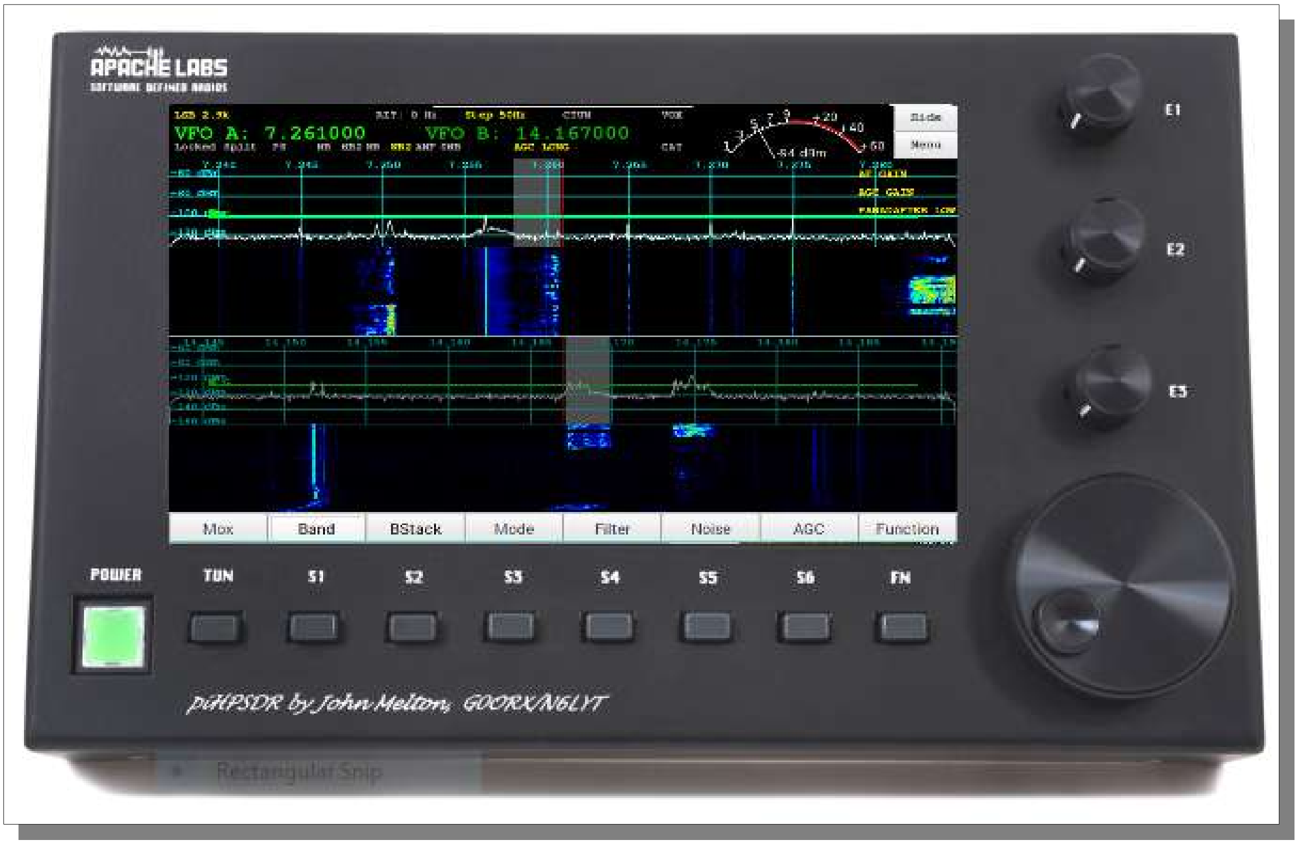


NEW
Version 1.2

Apache Labs LLC

piHPSDR Controller

Users Guide

by

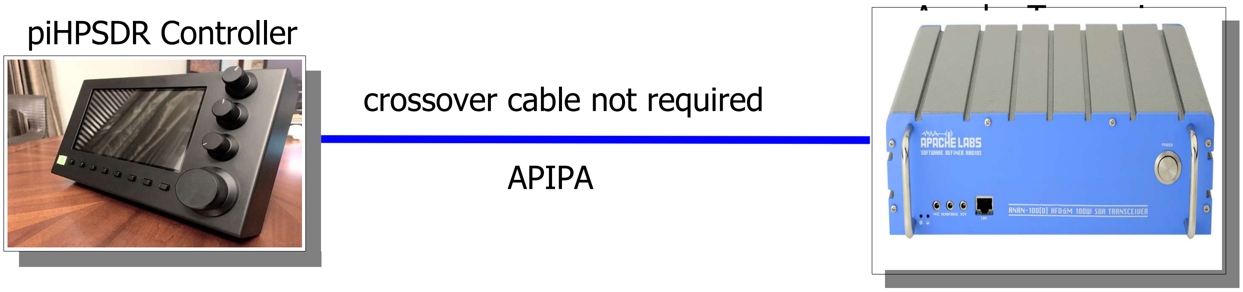
John Melton GØORX/N6LYT

Contents - piHPSDR Controller Users Guide

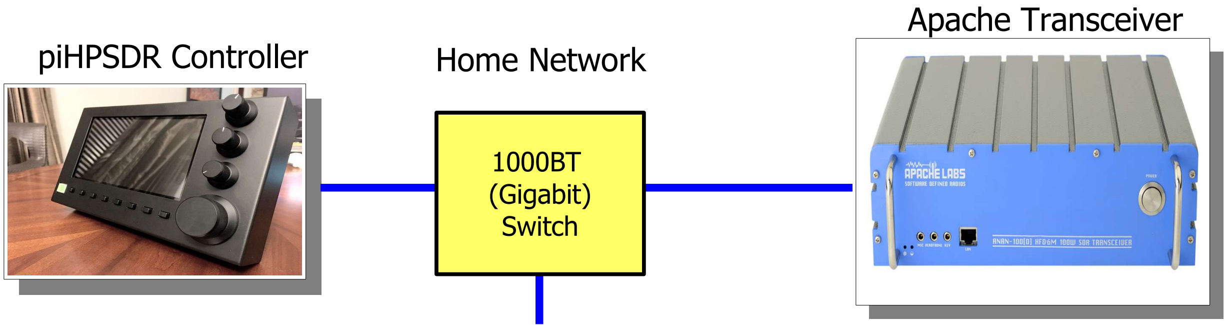
1. Controller connection requirements

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 to the LAN

Controller Hardware requirements

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

Note: when running the dual receiver version of pihpsdr, you may need to place passive heatsinks on the CPU and LAN chip on the RaspberryPi-3b. Be mindful of the temperature alarm shown on the right side of the display screen.

Heatsinks like these:

- **https://www.modmypi.com/blog/how-to-install-heat-sinks-on-the-raspberry-pi**
- **https://www.element14.com/community/community/raspberry-pi/raspberrypi_projects/blog/2016/03/03/raspberry-pi-3-cooling-heat-sink-ideas**
- **https://www.amazon.com/Addicore-Raspberry-Heatsink-Aluminum-Sinks/dp/B00HPQGTI4/ref=pd_lpo_147_bs_t_2?_encoding=UTF8&psc=1&refRID=6D4ND9MW55Y93DD05V26**

Controller software requirements:

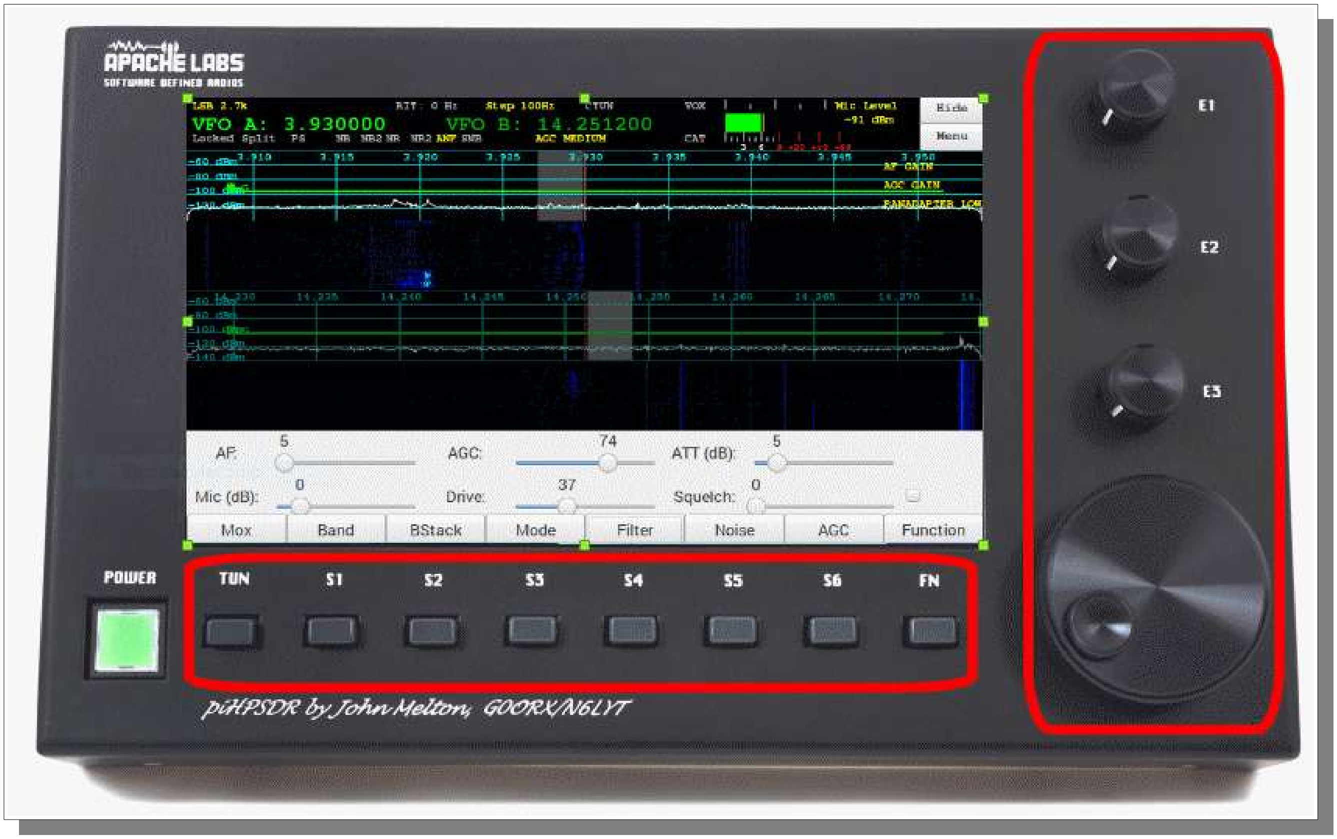
- **Pihpsdr Controller Operating System and application program are factory installed on an SDHC card.**
- If you have purchased the piHPSDR-Controller KIT, you will need to create and install an SDHC card with the RaspberryPi operating system and pihpsdr application.
- Instructions for Installation of Rpi software 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.

2. Front panel controls



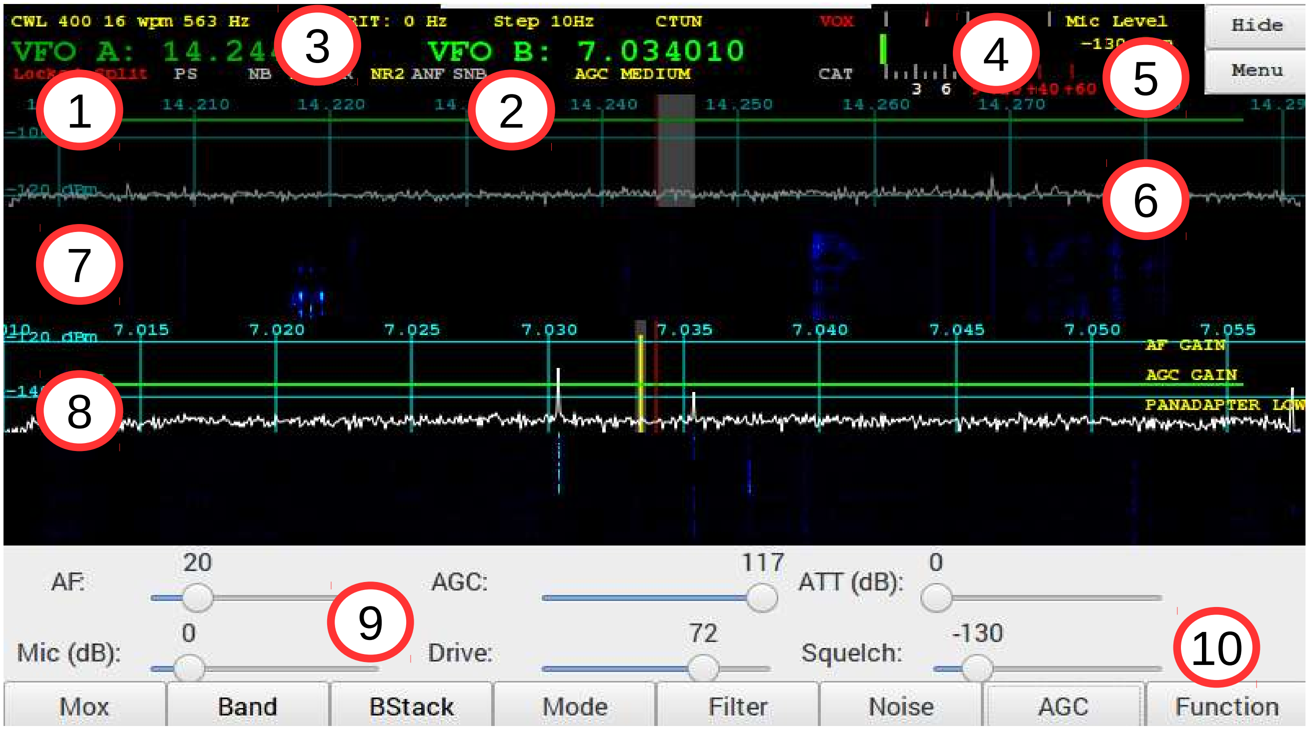
Switches Encoders

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/Reducer**
- **S6 – AGC**
- **FN – Function switch to toggle the functions of the S1-S6 buttons**
- **E1 – Rotate to change the value of the assigned function. Push Encoder knob to pop up menu for functions, rotate to select function, push to close the function menu.**
- **E2 – Rotate to change the value of the assigned function. Push Encoder knob to pop up menu for functions, rotate to select function, push to close the function menu.**
- **E3 – Rotate to change the value of the assigned function. Push Encoder knob to pop up menu for functions, rotate to select function, push to close the function menu.**
- **E4 – VFO – main tuning knob**

  Note: Appendix page 74 has a table showing the Switch, Encoder, and Touch Screen activation details.

3. Front Panel Display



① VFO-A Receiver Rx0

② VFO-B Receiver Rx1

③ Status Lines above and below VFO frequency

④ Meter displays S-Meter and ALC displays

⑤ Main MENU on screen control

⑥ Panadapter display Rx0

⑦ Waterfall Rx0

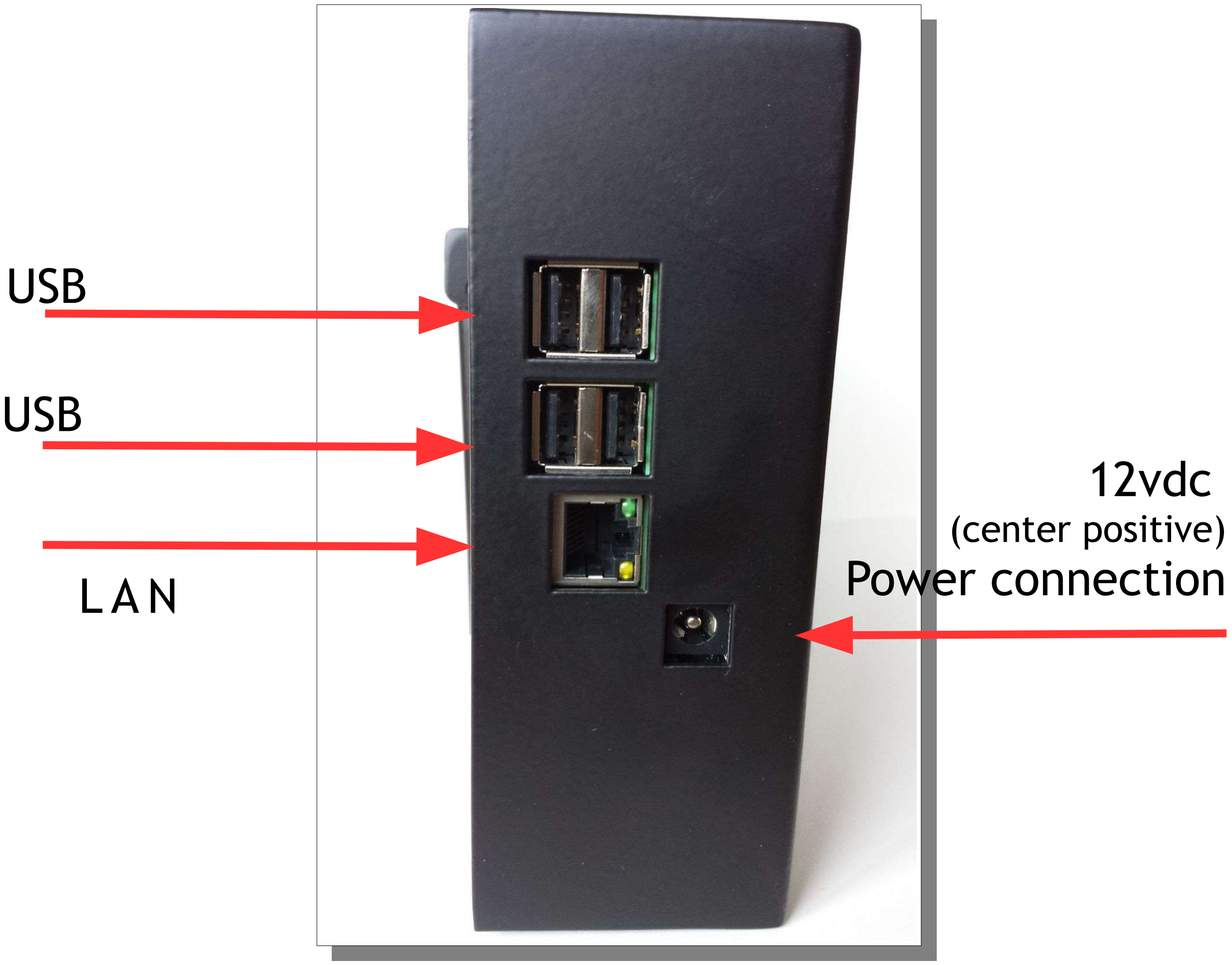
⑧ Panadapter display Rx1

⑨ 6 Slider controls

⑩ Function Menu activation

Note: if you are not using a <u>touch sensitive display panel</u>, you may wish to operate pihpsdr from your favorite Linux system using the RealVNC connection or connect a mouse/keyboard for easy access.

4. Side panel connections



5. 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 (13.8vdc) Amateur Radio station power supply.
- **Apache Transceiver connections to MIC, KEY, Headphones**

Software Setup

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

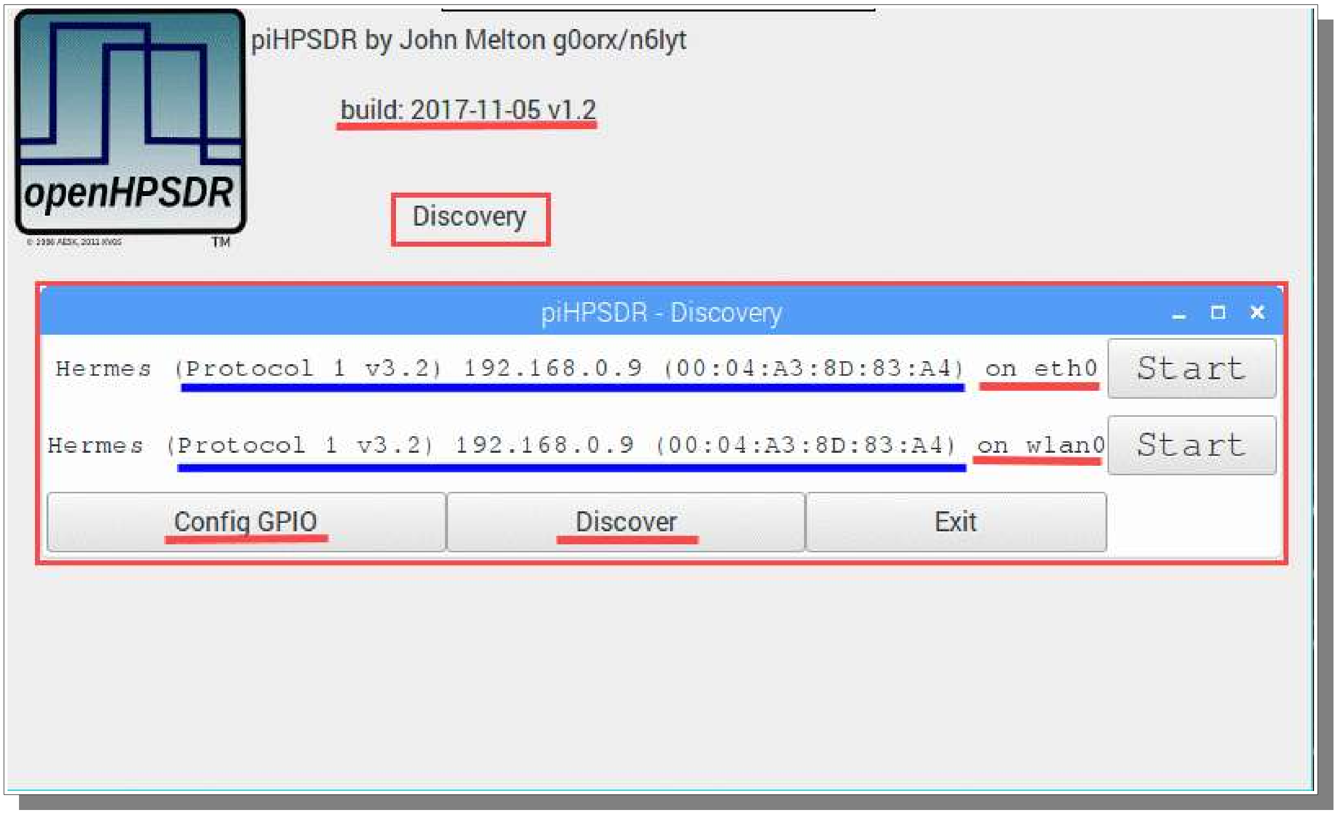
Alternative software configurations are not provided by Apache Labs.

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/pihpsdr/tree/master/release/documentation

6. Discovery Menu

Discover Ethernet Connections to RPi



piHPSDR by John Melton g0orx/n6lyt

build: 2017-11-05 v1.2

```
Hermes (Protocol 1 v3.2) 192.168.0.9 (00:04:A3:8D:83:A4) on eth0
Hermes (Protocol 1 v3.2) 192.168.0.9 (00:04:A3:8D:83:A4) on wlan0
```

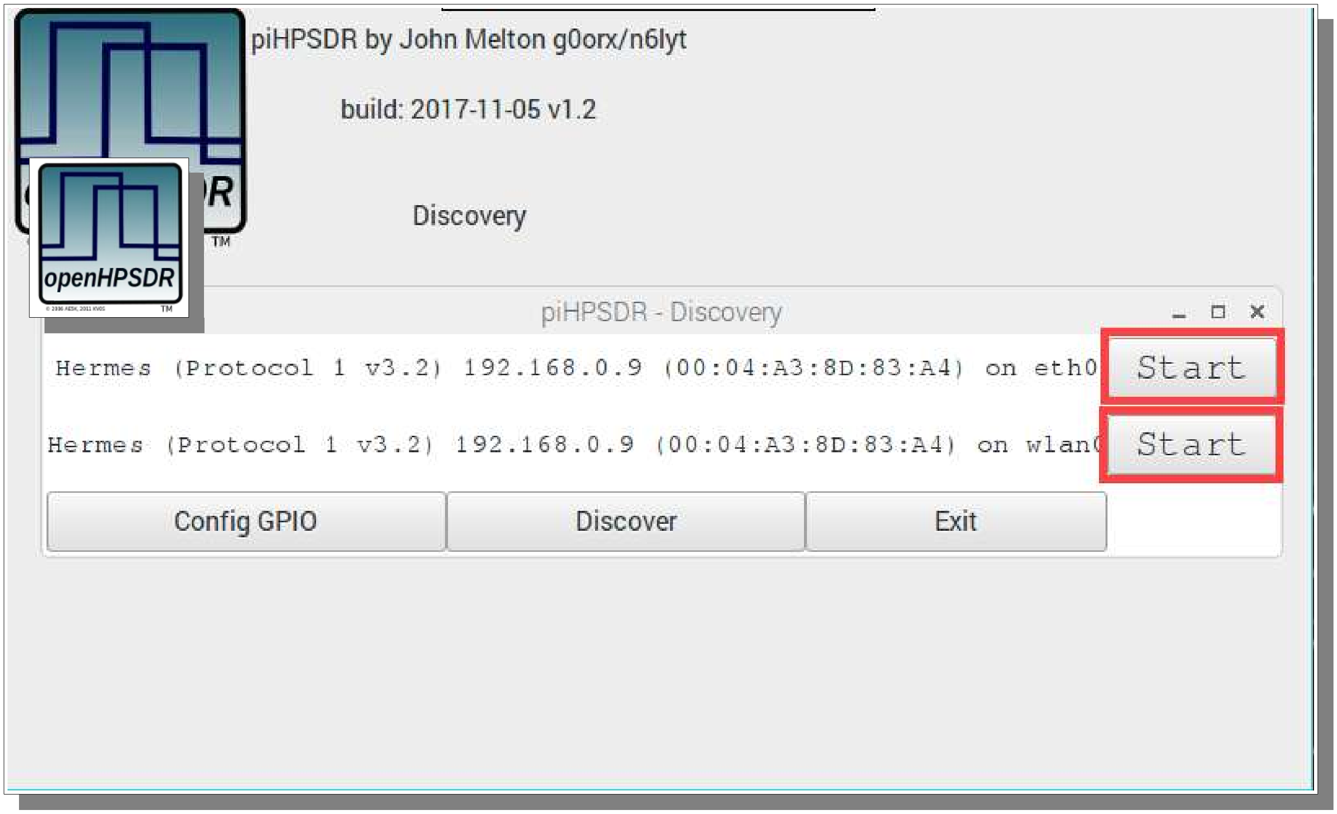
- **When first started, piHPSDR will try to discover all the HPSDR compatible radios on the network. It will look for devices running both protocol 1 and protocol 2 (the new protocol).**

- **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 Discover 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.

Pihpsdr program START button



piHPSDR by John Melton g0orx/n6lyt

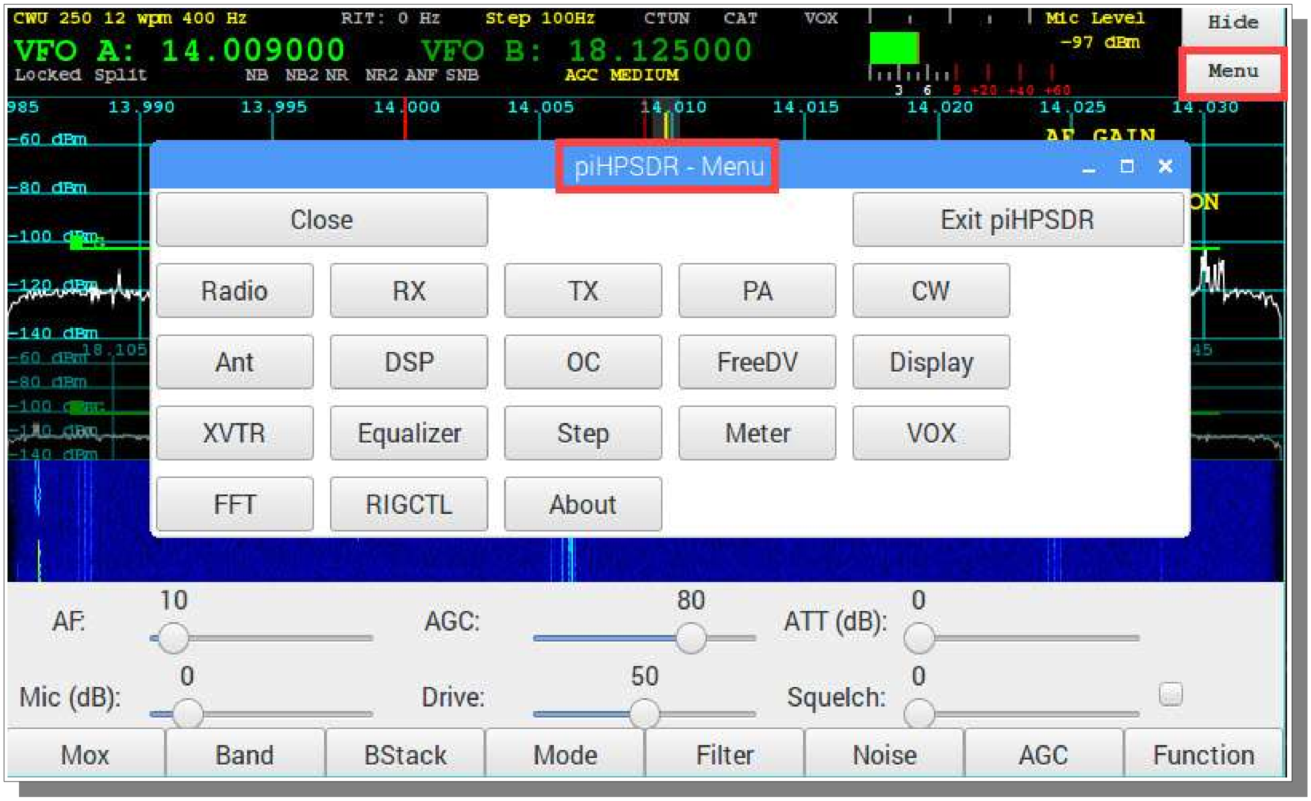
build: 2017-11-05 v1.2

Discovery

```
Hermes (Protocol 1 v3.2) 192.168.0.9 (00:04:A3:8D:83:A4) on eth0
Hermes (Protocol 1 v3.2) 192.168.0.9 (00:04:A3:8D:83:A4) on wlan
```

- **Tapping on the Start button will start the radio – on the selected Ethernet interface.**
- **In this example you see both the Rpi Wireless and Ethernet connections are active and were discovered properly.**
- **Depending on your particular WLAN and LAN setup, either will work well with pihpsdr 100Mbps network requirement. Both the Rpi and your Apache Transceiver must be on the same subnet.**
- Configuration of the **GPIO** default connections are shown on page **57**

7. Main Menu

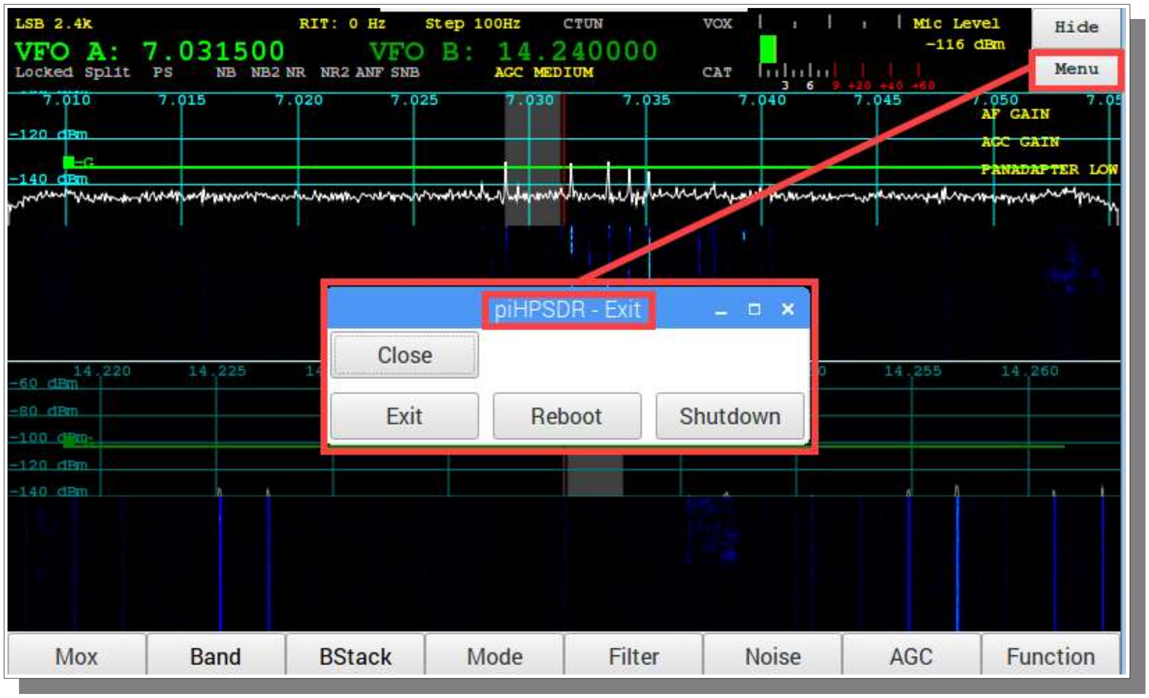


Tapping the Menu button brings up the menu dialog.

Menu items explained in detail in following pages

- **Radio** Receivers, Alex, RIT, VFO
- **RX** Sample Rate, Dither, Local Audio, Stereo
- **TX** MIC/Line, Boost, TX Filter, TX Display, FM, CTCSS, Tune Drive
- **PA** Calibration all bands
- **CW** Break-in delay, Modes, Speed, Sidetone Level, Sidetone Freq, Weight
- **ANT** ANT-1,2,3 EXT-1,2,3 XCTR Tx ANT-1,2,3 GEN, WWV, 136kHz, 472kHz
- **DSP** AGC Hang Threshold, NR/NR2/ANF pre-post-selection
- **OC** Open Collector Output pin assignments
- **FreeDV** FreeDV Digital Voice
- **DISPLAY** Fill, FPS, Pan High, Pan Low, Waterfall, Waterfall High, Waterfall Low, Detector, Averaging
- **XVTR** Transverter Min-Freq, Max-Freq, LO Freq, Disable PA
- **Equalizer** Enable Tx and Rx Equalizers in four octave groups
- **Step** VFO Step selection of 20 step sizes
- **Meter** S-Meter Analog, or S-meter Bar graph, S-Meter Peak/Average; ALC Peak/Average/Gain
- **VOX** MIC Level, VOX Threshold, VOX Hang Time
- **FFT** Filter Type - Linear Phase or Low Latency
- **RIGCTL** Rigctl Enable, RigCtl Port Number
- **About** software version, WDSP version, Device, MAC Address, Interface IP Addresses

Menu → Exit

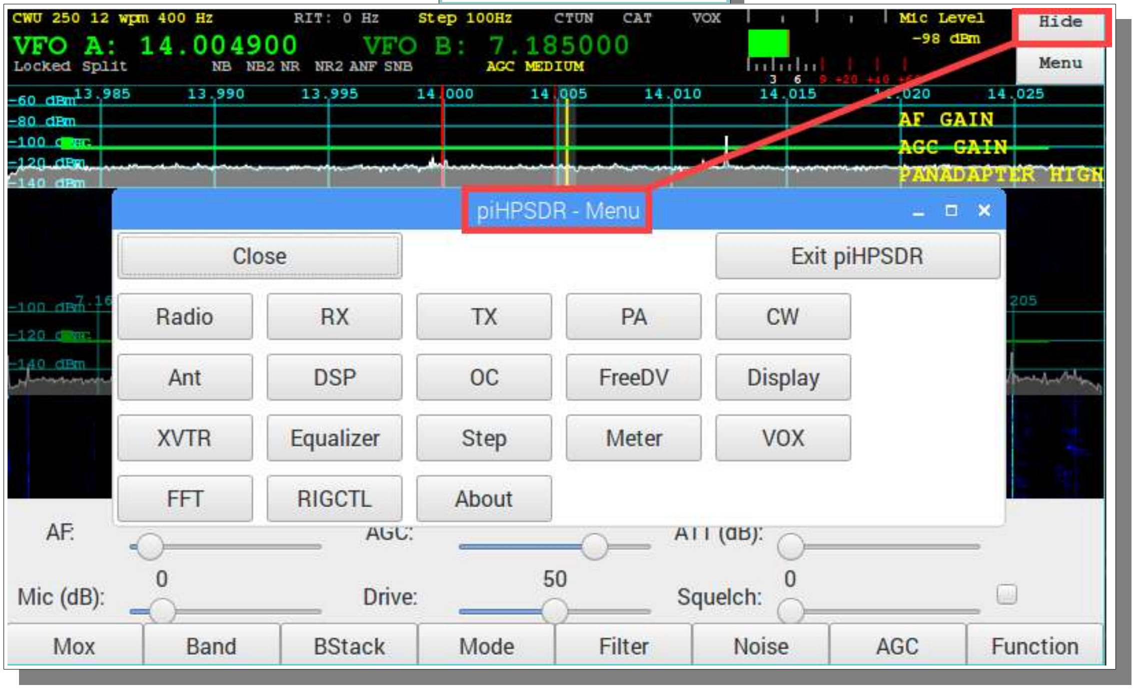


The Exit piHPSDR button will bring up a sub-dialog allowing you to end the piHPSDR application in a number of ways:

- **Close –** close this dialog box
- **Exit – exit piHPSDR back to the Raspberry Pi Desktop**
- **Reboot – reboot the Raspberry Pi**
- **Shutdown – Shutdown the Raspberry Pi - (note the Controller power button remains lighted)**

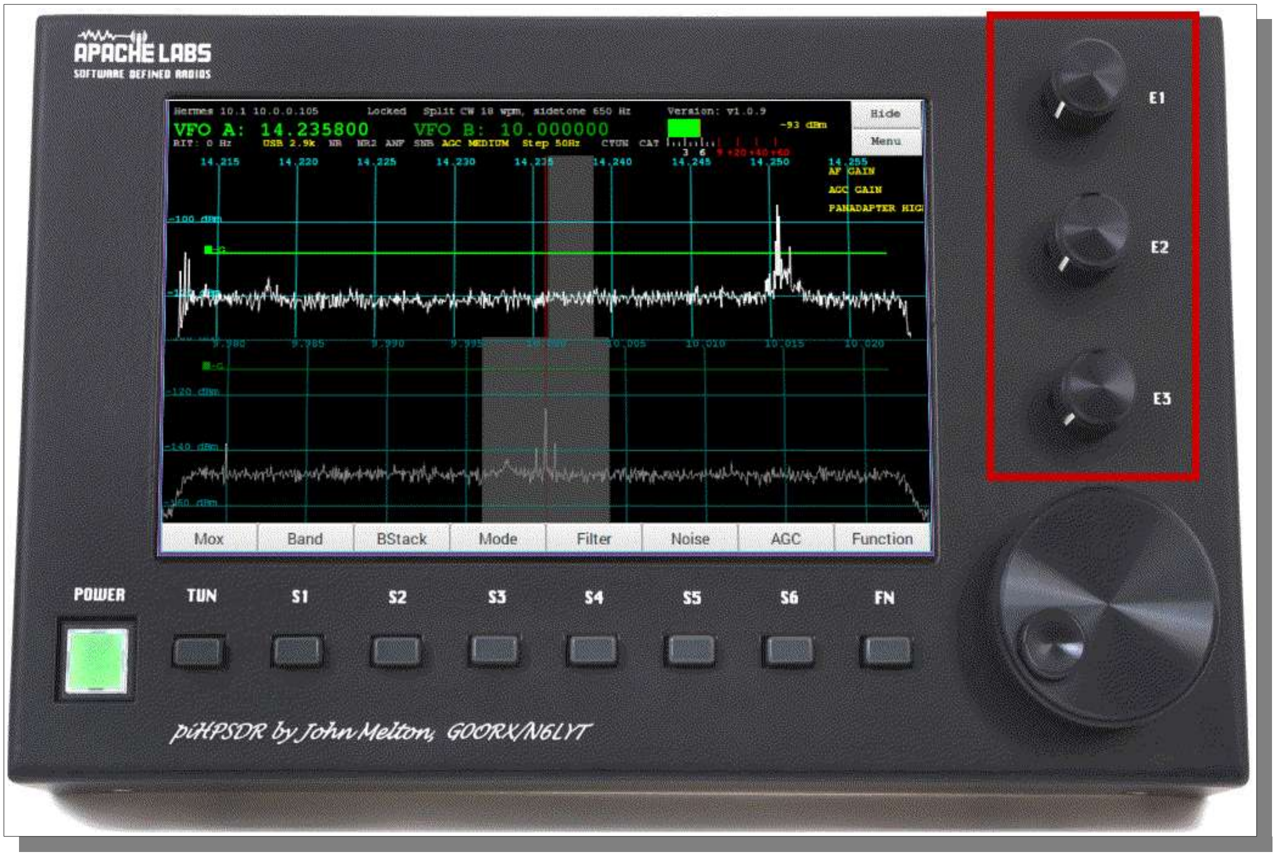
You can also tap the Menu button again to close the Exit Menu and bring up the Menu selection.

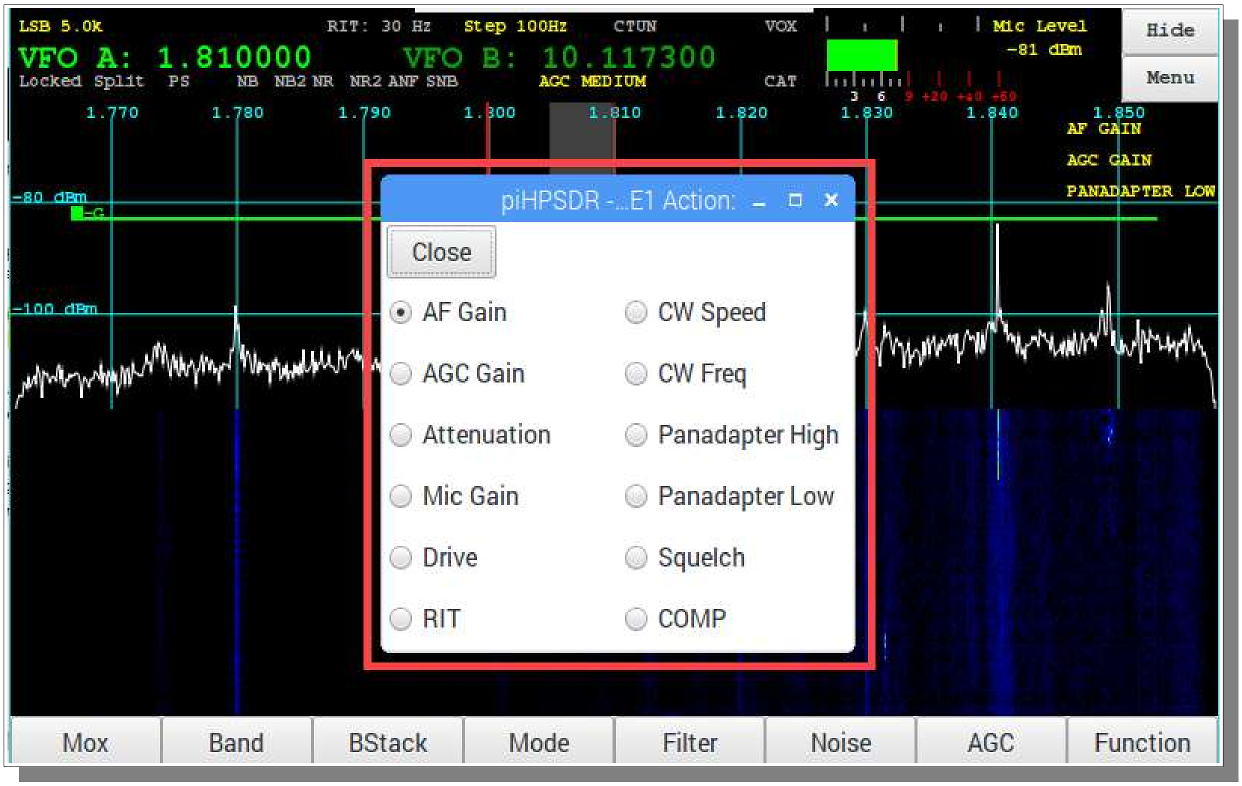
Menu → Hide



The purpose of the HIDE option is to minimize the pihpsdr application window and to then provide easy access to the RaspberryPi operating system.

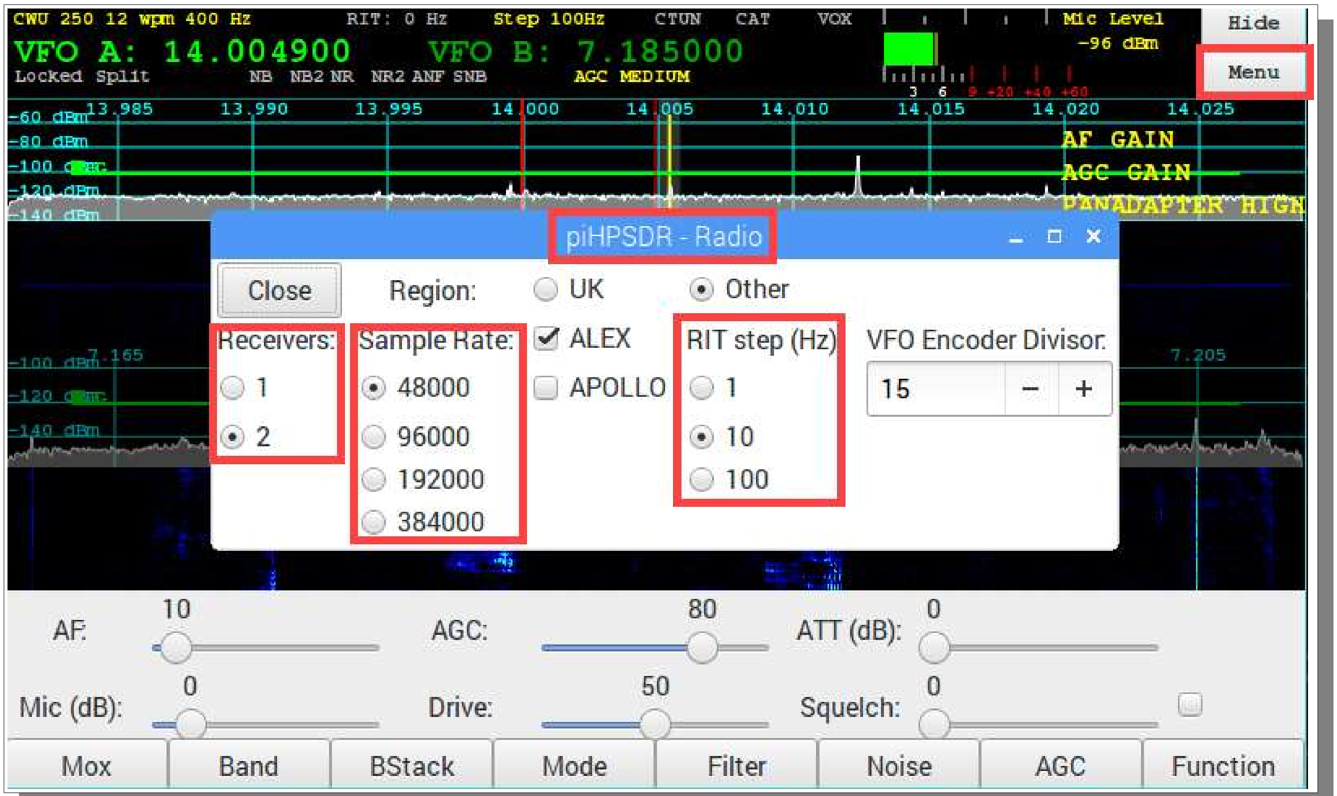
E1, E2, E3 Click-Set Assignments





The function of E1, E2, and E3 are shown in the Panadapter display. Push E1, E2, or E3 to pop up encoder selection. Rotate encoder or tap screen to select function. Push encoder or tap to close the menu.

