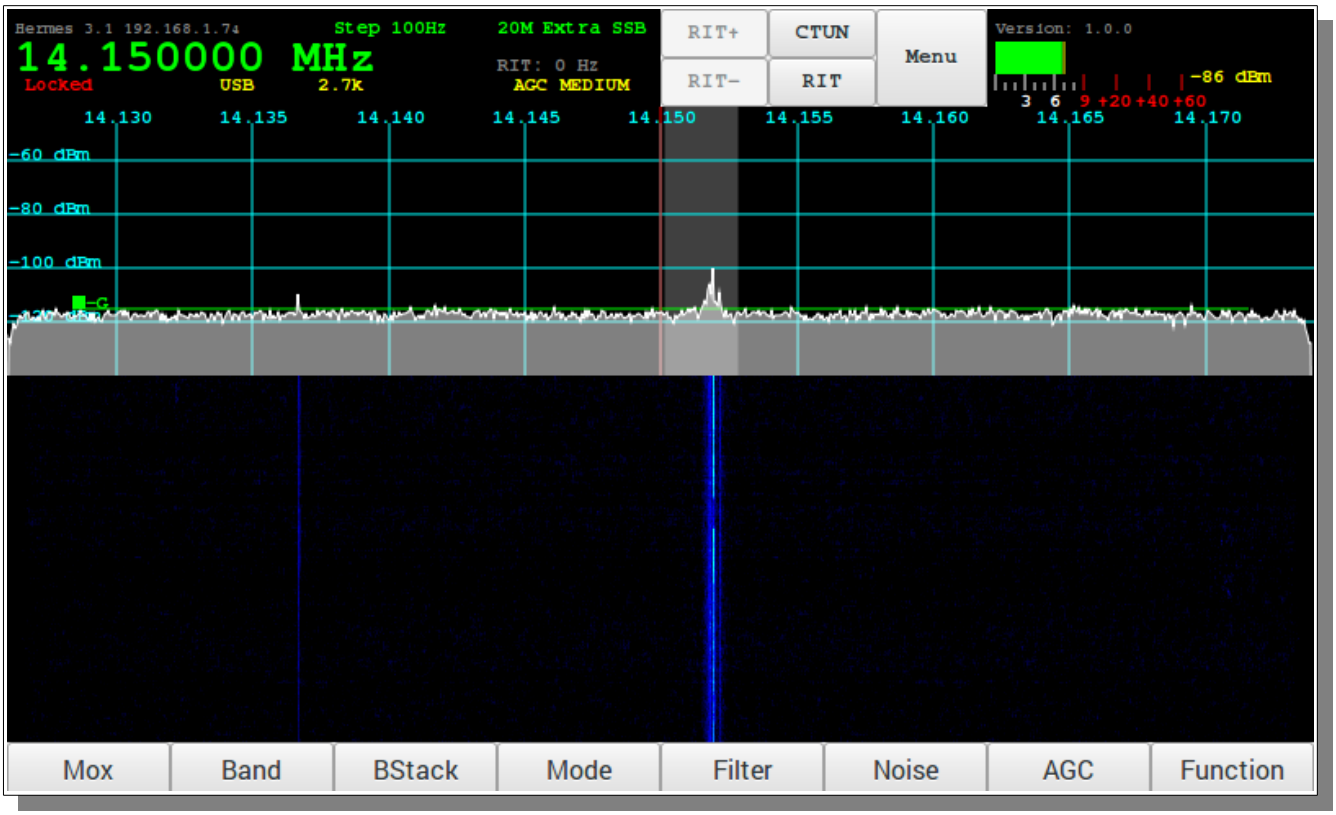



Tapping the AGC touch screen button or physical button will display the list of AGC options.

- Off – AGC disabled
- Long -Hangtime = 2000 ms. Decay_time_constant = 2000 ms.
- Slow - Hangtime = 1000 ms. Decay_time_constant = 500 ms
- Medium - Hang is turned OFF. Decay_time_constant = 250 ms
- Fast - Hang is turned OFF. Decay_time_constant = 50 ms

The AGC Gain levels are adjusted using the AGC encoder (or the AGC Slider if no encoders).