Menu → Radio



piHPSDR - Radio

Close

Region: ○ UK ● Other

Receivers:
○ 1
● 2

Sample Rate:
● 48000
○ 96000
○ 192000
○ 384000

☑ ALEX
☐ APOLLO

RIT step (Hz)
○ 1
● 10
○ 100

VFO Encoder Divisor: 15

AF: 10
AGC: 80
ATT (dB): 0
Mic (dB): 0
Drive: 50
Squelch: 0

Receivers - changing pihpsdr to 1 or 2 receivers.

Region – NOTE: Region is UK or Other. Be sure to select your Region.

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 60Mhz spectrum data from the Apache Transceiver.**

Note: Protocol 1 provides for an identical sample rate for both receivers, shown in the Menu → Radio

Protocol 2 - Receivers can have individual sample rates, shown in the Menu → Rx0/1

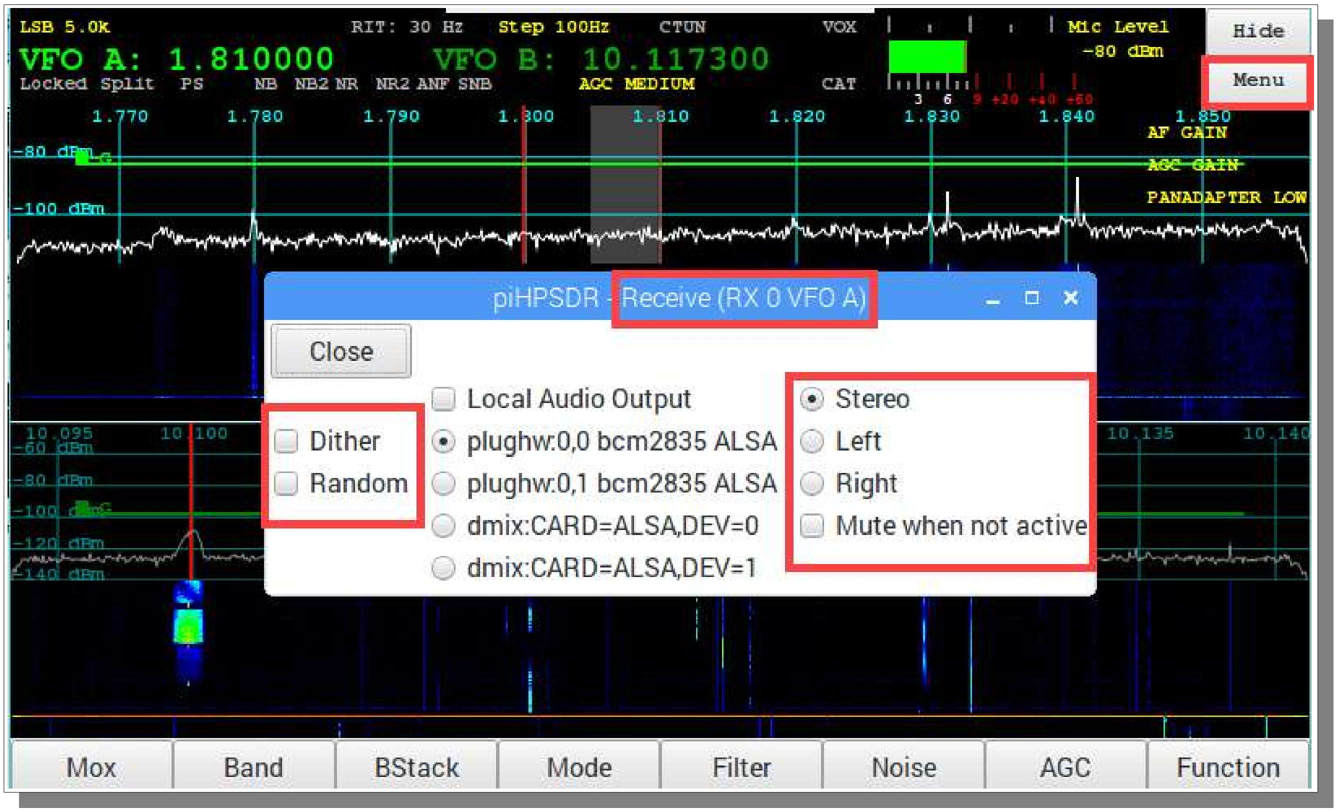
Note: 768k and 1.536K are not available on Rpi.

RIT Step size – This value is the increment or step size for the RIT control.

VFO Encoder Divisor is set at the factory to 15
set to 15, 1 revolution of the (600ppr) encoder = 4kHz/revolution
setting the Encoder Divisor to 30 slows the VFO Encoder tuning to 2kHz/revolution.
setting the Encoder Divisor to 7 increased the VFO Encoder tuning to 9kHz/revolution.

Menu → Receive (individual receiver settings)

When you touch or select either PANADAPTER you change to that VFO and the RX menu changes from Rx0 to Rx1.



piHPSDR - Receive (RX 0 VFO A)

Close

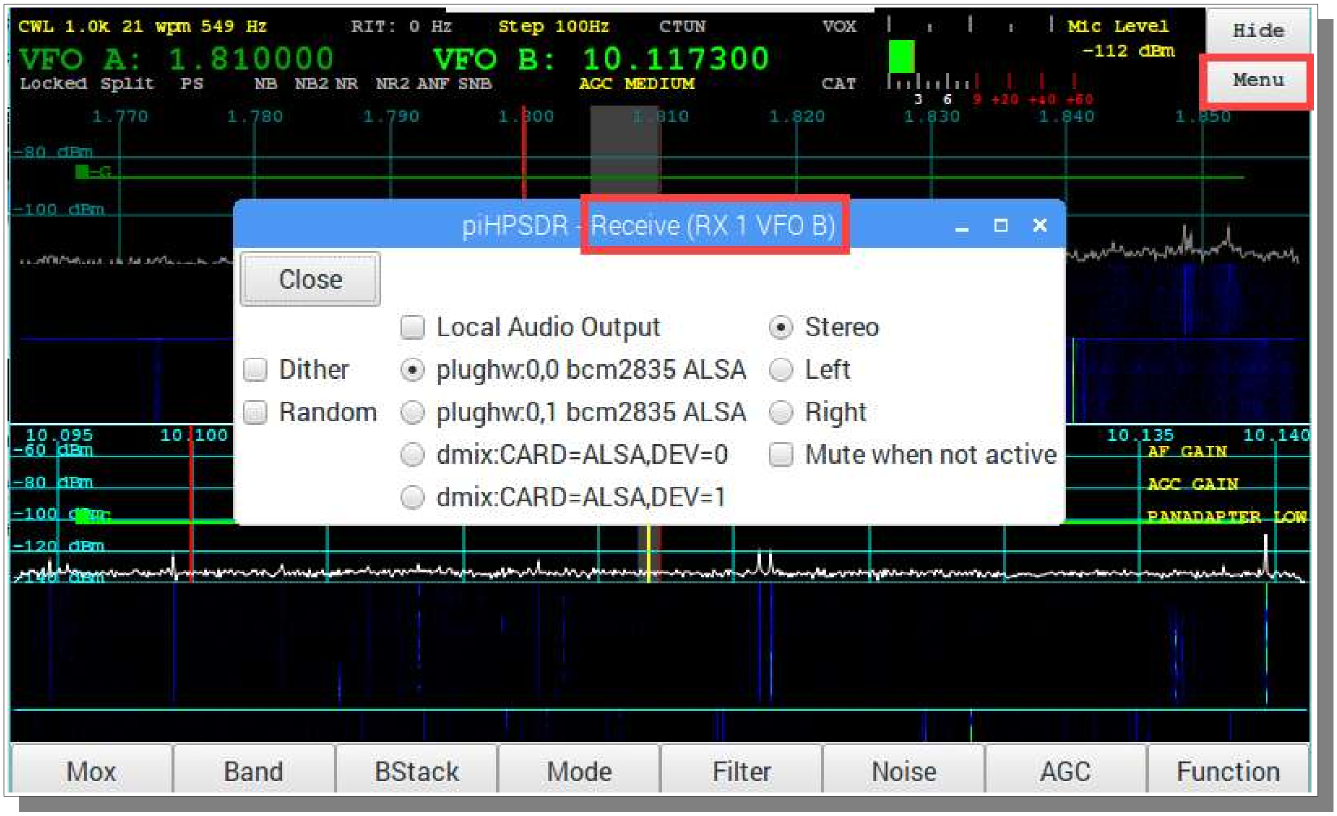
☐ Dither
☐ Random

☐ Local Audio Output
● plughw:0,0 bcm2835 ALSA
○ plughw:0,1 bcm2835 ALSA
○ dmix:CARD=ALSA,DEV=0
○ dmix:CARD=ALSA,DEV=1

● Stereo
○ Left
○ Right
☐ Mute when not active

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 - Random is a built-in electronic feature of the Linear Technologies LTC-2208 Analog to Digital Converter chip inside each Apache Software Defined Radio Transceiver.
Local Audio Output – This column identifies all the audio output connections that pihpsdr recognizes when it starts.
By default the audio output from the receiver is sent back to the radio for output to the audio connections on the radio.
You can also output the audio to either the audio output connector on the Raspberry Pi or a USB connected audio device.

To enable output to a selected device check the Local Audio Output check box. The output will be directed to the selected device.
plughw:0,0 bcm2835 ALSA is the Raspberry Pi output to the audio connector.
plughw:0,1 bcm2835 ALSA is the Raspberry Pi output to the HDMI interface.
plughw:1.0 USB Audio CODEC is a USB connected audio device.
Only one output stream can be directed to the devices listed above, but the ALSA sound system includes a mixer that lets multiple streams output to a single device.
dmix:CARD=ALSA,DEV=0 is a mixer for the plughw:0,0 bcm2835 ALSA output
dmix:CARD=ALSA,DEV=1 is a mixer for the plughw:0,1 bcm2835 ALSA output
dmix:CARD=CODEC,DEV=0 is a mixer for the plughw:1.0 USB Audio CODEC output
Stereo, Left, Right, Mute when not active
If you select the same dmix device for both receivers, the output from both will be mixed and sent to that device. You can select that the receivers output is either Stereo, Left or Right to select the channel(s) that the audio is sent on. This would allow one receiver inn the Left headphone/speaker and the other receiver in the Right headphone/speaker.



piHPSDR - Receive (RX 1 VFO B)

Close

☐ Local Audio Output
● Stereo
☐ Dither
● plughw:0,0 bcm2835 ALSA
○ Left
☐ Random
○ plughw:0,1 bcm2835 ALSA
○ Right
○ dmix:CARD=ALSA,DEV=0
☐ Mute when not active
○ dmix:CARD=ALSA,DEV=1

The Mute when not active when selected will simulate the output to the radio where only the active receiver is output to the device.

Note that if one receiver's audio is set to output to the plughw device the other receiver cannot be connected to the dmix device for that audio output.

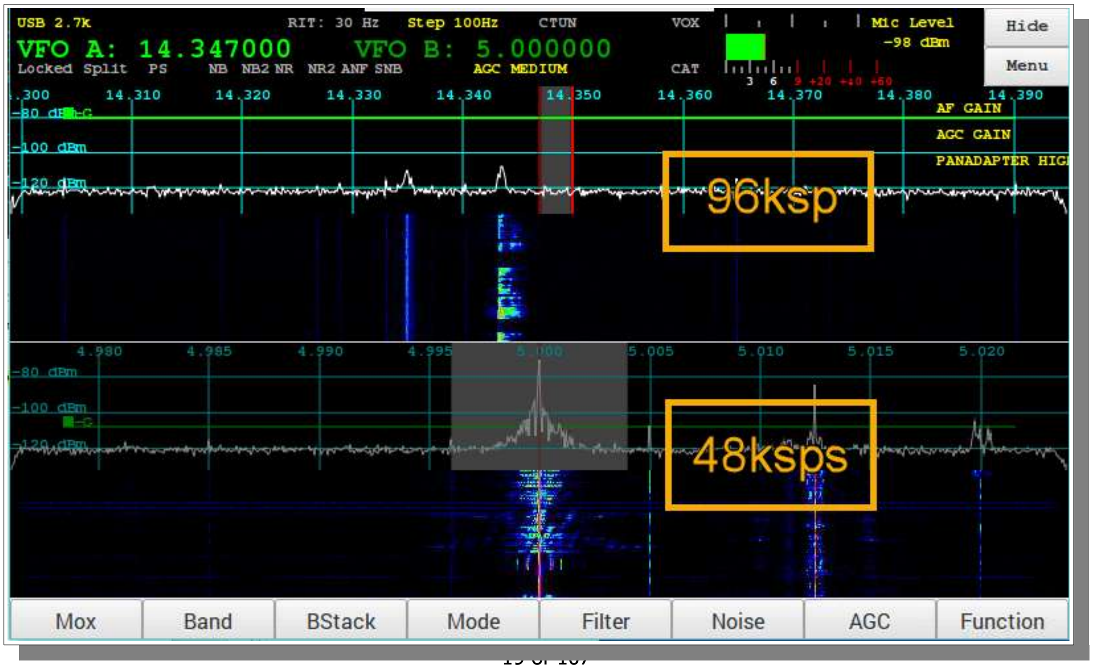
(protocol 1 = single sample rate, protocol 2 = independent sample rates)

Illustrates individual sample rates with new Ethernet protocol

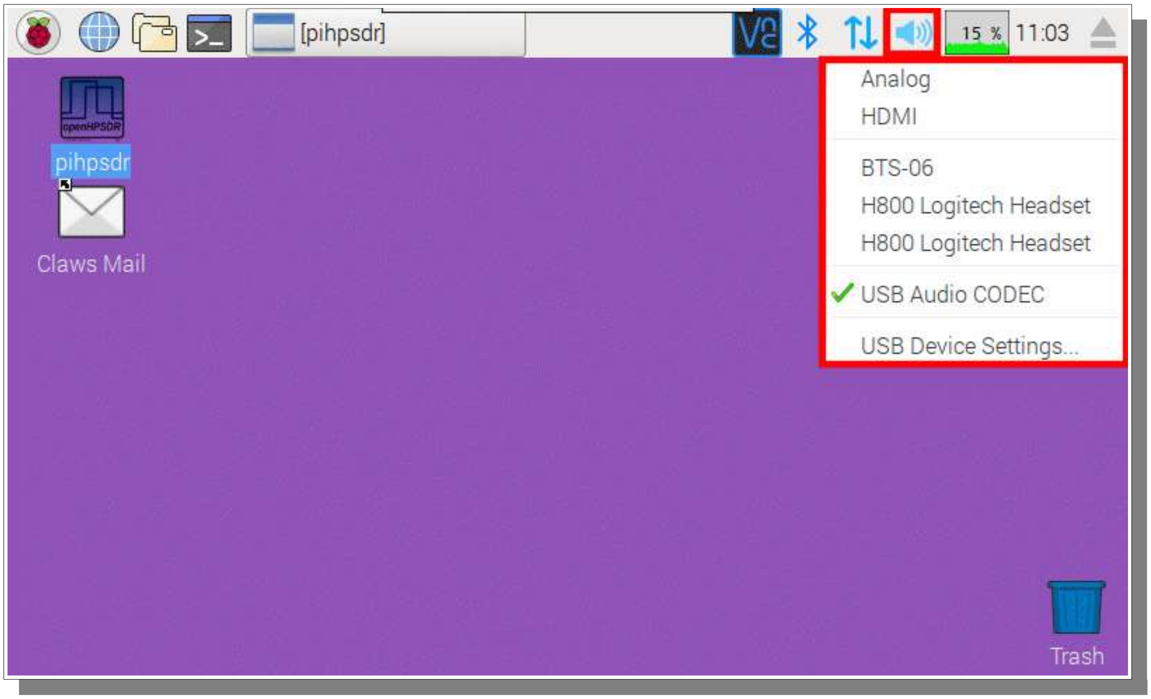
Note: Protocol 1 provides for an identical sample rate for both receivers, shown in the Menu → Radio

Protocol 2 - Receivers can have individual sample rates, shown in the Menu → Rx0/1

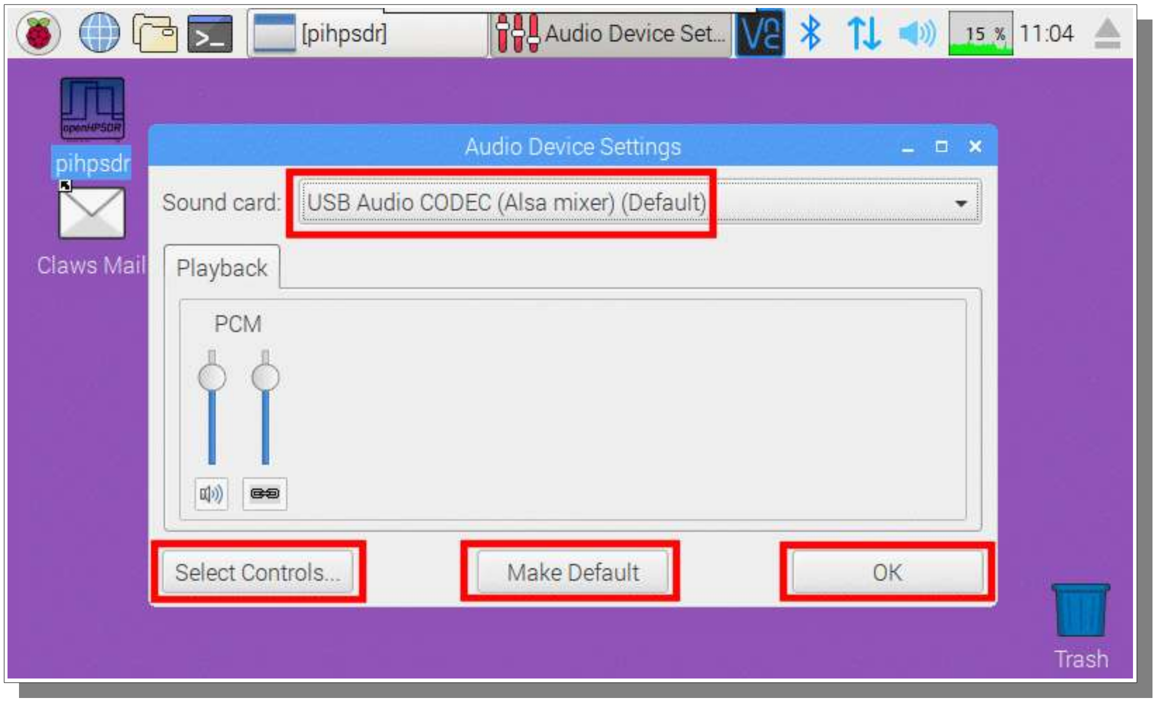
768k and 1.536K are not available on Rpi.



RaspberryPi (Audio Device Settings)



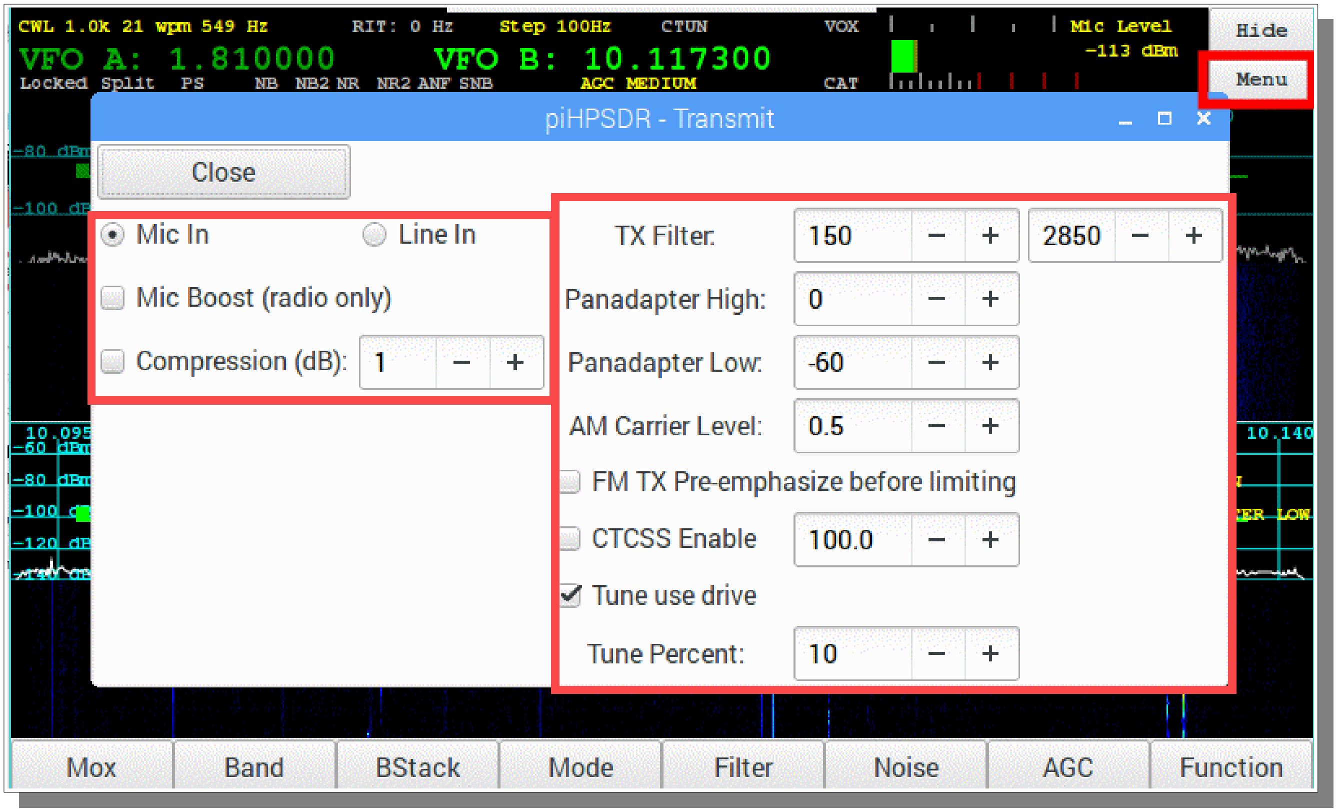
Note RIGHT click on RaspberryPi taskbar **AUDIO ICON** to assign RPi in/out devices



Sound card: USB Audio CODEC (Alsa mixer) (Default)

Note: USB DEVICE SETTINGS (from RPi taskbar ICON) detailed information

Menu → TX with USB Input Source



● Mic In ○ Line In

☐ Mic Boost (radio only)

☐ Compression (dB): 1

TX Filter: 150 2850

Panadapter High: 0

Panadapter Low: -60

AM Carrier Level: 0.5

☐ FM TX Pre-emphasize before limiting

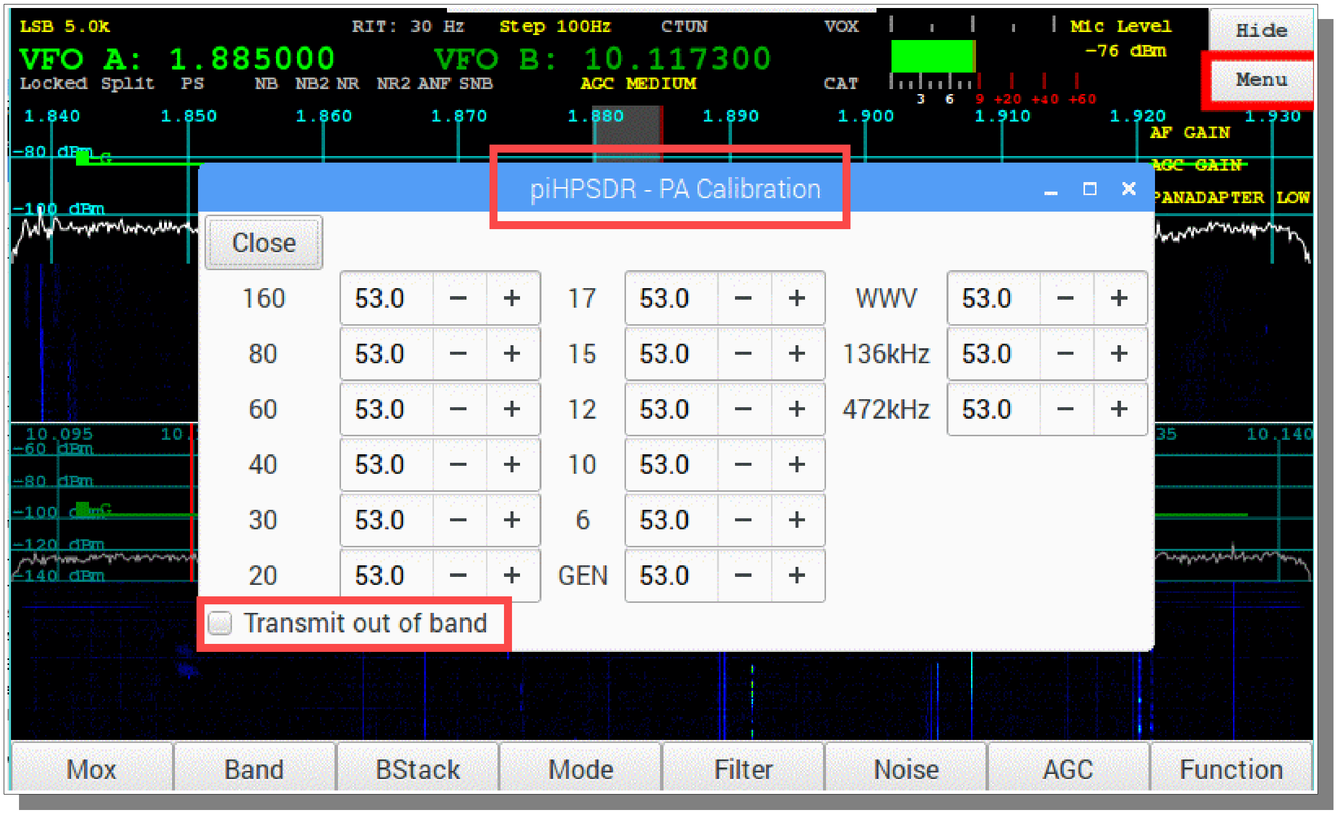
☐ CTCSS Enable 100.0

☑ Tune use drive

Tune Percent: 10

- Mic Boost (20db boost for transceiver front panel connection).
- Mic Line In (Transceiver ACC connector).
- Mic Compression (0 to 20db compression).
- Tx Filter width.
- Tx Panadapter display High, Tx Panadapter display Low.
- AM Carrier Level.
- FM Tx Pres-emphasis before limiting.
- CTCSS Enable and Frequency.
- Tune - Use Drive level (when checked Tune will use the drive setting, when not checked it will use the percentage of drive).
- Tune Percent of Drive (if box not checked)

Menu → PA Gain by Band



160	53.0	17	53.0	WWV	53.0
80	53.0	15	53.0	136kHz	53.0
60	53.0	12	53.0	472kHz	53.0
40	53.0	10	53.0		
30	53.0	6	53.0		
20	53.0	GEN	53.0		

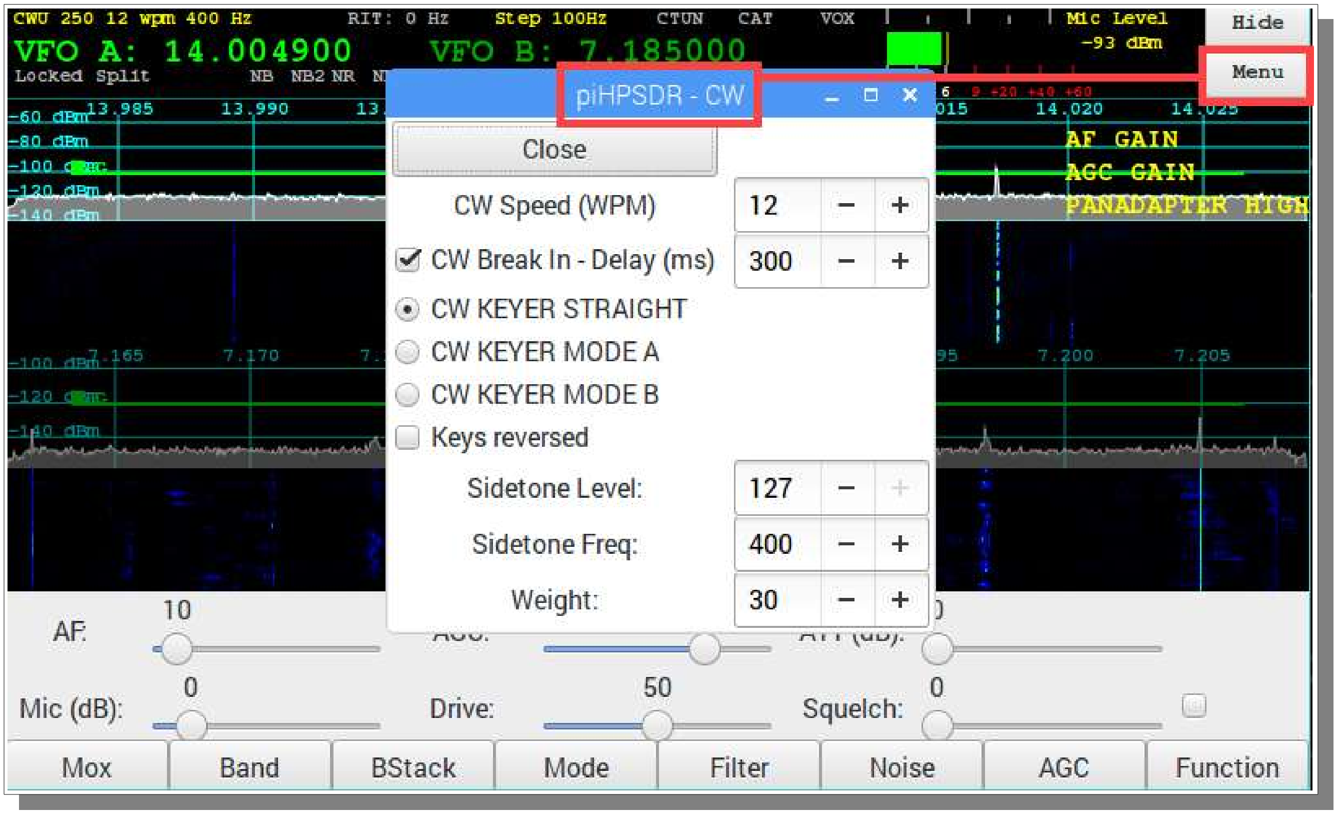
☐ Transmit out of band

- **Sets the PA gain for each band by adjusting the drive level. This adjustment allows you to set the Transceiver for maximum rated Transmit output.**
- **the setting are increased the drive level is decreased. This adjustment is set for maximum rated output protect the PA Final Amplifier.**

Caution: The User should follow the instructions for PA Gain settings in PowerSDR/Thetis. These instructions define setting the TUNE POWER to the Maximum Rated Output while adjusting the PA Gain settings. Operationally the user can use the pihpsdr Drive slider to adjust for a lower output power – for example when driving a Linear Amplifier.

- Note**: Transmit out of band – enables transmit outside of ham bands (for example MARS operation) or for use with external transverter with IF on non Ham bands.
Default disabled.**

Menu → CW



Close

CW Speed (WPM) 12

☑ CW Break In - Delay (ms) 300

● CW KEYER STRAIGHT

○ CW KEYER MODE A

○ CW KEYER MODE B

☐ Keys reversed

Sidetone Level: 127

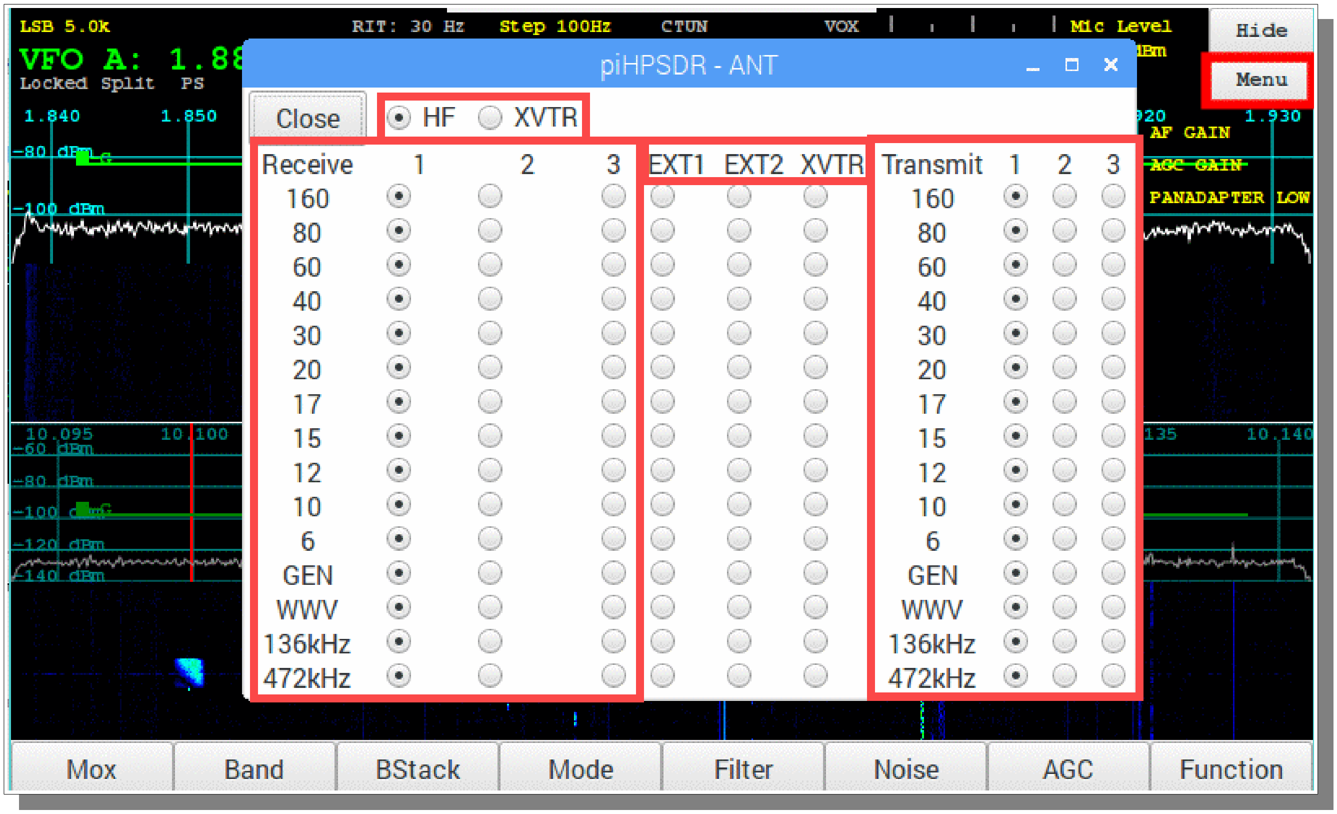
Sidetone Freq: 400

Weight: 30

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 at the Headphone or Line Out of the Apache Transceiver.**
- **Sidetone Freq – set the frequency of the sidetone at the Headphone or Line Out of the Apache Transceiver.**
- **Weight – sets the dot/dash weighting.**

Menu → ANT

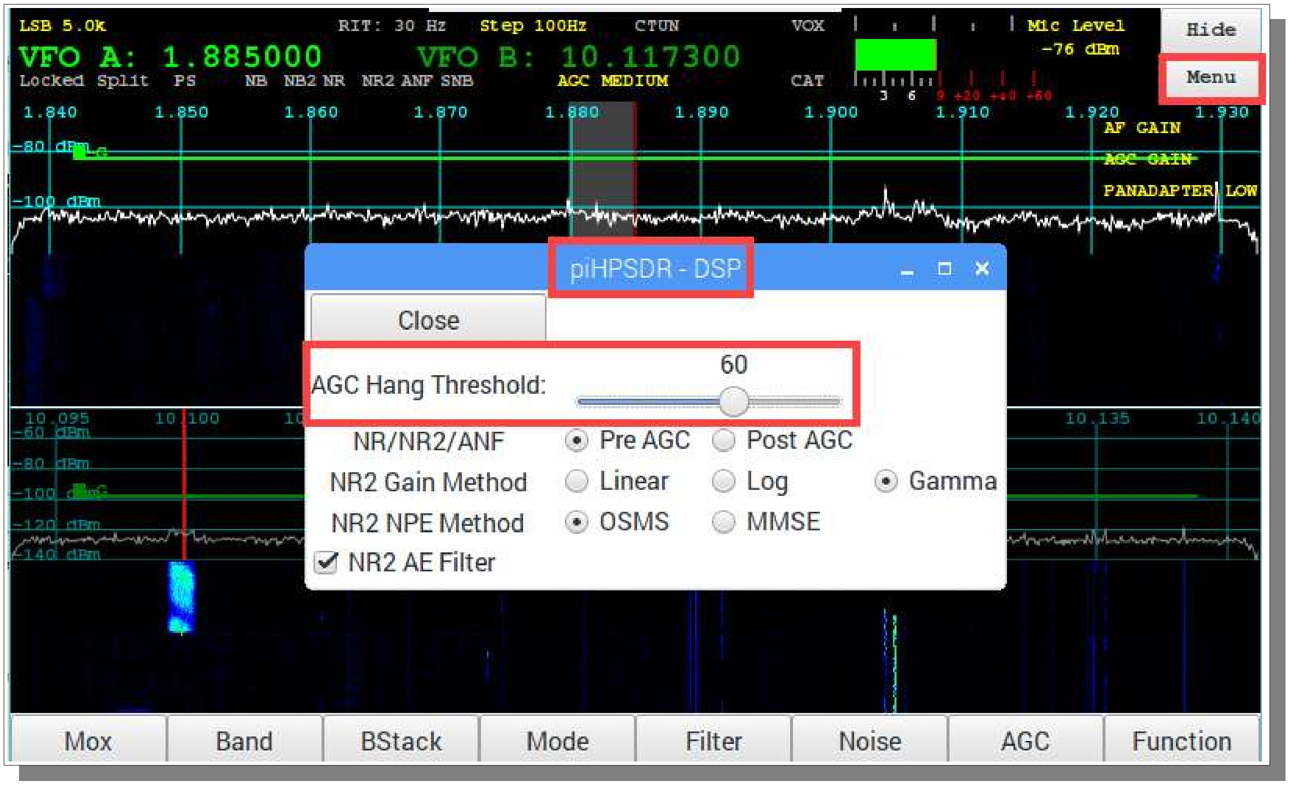


Close ● HF ○ XVTR

Receive	1	2	3	EXT1	EXT2	XVTR	Transmit	1	2	3
160	●	○	○	○	○	○	160	●	○	○
80	●	○	○	○	○	○	80	●	○	○
60	●	○	○	○	○	○	60	●	○	○
40	●	○	○	○	○	○	40	●	○	○
30	●	○	○	○	○	○	30	●	○	○
20	●	○	○	○	○	○	20	●	○	○
17	●	○	○	○	○	○	17	●	○	○
15	●	○	○	○	○	○	15	●	○	○
12	●	○	○	○	○	○	12	●	○	○
10	●	○	○	○	○	○	10	●	○	○
6	●	○	○	○	○	○	6	●	○	○
GEN	●	○	○	○	○	○	GEN	●	○	○
WWV	●	○	○	○	○	○	WWV	●	○	○
136kHz	●	○	○	○	○	○	136kHz	●	○	○
472kHz	●	○	○	○	○	○	472kHz	●	○	○

The Ant menu selects which antenna is used for receive and transmit on each band. The Antenna Menu will differ depending on the specific Apache Transceiver Model. This illustration shows the user selection of up to 3 Receive and Transmit Antennas. The EXT-1 and EXT-2 selections are available on the ANAN-100/100B/100D/200D.

Menu → DSP



AGC Hang Threshold: 60

NR/NR2/ANF ● Pre AGC ○ Post AGC

NR2 Gain Method ○ Linear ○ Log ● Gamma

NR2 NPE Method ● OSMS ○ MMSE

☑ NR2 AE Filter

The DSP has options for the DSP functions.

- **AGC Hang Threshold – sets the Hang Threshold for the AGC.**

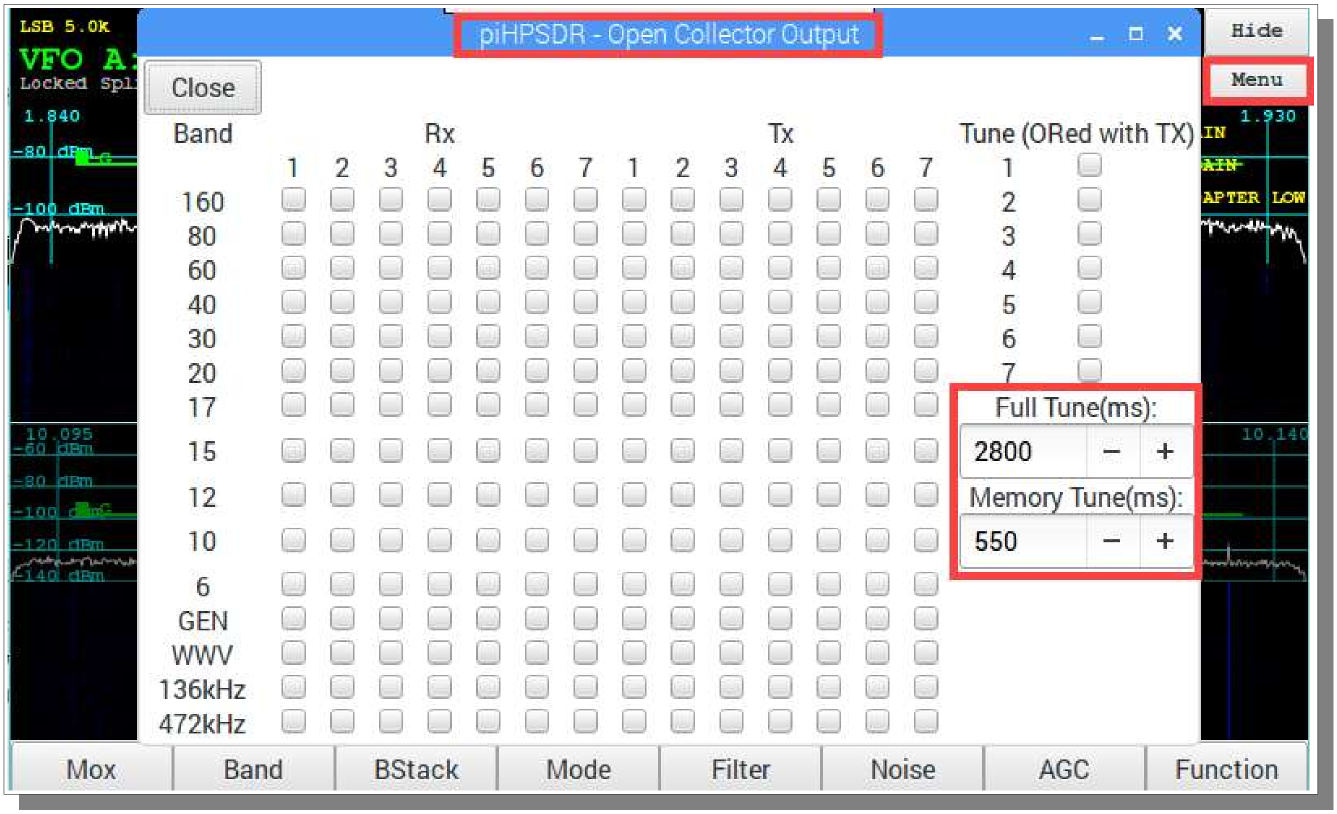
- **NR/NR2/ANF – selects where in the DSP processing the noise reduction functions are performed/ The default id Pre AGC processing.**
  Pre AGC – perform noise reduction pre AGC
  Post AGC – perform noise reduction post AGC

- **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 → OC Open Collector Aux I/O connectors



Band	Rx							Tx						
	1	2	3	4	5	6	7	1	2	3	4	5	6	7
160	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
80	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
60	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
40	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
30	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
20	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
17	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
15	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
12	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
10	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
6	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
GEN	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
WWV	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
136kHz	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
472kHz	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Tune (ORed with TX)	
1	☐
2	☐
3	☐
4	☐
5	☐
6	☐
7	☐

Full Tune(ms): 2800 − +

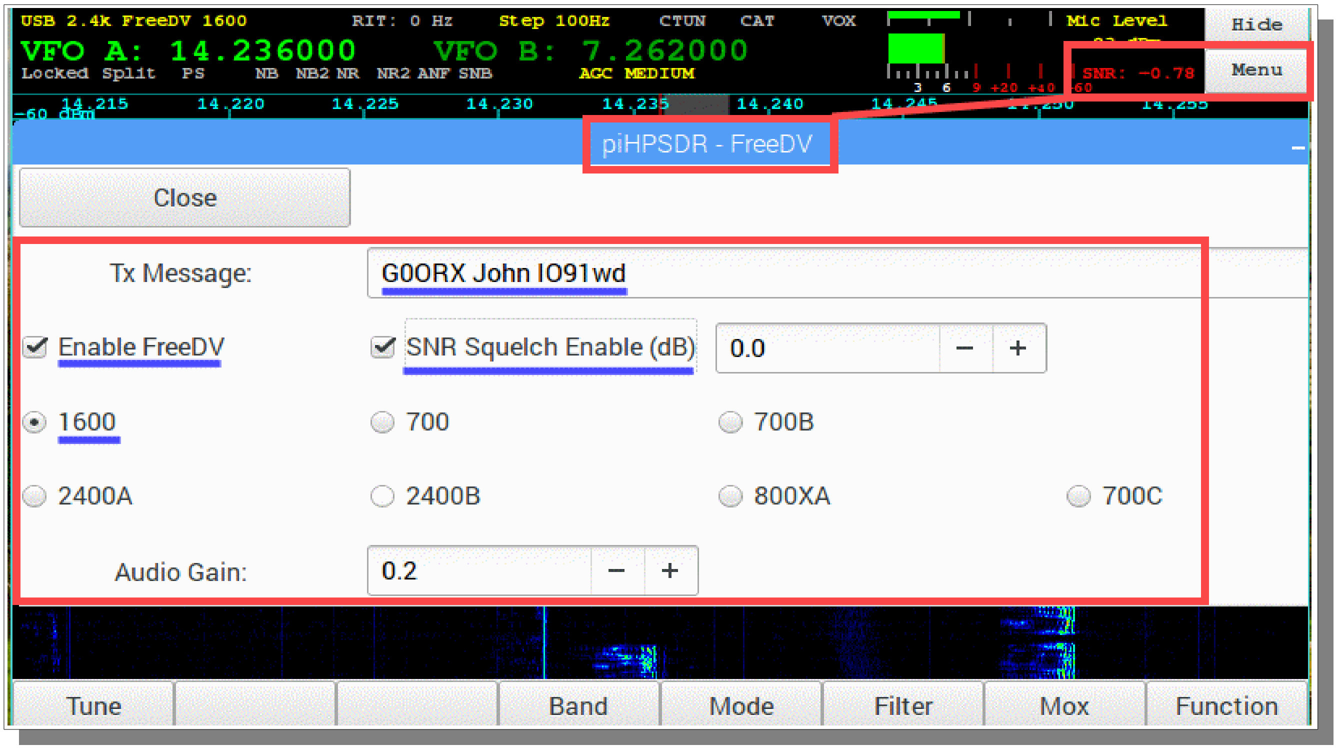
Memory Tune(ms): 550 − +

The user can configure the Open Collector outputs for each band for both Transmit and Receive. These can be used to control an external device such as bandpass filters or external Linear Amplifier band selection. The default is none are enabled.

Refer to the Apache Labs Users Guides for details and limitations of using Open Collector FET outputs. 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 Tx is enabled.
- **Full Tune milliseconds –** specifies the time the OC is enabled when the Tuning.
- **Memory Tune milliseconds –** keydown time for TUNE when using an externa**l Automatic Antenna Tuner**

Menu → FreeDV



Tx Message: G0ORX John IO91wd

☑ Enable FreeDV

☑ SNR Squelch Enable (dB) 0.0

● 1600 ○ 700 ○ 700B

○ 2400A ○ 2400B ○ 800XA ○ 700C

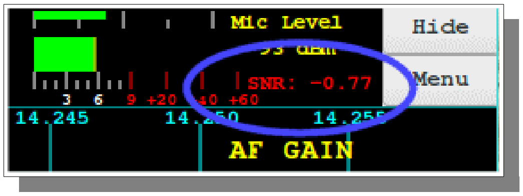
Audio Gain: 0.2

FreeDV: Open Source Amateur Digital Voice by The FreeDV.org. piHPSDR uses the FreeDV API. FreeDV provides a number of codecs that can run on HF over a typical SSB channel (1600, 700, 700B, 700C an 800XA). It also includes codec for running over FM channels on VHF/UHF (2400A and 2400B).

Speech is compressed down to 700-1600 bit/s then modulated onto a 1.25 kHz wide signal comprised of 16 QPSK carriers. FreeDV 700C is approaching SSB in it's low SNR performance. At high SNRs FreeDV 1600 sounds like FM, with no annoying analog HF radio noise.

Enable FreeDV - when enabled microphone input is passed to the FreeDV library for encoding and the output is passed into WDSP for processing into I/Q signals for transmitting. On receive the audio output from WDSP if passed to the FreeDV library to decode the audio.

SNR Squelch Enable - when enabled uses the SNR value to set the squelch. SNR shown in RED letters next to S-Meter. Adjust as needed to achieve quiet idle.



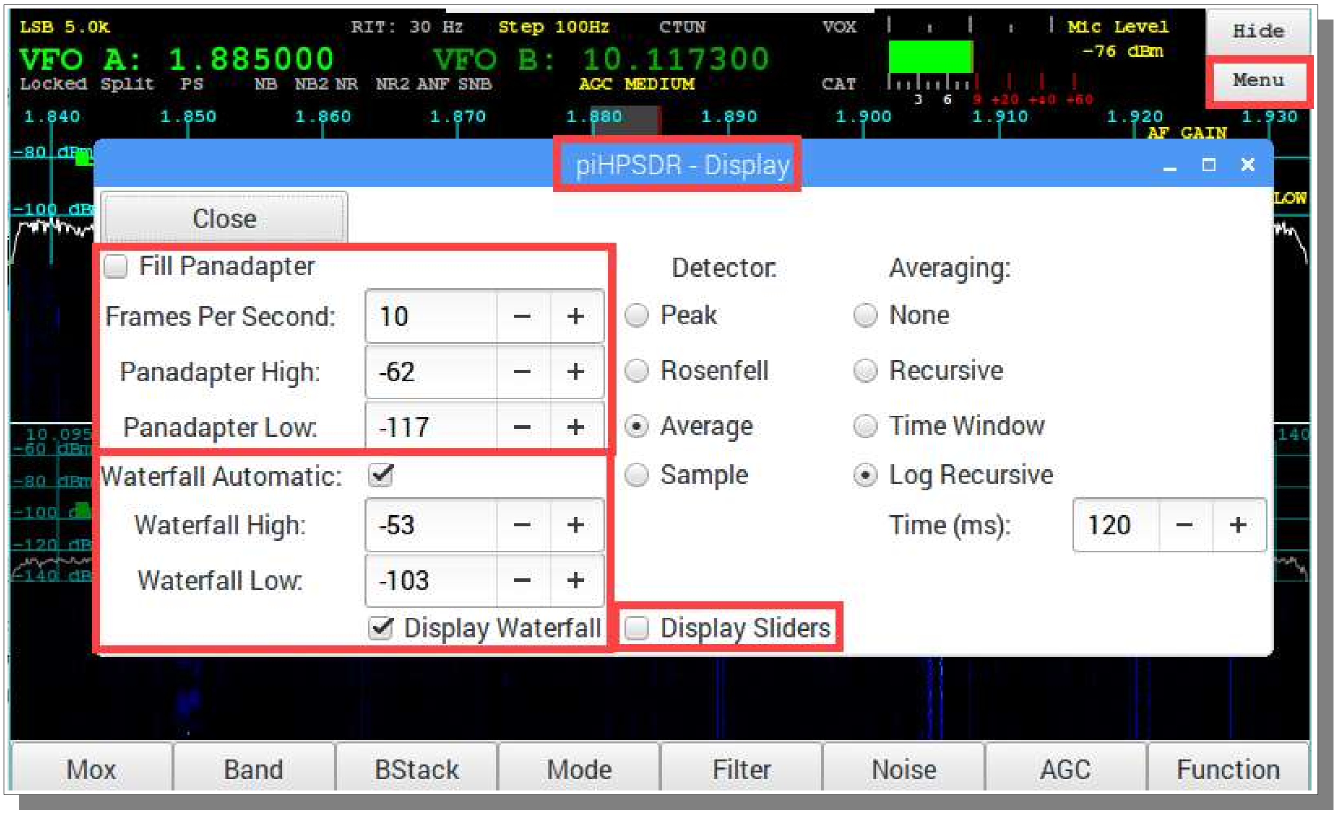
CODEC select

1600	select codec to use for HF
700	" "
700B	" "
700C	" "
800XA	" "
2400A	select codec to use for VHF/UHF
2400B	" "

Audio Gain sets the gain of the audio output from the codec passed to WDSP when transmitting.

Tx Message - specifies the text message that is transmitted along with the digital voice.

Menu → Display



☐ Fill Panadapter
Frames Per Second: 10
Panadapter High: -62
Panadapter Low: -117
Waterfall Automatic: ☑
Waterfall High: -53
Waterfall Low: -103
☑ Display Waterfall
☐ Display Sliders

Detector:
○ Peak
○ Rosenfell
● Average
○ Sample

Averaging:
○ None
○ Recursive
○ Time Window
● Log Recursive
Time (ms): 120

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

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

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

Note: now included as a sub-menu selection for E1, E2, E3 Encoder function assignment

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

Note: now included as a sub-menu selection for E1, E2, E3 Encoder function assignment

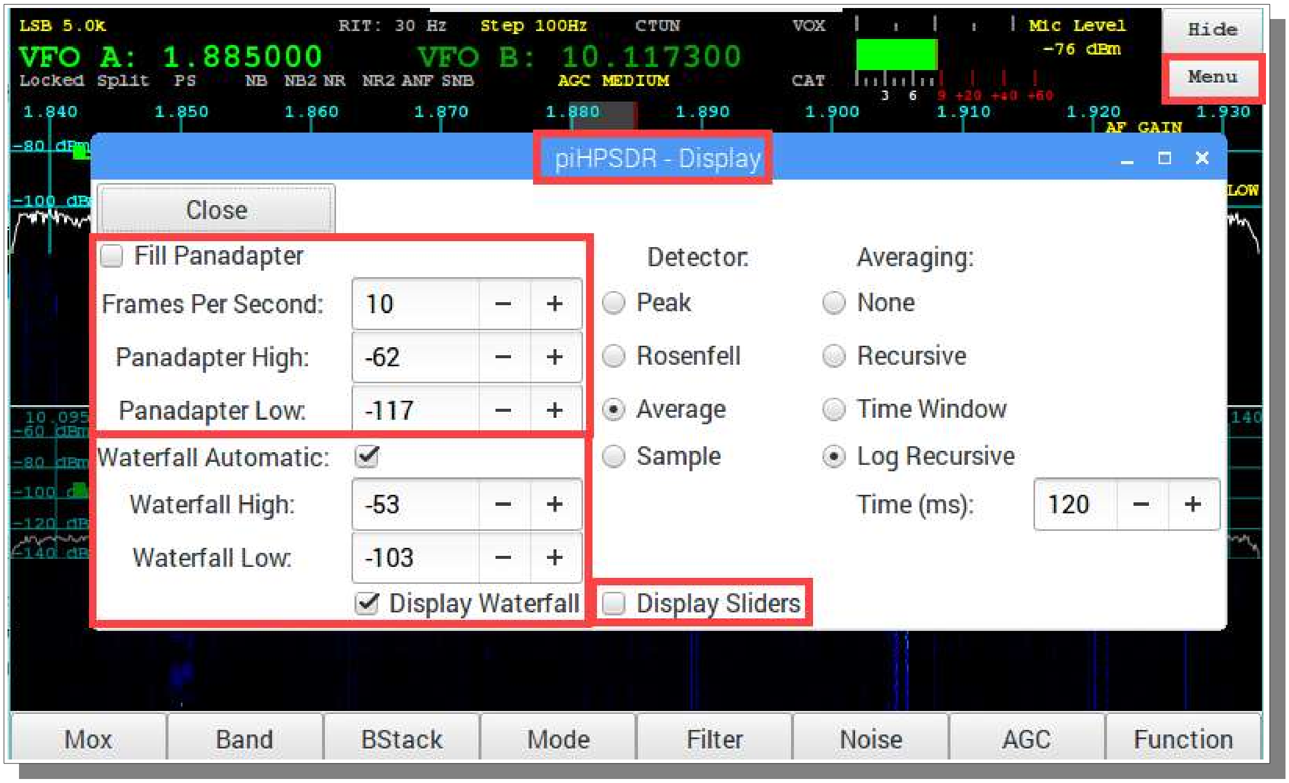
- **Waterfall Automatic – When enabled the Waterfall High and Waterfall Low are adjusted automatically.**

- **Waterfall High – Manual control to set the maximum signal used in the waterfall.**

- **Waterfall Low – Manual control to set set the minimum signal used in the waterfall.**

- **Display Sliders – displays the six on screen sliders**

Menu → Display (continued)



piHPSDR - Display

Close

☐ Fill Panadapter

Frames Per Second: 10

Panadapter High: -62

Panadapter Low: -117

Waterfall Automatic: ☑

Waterfall High: -53

Waterfall Low: -103

☑ Display Waterfall

Detector:

○ Peak
○ Rosenfell
● Average
○ Sample

Averaging:

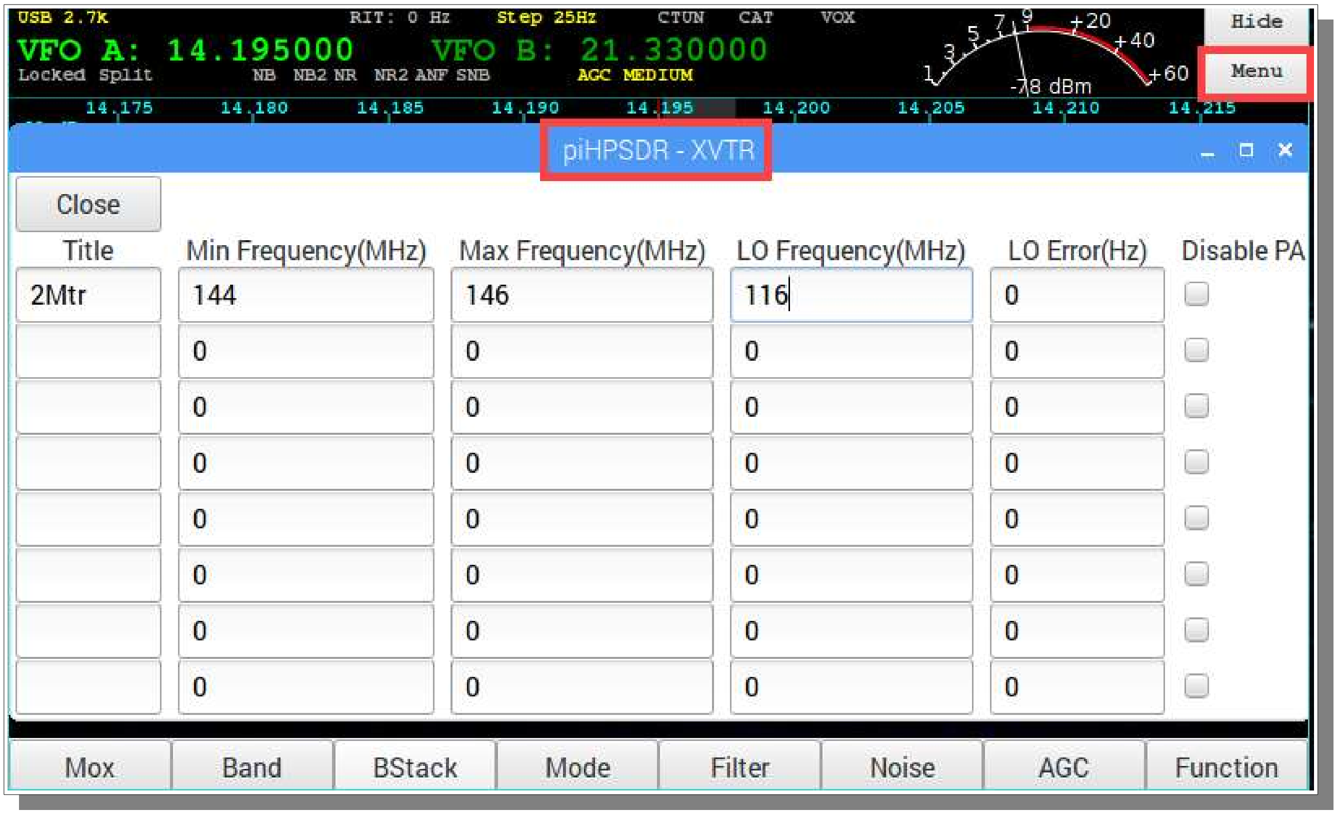
○ None
○ Recursive
○ Time Window
● Log Recursive

Time (ms): 120

☐ Display Sliders

- **Detector – Selects Peak, Rosenfell, Average or Sample for the Panadapter display.**
- **Averaging – Selects the method for averaging the Panadapter display.**
- **Display Waterfall – when selected the Waterfall display is displayed on the main screen.**
- **Display Sliders – when selected the slider controls are displayed as shown on the main screen.**

Menu → XVTR



Title	Min Frequency(MHz)	Max Frequency(MHz)	LO Frequency(MHz)	LO Error(Hz)	Disable PA
2Mtr	144	146	116	0	☐
	0	0	0	0	☐
	0	0	0	0	☐
	0	0	0	0	☐
	0	0	0	0	☐
	0	0	0	0	☐
	0	0	0	0	☐
	0	0	0	0	☐

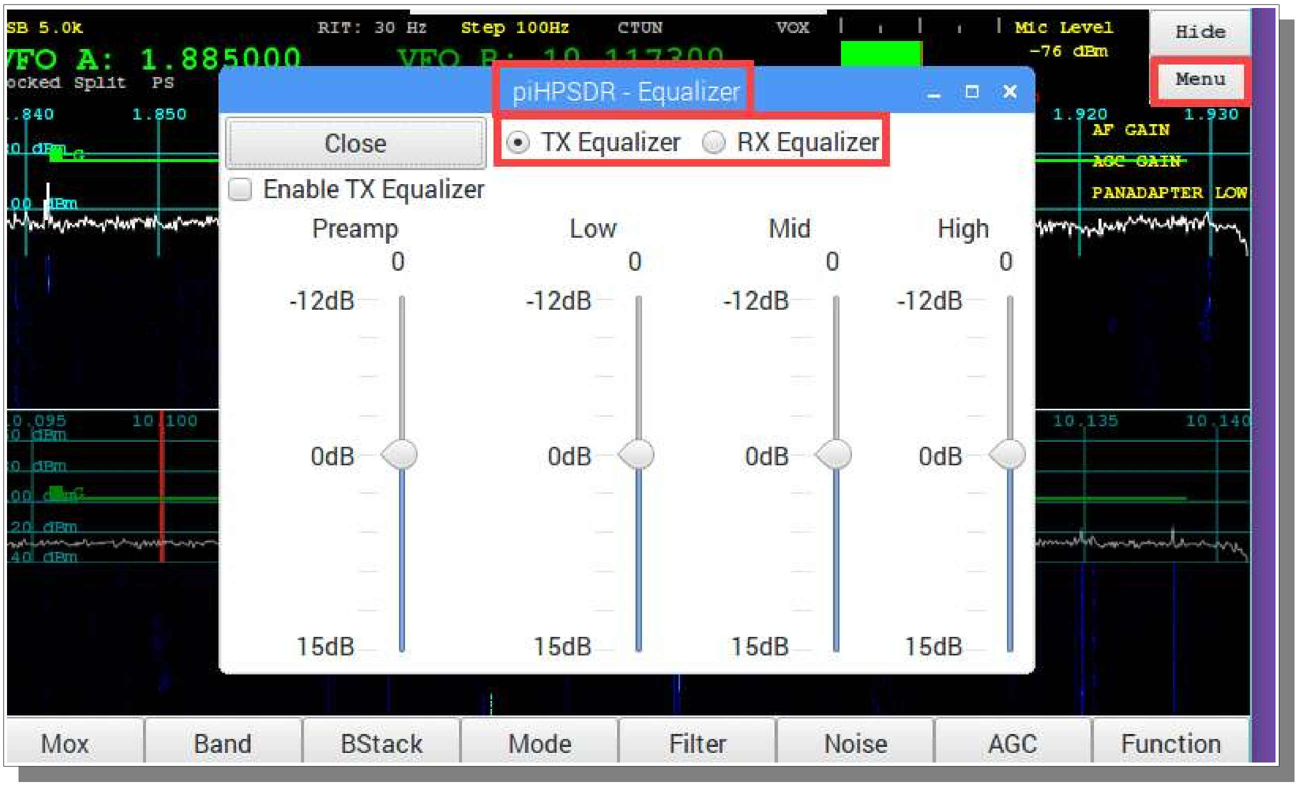
Configure up to 8 transverters

- **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.**
- **LO Error – adjust ± Hz for frequency error correction of transverter local oscillator.**
- **Disable PA – When checked, the Transceiver power amplifier 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.

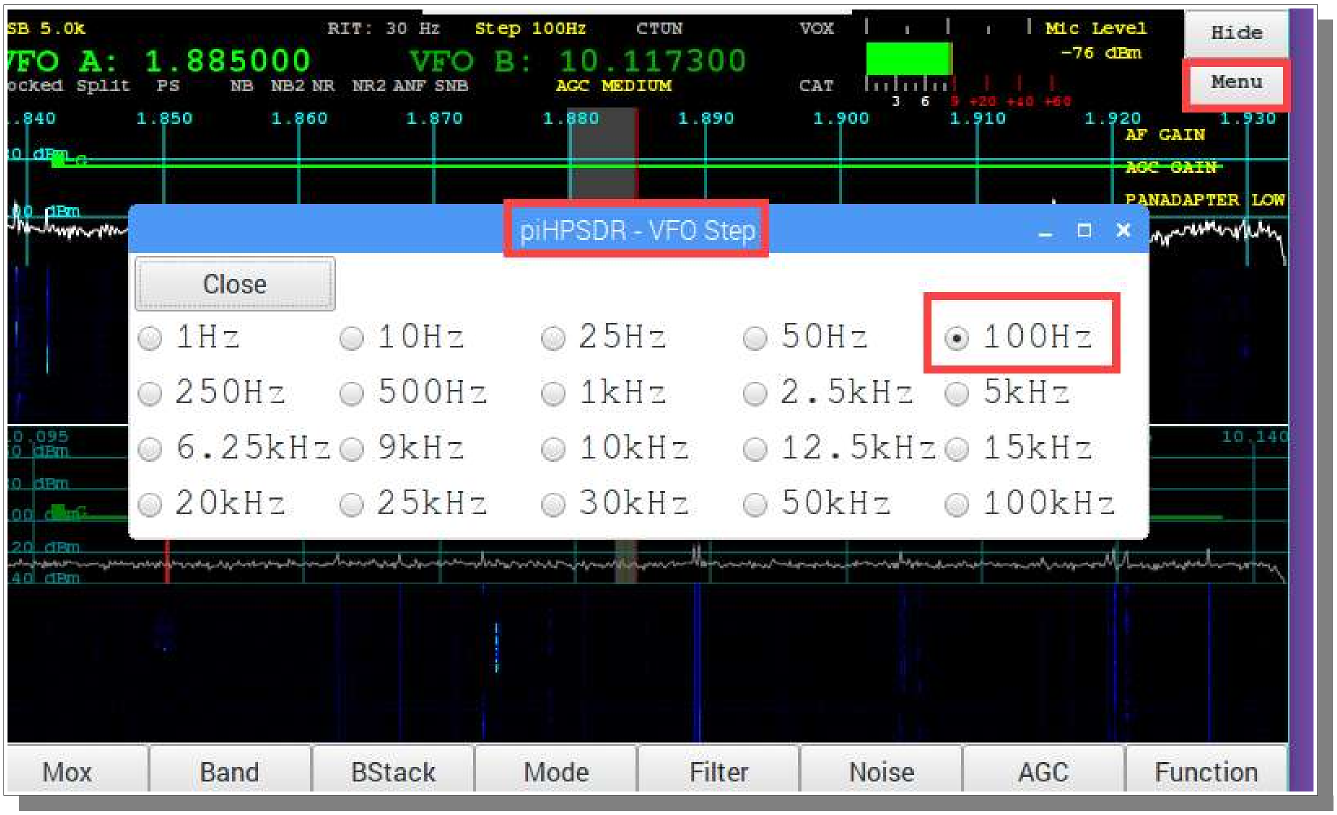
Menu → Equalizers



A 3 band graphic equalizer is implemented for both Transmit and Receive:

- **Preamp**
- **Low – 0-400 Hz slider dB**
- **Mid – 400-1500 Hz slider dB**
- **High – 1500-6000 Hz slider dB.**
- **The radio buttons TX Equalizer and RX Equalizer select which values are displayed.**
  The equalizers can be enabled by the checkbox Enable TX Equalizer or Enable Rx Equalizer.

Menu → Step



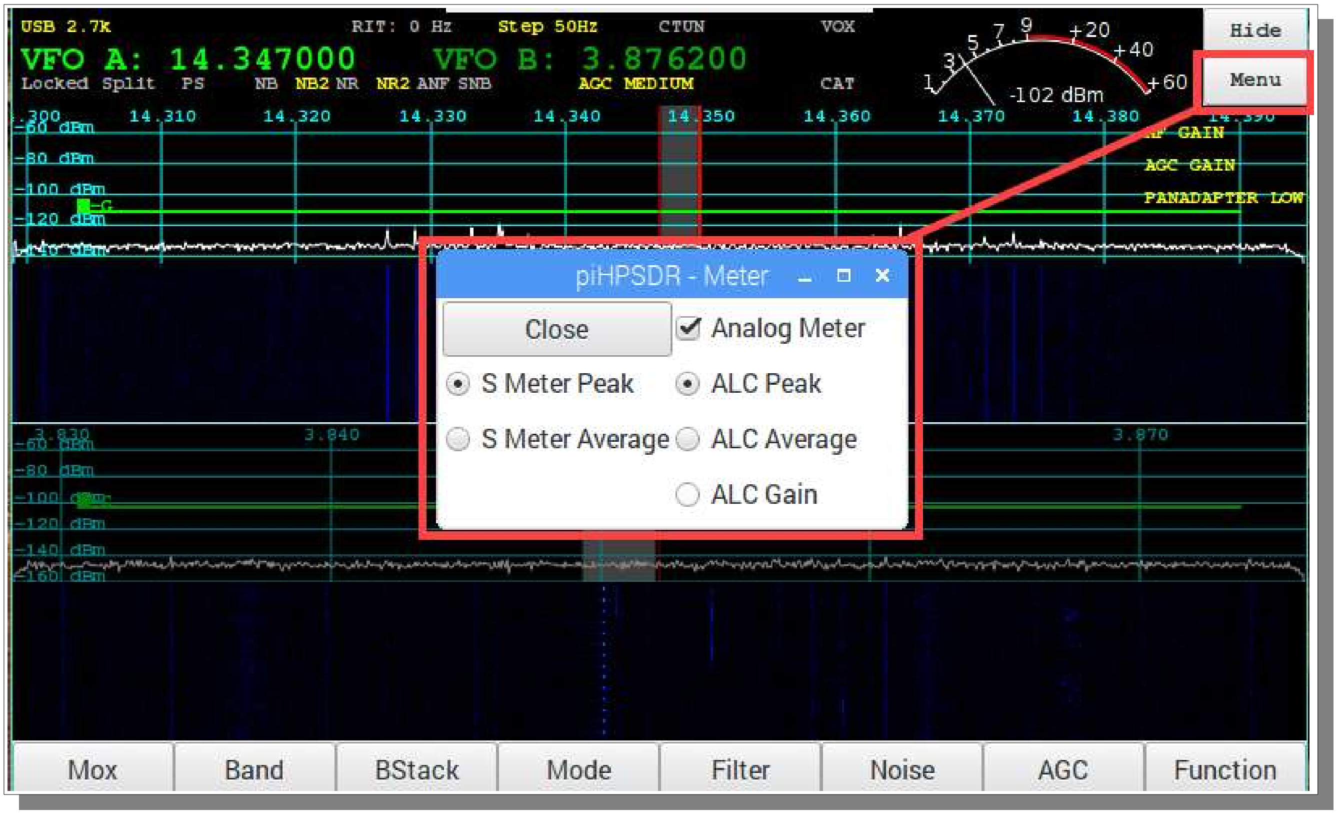
piHPSDR - VFO Step

Close

○ 1Hz	○ 10Hz	○ 25Hz	○ 50Hz	● 100Hz
○ 250Hz	○ 500Hz	○ 1kHz	○ 2.5kHz	○ 5kHz
○ 6.25kHz	○ 9kHz	○ 10kHz	○ 12.5kHz	○ 15kHz
○ 20kHz	○ 25kHz	○ 30kHz	○ 50kHz	○ 100kHz

Step sets the increment for VFO Tuning rate via Touch or VFO (E4) Encoder, and Mouse Wheel.

Menu → Meter – now includes new Analog S-Meter



piHPSDR - Meter

Close

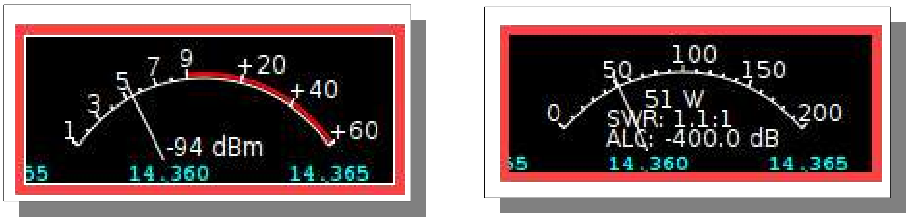
☑ Analog Meter

● S Meter Peak ● ALC Peak

○ S Meter Average ○ ALC Average

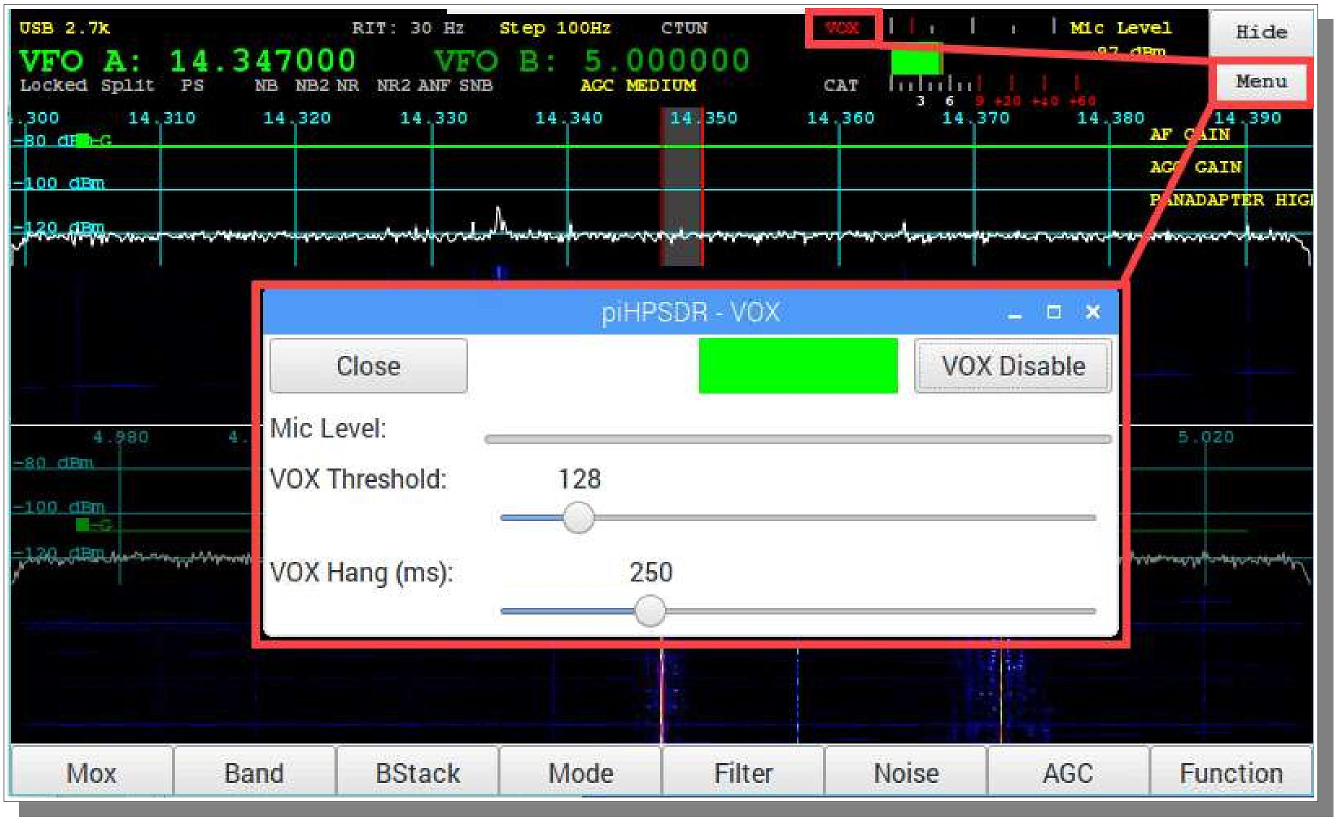
○ ALC Gain

- **NEW S-Meter Analog** meter movement or bar graph
- **S-Meter** Peak and Average
- **ALC** Peak Average or GAIN
- **Meter values appear in the upper right hand corner of the display.**
- You should adjust your **Microphone (dB)** slider so that ALC does not exceed zero on voice peaks.



Examples of Receive Analog S-Meter and Transmit TUNE SWR and Power Display

Menu → VOX

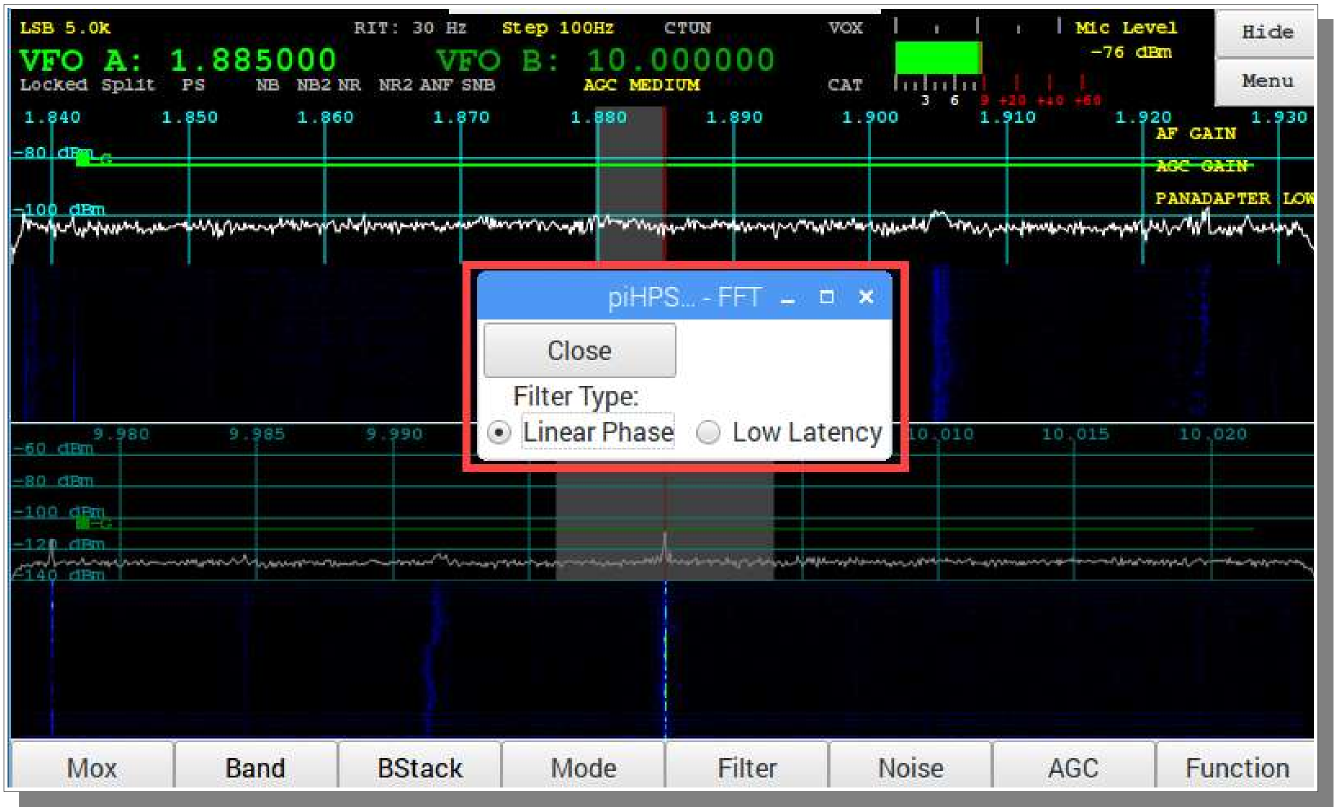


VOX Threshold: 128

VOX Hang (ms): 250

- **Highlighted box indicates Microphone Level as you speak to adjust Threshold and Hang**
- **Threshold** for VOX activation
- **Hang** for how long to hold VOX between words while speaking

Menu → FFT



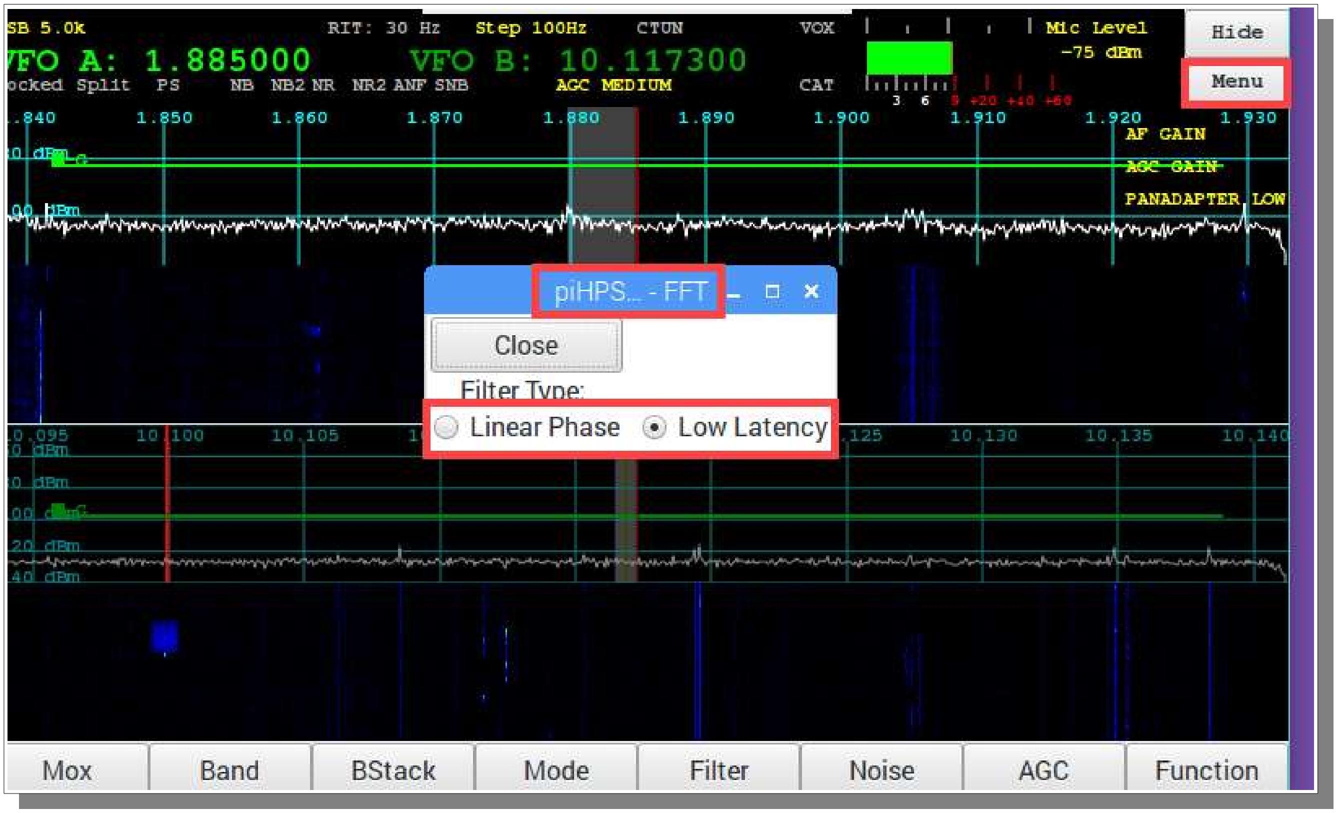
piHPS... - FFT

Close

Filter Type:

● Linear Phase ○ Low Latency

Linear Phase or Low Latency selection



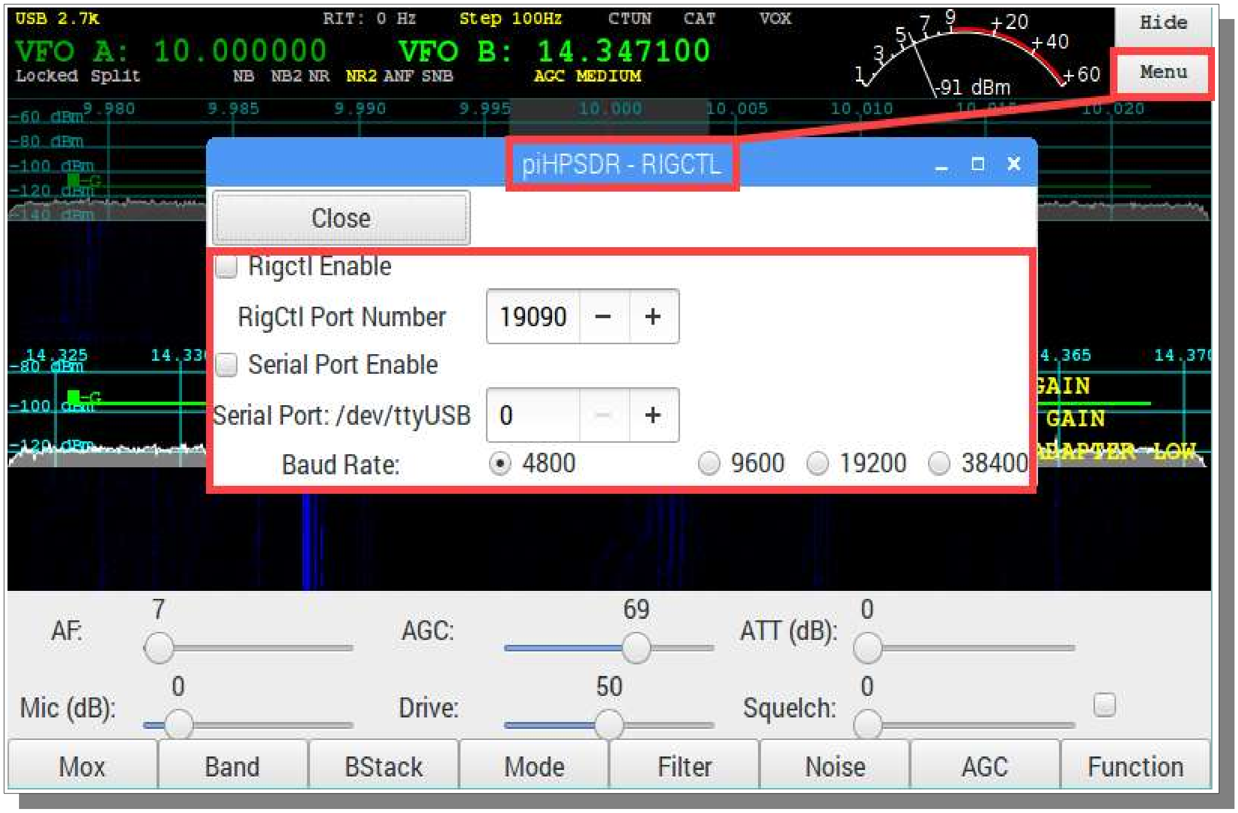
piHPS... - FFT

Close

Filter Type:

○ Linear Phase ● Low Latency

Menu → RIGCTL



☐ Rigctl Enable

RigCtl Port Number 19090

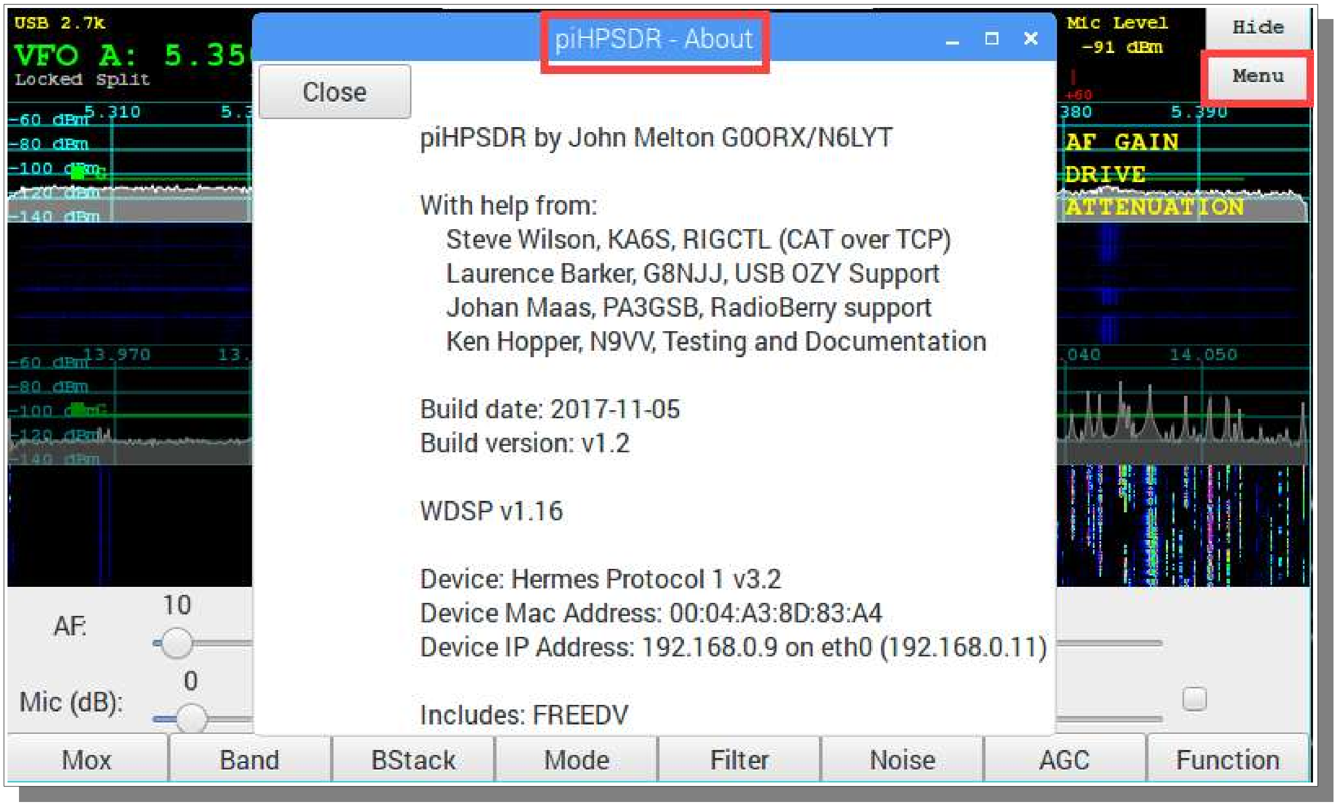
☐ Serial Port Enable

Serial Port: /dev/ttyUSB 0

Baud Rate: ● 4800 ○ 9600 ○ 19200 ○ 38400

Menu → RIGCTL Please refer to special Section 15 page 79 for specifics of the new KA6S Rig Control and Serial port option