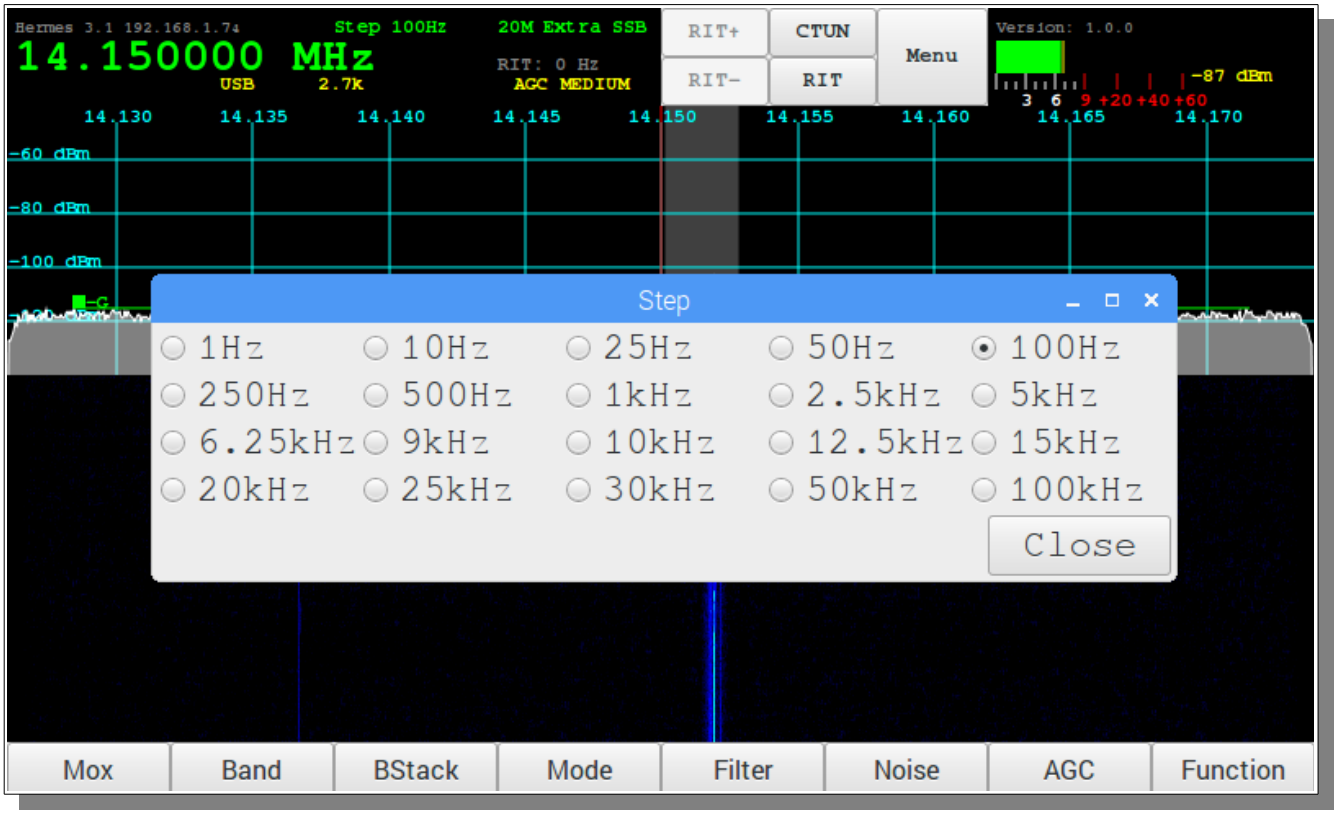
## VFO Lock



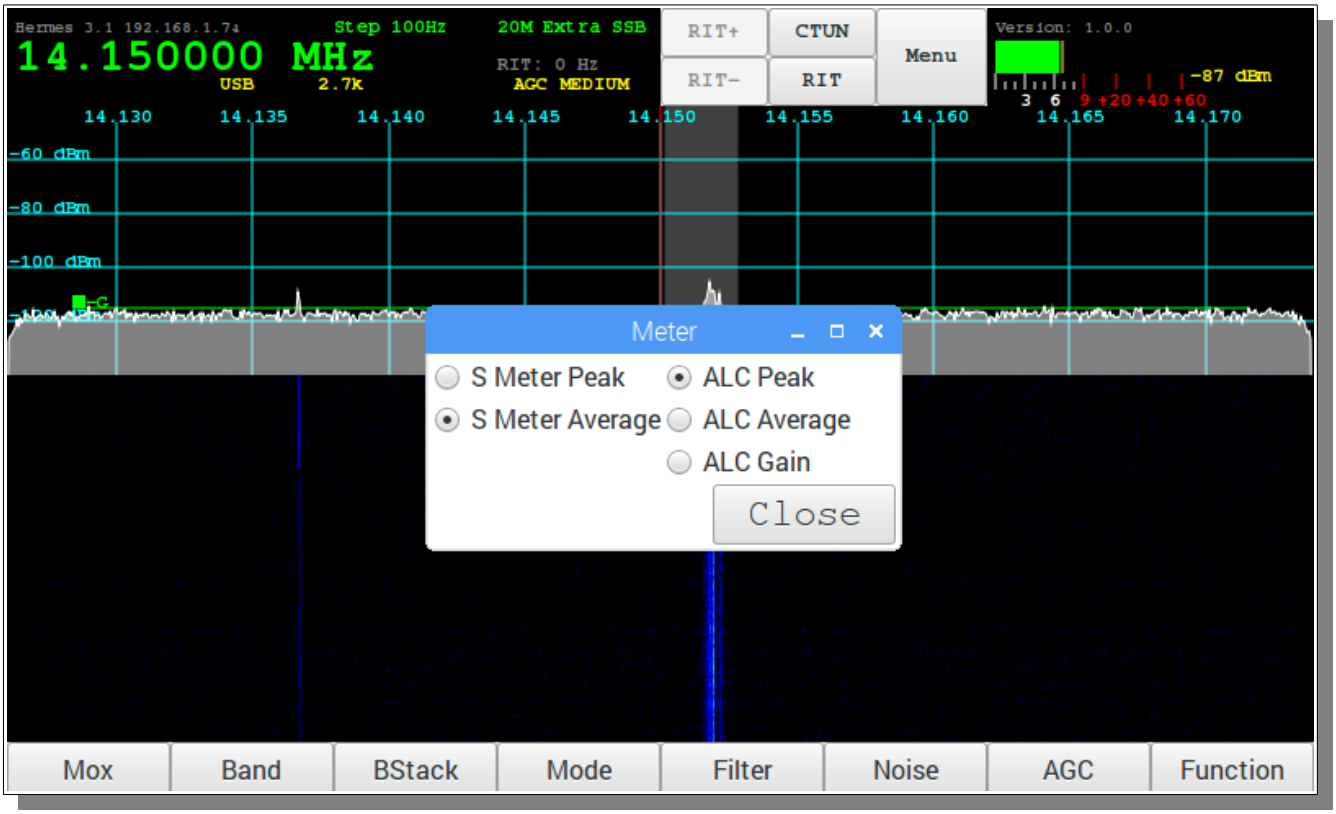

The VFO can be locked by pressing the button on the AF Gain encoder or by tapping on the left side of the VFO display. To unlock press the AF Gain encoder button again or tap on the left side of the VFO display. **When locked the text will be displayed in Red** on the VFO.

## VFO Step

By tapping on the right side of the VFO the menu to select the step rate can be selected. This will set the stepping rate when using the VFO encoder, dragging the panadapter or waterfall or using a mouse wheel



## Meter




**Tapping on the Meter will display a list of options** for updating the meter.

# 6. Tuning

## VFO Encoder

The VFO encoder can is used to tune the radio.  By turning the encoder clockwise and anticlockwise the frequency will increment or decrement by the amount of the step value.

The General Menu has a field to set the resolution of the encoder.

## Touch Screen

Touching and dragging on the panadapter or waterfall will move the frequency up or down.  Note that it will move in step increments.

Tapping a frequency on the panadapter or waterfall will move to that frequency.

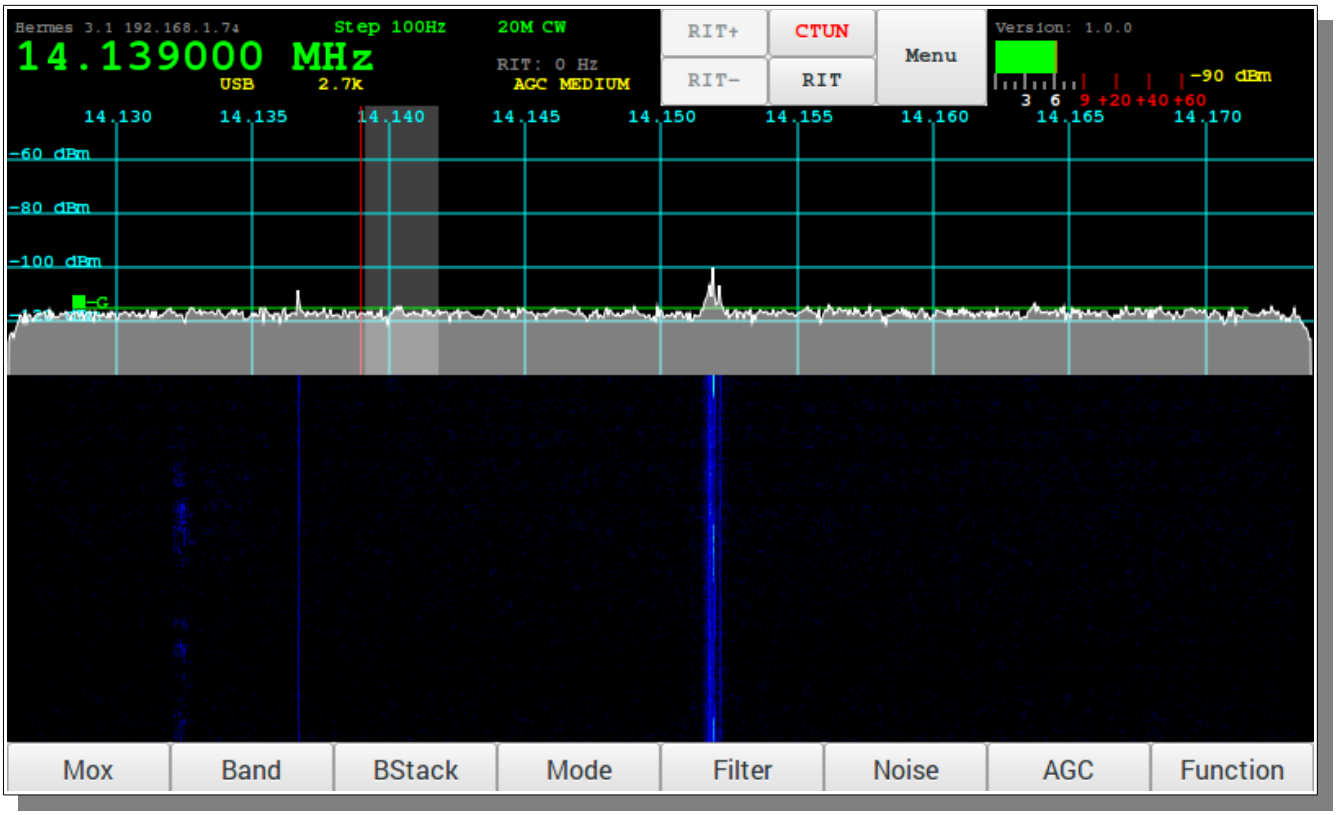
## Mouse

Left down and holding then dragging while on the panadapter or waterfall will move the frequency up or down.

Left clicking will move to the selected frequency.

Moving the scroll wheel will increment or decrement the frequency by the step value.

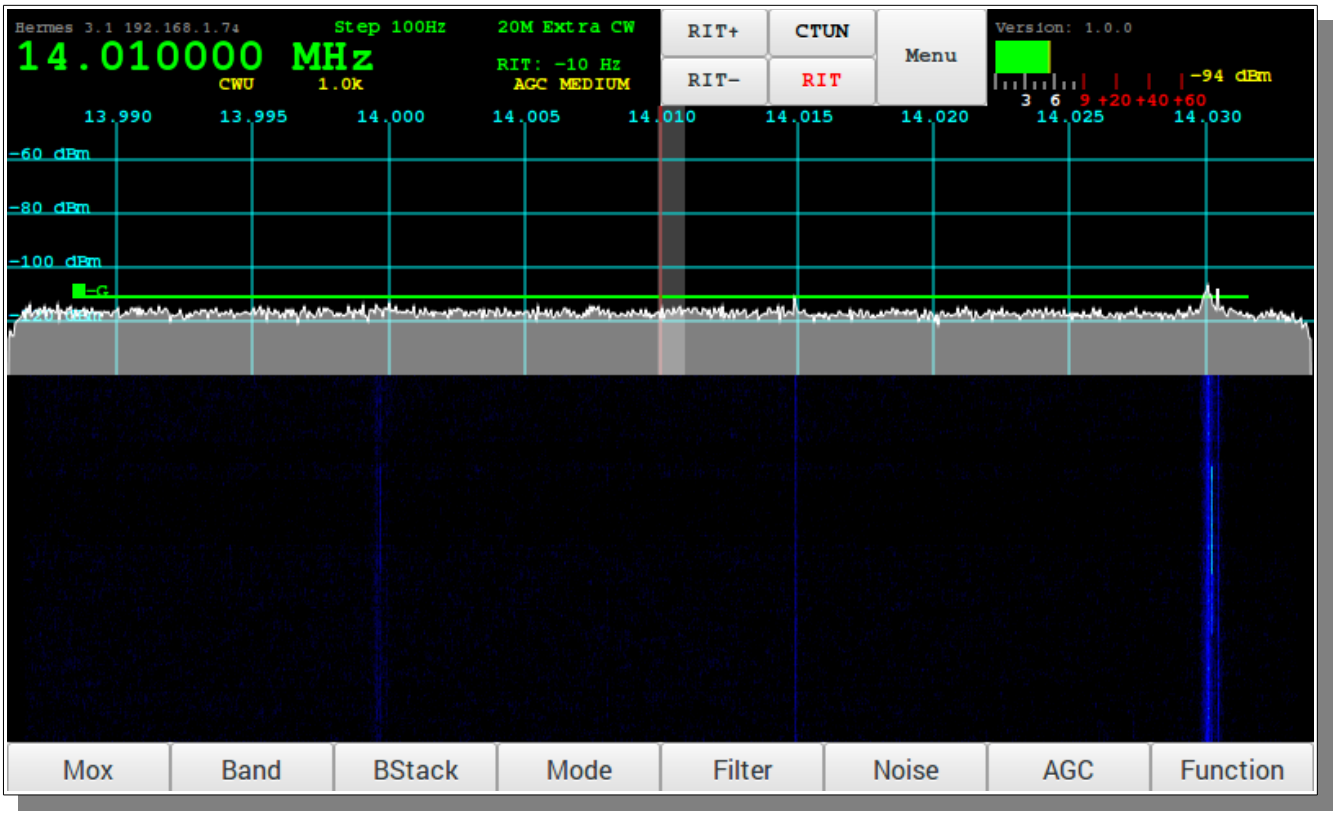
## CTUN – Click Tuning




Tapping on the CTUN button will enable or disable the click tuning function. When the function is enabled the CTUN button text will be red.