Menu → About



piHPSDR by John Melton G0ORX/N6LYT

With help from:
Steve Wilson, KA6S, RIGCTL (CAT over TCP)
Laurence Barker, G8NJJ, USB OZY Support
Johan Maas, PA3GSB, RadioBerry support
Ken Hopper, N9VV, Testing and Documentation

Build date: 2017-11-05
Build version: v1.2

WDSP v1.16

Device: Hermes Protocol 1 v3.2
Device Mac Address: 00:04:A3:8D:83:A4
Device IP Address: 192.168.0.9 on eth0 (192.168.0.11)

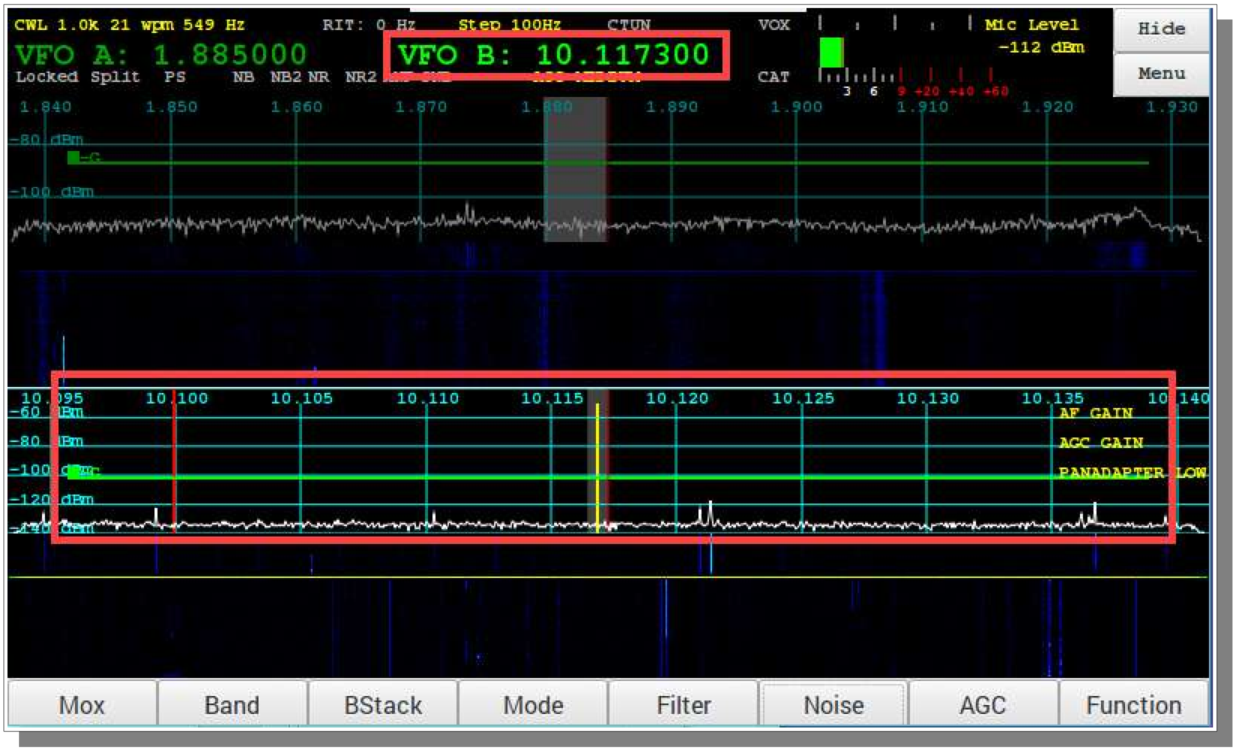
Includes: FREEDV

Note information in Menu → About display

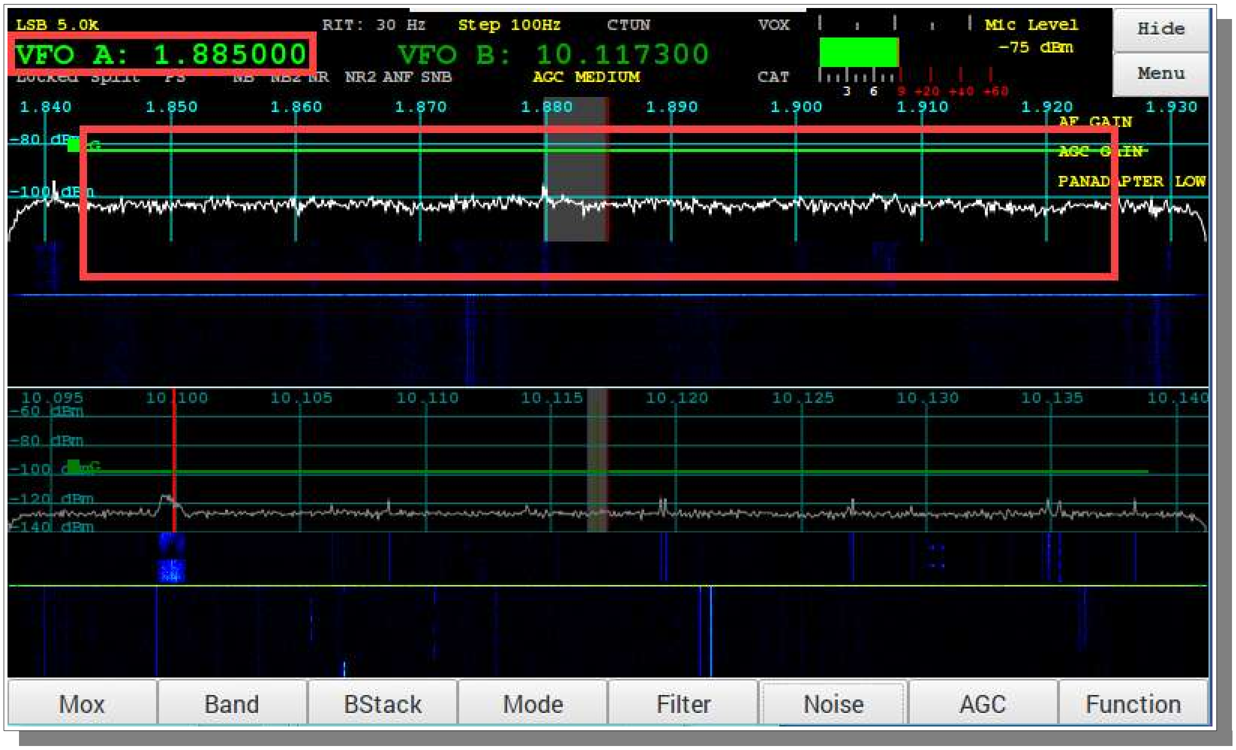
- Build date
- Build Version
- WDSP version
- Device Hermes Protocol
- Device MAC Addresses
- Device IP Addresses
- Includes: FreeDV

8. On-Screen Controls and toolbar Buttons

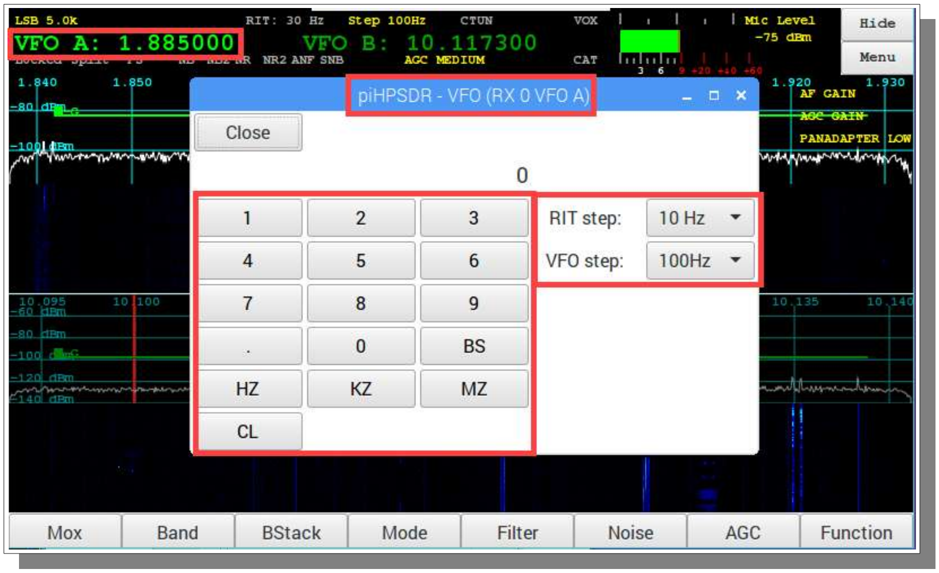
Touch selection of VFO-A (Rx0)Touch selection of VFO-B (Rx1)

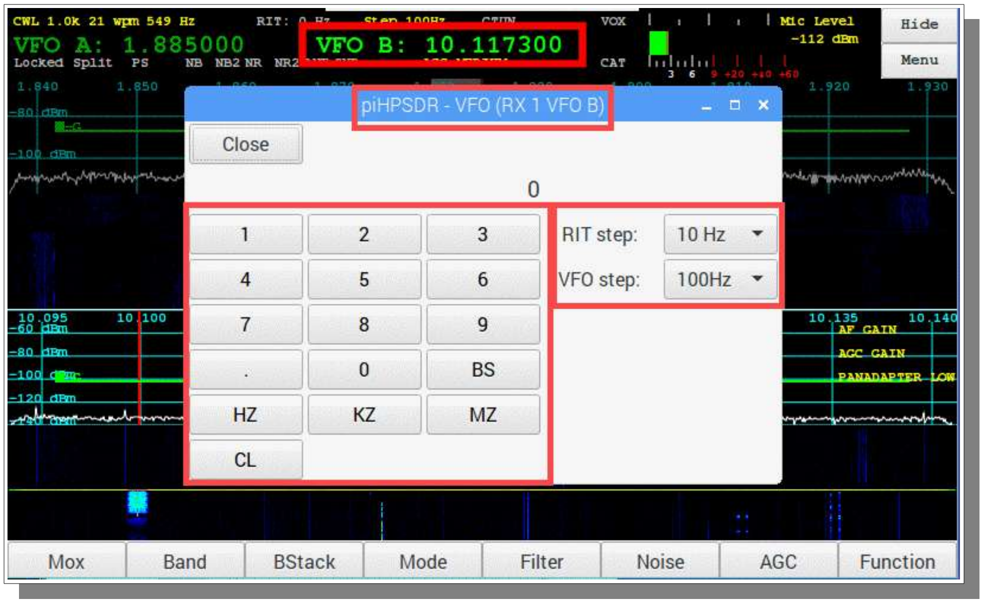


Note: dragging the panadapter image with finger or mouse changes VFO frequency



Touch Direct Frequency Entry





- **Touch or Mouse on VFO-A or VFO-B digits to bring up Direct Frequency Entry menu (example touch 1-4-3-2-0-KZ or 14.320 mHz = 14.320Mhz 20M)**
  Note: there is a separate direct Freq entry for each VFO when 2 Receivers are active

- **RIT Step** Receiver Incremental Tuning

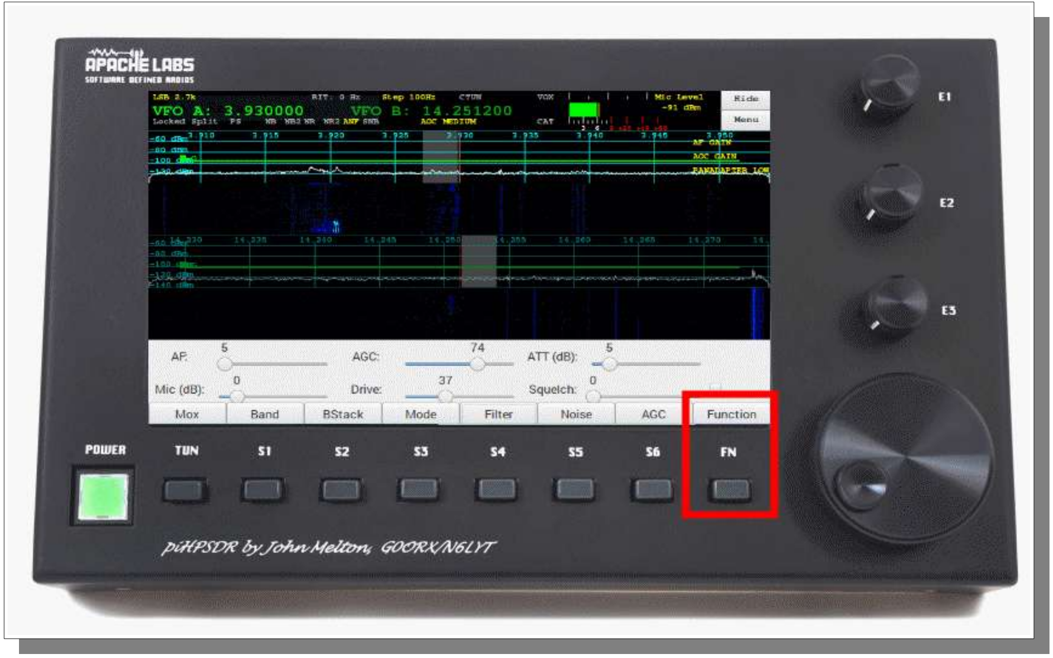
- **VFO Step** Frequency change per increment of VFO Encoder

Toolbar – Function button

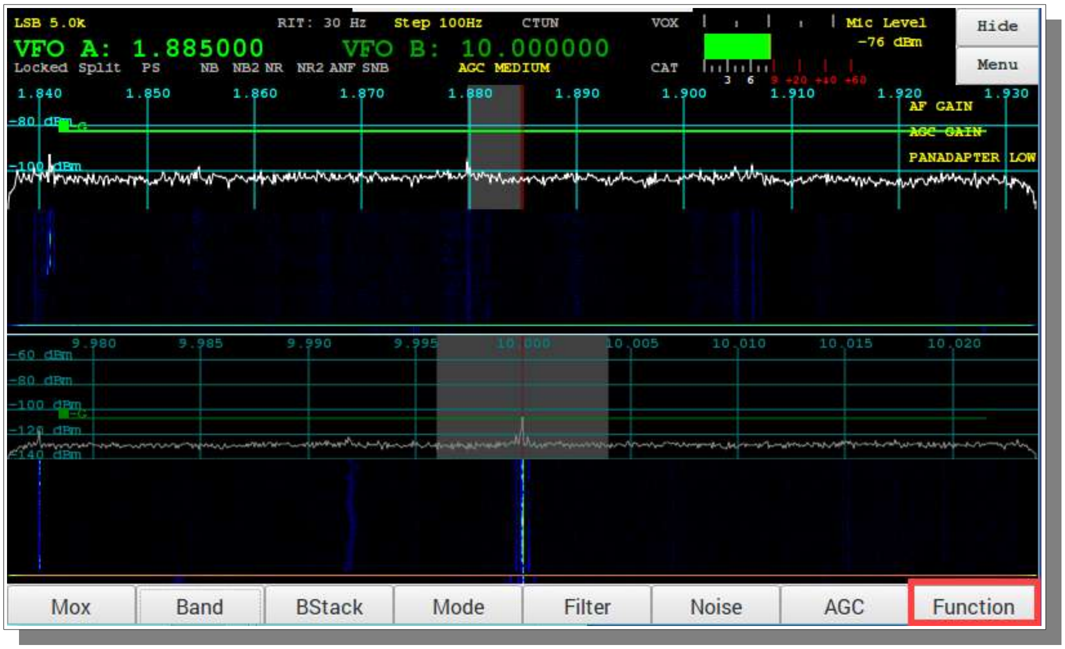
There are eight (8) switches on the pihpsdr Controller.

The functions are arranged in four groups.

Switches S1 through S6 change function as you push the FN key



Physical Function **FN** button
switches Tune/MOX and S1 – S6 functions
as you push the FN button.

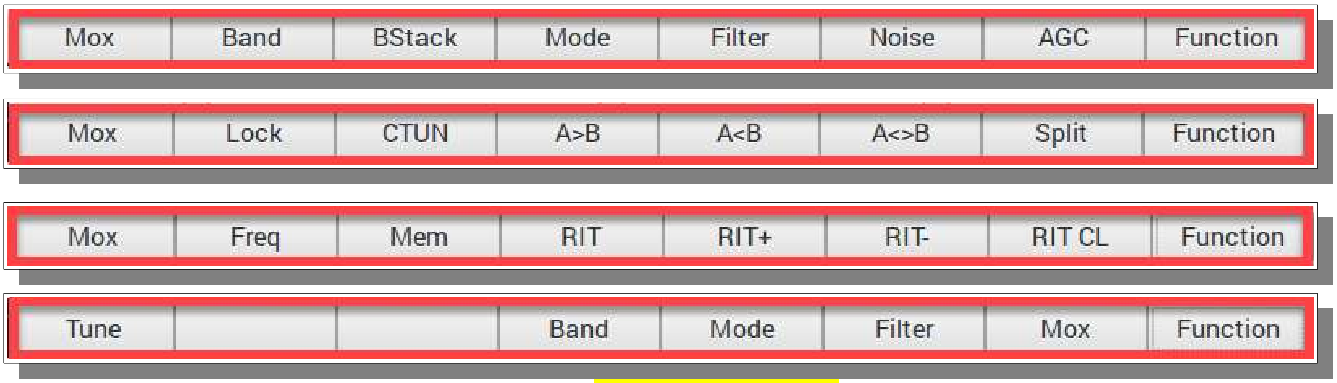


On-Screen **Function**
switches Tune/MOX and S1 – S6 functions
as you touch the Function button.

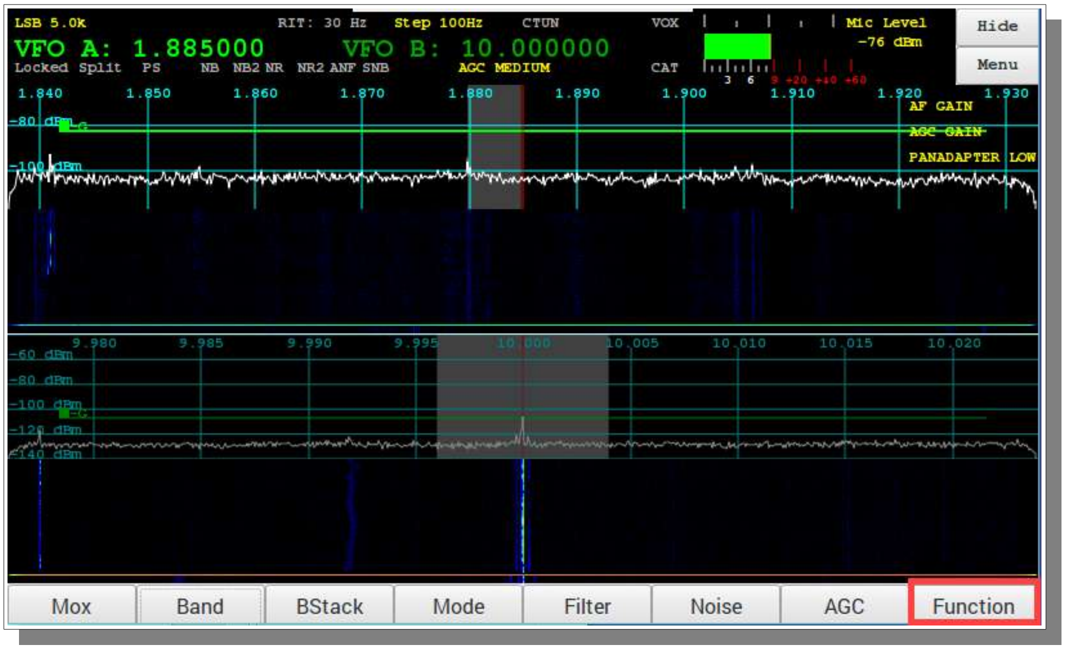
Toolbar – Function menu selections

The function touch screen button or physical FN button enable selection of the 4 function menu groups.

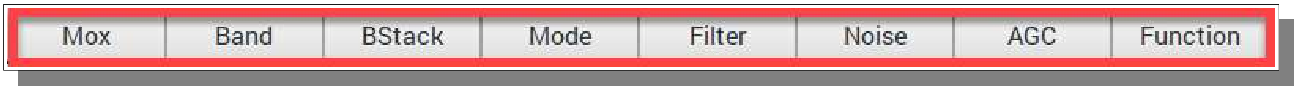
Four Function menus – select with Function button or front panel FN switch



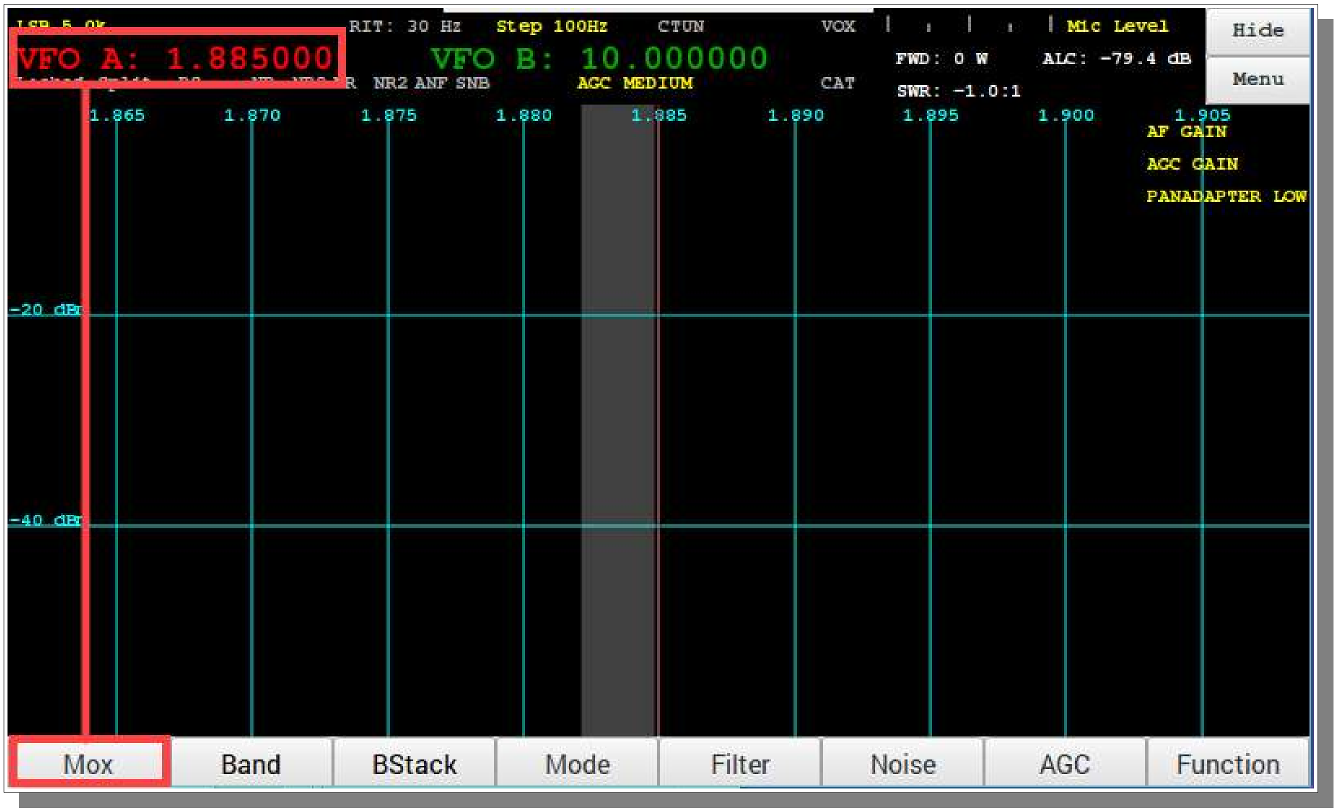
4 Function Groups



Function Group 1

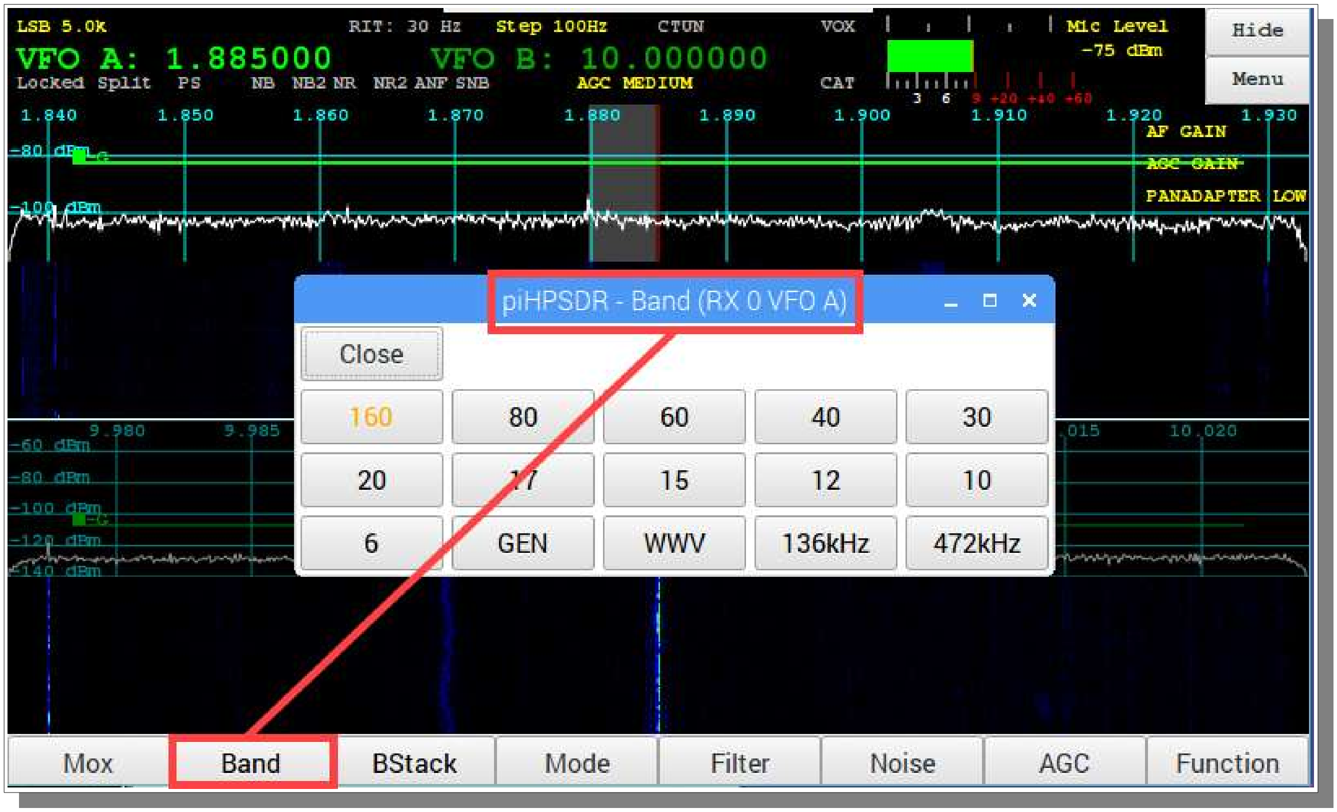


Toolbar → MOX



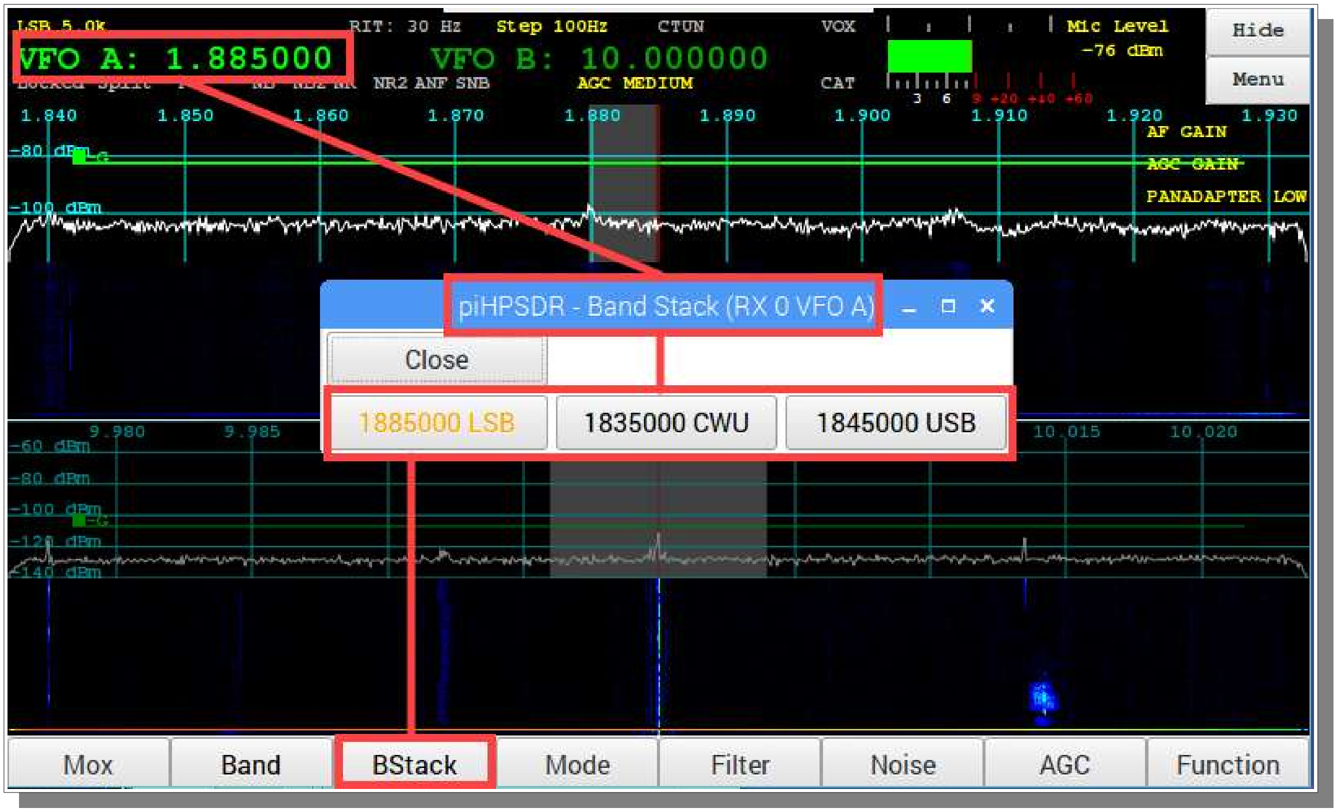
The MOX touch screen button is like Push-to-Talk and switches from Receive to Transmit as indicated by the RED VFO display.

Toolbar → BAND VFO-A

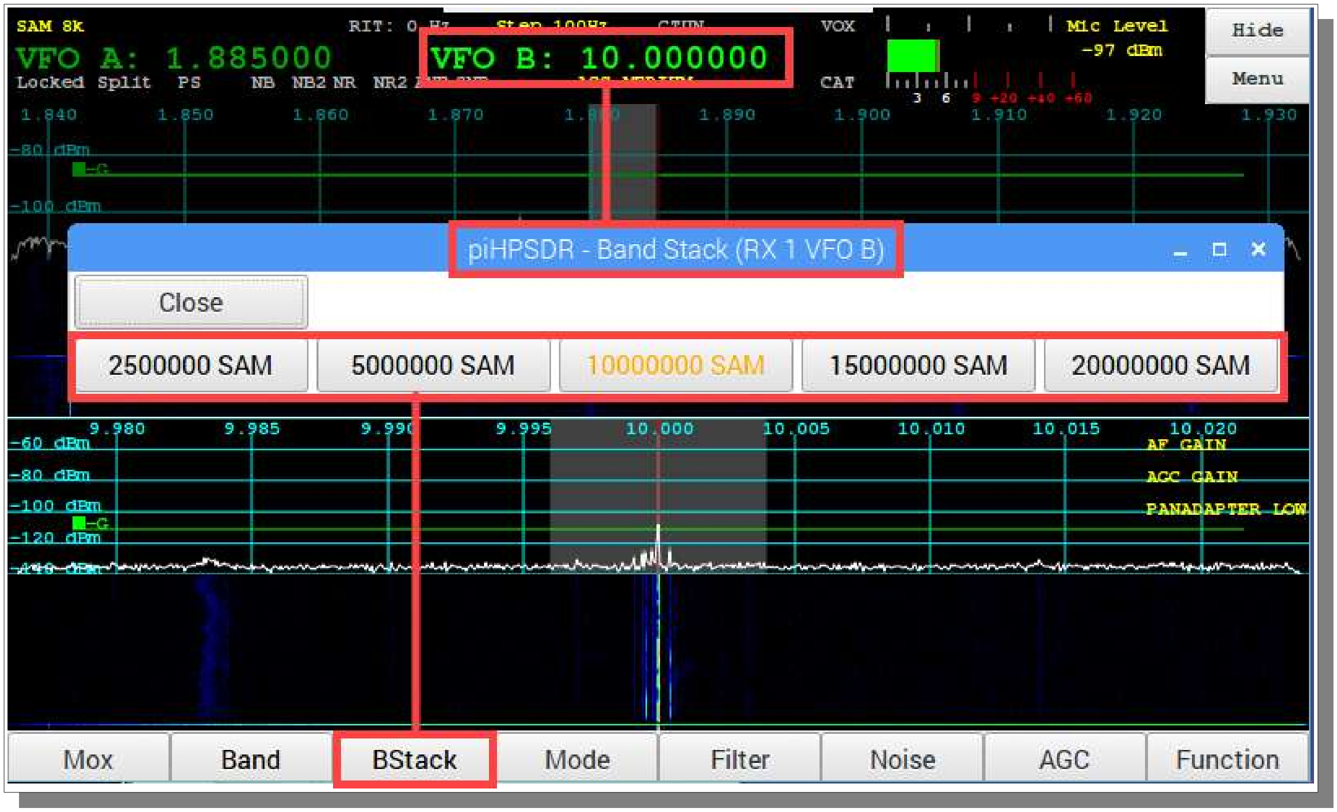


Tapping the Band button pops up the Band Selection Menu. Tap on a band to change to that band. If the band is the same as the current band, it will step to the next band stack entry.