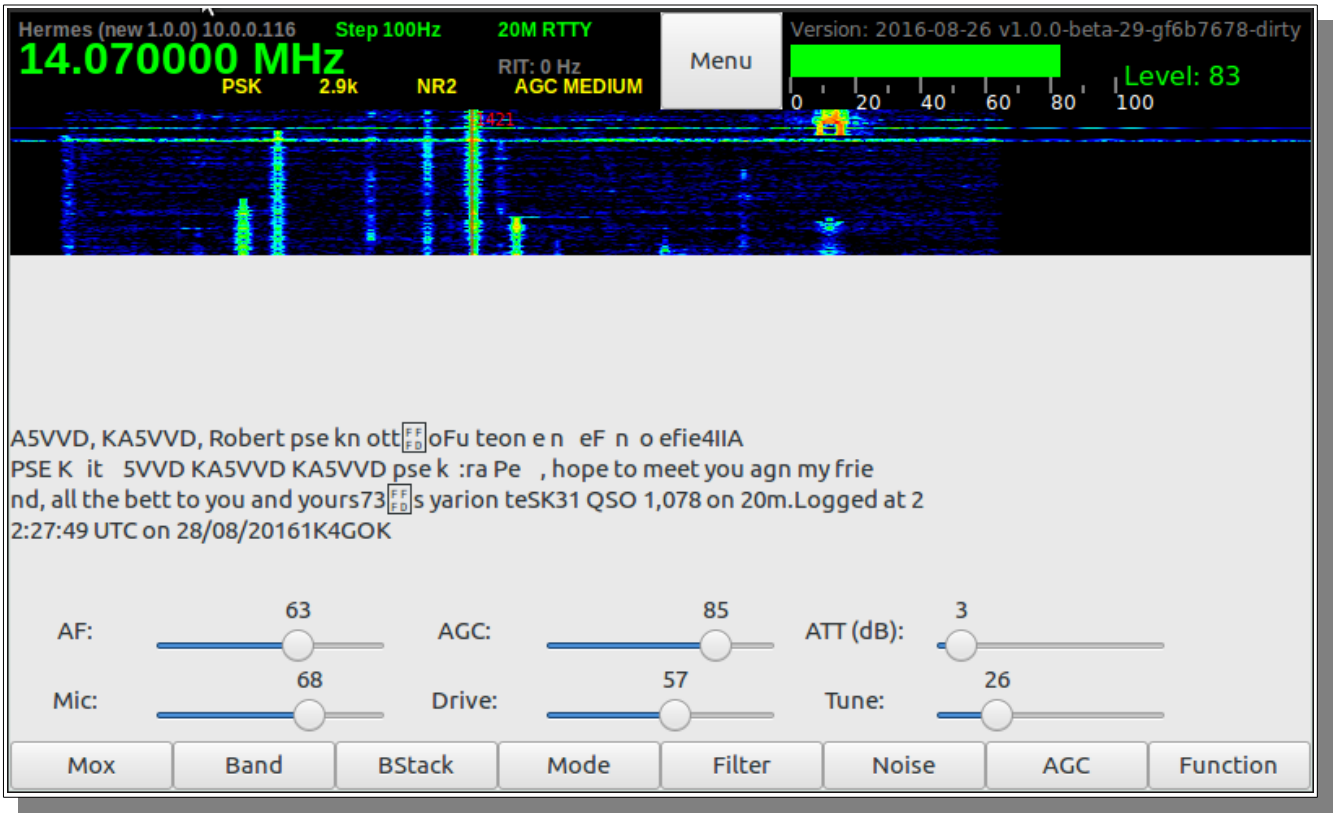
When CTUN is enabled tuning is restricted to be within the passband currently being received. The tuned frequency and filter moves withing the current passband without scrolling the Panadapter or waterfall left or right.

## RIT



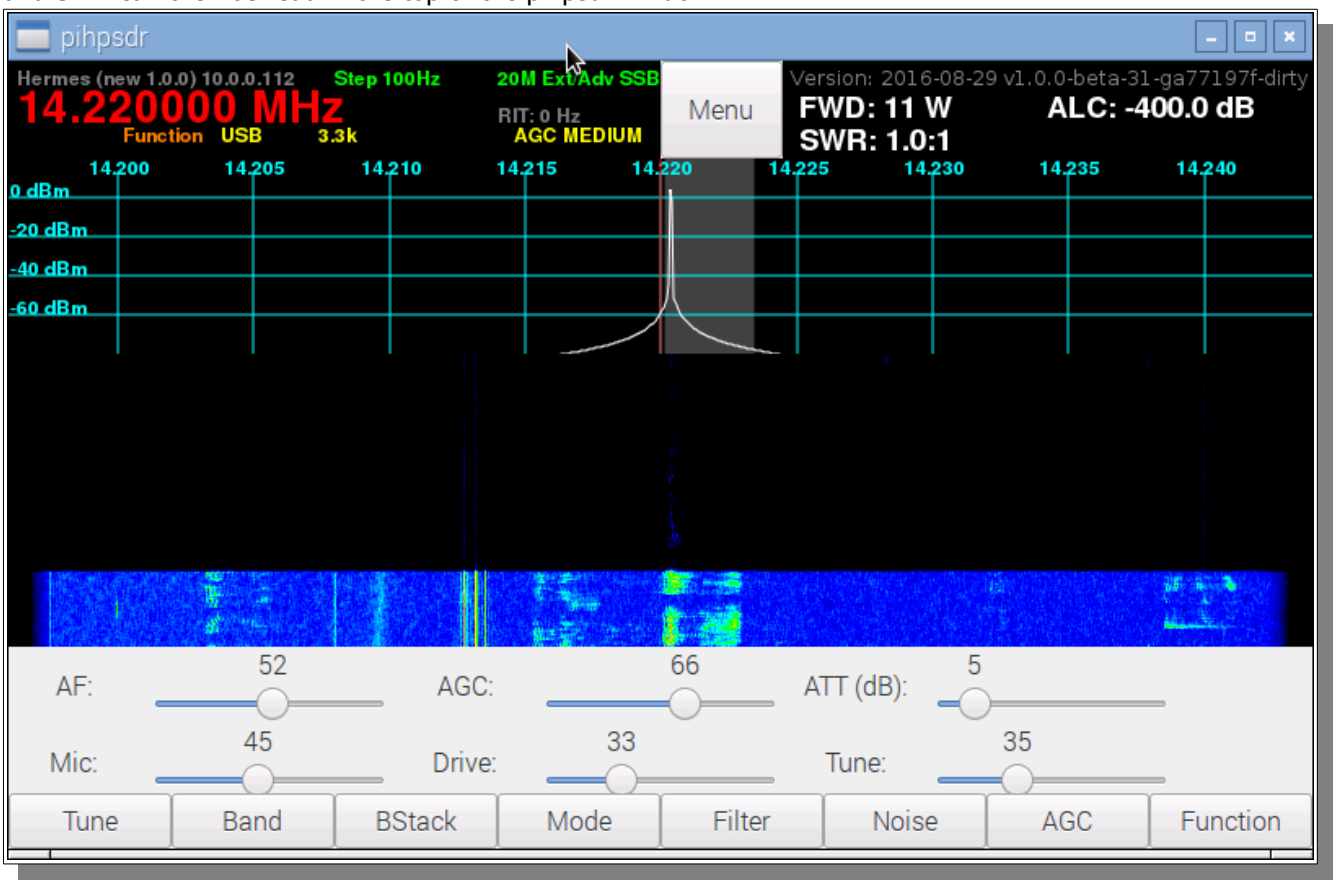

T**apping on the RIT button will enable RIT**. The RIT+ and RIT- buttons can be tapped or held to increment decrement the RIT frequency in Hz.

## 7. Mode PSK31 – Receive




## TUNE/SWR/FWD power

When the Function button is touched or selected, it toggles the TUNE/MOX button on the far left. FWD power and SWR can then be read in the top of the pihpsdr window.