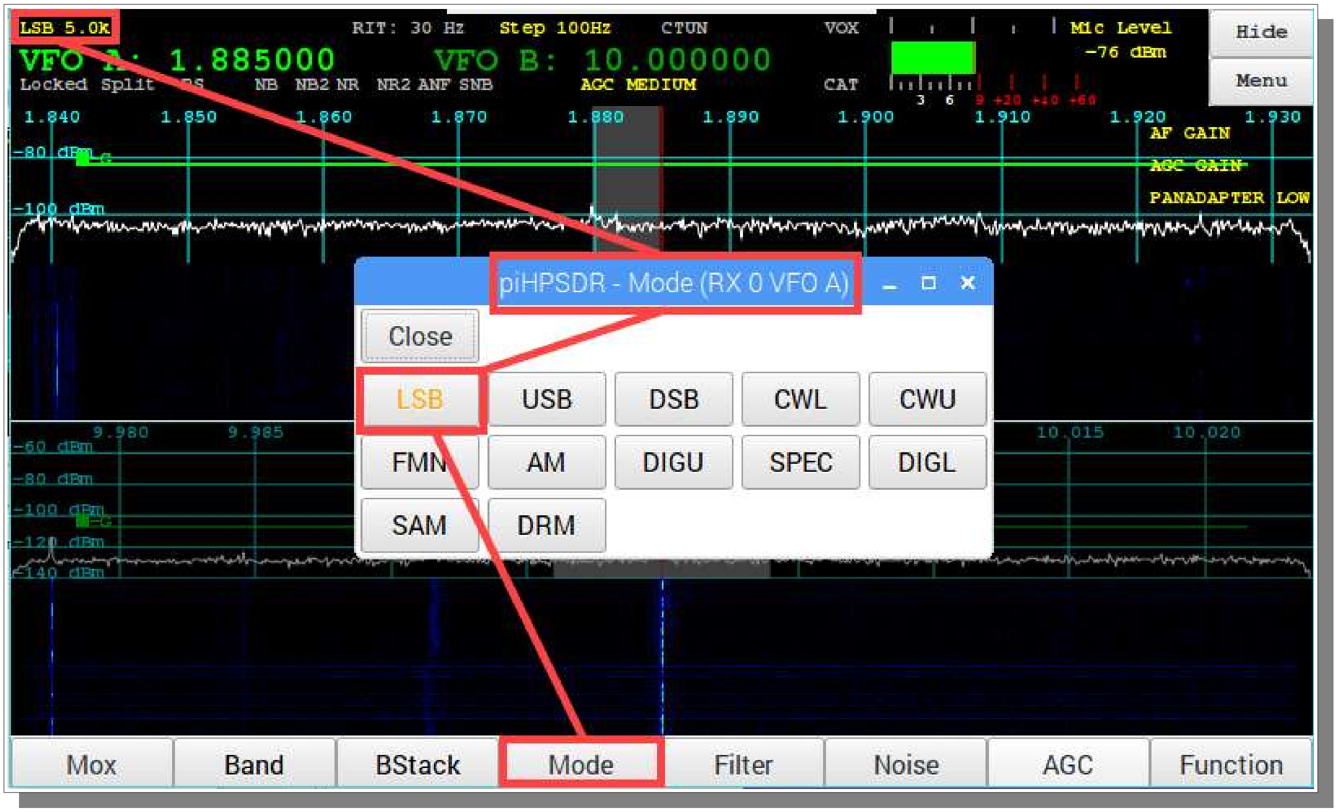
Toolbar → Band Stack for VFO-A selected



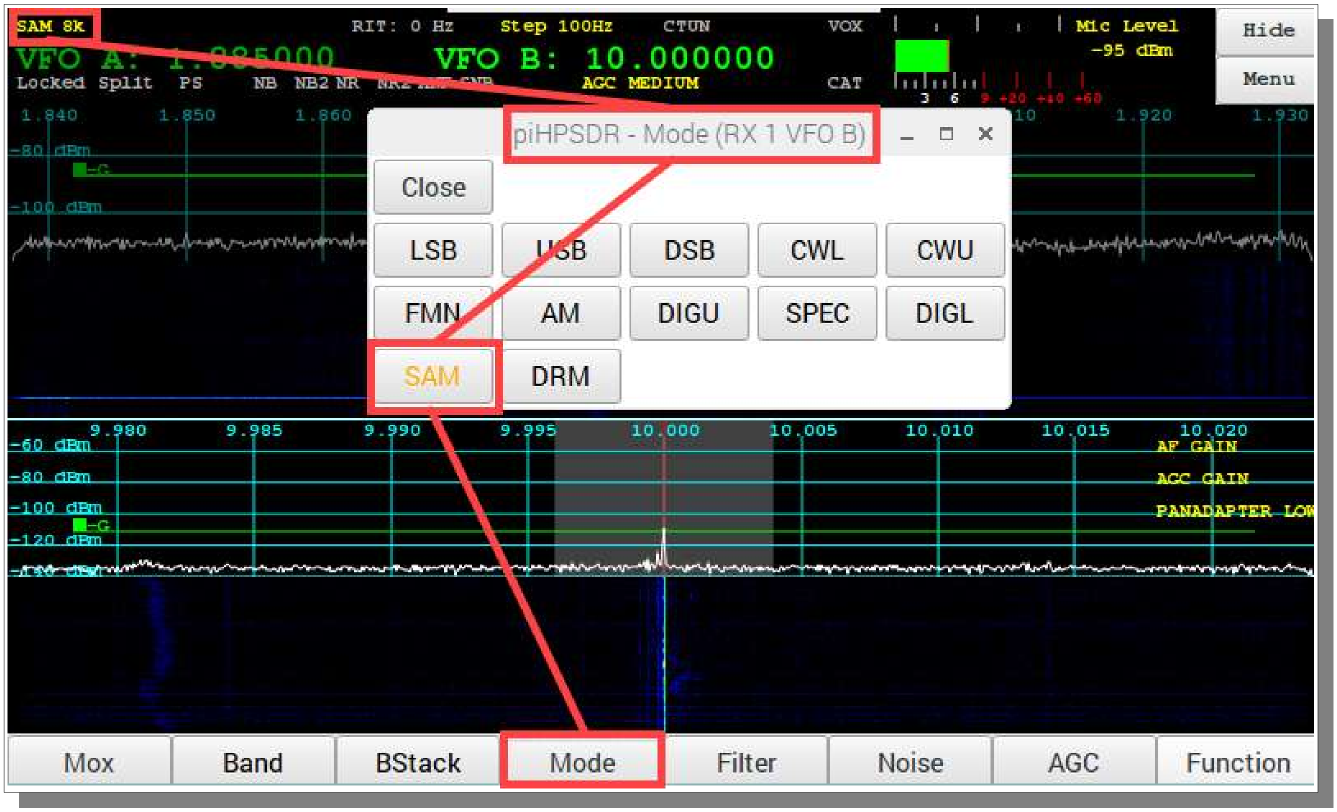
Toolbar → Band Stack for VFO-B selected



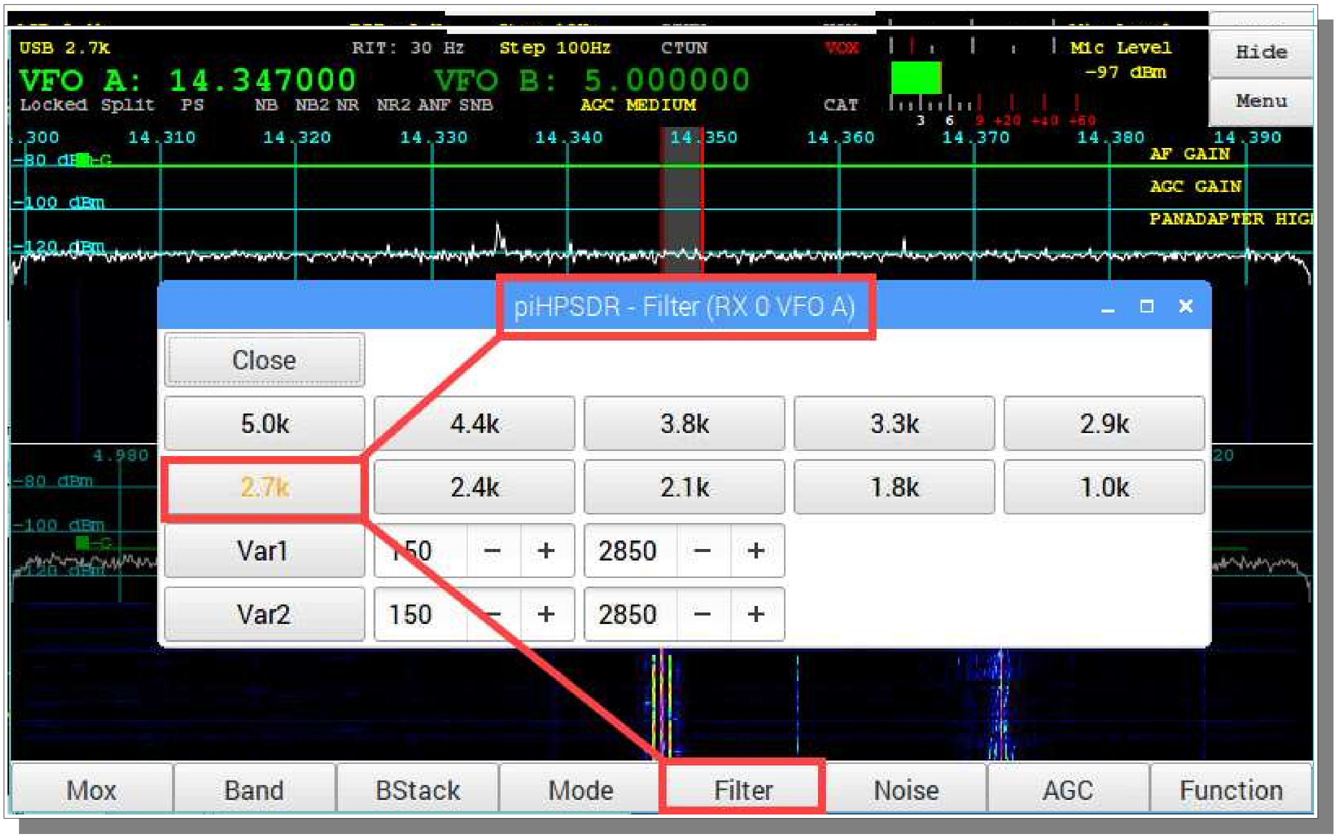
Toolbar → Mode VFO-A



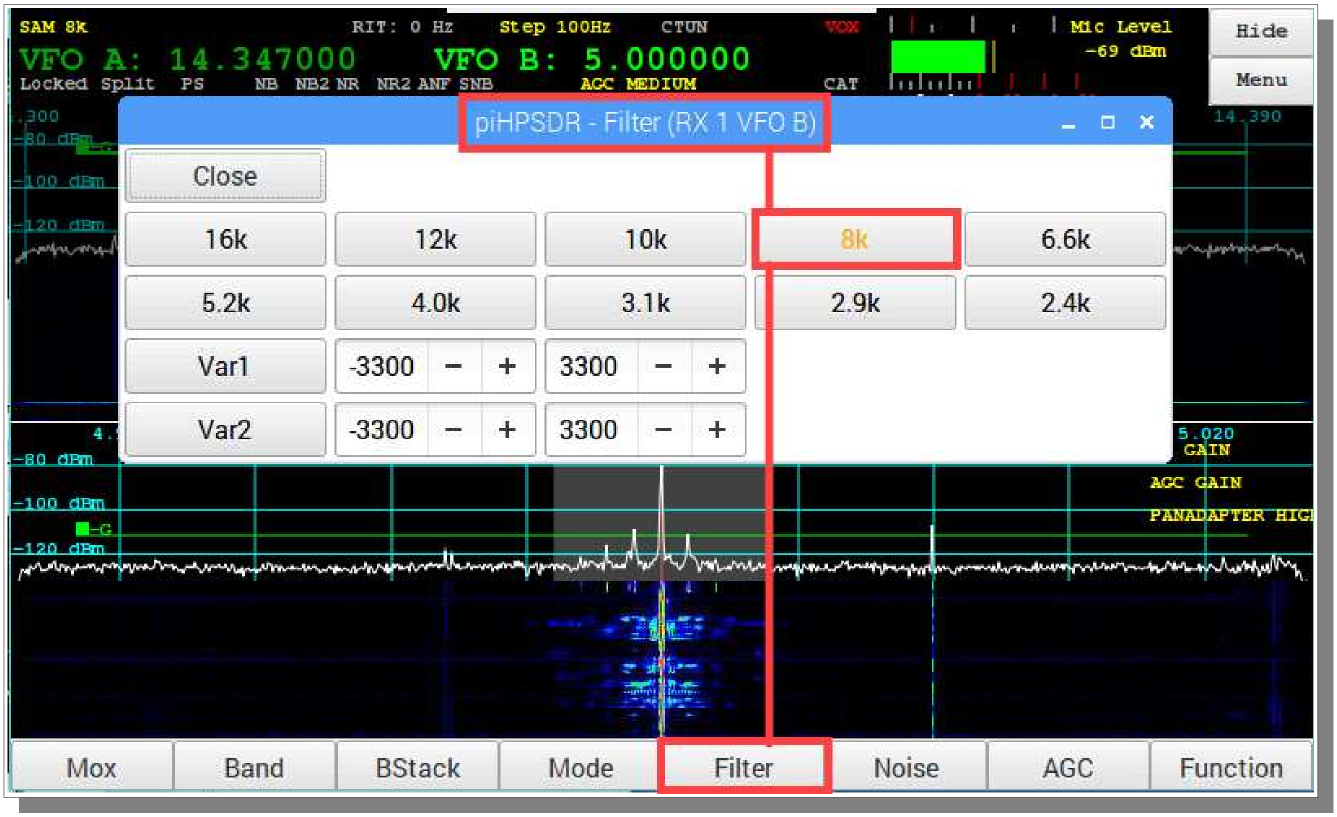
Toolbar → Mode VFO-B



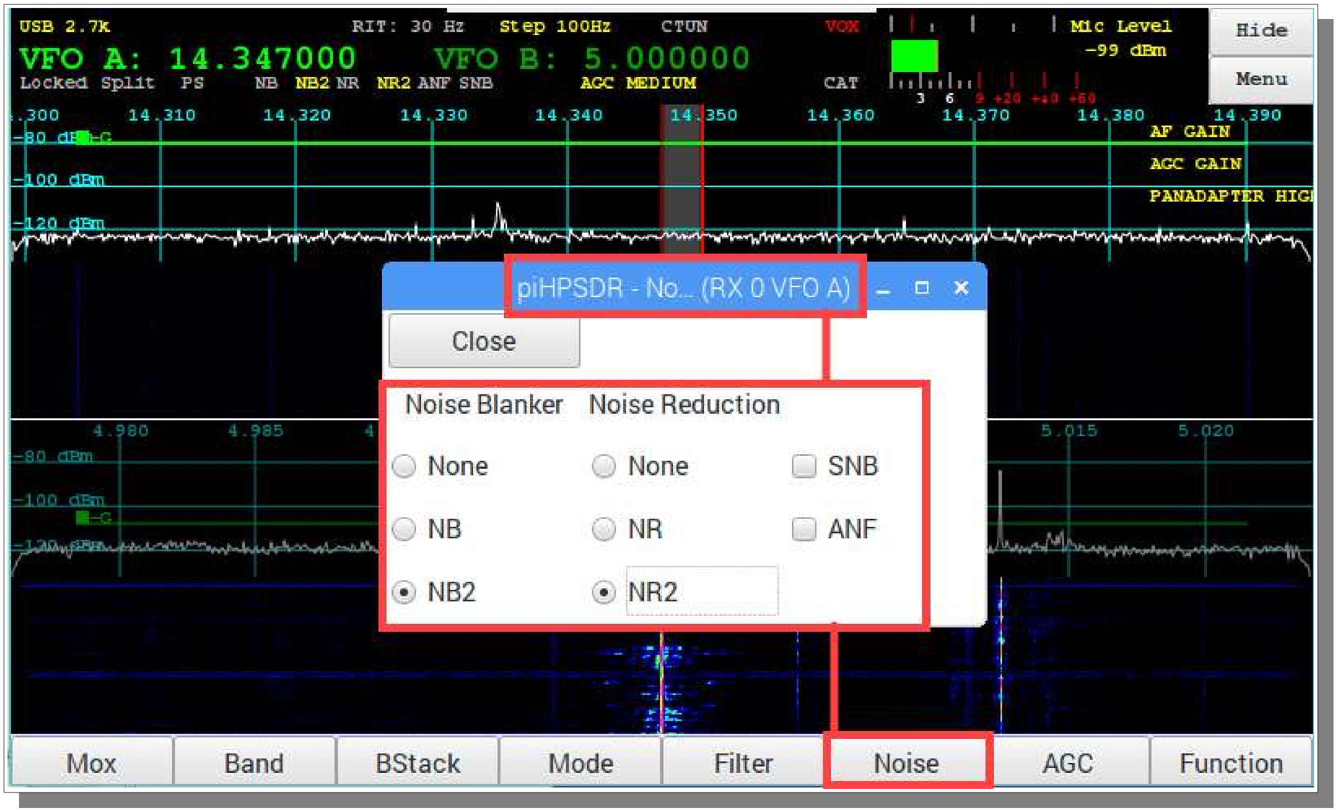
Toolbar → Filter VFO-A



Toolbar → Filter VFO-B



Toolbar → NOISE VFO-A

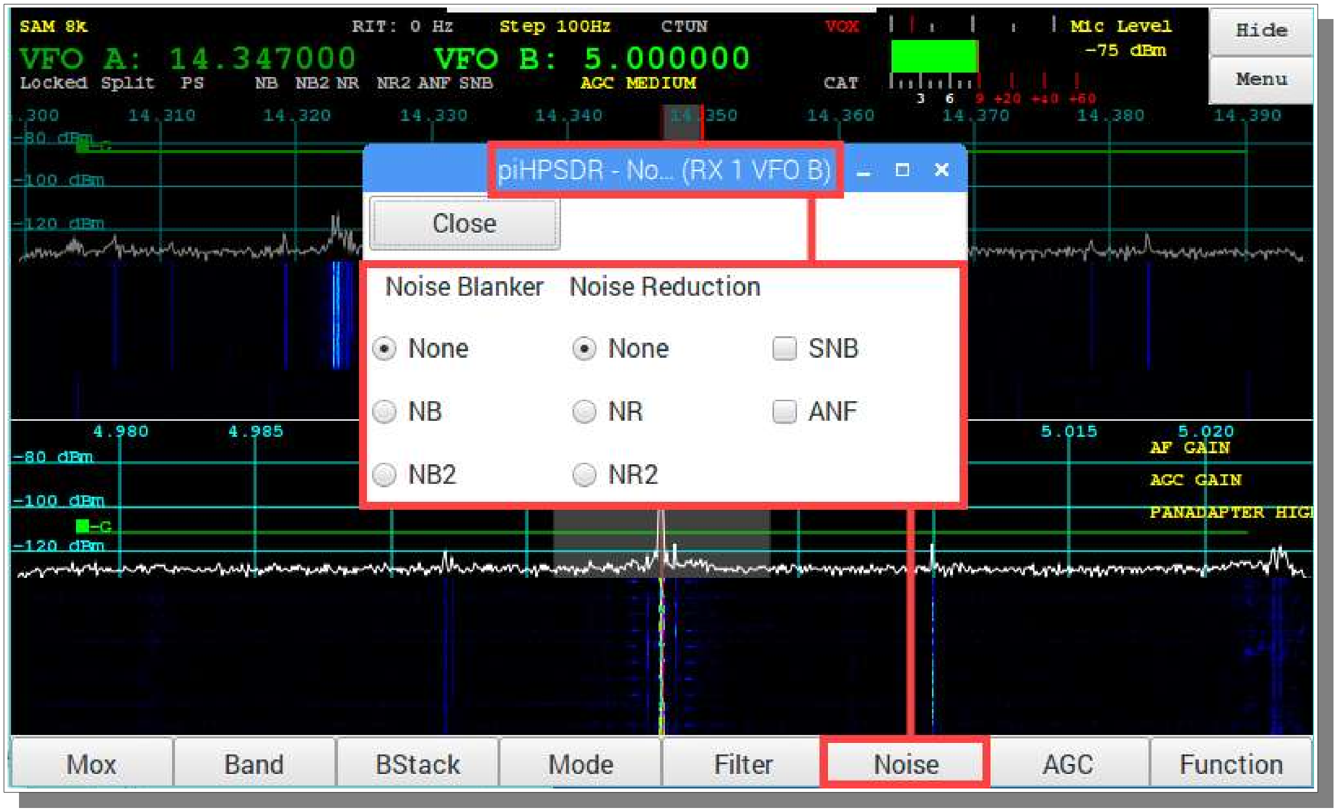


piHPSDR - No... (RX 0 VFO A)

Close

Noise Blanker	Noise Reduction	
○ None	○ None	☐ SNB
○ NB	○ NR	☐ ANF
● NB2	● NR2	

Toolbar → NOISE VFO-B

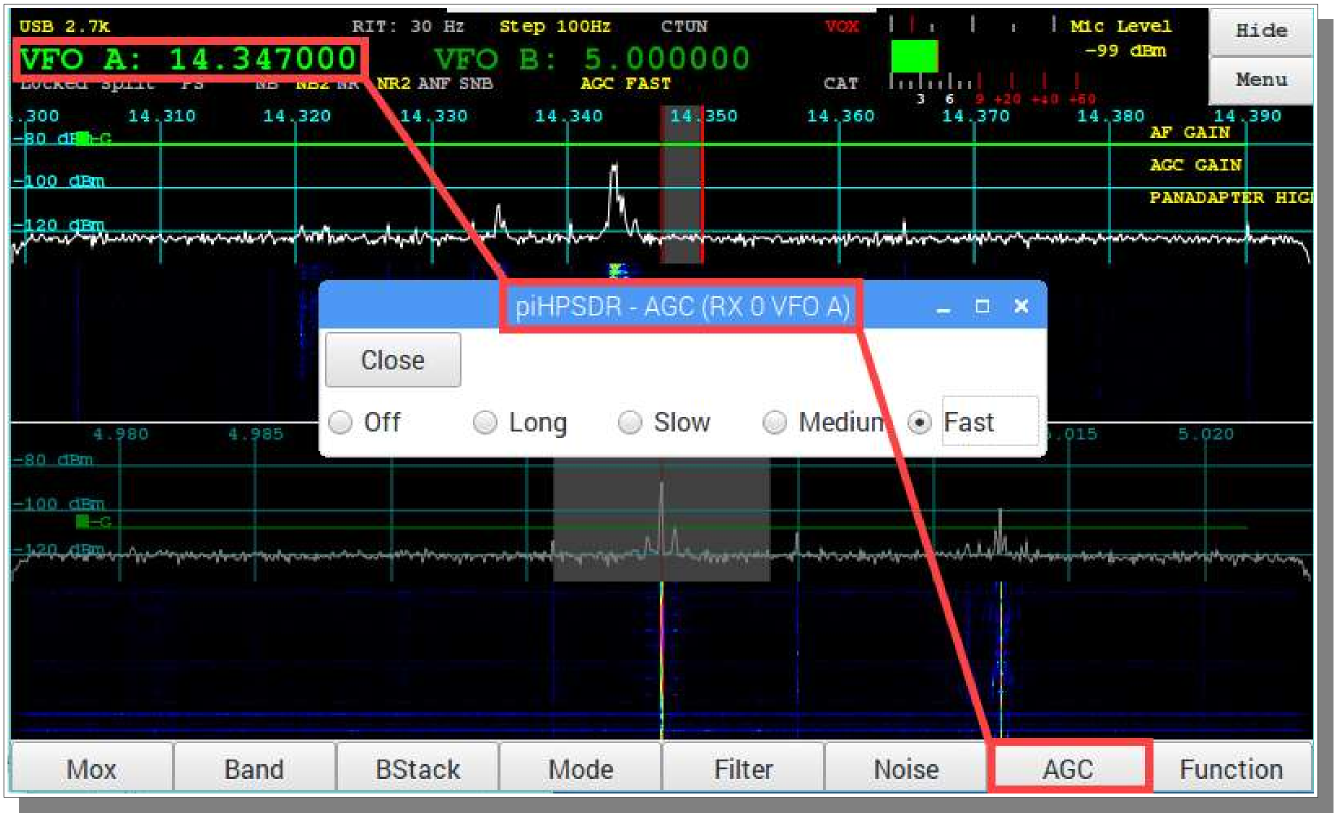


piHPSDR - No... (RX 1 VFO B)

Close

Noise Blanker	Noise Reduction	
● None	● None	☐ SNB
○ NB	○ NR	☐ ANF
○ NB2	○ NR2	

Toolbar → AGC VFO-A

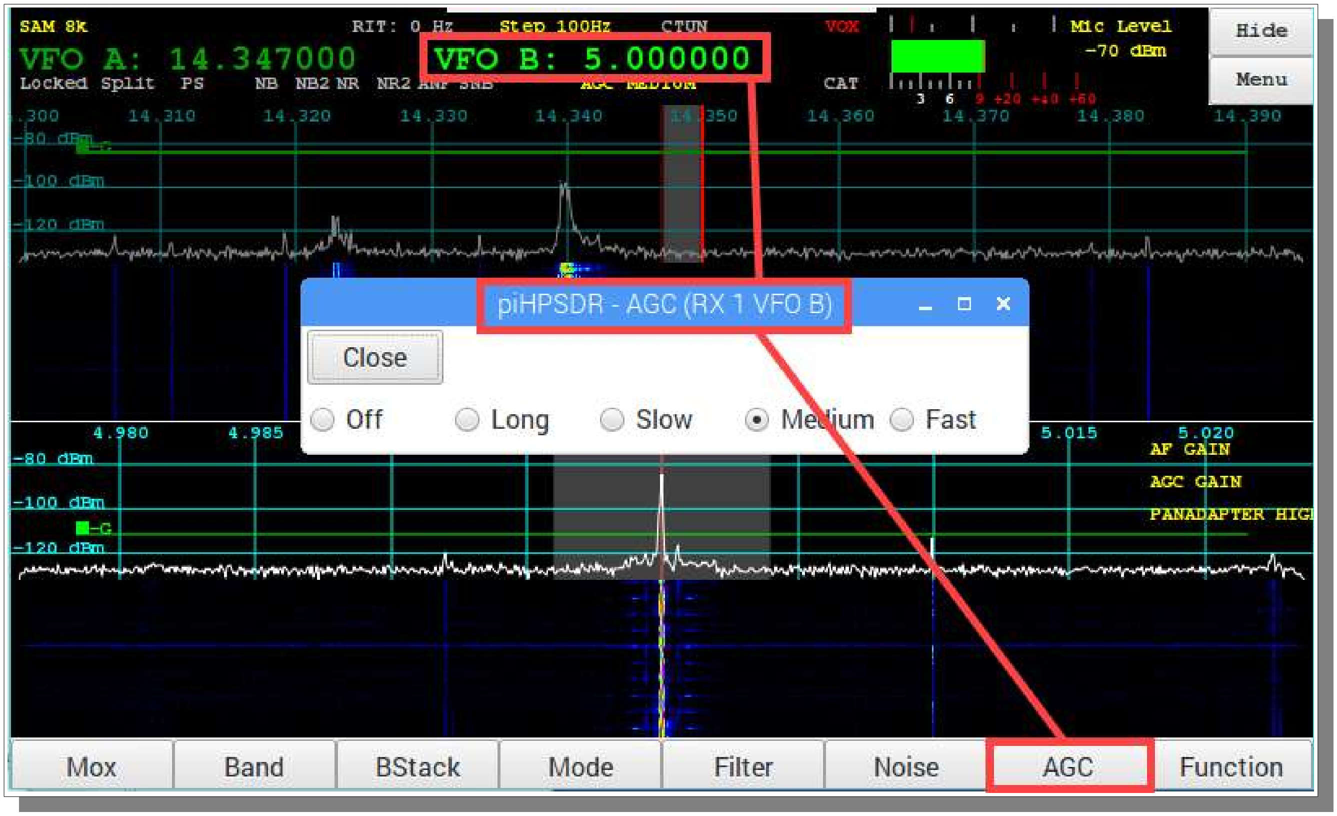


piHPSDR - AGC (RX 0 VFO A)

Close

○ Off ○ Long ○ Slow ○ Medium ● Fast

Toolbar → AGC VFO-B

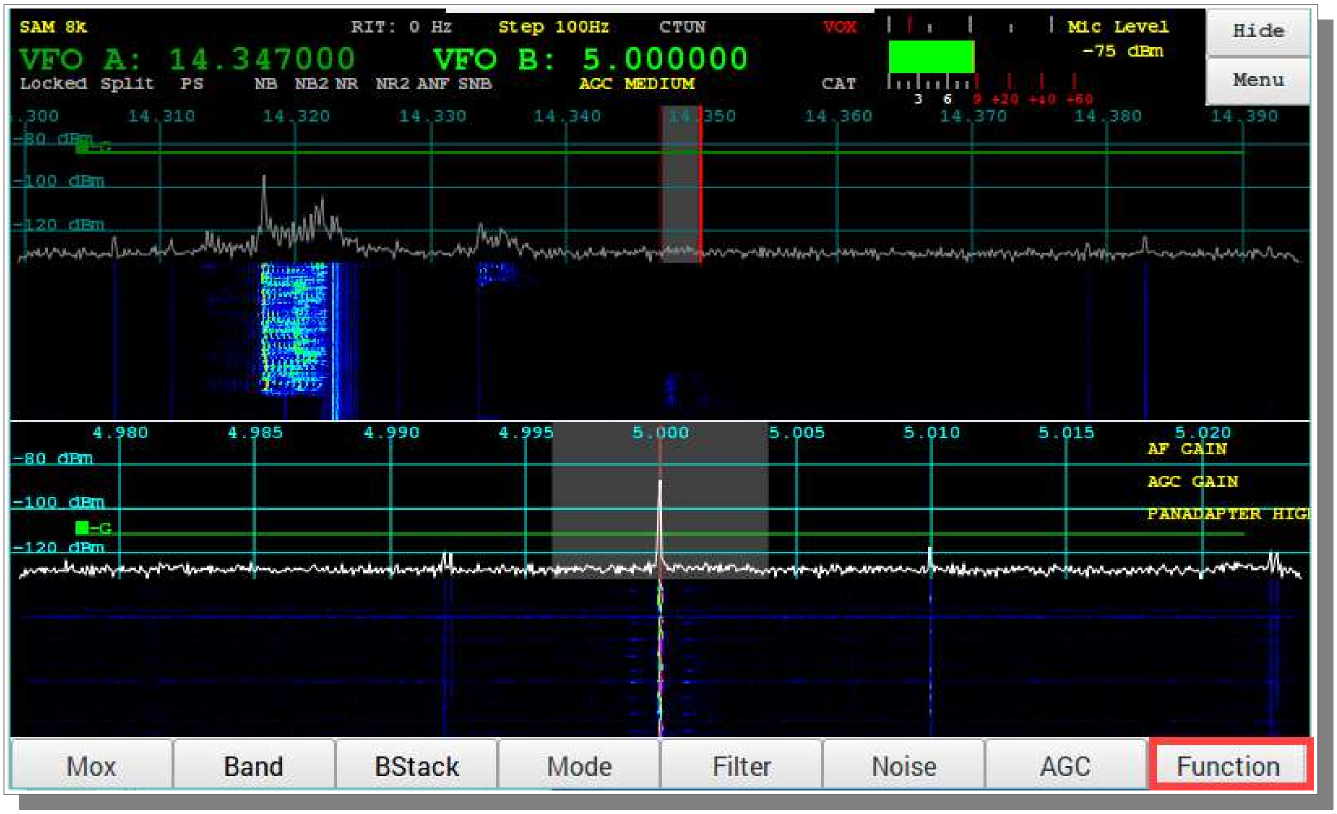


piHPSDR - AGC (RX 1 VFO B)

Close

○ Off ○ Long ○ Slow ● Medium ○ Fast

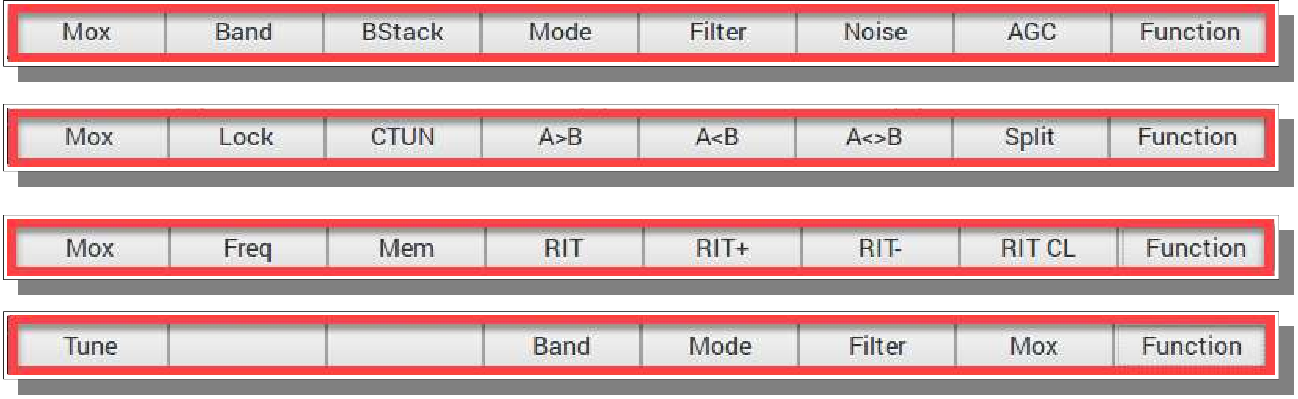
Toolbar → Function button



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

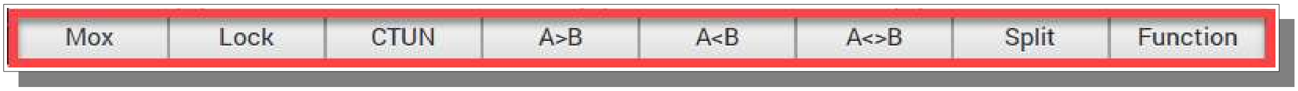
Four Function Toolbar Groups → Menu

select with FN button or front panel touch screen Function

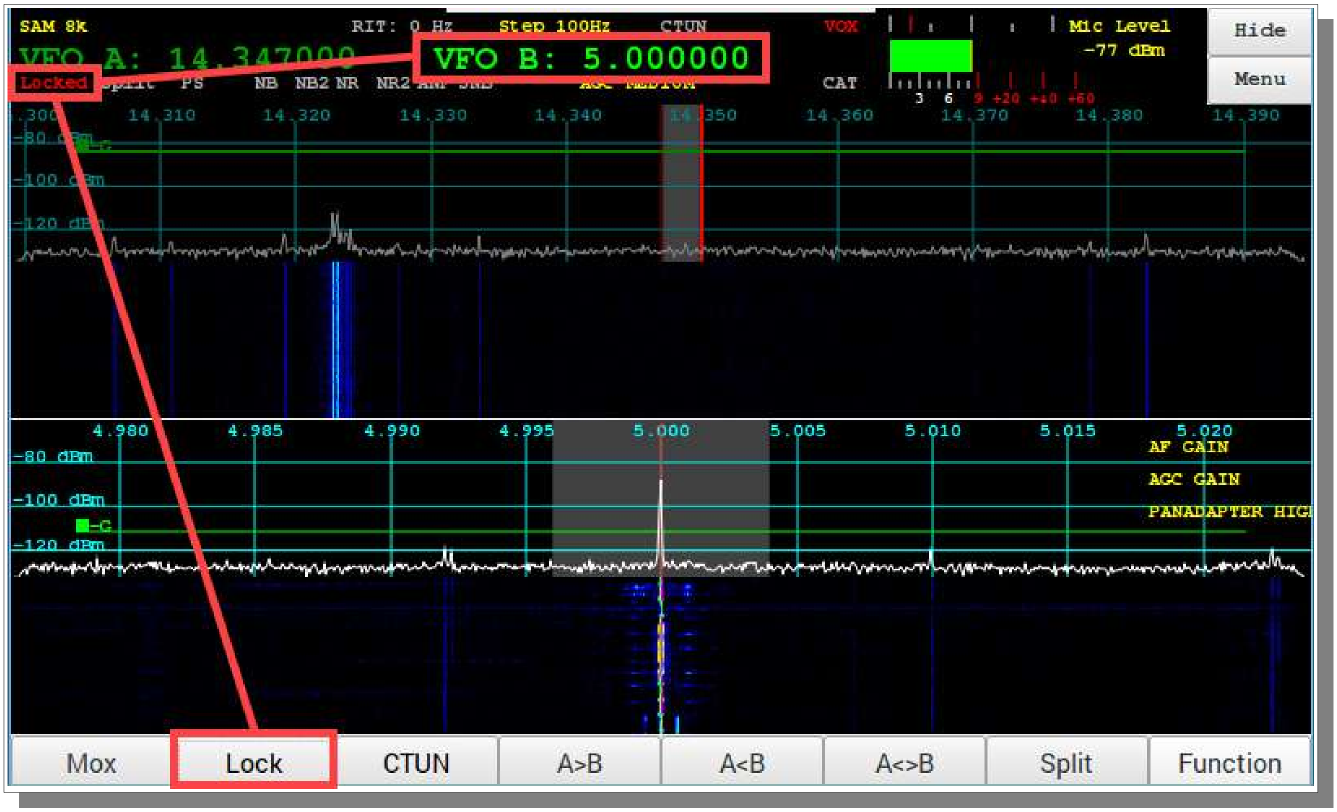


4 Function Groups

Toolbar → VFO Lock



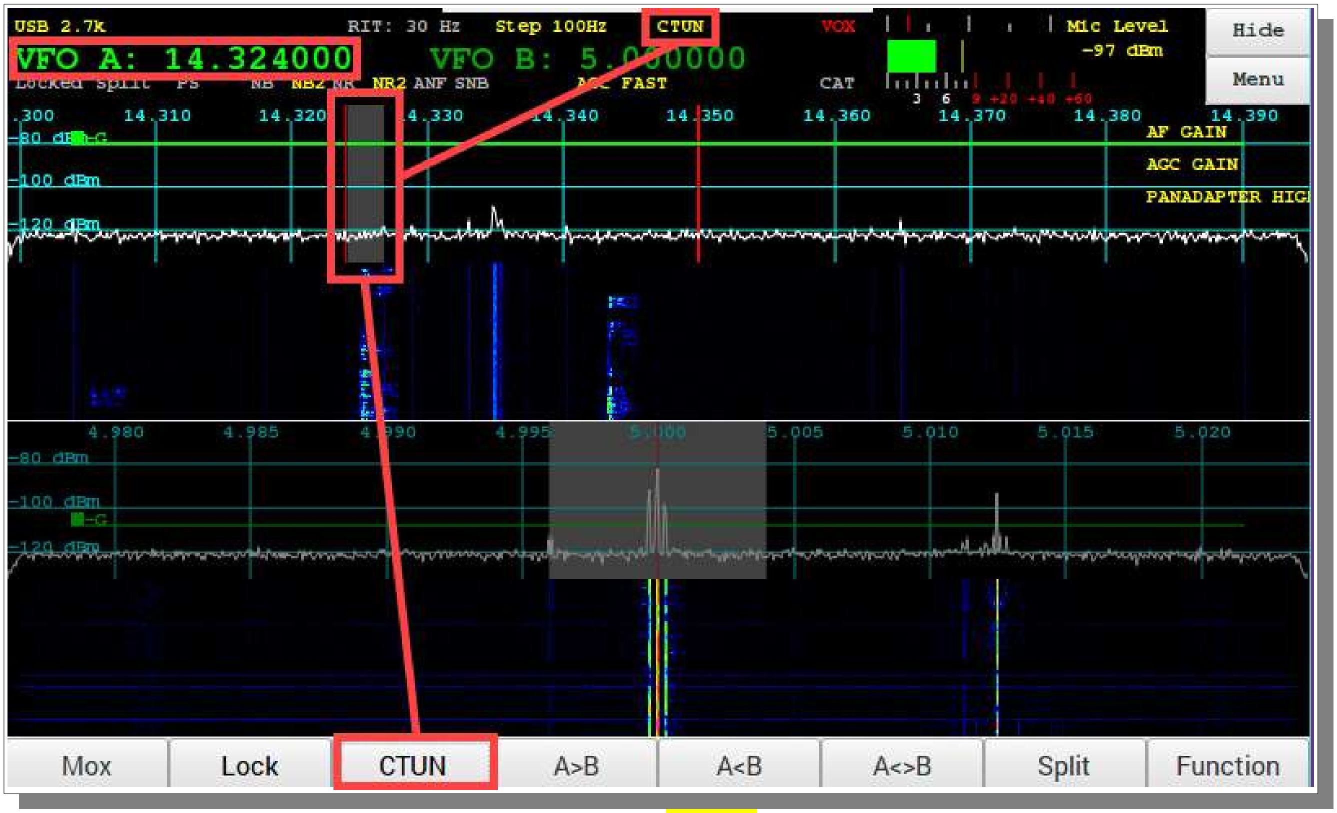
Function Group 2 Menu items



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 the VFO is locked, the red Locked text will be displayed near the VFO.

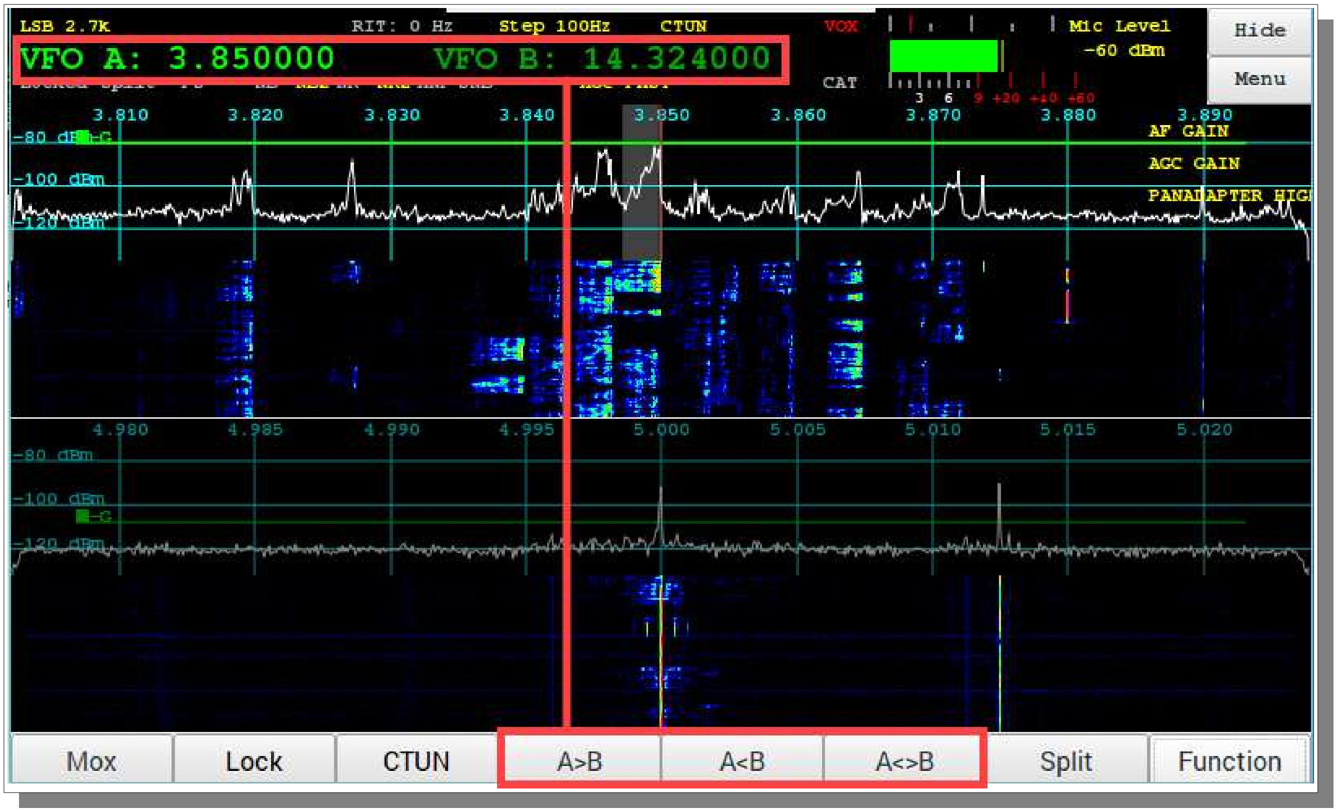
Toolbar → CTUN – Click Tuning



Note: how filter is Tuned to a new frequency and CTUNE is announced in the status bar in Yellow Tapping on the CTUN button will enable or disable the click tuning function. When the function is enabled the CTUN button text will be shown in yellow.

When CTUN is enabled, tuning is restricted to the passband currently displayed. The tuned frequency and filter moves withing the current passband display without moving the panadapter or waterfall left or right.

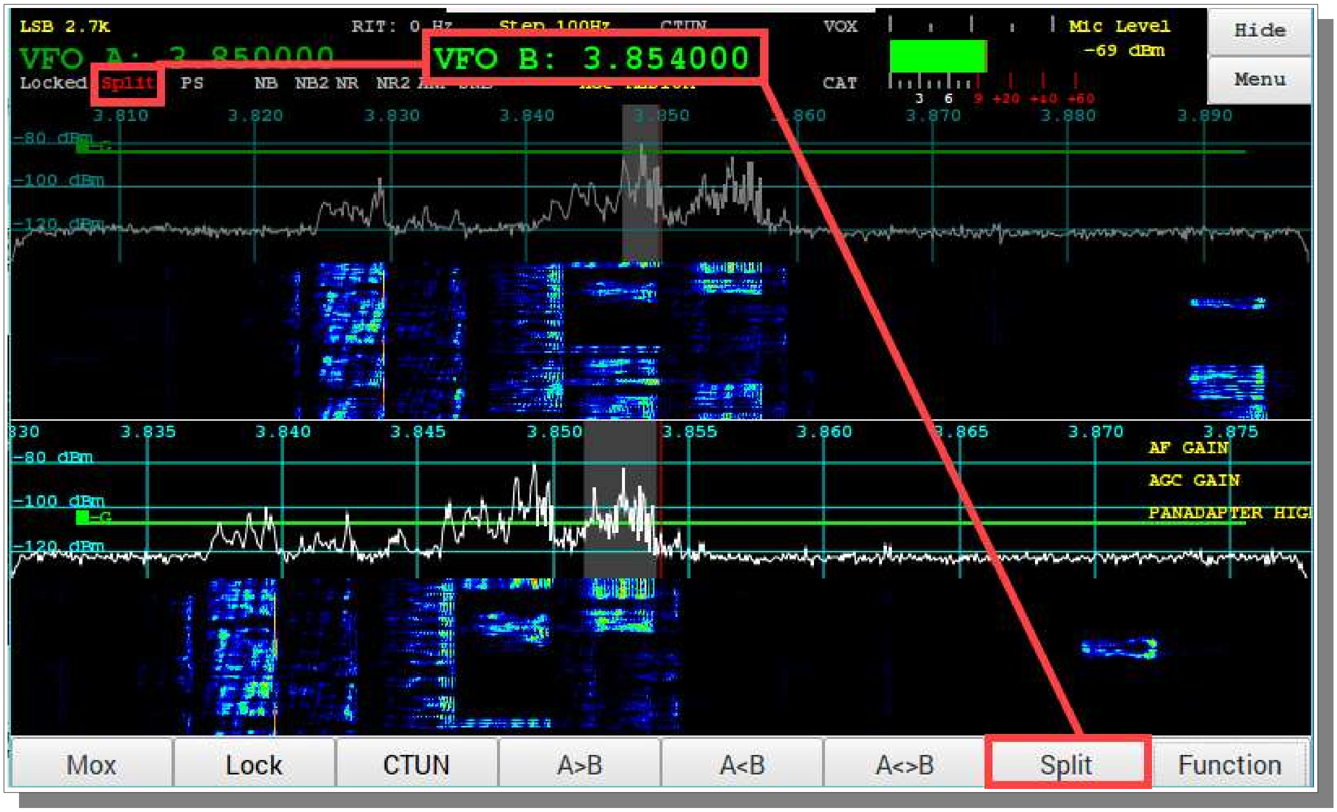
Toolbar → A>B, A<B, A<>B



The Function menu Group 2 - allows you to copy the VFO frequency:

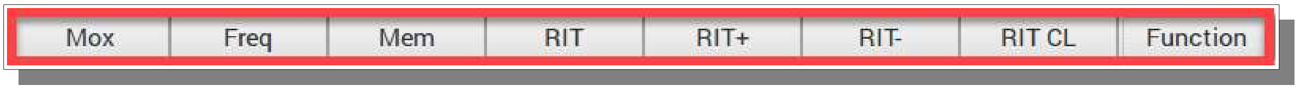
- **VFO-A into VFO-B (A>B)**
- **VFO-B into VFO-A (A<B)**
- **VFO-A swap with VFO-B (A<>B)**

Toolbar → SPLIT

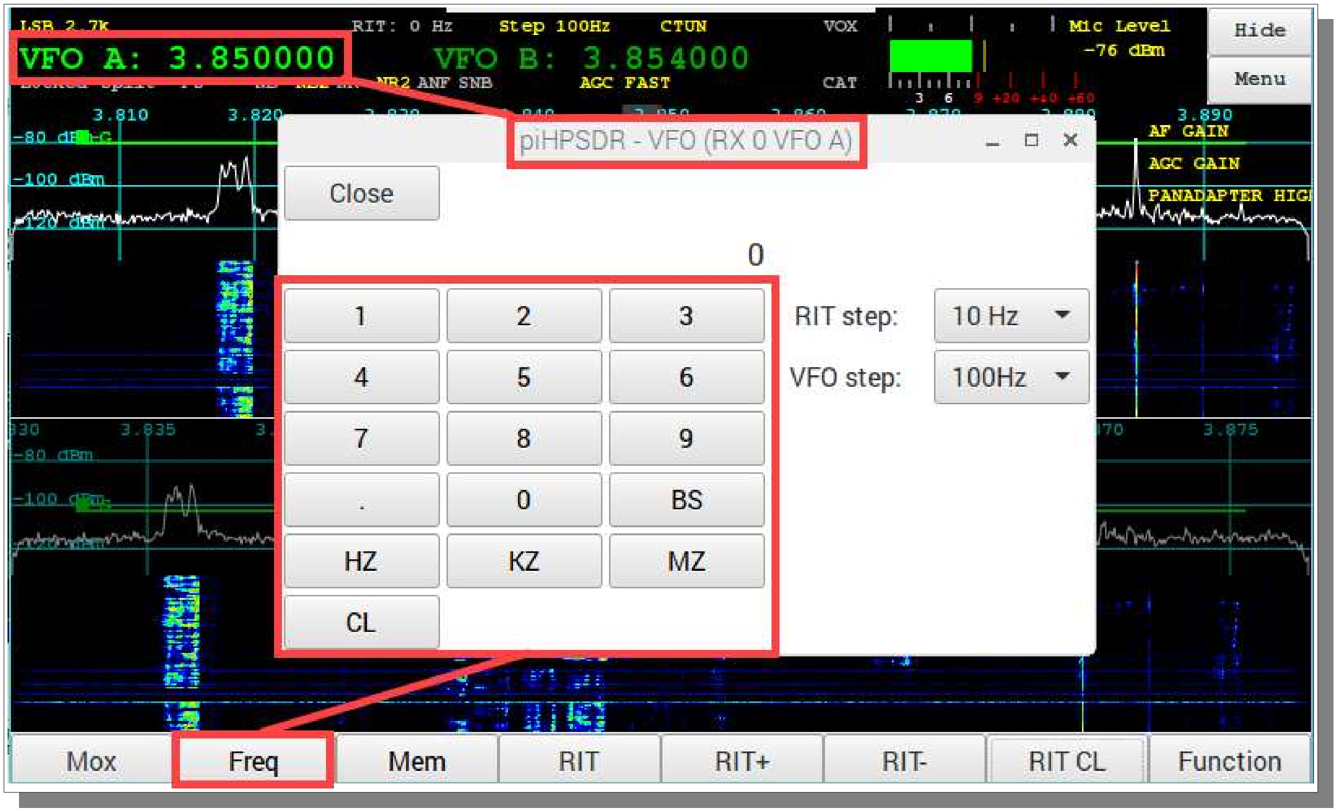


The Split function allows you to select VFO-A for Receive and VFO-B for Transmit. The illustration shows transmission on VFO-B. This is a common practice when working Contests, DX, or crossband.

Toolbar → FREQ



Function menu group 3



VFO A: 3.850000

VFO B: 3.854000

piHPSDR - VFO (RX 0 VFO A)

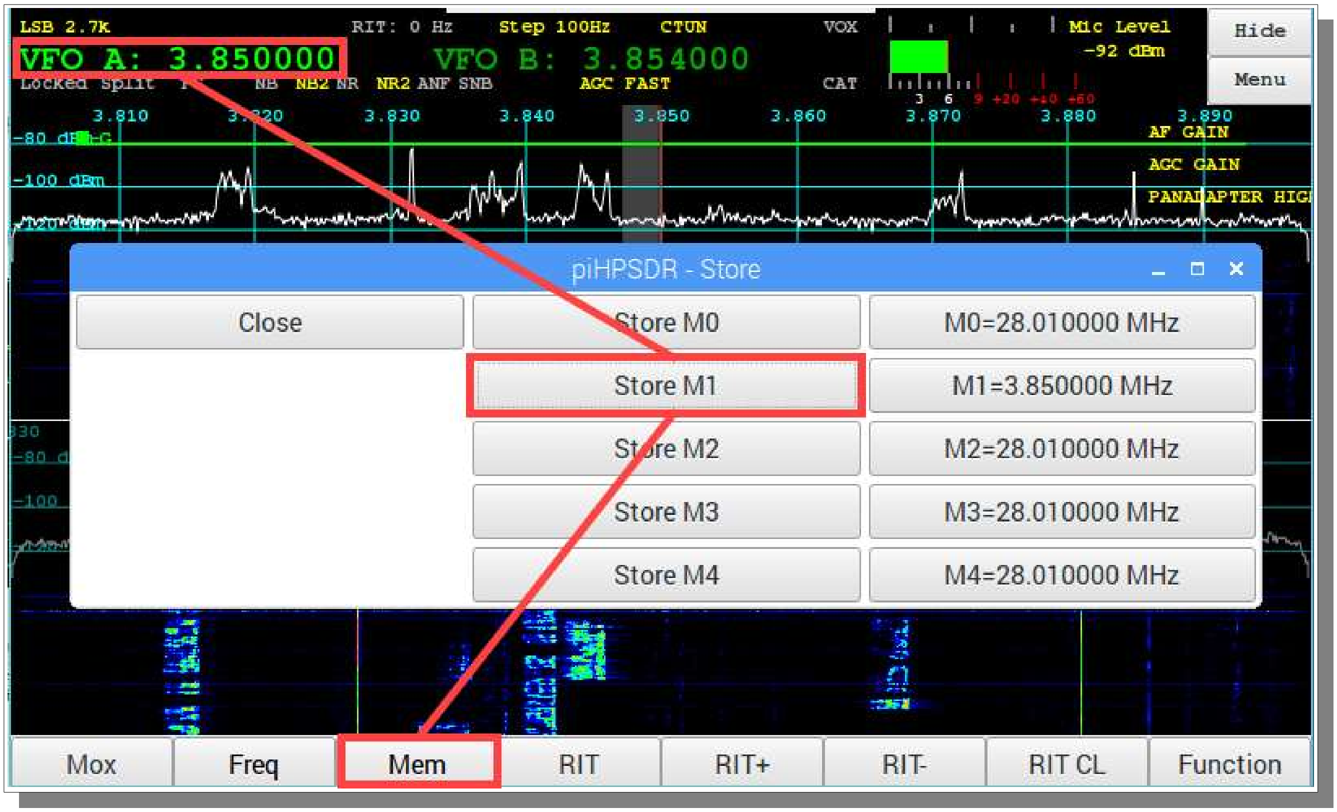
0

RIT step: 10 Hz

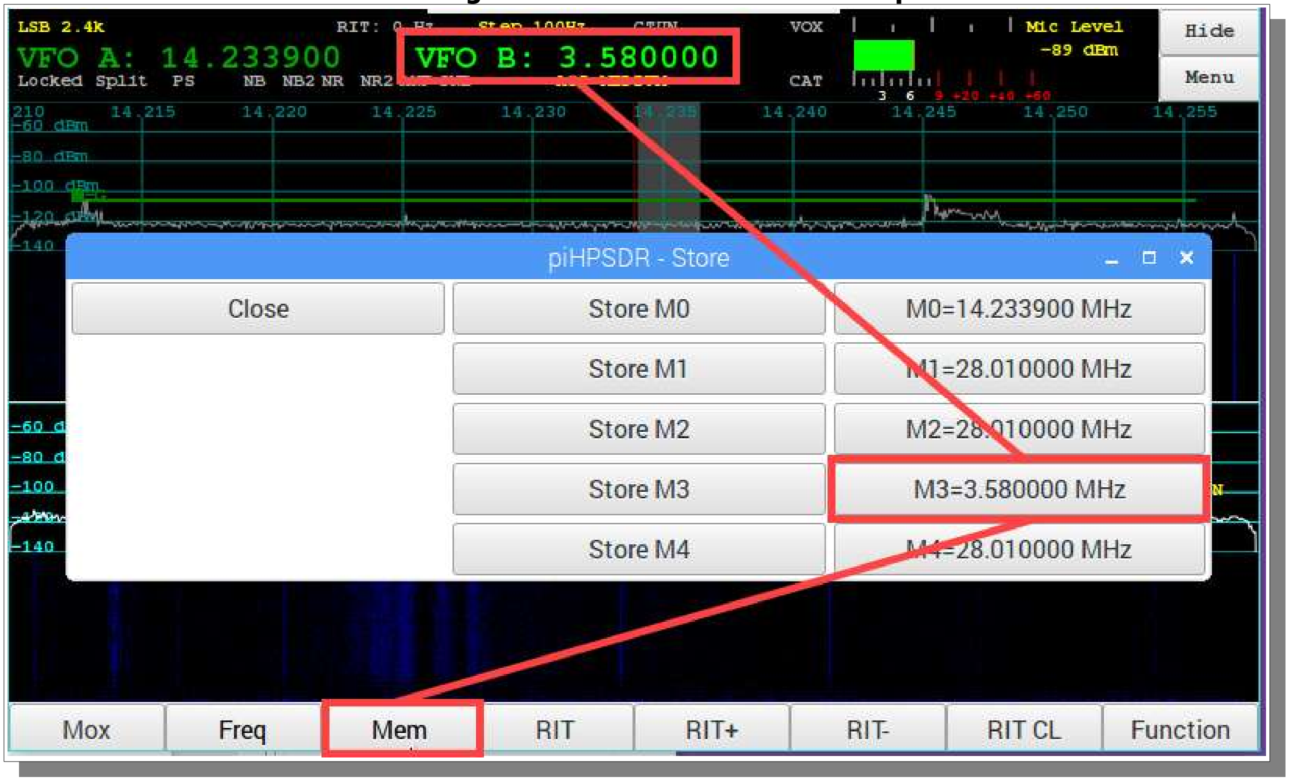
VFO step: 100Hz

- Direct frequency entry touching the screen display is easily done by first selecting the VFO you wish to change, and then entering in the frequency in kHz or mHz. This illustration shows selection of 7.228 LSB on 40M..
- RIT step the increment in Hz for the RIT+ and RIT- toolbar entries (shown on page 48).
- VFO step the increment in Hz for the VFO Knob (E4) or Mouse Wheel.

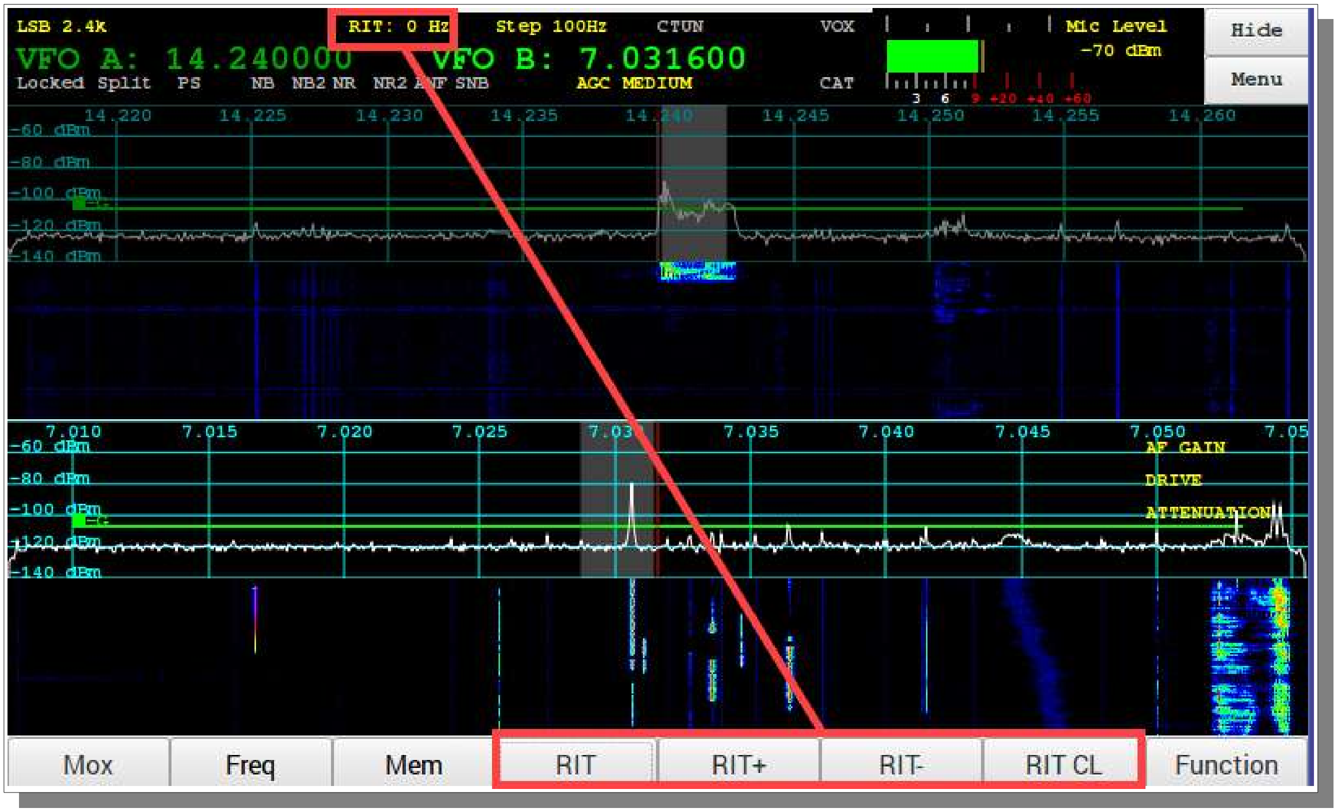
Toolbar → MEM



Convenient storage and recall of five favorite frequencies



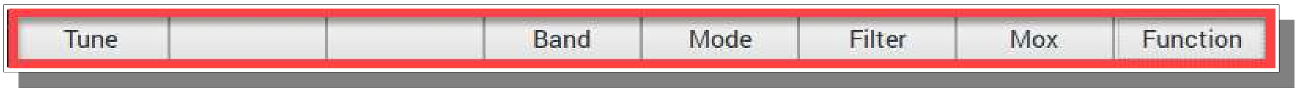
Toolbar → RIT – Receiver Incremental Tuning



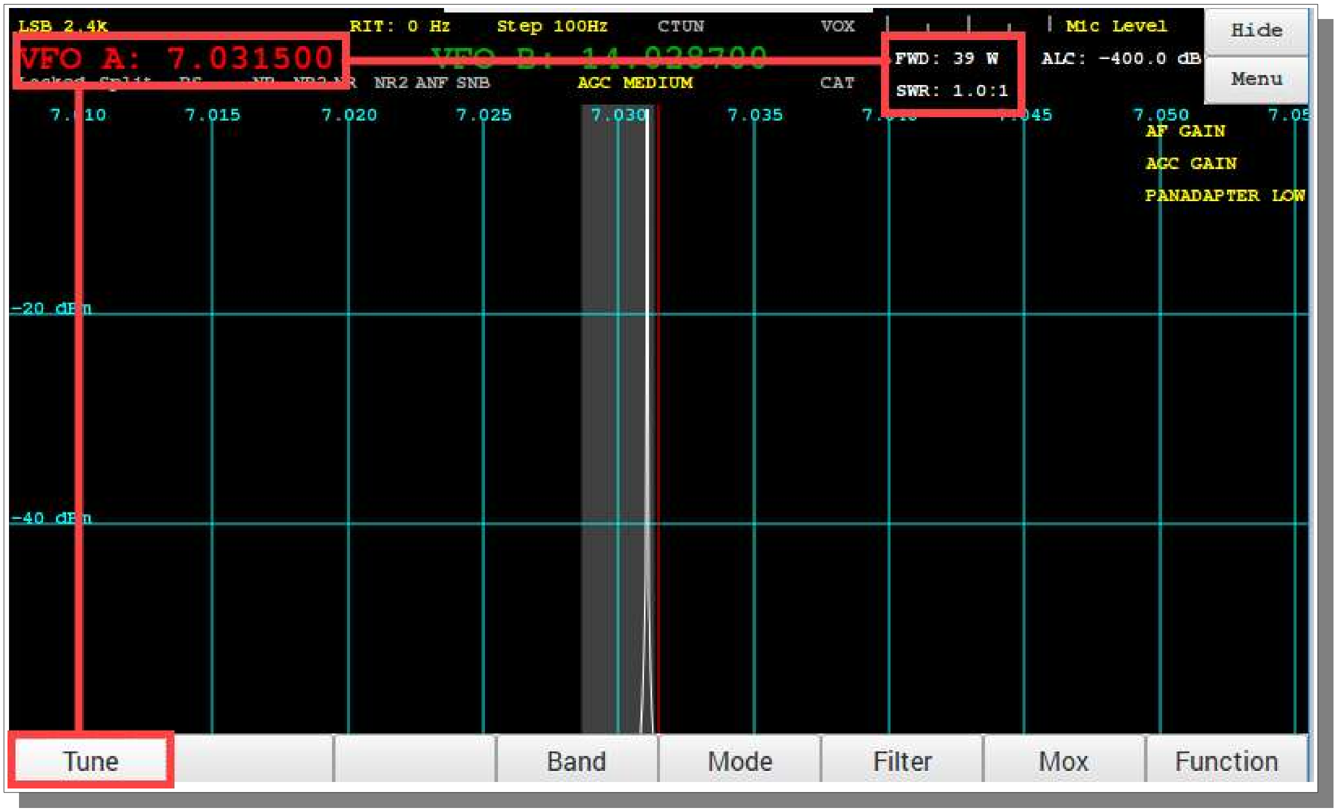
- **Tapping the RIT OFF/ON enables or disables Receiver Incremental Tuning**
- **Tapping on the RIT+ (plus) or RIT- (minus) button will**
- **Tapping RIT CL = RIT CLEAR**

Note: RIT offset shown in YELLOW TEXT above VFO-A

Toolbar → TUNE



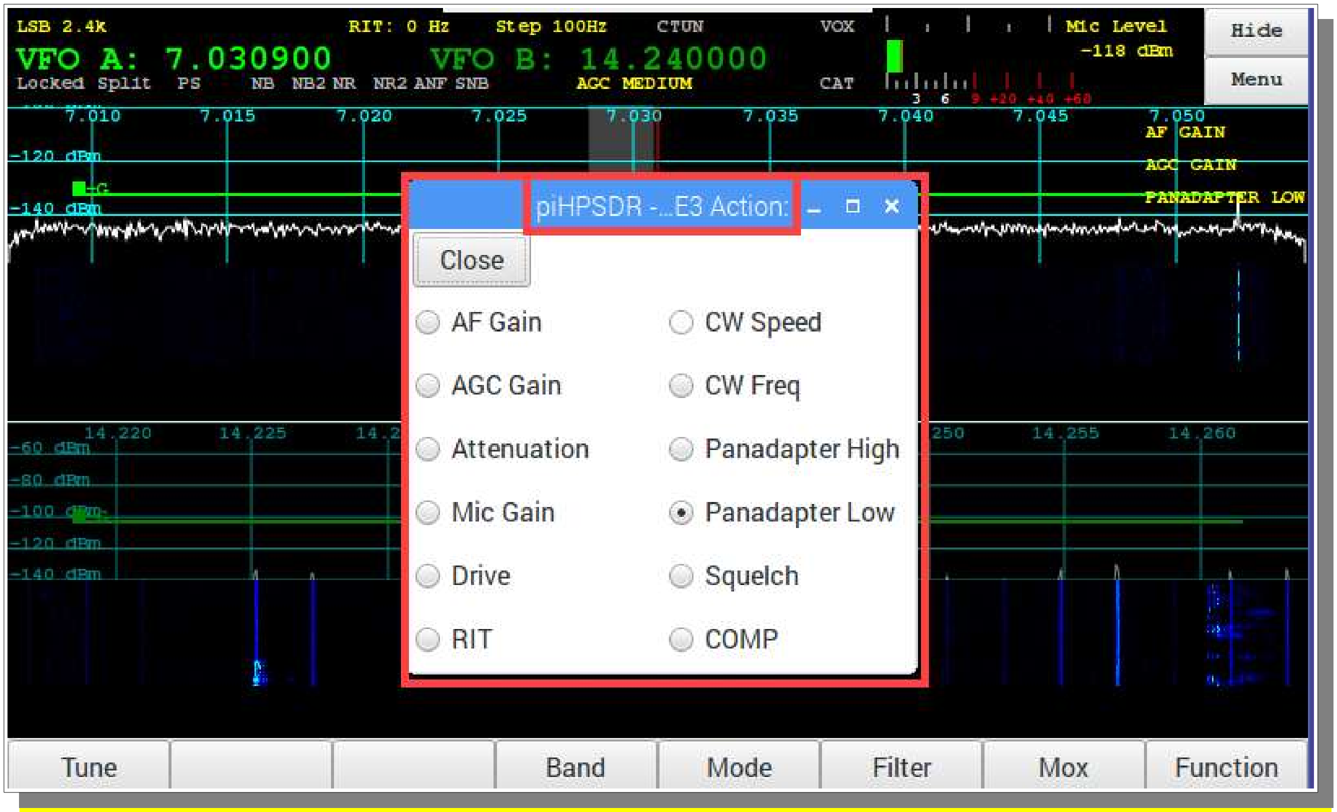
Function Group 4 menu items



The TUNE Toolbar gives quick access to keying the rig in CW and Generating TUNE watts.
Note: the FWD power and SWR measurements are indicated in the upper right**Shown here as 39W Forward Power and 1.0:1 SWR**

9. User assignable Actions for Encoders E1, E2, and E3

12 User Assignable actions for rotary encoders E1, E2, E3



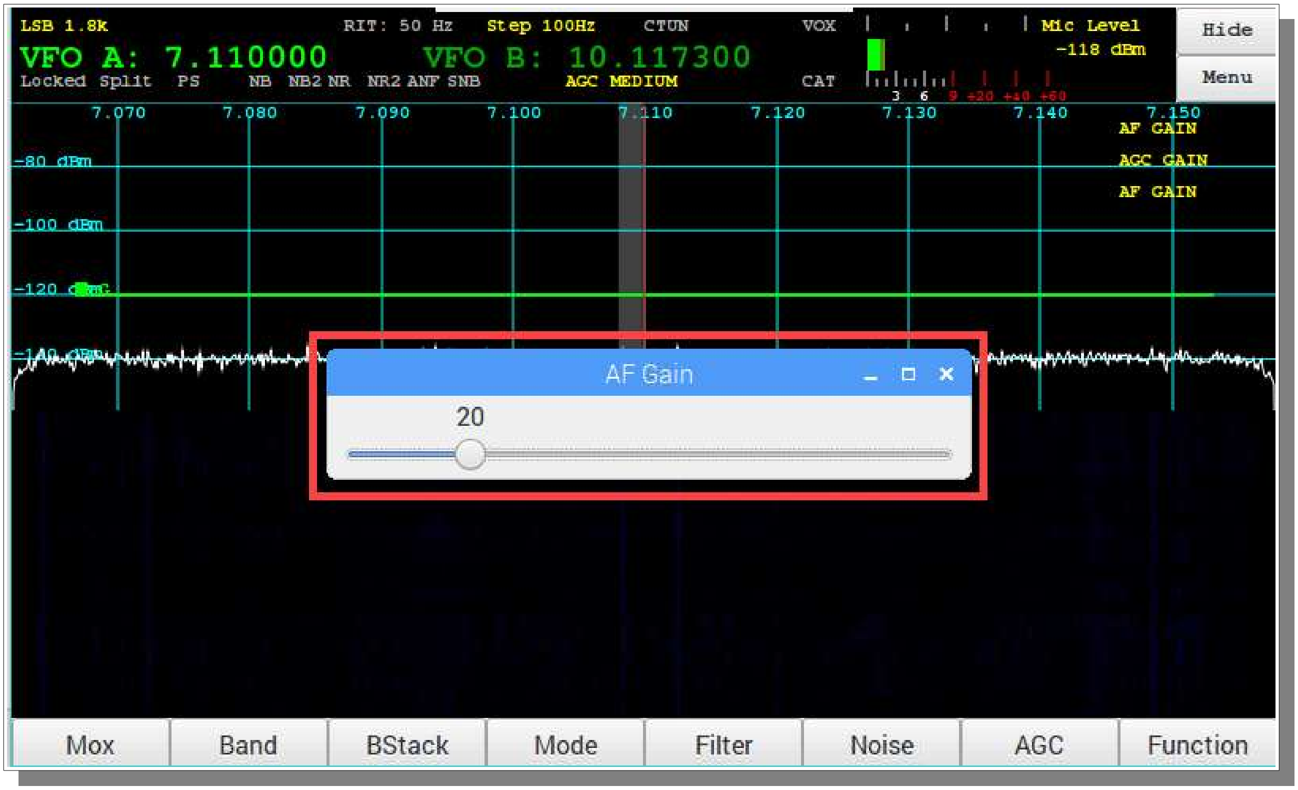
○ AF Gain ○ CW Speed
○ AGC Gain ○ CW Freq
○ Attenuation ○ Panadapter High
○ Mic Gain ● Panadapter Low
○ Drive ○ Squelch
○ RIT ○ COMP

Each rotary encoder E1, E2, E3 can be assigned to any one of the convenient twelve user selected items. Push on Encoder knob {1,2,3} and then rotate the knob to select the desired item:

- **AF Gain**
- **AGC Gain (Threshold)**
- **Attenuation**
- **Mic Gain**
- **Drive**
- **RIT**
- **CW Speed**
- **CW Sidetone Frequency**
- **Panadapter High**
- **Panadapter Low**
- **Squelch**
- **COMP compression**

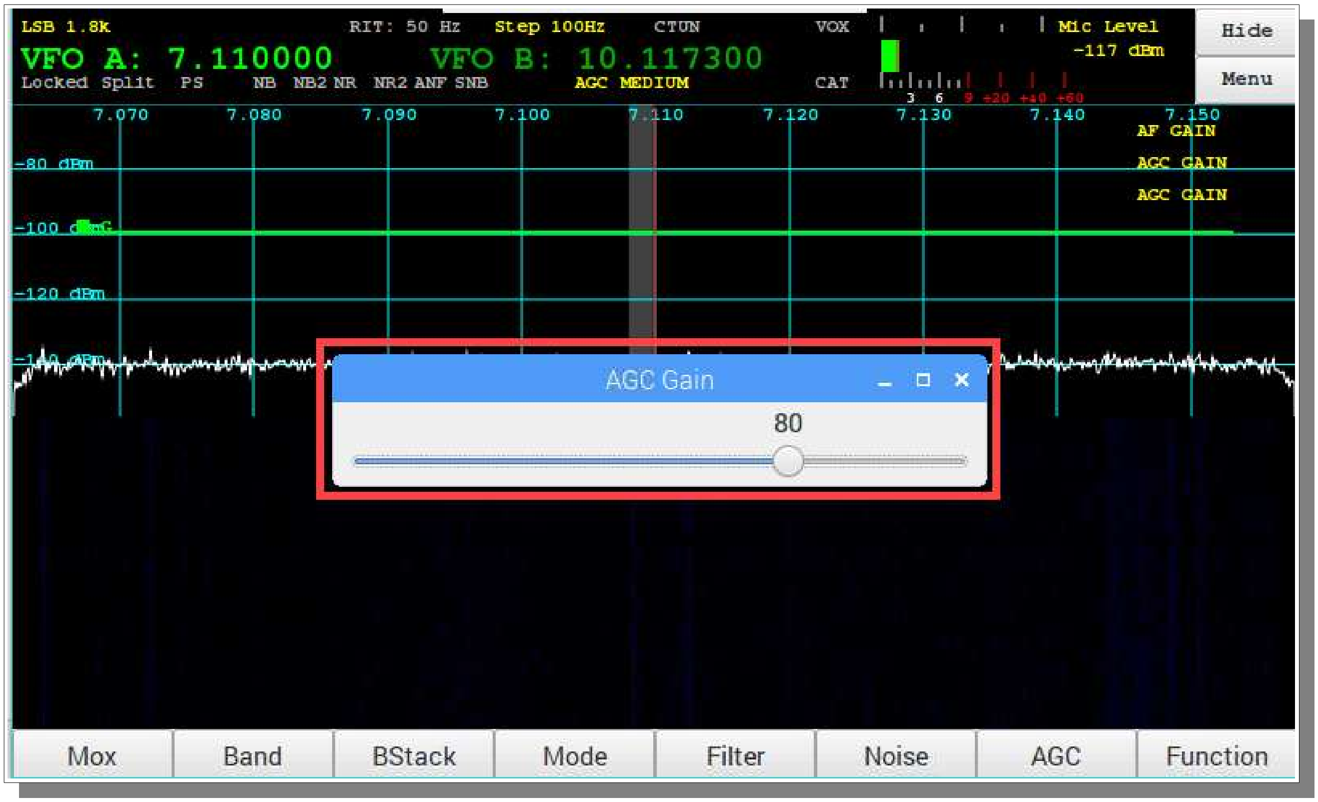
10. Individual Slider displays

The individual sliders are displayed when E1, E2, or E3 are rotated and Menu → Display → Sliders is turned off.



AF Gain slider is shown when DISPLAY → show sliders is unchecked

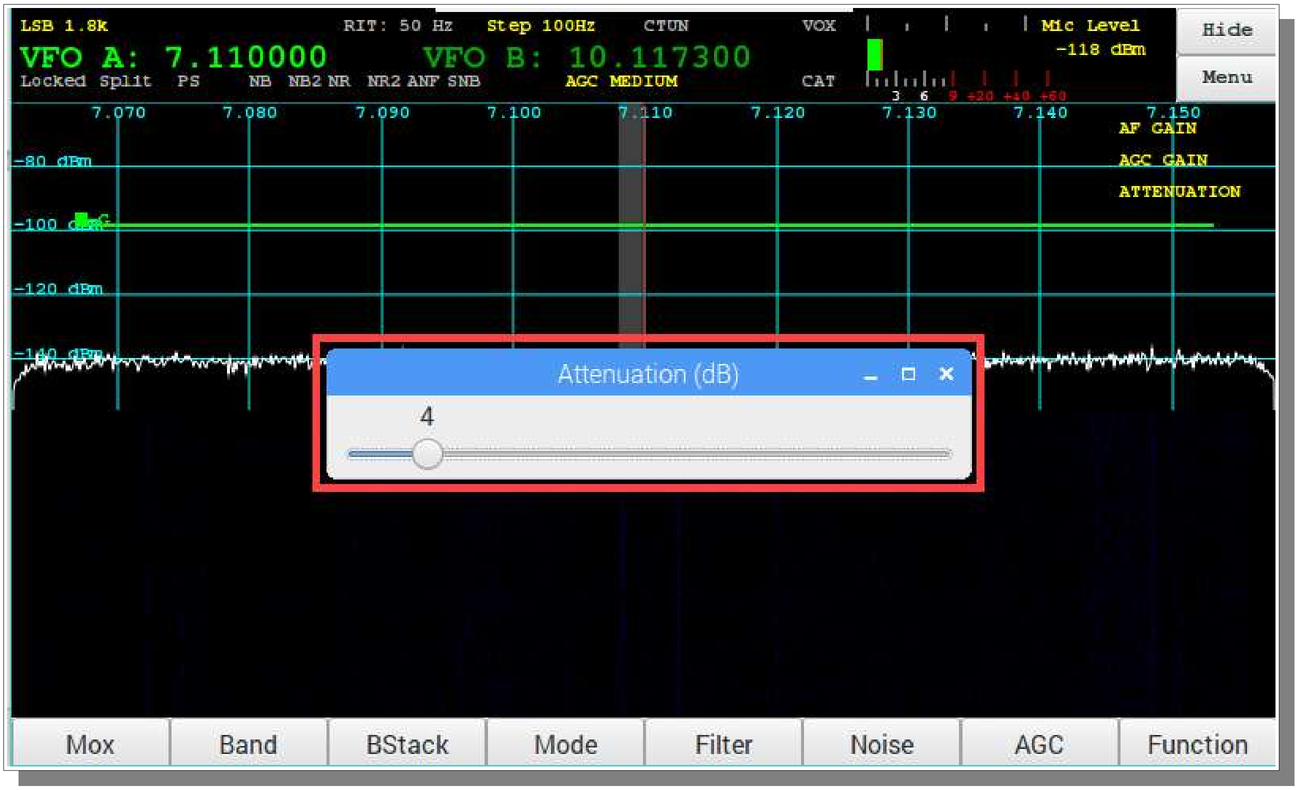
AGC Gain slider is shown when DISPLAY → show sliders is unchecked



AGC Gain

80

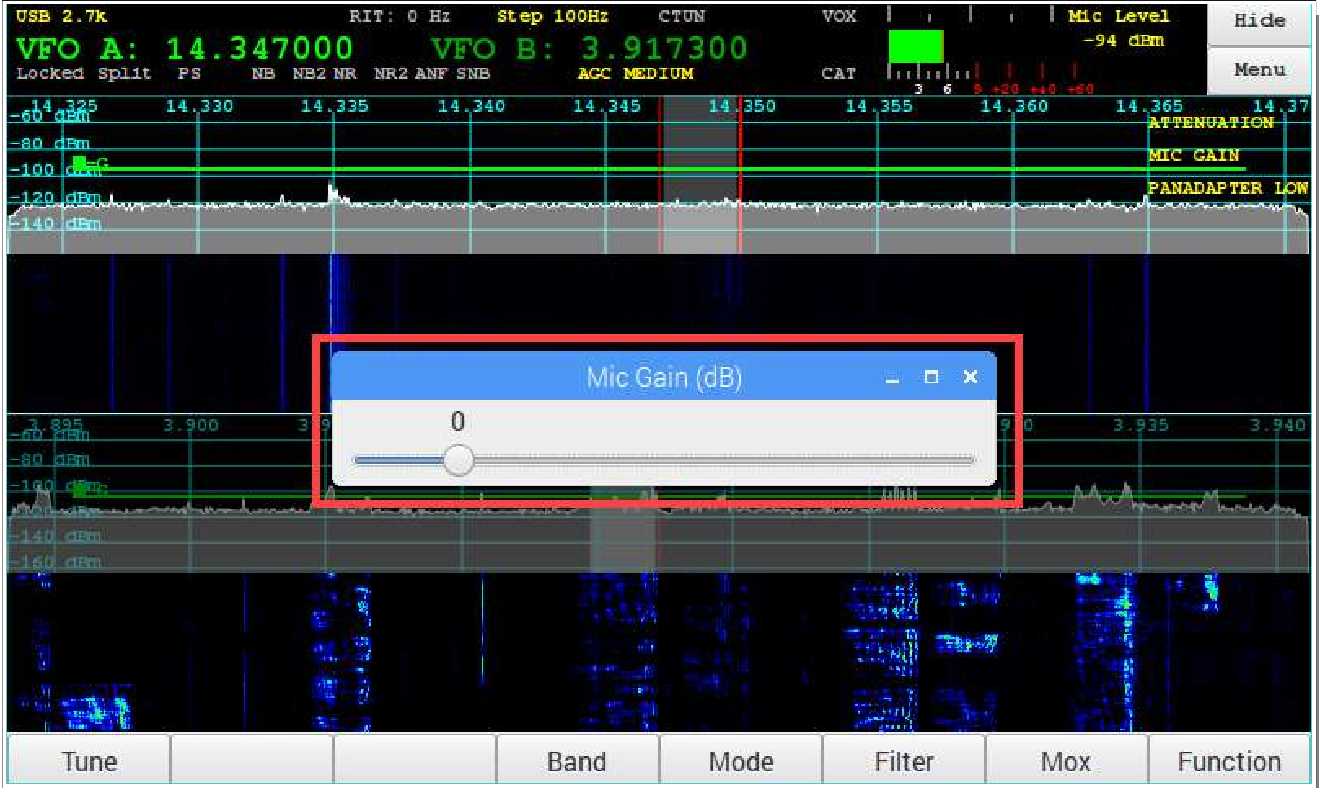
Receive Attenuation slider is shown when DISPLAY → show sliders is unchecked



Attenuation (dB)

4

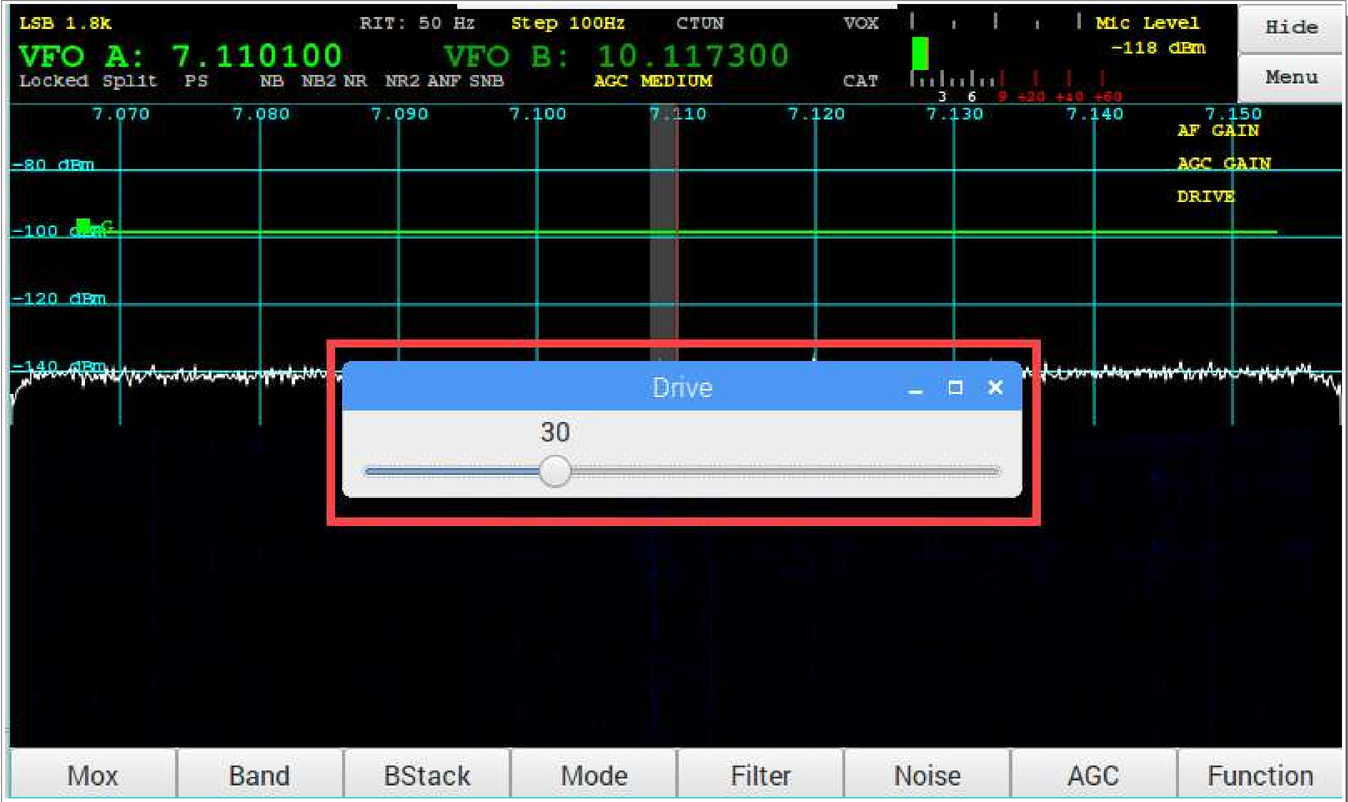
MIC Gain slider is shown when DISPLAY → show sliders is unchecked



Mic Gain (dB)

0

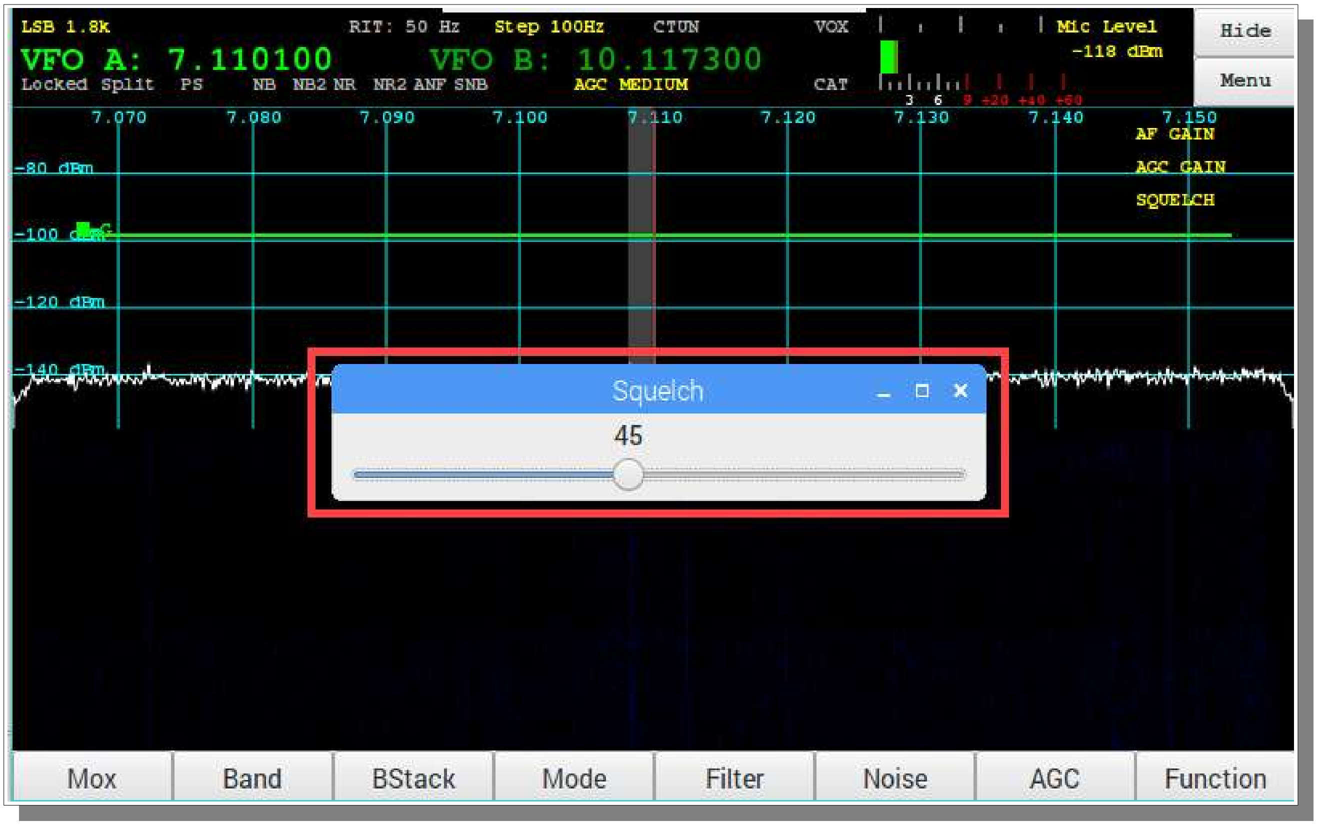
PA Drive slider is shown when DISPLAY → show sliders is unchecked



Drive

30

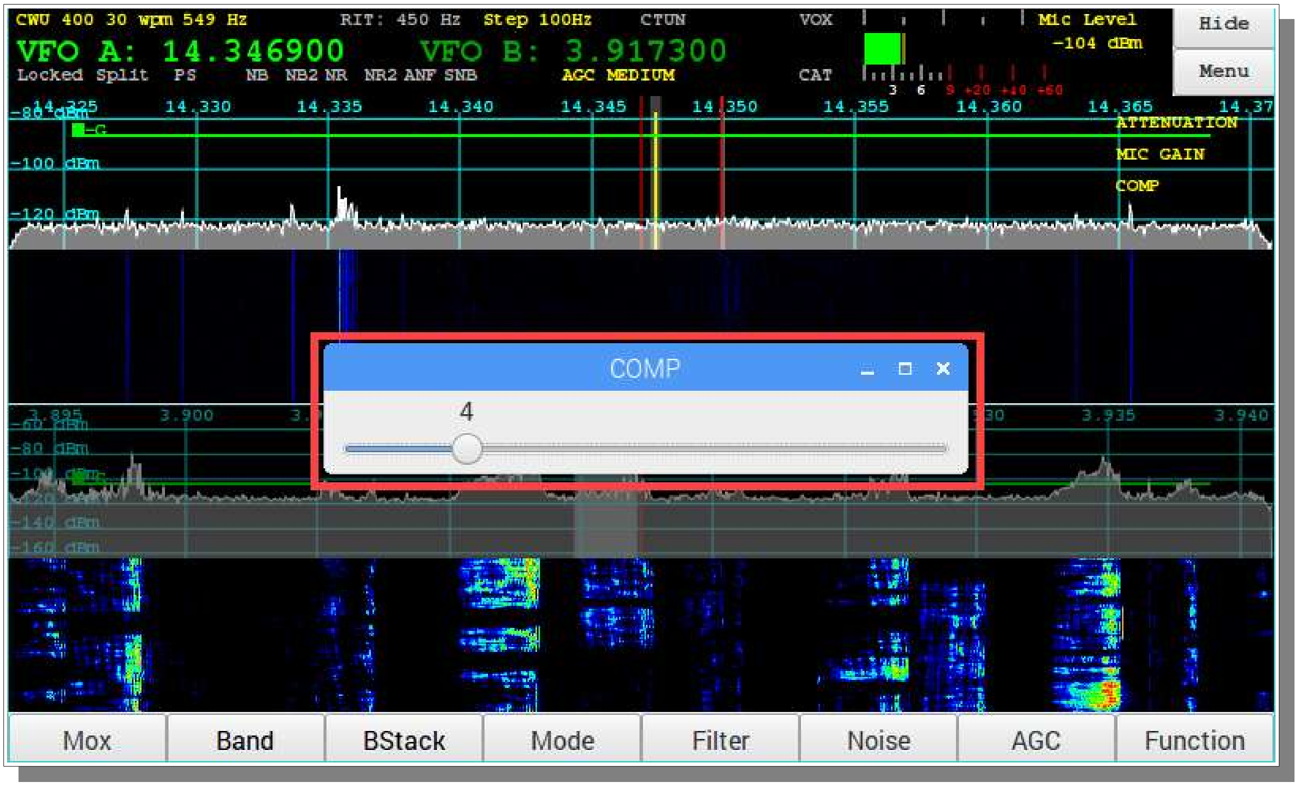
Squelch slider is shown when DISPLAY → **s**how sliders is unchecked



Squelch

45

MIC Compression is shown when DISPLAY → show sliders is unchecked



COMP

4

11. Tuning

VFO Encoder

The VFO encoder knob 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

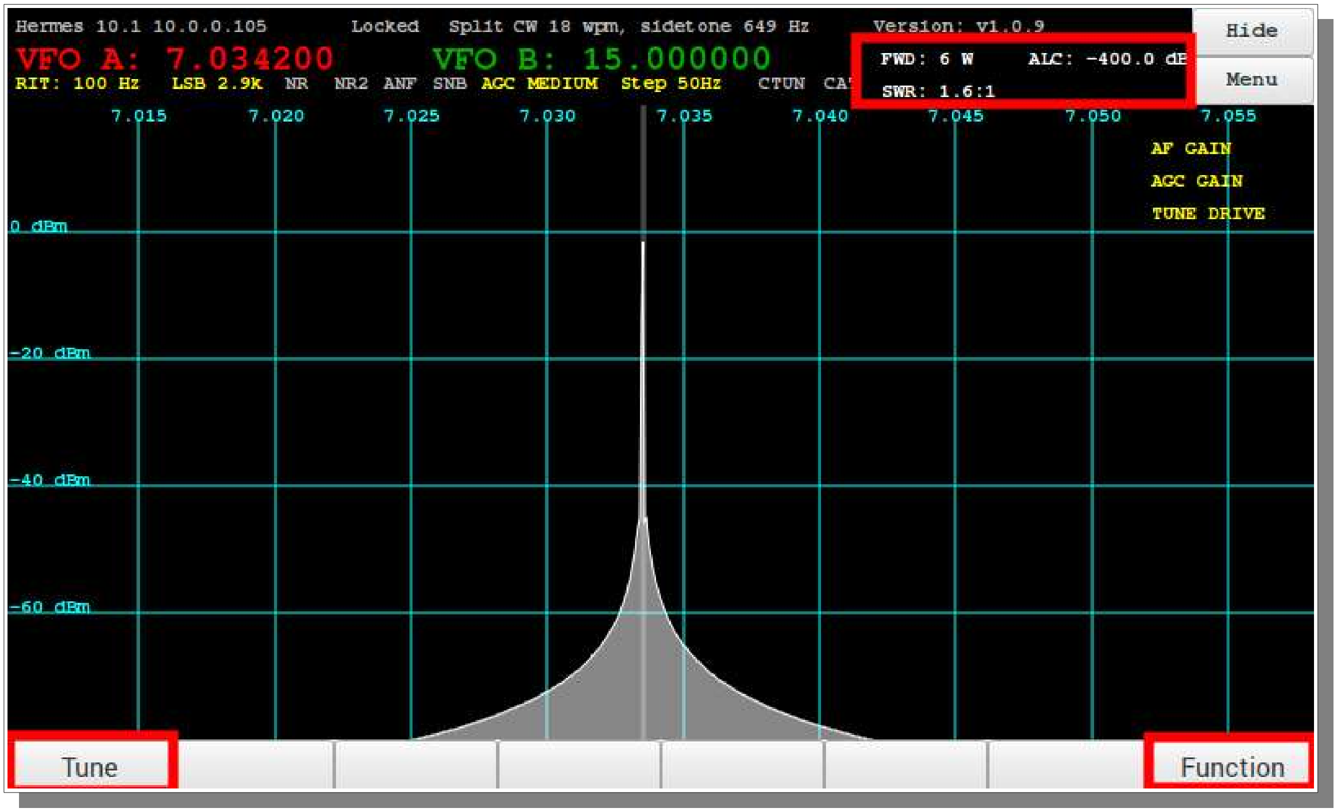
Left down and holding then dragging while on the panadapter or waterfall will move the frequency up or down.

Left clicking will move to the selected frequency.
Moving the scroll wheel will increment or decrement the frequency by the step value.

12. TUNE/SWR/FWD Power

TUNE/SWR/FWD power

When the Function button is 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.



Hermes 10.1 10.0.0.105 Locked Split CW 18 wpm, sidetone 649 Hz Version: v1.0.9

VFO A: 7.034200 VFO B: 15.000000

FWD: 6 W ALC: -400.0 dB

SWR: 1.6:1

RIT: 100 Hz LSB 2.9k NR NR2 ANF SNB AGC MEDIUM Step 50Hz CTUN CA

13. Appendix 13. Appendix

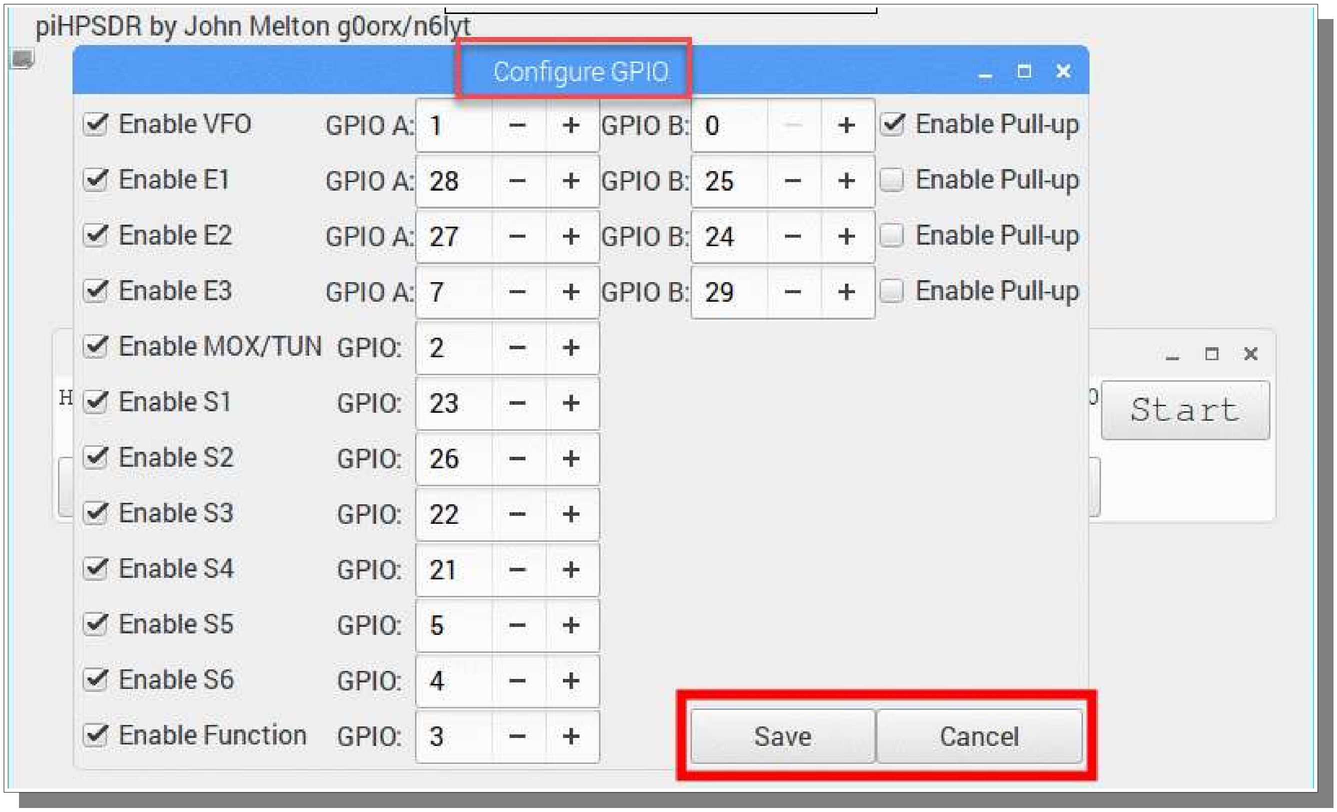
Encoders and Switches

Switch or Encoder	Function	Touch Screen	GPIO A	GPIO B
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	27	-----
MOX – MOX button	Space bar toggle for PTT or MOX button	Space bar toggle for PTT or MOX button	27	-----
S1 – Band	10 HF Bands + General Coverage + WWV + LF + XVRT	10 HF Bands + General Coverage + WWV + LF + XVRT	13	-----
S2 – BandStack	Four last used frequencies	Four last used frequencies	12	-----
S3 – Mode	LSB USB DSB CWL CWU FMN AM DRM FreeDV PSK	LSB USB DSB CWL CWU FMN AM DRM FreeDV PSK	6	-----
S4 – Filter	10 IF Filter widths	10 IF Filter widths	5	-----
S5 – Noise	5 Noise Reduction modes	5 Noise Reduction modes	24	-----
S6 – AGC	5 AGC Decay settings	5 AGC Decay settings	23	-----
FN – Function	Function switch to toggle TUNE or MOX button displayed on the bottom	Function switch to toggle TUNE or MOX and the action of E1, E2, E3	22	-----

Switch or Encoder	Function	Touch Screen	GPIO A	GPIO B
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	27	-----
MOX – MOX button	Space bar toggle for PTT or MOX button	Space bar toggle for PTT or MOX button	27	-----
S1 – Band	10 HF Bands + General Coverage + WWV + LF + XVRT	10 HF Bands + General Coverage + WWV + LF + XVRT	13	-----
S2 – BandStack	Four last used frequencies	Four last used frequencies	12	-----
S3 – Mode	LSB USB DSB CWL CWU FMN AM DRM FreeDV PSK	LSB USB DSB CWL CWU FMN AM DRM FreeDV PSK	6	-----
S4 – Filter	10 IF Filter widths	10 IF Filter widths	5	-----
S5 – Noise	5 Noise Reduction modes	5 Noise Reduction modes	24	-----
S6 – AGC	5 AGC Decay settings	5 AGC Decay settings	23	-----
	left of the screen	knobs		
E1 -- assignable	assignable	-------	20	26
E2 – assignable	assignable	-------	16	19
E3 – assignable	assignable	-------	4	21
E4 – VFO Main tuning knob	-------	-------	17	18

Switch or Encoder	**Function**	**Touch Screen**	**GPIO A**	**GPIO B**
Menu	**-------**	**Main piHPSDR Menu**	**13**	**-----**
Menu → Band		**Select band**	**-----**	**-----**
Menu → Band Stack	**3 to 5 level quick freq change**	**3 to 5 level quick freq change**	**12**	**-----**
Menu → Mode	**Mode**	**Mode**	**6**	**-----**
Menu → Filter	**IF Filter width**	**IF Filter Width**	**5**	**-----**
Menu → Noise	**Noise Blanker mode**	**Noise Blanker Mode**	**24**	**-----**
Menu → AGC	**Automatic Gain**	**Automatic Gain**	**4**	**21**
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**	**-----**	**-----**

GPIO pin Assignments (RaspberryPi



☑ Enable VFO	GPIO A: 1	GPIO B: 0	☑ Enable Pull-up
☑ Enable E1	GPIO A: 28	GPIO B: 25	☐ Enable Pull-up
☑ Enable E2	GPIO A: 27	GPIO B: 24	☐ Enable Pull-up
☑ Enable E3	GPIO A: 7	GPIO B: 29	☐ Enable Pull-up
☑ Enable MOX/TUN	GPIO: 2		
☑ Enable S1	GPIO: 23		
☑ Enable S2	GPIO: 26		
☑ Enable S3	GPIO: 22		
☑ Enable S4	GPIO: 21		
☑ Enable S5	GPIO: 5		
☑ Enable S6	GPIO: 4		
☑ Enable Function	GPIO: 3		

Note: You may wish to reverse the direction of Tuning on the VFO Encoder. The GIPIO table shown at the START of pihpsdr allows convenient text entry. For example, the VFO direction of TUNING can easily be changed by swapping GPIO-A from 1 to 0, and GPIO-B from 0 to 1.