# 8. Appendix

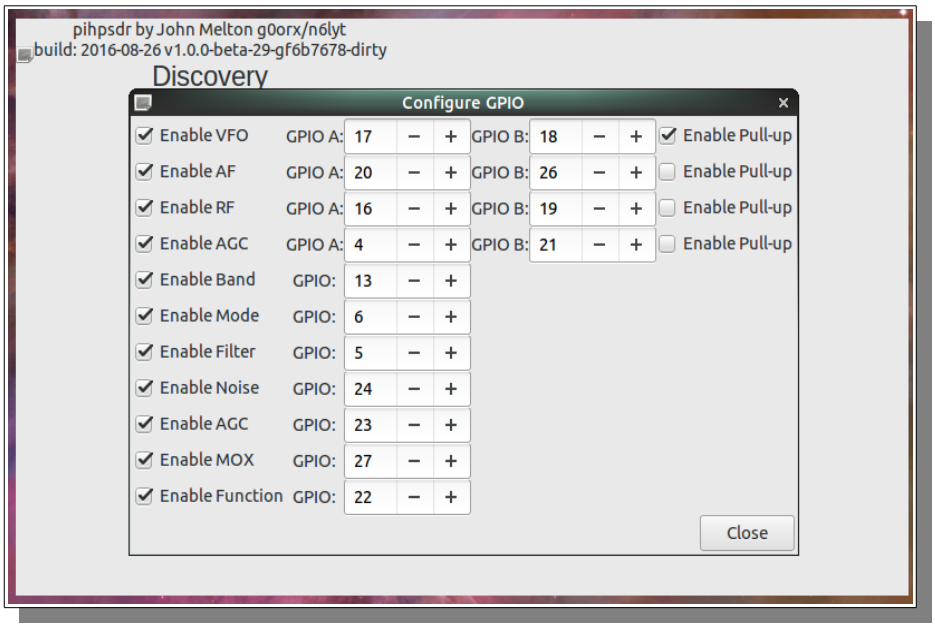
## Encoders and Switches

**Factory Switch/Encoder/Touch activation**

| Switch or Encoder | Function | Touch Screen | GPIO |
|---|---|---|---|
| Power ON/OFF | Controller power on/off | Controller power on/off | |
| TUN – TUNE button | generates a carrier with Tune power selection slider | generates a carrier with Tune power selection slider | |
| S1 - Band | 10 HF Bands + General Coverage + WWV | 10 HF Bands + General Coverage + WWV | 13 |
| S2 - Mode | LSB USB DSB CWL CWU FMN AM DRM FreeDV PSK | LSB USB DSB CWL CWU FMN AM DRM FreeDV PSK | 6 |
| S3 - Filter | 10 IF Filter widths | 10 IF Filter widths | 5 |
| S4 - Noise | 5 Noise Reduction modes | 5 Noise Reduction modes | 24 |
| S5 -AGC | 5 AGC Decay settings | 5 AGC Decay settings | 23 |
| S6 - MOX | Space bar toggle for PTT<br>or MOX button | Space bar toggle for PTT<br>or MOX button | 27 |
| FN – Function | Function switch to toggle TUNE or MOX button displayed on the bottom left of the screen | Function switch to toggle TUNE or MOX and the action of E1, E2, E3 knobs | 22 |
| E1 -- | two functions AF Gain / MIC Gain | two functions AF Gain / MIC Gain | 20 / 26 |
| E2 – | two function RF Drive / TUNE Drive | two function RF Drive / TUNE Drive | 16 / 19 |
| E3 – | two function AGC Gain / Rx ATTEN | two function AGC Gain / Rx ATTEN | 4 / 21 |
| E4 VFO – Main tuning knob | E4 large tuning knob | Touch to tune | 17 / 18 |
| Menu | ------- | Main piHPSDR Menu | ---- |
| | ------- | | ------- |
| Band | | | ------- |

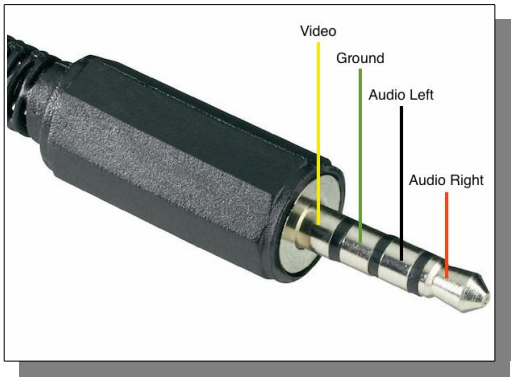
| Switch or Encoder | Function | Touch Screen | GPIO |
|---|---|---|---|
| Band Stack | 3 to 5 level quick freq change | 3 to 5 level quick freq change | |
| Mode | Mode | Mode | |
| Filter | IF Filter width | IF Filter Width | |
| Noise | Noise Blanker mode | Noise Blanker Mode | |
| AGC | Automatic Gain Attack speed | Automatic Gain Attack speed | |
| Locked | Right click on VFO Frequency to toggle Frequency Lock | Touch VFO Frequency to toggle Frequency Lock | |
| Meter | Right click on S-Meter for S-Meter and ALC peak/average | Touch S-Meter for S-Meter and ALC peak/average | |

## PIO pin Assignments (RaspberryPi)




## RaspberryPi-3b TRRS connector

The four wire **TRRS** audio/video connector on the RaspBerryPi-3b should be use carefully. The Video output may connect incorrectly on a standard 3-pin Stereo plug.




Adafruit has an adapter that may be helpful for experimenters:

<URL: https://www.adafruit.com/products/2914 >

SparkFun has an open wire plug and cable

<URL: https://www.sparkfun.com/products/11580 >

## MIC Input is easily done with an inexpensive USB adapter

<URL: https://www.adafruit.com/product/1475>

## 9. Reference materials

G0ORX Friedrichshaven pdf and video
Friedrichshafen 2016 publication

https://www.dropbox.com/sh/fva5d5mi93c93tq/AAD6dU-eBMR0cVK-E95hJC5Ia?dl=0

Jacinto Rebelo CU2ED for his Homebrew of a pihpsdr Controller

Kjell Lasen LA2NI for his complete RPi-e System

F'Hafen video on YouTube:
https://www.youtube.com/watch?v=U7QfP28YjCw

## 10. Volunteer Programmers can help with ideas

**PureSignal**

**PSK31 Transmit encoding**

**RTTY receive decoding**

**VOX**

**Thin Pipe Remote Audio/Video**

**Panadapter color selection**

**IF-shift, Hi-Cut, Low-Cut**

**SPLIT VFO-A/VFO-B**

## Apache Support

### Apache Labs International Support

Technical support for ANAN-10 from the factory is available via the Apache Labs Yahoo Group
**http://groups.yahoo.com/group/apache-labs/**

or directly via email **<support@apache-labs.com>**

### Apache Yahoo Support Group

**http://groups.yahoo.com/group/apache-labs/**

### OpenHPSDR Group

Instructions relating to joining the OpenHPSDR Group reflector are here:

http://lists.openhpsdr.org/listinfo.cgi/hpsdr-openhpsdr.org

The OpenHPSDR archives may also be searched here:

http://lists.openhpsdr.org/mmsearch.cgi/hpsdr-openhpsdr.org

The latest version of the OpenHPSDR User Manual can be obtained from

http://openhpsdr.org/documents.php Author Phil Harman VK6PH

### Apache Service and Repair

1023 Tower B4, Spaze I-Tech Park

Sector - 49, Sohna Road

Gurgaon - 122001

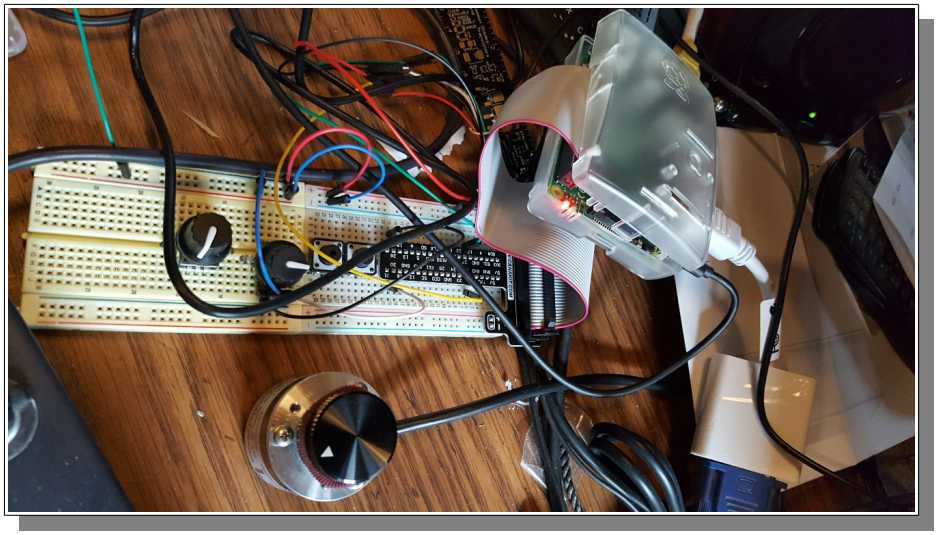
Haryana, India

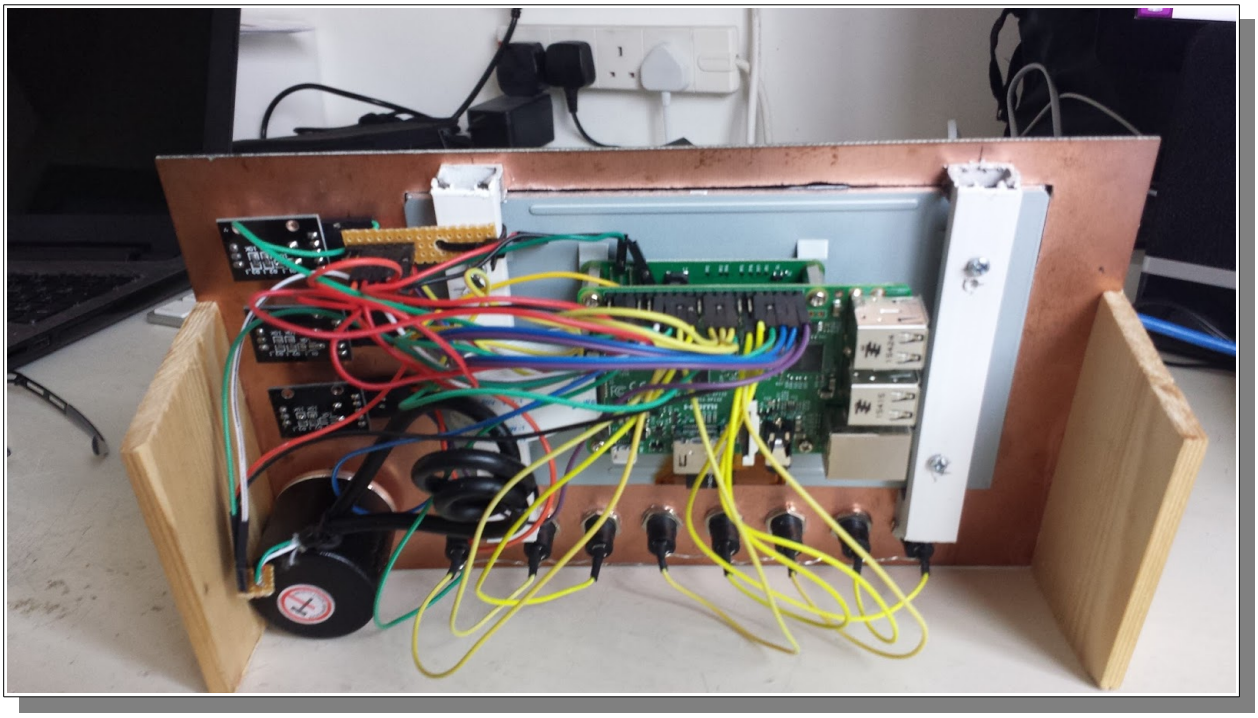
Tel: 91-0124-4245173/4/5 (10AM - 6PM IST]

Email: support@apache-labs.com

Website: **http://www.apache-labs.com**

Early prototypes by John Melton G0ORX





Note: pihpsdr is OpenSource. If you would like to run the program on a Linux system, the source, binaries, and documentation are located here: http://g0orx.blogspot.com/

# NOTES

# Alphabetical Index