14. Reference materials

GØORX

Jacinto Rebelo CU2ED for his homebrew of a pihpsdr Controller
Kjell Karlasen LA2NI for his complete RPi-e System
Scott WU2O homebrew Controller http://wu2o.dyndns.org/wu2o_pi_4.html
Bill Diaz KC9XG homebrew Controller and contributor to this manual
F'Hafen video on YouTube: https://www.youtube.com/watch?v=U7QfP28YjCw
Outstanding Video from F'Hafen 2016 with KV0S Editing http://openhpsdr.org/videos.php

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

In cooperation with **VK6PH**, **NRØV**, **W5WC**, **K5SO**, **KA6S**
and the OpenHPSDR Hardware and Software Projects
http://openhpsdr.org

all images and manufacturer data is copied here with permission of the owner

pihpsdr latest production version

https://github.com/gØorx/pihpsdr/releases

explanation of "Low Latency" buffer optional
June 10, 2016 - Warren, NR0V and Doug, W5WC

PowerSDR/OpenHPSDR_mRX_PS v3.4.2 has been released.
This release can be downloaded from the OpenHPSDR.org git repository.
This release contains the following changes:
SIGNIFICANTLY LOWER LATENCY
Receive latency is the time between when RF reaches your antenna and the corresponding audio is produced in your speaker or headphones. Similarly, transmit latency is, for example, the time between audio reaching your microphone and RF being on its way to your antenna. For many SDRs, especially those with sharp "brick wall" filters, the latency can be much larger than you might expect. Depending upon the radio design and various settings, SDR latencies can significantly exceed 100mS. Long latencies can create problems for the operator in contest operation, high-speed break-in CW, and even SSB rapid-turnaround VOX operation.
This release incorporates some technologies that allow us to achieve low latencies in the same category as leading conventional radios. Furthermore, we can do this with extremely sharp filters.

First of all, a couple basics:
*** Sometime ago, we moved CW Transmit from software to the FPGA in the radio hardware. This means that CW transmit latency was already very low, really based upon your delay settings which are chosen to avoid any hot-switching of relays.**

*** It has always been the case that the Buffer Size setting on the Setup=>Audio/Primary tab effects latency. The lower the size, the lower the latency. However, the lower the size, the more CPU cycles are required. Depending upon the speed of your computer, you may be limited in how low you can go. Fortunately, this is not likely to have such a large impact on your latency. For a very rough estimate of the latency due to this buffer, divide the buffer size by the sample rate. For example, a buffer of size 256, at a sample rate of 192K, contributes only about 256/192000 = 1.33mS.**

As of this release, there are some new features and corresponding controls to allow you to achieve much lower latency:
*** Up until this release, "Filter Size" and "DSP Buffer Size" have been the same and there has only been one setting, called "DSP Buffer Size." Filter Size determines how sharp your filters are; higher filter size leads to sharper filters. However, higher DSP Buffer Size leads to more latency because we must collect enough samples to fill the buffer before the buffer can be processed. As of this release, DSP Buffer Size and Filter Size are separate and can be set by mode on the Setup=>DSP/Options tab. So, using a very low DSP Buffer size minimizes latency and using a high Filter Size leads to sharper filters. The trade-off here is that using lower DSP buffer sizes requires somewhat more CPU cycles and using a high Filter Size does as well. With a reasonably fast computer, you will likely be able to run at a DSP Buffer Size of 64, the minimum, except, perhaps, for the FM mode. With filter sizes of 1024 or 2048, the sharpness of our filters rival the best radios. However, larger sizes, up to 16384, are available if you need them.**

*** You now have a choice of Filter Type, with two types available: Linear Phase and Low Latency. In the past, our filters have always been Linear Phase. Linear Phase filters have the property that all frequencies are delayed by the same amount of time as the signal is processed through the filter. This means that the time-domain waveform of a signal that is totally within the passband will look the same at the input of the filter and the output of the filter. The Low Latency filter does not strictly comply with this same type of operation. With the Low Latency filter, signals at frequencies very near the lower and upper edges of the passband may experience more delay than signals at other frequencies. Comparing the two types of filters, beta testers have reported little, if any, difference in sound quality, no problems with several digital modes that have been tested, and no significant negative impacts at all from using the Low Latency filters. However, both filter types are provided for your comparison and your choice. Of course, the Low Latency filters provide lower latency. In fact, the latency of Linear Phase filters increases linearly with Filter Size while the latency of the Low Latency filters is very low and nearly independent of Filter Size.**

Benchmark Comparisons:
*** For CW/SSB receive, using minimum Buffer Sizes and Low Latency filters, our beta testers have measured receive latencies in the 15mS to 20mS range. Using minimum Buffer Sizes and Linear Phase filters, the latencies are 25mS to 30mS for a Filter Size of 1024 and 35mS to 40mS for a Filter Size of 2048. Using features such as noise blankers, EQ, and noise reduction will add some amount to that, depending upon the feature(s) and settings. These numbers compare with ~65mS and ~120mS using DSP Buffer sizes of 1024 and 2048, respectively, in prior software releases.**

NOTE
You WILL need to reset your database. This release will build a new wisdom file on first time use. Depending on your system, it may take a very long time to complete. Please be patient.

Thanks & 73,
Warren, NR0V
Doug, W5WC

---

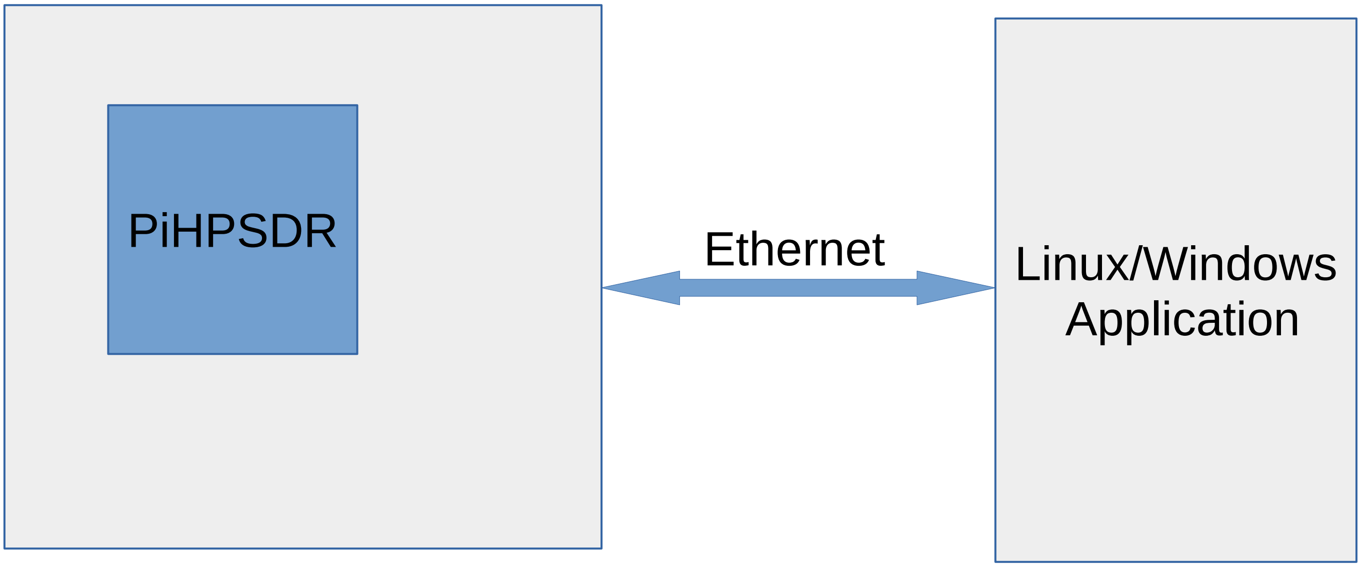
15. Steve Wilson KA6S special addition for Fldigi

Using PiHPSDR with other applications:
PiHPSDR emulates several commands available in the CAT control found in the TS-2000. However, it receives commands over TCP/IP instead of through an RS-232 connection. For some applications this is trivial – for others some internet plumbing is required.

The Basics

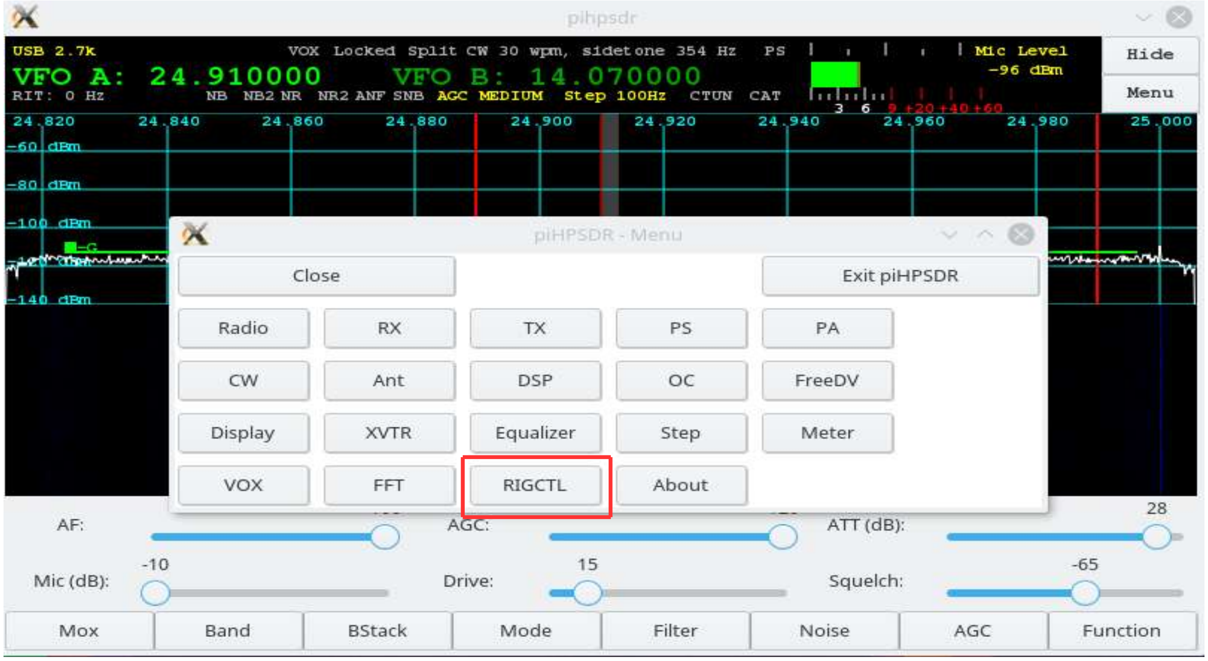
PiHPSDR listens on Port 19090 for TCP/IP connections. It interprets TS-2000 protocol commands and will respond appropriately. It is NOT a full implementation. Some things don't make sense, i.e. they are features not shared between the TS-2000 and PiHPSDR. Others are still to be implemented.

Note: The latest version of Hamlib has a radio definition for the PiHPSDR, thus any application that uses Hamlib will be able to talk to PiHPSDR in its native manner, i.e. directly through TCP/IP without using the serial port. (Thanks to Jae, K5JAE for the hamlib port!) The serial port methodology is presented here for older iterations of Hamlib. To get access to to native support you will need to compile Hamlib from source and install it. That is beyond the scope of this manual.

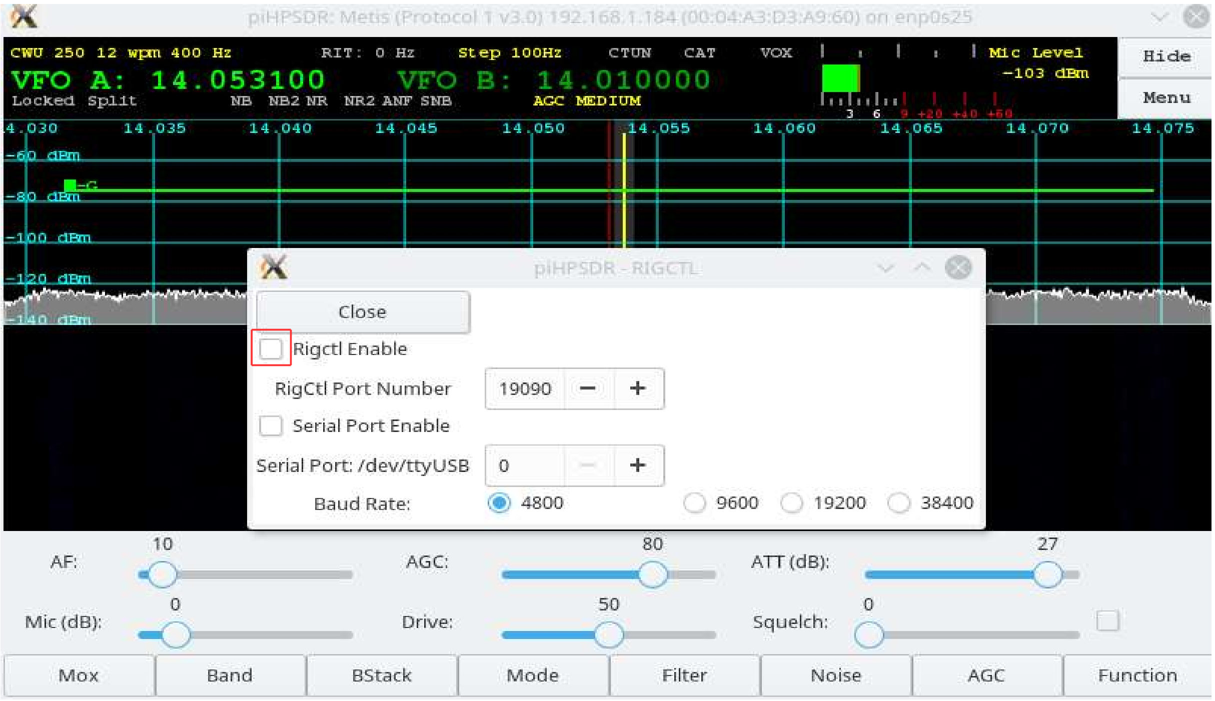


Use the Raspberry PI
TCP/IP Address
And set for Port 19090

Note that before you can begin using this connection you must configure and enable it through the main menu. This is done by choosing RIGCTL from the Main Menu:



Upon hitting the RIGCTL button the configuration menu is presented.



☐ Rigctl Enable

RigCtl Port Number: 19090 − +

☐ Serial Port Enable

Serial Port: /dev/ttyUSB 0 − +

Baud Rate: ● 4800 ○ 9600 ○ 19200 ○ 38400

By hitting the Rigctl Enable Button you can turn on the CAT control system. Hit Close to continue. Note that if you exit the program at this point RIGCTL will remain enabled. It is also possible to change the default Port number by using the -/+ buttons to decrement/increment the port number.

ping a Serial Port to TCP/IP port in Linux

Perhaps the easiest way to do this is via the utility “socat” available in most Linux distributions. This utility is sort of a swiss army knife for interconnectivity, and one its abilities is to map Linux Serial ports to TCP/IP ports.

To obtain “socat” and install it on an Ubuntu system type:

```
sudo apt-get install socat
```

To use it – first start PiPHSDR. You need to know the TCP/IP address of your Raspberry Pi. This will be a 32 bit number formatted as XX.XX.XX.XX. I'll use 192.168.1.73 in some examples, 192.168.1.76 in other examples. These are where I have HPSDR radios assigned in my local network.

To run it – type:

```
socat  pty,link=/tmp/vtty,raw  tcp:192.168.1.73:19090&
```

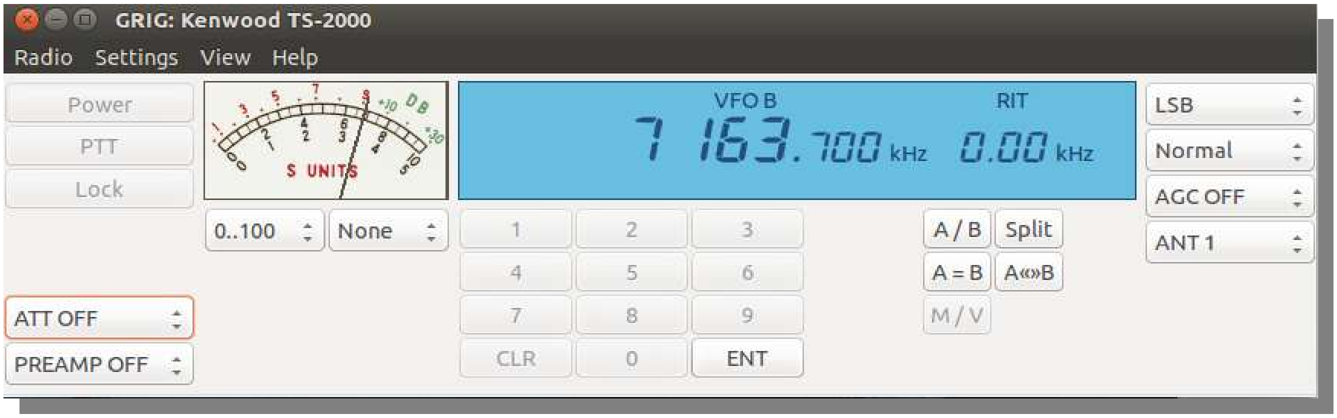
This will do two things. It creates a fake serial port called /tmp/vtty that can be opened by most applications and acts just like a serial port.

The next step is to point your application at /tmp/vtty where you would normally choose a serial port and set it to talk to a TS-2000. It really is that simple!

This allows applications that are built to use Hamlib to talk to PiHPSDR.

Example: Set up GRIG to talk to PiHPSDR

```
grig –-model=214 –-rig-file=/tmp/vtty *
```



/tmp/vtty is a temporary file – it disappears as soon as socat ends.

Grig is built around the Hamlib radio access library. The TS-2000 is model 214 in Hamlib. Any application that uses Hamlib should be able to communicate with PiHPSDR by using /dev/vtty and choosing model 214 as the radio.

Something else to realize is that more than one application can talk to PIHPSDR at a time. PiHPSDR supports multiple connections to the CAT port.

If you have the latest version of Hamlib that supports PiHPSDR natively – the command to start grig would be:

```
grig –model=240 –rig-file=192.168.1.73:19090 &
```

Example: Connect FLDIGI to PiHPSDR

FLDIGI can be connected to PiHPSDR using the native PiHPSDR interface built into the latest version of Hamlib. However, it polls PiHPSDR way to fast.

Have no fear – FLRIG is here. FLRIG is a companion application to FLDIGI that can act as a server for FLDIGI.

You can install FLRIG on with:

```
sudo apt-get install flrig
```

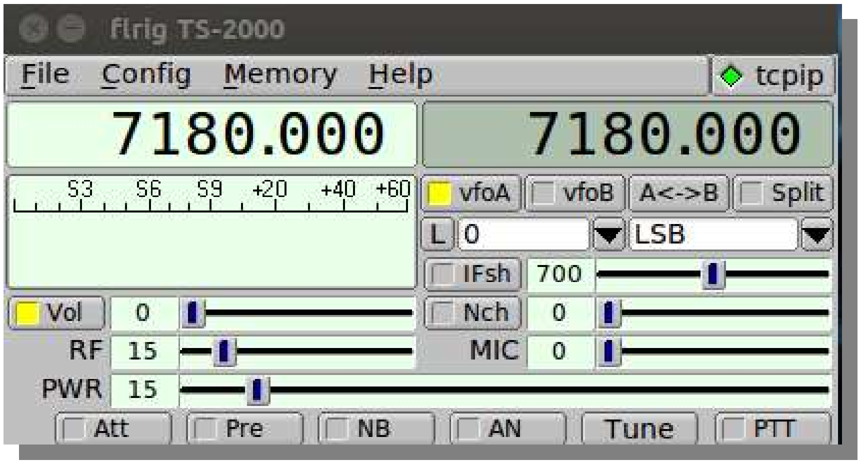
You can install FLDIGI with:

```
sudo apt-get install fldigi
```

Start FLRIG first with

```
flrig
```

No need to start it with sudo since it talks directly to PiHPSDR.



7180.000

7180.000

L 0

LSB

IFsh 700

Vol 0

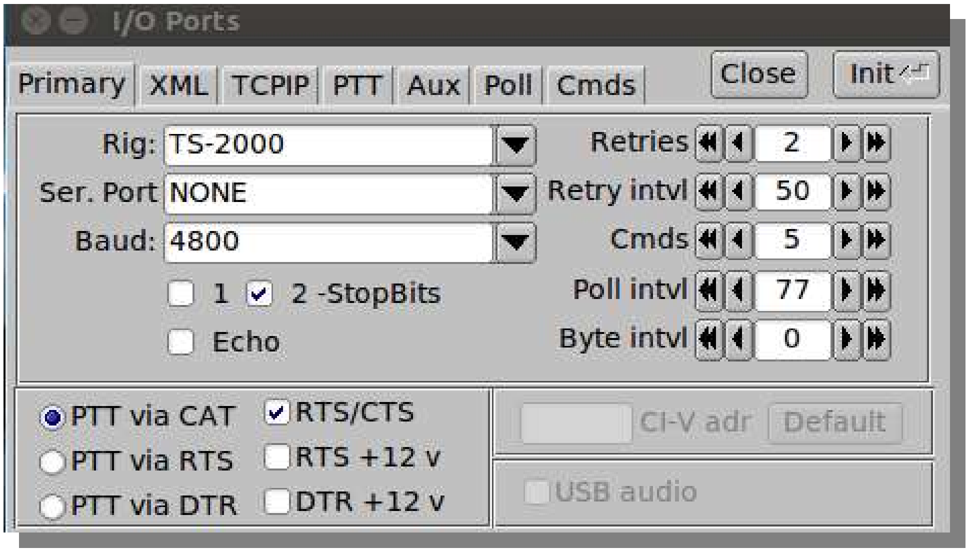
Nch 0

RF 15

MIC 0

PWR 15

To set up FLRIG – Hit the Config button on the top of the application and Choose Config/Setup/Transceiver – the Window below will pop up.



Rig: TS-2000
Ser. Port NONE
Baud: 4800
☐ 1 ☑ 2 -StopBits
☐ Echo
Retries 2
Retry intvl 50
Cmds 5
Poll intvl 77
Byte intvl 0
● PTT via CAT
○ PTT via RTS
○ PTT via DTR
☑ RTS/CTS
☐ RTS +12 v
☐ DTR +12 v
CI-V adr Default
☐ USB audio

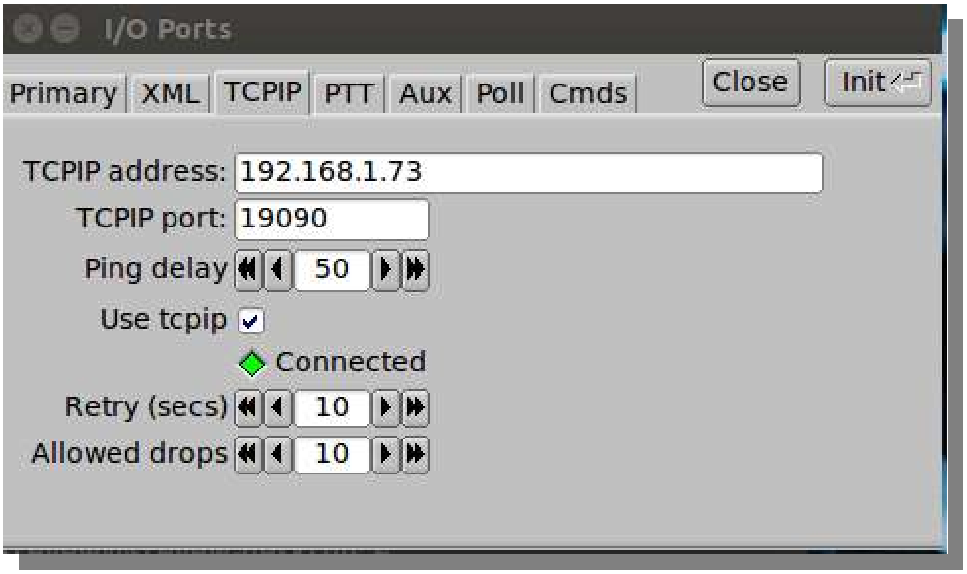
Use the arrow to the right of the Rig: box and slide it to TS-2000.

Also – check the PTT via CAT button in the lower left corner.

Hit Close

Now Choose Config/Setup/TCPIP

The window below will pop up.



TCPIP address: 192.168.1.73
TCPIP port: 19090
Ping delay 50
Use tcpip ☑
◆ Connected
Retry (secs) 10
Allowed drops 10

Select the TCPIP tab. Enter the TCPIP address of your Raspberry Pi – in our example you see 192.168.1.73. Type in the TCPIP box 19090 and that should do it. Hit the “Use tcpip” button. If the connected light isn't green – hit the Init button in the upper right portion of the window.

Hit Close and you should have FLRIG up and controlling your PiHPSDR.

The Frequency should work by using the scroll wheel over the digit you want to change.

The volume, RF, PWR, and MIC sliders should all work.

The PTT should cause the radio to transmit.

The S Meter should register.

Now get FLDIGI running.

Type:

fldigi

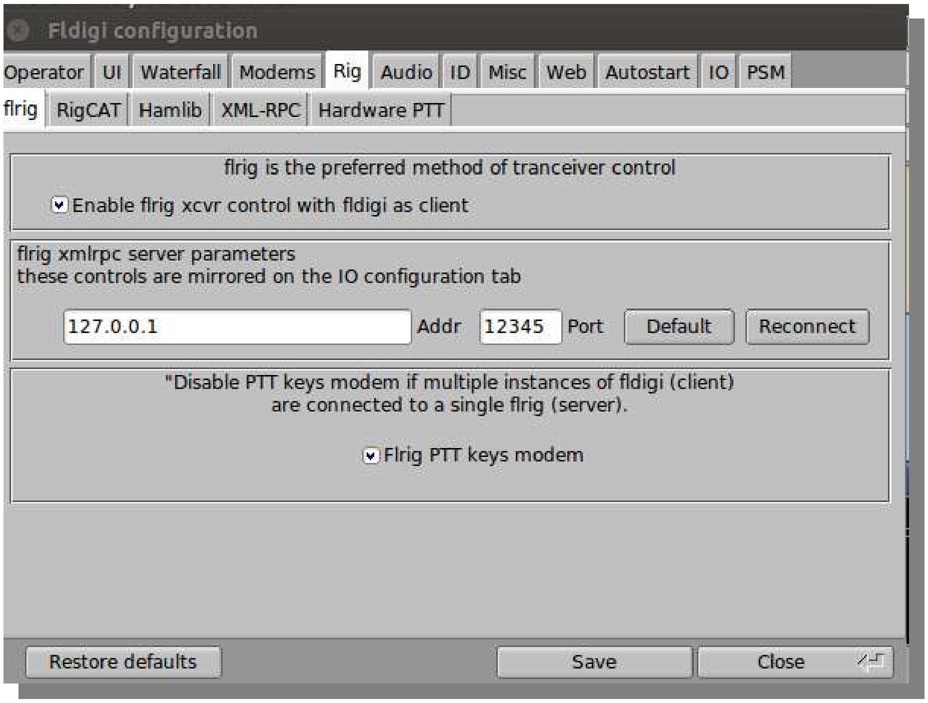
Choose Configure/Rig Control

Hit the flrig tab and the window below will pop up.

Hit the “Enable flrig xcvr control with fldigi as client” button.

Note that the TCP/IP address is 127.0.0.1 – which is the same as the machine you are running flrig and fldigi on. The TCP/IP port 12345 should already be configured.

Also hit the “Flrig PTT keys modem”



flrig is the preferred method of tranceiver control

☑ Enable flrig xcvr control with fldigi as client

flrig xmlrpc server parameters
these controls are mirrored on the IO configuration tab

Addr: 127.0.0.1
Port: 12345

"Disable PTT keys modem if multiple instances of fldigi (client)
are connected to a single flrig (server).

☑ Flrig PTT keys modem

That is it.

You need to use actual audio cables from the Radio to your Linux computer for the audio as of now – but this should be all that is necessary for FLDIGI to talk to your radio!

Setting up rigctld

The Hamlib system must be installed to use some packages – specifically to run the rigctld daemon.

On Ubuntu:

sudo apt-get install libhamlib-utils
sudo apt-get install libhamlib2

Note: These packages don't yet have PiHPSDR supported natively. It is necessary to compile the latest version from source and that is beyond the scope of this manual.

Multi-Client Support

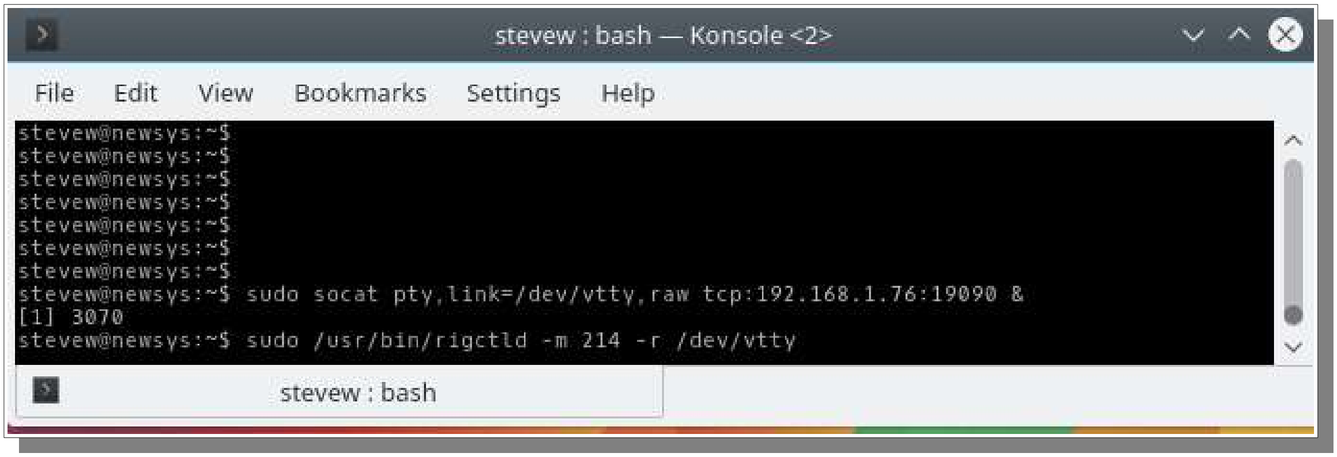
PiHPSDR can communicate with multiple independent clients simultaneously. An example of this might be running FLRIG/FLDIGI with the logging program CQLOG.

First – make sure you have CQLOG installed.

On Ubuntu:
sudo apt-get install CQLOG

We need do three things to get cqrlog running.
1) Create a virtual terminal – see the instructions above about running cqrlog. There is one change here. We're going to aim the virtual serial port /tmp/vtty at TCP/IP port 19090 again.
2) Start up the hamlib rigctld – this is a “daemon” in Unix parlance that is responsible for interfacing between applications and the virtual terminal using the internal hamlib protocol. Note that SOME applications have the hamlib software built in, while others use the daemon – cqrlog uses the daemon. The rigctld must be started using “sudo”
3) Start up cqrlog. Specifically do NOT use “sudo” for this command.

In a single xterm – you can start up both socat AND the rigctld daemon. Note again the use of “sudo” prior to the command AND note that the first program can be put into the background safely. Rigctld doesn't work if you do that..



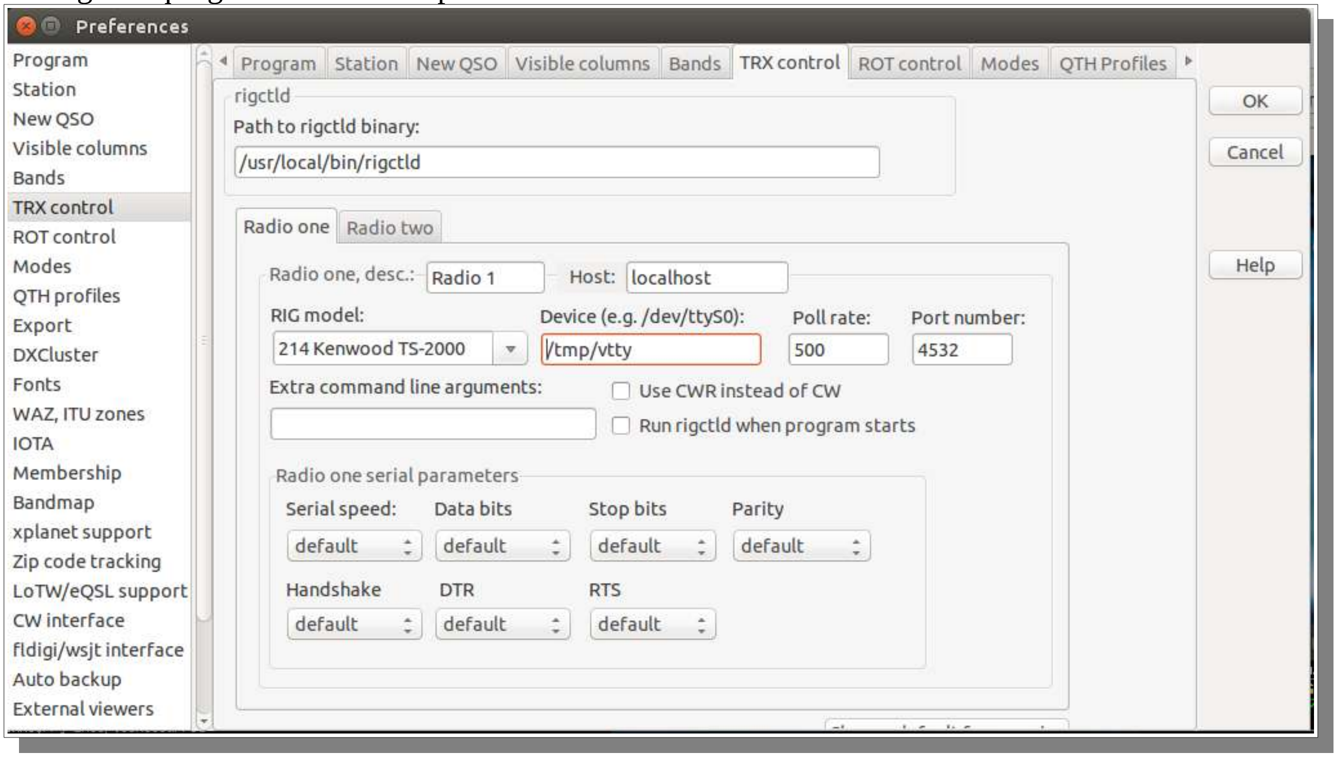
```
stevew@newsys:~$
stevew@newsys:~$
stevew@newsys:~$
stevew@newsys:~$
stevew@newsys:~$
stevew@newsys:~$
stevew@newsys:~$
stevew@newsys:~$ sudo socat pty,link=/dev/vtty,raw tcp:192.168.1.76:19090 &
[1] 3070
stevew@newsys:~$ sudo /usr/bin/rigctld -m 214 -r /dev/vtty
```

In another xterm – start up cqrlog

Enter: cqrlog &

This will start up cqrlog talking the PiHPSDR radio using port 19091. Note that this can be safely put into the background using the ampersand operator.

Configure cqrlog as shown in the picture below:



rigctld

Path to rigctld binary: /usr/local/bin/rigctld

Radio one, desc.: Radio 1

Host: localhost

RIG model: 214 Kenwood TS-2000

Device (e.g. /dev/ttyS0): /tmp/vtty

Poll rate: 500

Port number: 4532

Extra command line arguments:

☐ Use CWR instead of CW

☐ Run rigctld when program starts

Radio one serial parameters

Serial speed: default

Data bits: default

Stop bits: default

Parity: default

Handshake: default

DTR: default

RTS: default

Note that we are choosing to NOT launch rigctld when the program starts – but rather starting it by hand.

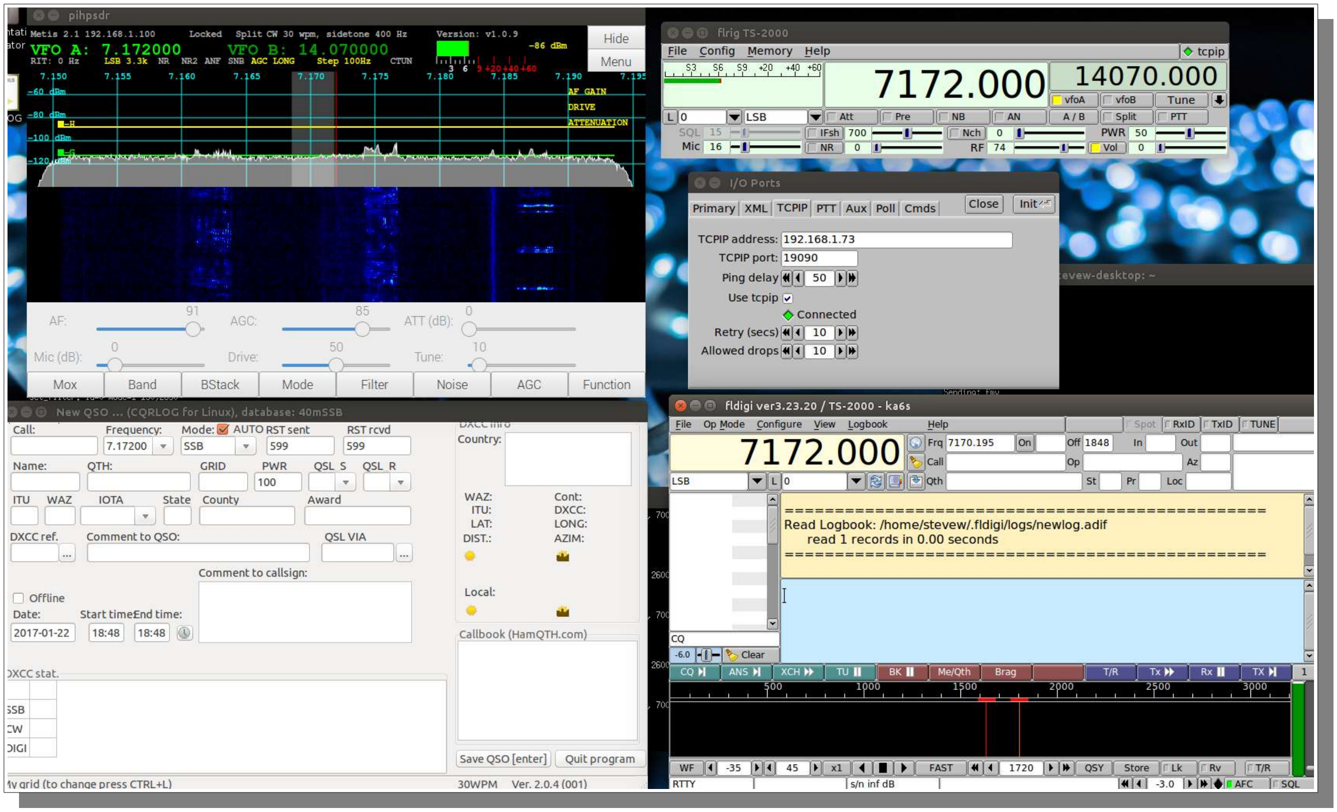
Next – start up flrig as you have previously using port 19090. It should be configured correctly from the first time you used it.   Since cqrlog spouts messages – you may need to use another xterm…

% flrig &

And this you can start up FLDIGI using the same xterm..

%fldigi

The picture below shows all applications running – using the two TCP/IP ports.


pihpsdr
Hide
Menu
AF GAIN
DRIVE
ATTENUATION
Mox
Band
BStack
Mode
Filter
Noise
AGC
Function
flrig TS-2000
File Config Memory Help
I/O Ports
Primary XML TCPIP PTT Aux Poll Cmds
Close
Init
fldigi ver3.23.20 / TS-2000 - ka6s
File Op Mode Configure View Logbook Help
CQ ANS XCH TU BK Me/Qth Brag T/R Tx Rx TX
New QSO ... (CQRLOG for Linux), database: 40mSSB
Save QSO [enter]
Quit program

Metis 2.1 192.168.1.100 Locked Split CW 30 wpm, sidetone 400 Hz Version: v1.0.9

VFO A: 7.172000 VFO B: 14.070000 -86 dBm

RIT: 0 Hz LSB 3.3k NR NR2 ANF SNB AGC LONG Step 100Hz CTUN

AF: 91 AGC: 85 ATT (dB): 0

Mic (dB): 0 Drive: 50 Tune: 10

flrig: 7172.000 14070.000 tcpip

L 0 LSB

SQL 15 IFsh 700 Nch 0 PWR 50

Mic 16 NR 0 RF 74 Vol 0

TCPIP address: 192.168.1.73

TCPIP port: 19090

Ping delay 50

Use tcpip ☑

Connected

Retry (secs) 10

Allowed drops 10

fldigi: 7172.000 Frq 7170.195 Off 1848

Read Logbook: /home/stevew/.fldigi/logs/newlog.adif
read 1 records in 0.00 seconds

Frequency: 7.17200 Mode: ☑ AUTO SSB RST sent 599 RST rcvd 599

PWR 100

Date: 2017-01-22 Start time 18:48 End time: 18:48

Callbook (HamQTH.com)

30WPM Ver. 2.0.4 (001)

Running applications with the latest version of Hamlib

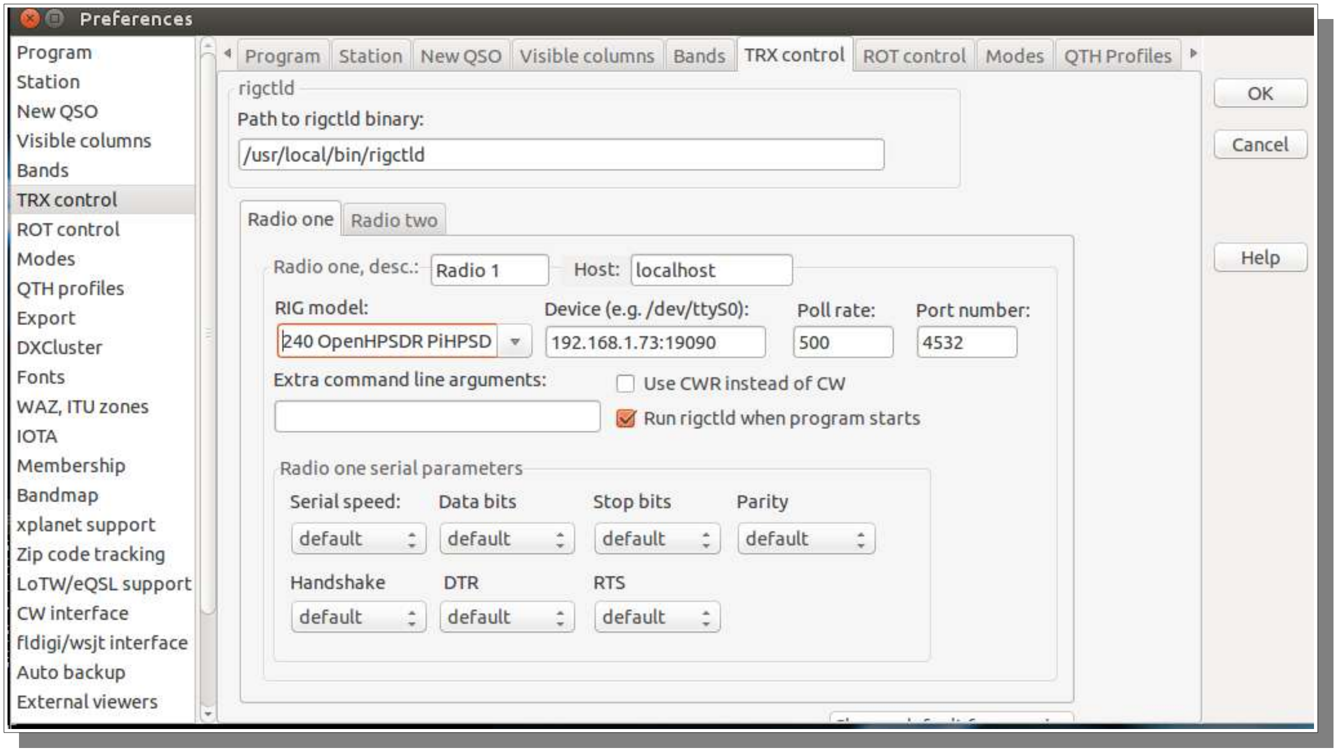
The biggest advantage of running applications through latest version of Hamlib is that the whole virtual terminal concept is removed! Applications can talk directly to PiHPSDR without any middleware. I'll use CQRLOG to illustrate the setup.

The first trick is to make rigctld runnable without being root. THIS IS A SECURITY HOLE – Proceed at your own risk! The following command allows rigctld to be run without being root…

```
sudo chown pi /usr/local/bin/rigctld
```

This makes the user “pi” owner of the rigctld daemon.

Next – start up cqrlog and set up its preferences as shown in the next picture.



rigctld

Path to rigctld binary: /usr/local/bin/rigctld

Radio one, desc.: Radio 1

Host: localhost

RIG model: 240 OpenHPSDR PiHPSD

Device (e.g. /dev/ttyS0): 192.168.1.73:19090

Poll rate: 500

Port number: 4532

Extra command line arguments:

☐ Use CWR instead of CW

☑ Run rigctld when program starts

Radio one serial parameters

Serial speed: default

Data bits: default

Stop bits: default

Parity: default

Handshake: default

DTR: default

RTS: default

Things to note:
Rig Model is OpenHPSDR PiHPSDR – radio model 240.
Device: Use the TCP/IP/Port number – so 192.168.1.73:19090
The “Run rigctld when program starts” is checked – since we don't have to root to run rigctld.

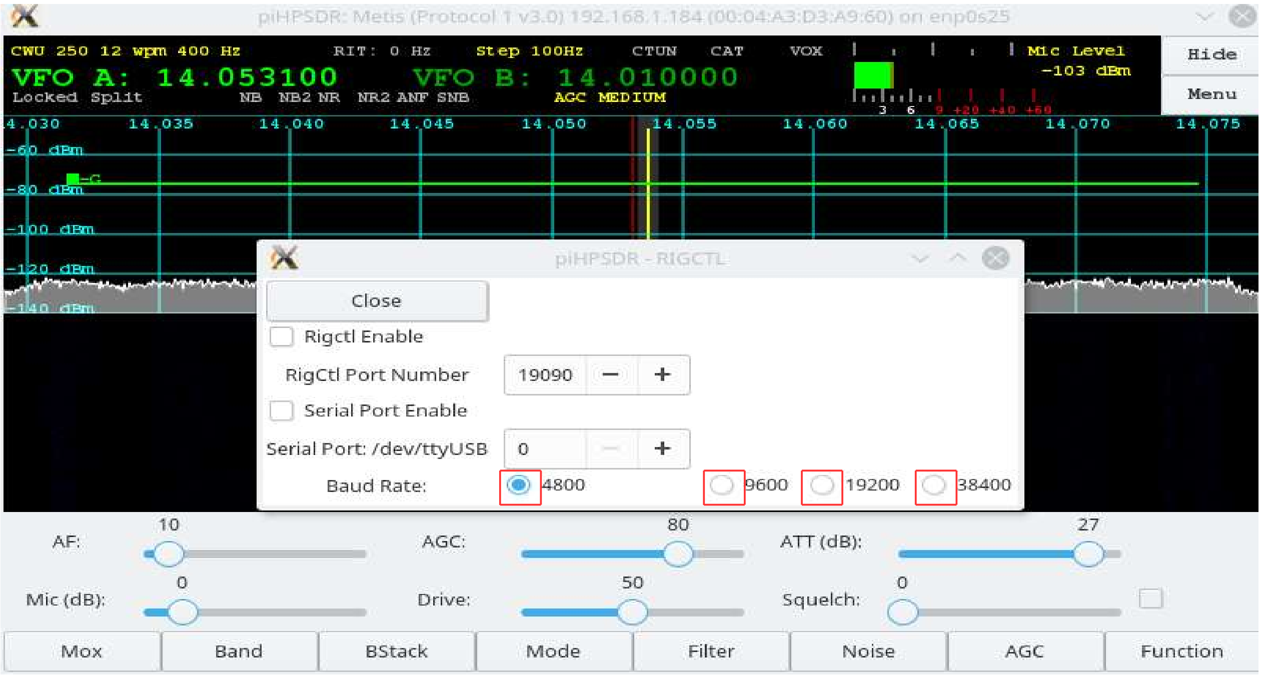
It is also possible to start cqrlog on OTHER than the Raspberry Pi. The settings would be the SAME since the device entry is what actually aims the communications at the PiHPSDR TCP/IP port. Rigctld can be started on another machine and it will be able to communicate via TCP/IP directly to the PiHPSDR application.

Using CAT through a Serial Port

The latest update to PiHPSDR can also operate over a USB Serial port. This will allow normal connection from a PC to PiHPSDR through a serial port similar to other radio CAT connections. Note that multiple connections are concurrently available through the TCP/IP connection as previously described after the Rigctl Enable button is lit. These two paths function independently.

It is important to set up the Baud Rate and choose the USB port BEFORE you enable the Serial Port! It is also important to plug in the USB to Serial adapter before you try and enable the function. Note that the Serial Port Enable will NOT stay lit if there is no /dev/ttyUSBx port available.

Configure the Baud Rate by selecting one of the 4 choices shown below:



☐ Rigctl Enable

RigCtl Port Number: 19090

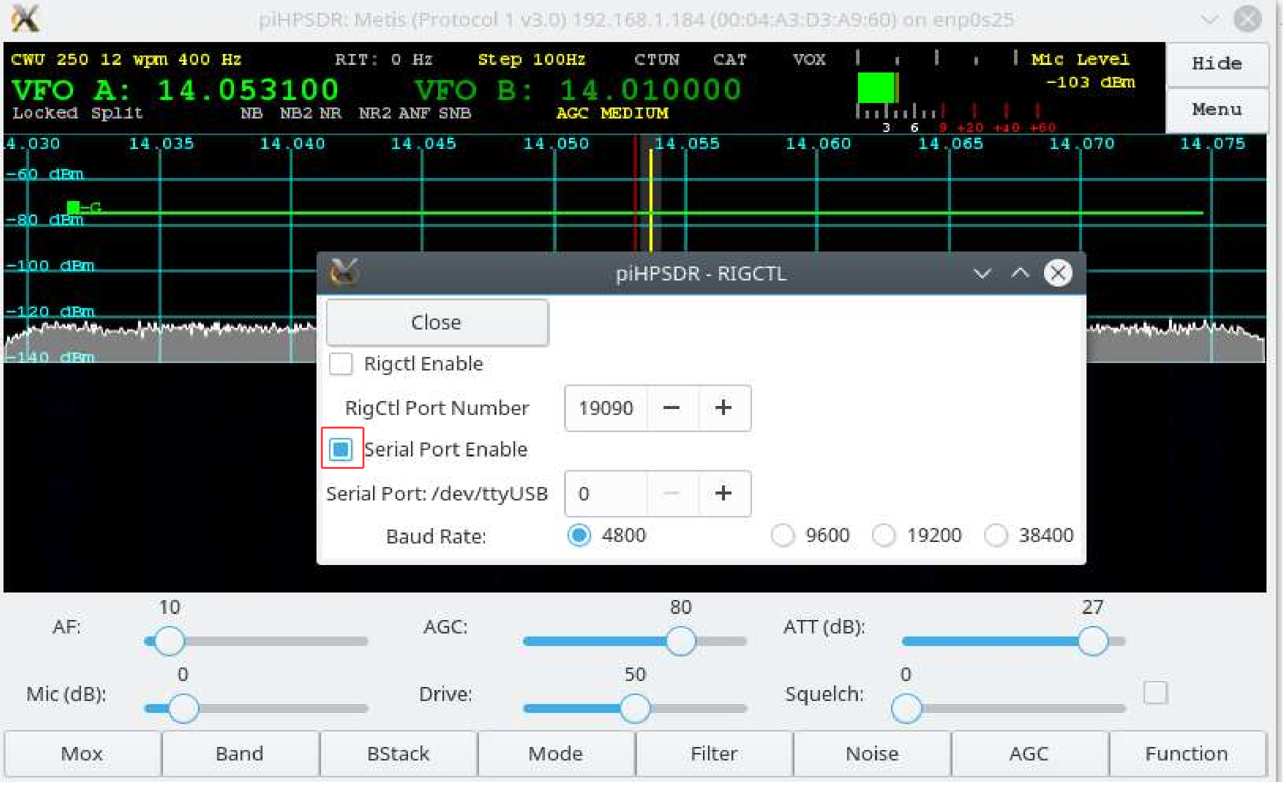
☐ Serial Port Enable

Serial Port: /dev/ttyUSB 0

Baud Rate: ● 4800 ○ 9600 ○ 19200 ○ 38400

Use the +/- buttons to change the USB serial port to match what is available on your system.

Once the Baud rate and /dev/ttyUSBx port are selected – enable the Serial Port by hitting the Serial Port Enable Button.



☐ Rigctl Enable

RigCtl Port Number: 19090

☑ Serial Port Enable

Serial Port: /dev/ttyUSB 0

Baud Rate: ● 4800 ○ 9600 ○ 19200 ○ 38400

If the Serial Port Enable button doesn't stay lit – no /dev/ttyUSBx was found.

KA6S Summary

PiHPSDR provides a TCP/IP channel on ports19090 which can communicate with other applications using an emulation of the TS-2000 command set. The port number CAN be configured on the RIGCTL menu, as well as enabled/disabled.

Some applications require a virtual serial port – use the socat command run to create the virtual serial port.

The very latest version of Hamlib supports PiHPSDR natively. Radio model is 240, and uses TCPIP Address/Port Number instead of the serial port.

Other applications require the hamlib rigctld to be present – start /usr/bin/rigctld with sudo.

To run FLDIGI – use FLRIG as the server for FLDIGI since FLDIGI has issues talking to the virtual serial port.

Conventional connection to PiHPSDR is available through a USB to Serial adapter plugged into the Raspberry PI by enabling the Serial Port Enable button.

16. Supported CAT Commands (TS-2000 Compatible)

AG Command Sets/Reads AF Slider

	1	2	3	4	5	6	7	8	9	10	
Set	A	G	P1	P2	P2	P2	;				Notes:
Read	A	G	;								P1 Ignored
Response	A	G	P1	P2	P2	P2	;				P2 000-255 linearly mapped to 0-100 range

BD Command Moves the Band Down

	1	2	3	4	5	6	7	8	9	10	Notes:
Set	B	D	;								No reply to this command -

Wraps from 1.8Mhz to 50Mhz

BU Command Moves the Band Up

	1	2	3	4	5	6	7	8	9	10	Notes:
Set	B	D	;								No reply to this command -

Wraps from 50MHz to 1.8MHz

CN Command Sets/Reads the CTCSS function

	1	2	3	4	5	6	7	8	9	10	
Set	F	R	P1	P1	;						P1- Values of 01-38
Read	F	R	;								
Response	F	R	P1	P1	;						

	P1	Freq	P1	Freq	P1	Freq	P1	Freq
P1 Values	1	67	11	97.4	21	136.5	31	192.8
	2	71.9	12	100	22	141.3	32	203.5
	3	74.4	13	103.5	23	146.2	33	210.7
	4	77	14	107.2	24	151.4	34	218.1
	5	79.7	15	110.9	25	156.7	35	225.7
	6	82.5	16	114.8	26	162.9	36	233.6
	7	85.4	17	118.8	27	167.9	37	241.8
	8	91.5	18	123	28	173.8	38	250.3
	9	91.5	19	127.3	29	179.9		
	10	94.8	20	131.8	30	186.2		

CT Command Sets/Reads CTCSS enable

	1	2	3	4	5	6	7	8	9	10	
Set	F	R	P1	;							Notes:
Read	F	R	;								0=CTCSS off
Response	F	R	P1	;							1=CTCSS on

FA Command Sets VFO A Frequency

	1	2	3	4	5	6	7	8	9	10	
Set	F	A	P1	P1	P1	P1	P1	P1	P1	P1	Frequency in Hertz (11- digits)
	P1	P1	P1	;							Note that blank digits should be 0
Read	F	A	;								Always reads VFO_A
Response	F	A	P1	P1	P1	P1	P1	P1	P1	P1	
	P1	P1	P1	;							

FB Command
Sets VFO B Frequency

	1	2	3	4	5	6	7	8	9	10
Set	F	B	P1	P1	P1	P1	P1	P1	P1	P1
	P1	P1	P1	;						
Read	F	B	;							
Response	F	B	P1	P1	P1	P1	P1	P1	P1	P1
	P1	P1	P1	;						

Frequency in Hertz (11- digits)
Note that blank digits should be 0
Reads VFO_A when one receiver operation
is selected.

FR Command
Sets/Reads which VFO is active receiver

	1	2	3	4	5	6	7	8	9	10
Set	F	R	P1	;						
Read	F	R	;							
Response	F	R	P1	;						

P1=0 – VFO A 1=VFO B
Only in 2 receiver mode will cause
frequency swapping to between
VFO A and VFO B and if Transmit != Receive
VFO – Split will set.

FS Command
Sets/Reads FINE RIT tuning operation (See RD/RU Commands)

	1	2	3	4	5	6	7	8	9	10
Set	F	R	P1	;						
Read	F	R	;							
Response	F	R	P1	;						

Notes:
0=FINE function off (RIT step=10Hz)
1=FINE function on (RIT step=1Hz)

FT Command
Sets/Reads which VFO is active transmitter

	1	2	3	4	5	6	7	8	9	10
Set	F	T	P1	;						
Read	F	T	;							
Response	F	T	P1	;						

P1=0 – VFO A 1=VFO B
Sets Split if Active Transmitter not the same
as active receiver

FW Command
Sets/Reads receive filter width (Only for CW mode)

	1	2	3	4	5	6	7	8	9	10
Set	F	W	P1	P1	P1	P1	;			
Read	F	W	;							
Response	F	W	P1	P1	P1	P1	;			

P1 – (0000-9999 in Hz)
Frequency are mapped from TS2000 to PiHPSDR
CW 25/50→50 100→100 250→300 400->400
500→500 600→600 750→1000 800→1000

GT Command
Set/Read AGC constant values

	1	2	3	4	5	6	7	8	9	10
Set										
Read	G	T	;							
Response	G	T	P1	P1	P1	;				

TS-2000 legal values 000-020
PiHPSDR = 000 = Off, 005=Fast, 010=Medium
010=Medium, 015=Slow, 020=Long
Note: Hamlib will send values as N*84 -
PiHPSDR can detect commands using
this scale and SET its internal state
accordingly – but can only read back legal
values according to the 000-020 mapping.

ID Command
Read the transceiver ID number

	1	2	3	4	5	6	7	8	9	10
Set										

P1 = 019: TS-2000

IF Command
Sets VFO A Frequency

	1	2	3	4	5	6	7	8	9	10
Set										
Read	I	F	;							
Response	I	F	P1	P1	P1	P1	P1	P1	P1	P1
	P1	P1	P1	P2	P2	P2	P2	P3	P3	P3
	P3	P3	P3	P4	P5	P6	P7	P7	P8	P9
	P10	P11	P12	P13	P14	P14	P15	;		

P1 = Frequency in Hz (11 digits)
P2 = Step in Hertz (5 digits)
P3 = Active Receiver RIT in Hertz (5 digits)
P4 = 0: Rit Off 1: Rit On
P5 = 0: Rit Off 1: Rit On
P8 = 0: Mox=0 1: Mox=1
P9 = Radio Mode (See MD command)
P12 = 0: Split off 1: Split on
All other parameters are 0.

KS Command
Sets and reads keyer speed

	1	2	3	4	5	6	7	8	9	10
Set	K	S	P1	P1	P1	;				
Read	K	S	;							
Response	K	S	P1	P1	P1	;				

P1 – 010 (min) – 060 (max) in WPM

LK Command
Sets/reads the lock function

	1	2	3	4	5	6	7	8	9	10
Set	L	K	P1	P2	;					
Read	L	K	;							
Response	L	K	P1	P2	;					

Notes:
P1 = 0: Unlock 1: Lock
P2 ignored

MD Command
Sets/Reads radio Mode

	1	2	3	4	5	6	7	8	9	10
Set	M	D	P1	;						
Read	M	D	;							
Response	M	D	P1	;						

P1 = 1: LSB 2: USB 3: CWU 4: FMN 5: AM
6: DIGL 7: CWL 9: DIGU

MG Command
Sets/Reads Mic Gain Slider

	1	2	3	4	5	6	7	8	9	10
Set	M	G	P1	P1	P1	;				
Read	M	G	;							
Response	M	G	P1	P1	P1	;				

P1 – 000 (min) – 100 (max)

NB Command
Sets/Reads the Noise Blanker function status

	1	2	3	4	5	6	7	8	9	10
Set	N	B	P1	;						
Read	N	B	;							
Response	N	B	P1	;						

P1 = 0: Off 1: On

NR Command
Sets/Reads the Noise Blanker function status

	1	2	3	4	5	6	7	8	9	10
Set	N	R	P1	;						
Read	N	R	;							
Response	N	R	P1	;						

P1 = 0: Off 1: NR On 2: NR2 On

NT Command
Sets ANF bit (Autonotch in TS2000)

	1	2	3	4	5	6	7	8	9	10
Set	N	T	P1	;						
Read	N	T	;							
Response	N	T	P1	;						

P1= 0: ANF Off 1: ANF On

PC Command
Sets/Reads Drive Slider

	1	2	3	4	5	6	7	8	9	10
Set	P	C	P1	P1	P1	;				
Read	P	C	;							
Response	P	C	P1	P1	P1	;				

P1 – 005 (min) – 100 (max)

PL Command
Sets/Reads Speech Processor Level

	1	2	3	4	5	6	7	8	9	10
Set	P	L	P1	P1	P1	P2	P2	P2	;	
Read	P	L	;							
Response	P	L	P1	P1	P1	P2	P2	P2	;	

P1 – 0 -100
P2 – Not used

PR Command

Sets/Reads Speech Processor Enable

	1	2	3	4	5	6	7	8	9	10
Set	P	R	P1	;						
Read	P	R	;							
Response	P	R	P1	;						

P1 – 0: Off, 1: On

RD Command

Move RIT off frequency Down

	1	2	3	4	5	6	7	8	9	10
Set	R	D	P1	P1	P1	P1	P1	;		
Read	R	D	;							
Response	R	D	P2	;						

P1 – RIT Minus Offset (0000-9999 in Hz)
When there is NO P1 – moves the RIT frequency
Moves offset down defined by FS setting (FINE)
P2- reports status of RIT function

RG Command

Sets/Reads AGC slider

	1	2	3	4	5	6	7	8	9	10
Set	R	G	P1	P1	P1	;				
Read	R	G	;							
Response	R	G	P1	P1	P1	;				

P1 = 000 (min) to 255 (max) linearly scaled to
-20 to 140 range of slider

RT Command

Sets/Read the RIT function status

	1	2	3	4	5	6	7	8	9	10
Set	R	T	P1	;						
Read	R	T	;							
Response	R	T	P1	;						

P1 – 0: Off 1: On
Returns state of active receiver RIT

RU Command

Move RIT off frequency Up

	1	2	3	4	5	6	7	8	9	10
Set	R	U	P1	P1	P1	P1	P1	;		
Read	R	U	;							
Response	R	U	P2	;						

P1 – RIT Plus Offset (0000-9999 in Hz)
When there is NO P1 – moves the RIT frequency
UP by by amount defined by FS setting (FINE)
P2- reports status of RIT function

RX Command

Set Mox to 0 (turn off transmitter)

	1	2	3	4	5	6	7	8	9	10
Set	R	X	;							
Read										
Response										

SD Command

Set /Read CW Breakin Delay

	1	2	3	4	5	6	7	8	9	10
Set	S	D	P1	P1	P1	P1	P1	;		
Read	S	D	;							
Response	S	D	P1	P1	P1	P1	P1	;		

P1 = 0000-1000 ms Breakin delay
0000= Set Full Breakin
If PiHPSDR has values above 1000 internally
a value of 1000 is reported

SM Command

Reads the S Meter

	1	2	3	4	5	6	7	8	9	10
Set										
Read	S	M	P1	;						
Response	S	M	P1	P2	P2	P2	P2	;		

P1 – 0: Main, 1: Sub
P2 = 0000 (min) to 0030 (max) main xcvr
0000 (min) to 0015 (max) sub xcvr

SQ Command

Set/Read Squech level

	1	2	3	4	5	6	7	8	9	10
Set	S	Q	P1	P2	P2	P2	P2	;		
Read	S	Q	;							
Response	S	Q	P1	P2	P2	P2	P2	;		

P1 = 0 – Not Used
P2- 0 – 255

ST Command

Sets/reads the frequency step

	1	2	3	4	5	6	7	8	9	10
Set	S	T	P1	P1	;					
Read	S	T	;							
Response	S	T	P1	P1	;					

Note that internal value is categorized into These slots for the read operation, Set uses the assigned value.

P1 – SSB 00-1 KHz 01: 2.5KHz 02 5KHz
03: 10KHz
AM/FM: 00: 5KHz 01: 6.25KHz
02: 10KHz 03: 12.5KHz
04: 15KHz 05: 20KHz 06: 25KHz 07: 30KHz
08: 50KHz 09: 100KHz

TX Command

Set the transmitter MOX

	1	2	3	4	5	6	7	8	9	10
Set	T	X	;							
Read										
Response										

VD Command

Set/Read Vox Delay

	1	2	3	4	5	6	7	8	9	10
Set	V	D	P1	P1	P1	P1	;			
Read	V	D	;							
Response	V	D	P1	P1	P1	P1	;			

P1 – 0000 (min) to 1000 (max)
Values are limited to these boundaries

VG Command

Set/Read Vox Threshold (Vox GAIN on TS2000)

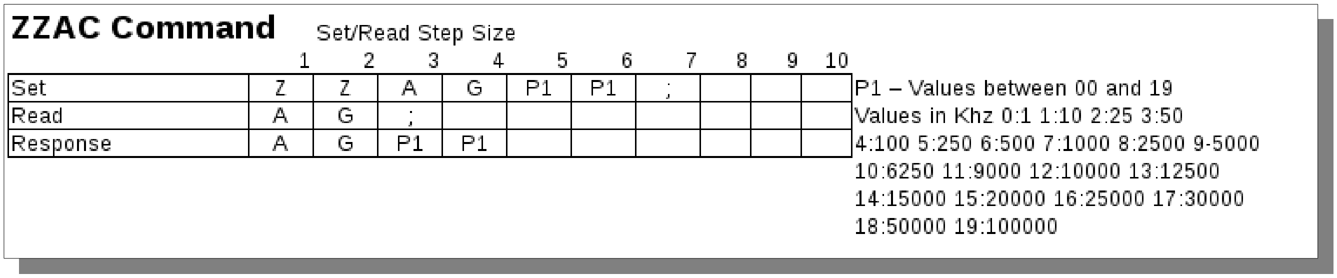
	1	2	3	4	5	6	7	8	9	10
Set	V	D	P1	P1	P1	;				
Read	V	D	;							
Response	V	D	P1	P1	P1	;				

Set Vox Threhold
P1 = (ooo) min to (oo9) max

17. Supported CAT Commands (ZZ Extension commands)

This is an extended command set which gives the CAT port access to most of the controls within PiHPSDR.

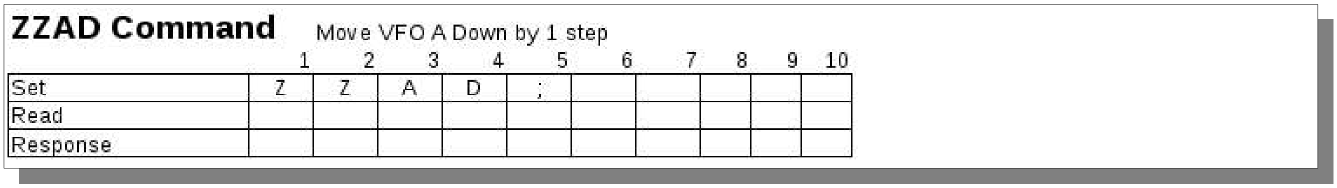
Note: These commands are not yet supported by Hamlib



ZZAC Command Set/Read Step Size

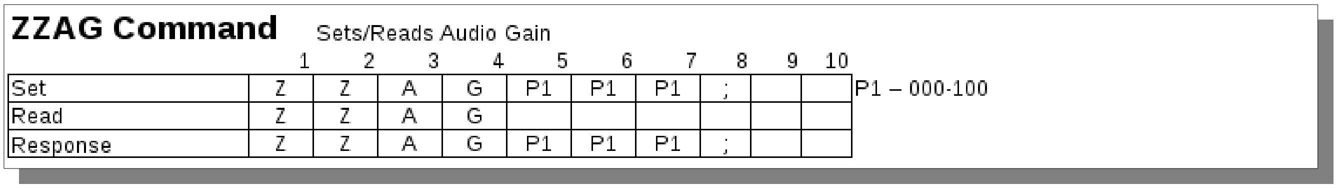
	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	G	P1	P1	;			
Read	A	G	;							
Response	A	G	P1	P1						

P1 – Values between 00 and 19
Values in Khz 0:1 1:10 2:25 3:50
4:100 5:250 6:500 7:1000 8:2500 9-5000
10:6250 11:9000 12:10000 13:12500
14:15000 15:20000 16:25000 17:30000
18:50000 19:100000



ZZAD Command Move VFO A Down by 1 step

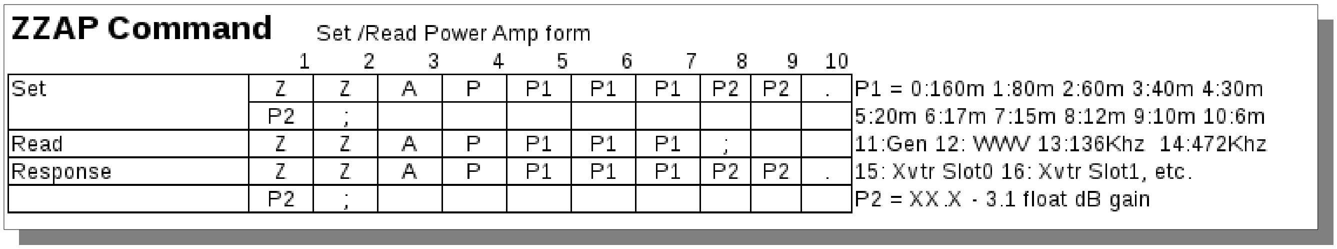
	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	D	;					
Read										
Response										



ZZAG Command Sets/Reads Audio Gain

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	G	P1	P1	P1	;		
Read	Z	Z	A	G						
Response	Z	Z	A	G	P1	P1	P1	;		

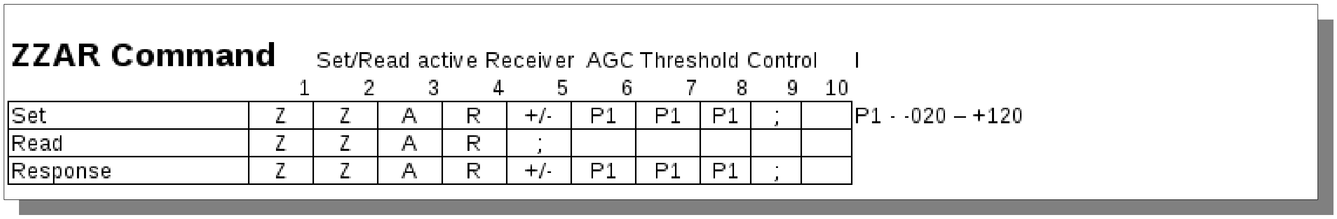
P1 – 000-100



ZZAP Command Set /Read Power Amp form

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	P	P1	P1	P1	P2	P2	.
	P2	;								
Read	Z	Z	A	P	P1	P1	P1	;		
Response	Z	Z	A	P	P1	P1	P1	P2	P2	.
	P2	;								

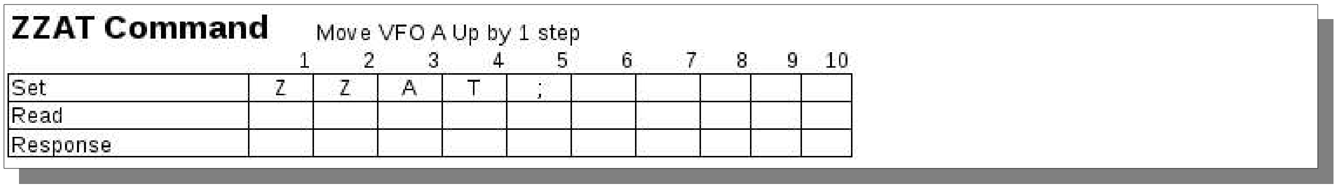
P1 = 0:160m 1:80m 2:60m 3:40m 4:30m
5:20m 6:17m 7:15m 8:12m 9:10m 10:6m
11:Gen 12: WWV 13:136Khz 14:472Khz
15: Xvtr Slot0 16: Xvtr Slot1, etc.
P2 = XX.X - 3.1 float dB gain



ZZAR Command Set/Read active Receiver AGC Threshold Control I

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	R	+/-	P1	P1	P1	;	
Read	Z	Z	A	R	;					
Response	Z	Z	A	R	+/-	P1	P1	P1	;	

P1 - -020 – +120



ZZAT Command Move VFO A Up by 1 step

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	T	;					
Read										
Response										

ZZAU Command

Start/Stop Audio UDP Stream

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	A	U	P1	P2	P2	P2	P2	P2
	P3	P4	;							
Read										
Response										

P1 = 0:RX0 1:RX1
P2 = Port to send UDP stream too
P3 = 0:aLaw 1:uLaw
P4 = 0:Stop 1:Start

ZZBD Command

Moves the Active Receiver Band Selector Down

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	B	D	;		;			
Read										
Response										

ZZBS Command

Read Active Receiver Band Switch

	1	2	3	4	5	6	7	8	9	10
Set										
Read	Z	Z	B	S	;					
Response	Z	Z	B	S	P1	P1	P1			

P1 -Band – position equal to location in the Band Menu with 0=left most entry. This encoding is used to allow for tranverters, when when define expand the menu:

P1 = 0:160m 1:80m 2:60m 3:40m 4:30m 5:20m 6:17m 7:15m 8:12m 9:10m 10:6m 11:Gen 12: WWV 13:136Khz 14:472Khz 15: Xvtr Slot0 16: Xvtr Slot1, etc.

ZZBU Command

Moves the Active Receiver Bad Selector Up

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	B	U	;		;			
Read										
Response										

ZZCB Command

Set/Read CW Break-In Check Box status

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	C	B	P1	;				
Read	Z	Z	C	B	;					
Response	Z	Z	C	B	P1	;				

P1 – 0:Off 1:On

ZZCD Command

Set /Read CW Break-in Delay

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	C	D	P1	P1	P1	P1	P1	;
Read	Z	Z	C	D	;					
Response	Z	Z	C	D	P1	P1	P1	P1	;	

P1 = 0000-1000 ms Breakin delay
0000= Set Full Breakin
If PiHPSDR has values above 1000
Internally a value of 1000 is reported

ZZCL Command

Set/Read CW Pitch

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	C	L	P1	P1	P1	P1	;	
Read	Z	Z	C	L	;					
Response	Z	Z	C	L	P1	P1	P1	P1	;	

P1- 0200-1200 Hz

ZZDA Command

Sets/Read Waterfall Automatic Enable

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	D	A	P1	;				
Read	Z	Z	D	A	;					
Response	Z	Z	D	A	P1	;				

P1 = 0:Disable 1:Enable

ZZDN Command

Set/Read Waterfall Lower Limit

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	D	N	P1	P1	P1	P1	;	
Read	Z	Z	D	N	;					
Response	Z	Z	D	N	P1	P1	P1	P1	;	

P1- -200-0200 legal values

ZZDO Command

Set/Read Waterfall Hi Limit

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	D	O	P1	P1	P1	P1	;	
Read	Z	Z	D	O	;					
Response	Z	Z	D	O	P1	P1	P1	P1	;	

P1- -200-0200 legal values

ZZDU Command

Read Active Receiver status

	1	2	3	4	5	6	7	8	9	10
Set										
Read	Z	Z	D	U	;					
Response	Z	Z	D	O	P1	P2	P3	P4	P5	P6
	P7	P8	P9	P10	P10	P10	P10	P10	P10	P11
	P12	P12	P12	+/-	P13	P13	P13	P14	P14	P14
	P15	+/-	P16	P16	P16	P16	P16	P16	P17	P18
	P18	P18	P18	P18	P18	P18	P18	P18	P18	P18
	P19	P19	P19	P19	P19	P19	P19	P19	P19	P19
	P19	+/-	P20	P20	P20	P21	P22	;		

P1 – split
P2 – tune
P3 – mox
P4- # of receivers (1 or 2)
P5 – act rcvr # (0 or 1)
P6 – act rcvr alex antenna
P7 – Rit On (o=No, 1=Yes0
P8 – act rcvr agc
P9 – act rcvr mode
P10 – step
P11 – act rcvr band (0-14)
P12 – drive
P13 – act rcvr agc gain
P14 – act rcvr volume
P15 – rit enabled
P16 – rit offset
P17 – Ctune
P18 – VFO Freq
P19 – CTUNE Freq
P20 – Smeter
P21 – act rcvr nr2
P22 – act rcvr snb

ZZEA Command

Set/Read RX Equalizer Values

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	E	A	P1	+/-	P2	P2	P2	+/-
	P3	P3	P3	+/-	P4	P4	P4	;		
Read	Z	Z	E	A	;					
Response	Z	Z	E	A	P1	+/-	P2	P2	P2	+/-
	P3	P3	P3	+/-	P4	P4	P4	;		

P1- 0:Disable Eq, 1:Enable Eq
P2 - -012 – 015db
P3 - -012 – 015db
P4 - -012 – 015db

ZZEB Command

Set/Read TX Equalizer Values

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	E	A	P1	+/-	P2	P2	P2	+/-	P1- 0=Disable Eq, 1=Enable Eq
	P3	P3	P3	+/-	P4	P4	P4	;			P2 - -012 – 015db
Read	Z	Z	E	A	;						P3 - -012 – 015db
Response	Z	Z	E	A	P1	+/-	P2	P2	P2	+/-	P4 - -012 – 015db
	P3	P3	P3	+/-	P4	P4	P4	;			

ZZFA Command

Set/Read VFO A

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	F	A	P1	P1	P1	P1	P1	P1	P1 – Freq in Hz
	P1	P1	P1	P1	P1	;					
Read	Z	Z	F	A	;						
Response	Z	Z	F	A	P1	P1	P1	P1	P1	P1	
	P1	P1	P1	P1	P1	;					

ZZFB Command

Set/Read VFO B

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	F	B	P1	P1	P1	P1	P1	P1	P1 – Freq in Hz
	P1	P1	P1	P1	P1	;					
Read	Z	Z	F	B	;						
Response	Z	Z	F	B	P1	P1	P1	P1	P1	P1	
	P1	P1	P1	P1	P1	;					

ZZFH Command

Set/Read Current DSP Filter High

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	F	H	P1	P1	P1	P1	P1	;	P1 - -9999 to 09999
Read	Z	Z	F	H	;						Note – Set only works if Variable 1 or 2
Response	Z	Z	F	H	P1	P1	P1	P1	P1	;	Selected.

ZZFI Command

Set/Read DSP Filter Selected

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	F	I	P1	P1	;				P1 – 0-9 Selected DSP Setting
Read	Z	Z	F	I	;						Filter value depends on Mode
Response	Z	Z	F	I	P1	P1	;			;	Values 11 and 12 select Variable 1 & 2

ZZFL Command

Set/Read Current DSP Filter Low

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	F	L	P1	P1	P1	P1	P1	;	P1 - -9999 to 09999
Read	Z	Z	F	L	;						Note – Set only works if Variable 1 or 2
Response	Z	Z	F	L	P1	P1	P1	P1	P1	;	Selected.

ZZFT Command

Set/Read Active Transmitter

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	F	T	P1	;					P1 – 0:VFOA 1:VFOB
Read	Z	Z	F	T	;						
Response	Z	Z	F	T	P1	;					

ZZGT Command

Set/Read AGC Constant

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	G	T	P1	;					P1 – 0:Off 1:Long 2:Slow 3:Medium 4:Fast
Read	Z	Z	G	T	;						
Response	Z	Z	G	T	P1	;					

ZZID Command

Read the transceiver ID number

	1	2	3	4	5	6	7	8	9	10	
Set											P1 = 240
Read	Z	Z	I	D	;						
Response	Z	Z	I	D	P1	P1	P1	;			

ZZKS Command

Sets and reads keyer speed

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	K	S	P1	P1	P1	;			P1 – 010 (min) – 060 (max) in WPM
Read	Z	Z	K	S	;						
Response	Z	Z	K	S	P1	P1	P1	;			

ZZLK Command

Sets/reads the lock function

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	L	K	P1	P2	;				Notes:
Read	Z	Z	L	K	;						P1 = 0: Unlock 1: Lock
Response	Z	Z	L	K	P1	P2	;				P2 ignored

ZZMD Command

Sets/Reads radio Mode

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	M	D	P1	P1	;				P1 = 1: LSB 2: USB 3: DSB 4: CWL 5:CWU 6:FMN 7:AM 8:DIGU 9:SPEC 10:DIGL 11:SAM: 12:DRM
Read	Z	Z	M	D	;						
Response	Z	Z	M	D	P1	P1	;				

ZZMG Command

Sets/Reads Mic Gain Slider

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	M	G	+/-	P1	P1	;			P1 = -10(min) to 50(max)
Read	Z	Z	M	G	;						Note +/- sign location mandatory
Response	Z	Z	M	G	+/-	P1	P1	;			

ZZMT Command

Set/Read TX meter Mode

	1	2	3	4	5	6	7	8	9	10	
Set	Z	Z	M	T	P1						P1 = 0: ALC Peak 1: ALC Average 2: ALC Gain
Read	Z	Z	M	T	;						
Response	Z	Z	M	T	P1	;					

ZZMU Command

Reads number of Receivers enabled

	1	2	3	4	5	6	7	8	9	10
Set										
Read	Z	Z	M	T	;					
Response	Z	Z	M	T	P1	;				

P1 = 0:1 Receivers 1:2 Receivers

ZZNB Command

Sets/Reads the Noise Blanker function status

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	N	B	P1	;				
Read	Z	Z	N	B	;					
Response	Z	Z	N	B	P1	;				

P1 = 0: Off 1: On

ZZNE Command

Sets/Reads the NPS AE Filter – DSP Menu

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	N	E	P1	;				
Read	Z	Z	N	E	;					
Response	Z	Z	N	E	P1	;				

P1 = 0:Off 1:On

ZZNG Command

Sets/Reads the Pre AGC/ Post AGC – DSP Menu

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	N	G	P1	;				
Read	Z	Z	N	G	;					
Response	Z	Z	N	G	P1	;				

P1 = 0:Pre AGC 1:Post AGC

ZZNM Command

Sets/Reads NR2 Gain Method – DSP Menu

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	N	M	P1	;				
Read	Z	Z	N	M	;					
Response	Z	Z	N	M	P1	;				

P1 = 0:Linear 1:Log

ZZNP Command

Sets/Reads NPS Method – DSP Menu

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	N	P	P1	;				
Read	Z	Z	N	P	;					
Response	Z	Z	N	P	P1	;				

P1 = 0:OSMS 1:MMSE

ZZNL Command

Sets/Reads AGC Hang Threshold – DSP Menu Slider

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	M	G	P1	P1	P1	;		
Read	Z	Z	M	G	;					
Response	Z	Z	M	G	P1	P1	P1	;		

P1 = 0(Min) – 100(Max)

ZZNR Command

Sets/Reads the Noise Blanker function status

	1	2	3	4	5	6	7	8	9	j
Set	Z	Z	N	R	P1	;				
Read	Z	Z	N	R	;					
Response	Z	Z	N	R	P1	;				

P1 = 0: Off 1: NR On 2: NR2 On

ZZNT Command

Sets ANF bit

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	N	T	P1	;				
Read	Z	Z	N	T	;					
Response	Z	Z	N	T	P1	;				

P1= 0: ANF Off 1: ANF On

ZZOA Command

Set/Read RX Antenna by band – ANT Menu

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	O	A	P1	P1	P1	P2	;	
Read	Z	Z	O	A	P1	P1	P1	;		
Response	Z	Z	O	A	P1	P1	P1	P2	;	

P1 = 0:160m 1:80m 2:60m 3:40m 4:30m
5:20m 6:17m 7:15m 8:12m 9:10m 10:6m
11:Gen 12: WWV 13:136Khz 14:472Khz
15: Xvtr Slot0 16: Xvtr Slot1, etc.
P2 = 0:1 1:2 2:3 3:EXT1 4:EXT2 5:XVTR

ZZOB Command

Set/Read TX Antenna by band – ANT Menu

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	O	B	P1	P1	P1	P2	;	
Read	Z	Z	O	B	;					
Response	Z	Z	O	B	P1	P1	P1	P2	;	

P1 = 0:160m 1:80m 2:60m 3:40m 4:30m
5:20m 6:17m 7:15m 8:12m 9:10m 10:6m
11:Gen 12: WWV 13:136Khz 14:472Khz
15: Xvtr Slot0 16: Xvtr Slot1, etc.
P2 = 0:1 1:2 2:3

ZZPA Command

Set/Read PreAmp F(x) status

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	P	A	P1					
Read	Z	Z	P	A	;					
Response	Z	Z	P	A	P1	;				

P1 = 0: Off 1: On

ZZPC Command

Sets/Reads Drive Slider

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	P	C	P1	P1	P1	;		
Read	Z	Z	P	C	;					
Response	Z	Z	P	C	P1	P1	P1	;		

P1 – 005 (min) – 100 (max)

ZZPZ Command

Set/Read Radio Sample rate

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	P	Z	P1	P1	P1	P1	P1	P1
	;					;				
Read	Z	Z	P	Z	;					
Response	Z	Z	P	Z	P1	P1	P1	P1	P1	P1
	;									

P1= 048000, 096000, 0192000, 384000

ZZRC Command

Clear the RIT Offset frequency

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	R	C	;					
Read										
Response										

ZZRD Command

Decrement Active Receiver RIT freq by rit_offset

	1	2	3	4	5	6	7	8	9	10
Set										
Read	Z	Z	R	D	;					
Response	Z	Z	R	D	P1	;				

P1 = Rit Enabled

ZZRG Command

Sets/Reads AGC slider

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	R	G	P1	P1	P1			
Read	Z	Z	R	G	;					
Response	Z	Z	R	G	P1	P1	P1			

P1 = -20 to 120 range of slider

ZZRT Command

Sets/Read the RIT function status

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	R	T	P1	;				
Read	Z	Z	R	T	;					
Response	Z	Z	R	T	P1	;				

P1 – 0: Off 1: On

Returns state of active receiver RIT

ZZRU Command

Increment Active Receiver RIT freq by rit_offset

	1	2	3	4	5	6	7	8	9	10
Set										
Read	Z	Z	R	U	;					
Response	Z	Z	R	U	P2					

P1 = Rit Enabled

ZZRX Command

Set Mox to 0 (turn off transmitter)

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	R	X	;					
Read										
Response										

ZZSM Command

Reads the S Meter

	1	2	3	4	5	6	7	8	9	10
Set										
Read	Z	Z	S	M	P1	;				
Response	Z	Z	S	M	P1	P2	P2	P2	P2	;

P1 = 0: Main, 1: Sub

P2 = 0000 (min) to 0030 (max) main xcvr
0000 (min) to 0015 (max) sub xcvr

ZZST Command

Sets/reads the frequency step

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	S	T	P1	P1	P1	P1	P1	P1
	;									
Read	Z	Z	S	T	;					
Response	Z	Z	S	T	P1	P1	P1	P1	P1	P1
	;									

P1 = 1, 10, 50, 100, 250, 500, 1000
2500, 5000, 6250, 9000, 10000
12500, 15000, 20000, 25000, 30000
50000, 100000

ZZTX Command

Set the transmitter MOX

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	T	X	;					
Read										
Response										

ZZUA Command

Set /Read XVTR Form

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	U	A	P1	P2	P3	P3	P3	P3
	P3	P3	P3	P3	P3	P3	P3	P3	P3	P3
	P3	P4	P4	P4	P4	P4	P4	P4	P4	P4
	P4	P4	P5	P5	P5	P5	P5	P5	P5	P5
	P5	P5	P5	P6	P6	P6	P6	P6	P6	P6
	P6	P6	P6	P6	;					
Read	Z	Z	U	A	;					
Response	Z	Z	U	A	P1	P2	P3	P3	P3	P3
	P3	P3	P3	P3	P3	P3	P3	P3	P3	P3
	P3	P4	P4	P4	P4	P4	P4	P4	P4	P4
	P4	P4	P5	P5	P5	P5	P5	P5	P5	P5
	P5	P5	P5	P6	P6	P6	P6	P6	P6	P6
	P6	P6	P6	P6	;					

P1 = Entry number 0-7
P2 = 0:EnablePA 1:DisablePA
P3 = Title – 15 characters
P4 = Min Frequency (Hz)
P5 = Max Frequency (Hz)
P6 = LO Frequency (Hz)

ZZVD Command

Set/Read Vox Hang Time

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	V	D	P1	P1	P1	P1	;	
Read	Z	Z	V	D	;					
Response	Z	Z	V	D	P1	P1	P1	P1	;	

P1 – 0000 (min) to 1000 (max)
Values are limited to these boundaries

ZZVG Command

Set/Read Vox Threshold (Vox GAIN on TS2000)

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	V	D	P1	P1	P1	P1	;	
Read	Z	Z	V	D	;					
Response	Z	Z	V	D	P1	P1	P1	P1	;	

Set Vox Threhold
P1 = 0 (Min) to 1000 (Max)

ZZVX Command

Sets/Read Vox Enable

	1	2	3	4	5	6	7	8	9	10
Set	Z	Z	V	X	P1	;				
Read	Z	Z	V	X	;					
Response	Z	Z	V	X	P1	;				

P1 = 0:Disable 1:Enable

18. 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

19. 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

NOTES

Alphabetical Index